NASA CR-159,279

NASA Contractor Report 159279

DOCUMENTATION OF PROGRAM COORDC TO GENERATE COORDINATE SYSTEM FOR 3-D CORNERS WITH OR WITHOUT FILLET USING BODY-FITTED CURVILINEAR COORDINATES-Part II

Dilip Kumar

SYSTEMS AND APPLIED SCIENCES CORPORATION
Hampton, Virginia 23666

~~FOR EARLY DOMESTIC DISSEMINATION~~

NASA Contract NAS1-15604
June 1980

NASA
National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

TABLE OF CONTENTS

ABSTRACT

The computer program COORDC generates a body-fitted curvilinear coordinate system for corner geometry with or without corner fillets. It is assumed that at any given ξ, x remains constant; consequently the only variation is in y and z. It is also assumed that for all ξ's in the physical plane the coordinate system in y-z plane is similar. This enables solution of coordinate system for one particular $\xi = 1$ (x for $\xi = 1$ is arbitrarily chosen to be 0.0) and the solution for all other ξ plane can be easily specified once the coordinates in the physical plane on the line $1 \leq \xi \leq$ IMAX, $\eta = 1$, $\zeta = 1$ are specified.

List of Symbols

IMAX	Maximum number of points in ξ - direction
JMAX	Maximum number of points in η - direction
KMAX	Maximum number of points in ζ - direction
J	Jacobian
P,Q,R	Inhomogeneous terms used for coordinate attraction
S,O,R	Successive-Over-Relaxation iteration
X,Y,Z	Physical Coordinates
α,β,γ	Metric Coefficients
ξ,η,ζ	Transformed Coordinates
$\Gamma_1,\Gamma_2,\Gamma_3,\Gamma_4$	Boundary contours in the physical plane
$\Gamma_1^*,\Gamma_2^*,\Gamma_3^*,\Gamma_4^*$	Boundary contours in the transformed plane
ω	Acceleration parameter for Gauss-Seidel iteration

Subscripts

ξ,η,ζ,x,y,z	Denotes first partial differentiation
$\xi\xi,\eta\eta$,xx,yy etc.	Denotes second partial differentiation
$\xi\eta$,xy,xz etc.	Denotes cross partial differentiation

Superscripts

(s)	Denotes the iteration number

I. Numerical Generation of Body-Fitted Curvilinear Coordinate Systems

The method for numerically generating boundary-fitted curvilinear coordinate systems is presented as applicable in a specialized 3-D corner with or without a fillet. The coordinate system is generated such that for a specified ξ, x remains constant and for all ξ's the solution in y-z plane stays similar. This is particularly beneficial because for $\xi = 1$ plane (x is chosen to be 0.0 for $\xi = 1$) y-z solution can be obtained numerically by the 2-D formulation instead of 3-D formulation of the body-fitted curvilinear coordinate system. The coordinates for all other ξ planes can be easily obtained once the physical plane coordinates on the line $\xi = 1$, IMAX, $\eta = 1$, $\zeta = 1$ are specified. In the present problem it is assumed that for a given ξ plane, right boundary and upper boundary have z and y constant respectively. In addition, on the left wall and the bottom wall, z and y respectively equal zero some distance from the corner when fillet is present; in case of no fillet z and y on left and bottom wall respectively is equal to zero. The clustering of η and ζ lines is specified by the point distribution on $\eta = 1$ and $\zeta = 1$ line.

Section A provides a brief discussion of the mathematical formulation and section B describes the numerical technique used to generate the coordinate system. Section C explains how the required initial guess is specified. Section D deals with handling of the x-coordinate and the metric coefficients.

A. Mathematics of Transformation

Consider the transformation of a simply connected region R, into a rectangular region as shown in Figure 1. We require that Γ_1, Γ_2, Γ_3, Γ_4 map into Γ_1^*, Γ_2^*, Γ_3^*, Γ_4^* respectively. For identification purposes, region D is referred to as the physical plane, and D^* as the transformed plane. In the physical plane it is assumed that x is constant for a specified ξ, z is constant on Γ_3 and y is constant on Γ_4. If for $\xi = 1$, $x = 0.0$ is assumed, then the transformation from x,y,z to ξ,η,ζ is defined as follows,

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \xi(x) \\ \eta(y,z) \\ \zeta(y,z) \end{bmatrix} \tag{1}$$

The above equation is solved only for η and ζ since ξ is explicitly known. Assuming that an inverse transformation exists, then

$$\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} y(\eta,\zeta) \\ z(\eta,\zeta) \end{bmatrix} \tag{2}$$

where the jacobian J is as follows:

$$J = y_\eta z_\zeta - y_\zeta z_\eta \tag{3}$$

The generating elliptic system is chosen to be the inhomogeneous Laplace equation,

$$\eta_{yy} + \eta_{zz} = Q(\eta,\zeta) \tag{4a}$$

$$\zeta_{yy} + \zeta_{zz} = R(\eta,\zeta) \tag{4b}$$

with the boundary conditions as follows:

on Γ_1
$$\begin{bmatrix} \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \eta_1(y,z) \\ \zeta_1 \end{bmatrix} \tag{5a}$$

on Γ_2
$$\begin{bmatrix} \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \eta_2 \\ \zeta_2(y,z) \end{bmatrix} \tag{5b}$$

on Γ_3
$$\begin{bmatrix} \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \eta_3(y,z) \\ \zeta_3 \end{bmatrix} \tag{5c}$$

and on Γ_4
$$\begin{bmatrix} \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \eta_4 \\ \zeta_4(y,z) \end{bmatrix} \tag{5d}$$

where ζ_1, η_2, ζ_3, η_4 are specified constants and η_1, ζ_2, η_3, ζ_4 are specified functions.

The inhomogeneous terms Q and R are selected to control the spacing of η = constant and ζ = constant lines in the physical plane; several forms for the inhomogeneous terms can be used. In the present case a special form of Q and R terms was used based on the point distribution on $\eta = \eta_2$ and $\zeta = \zeta_1$ line. (For convenience $\eta_2 = \zeta_1 = 1$ and η_4 = JMAX and ζ_4 = KMAX is chosen thus ensuring a field size of JMAX x KMAX for a given ξ plane).

In order that the transformed plane contain a uniform grid of spacing equal to unity, the dependent and independent variables

must be interchanged in equations (4a) and (4b). The resulting equations after transformation are as follows:

$$\alpha y_{\eta\eta} - 2\beta\, y_{\eta\zeta} + \gamma\, y_{\zeta\zeta} = -J^2 \left[y_\eta\, Q(\eta,\zeta) + y_\zeta\, R(\eta,\zeta) \right] \qquad (6a)$$

$$\alpha z_{\eta\eta} - 2\beta\, z_{\eta\zeta} + \gamma\, z_{\zeta\zeta} = -J^2 \left[z_\eta\, Q(\eta,\zeta) + z_\zeta\, R(\eta,\zeta) \right] \qquad (6b)$$

where,

$$\alpha = y_\zeta^{\,2} + z_\zeta^{\,2}$$

$$\beta = y_\eta\, y_\zeta + z_\eta\, z_\zeta$$

$$\gamma = y_\eta^{\,2} + z_\eta^{\,2}$$

$$J = y_\eta\, z_\zeta - y_\zeta\, z_\eta$$

The boundary conditions transform as follows:

on Γ_1^* $\begin{bmatrix} y \\ z \end{bmatrix} \quad \begin{bmatrix} a_1(\eta,\zeta_1) \\ a_2(\eta,\zeta_1) \end{bmatrix}$ (7a)

on Γ_2^* $\begin{bmatrix} y \\ z \end{bmatrix} \quad \begin{bmatrix} b_1(\eta_2,\zeta) \\ b_2(\eta_2,\zeta) \end{bmatrix}$ (7b)

on Γ_3^* $\begin{bmatrix} y \\ z \end{bmatrix} \quad \begin{bmatrix} c_1(\eta,\zeta_3) \\ c_2(\eta,\zeta_3) \end{bmatrix}$ (7c)

and on Γ_4^* $\begin{bmatrix} y \\ z \end{bmatrix} \quad \begin{bmatrix} d_1(\eta_4,\zeta) \\ d_2(\eta_4,\zeta) \end{bmatrix}$ (7d)

The functions a_1, a_2, b_1, b_2, c_1, c_2, d_1, d_2 are specified

by known contours Γ_1, Γ_2, Γ_3, Γ_4. If the coordinate point locations on these contours is fixed then dirchlet type boundary conditions can be used. On the other hand if the coordinate lines on these contours is to be made parallel or normal to some line then Neumann type conditions can be used. On Γ_1^* and Γ_2^* dirchlet type conditions were used. On Γ_3^* and Γ_4^* Neumann type conditions were used. Thus the following expressions are obtained on Γ_3^* and Γ_4^*:

on Γ_3^*

$$y(J,KMAX) = y(J,KMAX-1) \tag{8a}$$

$$z(J,KMAX) = \text{constant } z_1 \tag{8b}$$

$$1 \leq J \leq JMAX$$

and on Γ_4^*

$$z(JMAX,K) = z(JMAX-1,K) \tag{8c}$$

$$y(JMAX,K) = \text{constant } y_1 \tag{8d}$$

$$1 \leq K \leq KMAX$$

The last aspect to be discussed is how Q and R are computed. Q is computed based on the y variation of points on Γ_1 and R is computed based on the z variation of points on Γ_2. The following discussion will make this point clearer.

If one assumes on Γ_1 that coordinate points are specified such that the y derivatives with respect to ζ are zero then from (6a),

$$Q(\eta,\zeta) = -\frac{y_{\eta\eta}}{y_\eta^3} \tag{9a}$$

Also, Q is not a function of ζ. Thus, only one one-dimensional array of size JMAX is sufficient to store Q. Similarly for R we obtain,

$$R(\eta,\zeta) = \frac{z_{\zeta\zeta}}{z_\zeta^3} \tag{9b}$$

Again R is not a function of η. A one-dimensional array of size KMAX is sufficient to store R.

B. Numerical Solution Technique

Based on the above discussion, the following equations must be numerically solved:

$$\alpha\, y_{\eta\eta} - 2\beta\, y_{\eta\zeta} + \gamma\, y_{\zeta\zeta} = - J^2 \left[y_\eta\, Q(\eta) + y_\zeta\, R(\zeta) \right] \tag{10a}$$

$$\alpha\, z_{\eta\eta} - 2\beta\, z_{\eta\zeta} + \gamma\, z_{\zeta\zeta} = - J^2 \left[z_\eta\, Q(\eta) + z_\zeta\, R(\zeta) \right] \tag{10b}$$

where

$$Q(\eta) = - \frac{y_{\eta\eta}}{y_\eta^3} \quad \text{on } \Gamma_1 \tag{11a}$$

and

$$R(\zeta) = - \frac{z_{\zeta\zeta}}{z_\zeta^3} \quad \text{on } \Gamma_2 \tag{11b}$$

The first and second order derivaties in (10) and (11) are written in central difference form. The expressions for Q and R are as follows:

$$Q(J) = - \frac{y(J+1,1) - 2*y(J,1) + y(J-1,1)}{(0.5*(y(J+1,1) - y(J-1,1)))**3} \tag{12a}$$

$$2 \leq J \leq JMAX-1$$

and

$$R(K) = - \frac{z(1,K+1) - 2*z(1,K) + z(1,K-1)}{(0.5*(z(1,K+1) - z(1,K-1)))**3} \tag{12b}$$

$$2 \leq K \leq KMAX-1$$

Equations (10a) and (10b) are solved by successive over

relaxation (S.O.R) technique. The intermediate value for y and z are as follows:

$$\bar{Y}(J,K) = [\alpha(y(J+1,K) + y(J-1,K)) - 2\beta y_{\eta\zeta} + \gamma(y(J,K+1) + y(J,K-1)) + J^2(y_\eta Q(J) + y_\zeta R(K))] / [2(\alpha+\gamma)] \quad (13a)$$

where

$$y_{\eta\zeta} = 0.25\,[y(J+1,K+1) - y(J+1,K-1) - y(J-1,K+1) + y(J-1,K-1)] \quad (13b)$$

$$y_\eta = 0.5\,[y(J+1,K) - y(J-1,K)] \quad (13c)$$

$$y_\zeta = 0.5\,[y(J,K+1) - y(J,K-1)] \quad (13d)$$

$$\alpha = 0.25\,[(y(J,K+1) - y(J,K-1))^2 + (Z(J,K+1) - z(J,K-1))^2] \quad (13e)$$

$$\beta = 0.25\,[(y(J+1,K) - y(J-1,K))(y(J,K+1) - y(J,K-1)) + (z(J+1,K) - z(J-1,K))(z(J,K+1) - z(J,K-1))] \quad (13f)$$

$$\gamma = 0.25\,[(y(J+1,K) - y(J-1,K))^2 + z(J+1,K) - z(J-1,K))^2] \quad (13g)$$

$$J = 0.25\,[(y(J+1,K) - y(J-1,K))(z(J,K+1) - z(J,K-1)) - (y(J,K+1) - y(J,K-1))(z(J+1,K) - z(J-1,K))] \quad (13h)$$

similarly an expression for $\bar{z}$ can be written out. For each grid

point two finite difference equations (one for $\bar{y}$ and one for $\bar{z}$) are obtained. The subscripts J and K have a range as follows:

J = 2,3,4, ----, JMAX-1

K = 2,3,4, ----, KMAX-1

This results in a set of (JMAX-2)(KMAX-2). The above equations are solved simultaneously using S.O.R. For a set of non-linear differential equations for some variable f, S.O.R. iteration can be written as follows:

$$f^{(s+1)} = w\,\bar{f} + (1 - w)\,f^{(s)} \tag{14}$$

where superscripts denote the iteration number, ω is the acceleration parameter, and the latest values are used in the difference equations to solve $\bar{f}$. In the present problem f denotes y or z.

Since on Γ_3 and Γ_4, Neumann type boundary conditions are applied, these conditions are implemented as follows:

on Γ_3

$$y(J,KMAX) = y(J,KMAX-1) \qquad 2 \leq J \leq JMAX \tag{15a}$$

on Γ_4

$$z(JMAX,K) = z(JMAX-1,K) \qquad 2 \leq K \leq KMAX \tag{15b}$$

Since on Γ_3 z = constant and on Γ_4 y = constant, equations (15a) and (15b) ensure that coordinate lines on Γ_3 and Γ_4 are orthogonal even though the coordinate system is not necessarily orthogonal everywhere.

C. Initial Guess and Boundary Point Specification

The input to the program consists of points specification on Γ_1 and Γ_2 boundary. Γ_1 boundary is such that the variation of z close to the JMAX point is zero. Similarly on Γ_2 the y variation close to KMAX point is zero. Points on Γ_1 and Γ_2 are specified implied that {y(J,1), z(J,1)} $1 \leq J \leq$ JMAX and {y(1,K), z(1,K)} $1 \leq K \leq$ KMAX are known. Then for the initial guess we have,

$$z(J,KMAX) = z(1,KMAX), \quad 2 \leq J \leq JMAX \tag{16a}$$

$$y(JMAX,K) = y(JMAX,1), \quad 2 \leq K \leq KMAX \tag{16b}$$

also,

$$y(J,K) = y(1,K) + (y(JMAX,K) - y(1,K)) * ((y(J,1) - y(1,1))/(y(JMAX,1) - y(1,1))) \tag{16c}$$

$$z(J,K) = z(J,1) + (z(J,KMAX) - z(J,1))*((z(1,K) - z(1,1))/(z(1,KMAX) - z(1,1))) \tag{16d}$$

$$2 \leq J \leq JMAX-1 \quad \text{and} \quad 2 \leq K \leq KMAX-1$$

and

$$y(J,KMAX) = y(J,KMAX-1), \quad \leq J \leq JMAX-1 \tag{16e}$$

$$z(JMAX,K) = z(JMAX-1,K), \quad \leq K \leq KMAX-1 \tag{16f}$$

D. x-Coordinate and Metric Coefficients

As discussed before, the y and z variation is computed only for one plane $\xi = 1$; also x is constant for any specified ξ-plane. Thus if x,y,z variation on $1 \leq \xi \leq$ IMAX, $\eta = 1$ and $\zeta = 1$ is specified then coordinates of any point can be easily specified. For example say for $\xi =$ I1,

$$x(I1,1,1) = x1 \tag{17a}$$

$$y(I1,1,1) = y1 \tag{17b}$$

$$z(I1,1,1) = z1 \tag{17c}$$

Then for any point I1, J, K in ξ = I1 plane we have,

$$x(I1,J,K) = x1 \tag{17d}$$

$$y(I1,J,K) = y(I1,1,1) - y(1,1,1) + y(1,J,K) \tag{17e}$$

$$z(I1,J,K) = z(I1,1,1) - z(1,1,1) + z(1,J,K) \tag{17f}$$

If the coordinate system on $\xi = 1$ plane is continuous in y and z, and if on $\xi = 1$, IMAX, $\eta = 1$, $\zeta = 1$ line the variation in x, y, z is continuous then the solution given by above equations will be continuous. This is ensured by having continuous y and z variation on Γ_1 and Γ_2 and on line $\xi = 1$, IMAX, $\eta = 1$, $\zeta = 1$.

A brief discussion on metric coefficients can now be included. The nine metric coefficients β_{ij} are given by:

$$\beta_{11} = (y_\eta\, z_\zeta - y_\zeta\, z_\eta)/J \tag{18a}$$

$$\beta_{12} = (y_\zeta\, z_\xi - y_\xi\, z_\zeta)/J \tag{18b}$$

$$\beta_{13} = (y_\xi\, z_\eta - y_\eta\, z_\xi)/J \tag{18c}$$

$$\beta_{21} = (x_\zeta\, z_\eta - x_\eta\, z_\zeta)/J \tag{18d}$$

$$\beta_{22} = (x_\xi\, z_\zeta - x_\zeta\, z_\xi)/J \tag{18e}$$

$$\beta_{23} = (x_\eta\, z_\xi - x_\xi\, z_\eta)/J \tag{18f}$$

$$\beta_{31} = (x_\eta\, y_\zeta - x_\zeta\, y_\eta)/J \tag{18g}$$

$$\beta_{32} = (x_\zeta\, y_\xi - x_\xi\, y_\zeta)/J \tag{18h}$$

$$\beta_{33} = (x_\xi\, y_\eta - x_\eta\, y_\xi)/J \tag{18i}$$

Since x is constant for any specified ξ plane, we have $x_\eta = x_\zeta = 0$ which considerably simplifies the above equations. Also, we can ensure that $y_\xi = 0$, $z_\xi = 0$ if y and z are specified to be constant on $1 \leq \xi \leq IMAX$, $\eta = 1$, $\zeta = 1$ line. In such a case the metric coefficients become,

$$\beta_{11} = (y_\eta\, z_\zeta - y_\zeta\, z_\eta)/J \tag{19a}$$

$$\beta_{12} = 0.0 \tag{19b}$$

$$\beta_{13} = 0.0 \tag{19c}$$

$$\beta_{21} = 0.0 \tag{19d}$$

$$\beta_{22} = x_\xi\, z_\zeta/J \tag{19e}$$

$$\beta_{23} = -\, x_\xi\, z_\eta/J \tag{19f}$$

$$\beta_{31} = 0.0 \tag{19g}$$

$$\beta_{32} = -x_\xi\, y_\zeta/J \tag{19h}$$

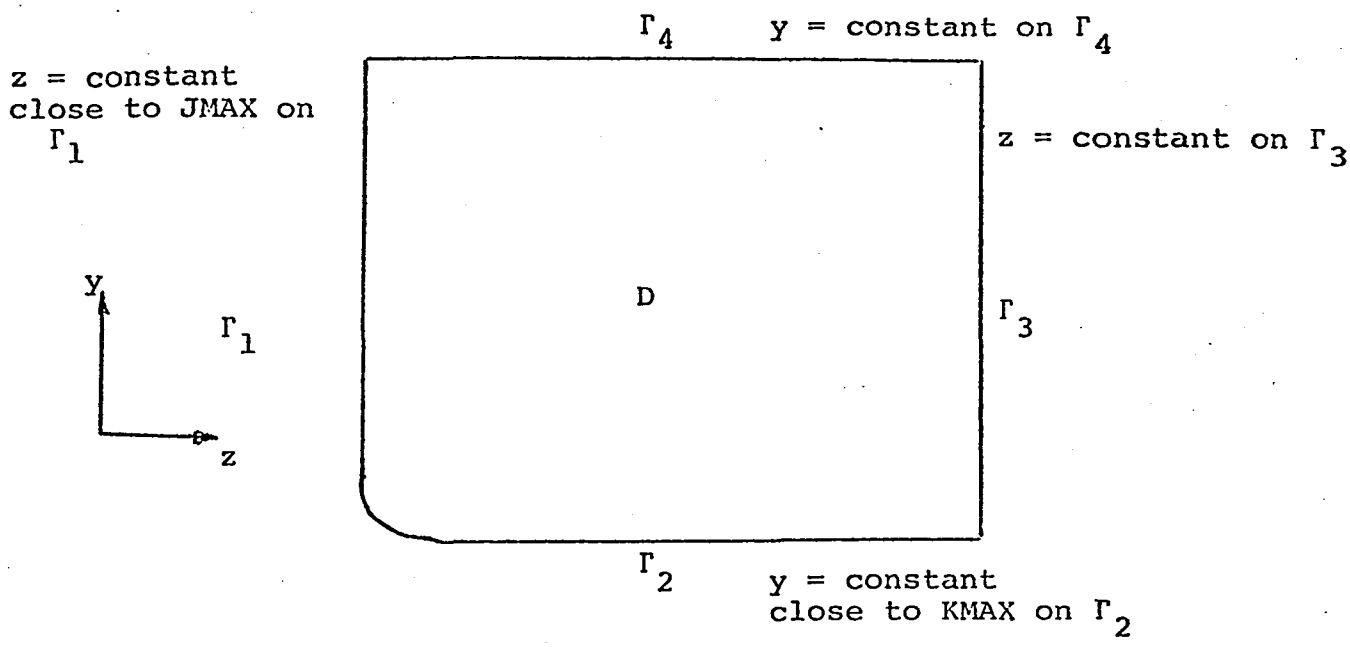
$$\beta_{33} = x_\xi\, y_\eta/J \tag{19i}$$

$$J = x_\xi(y_\eta\, z_\zeta - y_\zeta\, z_\eta) \tag{19j}$$

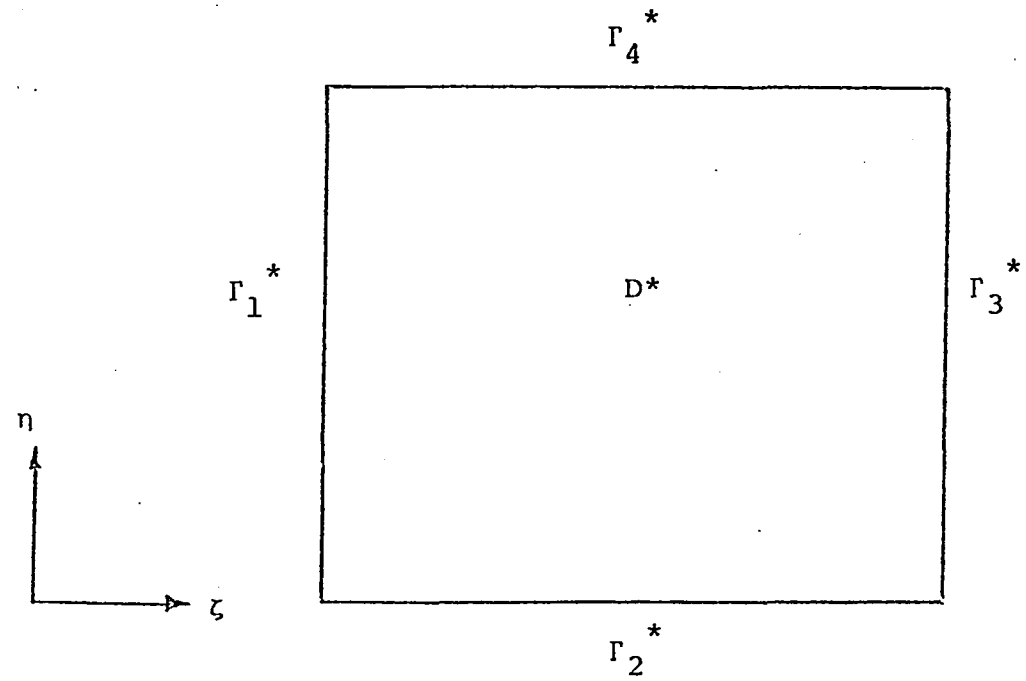
E. Concluding Remarks

A method for generating 3-D body-fitted coordinate system for corner flows has been presented. The computer code to generate coordinate systems using equations 12, 13, 14, 15, 16 is presented in Chapter 3. The coverged coordinate system is stored on a disk file which becomes a part of the input to the Navier Stokes code. Successful coordinate systems for coners

with or without fillet and for a 3-D flat plate have been generated (the program can generate suitable coordinate systems for a 3-D flat plate problem). The compressible laminar Navier Stokes solution for the above mentioned problems at low Reynolds number have already been obtained. Effort is now underway to obtain solution for higher Reynolds number with or without turbulence.



Physical Plane



Transformed Plane

II. Computer Program

A. Listing of the Computer Program

The program listing follows this page.

```
:REATION RUN                    CARDS ENCOUNTERED IN INPUT          UPDATE 1.2-452.    79/06/26. 13.30.37.      PAGE   1

****    *DECK  CORD1                                                                                    CORD1         1
            PROGRAM COORDC(INPUT,OUTPUT,SOLNXX,TAPE1=SOLNXX,TAPE5=INPUT,                                CORD1         2
           1     TAPE6=OUTPUT)                                                                         CORD1         3
        C**                                                                                             CORD1         4
        C*****************************************************************************                 CORD1         5
        C**                                                                           *                 CORD1         6
        C**       CORNER COORDINATE GENERATION PROGRAM BASED ON METHOD                *                 CORD1         7
        C**       DEVELOPED BY THOMPSON, THAMES, AND MASTIN OF MISSISSIPPI            *                 CORD1         8
        C**       STATE UNIVERSITY.                                                   *                 CORD1         9
        C**                                                                           *                 CORD1        10
        C**       FOR FURTHER ENQUIRIES CONTACT :                                     *                 CORD1        11
        C**                                                                           *                 CORD1        12
        C**            DR. JULIUS HARRIS                                              *                 CORD1        13
        C**            DR. DILIP KUMAR                                                *                 CORD1        14
        C**            NASA - LANGLEY RESEARCH CENTER                                 *                 CORD1        15
        C**            MAIL STOP 163                                                  *                 CORD1        16
        C**            HAMPTON, VA. 23665                                             *                 CORD1        17
        C**            PHONE:  804-827-3696                                           *                 CORD1        18
        C**            FTS CODE 928-1110                                              *                 CORD1        19
        C**                                                                           *                 CORD1        20
        C**                                                                           *                 CORD1        21
        C*****************************************************************************                 CORD1        22
        C**                                                                                             CORD1        23
        C**       THIS PROGRAM GENERATES COORDINATE SYSTEM FOR A 2-D CORNER                             CORD1        24
        C**       WITH OR WITHOUT FILLER. THE METHOD USED IS BASED ON METHOD                            CORD1        25
        C**       DEVELOPED BY THOMPSON, THAMES, MASTIN AND OTHERS AT MISS.                             CORD1        26
        C**       STATE UNIVERSITY. THE CORNER IS SUCH THAT EXCEPT IN THE                               CORD1        27
        C**       FILLER (WHEN PRESENT) L.H.S. BOUNDARY HAS Z = 0 AND BOTTOM                            CORD1        28
        C**       SIDE BOUNDARY HAS Y = 0. ALSO  R.H.S. BOUNDARY ALWAYS HAS                             CORD1        29
        C**       Z = CONSTANT  AND UPPER BOUNDARY HAS Y = CONSTANT. THE                                CORD1        30
        C**       PROGRAM GENERATES Q AND R SUCH THAT COORDINATE CLUSTERING                             CORD1        31
        C**       IS PROPORTIONAL TO COORDINATE POINTS DISTRIBUTION ON L.H.S.                           CORD1        32
        C**       AND BOTTOM BOUNDARY. ON R.H.S. AND TOP BOUNDARIES NEUMANN                             CORD1        33
        C**       TYPE BOUNDARY CONDITIONS ARE USED. THIS ENSURES THAT                                  CORD1        34
        C**       COORDINATE LINES ON R.H.S. AND UPPER BOUNDARY ARE PARALLEL                            CORD1        35
        C**       TO SIDES.                                                                             CORD1        36
        C**                                                                                             CORD1        37
        C**                                                                                             CORD1        38
        C**                                                                                             CORD1        39
        C**                    --- CARD INPUT ---                                                       CORD1        40
        C**                                                                                             CORD1        41
        C**       CARD1:  SPECIFIES FIELD SIZE AND PRINT, PLOT FLAGS                                    CORD1        42
        C**               JMAX,KMAX,IPRT1,IPRT2,IPLT1,IPLT2,ITERMX,R1,R2,R3                             CORD1        43
        C**               (FORMAT 7I5,5X,3F10.0)                                                        CORD1        44
```

```
C**          JMAX   : NO OF NODES IN Y-DIRECTION                              CORD1      45
C**          KMAX   : NO OF NODES IN Z-DIRECTION                              CORD1      46
C**          IPRT1  : INITIAL GUESS IS TO BE PRINTED OR NOT                   CORD1      47
C**                   IF IPRT1 = 1 THEN PARTIALLY CONVERGED                   CORD1      48
C**                   SOLUTION IS PRINTED EVERY 50 ITER.                      CORD1      49
C**          IPRT2  : FINAL SOLUTION TO BE PRINTED OR NOT                     CORD1      50
C**          IPLT1  : INITIAL GUESS TO BE PLOTTED ON NOT                      CORD1      51
C**          IPLT2  : FINAL  SOLUTION TO BE PLOTTED OR NOT                    CORD1      52
C**          ITERMX : MAXIMUM NO. OF ITERATIONS ALLOWED                       CORD1      53
C**          R1     : GAUSS SEIDELL ITERATION PARAMETER                       CORD1      54
C**                   (USUALLY 0.6)                                           CORD1      55
C**          R2     : CONVERGENCE CRITERION IN Y-DIRECTION                    CORD1      56
C**                   (USUALLY 0.000005 TO 0.00001)                           CORD1      57
C**          R3     : CONVERGENCE CRITERION IN Z-DIRECTION                    CORD1      58
C**                   (USUALLY SAME AS R2)                                    CORD1      59
C**                                                                           CORD1      60
C**                 NOTE : FOR ALL FLAGS 0 MEANS NO, 1 MEANS YES              CORD1      61
C**                                                                           CORD1      62
C**                                                                           CORD1      63
C**     CARD2:  CARDS TO SPECIFY POINT DISTRIBUTION ON L.H.S.                 CORD1      64
C**             BOUNDARY (Y-DIRECTION) NO OF CARDS = JMAX                     CORD1      65
C**             Y,Z  (FORMAT 2F10.0)                                          CORD1      66
C**             Y      : Y COORDINATE                                         CORD1      67
C**             Z      : Z COORDINATE                                         CORD1      68
C**                                                                           CORD1      69
C**     CARD3:  CARDS TO SPECIFY POINT DISTRIBUTION ON BOTTOM                 CORD1      70
C**             BOUNDARY (Z-DIRECTION). NO. OF CARDS = KMAX.                  CORD1      71
C**             Y,Z  (FORMAT 2F10.0)  SEE CARD TYPE 2)                        CORD1      72
C**                                                                           CORD1      73
C**     CARD4:  SPECIFIES NO. OF NODES IN X-DIRECTION.                        CORD1      74
C**             IMAX              (FORMAT(I5)                                 CORD1      75
C**             IMAX   : NO. OF POINTS IN X-DIRECTION.                        CORD1      76
C**                                                                           CORD1      77
C**     CARD5:  CARDS TO SPECIFY NO. OF POINTS IN X - DIRECTION.              CORD1      78
C**             X,Y1,Z1                (FORMAT 3F10.0)                        CORD1      79
C**             X      : X-COORDINATE ON J=1,K=1 LINE (X-DIRECTION)           CORD1      80
C**             Y1     : Y-COORDINATE ON J=1,K=1 LINE (X-DIRECTION)           CORD1      81
C**             Z1     : Z-COORDINATE ON J=1,K=1 LINE (X-DIRECTION)           CORD1      82
C**                                                                           CORD1      83
C**     CARD6:  PRINT,PLOT HEADINGS.                                          CORD1      84
C**             HED1(4)      (FORMAT  4A10)                                   CORD1      85
C**             HED1   : 40 BYTES OF PLOT HEADING                             CORD1      86
C**     CARD7:  ADDITIONAL PRINT PLOT HEADINGS.                               CORD1      87
C**             HED2(4)      (FORMAT  4A10)                                   CORD1      88
```

```
C**             HED2   : 40 BYTES OF ADDITIONAL PLOT HEADING.                CORD1      89
C**                                                                           CORD1      90
C**       SAMPLE RUNSTREAM.                                                   CORD1      91
C**                                                                           CORD1      92
C**                                                                           CORD1      93
C**       KVIS2,T777,CM70000.                    BIN 34    D KUMAR          CORD1      94
C**       USER,479019C.                                                       CORD1      95
C**       CHARGE,100718,LRC.                                                  CORD1      96
C**       GET(OLDPL=KMRCD1/UN=375732N).                                       CORD1      97
C**       UPDATE(F,C=VISFL1).                                                 CORD1      98
C**       COPYBR,INPUT,EDTFT1.                                                CORD1      99
C**       REWIND,VISFL1,EDTFT1.                                               CORD1     100
C**       EDIT,VISFL1,I=EDTFT1.                                               CORD1     101
C**       REWIND,VISFL1.                                                      CORD1     102
C**       FTN(OPT=1,I=VISFL1).                                                CORD1     103
C**       ATTACH,LRCGOSF/UN=LIBRARY,NA.                                       CORD1     104
C**       LDSET,LIB=LRCGOSF,PRESET=ZERO.                                      CORD1     105
C**       LGO.                                                                CORD1     106
C**       PLOT.VARIAN                                                         CORD1     107
C**       REWIND,SOLNXX.                                                      CORD1     108
C**       COPYBR,SOLNXX,FILETC.                                               CORD1     109
C**       REWIND,FILETC.                                                      CORD1     110
C**       SAVE,FILETC.                                                        CORD1     111
C**       7/8/9                                                               CORD1     112
C**       *IDENT DUM1                                                         CORD1     113
C**       *INSERT CORD1.3                                                     CORD1     114
C**       C**                                                                 CORD1     115
C**       7/8/9                                                               CORD1     116
C**       RS:/ MXLL/,/   30/;*                                                CORD1     117
C**       RS:/ MXLL/,/   23/;*                                                CORD1     118
C**       RS:/ MXLL/,/   23/;*                                                CORD1     119
C**        END                                                                CORD1     120
C**       7/8/9                                                               CORD1     121
C**            INPUT DATA.  SEE EXPLANATION ABOVE.                            CORD1     122
C**       6/7/8/9                                                             CORD1     123
C**                                                                           CORD1     124
C**                                                                           CORD1     125
C**                                                                           CORD1     126
C**       REWIND,VISFL1.                                                      CORD1     127
C**       NOTE: AFTER FIRST 7/8/9 CARD 3 CARDS ARE INPUT TO                   CORD1     128
C**             SATISFY INPUT FOR UPDATE UTILITY.                             CORD1     129
C**                                                                           CORD1     130
C**             AFTER 2ND 7/8/9 CARD EDIT CARDS ARE INPUT.  THE               CORD1     131
C**             PROGRAM IS WRITTEN SO THAT MAXIMUM NUMBER OF                  CORD1     132
```

```
C**            POINTS IN X,Y,Z DIRECTION CAN BE CHANGED BY SIMPLE          CORD1      133
C**            INPUT.  IN THE ABOVE CASE FIELD SIZE OF 30,23,23            CORD1      134
C**            IS ASSUMED.  IN THE THREE CARDS AFTER RS:/ AND MXLL         CORD1      135
C**            I,J,K SHOULD BE INPUT RESPECTIVELY.  THIS COULD NOT         CORD1      136
C**            BE SHOWN ABOVE BECAUSE TEXT EDITOR CHANGES TEXT IN          CORD1      137
C**            COMMENTS CARDS ALSO.                                        CORD1      138
C**                                                                        CORD1      139
C**            IT IS ASSUMED THAT PROGRAM RESIDES ON FILE KMRCD1           CORD1      140
C**            UNDER USER NO. OF 375732N AND USER 479019C HAS              CORD1      141
C**            PERMISSION TO USE IT.  FURTHERMORE, IT IS ALSO ASSUMED      CORD1      142
C**            THAT THE COORDINATE SYSTEM SOLUTION IS TO BE STORED         CORD1      143
C**            ON FILE FILETC.  THIS NAME CAN BE CHANGED BY                CORD1      144
C**            CHANGING THREE CARDS TOWARDS THE END OF THE RUNSTREAM.      CORD1      145
C**                                                                        CORD1      146
C**            THE PLOTS ON CALCOMP CAN BE OBTAINED BY REPLACING           CORD1      147
C**            PLOT.VARIAN CARD BY FOLLOWING CARDS.                        CORD1      148
C**              PLOT.CALPOST,11(XO=1.0,YO=0.2)                            CORD1      149
C**              CONT.  //PAPER 00,                                        CORD1      150
C**              CONT.    LEROY PEN SIZE 0.3                               CORD1      151
C**              CONT.    MULTIPLE PLOT MODE                               CORD1      152
C**              CONT.    THESE PLOTS ARE FOR REPORTS//                    CORD1      153
C**                                                                        CORD1      154
C**                                                                        CORD1      155
      COMMON /BXXX/ Y(JMXLL,KMXLL),Z(JMXLL,KMXLL),Q(JMXLL),R(KMXLL),       CORD1      156
     1    X(IMXLL),Y1(IMXLL),Z1(IMXLL)                                      CORD1      157
      COMMON /AXXX/ ILIMIT,JLIMIT,KLIMIT,IMAX,JMAX,KMAX,IPRT1,IPRT2,       CORD1      158
     1        IPLT1,IPLT2,ITERMX,R1,R2,R3,I1,HED1(4),HED2(4)                CORD1      159
C**                                                                        CORD1      160
      DATA KKK,YERRMX,ZERRMX,I1,IYLOCJ,IYLOCK,IZLOCJ,IZLOCK /0,2*0.,5*0/   CORD1      161
      ILIMIT = IMXLL                                                       CORD1      162
      JLIMIT = JMXLL                                                       CORD1      163
      KLIMIT = KMXLL                                                       CORD1      164
C**                                                                        CORD1      165
C**            INITIALIZE Q,R,Y AND Z TO ZERO.                             CORD1      166
C**                                                                        CORD1      167
      DO 10 J = 1,JLIMIT                                                   CORD1      168
         Q(J) = 0.0                                                        CORD1      169
   10 CONTINUE                                                             CORD1      170
      DO 20 K = 1,KLIMIT                                                   CORD1      171
         R(K) = 0.0                                                        CORD1      172
   20 CONTINUE                                                             CORD1      173
      DO 30 J = 1,JLIMIT                                                   CORD1      174
      DO 30 K = 1,KLIMIT                                                   CORD1      175
         Y(J,K) = 0.0                                                      CORD1      176
```

```
            Z(J,K) = 0.0                                                                    CORD1      177
      30  CONTINUE                                                                          CORD1      178
C**                                                                                         CORD1      179
C**       READ INPUT                                                                        CORD1      180
C**                                                                                         CORD1      181
      READ(5,1000)  JMAX,KMAX,IPRT1,IPRT2,IPLT1,IPLT2,ITERMX,R1,R2,R3                       CORD1      182
      IF(JMAX.GT.JLIMIT.OR.KMAX.GT.KLIMIT)  GO TO 50                                        CORD1      183
            GO TO 60                                                                        CORD1      184
C**                                                                                         CORD1      185
      50 WRITE(6,1020) JMAX,KMAX,JLIMIT,KLIMIT                                              CORD1      186
         CALL EXIT                                                                          CORD1      187
C**                                                                                         CORD1      188
      60 CONTINUE                                                                           CORD1      189
         READ(5,1030) ((Y(J,1),Z(J,1)),J=1,JMAX)                                            CORD1      190
         READ(5,1030) ((Y(1,K),Z(1,K)),K=1,KMAX)                                            CORD1      191
C**                                                                                         CORD1      192
C**       READ IN POINTS ALONG J = 1, K = 1 LINE (X-DIRECTION)                              CORD1      193
C**                                                                                         CORD1      194
      READ(5,1000) IMAX                                                                     CORD1      195
      IF(IMAX.LE.ILIMIT) GO TO 65                                                           CORD1      196
            WRITE(6,1090) IMAX,ILIMIT                                                       CORD1      197
            STOP                                                                            CORD1      198
      65 CONTINUE                                                                           CORD1      199
C**                                                                                         CORD1      200
      READ(5,1100) ((X(I),Y1(I),Z1(I)),I=1,IMAX)                                            CORD1      201
C**                                                                                         CORD1      202
C**   SPECIFY INITIAL GUESS ON THE WHOLE FIELD                                              CORD1      203
C**                                                                                         CORD1      204
      DO 70 J = 2,JMAX                                                                      CORD1      205
         Z(J,KMAX) = Z(1,KMAX)                                                              CORD1      206
      70 CONTINUE                                                                           CORD1      207
         DO 80 K = 2,KMAX                                                                   CORD1      208
         Y(JMAX,K) = Y(JMAX,1)                                                              CORD1      209
      80 CONTINUE                                                                           CORD1      210
C**                                                                                         CORD1      211
      J1 = JMAX - 1                                                                         CORD1      212
      K1 = KMAX - 1                                                                         CORD1      213
      DO 100 J = 2,J1                                                                       CORD1      214
      DO 100 K = 2,K1                                                                       CORD1      215
         Y(J,K) = (Y(JMAX,K) - Y(1,K)) * ((Y(J,1) - Y(1,1)) /                               CORD1      216
     1            (Y(JMAX,1) - Y(1,1))) + Y(1,K)                                          CORD1      217
         Z(J,K) = (Z(J,KMAX) - Z(J,1)) * ((Z(1,K) - Z(1,1)) /                               CORD1      218
     1            (Z(1,KMAX) - Z(1,1))) + Z(J,1)                                          CORD1      219
     100 CONTINUE                                                                           CORD1      220
```

```
      DO 102 J = 2,J1                                                   CORD1  221
          Y(J,KMAX) = Y(J,KMAX-1)                                       CORD1  222
  102 CONTINUE                                                          CORD1  223
      DO 104 K = 2,K1                                                   CORD1  224
          Z(JMAX,K) = Z(JMAX-1,K)                                       CORD1  225
  104 CONTINUE                                                          CORD1  226
C**                                                                     CORD1  227
C**          READ PRINT,PLOT HEADINGS.  TWO CARDS ARE READ IN.          CORD1  228
C**         CALL PLOT INITIALIZING ROUTINE PSEUDO IF PLOT OPTIONS       CORD1  229
C**         SET.                                                        CORD1  230
C**                                                                     CORD1  231
      READ(5,1080) (HED1(I),I=1,4)                                      CORD1  232
      READ(5,1080) (HED2(I),I=1,4)                                      CORD1  233
      IF(IPLT1.EQ.0.AND.IPLT2.EQ.0) GO TO 110                           CORD1  234
          CALL PSEUDO                                                   CORD1  235
          CALL FONTS(1)                                                 CORD1  236
  110 CONTINUE                                                          CORD1  237
C**                                                                     CORD1  238
C**       SEE IF INITIAL GUESS NEEDS TO BE PRINTED                      CORD1  239
C**                                                                     CORD1  240
      IF(IPRT1.EQ.1) CALL IPRTC(1,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK,   CORD1  241
     1                          IZLOCJ,IZLOCK)                         CORD1  242
C**                                                                     CORD1  243
C**       SEE IF INITIAL GUESS NEED TO BE PLOTTED                       CORD1  244
C**                                                                     CORD1  245
       IF(IPLT1.EQ.1) CALL  IPLTC(1,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK, CORD1  246
     1                           IZLOCJ,IZLOCK)                         CORD1  247
C**                                                                     CORD1  248
C**       COMPUTE  Q AND R. FIRST YETA  OR ZTAU IS COMPUTED.            CORD1  249
C**       Q =- YETAETA / (YETA*YETA*YETA)                               CORD1  250
C**       R =- ZTAUTAU / (ZTAU*ZTAU*ZTAU)                               CORD1  251
C**                                                                     CORD1  252
      DO 120 J = 2,J1                                                   CORD1  253
          A1 = 0.5 * (Y(J+1,1) - Y(J-1,1))                              CORD1  254
          Q(J) = - (Y(J+1,1) - 2.0*Y(J,1) + Y(J-1,1)) / (A1*A1*A1)      CORD1  255
  120 CONTINUE                                                          CORD1  256
C**                                                                     CORD1  257
      DO 140 K = 2,K1                                                   CORD1  258
          A1 = 0.5 * (Z(1,K+1) - Z(1,K-1))                              CORD1  259
          R(K) = - (Z(1,K+1) - 2.0*Z(1,K) + Z(1,K-1)) / (A1*A1*A1)      CORD1  260
  140 CONTINUE                                                          CORD1  261
C**                                                                     CORD1  262
C**        START  COMPUTING  SOLUTION.                                  CORD1  263
C**                                                                     CORD1  264
```

```
      KKK    = 0                                                                  CORD1       265
      IERR   = 0                                                                  CORD1       266
      DO  500  I1 = 1,ITERMX                                                      CORD1       267
          YERRMX = 0.0                                                            CORD1       268
          ZERRMX = 0.0                                                            CORD1       269
          CALL CALCOR(YERRMX,ZERRMX,IYLOCJ,IYLOCK,IZLOCJ,IZLOCK)                  CORD1       270
C**                                                                               CORD1       271
C**           CHECK IF MAX ERROR TO BE PRINTED OR NOT                             CORD1       272
C**                                                                               CORD1       273
          IERR  = IERR + 1                                                        CORD1       274
          IF(IERR.NE.50)  GO TO 160                                               CORD1       275
             IERR = 0                                                             CORD1       276
             WRITE(6,1040) I1,IYLOCJ,IYLOCK,YERRMX,IZLOCJ,IZLOCK,ZERRMX           CORD1       277
      IF(IPRT1.EQ.1) CALL IPRTC(2,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK,             CORD1       278
     1                          IZLOCJ,IZLOCK)                                   CORD1       279
  160     CONTINUE                                                                CORD1       280
C**                                                                               CORD1       281
C**           CHECK FOR CONVERGENCE                                               CORD1       282
C**                                                                               CORD1       283
          IF(YERRMX.GT.R2) GO TO 180                                              CORD1       284
          IF(ZERRMX.GT.R3) GO TO 180                                              CORD1       285
C**                                                                               CORD1       286
C**           SOLUTION CONVERGED, GET OUT OF THE LOOP, KKK = 1                    CORD1       287
C**           INDICATES SOLUTION CONVERGED                                        CORD1       288
C**                                                                               COPD1       289
             KKK = 1                                                              CORD1       290
             GO TO 520                                                            CORD1       291
C**                                                                               CORD1       292
  180    .CONTINUE                                                                CORD1       293
  500 CONTINUE                                                                    CORD1       294
  520 CONTINUE                                                                    CORD1       295
C**                                                                               CORD1       296
C**       SEE IF CONVERGED OR PARTIALLY CONVERGED SOLUTION IS TO                  CORD1       297
C**       BE PLOTTED AND PRINTED OR NOT                                           CORD1       298
C**                                                                               CORD1       299
      IF(IPRT2.EQ.1) CALL IPRTC(2,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK,             CORD1       300
     1               IZLOCJ,IZLOCK)                                             CORD1       301
C**                                                                               CORD1       302
      IF(IPLT2.EQ.1) CALL IPLTC(2,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK,             CORD1       303
     1               IZLOCJ,IZLOCK)                                             CORD1       304
C**                                                                               CORD1       305
C**           STORE CONVERGED SOLUTION                                            CORD1       306
C**                                                                               CORD1       307
      IF(KKK.EQ.0) GO TO  540                                                     CORD1       308
```

```
      WRITE(1,1110) ILIMIT,JLIMIT,KLIMIT,IMAX,JMAX,KMAX,                CORD1   309
     1     ((HED1(I),HED2(I)),I=1,4)                                   CORD1   310
      WRITE(1,1120) (((Y(J,K),Z(J,K)),K=1,KLIMIT),J=1,JLIMIT)          CORD1   311
      WRITE(1,1120) ((X(1),Y1(I),Z1(I)),I=1,ILIMIT)                    CORD1   312
          GO TO 560                                                    CORD1   313
C**                                                                    CORD1   314
C**       SOLUTION DID NOT CONVERGE. WRITE A MESSAGE                   CORD1   315
C**                                                                    CORD1   316
  540 CONTINUE                                                         CORD1   317
      WRITE(6,1060) YERRMX,IYLOCJ,IYLOCK,ZERRMX,IZLOCJ,IZLOCK,R2,R3,   CORD1   318
     1              ITERMX                                            CORD1   319
C**                                                                    CORD1   320
C**       ALL FUNCTIONS COMPLETE                                       CORD1   321
C**                                                                    CORD1   322
  560 CONTINUE                                                         CORD1   323
      STOP                                                             CORD1   324
C**                                                                    CORD1   325
C**       FORMAT STATEMENTS                                            CORD1   326
C**                                                                    CORD1   327
 1000 FORMAT(7I5,5X,3F10.0)                                            CORD1   328
 1020 FORMAT("1"////5X,"*** NO OF POINTS IN Y OR Z GREATER THAN",      CORD1   329
     1       " MAXIMUM ALLOWED."//5X,"*** POINTS INPUT = ",2I5,//5X,   CORD1   330
     2       " MAXIMUM ALLOWED = ",2I5,//5X,"*** RUN ABORTED")         CORD1   331
 1030 FORMAT(2F10.0)                                                   CORD1   332
 1040 FORMAT("0",5X,"ITER NO.,IYLOCJ,IYLOCK,YERRMX,IZLOCJ,IZLOCK,",    CORD1   333
     1       "ZERRMX = ",3I5,E15.6,2I5,E15.6 )                         CORD1   334
C**                                                                    CORD1   335
 1060 FORMAT("1"////5X,"*** SOLUTION DID NOT CONVERGE. SOLUTION",      CORD1   336
     1       " NOT STORED ON UNIT 1"//5X,"*** MAX ERROR IN Y = ",E15.6,CORD1   337
     2       " OCCURED AT LOCATION = ",I5,",",I5//5X,"*** MAX ERROR IN "CORD1   338
     3     ,"Z = ",E15.6," OCCURED AT LOCATION = ",I5,",",I5//5X,      CORD1   339
     4     "*** MAX ERROR ALLOWED IN Y AND Z = ",2E15.6//5X,"*** NO.", CORD1   340
     5     " OF ITERATIONS PERFORMED = ",I5)                           CORD1   341
 1080 FORMAT(4A10)                                                     CORD1   342
 1090 FORMAT("1"////5X,"*** NO. OF POINTS IN X-DIRECTION GREATER THAN",CORD1   343
     1    " MAXIMUM ALLOWED."//5X,"*** POINTS INPUT = ",I5,//5X,      CORD1   344
     2    " MAXIMUM ALLOWED = ",I5//5X,"*** RUN ABORTED.")            CORD1   345
 1100 FORMAT(3F10.0)                                                   CORD1   346
 1110 FORMAT(6I5,8A10)                                                 CORD1   347
 1120 FORMAT(1X,8E15.8)                                                CORD1   348
C**                                                                    CORD1   349
C**                                                                    CORD1   350
      END                                                              CORD1   351
      SUBROUTINE CALCOR(YERRMX,ZERRMX,IYLOCJ,IYLOCK,IZLOCJ,IZLOCK)     CORD1   352
```

```
C**                                                                    CORD1      353
C**       THIS SUBROUTINE ITERATES FOR COORDINATE SYSTEM SOLUTION.     CORD1      354
C**       THE SUBROUTINE IS CALLED FROM COORDC PROGRAM                 CORD1      355
C**                                                                    CORD1      356
      COMMON /BXXX/ Y(JMXLL,KMXLL),Z(JMXLL,KMXLL),Q(JMXLL),R(KMXLL),   CORD1      357
     1    X(IMXLL),Y1(IMXLL),Z1(IMXLL)                                 CORD1      358
      COMMON /AXXX/ ILIMIT,JLIMIT,KLIMIT,IMAX,JMAX,KMAX,IPRT1,IPRT2,   CORD1      359
     1        IPLT1,IPLT2,ITERMX,R1,R2,R3,I1,HED1(4),HED2(4)           CORD1      360
C**                                                                    CORD1      361
C**       COMPUTE  IN INNER FIELDS                                     CORD1      362
C**                                                                    CORD1      363
      J1 = JMAX - 1                                                    CORD1      364
      K1 = KMAX - 1                                                    CORD1      365
      DO 200 J = 2,J1                                                  CORD1      366
      DO 200 K = 2,K1                                                  CORD1      367
          YP0P0 = Y(J,K)                                               CORD1      368
          YP0P1 = Y(J,K+1)                                             CORD1      369
          YP0M1 = Y(J,K-1)                                             CORD1      370
          YP1P0 = Y(J+1,K)                                             CORD1      371
          YM1P0 = Y(J-1,K)                                             CORD1      372
          YM1M1 = Y(J-1,K-1)                                           CORD1      373
          YM1P1 = Y(J-1,K+1)                                           CORD1      374
          YP1M1 = Y(J+1,K-1)                                           CORD1      375
          YP1P1 = Y(J+1,K+1)                                           CORD1      376
          ZP0P0 = Z(J,K)                                               CORD1      377
          ZP0P1 = Z(J,K+1)                                             CORD1      378
          ZP0M1 = Z(J,K-1)                                             CORD1      379
          ZP1P0 = Z(J+1,K)                                             CORD1      380
          ZM1P0 = Z(J-1,K)                                             CORD1      381
          ZM1M1 = Z(J-1,K-1)                                           CORD1      382
          ZM1P1 = Z(J-1,K+1)                                           CORD1      383
          ZP1M1 = Z(J+1,K-1)                                           CORD1      384
          ZP1P1 = Z(J+1,K+1)                                           CORD1      385
          QQQ = Q(J)                                                   CORD1      386
          RRR = R(K)                                                   CORD1      387
          ALPHA = 0.25*((YP0P1 - YP0M1)**2.0 + (ZP0P1 - ZP0M1)**2.0)   CORD1      388
          BETA  = 0.25*((YP1P0 - YM1P0) * (YP0P1 - YP0M1) +            CORD1      389
     1                  (ZP1P0 - ZM1P0) * (ZP0P1 - ZP0M1))           CORD1      390
          GAMMA = 0.25*((YP1P0 - YM1P0)**2.0 + (ZP1P0 - ZM1P0)**2.0)   CORD1      391
          AJCB1 = 0.25*((YP1P0 - YM1P0) * (ZP0P1 - ZP0M1) -            CORD1      392
     1                  (YP0P1 - YP0M1) * (ZP1P0 - ZM1P0))           CORD1      393
C**                                                                    CORD1      394
          YYY1  = (0.5 * ALPHA * (YP1P0 + YM1P0) - 0.25 * BETA *       CORD1      395
     1        (YP1P1 - YP1M1 - YM1P1 + YM1M1) + 0.5 * GAMMA *          CORD1      396
```

```
     2        (YPOP1 + YPOM1)+ 0.25 * AJCB1 * AJCB1 *((YP1PO - YM1PO) *
     3        QQQ + (YPOP1 - YPOM1)*RRR)) / (ALPHA + GAMMA)
C**
      ZZZ1  = (0.5 * ALPHA * (ZP1PO + ZM1PO) - 0.25 * BETA *
     1         (ZP1P1 - ZP1M1 - ZM1P1 + ZM1M1) + 0.5 * GAMMA * (ZPOP1
     2         + ZPOM1) + 0.25*AJCB1*AJCB1*((ZP1PO-ZM1PO)*QQQ +
     3         (ZPOP1 - ZPOM1)*RRR)) / (ALPHA + GAMMA)
C**
C**     INTERMEDIATE VALUE CALCULATED. COMPUTE  NEW VALUE USING
C**     GAUSS SEIDELL ITERATION
C**
      YYY1 = R1 * YYY1 + (1.0 - R1) * YPOPO
      ZZZ1 = R1 * ZZZ1 + (1.0 - R1) * ZPOPO
C**
C**     FIND MAX ERROR AND ITS LOCATION
C**
      ERR1 = ABS(YYY1 - YPOPO)
      IF(ERR1.LE. YERRMX) GO TO 30
         IYLOCJ = J
         IYLOCK = K
         YERRMX = ERR1
   30 CONTINUE
C**
      ERR1 = ABS(ZZZ1 - ZPOPO)
      IF(ERR1.LE.ZERRMX) GO TO 60
         IZLOCJ = J
         IZLOCK = K
         ZERRMX = ERR1
   60 CONTINUE
C**
C**     SET NEW VALUES
C**
      Y(J,K)  = YYY1
      Z(J,K)  = ZZZ1
C**
  200 CONTINUE
C**
C**     APPLY NEUMANN CONDITIONS ON R.H.S. AND UPPER BOUNDARY
C**
      DO 220 K = 2,K1
         Z(JMAX,K) = Z(JMAX-1,K)
  220 CONTINUE
C**
      DO 240 J = 2,J1
```

```
CORD1     397
CORD1     398
CORD1     399
CORD1     400
CORD1     401
CORD1     402
CORD1     403
CORD1     404
CORD1     405
CORD1     406
CORD1     407
CORD1     408
CORD1     409
CORD1     410
CORD1     411
CORD1     412
CORD1     413
CORD1     414
CORD1     415
CORD1     416
CORD1     417
CORD1     418
CORD1     419
CORD1     420
CORD1     421
CORD1     422
CORD1     423
CORD1     424
CORD1     425
CORD1     426
CORD1     427
CORD1     428
CORD1     429
CORD1     430
CORD1     431
CORD1     432
CORD1     433
CORD1     434
CORD1     435
CORD1     436
CORD1     437
CORD1     438
CORD1     439
CORD1     440
```

```
            Y(J,KMAX) = Y(J,KMAX-1)                                          CORD1      441
  240 CONTINUE                                                               CORD1      442
C**                                                                          CORD1      443
      RETURN                                                                 CORD1      444
      END                                                                    CORD1      445
      SUBROUTINE IPRTC(ISOLN,KKK,YERRMX,ZERRMX,   IYLOCJ,IYLOCK,             CORD1      446
     1           IZLOCJ,IZLOCK)                                              CORD1      447
C**                                                                          CORD1      448
C**       THIS SUBROUTINE PRINTS THE COORDINATE SYSTEM IN Y AND Z            CORD1      449
C**       PLANE. ISOLN = 1 INDICATES INITIAL GUESS IS TO BE PRINTED.         CORD1      450
C**       ISOLN = 2 INDICATES INDICATES CONVERGED OR PARTIALLY               CORD1      451
C**       CONVERGED SOLUTION IS TO BE PRINTED. KKK = 0 INDICATES             CORD1      452
C**       PARTIALLY CONVERGED SOLUTION  AND KKK = 1 INDICATES CONVERGED      CORD1      453
C**       SOLUTION                                                           CORD1      454
C**                                                                          CORD1      455
      COMMON /BXXX/ Y(JMXLL,KMXLL),Z(JMXLL,KMXLL),Q(JMXLL),R(KMXLL),         CORD1      456
     1    X(IMXLL),Y1(IMXLL),Z1(IMXLL)                                        CORD1      457
      COMMON /AXXX/ ILIMIT,JLIMIT,KLIMIT,IMAX,JMAX,KMAX,IPRT1,IPRT2,         CORD1      458
     1        IPLT1,IPLT2,ITERMX,R1,R2,R3,I1,HED1(4),HED2(4)                  CORD1      459
C**                                                                          CORD1      460
C**       PRINT HEADINGS                                                     CORD1      461
C**                                                                          CORD1      462
      IF(ISOLN.EQ.1) WRITE(6,1000)                                           CORD1      463
      IF(ISOLN.EQ.2.AND.KKK.EQ.0) WRITE(6,1010)                              CORD1      464
      IF(ISOLN.EQ.2.AND.KKK.EQ.1) WRITE(6,1020)                              CORD1      465
      WRITE(6,1025) HED1                                                     CORD1      466
      WRITE(6,1025) HED2                                                     CORD1      467
      WRITE(6,1030) JMAX,KMAX                                                CORD1      468
      IF(ISOLN.EQ.1) GO TO 50                                                CORD1      469
        WRITE(6,1040) I1                                                     CORD1      470
        WRITE(6,1050) YERRMX,IYLOCJ,IYLOCK                                   CORD1      471
        WRITE(6,1060) ZERRMX,IZLOCJ,IZLOCK                                   CORD1      472
   50 CONTINUE                                                               CORD1      473
C**                                                                          CORD1      474
C**            PRINT ARRAYS                                                  CORD1      475
C**                                                                          CORD1      476
      DO 100 J = 1,JMAX                                                      CORD1      477
          WRITE(6,1070) J                                                    CORD1      478
          WRITE(6,1080) ((Y(J,K),Z(J,K)),K = 1,KMAX)                         CORD1      479
  100 CONTINUE                                                               CORD1      480
C**                                                                          CORD1      481
C**       PRINT J=1, K=1, I=1,IMAX                                           CORD1      482
C**                                                                          CORD1      483
      WRITE(6,1200)                                                          CORD1      484
```

```
      WRITE(6,1210) ((X(I),Y1(I),Z1(I)),I=1,IMAX)                        CORD1   485
C**                                                                      CORD1   486
C**       FORMAT STATEMENTS                                              CORD1   487
C**                                                                      CORD1   488
 1000 FORMAT("1"//51X,"*** INITIAL SOLUTION ***")                        CORD1   489
 1010 FORMAT("1"//44X,"*** PARTIALLY CONVERGED SOLUTION ***")            CORD1   490
 1020 FORMAT("1"//49X,"*** CONVERGED SOLUTION ***")                      CORD1   491
 1025 FORMAT(" ",42X,4A10)                                               CORD1   492
 1030 FORMAT("0",50X,"JMAX = ",I5,4X,"KMAX = ",I5)                       CORD1   493
 1040 FORMAT(" ",47X,"MAX ITERATIONS PERFORMED = ",I5)                   CORD1   494
 1050 FORMAT(" ",37X,"MAX Y-ERROR = ",E15.6,2X,"AT LOC. J,K = ",I5,      CORD1   495
     1       ",",I5)                                                     CORD1   496
 1060 FORMAT(" ",37X,"MAX Z-ERROR = ",E15.6,2X,"AT LOC. J,K = ",I5,      CORD1   497
     1       ",",I5)                                                     CORD1   498
 1070 FORMAT(/1X, "J = ",I5,"    (Y-Z ARRAY)")                           CORD1   499
 1080 FORMAT(1X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5,      CORD1   500
     1      2X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5)                         CORD1   501
 1200 FORMAT(/1X," J=1,K=1,X=1,IMAX  (X,Y,Z ARRAY)")                     CORD1   502
 1210 FORMAT(1X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5,2X,E11.5,2X,            CORD1   503
     1           E11.5,2X,E11.5,2X,E11.5,2X,E11.5)                       CORD1   504
C**                                                                      CORD1   505
C**            ALL  FUNCTIONS COMPLETE                                   CORD1   506
C**                                                                      CORD1   507
      RETURN                                                            CORD1   508
      END                                                               CORD1   509
      SUBROUTINE IPLTC(ISOLN,KKK,YERRMX,ZERRMX,IYLOCJ,IYLOCK,           CORD1   510
     1      IZLOCJ,IZLOCK)                                              CORD1   511
C**                                                                      CORD1   512
C**       THIS SUBROUTINE IS USED TO PLOT INITIAL, PARTIALLY CONVERGED   CORD1   513
C**       OR CONVERGED SOLN. ISOLN = 1 INDICATES INITIAL GUESS           CORD1   514
C**       ISOLN = 2 AND KKK = 0 INDICATES PARTIALLY CONVERGED SOLN.      CORD1   515
C**       ISOLN = 2 AND KKK = 1 INDICATES CONVERGED SOLN.                CORD1   516
C**                                                                      CORD1   517
C**          GLOBAL COMMON STATEMENTS                                    CORD1   518
C**                                                                      CORD1   519
      COMMON /BXXX/ Y(JMXLL,KMXLL),Z(JMXLL,KMXLL),Q(JMXLL),R(KMXLL),    CORD1   520
     1    X(IMXLL),Y1(IMXLL),Z1(IMXLL)                                  CORD1   521
      COMMON /AXXX/ ILIMIT,JLIMIT,KLIMIT,IMAX,JMAX,KMAX,IPRT1,IPRT2,    CORD1   522
     1       IPLT1,IPLT2,ITERMX,R1,R2,R3,I1,HED1(4),HED2(4)             CORD1   523
C**                                                                      CORD1   524
C**         LOCAL  DIMENSION STATEMENTS                                  CORD1   525
C**                                                                      CORD1   526
      DIMENSION YDUM(KMXLL),YDUM1(JMXLL),ZDUM(KMXLL),ZDUM1(JMXLL),      CORD1   527
     1     HED(4)                                                       CORD1   528
```

```
C**
C**                    WRITE HEADINGS. TWO CARDS ARE READ IN THE
C**           MAIN PROGRAM WITH 40 COLUMNS INFORMATION ON EACH CARD
C**            WHENEVER IPLT1 OR IPLT2 = 1.
C**
      CALL SCRIBE(0.3,1.5,0.15,0.5,HED1,90.,40,9)
      CALL SCRIBE(0.55,1.5,0.15,0.5,HED2,90.,40,9)
C**
C**         PRINT HEADINGS FIRST
C**
      IF(ISOLN.EQ.2) GO TO 50
          HED(1) = 10H--- INITIA
          HED(2) = 10HL GUESS --
          HED(3) = 1H-
          CALL SCRIBE(0.70,3.0,0.1,0.5,HED,90.0,21,9)
          GO TO 100
C**
C**             PARTIALLY CONVERGED OR CONVERGED SOLN
C**
   50 CONTINUE
          IF(KKK.EQ.1) GO TO 75
            HED(1) = 10H--- PARTIA
            HED(2) = 10HLLY CONVER
            HED(3) = 10HGED SOLUTI
            HED(4) =  7HON ---
            CALL SCRIBE(0.70,3.0,0.1,0.5,HED,90.0,37,9)
            GO TO 100
   75     CONTINUE
          HED(1) = 10H--- CONVER
          HED(2) = 10HGED SOLUTI
          HED(3) =   6HON ---
          CALL SCRIBE(0.70,3.0,0.1,0.5,HED,90.0,26,9)
  100 CONTINUE
C**
C**       WRITE ARRAY SIZES
C**
      ENCODE(40,1000,HED) JMAX,KMAX
      CALL SCRIBE(0.85,3.0,0.1,0.5,HED,90.0,27,9)
C**
C**          CALL NFRAME  TO ADVANCE FRAME AFTER WRITING HEADINGS
C**
      CALL NFRAME
C**
C**       FIND MIN MAX VALUE FOR PLOTTING
```

```
CORD1  529
CORD1  530
CORD1  531
CORD1  532
CORD1  533
CORD1  534
CORD1  535
CORD1  536
CORD1  537
CORD1  538
CORD1  539
CORD1  540
CORD1  541
CORD1  542
CORD1  543
CORD1  544
CORD1  545
CORD1  546
CORD1  547
CORD1  548
CORD1  549
CORD1  550
CORD1  551
CORD1  552
CORD1  553
CORD1  554
CORD1  555
CORD1  556
CORD1  557
CORD1  558
CORD1  559
CORD1  560
CORD1  561
CORD1  562
CORD1  563
CORD1  564
CORD1  565
CORD1  566
CORD1  567
CORD1  568
CORD1  569
CORD1  570
CORD1  571
CORD1  572
```

```
C**                                                                    CORD1      573
      YMIN =  10000.0                                                  CORD1      574
      ZMIN = 10000.0                                                   CORD1      575
      YMAX = -10000.0                                                  CORD1      576
      ZMAX = -10000.0                                                  CORD1      577
      DO 150 J = 1,JMAX                                                CORD1      578
      DO 150 K = 1,KMAX                                                CORD1      579
          IF(Y(J,K).LT.YMIN)  YMIN = Y(J,K)                            CORD1      580
          IF(Y(J,K).GT.YMAX)  YMAX = Y(J,K)                            CORD1      581
          IF(Z(J,K).LT.ZMIN) ZMIN = Z(J,K)                             CORD1      582
          IF(Z(J,K).GT.ZMAX) ZMAX = Z(J,K)                             CORD1      583
  150 CONTINUE                                                         CORD1      584
C**                                                                    CORD1      585
C**       PLOT DATA. ZETA LINES FIRST                                  CORD1      586
C**                                                                    CORD1      587
      DO 210 J = 1,JMAX                                                CORD1      588
         DO 200 K = 1,KMAX                                             CORD1      589
            YDUM(K)  = Y(J,K)                                          CORD1      590
            ZDUM(K)  = Z(J,K)                                          CORD1      591
  200    CONTINUE                                                      CORD1      592
         CALL INFOPLT(0,KMAX,YDUM,1,ZDUM,1,YMIN,YMAX,ZMIN,ZMAX,        CORD1      593
     1      1.0,0,HED,0,HED,0,5.0,5.0,0.0,2.0)                         CORD1      594
  210 CONTINUE                                                         CORD1      595
C**                                                                    CORD1      596
C**       PLOT ETA LINES                                               CORD1      597
C**                                                                    CORD1      598
      KK = 0                                                           CORD1      599
      DO 250 K = 1,KMAX                                                CORD1      600
          IF(K.EQ.KMAX) KK = 1                                         CORD1      601
         DO 240 J = 1,JMAX                                             CORD1      602
            YDUM1(J) = Y(J,K)                                          CORD1      603
            ZDUM1(J) = Z(J,K)                                          CORD1      604
  240     CONTINUE                                                     CORD1      605
          CALL INFOPLT(KK,JMAX,YDUM1,1,ZDUM1,1,YMIN,YMAX,ZMIN,ZMAX,    CORD1      606
     1      1.0,0,HED,0,HED,0,5.0,5.0,0.0,2.0)                         CORD1      607
  250 CONTINUE                                                         CORD1      608
C**                                                                    CORD1      609
      RETURN                                                           CORD1      610
C**                                                                    CORD1      611
 1000 FORMAT(7HJMAX = ,I5,10H   KMAX = ,I5)                            CORD1      612
C**                                                                    CORD1      613
      END                                                              CORD1      614
```

B. Job Control Cards

A list of job control cards needed to run this program are included in the comment section of the main program COORDC.

C. Input Explanation

The instructions for input preparation are included in the comment section of the main program COORDC.

D. Sample Input

Sample input listing follows on the next page.

E. Sample Output

The sample output generated by the sample input of section D follows sample input.

PROGRAM NO._____

CODED BY_____

DIVISION_____SECTION_____

LANGLEY RESEARCH CENTER

FORTRAN - DATA CODING FORM

DATE____

PAGE____

JOB ORDER_____

D. Sample Input

STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT
		23 23 1 1 1 1 300 0.6 0.00001 0.
0.03	0	1138 0.030138
0.03	3	67334 0.026722 8
0.03	8	73217 0.022416 7
0.04	6	4867 0.016804 8
0.05	7	8195 0.010370 4
0.07	3	3349 0.004133 3
0.09	3	3786 0.004339
0.11	8	072
0.14	7	336
0.18	0	917
0.21	8	419
0.25	9	325
0.30	3	027
0.34	8	856
0.39	6	102
0.44	4	050
0.49	2	001
0.53	9	304
0.58	5	380
0.62	9	744
0.67	2	044
0.71	2	094
0.75	0	000
0.03	0	1138 0.030138

NASA-Langley Form 67 (MAR 69)

NOTE: WRITE NUMBERS 10, LETTERS I Ø U G Z C, SYMBOLS / . , *

*U.S. GOVERNMENT

PROGRAM NO._____

CODED BY_____

DIVISION_____SECTION_____

LANGLEY RESEARCH CENTER

FORTRAN - DATA CODING FORM

DATE_____

PAGE_____

JOB ORDER_____

STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT
1 2 3 4 5	6	7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64
		0.026722 8 0.0336734
		0.022416 7 0.0387321
		0.016804 8 0.0464867
		0.010370 4 0.0578195
		0.004133 3 0.0733349
		0.000433 9 0.0933786
		0.1180720
		0.1473360
		0.1809170
		0.2184190
		0.2593250
		0.3030270
		0.3488560
		0.3961020
		0.4440500
		0.4920010
		0.5393040
		0.5853800
		0.6297440
		0.6720440
		0.7120940
		0.7500000
30		
	0.0	0.0301138 0.0301138
0.0068190		0.0301138 0.0301138

31

NASA-Langley Form 67 (MAR 69)

*US GOVERN

NOTE: WRITE NUMBERS 10, LETTERS I ∅ U G ≠ C, SYMBOLS / . , *

PROGRAM NO._____

CODED BY_____

DIVISION_____SECTION_____

LANGLEY RESEARCH CENTER

FORTRAN - DATA CODING FORM

DATE____

PAGE____

JOB ORDER_____

STATEMENT NUMBER	↓ CONTINUATION	FORTRAN STATEMENT
1 2 3 4 5	6	7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65
0.0	1	3640 0.03 01138 0.03 01138
0.0	2	0488 0.03 01138 0.03 01138
0.0	2	7377 0.03 01138 0.03 01138
0.0	3	4466 0.03 01138 0.03 01138
0.0	4	2122 0.03 01138 0.03 01138
0.0	5	1199 0.03 01138 0.03 01138
0.0	6	3233 0.03 01138 0.03 01138
0.0	8	0522 0.03 01138 0.03 01138
0.1	0	5800 0.03 01138 0.03 01138
0.1	4	1200 0.03 01138 0.03 01138
0.1	7	6600 0.03 01138 0.03 01138
0.2	1	2000 0.03 01138 0.03 01138
0.2	4	7400 0.03 01138 0.03 01138
0.2	8	2800 0.03 01138 0.03 01138
0.3	1	8200 0.03 01138 0.03 01138
0.3	5	3600 0.03 01138 0.03 01138
0.3	8	9000 0.03 01138 0.03 01138
0.4	2	4400 0.03 01138 0.03 01138
0.4	5	9800 0.03 01138 0.03 01138
0.4	9	5200 0.03 01138 0.03 01138
0.5	3	0600 0.03 01138 0.03 01138
0.5	6	6000 0.03 01138 0.03 01138
0.6	0	1400 0.03 01138 0.03 01138
0.6	3	6800 0.03 01138 0.03 01138
0.6	7	2200 0.03 01138 0.03 01138

NOTE: WRITE NUMBERS 10, LETTERS I Ø U G Z C, SYMBOLS / . , *

PROGRAM NO._____

CODED BY_____

DIVISION_____SECTION_____

LANGLEY RESEARCH CENTER

FORTRAN - DATA CODING FORM

DATE_____

PAGE_____

JOB ORDER_____

STATEMENT NUMBER	↓ CONTINUATION	FORTRAN STATEMENT
		1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64
		0.7076 0.03 0113 8 0.03 0113 8
		0.7430 0.03 0113 8 0.03 0113 8
		0.7840 0.03 0113 8 0.03 0113 8
		CØRNER FLØW WITH FILLET
		CØØRDINATE SYSTEM. SIZE= 30X23X23

*US GOVERN

NASA-Langley Form 67 (MAR 69)

NOTE: WRITE NUMBERS 10, LETTERS I Ø U G Z C, SYMBOLS / . , *

E. SAMPLE OUTPUT

*** INITIAL SOLUTION ***
CORNER FLOW WITH FILLET
COORDINATE SYSTEM. SIZE=30X23X23

JMAX = 23 KMAX = 23

J = 1 (Y-Z ARRAY)

.30114E-01	.30114E-01	.26723E-01	.33673E-01	.22417E-01	.38732E-01	.16805E-01	.46487E-01	.10370E-01	.57820E-01
.41333E-02	.73335E-01	.43390E-03	.93379E-01	0.	.11807E+00	0.	.14734E+00	0.	.18092E+00
0.	.21842E+00	0.	.25933E+00	0.	.30303E+00	0.	.34886E+00	0.	.39610E+00
0.	.44405E+00	0.	.49200E+00	0.	.53930E+00	0.	.58538E+00	0.	.62974E+00
	.67204E+00	0.	.71209E+00	0.	.75000E+00				

J = 2 (Y-Z ARRAY)

.33673E-01	.26723E-01	.30299E-01	.30299E-01	.26014E-01	.35382E-01	.20430E-01	.43173E-01	.14028E-01	.54559E-01
.78214E-02	.70147E-01	.41403E-02	.90286E-01	.37085E-02	.11510E+00	.37085E-02	.14450E+00	.37085E-02	.17824E+00
.37085E-02	.21592E+00	.37085E-02	.25701E+00	.37085E-02	.30092E+00	.37085E-02	.34697E+00	.37085E-02	.39443E+00
.37085E-02	.44261E+00	.37085E-02	.49079E+00	.37085E-02	.53831E+00	.37085E-02	.58460E+00	.37085E-02	.62918E+00
.37085E-02	.67168E+00	.37085E-02	.71192E+00	.37085E-02	.75000E+00				

J = 3 (Y-Z ARRAY)

.38732E-01	.22417E-01	.35382E-01	.26014E-01	.31127E-01	.31127E-01	.25582E-01	.38965E-01	.19225E-01	.50419E-01
.13063E-01	.66100E-01	.94075E-02	.86358E-01	.89788E-02	.11132E+00	.89788E-02	.14089E+00	.89788E-02	.17483E+00
.89788E-02	.21274E+00	.89788E-02	.25408E+00	.89788E-02	.29825E+00	.89788E-02	.34457E+00	.89788E-02	.39232E+00
.89788E-02	.44078E+00	.89788E-02	.48924E+00	.89788E-02	.53705E+00	.89788E-02	.58362E+00	.89788E-02	.62846E+00
.89788E-02	.67121E+00	.89788E-02	.71169E+00	.89788E-02	.75000E+00				

J = 4 (Y-Z ARRAY)

.46487E-01	.16805E-01	.43173E-01	.20430E-01	.38965E-01	.25582E-01	.33480E-01	.33480E-01	.27192E-01	.45023E-01
.21097E-01	.60825E-01	.17482E-01	.81239E-01	.17058E-01	.10639E+00	.17058E-01	.13619E+00	.17058E-01	.17040E+00
.17058E-01	.20859E+00	.17058E-01	.25025E+00	.17058E-01	.29476E+00	.17058E-01	.34144E+00	.17058E-01	.38956E+00
.17058E-01	.43839E+00	.17058E-01	.48723E+00	.17058E-01	.53541E+00	.17058E-01	.58234E+00	.17058E-01	.62752E+00
.17058E-01	.67060E+00	.17058E-01	.71139E+00	.17058E-01	.75000E+00				

J = 5 (Y-Z ARRAY)

.57820E-01	.10370E-01	.54559E-01	.14028E-01	.50419E-01	.19225E-01	.45023E-01	.27192E-01	.38836E-01	.38836E-01
.32839E-01	.54777E-01	.29282E-01	.75370E-01	.28865E-01	.10074E+00	.28865E-01	.13081E+00	.28865E-01	.16531E+00
.28865E-01	.20384E+00	.28865E-01	.24587E+00	.28865E-01	.29077E+00	.28865E-01	.33785E+00	.28865E-01	.38640E+00
.28865E-01	.43566E+00	.28865E-01	.48493E+00	.28865E-01	.53353E+00	.28865E-01	.58087E+00	.28865E-01	.62645E+00
.28865E-01	.66991E+00	.28865E-01	.71105E+00	.28865E-01	.75000E+00				

J = 6 (Y-Z ARRAY)

.73335E-01	.41333E-02	.70147E-01	.78214E-02	.66100E-01	.13063E-01	.60825E-01	.21097E-01	.54777E-01	.32839E-01
.48914E-01	.48914E-01	.45437E-01	.69681E-01	.45029E-01	.95266E-01	.45029E-01	.12559E+00	.45029E-01	.16038E+00
.45029E-01	.19923E+00	.45029E-01	.24162E+00	.45029E-01	.28690E+00	.45029E-01	.33430E+00	.45029E-01	.38333E+00
.45029E-01	.43301E+00	.45029E-01	.48269E+00	.45029E-01	.53170E+00	.45029E-01	.57944E+00	.45029E-01	.62540E+00
.45029E-01	.66923E+00	.45029E-01	.71073E+00	.45029E-01	.75000E+00				

J = 7 (Y-Z ARRAY)

.93379E-01	.43390E-03	.90286E-01	.41403E-02	.86358E-01	.94075E-02	.81239E-01	.17482E-01	.75370E-01	.29282E-01
.69681E-01	.45437E-01	.66307E-01	.66307E-01	.65911E-01	.92018E-01	.65911E-01	.12249E+00	.65911E-01	.15745E+00
.65911E-01	.19650E+00	.65911E-01	.23910E+00	.65911E-01	.28460E+00	.65911E-01	.33232E+00	.65911E-01	.38151E+00
.65911E-01	.43144E+00	.65911E-01	.48136E+00	.65911E-01	.53062E+00	.65911E-01	.57859E+00	.65911E-01	.62479E+00
.65911E-01	.66883E+00	.65911E-01	.71053E+00	.65911E-01	.75000E+00				

J = 8 (Y-Z ARRAY)

.11807E+00	0.	.11510E+00	.37085E-02	.11132E+00	.89788E-02	.10639E+00	.17058E-01	.10074E+00	.28865E-01
.95266E-01	.45029E-01	.92018E-01	.65911E-01	.91638E-01	.91638E-01	.91638E-01	.12213E+00	.91638E-01	.15711E+00
.91636E-01	.19618E+00	.91638E-01	.23880E+00	.91638E-01	.28433E+00	.91638E-01	.33208E+00	.91638E-01	.38130E+00
.91638E-01	.43125E+00	.91638E-01	.48121E+00	.91638E-01	.53049E+00	.91638E-01	.57849E+00	.91638E-01	.62471E+00
.91638E-01	.66878E+00	.91638E-01	.71051E+00	.91638E-01	.75000E+00				

J = 9 (Y-Z ARRAY)

.14734E+00	0.	.14450E+00	.37085E-02	.14089E+00	.89788E-02	.13619E+00	.17058E-01	.13081E+00	.28865E-01
.12559E+00	.45029E-01	.12249E+00	.65911E-01	.12213E+00	.91638E-01	.12213E+00	.12213E+00	.12213E+00	.15711E+00
.12213E+00	.19618E+00	.12213E+00	.23880E+00	.12213E+00	.28433E+00	.12213E+00	.33208E+00	.12213E+00	.38130E+00
.12213E+00	.43125E+00	.12213E+00	.48121E+00	.12213E+00	.53049E+00	.12213E+00	.57849E+00	.12213E+00	.62471E+00
.12213E+00	.66878E+00	.12213E+00	.71051E+00	.12213E+00	.75000E+00				

J = 10 (Y-Z ARRAY)

.18092E+00	0.	.17824E+00	.37085E-02	.17483E+00	.89788E-02	.17040E+00	.17058E-01	.16531E+00	.28865E-01
.16038E+00	.45029E-01	.15745E+00	.65911E-01	.15711E+00	.91638E-01	.15711E+00	.12213E+00	.15711E+00	.15711E+00
.15711E+00	.19618E+00	.15711E+00	.23880E+00	.15711E+00	.28433E+00	.15711E+00	.33208E+00	.15711E+00	.38130E+00
.15711E+00	.43125E+00	.15711E+00	.48121E+00	.15711E+00	.53049E+00	.15711E+00	.57849E+00	.15711E+00	.62471E+00
.15711E+00	.66878E+00	.15711E+00	.71051E+00	.15711E+00	.75000E+00				

J = 11 (Y-Z ARRAY)

.21842E+00	0.	.21592E+00	.37085E-02	.21274E+00	.89788E-02	.20859E+00	.17058E-01	.20384E+00	.28865E-01
.19923E+00	.45029E-01	.19650E+00	.65911E-01	.19618E+00	.91638E-01	.19618E+00	.12213E+00	.19618E+00	.15711E+00
.19618E+00	.19618E+00	.19618E+00	.23880E+00	.19618E+00	.28433E+00	.19618E+00	.33208E+00	.19618E+00	.38130E+00
.19618E+00	.43125E+00	.19618E+00	.48121E+00	.19618E+00	.53049E+00	.19618E+00	.57849E+00	.19618E+00	.62471E+00
.19618E+00	.66878E+00	.19618E+00	.71051E+00	.19618E+00	.75000E+00				

J = 12 (Y-Z ARRAY)

.25933E+00	0.	.25701E+00	.37085E-02	.25408E+00	.89788E-02	.25025E+00	.17058E-01	.24587E+00	.28865E-01
.24162E+00	.45029E-01	.23910E+00	.65911E-01	.23880E+00	.91638E-01	.23880E+00	.12213E+00	.23880E+00	.15711E+00
.23880E+00	.19618E+00	.23880E+00	.23880E+00	.23880E+00	.28433E+00	.23880E+00	.33208E+00	.23880E+00	.38130E+00
.23880E+00	.43125E+00	.23880E+00	.48121E+00	.23880E+00	.53049E+00	.23880E+00	.57849E+00	.23880E+00	.62471E+00
.23880E+00	.66878E+00	.23880E+00	.71051E+00	.23880E+00	.75000E+00				

J = 13 (Y-Z ARRAY)

.30303E+00	0.	.30092E+00	.37085E-02	.29825E+00	.89788E-02	.29476E+00	.17058E-01	.29077E+00	.28865E-01
.28690E+00	.45029E-01	.28460E+00	.65911E-01	.28433E+00	.91638E-01	.28433E+00	.12213E+00	.28433E+00	.15711E+00
.28433E+00	.19618E+00	.28433E+00	.23880E+00	.28433E+00	.28433E+00	.28433E+00	.33208E+00	.28433E+00	.38130E+00
.28433E+00	.43125E+00	.28433E+00	.48121E+00	.28433E+00	.53049E+00	.28433E+00	.57849E+00	.28433E+00	.62471E+00
.28433E+00	.66878E+00	.28433E+00	.71051E+00	.28433E+00	.75000E+00				

J = 14 (Y-Z ARRAY)

.34886E+00	0.	.34697E+00	.37085E-02	.34457E+00	.89788E-02	.34144E+00	.17058E-01	.33785E+00	.28865E-01
.33436E+00	.45029E-01	.33232E+00	.65911E-01	.33208E+00	.91638E-01	.33208E+00	.12213E+00	.33208E+00	.15711E+00
.33208E+00	.19618E+00	.33208E+00	.23880E+00	.33208E+00	.28433E+00	.33208E+00	.33208E+00	.33208E+00	.38130E+00
.33208E+00	.43125E+00	.33208E+00	.48121E+00	.33208E+00	.53049E+00	.33208E+00	.57849E+00	.33208E+00	.62471E+00
.33208E+00	.66878E+00	.33208E+00	.71051E+00	.33208E+00	.75000E+00				

J = 15 (Y-Z ARRAY)

.39610E+00	0.	.39443E+00	.37085E-02	.39232E+00	.89788E-02	.38956E+00	.17058E-01	.38640E+00	.28865E-01
.38333E+00	.45029E-01	.38151E+00	.65911E-01	.38130E+00	.91638E-01	.38130E+00	.12213E+00	.38130E+00	.15711E+00
.38130E+00	.19618E+00	.38130E+00	.23880E+00	.38130E+00	.28433E+00	.38130E+00	.33208E+00	.38130E+00	.38130E+00
.38130E+00	.43125E+00	.38130E+00	.48121E+00	.38130E+00	.53049E+00	.38130E+00	.57849E+00	.38130E+00	.62471E+00
.38130E+00	.66878E+00	.38130E+00	.71051E+00	.38130E+00	.75000E+00				

J = 16 (Y-Z ARRAY)

.44405E+00	0.	.44261E+00	.37085E-02	.44078E+00	.89788E-02	.43839E+00	.17058E-01	.43566E+00	.28865E-01
.43301E+00	.45029E-01	.43144E+00	.65911E-01	.43125E+00	.91638E-01	.43125E+00	.12213E+00	.43125E+00	.15711E+00
.43125E+00	.19618E+00	.43125E+00	.23880E+00	.43125E+00	.28433E+00	.43125E+00	.33208E+00	.43125E+00	.38130E+00
.43125E+00	.43125E+00	.43125E+00	.48121E+00	.43125E+00	.53049E+00	.43125E+00	.57849E+00	.43125E+00	.62471E+00
.43125E+00	.66878E+00	.43125E+00	.71051E+00	.43125E+00	.75000E+00				

J = 17 (Y-Z ARRAY)

.49200E+00	0.	.49079E+00	.37085E-02	.48924E+00	.89788E-02	.48723E+00	.17058E-01	.48493E+00	.28865E-01
.48269E+00	.45029E-01	.48136E+00	.65911E-01	.48121E+00	.91638E-01	.48121E+00	.12213E+00	.48121E+00	.15711E+00
.48121E+00	.19618E+00	.48121E+00	.23880E+00	.48121E+00	.28433E+00	.48121E+00	.33208E+00	.48121E+00	.38130E+00
.48121E+00	.43125E+00	.48121E+00	.48121E+00	.48121E+00	.53049E+00	.48121E+00	.57849E+00	.48121E+00	.62471E+00
.48121E+00	.66878E+00	.48121E+00	.71051E+00	.48121E+00	.75000E+00				

J = 18 (Y-Z ARRAY)

.53930E+00	0.	.53831E+00	.37085E-02	.53705E+00	.89788E-02	.53541E+00	.17058E-01	.53353E+00	.28865E-01
.53170E+00	.45029E-01	.53062E+00	.65911E-01	.53049E+00	.91638E-01	.53049E+00	.12213E+00	.53049E+00	.15711E+00
.53049E+00	.19618E+00	.53049E+00	.23880E+00	.53049E+00	.28433E+00	.53049E+00	.33208E+00	.53049E+00	.38130E+00
.53049E+00	.43125E+00	.53049E+00	.48121E+00	.53049E+00	.53049E+00	.53049E+00	.57849E+00	.53049E+00	.62471E+00
.53049E+00	.66878E+00	.53049E+00	.71051E+00	.53049E+00	.75000E+00				

J = 19 (Y-Z ARRAY)

.58536E+00	0.	.58460E+00	.37085E-02	.58362E+00	.89788E-02	.58234E+00	.17058E-01	.58087E+00	.28865E-01
.57944E+00	.45029E-01	.57859E+00	.65911E-01	.57849E+00	.91638E-01	.57849E+00	.12213E+00	.57849E+00	.15711E+00
.57849E+00	.19618E+00	.57849E+00	.23880E+00	.57849E+00	.28433E+00	.57849E+00	.33208E+00	.57849E+00	.38130E+00
.57849E+00	.43125E+00	.57849E+00	.48121E+00	.57849E+00	.53049E+00	.57849E+00	.57849E+00	.57849E+00	.62471E+00
.57849E+00	.66878E+00	.57849E+00	.71051E+00	.57849E+00	.75000E+00				

J = 20 (Y-Z ARRAY)

.62974E+00	0.	.62918E+00	.37085E-02	.62846E+00	.89788E-02	.62752E+00	.17058E-01	.62645E+00	.28865E-01
.62540E+00	.45029E-01	.62479E+00	.65911E-01	.62471E+00	.91638E-01	.62471E+00	.12213E+00	.62471E+00	.15711E+00
.62471E+00	.19618E+00	.62471E+00	.23880E+00	.62471E+00	.28433E+00	.62471E+00	.33208E+00.	.62471E+00	.38130E+00
.62471E+00	.43125E+00	.62471E+00	.48121E+00	.62471E+00	.53049E+00	.62471E+00	.57849E+00	.62471E+00	.62471E+00
.62471E+00	.66878E+00	.62471E+00	.71051E+00	.62471E+00	.75000E+00				

```
J =    21    (Y-Z ARRAY)
 .67204E+00  0.           .67168E+00  .37085E-02  .67121E+00  .89788E-02  .67060E+00  .17058E-01  .66991E+00  .28865E-01
 .66923E+00   .45029E-01  .66883E+00  .65911E-01  .66878E+00  .91638E-01  .66878E+00  .12213E+00  .66878E+00  .15711E+00
 .66878E+00   .19618E+00  .66878E+00  .23880E+00  .66878E+00  .28433E+00  .66878E+00  .33208E+00  .66878E+00  .38130E+00
 .66878E+00   .43125E+00  .66878E+00  .48121E+00  .66878E+00  .53049E+00  .66878E+00  .57849E+00  .66878E+00  .62471E+00
 .66878E+00   .66878E+00  .66878E+00  .71051E+00  .66878E+00  .75000E+00

J =    22    (Y-Z ARRAY)
 .71209E+00  0.           .71192E+00  .37085E-02  .71169E+00  .89788E-02  .71139E+00  .17058E-01  .71105E+00  .28865E-01
 .71073E+00   .45029E-01  .71053E+00  .65911E-01  .71051E+00  .91638E-01  .71051E+00  .12213E+00  .71051E+00  .15711E+00
 .71051E+00   .19618E+00  .71051E+00  .23880E+00  .71051E+00  .28433E+00  .71051E+00  .33208E+00  .71051E+00  .38130E+00
 .71051E+00   .43125E+00  .71051E+00  .48121E+00  .71051E+00  .53049E+00  .71051E+00  .57849E+00  .71051E+00  .62471E+00
 .71051E+00   .66878E+00  .71051E+00  .71051E+00  .71051E+00  .75000E+00

J =    23    (Y-Z ARRAY)
 .75000E+00  0.           .75000E+00  .37085E-02  .75000E+00  .89788E-02  .75000E+00  .17058E-01  .75000E+00  .28865E-01
 .75000E+00   .45029E-01  .75000E+00  .65911E-01  .75000E+00  .91638E-01  .75000E+00  .12213E+00  .75000E+00  .15711E+00
 .75000E+00   .19618E+00  .75000E+00  .23880E+00  .75000E+00  .28433E+00  .75000E+00  .33208E+00  .75000E+00  .38130E+00
 .75000E+00   .43125E+00  .75000E+00  .48121E+00  .75000E+00  .53049E+00  .75000E+00  .57849E+00  .75000E+00  .62471E+00
 .75000E+00   .66878E+00  .75000E+00  .71051E+00  .75000E+00  .75000E+00

 J=1,K=1,X=1,IMAX  (X,Y,Z ARRAY)
0.           .30114E-01  .30114E-01  .68190E-02  .30114E-01  .30114E-01  .13640E-01  .30114E-01  .30114E-01
 .20480E-01  .30114E-01  .30114E-01  .27370E-01  .30114E-01  .30114E-01  .34460E-01  .30114E-01  .30114E-01
 .42120E-01  .30114E-01  .30114E-01  .51190E-01  .30114E-01  .30114E-01  .63230E-01  .30114E-01  .30114E-01
 .80520E-01  .30114E-01  .30114E-01  .10560E+00  .30114E-01  .30114E-01  .14120E+00  .30114E-01  .30114E-01
 .17660E+00  .30114E-01  .30114E-01  .21200E+00  .30114E-01  .30114E-01  .24740E+00  .30114E-01  .30114E-01
 .28280E+00  .30114E-01  .30114E-01  .31820E+00  .30114E-01  .30114E-01  .35360E+00  .30114E-01  .30114E-01
 .38900E+00  .30114E-01  .30114E-01  .42440E+00  .30114E-01  .30114E-01  .45980E+00  .30114E-01  .30114E-01
 .49520E+00  .30114E-01  .30114E-01  .53060E+00  .30114E-01  .30114E-01  .56600E+00  .30114E-01  .30114E-01
 .60140E+00  .30114E-01  .30114E-01  .63680E+00  .30114E-01  .30114E-01  .67220E+00  .30114E-01  .30114E-01
 .70760E+00  .30114E-01  .30114E-01  .74300E+00  .30114E-01  .30114E-01  .77840E+00  .30114E-01  .30114E-01

    ITER NO.,IYLOCJ,IYLOCK,YERRMX,IZLOCJ,IZLOCK,ZERRMX =    50   13    9   .716194E-04    9   13   .715067E-04
```

*** PARTIALLY CONVERGED SOLUTION ***
CORNER FLOW WITH FILLET
COORDINATE SYSTEM. SIZE=30X23X23

JMAX = 23 KMAX = 23
MAX ITERATIONS PERFORMED = 50
MAX Y-ERROR = .716194E-04 AT LOC. J,K = 13, 9
MAX Z-ERROR = .715067E-04 AT LOC. J,K = 9, 13

J = 1 (Y-Z ARRAY)

.30114E-01	.30114E-01	.26723E-01	.33673E-01	.22417E-01	.38732E-01	.16805E-01	.46487E-01	.10370E-01	.57820E-01
.41333E-02	.73335E-01	.43390E-03	.93379E-01	0.	.11807E+00	0.	.14734E+00	0.	.18092E+00
0.	.21842E+00	0.	.25933E+00	0.	.30303E+00	0.	.34886E+00	0.	.39610E+00
0.	.44405E+00	0.	.49200E+00	0.	.53930E+00	0.	.58538E+00	0.	.62974E+00
0.	.67204E+00	0.	.71209E+00	0.	.75000E+00				

J = 2 (Y-Z ARRAY)

.33673E-01	.26723E-01	.31702E-01	.31701E-01	.27769E-01	.37616E-01	.21884E-01	.45740E-01	.14952E-01	.57186E-01
.83199E-02	.72725E-01	.42756E-02	.92742E-01	.36043E-02	.11739E+00	.35675E-02	.14666E+00	.35631E-02	.18028E+00
.35618E-02	.21764E+00	.35612E-02	.25881E+00	.35609E-02	.30256E+00	.35608E-02	.34845E+00	.35607E-02	.39575E+00
.35607E-02	.44375E+00	.35607E-02	.49175E+00	.35607E-02	.53911E+00	.35607E-02	.58523E+00	.35607E-02	.62964E+00
.35607E-02	.67198E+00	.35607E-02	.71206E+00	.35607E-02	.75000E+00				

J = 3 (Y-Z ARRAY)

.38732E-01	.22417E-01	.37617E-01	.27767E-01	.34763E-01	.34759E-01	.29291E-01	.43940E-01	.21859E-01	.55983E-01
.14507E-01	.71740E-01	.98591E-02	.91793E-01	.87786E-02	.11642E+00	.86534E-02	.14572E+00	.86337E-02	.17941E+00
.86275E-02	.21705E+00	.86249E-02	.25809E+00	.86237E-02	.30193E+00	.86231E-02	.34790E+00	.86228E-02	.39527E+00
.86227E-02	.44335E+00	.86227E-02	.49142E+00	.86227E-02	.53884E+00	.86227E-02	.58503E+00	.86228E-02	.62950E+00
.86228E-02	.67189E+00	.86228E-02	.71202E+00	.86228E-02	.75000E+00				

J = 4 (Y-Z ARRAY)

.46487E-01	.16805E-01	.45741E-01	.21882E-01	.43945E-01	.29285E-01	.39751E-01	.39741E-01	.32753E-01	.53135E-01
.24717E-01	.69736E-01	.18930E-01	.90143E-01	.16948E-01	.11490E+00	.16537E-01	.14430E+00	.16445E-01	.17810E+00
.16415E-01	.21587E+00	.16402E-01	.25705E+00	.16396E-01	.30101E+00	.16394E-01	.34709E+00	.16392E-01	.39458E+00
.16392E-01	.44276E+00	.16391E-01	.49094E+00	.16391E-01	.53846E+00	.16392E-01	.58474E+00	.16392E-01	.62929E+00
.16392E-01	.67177E+00	.16392E-01	.71197E+00	.16392E-01	.75000E+00				

J = 5 (Y-Z ARRAY)

.57820E-01	.10370E-01	.57187E-01	.14951E-01	.55987E-01	.21854E-01	.53145E-01	.32742E-01	.47573E-01	.47556E-01
.39878E-01	.65704E-01	.32993E-01	.87178E-01	.29490E-01	.11250E+00	.28340E-01	.14220E+00	.27993E-01	.17625E+00
.27870E-01	.21422E+00	.27818E-01	.25559E+00	.27794E-01	.29974E+00	.27783E-01	.34599E+00	.27777E-01	.39364E+00
.27774E-01	.44197E+00	.27774E-01	.49029E+00	.27773E-01	.53795E+00	.27774E-01	.58435E+00	.27774E-01	.62902E+00
.27775E-01	.67160E+00	.27775E-01	.71189E+00	.27775E-01	.75000E+00				

J = 6 (Y-Z ARRAY)

.73335E-01	.41333E-02	.72726E-01	.83196E-02	.71742E-01	.14506E-01	.69744E-01	.24710E-01	.65721E-01	.39862E-01
.59358E-01	.59335E-01	.52369E-01	.82352E-01	.47429E-01	.10885E+00	.45056E-01	.13926E+00	.44113E-01	.17374E+00

.43733E-01	.21205E+00	.43567E-01	.25370E+00	.43490E-01	.29810E+00	.43452E-01	.34458E+00	.43433E-01	.39243E+00
.43425E-01	.44097E+00	.43421E-01	.48947E+00	.43420E-01	.53730E+00	.43421E-01	.58387E+00	.43423E-01	.62869E+00
.43424E-01	.67139E+00	.43425E-01	.71179E+00	.43425E-01	.75000E+00				

J = 7 (Y-Z ARRAY)

.93379E-01	.43390E-03	.92742E-01	.42756E-02	.91794E-01	.98587E-02	.90148E-01	.18928E-01	.87190E-01	.32983E-01
.82373E-01	.52350E-01	.76281E-01	.76255E-01	.70797E-01	.10399E+00	.67247E-01	.13540E+00	.65414E-01	.17057E+00
.64547E-01	.20937E+00	.64137E-01	.25141E+00	.63936E-01	.29612E+00	.63836E-01	.34289E+00	.63787E-01	.39101E+00
.63763E-01	.43979E+00	.63753E-01	.48852E+00	.63750E-01	.53656E+00	.63751E-01	.58332E+00	.63755E-01	.62830E+00
.63758E-01	.67116E+00	.63760E-01	.71169E+00	.63760E-01	.75000E+00				

J = 8 (Y-Z ARRAY)

.11807E+00	0.	.11739E+00	.36043E-02	.11642E+00	.87785E-02	.11490E+00	.16947E-01	.11251E+00	.29486E-01
.10387E+00	.47416E-01	.10401E+00	.70775E-01	.98893E-01	.98866E-01	.94763E-01	.13103E+00	.92096E-01	.16692E+00
.90595E-01	.20630E+00	.89799E-01	.24879E+00	.89384E-01	.29390E+00	.89168E-01	.34100E+00	.89058E-01	.38942E+00
.89004E-01	.43848E+00	.88979E-01	.48746E+00	.88972E-01	.53574E+00	.88973E-01	.58271E+00	.88978E-01	.62788E+00
.88984E-01	.67090E+00	.88989E-01	.71157E+00	.88989E-01	.75000E+00				

J = 9 (Y-Z ARRAY)

.14734E+00	0.	.14666E+00	.35674E-02	.14572E+00	.86532E-02	.14430E+00	.16536E-01	.14221E+00	.28337E-01
.13927E+00	.45049E-01	.13542E+00	.67232E-01	.13105E+00	.94741E-01	.12698E+00	.12696E+00	.12386E+00	.16324E+00
.12179E+00	.20309E+00	.12056E+00	.24604E+00	.11986E+00	.29154E+00	.11947E+00	.33900E+00	.11926E+00	.38776E+00
.11916E+00	.43711E+00	.11911E+00	.48637E+00	.11909E+00	.53489E+00	.11909E+00	.58209E+00	.11910E+00	.62746E+00
.11911E+00	.67064E+00	.11911E+00	.71145E+00	.11911E+00	.75000E+00				

J = 10 (Y-Z ARRAY)

.18092E+00	0.	.18028E+00	.35631E-02	.17941E+00	.86336E-02	.17810E+00	.16444E-01	.17624E+00	.27991E-01
.17374E+00	.44107E-01	.17058E+00	.65404E-01	.16694E+00	.92079E-01	.16327E+00	.12384E+00	.16009E+00	.16007E+00
.15771E+00	.20012E+00	.15611E+00	.24339E+00	.15512E+00	.28924E+00	.15454E+00	.33704E+00	.15421E+00	.38611E+00
.15404E+00	.43576E+00	.15395E+00	.48530E+00	.15392E+00	.53407E+00	.15391E+00	.58148E+00	.15392E+00	.62704E+00
.15393E+00	.67039E+00	.15394E+00	.71134E+00	.15394E+00	.75000E+00				

J = 11 (Y-Z ARRAY)

.21842E+00	0.	.21784E+00	.35617E-02	.21704E+00	.86274E-02	.21587E+00	.16414E-01	.21422E+00	.27869E-01
.21205E+00	.43730E-01	.20938E+00	.64541E-01	.20631E+00	.90583E-01	.20311E+00	.12178E+00	.20014E+00	.15769E+00
.19771E+00	.19769E+00	.19592E+00	.24108E+00	.19471E+00	.28717E+00	.19396E+00	.33524E+00	.19351E+00	.38459E+00
.19326E+00	.43452E+00	.19313E+00	.48431E+00	.19307E+00	.53331E+00	.19306E+00	.58093E+00	.19307E+00	.62667E+00
.19306E+00	.67016E+00	.19309E+00	.71123E+00	.19309E+00	.75000E+00				

J = 12 (Y-Z ARRAY)

.25933E+00	0.	.25880E+00	.35612E-02	.25809E+00	.86249E-02	.25704E+00	.16402E-01	.25559E+00	.27817E-01
.25370E+00	.43566E-01	.25140E+00	.64133E-01	.24880E+00	.89791E-01	.24604E+00	.12055E+00	.24340E+00	.15610E+00
.24110E+00	.19590E+00	.23928E+00	.23927E+00	.23798E+00	.28546E+00	.23711E+00	.33372E+00	.23657E+00	.38329E+00
.23626E+00	.43344E+00	.23609E+00	.48345E+00	.23601E+00	.53265E+00	.23599E+00	.58045E+00	.23599E+00	.62634E+00
.23600E+00	.66997E+00	.23601E+00	.71114E+00	.23601E+00	.75000E+00				

J = 13 (Y-Z ARRAY)

.30303E+00	0.	.30256E+00	.35609E-02	.30193E+00	.86237E-02	.30101E+00	.16396E-01	.29973E+00	.27794E-01
.29809E+00	.43489E-01	.29612E+00	.63934E-01	.29389E+00	.89379E-01	.29154E+00	.11985E+00	[illegible]	[illegible]

.28717E+00	.19470E+00	.28547E+00	.23797E+00	.28418E+00	.28418E+00	.28328E+00	.33253E+00	.28270E+00	.38225E+00
.28234E+00	.43257E+00	.28214E+00	.48275E+00	.28205E+00	.53212E+00	.28201E+00	.58006E+00	.28201E+00	.62608E+00
.28202E+00	.66981E+00	.28202E+00	.71107E+00	.28202E+00	.75000E+00				

J = 14 (Y-Z ARRAY)

.34886E+00	0.	.34845E+00	.35608E-02	.34789E+00	.86231E-02	.34709E+00	.16394E-01	.34598E+00	.27782E-01
.34457E+00	.43451E-01	.34286E+00	.63835E-01	.34099E+00	.89166E-01	.33900E+00	.11947E+00	.33704E+00	.15453E+00
.33525E+00	.19395E+00	.33373E+00	.23711E+00	.33254E+00	.28328E+00	.33168E+00	.33167E+00	.33109E+00	.38148E+00
.33073E+00	.43191E+00	.33052E+00	.48222E+00	.33041E+00	.53171E+00	.33036E+00	.57976E+00	.33035E+00	.62588E+00
.33036E+00	.66969E+00	.33036E+00	.71101E+00	.33036E+00	.75000E+00				

J = 15 (Y-Z ARRAY)

.39610E+00	0.	.39575E+00	.35607E-02	.39527E+00	.86228E-02	.39458E+00	.16392E-01	.39363E+00	.27777E-01
.39243E+00	.43433E-01	.39101E+00	.63786E-01	.38942E+00	.89057E-01	.38775E+00	.11926E+00	.38611E+00	.15421E+00
.38460E+00	.19350E+00	.38329E+00	.23657E+00	.38225E+00	.28269E+00	.38148E+00	.33109E+00	.38095E+00	.38094E+00
.38060E+00	.43145E+00	.38039E+00	.48184E+00	.38028E+00	.53141E+00	.38023E+00	.57955E+00	.38021E+00	.62573E+00
.38021E+00	.66960E+00	.38021E+00	.71097E+00	.38021E+00	.75000E+00				

J = 16 (Y-Z ARRAY)

.44405E+00	0.	.44375E+00	.35607E-02	.44334E+00	.86227E-02	.44276E+00	.16392E-01	.44197E+00	.27775E-01
.44096E+00	.43425E-01	.43978E+00	.63763E-01	.43847E+00	.89003E-01	.43710E+00	.11916E+00	.43576E+00	.15404E+00
.43452E+00	.19326E+00	.43344E+00	.23626E+00	.43257E+00	.28234E+00	.43191E+00	.33073E+00	.43145E+00	.38060E+00
.43114E+00	.43114E+00	.43096E+00	.48158E+00	.43085E+00	.53121E+00	.43080E+00	.57940E+00	.43077E+00	.62563E+00
.43077E+00	.66954E+00	.43077E+00	.71095E+00	.43077E+00	.75000E+00				

J = 17 (Y-Z ARRAY)

.49200E+00	0.	.49175E+00	.35607E-02	.49142E+00	.86227E-02	.49094E+00	.16391E-01	.49029E+00	.27774E-01
.48947E+00	.43421E-01	.48851E+00	.63753E-01	.48746E+00	.88980E-01	.48637E+00	.11911E+00	.48529E+00	.15395E+00
.48431E+00	.19313E+00	.48345E+00	.23609E+00	.48275E+00	.28214E+00	.48222E+00	.33052E+00	.48184E+00	.38039E+00
.48159E+00	.43096E+00	.48143E+00	.48142E+00	.48133E+00	.53108E+00	.48128E+00	.57930E+00	.48126E+00	.62556E+00
.48125E+00	.66950E+00	.48125E+00	.71093E+00	.48125E+00	.75000E+00				

J = 18 (Y-Z ARRAY)

.53930E+00	0.	.53911E+00	.35607E-02	.53884E+00	.86227E-02	.53846E+00	.16391E-01	.53794E+00	.27774E-01
.53730E+00	.43421E-01	.53655E+00	.63750E-01	.53573E+00	.88972E-01	.53489E+00	.11909E+00	.53406E+00	.15392E+00
.53331E+00	.19307E+00	.53265E+00	.23601E+00	.53212E+00	.28205E+00	.53171E+00	.33041E+00	.53141E+00	.38028E+00
.53121E+00	.43085E+00	.53108E+00	.48133E+00	.53101E+00	.53101E+00	.53096E+00	.57924E+00	.53094E+00	.62552E+00
.53093E+00	.66947E+00	.53093E+00	.71092E+00	.53093E+00	.75000E+00				

J = 19 (Y-Z ARRAY)

.58530E+00	0.	.58523E+00	.35607E-02	.58503E+00	.86227E-02	.58474E+00	.16392E-01	.58435E+00	.27774E-01
.58387E+00	.43421E-01	.58331E+00	.63752E-01	.58271E+00	.88973E-01	.58208E+00	.11909E+00	.58140E+00	.15391E+00
.58093E+00	.19306E+00	.58045E+00	.23599E+00	.58006E+00	.28201E+00	.57976E+00	.33036E+00	.57955E+00	.38023E+00
.57940E+00	.43080E+00	.57930E+00	.48128E+00	.57924E+00	.53096E+00	.57921E+00	.57921E+00	.57919E+00	.62550E+00
.57918E+00	.66946E+00	.57917E+00	.71091E+00	.57917E+00	.75000E+00				

J = 20 (Y-Z ARRAY)

.62974E+00	0.	.62964E+00	.35607E-02	.62950E+00	.86228E-02	.62929E+00	.16392E-01	.62902E+00	.27774E-01
.62669E+00	.43423E-01	.62630E+00	.63758E-01	.62788E+00	.88978E-01	.62745E+00	.11910E+00	.62704E+00	.15392E+00

```
                                                                                  .62104E+00  .19392E+00

 .62646E+00   .19307E+00   .62634E+00   .23599E+00   .62608E+00   .28201E+00   .62588E+00   .33035E+00   .62573E+00   .38021E+00
 .62563E+00   .43077E+00   .62556E+00   .48126E+00   .62552E+00   .53094E+00   .62550E+00   .57919E+00   .62548E+00   .62548E+00
 .62547E+00   .66945E+00   .62547E+00   .71090E+00   .62547E+00   .75000E+00

J =    21     (Y-Z ARRAY)
 .67204E+00  0.            .67198E+00   .35607E-02   .67189E+00   .86228E-02   .67177E+00   .16392E-01   .67160E+00   .27775E-01
 .67139E+00   .43424E-01   .67116E+00   .63758E-01   .67090E+00   .88965E-01   .67064E+00   .11911E+00   .67039E+00   .15393E+00
 .67016E+00   .19308E+00   .66997E+00   .23600E+00   .66981E+00   .28202E+00   .66969E+00   .33036E+00   .66960E+00   .38021E+00
 .66954E+00   .43077E+00   .66950E+00   .48125E+00   .66947E+00   .53093E+00   .66946E+00   .57918E+00   .66945E+00   .62547E+00
 .66944E+00   .66944E+00   .66944E+00   .71090E+00   .66944E+00   .75000E+00

J =    22     (Y-Z ARRAY)
 .71209E+00  0.            .71206E+00   .35607E-02   .71202E+00   .86228E-02   .71197E+00   .16392E-01   .71189E+00   .27775E-01
 .71179E+00   .43425E-01   .71169E+00   .63760E-01   .71157E+00   .88989E-01   .71145E+00   .11911E+00   .71134E+00   .15394E+00
 .71123E+00   .19309E+00   .71114E+00   .23601E+00   .71107E+00   .28203E+00   .71101E+00   .33036E+00   .71097E+00   .38021E+00
 .71095E+00   .43077E+00   .71093E+00   .48125E+00   .71092E+00   .53093E+00   .71091E+00   .57917E+00   .71090E+00   .62547E+00
 .71090E+00   .66944E+00   .71090E+00   .71090E+00   .71090E+00   .75000E+00

J =    23     (Y-Z ARRAY)
 .75000E+00  0.            .75000E+00   .35607E-02   .75000E+00   .86228E-02   .75000E+00   .16392E-01   .75000E+00   .27775E-01
 .75000E+00   .43425E-01   .75000E+00   .63760E-01   .75000E+00   .88989E-01   .75000E+00   .11911E+00   .75000E+00   .15394E+00
 .75000E+00   .19309E+00   .75000E+00   .23601E+00   .75000E+00   .28203E+00   .75000E+00   .33036E+00   .75000E+00   .38021E+00
 .75000E+00   .43077E+00   .75000E+00   .48125E+00   .75000E+00   .53093E+00   .75000E+00   .57917E+00   .75000E+00   .62547E+00
 .75000E+00   .66944E+00   .75000E+00   .71090E+00   .75000E+00   .75000E+00

 J=1,K=1,X=1,IMAX  (X,Y,Z ARRAY)
0.            .30114E-01   .30114E-01   .68190E-02   .30114E-01   .30114E-01   .13640E-01   .30114E-01   .30114E-01
 .20480E-01   .30114E-01   .30114E-01   .27370E-01   .30114E-01   .30114E-01   .34460E-01   .30114E-01   .30114E-01
 .42120E-01   .30114E-01   .30114E-01   .51190E-01   .30114E-01   .30114E-01   .63230E-01   .30114E-01   .30114E-01
 .80520E-01   .30114E-01   .30114E-01   .10580E+00   .30114E-01   .30114E-01   .14120E+00   .30114E-01   .30114E-01
 .17660E+00   .30114E-01   .30114E-01   .21200E+00   .30114E-01   .30114E-01   .24740E+00   .30114E-01   .30114E-01
 .28280E+00   .30114E-01   .30114E-01   .31820E+00   .30114E-01   .30114E-01   .35360E+00   .30114E-01   .30114E-01
 .38900E+00   .30114E-01   .30114E-01   .42440E+00   .30114E-01   .30114E-01   .45980E+00   .30114E-01   .30114E-01
 .49520E+00   .30114E-01   .30114E-01   .53060E+00   .30114E-01   .30114E-01   .56600E+00   .30114E-01   .30114E-01
 .60140E+00   .30114E-01   .30114E-01   .63680E+00   .30114E-01   .30114E-01   .67220E+00   .30114E-01   .30114E-01
 .70760E+00   .30114E-01   .30114E-01   .74300E+00   .30114E-01   .30114E-01   .77840E+00   .30114E-01   .30114E-01

     ITER NO.,IYLOCJ,IYLOCK,YERRMX,IZLOCJ,IZLOCK,ZERRMX =   100    15    10    .166944E-04    10    15    .166981E-04
```

*** CONVERGED SOLUTION ***
CORNER FLOW WITH FILLET
COORDINATE SYSTEM. SIZE=30X23X23

JMAX = 23 KMAX = 23
MAX ITERATIONS PERFORMED = 119
MAX Y-ERROR = .998592E-05 AT LOC. J,K = 15, 10
MAX Z-ERROR = .999905E-05 AT LOC. J,K = 10, 15

J = 1 (Y-Z ARRAY)

.30114E-01	.30114E-01	.26723E-01	.33673E-01	.22417E-01	.38732E-01	.16805E-01	.46487E-01	.10370E-01	.57820E-01
.41333E-02	.73335E-01	.43390E-03	.93379E-01	0.	.11807E+00	0.	.14734E+00	0.	.18092E+00
0.	.21842E+00	0.	.25933E+00	0.	.30303E+00	0.	.34886E+00	0.	.39610E+00
0.	.44405E+00	0.	.49200E+00	0.	.53930E+00	0.	.58538E+00	0.	.62974E+00
0.	.67204E+00	0.	.71209E+00	0.	.75000E+00				

J = 2 (Y-Z ARRAY)

.33673E-01	.26723E-01	.31704E-01	.31704E-01	.27768E-01	.37626E-01	.21875E-01	.45759E-01	.14938E-01	.57216E-01
.83075E-02	.72769E-01	.42700E-02	.92804E-01	.36047E-02	.11748E+00	.35684E-02	.14678E+00	.35638E-02	.18043E+00
.35621E-02	.21800E+00	.35614E-02	.25897E+00	.35609E-02	.30273E+00	.35607E-02	.34861E+00	.35605E-02	.39590E+00
.35604E-02	.44389E+00	.35604E-02	.49187E+00	.35603E-02	.53920E+00	.35603E-02	.58531E+00	.35603E-02	.62969E+00
.35603E-02	.67201E+00	.35603E-02	.71208E+00	.35603E-02	.75000E+00				

J = 3 (Y-Z ARRAY)

.38732E-01	.22417E-01	.37626E-01	.27768E-01	.34772E-01	.34772E-01	.29284E-01	.43977E-01	.21831E-01	.56048E-01
.14478E-01	.71835E-01	.98447E-02	.91930E-01	.87795E-02	.11662E+00	.86566E-02	.14599E+00	.86361E-02	.17973E+00
.86290E-02	.21740E+00	.86257E-02	.25846E+00	.86239E-02	.30230E+00	.86228E-02	.34826E+00	.86222E-02	.39561E+00
.86219E-02	.44365E+00	.86215E-02	.49169E+00	.86214E-02	.53906E+00	.86213E-02	.58521E+00	.86212E-02	.62962E+00
.86212E-02	.67197E+00	.86212E-02	.71206E+00	.86212E-02	.75000E+00				

J = 4 (Y-Z ARRAY)

.46487E-01	.16805E-01	.45759E-01	.21875E-01	.43977E-01	.29283E-01	.39775E-01	.39775E-01	.32741E-01	.53227E-01
.24677E-01	.69891E-01	.18904E-01	.90370E-01	.16950E-01	.11523E+00	.16547E-01	.14474E+00	.16454E-01	.17864E+00
.16421E-01	.21648E+00	.16406E-01	.25769E+00	.16397E-01	.30166E+00	.16393E-01	.34772E+00	.16390E-01	.39517E+00
.16389E-01	.44330E+00	.16387E-01	.49141E+00	.16386E-01	.53885E+00	.16386E-01	.58505E+00	.16386E-01	.62952E+00
.16386E-01	.67191E+00	.16386E-01	.71203E+00	.16386E-01	.75000E+00				

J = 5 (Y-Z ARRAY)

.57820E-01	.10370E-01	.57216E-01	.14938E-01	.56048E-01	.21831E-01	.53227E-01	.32741E-01	.47634E-01	.47633E-01
.39886E-01	.65895E-01	.32980E-01	.87492E-01	.29503E-01	.11296E+00	.28369E-01	.14283E+00	.28021E-01	.17703E+00
.27891E-01	.21511E+00	.27831E-01	.25655E+00	.27799E-01	.30071E+00	.27781E-01	.34694E+00	.27770E-01	.39453E+00
.27763E-01	.44279E+00	.27759E-01	.49101E+00	.27756E-01	.53855E+00	.27755E-01	.58483E+00	.27754E-01	.62937E+00
.27753E-01	.67182E+00	.27753E-01	.71199E+00	.27753E-01	.75000E+00				

J = 6 (Y-Z ARRAY)

.73335E-01	.41333E-02	.72769E-01	.83075E-02	.71835E-01	.14478E-01	.69892E-01	.24676E-01	.65895E-01	.39886E-01
.59499E-01	.59498E-01	.52458E-01	.82704E-01	.47504E-01	.10942E+00	.45139E-01	.14005E+00	.44180E-01	.17475E+00

.43793E-01	.21322E+00	.43605E-01	.25498E+00	.43506E-01	.29941E+00	.43450E-01	.34588E+00	.43416E-01	.39367E+00
.43395E-01	.44210E+00	.43382E-01	.49047E+00	.43374E-01	.53814E+00	.43370E-01	.58453E+00	.43367E-01	.62917E+00
.43365E-01	.67169E+00	.43365E-01	.71193E+00	.43365E-01	.75000E+00				

J = 7 (Y-Z ARRAY)

.93379E-01	.43390E-03	.92804E-01	.42700E-02	.91930E-01	.98447E-02	.90371E-01	.18904E-01	.87493E-01	.32980E-01
.82704E-01	.52457E-01	.76576E-01	.76575E-01	.71043E-01	.10458E+00	.67461E-01	.13629E+00	.65596E-01	.17175E+00
.64688E-01	.21077E+00	.64230E-01	.25296E+00	.63982E-01	.29775E+00	.63839E-01	.34452E+00	.63754E-01	.39257E+00
.63702E-01	.44122E+00	.63669E-01	.48979E+00	.63649E-01	.53762E+00	.63637E-01	.58416E+00	.63629E-01	.62891E+00
.63625E-01	.67154E+00	.63624E-01	.71186E+00	.63624E-01	.75000E+00				

J = 8 (Y-Z ARRAY)

.11807E+00	0.	.11748E+00	.36047E-02	.11662E+00	.87795E-02	.11523E+00	.16950E-01	.11296E+00	.29503E-01
.10942E+00	.47504E-01	.10458E+00	.71042E-01	.99421E-01	.99419E-01	.95219E-01	.13193E+00	.92472E-01	.16816E+00
.90863E-01	.20783E+00	.89994E-01	.25053E+00	.89488E-01	.29575E+00	.89190E-01	.34288E+00	.89009E-01	.39124E+00
.88897E-01	.44017E+00	.88827E-01	.48896E+00	.88783E-01	.53700E+00	.88757E-01	.58371E+00	.88741E-01	.62861E+00
.88732E-01	.67136E+00	.88729E-01	.71178E+00	.88729E-01	.75000E+00				

J = 9 (Y-Z ARRAY)

.14734E+00	0.	.14678E+00	.35684E-02	.14599E+00	.86566E-02	.14474E+00	.16547E-01	.14283E+00	.28369E-01
.14005E+00	.45138E-01	.13629E+00	.67460E-01	.13193E+00	.95218E-01	.12778E+00	.12777E+00	.12452E+00	.16443E+00
.12231E+00	.20462E+00	.12092E+00	.24783E+00	.12006E+00	.29350E+00	.11954E+00	.34102E+00	.11921E+00	.38973E+00
.11900E+00	.43896E+00	.11887E+00	.48802E+00	.11879E+00	.53629E+00	.11874E+00	.58319E+00	.11871E+00	.62826E+00
.11869E+00	.67115E+00	.11869E+00	.71166E+00	.11869E+00	.75000E+00				

J = 10 (Y-Z ARRAY)

.18092E+00	0.	.18043E+00	.35638E-02	.17973E+00	.86361E-02	.17864E+00	.16454E-01	.17703E+00	.28021E-01
.17475E+00	.44189E-01	.17175E+00	.65595E-01	.16816E+00	.92471E-01	.16443E+00	.12452E+00	.16111E+00	.16111E+00
.15853E+00	.20152E+00	.15671E+00	.24508E+00	.15549E+00	.29114E+00	.15469E+00	.33904E+00	.15418E+00	.38811E+00
.15384E+00	.43766E+00	.15363E+00	.48701E+00	.15349E+00	.53552E+00	.15341E+00	.58264E+00	.15336E+00	.62788E+00
.15333E+00	.67092E+00	.15332E+00	.71158E+00	.15332E+00	.75000E+00				

J = 11 (Y-Z ARRAY)

.21842E+00	0.	.21800E+00	.35621E-02	.21740E+00	.86290E-02	.21648E+00	.16421E-01	.21511E+00	.27891E-01
.21323E+00	.43792E-01	.21078E+00	.64687E-01	.20783E+00	.90882E-01	.20462E+00	.12231E+00	.20152E+00	.15853E+00
.19886E+00	.19686E+00	.19679E+00	.24255E+00	.19528E+00	.28887E+00	.19424E+00	.33709E+00	.19353E+00	.38647E+00
.19306E+00	.43633E+00	.19275E+00	.48596E+00	.19255E+00	.53472E+00	.19242E+00	.58206E+00	.19234E+00	.62749E+00
.19230E+00	.67068E+00	.19228E+00	.71147E+00	.19228E+00	.75000E+00				

J = 12 (Y-Z ARRAY)

.25933E+00	0.	.25897E+00	.35614E-02	.25846E+00	.86257E-02	.25769E+00	.16406E-01	.25655E+00	.27831E-01
.25498E+00	.43605E-01	.25296E+00	.64229E-01	.25053E+00	.89993E-01	.24783E+00	.12092E+00	.24508E+00	.15670E+00
.24256E+00	.19679E+00	.24044E+00	.24043E+00	.23878E+00	.28686E+00	.23756E+00	.33528E+00	.23670E+00	.38492E+00
.23610E+00	.43505E+00	.23570E+00	.48494E+00	.23543E+00	.53394E+00	.23526E+00	.58149E+00	.23516E+00	.62710E+00
.23510E+00	.67045E+00	.23507E+00	.71137E+00	.23507E+00	.75000E+00				

J = 13 (Y-Z ARRAY)

.30303E+00	0.	.30273E+00	.35609E-02	.30230E+00	.86239E-02	.30166E+00	.16397E-01	.30071E+00	.27799E-01
.29941E+00	.43506E-01	.29775E+00	.63981E-01	.29575E+00	.89488E-01	.29350E+00	.12006E+00	.29114E+00	.15548E+00

.28887E+00	.19528E+00	.28686E+00	.23878E+00	.28520E+00	.28520E+00	.28391E+00	.33373E+00	.28296E+00	.38355E+00
.28227E+00	.43389E+00	.28180E+00	.48400E+00	.28148E+00	.53322E+00	.28127E+00	.58096E+00	.28114E+00	.62674E+00
.28107E+00	.67023E+00	.28104E+00	.71127E+00	.28104E+00	.75000E+00				

J = 14 (Y-Z ARRAY)

.34886E+00	0.	.34861E+00	.35607E-02	.34826E+00	.86228E-02	.34772E+00	.16393E-01	.34694E+00	.27781E-01
.34588E+00	.43450E-01	.34452E+00	.63839E-01	.34288E+00	.89190E-01	.34102E+00	.11954E+00	.33905E+00	.15469E+00
.33709E+00	.19424E+00	.33526E+00	.23756E+00	.33373E+00	.28391E+00	.33247E+00	.33247E+00	.33150E+00	.38240E+00
.33079E+00	.43289E+00	.33028E+00	.48318E+00	.32993E+00	.53258E+00	.32970E+00	.58049E+00	.32955E+00	.62642E+00
.32947E+00	.67003E+00	.32944E+00	.71118E+00	.32944E+00	.75000E+00				

J = 15 (Y-Z ARRAY)

.39610E+00	0.	.39590E+00	.35605E-02	.39561E+00	.86222E-02	.39517E+00	.16390E-01	.39453E+00	.27770E-01
.39367E+00	.43416E-01	.39257E+00	.63754E-01	.39125E+00	.89009E-01	.38974E+00	.11921E+00	.38811E+00	.15417E+00
.38643E+00	.19353E+00	.38493E+00	.23669E+00	.38355E+00	.28295E+00	.38240E+00	.33150E+00	.38149E+00	.38149E+00
.38080E+00	.43208E+00	.38029E+00	.48250E+00	.37994E+00	.53204E+00	.37970E+00	.58009E+00	.37955E+00	.62614E+00
.37946E+00	.66987E+00	.37943E+00	.71110E+00	.37943E+00	.75000E+00				

J = 16 (Y-Z ARRAY)

.44405E+00	0.	.44389E+00	.35604E-02	.44365E+00	.86218E-02	.44330E+00	.16388E-01	.44279E+00	.27763E-01
.44210E+00	.43395E-01	.44122E+00	.63702E-01	.44017E+00	.88897E-01	.43896E+00	.11900E+00	.43766E+00	.15384E+00
.43633E+00	.19306E+00	.43505E+00	.23610E+00	.43389E+00	.28227E+00	.43289E+00	.33079E+00	.43208E+00	.38079E+00
.43146E+00	.43146E+00	.43099E+00	.48197E+00	.43066E+00	.53161E+00	.43043E+00	.57977E+00	.43029E+00	.62592E+00
.43021E+00	.66973E+00	.43017E+00	.71104E+00	.43017E+00	.75000E+00				

J = 17 (Y-Z ARRAY)

.49200E+00	0.	.49187E+00	.35604E-02	.49169E+00	.86215E-02	.49141E+00	.16387E-01	.49101E+00	.27759E-01
.49047E+00	.43382E-01	.48978E+00	.63669E-01	.48896E+00	.88827E-01	.48802E+00	.11887E+00	.48701E+00	.15363E+00
.48596E+00	.19275E+00	.48494E+00	.23570E+00	.48400E+00	.28180E+00	.48318E+00	.33028E+00	.48251E+00	.38029E+00
.48197E+00	.43099E+00	.48157E+00	.48157E+00	.48128E+00	.53129E+00	.48108E+00	.57952E+00	.48095E+00	.62575E+00
.48087E+00	.66962E+00	.48084E+00	.71099E+00	.48084E+00	.75000E+00				

J = 18 (Y-Z ARRAY)

.53930E+00	0.	.53920E+00	.35603E-02	.53906E+00	.86214E-02	.53885E+00	.16386E-01	.53855E+00	.27756E-01
.53814E+00	.43374E-01	.53762E+00	.63649E-01	.53700E+00	.88783E-01	.53629E+00	.11879E+00	.53552E+00	.15349E+00
.53473E+00	.19255E+00	.53394E+00	.23543E+00	.53322E+00	.28148E+00	.53258E+00	.32993E+00	.53204E+00	.37994E+00
.53162E+00	.43066E+00	.53129E+00	.48128E+00	.53105E+00	.53105E+00	.53089E+00	.57934E+00	.53078E+00	.62562E+00
.53072E+00	.66955E+00	.53069E+00	.71095E+00	.53069E+00	.75000E+00				

J = 19 (Y-Z ARRAY)

.58538E+00	0.	.58531E+00	.35603E-02	.58520E+00	.86213E-02	.58505E+00	.16386E-01	.58483E+00	.27755E-01
.58453E+00	.43370E-01	.58416E+00	.63636E-01	.58371E+00	.88756E-01	.58319E+00	.11874E+00	.58264E+00	.15341E+00
.58206E+00	.19242E+00	.58149E+00	.23526E+00	.58096E+00	.28127E+00	.58049E+00	.32970E+00	.58009E+00	.37970E+00
.57977E+00	.43043E+00	.57952E+00	.48108E+00	.57934E+00	.53088E+00	.57921E+00	.57921E+00	.57913E+00	.62553E+00
.57908E+00	.66949E+00	.57906E+00	.71093E+00	.57906E+00	.75000E+00				

J = 20 (Y-Z ARRAY)

.62974E+00	0.	.62969E+00	.35603E-02	.62962E+00	.86212E-02	.62952E+00	.16386E-01	.62937E+00	.27754E-01
.62917E+00	.43367E-01	.62891E+00	.63629E-01	.62861E+00	.88741E-01	.62826E+00	.11871E+00	.62788E+00	.15336E+00

.62749E+00 .19234E+00 .62710E+00 .23515E+00 .62674E+00 .28114E+00 .62642E+00 .32955E+00 .62614E+00 .37955E+00
.62592E+00 .43029E+00 .62575E+00 .48095E+00 .62562E+00 .53078E+00 .62553E+00 .57913E+00 .62547E+00 .62547E+00
.62544E+00 .66945E+00 .62542E+00 .71091E+00 .62542E+00 .75000E+00

J = 21 (Y-Z ARRAY)
.67204E+00 0. .67201E+00 .35603E-02 .67197E+00 .86212E-02 .67191E+00 .16386E-01 .67182E+00 .27753E-01
.67169E+00 .43365E-01 .67154E+00 .63625E-01 .67136E+00 .88732E-01 .67115E+00 .11869E+00 .67092E+00 .15333E+00
.67068E+00 .19230E+00 .67045E+00 .23510E+00 .67023E+00 .28107E+00 .67003E+00 .32947E+00 .66987E+00 .37946E+00
.66973E+00 .43021E+00 .66962E+00 .48087E+00 .66955E+00 .53072E+00 .66949E+00 .57908E+00 .66945E+00 .62544E+00
.66943E+00 .66943E+00 .66942E+00 .71090E+00 .66942E+00 .75000E+00

J = 22 (Y-Z ARRAY)
.71209E+00 0. .71208E+00 .35603E-02 .71206E+00 .86212E-02 .71203E+00 .16386E-01 .71199E+00 .27753E-01
.71193E+00 .43365E-01 .71186E+00 .63624E-01 .71178E+00 .88729E-01 .71168E+00 .11869E+00 .71158E+00 .15332E+00
.71147E+00 .19228E+00 .71137E+00 .23507E+00 .71127E+00 .28104E+00 .71118E+00 .32943E+00 .71110E+00 .37943E+00
.71104E+00 .43017E+00 .71099E+00 .48084E+00 .71095E+00 .53069E+00 .71093E+00 .57906E+00 .71091E+00 .62542E+00
.71090E+00 .66942E+00 .71090E+00 .71090E+00 .71090E+00 .75000E+00

J = 23 (Y-Z ARRAY)
.75000E+00 0. .75000E+00 .35603E-02 .75000E+00 .86212E-02 .75000E+00 .16386E-01 .75000E+00 .27753E-01
.75000E+00 .43365E-01 .75000E+00 .63624E-01 .75000E+00 .88729E-01 .75000E+00 .11869E+00 .75000E+00 .15332E+00
.75000E+00 .19228E+00 .75000E+00 .23507E+00 .75000E+00 .28104E+00 .75000E+00 .32943E+00 .75000E+00 .37943E+00
.75000E+00 .43017E+00 .75000E+00 .48084E+00 .75000E+00 .53069E+00 .75000E+00 .57906E+00 .75000E+00 .62542E+00
.75000E+00 .66942E+00 .75000E+00 .71090E+00 .75000E+00 .75000E+00

J=1,K=1,X=1,IMAX (X,Y,Z ARRAY)
0. .30114E-01 .30114E-01 .68190E-02 .30114E-01 .30114E-01 .13640E-01 .30114E-01 .30114E-01
.20480E-01 .30114E-01 .30114E-01 .27370E-01 .30114E-01 .30114E-01 .34460E-01 .30114E-01 .30114E-01
.42120E-01 .30114E-01 .30114E-01 .51190E-01 .30114E-01 .30114E-01 .63230E-01 .30114E-01 .30114E-01
.80520E-01 .30114E-01 .30114E-01 .10580E+00 .30114E-01 .30114E-01 .14120E+00 .30114E-01 .30114E-01
.17650E+00 .30114E-01 .30114E-01 .21200E+00 .30114E-01 .30114E-01 .24740E+00 .30114E-01 .30114E-01
.28230E+00 .30114E-01 .30114E-01 .31820E+00 .30114E-01 .30114E-01 .35360E+00 .30114E-01 .30114E-01
.38900E+00 .30114E-01 .30114E-01 .42440E+00 .30114E-01 .30114E-01 .45980E+00 .30114E-01 .30114E-01
.49520E+00 .30114E-01 .30114E-01 .53060E+00 .30114E-01 .30114E-01 .56600E+00 .30114E-01 .30114E-01
.60140E+00 .30114E-01 .30114E-01 .63680E+00 .30114E-01 .30114E-01 .67220E+00 .30114E-01 .30114E-01
.70760E+00 .30114E-01 .30114E-01 .74300E+00 .30114E-01 .30114E-01 .77840E+00 .30114E-01 .30114E-01

THE PLOT CONTROL CARD IMAGE IS,

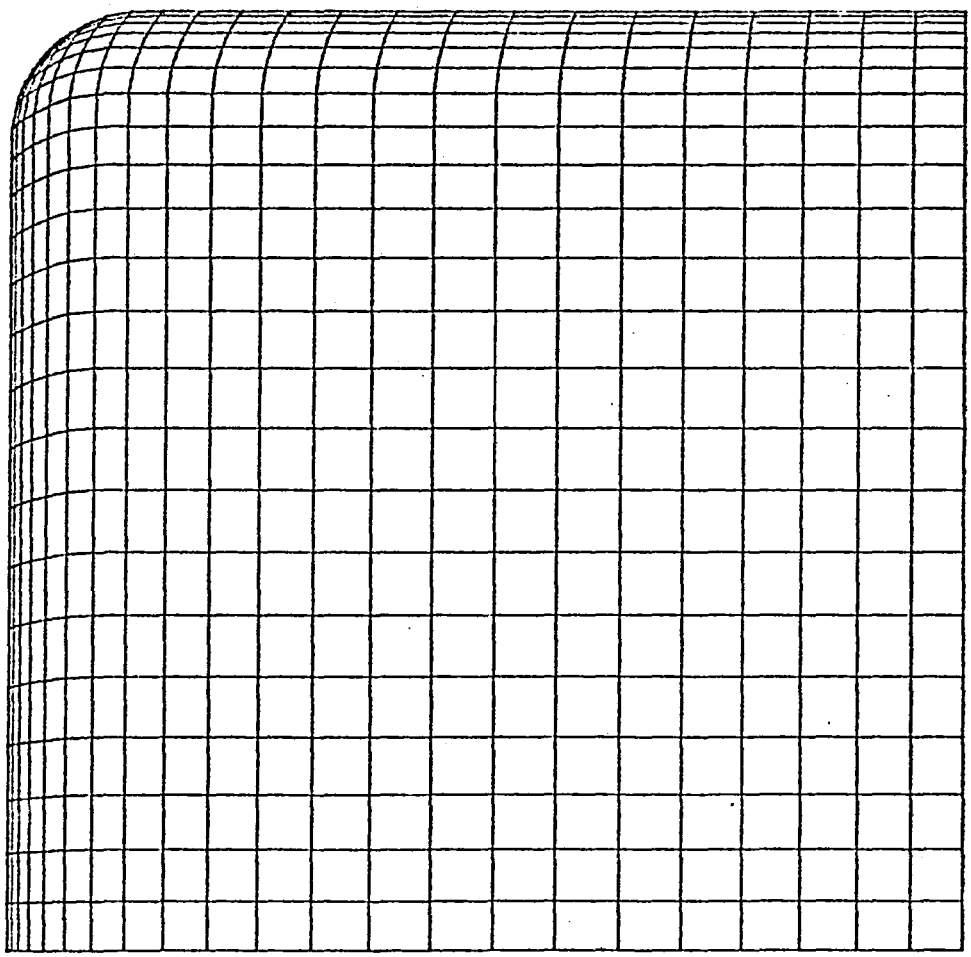
PLOT.VARIAN

FRAME	XO	YO	XM	YM	CAL. POS
1	0.	0.	1.000000E+00	1.000000E+00	0.
2	0.	0.	1.000000E+00	1.000000E+00	0.
3	0.	0.	1.000000E+00	1.000000E+00	0.
4	0.	0.	1.000000E+00	1.000000E+00	0.

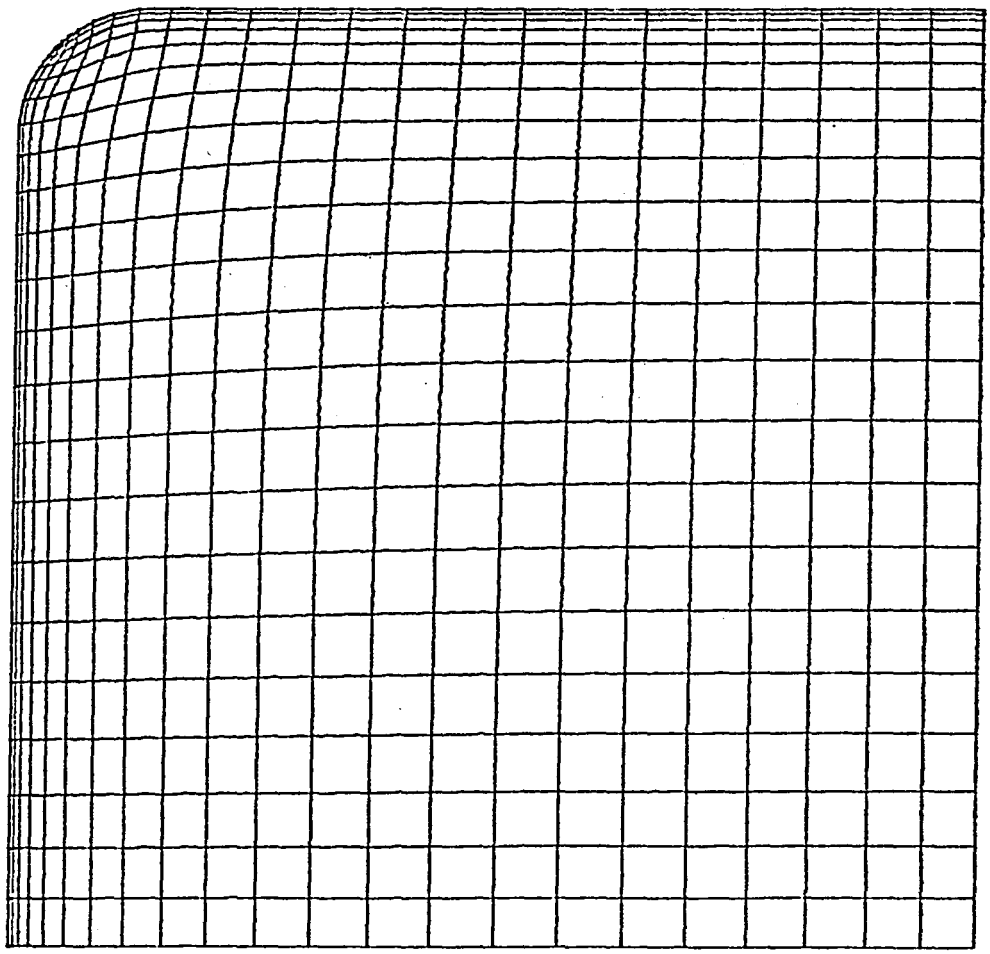
CORNER FLOW WITH FILLET
COORDINATE SYSTEM. SIZE 30X23X23

--- INITIAL GUESS ---

JMAX 23 KMAX 23



CORNER FLOW WITH FILLET
COORDINATE SYSTEM. SIZE 30X23X23
--- CONVERGED SOLUTION ---
JMAX 23 KMAX 23



III. Programmer/Analyst Section

The program is written such that the array size in ξ, η, ζ direction can be varied from problem to problem by the use of test editor on CDC 6600 machine. The array sizes are coded to be IMXLL, JMXLL, KMXLL in ξ, η and ζ direction respectively. These are changed to the appropriate numeric number by the test editor prior to compiling the program for execution. The job control cards, input explanation and other helpful hints are provided towards the beginning of main program COORDC. The program is extensively commented. Though the computer code is not structured in the true sense, extreme care was taken to avoid branching back in the code unless absolutely necessary. A brief discussion of each subroutine and their function follow.

A. Main Program COORDC

This program performs the following function,

i) Initialize variables to 0.0.

ii) Read in input data. If the field size is greater than the maximum field size allowed then the program aborts with a diagnostic message. The maximum field size allowed is ILIMIT, JLIMIT, KLIMIT and the field size input is IMAX, JMAX, KMAX. ILIMIT, JLIMIT, and KLIMIT are set to numerical value by the text editor. The storage requirement in N-S code require that IMAX, JMAX, KMAX be equal to ILIMIT, JLIMIT, KLIMIT respectively.

iii) Specify initial guess.

iv) Check to see if initial guess is to be printed, plotted or not. If so, then the appropriate subroutines are called.

v) Compute attraction parameter Q and R.

vi) Start computing solution. Do loop for maximum iterations (ITERMX) is set. For each iteration subroutine CALCOR is called. CALCOR computes solution for each iteration and also passes back maximum error and its location.

vii) Print maximum error and partially converged solution every 50 iterations. Partially converged solution is printed only if the flag to print the initial guess is set.

viii) Check maximum error in y and z for convergence. If convergence is reached, then an exit from the DO loop is taken.

ix) After ITERMX iterations or convergence, which ever comes first, some more print/plot and storing of data is done.

x) Check to see if partially or converged solution is to be printed/plotted.

xi) If convergence has occurred, then store solution on unit 1. If convergence has not occurred, then the maximum errors in y and z computation at the last iteration and their location are printed out.

B. Subroutine CALCOR

This subroutine advances the solution 1 iteration. The maximum errors and their location are computed and passed back to the calling program COORDC through the argument list. The argument list is,

YERRMX - maximum error in y

ZERRMX - maximum error in z

IYLOCJ - maximum y-error J location

IYLOCK - maximum y-error K location

IZLOCJ - maximum z-error J location

IZLOCK - maximum z-error K location

C. Subroutine IPRTC

This subroutine is used to print initial guess, partially converged or converged solution. The argument list is,

ISOLN = 1 indicates initial guess

= 2 indicates partially or fully converged solution

KKK = applicable only if ISOLN = 2

= 0 indicates partially converged solution

= 1 indicates fully converged solution.

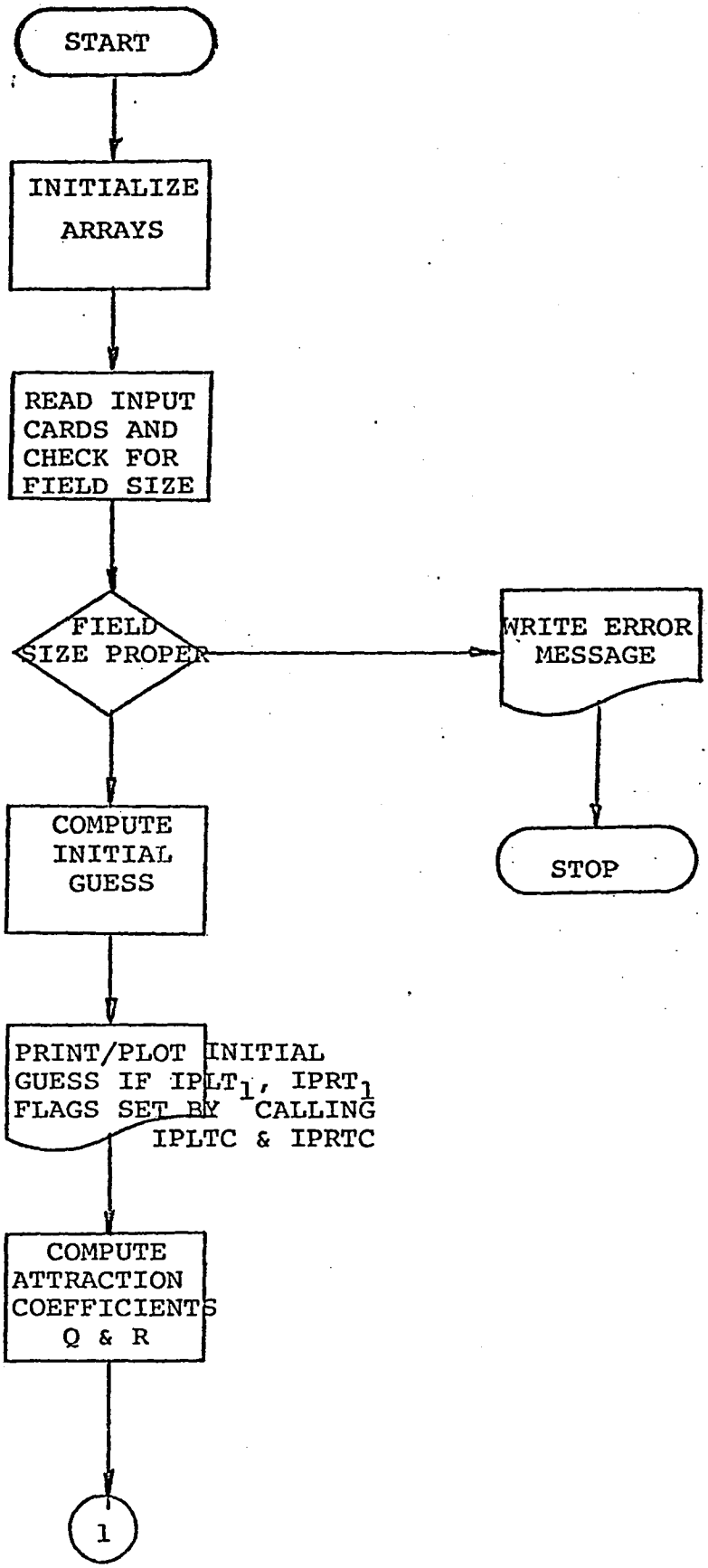
YERRMX, ZERRMX, IYLOCJ, IYLOCK, IZLOCJ, IZLOCK = same as in CALCOR.

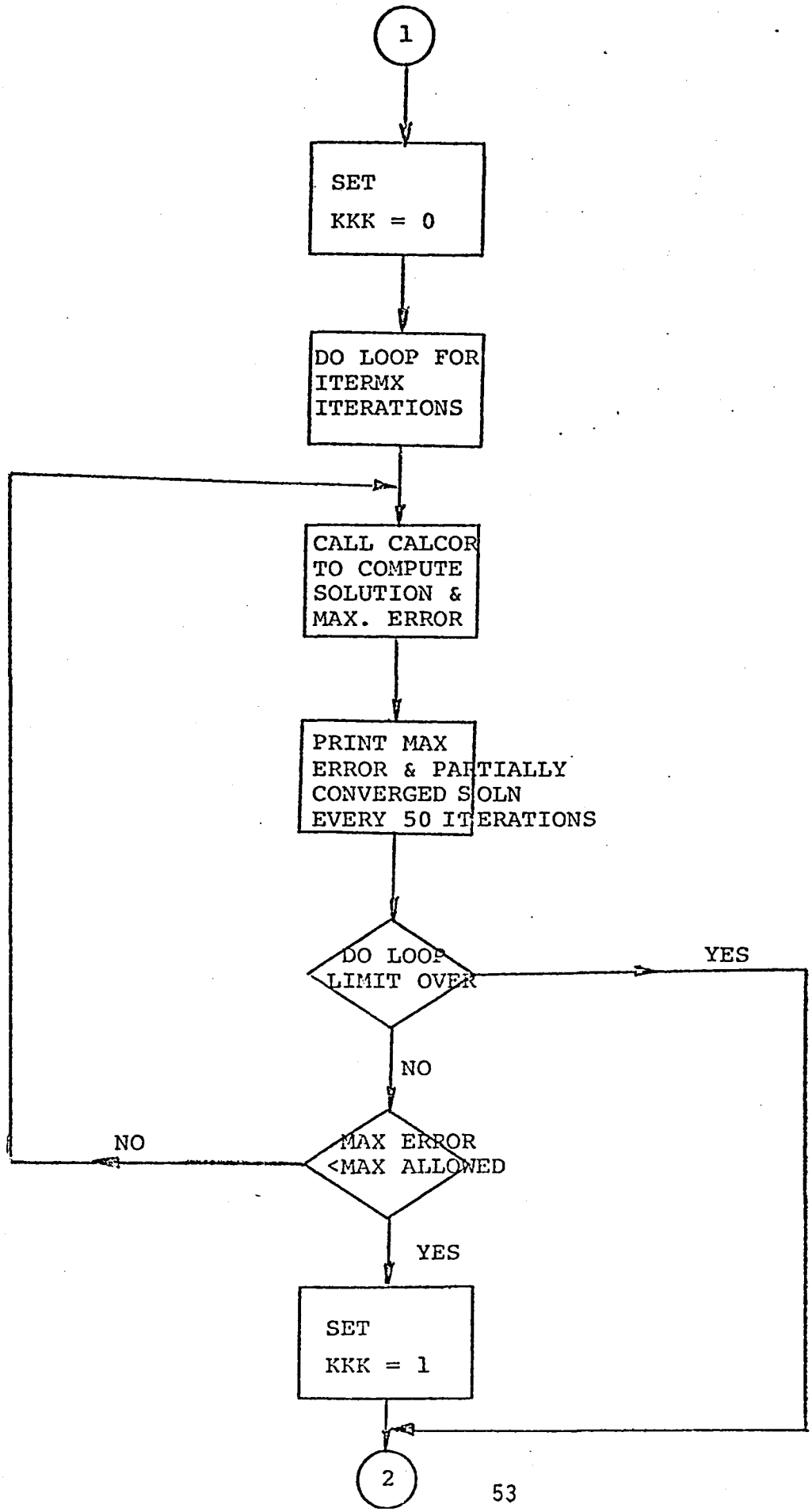
D. Subroutine IPLTC

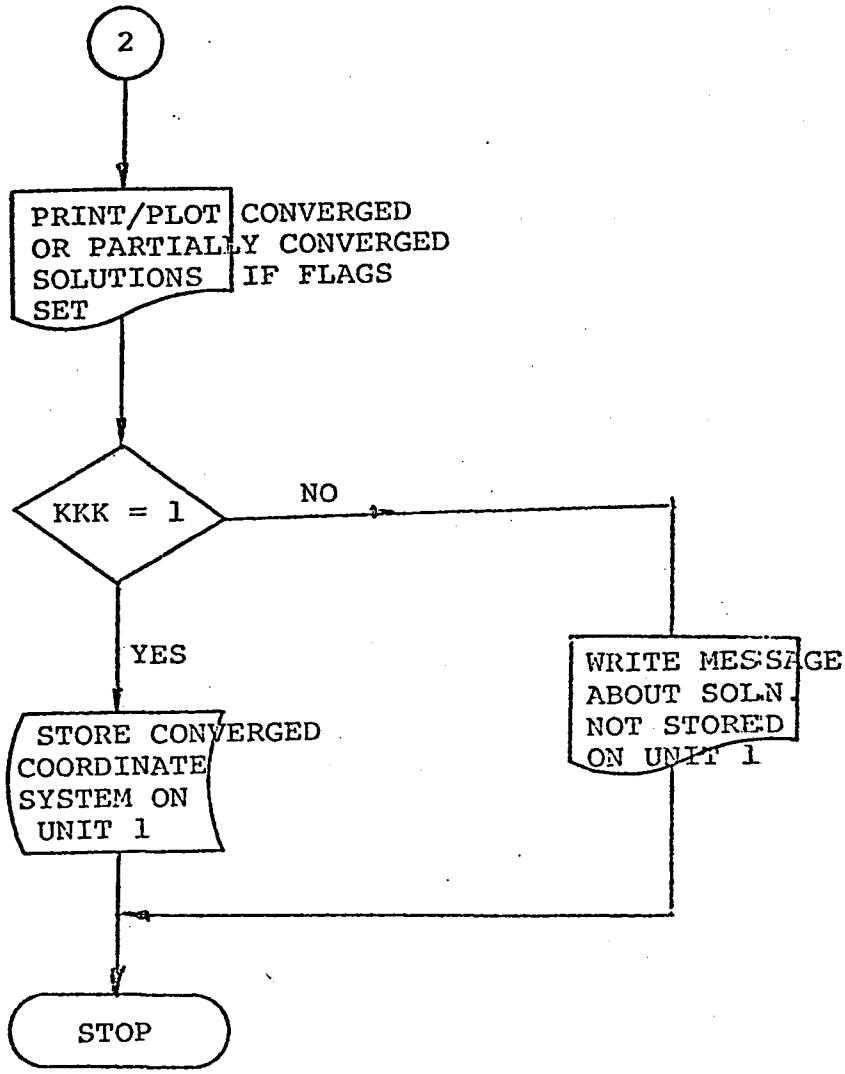
This subroutine is used to plot initial guess, partially converged solution or converged solution. The argument list is same as IPRTC.

E. Block Diagram

A Block Diagram is presented on the next page.


START
INITIALIZE ARRAYS
READ INPUT CARDS AND CHECK FOR FIELD SIZE
FIELD SIZE PROPER
WRITE ERROR MESSAGE
STOP
COMPUTE INITIAL GUESS
PRINT/PLOT INITIAL GUESS IF IPLT1, IPRT1 FLAGS SET BY CALLING IPLTC & IPRTC
COMPUTE ATTRACTION COEFFICIENTS Q & R
1


1
SET
KKK = 0
DO LOOP FOR
ITERMX
ITERATIONS
CALL CALCOR
TO COMPUTE
SOLUTION &
MAX. ERROR
PRINT MAX
ERROR & PARTIALLY
CONVERGED SOLN
EVERY 50 ITERATIONS
DO LOOP
LIMIT OVER
YES
NO
MAX ERROR
<MAX ALLOWED
NO
YES
SET
KKK = 1
2


2
PRINT/PLOT CONVERGED OR PARTIALLY CONVERGED SOLUTIONS IF FLAGS SET
KKK = 1
NO
YES
STORE CONVERGED COORDINATE SYSTEM ON UNIT 1
WRITE MESSAGE ABOUT SOLN. NOT STORED ON UNIT 1
STOP

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-159279		

4. Title and Subtitle	5. Report Date
Documentation of Program COORDC to Generate Coordinate System for 3-D Corners with or Without Fillet Using Body-Fitted Curvilinear Coordinates - Part II	June 1980
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Dilip Kumar	R-SAL-06/80-03
	10. Work Unit No.
9. Performing Organization Name and Address	11. Contract or Grant No.
Systems & Applied Sciences Corp. 17 Research Drive Hampton, VA 23666	NAS1-15604
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics & Space Administration Washington, DC 20546	14. Sponsoring Agency Code

15. Supplementary Notes

Technical Monitor - Dr. Julius Harris, NASA Langley Research Center

16. Abstract

The computer program COORDC generates a body-fitted curvilinear coordinate system for corner geometry with or without corner fillets. It is assumed that at any given ξ, x remains constant; consequently the only variation is in y and z. It is also assumed that for all ξ's in the physical plane the coordinate system in y-z plane is similar. This enables solution of coordinate system for one particular $\xi = 1$ (x for $\xi = 1$ is arbitrarily chosen to be 0.0) and the solution for all other ξ plane can be easily specified once the coordinates in the physical plane on the line $1 \leq \xi \leq IMAX$, $\eta = 1$, $\zeta = 1$ are specified.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Navier Stokes, compressible three dimensional, curvilinear coordinate systems, corner fillets, COORDC	[redacted] DISTRIBUTION

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	57	

*Available: NASA's Industrial Applications Centers

End of Document