NASA TECHNICAL NOTE

NASA TN D-8483

NASA TN D-8483

FINITE-ELEMENT COMPUTER PROGRAM FOR AXISYMMETRIC LOADING SITUATIONS WHERE COMPONENTS MAY HAVE A RELATIVE INTERFERENCE FIT

Christopher M. Taylor

Lewis Research Center
Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MAY 1977

1. Report No. NASA TN D-8483	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle FINITE-ELEMENT COMPUTER PROGRAM FOR AXISYMMETRIC LOADING SITUATIONS WHERE COMPONENTS MAY HAVE A RELATIVE INTERFERENCE FIT	5. Report Date May 1977 6. Performing Organization Code
7. Author(s) Christopher M. Taylor	8. Performing Organization Report No. E-9018
9. Performing Organization Name and Address Lewis Research Center National Aeronautics and Space Administration Cleveland, Ohio 44135	10. Work Unit No. 505-04 11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546	13. Type of Report and Period Covered Technical Note 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A finite-element computer program which enables the analysis of distortions and stresses occurring in components having a relative interference is presented. The program is limited to situations in which the loading is axisymmetric. As well as loads arising from the interference fit(s), external, inertial, and thermal loadings can be accommodated. The components may comprise several different homogeneous isotropic materials whose properties may be a function of temperature. An example illustrating the data input and program output is given.

17. Key Words (Suggested by Author(s)) Interference, finite-element, computer programs, axisymmetric bodies	18. Distribution Statement Unclassified - unlimited STAR Category 37

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 48	22. Price* A03

* For sale by the National Technical Information Service, Springfield, Virginia 22161

FINITE-ELEMENT COMPUTER PROGRAM FOR AXISYMMETRIC LOADING SITUATIONS WHERE COMPONENTS MAY HAVE A RELATIVE INTERFERENCE FIT

by Christopher M. Taylor*

Lewis Research Center

SUMMARY

This finite-element computer program permits the analysis of axisymmetrically loaded components having a relative interference fit. External, inertial, and thermal loads may be applied in addition to forces arising from an interference condition. The components under investigation may comprise different homogeneous isotropic materials whose properties may be temperature dependent. The theoretical background to the analysis of the interference condition is described, and the computer program structure outlined briefly. Particular attention is paid to the program input requirements and the output. The output comprises the input information, nodal distortions, element stresses, and details of the interference contact conditions. The running of multiple cases, with one or a number of different geometries, is possible. An example to illustrate the data input and program output is presented.

INTRODUCTION

The ability of the engineer to undertake the analysis of deformations and stresses occurring in the structures and components with which he deals has changed remarkably in the last 15 years. The development of finite-element approximation techniques, coupled with the widespread availability of digital computers, has paved the way for detailed examination of the behavior of loaded structures and machine parts of quite complex geometry. Numerous computer programs have been developed by workers in industry,

---

*Lecturer in Mechanical Engineering, University of Leeds, Leeds, England; National Research Council - National Aeronautics and Space Administration Senior Research Associate at the Lewis Research Center in 1976-1977.

research organizations, and institutions of higher education using the finite-element technique. Some of the more recently developed programs enable the study of situations in which three-dimensional loading pertains. Work to improve the accuracy of prediction in such situations, whilst most efficiently employing the facilities of the digital computer, progresses currently.

The program described here restricts itself to simple elements and axisymmetric loading situations. Unlike most finite-element programs, however, it can undertake a direct analysis of components that have relative interference. It is a frequent engineering practice to mate or prestress parts using shrink or press-fitting techniques. The analysis of such situations has, in general, been limited to axially invariant axisymmetric situations using, for example, the solution for a cylinder subjected to uniform internal or external pressure due to Lamé (ref. 1).

Wilson and Parsons (ref. 1) have described the approach adopted here to enable the solution of problems in which statically indeterminate conditions occur as a result of interference fits. They term the technique "differential displacements." It enables one node in a finite-element mesh to be displaced by a specified amount relative to another without having to know how the two nodes move in relation to some fixed coordinate system.

Since the finite elementry used in the computer program developed is quite conventional, the details are not dwelt upon. A number of texts (e.g., ref. 2) may be consulted for background information. The differential displacements approach and its incorporation into solution schemes for systems of equations is described. The computer program is presented as an appendix. As with many finite-element programs, it is long and its details are not discussed; its broad structure is, however, described. Fuller attention is paid to the input data for the program and the output to enable its easy use. An example of the inner ring of a roller bearing shrunk or press fitted onto a shaft is presented. The purpose of this example is not to highlight the technical aspects of the situation, but rather to give examples of the input and output.

The computer program presented uses some coding developed by Dr. B. Parsons, Reader in Mechanical Engineering, University of Leeds, England, whose encouragement and support are gratefully acknowledged.

SOME ASPECTS OF THE ANALYSIS

The finite-element types, designated 1, 2, and 0, that have been incorporated into the program are shown in figure 1. The geometrical restrictions of each element form are indicated. A linear displacement model within the elements was assumed, enabling the stiffness characteristics in matrix form to be determined by analytic integration of the strain energy integrals. The unknowns are normally node displacements in the radial and axial directions. Where a node displacement or a coordinate direction is prescribed, an unknown force results. The general equation enabling nodal distortions to be determined is

$$[K]\{U\} = \{F\} \tag{1}$$

where [K] is the overall stiffness matrix of the structure, $\{U\}$ is the vector of nodal displacements, and $\{F\}$ is the vector of nodal forces. The solution of the system of equations (1) may be achieved by many schemes. Direct methods of solution are now almost universally employed for solid mechanics problems, the method used here being that of Crout (ref. 3).

If it is known that two nodes in a finite-element structure have an effective prescribed movement relative to each other (obviously an interference), it is cumbersome with conventional finite elementry to analyze the situation. An influence coefficient approach can be adopted (e.g., ref. 4), but to use the differential displacement approach is more direct and less specific.

For example, consider, for convenience, consecutively numbered nodes 1, 2, 3, and 4 where it is known that nodes 2 and 3 have an interference $\{\delta\}$. With due regard for sign we may write,

$$\{U_3\} = \{U_2\} + \{\delta\} \tag{2}$$

and

$$\left.\begin{aligned}\{F_2\} &= \{F_2'\} - \{F_\delta\}\\ \{F_3\} &= \{F_3'\} + \{F_\delta\}\end{aligned}\right\} \tag{3}$$

where $\{F_2'\}$ and $\{F_3'\}$ are node forces due to external, inertial, and thermal loadings and $\{F_\delta\}$ is the interference force due to the differential displacement $\{\delta\}$ and is unknown.

Thus the equation of the form of (1) for the nodes may be adapted to

$$[K]\begin{Bmatrix} U_1 \\ U_2 \\ U_2 + \delta \\ U_4 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2' - F_\delta \\ F_3' + F_\delta \\ F_4 \end{Bmatrix} \tag{4}$$

Equation (4) can be reduced to give

$$[K_A]\begin{Bmatrix} U_1 \\ U_2 \\ U_4 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2' + F_3' \\ F_4 \end{Bmatrix} - [K_B]\begin{Bmatrix} 0 \\ \delta \\ 0 \end{Bmatrix} \qquad (5)$$

Having made this reduction, the set of equations (5) can now be solved for the unknown displacements, all other quantities being calculable. After the determination of the displacements, the set of equations (4) may be used to calculate the nodal forces due to interference $\{F_\delta\}$.

The differential displacements technique, already described in terms of four consecutively numbered nodes, two of which have relative interference, is amenable to programming on a digital computer for large systems of equations. Nodes having interference need not be numbered consecutively. The example presented in the report shows how the geometry of interferring nodes is described.

THE COMPUTER PROGRAM

The program listing is presented in appendix A. The program is written in Fortran and has been developed on a Univac 1100 machine with a Fortran V compiler. In appendix B the structure of the program is briefly described. To facilitate the use of the program, detailed attention will now be given to the required input and output.

Input Data

The required card input will be listed in sequence with comments on limitations. Some general remarks will first be made. Where detailed elucidation on some aspect of the input data is needed, this appears in appendix C.

The coordinate system adopted is shown in figure 2. For elements of types 1 and 0 the axial node positions (which are relative to node i) should be in accord with the sign convention of figure 2 (i.e., values to the left of node i being negative).

Any consistent set of units may be employed. The program output does not detail any units, but those actually employed could be indicated in the title.

The program arrangement permits multiple cases on one run with the same geometry but varying operating conditions, for example, loading or speed. In addition different geometries may be examined on the same run, though this is a less likely requirement.

The input requirements of finite-element computer programs are usually demanding in terms of punching of data. To simplify this aspect with the present program, it has been assumed that the most likely situation is that a single material (termed the "base" material) will be involved or will dominate. Other materials (termed "special"), unlimited in number, may be used in addition to the base material. However, when many special materials are used, the preparation of data cards will be disproportionately more time consuming than for a single material case.

The program includes a consideration of external, inertial, and thermal loadings. Material properties may be a function of temperature. The temperature distribution is an input requirement.

The limitations on the magnitude of certain variables indicated in the required input relate to the dimensioning of arrays. Alterations may be made as indicated in appendix C, where the storage requirement needed for operation on a Univac 1100 computer is also given.

It should also be noted that axisymmetric loading problems in which no interference fit occurs can also be treated with the program. The program variable for a parameter, on read-in, is indicated in brackets after its verbal description.

CARD A format (I2)

I2 number of geometries considered (IC)

The remaining cards are repeated for each geometry.

CARD B A title of up to 80 characters, commencing with a blank

CARD C format (I2)

I2 number of different cases for geometry under consideration (NOC)

CARD D format (6I4)

I4 number of elements (M)

I4 number of nodes ($\not> 200$) (N)

I4 maximum nodal number difference in any element ($\not> 19$) (See appendix C.) (NW)

I4 number of special material elements (NE)

I4 number of nodes constrained ($\not> 50$) (NCON)

I4 number of nodes with external forces applied (NFOR)

CARD E format (5E12.4)

E12.4	Poisson's ratio for base material	(PROP (1))
E12.4	Young's modulus for base material	(PROP (2))
E12.4	coefficient of expansion for base material	(PROP (3))
E12.4	density of base material	(PROP (4))
E12.4	reference temperature ($\nless 0$)	(PROP (5))

If any of the first four properties read in on this card are to be functions of the fifth, temperature, then any negative number should be placed in the appropriate format position. The reference temperature is that temperature the elements will be assured to take if the element temperature of CARD H is left blank. This reference temperature may be zero, which implies no relative thermal expansion for the element; that is, if the geometry read in is to suffer no thermal distortions, the reference temperature should be zero and apply to every element.

CARD F format (F10.0)

F10.0	rotational speed (rpm)	(W)

CARDS G, H

CARD G format (6I4)

I4	element number	(L)
I4	type of element	(NTYPE)
I4	number of node i	(NTRI (L, 1))
I4	number of node j	(NTRI (L, 2))
I4	number of node k	(NTRI (L, 3))
I4	number of node L (0 for a triangle)	(NTRI (L, 4))

CARD H format (6F10.0)

	Element type			
	1	2	0	
F10.0	r_i	r_i	r_i	(TRI (L, 1))
F10.0	r_j	$-r_j$	r_j	(TRI (L, 2))
F10.0	b_j	r_k	b_j	(TRI (L, 3))
F10.0	b_k	r_L	b_k	(TRI (L, 4))
F10.0	b_L	b	0	(TRI (L, 5))
F10.0	Element temperature ($\nless 0$)			(TRI (L, 6))

Figure 1 may be referred to for the meaning of symbols on cards G and H. If the element temperature is to equal the reference temperature of card E, a blank space may be left, otherwise the temperature of all elements must be indicated here. Note that if any temperatures are to be zero then the reference temperature must also be zero.

Cards G and H are repeated, consecutively, for each element. For type 1 rectangular elements only, element numbering need not be consecutive. In this case intermediate elements will be generated automatically. The automatic generation defines node numbers of elements by increasing the previous element values by unity; linear interpolation is used to determine r_i, r_j, and element temperature; and b_j, b_k, and b_L remain fixed. The automatic generation may be applied in either coordinate direction.

CARD I format (I4, 4E12.4) Only read if number of special material elements (NE) of card D is greater than zero

I4	number of special material element	(L)
E12.4	Poisson's ratio for special material	(TRI (L, 7))
E12.4	Young's modulus for special material	(TRI (L, 8))
E12.4	coefficient of expansion for special material	(TRI (L, 9))
E12.4	density of special element	(TRI (L, 10))

The properties should have a value appropriate to the element temperature of card E or card H. Card I should be repeated "NE times" corresponding to the number of special material elements.

CARDS J,K (For cases when properties of base material are temperature dependent)

CARD J format (I2)

I2	number of temperatures at which property value given (>10)	(IPROP (I))

CARD K format (F10.0,E12.4)

F10.0	temperature	(U (J))
E12.4	value of property at this temperature	(U (K))

Card K is repeated for the number of temperature values indicated on card J, with temperature values increasing.

Cards J and K are repeated for any property of card E that was read in as a negative number. The cards should be presented for properties in the same sequence as indicated for card E (i.e., Poisson's ratio, Young's modulus, coefficient of expansion, density) and omitted entirely if a property is not temperature dependent. The range of temperatures on card K must encompass the full range of element temperatures.

CARD L format (3I4,2F10.4)

I4	number of node	(NUV (I, 1))
I4	= 1 if node movement constrained radially, otherwise 0	(NUV (I, 2))
I4	= 1 if node movement constrained axially, otherwise 0	(NUV (I, 3))
F10.4	amount of radial movement if constrained	(ANUV (I, 1))
F10.4	amount of axial movement if constrained	(ANUV (I, 2))

Normally the constrained movement will have zero value. Nodes on the axis of revolution must be restrained to zero radial movement. Cases of axial symmetry can be accommodated considering only half the problem by using appropriate axial constraint. Card L should be read in for NCON nodes ($\ngtr 50$), as indicated on card D, and omitted if no nodes are constrained.

CARD M format (I4, 2E12.4)

I4	number of node	(L)
E12.4	applied radial force	(PROP (1))
E12.4	applied axial force	(PROP (2))

Card M should be repeated for NFOR nodes as indicated on card D and omitted if no nodes have applied forces. An applied distributed loading must be appropriately divided between nodes, the force at each being that for 2π radians (see appendix C.)

CARD N format (I2)

I2	= 1 if there is any interference, otherwise 0	(IDD)

If card N reads in the value 0, the following course is adopted: if the number of cases for the geometry under consideration is 1 (NOC of card C), then the data terminate, unless another geometry is to be considered, in which case the card sequence returns to card B and begins again. If there are (NOC - 1) cases for the same geometry still to be dealt with, then the following cards should be read for each of these cases: CARD F, CARD M, and CARD N. This sequence allows applied speed and load to be varied. If card N should now take the value 1, then the case should proceed as below.

If card N reads in the value 1 (an interference fit), the following card sequence should be adopted:

CARD O format (I5)

I5	number of node pairs with relative interference	(NODD)

CARD P format (1X,I6,5X,I6,5X,I6,E11.4,5X,I6,E11.4)

I6	lower numbered node of a pair in interference	(J)
I6	higher numbered node of a pair in interference	(K)
I6	= 1 if radial interference, otherwise 0	(IR)
E11.4	amount of any radial interference	(DR)
I6	= 1 if any axial interference, otherwise 0	(12)
E11.4	amount of any axial interference	(D2)

Card P should be repeated for the NODD pairs of nodes indicated on card 0. The sign of interference, radial or axial, should be in accord with the sign convention of figure 2, working from the lower to the higher numbered node. (See the example of appendix D.) It will be convenient in interpreting the output to have all the higher numbered nodes of pairs in radial interference "nearer" the axis of revolution. This scheme has been adopted in the development of the program and is recommended.

CARD Q format (I4)

I4	= 1 if radial interference pressures are to be calculated	(IND)

If card Q reads in 1, then the pressure distribution at a fixed radial location where interference occurs will be determined. Card Q may only be read in with a value of 1 if <u>all</u> the nodes having interference are at the same radius and are axially adjacent. If this card reads in 0, go to card T.

CARD R format (I4, E12.4)

I4	number of node pairs in interference ($\ngtr 50$)	(NP)
E12.4	constant radius of interference fit	(RAD)

CARD S format (I4, E12.4)

I4	number of the node in pair nearer the axis of revolution	(IP (I))
E12.4	axial distance to next node in sequence	AL (I)

Read in the node numbers nearer the axis of revolution in sequence, moving axially in either direction. The axial distance between nodes should be kept positive. The pressure distribution is the same for both components involved in the interference. The value of axial distance to the next node for the last node is immaterial.

Card S should be repeated for the number of node pairs (NP) indicated in card R.

CARD T format (I2)

I2	= 0 if no axial slip to be allowed automatically at nodes with radial interference, otherwise 1	(ISLP)

This card may only read in a value of 1 if it is permissible to do so with card Q. If card T reads in a value of 0, the following card sequence should be adopted for remaining cases with the same geometry: CARD F, CARDS L,M,N, et seq. Alternatively, if there are no more cases for the particular geometry, the data sequence terminates or returns to card B for a new geometry.

If card T reads in a value of 1, the following card should be inputted:

CARD U format (F5.0)

F5.0	coefficient of friction	(COF)

See appendix C for an explanation of the slip calculations. After this card, the data sequence should be the same as if card T had read in 0 (see above).

Although it is very convenient and saving in time to run multiple cases, particularly with the same geometry, it will be seen that the data is accordingly complicated. It is therefore recommended that new users of the program adopt single case runs initially. The data card series will then be as in table I.

Output

The program output is virtually self-explanatory. An example is to be found in appendix D. As well as outputting calculated information, all the input data are also presented. Thus, the first part of the output gives the data fed to the program on cards B, D through M, and P, as appropriate. Calculations to set element temperatures and physical properties are carried out on the basis of the above input but details are not outputted. In the case of multiple runs with the same geometry only the parameters which can be changed are re-presented.

Results calculated by the program and presented in the output are

(a) Nodal deformations

(b) Element stresses at the midpoint of the element. Stresses given are radial, tangential, axial, and shear together with the coordinates of the element midpoint with respect to node i. Positive stresses are tensile.

(c) If there is an interference fit, the forces due to this fit radially and axially acting on the higher numbered of each node in the pairs involved. The sign of these forces is opposite to the sign convention of figure 2. Thus in the example of appendix D, where all the higher numbered nodes are nearer the axis of revolution, the positive radial forces indicate that the two components are still entirely in interference. These forces are per radian unlike the input forces, which are for an extent of 2π radians.

(d) If the option is so chosen, the pressure distribution at the position of radial interference. Note that although the interference force output may indicate that all nodes of an interference remain in contact, small negative interference pressures may sometimes occur. This is because of the assumption that the pressure varies linearly between nodes.

(e) If slip between nodes in radial interference is to be accommodated automatically, the coefficient of friction and details of which nodes are allowed to slip on each cycle. On completion of slip the final nodal deformations and element stresses are given.

CONCLUDING REMARKS

A finite-element computer program that can directly analyze the behavior of axisymmetrically loaded components having an interference fit has been described. External, inertial, and thermal loads can be accommodated as well as loading due to the interference fit. In addition components constructed of a number of materials, the physical properties of which may be temperature dependent, can be dealt with. The program code is presented together with an example data input and program output.

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, February 15, 1977,
505-04

APPENDIX A

THE COMPUTER PROGRAM LISTING

```
C     AXISYMMETRIC FINITE  ELEMENT  PROGRAM  WITH  VARIABLE  MATERIALS ,
C     TEMPERATURE  EFFECTS  AND  INERTIAL  LOADING (CMT)
C     DIFFERENTIAL DISPLACEMENTS INCLUDED
      DIMENSION ALLK(400,41),TRI(200,10),NTRI(200,4),NODIS(400),FW(200),
     1PROP(5), U(400), NUV(50,3), ANUV(50,2),TU(400)
     2,DEL(400),DD(400),NDD(400),MOVE(400),ALLK1(400,41)
     3,ISTR(50,4),STR(50,2),NODIS1(400),IP(50),AL(50)
     4,IPROP(5)
      INTEGER  REPT
      COMMON/BLOCK1/TEMP,PR,YM,ALPHA,RHO
      COMMON/BLOCK2/MORD,NW221,NW22,NW
      IC=0
      READ(5,100)ICASE
1     IC=IC+1
      WRITE(6,150)
      READ(5,101)
      WRITE(6,101)
      WRITE(6,152)
      READ(5,100)NCC
      IC1=1
      READ(5,102)M,N,NW,NE,NCON,NFOR
      WRITE(6,154)M,N,NW,NE,NCON,NFOR
      WRITE(6,152)
      MORD = 2*N
      NW22 = 2*NW+2
      NW221 = NW22+1
      READ(5,103)(PROP(I),I=1,5)
      READ(5,104)W
      WRITE(6,155)(PROP(I),I=1,5),W
      WRITE(6,152)
      W = W*0.10471976
      CALL SETZ1(TRI,200,10)
      CALL SETZ3(NTRI,200,4)
      WRITE(6,156)
      WRITE(6,152)
      LST=0
      DO 2 I=1,M
      READ(5,102)L,NTYPE,(NTRI(L,J),J=1,4)
      READ(5,105)(TRI(L,J),J=1,6)
      L1=LST+1
      IF(L1.EQ.L)GOTO30
      L2=L-1
      L3=L-LST
      ZZ=(TRI(L,1)-TRI(LST,1))/L3
      YY=(TRI(L,6)-TRI(LST,6))/L3
      DO31 J1=L1,L2
      NTRI(J1,1)=NTRI(LST,1)+1
      NTRI(J1,2)=NTRI(LST,2)+1
      NTRI(J1,3)=NTRI(LST,3)+1
      NTRI(J1,4)=NTRI(LST,4)+1
      TRI(J1,3)=TRI(LST,3)
```

```
      TRI(J1,4)=TRI(LST,4)
      TRI(J1,5)=TRI(LST,5)
      TRI(J1,6)=TRI(LST,6)+YY
      IF(ABS(ZZ).GT.1.E-05)GOTO32
      TRI(J1,1)=TRI(LST,1)
      TRI(J1,2)=TRI(LST,2)
      GOTO33
   32 IF(ZZ.LT.0)GOTO35
      TRI(J1,1)=TRI(LST,2)
      TRI(J1,2)=TRI(LST,2)+ZZ
      GOTO33
   35 TRI(J1,1)=TRI(LST,1)+ZZ
      TRI(J1,2)=TRI(LST,1)
   33 WRITE(6,157)J1,NTYPE,(NTRI(J1,J),J=1,4),(TRI(J1,J),J=1,6)
   31 LST=LST+1
   30 WRITE(6,157)L,NTYPE,(NTRI(L,J),J=1,4),(TRI(L,J),J=1,6)
      IF(L.EQ.M)GOTO34
      LST=L
2     CONTINUE
   34 WRITE(6,152)
      IF(NE.LT.1)GOTO10
      WRITE(6,158)
      WRITE(6,152)
      DO3 I=1,NF
      READ(5,106) L,(TRI(L,J),J=7,10)
      WRITE(6,159)L,(TRI(L,J),J=7,10)
3     CONTINUE
      WRITE(6,152)
   10 CONTINUE
      DO4 I=1,5
      IF(PROP(I).GT. -1.E-10)GOTO 4
      READ(5,100)IPROP(I)
      WRITE(6,160)
      WRITE(6,151)
      IF(I.EQ.1)WRITE(6,161)
      IF(I.EQ.2)WRITE(6,162)
      IF(I.EQ.3)WRITE(6,163)
      IF(I.EQ.4)WRITE(6,164)
      N1 = (I-1)*10 + 1
      N2=N1-1+IPROP(I)
      DO5 J=N1,N2
      K = J+40
      READ(5,107)U(J),U(K)
    5 WRITE(6,107) U(J),U(K)
4     CONTINUE
      WRITE(6,152)
      DO6  I=1,M
      IF(TRI(I,6).GT.1.E-10) GOTO 7
      TRI(I,6) = PROP(5)
7     DO8  J=1,4
      K = J+6
      IF(TRI(I,K).GT.1.E-10) GO TO 8
      TRI(I,K) = PROP(J)
      IF(PROP(J).GT.-1.E-10) GOTO 8
      NS=IPROP(J)
      DO 9 K1=1,NS
      N1=(J-1)*10+K1
      Q = TRI(I,6)-U(N1)
      IF(ABS(Q).LT.1.E-05)TRI(I,K)=U(N1+40)
```

```
      IF(ABS(Q).LT.1.E-05)GOTO8
      IF(C.GT.0.0)GOTO 9
      M1 = N1 + 39
      M2 = M1 + 1
      M3 = N1 - 1
      TRI(I,K)=U(M1) +(U(M2) -U(M1))*(TRI(I,6) -U(M3))/(U(N1) -U(M3))
      GOTO8
9     CONTINUE
8     CONTINUE
6     CONTINUE
   36 CONTINUE
      CALL SETZ1(ALLK,400,41)
      CALL SETZ2(FW,200)
      CALL SETZ2(U,400)
      CALL SETZ4(NODIS,400)
      DO 11 L=1,M
      TEMP = TRI(L,6)
      PR = TRI(L,7)
      YM = TRI(L,8)
      ALPHA = TRI(L,9)
      RHO = TRI(L,10)
      CALL RECT(L,FW,NTRI,ALLK,U,TRI)
11    CONTINUE
      REPT=1
      DO 12 I=1,N
      J = 2*I-1
      J1 = J +1
      ALLK(J,NW221) = U(J) + FW(I)*W**2
      ALLK(J1,NW221) = U(J1)
      TU(J)=U(J)
      TU(J1)=U(J1)
      U(J) = 0.0
12    U(J1) = 0.0
      WRITE(6,176)
      WRITE(6,165)
      IF(NCON.EQ.0)GOTO80
      DO13 I=1,NCON
      READ(5,108)(NUV(I,J),J=1,3),(ANUV(I,J),J=1,2)
      WRITE(6,166)(NUV(I,J),J=1,3),(ANUV(I,J),J=1,2)
      NOD = 2*NUV(I,1) -1
      U(NOD) = ANUV(I,1)
      IF(NUV(I,2).GT.NODIS(NOD))NODIS(NOD)=NUV(I,2)
      NOD = NOD + 1
      U(NOD) = ANUV(I,2)
13    NODIS(NOD) = NUV(I,3)
80    CONTINUE
      WRITE(6,152)
14    WRITE(6,177)
      WRITE(6,167)
      IF(NFOR.EQ.0)GOTO81
      DO15 I=1,NFOR
      READ(5,106)L,(PROP(J),J=1,2)
      WRITE(6,168)L,(PROP(J),J=1,2)
      K = 2*L - 1
      KK = K +1
      ALLK(K,NW221)=ALLK(K,NW221) +PROP(1)*0.159154943
15    ALLK (KK,NW221) = ALLK(KK,NW221) + PROP(2)* 0.159154943
81    CONTINUE
      IDD=0
```

```
      READ(5,100)IDD
      DO20 I=1,MORD
      DO20 J=1,NW221
 20   ALLK1(I,J)=ALLK(I,J)
      DO37 I=1,400
 37   NODIS1(I)=NODIS(I)
      KT=MORD
      KCT=0
      ICY=0
      IWP=0
      GOTO38
 39   DO40 I=1,400
      NODIS(I)=NODIS1(I)
 40   U(I)=0
      DO41 I=1,MORD
      DO41 J=1,NW221
 41   ALLK(I,J)=ALLK1(I,J)
      CALL DIFDIS(KT,ALLK,NODIS,MOVE,NDD,DD,DEL,ISTR,STR,NORD,KCT)
      GOTO42
 38   IF(IDD.GT.0)
     1CALL DIFDIS(KT,ALLK,NODIS,MOVE,NDD,DD,DEL,ISTR,STR,NORD,KCT)
 42   CALL BCROUT(ALLK,U,MORD,REPT,NW,NODIS)
      IF(IDD.GT.0) GOTO22
      WRITE(6,150)
      WRITE(6,178)
      WRITE(6,169)
      DO16 I=1,N
      M1=2*I-1
      M2=M1+1
 16   WRITE(6,168)I,U(M1),U(M2)
      WRITE(6,152)
      WRITE(6,179)
      WRITE(6,170)
      TOTA=0.0
      ST=0.0
      DO17  L=1,M
      TEMP=TRI(L,6)
      PR=TRI(L,7)
      YM=TRI(L,8)
      ALPHA=TRI(L,9)
 17   CALL RECTST(L,ST,TOTA,U,TRI,NTRI)
      ST=ST/TOTA
      WRITE(6,171) ST
      IF(IC1.EQ.NOC)GOTO 18
      IC1 = IC1 + 1
      READ(5,104)W
      WRITE(6,172)W
      W=W*0.10471976
      DO19 I=1,N
      J=2*I-1
      J1=J+1
      ALLK(J,NW221)=FW(I)*W**2+TU(J)
 19   ALLK(J1,NW221)=TU(J1)
      REPT= -1
      WRITE(6,152)
      GOTO 14
 22   KT=MORD
      DO23II=1,KT
      I=KT-II+1
```

```fortran
      IF (NDD(I).GE.0) GOTO24
      I1=-NDD(I)
      I1=MOVE(I1)
      DEL(I)=DEL(I)+U(I1)
      GOTO23
24    I1=MOVE(I)
      DEL(I)=U(I1)
23    CONTINUE
46    CONTINUE
      IF(ICY.NE.0)GOTO45
      WRITE(6,150)
      WRITE(6,178)
      WRITE(6,169)
      DO25 I=1,N
      NOD=2*I-1
25    WRITE(6,168) I,DEL(NOD),DEL(NOD+1)
      WRITE(6,152)
      WRITE(6,179)
      WRITE(6,170)
      TOTA=0.0
      ST=0.0
      DO29  L=1,M
      TEMP=TRI(L,6)
      PR=TRI(L,7)
      YM=TRI(L,8)
      ALPHA=TRI(L,9)
29    CALL RECTST(L,ST,TOTA,DEL,TRI,NTRI)
      ST=ST/TOTA
      WRITE(6,171) ST
45    CONTINUE
      IF(IWR.NE.0)GOTO43
      CALL SETZ2(DD,400)
      DO26 I=1,KT
      MC1=I+1-NW22
      MC2=I+NW22-1
      IF (MC1.GT.1)J1=MC1
      IF (MC1.LE.1)J1=1
      IF (MC2.LT.MORD)J2=MC2
      IF(MC2.GE.MORD)J2=MORD
      DO27 J=J1,J2
      IF(I.LT.J)DD(I)=DD(I)+ALLK1(I,J-I+1)*DEL(J)
27    IF(I.GE.J)DD(I)=DD(I)+ALLK1(J,I-J+1)*DEL(J)
26    CONTINUE
      WRITE(6,152)
      WRITE(6,173)
      WRITE(6,152)
      DO28 I=1,N
      MC1=2*I-1
      MC2=MC1+1
      IF (NDD(MC1).GE.0.AND.NDD(MC2).GE.0)GOTO28
      IF (NDD(MC1).LT.0)DD(MC1)=ALLK1(MC1,NW221)-DD(MC1)
      IF (NDD(MC1).GE.0)DD(MC1)=0.
      IF (NDD(MC2).LT.0)DD(MC2)=ALLK1(MC2,NW221)-DD(MC2)
      IF(NDD(MC2).GE.0)DD(MC2)=0.
      IF(NODIS1(MC1).NE.0)DD(MC1)=0
      IF(NODIS1(MC2).NE.0)DD(MC2)=0
      WRITE(6,168)I,DD(MC1),DD(MC2)
28    CONTINUE
      CALL PRESS(DD,IP,AL,NP,RAD,KCT)
```

```
      IF(KCT.EQ.0)READ(5,100)ISLP
      IF(ISLP.EQ.0)GOTO43
      KCK=0
      DO44 I=1,NP
      MC1=2*IP(I)-1
44    IF(DD(MC1).LT.0)KCK=1
      IF(KCK.EQ.0)GOTO47
      WRITE(6,175)
      IF(KCT.EQ.0)READ(5,104)COF
      GOTO43
47    CALL SLIP(ISTR,DD,NODIS1,KCT,COF,ICY,IP,NP)
      IF(KCT.EQ.1)GOTO39
      IF(ICY.EQ.1)GOTO43
      IWR=1
      ICY=0
      GOTO46
43    IF(IC1.EQ.NOC)GOTO18
      IC1 = IC1 + 1
      READ(5,104)W
      WRITE(6,174)W
      W=W*0.10471976
      WRITE(6,152)
      GOTO36
18    IF(IC.LT.ICASE)GOTO 1
  100 FORMAT(I2)
  101 FORMAT(80H
     1
  102 FORMAT(6I4)
  103 FORMAT(5E12.4)
  104 FORMAT(F10.0)
  105 FORMAT(6F10.0)
  106 FORMAT(I4,4E12.4)
  107 FORMAT(F10.0,E12.4)
  108 FORMAT(3I4,2F10.4)
  150 FORMAT(1H1)
  151 FORMAT(1H )
  152 FORMAT(1H0)
  153 FORMAT(1H+)
  154 FORMAT(1X,'M=',I4,5X,'N=',I4,5X,'NW=',I4,5X,'NE=',I4,5X,
     1'NCON=',I4,5X,'NFOR=',I4)
  155 FORMAT(1X,'PR=',E12.4,5X,'YM=',E12.4,5X,'COEX=',E12.4,5X,
     1'RHO=',E12.4,5X,'TEMP=',E12.4,5X,'SPEED(RPM)=',F10.2)
  156 FORMAT(1X,'ELEMENT',4X,'TYPE',7X,'I',7X,'J',7X,'K',7X,'L',
     110X,'RI',10X,'RJ',7X,'RKOBJ',7X,'RLO K',8X,'BOBL',8X,'TEMP')
  157 FORMAT(6I8,6E12.4)
  158 FORMAT(1X,'SPECIAL ELEMENTS  '/1H ,'ELEMENT',10X,'PR',10X,'YM',
     18X,'COEX',9X,'RHO')
  159 FORMAT(I8,4E12.4)
  160 FORMAT(1X,'BASE MATERIAL PROPERTY VARIATIONS WITH TEMPERATURE')
  161 FORMAT(6X,'TEMP',10X,'PR')
  162 FORMAT(6X,'TEMP',10X,'YM')
  163 FORMAT(6X,'TEMP',8X,'COEX')
  164 FORMAT(6X,'TEMP',9X,'RHO')
  165 FORMAT(1X,'NODE' ,6X,'IU',6X,'IV',8X,'UU',8X,'VV')
  166 FORMAT(I5,2I8,2F10.4)
  167 FORMAT(1X, 'NODE', 10X,'FR', 10X, 'FZ')
  168 FORMAT(I5,2E12.4)
  169 FORMAT(1H ,'NODE',10X,'UR',10X,'UZ')
  170 FORMAT(1X,'ELMT',4X,'R STRESS' ,4X,'T STRESS',3X,'AX STRESS',3X,
```

```
     1'SH STRESS',9X,'R',9X,'Z')
  171 FORMAT(1H  ,'AVERAGE TANGENTIAL  STRESS=',E12.4)
  172 FORMAT(1H1, 'NEW  SPEED AND/OR  FORCES.W=',F10.2)
  173 FORMAT(1H0,'INTERFERENCE FORCE VECTORS'/1H ,'NODE',6X,'RADIAL',
     17X,'AXIAL')
  174 FORMAT(1H1,'SAME GEOMETRY.W=',F10.2)
  175 FORMAT(1H0,'LOSS OF RADIAL INTERFERENCE')
  176 FORMAT(1X,'NODAL RESTRAINTS')
  177 FORMAT(1X,'APPLIED NODAL FORCES')
  178 FORMAT(1X,'NODAL DEFORMATIONS')
  179 FORMAT(1X,'ELEMENT STRESSES')
      STOP
      END


      SUBROUTINE AINV(RI,RJ,RK,RL,B,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
C     INVERTS THE ARRAY A
      DIMENSION  A(8,8),TRI(200,10)
      CALL SETZ1(A,8,8)
      IF(TRI(L,2).GE.0.0)GOTO 1
      RI=TRI(L,1)
      RJ= -TRI(L,2)
      RK=TRI(L,3)
      RL=TRI(L,4)
      B=TRI(L,5)
      BJ=0.
      AI= 0.5*(RI+RK)
      AJ= 0.5*(RJ+RL)
      A(2,1)= 1/(RI-RJ)
      A(6,5)= A(2,1)
      A(2,2)= -A(2,1)
      A(6,6)= A(2,2)
      A(1,1)= RJ*A(2,2)
      A(5,5)= A(1,1)
      A(1,2)= RI*A(2,1)
      A(5,6)= A(1,2)
      A(4,1)= A(2,2)/B
      A(8,5)= A(4,1)
      A(4,2)= -A(4,1)
      A(8,6)= A(4,2)
      A(3,1)= RJ*A(4,2)
      A(7,5)= A(3,1)
      A(3,2)= RI*A(4,1)
      A(7,6)= A(3,2)
      A(4,3)= 1/(B*(RK-RL))
      A(8,7)= A(4,3)
      A(4,4)= -A(4,3)
      A(8,8)= A(4,4)
      A(3,3)= RL*A(4,4)
      A(7,7)= A(3,3)
      A(3,4)= RK*A(4,3)
      A(7,8)= A(3,4)
      DET= AI-AJ
      RETURN
1     AI=TRI(L,1)
      AJ=TRI(L,2)
      DET=AI-AJ
      BJ=TRI(L,3)
      BK=TRI(L,4)
```

```
      BL=TRI(L,5)
      B= (BK+BL-BJ)/2
      D1= AJ-AI
      D2= D1*(BJ-BL)
      A(1,1)= AJ/D1
      A(5,5)= A(1,1)
      A(1,2)= AI*BL/D2
      A(5,6)= A(1,2)
      A(1,4)= -AI*BJ/D2
      A(5,8)= A(1,4)
      A(2,1)= -1/D1
      A(6,5)= A(2,1)
      A(2,2)= -BL/D2
      A(6,6)= A(2,2)
      A(2,4)= BJ/D2
      A(6,8)= A(2,4)
      A(4,1)= 1/(BK*D1)
      A(8,5)= A(4,1)
      A(4,3)= -A(4,1)
      A(8,7)= -A(4,1)
      A(3,1)= -AJ*A(4,1)
      A(7,5)= A(3,1)
      A(3,3)= -A(3,1)
      A(7,7)= -A(3,1)
      A(4,2)= 1/D2
      A(8,6)= A(4,2)
      A(4,4)= -A(4,2)
      A(8,8)= -A(4,2)
      A(3,4)= AI/D2
      A(7,8)= A(3,4)
      A(3,2)= -A(3,4)
      A(7,6)= -A(3,4)
      RETURN
      END


      SUBROUTINE BCROUT(D,X,N,REPT,NW,NODIS)
C   APROCEDURE TO APPLY CROUT REDUCTION TO A BANDED MATRIX
C   D WHICH HOLDS THE UPPER TRIANGLE OF THE SYMMETRICAL
C   BANDED MATRIX A SO THAT D(I,J-I+I)=A(I,J).THE LAST COLUMN
C   (D(I,J-I+1) HOLDS THE VECTOR B FOR THE SET OF EQLATIONS A.X=B.
C   ON EXIT THE VECTOR X HOLDS THE SOLUTION (X) TO THESE EQUATIONS.
C   THE REDUCTION IS ARRANGED SO THAT IF CERTAIN SETS OF EQUATIONS
C   MUST NOT BE SOLVED.(I.E. NODIS(I)=1) THEN THE APPROPRIATE ROWS
C   AND COLUMNS ARE NEGLECTED
      DIMENSION D(400,41),X(400),NODIS(400)
      INTEGER REPT,CT,C1
      NW2=2*NW+1
      CT=0
91    CT=CT+1
      IF ((NODIS(CT)-0.1).GT.0) GO TO 91
      IF (REPT.LT.1) GOTO 25
      DO 92 I=1,NW2
      IF((NODIS(CT+I)-0.1).GT.0) GOTO 92
      D(CT,I+1) = D(CT,I+1)/D(CT,1)
92    CONTINUE
      C1=CT+1
      DO 93 J=C1,N
```

```
      IF ((NODIS(J )-0.1).GT.0) GO TO 93
      SUM=0.
      L1 =J-1
      K1 =J-NW2
      IF (K1.LT.CT) K1=CT
      DO94 K=K1,L1
      IF ((NODIS(K )-0.1).GT.0) GO TO 94
      SUM =SUM+(D(K,J-K+1)**2)*D(K,1)
94    CONTINUE
      D(J,1) =D(J,1)-SUM
      IF (J.EQ.N) GOTO 25
      L2=J+1
      N2 =J+NW2
      IF (N2.GE.N) N2=N
      DO 95 I=L2,N2
      IF ((NODIS(I )-0.1).GT.0) GOTO 95
      SUM =0.
      K1 =I-NW2-1
      IF (K1.LT.CT) K1=CT
      DO 96 K=K1,L1
      IF ((I-K-0.1).GT.NW2) GOTO 96
      IF ((NODIS(K)-0.1).GT.0) GOTO 96
      SUM =SUM+(D(K,I-K+1)*D(K,J-K+1)*D(K,1))
96    CONTINUE
      D(J,I-J+1) = (D(J,I-J+1)-SUM)/D(J,1)
95    CONTINUE
93    CONTINUE
25    NW3 =2*NW+3
      D(CT,NW3)=D(CT,NW3)/D(CT,1)
      C1=CT+1
      DO 97 I=C1,N
      IF ((NODIS(I) -0.1).GT.0) GOTO 97
      SUM =0.
      L1=I-1
      K1=I-NW2
      IF (K1.LT.CT) K1=CT
      DO 98 K=K1,L1
      IF ((NODIS( K)-0.1).GT.0) GOTO 98
      SUM =SUM+(D(K,I-K+1)*D(K,1)*D(K,NW3))
98    CONTINUE
      D(I,NW3) =(D(I,NW3)-SUM)/D(I,1)
97    CONTINUE
      IF((NODIS(N)-0.1).GT.0)GOTO26
      X(N)=D(N,NW3)
26    L1=N-1
      DO 99 I=1,L1
      L2=(N-I)+1
      SUM=0.0
      L3=N-I
      IF ((NODIS(L3)-0.1).GT.0) GOTO 99
      N2=L3+NW2
      IF (N2.GT.N) N2 =N
      DO 90 K=L2,N2
      IF ((NODIS (K)-0.1).GT.0) GOTO 90
      SUM =SUM+(D(L3,K-L3+1)*X(K))
90    CONTINUE
      X(L3)=D(L3,NW3)-SUM
99    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DIFDIS
     1(KT,D,NODIS,MOVE,NDD,DD,DEL,ISTR,STR,NODD,KCT)
      DIMENSION    D(400,41), NODIS(400), MOVE(400), NDD(400),
     1DD(400), DEL(400)
     2,ISTR(50,4),STR(50,2)
      COMMON/BLOCK2/MORD,NW221,NW22,NW
      CALL SETZ2(DEL,400)
      CALL SETZ2(DD,400)
      CALL SETZ4(NDD,400)
      CALL SETZ4(MOVE,400)
      NW2 = NW22-1
      DO20 I = 1,KT
20    DD(I) = D (I,NW22+1)
      IF(KCT.NE.0)GOTO90
      READ(5,1) NODD
      WRITE(6,2)
      WRITE(6,251)
      DO21 I = 1,NODD
      READ(5,252)J,K,IR,DR,IZ,DZ
      WRITE(6,252)J,K,IR,DR,IZ,DZ
      ISTR(I,1)=J
      ISTR(I,2)=K
      ISTR(I,3)=IR
      ISTR(I,4)=IZ
      STR(I,1)=DR
      STR(I,2)=DZ
      NDD(2*K-1) = (-2*J+1)*IR
      NDD(2*K) = (-2*J)*IZ
      DEL(2*K-1) = DR
21    DEL(2*K) = DZ
      GOTO91
90    DO92 I=1,NODD
      J=ISTR(I,1)
      K=ISTR(I,2)
      DR=STR(I,1)
      DZ=STR(I,2)
      IR=ISTR(I,3)
      IZ=ISTR(I,4)
      NDD(2*K-1)=(-2*J+1)*IR
      NDD(2*K)=(-2*J)*IZ
      DEL(2*K-1)=DR
92    DEL(2*K)=DZ
91    II=0
      DO22 I = 1,KT
      MOVE(I) = I
      IF (NDD(I).LT.0) GOTO 70
      II = II+1
      MOVE(I) = II
70    MC1 = I-NW2
      MC2=I+NW2
      IF (MC1.GT.1) J1 = MC1
      IF (MC1.LE.1) J1 = 1
      IF (MC2.LT.KT) J2 = MC2
      IF (MC2.GE.KT) J2 = KT
      DO23 J = J1,J2
      IF(NDD (J).GE.0) GOTO23
      IF (J.GE.I) DD(I) = DD(I)-DEL(J)*D(I,J-I+1)
      IF (J.LT.I) DD(I) = DD(I)-DEL(J)*D(J,I-J+1)
23    CONTINUE
```

```
 22     CONTINUE
        DO24 I = 1,KT
        IF(NDD(I).LT.0) GOTO71
        IK = MOVE(I)
        NFI = 0
        GOTO72
 71     IK = -NDD(I)
        NFI=1
        IK = MOVE (IK)
 72     J1 = I
        MC2 = I+NW2
        IF (MC2.LT.KT) J2 = MC2
        IF (MC2.GE.KT) J2 = KT
        DO 25 J = J1,J2
        IF(NDD(J).LT.0) GOTO74
        JK=MOVE(J)
        JJ = JK
        NFIJ = 0
        NFJ = 0
        GOTO73
 74     JK = -NDD(J)
        JK = MOVE (JK)
        NFJ = 1
        NFIJ = NFI*NFJ
 73     MC1 = NFJ+NFI-NFIJ
        IF (IK.GT.JK)D(JK,IK-JK+1)=D(I,J-I+1)+MC1*D(JK,IK-JK+1)
        IF(IK.LE.JK)D(IK,JK-IK+1)=D(I,J-I+1)+MC1*D(IK,JK-IK+1)
 25     CONTINUE
        J1=JJ+1
        J2=IK+NW2
        IF(J1.GT.J2)GOTO75
        DO26 J=J1,J2
 26     D(IK,J-IK+1) =0
 75     CONTINUE
        DD(IK)=DD(I)+NFI*DD(IK)
        NODIS(IK)=NODIS(I)+NFI*NODIS(IK)
 24     CONTINUE
        I=MOVE(KT)+1
        DO27 J=I,KT
        DD(J)=0.
        NODIS(J)=0
        DO28 JJ=1,NW22
 28     D(J,JJ)=0.
 27     CONTINUE
        DO29 I=1,KT
 29     D(I,NW221)=DD(I)
        KT=MOVE(KT)
        WRITE(6,2)
  251 FORMAT(1X,'DIFFERENTIAL DISPLACEMENTS'/1X ,'NODE A',5X,'NODE B',9X
     1,'IR',9X,'DR',9X,'IZ',9X,'DZ')
  252 FORMAT(1X,I6,5X,I6,5X,I6,E11.4,5X,I6,E11.4)
 1      FORMAT(5I5)
 2      FORMAT(1H0)
        RETURN
        END
```

```
      SUBROUTINE PRESS(DD,IP,AL,NP,RAD,KCT)
      DIMENSION DD(400),P(50),Z(50),B(50),IP(50),AL(50),A(50,3)
      CALL SETZ1(A,50,3)
      CALL SETZ2(B,50)
      CALL SETZ2(P,50)
      CALL SETZ2(Z,50)
      WRITE(6,5)
      IF(KCT.EQ.1)GOTO24
      READ(5,1)IND
      IF(IND.EQ.0)GOTO10
      READ(5,1)NP,RAD
      DO11 I=1,NP
   11 READ(5,1)IP(I),AL(I)
24    DO12 I=1,NP
      MC1=2*IP(I)-1
      B(I)=9*DD(MC1)/RAD
      IF(I.EQ.1.OR.I.EQ.NP)GOTO13
      A(I,1)=AL(I-1)
      A(I,2)=3*(AL(I-1)+AL(I))
      A(I,3)=AL(I)
      GOTO14
   13 IF(I.EQ.NP)GOTO15
      A(I,2)=3*AL(I)
      A(I,3)=AL(I)
      GOTO14
   15 A(I,1)=AL(I-1)
      A(I,2)=3*AL(I-1)
   14 CONTINUE
   12 CONTINUE
      ICT=0
      RF=1.5
      TOL=0.00001
   16 IF(ICT.GT.500)GOTO17
      MK=0
      ICT=ICT+1
      DO18 I=1,NP
      Z(I)=P(I)
      IF(I.EQ.1.OR.I.EQ.NP)GOTO19
      P(I)=(B(I)-A(I,1)*P(I-1)-A(I,3)*P(I+1))/A(I,2)
      GOTO20
   19 IF(I.EQ.1)P(1)=(B(1)-A(1,3)*P(2))/A(1,2)
      IF(I.EQ.NP)P(NP)=(B(NP)-A(NP,1)*P(NP-1))/A(NP,2)
   20 P(I)=Z(I)+RF*(P(I)-Z(I))
      IF(MK.GT.0)GOTO18
      IF(Z(I).EQ.0)GOTO21
      C1=(P(I)-Z(I))/Z(I)
      C1=ABS(C1)
      IF(C1.GT.TOL)GOTO21
      IF(I.EQ.NP)GOTO22
      GOTO18
   21 MK=1
   18 IF(I.EQ.NP)GOTO16
   22 DO23 I=1,NP
   23 WRITE(6,6)IP(I),P(I)
      GOTO10
   17 WRITE(6,7)
      GOTO10
1     FORMAT(I4,E12.4)
5     FORMAT(1H0,'NODE',5X,'INTERFERENCE PRESSURE(RADIALLY,P=CONST)')
```

```
6     FORMAT(I5,14X,E12.4)
7     FORMAT(1H0,'NON CONVERGENCE')
   10 CONTINUE
      RETURN
      END



      SUBROUTINE RECT(L,FW,NTRI,ALLK,U,TRI)
      DIMENSION NTRI(200,4),T(8),FW(200),U(400),ALLK(400,41),E(8,8),
     1A(8,8),TRI(200,10)
      COMMON/BLOCK1/TEMP,PR,YM,ALPHA,RHO
      IF(NTRI(L,4).EQ.0)GOTO8
      CALL AINV(RI,RJ,RK,RL,B,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
      GOTO9
8     CALL TAINV(RI,RJ,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
9     CONTINUE
      CALL SETZ1(E,8,8)
      IF (TRI(L,2).LT.0.0) GOTO 1
      IF (AI.LE.0.0) GOTO 2
      ELF = ALOG (AJ/AI)
      GOTO 3
2     ELF = 0
3     P1 = ((BK * AJ) - (BL * AI) + (BJ * AI))
      P2 = BL-BK-BJ
      P3 = (BL-BK)/(AJ-AI)
      P4 = BL-BK
      P5 = AJ-AI
      B1 = (P4**2-BJ**2)/(P5**2)
      B2 = (P4**3-BJ**3)/(P5**3)
      B3 = (BK**3-3*BK**2*AI*P3+3*BK*(AI*P3)**2-B2*AI**3)
      B4 = (3*BK**2*P3-6*BK*AI*P3**2+3*AI**2*B2)
      B5 = (3*BK*P3**2-3*AI*B2)
      B6 = (BK**2-2*AI*BK*P3)
      E1 = P2+P1*ELF/P5
      E2 = P1+P2*(AJ+AI)*0.5
      E3 = 0.5*((B6+B1*AI**2)*ELF+2*BK*P4-2*AI*B1*P5
     1+0.5*B1*(AJ**2-AI**2))
      E4 = 0.5*(BK**2*P5-2*AI*BK*P4+BK*P4*(AJ+AI)
     1+(P4**2-BJ**2)*P5/3)
      E5 = (P2*(AJ**3-AI**3)/3+P1*(AJ**2-AI**2)/2)/P5
      E6 = 0.5*(B6*(AJ**2-AI**2)/2+2*BK*P4*(AJ**3-AI**3)/(3*P5)
     1+B1*(0.25*P5**4+AI*(P5**3)/3))
      E7 = (P2*0.25*(AJ**4-AI**4)+P1*(AJ**3-AI**3)/3)/P5
      E8 = (B3*ELF+B4*P5+B5*(AJ**2-AI**2)/2
     1+B2*(AJ**3-AI**3)/3)/3
      E9 = (B3*P5+B4*(AJ**2-AI**2)/2+B5*(AJ**3-AI**3)/3
     1+B2*(AJ**4-AI**4)/4)/3
      E10 = (B3*(AJ**2-AI**2)/2+B4*(AJ**3-AI**3)/3
     1+B5*(AJ**4-AI**4)/4+B2*(AJ**5-AI**5)/5)/3
      E11 = (P2*(AJ**5-AI**5)/5+P1*(AJ**4-AI**4)/4)/P5
      E12 = (B6*(AJ**3-AI**3)/3+2*BK*P4*(AJ**4-AI**4)/(4*P5)
     1+B1*(0.2*P5**5+0.5*AI*P5**4+AI**2*P5**3/3))/2
      E13 = (B6*(AJ**4-AI**4)/4+2*BK*P4*(AJ**5-AI**5)/(5*P5)
     1+B1*((AJ**4*P5**2)/6-AI*(0.2*(AJ**5-AI**5)
     2-0.25*AI*(AJ**4-AI**4))/3))/2
      AJ1=(AJ-AI)*BK
      IF(AJ1.GT.0.0)GOTO4
```

```
      E1=-E1
      E2=-E2
      E3=-E3
      E4=-E4
      E5=-E5
      E6=-E6
      E7=-E7
      E8=-E8
      E9=-E9
      E10=-E10
      E11=-E11
      E12=-E12
      E13=-E13
      GOTO 4
1     RJRI = RJ-RI
      RLRK = RL-RK
      RLLN = ALOG(RL)
      RJLN = ALOG(RJ)
      RKRI = RK-RI
      RLRJ = RL-RJ
      RKLN = ALOG(RK)
      RILN = ALOG(RI)
      E2 = B*(RJRI+(RLRK-RJRI)/2)
      E4 = B**2*(RLRK+0.5*RJRI)/3
      E5 = 0.5*B*(RJ**2-RI**2+RJ*RLRJ-RI*RKRI+(RLRJ**2-RKRI**2)/3)
      E6 = B**2*((RJ**2-RI**2)/2+2*(RJ*RLRJ-RI*RKRI)/3
     1+(RLRJ**2-RKRI**2)/4)/2
      E7 = B*(RJ**3-RI**3+1.5*(RJ**2*RLRJ-RI**2*RKRI)
     1+RJ*RLRJ**2-RI*RKRI**2+(RLRJ**3-RKRI**3)/4)/3
      E9 = B**3*(RJRI+3*RLRK)/12
      E12 = B**2*((RJ**3-RI**3)/2+RJ**2*RLRJ-RI**2*RKRI
     1+0.75*(RJ*RLRJ**2-RI*RKRI**2)+0.2*(RLRJ**3-RKRI**3))/3
      IF (ABS(RLRJ).LT.1.E-08) GOTO 5
      E1 = (-RLRJ+RL*RLLN-RJ*RJLN)/RLRJ
      E3=RLLN-0.5+(RJ/RLRJ)*(1-RJ*(RLLN-RJLN)/RLRJ)
      E8=RLLN-1./3.+(RJ/RLRJ)*(0.5-(RJ/RLRJ)*(1-RJ*(RLLN-RJLN)/RLRJ))
      E10=RJ**2/6+RJ*RLRJ*0.25+RLRJ**2*0.1
      E11 = (RL**5-RJ**5)*0.05/RLRJ
      E13=RJ**4*0.5+4*RJ**3*RLRJ/3+1.5*RJ**2*RLRJ**2+0.8*RJ*RLRJ**3
     1+RLRJ**4/6
      GOTO 6
5     E1 = RJLN
      E3=RJLN
      E8=RJLN
      E10=RJ**2/6
      E11 = 0.25*RJ**4
      E13 = 0.5*RJ**4
6     IF (ABS(RKRI).LT.1.E-03) GOTO 7
      E1 = B*(E1-(-RKRI+RK*RKLN-RI*RILN)/RKRI)
      E3=B**2*(E3-RKLN+0.5-(RI/RKRI)*(1-RI*(RKLN-RILN)/RKRI))/2
      E8=B**3*(E8-RKLN+1./3.-(RI/RKRI)*(0.5-(RI/RKRI)
     1*(1-RI*(RKLN-RILN)/RKRI)))/3
      E10=B**3*(E10-RI**2/6-RI*RKRI*0.25-RKRI**2*0.1)
      E11 = B*(E11-(RK**5-RI**5)*0.05/RKRI)
      E13=B**2*(E13-RI**4*0.5-4*RI**3*RKRI/3-1.5*RI**2*RKRI**2
     1-0.8*RI*RKRI**3-RKRI**4/6)*0.25
      GOTO 4
```

```
7      E1 = B*(E1-RILN)
       E3 = B**2*(E3-RILN)/2
       E6 = B**3*(E8-RILN)/3
       E10 = B**3*(E10-(RI**2)/6)
       E11 = B*(E11-0.25*RI**4)
       E13 = B**2*(E13-0.5*RI**4)/4
4      E(1,1) = (1-PR)*E1
       E(1,2) = E2
       E(2,1) = E2
       E(1,3) = (1-PR)*E3
       E(3,1) = E(1,3)
       E(1,4) = E4
       E(4,1) = E4
       E(1,7) = PR*E2
       E(7,1) = PP*E2
       E(1,8) = PR*E5
       E(8,1) = PR*E5
       E(2,2) = 2*E5
       E(2,3) = E4
       E(3,2) = E4
       E(2,4) = 2*E6
       E(4,2) = 2*E6
       E(2,7) = 2*E(1,8)
       E(7,2) = E(2,7)
       E(2,8) = 2*PR*E7
       E(8,2) = E(2,8)
       E(3,3) = (1-PR)*E8+(1-2*PR)*E5/2
       E(3,4) = E9+(1-2*PR)*E7/2
       E(4,3) = E(3,4)
       E(3,6) = (1-2*PR)*E5/2
       E(6,3) = E(3,6)
       E(3,7) = PP*E4
       E(7,3) = E(3,7)
       E(3,8) = E6/2
       E(8,3) = E(3,8)
       E(4,4) = 2*E10+(1-2*PR)*E11/2
       E(4,6) = (0.5-PR)*E7
       E(6,4) = E(4,6)
       E(4,7) = PR*E(4,2)
       E(7,4) = E(4,7)
       E(4,8) = (0.5+PR)*E12
       E(8,4) = E(4,8)
       E(6,6) = E(3,6)
       E(6,8) = (0.5-PR)*E6
       E(8,6)=E(6,8)
       E(7,7) = (1-PR)*E5
       E(7,8) = (1-PR)*E7
       E(8,7) = E(7,8)
       E(8,8) = (1-PR)*E11+(0.5-PR)*E10
       T(1) = A(3,1)*E4+2*A(4,1)*E6+A(1,1)*E2+2*A(2,1)*E5
       T(2) = A(3,2)*E4+2*A(4,2)*E6+A(1,2)*E2+2*A(2,2)*E5
       T(3) = A(3,3)*E4+2*A(4,3)*E6+A(1,3)*E2+2*A(2,3)*E5
       T(4) = A(3,4)*E4+2*A(4,4)*E6+A(1,4)*E2+2*A(2,4)*E5
       T(5) = A(3,1)*E5+A(4,1)*E7
       T(6) = A(3,2)*E5+A(4,2)*E7
       T(7) = A(3,3)*E5+A(4,3)*E7
       T(8) = A(3,4)*E5+A(4,4)*E7
       IF(NTRI(L,4).NE.0)GOTO10
```

```
      DO11 I=1,8
      E(4,I)=0.
      E(8,I)=0.
      E(I,4)=0.
11    E(I,8)=0.
      T(4)=0.
      T(8)=0.
10    CONTINUE
      I = NTRI(L,1)
      FW(I) = (A(1,1)*E7+A(2,1)*E11+A(3,1)*E12+A(4,1)*E13)*RHO+FW(I)
      I = NTRI (L,2)
      FW(I) = (A(1,2)*E7+A(2,2)*E11+A(3,2)*E12+A(4,2)*E13)*RHO+FW(I)
      I = NTRI (L,3)
      FW(I) = (A(1,3)*E7+A(2,3)*E11+A(3,3)*E12+A(4,3)*E13)*RHO+FW(I)
      I = NTRI(L,4)
      IF(I.EQ.0)GOTO12
      FW(I) = (A(1,4)*E7+A(2,4)*E11+A(3,4)*E12+A(4,4)*E13)*RHO+FW(I)
12    CONTINUE
      CALL STEP1 (8,E,A,ALLK,T,U,NTRI,L)
      RETURN
      END



      SUBROUTINE RECTST (L,ST,TOTA,U,TRI,NTRI)
C     DETERMINES  ELEMENT  STRESSES
      DIMENSION  DB(4,8),TD(8),TRI(200,10),NTRI(200,4),STRESS(4),U(400),
     1A(8,8)
      COMMON/BLOCK1/TEMP,PR,YM,ALPHA,RHO
      IF(NTRI(L,4).EQ.0)GOTO6
      CALL AINV(RI,RJ,RK,RL,B,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
      AREA = -B*DET
      R= 0.5 * (AI + AJ)
      Z= 0.5 * (B + BJ)
      GOTO7
6     CALL TAINV(RI,RJ,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
      AREA=DET*0.5
      AREA=ABS(AREA)
      R=(2*RI+RJ)/3
      Z=(BJ+BK)/3
7     CONTINUE
      CALL SETZ1(DB,4,8)
      DB(1,1)= PP/R
      DB(1,2)= 1
      DB(1,3)= Z*DB(1,1)
      DB(1,4)= Z
      DB(1,7)= PR
      DB(1,8)= PR*R
      DB(2,1)= (1-PR)/R
      DB(2,2)= 1
      DB(2,3)= Z*DB(2,1)
      DB(2,4)= Z
      DB(2,7)= PR
      DB(2,8)= DB(1,8)
      DB(3,1)= DB(1,1)
      DB(3,2)= 2*PR
      DB(3,3)= DB(1,3)
      DB(3,4)= Z*DB(3,2)
      DB(3,7)= 1-PR
```

```
      DB(3,8)= R*DB(3,7)
      DB(4,3)= (1-2*PR)*0.5
      DB(4,4)= R*DB(4,3)
      DB(4,6)= DB(4,3)
      DB(4,8)= DB(4,3)*Z
      IF(NTRI(L,4).NE.0)GOTO4
      DO3 I=1,4
      DB(I,4)=0.
 3    DB(I,8)=0.
 4    CONTINUE
      NR1 = NTRI(L,1)*2 -1
      NR2 = NTRI(L,2)*2 -1
      NR3 = NTRI(L,3)*2 -1
      TD(1) = U(NR1)
      TD(2) = U(NR2)
      TD(3) = U(NR3)
      TD(5) = U(NR1+1)
      TD(6) = U(NR2+1)
      TD(7) = U(NR3+1)
      IF(NTRI(L,4).EQ.0)GOTO5
      NR4 = NTRI(L,4)*2 -1
      TD(4) = U(NR4)
      TD(8) = U(NR4+1)
 5    CONTINUE
      CALL STEP2(8,ST,TOTA,A,TD,DB,STRESS,AREA)
      WRITE(6,250)L, (STRESS(I),I=1,4),R,Z
  250 FORMAT (I5,4E12.4,2F10.5)
      RETURN
      END



      SUBROUTINE SETZ1(ADUM,N,M)
      DIMENSION ADUM(N,M)
      DO 62 I=1,N
      DO 62 J=1,M
 62   ADUM(I,J) =0.
      RETURN
      END


      SUBROUTINE SETZ2(ADUM,N)
      DIMENSION ADUM(N)
      DO 62 I=1,N
 62   ADUM(I) =0.
      RETURN
      END


      SUBROUTINE SETZ3(NDUM,N,M)
      DIMENSION NDUM(N,M)
      DO 62 I=1,N
      DO 62 J=1,M
 62   NDUM(I,J) =0
      RETURN
      END
```

```
      SUBROUTINE SET24(NDUM,N)
      DIMENSION NDUM(N)
      DO 62 I=1,N
62    NDUM(I) =0
      RETURN
      END


      SUBROUTINE SLIP(ISTR,DD,NODIS1,KCT,COF,ICY,IP,NP)
      DIMENSION ISTR(50,4),DD(400),NODIS1(400),IP(50)
      IF(KCT.EQ.1)GOTO11
      PEAD(5,1)COF
11    ICY=ICY+1
      WRITE(6,5)ICY
      WRITE(6,2)COF
      KCT=0
      DO10 I=1,NP
      MC1=2*IP(I)-1
      MC2=MC1+1
      IF(NODIS1(MC2).NE.0)GOTO10
      RATIO=DD(MC2)/DD(MC1)
      RATIO=ABS(RATIO)
      IF(RATIO.LT.COF)GOTO10
      KCT=1
      ISTR(I,4)=0
      WRITE(6,3)IP(I)
10    CONTINUE
      IF(KCT.EQ.0)WRITE(6,4)
    1 FORMAT(F5.0)
    2 FORMAT(1H ,'COEFFICIENT OF FRICTION=',F5.3)
    3 FORMAT(1H ,'NODE',I5,' ALLOWED TO SLIP AXIALLY')
    4 FORMAT(1H ,'NO SLIPPAGE')
    5 FORMAT(1H0,'SLIP CYCLE',I3)
      RETURN
      END


      SUBROUTINE STEP1(MS,BTDB,A,ALLK,T,U,NTRI,L)
C     POSITIONS IN ALLK ARRAY
      DIMENSION BTDB(8,8),A(8,8),ALLK(400,41),T(8),U(400),NTRI(250,4),
     1ELKA(8,8)
      COMMON/BLOCK1/TEMP,PR,YM,ALPHA,RHO
      M2 = MS/2
      M3=M2
      FACT = YM/((1 + PR) * (1-2 * PR))
      TEMP = YM*TEMP*ALPHA/(1-2 *PR)
      DO1 I = 1,MS
      DO1 J = 1,MS
      SUM = 0.0
      DO2 K = 1,MS
2     SUM = SUM + BTDB (I,K)*A(K,J)
      ELKA(I,J) = SUM
1     CONTINUE
      DO3 I = 1.MS
      DO3 J = 1,MS
      SUM = 0.0
      DO4 K = 1,MS
```

```
4      SUM = SUM + A(K,I)*ELKA(K,J)
       BTDB(I,J) = SUM*FACT
3      CONTINUE
       IF(NTRI(L,4).EQ.0)M3=3
       DO5 I=1,M3
       II = I + 4-M2
       MM = NTRI(L,II)*2-1
       KK = MM + 1
       II = I + M2
       U(MM) = T(I)*TEMP + U(MM)
       U(KK) = T(II)*TEMP + U(KK)
       DO6 J=1,M3
       JJ = J + 4-M2
       NN = NTRI(L,JJ)*2-1
       LL = NN+1
       JJ = J + M2
       IF (MM.LE.NN)ALLK(MM,NN-MM+1) = ALLK(MM,NN-MM+1) + BTDB(I,J)
       IF(MM.LE.LL)ALLK(MM,LL-MM+1) = ALLK(MM,LL-MM+1) + BTDB(I,JJ)
       IF (KK.LE.NN)ALLK(KK,NN-KK+1) = ALLK(KK,NN-KK+1) + BTDB(II,J)
       IF (KK.LE.LL)ALLK(KK,LL-KK+1) = ALLK(KK,LL-KK+1) + BTDB(II,JJ)
6      CONTINUE
5      CONTINUE
       RETURN
       END


       SUBROUTINE STEP2 (MS,ST,TOTA,A,TD,DB,STRESS,AREA)
       DIMENSION A(8,8),TD(8),DB(4,8),STRESS(4),ATD(8)
       COMMON/BLOCK1/TEMP,PR,YM,ALPHA,RHO
       FACT = YM/((1+PR)*(1-2*PR))
       DO1 I=1,MS
       SUM=0.0
       DO2 J=1,MS
2      SUM= SUM+A(I,J)*TD(J)
1      ATD(I)= SUM
       DO3 I=1,4
       SUM= 0.0
       DO4 J=1,MS
4      SUM= SUM + DB(I,J)*ATD(J)
3      STRESS(I)= FACT*SUM-TEMP*ALPHA*YM/(1-2*PR)
       STRESS(4)= FACT*SUM
       ST= ST+ STRESS(2)*AREA
       TOTA= TOTA + AREA
       RETURN
       END
```

```
      SUBROUTINE TAINV(RI,RJ,AI,AJ,BJ,BK,BL,DET,L,TRI,A)
C     INVERTS THE ARRAY A FOR THE TRIANGLE
      DIMENSION A(8,8),TRI(200,10)
      CALL SETZ1(A,8,8)
      RI=TRI(L,1)
      RJ=TRI(L,2)
      BJ=TRI(L,3)
      BK=TRI(L,4)
      BL=BJ
      AI=RI
      AJ=RJ
      DET=BK*(RJ-RI)
      A(1,1)=(BK*RJ-BJ*RI)/DET
      A(5,5)=A(1,1)
      A(1,2)=-RI*BK/DET
      A(5,6)=A(1,2)
      A(1,3)=RI*BJ/DET
      A(5,7)=A(1,3)
      A(2,1)=(BJ-BK)/DET
      A(6,5)=A(2,1)
      A(2,2)=BK/DET
      A(6,6)=A(2,2)
      A(2,3)=-BJ/DET
      A(6,7)=A(2,3)
      A(3,1)=-1/BK
      A(7,5)=A(3,1)
      A(3,3)=-A(3,1)
      A(7,7)=A(3,3)
      RETURN
      END
```

APPENDIX B

A BRIEF CONSIDERATION OF THE STRUCTURE OF THE PROGRAM

A flow chart of the computer program for a single case run is shown in figure 3. The functions of the program elements listed in appendix A will now be briefly described:

AXIFE	the main program element, which acts as executor, reading in data, calling appropriate subroutines, and routing according to options and multiple case processing requirements
AINV	inverts the shape function matrix (A) for trapezoidal elements
BCROUT	applies matrix reduction solution scheme to solve for deformations
DIFDIS	reduces system of equations according to interference (differential displacement) conditions
PRESS	calculates pressure distribution at interference location
RECT	prepares data for calculating element stiffness characteristics and thermal and inertial force effects
RECTST	prepares data for calculating element stresses
SETZ1, SETZ2, SETZ3, SETZ4	sets elements of arrays of various dimensions and types to zero
SLIP	allows axial slip of interfering nodes if appropriate
STEP1	determines element stiffness characteristics and organizes them into the main stiffness matrix
STEP2	calculates element stresses
TAINV	inverts the shape function matrix (A) for triangular elements

APPENDIX C

SOME PROGRAM DETAILS

Array dimensioning. - The following arrays would need redimensioning if the number of nodes required exceeds 200: ALLK, TRI, NTRI, NODIS, FW, U, TU, DEL, DD, NDD, MOVE, ALLK1, NODIS1, and D.

The following arrays would need redimensioning if the maximum nodal difference in an element (NW) were to exceed 19: ALLK and ALLK1 (see section Maximum nodal number difference).

The following arrays would need redimensioning if the number of constrained nodes were to exceed 50: NUV, ANUV, IP, AL, P, Z, B, and A.

The appropriate modifications, should redimensioning be required, are apparent without further detailing. Other dimensioning limitations are unlikely to be significant. The storage requirement of the program, as listed, on a Univac 1100 machine is 53 004 words. Note also that if redimensioning of arrays whose elements are set to zero is undertaken (through SETZ1 etc.), then the call of the subroutines SETZ should be amended accordingly.

Maximum nodal number difference. - The matrix reduction solution scheme after Crout (ref. 3) has been amended to a banded form for the purposes of the present program. The bandwidth of the solution is (2NW + 2), where NW is the maximum nodal number difference in any element. In fact, the width of the main stiffness matrix ALLK, and its copy ALLK1, has been dimensioned as (2NW + 3), the nodal force vector being added on in the last column.

Whilst it is apparent what the maximum nodal number difference is in an element where none of the nodes is "involved" in an interference, it is not so clear if this is not the case. If an interference is effected between components, then for elements involved in the interference the value of NW should be assessed as the maximum nodal number difference between nodes in the element and associated nodes in interference with the element. (Examples are shown in fig. 4.)

Equivalent loading. - If a continuously distributed load is applied to a structure, the loading must be reduced to an equivalent nodal distribution. An example is shown in figure 5.

Frictional condition at interference surfaces. - If two nodes have an imposed radial interference, they may also be constrained axially or, alternatively, allowed to move freely axially (to slip). If no slip is permitted between two radially constrained nodes, then equal and opposite forces will be set up on the nodes, there being an axial component. The reality of whether or not axial slip occurs between the nodes might be thought of as whether the ratio of axial to radial force exceeds some assumed coefficient of friction. An arrangement for automatically allowing slip has been incorporated into the program (cards T and U).

The option to allow slip to occur should only be chosen when all nodes in interference are located at the same radius and are axially adjacent. With the slip option effective, all the node pairs should initially be constrained axially. If slip of any node pair actually occurs, then a recalculation of distortions, etc., is carried out. Several cycles may be required until either no slip occurs in the remaining nodes constrained axially or all nodes have been freed to slip axially.

The process of slip can be investigated by manual intervention and, although this is more time consuming, it may be more appropriate in some circumstances.

APPENDIX D

AN EXAMPLE

To demonstrate the use of the program, an example will now be presented. This is the inner ring of a roller bearing having an interference fit with its hollow shaft. An extremely course mesh of elements has been chosen (see fig. 6) to avoid too much detail. Figure 6 shows node and element numbering chosen together with dimensions given in meters. A symmetrical situation is considered and therefore only half the geometry has been considered. The following data were chosen for the example:

Rotational speed, rpm	5000
Ring and shaft material	steel
Poisson's ratio	0.3
Young's modulus, N/m^2	2×10^{11}
Coefficient of expansion, $^{o}C^{-1}$	10^{-5}
Density, kg/m^3	7800
Ring temperature (relative to the shaft), ^{o}C	20
Shaft temperature (no thermal expansion), ^{o}C	0
Radial interference, m	4×10^{-5}
Coefficient of friction (for axial slip)	0.5

Note that nodes 1 to 7 are restrained axially to fulfill symmetry requirements and that the nodes in radial interference 4/5, 11/12, 20/21, 29/30, and 38/39 are at the same radial location; that is, there are two nodes at the same spatial position.

It should be noted that the technical aspects of this example are not the concern of this report. The example is solely to demonstrate a typical data input and the form of the output. The example used 13 seconds of central processing unit time on a Univac 1100 computer. The data input and program output are now presented.

```
1                                                                          CARD A
ROLLER BEARING INNER RING/SI UNITS                                         CARD B
1                                                                          CARD C
 28  47  11        7                                                       CARD D
  .3000E 00   .2000E 12   .1000E-04   .7800E 04                            CARD E
    5000.                                                                  CARD F
  1   1   9    8   2    1                                                  CARD G
     .064      .067                 .003      .003        20.              CARD H
  3   1  11   10   4    3
     .058      .061                 .003      .003        20.
  4   1  13   12   6    5
     .053      .058                 .003      .003
  5   1  14   13   7    6
     .048      .053                 .003      .003
  6   1  18   17   9    8
     .064      .067                 .003      .003        20.
  8   1  20   19  11   10
     .058      .061                 .003      .003        20.
  9   1  22   21  13   12
     .053      .058                 .003      .003
 10   1  23   22  14   13
     .048      .053                 .003      .003
 11   1  25   24  16   15
     .069      .071                 .003      .003        20.
 12   1  26   25  17   16
     .067      .069                 .003      .003        20.
 13   1  27   26  18   17
     .064      .067                 .003      .003        20.
 15   1  29   28  20   19
     .058      .061                 .003      .003        20.
 16   1  31   30  22   21
     .053      .058                 .003      .003
 17   1  32   31  23   22
     .048      .053                 .003      .003
 18   1  34   33  25   24
     .069      .071                 .003      .003        20.
 19   1  35   34  26   25
     .067      .069                 .003      .003        20.
 20   1  36   35  27   26
     .064      .067                 .003      .003        20.
 22   1  38   37  29   28
     .058      .061                 .003      .003        20.
 23   1  40   39  31   30
     .053      .058                 .003      .003
 24   1  41   40  32   31
     .048      .053                 .003      .003
 25   1  43   42  40   39
     .053      .058                 .003      .003
 26   1  44   43  41   40
     .048      .053                 .003      .003
 27   1  46   45  43   42
     .053      .058                 .003      .003
 28   1  47   46  44   43
     .048      .053                 .003      .003
```

```
   1    0    1                                                                                   CARD L
   2    0    1
   3    0    1
   4    0    1
   5    0    1
   6    0    1
   7    0    1
1                                                                                                CARD N
    5                                                                                            CARD O
      4             5              1 -.4000E-04          1                                       CARD P
     11            12              1 -.4000E-04          1
     20            21              1 -.4000E-04          1
     29            30              1 -.4000E-04          1
     38            39              1 -.4000E-04          1
   1
   5    .5800E-01                                                                                CARD Q
   5    .3000E-02                                                                                CARD R
  12    .3000E-02                                                                                CARD S
  21    .3000E-02
  30    .3000E-02
  39    .3000E-02
1                                                                                                CARD T
   .5                                                                                            CARD U

 ROLLER BEARING INNER RING/SI UNITS

 M=  28      N=  47      NW=  11      NE=   0      NCON=   7      NFOR=   0


 PR= .3000+00   YM= .2000+12  COEX=  .1000-04  RHO= .7800+04  TEMP=  .0000  SPEED(RPM)=  5000.00


 ELEMENT  TYPE   I     J     K     L     RI         RJ        RKOBJ      FLOBK      BCBL       TEMP


     1      1     9     8     2     1  .6400-01   .6700-01   .0000    .3000-02  .3000-02   .2000+02
     2      1    10     9     3     2  .6100-01   .6400-01   .0000    .3000-02  .3000-02   .2000+02
     3      1    11    10     4     3  .5800-01   .6100-01   .0000    .3000-02  .3000-02   .2000+02
     4      1    13    12     6     5  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
     5      1    14    13     7     6  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
     6      1    18    17     9     8  .6400-01   .6700-01   .0000    .3000-02  .3000-02   .2000+02
     7      1    19    18    10     9  .6100-01   .6400-01   .0000    .3000-02  .3000-02   .2000+02
     8      1    20    19    11    10  .5800-01   .6100-01   .0000    .3000-02  .3000-02   .2000+02
     9      1    22    21    13    12  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
    10      1    23    22    14    13  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
    11      1    25    24    16    15  .6900-01   .7100-01   .0000    .3000-02  .3000-02   .2000+02
    12      1    26    25    17    16  .6700-01   .6900-01   .0000    .3000-02  .3000-02   .2000+02
    13      1    27    26    18    17  .6400-01   .6700-01   .0000    .3000-02  .3000-02   .2000+02
    14      1    28    27    19    18  .6100-01   .6400-01   .0000    .3000-02  .3000-02   .2000+02
    15      1    29    28    20    19  .5800-01   .6100-01   .0000    .3000-02  .3000-02   .2000+02
    16      1    31    30    22    21  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
    17      1    32    31    23    22  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
    18      1    34    33    25    24  .6900-01   .7100-01   .0000    .3000-02  .3000-02   .2000+02
    19      1    35    34    26    25  .6700-01   .6900-01   .0000    .3000-02  .3000-02   .2000+02
    20      1    36    35    27    26  .6400-01   .6700-01   .0000    .3000-02  .3000-02   .2000+02
    21      1    37    36    28    27  .6100-01   .6400-01   .0000    .3000-02  .3000-02   .2000+02
    22      1    38    37    29    28  .5800-01   .6100-01   .0000    .3000-02  .3000-02   .2000+02
    23      1    40    39    31    30  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
    24      1    41    40    32    31  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
    25      1    43    42    40    39  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
    26      1    44    43    41    40  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
    27      1    46    45    43    42  .5300-01   .5800-01   .0000    .3000-02  .3000-02   .0000
    28      1    47    46    44    43  .4800-01   .5300-01   .0000    .3000-02  .3000-02   .0000
```

```
NODAL  RESTRAINTS
NODE          IU          IV          UU          VV
   1           0           1       .0000       .0000
   2           0           1       .0000       .0000
   3           0           1       .0000       .0000
   4           0           1       .0000       .0000
   5           0           1       .0000       .0000
   6           0           1       .0000       .0000
   7           0           1       .0000       .0000


APPLIED NODAL FORCES


NODE              FR              FZ


DIFFERENTIAL DISPLACEMENTS
NODE  A       NODE  B          IR           DR               IZ           DZ
      4             5           1   -.4000-04                 1   .0000
     11            12           1   -.4000-04                 1   .0000
     20            21           1   -.4000-04                 1   .0000
     29            30           1   -.4000-04                 1   .0000
     38            39           1   -.4000-04                 1   .0000


NODAL  DEFORMATIONS
NODE              UR              UZ
   1     .3274-04     .0000
   2     .3241-04     .0000
   3     .3211-04     .0000
   4     .3185-04     .0000
   5    -.8153-05     .0000
   6    -.8221-05     .0000
   7    -.8388-05     .0000
   8     .3273-04   -.3365-06
   9     .3240-04   -.2980-06
  10     .3210-04   -.2587-06
  11     .3184-04   -.2269-06
  12    -.8156-05   -.2269-06
  13    -.8221-05   -.2089-06
  14    -.8387-05   -.2185-06
  15     .3311-04   -.2061-06
  16     .3288-04   -.7463-06
  17     .3266-04   -.6843-06
  18     .3236-04   -.6058-06
  19     .3209-04   -.5238-06
  20     .3184-04   -.4509-06
  21    -.8160-05   -.4509-06
  22    -.8219-05   -.4155-06
  23    -.8379-05   -.4412-06
  24     .3302-04   -.1169-05
  25     .3278-04   -.1106-05
  26     .3257-04   -.1041-05
  27     .3229-04   -.9428-06
  28     .3207-04   -.8080-06
  29     .3184-04   -.6639-06
  30    -.8163-05   -.6639-06
```

```
   31     -.8198-05     -.6215-06
   32     -.8350-05     -.6669-06
   33      .3292-04     -.1533-05
   34      .3269-04     -.1466-05
   35      .3247-04     -.1402-05
   36      .3219-04     -.1295-05
   37      .3197-04     -.1166-05
   38      .3189-04     -.8572-06
   39     -.8113-05     -.8572-06
   40     -.8134-05     -.8265-06
   41     -.8291-05     -.8833-06
   42     -.7822-05     -.9279-06
   43     -.8029-05     -.1019-05
   44     -.8220-05     -.1073-05
   45     -.7714-05     -.1036-05
   46     -.7926-05     -.1159-05
   47     -.8161-05     -.1246-05


ELEMENT STRESSES
ELMT      R STRESS      T STRESS     AX STRESS     SH STRESS            R            Z
    1     -.9436+06      .5870+08     -.1521+07     -.2004+06       .06550       .00150
    2     -.3029+07      .6110+08     -.4024+07     -.2879+06       .06250       .00150
    3     -.5279+07      .6404+08     -.6183+07     -.3000+06       .05950       .00150
    4     -.5069+07     -.2981+08      .4065+07     -.1027+06       .05550       .00150
    5     -.1803+07     -.3222+08      .4039+07      .6720+05       .05050       .00150
    6     -.1724+07      .5850+08     -.1113+07     -.2380+06       .06550       .00150
    7     -.3869+07      .6086+08     -.3805+07     -.8573+06       .06250       .00150
    8     -.5967+07      .6376+08     -.6361+07     -.1067+07       .05950       .00150
    9     -.5409+07     -.3001+08      .3726+07     -.3891+06       .05550       .00150
   10     -.1941+07     -.3224+08      .4058+07      .1360+06       .05050       .00150
   11     -.6136+06      .5398+08      .9208+05      .1390+05       .07000       .00150
   12     -.1971+07      .5566+08     -.3603+04     -.2875+05       .06800       .00150
   13     -.3296+07      .5803+08     -.4447+06     -.1742+06       .06550       .00150
   14     -.6111+07      .6030+08     -.3037+07     -.1631+07       .06250       .00150
   15     -.7209+07      .6327+08     -.6606+07     -.2492+07       .05950       .00150
   16     -.6414+07     -.3056+08      .2876+07     -.8247+06       .05550       .00150
   17     -.2253+07     -.3228+08      .4029+07     -.8681+05       .05050       .00150
   18     -.6178+06      .5370+08      .7989+05     -.1152+05       .07000       .00150
   19     -.1928+07      .5542+08      .9309+05      .3017+05       .06800       .00150
   20     -.3920+07      .5766+08     -.1192+06     -.4837+05       .06550       .00150
   21     -.7082+07      .6061+08     -.2661+06     -.7896+06       .06250       .00150
   22     -.1355+08      .6109+08     -.7345+07     -.5204+07       .05950       .00150
   23     -.7756+07     -.3123+08      .1577+07     -.2016+07       .05550       .00150
   24     -.2440+07     -.3229+08      .3626+07     -.7835+06       .05050       .00150
   25     -.5458+07     -.3122+08     -.2213+07     -.4622+07       .05550       .00150
   26     -.1945+07     -.3218+08      .2520+07     -.1415+07       .05050       .00150
   27     -.3067+06     -.2858+08     -.3939+06     -.1048+07       .05550       .00150
   28     -.9778+06     -.3216+08      .4808+06     -.9920+06       .05050       .00150
AVERAGE TANGENTIAL  STRESS=    .7186+07
```

```
INTERFERENCE FORCE VECTORS
NODE        RADIAL        AXIAL


   5     .5674+03     .0000
  12     .1175+04     .9051+02
  21     .1296+04     .2456+03
  30     .1639+04     .4642+03
  39     .2036+04     .1349+04

NODE       INTERFERENCE PRESSURE(RADIALLY.R=CONST)
   5                       .6484+07
  12                       .6634+07
  21                       .7755+07
  30                       .6419+07
  39                       .2907+08

SLIP CYCLE   1
COEFFICIENT OF FRICTION= .500
NODE    39 ALLOWED TO SLIP AXIALLY



INTERFERENCE FORCE VECTORS
NODE        RADIAL        AXIAL

   5     .5590+03     .0000
  12     .1160+04     .1254+03
  21     .1304+04     .3115+03
  30     .1794+04     .1671+04
  39     .1910+04     .0000

NODE       INTERFERENCE PRESSURE(RADIALLY.R=CONST)
   5                       .6379+07
  12                       .6564+07
  21                       .7560+07
  30                       .8053+07
  39                       .2659+08

SLIP CYCLE   2
COEFFICIENT OF FRICTION= .500
NODE    30 ALLOWED TO SLIP AXIALLY



INTERFERENCE FORCE VECTORS
NODE        RADIAL        AXIAL


   5     .5514+03     .0000
  12     .1148+04     .2261+03
  21     .1367+04     .1740+04
  30     .1859+04     .0000
  39     .1820+04     .0000
```

```
NODE      INTERFERENCE PRESSURE(RADIALLY.R=CONST)
   5                   .6312+07
  12                   .6418+07
  21                   .7943+07
  30                   .8756+07
  39                   .2497+08

SLIP CYCLE  3
COEFFICIENT OF FRICTION= .500
NODE   21 ALLOWED TO SLIP AXIALLY



INTERFERENCE FORCE VECTORS
NODE       RADIAL       AXIAL


   5    .5436+03    .0000
  12    .1198+04    .1624+04
  21    .1411+04    .0000
  30    .1831+04    .0000
  39    .1776+04    .0000


NODE      INTERFERENCE PRESSURE(RADIALLY.R=CONST)
   5                   .6066+07
  12                   .6793+07
  21                   .8249+07
  30                   .8593+07
  39                   .2436+08

SLIP CYCLE  4
COEFFICIENT OF FRICTION= .500
NODE   12 ALLOWED TO SLIP AXIALLY



INTERFERENCE FORCE VECTORS
NODE       RADIAL       AXIAL


   5    .5921+03    .0000
  12    .1229+04    .0000
  21    .1385+04    .0000
  30    .1811+04    .0000
  39    .1754+04    .0000

NODE      INTERFERENCE PRESSURE(RADIALLY.R=CONST)
   5                   .6756+07
  12                   .6955+07
  21                   .8030+07
  30                   .8536+07
  39                   .2403+08

SLIP CYCLE  5
COEFFICIENT OF FRICTION= .500
NO SLIPPAGE
```

```
NODAL DEFORMATIONS
NODE          UR          UZ
   1     .3236-04     .0000
   2     .3201-04     .0000
   3     .3173-04     .0000
   4     .3152-04     .0000
   5    -.8479-05     .0000
   6    -.8640-05     .0000
   7    -.8801-05     .0000
   8     .3239-04    -.2722-06
   9     .3204-04    -.3145-06
  10     .3176-04    -.3577-06
  11     .3156-04    -.4189-06
  12    -.8442-05    -.3441-07
  13    -.8596-05    -.1693-06
  14    -.8754-05    -.3182-06
  15     .3290-04    -.4061-06
  16     .3267-04    -.4749-06
  17     .3245-04    -.5589-06
  18     .3213-04    -.6357-06
  19     .3186-04    -.7110-06
  20     .3167-04    -.8327-06
  21    -.8331-05    -.7011-07
  22    -.8464-05    -.3411-06
  23    -.8609-05    -.6330-06
  24     .3300-04    -.7711-06
  25     .3277-04    -.8355-06
  26     .3255-04    -.8945-06
  27     .3227-04    -.9702-06
  28     .3203-04    -.1057-05
  29     .3185-04    -.1237-05
  30    -.8146-05    -.1114-06
  31    -.8228-05    -.5216-06
  32    -.8361-05    -.9313-06
  33     .3309-04    -.1136-05
  34     .3286-04    -.1193-05
  35     .3265-04    -.1250-05
  36     .3239-04    -.1327-05
  37     .3222-04    -.1426-05
  38     .3219-04    -.1637-05
  39    -.7808-05    -.1700-06
  40    -.7878-05    -.7062-06
  41    -.8031-05    -.1188-05
  42    -.7265-05    -.2551-06
  43    -.7466-05    -.8584-06
  44    -.7657-05    -.1394-05
  45    -.6871-05    -.3558-06
  46    -.7071-05    -.9782-06
  47    -.7292-05    -.1554-05
```

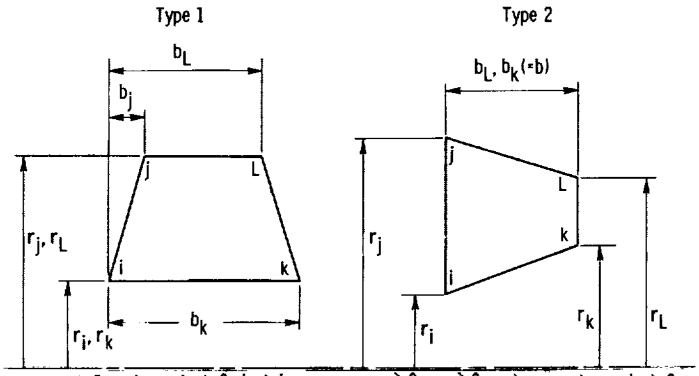
```
ELEMENT STRESSES
ELMT     R STRESS      T STRESS     AX STRESS     SH STRESS          R          Z
   1   -.8323+06     .5700+08    -.3595+07    -.2358+06      .06550     .00150
   2   -.2811+07     .6117+08    -.8954+05    -.2859+06      .06250     .00150
   3   -.5270+07     .6601+08     .4106+07    -.1301+06      .05950     .00150
   4   -.5352+07    -.3387+08    -.4976+07     .1540+05      .05550     .00150
   5   -.1917+07    -.3332+08     .5679+07    -.6740+04      .05050     .00150
   6   -.1602+07     .5708+08    -.3089+07    -.4264+06      .06550     .00150
   7   -.3335+07     .6120+08    -.1539+06    -.9331+06      .06250     .00150
   8   -.5822+07     .6593+08     .3608+07    -.3761+06      .05950     .00150
   9   -.5866+07    -.3373+08    -.4963+07     .3477+04      .05550     .00150
  10   -.2162+07    -.3303+08     .5662+07    -.1655+06      .05050     .00150
  11   -.5979+06     .5367+08     .1085+06    -.1777+05      .07000     .00150
  12   -.1992+07     .5506+08    -.8765+06     .1610+06      .06800     .00150
  13   -.2902+07     .5753+08    -.1278+07    -.1042+07      .06550     .00150
  14   -.5091+07     .6094+08    -.5596+06    -.1778+07      .06250     .00150
  15   -.7088+07     .6573+08     .2596+07    -.6713+06      .05950     .00150
  16   -.7185+07    -.3348+08    -.4803+07    -.1672+06      .05550     .00150
  17   -.2525+07    -.3245+08     .5469+07    -.8044+06      .05050     .00150
  18   -.6332+06     .5391+08     .3459+05     .1969+05      .07000     .00150
  19   -.1903+07     .5559+08    -.1542+06    -.1607+06      .06800     .00150
  20   -.4396+07     .5773+08    -.2527+06    -.8113+06      .06550     .00150
  21   -.8618+07     .6046+08    -.2579+06    -.1600+07      .06250     .00150
  22   -.1374+08     .6389+08     .6537+06    -.1744+07      .05950     .00150
  23   -.7913+07    -.3246+08    -.4009+07    -.1535+07      .05550     .00150
  24   -.2350+07    -.3152+08     .4554+07    -.1868+07      .05050     .00150
  25   -.3973+07    -.2921+08    -.2042+07    -.3472+07      .05550     .00150
  26   -.1557+07    -.3049+08     .2334+07    -.2233+07      .05050     .00150
  27    .1353+06    -.2590+08    -.3783+06    -.7005+06      .05550     .00150
  28   -.3622+06    -.2917+08     .4595+06    -.1197+07      .05050     .00150
AVERAGE TANGENTIAL  STRESS=     .7186+07
```

REFERENCES

1. Wilson, E. A.; and Parsons, B.: Finite Element Analysis of Elastic Contact Problems Using Differential Displacements. Int. J. Num. Methods Eng., vol. 2, no. 3, July-Sept. 1970, pp. 387-395.
2. Zienkiewicz, O. C.: The Finite Element Method in Structural and Continuum Mechanics. McGraw-Hill Book Co., Inc., 1967.
3. Crout, P. D.: A Short Method for Evaluating Determinants and Solving Systems of Linear Equations with Real or Complex Coefficients. Trans. Am. Inst. Electr. Eng., vol. 60, 1941, pp. 1235-1240.
4. Harris, T. A.; and Broschard, J. L.: Analysis of an Improved Planetary Gear Transmission Bearing. J. Basic Eng., vol. 86, Sept. 1964, pp. 457-462.

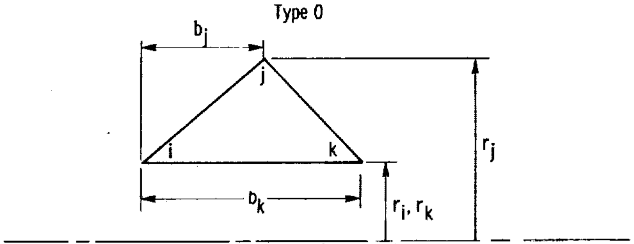
TABLE I. - DATA CARD SEQUENCE FOR A SINGLE CASE RUN

Card(s)	Comment
A	= 1 (number of geometries)
B	--
C	= 1 (number of cases for the geometry)
D	--
E	--
F	--
G,H	repeat as appropriate
I	if there are "special" material elements, repeat as appropriate
J,K	if base material properties are temperature dependent, repeat as appropriate
L	if nodes are constrained, repeat as appropriate
M	if nodal forces are applied, repeat as appropriate
N	if = 0, terminate data; if = 1, continue for an interference case
O	--
P	repeat as appropriate
Q	if = 0, go to card T
R	--
S	repeat as appropriate
T	if = 0, terminate data; if = 1, continue for slip analysis
U	--



$r_i \geq 0$, $r_j > r_i$, $b_k > 0$, $b_L > b_j$ $r_i \geq 0$, $r_k \geq 0$, $r_j > r_i$, $r_L > r_k$, $b_L > 0$

Trapezoidal axisymmetric elements



$r_i \geq 0$, $r_j \geq 0$, $r_i \neq r_j$, $b_k \neq 0$

Triangular axisymmetric element

Figure 1. - The three types of element incorporated into the computer program.

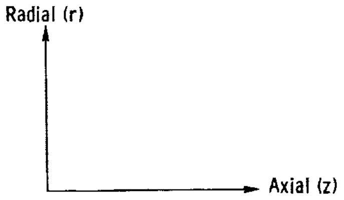


Figure 2. - The coordinate system.

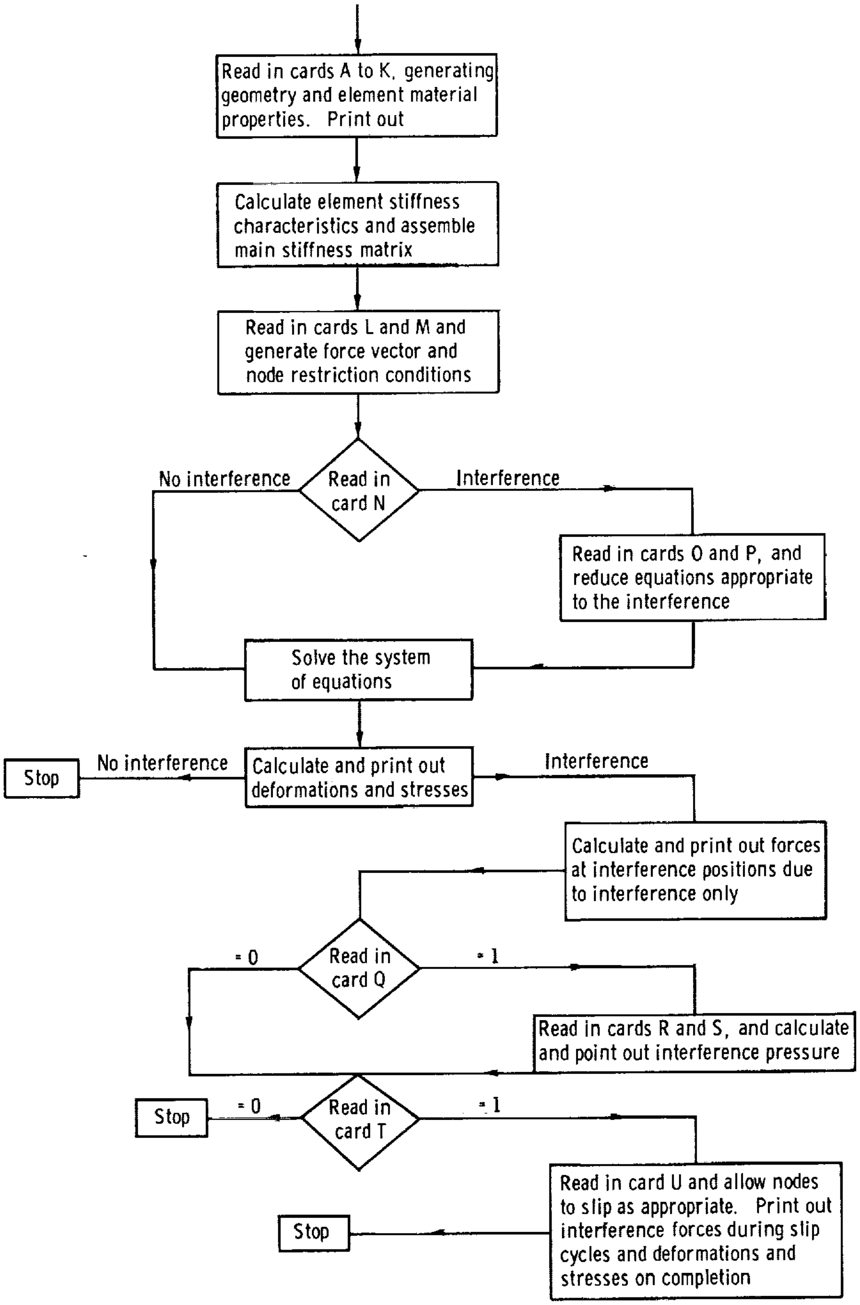
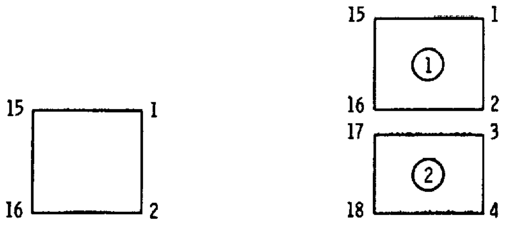


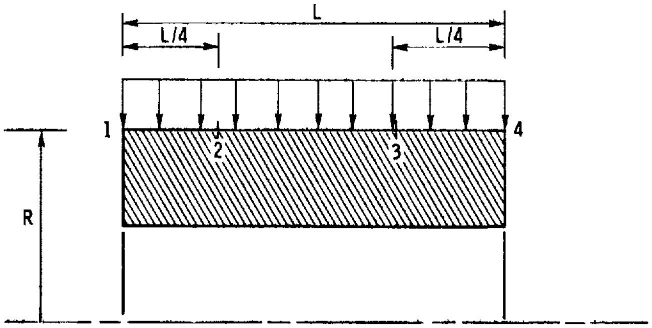
Figure 3. - Flow chart for a single case run.



Element not involved in an interference. NW = 16 - 1 = 15

Elements having interference at nodes 2-3 and 16-17. Element ① NW = 17 - 1 = 16; element ② NW = 18 - 2 = 16

Figure 4. - Determination of nodal number difference (NW) in element.



Cylinder with uniformly distributed compressive loading of Q/unit length on its outer surface

Equivalent forces at -

Nodes 1 and 4: $-Q\cdot 2\pi R\cdot\frac{L}{4}\cdot\frac{1}{2} = -\frac{\pi RLQ}{4}$

Nodes 2 and 3: $-\left(\frac{\pi RLQ}{4} + Q\cdot 2\pi R\cdot\frac{L}{2}\cdot\frac{1}{2}\right) = -\frac{3\pi RLQ}{4}$

Figure 5. - Determination of node applied forces due to distributed load.

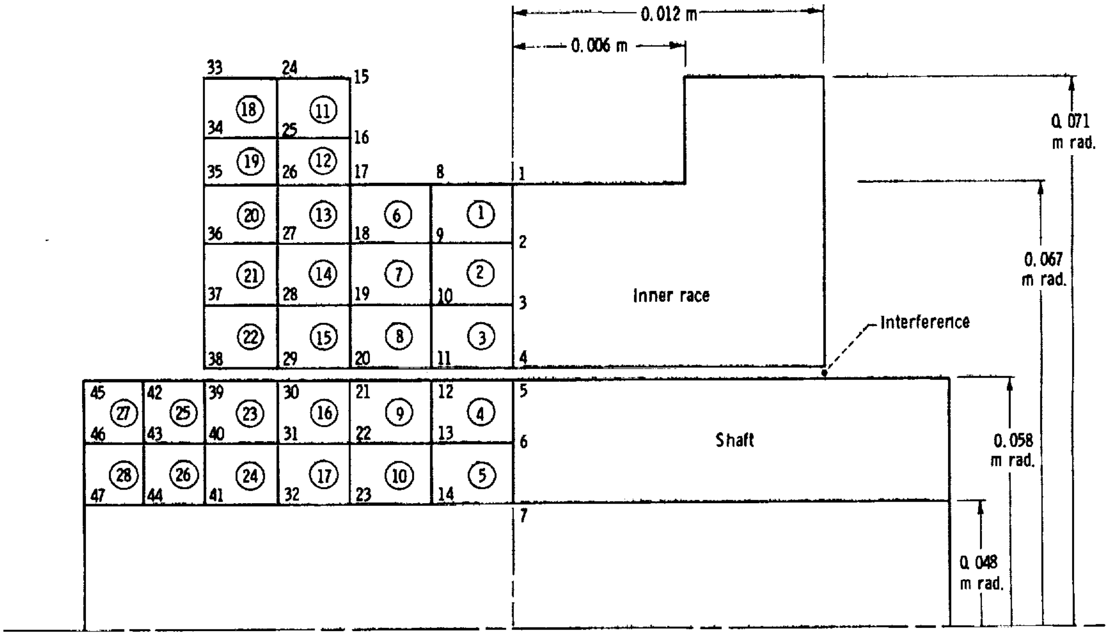


Figure 6. - The example: the inner ring and shaft of roller bearing. (Not to scale.)