FORTRAN PROGRAMS FOR CALCULATING NONLINEAR SEISMIC GROUND RESPONSE IN TWO DIMENSIONS

by

William B. Joyner

U. S. Geological Survey

OPEN FILE REPORT

78-287

This report is preliminary and has not been edited or reviewed for conformity with Geological Survey standards and nomenclature.

This page intentionally left blank

Introduction

The programs described here were designed for calculating the nonlinear seismic response of a two-dimensional configuration of soil underlain by a semi-infinite elastic medium representing bedrock. There are two programs. One is for plane strain motions, that is, motions in the plane perpendicular to the long axis of the structure, and the other is for antiplane strain motions, that is motions parallel to the axis. The seismic input is provided by specifying what the motion of the rock-soil boundary would be if the soil were absent and the boundary were a free surface. This may be done by supplying a magnetic tape containing the values of particle velocity for every boundary point at every instant of time. Alternatively, a punch card deck may be supplied giving acceleration values at every instant of time. In the plane strain program it is assumed that the acceleration values apply simultaneously to every point on the boundary; in the antiplane strain program it is assumed that the acceleration values characterize a plane shear wave propagating upward in the underlying elastic medium at a specified angle with the vertical.

The nonlinear hysteretic behavior of the soil is represented by a three-dimensional rheological model proposed by Iwan (1967). A boundary condition (Papastamatiou, written communication) is used which takes account of finite rigidity in the elastic substratum. The computations are performed by an explicit finite-difference scheme that proceeds step by step in space and time. Computations are done in terms of stress departures from an unspecified initial state.

Source listings are provided here along with instructions for preparing the input. A more detailed discussion of the method is presented elsewhere (Joyner, 1975). One provision of the programs that was not described in the

earlier publication permits irregularly shaped three-and four-sided elements. The programs relate velocity of the nodes to strain rate in the elements by a set of linear equations. The coefficients in these equations are determined by representing particle velocity within the element as a linear function of position and requiring the best least squares fit to the particle velocity at the nodes. Another provision not described in the earlier publication permits low-pass filtering of the output to remove the numerical noise (Joyner and Chen, 1975) that is generated at frequencies in the vicinity of 1/(4*DELT), where DELT is the time step. The filter is a zero-phase-shift digital filter with a frequency response, R, given by

$$R(f) = 1 \qquad f \leq F1$$

$$R(f) = 0.5 * (1.0 + \cos(\pi *(f-F1)/(F2-F1))) \qquad F1 \leq f \leq F2$$

$$R(f) = 0 \qquad f \geq F2$$

where f is frequency and F1 and F2 are parameters specified on an input card. It is recommended that F1 be assigned the value f_R, and F2 be assigned the value $2f_R$, where f_R is the highest frequency for which faithful representation is desired in the output. If filtering is done the number of points in the output time histories is reduced by 20.

In order to attain the desired frequency resolution the largest dimension of any element should be no more than about $VS/(10*f_R)$. The size of the time step must be limited to insure stability. The stability requirement for

the plane-strain problem in the linear elastic case with a square grid (Alterman and Loewenthal, 1972) is

$$VP*DELT \leq DELX/(1 + (VS/VP)^2)^{1/2}$$

where DELX is the grid size, VP is the compressional wave velocity, and VS is the shear wave velocity. For the anti plane-strain problem the requirement (Boore, 1972), is

$$VS*DELT \leq DELX/\sqrt{2}.$$

Satisfying these requirements for the linear case does not guarantee stability in the nonlinear case, but the relationships are useful as guides. It is recommended that DELT be taken as 75 to 80 percent of the smallest value that satisfies the appropriate elastic stability criterion. In computing DELT from the stability criterion a value should be used for DELX equal to the minimum dimension of any element. Relatively equidimensional elements of approximately uniform size are strongly recommended.

The Plane Strain Program

All input quantities are expressed in cgs units (cm/sec, dyne/cm^2, etc.) except for acceleration, which is given as a fraction of the acceleration of gravity. The input cards are listed below with an explanation of the variables.

Card No. 1	FORMAT (8A10)
ALPHA (J)	Any desired alphameric identification
Card No. 2	FORMAT(F8.0, I4, F8.0, 2E10.0, I4, I3, 3I1, 2F5.0)
DELT	Time interval for input data points and time step for finite difference computation.
NK	Number of points on input time histories. NK must not exceed 8000
RHON	The density of the elastic substratum.
VPN	The compressional velocity of the elastic substratum.
VSN	The shear velocity of the elastic substratum.
NPT	The total number of nodes in the mesh representing the soil. NPT must not exceed 1000.
JLIM	The number of nodes on the boundary between soil and elastic substratum. JLIM must not exceed 400.
IH	IH = 1 if the horizontal motion of the boundary nodes is specified on cards
IV	IV = 1 if the vertical motion of the boundary nodes is specified on cards.
ITAPE	ITAPE = 1 if an input magnetic tape is provided.
F1, F2	Parameters specifying the response of the low-pass filter. IF either F1 or F2 is zero, no filtering is done.
Card No. 3	FORMAT (E10.0)
FACTOR	A factor by which the coordinates of the nodes on the following set of cards will be multiplied.
Cards No. 4	FORMAT (10F8.0)
(X1(NP), X3(NP), NP = 1, NPT)	Horizontal (X1) and vertical (X3) coordinates of the nodes. X1 increases toward the right and X3 increases upward. NP is the identification number for the node.
Cards No. 5	FORMAT (I1, 4I4, F5.0, 3E10.0)
	One card for each element of the mesh. There must not be more than 900 elements.

MORE	MORE = 1 if additional cards of this type follow.
NP1, NP2, NP3, NP4	The identification numbers of the nodes defining the element, listed in order going counter clockwise around the element. If NP4 = 0 the element is triangular.
RHO	Density of the element.
VP	Compressional velocity of the element.
VS	Small-strain shear velocity of the element.
TAU	Dynamic shear strength of the element.
Cards No. 6	FORMAT (20I4)
(NPBP(J), J=1, JLIM)	The identification numbers of the nodes along the boundary between the soil and the elastic substratum, listed in order of increasing X1.
Card(s) No. 7	FORMAT(I6,11(I4,2I1)/12(I4,2I1))
LIM,(NSV(L),IOPS(L), NCMP(L), L = 1, LIM)	LIM is the number of time histories that are to be saved. Up to 400 time histories can be saved and will be written on an output tape. The first 16 only will also be kept in core. The maxima and minima of these will be listed in the printer output, and they will be available for plotting. NSV is the identification number of the element (or node). If IOPS = 1 , a strain time history for element NSV is saved. If IOPS = 2 , a stress time history is saved for that element. If IOPS = 3 , a particle velocity time history is saved for node NSV. If NCMP = 1 the X1 component of particle velocity or the 11 component of deviatoric stress or strain is saved. If NCMP = 2 the 13 component of deviatoric stress or strain is saved. If NCMP = 3 the X3 component of particle velocity or the 33 component of deviatoric stress or strain is saved. If NCMP = 4 the mean stress or strain is saved.
Cards No. 8	FORMAT(5E14.8) These cards must be omitted if IH = 1 on Card No. 2.
(SMALL(K), K = 1, NK)	Input horizontal acceleration.
Cards No. 9	FORMAT(5E14.8) These cards must be omitted if IV = 1 on Card No. 2.

(SMALL(K), K = 1, NK) Input vertical acceleration.

The remaining cards are input to the plotting program. For each plot one No. 10 card is read followed by the corresponding No. 11 card(s).

Card(s) No. 10 FORMAT(I1,I2,F3.0, F4.0, 2I2,3A10,3A10)

MORE MORE = 1 if this is <u>not</u> the last plot.

NCH Number of time histories to be plotted.

XMAX Total time corresponding to the length of the horizontal axis of the plot, which is 11 inches.

YMAX Total interval in velocity, stress, or strain units corresponding to the length of the vertical axis of the plot, which is 9 inches.

NX Number of divisions on the horizontal axis of the plot.

NY Number of divisions on the vertical axis of the plot.

XALPH Alphameric label of the horizontal axis.

YALPH Alphameric label of the vertical axis.

Card(s) No. 11 FORMAT(12(I2,F4.0))

(LY(J), YBASE(J), J = 1, NCH) If LY = 91 and if IH = 1 on card No. 2, the time history plotted will be the input horizontal particle velocity. If LY = 93 and if IV = 1 on card No. 2, the time history plotted will be the input vertical particle velocity. Otherwise LY refers to the sequence number of the time history on card No. 7 and may not exceed 16.

YBASE is the distance in units of velocity, stress, or strain which the zero line for a given time history is moved up from the bottom line of the plot.

If the vertical or horizontal motion of each boundary node is to be individually specified, then an input magnetic tape must be provided. The tape is unformatted and is read by a statement that begins "READ(17)". The first record is

(ALTP(J), J = 1,8), ITH, ITV

where ALTP is simply alphameric identification. ITH = 1 if horizontal particle velocity is provided and ITV = 1 if vertical particle velocity is provided. There are NK subsequent records if only one component is provided and twice that many if two components are provided. They are of the form

(VTP1(J), J = 1, JLIM)

(VTP3(J), J = 1, JLIM)

where VTP1 is the horizontal particle velocity and VTP3 is the vertical particle velocity specified for each boundary point in order. If both components are specified the records occur in pairs, one pair for each instant of time. The horizontal component is the first member of each pair.

The printer output lists the input parameters and also the maximum and minimum values of up to 16 of the time histories that were saved in the run. The output listing supplied with this report applies to the sample problem, illustrated in Figure 1, for which the input motion is a positive triangular spike of horizontal acceleration with amplitude 0.5 g and duration 0.2 sec followed by a negative triangular spike with the same amplitude and duration followed by 2.6 sec of zero acceleration (Figure 2). Other parameters are given in the output listing. The output is illustrated in Figure 3. A

listing of the input deck is included with this report.

The time histories that are saved are written on magnetic tape without low-pass filtering. The tape is unformatted, and is written by a statement that begins "WRITE(16)". These are NK records, one for each instant of time. Each record consists of the saved values in the same order as on input card(s) No. 7.

The plane strain program as written may require as many as 62,000 (decimal) words of small core memory and 404,000 words of large core memory on the CDC 7600. For large problems running time is essentially proportional to the product of the number of elements and the number of time steps. On the CDC 7600 the program requires about 220 microseconds per element per time step.

Important variables in the plane strain program are listed below with definitions.

ALL(J)	Coordinates in stress space of the "centers" of the yield surfaces. The variable is stored in large core memory.
ALS(J)	The same quantity as above stored in small core memory.
VF1(K), VF3(K)	Horizontal and vertical components of input motion.
SD11(NL), SD13(NL), SD22(NL), SD33(NL)	Components of normalized deviatoric stress for element NL.
SM(NL)	Normalized mean stress for element NL.
ST11, ST13, ST33	Components of normalized total stress.
ED11(NL), ED13(NL), ED33(NL)	Components of normalized deviatoric strain for element NL.
EM(NL)	Normalized mean strain for element NL.
DS11, DS13, DS33	Increments of normalized deviatoric stress.
DE11, DE13, DE33	Increments of normalized total strain.
DEM	Increments of normalized mean strain.
DED11, DED13, DED33	Increments of normalized deviatoric strain.
F1(NP), F3(NP)	Horizontal and vertical components of force on node NP.
V1(NP),V3(NP)	Horizontal and vertical components of particle velocity for node NP.
SIGY(J)	Normalized yield stress for the J yield surface.
N1(NL), N2(NL), N3(NL), N4(NL)	Identification numbers of the nodes that define element NL, listed in order going counterclockwise around the element. If N4(NL) equals zero the element is triangular.
RBM(NL)	Normalized reciprocal bulk modulus of element NL.
PMD(NP)	The mass associated with node NP divided by the time step DELT.
ECL(J)	Coefficients for determining increments of normalized strain in terms of particle velocities at the nodes. Variable is stored in large core memory.
ECS(J)	Same quantity as above stored in small core memory.
FCL(J)	Coefficients for determining force on the nodes in terms of normalized total stress. Variable is stored in large core memory.
FCS(J)	Same quantity as above stored in small core memory.

The Antiplane Strain Program

As with the plane strain program, all input quantities are expressed in cgs units except for acceleration, which is given as a fraction of the acceleration of the acceleration of gravity. The input cards are listed below with an explanation of the variables.

Card No. 1	FORMAT(8A10)
ALPHA(J)	Any desired alphameric identification.
Card No. 2	FORMAT(F8.0, I4, F8.0, E10.0, F5.0, I4, I3, I1, 2F5.0)
DELT	Time interval for input data points and time step for finite difference computations.
NK	Number of points on the input time histories NK must not exceed 8000.
RHON	The density of the elastic substratum.
VSN	The shear velocity of the elastic substratum.
THETA	The angle in degrees between the vertical and the ray direction of an incoming plane shear wave in the elastic substratum. The angle is taken as positive if the shear wave is incident from the left (i.e. the direction in which the horizontal coordinate decreases).
NPT	The total number of nodes in the mesh representing the soil. NPT must not exceed 1000.
JLIM	The number of nodes on the boundary between soil and elastic substratum. JLIM must not exceed 400.
ITAPE	ITAPE = 1 if an input magnetic tape is provided.
F1, F2	Parameters specifying the response of the low-pass filter. If either F1 or F2 is zero no filtering is done.
Card No. 3	FORMAT(E10.0)
FACTOR	A factor by which the coordinates of the nodes on the following set of cards will be multiplied.
Cards No. 4	FORMAT(10F8.0)
(X1(NP),X3(NP), NP = 1,NPT)	Horizontal (X1) and vertical (X3) coordinates of the nodes. X1 increases toward the right and X3 increases upward. NP is the identification number for the node.
Cards No. 5	FORMAT(I1, 4I4, F5.0, 2E10.0)
	One card for each element of the mesh. These must not be more than 1000 elements.

MORE	MORE = 1 if additional cards of this type follow.
NP1, NP2, NP3,NP4	The identification numbers of the nodes defining the element, listed in order going counter clockwise around the element. IF NP4 = 0 the element is triangular.
RHO	Density of the element.
VS	Small-strain shear velocity of the element.
TAU	Dynamic shear strength of the element.
Cards No. 6	FORMAT(20I4)
(NPBP(J),J = 1,JLIM)	The identification numbers of the nodes along the boundary between the soil and the elastic substratum, listed in order of increasing X1.
Card(s) No. 7	FORMAT(I6,11(I4,2I1)/(12(I4,2I1))
LIM,(NSV(L),IOPS(L), NCMP(L), L=1,LIM)	LIM is the number of time histories that are to be saved. Up to 400 time histories can be saved and will be written on an output tape. The first 16 only will also be kept in core. The maxima and minima of these will be printed out and they will be available for plotting. NSV is the identification number of the element (or node). If IOPS = 1, a strain time history for element NSV is saved. If IOPS = 2 a stress time history is saved for that element. If IOPS = 3, a particle velocity time history is saved for node NSV. If NCMP = 1, the 12 component of deviatoric stress or strain is saved. If NCMP = 2, the 23 component of deviatoric stress or strain is saved.
Cards No. 8	FORMAT(5E14.8)
	These cards must be omitted if ITAPE = 1 on Card No. 2
(SMALL(K), K = 1,NK)	Acceleration values for a plane shear wave incident in the elastic substratum.

The remaining cards are input to the plotting program. For each plot one No. 9 card is read followed by the corresponding No. 10 card(s).

Card(s) No. 9	FORMAT(I1,I2,F3.0,F4.0,2I2,3A10,3A10)
MORE	MORE = 1 if this is <u>not</u> the last plot.
NCH	Number of time histories to be plotted.
XMAX	Total time corresponding to the length of the horizontal axis of the plot which is 11 inches.
YMAX	Total interval in velocity, stress, or strain units corresponding to the length of the vertical axis of the plot, which is 9 inches.
NX	Number of divisions on the horizontal axis of the plot .
NY	Number of divisions on the vertical axis of the plot.
XALPH	Alphameric label of the horizontal axis.
YALPH	Alphameric label of the vertical axis.
Card(s) No. 10	FORMAT(12(I2,F4.0))
(LY(J),YBASE(J), J=1,NCH)	If LY = 92 and ITAPE = 1, the time history plotted will be the particle velocity for the shear wave incident in the elastic substratum. Otherwise LY refers to the sequence number of the time history on card No. 7 and may not exceed 16. YBASE is the distance in units of velocity, stress, or strain which the zero line for a given time history is moved up from the bottom of the plot.

If the motion of each boundary node is to be individually specified, then an input magnetic tape must be provided. The tape is unformatted and is read by a statement that begins "READ(17)". The first record is

```
(ALTP(J), J = 1,8)
```

ALTP is simply alphameric identification. There are NK subsequent records, each giving the particle velocities for the boundary nodes at an instant. A record consists simply of the velocity values in order.

The printer output lists the input parameters and also the maximum and minimum values of up to 16 of the time histories that were saved in the run. The output listing supplied with this report applies to the sample problem, illustrated in Figure 1, for which the input motion is a vertically propagating shear wave incident in the elastic substratum. The time history of acceleration is a positive triangular spike of amplitude 0.5 g and duration 0.2 sec followed by a negative triangular spike of the same amplitude and duration followed by 2.6 sec of zero acceleration (Figure 2). Other parameters are given in the output listing. The output is illustrated in Figure 4. A listing of the input deck is included with this report.

The time histories that are saved are written on magnetic tape without low-pass filtering. The tape is unformatted and is written by a statement that begins "WRITE(16)". These are NK records, one for each instant of time. Each record consists of the saved values in the same order as on input card(s) No. 7.

The antiplane strain program as written may require as many as 62,000 (decimal) words of small core memory and 404,000 words of large core memory on the CDC 7600. As with the plane strain program, running time for large

problems is essentially proportional to the product of the number of elements and the number of time steps. On the CDC 7600 the program requires about 120 microseconds per element per time step.

The important variables in the antiplane strain program are named in the same way as those in the plane strain program, for which definitions have already been given. The only difference is the different numbering corresponding to the fact that different components of stress, strain, force, and velocity are involved in the different problems.

Adaptation to Computer Systems Other Than CDC

The programs were written for the CDC 7600 at the Lawrence Berkeley Laboratory, University of California. Modifications will be necessary to run the programs on other systems. The PROGRAM statement at the beginning is a peculiarity of CDC FORTRAN and would have to be removed for use with other systems. The LARGE statements indicate that the listed variables are stored in Large Core Memory (LCM) on the 7600. Data stored in LCM on the 7600 must be transferred to Small Core Memory (SCM) before being used in computation. Data transfer is accomplished by the SMALLIN and SMALLOUT statements. Execution of the statement SMALLIN (A,B,n) causes the transfer of n words from the array B in LCM to the array A in SCM. Execution of the statement SMALLOUT (A, B, n) causes the transfer of n words from the array A in SCM to the array B in LCM. In other systems these variables named in the LARGE statements would have to be stored in main core storage or on a rapid access storage device such as a disk. Substitutes for the SMALLIN and SMALLOUT statements would have to be devised.

Subroutine NLPLT is a plotting subroutine that makes use of subroutines that were written for the CALCOMP plotter. If those subroutines are not available to the user, he will have to modify or replace NLPLT. At the end of both main programs, when the call to NLPLT occurs, the first 16 of time histories that have been saved are available for plotting in an array called VSAVE of dimensions (8000,16). In the planestrain program, if the input motion was specified on cards, the horizontal and vertical time histories of particle velocity for the input are available in arrays VF1 and VF3 with dimension (8000). In the antiplane-strain program, if the input motion was specified on cards, the time history of particle velocity for the input is available in array VF2 with dimension 8000.

REFERENCES

Alterman, Z., and Loewenthal, D., 1972, Computer generated seismograms, in Methods in computational physics, v. 12, Seismology, body waves and sources: Bruce A. Bolt, ed., Academic Press, New York, p. 35-164.

Boore, D. M., 1972, Finite difference methods for seismic wave propagation in heterogeneous materials, in Methods in computational physics, v. 11, Seismology, surface waves and earth oscillations: Bruce A. Bolt, ed., Academic Press, New York, p. 1-37.

Joyner, W. B., 1975, A method for calculating nonlinear seismic response in two dimensions: Seismol. Soc. America Bull., v. 65, p. 1337-1357.

Joyner, W. B., and Chen, A.T.F., 1975, Calculation of nonlinear ground response in earthquakes, Seismol. Soc. America Bull., v. 65, p. 1315-1336.

Figure 1. -- Diagram of soil embankment that constitutes the sample problem. Numbered nodes are those for which output is illustrated in Figures 3 and 4.

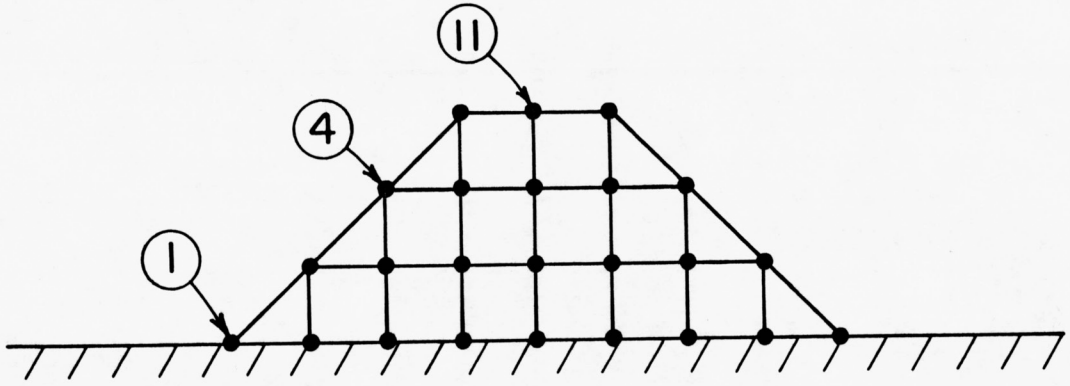

11
4
1

Figure 2. -- Input acceleration time history for sample problem.

SAMPLE PROBLEM 2

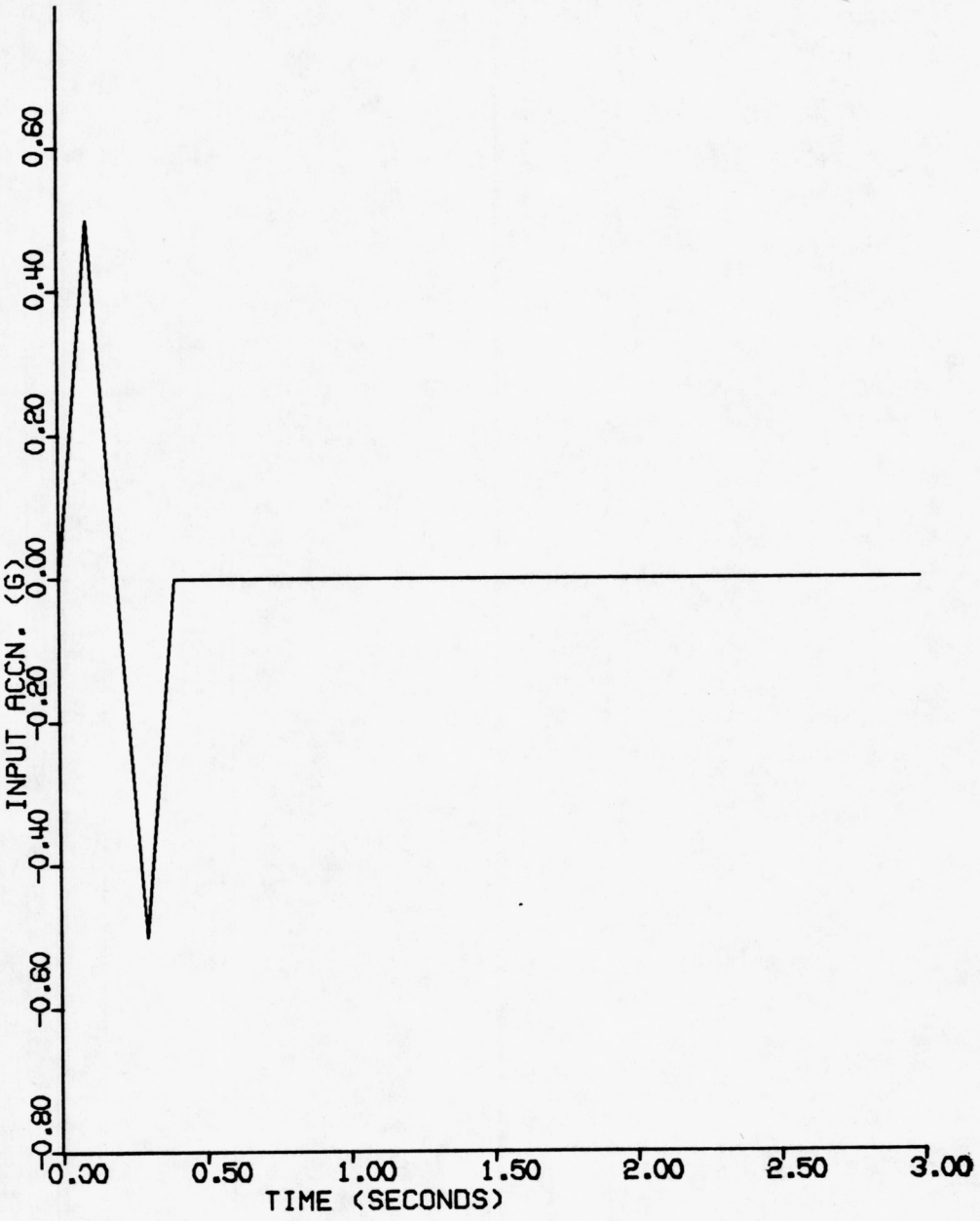


Figure 3. -- Particle velocity time histories for the plane strain problem. Numbers refer to nodes designated on Figure 1.

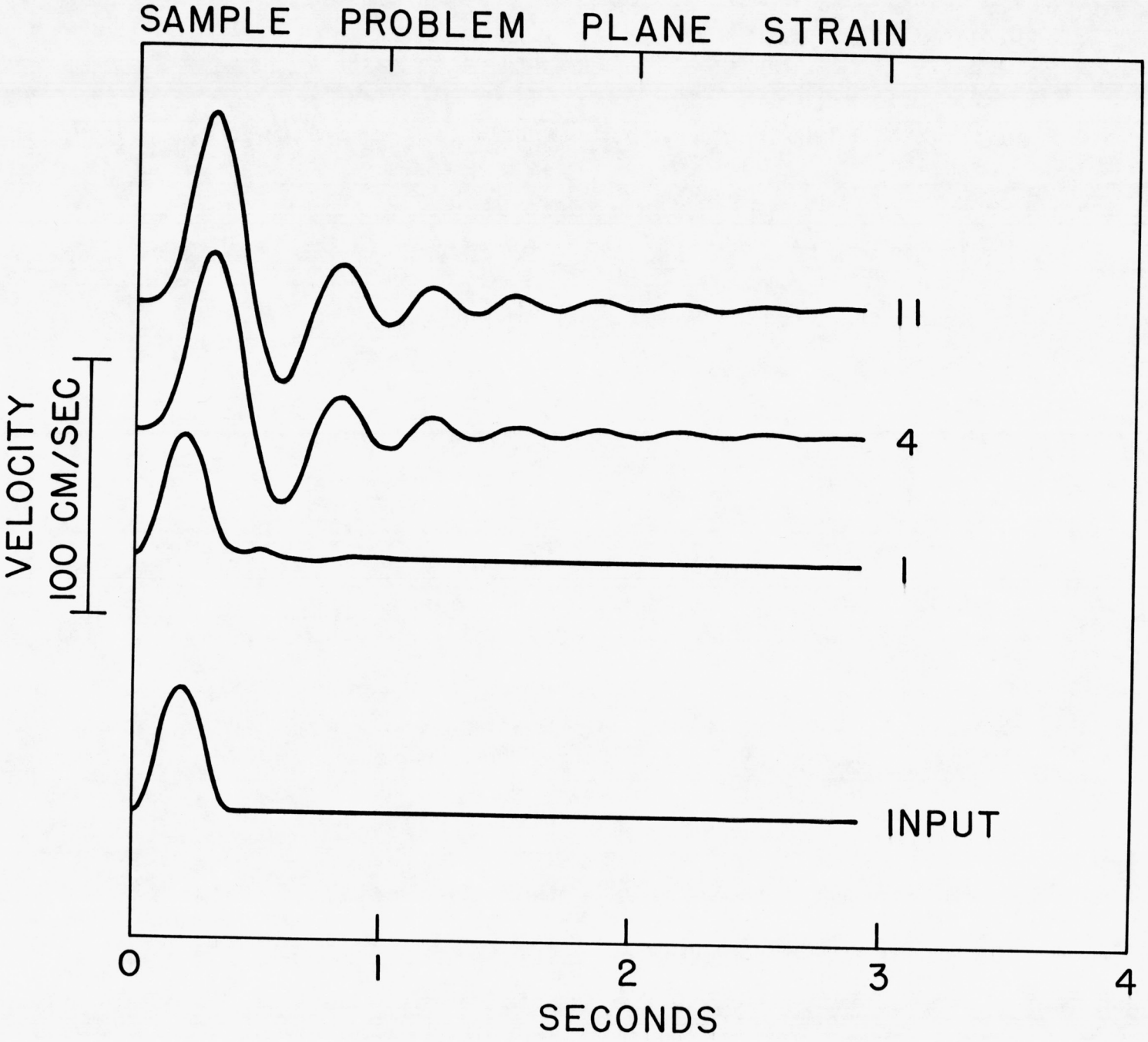
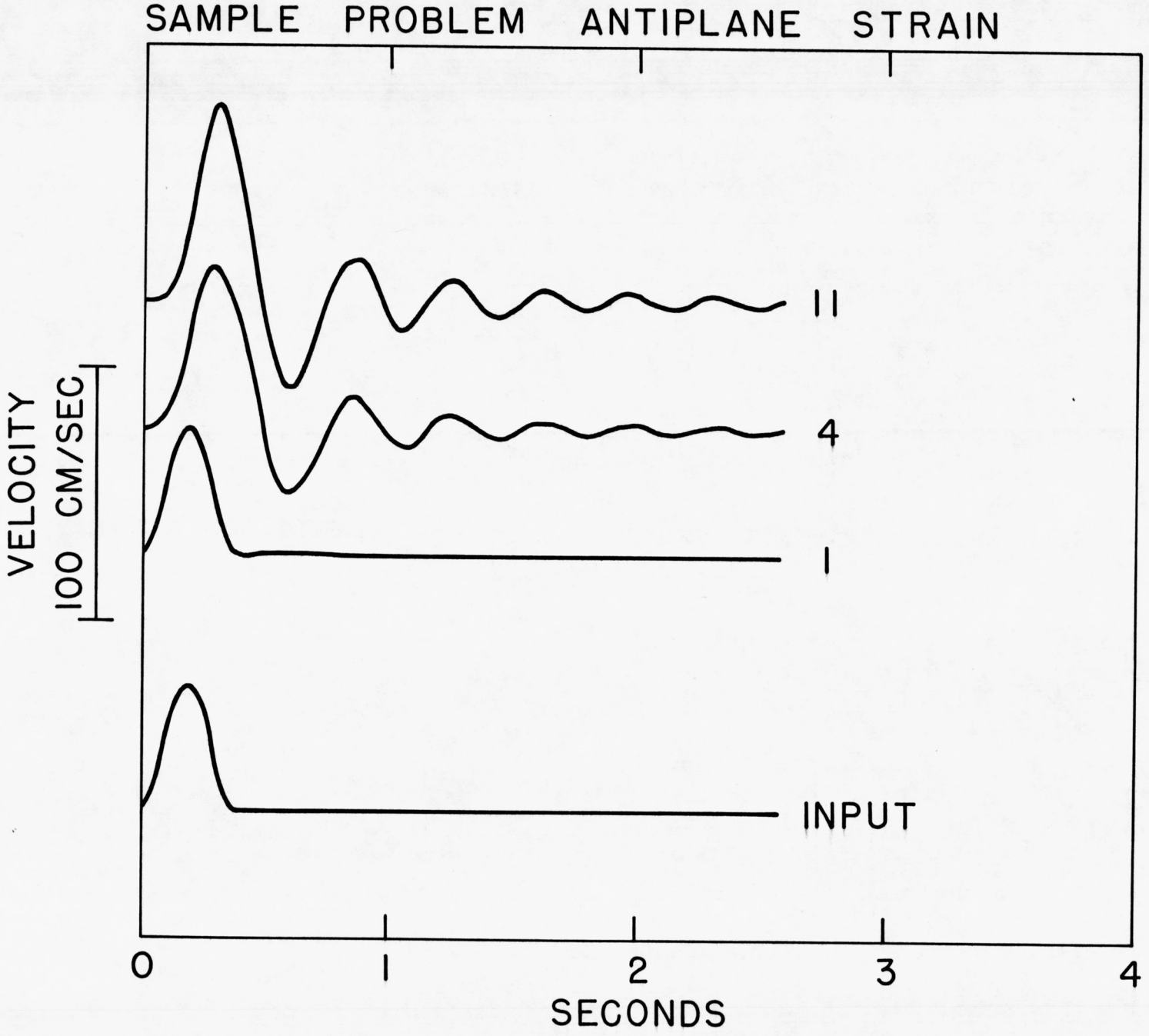

SAMPLE PROBLEM PLANE STRAIN
VELOCITY
100 CM/SEC
II
4
1
INPUT
0
1
2
3
4
SECONDS

Figure 4. -- Particle velocity time histories for the antiplane strain problem. Numbers refer to nodes designated on Figure 1.


SAMPLE PROBLEM ANTIPLANE STRAIN
VELOCITY
100 CM/SEC
II
4
1
INPUT
0
1
2
3
4
SECONDS

PLANESTRAIN PROGRAM

SOURCE LISTING

AND

OUTPUT LISTING FOR SAMPLE PROBLEM

```
      PROGRAM PGM(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE16,
     * TAPE17,TAPE98=101,PLOT,TAPE99=PLOT)
      LARGE ALL(27000),ECL(14400),FCL(7200)
      LARGE VF1(8000),VF3(8000)
      LARGE VSAVE(8000,16)
      REAL SMALL(8002)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      INTEGER ALPHA(8)
      REAL SIGY(10),SIGY2(10),CN(10)
  501 FORMAT(8A10)
  502 FORMAT(1H1,*PLANE STRAIN NONLINEAR GROUND RESPONSE*//
     * 1X,8A10/)
  503 FORMAT(F8.0,I4,F8.0,2E10.0,I4,I3,3I1,2F5.0)
  504 FORMAT(5X,*DELT=*,F10.7/5X,*NK=*,I6/
     * 5X,*RHON=*,F6.3/5X,*VPN=*,E12.4/5X,*VSN=*,E12.4/
     * 5X,*NPT=*,I6/5X,*JLIM=*,I5/5X,*IH=*,I2/5X,*IV=*,I2/
     * 5X,*ITAPE=*,I2/5X,*F1=*,F6.2/5X,*F2=*,F6.2)
  505 FORMAT(5E14.8)
  506 FORMAT(///1X,*EXTREME VALUES*/)
  507 FORMAT(3X,*INPUT HORIZONTAL*/)
  508 FORMAT(3X,*INPUT VERTICAL*/)
  509 FORMAT(3X,*L=*,I3,3X,*NSV=*,I5,
     * 3X,*IOPS=*,I2,3X,*NCMP=*,I2/)
      READ(5,501) (ALPHA(J),J=1,8)
   10 WRITE(6,502) (ALPHA(J),J=1,8)
   16 READ(5,503) DELT,NK,RHON,VPN,VSN,NPT,JLIM,
     * IH,IV,ITAPE,F1,F2
   52 WRITE(6,504) DELT,NK,RHON,VPN,VSN,NPT,JLIM,
     * IH,IV,ITAPE,F1,F2
  106 CALL SETUP(DELT,RHON,VPN,VSN,MAX,SIGY,SIGY2,CN,
     * NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
  125 LSV=LIM
  126 IF(LSV.GT.16) LSV=16
  132 DO 1 K=1,NK
  140 VF1(K)=0.0
  141 1 VF3(K)=0.0
  143 G=980.0
  144 IF(IH.NE.1) GO TO 11
  146 READ(5,505) (SMALL(K),K=1,NK)
  155 VOK=0.0
  156 DO 12 K=1,NK
  165 VOK=VOK+G*DELT*SMALL(K)
  167 12 SMALL(K)=VOK
  171 SMALLOUT(SMALL,VF1,NK)
  177 11 IF(IV.NE.1) GO TO 13
  201 READ(5,505) (SMALL(K),K=1,NK)
  210 VOK=0.0
  211 DO 14 K=1,NK
  220 VOK=VOK+G*DELT*SMALL(K)
  222 14 SMALL(K)=VOK
  224 SMALLOUT(SMALL,VF3,NK)
  232 13 WRITE(16) (ALPHA(J),J=1,8)
  237 CALL NLPSS(DELT,NK,VF1,VF3,VSAVE,ITAPE,
     * MAX,SIGY,SIGY2,CN,NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
  264 WRITE(6,506)
  270 IF(IH.NE.1) GO TO 21
```

```
PGM
    272         WRITE(6,507)
    276         SMALLIN(SMALL,VF1,NK)
    304         CALL XTRM(SMALL,NK)
    306      21 IF(IV.NE.1) GO TO 22
    310         WRITE(6,508)
    314         SMALLIN(SMALL,VF3,NK)
    322         CALL XTRM(SMALL,NK)
    324      22 NKM=NK
    325         IF(F1.EQ.0.0.OR.F2.EQ.0.0) GO TO 24
    334         NKM=NKM-20
    335      24 DO 23 L=1,LSV
    337         WRITE(6,509) L,NSV(L),IOPS(L),NCMP(L)
    355         SMALLIN(SMALL,VSAVE(1,L),NK)
    366         IF(F1.EQ.0.0.OR.F2.EQ.0.0) GO TO 23
    374         CALL FILTER(SMALL,F1,F2,80,DELT,NK)
    400         SMALLOUT(SMALL,VSAVE(1,L),NK)
    411      23 CALL XTRM(SMALL,NKM)
    416         CALL NLPLT(VF1,VF3,VSAVE,SMALL,NKM,DELT,ALPHA)
    431         STOP
    433         END


PROGRAM LENGTH INCLUDING I/O BUFFERS

25531

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
11     - 000200     13     - 000233     21     - 000307     22     - 000325     23     - 000412     24     - 000336
501    - 000453     502    - 000455     503    - 000466     504    - 000473     505    - 000524     506    - 000526
507    - 000532     508    - 000536     509    - 000542

BLOCK NAMES AND LENGTHS
  'LCM'- 0570130

VARIABLE ASSIGNMENTS
ALL    - 0000000/LC ALPHA  - 022554     CN     - 022610     DELT   - 022623     ECL    - 0064570/LC FCL    - 0120670/LC
F1     - 022635     F2     - 022636     G      - 022644     IH     - 022632     IOPS   - 020274     ITAPE  - 022634
IV     - 022633     J      - 022622     JLIM   - 022631     K      - 022643     L      - 022647     LIM    - 022641
LSV    - 022642     MAX    - 022637     NCMP   - 021114     NK     - 022624     NKM    - 022646     NLT    - 022640
NPT    - 022630     NSV    - 021734     RHON   - 022625     SIGY   - 022564     SIGY2  - 022576     SMALL  - 000572
VF1    - 0136730/LC VF3    - 0156430/LC VOK    - 022645     VPN    - 022626     VSAVE  - 0176130/LC VSN    - 022627

START OF CONSTANTS-000436       TEMPS--000563    INDIRECTS-000567

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 045100
```

```
NLPSS
      SUBROUTINE NLPSS(DELT,NK,VF1,VF3,VSAVE,ITAPE,
     * MAX,SIGY,SIGY2,CN,NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
C       VERSION OF 9/26/75
      LARGE ALL(27000),ECL(14400),FCL(7200)
      LARGE VF1(8000),VF3(8000)
      LARGE VSAVE(8000,16)
      REAL ALS(3),ECS(16),FCS(8)
      REAL A1(3),A2(3),A3(3),B(3)
      INTEGER ALTP(8)
      INTEGER MPIV(2)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      REAL VTAPE(400)
      REAL SM(900)
      REAL SD11(900),SD13(900),SD22(900),SD33(900)
      REAL ED11(900),ED13(900),ED33(900),EM(900)
      REAL F1(1000),F3(1000)
      REAL V1(1000),V3(1000)
      REAL VTP1(400),VTP3(400)
      REAL SIGY(10),SIGY2(10),CN(10)
      COMMON ERA(900),TAUA(900)
      COMMON N1(900),N2(900),N3(900),N4(900)
      COMMON RBM(900)
      COMMON IPTP(1000)
      COMMON X1(1000),X3(1000)
      COMMON PMD(1000)
      COMMON NPBP(400)
      COMMON CF11(400),CF13(400),CF31(400),CF33(400)
  501 FORMAT(///1X,8A10/3X,*ITH=*,I2/3X,*ITV=*,I2)
      ITH=0
   23 ITV=0
   24 IF(ITAPE.NE.1) GO TO 10
   26 READ(17) (ALTP(J),J=1,8),ITH,ITV
   37 WRITE(6,501) (ALTP(J),J=1,8),ITH,ITV
   60 10 LSV=LIM
   61 IF(LSV.GT.16) LSV=16
   65 DO 1 NL=1,NLT
   67 INDL=(NL-1)*30
   71 SM(NL)=0.0
   72 SD11(NL)=0.0
   73 SD13(NL)=0.0
   74 SD22(NL)=0.0
   75 SD33(NL)=0.0
   76 EM(NL)=0.0
   77 ED11(NL)=0.0
  100 ED13(NL)=0.0
  101 ED33(NL)=0.0
  103 DO 1 MM=1,MAX
  105 ALS(1)=0.0
  105 ALS(2)=0.0
  106 ALS(3)=0.0
  106 INDX=INDL+(MM-1)*3+1
  112   1 SMALLOUT(ALS,ALL(INDX),3)
  126 DO 11 NP=1,NPT
  134 F1(NP)=0.0
  135 F3(NP)=0.0
  135 V1(NP)=0.0
```

```
NLPSS
     135     11 V3(NP)=0.0
     137        DO 21 K=1,NK
     144        DO 22 NL=1,NLT
     145        INDL=(NL-1)*30
     147        INDE=(NL-1)*16+1
     151        INDF=(NL-1)*8+1
     153        N1T=N1(NL)
     155        N2T=N2(NL)
     157        N3T=N3(NL)
     161        N4T=N4(NL)
     164        SMALLIN(ECS,ECL(INDE),16)
     201        DE11=ECS(1)*V1(N1T)+ECS(5)*V1(N2T)+ECS(9)*V1(N3T)
     206        DE13=ECS(2)*V1(N1T)+ECS(3)*V3(N1T)+ECS(6)*V1(N2T)+
               * ECS(7)*V3(N2T)+ECS(10)*V1(N3T)+ECS(11)*V3(N3T)
     223        DE33=ECS(4)*V3(N1T)+ECS(8)*V3(N2T)+ECS(12)*V3(N3T)
     231        IF(N4T.EQ.0) GO TO 23
     235        DE11=DE11+ECS(13)*V1(N4T)
     237        DE13=DE13+ECS(14)*V1(N4T)+ECS(15)*V3(N4T)
     243        DE33=DE33+ECS(16)*V3(N4T)
     246     23 DEM=(DE11+DE33)/3.0
     251        DED11=DE11-DEM
     252        DED13=DE13
     253        DED33=DE33-DEM
     254        SM(NL)=SM(NL)+RBM(NL)*DEM
     260        DO 31 JJ=1,3
     265        A1(JJ)=0.0
     266        A2(JJ)=0.0
     266     31 A3(JJ)=0.0
     267        A1(1)=1.0
     270        A2(2)=1.0
     271        A3(3)=1.0
     273        DO 32 MM=1,MAX
     300        INDX=INDL+(MM-1)*3+1
     303        SMALLIN(ALS,ALL(INDX),3)
     310        CA11=SD11(NL)-ALS(1)
     312        CA13=SD13(NL)-ALS(2)
     315        CA22=SD22(NL)+ALS(1)+ALS(3)
     321        CA33=SD33(NL)-ALS(3)
     324        TEMP=CA13*CA13
     326        CALL=CA11*CA11+CA22*CA22+CA33*CA33+TEMP+TEMP
     333        IF(CALL.LE.SIGY2(MM)) GO TO 32
     337        FACT=CN(MM)*CALL
     341        TEMP=(CA11-CA22)/FACT
     344        A1(1)=A1(1)+CA11*TEMP
     346        A1(2)=A1(2)+CA13*TEMP
     351        A1(3)=A1(3)+CA33*TEMP
     353        TEMP=(CA13+CA13)/FACT
     354        A2(1)=A2(1)+CA11*TEMP
     357        A2(2)=A2(2)+CA13*TEMP
     360        A2(3)=A2(3)+CA33*TEMP
     362        TEMP=(CA33-CA22)/FACT
     364        A3(1)=A3(1)+CA11*TEMP
     367        A3(2)=A3(2)+CA13*TEMP
     371        A3(3)=A3(3)+CA33*TEMP
     373        FACT=SIGY(MM)/SQRT(CALL)
     377        ALS(1)=SD11(NL)-FACT*CA11
```

```
NLPSS
     403          ALS(2)=SD13(NL)-FACT*CA13
     406          ALS(3)=SD33(NL)-FACT*CA33
     412          SMALLOUT(ALS,ALL(INDX),3)
     424       32 CONTINUE
     427          B(1)=DED11
     430          B(2)=DED13
     431          B(3)=DED33
     433          PIV=0.0
     434          DO 33 JJ=1,3
     435          IF(ABS(A1(JJ)).LE.ABS(PIV)) GO TO 33
     442          PIV=A1(JJ)
     444          MPVT=JJ
     446       33 CONTINUE
     450          MPIV(1)=MPVT
     451          A2(MPVT)=A2(MPVT)/PIV
     454          A3(MPVT)=A3(MPVT)/PIV
     455          B(MPVT)=B(MPVT)/PIV
     457          MPVP=MPVT
     460          PIV=0.0
     461          DO 34 JJ=1,3
     462          IF(JJ.EQ.MPIV(1)) GO TO 34
     465          A2(JJ)=A2(JJ)-A1(JJ)*A2(MPVP)
     471          A3(JJ)=A3(JJ)-A1(JJ)*A3(MPVP)
     474          B(JJ)=B(JJ)-A1(JJ)*B(MPVP)
     500          IF(ABS(A2(JJ)).LE.ABS(PIV)) GO TO 34
     506          PIV=A2(JJ)
     510          MPVT=JJ
     511       34 CONTINUE
     513          MPIV(2)=MPVT
     514          A3(MPVT)=A3(MPVT)/PIV
     517          B(MPVT)=B(MPVT)/PIV
     520          MPVP=MPVT
     521          DO 35 JJ=1,3
     523          IF(JJ.EQ.MPIV(1)) GO TO 35
     524          IF(JJ.EQ.MPIV(2)) GO TO 35
     527          A3(JJ)=A3(JJ)-A2(JJ)*A3(MPVP)
     533          B(JJ)=B(JJ)-A2(JJ)*B(MPVP)
     537          MPVT=JJ
     540       35 CONTINUE
     547          DS33=B(MPVT)/A3(MPVT)
     551          MPVT=MPIV(2)
     553          DS13=B(MPVT)-A3(MPVT)*DS33
     557          MPVT=MPIV(1)
     560          DS11=B(MPVT)-A2(MPVT)*DS13-A3(MPVT)*DS33
     566          SD11(NL)=SD11(NL)+DS11
     567          SD13(NL)=SD13(NL)+DS13
     571          SD22(NL)=SD22(NL)-DS11-DS33
     574          SD33(NL)=SD33(NL)+DS33
     576          EM(NL)=EM(NL)+DEM
     601          ED11(NL)=ED11(NL)+DED11
     604          ED13(NL)=ED13(NL)+DED13
     607          ED33(NL)=ED33(NL)+DED33
     612          ST11=SD11(NL)+SM(NL)
     614          ST13=SD13(NL)
     616          ST33=SD33(NL)+SM(NL)
     620          SMALLIN(FCS,FCL(INDF),8)
```

```
NLPSS
    626         F1(N1T)=F1(N1T)+FCS(1)*ST11+FCS(2)*ST13
    633         F3(N1T)=F3(N1T)+FCS(1)*ST13+FCS(2)*ST33
    640         F1(N2T)=F1(N2T)+FCS(3)*ST11+FCS(4)*ST13
    645         F3(N2T)=F3(N2T)+FCS(3)*ST13+FCS(4)*ST33
    652         F1(N3T)=F1(N3T)+FCS(5)*ST11+FCS(6)*ST13
    657         F3(N3T)=F3(N3T)+FCS(5)*ST13+FCS(6)*ST33
    664         IF(N4T.EQ.0) GO TO 22
    665         F1(N4T)=F1(N4T)+FCS(7)*ST11+FCS(8)*ST13
    672         F3(N4T)=F3(N4T)+FCS(7)*ST13+FCS(8)*ST33
    677      22 CONTINUE
    702         DO 51 NP=1,NPT
    703         IF(IPTP(NP).NE.0) GO TO 51
    707         TKNP=1.0/PMD(NP)
    711         V1(NP)=V1(NP)+TKNP*F1(NP)
    715         V3(NP)=V3(NP)+TKNP*F3(NP)
    720         F1(NP)=0.0
    721         F3(NP)=0.0
    721      51 CONTINUE
    724         VF1X=VF1(K)
    726         VF3X=VF3(K)
    730         IF(ITH.EQ.1) READ(17) (VTP1(J),J=1,JLIM)
    744         IF(ITV.EQ.1) READ(17) (VTP3(J),J=1,JLIM)
    760         DO 52 J=1,JLIM
    762         NPX=NPBP(J)
    764         PMDX=PMD(NPX)
    766         IF(ITH.EQ.1) VF1X=VTP1(J)
    773         IF(ITV.EQ.1) VF3X=VTP3(J)
   1003         F1X=F1(NPX)
   1004         F3X=F3(NPX)
   1006         V1X=V1(NPX)
   1007         V3X=V3(NPX)
   1011         V1(NPX)=VF1X+CF11(J)*(F1X+PMDX*(V1X-VF1X))+
               * CF13(J)*(F3X+PMDX*(V3X-VF3X))
   1023         V3(NPX)=VF3X+CF31(J)*(F1X+PMDX*(V1X-VF1X))+
               * CF33(J)*(F3X+PMDX*(V3X-VF3X))
   1036         F1(NPX)=0.0
   1040      52 F3(NPX)=0.0
   1045         DO 61 L=1,LIM
   1046         NSVL=NSV(L)
   1050         IF(IOPS(L).EQ.1) GO TO 63
   1053         IF(IOPS(L).EQ.2) GO TO 64
   1055         IF(NCMP(L).EQ.1) VTAPE(L)=V1(NSVL)
   1063         IF(NCMP(L).EQ.3) VTAPE(L)=V3(NSVL)
   1072         GO TO 61
   1073      63 IF(NCMP(L).EQ.1) VTAPE(L)=ED11(NSVL)*ERA(NSVL)
   1102         IF(NCMP(L).EQ.2) VTAPE(L)=ED13(NSVL)*ERA(NSVL)
   1111         IF(NCMP(L).EQ.3) VTAPE(L)=ED33(NSVL)*ERA(NSVL)
   1120         IF(NCMP(L).EQ.4) VTAPE(L)=EM(NSVL)*ERA(NSVL)
   1127         GO TO 61
   1130      64 IF(NCMP(L).EQ.1) VTAPE(L)=SD11(NSVL)*TAUA(NSVL)
   1137         IF(NCMP(L).EQ.2) VTAPE(L)=SD13(NSVL)*TAUA(NSVL)
   1146         IF(NCMP(L).EQ.3) VTAPE(L)=SD33(NSVL)*TAUA(NSVL)
   1155         IF(NCMP(L).EQ.4) VTAPE(L)=SM(NSVL)*TAUA(NSVL)
   1164      61 CONTINUE
   1167         DO 62 L=1,LSV
   1176      62 VSAVE(K,L)=VTAPE(L)
```

```
NLPSS
   1201          WRITE(16) (VTAPE(L),L=1,LIM)
   1216       21 CONTINUE
   1220          RETURN
   1220          END


SUBPROGRAM LENGTH

33420

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
10      - 000061     22      - 000700     23      - 000247     32      - 000425     33      - 000447     34      - 000512
35      - 000541     51      - 000722     61      - 001165     63      - 001074     64      - 001131     501     - 001224

BLOCK NAMES AND LENGTHS
  'LCM'- 0136730
 //     - 030014/01

VARIABLE ASSIGNMENTS
ALL     - 0000000/LC ALS     - 001270     ALTP    - 001337     A1      - 001323     A2      - 001326     A3      - 001331
B       - 001334     CALL    - 033372     CA11    - 033365     CA13    - 033366     CA22    - 033367     CA33    - 033370
CF11    - 024714/01  CF13    - 025534/01  CF31    - 026354/01  CF33    - 027174/01  CN      - 000003     DED11   - 033361
DED13   - 033362     DED33   - 033363     DEM     - 033360     DE11    - 033355     DE13    - 033356     DE33    - 033357
DS11    - 033401     DS13    - 033400     DS33    - 033377     ECL     - 0064570/LC ECS     - 001273     ED11    - 013015
ED13    - 014621     ED33    - 016425     EM      - 020231     ERA     - 000000/01  FACT    - 033373     FCL     - 0120670/LC
FCS     - 001313     F1      - 022035     F1X     - 033412     F3      - 024005     F3X     - 033413     INDE    - 033347
INDF    - 033350     INDL    - 033342     INDX    - 033344     IOPS    - 000007     IPTP    - 014234/01  ITH     - 033335
ITV     - 033336     J       - 033337     JJ      - 033364     JLIM    - 000006     K       - 033346     L       - 033416
LIM     - 000012     LSV     - 033340     MAX     - 000000     MM      - 033343     MPIV    - 001347     MPVP    - 033376
MPVT    - 033375     NCMP    - 000010     NL      - 033341     NLT     - 000005     NP      - 033345     NPBP    - 024074/01
NPT     - 000004     NPX     - 033410     NSV     - 000011     NSVL    - 033417     N1      - 003410/01  N1T     - 033351
N2      - 005214/01  N2T     - 033352     N3      - 007020/01  N3T     - 033353     N4      - 010624/01  N4T     - 033354
PIV     - 033374     PMD     - 022124/01  PMDX    - 033411     RBM     - 012430/01  SD11    - 003775     SD13    - 005601
SD22    - 007405     SD33    - 011211     SIGY    - 000001     SIGY2   - 000002     SM      - 002171     ST11    - 033402
ST13    - 033403     ST33    - 033404     TAUA    - 001604/01  TEMP    - 033371     TKNP    - 033405     VF1X    - 033406
VF3X    - 033407     VTAPE   - 001351     VTP1    - 031675     VTP3    - 032515     V1      - 025755     V1X     - 033414
V3      - 027725     V3X     - 033415     X1      - 016204/01  X3      - 020154/01

START OF CONSTANTS-001223       TEMPS--001236    INDIRECTS-001260

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 047700
```

```
SETUP
      SUBROUTINE SETUP(DELT,RHON,VPN,VSN,MAX,SIGY,SIGY2,CN,
     * NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
      LARGE ALL(27000),ECL(14400),FCL(7200)
      REAL ECS(16),FCS(8)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      REAL SIGY(10),SIGY2(10),CN(10)
      REAL A1(3),A2(3),A3(3),B1(3),B2(3),B3(3),B4(3)
      INTEGER MPIV(3)
      COMMON ERA(900),TAUA(900)
      COMMON N1(900),N2(900),N3(900),N4(900)
      COMMON RBM(900)
      COMMON IPTP(1000)
      COMMON X1(1000),X3(1000)
      COMMON PMD(1000)
      COMMON NPBP(400)
      COMMON CF11(400),CF13(400),CF31(400),CF33(400)
  501 FORMAT(E10.0)
  502 FORMAT(10F8.0)
  503 FORMAT(///1X,*POINT LIST*//3X,*FACTOR=*,E14.5//
     * 1X,5(4X,*NP*,7X,*X1*,7X,*X3*)/)
  504 FORMAT(1X,I6,2F9.2,I6,2F9.2,I6,2F9.2,I6,2F9.2)
  505 FORMAT(///1X,*ELEMENT LIST*//1X,4X,*NL*,6X,*N1*,6X,*N2*,
     * 6X,*N3*,6X,*N4*,7X,*RHO*,12X,*VP*,12X,*VS*,11X,*TAU*/)
  506 FORMAT(I1,4I4,F5.0,3E10.0)
  507 FORMAT(1X,I6,4I8,F10.4,3E14.4)
  511 FORMAT(20I4)
  512 FORMAT(///1X,*BOUNDARY POINTS*//(3X,10I8))
  513 FORMAT(I6,11(I4,2I1)/(12(I4,2I1)))
  514 FORMAT(///1X,*SAVE POINTS*//4X,*LIM=*,I4//(1X,I7,2I2,
     * I7,2I2,I7,2I2,I7,2I2,I7,2I2,I7,2I2,I7,2I2,I7,2I2,
     * I7,2I2,I7,2I2))
  516 FORMAT(//1X,*POINT NUMBER EXCEEDS NPT - JOB TERMINATED*)
      CALL MODEL(MAX,SIGY,SIGY2,CN)
   24 DO 10 NP=1,NPT
   35 IPTP(NP)=0
   36 10 PMD(NP)=0.0
   37 READ(5,501) FACTOR
   50 READ(5,502) (X1(NP),X3(NP),NP=1,NPT)
   76 WRITE(6,503) FACTOR
  107 WRITE(6,504) (NP,X1(NP),X3(NP),NP=1,NPT)
  137 DO 11 NP=1,NPT
  151 X1(NP)=FACTOR*X1(NP)
  152 11 X3(NP)=FACTOR*X3(NP)
  154 MORE=1
  154 NL=0
  156 WRITE(6,505)
  171 101 IF(MORE.NE.1) GO TO 900
  173 READ(5,506) MORE,NP1,NP2,NP3,NP4,RHO,VP,VS,TAU
  221 NL=NL+1
  223 WRITE(6,507) NL,NP1,NP2,NP3,NP4,RHO,VP,VS,TAU
  254 IF(NP1.GT.NPT.OR.NP2.GT.NPT.OR.NP3.GT.NPT.OR.NP4.GT.NPT)
     * GO TO 901
  302 N1(NL)=NP1
  302 N2(NL)=NP2
  304 N4(NL)=NP4
  307 N3(NL)=NP3
```

```
SETUP
    311         GMAX=RHO*VS*VS
    314         RBM(NL)=(3.0*RHO*VP*VP-4.0*GMAX)/(2.0*GMAX)
    321         ER=TAU/(2.0*GMAX)
    324         ERA(NL)=ER
    325         TAUA(NL)=TAU
    327         ERI=DELT/ER
    331         INDE=(NL-1)*16+1
    334         INDF=(NL-1)*8+1
    335         TEMP=SQRT((X1(NP3)-X1(NP1))**2+(X3(NP3)-X3(NP1))**2)
    357         BX1=(X1(NP3)-X1(NP1))/TEMP
    362         BX3=(X3(NP3)-X3(NP1))/TEMP
    364         BY1=(X3(NP3)-X3(NP1))/TEMP
    366         BY3=-(X1(NP3)-X1(NP1))/TEMP
    371         YP2=BY1*(X1(NP2)-X1(NP1))+BY3*(X3(NP2)-X3(NP1))
    377         XP3=BX1*(X1(NP3)-X1(NP1))+BX3*(X3(NP3)-X3(NP1))
    406         XP2=BX1*(X1(NP2)-X1(NP1))+BX3*(X3(NP2)-X3(NP1))
    415         AA=XP2/YP2
    416         BB=(XP2-XP3)/YP2
    420         CC=XP3
    421         FMO=YP2*YP2*(BB-AA)/2.0+YP2*CC
    426         FMX=YP2*YP2*YP2*(BB*BB-AA*AA)/6.0+YP2*YP2*BB*
               * CC/2.0+YP2*CC*CC/2.0
    437         FMY=YP2*YP2*YP2*(BB-AA)/3.0+YP2*YP2*CC/2.0
    445         IF(NP4.EQ.0) GO TO 120
    457         XP4=BX1*(X1(NP4)-X1(NP1))+BX3*(X3(NP4)-X3(NP1))
    466         YP4=BY1*(X1(NP4)-X1(NP1))+BY3*(X3(NP4)-X3(NP1))
    473         AA=XP4/YP4
    474         BB=(XP4-XP3)/YP4
    477         FMO=FMO-YP4*YP4*(BB-AA)/2.0-YP4*CC
    504         FMX=FMX-YP4*YP4*YP4*(BB*BB-AA*AA)/6.0-YP4*YP4*BB*CC/2.0
               * -YP4*CC*CC/2.0
    516         FMY=FMY-YP4*YP4*YP4*(BB-AA)/3.0-YP4*YP4*CC/2.0
    530    120  XC=FMX/FMO
    531         YC=FMY/FMO
    533         X1C=X1(NP1)+BX1*XC+BY1*YC
    537         X3C=X3(NP1)+BX3*XC+BY3*YC
    543         X1P1=X1(NP1)-X1C
    545         X3P1=X3(NP1)-X3C
    547         X1P2=X1(NP2)-X1C
    551         X3P2=X3(NP2)-X3C
    554         X1P3=X1(NP3)-X1C
    556         X3P3=X3(NP3)-X3C
    561         IF(NP4.EQ.0) GO TO 121
    562         X1P4=X1(NP4)-X1C
    565         X3P4=X3(NP4)-X3C
    570    121  A1(1)=3.0
    571         IF(NP4.NE.0) A1(1)=4.0
    574         A1(2)=X1P1+X1P2+X1P3
    577         IF(NP4.NE.0) A1(2)=A1(2)+X1P4
    602         A1(3)=X3P1+X3P2+X3P3
    605         IF(NP4.NE.0) A1(3)=A1(3)+X3P4
    610         A2(1)=A1(2)
    611         A2(2)=X1P1*X1P1+X1P2*X1P2+X1P3*X1P3
    615         IF(NP4.NE.0) A2(2)=A2(2)+X1P4*X1P4
    621         A2(3)=X1P1*X3P1+X1P2*X3P2+X1P3*X3P3
    626         IF(NP4.NE.0) A2(3)=A2(3)+X1P4*X3P4
```

```
SETUP
      632         A3(1)=A1(3)
      633         A3(2)=A2(3)
      635         A3(3)=X3P1*X3P1+X3P2*X3P2+X3P3*X3P3
      641         IF(NP4.NE.0) A3(3)=A3(3)+X3P4*X3P4
      644         B1(1)=1.0
      645         B1(2)=X1P1
      647         B1(3)=X3P1
      650         B2(1)=1.0
      651         B2(2)=X1P2
      653         B2(3)=X3P2
      654         B3(1)=1.0
      655         B3(2)=X1P3
      657         B3(3)=X3P3
      660         IF(NP4.EQ.0) GO TO 122
      662         B4(1)=1.0
      663         B4(2)=X1P4
      664         B4(3)=X3P4
      666     122 PIV=0.0
      667         DO 123 JJ=1,3
      671         IF(ABS(A1(JJ)).LE.ABS(PIV)) GO TO 123
      676         PIV=A1(JJ)
      700         MPVT=JJ
      702     123 CONTINUE
      704         MPIV(1)=MPVT
      705         A2(MPVT)=A2(MPVT)/PIV
      710         A3(MPVT)=A3(MPVT)/PIV
      711         B1(MPVT)=B1(MPVT)/PIV
      713         B2(MPVT)=B2(MPVT)/PIV
      715         B3(MPVT)=B3(MPVT)/PIV
      717         IF(NP4.NE.0) B4(MPVT)=B4(MPVT)/PIV
      723         MPVP=MPVT
      724         PIV=0.0
      725         DO 124 JJ=1,3
      727         IF(JJ.EQ.MPIV(1)) GO TO 124
      732         A2(JJ)=A2(JJ)-A1(JJ)*A2(MPVP)
      736         A3(JJ)=A3(JJ)-A1(JJ)*A3(MPVP)
      741         B1(JJ)=B1(JJ)-A1(JJ)*B1(MPVP)
      745         B2(JJ)=B2(JJ)-A1(JJ)*B2(MPVP)
      751         B3(JJ)=B3(JJ)-A1(JJ)*B3(MPVP)
      755         IF(NP4.NE.0) B4(JJ)=B4(JJ)-A1(JJ)*B4(MPVP)
      764         IF(ABS(A2(JJ)).LE.ABS(PIV)) GO TO 124
      772         PIV=A2(JJ)
      774         MPVT=JJ
      775     124 CONTINUE
      777         MPIV(2)=MPVT
     1000         A3(MPVT)=A3(MPVT)/PIV
     1003         B1(MPVT)=B1(MPVT)/PIV
     1004         B2(MPVT)=B2(MPVT)/PIV
     1006         B3(MPVT)=B3(MPVT)/PIV
     1010         IF(NP4.NE.0) B4(MPVT)=B4(MPVT)/PIV
     1014         MPVP=MPVT
     1016         DO 125 JJ=1,3
     1017         IF(JJ.EQ.MPIV(1)) GO TO 125
     1020         IF(JJ.EQ.MPIV(2)) GO TO 125
     1023         A3(JJ)=A3(JJ)-A2(JJ)*A3(MPVP)
     1027         B1(JJ)=B1(JJ)-A2(JJ)*B1(MPVP)
```

```
SETUP
      B2(JJ)=B2(JJ)-A2(JJ)*B2(MPVP)
      B3(JJ)=B3(JJ)-A2(JJ)*B3(MPVP)
      IF(NP4.NE.0) B4(JJ)=B4(JJ)-A2(JJ)*B4(MPVP)
      MPVT=JJ
  125 CONTINUE
      MPIV(3)=MPVT
      AJ3=B1(MPVT)/A3(MPVT)
      MPVT=MPIV(2)
      AJ1=B1(MPVT)-A3(MPVT)*AJ3
      MPVT=MPIV(1)
      AJ0=B1(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
      PMD(NP1)=PMD(NP1)+RHO*FMO*AJ0/DELT
      ECS(1)=ERI*AJ1
      ECS(2)=ERI*AJ3/2.0
      ECS(3)=ERI*AJ1/2.0
      ECS(4)=ERI*AJ3
      MPVT=MPIV(3)
      AJ3=B2(MPVT)/A3(MPVT)
      MPVT=MPIV(2)
      AJ1=B2(MPVT)-A3(MPVT)*AJ3
      MPVT=MPIV(1)
      AJ0=B2(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
      PMD(NP2)=PMD(NP2)+RHO*FMO*AJ0/DELT
      MPVT=MPIV(3)
      ECS(8)=ERI*AJ3
      ECS(7)=ERI*AJ1/2.0
      ECS(6)=ERI*AJ3/2.0
      ECS(5)=ERI*AJ1
      AJ3=B3(MPVT)/A3(MPVT)
      MPVT=MPIV(2)
      AJ1=B3(MPVT)-A3(MPVT)*AJ3
      MPVT=MPIV(1)
      AJ0=B3(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
      PMD(NP3)=PMD(NP3)+RHO*FMO*AJ0/DELT
      ECS(9)=ERI*AJ1
      ECS(10)=ERI*AJ3/2.0
      ECS(11)=ERI*AJ1/2.0
      ECS(12)=ERI*AJ3
      IF(NP4.EQ.0) GO TO 126
      MPVT=MPIV(3)
      AJ3=B4(MPVT)/A3(MPVT)
      MPVT=MPIV(2)
      AJ1=B4(MPVT)-A3(MPVT)*AJ3
      MPVT=MPIV(1)
      AJ0=B4(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
      PMD(NP4)=PMD(NP4)+RHO*FMO*AJ0/DELT
      ECS(13)=ERI*AJ1
      ECS(14)=ERI*AJ3/2.0
      ECS(15)=ERI*AJ1/2.0
      ECS(16)=ERI*AJ3
      FCS(1)=TAU*(X3(NP4)-X3(NP2))/2.0
      FCS(2)=TAU*(X1(NP2)-X1(NP4))/2.0
      FCS(3)=TAU*(X3(NP1)-X3(NP3))/2.0
      FCS(4)=TAU*(X1(NP3)-X1(NP1))/2.0
      FCS(5)=TAU*(X3(NP2)-X3(NP4))/2.0
      FCS(6)=TAU*(X1(NP4)-X1(NP2))/2.0
```

```
SETUP
   1266          FCS(7)=TAU*(X3(NP3)-X3(NP1))/2.0
   1273          FCS(8)=TAU*(X1(NP1)-X1(NP3))/2.0
   1276          GO TO 127
   1306      126 FCS(1)=TAU*(X3(NP3)-X3(NP2))/2.0
   1311          FCS(2)=TAU*(X1(NP2)-X1(NP3))/2.0
   1314          FCS(3)=TAU*(X3(NP1)-X3(NP3))/2.0
   1321          FCS(4)=TAU*(X1(NP3)-X1(NP1))/2.0
   1323          FCS(5)=TAU*(X3(NP2)-X3(NP1))/2.0
   1330          FCS(6)=TAU*(X1(NP1)-X1(NP2))/2.0
   1333      127 SMALLOUT(ECS,ECL(INDE),16)
   1341          SMALLOUT(FCS,FCL(INDF),8)
   1347          GO TO 101
   1347      900 NLT=NL
   1351          READ(5,511) (NPBP(J),J=1,JLIM)
   1357          WRITE(6,512) (NPBP(J),J=1,JLIM)
   1371          A11=0.0
   1372          A13=0.0
   1372          A31=0.0
   1373          A33=0.0
   1373          JLS=JLIM-1
   1376          DO 131 J=1,JLS
   1403          NPY=NPBP(J+1)
   1405          NPX=NPBP(J)
   1407          IPTP(NPX)=1
   1412          DELS=SQRT((X1(NPY)-X1(NPX))**2+(X3(NPY)-X3(NPX))**2)
   1432          BT11=(X1(NPY)-X1(NPX))/DELS
   1436          BT13=(X3(NPY)-X3(NPX))/DELS
   1442          BT31=-BT13
   1443          BT33=BT11
   1444          BTI11=BT11
   1445          BTI13=BT31
   1447          BTI31=BT13
   1450          BTI33=BT33
   1452          FDL=DELS/2.0
   1454          B11=FDL*RHON*(BTI11*VSN*BT11+BTI13*VPN*BT31)
   1461          B13=FDL*RHON*(BTI11*VSN*BT13+BTI13*VPN*BT33)
   1467          B31=FDL*RHON*(BTI31*VSN*BT11+BTI33*VPN*BT31)
   1475          B33=FDL*RHON*(BTI31*VSN*BT13+BTI33*VPN*BT33)
   1503          A13=A13+B13
   1505          A11=A11+B11
   1507          A31=A31+B31
   1511          A33=A33+B33
   1512          DET=(A33+PMD(NPX))*(A11+PMD(NPX))-A13*A31
   1517          CF11(J)=(A33+PMD(NPX))/DET
   1521          CF13(J)=-A13/DET
   1523          CF31(J)=-A31/DET
   1526          CF33(J)=(A11+PMD(NPX))/DET
   1532          A11=B11
   1534          A13=B13
   1535          A31=B31
   1537      131 A33=B33
   1543          NPX=NPBP(JLIM)
   1545          IPTP(NPX)=1
   1546          DET=(A33+PMD(NPX))*(A11+PMD(NPX))-A13*A31
   1556          CF11(JLIM)=(A33+PMD(NPX))/DET
   1561          CF13(JLIM)=-A13/DET
```

```
SETUP
   1562          CF31(JLIM)=-A31/DET
   1564          CF33(JLIM)=(A11+PMD(NPX))/DET
   1571          READ(5,513) LIM,(NSV(L),IOPS(L),NCMP(L),L=1,LIM)
   1622          WRITE(6,514) LIM,(NSV(L),IOPS(L),NCMP(L),L=1,LIM)
   1657          RETURN
   1660      901 WRITE(6,516)
   1664          STOP
   1672          END


SUBPROGRAM LENGTH

02334

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
101    - 000172    120    - 000526    121    - 000571    122    - 000667    123    - 000703    124    - 000776
125    - 001054    126    - 001277    127    - 001334    501    - 001676    502    - 001700    503    - 001702
504    - 001715    505    - 001724    506    - 001741    507    - 001745    511    - 001751    512    - 001753
513    - 001760    514    - 001764    516    - 002004    900    - 001350    901    - 001661

BLOCK NAMES AND LENGTHS
  *LCM*- 0136730
 //    - 030014/01

VARIABLE ASSIGNMENTS
AA     - 002251    AJ0    - 002303    AJ1    - 002302    AJ3    - 002301    ALL    - 0000000/LC A1     - 002170
A11    - 002305    A13    - 002306    A2     - 002173    A3     - 002176    A31    - 002307    A33    - 002310
BB     - 002252    BTI11  - 002321    BTI13  - 002322    BTI31  - 002323    BTI33  - 002324    BT11   - 002315
BT13   - 002316    BT31   - 002317    BT33   - 002320    BX1    - 002242    BX3    - 002243    BY1    - 002244
BY3    - 002245    B1     - 002201    B11    - 002326    B13    - 002327    B2     - 002204    B3     - 002207
B31    - 002330    B33    - 002331    B4     - 002212    CC     - 002253    CF11   - 024714/01 CF13   - 025534/01
CF31   - 026354/01 CF33   - 027174/01 CN     - 000001    DELS   - 002314    DET    - 002332    ECL    - 0064570/LC
ECS    - 002140    ER     - 002235    ERA    - 000000/01 ERI    - 002236    FACTOR - 002221    FCL    - 0120670/LC
FCS    - 002160    FDL    - 002325    FMO    - 002254    FMX    - 002255    FMY    - 002256    GMAX   - 002234
INDE   - 002237    INDF   - 002240    IOPS   - 000005    IPTP   - 014234/01 J      - 002304    JJ     - 002276
JLIM   - 000004    JLS    - 002311    L      - 002333    LIM    - 000010    MORE   - 002222    MPIV   - 002215
MPVP   - 002300    MPVT   - 002277    NCMP   - 000006    NL     - 002223    NLT    - 000003    NP     - 002220
NPBP   - 024074/01 NPT    - 000002    NPX    - 002313    NPY    - 002312    NP1    - 002224    NP2    - 002225
NP3    - 002226    NP4    - 002227    NSV    - 000007    N1     - 003410/01 N2     - 005214/01 N3     - 007020/01
N4     - 010624/01 PIV    - 002275    PMD    - 022124/01 RBM    - 012430/01 RHO    - 002230    SIGY2  - 000000
TAU    - 002233    TAUA   - 001604/01 TEMP   - 002241    VP     - 002231    VS     - 002232    XC     - 002261
XP2    - 002250    XP3    - 002247    XP4    - 002257    X1     - 016204/01 X1C    - 002263    X1P1   - 002265
X1P2   - 002267    X1P3   - 002271    X1P4   - 002273    X3     - 020154/01 X3C    - 002264    X3P1   - 002266
X3P2   - 002270    X3P3   - 002272    X3P4   - 002274    YC     - 002262    YP2    - 002246    YP4    - 002260

START OF CONSTANTS-001675       TEMPS--002023    INDIRECTS-002126

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 051500
```

```
MODEL
                SUBROUTINE MODEL(MAX,SIGY,SIGY2,CN)
                REAL SIGY(10),SIGY2(10),CN(10)
                MAX=10
      7         DO 1 J=1,9
     13         SIGY(J)=0.1*FLOAT(J)
     15       1 SIGY2(J)=SIGY(J)*SIGY(J)
     17         SIGY(MAX)=0.99
     20         SIGY2(MAX)=SIGY(MAX)*SIGY(MAX)
     20         SIGYP=0.9999
     21         MM=MAX-1
     23         EML=SIGY(1)
     24         CFI=1.0
     26         DO 3 J=1,MM
     33         EM=SIGY(J+1)/(1.0-SIGY(J+1))
     35         CNI=(EM-EML)/(SIGY(J+1)-SIGY(J))-CFI
     41         CN(J)=1.0/CNI
     42         CFI=CFI+CNI
     43       3 EML=EM
     51         EM=SIGYP/(1.0-SIGYP)
     53         CNI=(EM-EML)/(SIGYP-SIGY(MAX))-CFI
     57         CN(MAX)=1.0/CNI
     60         SQ2=SQRT(2.0)
     63         DO 4 J=1,MAX
     73         SIGY(J)=SQ2*SIGY(J)
     74       4 SIGY2(J)=2.0*SIGY2(J)
     76         RETURN
     76         END


SUBPROGRAM LENGTH

00144

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
CFI    - 000140     CNI    - 000142     EM     - 000141     EML    - 000137     J      - 000134     MM     - 000136
SIGYP  - 000135     SQ2    - 000143

START OF CONSTANTS-000101       TEMPS--000107    INDIRECTS-000127

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 044000
```

```
FILTER
                SUBROUTINE FILTER(XX,F1,F2,N,DTF,NKF)
                REAL XX(1)
                REAL YY(8000)
                REAL P(1000)
                PI=3.14159
     11         P0=(F1+F2)*DTF
     13         W1=2.0*PI*F1
     15         W2=2.0*PI*F2
     16         C=DTF/(2.0*PI)
     17         Q=1.0/(2.0*(F2-F1))
     22         DO 1 I=1,N
     24         T=FLOAT(I)*DTF
     25         TMQ=T-Q
     27         ADD=PI*(F2-F1)
     31         IF(TMQ.NE.0) ADD=SIN(PI*TMQ*(F2-F1))/TMQ
     43         ADD=C*SIN((W1+W2)*T/2.0)*ADD
     56       1 P(I)=C*(SIN(W1*T)+SIN(W2*T))*(1.0/T-0.5/(T+Q))+ADD
    106         DO 2 J=1,NKF
    107         CUM=P0*XX(J)
    111         DO 3 I=1,N
    113         IF((I+J).GT.NKF) GO TO 4
    116         CUM=CUM+P(I)*XX(J+I)
    122       3 CONTINUE
    124       4 DO 5 I=1,N
    126         IF((J-I).LT.1) GO TO 2
    131         CUM=CUM+P(I)*XX(J-I)
    135       5 CONTINUE
    137       2 YY(J)=CUM
    144         DO 6 J=1,NKF
    150       6 XX(J)=YY(J)
    152         RETURN
    152         END


SUBPROGRAM LENGTH

21703

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
2       - 000140     4       - 000125

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
ADD     - 021700     C       - 021673     CUM     - 021702     I       - 021675     J       - 021701     P       - 017717
PI      - 021667     P0      - 021670     Q       - 021674     T       - 021676     TMQ     - 021677     W1      - 021671
W2      - 021672     YY      - 000217

START OF CONSTANTS-000155       TEMPS--000162    INDIRECTS-000216

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 044000
```

```
NLPLT
      SUBROUTINE NLPLT(VF1,VF3,VSAVE,Y,NK,DELT,ALPHA)
      COMMON/CCPOOL/XMIN,XMAX,YMIN,YMAX,CCXMIN,CCXMAX,CCYMIN,CCYMAX
      LARGE ALL(27000),ECL(14400),FCL(7200)
      LARGE VF1(8000),VF3(8000)
      LARGE VSAVE(8000,16)
      REAL X(8002),Y(8002)
      INTEGER ALPHA(8),XALPH(3),YALPH(3)
      REAL YBASE(24)
      INTEGER LY(24)
  501 FORMAT(///1X,*SUBROUTINE NLPLT*//)
  502 FORMAT(I1,I2,F3.0,F4.0,2I2,3A10,3A10)
  503 FORMAT(//5X,*MORE=*,I2/5X,*NCH=*,I3/5X,*XMAX=*,
     *F5.0/5X,*YMAX=*,F6.0/5X,*NX=*,I3/
     * 5X,*NY=*,I3/5X,*XALPH=*,1X,3A10/5X,*YALPH=*,1X,3A10/)
  504 FORMAT(12(I2,F4.0))
  505 FORMAT(5X,*LY(J)*,2X,*YBASE(J)*/(5X,I5,F10.0))
  506 FORMAT(F2.0)
  507 FORMAT(F4.0)
      WRITE(6,501)
   14 CCXMIN=100.
   15 CCXMAX=1200.
   17 CCYMIN=130.
   20 CCYMAX=1030.
   22 XMIN=0.0
   22 YMIN=0.0
   27 1 READ(5,502) MORE,NCH,XMAX,YMAX,NX,NY,
     * (XALPH(J),J=1,3),(YALPH(J),J=1,3)
   53 WRITE(6,503) MORE,NCH,XMAX,YMAX,NX,NY,
     * (XALPH(J),J=1,3),(YALPH(J),J=1,3)
  102 CALL CCGRID(1,NX,6HNOLBLS,1,NY)
  106 WRITE(98,506) XMIN
  117 CALL CCLTR(95.,100.,0,2)
  122 WRITE(98,507) XMAX
  133 CALL CCLTR(1180.,100.,0,2)
  136 CALL CCLTR(200.,1050.,0,2,ALPHA,80)
  143 CALL CCLTR(400.,20.,0,2,XALPH,30)
  147 CALL CCLTR(30.,430.,1,2,YALPH,30)
  153 READ(5,504) (LY(J),YBASE(J),J=1,NCH)
  200 WRITE(6,505) (LY(J),YBASE(J),J=1,NCH)
  225 DO 2 K=1,NK
  235 2 X(K)=FLOAT(K-1)*DELT
  240 DO 3 J=1,NCH
  245 LYJ=LY(J)
  247 YBASJ=YBASE(J)
  251 IF(LYJ.EQ.91) GO TO 4
  254 IF(LYJ.EQ.93) GO TO 5
  255 SMALLIN(Y,VSAVE(1,LYJ),NK)
  265 DO 6 K=1,NK
  271 6 Y(K)=Y(K)+YBASJ
  273 GO TO 3
  277 4 SMALLIN(Y,VF1,NK)
  306 DO 7 K=1,NK
  312 7 Y(K)=Y(K)+YBASJ
  314 GO TO 3
  320 5 SMALLIN(Y,VF3,NK)
  327 DO 8 K=1,NK
```

```
NLPLT
     333        8 Y(K)=Y(K)+YBASJ
     341        3 CALL CCPLOT(X,Y,NK)
     351          CALL CCNEXT
     352          IF(MORE.EQ.1) GO TO 1
     360          CALL CCEND
     361          RETURN
     362          END


SUBPROGRAM LENGTH

20301

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       - 000030     3       - 000342     4       - 000300     5       - 000321     501     - 000366     502     - 000372
503     - 000377     504     - 000423     505     - 000426     506     - 000434     507     - 000436

BLOCK NAMES AND LENGTHS
  'LCM'- 0136730
CCPOOL - 000010/01

VARIABLE ASSIGNMENTS
ALL     - 0000000/LC ALPHA   - 000000     CCXMAX - 000005/01 CCXMIN - 000004/01 CCYMAX - 000007/01 CCYMIN - 000006/01
ECL     - 0064570/LC FCL     - 0120670/LC J       - 020275     K       - 020276     LY      - 020241     LYJ     - 020277
MORE    - 020271     NCH     - 020272     NX      - 020273     NY      - 020274     X       - 000501     XALPH   - 020203
XMAX    - 000001/01  XMIN    - 000000/01  YALPH   - 020206     YBASE   - 020211     YBASJ   - 020300     YMAX    - 000003/01
YMIN    - 000002/01

START OF CONSTANTS-000365       TEMPS--000471    INDIRECTS-000477

7600 COMPILATION -- RUN76 LEVEL 20      23 NOV 77

ROUTINE COMPILES IN 045000
```

XTRM

```
                SUBROUTINE XTRM(Y,NK)
                REAL Y(8000)
              501 FORMAT(5X,*MAX=*,E12.4,1X,*AT K=*,I6/
                * 5X,*MIN=*,E12.4,1X,*AT K=*,I6//)
                KMAX=0
          4     KMIN=0
          5     YMAX=0.0
          6     YMIN=0.0
          7     DO 1 K=1,NK
         10     IF(Y(K).GT.YMAX) KMAX=K
         14     IF(Y(K).GT.YMAX) YMAX=Y(K)
         21     IF(Y(K).LT.YMIN) KMIN=K
         25     IF(Y(K).LT.YMIN) YMIN=Y(K)
         31   1 CONTINUE
         34     WRITE(6,501) YMAX,KMAX,YMIN,KMIN
         47     RETURN
         50     END
```

SUBPROGRAM LENGTH

00105

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1 - 000032 501 - 000054

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
K - 000104 KMAX - 000100 KMIN - 000101 YMAX - 000102 YMIN - 000103

START OF CONSTANTS-000053 TEMPS--000070 INDIRECTS-000072

7600 COMPILATION -- RUN76 LEVEL 20 23 NOV 77

ROUTINE COMPILES IN 044000

PLANE STRAIN NONLINEAR GROUND RESPONSE

SAMPLE PROBLEM PLANE STRAIN

DELT= .0040000
NK= 750
RHON= 2.600
VPN= 3.7000E+05
VSN= 2.0000E+05
NPT= 24
JLIM= 9
IH= 1
IV=-0
ITAPE=-0
F1= 3.75
F2= 7.50

POINT LIST

FACTOR= 1.00000E+03

NP	X1	X3	NP	X1	X3	NP	X1	X3	NP	X1	X3	NP	X1	X3
1	0.00	0.00	2	1.00	1.00	3	1.00	0.00	4	2.00	2.00	5	2.00	1.00
6	2.00	0.00	7	3.00	3.00	8	3.00	2.00	9	3.00	1.00	10	3.00	0.00
11	4.00	3.00	12	4.00	2.00	13	4.00	1.00	14	4.00	0.00	15	5.00	3.00
16	5.00	2.00	17	5.00	1.00	18	5.00	0.00	19	6.00	2.00	20	6.00	1.00
21	6.00	0.00	22	7.00	1.00	23	7.00	0.00	24	8.00	0.00			

ELEMENT LIST

NL	N1	N2	N3	N4	RHO	VP	VS	TAU
1	1	3	2	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
2	2	5	4	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
3	2	3	6	5	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
4	4	8	7	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
5	4	5	9	8	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
6	5	6	10	9	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
7	7	8	12	11	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
8	8	9	13	12	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
9	9	10	14	13	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
10	11	12	16	15	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
11	12	13	17	16	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
12	13	14	18	17	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
13	15	16	19	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
14	17	18	21	20	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
15	16	17	20	19	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
16	19	20	22	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
17	20	21	23	22	2.0000	1.6900E+05	3.0000E+04	1.8500E+06
18	22	23	24	0	2.0000	1.6900E+05	3.0000E+04	1.8500E+06

BOUNDARY POINTS

1 3 6 10 14 18 21 23 24

SAVE POINTS

LIM= 6

1 3 1 1 3 3 4 3 1 4 3 3 11 3 1 11 3 3

EXTREME VALUES

INPUT HORIZONTAL

MAX= 4.9000E+01 AT K= 50
MIN= -6.3047E-12 AT K= 100

L= 1 NSV= 1 IOPS= 3 NCMP= 1

MAX= 4.7717E+01 AT K= 51
MIN= -1.2938E+00 AT K= 175

L= 2 NSV= 1 IOPS= 3 NCMP= 3

MAX= 1.4762E+00 AT K= 112
MIN= -1.2925E+00 AT K= 50

L= 3 NSV= 4 IOPS= 3 NCMP= 1

MAX= 6.8757E+01 AT K= 78
MIN= -2.9132E+01 AT K= 147

L= 4 NSV= 4 IOPS= 3 NCMP= 3

MAX= 9.6804E-01 AT K= 103
MIN= -1.5576E+00 AT K= 65

L= 5 NSV= 11 IOPS= 3 NCMP= 1

MAX= 7.3423E+01 AT K= 79
MIN= -3.2068E+01 AT K= 147

L= 6 NSV= 11 IOPS= 3 NCMP= 3

MAX= 2.9191E-13 AT K= 83
MIN= -3.8088E-13 AT K= 49

SUBROUTINE NLPLT

MORE= 1
NCH= 4
XMAX= 5
YMAX= 400

```
NX=  5
NY=  8
XALPH= SECONDS
YALPH= VELOCITY H (CM/SEC)

LY(J)   YBASE(J)
   91        100
    1        200
    3        250
    5        300


MORE=-0
NCH=  4
XMAX=     5
YMAX=   400
NX=  5
NY=  8
XALPH= SECONDS
YALPH= VELOCITY V (CM/SEC)

LY(J)   YBASE(J)
   93        100
    2        200
    4        250
    6        300
```

PLANE STRAIN PROGRAM

LISTING OF INPUT DECK

FOR SAMPLE PROBLEM

INPUT CARDS FOR SAMPLE PROBLEM - PLANE STRAIN PROGRAM

```
        10        20        30        40        50        60        70        80
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
SAMPLE PROBLEM  PLANE STRAIN
 0.004    750 2.6         3.7E+05    2.0E+05  24  91    3.75 7.5
  10.0E+02
        0         0         1         1         1         0         2         2         2         1
        2         0         3         3         3         2         3         1         3         0
        4         3         4         2         4         1         4         0         5         3
        5         2         5         1         5         0         6         2         6         1
        6         0         7         1         7         0         8         0
1   1   3   2   0 2.0    1.69E+05    0.3E+05   1.85E+06
1   2   5   4   0 2.0    1.69E+05    0.3E+05   1.85E+06
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
1   2   3   6   5 2.0    1.69E+05    0.3E+05   1.85E+06
1   4   8   7   0 2.0    1.69E+05    0.3E+05   1.85E+06
1   4   5   9   8 2.0    1.69E+05    0.3E+05   1.85E+06
1   5   6  10   9 2.0    1.69E+05    0.3E+05   1.85E+06
1   7   8  12  11 2.0    1.69E+05    0.3E+05   1.85E+06
1   8   9  13  12 2.0    1.69E+05    0.3E+05   1.85E+06
1   9  10  14  13 2.0    1.69E+05    0.3E+05   1.85E+06
1  11  12  16  15 2.0    1.69E+05    0.3E+05   1.85E+06
1  12  13  17  16 2.0    1.69E+05    0.3E+05   1.85E+06
1  13  14  18  17 2.0    1.69E+05    0.3E+05   1.85E+06
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
1  15  16  19   0 2.0    1.69E+05    0.3E+05   1.85E+06
1  16  17  20  19 2.0    1.69E+05    0.3E+05   1.85E+06
1  17  18  21  20 2.0    1.69E+05    0.3E+05   1.85E+06
1  19  20  22   0 2.0    1.69E+05    0.3E+05   1.85E+06
1  20  21  23  22 2.0    1.69E+05    0.3E+05   1.85E+06
   22  23  24   0 2.0    1.69E+05    0.3E+05   1.85E+06
   1   3   6  10  14  18  21  23  24
     6   131   133   431   433  1131  1133
 .0       E+00 .02       E+00 .04       E+00 .06       E+00 .08       E+00
 .10      E+00 .12       E+00 .14       E+00 .16       E+00 .18       E+00
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
 .20      E+00 .22       E+00 .24       E+00 .26       E+00 .28       E+00
 .30      E+00 .32       E+00 .34       E+00 .36       E+00 .38       E+00
 .40      E+00 .42       E+00 .44       E+00 .46       E+00 .48       E+00
 .50      E+00 .48       E+00 .46       E+00 .44       E+00 .42       E+00
 .40      E+00 .38       E+00 .36       E+00 .34       E+00 .32       E+00
 .30      E+00 .28       E+00 .26       E+00 .24       E+00 .22       E+00
 .20      E+00 .18       E+00 .16       E+00 .14       E+00 .12       E+00
 .10      E+00 .08       E+00 .06       E+00 .04       E+00 .02       E+00
 .0       E+00-.02       E+00-.04       E+00-.06       E+00-.08       E+00
-.10      E+00-.12       E+00-.14       E+00-.16       E+00-.18       E+00
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
```

```
        10        20        30        40        50        60        70        80
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
-.20      E+00-.22      E+00-.24      E+00-.26      E+00-.28      E+00
-.30      E+00-.32      E+00-.34      E+00-.36      E+00-.38      E+00
-.40      E+00-.42      E+00-.44      E+00-.46      E+00-.48      E+00
-.50      E+00-.48      E+00-.46      E+00-.44      E+00-.42      E+00
-.40      E+00-.38      E+00-.36      E+00-.34      E+00-.32      E+00
-.30      E+00-.28      E+00-.26      E+00-.24      E+00-.22      E+00
-.20      E+00-.18      E+00-.16      E+00-.14      E+00-.12      E+00
-.10      E+00-.08      E+00-.06      E+00-.04      E+00-.02      E+00
            130 ADDITIONAL CARDS IDENTICAL TO THE PRECEDING
1 4  5 400 5 8SECONDS                          VELOCITY H (CM/SEC)
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
91 100 1 200 3 250 5 300
  4  5 400 5 8SECONDS                          VELOCITY V (CM/SEC)
93 100 2 200 4 250 6 300
```

ANTIPLANE STRAIN PROGRAM

SOURCE LISTING

AND

OUTPUT LISTING FOR SAMPLE PROBLEM

```
PGM
      PROGRAM PGM(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     * TAPE16,TAPE17,TAPE98=101,PLOT,TAPE99=PLOT)
      LARGE ALL(20000),ECL(8000),FCL(8000)
      LARGE VF2(8000)
      LARGE VSAVE(8000,16)
      REAL SMALL(8002)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      INTEGER ALPHA(8)
      REAL SIGY(10),SIGY2(10),CN(10)
  501 FORMAT(8A10)
  502 FORMAT(1H1,*ANTIPLANE STRAIN NONLINEAR GROUND RESPONSE*//
     * 1X,8A10//)
  503 FORMAT(F8.0,I4,F8.0,E10.0,F5.0,I4,I3,I1,2F5.0)
  504 FORMAT(5X,*DELT=*,F10.7/5X,*NK=*,I6/5X,*RHON=*,F6.3/
     * 5X,*VSN=*,E12.4/5X,*THETA=*,F6.1/5X,*NPT=*,I6/5X,
     * *JLIM=*,I5/5X,*ITAPE=*,I2/5X,*F1=*,F6.2/5X,*F2=*,F6.2)
  505 FORMAT(5E14.8)
  506 FORMAT(///1X,*EXTREME VALUES*/)
  507 FORMAT(3X,*INPUT*/)
  509 FORMAT(3X,*L=*,I3,3X,*NSV=*,I5,
     * 3X,*IOPS=*,I2,3X,*NCMP=*,I2/)
      READ(5,501) (ALPHA(J),J=1,8)
   10 WRITE(6,502) (ALPHA(J),J=1,8)
   16 READ(5,503) DELT,NK,RHON,VSN,THETA,NPT,JLIM,ITAPE,F1,F2
   46 WRITE(6,504) DELT,NK,RHON,VSN,THETA,NPT,JLIM,ITAPE,F1,F2
   76 THETA=THETA*3.14159/180.0
  100 CALL SETUP(DELT,RHON,VSN,THETA,MAX,SIGY,SIGY2,CN,
     * NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
  116 LSV=LIM
  117 IF(LSV.GT.16) LSV=16
  123 IF(ITAPE.EQ.1) GO TO 13
  125 READ(5,505) (SMALL(K),K=1,NK)
  134 G=980.0
  135 VOK=0.0
  136 DO 12 K=1,NK
  145 VOK=VOK+G*DELT*SMALL(K)
  147  12 SMALL(K)=VOK
  151 SMALLOUT(SMALL,VF2,NK)
  157  13 WRITE(16) (ALPHA(J),J=1,8)
  164 CALL NLAPS(DELT,NK,VF2,VSAVE,ITAPE,MAX,SIGY,SIGY2,CN,
     * NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
  207 WRITE(6,506)
  213 IF(ITAPE.EQ.1) GO TO 22
  215 WRITE(6,507)
  221 SMALLIN(SMALL,VF2,NK)
  227 CALL XTRM(SMALL,NK)
  231  22 NKM=NK
  232 IF(F1.EQ.0.0.OR.F2.EQ.0.0) GO TO 24
  241 NKM=NKM-20
  242  24 DO 23 L=1,LSV
  244 WRITE(6,509) L,NSV(L),IOPS(L),NCMP(L)
  262 SMALLIN(SMALL,VSAVE(1,L),NK)
  273 IF(F1.EQ.0.0.OR.F2.EQ.0.0) GO TO 23
  301 CALL FILTER(SMALL,F1,F2,20,DELT,NK)
  305 SMALLOUT(SMALL,VSAVE(1,L),NK)
  316  23 CALL XTRM(SMALL,NKM)
```

```
PGM
    323         CALL NLPLT(VF2,VSAVE,SMALL,NKM,DELT,ALPHA)
    332         STOP
    334         END


PROGRAM LENGTH INCLUDING I/O BUFFERS

25422

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
13      - 000160      22      - 000232      23      - 000317      24      - 000243      501     - 000354      502     - 000356
503     - 000367      504     - 000375      505     - 000421      506     - 000423      507     - 000427      509     - 000432

BLOCK NAMES AND LENGTHS
  *LCM*- 0517740

VARIABLE ASSIGNMENTS
ALL     - 0000000/LC ALPHA   - 022447      CN      - 022503      DELT    - 022516      ECL     - 0047040/LC FCL     - 0066540/LC
F1      - 022526     F2      - 022527      G       - 022535      IOPS    - 020167      ITAPE   - 022525      J       - 022515
JLIM    - 022524     K       - 022534      L       - 022540      LIM     - 022532      LSV     - 022533      MAX     - 022530
NCMP    - 021007     NK      - 022517      NKM     - 022537      NLT     - 022531      NPT     - 022523      NSV     - 021627
RHON    - 022520     SIGY    - 022457      SIGY2   - 022471      SMALL   - 000465      THETA   - 022522      VF2     - 0106240/LC
VOK     - 022536     VSAVE   - 0125740/LC  VSN     - 022521

START OF CONSTANTS-000337       TEMPS--000456    INDIRECTS-000462

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 044700
```

```
NLAPS
      SUBROUTINE NLAPS(DELT,NK,VF2,VSAVE,ITAPE,MAX,SIGY,SIGY2,CN,
     * NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
C       VERSION OF 9/26/75
      LARGE ALL(20000),ECL(8000),FCL(8000)
      LARGE VF2(8000)
      LARGE VSAVE(8000,16)
      REAL ALS(2),ECS(8),FCS(8)
      INTEGER ALTP(8)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      REAL VTAPE(400)
      REAL SD12(1000),SD23(1000)
      REAL ED12(1000),ED23(1000)
      REAL F2(1000)
      REAL V2(1000)
      REAL VTP2(400)
      REAL SIGY(10),SIGY2(10),CN(10)
      COMMON ERA(1000),TAUA(1000)
      COMMON N1(1000),N2(1000),N3(1000),N4(1000)
      COMMON IPTP(1000)
      COMMON X1(1000),X3(1000)
      COMMON PMD(1000)
      COMMON NPBP(400)
      COMMON CF2(400)
      COMMON KS(400)
      COMMON WTS(400)
  501 FORMAT(///1X,*INPUT TAPE*/1X,8A10)
      IF(ITAPE.NE.1) GO TO 10
   23 READ(17) (ALTP(J),J=1,8)
   30 WRITE(6,501) (ALTP(J),J=1,8)
   45 10 LSV=LIM
   46 IF(LSV.GT.16) LSV=16
   52 DO 1 NL=1,NLT
   54 INDL=(NL-1)*20
   56 SD12(NL)=0.0
   57 SD23(NL)=0.0
   61 ED12(NL)=0.0
   62 ED23(NL)=0.0
   64 DO 1 MM=1,MAX
   65 ALS(1)=0.0
   65 ALS(2)=0.0
   66 INDX=INDL+(MM-1)*2+1
   71  1 SMALLOUT(ALS,ALL(INDX),2)
  104 DO 11 NP=1,NPT
  111 F2(NP)=0.0
  112 11 V2(NP)=0.0
  113 DO 21 K=1,NK
  120 DO 22 NL=1,NLT
  121 INDL=(NL-1)*20
  123 INDE=(NL-1)*8+1
  125 INDF=(NL-1)*8+1
  127 N1T=N1(NL)
  131 N2T=N2(NL)
  133 N3T=N3(NL)
  135 N4T=N4(NL)
  140 SMALLIN(ECS,ECL(INDE),8)
  152 DED12=ECS(1)*V2(N1T)+ECS(3)*V2(N2T)+ECS(5)*V2(N3T)
```

```
NLAPS
    157         DED23=ECS(2)*V2(N1T)+ECS(4)*V2(N2T)+ECS(6)*V2(N3T)
    166         IF(N4T.EQ.0) GO TO 23
    171         DED12=DED12+ECS(7)*V2(N4T)
    173         DED23=DED23+ECS(8)*V2(N4T)
    175      23 A11=1.0
    176         A12=0.0
    177         A21=0.0
    177         A22=1.0
    201         DO 32 MM=1,MAX
    202         INDX=INDL+(MM-1)*2+1
    205         SMALLIN(ALS,ALL(INDX),2)
    212         CA12=SD12(NL)-ALS(1)
    214         CA23=SD23(NL)-ALS(2)
    217         TEMP=CA12*CA12+CA23*CA23
    221         CALL=TEMP+TEMP
    222         IF(CALL.LE.SIGY2(MM)) GO TO 32
    227         FACT=CN(MM)*CALL
    231         TEMP=(CA12+CA12)/FACT
    233         A11=A11+CA12*TEMP
    235         A21=A21+CA23*TEMP
    237         TEMP=(CA23+CA23)/FACT
    240         A12=A12+CA12*TEMP
    243         A22=A22+CA23*TEMP
    245         FACT=SIGY(MM)/SQRT(CALL)
    252         ALS(1)=SD12(NL)-FACT*CA12
    256         ALS(2)=SD23(NL)-FACT*CA23
    262         SMALLOUT(ALS,ALL(INDX),2)
    273      32 CONTINUE
    300         DET=A11*A22-A12*A21
    303         DS12=(A22*DED12-A12*DED23)/DET
    307         DS23=(A11*DED23-A21*DED12)/DET
    312         SD12(NL)=SD12(NL)+DS12
    314         SD23(NL)=SD23(NL)+DS23
    316         ED12(NL)=ED12(NL)+DED12
    321         ED23(NL)=ED23(NL)+DED23
    324         ST12=SD12(NL)
    325         ST23=SD23(NL)
    327         SMALLIN(FCS,FCL(INDF),8)
    335         F2(N1T)=F2(N1T)+FCS(1)*ST12+FCS(2)*ST23
    342         F2(N2T)=F2(N2T)+FCS(3)*ST12+FCS(4)*ST23
    347         F2(N3T)=F2(N3T)+FCS(5)*ST12+FCS(6)*ST23
    354         IF(N4T.EQ.0) GO TO 22
    355         F2(N4T)=F2(N4T)+FCS(7)*ST12+FCS(8)*ST23
    362      22 CONTINUE
    365         DO 51 NP=1,NPT
    366         IF(IPTP(NP).NE.0) GO TO 51
    371         TKNP=1.0/PMD(NP)
    373         V2(NP)=V2(NP)+TKNP*F2(NP)
    377         F2(NP)=0.0
    377      51 CONTINUE
    402         IF(ITAPE.EQ.1) READ(17) (VTP2(J),J=1,JLIM)
    415         DO 52 J=1,JLIM
    417         NPX=NPBP(J)
    421         IF(ITAPE.EQ.1) GO TO 151
    423         KS1=K-KS(J)
    425         KS2=KS1-1
```

```
NLAPS
     427          WTSJ=WTS(J)
     431          IF(KS1.GE.1) GO TO 152
     434          VF2X=0.0
     435          GO TO 153
     435      152 VSK1=VF2(KS1)
     440          IF(KS2.GE.1) GO TO 154
     443          VSK2=0.0
     444          GO TO 155
     444      154 VSK2=VF2(KS2)
     447      155 VF2X=WTSJ*VSK1+(1.0-WTSJ)*VSK2
     454          GO TO 153
     454      151 VF2X=VTP2(J)
     457      153 V2(NPX)=VF2X+CF2(J)*(F2(NPX)+PMD(NPX)*
                 * (V2(NPX)-VF2X))
     467       52 F2(NPX)=0.0
     474          DO 61 L=1,LIM
     475          NSVL=NSV(L)
     477          IF(IOPS(L).EQ.1) GO TO 63
     502          IF(IOPS(L).EQ.2) GO TO 64
     504          VTAPE(L)=V2(NSVL)
     507          GO TO 61
     510       63 IF(NCMP(L).EQ.1) VTAPE(L)=ED12(NSVL)*ERA(NSVL)
     517          IF(NCMP(L).EQ.2) VTAPE(L)=ED23(NSVL)*ERA(NSVL)
     526          GO TO 61
     527       64 IF(NCMP(L).EQ.1) VTAPE(L)=SD12(NSVL)*TAUA(NSVL)
     536          IF(NCMP(L).EQ.2) VTAPE(L)=SD23(NSVL)*TAUA(NSVL)
     545          IF(NCMP(L).EQ.2) VTAPE(L)=SD23(NSVL)*TAUA(NSVL)
     554       61 CONTINUE
     557          DO 62 L=1,LSV
     566       62 VSAVE(K,L)=VTAPE(L)
     571          WRITE(16) (VTAPE(L),L=1,LIM)
     606       21 CONTINUE
     610          RETURN
     610          END


SUBPROGRAM LENGTH

16164

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
10     - 000046    22     - 000363    23     - 000176    32     - 000274    51     - 000400    61     - 000555
63     - 000511    64     - 000530    151    - 000455    152    - 000436    153    - 000460    154    - 000445
155    - 000450    501    - 000614

BLOCK NAMES AND LENGTHS
  *LCM*- 0106240
 //    - 026520/01

VARIABLE ASSIGNMENTS
ALL    - 0000000/LC ALS    - 000642    ALTP   - 000664    A11    - 016134    A12    - 016135    A21    - 016136
A22    - 016137    CALL   - 016143    CA12   - 016140    CA23   - 016141    CF2    - 024240/01 CN     - 000002
DED12  - 016132    DED23  - 016133    DET    - 016145    DS12   - 016146    DS23   - 016147    ECL    - 0047040/LC
```

```
NLAPS
ECS    - 000644    ED12   - 005434    ED23   - 007404    ERA    - 000000/01  FACT   - 016144    FCL    - 0066540/LC
FCS    - 000654    F2     - 011354    INDE   - 016124    INDF   - 016125    INDL   - 016117    INDX   - 016121
IOPS   - 000006    IPTP   - 013560/01 J      - 016114    JLIM   - 000005    K      - 016123    KS     - 025060/01
KS1    - 016154    KS2    - 016155    L      - 016162    L1M    - 000011    LSV    - 016115    MM     - 016120
NCMP   - 000007    NL     - 016116    NLT    - 000004    NP     - 016122    NPBP   - 023420/01 NPT    - 000003
NPX    - 016153    NSV    - 000010    NSVL   - 016163    N1     - 003720/01  N1T    - 016126    N2     - 005670/01
N2T    - 016127    N3     - 007640/01 N3T    - 016130    N4     - 011610/01  N4T    - 016131    PMD    - 021450/01
SD12   - 001514    SD23   - 003464    SIGY   - 000000    SIGY2  - 000001    ST12   - 016150    ST23   - 016151
TAUA   - 001750/01 TEMP   - 016142    TKNP   - 016152    VF2X   - 016157    VSK1   - 016160    VSK2   - 016161
VTAPE  - 000674    VTP2   - 015274    V2     - 013324    WTS    - 025700/01  WTSJ   - 016156    X1     - 015530/01
X3     - 017500/01

START OF CONSTANTS-000613       TEMPS--000624    INDIRECTS-000636

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 046100
```

SETUP

```
      SUBROUTINE SETUP(DELT,RHON,VSN,THETA,MAX,SIGY,SIGY2,
     * CN,NPT,NLT,JLIM,IOPS,NCMP,NSV,LIM)
      LARGE ALL(20000),ECL(8000),FCL(8000)
      REAL ECS(8),FCS(8)
      INTEGER IOPS(400),NCMP(400),NSV(400)
      REAL SIGY(10),SIGY2(10),CN(10)
      REAL A1(3),A2(3),A3(3),B1(3),B2(3),B3(3),B4(3)
      INTEGER MPIV(3)
      REAL RAY(400)
      COMMON ERA(1000),TAUA(1000)
      COMMON N1(1000),N2(1000),N3(1000),N4(1000)
      COMMON IPTP(1000)
      COMMON X1(1000),X3(1000)
      COMMON PMD(1000)
      COMMON NPBP(400)
      COMMON CF2(400)
      COMMON KS(400)
      COMMON WTS(400)
  501 FORMAT(E10.0)
  502 FORMAT(10F8.0)
  503 FORMAT(///1X,*POINT LIST*//3X,*FACTOR=*,E14.5//
     * 1X,5(4X,*NP*,7X,*X1*,7X,*X3*)/)
  504 FORMAT(1X,I6,2F9.2,I6,2F9.2,I6,2F9.2,I6,2F9.2,I6,2F9.2)
  505 FORMAT(///1X,*ELEMENT LIST*//1X,4X,*NL*,6X,*N1*,6X,*N2*,
     * 6X,*N3*,6X,*N4*,7X,*RHO*,12X,*VS*,11X,*TAU*/)
  506 FORMAT(I1,4I4,F5.0,2E10.0)
  507 FORMAT(1X,I6,4I8,F10.4,2E14.4)
  511 FORMAT(20I4)
  512 FORMAT(///1X,*BOUNDARY POINTS*//(3X,10I8))
  513 FORMAT(I6,11(I4,2I1)/(12(I4,2I1)))
  514 FORMAT(///1X,*SAVE POINTS*//4X,*LIM=*,I4//(1X,I7,2I2,
     * I7,2I2,I7,2I2,I7,2I2,I7,2I2,I7,2I2,I7,2I2,
     * I7,2I2,I7,2I2))
  516 FORMAT(//1X,*POINT NUMBER EXCEEDS NPT - JOB TERMINATED*)
      CALL MODEL(MAX,SIGY,SIGY2,CN)
   24 DO 10 NP=1,NPT
   35 IPTP(NP)=0
   36 10 PMD(NP)=0.0
   37 READ(5,501) FACTOR
   50 READ(5,502) (X1(NP),X3(NP),NP=1,NPT)
   76 WRITE(6,503) FACTOR
  107 WRITE(6,504) (NP,X1(NP),X3(NP),NP=1,NPT)
  137 DO 11 NP=1,NPT
  151 X1(NP)=FACTOR*X1(NP)
  152 11 X3(NP)=FACTOR*X3(NP)
  154 MORE=1
  154 NL=0
  156 WRITE(6,505)
  171 101 IF(MORE.NE.1) GO TO 900
  173 READ(5,506) MORE,NP1,NP2,NP3,NP4,RHO,VS,TAU
  217 NL=NL+1
  221 WRITE(6,507) NL,NP1,NP2,NP3,NP4,RHO,VS,TAU
  250 IF(NP1.GT.NPT.OR.NP2.GT.NPT.OR.NP3.GT.NPT.OR.NP4.GT.NPT)
     * GO TO 901
  276 N1(NL)=NP1
  276 N2(NL)=NP2
```

```
SETUP
    300         N3(NL)=NP3
    303         N4(NL)=NP4
    305         GMAX=RHO*VS*VS
    310         ER=TAU/(2.0*GMAX)
    312         ERA(NL)=ER
    313         TAUA(NL)=TAU
    315         ERI=DELT/ER
    317         INDE=(NL-1)*8+1
    322         INDF=(NL-1)*8+1
    323         TEMP=SQRT((X1(NP3)-X1(NP1))**2+(X3(NP3)-X3(NP1))**2)
    345         BX1=(X1(NP3)-X1(NP1))/TEMP
    350         BX3=(X3(NP3)-X3(NP1))/TEMP
    352         BY1=(X3(NP3)-X3(NP1))/TEMP
    354         BY3=-(X1(NP3)-X1(NP1))/TEMP
    357         YP2=BY1*(X1(NP2)-X1(NP1))+BY3*(X3(NP2)-X3(NP1))
    365         XP3=BX1*(X1(NP3)-X1(NP1))+BX3*(X3(NP3)-X3(NP1))
    374         XP2=BX1*(X1(NP2)-X1(NP1))+BX3*(X3(NP2)-X3(NP1))
    403         AA=XP2/YP2
    404         BB=(XP2-XP3)/YP2
    406         CC=XP3
    407         FMO=YP2*YP2*(BB-AA)/2.0+YP2*CC
    414         FMX=YP2*YP2*YP2*(BB*BB-AA*AA)/6.0+YP2*YP2*BB*
               * CC/2.0+YP2*CC*CC/2.0
    425         FMY=YP2*YP2*YP2*(BB-AA)/3.0+YP2*YP2*CC/2.0
    433         IF(NP4.EQ.0) GO TO 120
    445         XP4=BX1*(X1(NP4)-X1(NP1))+BX3*(X3(NP4)-X3(NP1))
    454         YP4=BY1*(X1(NP4)-X1(NP1))+BY3*(X3(NP4)-X3(NP1))
    461         AA=XP4/YP4
    462         BB=(XP4-XP3)/YP4
    465         FMO=FMO-YP4*YP4*(BB-AA)/2.0-YP4*CC
    472         FMX=FMX-YP4*YP4*YP4*(BB*BB-AA*AA)/6.0-YP4*YP4*BB*CC/2.0
               * -YP4*CC*CC/2.0
    504         FMY=FMY-YP4*YP4*YP4*(BB-AA)/3.0-YP4*YP4*CC/2.0
    516    120  XC=FMX/FMO
    517         YC=FMY/FMO
    521         X1C=X1(NP1)+BX1*XC+BY1*YC
    525         X3C=X3(NP1)+BX3*XC+BY3*YC
    531         X1P1=X1(NP1)-X1C
    533         X3P1=X3(NP1)-X3C
    535         X1P2=X1(NP2)-X1C
    537         X3P2=X3(NP2)-X3C
    542         X1P3=X1(NP3)-X1C
    544         X3P3=X3(NP3)-X3C
    547         IF(NP4.EQ.0) GO TO 121
    550         X1P4=X1(NP4)-X1C
    553         X3P4=X3(NP4)-X3C
    556    121  A1(1)=3.0
    557         IF(NP4.NE.0) A1(1)=4.0
    562         A1(2)=X1P1+X1P2+X1P3
    565         IF(NP4.NE.0) A1(2)=A1(2)+X1P4
    570         A1(3)=X3P1+X3P2+X3P3
    573         IF(NP4.NE.0) A1(3)=A1(3)+X3P4
    576         A2(1)=A1(2)
    577         A2(2)=X1P1*X1P1+X1P2*X1P2+X1P3*X1P3
    603         IF(NP4.NE.0) A2(2)=A2(2)+X1P4*X1P4
    607         A2(3)=X1P1*X3P1+X1P2*X3P2+X1P3*X3P3
```

```
SETUP
    614         IF(NP4.NE.0) A2(3)=A2(3)+X1P4*X3P4
    620         A3(1)=A1(3)
    621         A3(2)=A2(3)
    623         A3(3)=X3P1*X3P1+X3P2*X3P2+X3P3*X3P3
    627         IF(NP4.NE.0) A3(3)=A3(3)+X3P4*X3P4
    632         B1(1)=1.0
    633         B1(2)=X1P1
    635         B1(3)=X3P1
    636         B2(1)=1.0
    637         B2(2)=X1P2
    641         B2(3)=X3P2
    642         B3(1)=1.0
    643         B3(2)=X1P3
    645         B3(3)=X3P3
    646         IF(NP4.EQ.0) GO TO 122
    650         B4(1)=1.0
    651         B4(2)=X1P4
    652         B4(3)=X3P4
    654     122 PIV=0.0
    655         DO 123 JJ=1,3
    657         IF(ABS(A1(JJ)).LE.ABS(PIV)) GO TO 123
    664         PIV=A1(JJ)
    666         MPVT=JJ
    670     123 CONTINUE
    672         MPIV(1)=MPVT
    673         A2(MPVT)=A2(MPVT)/PIV
    676         A3(MPVT)=A3(MPVT)/PIV
    677         B1(MPVT)=B1(MPVT)/PIV
    701         B2(MPVT)=B2(MPVT)/PIV
    703         B3(MPVT)=B3(MPVT)/PIV
    705         IF(NP4.NE.0) B4(MPVT)=B4(MPVT)/PIV
    711         MPVP=MPVT
    712         PIV=0.0
    713         DO 124 JJ=1,3
    715         IF(JJ.EQ.MPIV(1)) GO TO 124
    720         A2(JJ)=A2(JJ)-A1(JJ)*A2(MPVP)
    724         A3(JJ)=A3(JJ)-A1(JJ)*A3(MPVP)
    727         B1(JJ)=B1(JJ)-A1(JJ)*B1(MPVP)
    733         B2(JJ)=B2(JJ)-A1(JJ)*B2(MPVP)
    737         B3(JJ)=B3(JJ)-A1(JJ)*B3(MPVP)
    743         IF(NP4.NE.0) B4(JJ)=B4(JJ)-A1(JJ)*B4(MPVP)
    752         IF(ABS(A2(JJ)).LE.ABS(PIV)) GO TO 124
    760         PIV=A2(JJ)
    762         MPVT=JJ
    763     124 CONTINUE
    765         MPIV(2)=MPVT
    766         A3(MPVT)=A3(MPVT)/PIV
    771         B1(MPVT)=B1(MPVT)/PIV
    772         B2(MPVT)=B2(MPVT)/PIV
    774         B3(MPVT)=B3(MPVT)/PIV
    776         IF(NP4.NE.0) B4(MPVT)=B4(MPVT)/PIV
   1002         MPVP=MPVT
   1004         DO 125 JJ=1,3
   1005         IF(JJ.EQ.MPIV(1)) GO TO 125
   1006         IF(JJ.EQ.MPIV(2)) GO TO 125
   1011         A3(JJ)=A3(JJ)-A2(JJ)*A3(MPVP)
```

```
SETUP
   1015         B1(JJ)=B1(JJ)-A2(JJ)*B1(MPVP)
   1021         B2(JJ)=B2(JJ)-A2(JJ)*B2(MPVP)
   1025         B3(JJ)=B3(JJ)-A2(JJ)*B3(MPVP)
   1031         IF(NP4.NE.0) B4(JJ)=B4(JJ)-A2(JJ)*B4(MPVP)
   1037         MPVT=JJ
   1041     125 CONTINUE
   1043         MPIV(3)=MPVT
   1044         AJ3=B1(MPVT)/A3(MPVT)
   1047         MPVT=MPIV(2)
   1050         AJ1=B1(MPVT)-A3(MPVT)*AJ3
   1053         MPVT=MPIV(1)
   1054         AJ0=B1(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
   1062         PMD(NP1)=PMD(NP1)+RHO*FMO*AJ0/DELT
   1067         ECS(1)=ERI*AJ1/2.0
   1071         ECS(2)=ERI*AJ3/2.0
   1073         MPVT=MPIV(3)
   1075         AJ3=B2(MPVT)/A3(MPVT)
   1100         MPVT=MPIV(2)
   1101         AJ1=B2(MPVT)-A3(MPVT)*AJ3
   1105         MPVT=MPIV(1)
   1106         AJ0=B2(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
   1114         PMD(NP2)=PMD(NP2)+RHO*FMO*AJ0/DELT
   1121         ECS(3)=ERI*AJ1/2.0
   1123         ECS(4)=ERI*AJ3/2.0
   1125         MPVT=MPIV(3)
   1127         AJ3=B3(MPVT)/A3(MPVT)
   1132         MPVT=MPIV(2)
   1133         AJ1=B3(MPVT)-A3(MPVT)*AJ3
   1137         MPVT=MPIV(1)
   1140         AJ0=B3(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
   1146         PMD(NP3)=PMD(NP3)+RHO*FMO*AJ0/DELT
   1153         ECS(5)=ERI*AJ1/2.0
   1155         ECS(6)=ERI*AJ3/2.0
   1157         IF(NP4.EQ.0) GO TO 126
   1173         MPVT=MPIV(3)
   1174         AJ3=B4(MPVT)/A3(MPVT)
   1176         MPVT=MPIV(2)
   1200         AJ1=B4(MPVT)-A3(MPVT)*AJ3
   1204         MPVT=MPIV(1)
   1205         AJ0=B4(MPVT)-A2(MPVT)*AJ1-A3(MPVT)*AJ3
   1213         PMD(NP4)=PMD(NP4)+RHO*FMO*AJ0/DELT
   1220         ECS(7)=ERI*AJ1/2.0
   1222         ECS(8)=ERI*AJ3/2.0
   1224         FCS(1)=TAU*(X3(NP4)-X3(NP2))/2.0
   1230         FCS(2)=TAU*(X1(NP2)-X1(NP4))/2.0
   1235         FCS(3)=TAU*(X3(NP1)-X3(NP3))/2.0
   1240         FCS(4)=TAU*(X1(NP3)-X1(NP1))/2.0
   1244         FCS(5)=TAU*(X3(NP2)-X3(NP4))/2.0
   1246         FCS(6)=TAU*(X1(NP4)-X1(NP2))/2.0
   1253         FCS(7)=TAU*(X3(NP3)-X3(NP1))/2.0
   1255         FCS(8)=TAU*(X1(NP1)-X1(NP3))/2.0
   1262         GO TO 127
   1273     126 FCS(1)=TAU*(X3(NP3)-X3(NP2))/2.0
   1276         FCS(2)=TAU*(X1(NP2)-X1(NP3))/2.0
   1301         FCS(3)=TAU*(X3(NP1)-X3(NP3))/2.0
   1306         FCS(4)=TAU*(X1(NP3)-X1(NP1))/2.0
```

```
SETUP
   1310         FCS(5)=TAU*(X3(NP2)-X3(NP1))/2.0
   1315         FCS(6)=TAU*(X1(NP1)-X1(NP2))/2.0
   1320     127 SMALLOUT(ECS,ECL(INDE),8)
   1326         SMALLOUT(FCS,FCL(INDF),8)
   1334         GO TO 101
   1334     900 NLT=NL
   1336         READ(5,511) (NPBP(J),J=1,JLIM)
   1344         WRITE(6,512) (NPBP(J),J=1,JLIM)
   1356         CST=COS(THETA)
   1363         SNT=SIN(THETA)
   1372         DO 131 J=1,JLIM
   1405         NPX=NPBP(J)
   1406         IPTP(NPX)=1
   1406     131 RAY(J)=X3(NPX)*CST+X1(NPX)*SNT
   1414         RAYMIN=0.0
   1415         DO 132 J=1,JLIM
   1422         IF(RAY(J).LT.RAYMIN) RAYMIN=RAY(J)
   1426     132 CONTINUE
   1431         DO 133 J=1,JLIM
   1441         RAY(J)=RAY(J)-RAYMIN
   1442         KS(J)=RAY(J)/(VSN*DELT)
   1444         DTS=RAY(J)/(VSN*DELT)-FLOAT(KS(J))
   1446     133 WTS(J)=1.0-DTS
   1452         A22=0.0
   1453         JLS=JLIM-1
   1455         DO 134 J=1,JLS
   1462         NPY=NPBP(J+1)
   1464         NPX=NPBP(J)
   1467         FDL=0.5*SQRT((X1(NPY)-X1(NPX))**2+(X3(NPY)-X3(NPX))**2)
   1506         B22=FDL*RHON*VSN
   1510         A22=A22+B22
   1511         CF2(J)=1.0/(A22+PMD(NPX))
   1516     134 A22=B22
   1522         NPX=NPBP(JLIM)
   1524         CF2(JLIM)=1.0/(A22+PMD(NPX))
   1530         READ(5,513) LIM,(NSV(L),IOPS(L),NCMP(L),L=1,LIM)
   1562         WRITE(6,514) LIM,(NSV(L),IOPS(L),NCMP(L),L=1,LIM)
   1617         RETURN
   1620     901 WRITE(6,516)
   1624         STOP
   1632         END


SUBPROGRAM LENGTH

03063

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
101    - 000172    120    - 000514    121    - 000557    122    - 000655    123    - 000671    124    - 000764
125    - 001042    126    - 001264    127    - 001321    132    - 001427    501    - 001636    502    - 001640
503    - 001642    504    - 001655    505    - 001664    506    - 001700    507    - 001704    511    - 001710
512    - 001712    513    - 001717    514    - 001723    516    - 001743    900    - 001335    901    - 001621
```

```
SETUP

BLOCK NAMES AND LENGTHS
  'LCM'- 0106240
 //    - 026520/01

VARIABLE ASSIGNMENTS
AA      - 003014      AJ0     - 003046      AJ1     - 003045      AJ3     - 003044      ALL     - 0000000/LC A1      - 002114
A2      - 002117      A22     - 003055      A3      - 002122      BB      - 003015      BX1     - 003005      BX3     - 003006
BY1     - 003007      BY3     - 003010      B1      - 002125      B2      - 002130      B22     - 003061      B3      - 002133
B4      - 002136      CC      - 003016      CF2     - 024240/01   CN      - 000001      CST     - 003050      DTS     - 003054
ECL     - 0047040/LC  ECS     - 002074      ER      - 003000      ERA     - 000000/01   ERI     - 003001      FACTOR  - 002765
FCL     - 0066540/LC  FCS     - 002104      FDL     - 003060      FMO     - 003017      FMX     - 003020      FMY     - 003021
GMAX    - 002777      INDE    - 003002      INDF    - 003003      IOPS    - 000005      IPTP    - 013560/01   J       - 003047
JJ      - 003041      JLIM    - 000004      JLS     - 003056      KS      - 025060/01   L       - 003062      LIM     - 000010
MORE    - 002766      MPIV    - 002141      MPVP    - 003043      MPVT    - 003042      NCMP    - 000006      NL      - 002767
NLT     - 000003      NP      - 002764      NPBP    - 023420/01   NPT     - 000002      NPX     - 003052      NPY     - 003057
NP1     - 002770      NP2     - 002771      NP3     - 002772      NP4     - 002773      NSV     - 000007      N1      - 003720/01
N2      - 005670/01   N3      - 007640/01   N4      - 011610/01   PIV     - 003040      PMD     - 021450/01   RAY     - 002144
RAYMIN  - 003053      RHO     - 002774      SIGY2   - 000000      SNT     - 003051      TAU     - 002776      TAUA    - 001750/01
TEMP    - 003004      VS      - 002775      WTS     - 025700/01   XC      - 003024      XP2     - 003013      XP3     - 003012
XP4     - 003022      X1      - 015530/01   X1C     - 003026      X1P1    - 003030      X1P2    - 003032      X1P3    - 003034
X1P4    - 003036      X3      - 017500/01   X3C     - 003027      X3P1    - 003031      X3P2    - 003033      X3P3    - 003035
X3P4    - 003037      YC      - 003025      YP2     - 003011      YP4     - 003023

START OF CONSTANTS-001635       TEMPS--001763    INDIRECTS-002062

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 051300
```

```
MODEL
                SUBROUTINE MODEL(MAX,SIGY,SIGY2,CN)
                REAL SIGY(10),SIGY2(10),CN(10)
                MAX=10
      7         DO 1 J=1,9
     13         SIGY(J)=0.1*FLOAT(J)
     15       1 SIGY2(J)=SIGY(J)*SIGY(J)
     17         SIGY(MAX)=0.99
     20         SIGY2(MAX)=SIGY(MAX)*SIGY(MAX)
     20         SIGYP=0.9999
     21         MM=MAX-1
     23         EML=SIGY(1)
     24         CFI=1.0
     26         DO 3 J=1,MM
     33         EM=SIGY(J+1)/(1.0-SIGY(J+1))
     35         CNI=(EM-EML)/(SIGY(J+1)-SIGY(J))-CFI
     41         CN(J)=1.0/CNI
     42         CFI=CFI+CNI
     43       3 EML=EM
     51         EM=SIGYP/(1.0-SIGYP)
     53         CNI=(EM-EML)/(SIGYP-SIGY(MAX))-CFI
     57         CN(MAX)=1.0/CNI
     60         SQ2=SQRT(2.0)
     63         DO 4 J=1,MAX
     73         SIGY(J)=SQ2*SIGY(J)
     74       4 SIGY2(J)=2.0*SIGY2(J)
     76         RETURN
     76         END


SUBPROGRAM LENGTH

00144

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
CFI    - 000140     CNI    - 000142     EM     - 000141     EML    - 000137     J      - 000134     MM     - 000136
SIGYP  - 000135     SQ2    - 000143

START OF CONSTANTS-000101       TEMPS--000107    INDIRECTS-000127

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 044000
```

```
FILTER
                SUBROUTINE FILTER(XX,F1,F2,N,DTF,NKF)
                REAL XX(1)
                REAL YY(8000)
                REAL P(1000)
                PI=3.14159
     11         P0=(F1+F2)*DTF
     13         W1=2.0*PI*F1
     15         W2=2.0*PI*F2
     16         C=DTF/(2.0*PI)
     17         Q=1.0/(2.0*(F2-F1))
     22         DO 1 I=1,N
     24         T=FLOAT(I)*DTF
     25         TMQ=T-Q
     27         ADD=PI*(F2-F1)
     31         IF(TMQ.NE.0) ADD=SIN(PI*TMQ*(F2-F1))/TMQ
     43         ADD=C*SIN((W1+W2)*T/2.0)*ADD
     56       1 P(I)=C*(SIN(W1*T)+SIN(W2*T))*(1.0/T-0.5/(T+Q))+ADD
    106         DO 2 J=1,NKF
    107         CUM=P0*XX(J)
    111         DO 3 I=1,N
    113         IF((I+J).GT.NKF) GO TO 4
    116         CUM=CUM+P(I)*XX(J+I)
    122       3 CONTINUE
    124       4 DO 5 I=1,N
    126         IF((J-I).LT.1) GO TO 2
    131         CUM=CUM+P(I)*XX(J-I)
    135       5 CONTINUE
    137       2 YY(J)=CUM
    144         DO 6 J=1,NKF
    150       6 XX(J)=YY(J)
    152         RETURN
    152         END


SUBPROGRAM LENGTH

21703

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
2       - 000140     4       - 000125

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
ADD     - 021700     C       - 021673     CUM     - 021702     I       - 021675     J       - 021701     P       - 017717
PI      - 021667     P0      - 021670     Q       - 021674     T       - 021676     TMQ     - 021677     W1      - 021671
W2      - 021672     YY      - 000217

START OF CONSTANTS-000155       TEMPS--000162    INDIRECTS-000216

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 044000
```

```
NLPLT
      SUBROUTINE NLPLT(VF2,VSAVE,Y,NK,DELT,ALPHA)
      COMMON/CCPOOL/XMIN,XMAX,YMIN,YMAX,CCXMIN,CCXMAX,CCYMIN,CCYMAX
      LARGE ALL(20000),ECL(8000),FCL(8000)
      LARGE VF2(8000)
      LARGE VSAVE(8000,16)
      REAL X(8002),Y(8002)
      INTEGER ALPHA(8),XALPH(3),YALPH(3)
      REAL YBASE(24)
      INTEGER LY(24)
  501 FORMAT(///1X,*SUBROUTINE NLPLT*//)
  502 FORMAT(I1,I2,F3.0,F4.0,2I2,3A10,3A10)
  503 FORMAT(//5X,*MORE=*,I2/5X,*NCH=*,I3/5X,*XMAX=*,
     *F5.0/5X,*YMAX=*,F6.0/5X,*NX=*,I3/
     * 5X,*NY=*,I3/5X,*XALPH=*,1X,3A10/5X,*YALPH=*,1X,3A10/)
  504 FORMAT(12(I2,F4.0))
  505 FORMAT(5X,*LY(J)*,2X,*YBASE(J)*/(5X,I5,F10.0))
  506 FORMAT(F2.0)
  507 FORMAT(F4.0)
      WRITE(6,501)
   13 CCXMIN=100.
   14 CCXMAX=1200.
   16 CCYMIN=130.
   17 CCYMAX=1030.
   21 XMIN=0.0
   21 YMIN=0.0
   26 1 READ(5,502) MORE,NCH,XMAX,YMAX,NX,NY,
     * (XALPH(J),J=1,3),(YALPH(J),J=1,3)
   52 WRITE(6,503) MORE,NCH,XMAX,YMAX,NX,NY,
     * (XALPH(J),J=1,3),(YALPH(J),J=1,3)
  101 CALL CCGRID(1,NX,6HNOLBLS,1,NY)
  105 WRITE(98,506) XMIN
  116 CALL CCLTR(95.,100.,0,2)
  121 WRITE(98,507) XMAX
  132 CALL CCLTR(1180.,100.,0,2)
  135 CALL CCLTR(200.,1050.,0,2,ALPHA,80)
  145 CALL CCLTR(400.,20.,0,2,XALPH,30)
  151 CALL CCLTR(30.,430.,1,2,YALPH,30)
  155 READ(5,504) (LY(J),YBASE(J),J=1,NCH)
  202 WRITE(6,505) (LY(J),YBASE(J),J=1,NCH)
  227 DO 2 K=1,NK
  237 2 X(K)=FLOAT(K-1)*DELT
  242 DO 3 J=1,NCH
  247 LYJ=LY(J)
  251 YBASJ=YBASE(J)
  253 IF(LYJ.EQ.92) GO TO 4
  256 SMALLIN(Y,VSAVE(1,LYJ),NK)
  266 DO 6 K=1,NK
  272 6 Y(K)=Y(K)+YBASJ
  274 GO TO 3
  300 4 SMALLIN(Y,VF2,NK)
  307 DO 7 K=1,NK
  313 7 Y(K)=Y(K)+YBASJ
  321 3 CALL CCPLOT(X,Y,NK)
  331 CALL CCNEXT
  332 IF(MORE.EQ.1) GO TO 1
  340 CALL CCEND
```

```
NLPLT
    341         RETURN
    342         END


SUBPROGRAM LENGTH

20261

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       - 000027     3       - 000322     4       - 000301     501     - 000346     502     - 000352     503     - 000357
504     - 000403     505     - 000406     506     - 000414     507     - 000416

BLOCK NAMES AND LENGTHS
  *LCM- 0106240
CCPOOL - 000010/01

VARIABLE ASSIGNMENTS
ALL     - 0000000/LC CCXMAX - 000005/01  CCXMIN - 000004/01  CCYMAX - 000007/01  CCYMIN - 000006/01  ECL     - 0047040/LC
FCL     - 0066540/LC J      - 020255     K      - 020256     LY     - 020221     LYJ    - 020257     MORE    - 020251
NCH     - 020252     NX     - 020253     NY     - 020254     X      - 000461     XALPH  - 020163     XMAX    - 000001/01
XMIN    - 000000/01  YALPH  - 020166     YBASE  - 020171     YBASJ  - 020260     YMAX   - 000003/01  YMIN    - 000002/01

START OF CONSTANTS-000345       TEMPS--000451    INDIRECTS-000457

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 044700
```

```
XTRM
                SUBROUTINE XTRM(Y,NK)
                REAL Y(8000)
              501 FORMAT(5X,*MAX=*,E12.4,1X,*AT K=*,I6/
                * 5X,*MIN=*,E12.4,1X,*AT K=*,I6//)
                KMAX=0
         4      KMIN=0
         5      YMAX=0.0
         6      YMIN=0.0
         7      DO 1 K=1,NK
        10      IF(Y(K).GT.YMAX) KMAX=K
        14      IF(Y(K).GT.YMAX) YMAX=Y(K)
        21      IF(Y(K).LT.YMIN) KMIN=K
        25      IF(Y(K).LT.YMIN) YMIN=Y(K)
        31    1 CONTINUE
        34      WRITE(6,501) YMAX,KMAX,YMIN,KMIN
        47      RETURN
        50      END


SUBPROGRAM LENGTH

00105

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       - 000032     501     - 000054

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
K       - 000104     KMAX    - 000100     KMIN    - 000101     YMAX    - 000102     YMIN    - 000103

START OF CONSTANTS-000053       TEMPS--000070    INDIRECTS-000072

7600 COMPILATION -- RUN76 LEVEL 20      29 NOV 77

ROUTINE COMPILES IN 044000
```

ANTIPLANE STRAIN NONLINEAR GROUND RESPONSE

SAMPLE PROBLEM ANTIPLANE STRAIN

DELT= .0200000
NK= 150
RHON= 2.600
VSN= 2.0000E+05
THETA= 0.0
NPT= 24
JLIM= 9
ITAPE=-0
F1= 3.75
F2= 7.50

POINT LIST

FACTOR= 1.00000E+03

NP	X1	X3	NP	X1	X3	NP	X1	X3	NP	X1	X3	NP	X1	X3
1	0.00	0.00	2	1.00	1.00	3	1.00	0.00	4	2.00	2.00	5	2.00	1.00
6	2.00	0.00	7	3.00	3.00	8	3.00	2.00	9	3.00	1.00	10	3.00	0.00
11	4.00	3.00	12	4.00	2.00	13	4.00	1.00	14	4.00	0.00	15	5.00	3.00
16	5.00	2.00	17	5.00	1.00	18	5.00	0.00	19	6.00	2.00	20	6.00	1.00
21	6.00	0.00	22	7.00	1.00	23	7.00	0.00	24	8.00	0.00			

ELEMENT LIST

NL	N1	N2	N3	N4	RHO	VS	TAU
1	1	3	2	0	2.0000	3.0000E+04	1.8500E+06
2	2	5	4	0	2.0000	3.0000E+04	1.8500E+06
3	2	3	6	5	2.0000	3.0000E+04	1.8500E+06
4	4	8	7	0	2.0000	3.0000E+04	1.8500E+06
5	4	5	9	8	2.0000	3.0000E+04	1.8500E+06
6	5	6	10	9	2.0000	3.0000E+04	1.8500E+06
7	7	8	12	11	2.0000	3.0000E+04	1.8500E+06
8	8	9	13	12	2.0000	3.0000E+04	1.8500E+06
9	9	10	14	13	2.0000	3.0000E+04	1.8500E+06
10	11	12	16	15	2.0000	3.0000E+04	1.8500E+06
11	12	13	17	16	2.0000	3.0000E+04	1.8500E+06
12	13	14	18	17	2.0000	3.0000E+04	1.8500E+06
13	15	16	19	0	2.0000	3.0000E+04	1.8500E+06
14	16	17	20	19	2.0000.	3.0000E+04	1.8500E+06
15	17	18	21	20	2.0000	3.0000E+04	1.8500E+06
16	19	20	22	0	2.0000	3.0000E+04	1.8500E+06
17	20	21	23	22	2.0000	3.0000E+04	1.8500E+06
18	22	23	24	0	2.0000	3.0000E+04	1.8500E+06

BOUNDARY POINTS

1 3 6 10 14 18 21 23 24

```
SAVE POINTS

   LIM=   3

      1 3 2       4 3 2      11 3 2


EXTREME VALUES

  INPUT

    MAX=  4.9000E+01 AT K=     10
    MIN= -1.2292E-12 AT K=     20


  L=  1   NSV=     1   IOPS= 3   NCMP= 2

    MAX=  4.9579E+01 AT K=     11
    MIN= -7.6264E-01 AT K=     21


  L=  2   NSV=     4   IOPS= 3   NCMP= 2

    MAX=  6.2679E+01 AT K=     15
    MIN= -2.5890E+01 AT K=     30


  L=  3   NSV=    11   IOPS= 3   NCMP= 2

    MAX=  7.6086E+01 AT K=     16
    MIN= -3.4569E+01 AT K=     31


SUBROUTINE NLPLT


    MORE=-0
    NCH=  4
    XMAX=     5
    YMAX=   400
    NX=  5
    NY=  8
    XALPH= SECONDS
    YALPH= VELOCITY (CM/SEC)

    LY(J)  YBASE(J)
       92        100
        1        200
        2        250
        3        300
```

ANTIPLANE STRAIN PROGRAM

LISTING OF INPUT DECK

FOR SAMPLE PROBLEM

INPUT CARDS FOR SAMPLE PROBLEM - ANTIPLANE STRAIN PROGRAM

```
         10        20        30        40        50        60        70        80
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
SAMPLE PROBLEM  ANTIPLANE STRAIN
 0.02    150 2.6        2.0E+05 0.0   24  9  3.75 7.5
  10.0E+02
       0       0       1       1       1       0       2       2       2       1
       2       0       3       3       3       2       3       1       3       0
       4       3       4       2       4       1       4       0       5       3
       5       2       5       1       5       0       6       2       6       1
       6       0       7       1       7       0       8       0
1   1   3   2   0 2.0    0.3E+05  1.85E+06
1   2   5   4   0 2.0    0.3E+05  1.85E+06
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
1   2   3   6   5 2.0    0.3E+05  1.85E+06
1   4   8   7   0 2.0    0.3E+05  1.85E+06
1   4   5   9   8 2.0    0.3E+05  1.85E+06
1   5   6  10   9 2.0    0.3E+05  1.85E+06
1   7   8  12  11 2.0    0.3E+05  1.85E+06
1   8   9  13  12 2.0    0.3E+05  1.85E+06
1   9  10  14  13 2.0    0.3E+05  1.85E+06
1  11  12  16  15 2.0    0.3E+05  1.85E+06
1  12  13  17  16 2.0    0.3E+05  1.85E+06
1  13  14  18  17 2.0    0.3E+05  1.85E+06
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
1  15  16  19   0 2.0    0.3E+05  1.85E+06
1  16  17  20  19 2.0    0.3E+05  1.85E+06
1  17  18  21  20 2.0    0.3E+05  1.85E+06
1  19  20  22   0 2.0    0.3E+05  1.85E+06
1  20  21  23  22 2.0    0.3E+05  1.85E+06
   22  23  24   0 2.0    0.3E+05  1.85E+06
   1   3   6  10  14  18  21  23  24
     3   132   432  1132
 .0       E+00 .1       E+00 .2       E+00 .3       E+00 .4       E+00
 .5       E+00 .4       E+00 .3       E+00 .2       E+00 .1       E+00
....+....*....+....*....+....*....+....*....+....*....+....*....+....*....+....*
 .0       E+00-.1       E+00-.2       E+00-.3       E+00-.4       E+00
-.5       E+00-.4       E+00-.3       E+00-.2       E+00-.1       E+00
          26 ADDITIONAL CARDS IDENTICAL TO THE PRECEDING
  4  5 400 5 8SECONDS                        VELOCITY (CM/SEC)
92 100 1 200 2 250 3 300
```