A FORTRAN PROGRAM FOR CALCULATING
NONLINEAR SEISMIC GROUND RESPONSE

by

William B. Joyner

U.S. Geological Survey
OPEN FILE REPORT
77-671

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 program described here was designed for calculating the non-linear seismic response of a system of horizontal soil layers underlain by a semi-infinite elastic medium representing bedrock. Excitation is a vertically incident shear wave in the underlying medium. The non-linear hysteretic behavior of the soil is represented by a model (Figure 1) proposed by Iwan (1967) consisting of simple linear springs and Coulomb friction elements arranged as shown. 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. A brief program description is provided here with instructions for preparing the input and a source listing. A more detailed discussion of the method is presented elsewhere (Joyner and Chen, 1975) as is the description of a different program employing implicit integration (Chen, 1975).

Soil Layers

The physical properties of the soil layers are specified on input cards. For the purposes of computation the layers are divided into sublayers such that the transit time, DTT, of a shear wave at low strain across the sublayer is a constant. The division into sublayers is done automatically in such a way that a sublayer may span the boundary between layers. Since an integral number of sublayers is required, each with a fixed shear-wave transit time, the total thickness of the system of sublayers used in computation may
differ slightly from the sum of layer thicknesses specified on the input cards. The transit time DTT is ordinarily taken the same as the finite-difference time step, DELT, but the user is given the option of specifying a different value (which must be larger than DELT to prevent instability).

In the coordinate system used the vertical dimension is taken as positive downward. This choice controls the sign of stress and strain.

The dynamic behavior of the soil is specified in terms of the natural density, \( RHOI \), the low-strain shear modulus, \( GMAX \), and the dynamic shear strength, \( TAU \). \( RHOI \) is specified on the input cards, and \( GMAX \) and \( TAU \) can either be specified on the input cards or computed by the program. In the latter case the computations are done assuming \( GMAX \) and \( TAU \) are functions of effective stress. For normally consolidated soil the assumed relationships are (see Hardin and Drnevich, 1972)

\[
GMAX = GCFN \times PV \times 0.5 \tag{1}
\]

and

\[
TAU = TCF \times PV \tag{2}
\]

where \( GCFN \) and \( TCF \) are coefficients specified on the input cards and \( PV \), the vertical effective stress, is computed as a function of depth from data specified on the input cards. If there is a continuous water phase present in the soil, \( PV \) is given by

\[
PV = PVTL - SAT \times 980 \times (Z-PHREAT)
\]

where \( PVTL \) is the total vertical stress, \( SAT \) is the fractional water saturation, and \( PHREAT \) is the depth of the water table from the ground.
surface. This is an approximation to the relation proposed by Bishop and others (1961). If a continuous water phase is not present PV is set equal to PVTL. For overconsolidated soil the relationships for TAU and GMAX are

\[
G_{\text{MAX}} = G_{\text{CFN}} \times \text{OCR} \times XQ \times PV^{0.5} \tag{3}
\]

and

\[
\tau = T_{\text{CF}} \times \text{OCR} \times XT \times PV \tag{4}
\]

where the overconsolidation ratio,

\[
\text{OCR} = \frac{\text{SCV}}{\text{PV}}
\]

SCV is the preconsolidation vertical stress and is specified for a layer on the input card. The exponents XQ and XT are specified on the input card.

If the value of TAU for a sublayer is computed by the program rather than being specified on the input cards, the value assigned to the sublayer is the value computed by equation (2) or (4) for the point within the sublayer that represents the midpoint in terms of shear wave travel time at low strain. If the value of GMAX is computed by the program, the value assigned is a special kind of average which is equal to the shear modulus of a constant-modulus sublayer that would have the same thickness for the given shear wave transit time as implied by equation (2) or (4).

The stress-strain behavior of the soil in shear is that given by Iwan's (1967) model and is determined by specifying the initial loading curve. For convenience, stress and strain are normalized in the manner used by Hardin and Drnevich (1972). Stress is normalized by multiplying by \(1/\tau\) and strain is normalized by multiplying by
GMAX/TAU. The stress-strain curve normalized in this way has a stress limit of 1.0 for high strain and a slope at the origin of 1.0. A hyperbolic initial loading curve (Hardin and Drnevich, 1972) is used, and normalized strain, \( e \), is expressed in terms of normalized stress, \( s \), by the equation
\[
e = \frac{s}{1 - s}
\]
The initial loading curve for Iwan's model is a series of straight line segments. These are determined by selecting 51 normalized stress values, solving the preceding equation for the corresponding normalized strain values, and connecting the resulting stress-strain points by straight line segments. The 51 normalized stress values used are

\[
\begin{align*}
    s_1 &= 0 \\
    s_i &= 0.025 * (0.5)^{(6 - i)} & 2 \leq i \leq 6 \\
    s_i &= 0.025 * (i - 5) & 6 \leq i \leq 44 \\
    s_i &= 1.0 - 0.025 * (0.5)^{(i - 44)} & 45 \leq i \leq 51
\end{align*}
\]
The same normalized initial loading curve is used for all the soil sublayers. The differences in behavior from one sublayer to another result from differences in the values of GMAX and TAU assigned to the different sublayers.

**Input Motion**

The input motion is read in from punch cards and is the horizontal particle acceleration in \( g \) that would be expected at the bedrock interface if the soil layers were absent and the interface were a free surface. The acceleration values are given at a time...
interval DELT which is the same as the time step for the finite-difference computations. After the input acceleration values are read in, they are integrated to give particle velocity, which is used in the computations.

Resolution, Stability and Filtering

An important choice in setting up the input for a problem is the frequency resolution desired. Denote by $f_R$ the highest frequency for which faithful representation is desired in the output. This requirement implies 8 to 10 sublayers per wavelength. At low strain 10 sublayers per wavelength means a value of DTT equal to $1/(10\times f_R)$. In the ordinary case the program assigns to DTT a value equal to DELT, and the desired resolution at low strain can be obtained by specifying a value of $1/(10\times f_R)$ for DELT on the input cards. At high strain levels the resolution will be somewhat less.

The stability requirement for the linear elastic case (Richtmyer and Morton, 1967, p. 263) is that DTT must be greater than or equal to DELT. There is no guarantee that this will insure stability in a nonlinear problem, but the writer has encountered no cases of instability in runs where DTT was equal to DELT. If the user should happen to encounter a case of instability he can probably cure it by specifying a larger value of DTT. Provision is made in the program for specifying DTT separately.

The finite difference computations generate numerical noise at frequencies in the vicinity of $1/(4 \times \text{DELT})$. This noise is generally not noticeable on the surface particle velocity time history and it
has little effect on response spectral values at lower frequencies, but it is conspicuous in the surface acceleration time history. In general it is desirable to filter out this noise, and the program provides the option of digitally filtering output time histories by a zero-phase-shift filter with response, \( R \), given by

\[
R(f) = \begin{cases} 
1 & f \leq F_1 \\
0.5 \times (1.0 + \cos(\pi(f - F_1)/(F_2 - F_1))) & F_1 < f < F_2 \\
0 & f \geq F_2
\end{cases}
\]

where \( f \) is frequency and \( F_1 \) and \( F_2 \) are parameters specified on an input card. If either \( F_1 \) or \( F_2 \) is zero no filtering is done. It is recommended that \( F_1 \) be assigned a value equal to the desired frequency resolution \( f_R \) and that \( F_2 \) be assigned the value \( 2f_R \). If filtering is done the number of points in the output time histories is reduced by 20.

**General Description of Output**

All output quantities are expressed in cgs units except for acceleration, which is given as a fraction of the acceleration of gravity. The printout lists the values of the input parameters. There is also a list of sublayers, giving sequence number and physical properties along with depth, effective vertical stress and total vertical stress evaluated at the bottom of the sublayer. The input parameters for the layers are inserted into the sublayer list at the appropriate place. Also indicated in sublayer list are those places where time histories will be saved as designated on the input cards. Up to three time histories may be saved. They may be of particle
velocity, stress, or strain at the option of the user and they be located at any depth within the section.

The program as described here contains a dummy plot subroutine called SMPLT which simply lists time histories. The user may substitute a plot subroutine suitable for his installation that will produce graphic plots of the time histories.

Following the sublayer list the printout gives the maximum and minimum values of the input acceleration along with the sequence numbers of the time points at which the maximum and minimum occurred. This is followed by a listing of the input accleration from the dummy plot routine. Next is a list of sublayers giving the maximum absolute value of strain in each sublayer. These are unfiltered values and therefore may disagree slightly with subsequent listings which may be filtered. Following that, data are given in turn for surface particle velocity, surface acceleration and any time histories that are saved. For each the maximum and minimum values are given along with a listing from the dummy plot subroutine.

The output also includes a punch card deck for the surface acceleration. The cards have sequence numbers punched in columns 71 - 76, and the FORMAT is (5E14.8, I6).

For the output time histories, including both listings and punch card decks, the number of points, if no filtering is used, is NK, the same as the number of points on the input. If filtering is done the number of output points is (NK - 20).
Instructions for Preparing Input

As with the output, all input quantities are expressed in cgs units 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 (20A4)**

**ALPHA (J)** Any desired alphabetic identification

Card No. 2

**FORMAT (I5, F10.0, 2F5.0)**

**NK** Number of points on input time history. NK must not exceed 5000.

**DELT** Time intervals for input data points and time step for finite difference computations.

**F1, F2** Parameters specifying response of filter. If either F1 or F2 is zero, no filtering is done.

Card No. 3

**FORMAT (F10.0, E10.0, F10.0)**

**RHON** Density of elastic substratum

**VN** Shear velocity of elastic substratum

**DTT** Shear wave transit time through a sublayer at low strain. Ordinarily this should be left blank and the program will assign to DTT the value specified for DELT.

Card No. 4

**FORMAT (I1, 9X, 7(I1, E9.0))**

Input variable list: LIM, (IOPS(L), DS(L), L = 1, LIM)

**LIM** Number of time histories to be saved. May be zero or any number up to 3. (The limitation is imposed by dimension statements in the main program; the FORMAT statement above would allow for more.)
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOPS(L)</td>
<td>Code indicating kind of time history to be saved. IOPS(L) = 1 for strain, 2 for stress and 3 for particle velocity.</td>
</tr>
<tr>
<td>DS(L)</td>
<td>Depth for which time history is to be saved. If stress or strain is chosen, the time history saved applies to the sublayer containing the specified depth. If particle velocity is chosen the time history saved applies to the point at the top of the sublayer containing the specified depth.</td>
</tr>
<tr>
<td>Card(s) No. 5</td>
<td>FORMAT (I1, E8.0, F5.0, 2E8.0, F3.0, I1, E8.0, F3.0, E8.0, F3.0, 3E8.0)</td>
</tr>
<tr>
<td>MORE</td>
<td>Code indicating whether additional layer cards follow. MORE = 1 for all layer cards except the last, for which it may be anything other than 1.</td>
</tr>
<tr>
<td>THICK</td>
<td>Thickness (in cm, remember) of the layer.</td>
</tr>
<tr>
<td>RHOI</td>
<td>Density</td>
</tr>
<tr>
<td>SCV</td>
<td>Preconsolidation vertical stress.</td>
</tr>
<tr>
<td>PHREAT</td>
<td>Depth of water table from ground surface.</td>
</tr>
<tr>
<td>SAT</td>
<td>Degree of saturation (as a fraction not a percentage).</td>
</tr>
<tr>
<td>IFPH</td>
<td>IFPH = 1 if a continuous water phase is present and 0 if not. (A value of 0 for IFPH causes effective stress to be set equal to total stress).</td>
</tr>
<tr>
<td>GCFN</td>
<td>Parameter for computing GMAX by equations (1) or (3). May be left blank if GMAX is specified on card.</td>
</tr>
</tbody>
</table>
XQ Parameter for computing GMAX by equation (3). May be left blank if GMAX is specified on card.

TCF Parameter for computing TAU by equations (2) or (4). May be left blank if TAU is specified on the card.

XT Parameter for computing TAU by equation (4). May be left blank if TAU is specified on the card.

TAUC Specified value for TAU. Must be blank or zero if TAU is to be computed by equation (2) or (4).

GMAXC Specified value for GMAX. Must be blank or zero if GMAX is to be computed by equation (1) or (3).

VIC Specified value for the low strain shear velocity of the layer. If VIC is not equal to zero, GMAX is set equal to RHOI*VIC**2 overriding any specification in the GMAXC field and any computation by equations (1) or (3).

Card(s) No. 6 FORMAT (5E14.8)

Cards giving the values of input acceleration. A total of NK values at a time interval DELT.

Memory and Time Requirements

The program as written requires 64,000 words of core storage on an IBM 370 - 155. For large problems the time requirement on the same machine is approximately one millisecond per sublayer per time step. Running time for large problems is essentially proportional to the product of the number of elements and the number of time steps.
### Variables in Printer Output not Previously Defined

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Index number for sublayer. Numbering is from the top down.</td>
</tr>
<tr>
<td>VOUT</td>
<td>Low-strain shear wave velocity for sublayer.</td>
</tr>
<tr>
<td>DELZ</td>
<td>Thickness of sublayer.</td>
</tr>
<tr>
<td>ER</td>
<td>Reference strain used in normalizing stress-strain curve, equals ( \tau/G_{\text{MAX}} ).</td>
</tr>
<tr>
<td>Z</td>
<td>Depth to base of sublayer.</td>
</tr>
<tr>
<td>PV1</td>
<td>Vertical effective stress at base of sublayer.</td>
</tr>
<tr>
<td>PVTL</td>
<td>Vertical total stress at the base of sublayer.</td>
</tr>
<tr>
<td>L</td>
<td>Index number for time history to be saved.</td>
</tr>
<tr>
<td>IS(L)</td>
<td>Index number of sublayer for which time history is to be saved.</td>
</tr>
<tr>
<td>ZTOP</td>
<td>Depth to the top of sublayer IS(L). If the time history to be saved is particle velocity, then the time history applies to the point at depth ZTOP.</td>
</tr>
<tr>
<td>ZBASE</td>
<td>Depth to the base of sublayer IS(L).</td>
</tr>
<tr>
<td>K</td>
<td>Index number for time values.</td>
</tr>
</tbody>
</table>

### Other Important Variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIGY(J)</td>
<td>Normalized yield stress of friction element J in the Iwan soil model (equivalent to ( \gamma_i ) in Figure 1).</td>
</tr>
<tr>
<td>CF(M, I)</td>
<td>Normalized tangent shear modulus of the soil model for sublayer I where M is the index of the friction element with the largest yield stress of all the elements that are currently yielding. (As the program is now constituted the values of CF are the same for all sublayers. The I index</td>
</tr>
</tbody>
</table>
was included to facilitate program modifications that would permit different sets of CF values for different sublayers).

**NI**
Total number of sublayers.

**ERI(I)**
Parameter for sublayer I; equal to DELT/(DELZ*ER).

**SK(I)**
Parameter for sublayer I; equal to TAU.

**SL(I)**
Parameter for sublayer I; equal to DELT/DELZ.

**TK(I)**
Parameter characterizing the mass lumped at the boundary between sublayer I and the sublayer above; equal to 2*DELT/ (UMASS + BMASS), where BMASS is equal to RHOI * DELZ for sublayer I and UMASS is the corresponding quantity for the sublayer above. When I = 1, UMASS = 0, and when I = NI + 1, BMASS = 0.

**X(K)**
An array containing, at different stages of program execution, the input acceleration time history, the surface particle velocity time history and the surface acceleration time history.

**Y(K, L)**
Time history to be saved for index L.

**BRI**
Shear impedance of bedrock; equal to RHON * VN.

**MAX**
Total number of series units in the Iwan soil model. Each unit consists of a linear spring and a Coulomb friction element in parallel. As the program is now constituted MAX is fixed at 50.

**SIGYP**
Normalized stress for the point with the highest normalized stress of the set of points used in defining the initial loading curve for the soil model.

**VOK**
Current value of input particle velocity, obtained by integrating the input acceleration time history.
V(I)  Current value of particle velocity at the top of sublayer I.
SIG(I)  Current value of the normalized stress in sublayer I.
SIGS(J, I)  Current value of the normalized stress for spring J of the soil model representing sublayer I.
STRN(I)  Current value of strain (not normalized) in sublayer I.
SMAX(I)  Maximum value of strain (not normalized) in sublayer I.
M  Index of the friction element (in the Iwan soil model) with the largest yield stress of all the elements that are currently yielding.
MH(I)  Value of M for sublayer I.
L  (In subroutine NONLI only). A logical variable with the value TRUE if the strain is increasing and FALSE if it is decreasing.
LH(I)  Value of L for sublayer I.
DELE  Increment of normalized strain.
SIGP  Trial value of normalized stress.

Description of Subroutines

The main program reads input cards numbers 1, 2, and 6 and controls the sequence of operations. Subroutine SECT, called by the main program, reads input cards numbers 3, 4, and 5 and sets up the system of sublayers used in the computations, assigning to the sublayers the parameters that define their physical properties. Subroutine SECT also makes the necessary provisions for saving time histories for output.
Subroutine SIGMA, called by SECT, specifies the 51 normalized stress values used in defining the initial loading curve for the soil model. Subroutine MODEL, also called by SECT, generates the array CF(M, I), which describes the tangent modulus of the soil model.

Subroutine NONLI, called by the main program, performs all the dynamic response calculations and saves the output time histories.

The remaining subroutines are all called by the main program and perform functions related to output. Subroutine FILTER does a low-pass filter operation on the output time histories. Subroutine SMPLT is a dummy plot routine that simply lists the output time histories. The user may substitute a routine suitable for his own installation to give a graphic display of the output. Subroutine PUNCH generates a punch-card deck of the surface acceleration time history. Subroutine XTRM finds and prints out the maximum and minimum values of the output time histories along with the index of the time points at which the maximum and minimum occurred.

Sample Problem

To assist the reader in using the program a simplified sample problem is presented with a listing of the input deck and a copy of the printer output. The problem involves a soil layer 2500 cm thick with a density of 2.0 gm/cm$^3$. The water table is at the surface and the soil is fully saturated and normally consolidated throughout. (The assumption of normal consolidation all the way up to the surface was made for simplicity and is actually unrealistic. Some degree of overconsolidation near the surface would be expected for most soil
deposits.) GMAX and TAU are computed by the program using the following parameters:

\[ GCFN = 0.9 \times 10^6 \text{ dynes/cm}^2^{1/2} \]

\[ TAU = 0.33 \]

Values are given on the input card for XQ and XT but they are not needed because the soil is presumed to be normally consolidated.

The bedrock beneath the soil layer is assigned a density of 2.6 gm/cm\(^3\) and a shear wave velocity of 2.0 \(\times\) \(10^5\) cm/sec. The input acceleration is a positive triangular spike of amplitude 0.5g and duration 0.2 sec followed by a negative triangular spike of the same amplitude and duration (Figure 2). The input acceleration values are specified at intervals, DELT, of 0.01 seconds. A value of 300 was chosen for NK in order to allow sufficient time for the main features of the surface time histories to develop. The input acceleration just described is padded at the end with zeros to make up 300 points.

The output is filtered with F1 - 10 Hz and F2 - 20 Hz. Time histories of stress, strain, and particle velocity are saved at a depth of about 1250 cm.

The input acceleration for the problem is shown in Figure 2 and the surface particle velocity and surface acceleration are given in Figures 3 and 4 respectively.
References


Figure 1. - Model for nonlinear soil behavior. Model consists of simple elastic springs with spring constants $G_i$ and Coulomb friction elements with yield stresses $Y_i$. $Y_1$ equals zero.
Figure 2. - Input acceleration for sample problem.
SAMPLE PROBLEM 2

TIME (SECONDS)
Figure 3. - Surface particle velocity for sample problem.
SAMPLE PROBLEM 2

TIME (SECONDS)
Figure 4. - Surface acceleration for sample problem.
SAMPLE PROBLEM 2

TIME (SECONDS)

0.00 0.50 1.00 1.50 2.00 2.50 3.00

0.80 0.60 0.40 0.20 0.00
FORTRAN PROGRAM
C MAIN FOR NONLINEC - VERSION 1

0007 501 FORMAT(204)
0008 502 FORMAT(*1*204/)
0009 503 FORMAT(15*F10.0,2F5.0)
0010 504 FORMAT(5X+NNK=1,\(16/5X+DELT=1,F9.5/5X+FI=1,
0011 * F6.2/5X+FI2=1,F6.2/)
0012 507 FORMAT(SE14.8)
0013 511 FORMAT(1X+INPUT ACCELERATION/)
0014 512 FORMAT(1X+SURFACE VELOCITY/)
0015 513 FORMAT(1X+OUTPUT = 1*14,3*X+10PS=12/)
0016 G=0.00,0
0017 READ(5,501) (ALPHA(J),J=1,20)
0018 WRITE(6,502) (ALPHA(J),J=1,20)
0019 READ(5,503) NK,DELT,F1,F2
0020 WRITE(6,504) NK,DELT,F1,F2
0021 CALL SECTIE(SKI,SL,TK,NK,B1,L1,OPS,LIM,DELT,
0022 SIGV,CF,MAX)
0023 DO 20 L=1,LIM
0024 DO 20 K=1,NK
0025 20 Y(K+L)=0.0
0026 READ(5,507) (X(K),K=1,NK)
0027 WRITE(6,511)
0028 CALL XTRM(X,NK)
0029 CALL SMPLT(X,NK,DELT+ALPHA)
0030 CALL NONLIC(X,Y,IS,10PS,LIM,ERI,SK,SL,TK,NK,DELTI,
0031 SIGV,CF,NK+DELT)
0032 IF(F1.EQ.0.0 .OR.F2.EQ.0.0) GO TO 30
0033 CALL FILTER(X,F1,F2,20*DELT+NK)
0034 DO 30 L=1,LIM
0035 30 WRITE(6,512)
0036 CALL XTRM(X,NK)
0037 CALL SMPLT(X,NK,DELT+ALPHA)
0038 AT1=X(1)/DELT+G
0039 DO 10 K=2,NK
0040 AT2=(X(K)-X(K-1))/DELT+G
0041 X(K-1)=AT1
0042 10 AT1=AT2
0043 X(NK)=AT1
0044 WRITE(6,513)
0045 CALL XTRM(X,NK)
0046 CALL PUNCH(X,NK)
0047 CALL SMPLT(X,NK,DELT+ALPHA)
0048 DO 40 L=1,LIM
0049 40 WRITE(6,514)
FORTRAN IV G LEVEL 21

0049 WRITE(6,514) IS(L),IOPS(L)
0050 CALL XTHM(Y(1:L)+NK)
0051 CALL SMPLT(Y(1:L)+NK+DELT,ALPHA)
0052 STOP
0053 END
SUBROUTINE NONL1(X,Y,IS,IOPS,LIM+ERI,SK,SL,TK,NI,BRI,SIGY, 
# CF+MAX+NI+DELT)
C
NONL13C
REAL*8 VOK
REAL X(5000), Y(5000, 3)
REAL V(201)
REAL SIG(200), SIGS(50, 200)
REAL SIGY(50), CF(50, 200)
REAL SMAX(200), STRN(200)
REAL ERI(200), SK(200), SL(200)
REAL TK(201)
INTEGER IS(I), IOPS(I)
INTEGER NI, NH(200)
LOGICAL LM(200)
501 FORMAT('**MAXIMUM STRAIN PROFILE**')
DO 20 I=1, NI
LM(I)=.TRUE.
mh(I)=
V(I)=0.0
SIG(I)=0.0
SMAX(I)=0.0
 STRN(I)=0.0
DO 20 MH(I)=MAX
20 SIGS(MH(I))=0.0
V(NI+1)=0.0
G=980.0
VOK=0.0
DO 10 K=1, NK
DO 20 J=1, NJ
STRN(I)=STRN(I-1)+V(I-1)-V(I)*SL(I)
TEST=ABS(STRN(I))
IF (TEST.GT.SMAX(I)) SMAX(I)=TEST
CONTINUE
10 CONTINUE
L=LM(NI)
mn=MH(NI)
VOK=VOK+DEL*2(K)
DEL=(V(NI+1)-V(NI))*ERI(NI)
IF (DEL.EQ.0.0) GO TO 11
IF ((L,L,DELE,GT,0.0,0).OR.,(L,.NOT.,L).AND.,DELE,LT,0.0,0) GO TO 12
L=.NOT.,L
LH(NI)=L
M=1
MH(NI)=1
12 SIGP=SIG(NI)+DELE*CF(M,NI)
IF (M.FLT.MAX) GO TO 13
IF (ABS(SIGP-SIGS(M+1,NI)),LT.,SIGY(M+1)) GO TO 13
IF (SIGP,GT,SIGS(M+1,NI)) SIGP=SIGS(M+1,NI)+SIGY(M+1)
IF (SIGP,LT,SIGS(M+1,NI)) SIGP=SIGS(M+1,NI)-SIGY(M+1)
DELE=DELE-(SIGP-SIG(NI))/CF(M,NI)
SIG(NI)=SIGP
M=M+1
MH(NI)=M
0051   GO TO 12
0052   13 SIGS(I+1)=SIGP
0053   IF(M.EQ.1) GO TO 19
0054   DO 14 J=2,M
0055   IF(L) SIGS(J+1)=SIGP-SIGY(J)
0056   IF(.NOT.L) SIGS(J+1)=SIGP+SIGY(J)
0057   14 CONTINUE
0058   19 SIG(NI)=SIGP
0059   11 V(NI+1)=(1.0-1.0/(TK(NI+1)*BRK[1,0]))
0060   DO 5 J=2,NI
0061   I=NI-J+1
0062   L=LN(I)
0063   M=NN(I)
0064   DELE=(V(I+1)-V(I))*EI(I)
0065   IF(DELE.EQ.0.0) GO TO 9
0066   IF((L.AND.DELE.GT.0.0).OR,(.NOT.L).AND.DELE.LT.0.0)) GO TO 2
0067   L=.NOT.L
0068   L(M)=L
0069   M=1
0070   M=M+1
0071   2 SIGP=SIG(I)+DELE*CF(M+1)
0072   IF(M.EQ.MAX) GO TO 3
0073   IF(ABS(SIGP-SIGS(M+1)))LT.SIGY(M+1)) GO TO 3
0074   IF(SIGP.GT.SIGS(M+1)) SIGP=SIGS(M+1)+SIGY(M+1)
0075   IF(SIGP.LT.SIGS(M+1)) SIGP=SIGS(M+1)-SIGY(M+1)
0076   DELE=DELE-(SIGP-SIG(I))/CF(M+1)
0077   SIG(I)=SIGP
0078   M=M+1
0079   M=M+1
0080   GO TO 2
0081   3 SIGS(I+1)=SIGP
0082   IF(M.EQ.1) GO TO 9
0083   DO 4 J=2,M
0084   IF(L) SIGS(J+1)=SIGP-SIGY(J)
0085   IF(.NOT.L) SIGS(J+1)=SIGP+SIGY(J)
0086   4 CONTINUE
0087   9 SIG(I)=SIGP
0088   5 V(I+1)=V(I)+TK(I+1)*SK(I+1)*SIG(I+1)-SK(I)*SIG(I)
0089   40 CONTINUE
0090   10 X(K)=V(I)
0091   WRITE(6,501) (I,SMAX(I)*I=1,NI)
0092   RETURN
0093   500 CONTINUE
0094   END
SUBROUTINE SECT(ERI+5K,SL,TK+NI,RHI+IS,1OPS,LIM,
  * DELT*SIGY,CF,MAXY)
C SECTC3
  REAL ERI(201),SK(201),SL(201)
  REAL TK(201)
  REAL SIGY(50),CF(50,200)
  REAL DS(8)
  INTEGER IS(1):1OPS(1)
  REAL FORMAT/(1,E10.0,E10.0,F10.0)
  510 FORMAT(3X+DTT+*F9.5/)
  502 FORMAT(1ISUBROUTINE SECT+/5X)+RHON+*F8.3/
     5X+VNY+*E12.6/)
  503 FORMAT/(11+X7+9X+VOUTX+9X+DELZ+9X+OMAX+10X+TAU+)*9X+
      504 FORMAT(5X+LIM+*12/5X+1OPS(L)+*7I12)
      514 FORMAT(5X+DS(L)+*2X+7E12.4)
      505 FORMAT(/1X+3X+I+9X+VOUTX+9X+DELZ+9X+OMAX+10X+TAU+)*11X+
      506 FORMAT(/1X+MORE+6X+THICK,3X+RHO1,8X+SCV,5X+PHREAT+)*3X+SAT+2X+IPFH+7X+GCFN+4X+QX+9X+TCF+4X+TX+)*TX+TAUC,6X+GMAXC+LAT+VIC/)
      507 FORMAT(/1X+MORE+6X+THICK,AHON.YN+OTT+WAIT[6.5011]
      508 FORMAT(/1X+LIM+*12/3X+15(L)+*14/3X+1OPS(L)+*12,3X,
             * DS(L)+*1E12,4X+3X+2TOP+*E12,4X+ZRASE+*E12,4/)
      509 FORMAT(1X+4E13.4+F9.4+3E13.4)
      520 CALL SIGMA(MAX+SIGY,SIGYP)
      521 READ(5,501) RHON+VNY+DTT
      522 WRITE(6,502) RHON+VNY
      523 IF(DTT+NE.0,0) WRITE(6,510) DTT
      524 IF(DTT+EN.0,0) DTT=DELT
      525 WRITE(5,503) LIM+1OPS(L)+DS(L)+L+1LIM
      526 IF(LIM+EN.0,0) DS(I)+1,0E+15
      527 WRITE(6,504) LIM+1OPS(L)+L+1LIM
      528 WRITE(6,514) (DS(I)+L+1LIM)
      529 I=0
      530 Z0=0
      531 PVI=0,0
      532 ZLAST=0,0
      533 TUO=0,0
      534 PV1L=0,0
      535 UMAS=0,0
      536 DTH=DTT/2,0
      537 ISW=0
      538 L=1
      539 MORE=1
      540 WRITE(6,505)
      541 100 IF(MORE+NE.1) GO TO 900
      542 READ(5,506) MORE+THICK,RHO1,SCV,PHREAT,SAT+IPFH+
      543 WRITE(6,507) MORE+THICK,RHO1,SCV,PHREAT,SAT+IPFH+
      544
FORTRAN IV LEVEL 21

* GCFSXQ**T'C**XT*TAU*GMX**VIC
PTCF**RHO**90.0
PCOEF**(RHO)*SAT**90.0

0046 IF((IFPH, Equation 0.1) POC**PTCF

0047 PV**PVTL**SAT**90.0*(Z-PTRN Enrique)
0048 IF((IFPH, Equation 0.1) PV**PVTL

0049 GCFS**GCFS
0050 IF((SCV, Equation 0.0) GCFS**GCFS**SCV**XQ

0051 PNC**0.75
0052 POC**PNC**XQ/2.0
0053 PCN**SORT(GCFS/RHOI)**PCOEF**0.75
0054 PCN**SORT(GCFS/RHOI)**PCOEF**PCO

0055 IF((VIC, Equation 0.0) AND (GMX**NE, Equation 0.0) VIC=SQRT(GMEX/RHOI)

0056 SVC**PV**THICK**POCEF

0057 SLIM**SVB
0058 IF((SLIM, GT, SCV) SLIM=SCV

110 DX**DT=SU

0060 IF((SCV, LE, PV) GO TO 120

0061 PV**PV**POC**PCN**DTX**1.0/POC

0062 IF((VIC, NE, Equation 0.0) PV**PV**POC**VIC**DTX

0063 IF((PV**PV**SLIM) GO TO 130

0064 IF((VIC, Equation 0.0) TADD**SLIM**POC**PV**POC**PCO

0065 IF((VIC, NE, Equation 0.0) TADD**SLIM**PV**VIC**POCEF

0066 TU**TU**TADD

0067 ZADD**SLIM**PV**POC

0068 PVTL**PVTL**ZADD**POCEF

0070 PV**SLIM
0071 IF((SLIM, EQ, SVB) GO TO 100

0072 GO TO 110

120 PV**PV**POC**PCN**DTX**1.0/POC

0074 IF((VIC, NE, Equation 0.0) PV**PV**POC**VIC**DTX

0075 IF((PV**PV**SLIM) GO TO 130

0076 IF((VIC, EQ, Equation 0.0) TADD**SVB**POC**PV**PHC**PCO

0077 IF((VIC, NE, Equation 0.0) TADD**SVB**PV**VIC**PCO

0078 TU**TU**TADD

0079 ZADD**SVB**PV**POC

0080 PV**PV**SVB

0082 PV**SVB

0083 GO TO 100

0084 130 DV**PV**PV**POC

0085 ZADD**DV

0086 PV**PV**SVB

0087 ZADD**PV

0088 TP**TP**EQ**1 GO TO 140

0089 TAU**TP**PV

0090 IF((SCV, GT, PV) TAU**TAU**SCV**PV**XT

0092 IF((TAU, NE, Equation 0.0) TAU**TAU

0093 ISP**1

0094 GO TO 110
0095 140 DELZ=ZLAST
0096 141 I=I+1
0097 142 IF(I.GT.200) GO TO 920
0098 0099 WRITE(6,508) L-IS(L)+1OPS(L)+DS(L)+ZLAST+Z
0100 0101 L=L+1
0102 0103 IF(L.GT.LIM) GO TO 144
0104 0105 DS(L)=L*OE+15
0106 143 ZLAST+Z
0107 0108 VOUT=DELZ/DTT
0109 0110 BMSS=RHOI*DELZ
0111 0112 TK(I)=2.0*DELT/(UMASS*BMSS)
0113 0114 SL(I)=DELT/DELZ
0115 0116 GMAX=RHOI*(DELZ/DTT)**2
0117 0118 ER=TAU/GMAX
0119 0120 ER(I)=DELT/(DELZ*ER)
0121 0122 SK(I)=TAU
0123 0124 UMASS*BMSS
0125 0126 CALL MODEL(MAX+SIGY+SIGYP+CF*I)
0127 0128 WRITE(6,509) I*VOUT+DELZ+GMAX+TAU+ER+RHOI+Z+PVI+PVT1
0129 0130 GO TO 110
0131 0132 900 IF(ISW.EQ.0) GO TO 905
0133 0134 DELZ=ZLAST
0135 0136 ZADD=DELZ/(DTT/TUU+DTA)-I,0
0137 0138 Z=ZADD
0139 0140 PVTL=PVT1+ZADD*PTC1
0141 0142 PV1=PV1+ZADD*PCOF
0143 DELZ=DELZ*ZADD
0144 I=I+1
0145 0146 920 IF(I.GT.200) GO TO 920
0147 0148 IF(Z.LT.DSL(L)) GO TO 943
0149 0150 IS(L)=I
0151 0152 WRITE(6,508) L-IS(L)+1OPS(L)+DS(L)+ZLAST+Z
0153 0154 L=L+1
0155 0156 IF(L.GT.LIM) GO TO 944
0157 0158 944 DS(L)=L*OE+15
0159 0160 943 VOUT=DELZ/DTT
0161 0162 BMSS=RHOI*DELZ
0163 0164 TK(I)=2.0*DELT/(UMASS*BMSS)
0165 0166 SL(I)=DELT/DELZ
0167 0168 GMAX=RHOI*(DELZ/DTT)**2
0169 0170 ER=TAU/GMAX
0170 0171 ER(I)=DELT/(DELZ*ER)
0172 0173 SK(I)=TAU
0174 0175 UMASS*BMSS
0176 0177 CALL MODEL(MAX+SIGY+SIGYP+CF*I)
0178 0179 WRITE(6,509) I*VOUT+DELZ+GMAX+TAU+ER+RHOI+Z+PVI+PVT1
0180 0181 905 NI=I


```
FORTAN IV G LEVEL 21

0147           TK(NI+1)=2,0*DELT/UMASS
0148           IF(1,L0,LIM) GO TO 910
0149           DO 910 JL=L+LIM
0150     IS(JL)=NI
0151     IF (IOPS(JL),EQ,3) IS(JL)=NI+1
0152            ZTOP=Z-DELT
0153     IF (IOPS(JL),EQ,3) ZTOP=Z
0154  901 WRITE(6+508) JL,IS(JL),IOPS(JL),DS(JL),ZTOP,Z
0155  910 RETURN
0156  920 WRITE(6+520)
0157           STOP
0158           END
```
0001 C SUBROUTINE SIGMA(SIGMAX,SIGY,SIGYP)
0002
0003 REAL SIGY(50)
0004 MAX=50
0005 SIGY(1)=0.0
0006 DO 1 J=2,6
0007 1 SIGY(J)=0.025*(6-J)
0008 DO 2 J=7,44
0009 2 SIGY(J)=0.025*FLOAT(J-5)
0010 DO 3 J=45,50
0011 3 SIGY(J)=1.0-0.025*(J-44)
0012 SIGYP=1.0-0.025*0.5**7
0013 RETURN
0014 END
SURROUNTOE MODULE MAX, SIGY, SIGYP, CF, II
C
REAL SIGY(50), CF(50, 200)
MM = MAX = 1
EML = 0, 0
DO 1 J = 1, MM
EM = SIGY(J, 1) / (1.0 - SIGY(J, 1))
CF(J, 1) = (SIGY(J, 1) - SIGY(J)) / (EM - EML)
1 EM = EM
EM = SIGYP / (1.0 - SIGYP)
CF(MAX, I) = SIGYP - SIGY(MAX) / (EM - EML)
RETURN
END
SUBROUTINE FILTER(X1,F1,F2,TDF,NKF)

REAL X(1)
REAL Y(5000)
REAL P(1000)
P=3.14159
P0=(F1+F2)*TDF
W=2.0*PI*F1
W2=2.0*PI*F2
C=TDF/(2.0*PI)
Q=1.0/(2.0*(F2-F1))

DO 1=1,N
1=FLOAT(I)*TDF

DO 2=1,N
ADD=P*(F2-F1)

IF(YI.NE.0) ADD=SIN(P*I*Q)*TDF
ADD=C*SIN(QI+WZ)*Y/(2.0)*ADD

1 P(1)=C*SIN(QI+W1)+SIN(QI+W2)*Y/(1.0/T-0.5/(T+Q))
ADD

DO 2=1,N
2=1+N

IF(I+J,GT.NKF) GO TO 4
2 CUM=CUM+P(I)*XX(J)
3 CONTINUE

4 DO 5=1,N
5=1+N

IF(J+I,LT.1) GO TO 2
2 CUM=CUM+P(I)*XX(J)
3 CONTINUE

6 CUM=CUM+P(I)*XX(J)

RETURN
END
SUBROUTINE SMPLT(Y,NK,DELT,ALPHA)
 DUMMY PLOT SUBROUTINE
 REAL Y(5000)
 501 FORMAT((1E12.3))
 WRITE(6,501) Y(K),K=1,NK
 RETURN
 END
FORTRAN IV G LEVEL 21

SUBROUTINE PUNCH(Y,NK)

REAL Y(5000)

501 FORMAT(5E14.8,16)

J1=1

J2=5

NC=NK/5

DO 10 KC=1,NK

WRITE (3,501) (Y(J)+J1+J2)+KC

J1=J1+5

J2=J2+5

10 RETURN

END
SUBROUTINE XTRM(Y,NK)
REAL Y(5000)
501 FORMAT(5X*MAX=F12.4,1X*AT K=16/5X*MIN=F12.4,1X*)

0004 KMAX=0
0005 KMIN=0
0006 YM=0.0
0007 YMIN=0.0
0008 DO 1 K=1,NK
0009 IF(Y(K),GT,YMAX) KMAX=Y(K)
0010 IF(Y(K),GT,YMAX) YMAX=Y(K)
0011 IF(Y(K),LT,YMIN) KMIN=Y(K)
0012 IF(Y(K),LT,YMIN) YMIN=Y(K)
0013 1 CONTINUE
0014 WRITE(6+501) YM,KMAX,YMIN,KMIN
0015 RETURN
0016 END
SAMPLE PROBLEM

INPUT DECK
<table>
<thead>
<tr>
<th>SAMPLE PROBLEM 2</th>
<th>10.0</th>
<th>20.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>30C 0.01</td>
<td>1.0</td>
<td>2.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3</th>
<th>12.5E+02</th>
<th>12.5E+03</th>
<th>12.5E+04</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0.0E+00</td>
<td>0.0E+00</td>
<td>0.0E+00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>25J 0.02</th>
<th>2.0</th>
<th>0.0E+00</th>
<th>0.0E+00</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>E+00</td>
<td>-0.05</td>
<td>E+00</td>
</tr>
<tr>
<td>0.25</td>
<td>E+00</td>
<td>-0.30</td>
<td>E+00</td>
</tr>
<tr>
<td>0.50</td>
<td>E+00</td>
<td>-0.45</td>
<td>E+00</td>
</tr>
<tr>
<td>0.25</td>
<td>E+00</td>
<td>-0.20</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
<td>-0.05</td>
<td>E+00</td>
</tr>
</tbody>
</table>

| -0.25    | E+00 | -0.35  | E+00  |
| -0.50    | E+00 | -0.40  | E+00  |
| -0.25    | E+00 | -0.15  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |

| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |
| 0        | E+00 | -0.00  | E+00  |

### Input Cards

<table>
<thead>
<tr>
<th>Card</th>
<th>Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>50</td>
<td>60</td>
</tr>
<tr>
<td>70</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>10</td>
</tr>
<tr>
<td>---</td>
<td>----</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>F+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
<tr>
<td>0</td>
<td>E+00</td>
</tr>
</tbody>
</table>
SAMPLE PROBLEM 2

NK = 300
DEL = 0.01000
F1 = 10.00
F2 = 20.00
```
SUBROUTINE SECT

RHON= 2.600
VM= 0.2000E 06

L= 1
IOPS(L)= 1
DS(L)= 0.1250E 04, 0.1250E 04, 0.1250E 04

MORE THICK RHOI SCV PHREAT SAT IFPH GCFL XG TCF AT TAUC GMAX VIC
0 0.2500E 04 2.000 0.0 1.00 1.0000E 06 0.28 0.3300E 00 0.75 0.0 0.0

1 0.0563E 04 0.0563E 02 0.1467E 09 0.1990E 05 0.7494E-04 2.0000 0.0563E 02 0.8392E 05 0.1678E 06
2 0.1301E 05 0.1301E 03 0.3388E 09 0.4755E 05 0.1304E-03 2.0000 0.2158E 03 0.2115E 06 0.4229E 06
3 0.1547E 05 0.1547E 03 0.4788E 09 0.9396E 05 0.1962E-03 2.0000 0.3705E 03 0.3631E 06 0.7262E 06
4 0.2132E 05 0.2132E 03 0.6001E 09 0.1472E 06 0.2452E-03 2.0000 0.5437E 03 0.5328E 06 0.1066E 07
5 0.2015E 05 0.2015E 03 0.8118E 09 0.2689E 06 0.3312E-03 2.0000 0.9338E 03 0.9175E 06 0.1435E 07
6 0.2130E 05 0.2130E 03 0.9076E 09 0.3350E 06 0.3701E-03 2.0000 0.1147E 04 0.1124E 07 0.2247E 07

L= 2
IOPS(L)= 1
DS(L)= 0.1250E 04, ZTOP= 0.1147E 04, ZBASE= 0.1370E 04

L= 3
IOPS(L)= 1
DS(L)= 0.1250E 04, ZTOP= 0.1147E 04, ZBASE= 0.1370E 04

INPUT ACCELERATION

MAX= 0.5000E 00 AT K= 11
MIN= -0.5000E 00 AT K= 31

0.0 0.5000E-01 0.1000E 00 0.1500E 00 0.2000E 00 0.2500E 00 0.3000E 00 0.3500E 00 0.4000E 00 0.4500E 00
```

### Maximum Strain Profile

<table>
<thead>
<tr>
<th>MAXIMUM STRAIN PROFILE</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.125E-01</td>
</tr>
<tr>
<td>2</td>
<td>0.294E-02</td>
</tr>
<tr>
<td>3</td>
<td>0.328E-02</td>
</tr>
<tr>
<td>4</td>
<td>0.323E-02</td>
</tr>
<tr>
<td>5</td>
<td>0.338E-02</td>
</tr>
<tr>
<td>6</td>
<td>0.310E-02</td>
</tr>
<tr>
<td>7</td>
<td>0.242E-02</td>
</tr>
<tr>
<td>8</td>
<td>0.305E-02</td>
</tr>
<tr>
<td>9</td>
<td>0.233E-02</td>
</tr>
<tr>
<td>10</td>
<td>0.371E-02</td>
</tr>
<tr>
<td>11</td>
<td>0.431E-02</td>
</tr>
<tr>
<td>12</td>
<td>0.520E-02</td>
</tr>
<tr>
<td>13</td>
<td>0.620E-02</td>
</tr>
</tbody>
</table>

### Surface Velocity

<table>
<thead>
<tr>
<th>SURFACE VELOCITY</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX = 0.3703E 02 AT K = 48</td>
<td></td>
</tr>
<tr>
<td>MIN = -0.8119E 01 AT K = 88</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.892E-02</td>
<td>0.114E-01</td>
<td>0.115E-01</td>
<td>0.802E-02</td>
<td>0.271E-02</td>
</tr>
<tr>
<td>0.582E-01</td>
<td>0.687E-01</td>
<td>0.795E-01</td>
<td>0.907E-01</td>
<td>0.102E-01</td>
</tr>
<tr>
<td>0.173E-02</td>
<td>0.189E-02</td>
<td>0.210E-02</td>
<td>0.222E-02</td>
<td>0.235E-02</td>
</tr>
<tr>
<td>0.235E-02</td>
<td>0.223E-02</td>
<td>0.211E-02</td>
<td>0.199E-02</td>
<td>0.187E-02</td>
</tr>
<tr>
<td>0.113E-02</td>
<td>0.101E-02</td>
<td>0.881E-01</td>
<td>0.756E-01</td>
<td>0.629E-01</td>
</tr>
<tr>
<td>0.690E-01</td>
<td>0.640E-01</td>
<td>0.612E-01</td>
<td>0.577E-01</td>
<td>0.540E-01</td>
</tr>
<tr>
<td>0.275E-01</td>
<td>0.229E-01</td>
<td>0.181E-01</td>
<td>0.129E-01</td>
<td>0.745E-01</td>
</tr>
<tr>
<td>0.228E-01</td>
<td>0.270E-01</td>
<td>0.316E-01</td>
<td>0.364E-01</td>
<td>0.387E-01</td>
</tr>
<tr>
<td>0.274E-01</td>
<td>0.252E-01</td>
<td>0.229E-01</td>
<td>0.207E-01</td>
<td>0.184E-01</td>
</tr>
<tr>
<td>0.383E-01</td>
<td>0.328E-01</td>
<td>0.258E-01</td>
<td>0.194E-01</td>
<td>0.180E-01</td>
</tr>
<tr>
<td>0.256E-01</td>
<td>0.263E-01</td>
<td>0.265E-01</td>
<td>0.268E-01</td>
<td>0.264E-01</td>
</tr>
<tr>
<td>0.215E-01</td>
<td>0.258E-01</td>
<td>0.305E-01</td>
<td>0.129E-00</td>
<td>0.103E-00</td>
</tr>
<tr>
<td>0.144E-01</td>
<td>0.160E-01</td>
<td>0.172E-01</td>
<td>0.181E-01</td>
<td>0.187E-01</td>
</tr>
<tr>
<td>0.153E-01</td>
<td>0.137E-01</td>
<td>0.116E-01</td>
<td>0.907E-01</td>
<td>0.623E-01</td>
</tr>
<tr>
<td>0.139E-01</td>
<td>0.134E-01</td>
<td>0.126E-01</td>
<td>0.118E-01</td>
<td>0.104E-01</td>
</tr>
<tr>
<td>0.134E-02</td>
<td>0.221E-00</td>
<td>0.428E-00</td>
<td>0.601E-00</td>
<td>0.733E-00</td>
</tr>
<tr>
<td>0.139E-01</td>
<td>0.199E-01</td>
<td>0.108E-01</td>
<td>0.106E-01</td>
<td>0.102E-01</td>
</tr>
<tr>
<td>0.430E-00</td>
<td>0.293E-00</td>
<td>0.152E-00</td>
<td>0.875E-02</td>
<td>0.135E-02</td>
</tr>
<tr>
<td>0.876E-00</td>
<td>0.900E-00</td>
<td>0.899E-00</td>
<td>0.866E-00</td>
<td>0.870E-00</td>
</tr>
<tr>
<td>0.588E-00</td>
<td>0.487E-00</td>
<td>0.399E-00</td>
<td>0.292E-00</td>
<td>0.148E-00</td>
</tr>
<tr>
<td>0.584E-00</td>
<td>0.549E-00</td>
<td>0.633E-00</td>
<td>0.712E-00</td>
<td>0.770E-00</td>
</tr>
<tr>
<td>0.588E-00</td>
<td>0.531E-00</td>
<td>0.472E-00</td>
<td>0.403E-00</td>
<td>0.327E-00</td>
</tr>
<tr>
<td>0.137E-00</td>
<td>0.263E-00</td>
<td>0.340E-00</td>
<td>0.407E-00</td>
<td>0.464E-00</td>
</tr>
<tr>
<td>0.596E-00</td>
<td>0.593E-00</td>
<td>0.622E-00</td>
<td>0.639E-00</td>
<td></td>
</tr>
</tbody>
</table>

### Surface Acceleration

<table>
<thead>
<tr>
<th>SURFACE ACCELERATION</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX = 0.1423E 00 AT K = 44</td>
<td></td>
</tr>
<tr>
<td>MIN = -0.1401E 00 AT K = 84</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.946E-03</td>
<td>0.216E-03</td>
<td>0.127E-04</td>
<td>0.357E-03</td>
<td>0.541E-03</td>
</tr>
<tr>
<td>0.106E-02</td>
<td>0.202E-02</td>
<td>0.148E-02</td>
<td>0.108E-02</td>
<td></td>
</tr>
<tr>
<td>0.944E+00</td>
<td>0.103E+01</td>
<td>0.110E+01</td>
<td>0.115E+01</td>
<td>0.117E+01</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>0.882E+00</td>
<td>0.800E+00</td>
<td>0.707E+00</td>
<td>0.601E+00</td>
<td>0.486E+00</td>
</tr>
<tr>
<td>0.342E+00</td>
<td>0.406E+00</td>
<td>0.615E+00</td>
<td>0.722E+00</td>
<td>0.804E+00</td>
</tr>
<tr>
<td>0.892E+00</td>
<td>0.849E+00</td>
<td>0.789E+00</td>
<td>0.712E+00</td>
<td>0.528E+00</td>
</tr>
<tr>
<td>0.548E+01</td>
<td>0.452E+01</td>
<td>0.146E+00</td>
<td>0.245E+00</td>
<td>0.340E+00</td>
</tr>
<tr>
<td>0.751E+00</td>
<td>0.759E+00</td>
<td>0.766E+00</td>
<td>0.717E+00</td>
<td>0.677E+00</td>
</tr>
<tr>
<td>0.256E+00</td>
<td>0.163E+00</td>
<td>0.729E+00</td>
<td>0.109E+00</td>
<td>0.827E+00</td>
</tr>
<tr>
<td>0.467E+00</td>
<td>0.510E+00</td>
<td>0.546E+00</td>
<td>0.575E+00</td>
<td>0.597E+00</td>
</tr>
<tr>
<td>0.393E+00</td>
<td>0.393E+00</td>
<td>0.262E+00</td>
<td>0.193E+00</td>
<td>0.118E+00</td>
</tr>
<tr>
<td>0.286E+00</td>
<td>0.326E+00</td>
<td>0.361E+00</td>
<td>0.393E+00</td>
<td>0.420E+00</td>
</tr>
<tr>
<td>0.434E+00</td>
<td>0.408E+00</td>
<td>0.369E+00</td>
<td>0.318E+00</td>
<td>0.256E+00</td>
</tr>
<tr>
<td>0.115E+00</td>
<td>0.172E+00</td>
<td>0.226E+00</td>
<td>0.272E+00</td>
<td>0.308E+00</td>
</tr>
</tbody>
</table>