EVALUATION OF CONSTITUTIVE PROPERTIES FROM VELOCITY INTERFEROMETER DATA

D. E. Grady and E. G. Young

Prepared by Sandia Laboratories, Albuquerque, New Mexico 87115 and Livermore, California 94550 for the United States Energy Research and Development Administration under Contract AT (20-1)-789

Printed August 1976
EVALUATION OF CONSTITUTIVE PROPERTIES
FROM VELOCITY INTERFEROMETER DATA

D. E. Grady and E. G. Young
Geomechanics Research Division 5163
Sandia Laboratories
Albuquerque, New Mexico 87115

ABSTRACT

Accurate constitutive relations are required to predict the transient response of rock and soil media in a number of engineering and geophysical applications. Diffuse velocity interferometry has been specialized to this task. In this work, we report a method of analysis which accounts for the laser window-sample material impedance mismatch in determining material constitutive properties and describe two computer programs which have been developed to provide for rapid evaluation of these properties.
I. INTRODUCTION

There is currently a need for constitutive models which are capable of predicting the transient dynamic response of rock and soil media in a number of engineering and geophysical applications. Controlled plate impact experiments on samples of geological media provide one technique for determining the experimental data base necessary to develop adequate models. Attractive features of this approach are: (1) a controlled laboratory situation which allows measurement of the input loading pulse and subsequent evolution of the deformational stress wave and; (2) the fact that the experiment is dynamic, consistent with most of the application requirements.

Instrumentation capable of providing accurate data during a plate impact test has evolved over the past decade. At present, transducer techniques which measure stress-time or particle velocity-time histories have been the most successful. These include piezoresistant, piezoelectric, magnetic induction, and laser interferometry techniques. A notable advance has been the recent development of diffuse surface laser interferometry which does not have the severe optical reflectivity requirements of earlier laser systems. Development of this very accurate instrumentation capability has made feasible the study of transient wave propagation in rock and soil media, and several recent studies on rocks have been conducted in this laboratory. A particularly attractive feature has been the capability of making accurate relief wave studies which has been a limiting problem with other transducer techniques.

A critical problem in controlled impact studies has been the difficulty in relating the large amplitude transient wave propagation data to the stress-
strain constitutive properties and to the physical effects underlying the observed propagation characteristics. The measured stress or velocity profiles are only indirectly related to these properties and methods must be developed to effectively analyze the data. This problem has been considered by a number of authors. \textsuperscript{8,9,10,11}

A further difficulty occurs with velocity interferometer techniques where velocity is measured at an interface between the sample under test and a laser window material. Since the window material is never an exact impedance match for the test material, a region of wave interaction is created in the test material near the sample-window interface and the measured interface velocity is a distorted version of the profile originally propagating in the undisturbed test medium. This sample-window impedance mismatch also complicates the problem of relating the wave propagation data to the rock material properties.

The present effort was undertaken to develop a systematic method for reducing velocity interferometer data to determine constitutive rock properties. A computer program, VISAR, has been developed by Barker\textsuperscript{12} to determine particle velocity profiles from the raw interferometer fringe data. In the present work two additional programs, MODIFY and IMP have been developed to obtain loading and unloading stress-strain, stress-particle velocity, and material moduli data for the test material from the measured particle velocity profiles. These programs were developed for the PDP-10 computer system and make full use of the interactive graphics feature of this system.

In this report we first outline the experimental conditions upon which the method of analysis is based. We then provide the mathematical background and assumptions necessary to treat the problem of wave interaction.
due to the sample-window impedance mismatch. In many rock materials of interest, the dispersive and dissipative character of the medium makes it difficult to design experiments in which the loading wave and the overtaking relief wave do not interact. Wave attenuation, therefore, results and introduces special problems in the analysis. These problems are discussed in the succeeding section. In the remaining sections an operational description of the analysis programs is provided.
II. EXPERIMENTAL CONDITIONS

The experimental methods which have been employed in this laboratory to investigate the dynamic properties of rock and soil media under conditions of plate impact are illustrated in Fig. 1. A 10 cm diameter light gas gun is used to impel flat-nosed aluminum projectiles at the target media. Projectiles are faced with thin plates of mechanically well known impact material which in turn are backed with low impedance solid foam. Fused quartz has been the standard impact material in most studies. Impact on the target material provides a step input stress wave in the sample material followed by an unloading wave originating at the impact plate-solid foam interface. Projectile velocities, ranging from 0.01 to 1 km/s, can be controlled within about 2 percent and impact planarity can be maintained within approximately one milliradian. The projectile velocity is measured by offset pins to within 0.2 percent and impact time at center of impact is measured to within about 10 ns using an offset impact fiducial pin and a planarity measurement with coplanar flush pins.

A laser window material is mounted on the back surface of the sample specimen. Although fuzed quartz, plexiglas, and sapphire have been calibrated for laser window materials, only fused quartz has been used in the current study of rock properties to date. The window material surface which is to be placed in intimate contact with the sample surface is first vapor deposited with silver or aluminum to provide a diffusely reflecting surface. Upon emergence of the impact-produced stress wave at the sample-window interface, the particle velocity is continuously and accurately recorded through the complete history of loading and unloading with the diffuse surface velocity interferometer.
Fig. 1: EXPERIMENTAL CONFIGURATION FOR STUDY OF CONSTITUTIVE PROPERTIES OF ROCKS.
A series of tests on one rock material consisted of selecting a number of impact amplitudes within the range of interest and conducting several experiments at each impact amplitude. Sample thicknesses vary, usually ranging between 5 and 25 mm. By this procedure we can determine the evolution of the wave propagation which will be determined by the particular dispersive and dissipative properties of the rock under test. A set of particle velocity profiles obtained in such a series of tests on Solenhofen limestone are illustrated in Fig. 9 of this text.
III. ANALYTIC BASIS

In Fig. 2 an x-t diagram for a typical plate slap experiment using a window material is shown. The test material is impacted from the left producing a simple loading stress wave propagating to the right in the sample material and to the left in the impactor material. The wave in the impactor material reflects off of the foam-impactor interface producing a right-going unloading wave which enters the sample material at the impact interface. The impactor plate, therefore, produces a complete loading and unloading stress pulse which propagates to the right in the test material and emerges at the sample-window interface where the resulting velocity profile is measured with laser interferometry. As was mentioned in the previous section, experiments using several sample thicknesses are conducted so the propagation velocities and dispersion characteristics of the wave pulse can be determined. Since the window material is never an exact impedance match for the test material, a region of wave interaction is created in the sample to the left of the sample-window interface and the measured velocity profile is a distorted version of the input profile, depending on the coupled mechanical properties of the window material and the unknown sample material. The objective is to subtract out properties of the window material so that the constitutive properties of the test material can be determined from the experimental wave profiles.

The method we have used to account for the interface mismatch has been to correct the distorted profiles and obtain the profiles which would have been measured at the sample-window interface if no impedance mismatch occurred. These profiles are then used to determine the stress-strain and stress-particle velocity properties of the test material. This
Fig. 2: DISTANCE-TIME PLOT
Illustrates the initial step input loading of the sample on impact and interaction regions due to impedance differences between sample and window material.
approach is based on invariance of the Riemann integrals in regions of wave interaction and, hence, precludes the analysis of experimental profiles which exhibit highly rate dependent material behavior. This is not a critical limitation, however, to a large class of geological and engineering materials where working constitutive relations are required. When significant rate dependence occurs, it is recognized as curvature in the distance-time characteristics of constant particle velocity amplitude.

Wave propagation in the unknown sample material is governed by the equations of continuity and momentum. In Lagrangian coordinates they are:

\[
\frac{\partial \varepsilon}{\partial t} + \frac{\partial u}{\partial h} = 0 \tag{1}
\]

\[
\frac{\partial u}{\partial t} + \frac{1}{\rho_0} \frac{\partial \sigma}{\partial h} = 0 \tag{2}
\]

where  \( \varepsilon = 1 - \rho_0/\rho \). \hspace{1cm} (3)

The time coordinate and Lagrangian space coordinate are \( t \) and \( h \), respectively, and \( \varepsilon, u, \) and \( \sigma \) are the strain, particle velocity, and stress referred to the direction of wave propagation. \( \rho \) and \( \rho_0 \) are the present and initial densities. Equations 1 and 2 can be rewritten as

\[
\frac{\partial \varepsilon}{\partial t} + \frac{\partial u}{\partial h} = 0 \tag{4}
\]

\[
\frac{\partial u}{\partial t} + c^2 \frac{\partial \varepsilon}{\partial h} = 0 \tag{5}
\]

where

\[
c^2 = \frac{1}{\rho_0} \frac{\partial \sigma}{\partial h} \tag{6}
\]
By adding and subtracting, we can put Eqs. 4 and 5 into the following form.

\[
\frac{\partial u}{\partial \alpha} + c \frac{\partial e}{\partial \alpha} = 0 \tag{7}
\]

\[
\frac{\partial u}{\partial \beta} - c \frac{\partial e}{\partial \beta} = 0 \tag{8}
\]

\[
\frac{\partial x}{\partial \alpha} - c \frac{\partial t}{\partial \alpha} = 0 \tag{9}
\]

\[
\frac{\partial x}{\partial \beta} + c \frac{\partial t}{\partial \beta} = 0 \tag{10}
\]

Equations 9 and 10 define families of right- and left-facing characteristics, \(C_+\) and \(C_-\) and \(\alpha\) and \(\beta\) are new independent variables referred to these characteristic coordinates. Equations 7 and 8 can be integrated to obtain the Riemann invariants

\[
J_+(\beta) = u + f(\sigma) \tag{11}
\]

and

\[
J_-(\sigma) = u - f(\sigma) \tag{12}
\]

where

\[
f(\sigma) = \int \frac{d\sigma}{\rho_c c} \tag{13}
\]

is a function, to be determined, which described the stress-strain response of the unknown sample material. Continuity of stress and particle velocity require that, at the sample-window interface,

\[
\frac{1}{2} (J_+ + J_-) = U(t) \tag{14}
\]

and

\[
\frac{1}{2} (J_+ - J_-) = f(P(t)) \tag{15}
\]
where \( U(t) \) is the measured particle velocity profile and \( P(t) \) is the stress profile determined from the known properties of the window material (for instance, a nonlinear elastic equation of state for fused silica). Solving Eqs. 14 and 15 for the Riemann invariants, results in

\[
J_+ = U(t) + f(P(t)),
\]

and

\[
J_- = U(t) - f(P(t)).
\]

The \( x-t \) diagram in Fig. 2 shows that the propagating stress wave is originally a simple wave (all wave information is carried on \( C_+ \) characteristics) which enters an interaction region caused by reflection at the sample-window interface. Due to invariance, the same value of \( J_+ \) in the simple wave region is carried into the interaction wave region. Therefore, if no interface were present the Riemann invariants would be

\[
J_+ = U(t) + f(P(t))
\]

\[
J_- = 0.
\]

Equations 18 and 19, with the definition of the Riemann invariants in Eq. 11 and 12 provide the in-material particle velocity profile if no mismatched interface were present,

\[
u(t) = \frac{1}{2} \left[ U(t) + f(P(t)) \right].
\]

Neglected in this analysis is the slight refraction of \( J_+ \) characteristics when they enter the interaction region which causes slight arrival time errors upon reaching the sample-window interface. These time errors will be small if propagation time in the region of interaction is maintained small compared to propagation time in the noninteraction region, and if
the impedance mismatch is kept small so that the bending of characteristics is minimized. Also, since the refraction of characteristics will be in the same direction for experiments using different sample thicknesses, the errors in time of arrival used to calculate wave velocities will tend to cancel.

A geometric interpretation of equation (20) is helpful in visualizing the impedance mismatch correction. In Fig. 3 we show a representative loading response curve for the window material and the sample material in the pressure-particle velocity plane. Also shown is the mirror image of the sample material loading curve about a point \( u(t) \) on the particle velocity axis. Referring to Eq. (20) the quantities \( U(t), P(t), f(P(t)) \), and \( u(t) \) are shown in the figure. A similar geometric representation can be drawn for the unloading wave.

It is necessary to clarify one further point concerning Fig. 3. We see that it conforms with a familiar interpretation of a wave of amplitude \( u(t) \) "unloading" to an amplitude \( U(t) \) upon encountering the lower impedance window material. The concept of unloading, however, is misleading in the case of a continuous loading wave arriving at the sample-window interface. In the case of a hysteretic material (loading and unloading response paths are different), it is important to note that it is the loading and not unloading curve that is used as the reflected curve in Fig. 3. This geometric procedure is then consistent, as it must be, with Eq. (20).

Eq. (20) provides the analytic expression for correcting the experimental profiles for impedance mismatch of the window material. The function \( f(P(t)) \) in Eq. (20) or, equivalently, the response curve for the sample material in Fig. 2 is not known before the fact. Since
Fig. 3: GEOMETRIC REPRESENTATION OF THE SAMPLE-WINDOW IMPEDANCE MISMATCH CORRECTION.
we see that knowledge of the functional dependence of the wave velocity within the integral in Eq. 21 is equivalent to knowing \( f(\sigma) \). The procedure followed is to use the multiple uncorrected particle velocity profiles obtained at different sample thicknesses to provide an estimate of the functional dependence of the Lagrangian wave velocity on particle velocity. Eq. (20) is then used to provide an estimate for the corrected particle velocity profiles (those occurring if no interface were present) which in turn provides a better estimate of the functional dependence of the wave velocity. By this iterative procedure, the value of \( f(\sigma) \) or equivalently the \( \sigma-u \) curve for the unknown sample material is determined. Having this, the stress-strain paths and longitudinal modulus for material loading and unloading readily follow.

In practice, it has been found that no iterating was necessary to obtain satisfactory results in the materials studied to this date. Further iteration provides improvement only if the mesh size is sufficiently fine. To check the procedure, a test problem for a sample material with a variable impedance differing from fused silica window material by about 25 percent on loading and by about 100 percent on unloading was developed. A total of 28 mesh points were used to describe the total loading and unloading profile. The analysis scheme reproduced the correct in-material particle velocity profiles within about 0.5 percent on loading and about 2.5 percent on unloading after one iteration.
IV. ATTENUATING WAVE

When a one-dimensional loading stress wave propagates into a region of uniform state, as in the situation of a typical plate impact experiment, the wave velocity at any amplitude on the loading wave relates directly to the loading stress-strain modulus through the relation

\[
\frac{d\sigma}{d\epsilon} = \rho_0 c^2(\sigma) .
\]

(22)

Similarly, if an overtaking relief wave propagates into the uniform state remaining after passage of the loading wave then the relief wave velocity at any amplitude is related to the unloading stress-strain modulus through the same relation (Eq. 22). If, however, the unloading wave overtakes the loading wave and attenuation of the wave peak occurs then wave velocities determined from the unloading wave do not provide the unloading stress-strain modulus as in Eq. 22. Care must be exercised in extracting unloading stress-strain data from wave velocities determined from attenuating wave propagation data.

We have found attenuating wave data difficult to circumvent in the study of some rock types. A good example is Solenhofen limestone\(^7\) (See Fig. 9) which has an extremely slow loading wave velocity above the axial failure stress allowing rapid overtaking of the unloading wave and subsequent attenuation of the peak.

The approach we have taken to extract the unloading stress-strain response from attenuating data has been to examine the equations governing attenuating wave propagation and determine, quantitatively, how significantly the measured unloading wave velocities differ from Eq. 22. We have found a

*This observation is strictly true only if rate dependent effects are not present. The experimental situations in which rate dependence will adversely effect the results are limited, however, and will not be considered here.
relation for this difference which depends on the rate of attenuation of the peak and the slope of the wave in front of and in back of the peak. We have found from this analysis that in a majority of experimental situations the unloading wave velocity does not differ significantly from Eq. 22, and therefore, analysis methods valid for nonattenuating wave propagation can be used with little error on attenuating data. The results of that calculation follow.

In Fig. 4 we depict a set of particle velocity histories which might be obtained in a typical experiment where wave attenuation has occurred. Let the Lagrangian distance and time coordinates be $h$ and $t$, respectively, and $u$ the particle velocity. The velocity of the wave peak will be denoted by $D$ and quantities immediately in front of and immediately in back of the peak will be denoted by a subscript 1 or a subscript 2, respectively.

We ultimately wish to relate the unloading wave velocity at a given amplitude to the unloading stress-strain modulus

$$\frac{d\sigma}{d\varepsilon} = \frac{\sigma_t}{\varepsilon_t} = \rho_o c^2. \quad (23)$$

From the equation of continuity

$$\varepsilon_t = - u_t = \frac{1}{D} (u_t - \dot{u}). \quad (24)$$

The respective partial derivatives are expressed by subscript $t$ or $h$ and the dot refers to a directional time derivative along the path of the attenuating peak. The equation of motion gives

$$\sigma_h = - \rho_0 u_t \quad (25)$$

and since

$$\sigma_t = \dot{\sigma} - D\varepsilon_t \quad (26)$$
Fig. 4: ATTENUATING WAVE PROPAGATION.
we obtain

$$\sigma_t = \dot{\sigma} + \rho_0 D u_t$$  \hspace{1cm} (27)

Equations 23, 24 and 27 provide

$$\rho_0 c^2 = \frac{D(\dot{\sigma} + \rho_0 D u_t)}{(u_t - \dot{u})}$$  \hspace{1cm} (28)

Since Eq. 28 applies either in front of or behind the peak, we obtain

the relations

$$\rho_0 c_1^2 (u_{t1} - \dot{u}) - \rho_0 D^2 u_{t1} = D \dot{\sigma}$$ \hspace{1cm} (29)

$$\rho_0 c_2^2 (u_{t2} - \dot{u}) - \rho_0 D^2 u_{t2} = D \dot{\sigma}$$ \hspace{1cm} (30)

where \( \rho_0 c_1^2 \) and \( \rho_0 c_2^2 \) are the slope of the stress-strain curve immediately
in front of and immediately behind the peak. Similarly, \( u_{t1} \) and \( u_{t2} \) are
slopes of the particle velocity profile on opposite sides of the peak.

To eliminate \( \sigma \) it is necessary to restrict the form of the loading
wave. We will make a simple wave assumption

$$\ddot{\sigma} = \rho_0 c_1 (u) \dot{u}$$ \hspace{1cm} (31)

which includes as limiting cases shock loading, \( \ddot{\sigma} = \rho_0 D \dot{u} \) or a steady wave,
\( \ddot{\sigma} = \rho_0 c \dot{u} \), where \( c = \) constant.

Substitute Eq. 31 into Eq. 29 and 30 and define \( \alpha = \dot{u}/u_{t1} \) and
\( \beta = \dot{u}/u_{t2} \) which are ratios of the rate of profile decay in front of and
behind the peak. The resulting equations are

$$c_1^2 (1 - \alpha) - D^2 = D c_1 \alpha$$  \hspace{1cm} (32)

$$c_1^2 (1 - \beta) - D^2 = D c_1 \beta$$  \hspace{1cm} (33)
In region two we will define an unloading wave velocity
\[ c_u = \left( \frac{\partial h}{\partial t} \right)_u \]  
(34)
which is the experimentally measured wave velocity at particle velocity amplitude \( u \). The purpose of this calculation is to determine how \( c_u \) differs from \( c_2 \) which in turn is identified with the unloading stress-strain modulus through Eq. 22. In region 2 (at the peak) we obtain the relation
\[ c_u = D/(1 - \beta) \]  
(35)
which is used to eliminate \( D \) from Eqs. 32 and 33. Let \( x = c_2/c_u \) and eliminating \( c_1 \) from the resulting expression provides, after some algebra, the relation
\[ x^2 = (1 - \beta) \left[ 1 + \frac{\beta}{1 - \alpha} \right] \]  
(36)
when there is no attenuation \( x = 1 \) (\( c_u = c_2 \)) and the measured wave velocity identically determines the stress-strain modulus. When attenuation occurs \( \alpha \) and \( \beta \) can be determined from the rate of attenuation and the local slopes of the profiles in front of and behind the peak and Eq. 36 can be used to determine the error in assuming that the measured wave velocity determines the stress-strain modulus. In Fig. 5, Eq. 36 is displayed graphically. The curves shown correspond to 1 percent, 2 percent, and 3 percent errors, respectively. Also shown is the point corresponding to the Solenhofen limestone data presented in Fig. 9 which places the order of error expected in perspective.
Fig. 5: PLOT OF EQUATION 36 AT CONSTANT VALUES OF X.

The ordinate and abcissa are the relative slopes of the loading and unloading waves to the slope of the peak attenuation. The point shown corresponds to the Solenhofen limestone data plotted in Fig. 9.
V. OPERATIONAL DESCRIPTION

For each individual experiment, program "VISAR"\(^{12}\) will provide a particle velocity-time history from the raw interference fringe records. The profile shape is determined by the impact velocity, impactor thickness, constitutive properties of the test medium, and impedance differences between the test sample and the window material. The profiles from a series of experiments are used to determine the constitutive behavior of the test material. Programs "MODIFY" and "IMP" have been developed to assist in this analysis.

Program "MODIFY" Analysis begins by taking the velocity profiles from the same series (similar impact velocities and impactor thicknesses but different sample thicknesses) and plotting them on the same particle velocity-time graph. This can be done using program "PLOTPL" which is available on the PDPI0, and provides a qualitative picture of the propagation properties of the wave profiles. Since profiles were obtained from individual experiments, there are obvious inconsistencies due to slight differences in impact velocities and impactor thicknesses. Adjustment of the amplitudes and unloading wave times to account for these slight differences is required in order to make the profiles from a given series compatible. Also, individual profiles may have irregularities due to superpose time marks, triggering noise or other sources which would be desirable to eliminate. Program "MODIFY" was constructed to account for these problems.

When "MODIFY" is executed (EX@MOD) the user is asked the input file name and then the desired output or modified file name. If the user is sufficiently confident of the adjustments required, the new file may be read into the old file name. This will irreversibly kill the old file,
however. The input profile is then automatically plotted and provides
the user with a working plot of the velocity-time history.

The user is then asked to distinguish the point in time separating the
loading wave from the unloading wave ("DO YOU WISH TO MARK THE PEAK?").
This has two purposes. First, if adjustment of the unloading wave due
to different impactor thicknesses is desired this separation time is
necessary. Second, this time is required in program IMP. When the user
responds with yes ("y") he or she is given the option to replot. This
can be done on a magnified scale so that the time of the peak can be more
closely identified. The user is then requested to identify the time
("ENTER TIME OF PEAK").

The next operation allows adjustment of the data in time and amplitude.
The user is asked "DO YOU WISH TO SCALE OR TRANSLATE THE DATA." An affir-
mative response is immediately followed by "DO YOU WISH TO TRANSLATE IN
TIME?" If yes, then the user is asked whether they wish to shift the entire
profile or the unloading wave only. The former provides a rigid trans-
lation of the profile while the latter shifts only the unloading wave.
The user then has an opportunity to scale the profile in time by some con-
stant scale factor. This operation is used primarily to compare profiles
for self similar flow where the profile is some function of the distance
divided by the time. Finally, the user is asked "DO YOU WISH TO SCALE IN
VELOCITY?" This operation allows the user to adjust the profile amplitude
for differences in impact velocity.

In the next operation the user is given a chance to alter the profile
("ALTER ANY PORTION OF THE PROFILE?"). If yes, the user is allowed to re-
plot and focus on any portion of the profile. The user may then select
any time interval of the data file to be eliminated and then specify the
points to replace the eliminated data. The process is repeated until the user has no further alterations to perform.

Lastly, the user is given an opportunity to plot both the old and the new data file on the same velocity-time plot. The old data appears as a solid line while the new data is plotted as a dashed line. This provides a check on the adjustments and alterations made on the data file. This comparison cannot be made if the modified file has been given the same as the input file.

Program "IMP" After a set of particle velocity profiles from any given series have been satisfactorily correlated using "MODIFY," program "IMP" is executed (EX@IMP). This program is intended to evaluate the constitutive properties of the test material (stress-strain, stress-particle velocity, and wave velocity-stress behavior) and account for the impedance difference between the sample material and the laser window material as was discussed in Section III. At execution, the user is asked the number of input files and their names. Two to five files can presently be analyzed with the program. Following this, the size of the velocity increment must be input. We have found that about 20 to 30 increments from wave foot to peak convenient and sufficient and the increment is selected accordingly. The program then asks whether the user desires a plot of the incremented velocity profiles. An example of such a plot is provided in Fig. 6. If this is the first program run such a plot is necessary as a working plot for subsequent user interaction operations. The user is then asked to supply sample thicknesses in millimeters corresponding to the input profiles in order (increasing sample thickness). Finally, the user is given the opportunity to adjust timing of the profiles such that initial breakaway of the wave (foot of wave) corresponds to a prescribed velocity. This is convenient if a break-
Fig. 6: INCREMENTED PARTICLE VELOCITY PROFILES.

Selected increment size is 0.02 km/s. Data was obtained on Blair dolomite.
away velocity has been determined from a number of experiments and it is desired to have all profiles consistent with this velocity. The program then proceeds with the incremental velocity plot shown in Fig. 6.

The next program section provides distance-time plots for data evaluation and possible adjustment. The user is asked which characteristics he wishes to plot. Response is governed by a systematic numbering of the particle velocity levels. Referring to Fig. 6 the zero velocity foot of the wave is numbered one. Each consecutive velocity level through the full profile (loading and unloading) is numbered with consecutive integers. A response of '5', for example, would plot the first, sixth, eleventh, and so forth characteristics. An irregularity occurs if the wave attenuates with propagation distance as occurs in the example in Fig. 6. In such a case the level count up the loading wave proceeds as long as any two profiles can contribute to that characteristic. Failing this, the count switches to the unloading wave and continues.

The user is then asked to provide the minimum and maximum time desired in the plot. This can be estimated from the velocity plot in Fig. 6. The distance axis is determined automatically from the sample dimensions entered. A distance-time plot corresponding to the velocity data in Fig. 6 is shown in Fig. 7. The small circles indicate the exact distance-time point for a given profile and particle velocity level. The line is a best linear fit to equivalent particle velocity level points on different contributing profiles. The user is then provided an opportunity to add or delete characteristics, adjust timing in terms of a rigid translation in time of any profile, and reploting. This process is iterated until the user is satisfied with the distance-time properties of the series.
Fig. 7: DISTANCE-TIME PLOT CORRESPONDING TO INCREMENTED PROFILES DISPLAYED IN FIG. 6.
The user may then plot the wave velocity data determined from the linear fit to the incremented profile data. This plot is shown in Fig. 8. Note that there is a one-to-one correspondence between the points in the wave velocity-particle velocity plot in Fig. 8 and the particle velocity-time plot in Fig. 6. Adjustment of the wave velocity may be made at this point, a feature which is particularly valuable in assessing the effect of velocity variations on the ultimate constitutive properties of the material. Adjustment may be made in two ways: (1) The user may adjust the velocity corresponding to particular particle velocity level points on loading or unloading. The numbering scheme is the same as that incorporated for the characteristics in the distance-time plot. Or, (2) the user may adjust the total velocity spectrum in order to bring a given level point into correspondence with a desired wave velocity. This acts as a rigid translation of the wave profiles to bring about the desired correspondence.

At this point the user must enter the initial sample density (g/cc) and then the program uses the final wave velocity-particle velocity data to determine, and account for, the impedance difference between the sample material and the window material according to the methods discussed in Section III and to provide the final constitutive properties for the sample material. In Fig. 9 measured particle velocity profiles for Solenhofen limestone are shown and compared with profiles corrected with program IMP for the limestone-fused silica interface mismatch. The user may, at this point, print out in tabular form the incremental stress, strain, particle velocity and wave velocity data for both loading and unloading. The user may also choose to write a data file for the stress-strain, stress-particle velocity,
Fig. 8: WAVE VELOCITY VALUES CORRESPONDING TO THE INCREMENTED PROFILES IN FIG. 6.

Lower values are for the loading wave; upper values for the unloading wave. Note the correspondence between points in Fig. 6 and Fig. 8.
Fig. 9: PARTICLE VELOCITY PROFILES IN SOLENHOFEN LIMESTONE.

Solid lines are the uncorrected profiles. Dashed lines represent the in-material profiles corrected for the limestone-fused silica interface impedance mismatch.
or wave velocity-stress data. This is convenient if, for example, a combined plot of the stress-strain data from several series is desired. As mentioned earlier, program "PLOTPL" may be used for this purpose. Finally, the user is given an opportunity to make immediate plots of the constitutive data.

Program "IMP" has been found convenient for rapid evaluation of the constitutive properties of a large number of experiments. It has also proved useful where questions of how errors in timing and amplitude in the original profiles effect the final constitutive properties since such variations may be introduced in the initial profiles or, at some other point and the resulting variation observed in the final output. The program was developed with velocity interferometer data from a sample-window interface configuration. It could be extended for use in other experimental configurations without undue difficulty.
APPENDIX I

Listing of Program MODIFY
PROGRAM MODIFY

DIMENSION X(500), Y(500), LABX(25), LABY(25), LTITLE(25)

CALL INITT(3)
CALL TERM(1, 1024)

CALL BINIT
CALL ANMODE

C PLOT FILE -- CARTESIAN COORDINATES ONLY

14 TYPE 14
FORMAT(' INPUT FILENAME 'S).
ACCEPT 160, IFIL

123 TYPE 25, IFIL
FORMAT( a5)

25 TYPE 25, IFIL
FORMAT( a5) FOR MODIFIED OR CORRECTED 'A5', 'B5', 'S')
ACCEPT 160, MOUT

CALL NEWPAG

C TEK SUBPROGRAMS FOR PLOTTING EITHER REQUIRE MUCH STORAGE OR AN
C EXTRA-READ OF THE FILES TO PREDETERMINE PLOTTING PARAMETERS,
C THE LATTER WAS CHOSEN IN THIS PROGRAM.

C

C PRESET GLOBAL MAXIMA AND MINIMA FOR ALL X AND Y ARRAYS
C FROM ALL FILES,

CALL PRESET( XMIN, XMAX, YMIN, YMAX)
CALL READ( IFIL, X, Y)
CALL MINMAX( X, XMIN, XMAX)
CALL MINMAX( Y, YMIN, YMAX)

C PLOT UNALTERED DATA

18 CALL PLSCAL( X, Y, IFIL, 1, XMIN, XMAX, YMIN, YMAX, 'Y', 1)
CALL TINPUT( M)
CALL NEWPAG
CALL ANMODE

1058 TYPE 150

150 FORMAT( ' DO YOU WISH TO MARK THE PROFILE PEAK? ' S )

C DETERMINE PROFILE PEAK

ACCEPT 21, LYES
"ARPEK#0"
IF( LYES.EQ. 'N' ) GO TO 199
"ARPEK#1"

151 TYPE 152

152 FORMAT( ' REPLOT? ' S)

ACCEPT 21, LYES
IF( LYES.EQ. 'N' ) GO TO 156

153 FORMAT( ' ENTER XMIN, XMAX, YMIN, YMAX ' )
ACCEPT 154, YMIN, YMAX, XMIN, XMAX

154 FORMAT( 4G)
CALL BINIT
CALL NEWPAG
CALL PLSCAL( X, Y, IFIL, 1, XMIN, XMAX, YMIN, YMAX, 'Y', 1)
CALL TINPUT( M)
CALL NEWPAG
CALL ANMODE

156 TYPE 157

157 FORMAT( ' ENTER TIME OF PEAK ' S)
ACCEPT 158, TPEAK
FORMAT(G)
CALL READ(1,FILE,X,Y)
READ=Y

I = X(1)
GO TO 160 I = 2, N

IF (X(I) = TPEAK)
IF (DIFGE, E, 2) GO TO 162
CONTINUE

160 UNLOR=N+1
NL=1
GO 164, J = 1, MUNLOR
V(N+2-J) = Y(N+1-J)
X(N+2-J) = X(N+1-J)
CONTINUE

X(I) = N
Y(I) = N

199 TYPE 40
FORMAT(' DO YOU WISH TO SCALE OR TRANSLATE THE PROFILE? ' )
ACCEPT 21,NYES
IF (NYES.EQ..N1) GO TO 169
I = X(1)
TYPE 26
26 FORMAT(' DO YOU WISH TO SCALE IN TIME? ' )
ACCEPT 21,NYES
IF (NYES.EQ..N1) GO TO 229
TYPE 202
229 FORMAT(' ENTIRE PROFILE OR UNLOADING ONLY? ' )
TYPE 206
206 FORMAT(' (ANSWER "ENT" OR "UNL" ) ' )
ACCEPT 203,PSHIFT

233 FORMAT(A3)
TYPE 204
234 FORMAT(' ENTER TIME SHIFT ' )
ACCEPT 28,XTS
28 FORMAT(G)
IF (PSHIFT,EQ..UNL') GO TO 207

207 CONTINUE
X(I) = X(1)*XTS
GO TO 220
208, IF (X(I) EQ..X(1)) GO TO 240
TYPE 241
241 FORMAT( ' YOU FORGOT TO MARK THE PEAK! ' )
GO TO 1608

242 MNL = NNL+1
GO 208 I = MNL+1, N
208 X(I) = X(1)*XTS
GO 209 I = 2, NNL
IF (X(I) .GE..X(NNL1)) GO TO 210
CONTINUE

X(NNL) = X(NNL1)
GO TO 220

210 CONTINUE
GO 215 I = IL, NNL
GO 214 J = NNL1, N
IF (X(J), LT, X(I)) GO TO 214
IF (Y(J), LT, Y(I)) GO TO 216
GO TO 215
CONTINUE
215 CONTINUE
216 X(J+1) = X(J)
    Y(J+1) = Y(J)
217 CONTINUE
    X(J) = X(J+1)
    Y(J) = Y(J+1)
    J = J + 1
221 IF (J > N) GO TO 222
222 CONTINUE
223 CONTINUE
224 CONTINUE
225 CONTINUE
226 FORMAT (\"DO YOU WISH TO SCALE IN TIME? \"\")
    ACCEPT 21, YES
    IF (YES, EQ, \"N\") GO TO 222
227 CONTINUE
228 CONTINUE
229 CONTINUE
230 FORMAT (\"DO YOU WISH TO SCALE IN VELOCITY? \"\")
    ACCEPT 21, YES
    IF (YES, EQ, \"N\") GO TO 169
231 CONTINUE
232 CONTINUE
233 CONTINUE
234 CONTINUE
235 CONTINUE
236 CONTINUE
237 CONTINUE
238 CONTINUE
239 CONTINUE
240 CONTINUE
241 CONTINUE
242 CONTINUE
243 CONTINUE
244 CONTINUE
245 CONTINUE
246 CONTINUE
247 CONTINUE
248 CONTINUE
249 CONTINUE
250 CONTINUE
251 CONTINUE
252 CONTINUE
253 CONTINUE
254 CONTINUE
255 CONTINUE
256 CONTINUE
257 CONTINUE
258 CONTINUE
259 CONTINUE
260 CONTINUE
261 CONTINUE
262 CONTINUE
263 CONTINUE
264 CONTINUE
265 CONTINUE
266 CONTINUE
267 CONTINUE
268 CONTINUE
269 CONTINUE
270 CONTINUE
271 CONTINUE
272 CONTINUE
273 CONTINUE
274 CONTINUE
275 CONTINUE
276 CONTINUE
277 CONTINUE
278 CONTINUE
279 CONTINUE
280 CONTINUE
281 CONTINUE
282 CONTINUE
283 CONTINUE
284 CONTINUE
285 CONTINUE
286 CONTINUE
287 CONTINUE
288 CONTINUE
289 CONTINUE
290 CONTINUE
291 CONTINUE
292 CONTINUE
293 CONTINUE
294 CONTINUE
295 CONTINUE
296 CONTINUE
297 CONTINUE
298 CONTINUE
299 CONTINUE
300 CONTINUE
301 CONTINUE
302 CONTINUE
303 CONTINUE
304 CONTINUE
305 CONTINUE
306 CONTINUE
307 CONTINUE
308 CONTINUE
309 CONTINUE
310 CONTINUE
311 CONTINUE
312 CONTINUE
313 CONTINUE
314 CONTINUE
315 CONTINUE
316 CONTINUE
317 CONTINUE
318 CONTINUE
319 CONTINUE
320 CONTINUE
321 CONTINUE
322 CONTINUE
323 CONTINUE
324 CONTINUE
325 CONTINUE
326 CONTINUE
327 CONTINUE
328 CONTINUE
329 CONTINUE
330 CONTINUE
331 CONTINUE
332 CONTINUE
333 CONTINUE
334 CONTINUE
335 CONTINUE
336 CONTINUE
337 CONTINUE
338 CONTINUE
339 CONTINUE
340 CONTINUE
341 CONTINUE
342 CONTINUE
343 CONTINUE
344 CONTINUE
345 CONTINUE
346 CONTINUE
347 CONTINUE
348 CONTINUE
349 CONTINUE
350 CONTINUE
351 CONTINUE
352 CONTINUE
353 CONTINUE
354 CONTINUE
355 CONTINUE
356 CONTINUE
357 CONTINUE
358 CONTINUE
359 CONTINUE
360 CONTINUE
361 CONTINUE
362 CONTINUE
363 CONTINUE
364 CONTINUE
365 CONTINUE
366 CONTINUE
367 CONTINUE
368 CONTINUE
369 CONTINUE
370 CONTINUE
371 CONTINUE
372 CONTINUE
373 CONTINUE
374 CONTINUE
375 CONTINUE
376 CONTINUE
377 CONTINUE
378 CONTINUE
379 CONTINUE
380 CONTINUE
381 CONTINUE
382 CONTINUE
383 CONTINUE
384 CONTINUE
385 CONTINUE
386 CONTINUE
387 CONTINUE
388 CONTINUE
389 CONTINUE
390 CONTINUE
391 CONTINUE
392 CONTINUE
393 CONTINUE
394 CONTINUE
395 CONTINUE
396 CONTINUE
397 CONTINUE
398 CONTINUE
399 CONTINUE
400 CONTINUE
401 CONTINUE
402 CONTINUE
403 CONTINUE
404 CONTINUE
405 CONTINUE
406 CONTINUE
407 CONTINUE
408 CONTINUE
409 CONTINUE
410 CONTINUE
411 CONTINUE
412 CONTINUE
413 CONTINUE
414 CONTINUE
415 CONTINUE
416 CONTINUE
417 CONTINUE
418 CONTINUE
419 CONTINUE
420 CONTINUE
421 CONTINUE
422 CONTINUE
423 CONTINUE
424 CONTINUE
425 CONTINUE
426 CONTINUE
427 CONTINUE
428 CONTINUE
429 CONTINUE
430 CONTINUE
431 CONTINUE
432 CONTINUE
433 CONTINUE
434 CONTINUE
435 CONTINUE
436 CONTINUE
437 CONTINUE
438 CONTINUE
439 CONTINUE
440 CONTINUE
441 CONTINUE
442 CONTINUE
443 CONTINUE
444 CONTINUE
445 CONTINUE
446 CONTINUE
447 CONTINUE
448 CONTINUE
449 CONTINUE
450 CONTINUE
451 CONTINUE
452 CONTINUE
453 CONTINUE
454 CONTINUE
455 CONTINUE
456 CONTINUE
457 CONTINUE
458 CONTINUE
459 CONTINUE
460 CONTINUE
461 CONTINUE
462 CONTINUE
463 CONTINUE
464 CONTINUE
465 CONTINUE
466 CONTINUE
467 CONTINUE
468 CONTINUE
469 CONTINUE
470 CONTINUE
471 CONTINUE
472 CONTINUE
473 CONTINUE
474 CONTINUE
475 CONTINUE
476 CONTINUE
477 CONTINUE
478 CONTINUE
479 CONTINUE
480 CONTINUE
481 CONTINUE
482 CONTINUE
483 CONTINUE
484 CONTINUE
485 CONTINUE
486 CONTINUE
487 CONTINUE
488 CONTINUE
489 CONTINUE
490 CONTINUE
491 CONTINUE
492 CONTINUE
493 CONTINUE
494 CONTINUE
495 CONTINUE
496 CONTINUE
497 CONTINUE
498 CONTINUE
499 CONTINUE
500 CONTINUE
501 CONTINUE
502 CONTINUE
503 CONTINUE
504 CONTINUE
505 CONTINUE
506 CONTINUE
507 CONTINUE
508 CONTINUE
509 CONTINUE
510 CONTINUE
511 CONTINUE
512 CONTINUE
513 CONTINUE
514 CONTINUE
515 CONTINUE
516 CONTINUE
517 CONTINUE
518 CONTINUE
519 CONTINUE
520 CONTINUE
521 CONTINUE
522 CONTINUE
523 CONTINUE
524 CONTINUE
525 CONTINUE
526 CONTINUE
527 CONTINUE
528 CONTINUE
529 CONTINUE
530 CONTINUE
531 CONTINUE
532 CONTINUE
533 CONTINUE
534 CONTINUE
535 CONTINUE
536 CONTINUE
537 CONTINUE
538 CONTINUE
539 CONTINUE
540 CONTINUE
541 CONTINUE
542 CONTINUE
543 CONTINUE
544 CONTINUE
545 CONTINUE
546 CONTINUE
547 CONTINUE
548 CONTINUE
549 CONTINUE
550 CONTINUE
551 CONTINUE
552 CONTINUE
553 CONTINUE
554 CONTINUE
555 CONTINUE
556 CONTINUE
557 CONTINUE
558 CONTINUE
559 CONTINUE
560 CONTINUE
561 CONTINUE
562 CONTINUE
563 CONTINUE
564 CONTINUE
565 CONTINUE
566 CONTINUE
567 CONTINUE
568 CONTINUE
569 CONTINUE
570 CONTINUE
571 CONTINUE
572 CONTINUE
573 CONTINUE
574 CONTINUE
575 CONTINUE
576 CONTINUE
577 CONTINUE
578 CONTINUE
579 CONTINUE
580 CONTINUE
581 CONTINUE
582 CONTINUE
583 CONTINUE
584 CONTINUE
585 CONTINUE
586 CONTINUE
587 CONTINUE
588 CONTINUE
589 CONTINUE
590 CONTINUE
591 CONTINUE
592 CONTINUE
593 CONTINUE
594 CONTINUE
595 CONTINUE
596 CONTINUE
597 CONTINUE
598 CONTINUE
599 CONTINUE
600 CONTINUE
601 CONTINUE
602 CONTINUE
603 CONTINUE
604 CONTINUE
605 CONTINUE
606 CONTINUE
607 CONTINUE
608 CONTINUE
609 CONTINUE
610 CONTINUE
611 CONTINUE
612 CONTINUE
613 CONTINUE
614 CONTINUE
DO 188 I=MIN,N
188 CONTINUE
189 IF(X(I)=MAX)
189 CONTINUE
190 I=1,NCORP
190 I=1,NCORP
X(IMIN+I)=XC(I)
Y(IMIN+I)=YC(I)
X(IMIN+NCORP+I)=X(IMAX+I)
Y(IMIN+NCORP+I)=Y(IMAX+I)
Y(I)=N
Y(I)=N
TYPE 171
171 FORMAT(' REPLAY? 'S)
ACCEPT 21,LYES
IF(LYES.EQ.'Y') GO TO 174
CALL ALTPLOT(MOUT,X,Y)
174 TYPE 192
192 FORMAT(' FURTHER ALTERATIONS? 'S)
ACCEPT 21,LYES
IF(LYES.EQ.'Y') GO TO 176
WRITE MODIFIED FILE
42 CALL WRITE(MOUT,X,Y)
24 CONTINUE
C
37 TYPE 39
39 FORMAT(' DO YOU WISH TO PLOT INPUT AND OUTPUT FILES? 'S)
ACCEPT 21,LYES
IF(LYES.EQ.'N') GO TO 47
TYPE 35
35 FORMAT(' DO YOU WISH TITLES AND LABELS?')
ACCEPT 21,MT
21 FORMAT(A1)
IF(MT.EQ.1MN) GO TO 1000
CALL MYLABL(TITLE,LNBK,LABY)
TYPE 33
33 FORMAT(' PLOT WITH GRID?')
ACCEPT 21,MO
1000 CALL BINIT
TYPE 15
15 FORMAT(' ENTER UMIN, UMAX, TMIN, TMAX 'S)
ACCEPT 16,YMIN,YMAX,XMIN,XMAX
16 FORMAT(4G)
C
PLOT INPUT AND MODIFIED FILES,
36 CALL NEWPAG
DO 23 K = 1,2
HM=MOD(K,2)
Mprofil(K)=IPIL
IF(HM.EQ.0) MPROF(K)=MOUT
22 CALL PLSCAL(X,Y,MPROF,K,XMIN,XMAX,YMIN,YMAX,MQ,MH)
23 CONTINUE
IF(MT.EQ.1MN) GO TO 38
C OUTPUT TITLE AND LABELS
CALL MOVARS(380,780)
CALL HLABEL(25,TITLE)
CALL MOVARS(5,680)
CALL VLABEL(25, LAY)
CALL MOVABS(200, 'U')
CALL HLABEL(25, LABX)
38 CALL TINPUT('M')
CALL NEWPAG
CALL ANMODE
C REPLY AGAIN?
TYPE 34
34 FORMAT(' REPLY FILES? 'S')
ACCEPT 21, MG
IF (MG, EQ, 'N') GO TO 17
GO TO 1000
17 TYPE 30
30 FORMAT(' MODIFY OTHER FILES? 'S')
ACCEPT 21, MG
IF (MG, EQ, 'Y') GO TO 1002
1221 CONTINUE
CALL NEWPAG
300 CALL FINITX('P, ?)
CALL CLR4TK
END

C C PROFILE PLOT DURING INTERMEDIATE ALTER STEP,
C
SUBROUTINE ALTPLT(MOUT, X, Y)
DIMENSION X(1), Y(1)
CALL WRITE(MOUT, X, Y)
TYPE 10
10 FORMAT(' ENTER XMIN, XMAX, XMIN, XMAX ') 
ACCEPT 23, YMIN, YMAX, XMIN, XMAX
20 FORMAT(4G)
CALL BINITY
CALL NEWPAG
CALL PLSCAL(X, Y, MOUT, XMIN, XMAX, YMIN, YMAX, 'Y', 1)
CALL TINPUT('M')
CALL NEWPAG
CALL ANMODE
RETURN
END

C C READ ONE FILE FROM DISK
SUBROUTINE READ(IFIL, X, Y)
DIMENSION X(1), Y(1)
OPEN(UNIT = '20', FILE='IFIL', ACCESS='SEQIN')
Q = 0
10 IF Q = 0 GO TO 106
READ(20, 106, END = 20) X(1), Y(1)
106 FORMAT(2G)
10 Q = NO + 1
22 TYPE 22, IFI
22 FORMAT(' NOT ALL DATA READ FROM FILE ', AS, ' 500 PTS., MAX, ?')
20 X(1) = NO - 1
20 Y(1) = NO - 1
CALL RELEASE(20)
RETURN
END

C C SUBROUTINE PLSCAL(X, Y, KFIL, K, XMIN, XMAX, YMIN, YMAX, MO, HH)
C PLOT ONE DISK FILE USING GLOBAL MAXIMA AND MINIMA FOR X AND Y ARRAYS.
C
"M IS 0--PLOT A DOTTED LINE FOR FILES 2, 4, 6, ETC.
C "M IS 1--PLOT A SOLID LINE FOR FILES 1, 3, 5, ETC.
C DIMENSION X(1), Y(1), KFIL(1)
C IF M = KFIL(K)
C CALL READ(IFIL, X, Y)
C CALL DLIMX(XMIN, XMAX)
C CALL DLIMY(YMIN, YMAX)
C IF (NO.EQ.1) GO TO 10
C
C DELETE GRID WITH NEXT TWO CALLS
C CALL YFRM(2)
C CALL XFRM(2)
C
CALL LINE(K)
C IF MM IS 0, THEN DOT LINES--DOTTED LINES FOR 2, 4, ETC
C IF (K.EQ.1) GO TO 12
C IF (MM.EQ.0) CALL LINE(1212)
C
C PLOT FILES 2, 3, ETC,
C CALL C PLOT(X, Y)
C RETURN
C
C PLOT FILE 1
C CALL CHECK(X, Y)
C CALL DSPLAY(X, Y)
C RETURN
C
C
C THIS SUBROUTINE FINDS THE MAXIMUM AND MINIMUM OF AN ARRAY
C SUBROUTINE MINMAX(XX, XX1, XX2)
C DIMENSION XX(1)
C XX = XX(1)
C DO 10 K = 2, N
C XX1 = AMIN1(XX(K), XX1)
C XX2 = AMAX1(XX(K), XX2)
C 10 CONTINUE
C RETURN
C
C END
C
C THIS SUBROUTINE WRITES ONE FILE TO DISK
C SUBROUTINE WRITEOFIL(X, Y)
C DIMENSION X(1), Y(1)
C OPEN (UNIT=21, FILE=OFIL, ACCESS='SEQOUT')
C GO, OF POINTS RESIDES IN X(1), Y(1)
C DO = X(1)
C NOP1 = NO + 1
C DO 20 I = 2, NOP1
C WRITE (21, 100) X(I), Y(I)
C 100 FORMAT(2F)
C 20 CONTINUE
C CALL RELEAS (21)
C RETURN
C
C END
C
C PRESET MINIMA AND MAXIMA OF X AND Y ARRAYS
C SUBROUTINE PRESET(XMIN, XM, YMIN, YM)
C XMIN = 1.48
C XM = 1.48
C YMIN = XMIN
C YM = XM
C RETURN
C
C
END

THIS SUBROUTINE ACCEPTS LABELS AND TITLES FOR GRAPH
FROM THE KEYBOARD, 25 CHARACTERS PER TITLE OR LABEL.

SUBROUTINE MYLABL(LTITLE,LABX,LABY)
DIMENSION LTITLE(1),LABX(1),LABY(1),KTITLE(5)

16 CALL ANMODE
TYPE 8
READ (5,11) KTITLE
FORMAT(1'A TITLE? ' /)
11 FORMAT(5A5)
CALL KAN2AS(25,KTITLE,LTITLE)
DO 15 J = 1,5
15 KTITLE(J) = '
CALL ANMODE
TYPE 10

READ (5,11) KTITLE
FORMAT(' LABEL FOR X-AXIS? ' /)
CALL KAN2AS(25,KTITLE,LABX)
DO 20 J = 1,5
20 KTITLE(J) = '
CALL ANMODE
TYPE 12
READ (5,11) KTITLE
FORMAT(' LABEL FOR Y-AXIS? ' /)
CALL KAN2AS(25,KTITLE,LABY)
TYPE 14

FORMAT(' ALL TITLES AND LABELS OK? ' /)
ACCEPT S, M0
FORMAT(A1)
IF (M0,EQ.,'Y') GO TO 16
RETURN
END
APPENDIX II

Listing of Program IMP
PROGRAM IMP

C	TENNIS FRANCY, FEBRUARY, 1975
C	THE PRESENT PROGRAM PROVIDES A REDUCTION
C	OF VELOCITY INTERFEROMETER DATA TO STRESS-STRAIN
C	PATHS AND MODULUS.
C	COMMON/BRDF/VSAM(5),A(100),B(100),C(100),TV(5,100),NOF
C	COMMON/BRDCS/NSAM,NC,4ORDER(5,100)
C	COMMON/BRDCS/V,RO
C	COMMON/BRDCV/UX
C	COMMON/BRIV/VU(500),TV(500),TV(1000),XL(5),NT(5)
C	COMMON/BRIV/S(500),E(100)
C	COMMON/BRIC/VCOR(100)
C	DIMENSION IFIL(10),DELU(100),NDEL(100),X1(100)
C	LOGICAL S,ICOPY
C
C	PRESET FOR TEK PLOTTING
C
C	CALL SET4TK
C	CALL INIT(30)
C	CALL TERM(1,1024)
C	CALL EDIT
C	CALL ANMODE
C
C	OPERATION: 1,2,3.
C
C	INPUT AND IDENTIFY ALL PROFILES, NORMALIZE, AND INCREMENT.
C
1 TYPE 7
7 FORMAT(1,'ENTER NUMBER OF FILES TO READ',S)
9 ACCEPT 9, NOF
11 FORMAT(I)
DO 8 I = 1, NOF
8 NDEL(I) = 0
CONTINUE
DO 36 J=1,NOF
10 TYPE 11
11 FORMAT(1,'INPUT FILE',S)
12 ACCEPT 12, IFIL(J)
36 CONTINUE
21 TYPE 21
21 FORMAT(1,'ENTER VELOCITY INCREMENT',S)
22 ACCEPT 22, DUX
23 FORMAT(F)
23 TYPE 23
23 ACCEPT 47, NYES
23 FORMAT(1,'PLOT VELOCITY VS. TIME',S)
DO 41 I = 1, NOF
39 CONTINUE
39 FORMAT(1,'ENTER SAMPLE',I4,' THICKNESS',S)
40 ACCEPT 49, XSAH(I)
41 CONTINUE
CALL NEWPNG
XMIN = 1.ER
XMAX = 1.ES
DO 35 J = 1, NOF
35 CALL READ(IFIL(J),NF)
CALL VINCNT(NF,NT(J),IFIL(J))
NLT(J) = NL(T(J))
DO 24 K = 1, NLT
DO 24 K = 1, NLT
TV(J,K) = TV(K)
CONTINUE

TYPE 20
70 FORMAT(' FIT PROFILE FEET TO PRESUMED VELOCITY? 'F)
ACCEPT 47, MYES
IF(MYES, EQ., 'Y') GO TO 38

TYPE 72
72 FORMAT(' ENTER FOOT VELOCITY? 'F)
ACCEPT 46, CFoot
CALL EFIT(CFOOT)

36 IF(MYES, EQ., 'Y') GO TO 77
DO 76 J = 1, NOF
NNT = NT(J)
    DO 75 I = 1, NNT
        XMIN = AMIN1(XMIN, TV(J,I))
        XMAX = AMAX1(XMAX, TV(J,I))
    CONTINUE
76 CONTINUE
YMIN = 0,
YMAX = YMIN
DO 37 K = 1, NDF
    YMA = DUU * FLOAT(NLL(K))
    YMAX = AMAX1(YMAX, YMA)
    CONTINUE
37 CONTINUE
SW = .FALSE.
DO 26 J = 1, NDF
    J1 = J
    NNT = NT(J)
    DO 28 K = 1, NNT
        X1(K) = TV(J,K)
    CONTINUE
26 CONTINUE
NLL = NLL(J)
    DO 25 K = 1, NLL
        DELU(K) = DUU * FLOAT(K-1)
    CONTINUE
25 CONTINUE
NLL = NLL + 1
    DO 27 K = NLL1, NNT
        DELU(K) = DELU(NLL) - DUU * (FLOAT(K-NLL1))
    CONTINUE
27 CONTINUE
CALL PLOT(X1, DELU, XMIN, XMAX, YMIN, YMAX, IFIL, SW, J1, NNT)
CONTINUE
28 CONTINUE
CALL INPUT(KK)
CALL ANMODE
C C
C DETERMINE WAVE VELOCITIES AT EACH
C PARTIAL VELOCITY LEVEL.
C
CALL OFIT
C
SUBRNN OFIT PROVIDES A LINEAR FIT TO EACH VELOCITY
C LEVEL AND RETURNS A(I) AND B(I) IN THE RELATION T = A + Bx,
C AND C(I), THE SOUND SPEED,
C TYPE 65
C
65 FORMAT(' WHAT CHARACTERISTICS TO PLOT? 'S)
ACCEPT 9, ND
DO 45 K = 1, NDT, ND
    ND = NL(K) = 1
    CONTINUE
XMAX = -1.E5
DO 63 K = 1, NDF
   DO 63 K = 1, NDF
50 CONTINUE
XMAX = AMAX(XMAX,XSAM(K))
XMIN = AMIN(XMIN,XSAM(K))

63 CONTINUE
61 TYPE 64
60 FORMAT(1X,'ENTER THIN,THMAX 'S)
59 ACCEPT 49, THIN, THMAX
49 FORMAT(2G)
48 CALL NEWPAG
47 CALL XTPLOT(ND, THIN, THMAX)
46 CALL HOMN
45 CALL TINPUT(KK)
44 CALL ANMODE
43 TYPE 46
42 FORMAT(1X,'SHIFT TIME ON ANY PROFILE? 'S)
41 ACCEPT 47, MYES
40 FORMAT(1X)
39 IF (MYES, EQ, 'Y') GO TO 60
38 TYPE 48
37 FORMAT(1X,'WHICH PROFILE? 'S)
36 ACCEPT 9, N
35 TYPE 59
34 FORMAT(1X,'TIME SHIFT? 'S)
33 ACCEPT 27, TS
32 CALL TSHIFT (NS, TS)
31 CALL CFIT
30 TYPE 51
29 FORMAT(1X,'REPLAT? 'S)
28 ACCEPT 47, MYES
27 IF (MYES, EQ, 'Y') GO TO 61
26 CALL ANMODE
25 TYPE 52
24 FORMAT(1X,'ADD ANY CHARACTERISTICS? 'S)
23 ACCEPT 47, MYES
22 IF (MYES, EQ, 'Y') CALL DELADD(ND, 1)
21 TYPE 54
20 FORMAT(1X,'DELETE ANY CHARACTERISTICS? 'S)
19 ACCEPT 47, MYES
18 IF (MYES, EQ, 'Y') CALL DELADD(ND, 2)
17 TYPE 55
16 ACCEPT 47, MYES
15 IF (MYES, EQ, 'Y') GO TO 79
14 GO TO 61
13
12 TYPE 80
11 FORMAT(1X,'DO YOU WISH TO ADJUST THE WAVE VELOCITY DATA? 'S)
10 ACCEPT 47, LYES
9 IF (LYES, EQ, 'Y') GO TO 66
8 CALL CADJST(IFIL,DELU)
7
6 CORRECT FOR THE MATERIAL-WINDOW INTERFACE IMPEDANCE MISMATCH.
5 CALL NEWPAG
4 CALL ANMODE
3 TYPE 13
2 FORMAT(1X,'ENTER SAMPLE DENSITY IN GRAMS/CC 'S)
1 ACCEPT 14, RHO
0 FORMAT(G)
-1 RHO = RHO * 10,
-2 CALL NEWPAG
-3 CALL CORVEL
-4 SUBROUTINE CORVEL RETURNS THE CORRECTED WAVE VELOCITIES.
-5 DETERMINE THE STRESS-STRAIN DATA.
CALL SSRT

**SUBROUTINE SSRT RETURNS CORRESPONDING STRESS-STRAIN DATA.**

**TYPE 81**

FORMAT(?, 'DO YOU WISH A PRINT OUT OF THE DATA? ?')
ACCEPT 47, LYES
IF(LYES.EQ,'N') GO TO 105
CALL NEWPAG
CALL ANMODE

**TYPE 82**

**LOADING DATA **

**TYPE 83**

**STRESS STRAIN PARTICLE WAVE */

**TYPE 84**

**FORMAT(?, 'VELOCITY VELOCITY')**

LINE = 5
DO 88 I=1,NCL
    LINE = LINE + 1
    ICOPY = LINE .GT. 30
    IF (ICOPY) CALL HDCOPY
    IF (ICOPY) TYPE 83
    IF (ICOPY) LINE = 0
    TYPE 85,1,ST(I),T(I),UCOR(I),C(I)
88    CONTINUE

**FORMAT(?, 'UNLOADING DATA */')**

CALL TINPUT(KKK)
CALL NEWPAG
CALL ANMODE

**TYPE 86**

**TYPE 84**

LINE = 5
ICOPY = ,FALSE.
NCLP1=NCL+1
DO 89 I=NCLP1,NCT
    LINE = LINE + 1
    ICOPY = LINE .GT. 30
    IF (ICOPY) CALL HDCOPY
    IF (ICOPY) TYPE 83
    IF (ICOPY) LINE = 0
    TYPE 85,1,ST(I),T(I),UCOR(I),C(I)
89    CONTINUE

CALL TINPUT(KKK)
CALL NEWPAG

**TYPE 119**

**FORMAT(?, 'DO YOU WISH TO MAKE OUTPUT DATA FILES ? ?')**
ACCEPT 47, LYES
IF(LYES.EQ,'N') GO TO 119
CALL WRITE(I,E,S,I,NCT)
CALL WRITE(S,UCOR,Y,I,NCT)
CALL WRITE(S,S,C,I,NCL)
NCL1=NCL+1
CALL WRITE(4,S,C,NCL1,NCT)

CALL RESULT

**TYPE 67**

**FORMAT(?, 'REDUCE ANOTHER SET OF DATA ? ?')**
ACCEPT 47, HYES
IF (HYES.EQ.'Y') GO TO 1
CALL FINIT(0,0)
CALL CLRT4K
END

SUBROUTINE WRITE(NP,A,B,N1,N2)
DIMENSION A(1),B(1)
IF(N0.EQ.1) TYPE 111
IF(N0.EQ.2) TYPE 112
IF(N0.EQ.3) TYPE 113
IF(N0.EQ.4) TYPE 114
111 FORMAT(' NAME STRESS-STRAIN FILE 'S)
112 FORMAT(' NAME STRESS-PARTICLE VELOCITY FILE 'S)
113 FORMAT(' NAME LOADING WAVE VELOCITY-STRESS FILE 'S)
114 FORMAT(' NAME UNLOADING WAVE VELOCITY-STRESS FILE 'S)
ACCEPT 1A, IDFIL
10 OPEN(UNIT=20,FILE=IDFIL,ACCESS='SEQOUT')
WRITE(20,120)(A(I),B(I),I=N1,N2)
120 FORMAT(2F)
CALL RELEASE(20)
RETURN
END

SUBROUTINE DELAG(NDEL,ISW)
C THIS ROUTINE SETS FLAGS TO DELETE OR ADD CHARACTERISTIC LINES
DIMENSION NDEL(1), NDELTA(10)
TYPE 10
10 FORMAT(' HOW MANY? (K=10) 'S)
ACCEPT 12, NUMR
12 FORMAT(13I1)
TYPE 14
14 FORMAT(' ENTER ALL CHARACTERISTIC NOS., ON ONE LINE 'S)
ACCEPT 12, (NDELTA(K), K = 1, NUMB)
DO 16 K = 1, NUMB
   I = NDELTA(K)
   IF(ISW.EQ.1) NDEL(1) = 1
   IF(ISW.EQ.0) NDEL(1) = 0
16 CONTINUE
RETURN
END

SUBROUTINE TSHIFT(NS,TS)
C TIME SHIFT ARRAY TV BY TIME, TS
COMMON / BIV/U(500), T(500), TVV(100), NL(5), NT(5)
COMMON / BICF/XSAM(5), A(100), B(100), C(100), TVV(5,100), NOF
NTY = NT(5)
DO 10 K = 1, NTT
   TVV(NS,K) = TVV(NS,K)*TS
10 CONTINUE
RETURN
END

SUBROUTINE FFFT(CFOOT)
COMMON / BIV/U(500), T(500), TVV(100), NL(5), NT(5)
COMMON / BICF/XSAM(5), A(100), B(100), C(100), TVV(5,100), NOF
DO 10 I = 1, NOF
   TCC = TVV(I,1)-XSAM(I)/CFOOT
   NTT = NT(1)
   DO 0 J = 1, NTT
   CONTINUE
10 CONTINUE
TV(I,J) = TV(I,J) - TCOR
CONTINUE
RETURN
END

PLOT X VS. TIME
SUBROUTINE XTPLOT(NOEL,THIN,TMAX)
COMMON/BCFF/XSAM(5),A(100),B(100),C(100),TV(5,100),NOF
COMMON/BICCCS/ NCL,ACT,TORDER(5,100)
LOGICAL ITFIP
DIMENSION TIME(100),X(100),APT(3),XX(3),NOEL(1),LAB1(6),LAB2(6)
DATA LAB1/5H,5H,5H,5H,5H,5H /
DATA LAB2/5H,5H,5H,5H,5H,5H /
CALL BINIT
XMIN = 0.
XMAX = XSAM(0) + 1.
ITFIP = TRUE,
TO 10 J = 1, NOT
IF (NOEL(J) .EQ. 0) GO TO 10
N = 1
DO 12 K = 1, NOE
IF (TORDER(K,J),LT.,1,E-4) GO TO 12
N = N + 1
X(N) = XSAM(K)
TIME(N) = TORDER(K,J)
CONTINUE
12
TIME(1) = FLOAT(N-1)
X(1) = FLOAT(N-1)
APT(1) = 2.
APT(2) = A(J)
APT(3) = A(J) + B(J) * (XSAM(0) + 1.)
XX(1) = 2.
XX(2) = 0.
XX(3) = XMAX
CALL XFRM(2)
CALL YFRM(2)
CALL YMAX(0)
CALL DLHXY(XMIN,XMAX)
CALL DLHYY(THIN,TMAX)
CALL LINE(-1)
CALL SI Zus(2,29)
CALL SYM(2)
CALL CHECK(X,TIME)
IF (ITFIP) GO TO 14
CALL CLOT(X,TIME)
CALL LINE(0)
CALL SYM(0)
CALL CLOT(XX, APT)
GO TO 10
14
CALL DISPLAY(X,TIME)
CALL LINE(0)
CALL SYM(0)
CALL CLOT(XX, APT)
IF (ITFIP) .FALSE.
CONTINUE
10
IF (THIN .GE. 0.) GO TO 18
APT(2) = 0.
APT(3) = 0.
CALL LINE(5252)
CALL CPOST (XX, APT)
CONTINUE
CALL PUTON (LAB1, LAB2)
RETURN
END

SUBROUTINE VINC(NNT, NNL, NF, IFI)
LOGICAL JHI, PEAK
COMMON/BIV/DUX
COMMON/BIV /U(500), T(500), TTV(100), NL(5), NT(5)
C
NOISE NOT USED IN THE RAW DATA SPACE
DO 8 I=1,50
   DIF=ABS(U(I+1)-U(I))
   IF(DIF,GE.,0.01) GO TO 9
   CONTINUE
8 TYPE 10, IFI
10 FORMAT(' TVV(I) NOT FOUND IN FIRST 50 DATA POINTS
1 FROM FILE 'AS.' IN S, R, VINC')
CALL EXIT
C
FIRST POINT OF TTV ARRAY IS T(1)
TTV(1) = T(1)
JHI = ,FALSE,
PEAK = ,FALSE,
UX = DUX
NNL = 1
J = I + 1
20 J = J + 1
C
J WITHIN RAW DATA?
IF J.GT. (NF-1) JHI = ,TRUE.
IF (.NOT. JHI) GO TO 21
C
CALCULATE LOADING DATA
21 DIF = T(J+1) - T(J)
   IF (DIF,LT. (1.,3E-5)): PEAK = ,TRUE.
   IF (PEAK) GO TO 26
   IF (UJ(LT. UX)) GO TO 20
   NNL = NNL + 1
   S = (U(J) - U(J-1))/(T(J) - T(J-1))
   TTV(NNL) = T(J) - (U(J) - UX)/S
   UX = UX + DUX
   J = J + 1
GO TO 21
C
PEAK FOUND
C
ALL LOADING DATA CALCULATED
26 NNT = NNL
   UX = UX - DUX
C
J WITHIN RAW DATA?
30 IF (J.GT. (NF-1)) JHI = ,TRUE.
C
ALL RAW DATA HAS BEEN USED
IF (JHI) RETURN
IF (U(J),LE. UX) GO TO 32
C
CALCULATE UNLOADING DATA
32 NNT = NNT + 1
8. (U(J) = U(J-1)/(T(J) - T(J-1))
TTV(NNT) = T(J) - (U(J) - UX)/S
J = J + 1
GO TO 30
END

SUBROUTINE CFIT
COMMON/BICE / XSAH(5),A(100),B(100),C(100),TV(5,100),NOF
COMMON/BIV / U(500),T(500),TTV(100),NL(5),NT(5)
COMMON/BICCCS/NCL,NCT,TORDER(5,100)
DIMENSION NSH(5),NX(5)
NMAX = 0
DO 3 I = 1, NOF
NMAX = MAX(NMAX, NL(I))
3 CONTINUE
DO 4 I = 1, NOF
NX(I) = NL(I)
4 CONTINUE
ARRANGE ARRAY NX IN HI TO LO ORDER
NOF1 = NOF - 1
DO 5 I = 1, NOF1
IST = I + 1
DO 7 J = IST, NOF
IF (NX(J) = NX(I),LT, 0) GO TO 7
SAVE = NX(J)
NX(J) = NX(I)
NX(I) = SAVE
7 CONTINUE
CONTINUE
NX = NX(2)
ORDER TIME ARRAY TO AN ARRAY FOR FITTING CHARACTERISTICS
DO 14 J = 1, NOF
DO 14 1 = 1, 100
TORDER(J, 1) = 0,
14 CONTINUE
CONTINUE
DO 30 J = 1, NOF
IDIF = NCL - NL(J)
LENGTH = NT(J) + 2 * IDIF
DO 30 I = 1, LENGTH
IF (I,LE, NCL) TORDER(J,I) = TV(J,I)
IF (I,GT, NCL) TORDER(J,I) = TV(J,I-2*IDIF)
30 CONTINUE
CONTINUE
IF (IDIF,LE, 2) GO TO 30
LMIN = NCL - IDIF + 1
LMAX = NCL + IDIF
DO 45 M = LMIN, LMAX
TORDER(J,M) = 0,
45 CONTINUE
CONTINUE
NCT = 1
CONTINUE
CONTINUE
NX = UX - DUX
J = J + 1
GO TO 30
END
NSW(J) = 1
NTOT = NTOT + 1

17 CONTINUE
IF (NTOT,LT, 2) GO TO 55
C DETERMINE LEAST SQUARE POINTS
SUMX = 0,
SUMT = 0,
SUMXT = 0,
SUMX2 = 0,
NSUM = NTOT
DO 18 IS = 1,NSUM
SUMX = SUMX + NSW(IS) * XSAM(IS)
SUMT = SUMT + NSW(IS) * TORDER(IS,NCT)
SUMXT = SUMXT + NSW(IS) * XSAM(IS) * TORDER(IS,NCT)
SUMX2 = SUMX2 + NSW(IS) * XSAM(IS) ** 2
18 CONTINUE
DEN = NSUM * SUMX2 - SUMX ** 2
ANUM = SUMT * SUMX2 - SUMXT * SUMX
A(NCT) = ANUM / DEN
B(NCT) = BNUM / DEN
C(NCT) = 1. / B(NCT)
NCT = NCT + 1
GO TO 12

55 NCT = NCT - 1
RETURN

SUBROUTINE CORVEL
C << COMPUTES CORRECTED PARTICLE VELOCITIES >>
COMMON/BIGCSS/NCL,NCT,TORDER(5,100)
COMMON/BIGCS/UOR(102)
COMMON/BICOS/RHO
COMMON/BIC0V/DUX
COMMON/BICF/XSAM(5),A(102),B(102),C(102),TV(5,100),NSUM
DIMENSION FP(102)
C << INITIAL SETTING OF F >>
10 FP(I)=RHO*C(1)
C << EVALUATION OF CORRECTED U >>
15 UCOR(1)=?
DO 20 I=2,NCL
UCOR(1)=UCOR(I-1)*P,5*(1,0*GP(I-1)/FP(I-1))@DUX
20 CONTINUE
UCOR(NCL+1)=UCOR(NCL)
UCLP2=NCL+2
DO 30 I=NCLP2,NCT
UCOR(I)=UCOR(I-1)-0.5*(1,0*GP(I-1)/FP(I-1))@DUX
30 CONTINUE
IF(NSUM.EQ.1) RETURN
NSUM=NSUM+1
C << REEVALUATE F >>
DO 40 I=2,NCT
ITEM*FP(J)
40 FP(I)=RHO*ITEM
END
FUNCTION FPP(I)
C<br>** REEVALUATE F' FOR SOME CORVOL >**<br>COMMON/BICCSS/NCL,NCT,TORDER(5,100)<br>COMMON/BICF/XSAW(5),A(100),B(100),C(100),TV(5,100),NOF<br>IF(I.GT.NCL) GO TO 10
C<br>** LOADING >>**<br>* TEST=GG(I)<br>CALL SINT(STEST,STF,CF,1)<br>IF(SF.GT.,) FPP=CF<br>IF(SF.GT,3) RETURN<br>FPP=C(NCL)<br>RETURN
C<br>** UNLOADING >>**<br>10 TEST=GG(NCL)<br>CALL SINT(STEST,STF,CF,1)<br>IF(SF.GT,.) GO TO 20<br>IF(GG(I),GE,STF) FPP=C(MCL+1)<br>IF(GG(I),GE,STF) RETURN<br>TEST=GG(NCL)<br>CALL SINT(STEST,STF,CF,2)<br>IF(SF.GT,.) FPP=CF<br>IF(SF.GT,2) RETURN<br>FPP=C(NCT)<br>RETURN<br>END<br>C<br>** SUBROUTINE SINT(STEST,STF,CF,IS)**
C<br>** INTEGRATES CURRENT CORRECTED P-U PATH >>**<br>COMMON/BICCSS/NCL,NCT,TORDER(5,100)<br>COMMON/BICF/XSAW(5),A(100),B(100),C(100),TV(5,100),NOF<br>COMMON/BICCSS/RHOR<br>IF(IS.EQ.2) GO TO 30<br>S=0.,<br>DO 20 I=2,NCL<br>Z=RHO*(C(I)+C(I-1))/2.,<br>S=S+(UCOR(I)-UCOR(I-1))/2.<br>IF(S.GT.STEST) GO TO 20<br>CF=C(I-1)+(C(I)-C(I-1))*(STEST-R)/(S-R)<br>SF=S<br>CONTINUE<br>20 CONTINUE<br>SF=1.0<br>RETURN<br>30 DO 40 I=2,NCT<br>Z=RHO*(C(I)+C(I-1))/2.,<br>S=S+(UCOR(I)-UCOR(I-1))/2.<br>IF(I.LE.NCL) GO TO 40<br>IF(S.GT.STEST) GO TO 40<br>CF=C(I-1)+(C(I)-C(I-1))*(STEST-R)/(S-R)<br>SF=S<br>RETURN<br>40 CONTINUE<br>SF=1.0<br>RETURN
SUBROUTINE ST  
COMPUTE STRESS STRAIN DATA
COMMON/BICS/RHCL, TICM, TVT(5), NT(5)
COMMON/BICS/RHCL, TICM, TVT(5), NT(5)

FUNCTION G11
COMPUTE EQUATION FOR MOLD MATERIAL
COMMON/BICS/RHCL, TICM, TVT(5), NT(5)

TYPE 82
CONTINUE
CALL RELEAS(20)
RETURN
END

READ FILE FROM DISK
COMMON/BICS/RHCL, TICM, TVT(5), NT(5)
COMMON/BICS/RHCL, TICM, TVT(5), NT(5)

SUBROUTINE PERFILE( )
RETURN
END

END
RETURN
END

SUBROUTINE PLOT(X1,Y,XMI,YMI,XMA,YMA,IFIL,SW,K,NO)
DIMENSION Y(1),X1(1),ISYM(12),IFIL(1)
LOGICAL SW
COMMON /BICF/ XSAM(5),A(100),B(100),C(100),TV(5,100),NOF
DIMENSION LAB1(6), LAB2(4), LAB3(6), LAB4(6)
DATA ISYM/1,1,1,1,1,1,1,1,1,1/ 
DATA LAB1/5H ,5H ,5H ,5H ,5H ,5H ,5H / 
DATA LAB2/5H ,5H ,SHSOUND,SHSPEED,SHCITY ,5H ,5H / 
DATA LAB3/5H ,5H ,SHPART,SHCABLE,SHSPEED,SHCITY,5H / 
DATA LAB4/5H ,5H ,SHPART,SHCABLE,SHSPEED,SHCITY,5H / 
CALL DLINX(XMI,XMA)
CALL DLINY(YMI,YMA)
CALL LINE (-1)
CALL SIZE (0,15)
CALL SYMBOL (ISYM(K))
IF (K.EQ.1) GO TO 27
IF (K.LT. NOF) RETURN
CALL PUTON (LAB1,LAB2)
CALL MOVA95(40,760)
CALL ANMODE
TYPE 26,(IFIL(K1),K1=1,NOF)
FORMAT (* FILES=*,10A7)
RETURN
CONTINUE
CALL PUTON(LAB4,LAB3)
RETURN

CALL FRAME
CALL NPTS(0)
CALL CHECK(X1,Y)
CALL DISPLAY(X1,Y)
RETURN

SUBROUTINE RESULT
COMMON /BICF/ XSAM(5),A(100),B(100),C(100),TV(5,100),NOF
COMMON /BICF/ X1V(50),I5(500),TTV(100),NL(5),NT(5)
COMMON /BICF/ X1S(100),E(100)
COMMON /BICF/ X1DCS/UOC(100)
COMMON /BICF/ X1CS/NOL,NCT,TORDER(5,100)
DIMENSION LAB1(6),LAB2(6), LAB3(6), LAB4(6)
DATA LAB1/5H ,5H ,5H ,5H ,5H ,5H ,5H / 
DATA LAB2/5H ,5H ,5H ,5H ,5H ,5H ,5H / 
DATA LAB3/5H ,5H ,5H ,5H ,5H ,5H ,5H / 
DATA LAB4/5H ,5H ,5H ,5H ,5H ,5H ,5H / 

KEY TO REQUESTS FOR PLOTTING
CALL NEWPAG

CALL NEWPAC
CALL ANMODE
TYPE 4
FORMAT(' TO OBTAIN GRAPH STRESS-STRAIN, TYPE "SE"/ 
119X, "STRESS-PARTICLE VELOCITY, TYPE "SV"/ 
219X, "SOUND VELOCITY-STRESS, TYPE "CS"/ 
319X, "SOUND VELOCITY-PARTICLE VELOCITY, TYPE "CV"/ 
CALL ANMODE
TYPE 12
FORMAT(' WHICH GRAPH? ..., TYPE "NONE" TO EXIT ') 
ACCEPT 1X, MEM
CALL NEWPAGE
IF (MEM,NE,'NONE') RETURN
CALL BINIT
CALL NPTS(NCT)
CALL STRESS-STRAIN PLOT
IF (MEM,NE,'SE') GO TO 20
CALL CHECK (E,S)
call display (E,S)
call puton (LAB3,LAB1)
call tinput(KKK)
go to 60
call stress-particle velocity (corrected) plot
20 IF (MEM,NE,'SV') GO TO 30
CALL CHECK (UCON,S)
call display (UCOR,S)
call puton (LAB4,LAB1)
call tinput(KKK)
go to 60
call stress- sound velocity plot
30 IF (MEM,NE,'CS') GO TO 50
CALL CHECK (S,C)
call display (C)
call puton (LAB1,LAB2)
call tinput(KKK)
go to 50
50 IF (MEM,NE,'CV') GO TO 5
CALL CHECK (UCOR,C)
call display (UCOR,C)
call puton (LAB4,LAB2)
call tinput(KKK)
go to 50
60 CALL HOME
GO TO 5
END

THIS SUBROUTINE PUTS TITLE AND LABELS ON A GRAPH
SUBROUTINE PUTON(LABX,LABY)
DIMENSION LTITLE(1),LABX(1),LABY(1),LAB(3)
CALL MOVABS(30°,780)
call hlable(30°,LTITLE)
call movabs(5°,663)
call kampas(30°,LABY,LAB)
call vlarel(30°,LAB)
call movabs(30°,25)
call kampas(30°,LABX,LAB)
call hlable(30°,LAB)
CALL ANMODE
RETURN
END

SUBROUTINE CADIUS(IFIL,DELU)
ADJUST THE WAVE VELOCITY DATA

COMMON SIGF /XSAM(5),A(100),B(100),C(100),TV(5,100),NOF
COMMON /HICOV/UX
COMMON /RICECS/NCL,NCT,TORDER(5,100)

LOGICAL SW
SW = .TRUE.
DO 5 K = 1, NCL
   DELU(K) = DUX * FLOAT(K-1)
   CONTINUE
   N1 = NCL -1
   DO 6 K = N1, NCT
      DELU(K) = DELU(NCL) - DUX * FLOAT(K-N1)
      CONTINUE
5 CONTINUE
CALL BINIT
UMIN = 0,
UMAX = DUX * FLOAT(NCL)
CMIN = 1,E5
CMAX = -CHIN
DO 8 K = 1, NCT
   CHIN = A(1M1) CHIN, C(K))
   CMAX = AMAX1( CMAX, C(K))
8 CONTINUE
CALL NREP
CALL PLUT(DELU,C,UMIN,UMAX,CHIN,CMAX,IFIL,SW,1,NCT)
CALL HOME
CALL TINPUT(KKK)
CALL ANMODE

TYPE 13
FORMAT(' DO YOU WISH TO ADJUST PARTICULAR POINTS? Y S)
ACCEPT 14, 'YES

14 FORMAT(A1)
IF (YES, EQ, 'Y') GO TO 40

TYPE 15
FORMAT(' ENTER NUMBER OF POINTS TO CORRECT, S)
ACCEPT 20, NTS

20 FORMAT()
DO 30 I = 1, NTS
30 FORMAT(' ENTER POINT NUMBER AND NEW VELOCITY, S)
ACCEPT 25,NPT,CNEW

25 FORMAT(2G)
C(NPT) = CNEW
CONTINUE

TYPE 45
FORMAT(' DO YOU WISH PROFILE VELOCITIES SCALED AT A
PITICULAR LEVEL? Y S)
ACCEPT 14, 'YES
IF (YES, EQ, 'Y') GO TO 80

TYPE 55
FORMAT(' ENTER LEVEL NUMBER AND NEW VELOCITY, S)
ACCEPT 60, NLEV, CNEW

60 FORMAT(2G)
C(NLEV) = C(NLEV)
DO 70 I = 1, NCT
   C(I) = 1, / (1./C(I) -1, /CNLEV + 1, /CNEW)
70   CONTINUE
80   TYPE 85
85 FORMAT(' REPLLOT?')
   ACCEPT 14, IYES
   IF (IYES, EQ, 'Y') GO TO 2
   RETURN
END
REFERENCES


DISTRIBUTION (CONT.)

T. Stong, SPSS
Director, Defense Nuclear Agency
Washington, DC 20305

J. B. Walsh
Massachusetts Inst. of Technology
Cambridge, MA 02139

2513 D. B. Hayes
2513 D. E. Mitchell
4363 W. E. Alzheimer
5000 A. Narath
  Attn: 5200 E. H. Beckner
  5400 A. W. Snyder
  5700 J. H. Scott
  5800 R. S. Claassen
5100 J. K. Galt
  Attn: 5110 F. L. Vook
  5120 G. J. Simmons
  5130 G. A. Samara
  5150 J. E. Schirber
5131 J. E. Kennedy
5160 W. Herrmann
5162 L. D. Bertholf
5163 D. E. Munson
5163 D. E. Grady (25)
5163 R. E. Hollenbach
5163 K. W. Schuler
5163 R. A. Schmidt
5163 E. G. Young (5)
5163 W. R. Wawersik
5166 A. J. Chabai
5166 R. K. Byers
5167 B. M. Butcher
5167 J. R. Asay
5734 A. L. Stevens
8266 Tech. Library (2)
3141 Tech. Library (5)
3151 Tech. Writing (3)
  For ERDA/TIC (Unlimited Release)
  ERDA/TIC (25)
  (3171-1 R. P. Campbell)

L. Lee
KTech.
P.O. Box 5490
Kirtland AFB
Albuquerque, NM 87115