Procurement Executive - Ministry of Defence

AWRE, Aldermaston

AWRE REPORT NO. 08/76

A Computer Program for the Determination of the Phase Velocity of Seismic Surface Waves between Pairs of Stations

G W Stuart (University of Reading)
A Douglas
C Blamey

SBN 85518099 4

CONTENTS

SUMMARY

The program described determines the phase velocity of a dispersed seismic surface wave as a function of period between a pair of stations from their digitized seismograms. The cross-multiplication technique of Bloch and Hales [1] is used to compute the phase velocity curve.

1. INTRODUCTION

Seismic surface waves generated by a distant earthquake or explosion are dispersed; that is the velocity at which their energy (the group velocity) and their individual harmonic components (the phase velocity) propagate is a function of frequency. This is because the longer period surface waves usually travel faster than the shorter periods, as they sample deeper down in the Earth's interior where the shear and compressional wave velocities are greater.

From the dispersion of surface waves deductions can be made about the P and S wave velocity and density structure of the crust and upper mantle through which they have passed; this is usually carried out by comparing theoretical dispersion curves computed for layered Earth models with the observational data to obtain a reasonably fitting model.

The relationship between group and phase velocity is $U = C - \lambda(dC/d\lambda)$, where U is the group velocity, C is the phase velocity and λ is the wavelength.

As group velocity depends on the derivative of phase velocity, it has an inherent non-uniqueness [2], hence models are best tested against phase velocity data.

The Fourier transform of a surface wave signal $x(t)$, ($x(t) = 0$, for $t < 0$) recorded by a seismological station at a distance r, and azimuth θ from the source is

$$X(\omega) = \int_0^\infty x(t).e^{-i\omega t}dt,$$

where t is time and ω is angular frequency.

The complex spectrum of the signal, $X(\omega)$, can be represented by its amplitude, A, and phase, ϕ, functions

$$X(\omega) = A(r,\theta,\omega).e^{i\phi(r,\theta,\omega)}.$$

The phase spectrum can be represented as

$$\phi(r,\theta,\omega) = k(\omega).r + \phi_o(\theta,\omega) + \phi_i(\omega) + 2n\pi, \qquad ....(1)$$

where $k(\omega)$ is the wavenumber (2π/wavelength), $\phi_o(\theta,\omega)$ is the source phase shift and $\phi_i(\omega)$ is the instrumental phase shift. The term $2n\pi$ arises ($n = 1, 2, ...$) since the Fourier transform contains phase information only for an angle interval $(0, 2\pi)$.

Now, $$C(\omega) = \frac{\omega}{k(\omega)}.$$

To evaluate $k(\omega)$ it is necessary to determine ϕ_o, ϕ_i, and the integer n (see equation (1)).

Due to the difficulty in measuring the source phase shift ϕ_o, which can only be calculated with a knowledge of the earthquake mechanism, single station measurements of phase velocity are not frequently used. This problem may be overcome by using two stations and an earthquake or explosion on a great circle path (the interstation phase velocity method) and measuring the phase difference between the two seismograms.

From equation (1)

$$k(\omega).r_1 = \phi_1(r_1,\theta,\omega) - \phi_o(\theta,\omega) - \phi_{i1}(\omega) + 2n_1\pi \qquad ....(2)$$

for the nearest station, and

$$k(\omega).r_2 = \phi_2(r_2,\theta,\omega) - \phi_o(\theta,\omega) - \phi_{i2}(\omega) + 2n_2\pi \qquad ....(3)$$

for the furthest station.

Thus, assuming that ϕ_o is constant for a given azimuth and that both stations are at the same azimuth from the epicentre, then, by subtracting equation (1) from equation (2) and rearranging, we get

$$C(\omega) = \frac{\omega}{k(\omega)} = \frac{\omega(r_2 - r_1)}{\phi_2 - \phi_1 - \phi_{i2} + \phi_{i1} + 2m\pi}.$$

The source phase shift is thus eliminated and ϕ_{i1} and ϕ_{i2} can be calculated if the instrumental constants are known. The integer m reflects the ambiguity in the identification of peaks between the two stations, and it may be determined by a knowledge of possible phase velocity values.

2. OUTLINE OF METHOD

Interstation phase velocity may be determined using several techniques [3]. The cross-multiplication method of Bloch and Hales [1] is used in this program.

The method is based on the principle that, when two signals are in phase for a given frequency, the average of their cross-multiplication function is a maximum.

The computation procedure is as follows:-

(a) Suitable earthquakes with epicentres within 4 - 5 degrees of the interstation great circle have to be chosen in order to eliminate the influence of the source radiation pattern. Their surface wave traces are then digitized at a constant sampling rate.

(b) If the instruments at the two recording stations are not identical, their instrumental phase delays have to be taken into account before further processing.

(c) The seismograms are windowed around the group arrival time of the period of interest to eliminate any disturbance in the phase velocity determination due to microseismic noise, non-least time path arrivals and other modes. The group arrival times are calculated from group velocity values which are computed outside the program [4].

(d) The signals are then narrow band-pass filtered so that both are effectively single frequency seismograms and can be represented as $A_1 . \cos \omega t$ and $A_2 . \cos (\omega t + \phi)$ where A_1 and A_2 are constants, and ϕ is the phase difference between the seismograms.

(e) The two records are cross-multiplied to produce:-

$$A_1 . \cos \omega t . A_2 . \cos (\omega t + \phi) = (A_1 . A_2/2) . (\cos (2\omega t + \phi) + \cos \phi) \quad ..(4)$$

which represents a wave with twice the original frequency superimposed on a dc level proportional to $\cos \phi$. When the two records are in phase, $\phi = 0$, $\cos \phi = 1$ and the dc level is thus at a maximum.

(f) In practice the two filtered signals are cross-multiplied for various relative time shifts, corresponding to constant phase velocity steps, the maximum value of the dc level being recorded in each case.

(g) Steps (c) to (f) are repeated for the various periods of interest, building up to 2-D matrix of values of the maximum dc level as a function of phase velocity and period. The maxima in the contoured matrix define positive ridges which represent velocity values at which the various periods in the two seismograms are in phase. The phase velocity curve is represented by the ridge with the most reasonable velocity values.

3. PROGRAM PROCEDURE

3.1 Input data - program MAIN

Variables have the suffix 1 when referring to the station nearer to the source, and the suffix 2 for the station further away.

Each surface wave train is represented in the program as a series of digits (SEIS1(I), I = 1, NEIS1) sampled at equal time intervals, DELA, beginning from a known point in time (GMT1).

The group velocities (GV1(I), I = 1, NGV) at the periods of interest PERODl(I) are read in to construct the group arrival time windows, enabling one to reduce the extraneous noise in the analysis.

The variables for the second station are similarly represented.

Title cards and parameter cards, which specify the operations to be carried out, are read in, together with the data (see INPUT DATA CARDS).

3.2 Preparation of the time series for analysis - program MAIN

There are options in the program to remove the mean or linear trend from the time series and to cosine taper both ends (see INPUT DATA CARDS). This eliminates the effects on the analysis of a time series superimposed on a non-zero or sloping base line and it reduces Gibb's phenomenon.

If required, either of the seismograms may be inverted by the specification of IVSEIS.

For the efficient application of the Cooley-Tukey algorithm used in the Fast Fourier Transform routine COOL, the number of points in the digital series has to be a power of two. In accordance with this, the number of points in each seismogram is increased to N points by adding zeros, where N = 2**(I + 1) and I is the first integer which makes N both greater than NSEIS1 and NSEIS2.

Examples of observed and synthetic seismograms are shown in figures 1 and 3.

3.3 Windowing around the group arrival time - subroutine PACKET

Both seismograms are windowed around the group arrival time of each period of interest with a time window 4.5 times the period. A symmetrical cosine taper centred on the group arrival time is then applied to these windowed seismograms.

At longer periods where the group arrival time is near the start of the time series and the time window is large, the window may extend beyond the data block. In these circumstances the program will use an asymmetrical cosine taper. In order to avoid this happening a specified number of zeros, NOZERO, can be added to the front of both seismograms (see INPUT DATA CARDS).

Examples of the windowed synthetic seismograms are shown in figure 4(a).

3.4 Band-pass filtering of the seismograms - subroutine CRUNCH

To band-pass filter both windowed seismograms, SA(I) and SB(I) simultaneously, one is put as the real, the other as the complex part of a complex array Z. The filtering is carried out in the frequency domain by the multiplication of the spectrum of Z by that of the filter function (see section 3.4.1) for both positive and negative frequency components. When the resultant spectrum is transformed back into the time domain it has as its real and imaginary parts the band-pass filtered seismograms SA'(I) and SB'(I).

3.4.1 The filter

A Gaussian function is used as the band-pass filter because of its good resolution in both the time and frequency domains. The harmonic (OMEGAC) of the Fourier analysed time series nearest to the period of interest is chosen as the centre frequency of the filter and it is at this period (2π/OMEGAC) that the phase velocity determinations are calculated.

The band-pass Gaussian filter function, with a centre frequency ω_c, is given by

$$\exp\left\{-\alpha\left(\frac{\omega - \omega_c}{\omega_c}\right)^2\right\},$$

where α determines the filter resolution and is specified by BAND and DWF (α = ln DWF/BAND^2). BAND and DWF have typical values of 0.2 and 10.0 respectively.

Examples of the band-pass filtered seismograms are given in figure 4(b).

3.5 Parameters for time shifting in cross-multiplication - MAIN

The highest phase velocity VELHI and the lowest phase velocity of interest VELLOW are read in as input parameters. The relative time shifts of the two seismograms needed to correspond with VELHI and VELLOW are specified by NMIN and NMAX, respectively. If the first point of SEIS1(I) travels at a velocity VELHI, it would arrive at the second station at a time NMIN × DELA seconds before the first point in the second seismogram (GMT2). If SEIS1(1) travels at velocity VELLOW, it would arrive at the second station NMAX × DELA seconds after GMT2.

An array TT(I) is created which corresponds to all possible time shifts required for the phase velocity range VELHI to VELLOW. The array TSTEP(I) corresponds to the time shifts required through specification of the velocity interval DV between VELHI and VELLOW.

3.6 Cross-multiplication process - subroutine OUTPUT

The cross-multiplication function, SAMSBS(I), of the windowed band-pass filtered seismograms (cf, equation (4)) for various time shifts TT(I), is computed within this subroutine, see figure 4(c).

If the first windowed filtered seismogram is represented as SA(1), SA(2), ... SA(N), and the second SB(1), SB(2), ... SB(N), then when the point SA(NMIN) travels at a velocity VELHI, it will arrive at the second station at a time corresponding to SB(1). Hence, the cross-multiplication function SAMSBS(I) for a time shift corresponding to a phase velocity of VELHI is formed by the product of SA(NMIN).SB(1), SA(NMIN+1).SB(2), ... SA(NMIN+n).SB(n+1), ... SA(N).SB(N-NMIN+1).

The value of the dc shift is determined by finding the value X_{max} of the maximum amplitude of the product seismogram and the following minimum X_{min}. The dc level for this particular time shift is then $(X_{max} + X_{min})/2$. This value is then stored in E(1) of an array of dc levels E(k), k = 1,NP. NP is the number of points between NMIN and NMAX.

This operation is repeated for the next time shift, which corresponds to a lower phase velocity, and an array of dc levels E(k) is built up.

	TIME SHIFT →			TT(I),I=1,NP
CROSS	SA(NMIN).SB(1),	SA(NMIN-1).SB(1),	.....	SA(NMAX).SB(1)
MULTI-	SA(NMIN+1).SB(2),	SA(NMIN).SB(2),	.....	SA(NMAX+1).SB(2)
PLICATION	"	"		"
FUNCTION	SA(NMIN+n).SB(n+1),	SA(NMIN+n-1).SB(n+2),	.....	SA(NMAX+n).SB(n+1)
SAMSBS(I)	"	"		"
↓	SA(N).SB(N-NMIN+1),	SA(N).SB(N-NMIN),	.....	SA(N).SB(N-NP)

The operation continues until a dc level has been found for all the time shift values TT(I).

Commonly, the point SA(NMIN-n), where n < NP, is equal to SA(1) before the whole range of time shifts has been computed. When this occurs, the role of each seismogram is reversed. SB(I) is now time shifted relative to SA(I) until the appropriate amount of time shifts are completed.

The dc levels stored in E(k) correspond to time shift values TT(I), ie. D21/VELHI,(D21/VELHI)+DELA, ... D21/VELLOW, where D21 is the distance between the two stations. These values have to be interpolated in subroutine INTPOL to produce a function X(KPER,I) (KPER is a period index) of dc levels for time shifts TSTEP(I) corresponding to the velocities of interest VSTEP(I).

3.7 The phase velocity values

The sequence of operations starting at the windowing of the seismogram (see section 3.3) is repeated for each period of interest and the dc levels are stored in subsequent columns of X(I,J) (where I is a period index and J is the dc level) which has a maximum size of 120 × 120. The maximum value of X is determined and set to 99, all other values of X are then scaled accordingly. X is a 2-D matrix of dc levels as a function of period and phase velocity (see figure 2 and 5). The phase velocity curve is determined from the most reasonable maxima of the contoured matrix.

3.8 Output

The output from the program is in the form of paper print-out and graphical display.

The print-out consists of:-

(a) The input parameters, seismograms and group velocity data.

(b) The 2-D matrix of the dc level of the cross-multiplied trace as a function of period and phase velocity.

(c) An extended print-out if required (set IMTRX=1) of many of the variables calculated during the computations.

The graphical output consists of:-

(a) The group velocity curves used in the formation of the group arrival time window.

(b) The two seismograms prior to analysis, SEIS1 and SEIS2.

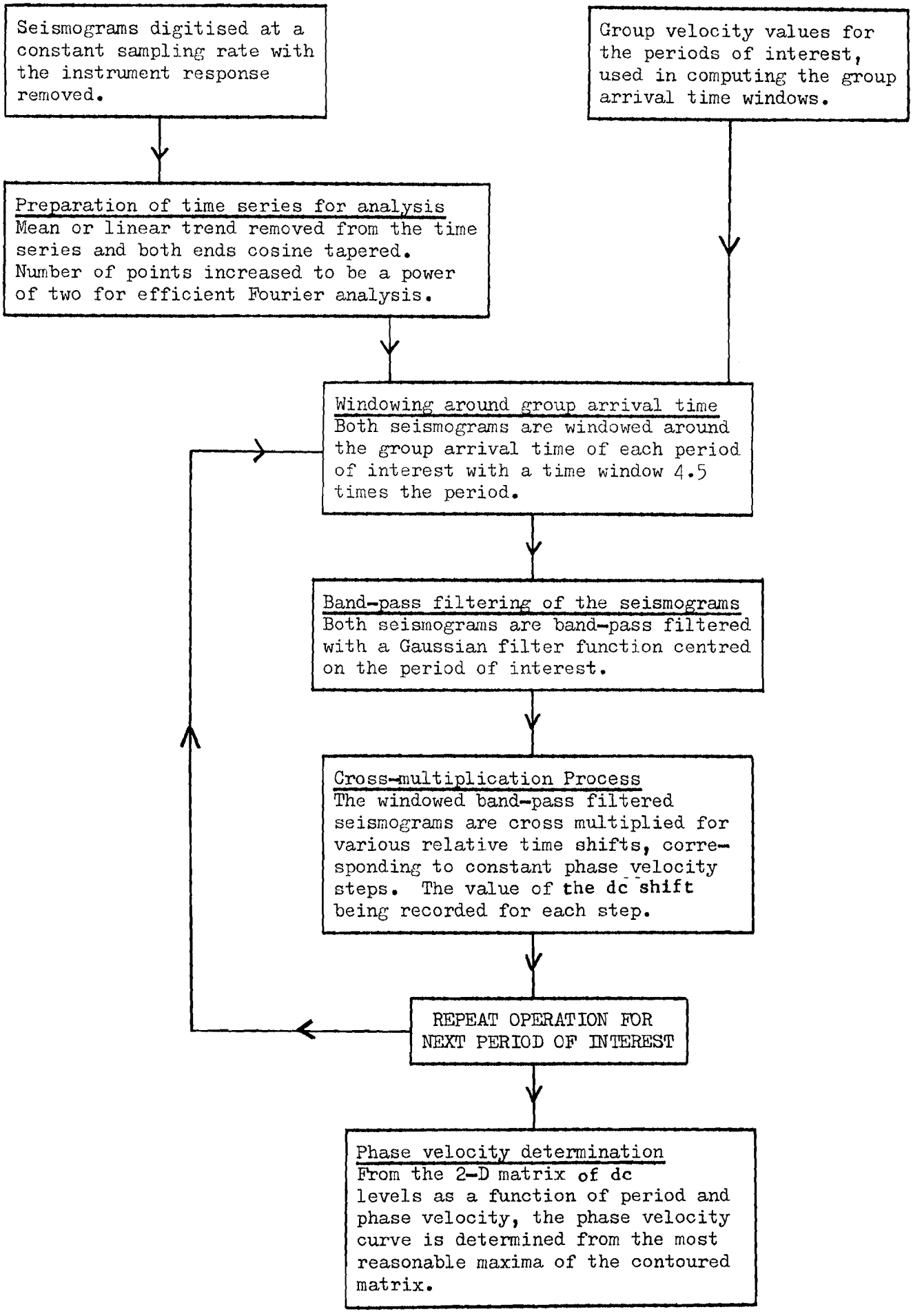
(c) For the largest, the centre and the smallest periods of interest, the windowed seismograms (WIND1,2), the band-pass filtered seismograms (FILT1,2) and the cross-multiplication function (XMULT) for a particular time shift (when SA(NMIN-n) = SA(1), n < NP) are graphed out.

(d) A contour plot (positive values only) of the dc levels for the phase velocity/period array.

4. PROGRAM SPECIFICATION

The program is written in FORTRAN IV. A storage of 384K is required. The output is in the form of paper print-out and SC 4060 graphics. The maximum number of digits in the time series is 1024. The maximum size of the phase velocity/period matrix that may be requested is 120 × 120.

5. FLOW DIAGRAM OF PROCEDURE



REFERENCES

1. S Bloch and A Hales: "New Techniques for the Determination of Surface Wave Phase Velocities". Bull Seism Soc Am, 58, 1021-1034 (1968)

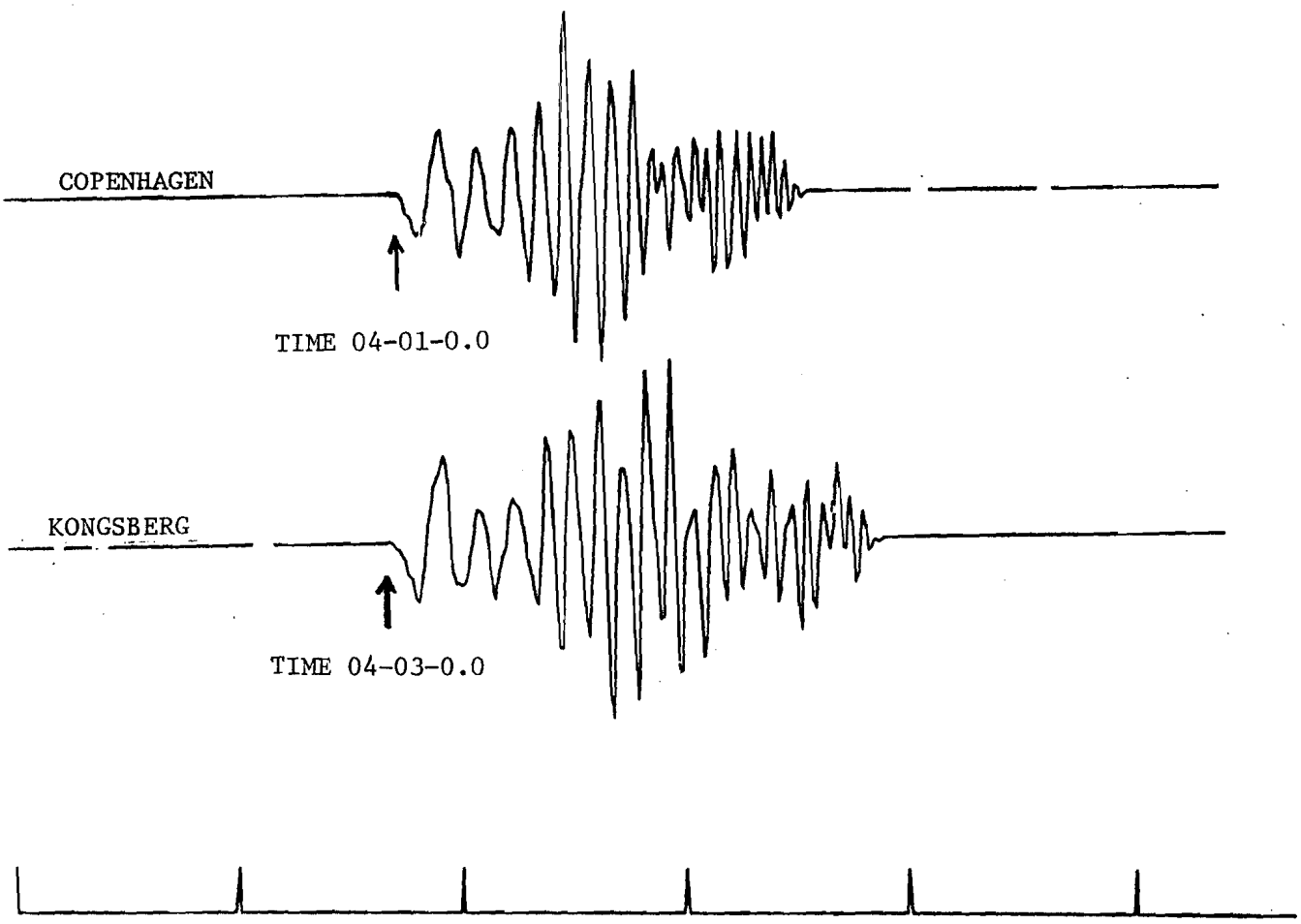
2. W Pilant and L Knopoff: "Inversion of Phase and Group Slowness Dispersion". J Geophys Res, 75, 2135-2136 (1970)

3. A Dziewonski and A Hales: "Numerical Analysis of Dispersed Seismic Waves". Methods in Computational Physics, 11, Edited by B A Bolt. Academic Press, pp39-85 (1972)

4. P Burton and C Blamey: "A Computer Program to Determine the Spectrum and a Dispersion Characteristic of a Transient Signal". AWRE Report 048/72 (1972)

5. A Douglas, J Hudson and C Blamey: "A Quantitative Evaluation of Seismic Signals at Teleseismic Distances. III: Computed P and Rayleigh Wave Seismograms". Geophys J Roy Astr Soc, 28, 385-410 (1972)

6. H Kanamori: "Attenuation of P Waves in the Upper and Lower Mantle". Bull Earth Res Inst, Tokyo, 45, 299-312 (1967)



TIME MARKS AT INTERVALS OF 200 SECONDS

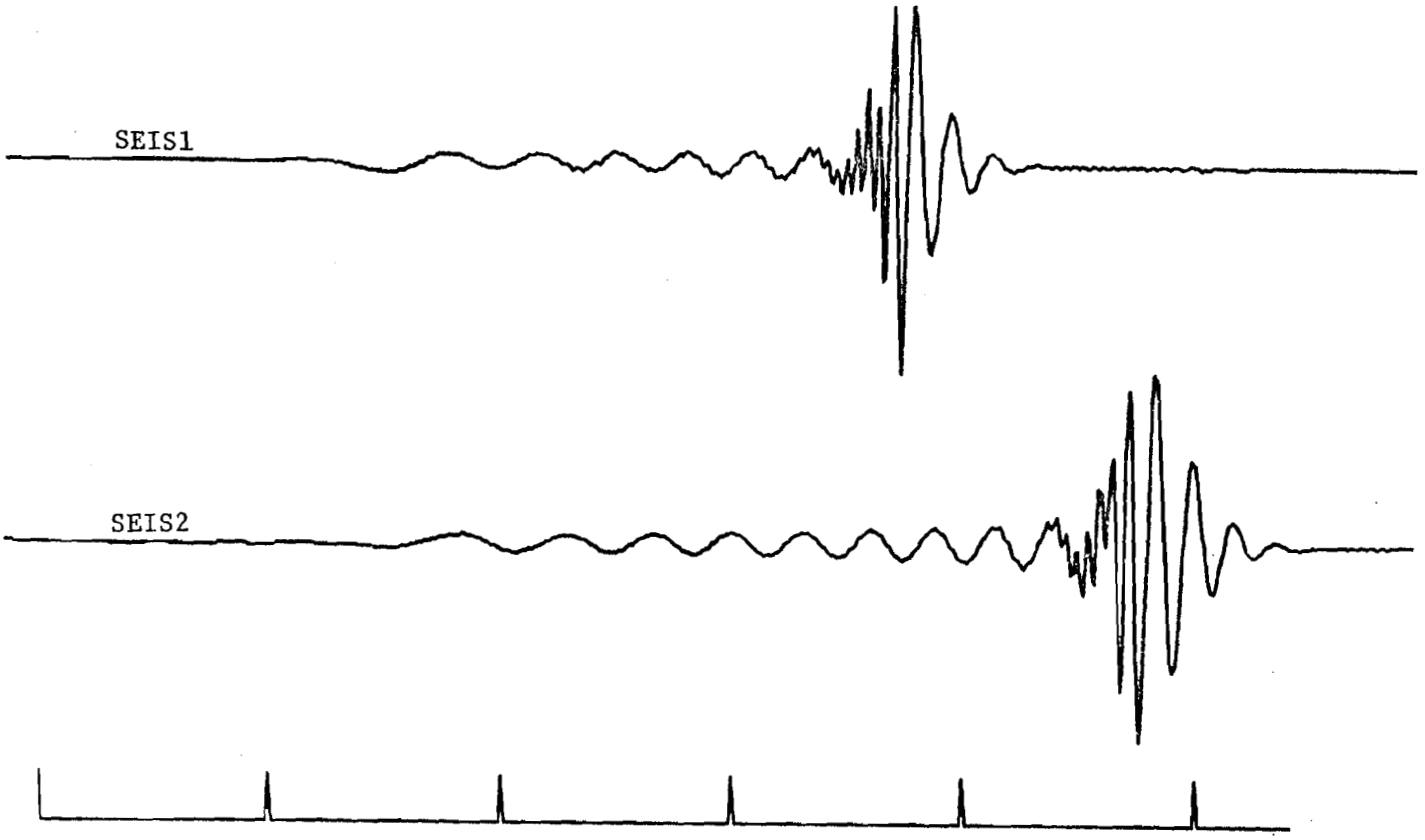
FIGURE 1. OBSERVED SEISMOGRAMS ON THE LONG PERIOD WWSSN INSTRUMENTS AT COPENHAGEN (DISTANCE = 16.10 DEGREES, AZIMUTH 345.2) AND KONGSBERG (DISTANCE = 20.36 DEGREES, AZIMUTH 345.0) OF THE FUNDAMENTAL MODE RAYLEIGH WAVES FROM AN EARTHQUAKE IN ALBANIA ON THE 16TH SEPTEMBER 1972. ORIGIN TIME 03-53-26.5, EPICENTRE 40.32N, 19.69E, Ms = 4.7, m_b = 5.1

MATRIX OF X(I)

PERIOD, s

PHASE VELOCITY, km s^{-1}	95.15	71.36	57.09	47.57	40.78	35.68	31.72	28.54	25.95	23.79	21.96	20.39	19.03
4.25	0.2 0.5	1.0 1.7	2.1 3.0	3.7 4.4	4.8 4.7	3.8 2.9	2.0 -1.2	-4.5 -13.9	-20.4 -5.5	-45.7 -40.9	-43.4 -7.3	26.5 38.0	29.6
4.23	0.2 0.5	1.1 1.7	2.2 3.1	4.0 4.9	5.4 5.5	4.8 4.0	3.1 0.1	-3.0 -12.2	-20.3 -13.9	-43.9 -50.5	-53.6 -17.8	17.6 33.8	30.4
4.21	0.2 0.6	1.1 1.8	2.3 3.3	4.4 5.3	6.0 6.2	5.8 5.1	4.2 1.4	-1.5 -9.8	-20.5 -24.7	-41.6 -59.6	-59.9 -26.9	7.8 28.4	30.5
4.19	0.2 0.6	1.1 1.8	2.4 3.4	4.7 5.6	6.5 6.9	6.8 6.2	5.2 2.8	0.0 -6.9	-20.3 -35.4	-39.3 -66.8	-61.0 -33.8	-2.4 22.1	29.5
4.17	0.2 0.6	1.1 1.8	2.5 3.6	4.9 6.0	7.0 7.5	7.7 7.2	6.2 4.2	1.8 -3.8	-18.9 -41.6	-38.3 -69.9	-54.0 -37.0	-13.2 15.3	27.0
4.15	0.2 0.6	1.1 1.9	2.6 3.7	5.0 6.4	7.5 8.2	8.4 8.1	7.3 5.5	3.8 -0.8	-15.5 -39.9	-39.1 -67.5	-37.9 -35.9	-24.1 8.4	22.8
4.13	0.2 0.6	1.1 1.9	2.6 3.8	5.2 6.7	8.0 8.8	9.1 8.9	8.3 6.9	5.9 2.2	-11.2 -34.6	-40.0 -62.6	-20.6 -33.8	-32.6 1.3	17.5
4.11	0.2 0.6	1.1 1.9	2.7 3.9	5.3 7.0	8.4 9.4	9.8 9.7	9.3 8.1	7.9 5.1	-6.2 -26.7	-40.4 -56.3	-6.2 -32.4	-37.2 -6.1	11.6
4.09	0.2 0.6	1.1 2.0	2.7 4.0	5.4 7.2	8.7 9.9	10.3 10.3	10.2 9.1	9.7 8.0	-1.0 -18.4	-38.9 -50.6	-1.7 -34.7	-35.0 -13.7	5.5
4.07	0.2 0.5	1.1 2.0	2.8 4.0	5.5 7.4	9.0 10.2	10.9 10.9	11.0 10.0	11.1 10.6	3.9 -11.6	-34.7 -47.3	-12.2 -43.0	-24.3 -21.5	-0.3
4.05	0.2 0.5	1.1 2.0	2.8 4.1	5.7 7.5	9.2 10.4	11.3 11.4	11.6 10.8	12.2 13.2	8.6 -4.8	-28.7 -43.8	-27.4 -51.7	-12.5 -28.7	-5.9
4.03	0.2 0.5	1.1 2.0	2.8 4.1	5.8 7.6	9.3 10.6	11.6 11.8	12.2 11.4	13.2 15.6	13.2 1.8	-20.9 -39.3	-42.9 -58.6	-3.0 -34.7	-11.3
4.01	0.2 0.5	1.1 2.0	2.8 4.1	5.8 7.6	9.4 10.7	11.8 12.0	12.5 12.0	14.3 17.8	17.5 8.4	-11.8 -32.1	-51.1 -59.1	-3.1 -39.0	-16.4
3.99	0.2 0.5	1.1 2.0	2.8 4.1	5.7 7.7	9.4 10.7	11.8 12.0	12.7 12.6	15.4 19.9	21.4 14.9	-1.7 -21.7	-48.4 -51.8	-14.3 -41.1	-21.1
3.97	0.2 0.5	1.1 2.0	2.8 4.1	5.6 7.6	9.3 10.7	11.7 11.9	12.6 13.0	16.4 21.7	25.0 21.4	8.8 -9.6	-39.2 -40.5	-28.5 -41.7	-24.9
3.95	0.2 0.5	1.1 1.9	2.8 4.0	5.5 7.5	9.2 10.6	11.5 11.7	12.4 13.2	17.1 23.0	28.3 27.7	19.5 3.4	-24.8 -27.0	-41.7 -40.8	-27.5
3.93	0.2 0.5	1.0 1.9	2.7 3.9	5.4 7.4	9.0 10.2	11.2 11.4	12.1 13.1	17.4 23.8	31.3 34.1	29.4 16.1	-7.9 -14.8	-45.3 -39.0	-28.2
3.91	0.2 0.5	1.0 1.9	2.7 3.8	5.3 7.1	8.6 9.7	10.8 10.9	11.6 12.6	17.2 24.0	33.8 40.1	38.6 28.2	10.1 -3.7	-40.6 -36.3	-27.4
3.89	0.2 0.5	1.0 1.8	2.6 3.7	5.2 6.9	8.2 9.2	10.3 10.3	11.0 11.9	16.6 23.7	35.6 45.1	46.8 39.8	27.8 6.8	-30.5 -32.3	-25.7
3.87	0.2 0.5	1.0 1.8	2.6 3.6	5.0 6.5	7.8 8.6	9.6 10.2	11.0 15.8	23.0 36.4	48.8 54.1	50.7 43.4	17.0 -16.8	-27.1 -23.6	
3.85	0.2 0.4	0.9 1.7	2.5 3.5	4.7 6.2	7.3 8.0	8.7 8.6	9.4 10.2	14.8 22.0	36.0 50.3	60.3 60.9	53.9 27.1	-2.7 -20.4	-22.2
3.83	0.2 0.4	0.9 1.7	2.3 3.3	4.4 5.8	6.7 7.4	7.8 7.4	8.4 9.2	13.5 20.6	34.6 50.4	65.0 69.6	61.9 37.4	12.1 -12.4	-20.2
3.81	0.2 0.4	0.8 1.6	2.2 3.1	4.0 5.3	6.0 6.6	6.7 6.2	7.2 8.1	11.9 18.6	32.4 49.1	67.8 76.3	68.5 47.9	26.7 -3.3	-16.9
3.79	0.2 0.4	0.8 1.5	2.1 2.9	3.7 4.8	5.3 5.6	5.6 5.0	5.7 6.7	10.1 16.1	29.6 47.1	68.3 80.0	76.6 60.0	39.7 7.0	-11.3
3.77	0.2 0.4	0.8 1.5	2.0 2.7	3.4 4.2	4.5 4.4	4.4 3.7	4.0 5.1	8.0 13.0	26.3 44.4	66.4 80.9	84.8 72.0	50.9 18.0	-3.6
3.75	0.2 0.3	0.7 1.4	1.9 2.4	3.0 3.6	3.6 3.2	3.2 2.4	2.2 3.3	5.7 9.6	22.4 40.5	62.6 79.1	90.2 81.5	60.1 29.1	5.1
3.73	0.1 0.3	0.7 1.3	1.8 2.2	2.6 3.0	2.7 2.0	1.9 1.0	0.6 1.4	3.4 5.9	17.9 35.5	57.4 75.1	90.1 85.8	67.6 39.8	13.7
3.71	0.1 0.3	0.6 1.2	1.6 1.9	2.1 2.3	1.7 1.1	0.5 -0.7	-0.7 -0.4	0.9 2.1	12.8 29.2	51.1 69.2	82.3 83.1	73.4 49.4	21.3
3.69	0.1 0.3	0.6 1.1	1.4 1.6	1.6 1.6	0.8 0.3	-0.9 -2.3	-2.0 -2.3	-1.6 -1.6	7.2 21.6	42.9 60.8	71.0 76.4	76.5 57.0	28.6
3.67	0.1 0.3	0.5 1.0	1.2 1.3	1.1 0.9	-0.0 -0.8	-2.3 -3.9	-3.5 -4.0	-4.1 -5.1	0.9 12.8	32.7 49.7	58.2 67.8	76.3 61.8	35.3
3.65	0.1 0.2	0.4 0.8	1.1 1.0	0.6 0.2	-0.7 -2.2	-3.7 -5.1	-5.3 -5.5	-6.6 -7.9	-6.1 3.2	19.6 35.0	48.8 60.9	72.1 62.5	42.3
3.63	0.1 0.2	0.4 0.7	0.9 0.7	0.2 -0.6	-1.3 -3.7	-5.0 -6.3	-7.2 -7.0	-9.0 -10.7	-12.8 -6.9	5.6 19.2	38.0 52.4	64.6 60.2	47.5
3.61	0.1 0.2	0.4 0.6	0.7 0.3	-0.2 -1.3	-2.1 -5.0	-6.1 -7.4	-9.0 -8.5	-11.2 -13.8	-18.3 -16.9	-8.1 3.9	24.1 40.8	54.5 55.3	49.8
3.59	0.1 0.1	0.3 0.5	0.5 0.0	-0.8 -2.1	-3.2 -5.9	-6.7 -8.9	-10.2 -10.7	-13.4 -18.4	-22.3 -27.4	-19.7 -7.2	2.8 23.8	43.0 49.2	47.3
3.57	0.1 0.1	0.2 0.3	0.3 -0.3	-1.4 -2.8	-4.5 -6.5	-7.2 -10.2	-10.8 -12.2	-14.7 -22.2	-23.5 -34.7	-29.0 -17.5	-21.1 3.5	29.3 41.0	41.4
3.55	0.1 0.1	0.1 0.2	0.0 -0.7	-2.0 -3.4	-5.6 -7.0	-7.7 -10.9	-10.6 -12.5	-14.6 -23.7	-21.4 -36.4	-35.5 -29.0	-44.2 -17.9	13.1 30.4	33.3
3.53	0.0 0.1	0.0 0.1	-0.2 -1.0	-2.4 -4.1	-6.2 -7.7	-8.9 -10.3	-9.0 -9.7	-11.6 -19.5	-13.7 -26.1	-38.3 -46.5	-63.9 -40.8	-7.0 17.2	25.0
3.51	0.0 0.0	0.0 -0.1	-0.4 -1.3	-2.8 -4.7	-6.5 -8.5	-10.1 -9.3	-7.3 -6.3	-8.0 -13.7	-5.8 -13.9	-39.6 -63.5	-74.2 -56.9	-27.5 1.8	15.7
3.49	0.0 -0.0	-0.0 -0.2	-0.6 -1.7	-3.1 -5.3	-6.3 -9.1	-11.0 -8.7	-6.1 -4.0	-5.6 -9.3	-1.8 -6.7	-41.0 -75.8	-71.4 -60.6	-45.4 -15.0	5.2
3.47	0.0 -0.0	-0.1 -0.4	-0.8 -1.9	-3.6 -6.1	-5.8 -9.4	-10.7 -9.5	-7.0 -5.9	-7.5 -11.9	-9.4 -18.1	-45.7 -75.0	-44.4 -38.2	-56.0 -34.6	-8.0
3.45	-0.0 -0.1	-0.2 -0.5	-1.0 -2.2	-4.0 -6.5	-5.4 -9.6	-10.0 -10.5	-8.6 -9.0	-10.6 -16.1	-20.0 -34.2	-50.6 -68.8	-15.5 -13.1	-63.1 -48.7	-20.2
3.43	-0.0 -0.1	-0.3 -0.6	-1.2 -2.6	-4.5 -6.4	-5.4 -9.6	-9.4 -11.1	-9.8 -11.3	-12.8 -18.6	-27.7 -46.1	-53.9 -62.0	-1.5 -2.4	-68.9 -50.3	-28.6

FIGURE 2. 2-D MATRIX OF NORMALISED DC LEVELS AS A FUNCTION OF PERIOD AND PHASE VELOCITY FOR THE INTERSTATION PHASE VELOCITY DETERMINATION BETWEEN THE WWSSN STATIONS OF COPENHAGEN AND KONGSBERG. THE DOTS REPRESENT THE COMPUTED PHASE VELOCITY CURVE

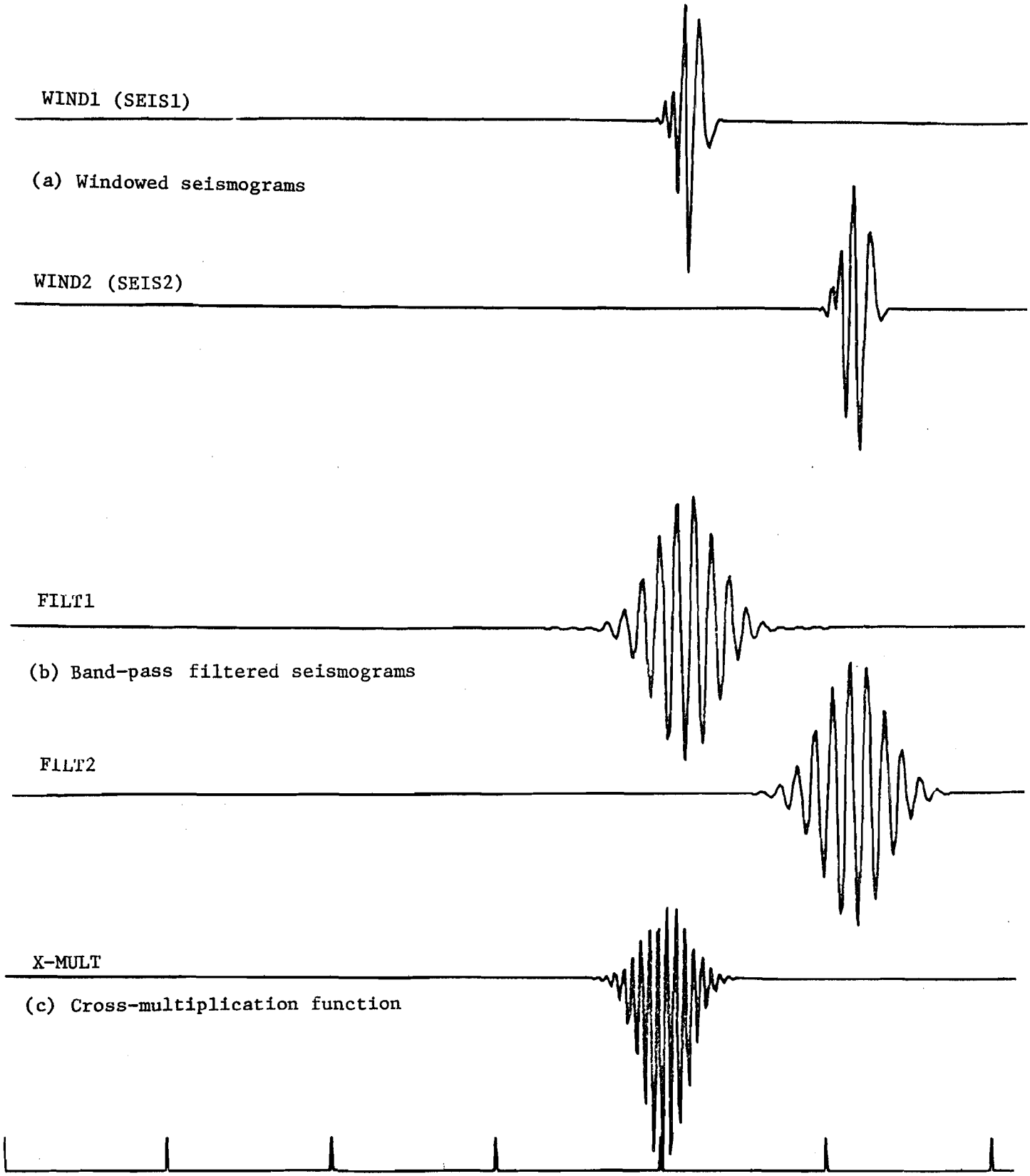


TIME MARKS AT INTERVALS OF 100 SECONDS

FIGURE 3. SYNTHETIC, FUNDAMENTAL MODE, RAYLEIGH WAVE SEISMOGRAMS COMPUTED (SEE DOUGLAS, HUDSON AND BLAMEY [5]) FOR A 25000 KTON EXPLOSION IN AIR AT EPICENTRAL DISTANCES OF 30.0 (SEIS1) AND 40.0 (SEIS2) DEGREES, RECORDED BY STANDARD WWSSN LONG PERIOD INSTRUMENTS. THE DETAILS OF THE STANDARD CONTINENTAL STRUCTURE USED IN THE COMPUTATION IS SHOWN BELOW

STANDARD CONTINENTAL CRUST (KANAMORI [6])

	P WAVE VELOCITY ($km\ s^{-1}$)	S WAVE VELOCITY ($km\ s^{-1}$)	DENSITY ($g\ cm^{-3}$)	THICKNESS (km)
1st LAYER	6.10	3.50	2.70	11.0
2nd LAYER	6.40	3.68	2.90	9.0
3rd LAYER	6.70	3.94	2.90	18.0
HALF SPACE	8.15	4.75	3.30	∞



TIME MARKS AT INTERVALS OF 100 SECONDS

FIGURE 4. (a) SEISMOGRAMS WINDOWED AROUND THE GROUP ARRIVAL TIME OF 10.5 SECOND PERIOD WITH A WINDOW 4.5 × 10.5 s IN LENGTH AND THEN SYMMETRICALLY COSINE TAPERED
(b) BAND-PASS FILTERED SEISMOGRAMS, FILTERED IN THE FREQUENCY DOMAIN BY A GAUSSIAN FILTER CENTRED ON 10.5 s PERIOD
(c) CROSS-MULTIPLICATION FUNCTION OF THE TWO FILTERED SEISMOGRAMS FOR A PARTICULAR TIME SHIFT

MATRIX OF X(I)

PERIOD, s

PHASE VELOCITY, km s^{-1}	56.89	39.38	31.12	24.38	20.48	17.66	15.52	13.84	12.49	11.38	10.45
4.50	0.1 -0.1 -0.3 -0.1 0.4	0.0 -0.6	1.1 -0.4	1.2 -0.8	0.2	0.3 -0.3	0.3 -0.1	0.1 -0.0	0.0	0.0	0.0
4.48	0.1 -0.0 -0.3 -0.2 0.4	0.2 -0.5	1.2 -0.9	1.0 -0.3	-0.2	0.6 -0.3	0.4 -0.1	0.1 -0.0	-0.0	0.0	0.0
4.46	0.1 -0.0 -0.3 -0.3 0.3	0.3 -0.8	1.2 -1.0	0.6	0.1 -0.7	0.9 -0.1	0.3 -0.1	0.1	0.0 -0.0	0.0	0.0
4.44	0.1 0.0 -0.2 -0.3 0.2	0.5 -0.6	1.1 -0.6	0.1	0.7 -1.1	1.0 -0.5	0.2 -0.0	-0.0	0.1	0.0	0.0
4.42	0.1 0.1 -0.2 -0.3 0.1	0.6 -0.8	0.8 -0.3	-0.4	1.2 -0.8	0.9 -0.2	0.0	0.2 -0.1	0.2	0.0	0.0
4.40	0.1 0.1 -0.2 -0.3 0.0	0.7 -0.7	0.5	0.2 -1.0	1.6 -0.8	0.6 -0.0	-0.2	0.3 -0.1	0.2	-0.0 -0.0	0.0
4.38	0.1 0.1 -0.1 -0.3 -0.1	0.7 -0.5	0.2	0.8 -1.6	1.8 -1.0	0.1	0.4 -0.4	0.4 -0.1	0.1	0.0 -0.1	0.1
4.36	0.1 0.2 -0.1 -0.4 -0.2	0.7 -0.3	-0.2	1.3 -0.2	1.6 -0.5	-0.5	0.8 -0.0	0.4 -0.1	-0.0	0.1 -0.0	0.0
4.34	0.1 0.2 0.0 -0.3 -0.3	0.6 0.1	-0.7	1.6 -1.7	1.2	0.3 -1.1	1.1 -0.4	0.3	0.1 -0.2	0.2 -0.1	0.0
4.32	0.1 0.2 0.1 -0.3 -0.4	0.5 0.4	-1.0	1.8 -1.6	0.4	1.0 -0.9	1.1 -0.3	0.0	0.3 -0.2	0.2 -0.0	-0.0
4.30	0.1 0.2 0.1 -0.2 -0.5	0.3 0.7	-1.0	1.8 -1.0	-0.5	1.8 -0.2	0.8	0.1 -0.3	0.4 -0.1	0.1	0.1 0.0
4.28	0.1 0.2 0.2 -0.2 -0.2	0.1 1.0	-1.3	1.5 -0.2	-1.4	2.2 -1.2	0.2	0.5 -0.4	0.5 -0.1	-0.1	0.2 -0.0
4.26	0.1 0.2 0.2 -0.1 -0.5	-0.1 1.1	-1.5	1.0	0.7 -2.3	2.2 -0.7	-0.5	1.0 -0.3	0.4	0.1 -0.2	0.2 -0.0
4.24	0.1 0.2 0.3 0.0 -0.5	-0.3 1.2	-1.2	0.3	1.6 -0.1	1.7	0.3 -1.2	1.2 -0.3	0.1	0.4 -0.0	0.1 0.1
4.22	0.1 0.2 0.3 0.1 -0.4	-0.5 1.2	-0.8	-0.5	2.4 -2.5	0.7	1.4 -1.3	1.1 -0.1	-0.4	0.6 -0.2	-0.0 0.2
4.20	0.1 0.2 0.3 0.2 -0.3	-0.6 1.0	-0.1	-1.3	2.8 -2.0	-0.5	2.3 -0.8	0.6	0.5 -0.4	0.6 -0.0	-0.3 0.2
4.18	0.1 0.2 0.3 0.3 -0.2	-0.8 0.7	0.5	-1.6	2.7 -0.8	-1.9	2.8 -1.1	-0.3	1.1 -0.6	0.3	0.3 -0.0 0.1
4.16	0.1 0.2 0.3 0.3 -0.1	-0.7 0.4	1.2	-0.8	2.3 0.6	-2.4	2.6 -0.3	-1.2	1.5 -0.5	-0.1	0.6 -0.2 -0.1
4.14	0.0 0.2 0.3 0.4 0.1	-0.8 -0.0	1.8	-2.3	1.2 2.2	-1.9	1.6	1.3 -0.9	1.3	0.2 -0.7	0.7 -0.0 -0.1
4.12	0.0 0.1 0.3 0.4 0.2	-0.7 -0.5	2.2	-2.0	-0.2 3.5	-3.1	-0.1	2.5 -1.0	0.7	0.8 -0.2	0.4 0.3 -0.1
4.10	0.0 0.1 0.2 0.4 0.4	-0.6 -0.8	2.3	-1.4	-1.6 4.0	-1.4	-2.0	3.2 -1.0	-0.6	1.5 -0.5	-0.2 0.7 -0.3
4.08	-0.0 0.0 0.2 0.4 0.5	-0.3 -1.3	2.1	-0.2	-3.3 3.9	0.0	-3.4	3.0 0.3	-1.5	1.6 -0.0	-0.7 0.7 0.1
4.06	-0.0 -0.0 0.1 0.4 0.6	-0.1 -0.7	1.5	1.1	-1.2 2.8	2.4	-4.3	1.7 1.7	-1.4	1.0 0.9	-0.7 0.3 0.6
4.04	-0.0 -0.0 0.1 0.3 0.6	0.2 -1.3	0.8	2.3	-3.6 0.7	4.3	-2.0	-0.4 3.2	-1.6	-0.3 1.8	-0.7 -0.6 0.9
4.02	-0.1 -0.1 -0.0 0.2 0.6	0.5 -1.4	0.0	3.2	-3.2 -1.6	5.4	-2.3	-2.8 3.6	-0.3	-1.9 1.9	0.2 -0.6 0.5
4.00	-0.1 -0.1 -0.1 0.1 0.8	0.7 -1.0	-0.9	3.6	-1.7 -3.8	5.2	0.9	-4.6 2.5	1.4	-0.5 1.0	1.4 -0.8 -0.3
3.98	-0.1 -0.2 -0.2 0.0 0.5	0.9 -0.5	-1.6	3.4	0.5 -3.2	3.2	3.9	-2.5 0.4	3.5	-2.1 -0.8	2.2 -0.3 -0.7
3.96	-0.1 -0.2 -0.2 -0.1 0.3	1.0 0.0	-2.2	2.5	2.8 -3.7	0.2	6.2	-3.5 -2.7	4.2	-0.1 -2.3	1.8 1.2 -0.7
3.94	-0.1 -0.2 -0.3 -0.2 0.2	1.0 0.7	-2.4	1.2	4.6 -4.5	-3.3	6.8	0.3 -2.5	3.2	2.0 -0.2	0.2 2.2 -0.7
3.92	-0.1 -0.2 -0.3 -0.3 -0.0	0.8 1.2	-2.6	-0.5	5.7 -1.9	-7.0	5.1	3.9 -2.8	0.4	4.1 -1.9	-2.2 2.2 1.1
3.90	-0.1 -0.2 -0.3 -0.4 -0.2	0.6 1.6	-1.9	-2.8	5.6 1.5	-1.5	1.3	7.0 -2.6	-2.5	4.4 1.1	-0.7 0.4 2.6
3.88	-0.0 -0.2 -0.3 -0.4 -0.4	0.3 1.8	-0.7	-4.1	4.0 5.2	-6.9	-3.6	8.0 0.3	-2.7	2.3 4.0	-2.1 -2.2 2.3
3.86	-0.1 -0.2 -0.3 -0.0 -0.5	-0.0 1.3	0.5	-2.7	1.3 8.0	-3.3	-7.8	5.8 4.8	-2.9	-1.3 5.3	0.7 -1.3 0.1
3.84	-0.1 -0.2 -0.3 -0.5 -0.7	-0.4 1.3	1.9	-4.6	-1.9 8.4	1.6	-8.0	0.9 8.4	-2.0	-5.9 3.6	4.1 -2.6 -3.1
3.82	-0.1 -0.2 -0.3 -0.4 -0.3	-0.6 0.9	3.1	-3.2	-4.9 6.5	6.7	-6.4	-5.5 8.5	3.5	-2.0 -1.1	6.1 1.2 -0.7
3.80	-0.1 -0.1 -0.2 -0.3 -0.7	-0.9 0.2	[illegible]	-1.0	-3.4 2.7	10.5	-3.2	-7.0 3.8	8.3	-3.2 -5.6	3.9 5.1 -2.3
3.78	-0.1 -0.1 -0.1 -0.2 -0.5	-1.1 -0.5	3.5	1.6	-7.5 -2.3	11.1	4.4	-10.1 -3.8	9.4	2.3 -0.9	-1.5 6.4 2.5
3.76	-0.1 -0.1 -0.0 -0.1 -0.4	-1.0 -1.2	2.8	4.1	-5.6 -7.7	7.6	10.9	-6.2 -7.1	6.4	8.5 -4.2	-7.3 2.8 6.9
3.74	-0.1 -0.0 0.0 0.0 -0.2	-0.8 -1.6	1.6	5.8	-2.0 -9.1	0.9	14.0	1.5 -5.8	-1.8	10.7 3.8	-3.3 -4.6 6.1
3.72	-0.0 0.0 0.1 0.2 0.4	-0.7 -0.6	0.1	6.2	2.6 -10.4	-6.8	11.9	11.0 -7.2	-9.0	6.2 10.6	-2.7 -2.8 -0.6
3.70	-0.0 0.1 0.2 0.3 0.3	-0.3 -2.2	-1.7	5.1	6.8 -6.7	-12.2	4.2	16.5 0.3	-10.6	-3.3 11.8	7.5 -6.1 -7.9
3.68	0.0 0.1 0.3 0.4 0.5	0.1 -1.6	-3.1	2.8	9.5 0.1	-14.6	-6.2	14.7 10.9	-6.4	-11.4 4.2	13.3 3.8 -4.4
3.66	0.0 0.2 0.3 0.4 0.6	0.6 -0.9	-2.4	-0.5	9.8 7.5	-9.7	-16.6	5.9 17.7	2.1	-6.2 -8.7	10.0 12.6 -1.2
3.64	0.1 0.2 0.3 0.5 0.7	0.9 0.1	-4.0	-4.1	7.1 13.0	-1.9	-8.4	-7.4 15.4	15.4	-6.7 -9.4	-2.3 14.2 12.6
3.62	0.1 0.2 0.4 0.5 0.7	1.2 1.0	-3.1	-6.9	2.0 14.5	8.8	-13.9	-18.9 3.6	20.8	9.1 -11.6	-16.2 2.0 16.8
3.60	0.1 0.2 0.3 0.4 0.6	1.2 1.8	-1.3	-6.7	-4.4 11.1	17.4	-0.9	-16.6 -13.9	12.9	20.2 3.4	-12.2 -14.5 5.0
3.58	0.1 0.2 0.3 0.3 0.5	1.1 2.3	1.0	-6.2	-10.8 3.2	19.6	14.2	-10.7 -13.5	-4.4	18.9 19.5	-1.0 -7.6 -12.5
3.56	0.1 0.2 0.2 0.2 0.3	0.8 2.4	3.2	-3.4	-1.4 -6.6	13.4	23.9	6.8 -14.1	-21.1	3.1 23.7	18.5 -1.7 -11.5
3.54	0.1 0.2 0.1 0.0 0.0	0.4 2.0	4.7	1.1	-10.6 -14.8 0.6	23.0	23.1	-5.1	-20.5 -18.8	8.3 26.4	17.9 -6.0
3.52	0.1 0.2 0.0 -0.1 -0.3	-0.2 1.2	5.2	5.5	-4.8 -14.8 -14.1	10.4	28.4	17.7	-8.3 -14.6	-18.2 10.9	28.6 21.5
3.50	0.1 0.1 -0.0 -0.3 -0.5	-0.7 0.1	4.3	8.6	3.1 -13.6 -11.9	-9.3	19.1	31.6	13.6 -17.6	-17.8 -17.3	11.9 31.6
3.48	0.1 0.1 -0.1 -0.4 -0.7	-1.0 -1.1	2.2	9.1	10.7 -2.8 -20.3	-26.3	-2.2	25.9	33.7 12.5	-17.8 -17.7	-23.9 7.0
3.46	0.1 0.0 -0.2 -0.5 -0.7	-0.9 -1.6	-0.4	6.7	14.9 10.0 -10.3	-11.0	-27.1	1.9	30.8 36.3	13.4 -17.7	-8.3 -29.6
3.44	0.1 -0.0 -0.3 -0.4 -0.7	-1.2 -2.3	-2.7	2.0	13.9 19.6 8.2	-17.9	-16.4	-24.7 3.6	31.3 39.0	21.6	-9.5 -27.5
3.42	0.1 -0.1 -0.3 -0.4 -0.6	-1.0 -2.5	-4.7	-3.7	7.6 21.3 24.2	6.4	-22.4	-32.6 -29.5	-0.1 31.8	46.5 35.6	5.7
3.40	0.1 -0.1 -0.2 -0.3 -0.3	-0.6 -1.9	-5.2	-8.5	-2.1 13.4 29.2	29.5	8.2	-21.8 -42.3	-37.4 -9.3	26.2 49.7	49.4
3.38	0.0 -0.2 -0.3 -0.2 -0.0	-0.0 -0.7	-4.4	-9.9	-11.2 -1.4 19.2	37.0	36.9	13.4 -17.9	-33.3 -37.8	-25.6 9.9	40.8
3.36	0.0 -0.2 -0.2 -0.1 0.2	0.6 0.6	-2.1	-8.8	-2.6 -16.6 -1.9	23.0	43.7	44.4 22.6	-9.7 -34.3	-25.1 -43.8	-18.2
3.34	-0.0 -0.2 -0.2 0.1 0.5	1.1 1.8	1.2	-4.0	-15.3 -7.9 -19.4	-6.7	21.5	45.5 51.8	36.8 8.7	-24.8 -33.8	-49.5
3.32	-0.0 -0.1 -0.1 0.3 0.7	1.3 2.6	4.3	2.7	-5.7 -22.6 -30.9	-32.1	-16.3	11.8 39.5	54.5 55.3	40.0 14.7	-22.3
3.30	-0.1 -0.2 -0.0 0.4 0.7	1.3 2.7	6.1	8.8	6.1 -6.5 -25.1	-39.3	-45.7	-37.2 -8.1	21.2 48.4	66.2 69.4	55.0
3.28	-0.1 -0.2 0.1 0.4 0.7	1.0 2.1	6.0	11.6	16.1 12.5 -0.7	-22.4	-42.6	-45.2 -52.6	-35.9 -12.6	18.5 49.3	69.1
3.26	-0.1 -0.1 0.2 0.5 0.5	0.5 1.0	3.9	10.0	19.4 26.0 25.3	14.0	-5.7	-28.0 -48.6	-60.4 -36.1	-55.0 -32.6	-7.7
3.24	-0.1 -0.1 0.3 0.4 0.2	-0.2 -0.5	0.4	4.2	14.0 26.1 37.5	42.9	39.0	24.6 1.2	-22.6 -48.9	-65.0 -31.2	-52.9
3.22	-0.1 -0.0 0.3 0.3 -0.1	-0.8 -1.9	-3.3	-3.5	1.7 11.8 27.0	43.7	57.1	61.8 56.4	43.3 25.8	6.8	-11.7 -33.9
3.20	-0.1 0.0 0.3 0.1 -0.4	-1.1 -2.1	-5.4	-9.9	-11.4 -9.6 -1.1	13.9	32.7	48.1 61.1	68.4 75.1	77.5 76.2	67.0
3.18	-0.1 0.1 0.3 -0.0 -0.6	-1.2 -2.5	-6.1	-11.5	-8.2 -24.6 -30.2	-27.8	-17.5	-8.2 6.6	20.6 36.5	52.5 67.5	77.9
3.16	-0.1 0.2 0.2 -0.2 -0.3	-1.1 -1.9	-4.6	-8.7	-18.5 -26.2 -37.7	-47.8	-54.1	-12.7 -46.2	-58.8 -47.9	-34.8 -30.5	-25.3
3.14	-0.1 0.2 0.1 -0.3 -0.6	-0.6 -0.5	-0.8	-1.9	-5.4 -11.8 -22.0	-32.7	-41.0	-58.7 -62.7	-63.1 -68.2	-72.5 -55.5	-87.7
3.12	-0.1 0.2 0.0 -0.4 -0.5	0.0 1.1	3.3	6.6	9.6 11.9 11.3	9.4	7.9	-0.2 -2.6	-5.4 -9.5	-6.4 -9.3	-11.4
3.10	-0.0 0.2 -0.1 -0.3 -0.2	0.7 2.3	6.1	12.1	19.9 28.9 37.5	45.9	52.8	58.2 61.8	65.3 70.9	77.0 84.3	88.2
3.08	0.0 0.2 -0.2 -0.3 0.2	1.1 2.7	6.4	11.7	19.3 27.9 37.4	46.8	54.5	59.6 60.7	59.0 58.2	58.0 57.5	49.0
3.06	0.0 0.2 -0.2 -0.2 0.4	1.3 2.2	4.0	5.3	7.7 9.1 10.1	10.0	7.3	1.1 -5.1	-17.6 -33.9	-42.4 -49.1	-53.4
3.04	0.1 0.1 -0.3 -0.0 0.6	1.0 1.0	-0.1	-3.8	-8.4 -15.3 -25.0	-37.3	-48.6	-16.0 -29.1	-29.1 -43.8	-48.4 -65.8	-52.6
3.02	0.1 0.1 -0.1 0.1 0.7	0.5 -0.6	-3.7	-10.0	-19.4 -24.7 -1.4	-47.3	-50.2	-53.7 -44.4	-31.3 -16.0	3.0 24.8	47.3
3.00	0.0 0.0 0.0 0.0 0.0	0.0 0.0	0.0	0.0	0.0 0.0 0.0	0.0	0.0	0.0 0.0	0.0 0.0	0.0 0.0	0.0

FIGURE 5. 2-D MATRIX OF NORMALISED DC LEVELS AS A FUNCTION OF PERIOD AND PHASE VELOCITY FOR THE SYNTHETIC SEISMOGRAMS. BLACK DOTS REPRESENT THE COMPUTED PHASE VELOCITY, THE DASHED LINE REPRESENTS THE ACTUAL PHASE VELOCITY CURVE. PROCESSING ERROR FOR ALL PERIODS IS BETTER THAN ∓ 0.015 km/s

APPENDIX A

PROGRAM LISTING

```
C
C                    INTERSTATION PHASE VELOCITY PROGRAM (IPV)
C                    -----------------------------------------
C
C
C        THE PROGRAM COMPUTES THE PHASE VELOCITY OF A DISPERSED SEISMIC
C     SIGNAL BETWEEN TWO STATIONS FROM A SOURCE WHICH IS ON OR NEAR THE
C     INTERSTATION GREAT CIRCLE. THE SEISMOGRAMS,WHICH ARE REPRESENTED
C     IN THE PROGRAM AS DIGITS SAMPLED AT EQUAL INTERVALS,ARE INITIALLY
C     WINDOWED AROUND THE GROUP ARRIVAL TIME OF THE VARIOUS PERIODS OF
C     INTEREST IN ORDER TO ELIMINATE EXTRANEOUS NOISE. BOTH SEISMOGRAMS
C     ARE THEN PASSED THROUGH A NARROW BAND PASS DIGITAL FILTER CENTRED
C     AT VARIOUS PERIODS AND THEIR CROSS PRODUCT FORMED FOR A NUMBER OF
C     TIME SHIFTS CORRESPONDING TO CONSTANT PHASE VELOCITY STEPS. THE
C     AVERAGE OF THE RESULTANT TIME SERIES IS A MAXIMUM WHEN THE TWO
C     SIGNALS ARE IN PHASE. THE PHASE VELOCITY DISPERSION IS DETERMINED
C     FROM A CONTOURED MATRIX CONSISTING OF AVERAGES AS A FUNCTION OF
C     PHASE VELOCITY AND PERIOD.
C
C
C     GENERAL REFERENCE ---'NEW TECHNIQUES FOR THE DETERMINATION OF
C     -----------------           SURFACE WAVE PHASE VELOCITIES'   S.BLOCH AND
C     A.L.HALES,  BULL.SEISM.SOC.AM. 58,  1021-1034.
C                                     --
C
C               * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C     NOTES FOR USERS.
C     ----------------
C
C     1)     ENSURE THAT GROUP VELOCITY VALUES ARE AT THE SAME PERIODS
C            AND OVER THE SAME RANGE IN EACH RECORD.
C     2)     SELECT THE RANGE OF PERIODS OVER WHICH THE GROUP VELOCITY
C            VALUES LOOK SENSIBLE.
C     3)     FIRST SEISMOGRAM,STATION ETC. IS NEARER THE SOURCE, THE
C            SECOND SEISMOGRAM,STATION ETC. IS FURTHER FROM THE SOURCE.
C
C
C     INPUT DATA CARDS
C     ----------------
C
C     CARD 1
C     ------  FORMAT (10A8)
C



C            TITLEA----TITLE FOR THE DATA SET (10A8)
C
C
C     CARD 2
C     ------  FORMAT (8A1,F12.5,I2,I2,F3.1,3X,8A1,F12.5,I2,I2,F3.1,16X,
C                    I2,I2,F3.1)
C
C            STNAM1----NAME OF THE RECORDING STATION NEARER THE SOURCE
C                      (8A1)
C
C            DELTA1----DISTANCE IN DEGREES BETWEEN STNAM1 AND THE SOURCE
C                      1 DEGREE = 111.1 KM  (F12.5)
C
C            MHGMT1----HOURS (I2)
C                                          GMT TIME OF FIRST SAMPLE IN THE
C            MNGMT1----MINUTES (I2)
C                                          FIRST SEISMOGRAM
C            SCGMT1----SECONDS (F3.1)
C
C            STNAM2----NAME OF RECORDING STATION FURTHER FROM
C                      THE SOURCE (8A1)
C
C            DELTA2----DISTANCE IN DEGREES BETWEEN STNAM2 AND SOURCE
C                      (F12.5)
C
C            MHGMT2----HOURS (I2)
C                                          GMT TIME OF FIRST SAMPLE IN THE
C            MNGMT2----MINUTES (I2)
C                                          SECOND SEISMOGRAM
C            SCGMT2----SECONDS (F3.1)
C
C            MHOUR ----HOURS (I2)
C                                          GMT ORIGIN
C            MIN   ----MINUTES (I2)
C                                          TIME
C            SEC   ----SECONDS (F3.1)
C
C
C     CARD 3
C     ------  FORMAT (4I10,3F10.5)
C
C            NSEIS1----NUMBER OF SAMPLES IN THE FIRST SEISMOGRAM, NOT
C                      MORE THAN 1024 (I10)
C
C
C            NSEIS2----NUMBER OF SAMPLES IN THE SECOND SEISMOGRAM, NOT
```

```
C                    MORE THAN 1024 (I10)
C
C          NGV   ---- NUMBER OF GROUP VELOCITY / PERIOD VALUES FOR THE
C                     TWO SEISMOGRAMS,  NOT MORE THAN 120 (I10)
C
C          NCOSTP----NUMBER OF POINTS IN COSINE TAPER OF BOTH ENDS OF
C                    SIGNAL, IF NCOSTP=0 NO COSINE TAPER APPLIED
C
C          DELA  ----INTERVAL BETWEEN SAMPLES IN SECONDS (F10.5)
C
C          VELHI ----HIGHEST PHASE VELOCITY OF INTEREST (F10.5)
C
C          VELLOW----LOWEST PHASE VELOCITY OF INTEREST (F10.5)
C
C
C     CARD 4
C     ------  FORMAT (1X,A8,1X,3I10,3I5)
C
C          TYPE  ----IF'TYPE'=MEAN, MEAN BASELINE REMOVED FROM SIGNALS
C                    IF'TYPE'=LEAST,LEAST SQUARES BASELINE IS REMOVED
C                    (A8)
C
C          NCZERO----'NCZERO' NUMBER OF ZEROS ADDED TO THE FRONT OF
C                    BOTH SEISMOGRAMS (I10)
C
C          IMTRX ----IF'IMTRX'=1, AN EXTENDED PRINT OUT IS OBTAINED.
C                    (I10)
C
C          IVSEIS----INVERT THE FIRST SIGNAL IF'IVSEIS'=1, THE SECOND
C                    IF'IVSEIS'=2 (I10)
C
C          NG1   ----IF NG1=1,GROUP VELOCITY CURVES AND ORIGINAL
C                    SEISMOGRAMS GRAPHED. (I5)
C
C          NG2   ----IF NG2=1, WINDOWED, BAND-PASS FILTERED AND CROSS-
C                    MULTIPLIED TRACES GRAPHED FOR THE LONGEST, CENTRE
C                    AND SHORTEST PERIODS.
C
C          NG3   ----IF NG3=1, CONTOUR PLOT OF POSITIVE VALUES OF 2-D
C                    MATRIX ARE GRAPHED.
C
C
C     CARD 5
C     ------  FORMAT (3F10.5)
C
C
C
C
C
C          BAND  ----RELATIVE BANDWIDTH OF GAUSSIAN FILTER (F10.5)
C
C          DWF   ----DECAY RATE OF GAUSSIAN FILTER FUNCTION (F10.5)
C
C          DV    ----THE VELOCITY INTERVAL BETWEEN PHASE VELOCITY
C                    VALUES, IF DV LEFT BLANK IT IS SET TO 0.02 KM/SEC
C                    (NOT MORE THAN 120 INTERVALS BETWEEN VELHI AND
C                    VELLOW)
C
C
C     CARDS 6
C     ------  FORMAT (2E15.7)
C
C          PERIOD1 (I)     PERIODS AT WHICH GROUP VELOCITIES ARE GIVEN
C           I=1,NGV        FOR FIRST SEISMOGRAM
C
C          GV1 (I)         GROUP VELOCITY VALUES FOR THE ABOVE PERIODS
C           I=1,NGV
C
C     CARD 7
C     ------  FORMAT (2E15.7)
C
C          PERIOD2 (I)     PERIODS AT WHICH GROUP VELOCITIES ARE GIVEN
C           I=1,NGV        FOR SECOND SEISMOGRAM
C
C          GV2 (I)         GROUP VELOCITY VALUES FOR THE ABOVE PERIODS
C           I=1,NGV
C
C     CARD 8
C     ------  FORMAT (10A8)
C
C          TITLE1----TITLE FOR THE FIRST SEISMOGRAM (10A8)
C
C     CARD 9
C     ------  FORMAT (10A8)
C
C          FMT1  ----INPUT FORMAT FOR FIRST SEISMOGRAM (10A8)
C                    BRACKETS MUST BE PLACED AROUND ARGUMENT
C
C     CARDS 10
C     -------  FORMAT FMT1
C
C          SEIS1 (I)       DIGITAL SEISMIC SIGNAL FROM FIRST STATION,
C          I=1,NSEIS1      NOT MORE THAN 1024 POINTS (FMT1)
C
```

```
C
C     CARD 11
C     -------  FORMAT (10A8)
C
C            TITLE2----TITLE FOR SECOND SEISMOGRAM (10A8)
C
C
C     CARD 12
C     -------  FORMAT (10A8)
C
C            FMT2  ----INPUT FORMAT FOR SECOND SEISMOGRAM (10A8)
C                      BRACKETS MUST BE PLACED AROUND ARGUMENT
C
C
C     CARDS 13
C     -------  FORMAT FMT2
C
C            SEIS2 (I)      DIGITAL SEISMIC SIGNAL FROM SECOND STATION,
C            I=1,NSEIS2     NOT MORE THAN 1024 POINTS (FMT2)
C
C
C     CARDS 1 TO 13 CAN BE REPEATED FOR ANY NUMBER OF DATA SETS
C
C     FOR THE OUTPUT FROM THE PROGRAM SEE O REPORT 8/76, SECTION 3.8
C
C
C                * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C
C     MAIN
C     ----
C
C
      COMMON/PHASE1/NSEIS1,SEIS1(2048),NSEIS2,SEIS2(2048),N,DELA,IMTRX,
     1P1,N2,NBY,DOMEGA,ALPHA,BETA
      COMMON/PHASE2/NGV1,PEROD1(120),GV1(120),NGV2,PEROD2(120),GV2(120)
      COMMON/PHASE3/Z(8192),SA(2048),SB(2048)
      COMMON/PHASE4/X(120,120),VSTEP(120),NR,NX,KPER,NC,PERIOD(120)
      COMMON/PHASE5/VELHI,VELLOW,NVEL,NP,NMAX,NMIN,TT(2048),TSTEP(120),N
     1G3
      COMMON /OUT/ Y(2048,5)
      COMMON /GRAPH/ TITLE(31),DDELA
      COMMON /POUT/ NT,NXL,RANGE,CONST,AINTER,T(2048),XX,BMAX(5),BMIN(5)
C
      DIMENSION STNAM1(8),STNAM2(8)
      DIMENSION FMT1(10),FMT2(10)
      DIMENSION TT1(2),TT2(5)
      DIMENSION TITLE1(20),TITLE2(20),TITLEA(20)
C
      REAL*8 DATE,TIME,TITLE1,TITLE2,FMT1,FMT2,TYPE,SECS,BLANK
      REAL*8 TITLE,TITLEA,TT1,TT2
      COMPLEX Z
C
      DATA TYPE/8H        /
      DATA BLANK/8H        /
      DATA SECS/8HSECONDS /
      DATA TT1/8HSEIS 1  ,8HSEIS 2  /
      DATA TT2/8HWIND 1  ,8HWIND 2  ,8HFILT 1  ,8HFILT 2  ,8HX-MULT   /
C
      CALL TIMER
      CALL DATIM(TITLE1(16),TITLE1(17))
      CALL DATIM(TITLE2(16),TITLE2(17))
      PI=4.0*ATAN(1.0)
C
C
 1    CALL DATIM(DATE,TIME)
      DO 2 I=1,20
      TITLE1(I)=BLANK
      TITLE2(I)=BLANK
 2    CONTINUE
      TITLE1(3)=SECS
      TITLE2(3)=SECS
      TITLE(31)=SECS
      DO 3 I=1,2048
      SEIS1(I)=0.0
      SEIS2(I)=0.0
 3    CONTINUE
      DO 200 I=1,2
 200  TITLE(I)=TT1(I)
C
C     READ IN THE INPUT DATA
C
      READ (5,4,END=999) (TITLEA(I), I=6,15)
 4    FORMAT (10A8)
      READ 5,STNAM1,DELTA1,MHGMT1,MNGMT1,SCGMT1,
     1       STNAM2,DELTA2,MHGMT2,MNGMT2,SCGMT2,MHOUR,MIN,SEC
 5    FORMAT(8A1,F12.5,I2,I2,F3.1,3X,8A1,F12.5,I2,I2,F3.1,16X,I2,I2,
     1F3.1)
      READ 6,NSEIS1,NSEIS2,NGV,NCOSTP,DELA,VELHI,VELLOW
```

```
 6      FORMAT(4I10,3F10.5)
        NGV1=NGV
        NGV2=NGV
        READ 7,TYPE,NOZERO,IMTRX,IVSEIS,NG1,NG2,NG3
 7      FORMAT(1X,A8,1X,3I10,3I5)
        READ 8,BAND,DWF,DV
 8      FORMAT(3F10.5)
        READ 9,(PEROD1(I),GV1(I),I=1,NGV1)
        READ 9,(PEROD2(I),GV2(I),I=1,NGV2)
 9      FORMAT(2E15.7)
C
C       CONVERSION TO BASIC UNITS OF TIME(SECS) AND DISTANCE(KMS.)
        GMT1=3600.0*FLOAT(MHGMT1)+60.0*FLOAT(MNGMT1)+SCGMT1-
       1FLOAT(NOZERO)*DELA
        GMT2=3600.0*FLOAT(MHGMT2)+60.0*FLOAT(MNGMT2)+SCGMT2-
       1FLOAT(NOZERO)*DELA
        ORIGTM=3600.0*FLOAT(MHOUR)+60.0*FLOAT(MIN)+SEC
        DELTA1=DELTA1*111.1
        DELTA2=DELTA2*111.1
        D21=DELTA2-DELTA1
        IF(DV.EQ.0.0)DV=0.02
C
C       FILTER CHARACTERISTICS
        BETA=ALOG(DWF)
        ALPHA=BETA/BAND**2
C
C
C       CALCULATE TIME SHIFT FACTORS FOR CROSS-MULTIPLICATION
        NOZER1 = NOZERO
        TMAX=D21/VELLOW+3.0*DELA
        TMIN=D21/VELHI-3.0*DELA
        NMIN=(GMT2-GMT1-TMIN)/DELA+1.0
        IF (NMIN.GT.0) GO TO 33
        NOZER1 = NOZER1-NMIN + 1
        GMT1 = GMT1 + FLOAT (NMIN-1) * DELA
 33     NSEIS1=NSEIS1+NOZER1
        NSEIS2=NSEIS2+NOZERO
        NOZERO=NOZERO+1
        NOZER1=NOZER1+1
        NMIN=(GMT2-GMT1-TMIN)/DELA+1.0
        NMAX=(TMAX-GMT2+GMT1)/DELA+1.0
        TMIN=GMT2-GMT1-FLOAT(NMIN-1)*DELA
        TMAX=GMT2-GMT1+FLOAT(NMAX-1)*DELA
        NP=NMAX+NMIN-1
        TT(1)=TMIN
```

```
        DO 34 I=2,NP
        TT(I)=TT(I-1)+DELA
 34     CONTINUE
C
        READ 4,(TITLE1(I),I=6,15)
        READ 4,(FMT1(I),I=1,10)
        READ FMT1,(SEIS1(I),I=NOZER1,NSEIS1)
        READ 4,(TITLE2(I),I=6,15)
        READ 4,(FMT2(I),I=1,10)
        READ FMT2,(SEIS2(I),I=NOZERO,NSEIS2)
C
C       PRINT OUT INPUT DATA AND PARAMETERS
        PRINT 11,DATE
        PRINT 12
        PRINT 13,(TITLEA(I),I=6,15)
        PRINT 14,MHOUR,MIN,SEC,ORIGTM
        PRINT 15,STNAM1,DELTA1,MHGMT1,MNGMT1,SCGMT1,GMT1
        PRINT 15,STNAM2,DELTA2,MHGMT2,MNGMT2,SCGMT2,GMT2
        PRINT 16,NSEIS1,NSEIS2,NGV1,NGV2,NCOSTP,DELA,VELHI,VELLOW,DV
        PRINT 17,BAND,DWF,NG1,NG2,NG3
        PRINT 18,TYPE,NOZERO,IMTRX,IVSEIS
        PRINT 19
        PRINT 13,(TITLE1(I),I=6,15)
        PRINT 4,(FMT1(I),I=1,10)
        PRINT 20,(I,SEIS1(I),I=NOZER1,NSEIS1)
        PRINT 21
        PRINT 13,(TITLE2(I),I=6,15)
        PRINT 4,(FMT2(I),I=1,10)
        PRINT 20,(I,SEIS2(I),I=NOZERO,NSEIS2)
        PRINT 22
        PRINT 23,(I,PEROD1(I),GV1(I),I=1,NGV1)
        PRINT 24
        PRINT 23,(I,PEROD2(I),GV2(I),I=1,NGV2)
        PRINT 25
C
C       INVERT SEISMOGRAM
        IVSEIS=IVSEIS+1
        GO TO (26,27,28),IVSEIS
 27     DO 29 I=1,NSEIS1
        SEIS1(I)=-SEIS1(I)
 29     CONTINUE
        GO TO 26
 28     DO 30 I=1,NSEIS2
        SEIS2(I)=-SEIS2(I)
 30     CONTINUE
```

```
 26   IMTRX=IMTRX+1
C
C     INCREASE NO. OF POINTS TO A POWER OF 2 .GT. NSEIS1 OR NSEIS2
      IF(NSEIS1.GT.1024.OR.NSEIS2.GT.1024)CALL EXIT
      N=128
      N2=7
 31   IF(NSEIS1.LE.N.AND.NSEIS2.LE.N)GO TO 32
      N=2*N
      N2=N2+1
      GO TO 31
C
 32   DDELA=DELA
      NGB2=NGV1/2
      NBY=N/2
      DOMEGA=PI/(DELA*FLOAT(NBY))
C
      NVEL=(VELHI-VELLOW)/DV+1.0
      VSTEP(1)=VELHI
      TSTEP(1)=D21/VSTEP(1)
      DO 35 I=2,NVEL
      VSTEP(I)=VSTEP(I-1)-DV
      TSTEP(I)=D21/VSTEP(I)
 35   CONTINUE
C
C     REMOVE MEAN OR LINEAR TREND FROM SEISMOGRAMS
      IP1=IMTRX-1
      CALL BASE(SEIS1(NOZER1),NSEIS1-NOZER1+1,TYPE,IP1)
      CALL BASE(SEIS2(NOZERO),NSEIS2-NOZERO+1,TYPE,IP1)
C
C     COS TAPER BOTH ENDS OF SEISMOGRAM
      IF(NCOSTP.EQ.0) GO TO 36
      CALL CSTP(SEIS1(NOZER1),NSEIS1-NOZER1+1,NCOSTP,1)
      CALL CSTP(SEIS2(NOZERO),NSEIS2-NOZERO+1,NCOSTP,1)
C
      IF(NG1.EQ.0) GO TO 42
C     GRAPH OUT GROUP VELOCITY VALUES
 36   CALL GRFGPV(NGV1,PERGD1,GV1)
      CALL GRFGPV(NGV2,PERGD2,GV2)
C
C     GRAPH OUT SEISMOGRAMS
      DO 37 I=1,1024
      Y(I,1)=SEIS1(I)
      Y(I,2)=SEIS2(I)
      Y(I,3)=0.0
      Y(I,4)=0.0
```

```
      Y(I,5)=0.0
 37   CONTINUE
      CALL CHAN(N,2,3,1)
 42   DO 201 I=1,5
 201  TITLE(I)=TT2(I)
C
      GO TO (100,38),IMTRX
 38   PRINT 39,NP,NMIN,NMAX,TMIN,TMAX
      PRINT 40,(TT(I),I=1,NP)
      PRINT 41,(VSTEP(I),TSTEP(I),I=1,NVEL)
C
C
 100  NC=0
      KPER=1
 101  CALL PACKET(N,DELA,NSEIS1,SEIS1,CRIGTM,DELTA1,GMT1,NGV1,PERGD1,
     1GV1,KPER,SA)
      CALL PACKET(N,DELA,NSEIS2,SEIS2,CRIGTM,DELTA2,GMT2,NGV2,PERGD2,
     1GV2,KPER,SB)
      DO 102 I=1,N
      Y(I,1)=SA(I)
      Y(I,2)=SB(I)
 102  CONTINUE
      CALL CRUNCH
      DO 103 I=1,N
      Y(I,3)=SA(I)
      Y(I,4)=SB(I)
 103  CONTINUE
      CALL OUTPUT
      IF (NG2.EQ.0) GO TO 104
      IF(KPER.EQ.1.OR.KPER.EQ.NGB2.OR.KPER.EQ.NGV1)CALL CHAN(N,5,3,1)
 104  KPER=KPER+1
      NC=NC+1
      IF(KPER.LE.NGV1)GO TO 101
      CALL EXHBIT
      CALL TIMER
      GO TO 1
C
C
 11   FORMAT(1H1///10X,62HSURFACE WAVE ANALYSIS(PHASE VELOCITY (KMS/SEC)
     1/ PERIOD (SECS),49X,A8)
 12   FORMAT(10X,62H-------------------------------------------------------
     1---------,49X,8H--------)
 13   FORMAT(/20X,10A8//)
 14   FORMAT(///10X,20HORIGIN TIME OF EVENT,2X,I2,1X,5HHOURS,2X,I2,1X,
     14HMINS,2X,F4.1,1X,4HSECS,5X,2HOR,5X,F12.4,1X,4HSECS)
```

```
15    FORMAT(///10X,'STATION NAME OF SEISMOGRAM 1 IS',2X,8A1//
     110X,45HDISTANCE OF RECORDING STATION FROM EPICENTRE ,F12.5,2X,
     13HKMS//
     110X,25HGMT TIME OF FIRST SAMPLE ,I2,2X,5HHOURS,2X,I2,2X,4HMINS,2X,
     1F4.1,2X,4HSECS,5X,2HOR,5X,F12.4,1X,4HSECS)
16    FORMAT(///10X,'NSEIS1 =',I6,5X,'NSEIS2 =',I6,5X,'NGV1  =',I4,5X,'N
     1GV2  =',I4,5X,'NCOSTP =',I5,5X,'DELA =',F6.2,//10X,'VELHI  =',F6.2
     2,5X,'VELLOW =',F6.2,5X,'DV    =',F6.2)
17    FORMAT(//10X,'BAND   =',F6.2,5X,'DWF    =',F6.2,5X,'NG1   =',I4,5X
     1,'NG2   =',I4,5X,'NG3   =',I4)
18    FORMAT(//10X,A8,11X,'NOZERO =',I6,5X,'IMTRX =',I4,5X,'IVSEIS =',I4
     1)
19    FORMAT(1H1//10X,12HSEISMOGRAM 1,3X,'ORIGINAL DATA')
20    FORMAT(6(I5,E15.7))
21    FORMAT(1H1//10X,'SEISMOGRAM 2',3X,'ORIGINAL DATA')
22    FORMAT(1H1//20X,53HPERIOD(SECS)/GROUP VELOCITY(KMS/SEC) FOR SEISMO
     1GRAM 1//)
23    FORMAT(3(I5,2E15.7))
24    FORMAT(1H1//20X,53HPERIOD(SECS)/GROUP VELOCITY(KMS/SEC) FOR SEISMO
     1GRAM 2//)
25    FORMAT(1H1)
39    FORMAT(1H1//2X,3HNP=,I5,1X,5HNMIN=,I5,1X,5HNMAX=,I5,1X,5HTMIN=,
     1G12.5,1X,5HTMAX=,G12.5)
40    FORMAT(//,1X,'TT(I),I=1,NP',/(1X,10G12.5))
41    FORMAT(/,1X,'VSTEP(I),TSTEP(I),I=1,NVEL',/(1X,5(2G12.5)))
C
C
999   CALL FINISH
      STOP
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE PACKET(N,DELA,NSEIS,SEIS,ORIGTM,DELTA,GMT,NGV,PERIOD,
     1GV,KPEROD,WAVPKT)
C
C
      DIMENSION SEIS(1),PERIOD(1),GV(1),WAVPKT(1)
C
C     GROUP ARRIVAL TIME FOR PERIOD OF INTEREST (TMARR)
      TMARR=DELTA/GV(KPEROD)+ORIGTM
C
C     LENGTH (WINDOW) AND NO. OF SAMPLES (NWIN) OF COSINE TAPER WINDOW
      WINDOW=4.5*PERIOD(KPEROD)
C
C     NWIN=WINDOW/DELA
      NWBY2=NWIN/2
      NWIN=2*NWBY2
C
C     TIME (TARRG) AND NO. OF POINTS (NSACW) FROM START OF SEISMOGRAM
C     TO 'TMARR'
      TARRG=TMARR-GMT
      NSACW=TARRG/DELA
C
C     NO. OF POINTS FROM START OF SEISMOGRAM TO START (NSTARW) AND END
C     (NFINW) OF WINDOW
      NSTARW=NSACW-NWBY2
      NFINW=NSACW+NWBY2
      NFRONT=NWBY2
C
C     CHECK THAT WINDOW IS INSIDE DATA SET
      IF(NSTARW.GT.1)GO TO 2
      NWIN=NWIN+NSTARW
      NFRONT=NWIN-NWBY2
      NSTARW=1
      PRINT 1,NSACW,NWBY2,PERIOD(KPEROD)
 1    FORMAT(///10X,'FOR THIS PERIOD THE PROGRAM IS USING AN ASYMMETRICA
     1L COSINE TAPERED WINDOW'/10X,
     1'BECAUSE A SYMMETRICAL WINDOW WOULD EXTEND BEYOND THE EDGE OF THE
     1SEISMOGRAM.'/10X,
     1'THE USER CAN OVERCOME THIS ASYMMETRY BY LENGTHENING THE SEISMOGRA
     1M'/10X,
     1'IF NO MORE ACTUAL DATA IS AVAILIBLE THEN ZEROS CAN BE ADDED TO TH
     1E FRONT.  (SEE MAIN PROGRAM LISTING) '/10X,
     1'NSACW=',I7,10X,'NWBY2=',I7,10X,'PERIOD=',F12.5)
C
C     LOAD AND COSINE TAPER THE WINDOWED SEISMOGRAM
 2    DO 3 I=1,N
      WAVPKT(I)=0.0
 3    CONTINUE
      DO 4 I=NSTARW,NFINW
      WAVPKT(I)=SEIS(I)
 4    CONTINUE
      CALL COSTP(WAVPKT(NSTARW),NWIN+1,NFRONT,1)
C
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
```

```
C
      SUBROUTINE CRUNCH
C
C
      COMMON/PHASE1/NSEIS1,SEIS1(2048),NSEIS2,SEIS2(2048),N,DELA,IMTRX,
     1PI,N2,NBY,DOMEGA,ALPHA,BETA
      COMMON/PHASE2/NGV1,PEROD1(120),GV1(120),NGV2,PEROD2(120),GV2(120)
      COMMON/PHASE3/Z(8192),SA(2048),SB(2048)
      COMMON/PHASE4/X(120,120),VSTEP(120),NR,NX,KPER,NC,PERIOD(120)
C
      COMPLEX Z
C
C     WINDOWED SEISMOGRAMS (SA(I),SB(I)) ARE PUT INTO A COMPLEX ARRAY Z
C     AND FOURIER ANALYSED
      DO 1 I=1,N
      Z(I)=CMPLX(SA(I),SB(I))
    1 CONTINUE
      CALL COOL(N2,Z,1.0)
      Z(1)=CMPLX(0.0,0.0)
      Z(NBY+1)=Z(1)
C
C     CHOOSE THE CENTRE FREQUENCY OF THE FILTER (OMEGAC) FROM THE
C     HARMONICS IN THE TIME SERIES (DOMEGA*L) SO THAT IT IS NEAREST TO
C     THE PERIOD OF INTEREST (PEROD1(I))
      OMEGAC=2.0*PI/PEROD1(KPER)
      L=OMEGAC/DOMEGA+0.5
      L=L-1
      OMEGAC=DOMEGA*FLOAT(L)
      PERIOD(KPER)=2.0*PI/OMEGAC
      OMEGA=0.0
C
C     BAND PASS FILTER THE WINDOWED SEISMOGRAMS
C     GAUSSIAN FILTER FUNCTION P IS MULTIPLIED BY BOTH +VE AND -VE
C     FREQUENCIES, J,K AND NJ,NK.
      DO 2 I=1,NBY
      J=L-I
      K=L+I
      IF(J.LT.2.AND.K.GT.NBY)GO TO 5
      NJ=N-J+2
      NK=N-K+2
      OMEGA=OMEGA-DOMEGA
      P=0.0
      PONENT = (-ALPHA*(OMEGA/OMEGAC)**2)
      IF (ABS(PONENT).GT. 100.00) GO TO 3
      P=EXP(PONENT)
 3    IF(J.LT.2)GO TO 4
      Z(J)=Z(J)*P
      Z(NJ)=Z(NJ)*P
 4    IF(K.GT.NBY)GO TO 2
      Z(K)=Z(K)*P
      Z(NK)=Z(NK)*P
 2    CONTINUE
C
C     TRANSFORM FILTERED SEISMOGRAMS BACK INTO TIME DOMAIN
 5    CALL COOL(N2,Z,-1.0)
      DO 6 I=1,N
      SA(I)=REAL(Z(I))
      SB(I)=AIMAG(Z(I))
 6    CONTINUE
C
      PRINT 7,  NC,L,OMEGAC,J,PERIOD(KPER)
 7    FORMAT(//2X,3HNC=,I5,1X,2HL=,I5,1X,7HOMEGAC=,F10.5,1X,2HJ=,I5,5X,
     1F10.5)
C
      GO TO (8,9),IMTRX
 9    IF(KPER.GT.5) GO TO 8
      PRINT 10,NC,N,KPER,L,N2
 10   FORMAT(//2X,'NC =',I5,2X,'N =',I5,2X,'KPER =',I5,2X,'L =',I5,2X,'N
     12 =',I5)
      PRINT 11,(SA(I),I=1,N)
 11   FORMAT(//,1X,'SA(I),I=1,N',/(1X,10G12.5))
      PRINT 12,(SB(I),I=1,N)
 12   FORMAT(/,1X,'SB(I),I=1,N',/(1X,10G12.5))
C
 8    RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE OUTPUT
C
C
      COMMON/PHASE1/NSEIS1,SEIS1(2048),NSEIS2,SEIS2(2048),N,DELA,IMTRX,
     1PI,N2,NBY,DOMEGA,ALPHA,BETA
      COMMON/PHASE3/Z(16384),SA(2048),SB(2048)
      COMMON/PHASE4/X(120,120),VSTEP(120),NR,NX,KPER,NC,PERIOD(120)
      COMMON/PHASE5/VELHI,VELLOW,NVEL,NP,NMAX,NMIN,TT(2048),TSTEP(120),N
     1G3
      COMMON /CUT/ Y(2048,5)
```

```
C
      DIMENSION E(2048),SAMSBS(2048)
C
C     SET UP CONSTANTS
      NEND=N
      NSTA=NMIN
      NSTB=0
      ISW=1
      IF(NSTA.LT.2) ISW=2
      K=1
C
C
C     CROSS-MULTIPLY TRACES FOR VARIOUS TIME SHIFTS
 1    XMAX=0.0
      IC=NSTA
      DO 2 I=NSTA,NEND
      IN=I-NSTA+NSTB+1
      SAMSBS(I)=SA(I)*SB(IN)
      IF (XMAX-SAMSBS(I))3,2,2
 3    IC=I
      XMAX=SAMSBS(I)
 2    CONTINUE
C
C     SAMSBS(I) GRAPHED IF NSTA = 1
      IF(NSTA.NE.1)GO TO 4
      DO 5 I=NSTA,NEND
      Y(I,5)=SAMSBS(I)
 5    CONTINUE
C
 4    NEND1=NEND-1
      E(K)=0.0
      IF(NEND1.LT.IC)GO TO 6
C     FIND MIN VALUE OF X-MULT TRACE AFTER THE MAX AT 'IC' AND CALCULATE
C     THE D.C. LEVEL
      DO 7 I=IC,NEND1
      IF (SAMSBS(I+1).GT.SAMSBS(I))GO TO 8
 7    CONTINUE
 8    E(K)=(SAMSBS(IC)+SAMSBS(I))*0.5
 6    K=K+1
C     CHECK RANGE OF PHASE VEL. VALUES COMPLETED
      IF (K.GT.NP)GO TO 9
C
C     IF RANGE NOT COMPLETED REVERSE TRACES IN X-MULT PROCESS
      GO TO (10,11),ISW
 10   IF (NSTA.EQ.2)ISW=2
      NSTA=NSTA-1
      GO TO 1
 11   NEND=NEND-1
      NSTB=NSTB+1
      IF (NEND+NSTB.GT.N.OR.NSTB+1.GT.NMAX)GO TO 9
      GO TO 1
C
C     INTERPOLATE TO FIND D.C. LEVEL AT VEL. STEPS OF INTEREST
 9    KLIM=K-1
      DO 12 I=1,NVEL
      X(KPER,I)=0.
      IND=1
      CALL INTPOL (NP,3,TT,E,TSTEP(I),X(KPER,I),IND)
 12   CONTINUE
C
      GO TO (13,14),IMTRX
 14   IF (KPER.GT.2) GO TO 13
      PRINT 15,NEND,NSTA,NSTB,ISW,K,KLIM,KPER
 15   FORMAT(1H1//2X,'NEND =',I5,2X,'NSTA =',I5,2X,'NSTB =',I5,2X,'ISW =
     1',I5,2X,'K =',I5,2X,'KLIM =',I5,2X,'KPER =',I5)
      PRINT 16,(TT(I),E(I),I=1,KLIM)
 16   FORMAT(//,1X,'TT(I),E(I),I=1,KLIM',/(1X,10G12.5))
      PRINT 17,(TSTEP(I),X(KPER,I),I=1,NVEL)
 17   FORMAT(//,1X,'TSTEP(I),X(KPER,I),I=1,NVEL',/(1X,10G12.5))
C
 13   RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE EXHBIT
C
C
      COMMON/PHASE4/X(120,120),VSTEP(120),NR,NX,KPER,NC,PERIOD(120)
      COMMON/PHASE5/VELHI,VELLOW,NVEL,NP,NMAX,NMIN,TT(2048),TSTEP(120),N
     1G3
C
C
      NR=NVEL
      NC=KPER-1
C
C     FIND MAXIMUM OF MATRIX
C
      XMAX=0.0
```

```
      DO 1 J=1,NR
      CALL AMAX(X(1,J),NC,HOLD)
      IF(HOLD.GT.XMAX)XMAX=HOLD
    1 CONTINUE
C
      DO 2 I=1,NR
      DO 3 J=1,NC
      X(J,I)=X(J,I)/XMAX*99.0
 3    CONTINUE
 2    CONTINUE
C
      N25=(NC-1)/25+1
      DO 4 IZ=1,N25
      NLOW=(IZ-1)*25+1
      NHY=NLOW+24
      IF(NHY.GT.NC)NHY=NC
      PRINT 5
 5    FORMAT(1H1//50X,14HMATRIX OF X(I)///)
      PRINT 6,(PERIOD(I),I=NLOW,NHY,2)
 6    FORMAT(3X,13F10.2)
      DO 7 I=1,NR
      PRINT 8 ,VSTEP(I),(X(J,I),J=NLOW,NHY)
 8    FORMAT(1X,F4.2,3X,25F5.2)
 7    CONTINUE
 4    CONTINUE
C
      IF (NG3.EQ.0) GO TO 9
      CALL CONTUR(X,NC,NR,120,5.0,5.0,100.0,3)
C
 9    RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C     SUBROUTINE BASE(X,N,TYPE,IP1)
C     ENTRY DBASE(DX,N,TYPE,IP1)
C     ENTRY CBASE(X,N,TYPE,IP1)
C     ENTRY DCBASE(DX,N,TYPE,IP1)
C
C     REMOVES EITHER THE LEAST SQUARES OR MEAN BAELINE OR THE FIRST POINT
C
C
      SUBROUTINE BASE(X,N,TYPE,IP1)
      DIMENSION X(N),DX(N)
```

```
      DOUBLE PRECISION DX,SUMX,SUMIX,FX,PHI,XINTO,XBAR,AN,X1,X2,
     1                 SUMX2,SUMX3,VARX,A3MNT,SKEW
      REAL*8 TYPE,MEAN,LEAST
      DATA MEAN/8HMEAN    /,LEAST/8HLEAST   /
C
      IPC=1
      IF(N.LE.0)GO TO 210
      N2=N
      N3=1
      FX=DBLE(X(1))
      GO TO 100
C
      ENTRY DBASE(DX,N,TYPE,IP1)
      IPC=2
      IF(N.LE.0)GO TO 210
      N2=N
      N3=1
      FX=DX(1)
      GO TO 100
C
      ENTRY CBASE(X,N,TYPE,IP1)
      IPC=1
      IF(N.LE.0)GO TO 210
      N2=N*2-1
       N3=2
      FX=DBLE(X(1))
      GO TO 100
C
      ENTRY DCBASE(DX,N,TYPE,IP1)
      IPC=2
      IF(N.LE.0)GO TO 210
      N2=N*2-1
       N3=2
      FX=DX(1)
C
C
 100  INDE=1
      IF(TYPE.EQ.MEAN)INDE=2
      IF(TYPE.EQ.LEAST)INDE=3
C
      AN = DFLOAT(N)
      SUMX = 0.D0
      SUMIX = 0.D0
      DO 104 I=1,N2,N3
      GO TO (101,102),IPC
```

```
 101   X1 = DBLE(X(I))
       GO TO 103
 102   X1 = DX(I)
 103   SUMX = SUMX + X1
       SUMIX = SUMIX + DFLOAT(I)*X1
 104   CONTINUE
       XBAR = SUMX/AN
       XINTO = (((4.D0*AN)+2.D0)*SUMX-6.D0*SUMIX)/(AN*(AN-1.D0))
       PHI = ((12.D0*SUMIX)-6.D0*(AN+1.D0)*SUMX)/(AN*(AN+1.D0)*(AN-1.D0))
C
       GO TO (110,120,130),INDE
 110   DO 113 I=1,N2,N3
       GO TO (111,112),IPC
 111   X(I) = DBLE(X(I)) - FX
       GO TO 113
 112   DX(I) = DX(I) - FX
 113   CONTINUE
       PRINT 1
 1     FORMAT(//4X,42HDATA HAS BEEN CORRECTED BY THE FIRST POINT)
       GO TO 200
 120   DO 123 I=1,N2,N3
       GO TO (121,122),IPC
 121   X(I) = DBLE(X(I)) - XBAR
       GO TO 123
 122   DX(I) = DX(I) - XBAR
 123   CONTINUE
       PRINT 2
 2     FORMAT(//4X,44HDATA HAS BEEN CORRECTED TO THE MEAN BASELINE)
       GO TO 200
 130   DO 133 I=1,N2,N3
       GO TO (131,132),IPC
 131   X(I) = DBLE(X(I)) - DFLOAT(I)*PHI - XINTO
       GO TO 133
 132   DX(I) = DX(I) - DFLOAT(I)*PHI - XINTO
 133   CONTINUE
       PRINT 3
 3     FORMAT(//4X,53HDATA HAS BEEN CORRECTED TO THE LEAST SQUARES BASELI
      1NE)
C
 200   SUMX2 = 0.D0
       SUMX3 = 0.D0
       DO 204 I=1,N
       GO TO (201,202),IPC
 201   X1 = DBLE(X(I))
       GO TO 203
```

```
 202   X1 = DX(I)
 203   X2 = X1*X1
       SUMX2 = SUMX2 + X2
       SUMX3 = SUMX3 + X2*X1
 204   CONTINUE
       VARX = SUMX2/AN
       A3MNT = SUMX3/AN
       SKEW = (A3MNT*A3MNT)/(VARX**3)
       PRINT 4, XBAR, FX, PHI, VARX, SKEW
 4     FORMAT(4X,14HMEAN OF DATA =,G12.5,8X,22HVALUE OF FIRST POINT =,
      1G12.5/4X,32HGRADIENT OF LEAST SQUARES LINE =,G12.5,8X,18HVARIANCE
      2OF DATA =,G12.5,8X,10HSKEWNESS =,G12.5)
C
       IF(IP1.EQ.1)PRINT 5, (I, X(I), I=1,N2,N3)
 5     FORMAT(//4X,24HTHE BASELINED DATA IS --//5(4X,6HSAMPLE,4X,4HX(I),
      13X)/(5(4X,I5,G12.5)))
C
 210   RETURN
       END
C
C                * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C      SUBROUTINE CSTP(X,N,NC,INDE)
C      ENTRY DCSTP(DX,N,NC,INDE)
C      ENTRY CCSTP(X,N,NC,INDE)
C      ENTRY DCCSTP(DX,N,NC,INDE)
C
C      ----------------------------------------------------------------
C
C      INDE=1--------COSINE TAPER BOTH ENDS OF CURVE.
C      INDE=2--------COSINE TAPER FRONT END OF CURVE.
C      INDE=3--------COSINE TAPER COSINE TAPER BACK END OF CURVE.
C
C      ----------------------------------------------------------------
C
C      X------IS THE ARRAY.
C
C      N-----IS THE NUMBER OF POINTS IN THE ARRAY.
C
C      NC-------IS THE STARTING NUMBER.
C
C      ----------------------------------------------------------------
C
C
```

```
      SUBROUTINE CSTP(X,N,NC,INDE)
      DIMENSION X(N),DX(N)
      DOUBLE PRECISION DX,PI,PHI,CPHI,SPHI,CTHET1,STHET1,CTHET2,STHET2
C
      IPC=1
      IC=1
      GO TO 1
C
      ENTRY DCSTP(DX,N,NC,INDE)
      IPC=2
      IC=1
      GO TO 1
C
      ENTRY CCSTP(X,N,NC,INDE)
      IPC=1
      IC=2
      GO TO 1
C
      ENTRY DCCSTP(DX,N,NC,INDE)
      IPC=2
      IC=2
C
C
 1    PI = 4.D0*DATAN(1.D0)
      PHI = PI/DFLOAT(NO-1)
      CPHI = DCOS(PHI)
      SPHI = DSIN(PHI)
      CTHET1 = 1.D0
      STHET1 = 0.D0
      CTHET2 = 1.D0
      STHET2 = 0.D0
      IF(NO.LE.0)RETURN
C
      GO TO (10,20),IPC
C
 10   DO 17 I = 1,NO
      GO TO (11,12,13),INDE
 11   IA = (N-I)*IC+1
 12   IN = (I-1)*IC+1
      GO TO 14
 13   IN = (N-I)*IC+1
 14   X(IN) = 5.D-1*DBLE(X(IN))*(1.D0-CTHET2)
      GO TO (15,16,16),INDE
 15   X(IA) = 5.D-1*DBLE(X(IA))*(1.D0-CTHET2)
 16   CALL SINCOS(CPHI,SPHI,CTHET1,STHET1,CTHET2,STHET2)
```

```
 17   CONTINUE
      GO TO 30
C
 20   DO 27 I = 1,NO
      GO TO (21,22,23),INDE
 21   IA = (N-I)*IC+1
 22   IN = (I-1)*IC+1
      GO TO 24
 23   IN = (N-I)*IC+1
 24   DX(IN) = 5.D-1*DX(IN)*(1.D0-CTHET2)
      GO TO (25,26,26),INDE
 25   DX(IA) = 5.D-1*DX(IA)*(1.D0-CTHET2)
 26   CALL SINCOS(CPHI,SPHI,CTHET1,STHET1,CTHET2,STHET2)
 27   CONTINUE
C
 30   RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE SINCOS(CPHI,SPHI,CTHET1,STHET1,CTHET2,STHET2)
C
      DOUBLE PRECISION CPHI,SPHI,CTHET1,STHET1,CTHET2,STHET2
C
      CTHET2 = CTHET1*CPHI - STHET1*SPHI
      STHET2 = STHET1*CPHI + CTHET1*SPHI
      CTHET1 = CTHET2
      STHET1 = STHET2
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C                 GIP SUBROUTINE INTPOL
C                 --- ---------- ------
C
C
C     BESSELS INTERPOLATION FORMULA TO THE FIFTH DIFFERENCE
C                 REFERENCE - I.A.T. P.56
C
C
C     FROM THE  CALL INTPOL(N,L,X,Y,XP,YP,IND)
C
```

```
C       N    NUMBER OF POINTS
C       L    STARTING POSITION IN TABLE OF VALUES  (NOT LESS THAN 3)
C       X    INDEPENDENT VARIABLE WHICH MUST BE EQUALLY SPACED
C       Y    VALUES OF THE FUNCTION
C       XP   INTERMEDIATE VALUE OF REQUIRED YP
C       YP   VALUE OF FUNCTION OF REQUIRED XP
C       IND = 1  NORMAL INTERPOLATION  YP FROM XP
C       IND = 2  INVERSE INTERPOLATION  XP FROM YP
C       IND = 3  NO INTERPOLATION  VALUE OUTSIDE TABLE
C
C
C          FORMS THE FOLLOWING DIFFERENCE TABLE --
C
C       X          Y          D1     D2     D3     D4     D5
C       -          -          --     --     --     --     --
C
C       X(K-2)     Y(K-2)
C                             D11
C       Y(K-1)     Y(K-1)            D21
C                             D12           D31
C       X(K)       Y(K)              D22           D41
C       XP         YP         D13           D32           D51
C       X(K+1)     Y(K+1)            D23           D42
C                             D14           D33
C       X(K+2)     Y(K+2)            D24
C                             D15
C       X(K+3)     Y(K+3)
C
C
      SUBROUTINE INTPOL(N,L,X,Y,XP,YP,IND)
      DIMENSION X(N),Y(N)
C
      DOUBLE PRECISION P,B2,B3,B4,B5,D11,D12,D13,D14,D15,
     1   D21,D22,D23,D24,D31,D32,D33,D41,D42,D51
C
C
      NL=N-L
      GO TO (10,20),IND
C
C        SET K FOR INTERPOLATION
 10   K=L
      IF(X(L)-XP)11,11,14
 11   DO 13 I=L,NL
      IF(X(I)-XP)12,15,100
 12   K=I
```

```
 13   CONTINUE
C        XP OUTSIDE TABLE
 14   IND=3
      GO TO 300
C        XP IS A TABULAR VALUE
 15   YP=Y(I)
      GO TO 300
C
C        SET K FOR INVERSE INTERPOLATION
 20   DO 23 K=L,NL
      P=(YP-Y(K))/(Y(K+1)-Y(K))
      IF(P)23,25,22
 22   IF(P-1.0D0)100,23,23
 23   CONTINUE
C        YP OUTSIDE TABLE
 24   IND=3
      GO TO 300
C        YP IS A FUNCTION VALUE
 25   XP=X(K)
      GO TO 300
C
C        FORM DIFFERENCES
 100  D11=Y(K-1)-Y(K-2)
      D12=Y(K)-Y(K-1)
      D13=Y(K+1)-Y(K)
      D14=Y(K+2)-Y(K+1)
      D15=Y(K+3)-Y(K+2)
      D21=D12-D11
      D22=D13-D12
      D23=D14-D13
      D24=D15-D14
      D31=D22-D21
      D32=D23-D22
      D33=D24-D23
      D41=D32-D31
      D42=D33-D32
      D51=D42-D41
      GO TO (210,220),IND
C
C        INTERPOLATION SECTION
C        FORM BESSEL COEFFICIENTS
 210  P=(XP-X(K))/(X(K+1)-X(K))
      B2=P*(P-1.0D0)
      B3=B2*(P-0.5D0)
      B4=B2*(P+1.0D0)*(P-2.0D0)
```

```
      B5=B4*(P-0.5D0)
      B2=B2/4.0D0
      B3=B3/6.0D0
      B4=B4/48.0D0
      B5=B5/120.0D0
C
C        THE FORMULA
      YP=Y(K)+P*D13+B2*(D22+D23)+B3*D32+B4*(D41+D42)+B5*D51
      GO TO 300
C
C        INVERSE INTERPOLATION SECTION
 220  DO 230 I=1,20
C        FORM BESSEL COEFFICIENTS
      B2=P*(P-1.0D0)
      B3=B2*(P-0.5D0)
      B4=B2*(P+1.0D0)*(P-2.0D0)
      B5=B4*(P-0.5D0)
      B2=B2/4.0D0
      B3=B3/6.0D0
      B4=B4/48.0D0
      B5=B5/120.0D0
C        THE INVERSE FORMULA
      P=(YP-Y(K)-B2*(D22+D23)-B3*D32-B4*(D41+D42)-B5*D51)/D13
 230  CONTINUE
      XP=X(K)+P*(X(K+1)-X(K))
C
 300  RETURN
      END
      SUBROUTINE FILLUP(NOPTS,Z)
C
C
      DIMENSION Z(1)
      COMPLEX Z,CZERO
      CZERO=CMPLX(0.0,0.0)
      AN=NOPTS-2
      N=ALOG10(AN)/ALOG10(2.0)+1.0
      N1=(2**N)+1
      N2=2**(N+1)
      N1M1=N1-1
      DO 1 I=2,N1M1
      NN=N2-I+2
      Z(NN)=CONJG(Z(I))
 1    CONTINUE
      Z(N1)=CZERO
      RETURN
```

```
      END
C
C                * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C  NEW CARGRF
C
C
C        FROM THE  CALL CARGRF(X,Y,N)  THIS PACKAGE PLOTS N POINTS
C     THE CARTESIAN CO-ORDINATES OF THE ITH POINT BEING SPECIFIED AS
C     X(I),Y(I)
C
C        THE OPTIONS ARE SET BY USING THE COMMON -
C
C     COMMON /GRFF/ TITLE(20), XMAX, XMIN, YMAX, YMIN, INDX, INDY, INC,
C    1IDOT, ANSTR1, IF, XLIMIT, YLIMIT, SCALX, SCALY
C
C        THE TITLE ARRAY CARRIES INFORMATION FOR ANNOTATING THE OUTPUT
C     GRAPH.  THIS ARRAY MUST BE SET UP AS FOLLOWS -
C
C         TITLE(1)  -)
C           .        )CONTAINS 24 HOLLERITH CHARACTERS GIVING THE UNITS
C         TITLE(3)  -)OF THE ABSCISSAE
C
C         TITLE(4)   )CONTAINS 16 HOLLERITH CHARACTERS GIVING THE UNITS
C         TITLE(5)   )OF THE ORDINATE
C
C         TITLE(6)  -)
C           .        )CONTAINS 80 HOLLERITH CHARACTERS GIVING A TITLE TO
C           .        )THE GRAPH
C         TITLE(15) -)
C
C         TITLE(16) -CONTAINS 8 HOLLERITH CHARACTERS GIVING DATE OF
C                    PROCESSING
C         TITLE(17) -CONTAINS 8 HOLLERITH CHARACTERS GIVING TIME OF
C                    PROCESSING
C         TITLE(18) -)
C           .        )UNUSED
C         TITLE(20) -)
C
C         XMAX )SET BOTH EQUAL IF PROGRAM TO CHOOSE THE ABSCISSAE
C         XMIN )SCALE. OTHERWISE SET TO CHOSEN LIMITS OF ABSCISSAE SCALE
C
C         YMAX )SET BOTH EQUAL IF PROGRAM TO CHOOSE THE ORDINATE SCALE
C         YMIN )OTHERWISE SET TO CHOSEN VALUES OF ORDINATE SCALE
```

```
C
C        INDX IS AN INDICATOR FOR PLOTTING THE ABSCISSAE ON A LOG SCALE
C          INDX=1 ABSCISSAE ON LINEAR SCALE
C          INDX=2 ABSCISSAE ON LOG SCALE
C
C        INDY IS A SIMILAR INDICATOR FOR THE ORDINATE SCALE
C
C        N.B. CONTENTS OF ARRAYS ARE MODIFIED USING LOG SCALE
C
C        IND IS AN INDICATOR FOR CONTROLLING FRAME CALLS -
C          IND=1 CARGRF CALLS ADVFLM AND PLOTS ON A NEW FRAME
C              =2 CARGRF PLOTS ON THE CURRENT FRAME
C
C        IDOT IS THE SC4020 CODE OF THE REQUIRED PLOTTING SYMBOL
C
C        ANSTR1 INDICATES WHETHER THE PLOTTED POINTS HAVE TO BE JOINED UP
C          ANSTR1=1. POINTS NOT JOINED
C                 =2. POINTS JOINED
C
C        IF SPECIFIES TYPE OF OUTPUT
C          IF=1 OUTPUT ON MICROFILM
C
C    REWRITTEN BY J.B.YOUNG FOR THE IBM 370/168 ON 01/07/74
C
C
C
      SUBROUTINE GRAF(X,Y,N)
      ENTRY CARGRF(X,Y,N)
C
      DIMENSION X(N), Y(N)
C
      COMMON /GRFF/ TITLE(20), XMAX, XMIN, YMAX, YMIN, INDX, INDY, IND,
     1IDOT, ANSTR1, IF, XLIMIT, YLIMIT, SCALX, SCALY, XFACTR, YFACTR
C
      REAL*8 TITLE,DATE
C
      REAL*4 INSTR1,JOIN/4HJOIN/,BLANK/4H    /
C
      INTEGER*4 PLACEX,PLACEY,XPLOT1,XPLOT2,YPLOT1,YPLOT2,XPOS,YPOS
C
C
      CALL DATIM(DATE,TITLE(17))
      IF(IF)2,2,1
 1    IF(IF-4)3,3,2
 2    IF=2
```

```
 3    IF(IND.NE.2)IND=1
      IF(INDX.NE.2)INDX=1
      IF(INDY.NE.2)INDY=1
      IF(IDOT.GT.63)IDOT=48
      INSTR1=JOIN
      IF(ANSTR1.EQ.1.)INSTR1=BLANK
C
      GO TO (30,10),INDX
 10   POSXT=PMIN(X,N)*0.8
      DO 20 I=1,N
      IF(X(I).LT.POSXT)X(I)=POSXT
      X(I)=ALOG10(X(I))
 20   CONTINUE
 30   GO TO (60,40),INDY
 40   POSYT=PMIN(Y,N)*0.8
      DO 50 I=1,N
      IF(Y(I).LT.POSYT)Y(I)=POSYT
      Y(I)=ALOG10(Y(I))
 50   CONTINUE
C
 60   GO TO (100,200),IND
C
C
 100  IF(XMAX-XMIN)110,120,110
 110  XMX=XMAX
      XMN=XMIN
      GO TO 130
 120  CALL AMAX(X,N,XMX)
      CALL AMIN(X,N,XMN)
 130  IF(YMAX-YMIN)140,150,140
 140  YMX=YMAX
      YMN=YMIN
      GO TO 160
 150  CALL AMAX(Y,N,YMX)
      CALL AMIN(Y,N,YMN)
 160  CALL ADVFLM(IF)
C
C
 200  CALL SCALEN(X,XLIMIT,SCALX,PLACEX,XFACTR,N,XMX,XMN)
      CALL SCALEN(Y,YLIMIT,SCALY,PLACEY,YFACTR,N,YMX,YMN)
      XPLOT1=(X(1)-XLIMIT)*SCALX+123.
      YPLOT1=923.-(Y(1)-YLIMIT)*SCALY
      CALL PLOT(XPLOT1,IDOT,YPLOT1)
      DO 230 I=2,N
      XPLOT2=(X(I)-XLIMIT)*SCALX+123.
```

```
      YPLOT2=923.-(Y(I)-YLIMIT)*SCALY
      CALL PLOT(XPLOT2,IDCT,YPLOT2)
      IF(INSTR1-JOIN)220,210,220
 210  CALL VECTOR(XPLOT1,YPLOT1,XPLOT2,YPLOT2)
 220  YPLOT1=YPLOT2
      XPLOT1=XPLOT2
 230  CONTINUE
C
      GO TO (300,400),IND
C
C
 300  CALL VECTOR(115,923,1003,923)
      CALL VECTOR(123,931,123,43)
      DO 310 I=1,11
      XPOS=203+(I-1)*80
      YPOS=43+(I-1)*80
      CALL VECTOR(XPOS,923,XPOS,931)
      CALL VECTOR(115,YPOS,123,YPOS)
 310  CONTINUE
      DO 370 I=1,3
      XLIM=XLIMIT+FLOAT(I-1)*XFACTR*4.
      GO TO (330,320),INDX
 320  XPOS=92+(I-1)*320
      CALL TSP(XPOS,48,942)
      CALL C4020E(10.**XLIM,11,4)
      GO TO 340
 330  XPOS=20+PLACEX*8+(I-1)*320
      CALL TSP(XPOS,48,942)
      CALL C4020F(XLIM,13,PLACEX)
 340  YPOS=915-(I-1)*320
      YLIM=YLIMIT+FLOAT(I-1)*YFACTR*4.
      GO TO (360,350),INDY
 350  CALL TSP(28,48,YPOS)
      CALL C4020E(10.**YLIM,11,4)
      GO TO 370
 360  CALL TSP(12,48,YPOS)
      CALL C4020F(YLIM,13,PLACEY)
 370  CONTINUE
C
      IF(TITLE(16).NE.DATE)GO TO 400
      CALL TSP(760,48,958)
      CALL HORAM(TITLE(1),24)
      CALL TSP(48,48,291)
      CALL VERAM(TITLE(4),16)
      CALL TSP(130,48,23)




      CALL HORAM(TITLE(6),80)
C
C
 400  RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE SCALEN(X,XLIMIT,SCALX,IPLACE,FACTOR,N,XMAX,XMIN)
C
C     COMPUTES SCALING VALUES FOR CARGRF
C
C
      DIMENSION X(N)
      DOUBLE PRECISION XRG,R,FACT,S
C
      XRG=XMAX-XMIN
      IF(XRG.LT..0000000001)XRG=1.0D0
C
      R=0.0D0
 1    IF(XRG.LT.(10.0D0**R*.0000000001)) GO TO 2
      R=R+1.0D0
      GO TO 1
 2    FACT=(10.0D0**(R-1.0D0)*.0000000000125)
      S=0.0D0
 3    IF(XRG.LE.FACT*(2.0D0**S)) GO TO 4
      S=S+1.0D0
      GO TO 3
 4    FACTOR=(FACT*(2.0D0**S))*10.0D-2
C
      XLIMIT=FLOAT(IFIX(XMIN/FACTOR))*FACTOR
      IF(XMIN.LT.XLIMIT)XLIMIT=XLIMIT-FACTOR
      SCALX=80./FACTOR
      IPLACE=13.-R
      IF(IPLACE.LT.1)IPLACE=1
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      FUNCTION PMIN (X,N)
C
C     FINDS MINIMUM POSITIVE VALUE OF ARRAY X
```

```
C
C
      DIMENSION X(N)
C
      DO 1 KQ=1,N
      IF(X(KQ).LE.0.0)GO TO 1
      GO TO 5
    1 CONTINUE
      PMIN=1.25
      GO TO 6
    5 KP=KQ
    2 IF(KQ-N)3,4,4
    3 KQ=KQ+1
      IF(X(KQ).LE.0.0)GO TO 2
      IF(X(KP)-X(KQ))2,5,5
    4 PMIN=X(KP)
    6 RETURN
      END
C
C                * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C     SUBROUTINE CHAN(N,NC,IF,IND)
C
C
C        THIS PACKAGE PLOTS UP TO 20 CHANNELS OF A DOUBLE SUBSCRIPTED
C     ARRAY, X(I,J) WHERE J DEFINES THE CHANNEL NUMBER AND I THE SAMPLE
C     NUMBER IN THE JTH CHANNEL. THE DATA IS ASSUMED TO BE SAMPLED AT
C     EQUAL INTERVALS.
C
C        FIVE CHANNELS ARE PLOTTED TO A FRAME, THE TIME AXIS IS PARALLEL
C     TO THE Y AXIS OF THE PLOTTER.
C
C        THE PACKAGE USES THE COMMONS -
C
C     COMMON  /OUT/     X(2048,5)
C
C     COMMON /GRAPH/ TITLE(31), DELA
C
C      X IS THE ARRAY TO BE PLOTTED, TITLE IS AN ARRAY FOR ANNOTATING
C     THE OUTPUT AND DELA IS THE SAMPLING INTERVAL OF THE TIME SERIES.
C        THE TITLE ARRAY IS MADE UP AS FOLLOWS -
C
C         TITLE(1)  -)
C           .       -)
C           .       -)
C           .       -)
C         TITLE(J)  -) EACH ELEMENT CARRIES AN 8 CHARACTER TITLE
C           .       -) DESCRIBING THE DATA IN THE CHANNEL
C           .       -)
C         TITLE(20) -)
C
C         TITLE(21) -)
C           .       -) CARRIES 80 HOLLERITH CHARACTERS GIVING A TITLE TO
C           .       -) THE OUTPUT
C         TITLE(30) -)
C
C         TITLE(31) - CARRIES 8 HOLLERITH CHARACTERS GIVING UNITS OF THE
C                     TIME SERIES
C
C           IF IS AN INDICATOR TO SPECIFY THE OUTPUT REQUIRED
C
C         IF IF =1 OUTPUT IS ON MICROFILM
C            IF =2 OUTPUT IS ON HARD COPY
C            IF =3 OUTPUT IS ON BOTH MICROFILM AND HARD COPY
C
C           IND IS AN INDICATOR IF SET TO 1 THE MAXIMUM AMPLITUDE IN
C     EACH CHANNEL IS SCALED TO THE FULL RANGE AVAILABLE, IF SET TO 2
C     THE MAXIMUM RANGE OF THE X ARRAY IS SCALED TO THE FULL RANGE
C     AVAILABLE.
C           SETTING IND TO 1 DESTROYS THE ORIGINAL DATA IN THE X ARRAY,
C     PLOTTING USING CHAN SHOULD THEREFORE BE DONE ONLY WHEN ALL
C     REQUIRED COMPUTATION HAS BEEN CARRIED OUT ON X - THIS RESTRICTION
C     DOES NOT APPLY IF IND IS 2.
C
C
      SUBROUTINE CHAN(N,NC,IF,IND)
C
      COMMON  /OUT/  X(2048,5)
C
      COMMON /GRAPH/ TITLE(31), DELA
      REAL*8 TITLE
C
      COMMON/PLOT/NT,NXL,RANGE,CONST,AINTER,
     1TT(2048),XX,BMAX(5),BMIN(5)
C
C
      GO TO (10,20),IND
 10   XMAX=1.
      XMIN=0.
C
```

```
 20     DO 60 I=1,NC
        CALL AMAX(X(1,I),N,ZMAX)
        CALL AMIN(X(1,I),N,ZMIN)
        BMAX(I)=ZMAX
        BMIN(I)=ZMIN
        GO TO (30,60),IND
 30     ZRG=ZMAX-ZMIN
        IF(ZRG-1.0E-30)60,60,40
C
 40     DO 50 K=1,N
        X(K,I)=(X(K,I)-ZMIN)/ZRG
 50     CONTINUE
 60     CONTINUE
C
        GO TO (80,70),IND
 70     CALL AMAX(BMAX,NC,XMAX)
        CALL AMIN(BMIN,NC,XMIN)
C
 80     N55=(NC-1)/5+1
        NN=5
        PRINT 77
 77     FORMAT(//,2X,'PRINT OUT FROM SUBROUTINE CHAN')
        PRINT 66, N55,NN,NC,N,XMAX,XMIN
 66     FORMAT(2X,4I6,2X,2G14.5)
        CALL SCHAN(XMAX,XMIN,N)
        CALL TCHAN(N,DELA)
C
        DO 90 I=1,N55
        NC5=(I-1)*5+1
        IF(I.EQ.N55)NN=NC-NC5+1
        CALL PMULTI(N,NN,NC5,IF)
 90     CONTINUE
C
        RETURN
        END
C
C               * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
        SUBROUTINE PMULTI(N,NN,NC5,IF)
C
C
        COMMON  /OUT/  X(2048,5)
C
        COMMON /GRAPH/ TITLE(31), DELA




        REAL*8 TITLE
C
        COMMON/POUT/NT,NXL,RANGE,CONST,AINTER,
       1TT(2048),XMIN,BMAX(5),BMIN(5)
C
C
        NC=NN
        IZ=NC5
        NM=NT
        DO 10 I=1,NT
        IF(I.GT.NM)GO TO 99
        CALL ADVFLM(IF)
        CALL TSP(100,48,8)
        CALL HCRAM(TITLE(21),80)
        CALL TSP(1014,48,124)
        CALL VERAM(32H     TIME MARKS AT INTERVALS OF ,32)
        CALL TSP(956,48,640)
        CALL C4020F(AINTER,8,3)
        CALL TSP(1014,48,648)
        CALL VERAM(TITLE(31),8)
C
        IBEGIN=(I-1)*NXL+1
        IEND=NXL
        IF(I*NXL.GT.N)IEND=N-((I-1)*NXL)
C
        DO 50 J=1,NC
        JA=J
        K=J+IZ-1
        IPOS=1000.-FLOAT(JA-1)*200.
        CALL TSP(IPOS,48,20)
        CALL VERAM(TITLE(K),8)
        PRINT 66, X(IBEGIN,K),IEND,JA,XMIN,RANGE,CONST,BMAX(K),BMIN(K)
 66     FORMAT(2X,G14.5,2X,2I6,2X,5G14.5)
        CALL PEN(X(IBEGIN,K),IEND,JA,XMIN,RANGE,CONST)
 50     CONTINUE
C
        CALL PEN(TT(IBEGIN),IEND,5,0.0,1.,CONST)
 10     CONTINUE
C
 99     RETURN
        END
C
C               * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
```

```
      SUBROUTINE SCHAN(XMAX,XMIN,N)
C
C
      COMMON /GRAPH/ TITLE(31), DELA
      REAL*8 TITLE
C
      COMMON/PLOT/NT,NXL,RANGE,CONST,AINTER,
     1TT(2048),XX,BMAX(5),BMIN(5)
C
      REAL*8 SECS/8HSECONDS /
C
C
      XRG=XMAX-XMIN
      XX=XMIN
      RANGE=200./XRG
      IF(DELA.EQ..04444.AND.TITLE(31).EQ.SECS)GO TO 3
      S=6.
      IF(N.LE.100)S=7.
      R=0.
 1    IF(DELA.LT.(10.**R*.000001))GO TO 2
      R=R+1.
      GO TO 1
 2    IF(R.LT.1.)GO TO 99
      INTER=10.**(8.-R)*DELA+0.5
      AINTER=FLOAT(INTER)/(10.**(S-R))
      ANXL=(8.*AINTER)/DELA
      GO TO 4
C
 3    AINTER=5.
      ANXL=1125.
C
 4    NXL=ANXL
      CONST=1000./ANXL
      NT=N/NXL+1
C
 99   RETURN
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE PEN(X,IEND,JA,XMIN,RANGE,CONST)
C
C
      DIMENSION X(IEND)
C
```

```
      INTEGER  XPLOT1, XPLOT2, TPLOT1, TPLOT2
C
C
      AJ=812. - FLOAT(JA-1)*200.
      XPLOT1=AJ + (X(1) - XMIN)*RANGE
      TPLOT1=19.
C
      DO 15 I=2,IEND
      XPLOT2=AJ + (X(I) - XMIN)*RANGE
      TPLOT2=FLOAT(I-1)*CONST + 19.
      CALL VECTOR(XPLOT1,TPLOT1,XPLOT2,TPLOT2)
      XPLOT1=XPLOT2
      TPLOT1=TPLOT2
   15 CONTINUE
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE TCHAN(N,DELA)
C
C
      COMMON/PLOT/NT,NXL,RANGE,CONST,AINTER,
     1TT(2048),XX,BMAX(5),BMIN(5)
C
C
      DO 1 I=1,N
      TT(I)=0.
 1    CONTINUE
C
      LT=(FLOAT(N)*DELA)/AINTER+1.
      TT(2)=25.
      AD=AINTER/DELA
C
      DO 2 I=1,LT
      LL=FLOAT(I-1)*AD+1.
      TT(LL)=25.
 2    CONTINUE
C
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
```

```
C
      SUBROUTINE AMAX (X,N,XMAX)
C
C     FINDS MAXIMUM VALUE OF ARRAY X
C
C
      DIMENSION X(N)
C
      KQ = 1
    2 KP = KQ
    5 IF(KQ -N)3,4,4
    3 KQ = KQ + 1
      IF(X(KP) - X(KQ))2,5,5
    4 XMAX = X(KP)
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE AMIN (X,N,XMIN)
C
C     FINDS MINIMUM VALUE OF ARRAY X
C
C
      DIMENSION X(N)
C
      KQ = 1
    5 KP = KQ
    2 IF(KQ-N)3,4,4
    3 KQ = KQ + 1
      IF(X(KP) - X(KQ))2,5,5
    4 XMIN = X(KP)
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE CONTOR(A,NX,NY,N,CSTEP,CLOW,CHIGH,IF)
      DIMENSION A(N,N)
      CALL ADVFLM(IF)
C
      HX=10.2/FLOAT(NX)
      HY=10.2/FLOAT(NY)




      QX=0.02
      QY=0.02
      DC=CSTEP
      C3=CLOW
      C4=CHIGH
      C1=C4+DC
      C2=C1+DC
C
      CALL OB07A(A,DC,C1,C2,C3,C4,NX,NY,HX,HY,QX,QY,0.,1,N)
C
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE OB07A(F,DC,C1,C2,C3,C4,NX,NY,HX,HY,QX,QY,COSS,NQ,IDF)
      DIMENSION F(IDF,IDF)
      COMMON /CONT/ CX,CY,COSG,SING,DX,DY,QX,QY,X,Y,A,B,C,D,CEN,IHL,IHK
C
      IHL=1
      IHK=2
      IHC=3
      CEN=100.
      OX = QX
      OY = QY
      COSG = COSS
      DX = HX
      DY = HY
      IX = NX -1
      IY = NY -1
      AA = QX * FLOAT(IX)
      BB = QY * FLOAT(IY)
      CX = 0.5 * DX
      CY = 0.5 * DY
      SING = SQRT(1.-COSG*COSG)
      DELR = 1.0/DC
      IF (COSG) 101,102,102
  101 OX = OX - DY*FLOAT(IY)*COSG
  102 IF (NQ) 105,105,104
  104 XC = CEN*(CX)
      YC = CEN*(OY)
      CALL QUACK(XD,YD,IHL)
      XC = CEN*(CX+AA)
      CALL QUACK(XD,YD,IHK)
```

```
      XC = CEN*(DX+AA+BE*COSG)
      YC = CEN*(DY+BB*SING)
      CALL QUACK(XC,YC,IHK)
      XC = CEN*(DX+BE*COSG)
      CALL QUACK(XC,YC,IHK)
      XC = CEN*(DX)
      YC = CEN*(DY)
      CALL QUACK(XC,YC,IHK)
      IF (NC-1) 105,105,106
 106  CONTINUE
      IX1=IX-1
      IY1=IY-1
      DO 107 J=1,IY1
      DO 108 I=1,IX1
      X = FLOAT(I) * DX
      Y = FLOAT(J) * DY
      XC = CEN*(DX+X+Y*COSG)
      YC = CEN*(DY+Y*SING)
  108 CALL QUACK(XC,YC,IHC)
  107 CONTINUE
  105 CONTINUE
      SMALL=1.E-6
      DO 112 J=1,IY
      DO 113 I=1,IX
C
C
C     FOR REASONS WHICH WILL BECOME APPARENT, CONTOUR VALUES SHOULD NOT
C     EXACTLY COINCIDE WITH FIELD VALUES.  TO TAKE AN EXAMPLE, SUPPOSE
C     FIELD VALUES ARE READ IN AS  A=25.3  B=29.4  C=29.4  D=26.5  AND
C     THE CONTOURS ARE REQUESTED BY PUNCHING  C1=20.4  C2=38.4  DC=4.5 .
C     THERE IS THEN THE DANGER THAT CONTOUR 29.4 MIGHT BE IDENTIFIED AS
C     INTERSECTING SIDE BC  (SEE LINE OF EXECUT-
C     302  IF ((B-CON)*(C-CON).LT.0) 303,304     ).
C     IF BY CHANCE WE GO TO 303, WE WILL GET TO 402,404,410,OR 412 AND
C     GET OVERFLOW ON DIVIDING BY (C-B).
C         AS ANOTHER EXAMPLE, A MALFUNCTION CAN OCCUR WHEN ONLY ONE OF
C     THE 4 FIELD VALUES IS EQUAL TO A CONTOUR LEVEL.   SUPPOSE A=25.3,
C     B=29.4, C=30.0, D=26.4, CONTOUR=29.4 . CENTRE POINT E IS 27.78 AND
C     CONTOUR SHOULD BE DRAWN FROM SIDE CD TO SEMIDIAGONAL CE TO CORNER
C     B.   HOWEVER, THE COMPUTER MIGHT FIND THAT BOTH AB AND BC ARE
C     INTERSECTED BY THE CONTOUR.   SINCE CD IS INTERSECTED THE LOGIC
C     MUST ASSUME THAT AD IS ALSO INTERSECTED.   A WILD LINE WILL BE
C     DRAWN FROM A POINT WELL OUTSIDE THE GRID-BOX, LYING ON
C     EXTRAPOLATED LINE AD.
C         THE FOLLOWING TRANSFORMATION REDUCES ALL SUCH RISKS TO A
C     PRACTICAL ZERO.  MODIFY THE 4 FIELD VALUES (Z) TO
C                 Z(NEW)  =  Z(OLD)  +  DZ          WHERE
C                DZ  =  Z * 10**5  +  DELBIT       (FOR Z.GE.0)
C     OR         DZ  =  Z * 10**5  -  DELBIT       (FOR Z.LT.0) .
C     THE DISCONTINUITY OF 2*DELBIT IS INTRODUCED IN CASE Z IS SMALL,
C     BECAUSE THE TRANSFORMATION MIGHT THEN DEGENERATE TO AN IDENTITY
C     MAPPING.
C
      A = F(I,J)
      A= 1.00001*A + SIGN(SMALL,A)
      B = F(I+1,J)
      B= 1.00001*B + SIGN(SMALL,B)
      C = F(I+1,J+1)
      C= 1.00001*C + SIGN(SMALL,C)
      D = F(I,J+1)
      D= 1.00001*D + SIGN(SMALL,D)
      X = DX * FLOAT(I-1)
      Y = DY * FLOAT(J-1)
C
C
C     TO DETERMINE WHICH CONTOURS (IF ANY) ENTER THE BOX, APPLY INTEGRAL
C     PARTS TEST.  TO DO THIS CORRECTLY WE OPERATE UPON SHIFTED FIELD
C     VALUES AND RETURN TO PURE FIELD AFTER
C
 103  DS = SIGN (DC,C1)
      EMOD = C1 - DS*AINT (ABS (C1*DELR))
      JA = IFIX((A-EMOD)*DELR)
      JB = IFIX((B-EMOD)*DELR)
      JC = IFIX((C-EMOD)*DELR)
      JD = IFIX((D-EMOD)*DELR)
      J1 = MAX0(JA,JB,JC,JD)
      J3 = MIN0(JA,JB,JC,JD)
      A1 = FLOAT(J1)
      A3 = FLOAT(J3)
      IF(J3.EQ.J1) GO TO 123
C
C     JUMP ON IF NO CONTOURS EXIST, OTHERWISE FIND WHAT THEY ARE
      J3 = J3 + 1
      AJC1 = EMOD + AMAX1(C1-EMOD,A3*DC)
      AJC2 = EMOD + AMIN1(C2-EMOD,A1*DC) + 0.01*DC + SMALL
      CON=AJC1
C
C     BROKEN CONTOUR
 41   CALL EXECUT(1)
      CON=CON+DC
```

```
      IF(CON-AJC2)41,41,123
C
 123  DS = SIGN (DC,C3)
      EMOD = C3 - DS*AINT (ABS (C3*DELR))
      JA = IFIX((A-EMOD)*DELR)
      JB = IFIX((B-EMOD)*DELR)
      JC = IFIX((C-EMOD)*DELR)
      JD = IFIX((D-EMOD)*DELR)
      J1 = MAX0(JA,JB,JC,JD)
      J3 = MIN0(JA,JB,JC,JD)
      A1 = FLOAT(J1)
      A3 = FLOAT(J3)
      IF(J3.EQ.J1) GO TO 113
C
C     JUMP OUT IF NO CONTOURS, OTHERWISE FIND THEM
      J3 = J3 + 1
      AJC3 = EMOD + AMAX1(C3-EMOD,A3*DC)
      AJC4 = EMOD + AMIN1(C4-EMOD,A1*DC) + 0.01*DC + SMALL
      CON=AJC3
C
C     FULL CONTOUR
 42   CALL EXECUT(2)
      CON=CON+DC
      IF(CON-AJC4)42,42,113
 113  CONTINUE
 112  CONTINUE
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE QUACK(PX,PY,IND)
      GO TO (2,3,1,5),IND
 1    CALL PLOT(PX,42,PY)
      GO TO 5
 2    IXS=IFIX(PX)
      IYS=IFIX(PY)
      GO TO 5
 3    IXF=IFIX(PX)
      IYF=IFIX(PY)
      IF((IXS-IXF)**2+(IYS-IYF)**2)5,5,4
 4    CALL VECTOR(IXS,IYS,IXF,IYF)
      IXS=IXF
      IYS=IYF
```

```
 5    RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE EXECUT(KLINK)
      COMMON /CONT/ CX,CY,COSG,SING,DX,DY,CX,CY,X,Y,A,B,C,D,CON,IHL,IHK
      REAL I
      I=1.
      CEN = 0.25 * (A+B+C+D)
      PIP = CEN - CON
      KZ = 16
      IF ((A-CON)*(B-CON).LT.0.) KZ = KZ - 8
      IF ((B-CON)*(C-CON).LT.0.) KZ = KZ - 4
      IF ((C-CON)*(D-CON).LT.0.) KZ = KZ - 2
      IF ((A-CON)*PIP.LT.0.) KZ = KZ - 1
      GO TO (401,402,403,404,405,406,407,408,409,410,411,412,413,414,
     1 415,416),KZ
 401  CALL QUICK(X,Y+CY*(CON-A)/(D-A),IHL)
      P = PIP/(CEN-A)
      CALL QUICK(X+CX*(I-P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 201
      CALL QUICK(X+DX*(CON-A)/(B-A),Y,IHK)
 201  CALL QUICK(X+DX*(CON-D)/(C-D),Y+CY,IHL)
      P = PIP/(CEN-C)
      CALL QUICK(X+CX*(I+P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 202
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
 202  RETURN
 402  CALL QUICK(X,Y+CY*(CON-A)/(D-A),IHL)
      P = PIP/(CEN-D)
      CALL QUICK(X+CX*(I-P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 203
      CALL QUICK(X+DX*(CON-D)/(C-D),Y+CY,IHK)
 203  CALL QUICK(X+DX*(CON-A)/(B-A),Y,IHL)
      P = PIP/(CEN-B)
      CALL QUICK(X+CX*(I+P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 204
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
 204  RETURN
 403  CALL QUICK(X+DX*(CON-A)/(B-A),Y,IHL)
      P = PIP/(CEN-D)
      CALL QUICK(X+CX*(I-P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 205
```

```
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
205   RETURN
404   CALL QUICK(X+DX*(CCN-A)/(B-A),Y,IHL)
      P = PIP/(CEN-B)
      CALL QUICK(X+CX*(I+P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 206
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
206   RETURN
405   P = PIP/(CEN-D)
      Q = PIP/(CEN-A)
      GO TO (0501,0502),KLINK
0501  CALL QUICK(X+CX*(I-0.5*(P+Q)),Y+CY*(I+0.5*(P-Q)),IHL)
      GO TO 0503
0502  CALL QUICK(X+DX*(CCN-D)/(C-D),Y+DY,IHL)
      CALL QUICK(X+CX*(I-P),Y+CY*(I+P),IHK)
0503  CALL QUICK(X+CX*(I-Q),Y+CY*(I-Q),IHK)
      CALL QUICK(X+DX*(CON-A)/(B-A),Y,IHK)
      RETURN
406   P = PIP/(CEN-C)
      Q = PIP/(CEN-B)
      GO TO (0601,0602),KLINK
0601  CALL QUICK(X+CX*(I+0.5*(P+Q)),Y+CY*(I+0.5*(P-Q)),IHL)
      GO TO 0603
0602  CALL QUICK(X+DX*(CCN-C)/(C-D),Y+DY,IHL)
      CALL QUICK(X+CX*(I+P),Y+CY*(I+P),IHK)
0603  CALL QUICK(X+CX*(I+Q),Y+CY*(I-Q),IHK)
      CALL QUICK(X+DX*(CON-A)/(B-A),Y,IHK)
      RETURN
407   CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      P = PIP/(CEN-A)
      CALL QUICK(X+CX*(I-P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 207
      CALL QUICK(X+DX*(CCN-A)/(B-A),Y,IHK)
207   RETURN
408   CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      P = PIP/(CEN-C)
      CALL QUICK(X+CX*(I+P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 208
      CALL QUICK(X+DX*(CCN-A)/(B-A),Y,IHK)
208   RETURN
409   CALL QUICK(X+DX*(CCN-D)/(C-D),Y+DY,IHL)
      P = PIP/(CEN-A)
      CALL QUICK(X+CX*(I-P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 209
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
209   RETURN
410   CALL QUICK(X+DX*(CCN-D)/(C-D),Y+DY,IHL)
      P = PIP/(CEN-C)
      CALL QUICK(X+CX*(I+P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 210
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
210   RETURN
411   P = PIP/(CEN-A)
      Q = PIP/(CEN-B)
      GO TO (1101,1102),KLINK
1101  CALL QUICK(X+CX*(I-0.5*(P-Q)),Y+CY*(I-0.5*(P+Q)),IHL)
      GO TO 1103
1102  CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      CALL QUICK(X+CX*(I-P),Y+CY*(I-P),IHK)
1103  CALL QUICK(X+CX*(I+Q),Y+CY*(I-Q),IHK)
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
      RETURN
412   P = PIP/(CEN-D)
      Q = PIP/(CEN-C)
      GO TO (1201,1202),KLINK
1201  CALL QUICK(X+CX*(I-0.5*(P-Q)),Y+CY*(I+0.5*(P+Q)),IHL)
      GO TO 1203
1202  CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      CALL QUICK(X+CX*(I-P),Y+CY*(I+P),IHK)
1203  CALL QUICK(X+CX*(I+Q),Y+CY*(I+Q),IHK)
      CALL QUICK(X+DX,Y+DY*(CON-B)/(C-B),IHK)
      RETURN
413   CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      P = PIP/(CEN-B)
      CALL QUICK(X+CX*(I+P),Y+CY*(I-P),IHK)
      IF(KLINK.NE.2) GO TO 211
      CALL QUICK(X+DX*(CCN-D)/(C-D),Y+DY,IHK)
211   RETURN
414   CALL QUICK(X,Y+DY*(CCN-A)/(C-A),IHL)
      P = PIP/(CEN-D)
      CALL QUICK(X+CX*(I-P),Y+CY*(I+P),IHK)
      IF(KLINK.NE.2) GO TO 212
      CALL QUICK(X+DX*(CCN-D)/(C-D),Y+DY,IHK)
212   RETURN
415   CONTINUE
416   CONTINUE
      RETURN
      END
C
C               * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C
C
      SUBROUTINE QUICK(XIND,YIND,IND)
      COMMON /CONT/ CX,CY,COSG,SING,DX,DY,CX,CY,X,Y,A,B,C,D,CON,IHL,IHK
      PX=100.*(DX+XIND+YIND*COSG)
      PY=100.*(DY+YIND*SING)
      CALL QUACK(PX,PY,IND)
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE GRFGPV(N,X,Y)
C
C
C     GRAPHS OUT GROUP VELOCITY VALUES
C
      COMMON /GRFF/ ATITLE(20), XMAX, XMIN, YMAX, YMIN, INCX, INDY, INC,
     1ICCT, ANSTR1, IF, XLIMIT, YLIMIT, SCALX, SCALY, XFACTR, YFACTR
      DIMENSION X(1),Y(1)
C
      DIMENSION TITLE(20)
      DIMENSION TITLX(20)
C
      REAL*8 ATITLE,TITLE,TITLX
C
      DATA TITLE/8HPERIOD 1,8HN SECOND,8HS        ,8HGRP.VEL.,8H KM/SEC ,
     18HGRAPH OF,8HGROUP VE,8HLOCITY (,8HKM/SEC) ,8HAGAINST ,8HPERIOD (,
     28HSEC) FOR,8H SEISMOG,8HRAM 1    ,8H        ,8HDATE    ,8HTIME    ,
     38H        ,8H        ,8H        /
      DATA TITLX/8HPERIOD 1,8HN SECOND,8HS        ,8HGRP.VEL.,8H KM/SEC ,
     18HGRAPH OF,8HGROUP VE,8HLOCITY (,8HKM/SEC) ,8HAGAINST ,8HPERIOD (,
     28HSEC) FOR,8H SEISMOG,8HRAM 2    ,8H        ,8HDATE    ,8HTIME    ,
     38H        ,8H        ,8H        /
C
      DATA NH/0/
C
C
      CALL DATIM(TITLE(16),TITLE(17))
      CALL DATIM(TITLX(16),TITLX(17))
C
      NH=NH+1
      GO TO (1,3),NH
C
 1    DO 2 I=1,16
      ATITLE(I)=TITLE(I)
 2    CONTINUE
      GO TO 5
C
 3    DO 4 I=1,16
      ATITLE(I)=TITLX(I)
 4    CONTINUE
      NH=0
C
 5    XMAX=0.0
      XMIN=0.0
      YMAX=0.0
      YMIN=0.0
      INCX=1
      INCY=1
      IND=1
      ICCT=42
      ANSTR1=2.0
      IF=3
      CALL CARGRF(X,Y,N)
C
      RETURN
      END
C
C              * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE COOL(N,XX,SIGNI)
C
C     A SPECIAL VERSION USING THE DOUBLE PRECISION COOL PROGRAMME
C
C     N.B.  ARRAY  XX  MUST HAVE TWICE AS MUCH STORAGE ALLOCATED AS USED
C
C
      DIMENSION XX(1)
C
      NX=2**(N+1)
      NXD=NX*2
      DO 1 I=1,NX
      J=NX-I+1
      JD=NXD-I*2+1
      XX(JD)=XX(J)
      XX(JD+1)=0.
 1    CONTINUE
```

```
C
      CALL CCOCL(N,XX,SIGNI)
C
      DO 2 J=1,NX
      JD=J*2-1
      XX(J)=XX(JD)
 2    CONTINUE
C
      RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * *
C
C
      SUBROUTINE DCLCL(N,XX,SIGNI)
C
C     THIS SUBROUTINE WAS PROGRAMMED BY I.MACLEOD, DEPT. OF
C     ENGINEERING PHYSICS,A.N.U. AND HAS BORROWED FROM C. MCCOWAN'S
C     COOL AND IBM'S HARM.
C
C     DOUBLE PRECISION VERSION MODIFIED BY J.B.YOUNG FOR THE 360/75.
C
C
C
      DIMENSION W(14),XX(1),NBIT(20),JNT(20)
C
C
      DOUBLE PRECISION XX,ROOT2,PI2,CON1,ARG,W,CSSQA,CSSQ2A,CSSQ3A,
     1A0R,A0I,A1R,A1I,A2R,A2I,A3R,A3I,A4R,A4I,A5R,A5I,A6R,A6I,A7R,A7I,
     2A8R,A8I,XK0WR,XK0WI,XK1WR,XK1WI,XK2WR,XK2WI,XK3WR,XK3WI,
     3XK4WR,XK4WI,XK5WR,XK5WI,XK6WR,XK6WI,XK7WR,XK7WI,HOLDR,HOLDI
C
C
      INTEGER OFFSET
C
C
      DATA NX/0/
C
C
      IF(NX.GT.0)GO TO 100
      ROOT2=DSQRT(2.0D0)
      PI2=8.0D0*DATAN(1.0D0)
C
 100  NX=2**N
      NX2=NX+NX
```

```
      NX2LS1=NX2-1
      NX2LS2=NX2-2
      NXON8=NX/8
      NXON4=NXON8+NXON8
      NXON2=NXON4+NXON4
      CON1=PI2/DFLOAT(NX)
      IF(SIGNI.GT.0.0)GO TO 120
C
      DO 110 K=1,NX2LS1,2
      XX(K+1)=-XX(K+1)
 110  CONTINUE
C
 120  DO 130 K=1,N
      JNT(K)=2**(N-K)
 130  CONTINUE
C
      LSTART=N-N/3*3+1
      IF(LSTART.EQ.1)GO TO 200
      IF(LSTART.EQ.2)GO TO 150
      LBLOK2=NXON2
      L2BLOK=LBLOK2-1
C
      DO 140 K0=1,L2BLOK,2
      K1=K0+LBLOK2
      K2=K1+LBLOK2
      K3=K2+LBLOK2
      A0R=XX(K0)+XX(K2)
      A0I=XX(K0+1)+XX(K2+1)
      A1R=XX(K0)-XX(K2)
      A1I=XX(K0+1)-XX(K2+1)
      A2R=XX(K1)+XX(K3)
      A2I=XX(K1+1)+XX(K3+1)
      A3R=XX(K1)-XX(K3)
      A3I=XX(K1+1)-XX(K3+1)
      XX(K0)=A0R+A2R
      XX(K0+1)=A0I+A2I
      XX(K1)=A0R-A2R
      XX(K1+1)=A0I-A2I
      XX(K2)=A1R-A3I
      XX(K2+1)=A1I+A3R
      XX(K3)=A1R+A3I
      XX(K3+1)=A1I-A3R
 140  CONTINUE
      GO TO 200
C
```

```
 150  LBLOK2=NX
      L2BLOK=LBLOK2-1
      DO 160 KO=1,L2BLOK,2
      K1=KO+LBLOK2
      A1R=XX(K1)
      A1I=XX(K1+1)
      XX(K1)=XX(KO)-A1R
      XX(K1+1)=XX(KO+1)-A1I
      XX(KO)=XX(KO)+A1R
      XX(KO+1)=XX(KO+1)+A1I
 160  CONTINUE
C
C
 200  DO 300 M=LSTART,N,3
      LBLOK2=NX/2**(M+1)
      L2BLOK=LBLOK2-1
      LBLCK1=L2BLOK-1
      LBLCK8=LBLCK2*8
      LBLAST=NX2-LBLCK8+1
C
      DO 210 K=4,N
      NBIT(K)=0
 210  CONTINUE
C
      NW=0
C
      DO 290 OFFSET=1,LBLAST,LBLOK8
      IF(OFFSET.EQ.1)GO TO 220
      ARG=CON1*DFLOAT(NW)
      W(1)=DCOS(ARG)
      W(2)=DSIN(ARG)
      CSSQA=W(1)*W(1)
      W(3)=CSSQA+CSSQA-1.0D0
      W(4)=W(1)*W(2)
      W(4)=W(4)+W(4)
      W(5)=W(3)*W(1)-W(4)*W(2)
      W(6)=W(4)*W(1)+W(3)*W(2)
      CSSQ2A=W(3)*W(3)
      W(7)=CSSQ2A+CSSQ2A-1.0D0
      W(8)=W(4)*W(3)
      W(8)=W(8)+W(8)
      W(9)=W(7)*W(1)-W(8)*W(2)
      W(10)=W(8)*W(1)+W(7)*W(2)
      CSSQ3A=W(5)*W(5)
      W(11)=CSSQ3A+CSSQ3A-1.0D0
      W(12)=W(6)*W(5)
      W(12)=W(12)+W(12)
      W(13)=W(7)*W(5)-W(8)*W(6)
      W(14)=W(8)*W(5)+W(7)*W(6)
 220  LBLOKO=OFFSET+LBLCK1
C
      DO 260 KO=OFFSET,LBLOKO,2
      K1=KO+LBLOK2
      K2=K1+LBLOK2
      K3=K2+LBLOK2
      K4=K3+LBLCK2
      K5=K4+LBLOK2
      K6=K5+LBLOK2
      K7=K6+LBLOK2
      XKOWR=XX(KO)
      XKOWI=XX(KO+1)
      IF(OFFSET.NE.1) GO TO 240
      XK1WR=XX(K1)
      XK1WI=XX(K1+1)
      XK2WR=XX(K2)
      XK2WI=XX(K2+1)
      XK3WR=XX(K3)
      XK3WI=XX(K3+1)
      XK4WR=XX(K4)
      XK4WI=XX(K4+1)
      XK5WR=XX(K5)
      XK5WI=XX(K5+1)
      XK6WR=XX(K6)
      XK6WI=XX(K6+1)
      XK7WR=XX(K7)
      XK7WI=XX(K7+1)
      GO TO 250
 240  XK1WR=XX(K1)*W(1)-XX(K1+1)*W(2)
      XK1WI=XX(K1)*W(2)+XX(K1+1)*W(1)
      XK2WR=XX(K2)*W(3)-XX(K2+1)*W(4)
      XK2WI=XX(K2)*W(4)+XX(K2+1)*W(3)
      XK3WR=XX(K3)*W(5)-XX(K3+1)*W(6)
      XK3WI=XX(K3)*W(6)+XX(K3+1)*W(5)
      XK4WR=XX(K4)*W(7)-XX(K4+1)*W(8)
      XK4WI=XX(K4)*W(8)+XX(K4+1)*W(7)
      XK5WR=XX(K5)*W(9)-XX(K5+1)*W(10)
      XK5WI=XX(K5)*W(10)+XX(K5+1)*W(9)
      XK6WR=XX(K6)*W(11)-XX(K6+1)*W(12)
      XK6WI=XX(K6)*W(12)+XX(K6+1)*W(11)
      XK7WR=XX(K7)*W(13)-XX(K7+1)*W(14)
```

```
      XK7WI=XX(K7)*W(14)+XX(K7+1)*W(13)
 250  AOR=XK0WR+XK4WR
      AOI=XK0WI+XK4WI
      A1R=XK1WR+XK5WR
      A1I=XK1WI+XK5WI
      A2R=XK2WR+XK6WR
      A2I=XK2WI+XK6WI
      A3R=XK3WR+XK7WR
      A3I=XK3WI+XK7WI
      A4R=AOR+A2R
      A4I=AOI+A2I
      A5R=AOR-A2R
      A5I=AOI-A2I
      A6R=A1R+A3R
      A6I=A1I+A3I
      A7R=A3I-A1I
      A7I=A1R-A3R
      XX(KO)=A4R+A6R
      XX(KO+1)=A4I+A6I
      XX(K1)=A4R-A6R
      XX(K1+1)=A4I-A6I
      XX(K2)=A5R+A7R
      XX(K2+1)=A5I+A7I
      XX(K3)=A5R-A7R
      XX(K3+1)=A5I-A7I
      ACR=XK0WR-XK4WR
      ACI=XK0WI-XK4WI
      A8R=XK1WR-XK5WR
      A8I=XK1WI-XK5WI
      A1R=A8R-A8I
      A1I=A8R+A8I
      A2R=XK6WI-XK2WI
      A2I=XK2WR-XK6WR
      A8R=XK3WR-XK7WR
      A8I=XK3WI-XK7WI
      A3R=A8R-A8I
      A3I=A8R+A8I
      A4R=AOR+A2R
      A4I=AOI+A2I
      A5R=ACR-A2R
      A5I=ACI-A2I
      A6R=(A1R-A3I)/RCOT2
      A6I=(A1I+A3R)/RCOT2
      A7R=(A3R-A1I)/RCOT2
      A7I=(A3I+A1R)/RCOT2


      XX(K4)=A4R+A6R
      XX(K4+1)=A4I+A6I
      XX(K5)=A4R-A6R
      XX(K5+1)=A4I-A6I
      XX(K6)=A5R+A7R
      XX(K6+1)=A5I+A7I
      XX(K7)=A5R-A7R
      XX(K7+1)=A5I-A7I
 260  CONTINUE
C
      DO 280 K=4,N
      IF(NBIT(K).NE.0)GO TO 270
      NBIT(K)=1
      NW=NW+JNT(K)
      GO TO 290
 270  NBIT(K)=0
      NW=NW-JNT(K)
 280  CONTINUE
C
 290  CONTINUE
 300  CONTINUE
C
C
      NW=0
C
      DO 310 K=1,N
      JNT(K)=JNT(K)+JNT(K)
      NBIT(K)=0
 310  CONTINUE
C
      K=0
      IF(NW.LE.K)GO TO 320
      HOLDR=XX(NW+1)
      HOLDI=XX(NW+2)
      XX(NW+1)=XX(1)
      XX(NW+2)=XX(2)
      XX(1)=HOLDR
      XX(2)=HOLDI
C
 320  DO 340 M=1,N
      IF(NBIT(M).NE.0)GO TO 330
      NBIT(M)=1
      NW=NW+JNT(M)
      GO TO 350
 330  NBIT(M)=0
```

```
      NW=NW-JNT(M)
 340  CONTINUE
C
 350  DO 390 K=2,NX2LS2,2
      IF(NW.LE.K)GO TO 360
      HOLDR=XX(NW+1)
      HOLDI=XX(NW+2)
      XX(NW+1)=XX(K+1)
      XX(NW+2)=XX(K+2)
      XX(K+1)=HOLDR
      XX(K+2)=HOLDI
C
 360  DO 380 M=1,N
      IF(NBIT(M).NE.0)GO TO 370
      NBIT(M)=1
      NW=NW+JNT(M)
      GO TO 390
 370  NBIT(M)=0
      NW=NW-JNT(M)
 380  CONTINUE
C
 390  CONTINUE
C
      IF(SIGNI.GT.0.0)GO TO 420
C
      DO 410 K=1,NX2LS1,2
      XX(K+1)=-XX(K+1)
 410  CONTINUE
C
C
 420  RETURN
      END
C
C
C
C                 * * * * * * * * * * * * * * * * * * * * * * * * *
C                    GENERAL USER SUBROUTINE TIMER AND CHECK
C                    ---------------------------------------
C
C
C     FROM THE CALL  -  CALL TIMER
C     THE ELAPSED TIME AND CPU TIMES FROM THE LAST CALL TIMER ARE PRINTED OUT
C     THE FIRST CALL SETS UP TIMER
C
C
C
C     FROM THE CALL  -  CALL CHECK(&....)
C     TIMES ARE PRINTED OUT AS FOR CALL TIMER BUT WHEN THE CPUAV
C     IS LARGER THAN THE CPU TIME LEFT CONTROL IS RETURNED TO
C     STATEMENT NUMBER .... IN THE CALLING PROGRAM
C
C
C
      SUBROUTINE TIMER
      DATA DIFF/0./
      IND=1
      IF(DIFF) 2,1,2
C
      ENTRY CHECK(*)
      IND=2
      IF(DIFF) 2,1,2
C
 1    CALL CLOCK(DIFF)
      DIFF=DIFF*60.
      CALL TXLIBR(CTIME)
      CTIME=CTIME*0.01
      CPUAV=CTIME*0.2
      RETURN
C
 2    CALL CLOCK(TIME)
      TIME=TIME*60.
      DIFF=TIME-DIFF
      CALL TXLIBR(CPU)
      CPU=CPU*0.01
      CTIME=CTIME-CPU
      CPUAV=CPUAV*0.8+CTIME*0.2
      PRINT 3, DIFF,CTIME,CPU
 3    FORMAT(41H0FROM LAST TIMER CALL   1) TIME ELAPSED =,F8.3,29H SECON
     1DS   2) CPU TIME USED =,F8.3,29H SECONDS   3) CPU TIME LEFT =,F8.3
     2,8H SECONDS/)
      DIFF=TIME
      CTIME=CPU
C
      GO TO (6,4),IND
 4    IF(CPUAV-CPU)6,6,5
 5    RETURN 1
C
 6    RETURN
      END
C
C                 * * * * * * * * * * * * * * * * * * * * * * * * *
```

DOCUMENT CONTROL SHEET

Overall security classification of sheet .....Unclassified.....

(As far as possible this sheet should contain only unclassified information. If it is necessary to enter classified information, the box concerned must be marked to indicate the classification eg (R), (C) or (S)).

1. DRIC Reference (if known)	2. Originator's Reference	3. Agency Reference	4. Report Security Classification
-	AWRE Report No. 08/76	-	On Sale

5. Originator's Code (if known)	6. Originator (Corporate Author) Name and Location
-	Atomic Weapons Research Establishment, Aldermaston, Berkshire
5a. Sponsoring Agency's Code (if known)	**6a. Sponsoring Agency (Contract Authority) Name and Location**
-	-

7. Title

A Computer Program for the Determination of the Phase Velocity of Seismic Surface Waves between Pairs of Stations

7a. Title in Foreign Language (in the case of Translation)

-

7b. Presented at (for Conference Papers). Title, Place and Date of Conference

-

8. Author 1.Surname, Initials	9a. Author 2	9b. Authors 3, 4	10. Date	pp	ref
Stuart G W	Douglas A	Blamey C	May 1976	44	6

11. Contract Number	12. Period	13. Project	14. Other References
-	-	-	-

15. Distribution Statement

No restriction

16. Descriptors (or Keywords) (TEST)

Seismology, Earthquakes, Seismic Waves, Surface Waves, Seismological Stations, Phase Velocity, Computer Programs

Abstract

The program described determines the phase velocity of a dispersed seismic surface wave as a function of period between a pair of stations from their digitized seismograms. The cross-multiplication technique of Bloch and Hales is used to compute the phase velocity curve.