AN ABSTRACT OF THE THESIS OF

KEITH H. WROLSTAD for the degree DOCTOR OF PHILOSOPHY
(Name of Student) (Degree)

GEOPHYSICS presented on AUGUST 4, 1978
(Major) (Date)

Title: Applications of Source Signature Deconvolution to Air Gun Seismic Profiling and the Measurement of Attenuation from Reflection Seismograms.

Abstract approved: Redacted for privacy

Deterministic source signature deconvolution is applied to the processing of marine wide angle and vertical profiler data with air-gun sources. Optimum results are obtained with a source signature measured by stacking the signal reflected from a relatively homogeneous abyssal plain sedimentary environment. This eliminates the need for the unstable inverse source-receiver ghost filter. Improved resolution of reflection event timing allows the computation of more reliable interval velocities by the $T^2 - X^2$ method, provided the layer thickness limitation of the method is not exceeded. Accurate timing of primary reflection events in the deconvolved vertical profiler data permits computation of frequency dependent attenuation by univariate least-squares regression in the Fourier transform domain. The technique successfully extracts input amplitude attenuation functions from model reflection coefficient sequences with additive random noise. This success is attributed to the stability of singular value analysis in solving the least-squares regression model.
Statistical tests on the solution vectors for model and field data give criteria for evaluating their reliability. The model data studies suggest that multiple and primary events not included in the regression may be considered part of the noise term without seriously affecting the accuracy of the computed spectral ratios.

The method is tested on field data from the following sedimentary environments off the coast of Oregon and northern California: a continental shelf basin, an abyssal plain environment, the base of the continental slope and two locations on the Astoria sea fan, one near the Cascadia sea channel and one north of DSDP site 174. Velocity versus depth and frequency dependent spectral ratio plots are determined for each environment. The computed surface layer interval velocity of 1.77 km/sec over a thickness of 455 m for the station north of DSDP site 174 is in good agreement with the average material type found in the drill core (sandy-silt with greater than 60% sand). Maximum attenuation coefficients are estimated from the spectral ratios for the upper sediment intervals of the study areas using typical acoustic impedance values of surface sediment types determined from nearby piston cores. Some maximum attenuation coefficients are too high suggesting the possibility of a stratigraphic component. The maximum attenuation in the upper interval for SB 46 over the Tufts abyssal plain where fine-grained material (silts and clays) is expected is 0.025 dB/m at 127 Hz compared with 0.004 dB/m at 80 Hz for the upper interval of the turbidite environment north of DSDP site 174.
Applications of Source Signature Deconvolution to Airgun Seismic Profiling and the Measurement of Attenuation from Reflection Seismograms

by

Keith H. Wrolstad

A THESIS submitted to Oregon State University

in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Completed August 4, 1978

Commencement June 1979
APPROVED:

Redacted for privacy

Associate Professor of OceanPhysics
in charge of major

Redacted for privacy

Chairman of the School of Oceanography

Redacted for privacy

Dean of the Graduate School

Date thesis is presented August 4, 1978

Typed by Rebecca Rakish for Keith H. Wrolstad
PLEASE NOTE:

Some appendix pages have very small and indistinct print. Filmed in the best possible way.

UNIVERSITY MICROFILMS.
ACKNOWLEDGEMENTS

This research could not have been successfully completed without the support and encouragement of my major professor, Dr. Stephen H. Johnson. I am grateful for his advice and for critically reviewing the manuscript.

Discussions with several professors were important to the development of the method presented in this dissertation. In this regard, sincere appreciation is extended to Dr. F. Thomas Lindstrom and to Dr. Richard K. Johnson. Thanks also go to Dr. Gunnar Bodvarsson for discussions which were helpful in organizing concepts. My sincere gratitude is extended to fellow graduate students Jonathan M. Hanson, Paul R. Jones and Robert Karlin for their contributions and helpful advice. I am indebted also to Rick McAlister for the digitizing equipment and for providing much necessary technical information, to Bart Brown for helping with the instrumentation and to Ken Keeling and Gerry Connard for invaluable assistance with the computer and computer programs. Sincere thanks go to Barbara Priest and Darcy Burt for drafting the figures, to Dave Reinert for photographic work and to Becky Rakish for her patience in typing the manuscript. Appreciation is extended to the geophysics staff and students of OSU and to the personnel of the R/V Yaquina and R/V Wecoma for their hard work and expertise in the efficient collection of data at sea.

I am very grateful to my wife, Erma, for her patience and unselfish support during this educational period.

This research was supported by the United States Navy under ONR contract N00014-76-C-0067.
# TABLE OF CONTENTS

Chapter 1. INTRODUCTION

Background Information and Purpose of this Research ................................................. 1
Description of Deconvolution Techniques ......................................................................... 3
Wavelet or Source Signature Estimation ........................................................................... 11
Acoustic Velocity and Attenuation in Sediments and Rocks ........................................... 14
Attenuation Mechanisms ................................................................................................. 17
The Measurement of Attenuation from Seismic Reflection Data ....................................... 20
Method of Approach ........................................................................................................ 23

Chapter 2. METHOD AND DERIVATIONS

The Reflection Seismogram .............................................................................................. 26
Filters in the Linear System ............................................................................................. 29
Source-Receiver Directivity Function ............................................................................. 30
Instrument Transfer Function ......................................................................................... 32
Surface Ghost Transfer Function .................................................................................... 33
Geometric Spreading ....................................................................................................... 38
Acoustic Wave Propagation in a Homogeneous Half Space ............................................. 38
Propagation in a Layered Half Space ............................................................................. 40
Proposed Method for the Measurement of Attenuation in Reflection Seismograms ....... 45
Singular Value Analysis ................................................................................................. 49
Error Analysis .................................................................................................................. 52

Chapter 3. EXPERIMENTAL RESULTS

Section 1. Model Data
Source Signature Deconvolution .................................................................................... 59
Frequency Domain Least-Squares Regression ................................................................. 63
Bandpass Filtering .......................................................................................................... 72
Frequency Independent Phase Shifts .............................................................................. 72
Inverse Ghost Filtering ................................................................................................... 75

Section 2. Field Data
Data Processing Procedure ............................................................................................. 80
Noise Suppression ............................................................................................................ 88
Interval Velocity Analysis ............................................................................................... 92
Attenuation Analysis ........................................................................................................ 94
Analysis and Discussion of Sedimentary Environments ................................................ 95
Sonobuoy 3 .................................................................................................................... 95
Sonobuoy 5 .................................................................................................................... 101
Sonobuoy 46 .................................................................................................................. 105
Sonobuoy 57 .................................................................................................................. 111
Sonobuoy 61 .................................................................................................................. 115
Chapter 4. DISCUSSION AND CONCLUSIONS

Discussion of the Reliability of Solution Vectors for Field Data 119
Conclusions 124

BIBLIOGRAPHY 127

APPENDICES

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Formulas for Interval Velocity Computations</td>
<td>133</td>
</tr>
<tr>
<td>2.</td>
<td>Derivations of the Normal Equations for a Least Squares Polynomial Fit.</td>
<td>142</td>
</tr>
<tr>
<td>3.</td>
<td>Frequency Domain Bandpass and Notch Filters</td>
<td>144</td>
</tr>
<tr>
<td>4.</td>
<td>Inverse Ghost Filter</td>
<td>146</td>
</tr>
<tr>
<td>5.</td>
<td>Trigonometry of Ghost Delay Calculations.</td>
<td>149</td>
</tr>
<tr>
<td>6.</td>
<td>Frequency Independent Phase Changes</td>
<td>152</td>
</tr>
<tr>
<td>7.</td>
<td>Derivation of the Normal Equations for Least Squares Regression of a Linear System With Complex Numbers.</td>
<td>159</td>
</tr>
<tr>
<td>8.</td>
<td>Error Propagation for Average Complex Moduli of Spectral Ratios.</td>
<td>163</td>
</tr>
<tr>
<td>9.</td>
<td>Input and Processed Field Data.</td>
<td>166</td>
</tr>
<tr>
<td>10.</td>
<td>Program Listings and Flow Charts.</td>
<td>184</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1.</td>
<td>Source-receiver geometry of vertical profiler and wide angle reflection systems.</td>
<td>24</td>
</tr>
<tr>
<td>2.</td>
<td>Streamer directivity function geometry.</td>
<td>31</td>
</tr>
<tr>
<td>3.</td>
<td>Amplitude and phase spectrum of streamer and sonobuoy directivity functions.</td>
<td>32</td>
</tr>
<tr>
<td>4.</td>
<td>Theoretical attenuation vs. frequency in sands and muds or clays according to Stoll (1977).</td>
<td>40</td>
</tr>
<tr>
<td>5.</td>
<td>Layer model for attenuation calculations.</td>
<td>46</td>
</tr>
<tr>
<td>6.</td>
<td>Arbitrary source function for model data and its power spectrum.</td>
<td>61</td>
</tr>
<tr>
<td>7.</td>
<td>Synthetic seismogram of Model #1 with no noise; includes frequency dependent attenuation.</td>
<td>61</td>
</tr>
<tr>
<td>8.</td>
<td>Power spectrum of signal on model #1.</td>
<td>62</td>
</tr>
<tr>
<td>9.</td>
<td>Impulse response of Model #1 with no noise.</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>(a) with frequency dependent attenuation;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b) with attenuation plus simulated dispersion.</td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Impulse response of model #1(a) with 20% relative RMS noise level; no bandpass filtering.</td>
<td>64</td>
</tr>
<tr>
<td>11.</td>
<td>Results of least squares regression on Model 1(a).</td>
<td>65</td>
</tr>
<tr>
<td>12.</td>
<td>Results of least squares regression on Model 1(b).</td>
<td>66</td>
</tr>
<tr>
<td>13.</td>
<td>Impulse response of model 2; 10% relative RMS noise; no bandpass filtering.</td>
<td>69</td>
</tr>
<tr>
<td>14.</td>
<td>Results of least squares regression on model 2.</td>
<td>70</td>
</tr>
<tr>
<td>15.</td>
<td>Bandpass filtering of Model 1(a) (0%-68.5% Nyquist)</td>
<td>73</td>
</tr>
<tr>
<td>16.</td>
<td>Bandpass filtering of Model 1(a) (0%-38.5% Nyquist)</td>
<td>73</td>
</tr>
<tr>
<td>17.</td>
<td>Source signature deconvolution of model with frequency independent phase shifts.</td>
<td>74</td>
</tr>
<tr>
<td>18.</td>
<td>Amplitude spectrum of surface interference effect.</td>
<td>75</td>
</tr>
<tr>
<td>19.</td>
<td>Synthetic seismogram of the impulse response of a source-receiver ghost model convolved with an arbitrary source wavelet.</td>
<td>76</td>
</tr>
</tbody>
</table>
20. Impulse response of source-receiver ghost model; ghost delays at 7 and 11 data points. 

21. Impulse response of source-receiver ghost model after application of inverse ghost filter with truncation point at 6% of Nyquist. 

22. Impulse response of source-receiver ghost model after application of inverse ghost filter with truncation point at 2% of Nyquist. 

23. Impulse response of source-receiver ghost model with one ghost delay in error; truncation point at 6% of Nyquist. 

24. Impulse response of source-receiver ghost model with two ghost delays in error; truncation point at 6% of Nyquist. 

25. Block diagram for digitization of analogue-recorded data. 

26. Data processing flow chart. 

27. Stacked streamer source signature for SB 3; 30 traces. 

28. Typical power spectrum of stacked direct wave recorded by streamer. 

29. Stacked source signature received by SB 3; 5 traces; used for SB 3 WAR deconvolution. 

30. Amplitude spectrum of SB 3 WAR source signature. 

31. Stacked bottom reflected arrival at SB 46; 21 traces; for deconvolution of SB 46, 57 and 61. 

32. Amplitude spectrum of 21 trace stack of bottom reflected arrival at SB 46. 

33. Spectrum of signal in a single trace received by SB 3; (a) without and (b) with noise subtraction. 

34. Field data processing examples. 

35. Map showing location of tracklines, sonobuoys and core sites. 

36. Time section and velocities for SB 3.
<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>37.</td>
<td>Velocity vs. depth and spectral ratios for SB 3.</td>
</tr>
<tr>
<td>38.</td>
<td>Time section and interval velocities for SB 5 over the upper Astoria sea fan.</td>
</tr>
<tr>
<td>39.</td>
<td>Velocity versus depth and average moduli of complex spectral ratios relative to arrival from sea floor for SB 5; spectral ratios are not corrected for geometric spreading.</td>
</tr>
<tr>
<td>40.</td>
<td>Attenuation data compiled by Hamilton (1974)</td>
</tr>
<tr>
<td>41.</td>
<td>Time section and interval velocities for SB 46 over the Tufts abyssal plain near the west end of the Blanco fracture zone.</td>
</tr>
<tr>
<td>42.</td>
<td>Velocity versus depth and average spectral ratios relative to bottom arrival for SB 46.</td>
</tr>
<tr>
<td>43.</td>
<td>Velocity versus depth and average spectral ratios relative to layer above for SB 46.</td>
</tr>
<tr>
<td>44.</td>
<td>Time section and interval velocities for SB 57 near the Cascadia sea channel north of the Blanco fracture zone.</td>
</tr>
<tr>
<td>45.</td>
<td>Velocity versus depth and spectral ratios relative to arrival from sea floor for SB 57.</td>
</tr>
<tr>
<td>46.</td>
<td>Velocity versus depth and spectral ratios relative to arrival from layer above for SB 57.</td>
</tr>
<tr>
<td>47.</td>
<td>Power spectrum of single trace of signal received by streamer for SB 57.</td>
</tr>
<tr>
<td>48.</td>
<td>Time section and interval velocities for SB 61 at the base of the Oregon continental slope.</td>
</tr>
<tr>
<td>49.</td>
<td>Velocity versus depth and spectral ratios relative to arrival from sea floor for SB 61.</td>
</tr>
<tr>
<td>50.</td>
<td>Velocity versus depth and spectral ratios relative to layer above for SB 61.</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>1.</td>
<td>F-test for significance of regression.</td>
</tr>
<tr>
<td>2.</td>
<td>F-test for extra sum of squares for model 2.</td>
</tr>
<tr>
<td>3.</td>
<td>Parameters for attenuation estimates.</td>
</tr>
<tr>
<td>4.</td>
<td>Complete error analysis for sonobuoy 3.</td>
</tr>
<tr>
<td>5.</td>
<td>Complete error analysis for sonobuoy 46.</td>
</tr>
<tr>
<td>6.</td>
<td>Perturbation bounds for the solution vectors.</td>
</tr>
</tbody>
</table>
APPLICATIONS OF SOURCE SIGNATURE DECONVOLUTION TO AIRGUN SEISMIC PROFILING AND THE MEASUREMENT OF ATTENUATION FROM REFLECTION SEISMOGRAMS

INTRODUCTION

CHAPTER 1

An important goal of exploration seismology is the determination of near-surface crustal structure using reflection and refraction methods with artificial acoustic sources. The variation of velocity with depth may be computed from travel times of seismic pulses reflected from subsurface interfaces separating material with different densities and acoustic velocities. A common technique is to detect and accurately measure these arrival times at various distances from the source. Reflection events in a seismogram are often obscured, however, by noise and reverberation of the oscillatory source wavelet or signature. In practice the source signature is rarely a pure spike or delta function, especially in the marine environment where the airgun bubble pulse lasts for a considerable fraction of a second. Data processing methods are therefore usually employed to enhance the resolution of these events. These methods are commonly called deconvolution because the data is processed to remove the overlapping oscillations of the source wavelet resulting in impulses which ideally represent only the primary and reverberating reflection events as spikes.

In addition to the timing of reflection events, seismic energy is also of diagnostic value (Hermont, 1969). Recognition that amplitudes relay important information on subsurface lithology led to an important advance in exploration geophysics which came to be known as "bright spot" technology. Large amplitude reflections which can be indicators
of gas and oil deposits had previously gone unnoticed because of the widespread use of automatic gain control in seismic data collection. Currently, the measurement of frequency dependent amplitude information from seismic reflection data is an important topic of research (Sheriff, 1978), because of the need to gain more information about the subsurface from the seismic waveform. Not all physical properties of the bulk medium can be measured directly from the seismogram, but attenuation can in principle be measured as a lumped parameter. This information in conjunction with velocity, reflection coefficient and density logs will provide additional clues for making judgments about subsurface lithology and structure. These judgments will also depend on a reliable compilation of velocity and attenuation data for various media in particular geologic provinces.

Seismic reflection data processing in the oil exploration industry has largely been concerned with enhancing the resolution of events by statistical techniques which ignore frequency dependent amplitude information in the seismogram. It is the purpose of this thesis research to study the advantages and problems associated with the application of deterministic source signature deconvolution techniques to the processing of marine wide angle and vertical profiler data to obtain event timing resolution as well as frequency dependent amplitude information. The ultimate purpose of this research is twofold: the first is to improve the resolution of acoustic impedance fine structure in various marine sedimentary environments and the second is to test the effectiveness of a technique to measure frequency dependent attenuation from the reflection seismogram using event timing as input.
It will be shown that event resolution and frequency dependent amplitude information can both be obtained from the seismogram if deterministic methods are used as much as possible in the data processing, leaving statistical averaging or least squares procedures for the final step to improve the accuracy of amplitude measurements.

A number of methods exist for deconvolving seismic signals and estimating seismic wave attenuation. The following is background information on the most common techniques including their advantages and disadvantages relative to the problem of measuring frequency dependent amplitudes.

Description of Deconvolution Techniques

There are two fundamentally different approaches to the problem of resolving events in a reflection seismogram. One is statistical and commonly employs least squares techniques and the other is deterministic. The methods for inverting or unfolding a seismogram to obtain the crustal impulse response or reflection coefficient series (deconvolution) incorporate both statistical and deterministic assumptions but emphasis will vary. The most common methods are predictive deconvolution, Wiener least-squares inverse filtering, matched filtering and frequency domain source signature deconvolution. Brief descriptions of these techniques are given below.

The statistical approach relies on two basic hypotheses first defined by Robinson (1967). The first is the statistical hypothesis that the strengths and arrival times of the information-bearing events
in the seismic trace, or the reflector series can be represented as a random spike series, and the second is the deterministic hypothesis that the basic wave form associated with each of these events is minimum phase. The minimum phase hypothesis is the frequency or transform domain equivalent of minimum delay in the time domain. A minimum delay waveform is expected in a layered earth model because reflection coefficients are less that unity. Hence repeated reflections delay and attenuate the acoustic pulse causing a concentration of energy to appear at its beginning rather than at its end. An entirely non-deterministic stationary stochastic process has infinitely many representations as a convolution of the general form;

\[ x_t = \sum_{s=0}^{\infty} a_s \epsilon_{t-s} = a_t^* \epsilon_t \]  

(1.1)

where \( a_t \) is a one sided wavelet (i.e. \( a_t = 0 \) for \( t < 0 \)), and \( \epsilon_t \) is a random or white noise time series. As a result, the autocorrelation of the stationary time series \( x_t \) is equal to a constant times the autocorrelation of the wavelet \( a_t^* \), since the autocorrelation of white noise is zero at all lags except zero where it is equal to the variance of the noise series. We can determine uniquely the wavelet from knowledge of the autocorrelation of the series \( x_t \) only if the wavelet is restricted to be minimum delay. The statistical approach incorporates these assumptions and solves for a filter by means of least squares which on the basis of previous data points will predict future data points. In predictive deconvolution the wavelet need not
actually be computed. The received signal is assumed to be a convolution of a minimum delay source wavelet $w_t$ with a minimum delay reflector series $r_t$ plus additive random noise,

$$x_t = w_t * r_t + n_t$$  \quad (1.2)

From the autocorrelation of $x_t$ the prediction error operator for the waveform $w_t$ is computed, and when it is applied to the seismic trace, the predictable or reverberation components of the trace are suppressed leaving the unpredictable components which increase the prediction error and thereby resolve reflection events. Wiener filter theory is the basis for the computation of the filter coefficients. The purpose is to minimize the sums of squares of prediction errors given some prediction distance, $\alpha$,

$$x_{t+\alpha} = y_t = f_0 x_t + f_1 x_{t-1} + \cdots + f_m x_{t-m}$$  \quad (1.3)

The mean-square-error is,

$$I = E \{(x_{t+\alpha} - y_t)^2\} = \sum_{t} (x_{t+\alpha} - y_t)^2.$$  \quad (1.4)

Partial derivatives are taken with respect to the filter coefficients $f_i$ and the results set equal to zero. The coefficients of the prediction error operator $f_m$ are given as the solution of the normal equations

$$f_0 \phi_s + f_1 \phi_{s-1} + \cdots + f_m \phi_{s-m} = \phi_{\alpha+s}$$  \quad (1.5)
for \( s = 0,1,2\ldots m \), where \( \phi_s \) is the autocorrelation of \( x_t \). The prediction error time series is,

\[
e_{t+a} = x_{t+a} - y_t.
\]  

(1.6)

The filter is computed from a segment or window of the input trace and is convolved with the entire trace. If the number of terms in the window of \( x_t \) is not large enough there will be an undesirable biasing effect in the autocorrelation. The method of Burg (1972) solves the normal equations and avoids the end effect problems by estimating a minimum phase prediction error filter directly from the data rather than calculating the autocorrelation. This is accomplished by computing a filter which transforms the input series into a white noise series. The spectrum of the filter is then the inverse of the spectrum of the input series. The method is called maximum entropy spectral analysis. By using the efficient Levinson recursion (for example; Claerbout, 1976, p. 56-57), the prediction errors are minimized and, in the process, a series of coefficients \( r_t \) are generated which are indicative of the unpredictable points in the series \( x_t \). These are interpreted as reflection coefficients. Therefore a reflection coefficient series is generated from the data without having to know what the form of the source wavelet is, the success of which is contingent upon the validity of the assumptions of minimum delay and a random reflector series. Modern data processing routinely employs adaptive multi-channel versions of the Burg algorithm based on techniques such as that described by Widrow et.al. (1967).
The adaptive filter is one which is recomputed or updated continuously as each new data point is encountered.

Predictive deconvolution does not perform satisfactorily when the assumptions of minimum phase and a white reflection coefficient series break down. Commonly the source function is not minimum phase and in cases where the geologic structure is cyclic in nature, predictive deconvolution will not resolve all reflection events.

The other widely used method of deconvolution is wavelet processing. This approach incorporates both deterministic and statistical aspects. The source waveform is estimated first by field measurements or some robust technique such as structural deconvolution and followed by a Wiener filter which will operate on the data to give the earth response or reflection coefficient series,

\[ r(t) = f(t) * x(t). \]

This technique is particularly well suited to multi-channel deconvolution because ensemble averages can be taken which are necessary for the derivation of the optimal filter. Following Berkhout (1977), the method is predicated upon minimizing the sum of squares of the errors between the input time series \( x_i(t) \), and a desired time series \( d_i(t) \). If a large number, \( N \), of traces are available, one filter is computed which minimizes the total error of all of them,

\[
E = \frac{1}{N} \sum_{i=1}^{N} \left[ \sum_{t} (d_i(t) - r_i(t))^2 \right] = \frac{1}{N} \sum_{i=1}^{N} \left[ \sum_{t} (d_i(t) - \sum_{m} f(m)x_i(t-m))^2 \right].
\] (1.7)
Differentiation of (1.7) with respect to the unknown filter coefficients give the normal equations,

$$\sum \frac{f(m)}{m} R_{x,x}(n-m) = R_{x,d}(n)$$  \hspace{1cm} (1.8)

where $R_{x,x}(\tau) = \frac{1}{N} \sum_{i=1}^{N} x_i x_i(\tau)$ is the ensemble average of the autocorrelation of the input trace, and $R_{x,d}(\tau) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T} \sum_{t=1}^{T} x_i(t-\tau) d_i(t)$ is the ensemble average of the cross correlation of the input trace with the desired output trace. If the desired output trace is the reflection coefficient series $r(t)$, the normal equations reduce to the form,

$$\sum \frac{f(m)}{m} R_{x,x}(n-m) = w(-n),$$  \hspace{1cm} (1.9)

under the assumptions that the reflection coefficient series is a white noise series and the additive noise is uncorrelated with the reflector series $r(t)$. The right hand term, $w(-n)$, is the time reverse of the measured or assumed waveform. The reflection coefficient series generated by the Burg algorithm is often compared with the result of the application of the Wiener-Hopf optimum least squares inverse filter to check for consistency. The filter obtained from the solution of the simultaneous equations in (1.9) is called two-sided because the filter and the wavelet may have a value for $t < 0$. 
If \( w(t) = 0 \) and \( f(t) = 0 \) for \( t < 0 \), the filter is called one-sided and the normal equations become,

\[
\sum_{m=0}^{M} f(m) R_{x,x}(k-m) = w(o) \delta_{o,k}, \text{ for } k = 0, 1, \ldots, M.
\]

In this case the form of the wavelet need not be known so this filter is widely used in the seismic exploration industry.

Berkhout (1977) points out, however, that even with minimum phase wavelets, one-sided least squares inverse filtering is not very suitable to produce high resolution results due to the presence of noise. That is, it is difficult to determine the polarity of arrivals and their arrival times are seriously in error. Two-sided least squares inverse filtering is however greatly superior to one-sided least squares inverse filtering if preceded by an adequate wavelet estimation procedure. If the wavelet is chosen to be a symmetrical zero-phase wavelet with some desired bandwidth the process is called wavelet deconvolution and it closely resembles two-sided least squares inverse filtering provided the noise on the input data is white.

As with predictive deconvolution, wavelet deconvolution and least squares inverse filtering rely on the assumption that the reflection coefficient series can be represented by a white noise series. Similarly, the technique is less effective for those geologic sections that have cyclic stratification with large acoustic impedances at each interface and nearly equal thicknesses between beds.
The matched filter is a deterministic method that relies on knowing the shape of the input signal or source function. It seeks to maximize the signal to noise ratio $\mu$ (Treitel and Robinson, 1969):

$$\mu = \frac{\text{(value of filtered signal at time } t_0)^2}{\text{Average power of filtered noise at time, } t_0}$$

The filter, $a$, is computed from the known wavelet $w$ by maximizing the central output value of the convolution, $c = a \ast w$ subject to the unit energy constraint on the filter coefficients,

$$\sum_{i=1}^{n} \hat{a}_i^2 = 1.$$

Another filter in the same class is the output energy filter. It seeks to maximize the sum of squares of all output values, $\sum_{i=1}^{n} c_i^2$, subject to the same unit energy constraint on the filter coefficients. In this case only the autocorrelation or the power density spectrum of the source wavelet need be known. This is particularly applicable in cases where dispersion may be present but the amplitude spectrum of the propagating wavelet is not changing appreciably. Both of these filters will increase the signal to noise ratio in a seismic trace and will tend to compress the energy of the wavelet into a single spike. Although the performance of these filters is good in a noisy environment they have disadvantages, namely that the time of the detected event will not be its true onset time and the other is difficulty in identification of overlapping reflections. Use of these filters to resolve thin beds in a sedimentary environment is therefore not advisable.
A method developed recently by Wiggins et al. (1977) called minimum entropy deconvolution is an entirely different approach from the previously described techniques for extraction of the reflection coefficient series from the seismogram. Like predictive deconvolution and wavelet processing it seeks to remove the effects of the source wavelet, but where predictive deconvolution whitens the data (maximizes entropy), minimum entropy deconvolution finds the smallest number of large spikes that is consistent with the data (minimizes entropy or maximizes order). It makes no assumptions about the source wavelet shape or phase characteristics and whereas predictive deconvolution will sometimes amplify noise, minimum entropy deconvolution maximizes the spikiness of the output trace and suppresses frequency bands with low signal to noise. One disadvantage of the technique is that the polarity of a reflection event is oftentimes ambiguous.

A disadvantage shared by all of the aforementioned techniques lies in the fact that the purpose is only to resolve reflection events. After such processing it is uncertain whether any additional information such as attenuation can be gained from the seismogram.

Wavelet or Source Signature Estimation

The source signature or source wavelet can be estimated in a number of ways. A robust wavelet estimation procedure called structural deconvolution (Stone, 1977a) relies on the knowledge of the reflection coefficient series by means of well logs. The basic equation (1.2) is used to compute the wavelet,
\[ w(t) = r^{-1}(t) * x(t), \]

where \( x(t) \) is a reflection seismogram taken near the well.

Homomorphic filtering estimates the source function from the seismogram by utilizing the fact that the inverse Z-transform (Fourier transform) of the log of the Z-transform of a seismic trace, called the complex cepstrum, will have components due to the source function at smaller data point values than the components due to echoes (Ulrych, 1971; Stoffa et al., 1974). By low pass filtering the cepstrum and then inverse transforming back to time the resulting trace will be an estimate of the source function. No prior assumption about the nature of the seismic wavelet or the impulse response of the transmission path need be made. Another method of homomorphic filtering called log spectral averaging (Otis and Smith, 1977) averages the log spectra of several reflection records. The source function is assumed to be stationary whereas the earth's response for the various traces is chosen to be spatially nonstationary. Thus the log spectrum of the source function will be enhanced and that of the earth's response will tend to average out. Inverse transforming back to time yields an estimate of the source function.

Direction measurement of the source signature is another commonly used technique for wavelet estimation and is the method used in this thesis. Four types of measurement are generally done in marine seismic work. One is the near field recording of the signature by sensors mounted on or close to the airgun at distances on the order of a meter. The second is to drop a hydrophone below the airgun in deep water
and do a static measurement of the source signature assuming it to be a satisfactory representation of the true wavelet under operating conditions. The third is to measure the direct arrival by a towed streamer or sensor at near surface horizontal distances of several to tens of meters and the fourth is record a time gated bottom reflected signal in areas where there is an evident lack of subsurface structure. The last two techniques were tested on the data in this study.

After wavelet estimation by any of the preceding methods, deconvolution to remove the ringing effects of the oscillating bubble pulse can be accomplished by time domain Wiener least squares inverse filtering, matched filtering or by division in the Fourier transform domain. In both cases it is usually necessary to apply bandpass filtering as a post operation. Deconvolution by division in the Fourier transform domain was the technique applied in this research because it has been shown to give similar results as Wiener inverse filtering (Otis and Smith, 1977) for the same estimated wavelet and it has the advantage of not altering the original data in such a way as to complicate further analysis of attenuation in the subsurface strata. In addition no assumptions about the statistics of the reflection coefficient series need be made. The main disadvantages of deconvolution by division in the Z-transform or Fourier transform domain are the possible instabilities caused by the existence of poles in the inverse filter (Treitel and Robinson, 1969) and its sensitivity to the presence of noise.
Deconvolution attempts to extract the crustal impulse response or reflection coefficient series from the data, in order to increase the possibility of resolving thin layers and intrabed multiples. Resolution of primary reflection events by deconvolution permits increased accuracy in the computation of a velocity log or a velocity versus depth model. From velocities alone some inferences can be made about the general class of material present at depth based upon experimental work done by many authors (Hamilton et al., 1970a; Hamilton, 1970b; Hamilton, 1971b; Morgan, 1969; Shumway, 1960a and 1960b; and others). Seismic velocities have been related to various elastic constants such as bulk modulus and rigidity (Hamilton, 1971a) as well as to sediment properties such as porosity and density (Morgan, 1969; Nafe and Drake, 1957; Hamilton, 1970b). Variations in the speed of sound in natural unconsolidated sediments are caused by, in the order of importance, porosity, rigidity, pressure, temperature and the compressibility of the grain aggregate (Shumway, 1960a). Keller (1974) gives a summary of sediment geotechnical properties, their interrelationships and relationships to depth of burial. Another parameter which is an indicator of material type but is independent of the measurement of velocity is attenuation. Hamilton (1972) measured compressional wave attenuation in situ in marine sediments and plotted it versus porosity and mean grain diameter. Shumway (1960a and 1960b) made laboratory measurements of unconsolidated marine sediment samples
and found that an absorption maximum occurs for sediments of intermediate porosity (0.45-0.6) and intermediate grain size (0.031-0.25mm). McCann and McCann (1969) also discussed this effect and measured attenuation in North Atlantic sediment cores. Hampton (1967) made laboratory measurements of velocity and attenuation in sediments over a frequency range of 400 kHz to 600 kHz. Tyce (1976) measured attenuation in sediments in situ at 4 kHz. Laboratory measurements of body wave attenuation in rocks have been done by many investigators (Toksoz et al., 1977; Birch and Bancroft, 1938; Peselnick and Zietz, 1959; Born, 1941; Wyllie, et al., 1962; and others). Most measurements on sediments and rocks have been done at frequencies in the kilohertz to the megahertz range. A study at high frequencies by Johnston and Toksoz (1977) concludes that: 1) attenuation is linear with frequency for P waves in both dry and water saturated rocks, 2) attenuation in water saturated rocks is greater than in dry rocks and 3) attenuation decreases with increasing differential pressure. The high frequency studies for sediments suggest that geotechnical properties such as grain size, porosity and percent sand-clay mixture may be predicted from acoustic attenuation measurements (Hamilton, 1972 and 1974; Buchan, 1972). The gap in attenuation measurements at high and low frequencies is due primarily to the difference in the measurement techniques required. Test samples must have dimensions on the order of or larger than the acoustic wavelength. High frequency measurements can therefore be conveniently accomplished in the laboratory, whereas measurements at seismic frequencies require excessively large volumes of material. Only a
few measurements have been made at seismic frequencies in rocks (for example, Ricker, 1953; Bruckshaw and Mahanta, 1954; McDonal et al., 1958). Attewell and Ramana (1966) give a summary of reported values of attenuation in rocks over the frequency range from one Hz to $10^8$ Hz. Similarly, very few attenuation studies have been done at seismic frequencies in sediments (for example, Cole, 1965; Tullos and Reid, 1969; Li and Smith, 1969; Anderson and Blackman, 1971).

The important question to address is whether or not there is enough variability and predictability of attenuation at seismic frequencies as a function of geotechnical rock properties to aid in the identification of material types. At present there is not enough data available to make a judgment. Jankowsky (1975) applied a method developed by Koerner et al. (1975) to extract energy decay functions from seismic reflection data in offshore areas. He suggests on the basis of his results that it seems possible to identify lithologic sequences and their pore space and saturation through their specific "absorption functions".

The main reason for the difficulty of attenuation measurements from seismic reflection data is that there are two primary causes for attenuation. The first is attenuation due to the anelastic properties of the material itself and the second is the category of various geometric effects including scattering and losses due to the reflection and transmission coefficients at the contacts of dissimilar geologic strata.
Attenuation Mechanisms

Theoretical and experimental studies of attenuation due to the anelastic properties of fluid-filled porous media have verified that there are two basic mechanisms operating: 1) viscous losses and 2) solid friction or frame losses (McCann and McCann, 1969; Born, 1941; Wyllie et al., 1962; Stoll, 1977; and others). The viscous losses due to fluid flow have been described theoretically by Biot (1956a and 1956b). Other viscous mechanisms include relaxation phenomena, squirting effects, gas pocket squeezing, etc. Frame losses include grain boundary and crack surface frictional dissipation and inelastic relaxation phenomena. All of these effects will be dependent upon frequency, fluid saturation and pressure as previously mentioned. The conclusion that attenuation is proportional to the first power of frequency over a broad frequency range (10^{-2} to 10^7 Hz) has been verified by many investigators (Birch and Bancroft, 1938; Born, 1941; McDonal et al, 1958; Attewell and Ramana, 1966; and others). This conclusion has also been verified by Hamilton (1972) for 3.5 kHz to 100 kHz in unconsolidated marine sediments. Stoll (1977), in a theoretical study, concludes that in finer sediments where fluid mobility is low, the losses in the skeletal frame are significantly larger than the fluid losses at all frequencies of interest and the attenuation is proportional to the first power of frequency. In coarser sediments, however, fluid losses become larger than frame losses and the first power dependency is no longer valid. Extrapolation of attenuation coefficients from high frequency data to seismic
frequencies for coarse sediments is therefore inadvisable. For clays and muds it appears to be an acceptable procedure.

The second major class of attenuation mechanisms is acoustic scattering. These include Rayleigh scattering from inhomogeneities in the bulk material, diffuse reflection from irregular interfaces, diffraction, reflection and transmission losses and apparent attenuation due to inter and intrabed multiple reflections. The term intrabed will hereafter be used exclusively throughout to designate reflections between and within beds. The latter effect has been treated recently by O'Doherty and Anstey (1971), Schoenberger and Levin (1974 and 1977) and Spencer (1977). O'Doherty and Anstey (1971) suggested that for many geologic sections part or all of the attenuation measured from reflection seismograms can be attributed to intrabed multiples rather than to absorption in the rock material itself. Schoenberger and Levin (1974 and 1977) computed synthetic reflection seismograms from density and velocity logs of wells by placing an isolated reflector into the uniform section below the wells. The synthetic seismogram program divides the subsurface into layers of 2 m sec two way travel time and computes 50 orders of intrabed multiples for a spike input. Amplitude spectra were computed for time gates or windows of the synthetic seismogram at successively increasing record times. Attenuation constants computed from surface seismic reflection data near four of the 31 wells investigated gave values 1.6 to 10 times the corresponding intrabed constants computed from the synthetic seismograms. They conclude that attenuation due
to intrabed multiples accounts for an appreciable fraction of the total attenuation but by no means all of it. Spencer (1977) computed the expected seismic wave attenuation in non-resolvable cyclic stratification and concluded that the effect of intrabed multiples should not be significant unless the reflection coefficients are unusually large, i.e. greater than 0.1.

The conclusions of Schoenberger and Levin and Spencer suggest that if attenuation can be measured for the intervals between resolved acoustic interfaces in sedimentary environments, it can be expected to be due largely to material absorption. That is, if all primary reflection events are accounted for, the effect of non-resolvable stratification should be a small fraction of the total observed attenuation. Thus it would seem possible to estimate such sediment geotechnical properties as mean grain size or porosity. Under certain circumstances cyclic stratification with large velocity contrasts and large reflection coefficients can occur. Domenico (1974) demonstrated that gas charged sands will exhibit significantly larger reflection coefficients than the same water saturated sands. High velocity stringers of coarse sands or gravels interbedded with fine sediments such as may be found in a turbidite environment may have a measurable effect if the thickness of the zone is large enough. Spencer (1977) shows that such cyclic systems exhibit frequency selective attenuation associated with short path multiples. Propagation through them results in low pass filtering or spectral shaping with the attenuation coefficient having a frequency to the 2.5 power dependence over the low frequency band. It is interesting to note the comparison with
the linear dependence associated with absorption and the second power
dependence associated with Rayleigh Scattering. The bandwidth is
dependent on the average thickness of the scatters -- as the thickness
decreases, the bandwidth increases. If such frequency dependence
is observed it is conceivable that the effect of stratification on
the attenuation can be exploited to measure the average non-resolvable
layer thickness or to infer the presence of gas by comparing lateral
changes in the spectral shaping of reflections from above and below
the zone.

The Measurement of Attenuation from Seismic Reflection Data

Published information is scarce on in situ measurements of
attenuation from seismic reflection data at frequencies from zero to
500 Hz. Li and Smith (1969) made in situ measurements of attenuation
in a sedimentary wedge in the Irish Sea. The difference in pressure,
or amplitude of the reflection from sea bottom and the reflection
from the base of the layer expressed in decibels at two different
thicknesses and at a given frequency defines the reflection coefficient.
From the data they estimated the mean grain size in the sediment
column. A similar method was employed by Anderson and Blackman (1970)
to estimate acoustic attenuation in a sediment wedge on the flanks
of Bermuda. They plotted a difference of the subbottom to bottom
amplitudes in decibels versus range (two times thickness) and computed
$\alpha$ from the slope of a linear least square fit to the data for a
frequency range from 200 to 400 Hz. Plotting $\log \alpha$ versus $\log f$
gave a value of $n = 3$ in the equation, $\alpha = kf^n$. They attribute this
large deviation from the expected linear dependence of attenuation on frequency to the presence of thin layering in the sedimentary wedge. These techniques are of limited usefulness in that they require the presence of a thick sedimentary wedge and are inaccurate for the purpose of measuring material absorption unless the wedge is vertically homogeneous. This was apparently the case for Tyce (1976) who used the technique for a wedge of sedimentation in the San Diego Trough and on the Carnegie Ridge. His values for the San Diego Trough are consistent with those measured directly by Hamilton (1972) for clayey silt of approximately 75% porosity. Low attenuation in the Carnegie Ridge sediments suggested that lithification may be occurring in them.

Tullos and Reid (1969) measured acoustic attenuation in the frequency range 50 to 400 Hz of normally or flat layered Gulf coast sediments. Their technique differed in that strings of geophones of equal separation were placed in wells near the shotpoints. The seismic signals were gated to separate the outgoing pulse or direct wave from the reflected or refracted energy and scaled with a factor that included calibration constants, estimated reflection coefficients and the effect of geometric spreading. The Fourier amplitude spectra of the gated and scaled traces were smoothed by a 25 point running average and the spectra and spectral ratios of successive geophones were averaged for a number of geophone stations. Their data confirmed the linear dependence of attenuation with frequency and suggest that if spectral ratios are taken over small intervals and smoothed and averaged, the apparent attenuation due to intrabed multiples becomes negligible. Placing geophones in wells is a very expensive and time
consuming procedure, however, so there is a need for measuring attenuation routinely from seismic reflection data recorded by surface detectors. The standard technique in the literature for doing this is by time gating the reflection seismogram (multiplication by a time domain window) (Schoenberger and Levin, 1974). This technique was also employed by Millahn and Jurczyk (1977) except that spectra were estimated using maximum entropy spectral analysis in order to obtain stable spectra from short time gates (Ulrych et al., 1973). Short time-gates would supposedly increase the reliability of spectral information related to a single arrival by excluding more arrivals caused by multiple reflections. Spectral ratios are taken from the spectra of selected time windows which contain primary reflections. Field data gave very uncertain results with large error. A disadvantage of the maximum entropy spectral analysis technique in this application is its tendency to shift the location of spectral peaks (Chen and Stegen, 1974). Spectral ratios can therefore be in considerable error if narrow spectral peaks are present. The other major disadvantage of the time gating technique is the inclusion of unknown extraneous spectral information in the window. Spectral smoothing and averaging may help but the unknown effect of multiples may still remain as they are deterministic and quite repeatable from shot to shot for the same shotpoint-geophone location. Clowes and Kanasewich (1970) use a different approach. They compared the spectra of windows centered about chosen reflection events of near vertical incidence deep reflection seismograms with the spectra of the same windows for synthetic seismograms which include frequency and depth-dependent attenuation. When the spectral shapes
are nearly coincident, the input Q structure of the synthetic seismograms are accepted. Thus inversion is accomplished laboriously by trial and error techniques.

Method of Approach

The purpose of this research is to investigate an approach to the analysis of seismic reflection data that is completely deterministic in nature, making as few assumptions as possible about the statistical properties of the data, at least during the initial deconvolution stages of data processing. Wide angle reflection and near vertical incidence reflection data are processed by source signature deconvolution to increase the resolution and accuracy of intervals and their velocities. A frequency domain least squares regression routine is proposed to process the data for attenuation information.

The data was collected from Oregon State University vessels, R/V YAQUINA, in 1975 and the R/V WECOMA, in 1976. The configuration for routine collection of single channel vertical seismic reflection data and wide angle reflection data is shown in Figure 1. The wide angle reflection data is analyzed for interval velocities using the vertical profiler data for computing dips of major acoustic horizons. Both types of data were digitized from analog magnetic tape and processed for the removal of the effects of the bubble pulse by frequency domain source signature deconvolution. A Data General NOVA 1200 computer processed the digitized data and the OSU CDC 3300 computer solved for interval velocities. The source signatures were measured by three methods: 1) stacking the direct wave received
by the short streamer shown in Figure 1, 2) stacking the direct wave as received by the sonobuoy near the origin time of a wide angle reflection run, and 3) stacking the bottom reflected arrival as received by the sonobuoy in a sedimentary environment with a relatively homogeneous surface sediment layer. The first method was used to deconvolve vertical profiler data and the last two were used with wide angle reflection data. The vertical profiler data is taken at near normal angles of incidence and is therefore suitable for the analysis of attenuation. Frequency domain source signature deconvolution increases the resolution of all events including multiples in the seismogram. Subjective selection of primary events in the deconvolved wide angle reflection data allows the computation of interval velocities in the sedimentary column. Accurate timing of primary reflection
events in the deconvolved vertical profiler data allows the computation of frequency dependent attenuation by a linear least squares regression model. Frequency dependent spectral ratios are computed for reflections from the interfaces resolved by deconvolution. The OSU CDC Cyber 73 computer was utilized for the frequency dependent attenuation analysis. The method was tested on field data from the following sedimentary environments off the coast of Oregon and Northern California which were chosen for their variability: a continental shelf basin, an abyssal plain environment, the base of the continental slope, and two locations on the Astoria sea fan, one near the Cascadia sea channel and one near DSDP site 174. The theoretical development of the deconvolution method and the least squares regression model will present some of the advantages as well as problems with the proposed technique for the measurement of attenuation over previous methods described above and in the literature.
CHAPTER 2  
METHOD AND DERIVATIONS

The Reflection Seismogram

The mathematical representation of the reflection seismogram is the linear system previously mentioned (eq. 1.2):

\[ x_t = s_t * r_t + n_t = \sum_{\tau=1}^{N-t} s_t r_{t-\tau} + n_t \]  

(2.1)

for a discrete time series of finite length, where \( * \) denotes convolution, \( s_t \) is the discreet form of the source function or source signature, \( r_t \) is the discreet form of a reflection coefficient series and \( n_t \) is a random noise series. According to Bath (1974, p. 37) the Fourier transform of a continuous function \( f(t) \) is:

\[ F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} \, dt \]  

(2.2)

and the inverse transform is:

\[ f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} \, d\omega \]  

(2.3)

where \( i = \sqrt{-1} \). The Fourier transform of \( x_t \) is simplified by using the following theorems (Bath, 1974, pp. 79 and 44, respectively):

1) The Fourier convolution theorem

The Fourier transform of the convolution of two functions, \( a(t) \) and \( b(t) \) is equal to the product of the Fourier transforms of each function:

\[ FT\{a*b\} = A(\omega) B(\omega) \]  

(2.4)

2) The time shifting theorem
If the function \( a(t) \) is shifted in time by \( \pm \tau \) the Fourier transform becomes:

\[
\text{FT} \{ a(t \pm \tau) \} = e^{\pm i \omega \tau} A(\omega).
\] (2.5)

These theorems hold for the discreet Fourier transform also which are given by (Brigham, 1974, p. 98):

\[
F(n/N\Delta t) = \sum_{k=0}^{N-1} f(k\Delta t) e^{-i2\pi nk/N},
\] (2.6)

and the inverse transform by:

\[
f(k\Delta t) = (1/N) \sum_{n=0}^{N-1} F(n/N\Delta t) e^{i2\pi nk/N},
\] (2.7)

where \( k \) is the data point in time, \( n \) is the data point in frequency, \( N \) is the number of data points in the finite length series and \( \Delta t \) is the sample or digitizing interval.

The reflection coefficient series for a digitized time sequence is derived using the Kronecker delta function:

\[
r_t = \sum_j a_{j,n} \delta_{jk}, \text{ where } \delta_{jk} = \begin{cases} 1 & j=k \ 0 & j \neq k \end{cases}
\] (2.8)

The spikes are shifted by the time delays \( \tau_j = j\Delta t \) and \( a_{j,n} \) represents a complex time delay dependent and frequency dependent function which includes the reflection and transmission coefficients as well as the effects of attenuation, dispersion, and geometric spreading.

The discreet Fourier transform of the reflection seismogram is then, using 2.1, 2.4, 2.6 and 2.8:
\[ X(n/N\Delta t) = X_n = \left[ \sum_{k=0}^{N-1} s(k\Delta t)e^{-i2\pi nk/N} \right] \left[ \sum_{j=0}^{N-1} \left( \sum_{j=0}^{N-1} a_{j,n} \delta_{jk} \right)e^{-i2\pi nk/N} \right] \]

\[ = S_n \sum_j a_{j,n} \left[ \sum_{k=0}^{N-1} \delta_{jk} e^{-i2\pi nk/N} \right] \]

\[ = S_n \sum_j a_{j,n} e^{-i2\pi nj/N} \quad (2.9) \]

which is equivalent to:

\[ X(\omega) = S(\omega) \sum_j a_j(\omega,\tau_j)e^{-i\omega \tau_j} \quad 2.10 \]

for continuous functions.

The equivalence between the two representations can be seen by realizing that the highest possible frequency in a digitized time function is called the sampling frequency, \( f_s \):

\[ f_s = 1/\Delta t \quad \text{cycles/sec if } \Delta t \text{ is in seconds,} \]

or in units of radians/sec, \( \omega_s = 2\pi/\Delta t \). The sample interval in the frequency domain is \( \omega_s/N = 2\pi/N\Delta t \). The discreet frequency values are then \( \omega = 2\pi n/N\Delta t \) and the discreet time delays are \( \tau_j = j\Delta t \), therefore, \( \omega \tau = 2\pi nj/N \). The Nyquist or folding frequency is \( f_N = f_s/2 = 1/2\Delta t \) or \( \omega_N = \pi/\Delta t \) radians/sec. The spectrum of a time series is unique only over the frequency range, zero to Nyquist.

The fast Fourier transform (Cooley and Tukey, 1965) was used in the data analysis. See Brigham (1974) for a complete description. It is required that the number of data points in the time series be equal to a power of two. The usual procedure is to "pad" with zeroes on either end or just at the end of the sampled time series.
to satisfy this requirement. The Nyquist frequency is located at the \((N/2+1)\) data point and the last data point in the frequency domain is one less than the sampling frequency.

**Filters in the Linear System**

Before discussing the propagation components or factors of the coefficients \(a_{1,n}\) it will be useful to consider additional filters in the linear system. An acoustic wave propagating in a homogeneous half space reflecting from a single boundary at depth and recorded at the surface can be presented as a product of the cascaded filters similar to that given by Bath (1974, pp. 272-274):

\[
X(\theta, \omega, r) = S(\omega) B(\theta, \omega) I(\omega) G_{h}(\theta, \omega) G(r) L(\omega, r),
\]

where

- \(S(\omega)\) = true source signature spectrum
- \(B(\theta, \omega)\) = source-receiver space or directivity function
- \(I(\omega)\) = instrument transfer function
- \(G_{h}(\theta, \omega)\) = surface ghost transfer function
- \(G(r)\) = geometrical spreading
- \(L(\omega, r)\) = loss function which includes attenuation and dispersion.

Let \(G(r)L(\omega, r) = a(\omega, \tau_j)\) where \(\tau_j\) is now a function of \(r\) through:

\[
\tau_j = \sum \Delta r_i / v_i,
\]

where \(m_i\) is the number of transits through interval \(i\), \(\Delta r_i\) is the thickness and \(v_i\) is the interval velocity. Then,

\[
X(\theta, \omega, r) = S(\omega) B(\theta, \omega) I(\omega) G(\theta, \omega) a(\omega, \tau_j)
\]

(2.12)
and for a layered half space, equation (2.10) becomes, excluding noise:

\[ X(\theta, \omega, r) = S(\omega)B(\theta, \omega)I(\omega)G_{h}(\theta, \omega) \sum a(\omega, \tau) e^{-i\omega \tau} j \]  

where \( \theta \) is the angle of incidence of the upcoming ray.

The various filters will now be treated separately.

**Source-Receiver Directivity Function**

The streamer and the sonobuoy hydrophone string may be considered a line array of equally spaced receiving elements with the ray impinging at angle \( \theta \) relative to the array axis as shown in Figure 2. The amplitude and phase spectrum of the directivity function is derived according to the formulation given by Urick (1967, pp. 41-42).

The time delay between the arrivals at each element is \( \Delta \tau = \frac{d \cos \theta}{v_w} \) where \( v_w \) is the velocity of sound in water. The filter is the sum of the delayed responses for each element.

\[ B(\theta, \omega) = 1 + e^{-i\omega \Delta \tau} + e^{-2i\omega \Delta \tau} + e^{-3i\omega \Delta \tau} + \ldots \]  

where \( z = e^{-i\omega \Delta \tau} \) and \( N \) is the number of elements. Multiply (2.14) by \( z \) and subtract (2.14) from the result,

\[ zB - B = z^N - 1 \]

\[ B(\theta, \omega) = \frac{z^N - 1}{z - 1} = \frac{e^{-iN\omega \Delta \tau} - 1}{e^{-i\omega \Delta \tau} - 1}. \]
The modulus of this filter is:

\[ |B(\theta, \omega)| = \frac{\sin(N\omega \Delta t/2)}{\sin(\omega \Delta t/2)}, \tag{2.16} \]

and the phase is:

\[ \varphi = \tan^{-1} \left( \frac{\sin N\omega \Delta t - \sin \omega \Delta t - \sin(N-1)\omega \Delta t}{\cos(N-1)\omega \Delta t - \cos \omega \Delta t - \cos N\omega \Delta t + 1} \right). \tag{2.17} \]

The geometrical configuration for the data collection is shown in Figure 1. The streamer is 100 feet long with hydrophones spaced every foot. It has two channels with alternate geophones on each channel. The sonobuoy string is composed of four hydrophones separated by 20 inches with the midpoint at about 19.3 meters depth.

*Figure 2. Streamer directivity function geometry.*
The normalized amplitude and phase spectrum of the directivity filter function for these array configurations at $\theta = 0^\circ$ is shown in Figure 3. The amplitude is one and the phase is zero for all frequencies at $\theta = 90^\circ$. The angle difference between the signal and the source function for both sonobuoy and streamer is about $90^\circ$ which corresponds to the plot for $\theta = 0^\circ$. The implications of this in the data analysis will be discussed in the results.

Instrument Transfer Function

The instrument transfer function is assumed to cancel or to be equalized upon division by the Fourier transform of the measured source function if the source signature for the deconvolution

![Figure 3. Amplitude and phase spectrum of streamer and sonobuoy directivity functions.](image)
of vertical profiler data is detected by the streamer and the signature for the wide angle reflection data is detected by the sonobuoy.

**Surface Ghost Transfer Function**

The surface interference effect, also called the Lloyd's mirror effect, is the result of the interference of the direct wave with the surface reflected wave which has been phase shifted by \( \pi \) radians. The effect of ghosts in a seismogram is commonly attenuated with single loop feedback filters (see for example, Lindsey, 1960). The inverse ghost filters in this work are applied in the Fourier transform domain (Appendix 4), and the transfer functions of those applicable to the wide angle reflection (WAR) and vertical profiler source-receiver geometries are derived below. The received signal is the sum:

\[
s'(t) = s(t) + Rs(t-\tau).
\]

The Fourier transform is, using equation (2.5):

\[
S'(\theta,\omega) = S(\omega) [1 - e^{-i\omega T}] = S(\omega)G_h(\theta,\omega)
\]

where \( \tau \) is the time delay of the ghost relative to the direct wave and is related to the geometry of the source-receiver configuration and the reflection coefficient of the surface-air interface \( R = -1 \). A further sophistication of this filter is to replace \( R \) by a Rayleigh scattering coefficient \( \mu(\omega) \) which is dependent upon
frequency and the sea state. According to Urick (1967):

$$\mu(\omega) = \exp(-R')$$

(2.18)

where the Rayleigh parameter, $R' = kH\sin\theta, k = 2\pi/\lambda, H = \text{rms crest to trough wave height}$ and $\theta = \text{grazing angle (with respect to horizontal) of ray at the surface reflection point}$. The amplitude spectrum of the filter is shown in figure 18. The inverse filter must be truncated at zero points but problems may still occur as is shown in the results with model data. Inclusion of the Rayleigh scattering criterion smooths the spectrum and eliminates the zeros at high frequencies. This is included in the deghost filter algorithm (Appendix 10).

Three different inverse ghost filters are used in the data processing. The first is for deconvolution of vertical profiler data. The source signature received by the streamer has the transfer function (excluding all but the ghost filter):

$$S'(\omega) = S(\omega)(1 - e^{-i\omega\tau_1}).$$

The downgoing ray is:

$$S''(\omega) = S(\omega)(1 - e^{-i\omega\tau_2}),$$

and the final received signal is:

$$Y(\omega) = S''(\omega) \left[ \sum_{\tau} a(\omega, \tau)e^{-i\omega\tau} \right] (1 - e^{-i\omega\tau_3}).$$
The deconvolved trace will be:

\[
\frac{Y(\omega)}{S'(\omega)} = \frac{S(\omega)(1 - e^{-i\omega \tau_2})[\sum a(\omega, \tau)e^{-i\omega \tau}]}{S(\omega)(1 - e^{-i\omega \tau_1})} (1 - e^{-i\omega \tau_3})
\]

The inverse ghost filter needed to extract the crustal impulse response is therefore:

\[
Gh^{-1}_{STR}(\theta, \omega) = \frac{(1 - \mu e^{-i\omega \tau_1})}{(1 - \mu e^{-i\omega \tau_2})(1 - \mu e^{-i\omega \tau_3})}, \quad (2.19)
\]

so that,

\[
\frac{Y(\omega)}{S'(\omega)} \cdot Gh^{-1}_{STR}(\theta, \omega) = \sum a(\omega, \tau)e^{-i\omega \tau}
\]

where,

- \(\mu\) = reflection coefficient which includes the Rayleigh scattering criterion,
- \(\tau_1\) = time delay between direct and reflected ray for source signature,
- \(\tau_2\) = ghost time delay for downgoing ray at airgun
- \(\tau_3\) = ghost time delay for upcoming ray at streamer.

This operation is performed in subroutine VGHHOST.

The second deghost filter is intended for deconvolution of WAR data. The source signature is measured by stacking or averaging a number of direct wave arrivals to the sonobuoy near the beginning of a WAR profile. This is necessary because the surface interference effect attenuates low frequencies more as \(\tau\) decreases. After a
few minutes of ship travel time, some low frequencies are in the noise and no longer recoverable. Assume that the true source function is invariant. As the ship moves away from the sonobuoy, the delay times decrease, resulting in a different ghost filter for each shot. If the stacked source function can be represented as:

\[
s_{SON}(t) = \frac{1}{N} \sum_{j=1}^{N} [s(t) * gh_j(t) + \epsilon_j(t)].
\]

the noise series \( \epsilon_j(t) \) will tend to zero in the stack if it is white and stationary. The Fourier transform is:

\[
S_{SON}(\omega) = S(\omega) \frac{1}{N} \sum_j \text{Gh}_j (\theta, \omega)
\]

The inverse filter for the sonobuoy source signature is:

\[
\text{Gh}^{-1}(\theta, \omega) = [1 - \frac{1}{N} \sum_j e^{-i\omega \tau_j/N}]
\]

This operation is done in program XSNSFW (Appendix 10). Program STKSS computes the ghost time delays necessary for the computation. The deconvolved sonobuoy trace is then:

\[
\frac{Y(\omega)}{S(\omega)} = \frac{S(\omega) (1 - e^{-i\omega \tau_2})(1 - e^{-i\omega \tau_3}) \sum_j a(\omega, \tau_j) e^{-i\omega \tau_j}}{S(\omega)}
\]

The inverse ghost filter for extracting the crustal impulse response from WAR data is:

\[
\text{Gh}_{SON}^{-1} = \frac{1}{(1 - e^{-i\omega \tau_2})(1 - e^{-i\omega \tau_3})}.
\]
This operation is performed in subroutine WGHOST.

In order to compute the source ghost delay time the depth of the air gun must be known. The depth can be measured knowing the bubble pulse oscillation period from a form of the Rayleigh-Willis formula (Kramer, 1968, p. 17):

\[
H = \frac{(PV)^{4}}{27.7(T)^{1.2}} - 33
\]  

(2.22)

where,  
H = depth in feet,

P = air gun chamber pressure in pounds per square inch,

V = air gun chamber volume in in³

T = bubble pulse oscillation period in seconds.

The bubble pulse period can best be measured by inspection of the power spectrum of the source signature as shown in Figures 27 and 30. The period is found by taking the inverse of the frequency at which the first maximum occurs. This value agrees with estimates of the depth from the geometry of the airgun-ship configuration and with the improved appearance of the deconvolved traces when the correct ghost delay times are entered as shown in the results with field data.

The sonobuoy hydrophone depth was assumed equal to the depth set during deployment and the streamer depth was estimated from optimum ghost delay times.

The formulas for the computation of variable and fixed water surface ghost delay times according to the WAR and vertical profiler source-receiver geometries are derived in Appendix 5.
Geometric Spreading

The receivers (hydrophones) are sensitive to pressure. The sound intensity level is (Officer, 1958, p. 29):

\[
I = \frac{\text{Power}}{\text{Area}} = \frac{p^2}{\rho v},
\]

where \( p \) = pressure, \( \rho \) = density of medium and \( v \) = acoustic velocity of medium. The sound intensity level will decrease as the square of the distance. The pressure or amplitude level will therefore be inversely proportional to the distance. The ratio of the amplitude levels at two distances will then be:

\[
\frac{A_1}{A_2} = \frac{p_1}{p_2} = \sqrt{\frac{I_1}{I_2}} = \sqrt{\frac{\text{power}}{4\pi r_1^2}} = \frac{r_2}{r_1}
\]

The function is real and frequency independent.

Acoustic Wave Propagation in a Homogenous Half Space

In general, propagation effects may be classified as frequency independent and dependent losses and phase changes. The lumped attenuation coefficients of homogeneous media are frequency dependent and two pertinent examples are given below. The analytical solution to the wave equation for dilatational wave propagation in a viscous fluid is (Officer, 1958, p. 252):

\[
u = Ae^{-ax}e^{i\omega(t - x/c)}
\]
where $u =$ displacement in the $x$ direction,

$$\alpha = \frac{2}{3}(\omega^2 \nu / \rho c^3) = \text{attenuation coefficient}$$

$\nu =$ coefficient of viscosity, $\rho =$ density,

$$c = \sqrt{k/\rho} = \text{velocity of propagation of the dilatational wave in the fluid,}$$

$k = \text{incompressibility or bulk modulus.}$

Stoll (1977) applied Biot's theory to the propagation of low frequency acoustic waves in sediments (pore fluid in a skeletal frame) and again the form of the analytical solution for volumetric strain of the frame, $e$, and fluid flow, $\zeta$, is:

$$e = A_1 \exp[i(\omega t - lx)]$$

$$\zeta = A_2 \exp[i(\omega t - lx)]$$

where $l = l_r + il_m$ give the attenuation $l_m$ and the phase velocity, $\omega/l_r$, or:

$$e = A_1 e^{lm_x} \exp[i(\omega(t - l_r/\omega) x)]$$

$$\zeta = A_2 e^{lm_x} \exp[i(\omega(t - l_r/\omega) x)].$$

A plot of attenuation versus frequency according to Stoll (1977) is shown in Figure 4. In general, then, propagation in a homogeneous half space of fluid or sediment results in attenuation and dispersion given by the complex factor:

$$L(\omega, r) = e^{-\alpha r} \exp[i\omega r / v],$$

where dispersion is determined by the frequency dependence of the phase velocity $v$. 
Figure 4. Theoretical attenuation vs. frequency in sands and muds or clays according to Stoll (1977).

Propagation in a Layered Half Space

In a layered half space we must consider additional propagation effects. Frequency independent losses represent reflection and transmission coefficients at normal incidence for plane waves incident on the interfaces between fluid and non-dispersive elastic media. Reflection coefficients can however be frequency dependent even at normal incidence as in the case of reflection from a layer with a velocity gradient (Officer, 1958). Stoll (1977) derived a frequency dependent reflection coefficient for normally incident dilatational waves in water over sand, although the effect takes place above 100 Hz. Frequency dependent loss can be due to material attenuation as stated above, or due to intrabed multiples in thin
layers. For a layered half space we assume that material attenuation is constant in intervals.

Spencer (1977) discussed the effect of thin layers (thickness < \( \lambda \)) on the propagation of a seismic pulse. A delta function wavelet reflecting from a single interface is attenuated by the frequency independent reflection coefficient and its spectrum is flat. A single pulse into a system of layers will be dispersed in time into a multitude of output pulses and the spectrum of a window which includes a number of these will have an apparent spectral shape although each pulse has a flat spectrum. This is the apparent attenuation due to intrabed multiples discussed by Schoenberger and Levin (1974 and 1978). According to Spencer (1977) however, if the pulse length is longer than the two way intrabed travel times, the transmission coefficient is frequency dependent and a pulse propagating through a cyclic system of these non-resolvable layers will be broadened, approaching a Gaussian shape for long propagation path lengths, and the attenuation coefficient will be proportional to \( f^{2.5} \).

Enhanced backscattering may be expected from thin layers of thickness equal to \((2k+1) \lambda/4\) or odd multiples of \(\lambda/4\) where \(\lambda\) is the acoustic wavelength in the thin layer and it is assumed that there is a phase difference of \(\pi\) between the reflection from the upper and lower interfaces. This is the acoustic analog of the thin film effect. A distribution of layers with varying thicknesses will selectively backscatter acoustic waves of frequency, \( f = v(k+.5)/2d \), where \(k\) is an integer, \(v\) is the acoustic velocity of the material in the layer and \(d\) is the layer thickness (Rossi, 1957). A distribution of layers with an average thickness of \(\bar{d}\) would be expected to act as a notch filter with maximum
backscatter at $f = v(k+.5)/2d$ on an acoustic pulse propagating through it. Spencer gives the absolute value of the two-way transmission coefficient for a single thin layer embedded in an infinite homogeneous medium:

$$T(\omega) = \frac{(1 - R_1^2)^2}{(1 + R_1^2)^2 - 4R_1^2 \cos^2\omega \Delta_1}$$

where $\Delta_1$ is the two way travel time in the thin layer. He also gives a formula for the bandwidth, $\omega_0$, to the first minimum in the transmission coefficient of a non-interacting cyclic system for the special case in which all scattering layers are identical:

$$\omega_0 = \frac{v}{(R_1\pi \sqrt{8\phi z d})}$$  \hspace{1cm} (2.23)

where $\phi$ is the fraction of space occupied by the scatterers and $z$ is two times the total distance traveled by the wave in the system. The non-interaction assumption neglects multiples in the layers in between scatterers. Spencer considers a uniform intrabed thickness distribution, removes the non-interaction constraint and concludes that reflection coefficients greater than about 0.08 are necessary before noticeable pulse broadening occurs. These considerations may allow the detection of unresolved structure, if the intrabed reflection coefficients are large enough, in situations where the spectral shape of a reflection event is not a smooth monotonic function of frequency as predicted by material absorption theory.

Frequency independent phase changes can occur upon reflection from an elastic or fluid-elastic medium boundary for angles of incidence greater than the critical angle:

$$\theta_c = \sin^{-1}(v_1/v_2)$$
where \( v_1 \) = compressional sound velocity in medium in which ray is propagating and \( v_2 > v_1 \). The effect in the time domain after source signature deconvolution of a frequency independent phase shifted wavelet is given in Appendix 6. The result is the time domain representation of a Hilbert transform superimposed on a delta function. This was verified in tests on model data. The important point is that the location of the delta function is not shifted from its true location in time.

Frequency dependent phase changes are represented by dispersion. This type of phase change does cause shifts in peak locations in the time domain after deconvolution. This degrades somewhat the accuracy of attenuation measurements by the proposed least squares regression method. Fortunately however, dispersion is considered negligible for frequencies under 100 Hz (Ewing, Jardetzky and Press, 1957, p. 277). Molotova (1966) however concludes that dispersion will occur in highly absorbing terrigeneous rocks in the frequency range of 20 to 400 Hz. The magnitude of peak shifts in the time domain for a frequency dependent phase function will be discussed in the results of tests on model data.

The Fourier transform of the recorded reflection seismogram, including all the filters previously discussed, is given by equation (2.13). The measured source signature spectrum will be:

\[
S'(\theta, \omega) = S(\omega)B'(\theta, \omega)I(\omega)Gh'(\theta, \omega)
\]

(2.24)

where \( B'(\theta, \omega) \) is the directivity function at \( \theta \) for the direct wave impinging on the receiver and \( Gh'(\theta, \omega) \) is the ghost filter of the
recorded source signature. After division of (2.13) by (2.24) we have:

\[ X'(\theta,\omega,r) = B''(\theta,\omega)G''(\theta,\omega) \sum_{j} a_{j}(\omega,\tau_{j}) e^{-i\omega \tau_{j}}. \]  

(2.25)

where \( \tau_{j} \) is a function of \( r \) through velocity,

\[ B''(\theta,\omega) = B(\theta,\omega)/B'(\theta,\omega) \text{ and } G''(\theta,\omega) = Gh(\theta,\omega)/Gh'(\theta,\omega). \]

Now rewrite (2.25) and include a noise term \( \xi(\omega) \):

\[ Y(\theta,\omega,r) = \sum_{j} W_{j}(\theta,\omega,r)e^{-i\omega \tau_{j}} + \xi(\omega) \]  

(2.26)

where \( Y(\theta,\omega,r) = X'(\theta,\omega,r) \) and

\[ W_{j} = B''(\theta,\omega)G''(\theta,\omega)a_{j}(\omega, \tau_{j}). \]

The factors, \( a_{j}(\omega, \tau_{j}) \), for the layered half space include the product of all reflection and transmission coefficients and all accumulated attenuation and dispersion along the unique travel path encountered by the acoustic wave received at time \( \tau_{j} \):

\[ a_{j}(\omega, \tau_{j}) = \frac{1}{(2\pi r_{j})}\prod_{k} R_{k}^{m_{k}} T_{1}^{n_{1}} \exp \left( -\sum_{p} m_{p} \Delta r_{p} \right) \exp \left( i\omega \sum_{p} m_{p} \Delta r_{p}/v_{p}(\omega) \right) \]  

(2.27)

where \( r_{j} = \text{total travel path length} = \sum_{p} \Delta r_{p} \),

\( k = \text{number of different reflection coefficients, } R_{K} \)

\( m_{k} = \text{number of times } R_{K} \text{ was encountered} \)

\( l = \text{number of different transmission coefficients, } T_{l} \)

\( n_{l} = \text{number of times } T_{l} \text{ was encountered} \)

\( \alpha_{p} = \text{attenuation coefficient in interval } p \)
\[ m_p = \text{number of times thickness of interval } p, \Delta r_p, \text{ is traversed and} \]
\[ v_p(\omega) = \text{phase velocity of interval } p \text{ at frequency, } \omega. \]

The complex coefficients, \( W_j \), will now be estimated by means of a linear least squares regression model in the frequency domain.

Proposed Method for the Measurement of Attenuation in Reflection Seismograms

If the complex coefficients, \( W_j \), can be computed, spectral ratios relative to some reference layer at \( \tau_1 \) (bottom arrival) or at \( \tau_{j-1} \) (layer above) will allow equalization or cancellation of the common factors, \( B''(\theta, \omega) \) and \( G''(\theta, \omega) \), or for that matter \( S(\omega) \) if source signature deconvolution was not done. Spectral ratios relative to the layer above, if only primary events are chosen, will give:

\[
P_j(\omega, r_j) = \frac{W_j}{W_{j-1}} = \frac{a_j}{a_{j-1}} = \frac{r_j^{-1}T(j-1)d^{R_j}T(j-1)u}{r_jR_{j-1}} \exp(-2a_j\Delta r_j + i2\omega \Delta r_j/v_j)
\]

where \( R_j = \frac{\rho j+1 v_{j+1} - \rho_j v_j}{\rho j+1 v_{j+1} + \rho_j v_j} \), and the transmission coefficients for pressure are:

\[
\]
Figure 5 shows the single layer model. The modulus of this spectral ratio defined as \( \sqrt{p \times p} \) is:

\[
|P_j| = \frac{r_j^{-1} T(j-1)dR_j T(j-1)u}{r_j R_{j-1}} \exp(-2\alpha_j \Delta r_j). 
\]

In principle the attenuation coefficient \( \alpha_j \) can be computed from the spectral ratio, the interval velocity and the interval density as estimated from the velocity through empirical curves compiled by Nafe and Drake in Ludwig (1970) and from Hamilton (1970b);
\[ \alpha_j = (1/2 \Delta r_j) \ln \frac{r_{j-1}^T(j-1) d_{j} R_j T(j-1) u}{r_{j}^T R_j - 1 |p_j|} \quad (2.29) \]

The interval \( Q \) for the upper sedimentary layer is given by:

\[ Q = \pi f/\alpha v. \]

This of course neglects velocity and density gradients and the possibility of frequency dependent reflection and transmission coefficients and can therefore be only an estimate.

The following model is proposed for the solution of the complex coefficients, \( W_j \) in (2.26) from the Fourier transformed single channel reflection seismic data:

\[
\begin{bmatrix}
Y(\omega_j) & z(\omega_j)^n_1 & z(\omega_j)^n_2 & z(\omega_j)^n_3 & \cdots \\
Y(\omega_{j+\Delta j}) & z(\omega_{j+\Delta j})^n_1 & z(\omega_{j+\Delta j})^n_2 & z(\omega_{j+\Delta j})^n_3 & \cdots \\
Y(\omega_{j+2\Delta j}) & z(\omega_{j+2\Delta j})^n_1 & z(\omega_{j+2\Delta j})^n_2 & z(\omega_{j+2\Delta j})^n_3 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots \\
\end{bmatrix}
\begin{bmatrix}
W_1 \\
W_2 \\
W_3 \\
\vdots \\
\end{bmatrix}
\]

where, \( z(\omega) = e^{-i \omega \Delta t} \), \( \Delta t = \) digitizing interval, \( n_k = \) data point locations of chosen arrivals and \( \Delta j = \) number of frequency intervals skipped per equation. This model can be rewritten in terms of real numbers:
\[ Y = A W + \Xi \]

\[ Y_1 + iY_2 = (A_1 + iA_2)(W_1 + iW_2) + (\Xi_1 + i\Xi_2) \]

\[ Y_1 + iY_2 = A_1W_1 - A_2W_2 + i(A_1W_2 + A_2W_1) + (\Xi_1 + i\Xi_2) \]

\[ Y_1 = A_1W_1 - A_2W_2 + \Xi_1 \]

\[ Y_2 = A_2W_1 + A_1W_2 + \Xi_2 \]

or in matrix form,

\[
\begin{bmatrix}
Y_1 \\
Y_2
\end{bmatrix} = \begin{bmatrix}
[A_1] & [-A_2] \\
[A_2] & [A_1]
\end{bmatrix}
\begin{bmatrix}
(W_1) \\
(W_2)
\end{bmatrix} + \begin{bmatrix}
(\Xi_1) \\
(\Xi_2)
\end{bmatrix},
\]

(2.30)

where \( \Xi \) is the Fourier transform of the noise term.

The normal equations for the estimates, \( \hat{W} \), of the coefficients, \( W \) for the model (2.30) are:

\[
\hat{W} = (A^TA)^{-1}A^TY
\]

(2.31)

where \( A^T \) denotes \( A \) transpose. See Appendix 7 for a derivation of the normal equations for complex data. The solution vector represents the least squares best fit of the components \( W_j \) to the data in the frequency band of width, \( \Delta f = (\lambda-1)\Delta f/T \) where \( T \) is the record length of the Fourier transformed data and \( \lambda \) is the number of equations in the regression. The actual value of \( W_j \) will not necessarily be frequency independent in the band so it will be necessary to construct model data and test the effectiveness of the regression procedure for its ability to solve for the correct average value of \( W_j \). The regression is then done for successive, adjacent frequency
bands over a desired range within the interval from zero to Nyquist. The complex solution vector, \( W \), is reconstructed and spectral ratios are taken relative to the previous vector component or to the component representing the bottom arrival as desired. Spectral ratios corresponding to certain chosen acoustic impedance interfaces are stored and the process is repeated for another seismic trace. The final results are the averages for the spectral ratios corresponding to arrivals correlatable on all traces.

**Singular Value Analysis**

Rather than solve the normal equations (2.31) by straight forward techniques such as inverting \((A^T A)\) it was found necessary to obtain the solution of the linear least squares regression model by applying singular value analysis (Lawson and Hanson, 1974). Briefly, the \( m \times n \) matrix \( A \) of the model (2.30) is decomposed into:

\[
A = USV^T
\]

where \( U \) is an \( m \times m \) orthogonal matrix, \( V \) is an \( n \times n \) orthogonal matrix and \( S \) is an \( m \times n \) diagonal matrix. A square matrix \( B \) is orthogonal if \( B^T B = I \), the identity matrix. The diagonal entries of \( S \), \( (s_{ij}) \), are called the singular values of \( A \). The model \( A w = x \) is replaced by the equivalent problem,

\[
\begin{bmatrix}
S \\
0
\end{bmatrix} p = g,
\]

where \( g = U^T y \) and the trivial solution vector, \( p \), allows the computation of a sequence of candidate solutions, \( w(k) = \sum_{j=1}^{k} p_j v_j \).

where \( v_j \) is the \( j \)th column vector of \( V \).
This singular value decomposition is computed in two stages. First the \( m \times n \) matrix \( A \) is transformed into an upper bidiagonal matrix
\[
\begin{bmatrix}
B \\
0
\end{bmatrix}
\]
by a sequence of at most \( 2n-1 \) Householder transformations. The Householder transformation matrix, \( Q \), operates on a vector, \( v \), (column of \( A \)) and rotates it:
\[
Q = I_m + b^{-1} uu^T
\]
where \( u \) is an \( m \)-vector satisfying \( \| u \| \neq 0 \), \( b = -\frac{\| u \|^2}{2} \) and \( \| u \| \) is the norm of \( u = (\sum_{i=1}^{m} u_i^2)^{1/2} \).

The second stage is the application of a specially adapted QR algorithm to compute the singular value decomposition of \( B \):
\[
B = \hat{U}S\hat{V}^T.
\]
where \( \hat{U} \) and \( \hat{V} \) are orthogonal and \( S \) is diagonal. The QR algorithm is a very stable iterative procedure with global quadratic convergence, i.e. the superdiagonal elements of successive resulting bidiagonal matrices, \( B_k \), converge quadratically to zero as \( k \to \infty \). The matrices \( \hat{U}^T \) and \( \hat{V} \) for the successive iterations, \( B_{k+1} = \hat{U}_k^T B_k \hat{V}_k \), \( k = 1,2,... \), are chosen such that the matrix, \( \tilde{S} = \lim_{k \to \infty} B_k \), exists and is diagonal. Post processing to insure that the elements of \( S = \tilde{S}D \) are ordered and positive allow the final computation of \( U^T \), \( S \) and \( V \).

According to Lawson and Hanson (1974, p. 121-129) there are two important advantages to the use of singular value analysis. One is that methods based on forming and solving the normal equations typically require only about half as many operations as the House-
holder algorithm, but these operations must be performed using precision $\eta^2$ to satisfactorily handle the classes of problems that can be treated using the Householder algorithm with precision $\eta$. Furthermore, use of normal equations requires more storage locations in addition to the extra storage required by the higher precision. Cholesky decomposition, for example, is a method of solving the normal equations (2.31) by breaking them up into two triangular systems,

$$U^T \zeta = A^T \gamma$$
$$Uw = \zeta$$

where $U$ is an upper triangular matrix. The Cholesky factorization algorithm is:

a) \[ \sum_{k=1}^{i} u_{ki}u_{kj} = p_{ij} \text{ for } i = 1, \ldots, n; j = 1, \ldots, n \]
b) \[ v_i = p_{ii} - \sum_{k=1}^{i-1} u_{kii}^2 \]
c) \[ u_{ii} = \sqrt{v_i} \]
d) \[ u_{ij} = (p_{ij} - \sum_{k=1}^{i-1} u_{kii}u_{kj})/u_{ii}. \]

If the elements of $A$ are nearly parallel or linearly dependent, negative numbers may be produced at step 2.32(b) due to round off error and are set to zero to avoid imaginary elements in $U$. The result may be a matrix $U$ that is singular. Householder triangularization would not produce a singular matrix. This is important to the work in this thesis because this situation may arise due to the near linear dependence of the successive equations in the model.
where the frequency increment between equations must be kept small to have a large number of equations and yet a small bandwidth for each solution vector.

Error Analysis

It will be necessary in the analysis of field data to establish criteria by which the reliability of the solution vectors can be judged. Three types of criteria will be utilized: 1) computation of estimates of the standard deviations of the solution vector components, 2) use of the condition number, \( \kappa \), to estimate perturbation bounds for the least squares solution vector and 3) use of the F-test for significance of regression.

Estimates of the standard deviations of the solution vector components are obtained from the variance-covariance matrix, \( C \), of the solution parameters,

\[
C = (A^TA)^{-1}s^2,
\]

where \( s^2 \) is an estimate of the variance of the noise in the model (2.30), and is usually given by the normalized cumulative sum of squares of residuals,

\[
s^2 = \frac{||A\hat{\omega} - Y||^2}{m - n}
\]

where \( m \) and \( n \) are, respectively, the number of rows and columns of the matrix \( A \). This value of \( s^2 \) is a true estimate of the noise in the model assuming the model is correct. The variances are
given by the diagonal elements of C and the covariances by the off diagonal elements. The matrix C is computed in singular value analysis by,

\[
C = s^2 V s^2 V^T,
\]
or,

\[
c_{ij} = s^2 \sum_{k=1}^{n} v_i v_j k/(\text{sing})^2_{kk}
\]  \hspace{1cm} (2.33)

The standard deviations are the square root of the variances, \(\sigma_i = c_{ij}^{1/2}\). These are output along with the solution vectors in program FDPLOSS. The propagation of error for the average ratios of solution vectors is given in Appendix 8.

Lawson and Hanson (1974, p. 27) define the condition number, \(\kappa\), to be the ratio of the largest singular value of A to the smallest nonzero singular value of A. Perturbation bounds on the solution vector can be computed from this parameter:

\[
\frac{\| dw \|}{\| w \|} \leq \kappa [\gamma + (2 + \kappa \rho) \alpha + \gamma \beta]
\]

where \(\alpha = \frac{\| E \|}{\| A \|}\), \(\beta = \frac{\| dY \|}{\| Y \|}\), \(\gamma \leq \frac{\| Y \|}{\| Aw \|}\), \(\rho \leq \frac{\| Y - Aw \|}{\| Y \|}\),

\[K = \frac{\kappa}{1 - \kappa \alpha}\] and \(\| E \|\) is the norm of the perturbation matrix for A.

Neglecting the unknown perturbations, \(\alpha\), of A,

\[
\frac{\| dw \|}{\| w \|} \leq K \gamma = K \frac{\| dY \|}{\| Aw \|}, \text{ and since } Aw = Y,
\]

\[
\frac{\| dw \|}{\| w \|} \leq K \frac{\| dY \|}{\| Y \|}. \text{ Under the hypothesis that the noise in model (2.30) }$
\]
has zero mean and variance, $s^2$, $s$ can be interpreted as an unbiased estimate of the standard deviation of the errors in the data vector $Y$. Therefore,

$$
\frac{\|dw\|}{\|w\|} \leq \frac{s}{\|Y\|}
$$

(2.34)

This relation will be applied to field data and discussed in the results.

The F-test for significance of regression is used to determine if the least squares fit obtained by the proposed model accounts for the variation in the data vector $Y$. More specifically it is to test the null hypothesis, $H_0$, that $W_1 = W_2 = \ldots = W_{n-1} = 0$ against the hypothesis, $H_1$, that not all $W_i = 0$. The test is to compare the ratio of the regression mean square to the residual mean square with a selected percentage point of the F-distribution. That is, to compare:

$$
\frac{W^TATY/n}{s^2} = \frac{[(YNORM)^2 - CSS]/n}{(NSRCSS)^2}
$$

(2.35)

where $n$ is the number of degrees of freedom of the regression mean square assuming a negligible mean, $m-n$ is the number of degrees of freedom of the residual mean square, $CSS$ is the cumulative sum of squares and $NSRCSS$ is the normalized square root of the cumulative sum of squares from subroutine SVA, with the $F(n,m-n,\alpha)$ statistic (Draper and Smith, 1966). The Fisher's F-distribution is
used to test whether or not two measured variances or sums of squares come from the same normal population. If the calculated ratio is less than the F statistic given in the tables and if the errors are normally distributed, it can be reasonably assumed at the 1-\(\alpha\) probability level for a one-sided test that the two variances are from the same distribution. In the F-test for significance of regression, the regression sum of squares (variation explained by model) should be large compared to the residual sum of squares (difference between actual and predicted variation). For a statistically significant regression, the calculated F-ratio should be greater than the selected percentage point of the F-distribution by about four times in order to insure that the equation has not been "fitted to the errors only" (Draper and Smith, 1966, p. 64).

Another test that will be used with field data is the F-test for the extra sum of squares which is a special case of a general linear hypothesis that some of the variables or solution vector components are actually linear functions of one another. Specifically, we wish to test whether including more variables in the model allows a better fit to the data. That is, to test \(H_0: W_{q+1} = W_{q+2} = \ldots = W_p = 0\). The ratio:

\[
\frac{(SSR_R - SSR_F)/q}{s_F^2} = \frac{(CSS_R - CSS_F)/q}{(NSR"CSS_F)^2}
\]

is calculated where, \(SSR_R = (Y^T Y - W^T A^T Y)_R\) which is the sum of squares of residuals for the reduced model (excluding the variables
in question), SSR_F is the sum of squares of residuals for the full model and q is the number of variables in the null hypothesis. This ratio is compared to the F(q, m-n, 1-α) statistic. In this case H_0 is retained if the computed ratio is less than the F-value in the tables. That is, the residual sum of squares for the reduced model should not be too much greater than the residual SS for the full model if the additional q variables do not allow a significantly better fit to the data vector Y. A computed ratio less than F(q, m-n, 1-α) means that the additional variables are not justified at a 1-α level of significance for a one-sided test.

These tests will be used to determine if the model provides a suitable fit to the data and to determine if the exclusion of some events in the time domain will significantly decrease the ability of the regression to account for the variability in the data.

The assumption that the errors have a normal distribution of amplitudes is required for the validity of these tests but not the validity of the least squares solution. The question arises - what is the amplitude distribution of the Fourier transform of the noise term in (2.13) and (2.30)? This term may include the spectrum of random time domain noise plus the spectrum of a great number of multiple arrivals or unresolved primary events. According to Jenkins and Watts (1968, p. 233) even if the time domain representation of this noise process is not normal, the distribution of its power spectrum will be very nearly a χ^2 distribution with two degrees of freedom. The χ^2 distribution is a sampling distribution
of the variance where $\chi^2$ is defined as,

$$\chi^2 = x_1^2 + x_2^2 + x_3^2 + \ldots + x_v^2$$

and where $x_i$ are normally distributed random variables and $v$ is the number of degrees of freedom. By the Central Limit Theorem a $\chi^2$ distribution will approach normality at large degrees of freedom ($n > 30$), therefore even though the tests are not strictly applicable they are approximately so.

A final consideration is that of a three way fundamental limitation on the resolution of time, frequency and the statistical accuracy in computing spectral amplitudes of random series. In order to attain a frequency resolution of $\Delta f$ and a relative accuracy of $\Delta p/p$ of the spectral amplitudes, a time sample of duration at least,

$$T \geq \frac{1}{\Delta f (\Delta p/p)^2}$$

will be required (Gaaerbout, 1976, p. 83). This means for example that if a 10% uncertainty in the prediction of adjacent spectral amplitudes is required with a sample length of two seconds, the frequency resolution is only 50 Hz. For the same sample length, a 16% uncertainty corresponds to a resolution of 20 Hz and a 22% uncertainty corresponds to a resolution of 10 Hz. There is therefore a fundamental limitation on the accuracy with which spectral ratios can be compared in the adjacent frequency bands of the successive applications of the least squares regression routine.
The limitation imposed by the proposed least squares regression technique is, however, not as severe as with techniques which compute spectra from short window lengths in the time domain.
Section 1. Model Data

Source Signature Deconvolution

Synthetic seismograms were computed to study the effects of source signature deconvolution on time series which include frequency dependent and frequency independent losses and phase shifts, to test the effectiveness of bandpass and deghost filters, and to test the ability of the least squares regression technique to extract input frequency dependent loss functions.

An arbitrary source function (Figure 6) was convolved with a simple reflection coefficient series to produce the synthetic seismograms. Each impulse of the reflector series was given a different attenuation function by applying a window in the frequency domain which was symmetric about the Nyquist frequency. The window may be a complex function allowing the option of frequency dependent phase shifting. The synthetic seismogram is computed by Program WVW12 and the frequency domain windowing is done by subroutine WNDKW (appendix 10). Program WVW12 also has the option of additive random noise at any desired relative rms level. The square root of the variance of the noise free synthetic seismogram defined as,

$$\sigma_s = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})^2},$$

where $\bar{x}$ is the mean of the series $x_i$, is compared with the square root of the variance of a random
series $\sigma_N$ generated by subroutine RANBW, and a factor, FACT, is computed,

$$\text{FACT} = \frac{\sigma_S}{\sigma_N} (\text{RMSN})$$

which when multiplied by the random series will adjust the square root of its variance to the desired level, $(\text{RMSN})$, relative to that of the synthetic seismogram.

Model #1 is a reflector series of 4 impulses with the first being lossless having a reflection coefficient, $R_1 = 1.0$, the second has a linear ramp frequency dependent loss function with a frequency independent reflection coefficient, $R_2 = 0.5$, the third has a linear ramp loss function of greater slope than for pulse #2 with $R_3 = -0.5$ and the fourth pulse has an exponential loss function with $R_4 = 0.5$. The synthetic seismogram of Model #1 with no noise is shown in Figure 7, and its power spectrum in Figure 8. Figure 9(a) is the inverse Fourier transform from zero to Nyquist after source signature deconvolution by division in the frequency domain. This will be termed model 1(a). Notice the pulse broadening that occurs with frequency dependent attenuation although the peak locations are correct. Figure 9(b) shows the effect of including different frequency dependent linear phase ramps on pulses 2, 3 and 4. The result is a delay in peak location for a negative phase ramp and an advance for a positive phase ramp. This will be called model 1(b).
Figure 6. Arbitrary source function for model data and its power spectrum.

Figure 7. Synthetic seismogram of Model #1 with no noise; includes frequency dependent attenuation.
Figure 8. Power spectrum of signal on model #1.

Figure 9. Impulse response of Model #1 with no noise, (a) with frequency dependent attenuation, (b) with attenuation plus simulated dispersion.
**Frequency Domain Least-Squares Regression**

To test the ability of the proposed least squares regression technique to extract the input complex functions from the impulse response, a random noise series at a 20% relative RMS level was added to the synthetic seismogram of model 1(a). The impulse response after source signature deconvolution is shown in Figure 10. The input echo delays are at points 0, 20, 30 and 50. Figure 11 summarizes the results of regression on model 1(a). The plotted points are the 4 solution vector components at each frequency band and the error bars represent ± two standard deviations. Each frequency band is .125 Nyquist in width with 8 equations per regression. The error and deviation from the input functions increases at a frequency greater than .5 Nyquist because, as can be seen in Figures 6 and 8, there is no significant power above this frequency. Figure 12 summarizes the results of least squares regression (Program FDPLOSS) on model 1(b) to extract the input frequency dependent phase functions. Again the relative RMS noise level was 20%. In this case, the solution vector components follow the input phase ramps only if the true peak locations, not the actual measured peak locations are known. Some of the error bars representing ± two standard deviations are shown indicating the greater difficulty in extracting phase information than amplitude information. The number of equations per regression was 16 giving a bandwidth of .25 Nyquist. By running Program FDPLOSS again at a different starting frequency, a series of four overlapping bands gave solution components that follow the
input phase ramps reasonably well. It must be pointed out that a pulse at delay $\tau$ has a Fourier transform representation as $e^{-i\omega \tau}$. Its phase is therefore a linear ramp with frequency. The least squares regression routine is finding the additional phase information superimposed on the linear phase ramp corresponding to the delay $\tau$. It would not make any difference what the form of the phase function was; the choice of a linear ramp was arbitrary. The moduli of the complex solution vector components however were not sensitive to error in the peak locations. The overall average deviation of the moduli of the solutions using the apparent peak locations from those with the true peak location was only 5%.

Another model (model #2) was constructed with 10% relative RMS noise resulting in an impulse response after deconvolution as shown
Figure 11. Results of least squares regression on model 1(a) with 20% relative RMS noise; solid lines are the actual zero phase attenuation functions; error bars represent $\pm 2\sigma$; number of equations per regression is 8.
Figure 12. Results of least squares regression on model 1(b) with frequency dependent attenuation plus simulated dispersion; relative RMS noise level is 20%; error bars represent ±2σ; number of equations per regression is 16.
in Figure 13. Figure 14 shows the results of least squares regression compared to the input zero phase loss functions (solid lines). Again the agreement is good for the 4 overlapping frequency bands each of width .25 Nyquist up to about .5 Nyquist.

The model data studies verify the ability of the proposed least squares regression technique to extract the correct complex attenuation factors from an impulse series, even though the successive equations in each regression vary by only one frequency interval. This is attributed to the stability of singular value analysis. A test on the same model data to solve the normal equations using the standard Cholesky decomposition resulted in failure of the algorithm to give a solution vector. It is apparent, however, that the correct pulse delay times must be known in order to extract accurate phase information, but amplitude information is less sensitive to errors in peak location.

F-tests for significance of regression were done on model 1(a). The results are shown in Table 1. In every case where the standard deviations were small the computed F-value was larger than the corresponding F-statistic at the 95% level of significance, although all were not greater than four times the F(.95) value even though their solutions were acceptable. These results suggest that even though the noise does not have a normal distribution, the F-test for significance of regression is a good measure of the reliability of the solution vectors. Perturbation bounds for the solution vectors computed from the condition number were not reliable indicators of solution accuracy except in an approximate relative sense (Table 6).
Table 1. F-test for significance of regression

<table>
<thead>
<tr>
<th>Sonobuoy Number</th>
<th>p</th>
<th>n</th>
<th>Frequency</th>
<th>F-Ratio</th>
<th>$F_{p,v}(.95)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>10</td>
<td>80</td>
<td>39</td>
<td>6.72</td>
<td>1.98</td>
</tr>
<tr>
<td></td>
<td>78</td>
<td></td>
<td>8.18</td>
<td></td>
<td>1.98</td>
</tr>
<tr>
<td></td>
<td>117</td>
<td></td>
<td>2.35</td>
<td></td>
<td>1.98</td>
</tr>
<tr>
<td>57</td>
<td>8</td>
<td>80</td>
<td>39</td>
<td>1.81</td>
<td>2.08</td>
</tr>
<tr>
<td></td>
<td>78</td>
<td></td>
<td>5.72</td>
<td></td>
<td>2.08</td>
</tr>
<tr>
<td></td>
<td>117</td>
<td></td>
<td>17.81</td>
<td></td>
<td>2.08</td>
</tr>
<tr>
<td>61</td>
<td>10</td>
<td>80</td>
<td>39</td>
<td>2.90</td>
<td>1.98</td>
</tr>
<tr>
<td></td>
<td>78</td>
<td></td>
<td>5.54</td>
<td></td>
<td>1.98</td>
</tr>
<tr>
<td></td>
<td>117</td>
<td></td>
<td>4.48</td>
<td></td>
<td>1.98</td>
</tr>
</tbody>
</table>

fraction of Nyquist

<table>
<thead>
<tr>
<th>Model #1</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>.06</td>
<td>81.1</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.19</td>
<td>38.1</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.31</td>
<td>403.2</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.44</td>
<td>8.6</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.56</td>
<td>.67</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.69</td>
<td>.59</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.81</td>
<td>1.57</td>
<td>3.44</td>
</tr>
</tbody>
</table>
F-tests for extra sum of squares were run on model #2 to determine if eliminating variables had an effect on the residual norms of the regression. Table 2 summarizes the results for two reduced models. It appears that this test is erratic at frequencies where solutions are unacceptable but clearly favors the full model at frequencies where the solution vectors are acceptable. However, the solution components of the reduced model vary from the same ones in the full model by averages varying from only 1.3% to 12%. These tests with model data increase optimism that primary events and multiples not included in the regression can be considered part of the noise.
Figure 14. Results of least squares regression on model 2, with 10% relative RMS noise; solid lines are the actual zero phase attenuation functions; error bars represent $\pm 2\sigma$; number of equations per regression is 16.
Table 2. F-test for extra sum of squares for Model 2.

Model A (full) = 4 unknowns at data points 0, 20, 30 and 50
Model B (reduced) = 3 unknowns at data points 0, 20 and 50

<table>
<thead>
<tr>
<th>Frequency (% of Nyquist)</th>
<th>F-Ratio</th>
<th>F_{2,24}(.95)</th>
<th>Conclusion</th>
<th>% Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>.13</td>
<td>33.0</td>
<td>3.4</td>
<td>reject model B</td>
<td>3.3</td>
</tr>
<tr>
<td>.38</td>
<td>28.1</td>
<td>3.4</td>
<td>reject model B</td>
<td>1.3</td>
</tr>
<tr>
<td>.63</td>
<td>3.4</td>
<td>3.4</td>
<td>marginal</td>
<td></td>
</tr>
<tr>
<td>.88</td>
<td>.5</td>
<td>3.4</td>
<td>retain model B</td>
<td></td>
</tr>
</tbody>
</table>

Model A (full) = 4 unknowns at data points 0, 20, 30 and 50
Model B (reduced) = 3 unknowns at data points 0, 30 and 50

<table>
<thead>
<tr>
<th>Frequency (% of Nyquist)</th>
<th>F-Ratio</th>
<th>F_{2,24}(.95)</th>
<th>Conclusion</th>
<th>% Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>.13</td>
<td>150.7</td>
<td>3.4</td>
<td>reject model B</td>
<td>12</td>
</tr>
<tr>
<td>.38</td>
<td>172.0</td>
<td>3.4</td>
<td>reject model B</td>
<td>7</td>
</tr>
<tr>
<td>.63</td>
<td>.3</td>
<td>3.4</td>
<td>retain model B</td>
<td></td>
</tr>
<tr>
<td>.88</td>
<td>7.9</td>
<td>3.4</td>
<td>reject model B</td>
<td></td>
</tr>
</tbody>
</table>
Bandpass Filtering

Bandpass filtering is done by multiplication in the frequency domain after source signature deconvolution. Details of the algorithm are given in Appendix 3. A zero phase rectangular window with a cosine taper is applied to the Fourier transformed data. Results of bandpass filtering from 0-68.5% of Nyquist on the impulse response of model 1(a) is shown in Figure 15. Restricting the bandwidth to 0-38.5% of Nyquist results in a greatly improved impulse response as shown in Figure 16. The filter has a zero phase response since the peaks are not shifted from their true delay times. Improvement of a noisy impulse response can be achieved if the noise and signal spectra are not identical. Use of the bandpass filter (Subroutines LPFKW and HPFKW) and a notch filter (Subroutine SNOTCH) to suppress coherent and random noise will be discussed in conjunction with field data.

Frequency Independent Phase Shifts

The effects of frequency dependent and independent loss and frequency dependent phase shifts on the impulse response after source signature deconvolution have been discussed. One final condition, that of frequency independent phase shifts, needs to be treated to determine its effect on the detection of reflection events. Although this effect need not be considered for reflection at normal incidence from an acoustic impedance interface, it will be discussed for completeness. Frequency independent phase changes can take place
Figure 15. Bandpass filtered impulse response of model 1(a); 20% relative RMS noise; bandpass: no distortion from 0% to 68.5% of Nyquist.

Figure 16. Bandpass filtered impulse response of model 1(b); 20% relative RMS noise; bandpass: no distortion from 0% to 38.5% of Nyquist.
at angles of incidence greater than the critical angle for the establishment of head waves,

\[ \theta_c = \sin^{-1} \frac{V_1}{V_2} \]

where \( V_1 \) is the velocity of the medium which contains the incident wave and \( V_2 \) is the velocity of the medium below the boundary.

Derivation of the equations and a description of the algorithm for computation of a synthetic seismogram (Program WVKW8) with different phase shifts for each echo is given in appendix 6. Figure 17 shows the impulse response after source signature deconvolution of a noise free model with echo delays at 0, 10, 40 and 80 time units and phase shifts of 0°, 10°, 45° and 90° respectively. The result in curve (a)

![Figure 17. Source signature deconvolution of model with frequency independent phase shifts; a) no bandpass filtering, b) with bandpass filtering.](image-url)
is the superposition of a delta function at the true echo delay and the time domain representation of a Hilbert transform, the relative magnitudes of which depend on the phase shift. Curve (b) shows the effect of bandpass filtering. Appendix 6 gives an analytical treatment.

**Inverse Ghost Filtering**

Subsurface sources and receivers record the direct arrival plus a reflection from the surface phase shifted by 180°. This is the surface interference or Lloyd's mirror effect. The amplitude spectrum of the direct arrival and its "ghost" is shown in Figure 18(a). If the effect of scattering due to surface roughness (Rayleigh scattering)
is taken into account, the amplitude spectrum is smoothed as shown in figure 18(b). As the ghost delay time decreases, the nodes move to higher frequencies. Notice that low frequencies are attenuated relative to higher frequencies which explains why the low frequencies of the direct wave are attenuated as the distance from the source increases. The inverse or "deghost" filter will have poles at the zero locations in frequency of the forward filter. The frequency domain inverse filter must therefore be truncated to avoid noise amplification at these points. The details of the algorithm (Subroutines WGHOST and VGHST) are given in Appendices 4 and 10.

Models were computed to test the deghost subroutines. Figure 19 is a noise free synthetic seismogram of an arbitrary source.

Figure 19. Synthetic seismogram of the impulse response of a source-receiver ghost model convolved with an arbitrary source wavelet.
wavelet "ghosted" at the source location, reflected from a single deep interface and "ghosted" again at the receiver location. The resulting impulse response is shown in Figure 20, where the positive spike is delayed from the first arrival by the sum of the ghost time delays for the source and receiver. The deghosted result is shown in Figure 21 where the inverse filter was truncated at all poles at the amplitude of the spectrum for a frequency of 0.06 Nyquist. Figure 22 shows less noise when the truncation point is at 2% of Nyquist. The extreme sensitivity of the deghost filter to inaccuracy in the delay times is illustrated by Figures 23 and 24 where one ghost delay and both ghost delay times respectively are in error by only one data point. The implications of this problem to field data processing will be discussed in the next section.

Figure 20. Impulse response of source-receiver ghost model; ghost delays at 7 and 11 data points.
Figure 21. Impulse response of source-receiver ghost model after application of inverse ghost filter with truncation point at 6% of Nyquist.

Figure 22. Impulse response of source-receiver ghost model after application of inverse ghost filter with truncation point at 2% of Nyquist.
Figure 23. Impulse response of source-receiver ghost model with one ghost delay in error; truncation point at 6% of Nyquist.

Figure 24. Impulse response of source-receiver ghost model with two ghost delays in error; truncation point at 6% of Nyquist.
Section 2. **Field Data**

**Data Processing Procedure**

The data to be analyzed was recorded on seven track, analog magnetic tape. Four channels were digitized using the arrangement shown in Figure 25. One channel was the universal time clock code, two were streamer channels at different amplifications and the fourth was the sonobuoy channel. The Krohn-Hite bandpass filters were set with a high frequency cutoff at 250 Hz to provide an anti-alias filter for the digitizing frequency of 500 Hz.

The data processing procedure is shown in Figure 26. See Appendix 10 for a listing and flow charts of the programs and subroutines. The near vertical incidence reflection data recorded by the streamer was analyzed for frequency dependent attenuation and the wide angle reflection (WAR) data as recorded by the sonobuoys was analyzed for velocity structure. Sonobuoy data was rejected for attenuation analysis because of the problems associated with the unknown amplitude dependent filtering of automatic gain control in the sonobuoy package. The problems with streamer data include the phase characteristics of its directivity function and the relatively low amplitude level of the reflection coda. As shown in model data studies, lack of phase equalization can produce peak shifts in the time domain which can possibly be deleterious to the accuracy of the least squares regression model. Small shifts, however, were determined not to have a significantly degrading effect on the extraction of amplitude information. Phase information will not be analyzed in this study. Tests of the effect
Figure 25. Block diagram for digitization of analogue-recorded data.
Figure 26. Data processing flow chart.
of the directivity function or the location of the bottom arrival showed no measurable shift between the input data and the deconvolved, processed data. This is attributed to the near symmetry of the phase spectrum for the streamer over the band from zero to 50 Hertz that contains most of the power of the source function (Figure 3). That is, an equal amount of phase is positive as is negative, minimizing the net peak shift. For this reason and to avoid noise amplification at filter poles, an inverse filter for the streamer directivity function was not included in the data processing. As shown in Figure 3 the amplitude and phase spectrum of the sonobuoy is nearly a constant at one and zero respectively over the seismic band, so its effect may be neglected.

The source signature was measured differently for streamer and sonobuoy data as previously mentioned. A typical example of a 30 trace stack of the air gun direct arrival received by the streamer and its power spectrum is shown in Figures 27 and 28, respectively. Program STACKW handles the magnetic tape and stacks the streamer direct wave.

The source signature used with sonobuoys #3 and #5 was a stacked direct wave as received by the sonobuoy near the origin of the respective wide angle reflection runs (Program STKSS). Since the surface interference effect acts as a high pass filter, only a few arrivals near the beginning of the run were suitable for stacking. Figure 29 shows the five trace stack comprising the source signature for sonobuoy #3. Its amplitude spectrum is shown in Figure 30. Another problem with this method of developing a source signature is that the
Figure 27. Stacked streamer source signature for SB 3; 30 traces.

Figure 28. Typical power spectrum of stacked direct wave recorded by streamer.
Figure 29. Stacked source signature received by SB 3; 5 traces; used for SB 3 WAR deconvolution.

Figure 30. Amplitude spectrum of SB 3 WAR source signature.
automatic gain control is in operation for the first few direct wave arrivals. Its response is of course different for the bottom reflected signal. A third type of source signature was used to deconvolve the wide angle reflection data of sonobuoys #46, #57, and #61. The first .2 seconds of the bottom reflected arrival to sonobuoy #46 was averaged (Program STKSB). The 21 trace stack is shown in Figure 31 and its amplitude spectrum in Figure 32. The improvement in the deconvolved wide angle reflection data can be seen by comparing the results of SB 46, 57 or 61 with that of SB 3, (Appendix 9). This is attributed to the fortuitous homogeneity of the upper sediment layer of the abyssal plain environment at SB 46. The advantages of this method are: 1) the AGC level is about the same for the source signature and the signal to be deconvolved, 2) the source and receiver ghosts are included in the source signature (neglecting small variations in delay over the 21 trace stack) and 3) the propagation path lengths of the signal to be deconvolved and the source signature are about the same, thus insuring that the low frequency absorption of sea water, though small (Urick, 1967, p. 91) will not have an effect. Use of this source signature for deconvolution of the WAR data of SB 3 was not fruitful, possibly because of the inapplicability to SB 3 of some combination of the three advantages given above. Stacking a bottom reflected source signature for SB 5 was not fruitful presumably because of the presence of subsurface structure.

Deconvolution to remove the oscillatory effects of the bubble pulse is done by dividing the complex Fourier transform of the signal
Figure 31. Stacked bottom reflected arrival at SB 46; 21 traces; source signature for WAR deconvolution of SB 46, 57 and 61.

Figure 32. Amplitude spectrum of 21 trace stack of bottom reflected arrival at SB 46.
by the complex Fourier transform of the source signature. The number of data points in the FFT for the numerator and the denominator must be identical powers of two. Results with field data show that slight mismatches of the signal and source signature spectra, especially at points of near zero amplitude in the spectrum of the denominator, amplify noise at these frequencies. It is essential to use a small digitizing interval (<.002 sec) in order to assure that frequency components in the seismic band of the signal and source signature can be matched as accurately as possible.

Noise Suppression

The two kinds of noise from the natural environment are either random or coherent. The random noise from streamer drag, instrumentation, etc. may be colored; that is, its amplitude spectrum may not be constant or white. Two troublesome types of coherent noise were caused by the ships propeller (~10 Hz) and instrumentation (60 Hz). In addition these noises may be amplified by the data processing steps involving the inverse filters -- deconvolution and deghosting.

The most effective method for suppressing random noise is by stacking or averaging redundant traces. The success of this technique is exemplified by the smoothness of the stacked source signature in Figure 27. This cannot be done with the signal to be deconvolved without introducing an unknown structure dependent filter, since this data was recorded from a moving platform and there is therefore a moving reflection point at depth. Subtraction of the spectrum of a noise sample from the spectrum of a signal will help if large amplitude
coherent noise is present in the seismic band as is the case with vertical profiler data. Figure 33 shows the effect on the spectrum of SB 3 streamer data with and without noise subtraction. The variance of the random component is doubled by this procedure but the least squares regression method which uses the streamer data is more adversely affected by the presence of coherent noise than random noise. Noise subtraction was not done for WAR data. The best and easiest technique for suppressing coherent and processing noise without altering the amplitude information in the seismogram in a non-deterministic way was found to be judicious bandpass and notch filtering. See Appendices 3 and 10 for descriptions and flow charts of these subroutines.

Examples of the processing steps on a single trace of sonobuoy data is shown in Figure 34. The input unfiltered trace is shown in Figure 34(a) and after bandpass filtering in 34(b). If the source signature received by the streamer is used for deconvolution, the different instrument responses cause error in the polarity of the bottom arrival 34(c). Deconvolution with a source signature which is a five trace stack of the direct wave received by the sonobuoy is shown in 34(d). The source signature includes the "stack deghost" filter of equation (2.20) and is bandpass filtered from 2.5 to 93.7 Hz where the stated frequencies are at the zero level of the amplitude spectrum. Notice that the first arrival is of the correct polarity. The second pulse is negative and is due to the surface ghost. Applying the source-receiver deghost filter attenuates this negative arrival (34(e)). If the source and receiver ghost time
Figure 33. Spectrum of signal in a single trace received by SB 3; a) without and b) with noise subtraction.
Source signature deconvolution with deghost filter; time delays in error bandpass: 5 to 150 Hz

Same as (d) except with source-receiver deghost filter

Deconvolution with stacked source signature received by sonobuoy; bandpass: 2.5 to 93.7 Hz, no source-receiver deghost filter

Deconvolution with stacked source signature received by streamer; bandpass: 0. to 150 Hz

Input data, bandpass filtered from 2.5 to 93.7 Hz

Input data, no filtering, sonobuoy #3

Figure 34. Field data processing examples.
delays were in error, noise at the filter poles is amplified as shown in 34(f). The inverse ghost filter was found to be very sensitive to the ghost delay times. Even though these delay times were measured from the signature amplitude spectra as described in Chapter 2, considerable experimentation was necessary to optimize them. Only sonobuoys #3 and #5 required deghosting since the source signature for sonobuoys #46, #47 and #61 include the effect of the source and receiver ghosts.

After deconvolution of WAR and streamer data, a scanning algorithm may be opted which prints the data point and time locations of all peaks above a desired threshold level. The arrival times of selected reflection events are then read from the printout.

**Interval Velocity Analysis**

Interval velocities between selected reflection events are computed by Program LAYRVEL which implements the modified \( T^2 - X^2 \) method developed by LePichon et al. (1968). See Appendix 1 for a complete description and derivation of formulas used in the algorithm. The round trip travel time-direct wave (air gun to sonobuoy) travel time data pairs for selected reflection hyperbolas are obtained from the output of Program WARDCN. The direct wave travel times (D-times) are computed by searching for the direct wave on the sonobuoy channel after the shotbreak on the streamer channel. Because of the rapid decrease in intensity of the direct wave with distance only a few D-times were measured directly and these were used to
determine an average difference in D-time, $\bar{T}_D$. Subsequent D-times were computed according to:

$$T_{D_i} = T_{D_{i-1}} + \bar{T}_D .$$

Inaccuracies in the computed $T_{D_i}$ for some sonobuoys required a corrected $T_{D_i}$ to be computed by the plane dipping layer formula (equation (A.1.1)):

$$T_D = \frac{T_0 V}{V_H} \left( \sqrt{\sin^2 \theta + \left( \frac{T^2}{T_0^2} - 1 \right) - \sin \theta} \right)$$

where $T_0$ is the two way travel time at the origin,

$V_H$ is the sea surface sound channel acoustic velocity estimated from the unprocessed original data,

$T$ is the two way travel time to the bottom as read from the computer printout, and

$\theta$ is the dip of the sea floor as measured from the original vertical profiler data.

Using this formula for SB 46, 57 and 61 was justified since the sea floor was very flat at these locations. Dips for layers below the sea floor are computed from structure discernable in the unprocessed vertical profiler data by first using estimated interval velocities and then iterating with computer determined velocities. The vertical water column velocity is read from Matthew's Tables (1939). Program LAYRVEL prints the interval velocity and its standard deviation computed from the least squares best fit of a straight line to the
reduced $T^2 - X^2$ plot for each layer at the origin and the thickness and depth of each layer at the origin.

The $T^2 - X^2$ method begins to fail altogether for water depth to layer thickness ratios greater than 50 (Bryan, 1974) and becomes unreliable for ratios greater than 12 (Houtz, 1974). This is attributed to errors in the assumption of straight line paths in the water column. Increasing the accuracy of reflection event timing through deconvolution techniques can help increase the accuracy of interval velocity computations only if 1) the thickness ratio limits are not exceeded, and 2) the layers are truly plane dipping and homogeneous. The thickness limitation affected some of the velocity computations requiring consolidation of thin layers to get a reasonably accurate average interval velocity (see for example, the results of SB 61).

Other causes for error in the interval velocity computations are non-constant ship velocity, departure from the plane dipping layer assumption, horizontal and under some conditions vertical velocity gradients, choice of multiples rather than primary reflection hyperbolas, uncertainty in the location of primary events due to noise interference, dip uncertainties and short reflection hyperbolas.

**Attenuation Analysis**

Data preparation for attenuation analysis involved five steps because of constraints on system usage and capabilities: 1) disk storage of selected Fourier transformed signal traces of vertical profiler data on the CDC NOVA computer system; 2) conversion to EBCDIC
coding and storage on 9 track magnetic tape; 3) conversion of 9 track tape to 7 track tape by OSU computer services; 4) copy to disk file storage on the OSU CYBER computer system; and 5) computation of spectral ratios using Program BFLSVA on the CYBER system.

Program FDPLOSS (binary version-BFLSVA) computes the average complex moduli of spectral ratios for any number of input traces. See Appendix 10 for a description of input parameters with example input and output. Output on the line printer is spectral ratios and standard deviations at each frequency for each data set, and the averaged spectral ratios and standard deviations for all data sets both uncorrected and corrected for geometric spreading. There is an option for interval Q computation but these results will be neglected in this dissertation because error propagation was found to be excessive.

Analysis and Discussion of Sedimentary Environments

Figure 35 is a map showing the locations of the five sedimentary environments studied and nearby piston cores including a DSDP drill site. Appendix 9 contains the input time domain traces and the deconvolved traces for all WAR and vertical reflection profiles.

Sonobuoy 3

This site is over an outer shelf basin on the northern Californian margin at 41°40' N., 125° W. The vertical profiler record and interval velocity solutions are shown in Figure 36. Piston core site 6706-1 at 42°6' N, 124°56' W (Spigai, 1971) penetrated 430 cm and
Figure 35. Map showing location of tracklines, sonobuoys and core sites.
Figure 36. Vertical seismic record of an outer shelf basin of northern California and interval velocity results for SB 3.
revealed surface sediments of olive gray to gray lutite with a sand-silt layer at 11 cm. Lutite is sediment composed of silt and clay-sized particles. Surface sediment compressional wave velocities in continental terrace environments have been measured by Hamilton (1970b). Sand-silt-clay is 1.578 km/sec and silty sand is 1.677. Very fine sands to coarse sands range from 1.711 to 1.836. Taking compaction into consideration, the measured velocity of 1.63 km/sec over an interval 174 m thick suggests the presence of sandy silt with approximately 40-50% sand and an upper limit mean grain diameter of 0.07 to 0.1 mm (Hamilton, 1970b; Smith, 1974).

The velocity versus depth and spectral ratio plots are shown in Figure 37. Error bars represent ± one standard deviation. Spectral ratios relative to the layer above are not shown because of large error propagation. Spectral shaping occurs over the band from 20 to 50 Hz, with a minimum for the first layer at 40 Hz, for the second layer at 30 Hz and for the third layer at 50 Hz. If this is attributed to non-resolvable cyclic stratification with layers of equal thickness, the non-interaction formula of Spencer (1977) (equation 2.29) suggests average thicknesses of 13 m for the first layer, 30 m for the second layer and 3 m for the third layer assuming $\phi = 60\%$ and $R_i = 0.1$. The acoustic impedances of the upper and lower boundaries of the first interval were computed from the parameters in Table 3. The density and acoustic velocity of the sediment upper boundary were estimated from the measured average grain sizes (Phi mean diameter = 8.6) in core 6706-1. The bottom water sound velocity was computed from the temperature T, salinity S, and depth D according to (Medwin, 1975):
Figure 37. Velocity versus depth and average moduli of complex spectral ratios versus frequency for a continental shelf basin at SB 3. Spectral ratios are relative to arrival from sea floor and are not corrected for geometric spreading.
Table 3. Parameters for attenuation estimates.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Reference</th>
<th>SB 3</th>
<th>SB 5</th>
<th>SB 61</th>
<th>SB 57</th>
<th>SB 46</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depth (m)</td>
<td></td>
<td>952.3</td>
<td>2592</td>
<td>3041</td>
<td>2913</td>
<td>3658.7</td>
</tr>
<tr>
<td>Thickness (m)</td>
<td></td>
<td>174.1</td>
<td>455.2</td>
<td>74.5</td>
<td>172.1</td>
<td>213.5</td>
</tr>
<tr>
<td>Bottom water temp. (°C)</td>
<td>Sverdrup (1942)</td>
<td>4.5</td>
<td>2.1</td>
<td>1.7</td>
<td>1.7</td>
<td>1.56</td>
</tr>
<tr>
<td>Bottom water salinity (o/oo)</td>
<td></td>
<td>34.3</td>
<td>34.6</td>
<td>34.7</td>
<td>34.7</td>
<td>34.7</td>
</tr>
<tr>
<td>$V_1$ (water)</td>
<td>Medwin (1975)</td>
<td>1483.14</td>
<td>1499.0</td>
<td>1505.12</td>
<td>1503.03</td>
<td>1514.35</td>
</tr>
<tr>
<td>Specific volume of seawater, $\alpha_{s,0,p}$</td>
<td>Neumann and Pierson (1966)</td>
<td>.9684</td>
<td>.96136</td>
<td>.96005</td>
<td>.96057</td>
<td>.95737</td>
</tr>
<tr>
<td>$V_2$ (sed. upper boundary)</td>
<td></td>
<td>1.519</td>
<td>1.6</td>
<td>1.521</td>
<td>1.520</td>
<td>1.505</td>
</tr>
<tr>
<td>$V_2$ (sed. lower boundary)</td>
<td></td>
<td>1.634</td>
<td>1.7722</td>
<td>1.560</td>
<td>1.462</td>
<td>1.71</td>
</tr>
<tr>
<td>$\rho_2$ (sed. upper boundary)</td>
<td>(g/cm$^3$)</td>
<td>1.423</td>
<td>1.6</td>
<td>1.24</td>
<td>1.24</td>
<td>1.26</td>
</tr>
<tr>
<td>$\rho_2$ (sed. lower boundary)</td>
<td>Hamilton (1970b)</td>
<td>1.85</td>
<td>1.98</td>
<td>1.63</td>
<td>1.36</td>
<td>1.81</td>
</tr>
<tr>
<td>$V_3$</td>
<td></td>
<td>1.963</td>
<td>2.60</td>
<td>2.56</td>
<td>2.04</td>
<td>3.18</td>
</tr>
<tr>
<td>$\rho_3$</td>
<td>Ludwig (1970)</td>
<td>2.0</td>
<td>2.22</td>
<td>2.13</td>
<td>1.95</td>
<td>2.25</td>
</tr>
<tr>
<td>Maximum attenuation, $\alpha$ (nepers/m)</td>
<td></td>
<td>$2.3\times10^{-3}$</td>
<td>$4.6\times10^{-4}$</td>
<td>.012 @</td>
<td>$8\times10^{-3}$ @</td>
<td>$2.9\times10^{-3}$ @</td>
</tr>
<tr>
<td></td>
<td></td>
<td>@ 40 Hz</td>
<td>@ 80 Hz</td>
<td>195 Hz</td>
<td>117 Hz</td>
<td>127 Hz</td>
</tr>
</tbody>
</table>

1 OSU Rep. 63-33
2 Sverdrup, 1942
3 Hamilton, 1970b
4 Kulm, 1973
5 Ludwig, 1970
\[ C = 1449.2 + 4.6T - 0.055 T^2 + 0.00029T^3 + (1.34 - 0.010T(S-35)) + 0.16D. \]

This gave a bottom reflection coefficient of 0.17. From formula (2.28), the maximum attenuation at 40 Hz for the first sediment interval was 0.02 dB/m with a Q of 33. This value is too high but represents an upper limit on the material attenuation since the reflectivity of the lower interface was estimated from the interval velocities and it is possible that there is a stratigraphic component present.

**Sonobuoy 5**

Data for sonobuoy 5 was taken on cruise W7612-A of the R/V WECOMA on December 3, 1976. It was located over the middle Astoria sea fan at 45°48' N, 126°10' W. DSDP drill core site 174 is at 44°53' N, 126° 21' W (Kulm, von Huene, et al., 1973). The sediment column at site 174 is composed of two units. The first (0-284 m) consists mainly of upper Pleistocene medium to very fine turbidite sands. Individual beds have a basal sand which grade upward into a silt or silty clay with sharp contacts commonly occurring between the underlying silty clay and the overlying sand interval. Occasionally thin interbeds of graded silts and silty clays occur within the main sandy sequence. Sand intervals comprise 62 percent, silty clay, 29 percent and silt, 9 percent of unit 1. The beds range in thickness from 5 to 733 cm
but generally are between 50 and 100 cm. The second unit (284-879 m) consists of Upper Pleistocene to Pliocene sediments. It is characterized by thin beds of compacted silty clay with a basal silt grading upward into an olive gray silty clay and then into a dark greenish gray fissile silty clay. Most beds are separated by a sharp basal contact and display the graded bedding typical of turbidites. The contact between the two units represents the base of the Astoria fan which is prograding over the abyssal plain sediments of unit 2.

The sediment column at SB 5 has a computed thickness of 1835 m compared to a measured thickness of 911 m for site 174. It is probable that unit 1 corresponds to the first interval with a computed velocity of 1.77 km/sec and thickness of 455 m. Figure 38 summarizes the morphology and interval velocities. Difficulty with measuring an adequate source signature and the presence of non-planar subbottom layers prevented computation of the velocities of thinner intervals. A very narrow bandwidth (2.5-29.5 Hz) allowed resolution of a surface layer 103 m thick but the velocity of 1.497 km/sec was inaccurate due to the thickness limitation of the $T^2 - X^2$ method and, in addition, the hyperbola was short. Hamilton (1970b) reports the velocity of fine sand in continental shelf-slope environments to be 1.742 km/sec and for a proportion of sand greater than 62% he reports a velocity of about 1.68 km/sec (continental terrace). Taking compaction and relative proximity to source into consideration the measured interval velocity agrees well with the expected sediment type in this location.

Figure 39 shows the velocity versus depth and spectral ratio plot for SB 5. Only the band from 50 to 90 Hertz had acceptable
Figure 38. Time section and interval velocities for SB 5 over the upper Astoria sea fan.
Figure 39. Velocity versus depth and average moduli of complex spectral ratios relative to arrival from sea floor for SB 5 over the middle Astoria sea fan. Spectral ratios are not corrected for geometric spreading.
standard deviations. Using the parameters shown in Table 3 to compute acoustic impedances the maximum attenuation in the 1.77 km/sec interval at 80 Hz was 0.004 dB/m or 0.12 dB/100' with a Q value of 308. Compiled data by Hamilton (1974) and Usher (1962) indicates a range of .005-.008 dB/m in natural saturated sediments. Anderson and Blackman measured an attenuation coefficient of 0.0031 at 200 Hz in sediments. Attenuation for SB 5 is less than expected for sand if linear attenuation with frequency is assumed and the data compiled by Hamilton (1974) is extrapolated to 80 Hz (Figure 40). Stoll (1977) however, predicts a departure from the linear dependence for sands, with less attenuation at lower frequency (Figure 4). The theoretical curve for an interval of this thickness is consistent with a predominantly sand composition as shown in Figure 38. Using an average bed thickness of 75 cm or 2.5 feet, 0.1 dB/100' of stratigraphic attenuation at 100 Hz would require intrabed reflection coefficients of about 0.24 (Spencer, 1977). The maximum possible reflection coefficient for a silty clay-fine sand contact would be 0.22, from Hamilton, (1970b). It is not likely that such large intrabed reflection coefficients exist in situ over the entire thickness. Considering its low value, the data suggests that the measured attenuation in this case is largely that of the sediment itself.

Sonobuoy 46

Sonobuoy #46 was located over the Tufts abyssal plain west of the west end of the Blanco fracture zone at 45°10' N, 132° W. A catcher at 44°57' N, 132° W revealed brown clay (OSU Core Library). Piston
Figure 40. Attenuation versus frequency in natural, saturated sediments and sedimentary strata (Hamilton, 1974). Symbols: circles: sands (all grades); squares: clayey silt, silty clay; triangles: mixed sizes (e.g., silty sand, sandy silt, sand-silt-clay. See Hamilton (1974), figure 1, p. 196 for data sources. Open circle is maximum attenuation in the surface layer at SB 5 and open square is maximum attenuation in the surface layer at SB 46.
core 6709-21 at 45°31' N, 131°33' W was 83 cm long and contained brown clay with about a 20 cm thick calcareous layer near its mid section. Core 6709-20 at 46°4' N, 131° W was 1168 cm long and contained brown clay in the bottom meter. Cores 6709-20 and 21 were examined by the author.

The deconvolved records reveal an acoustically transparent layer with a two-way travel time thickness of .20 seconds and a thin layer .040 seconds thick overlies acoustic basement. The data suggests a low velocity (1.51 ± .3 km/sec) for the first interval and 2.63 ± .4 for the thin layer but the errors were excessive due to a short hyperbola length for the reflection from the upper boundary of the thin layer. Therefore, the reported value of 1.71 km/sec is for the combined layers to acoustic basement with a total thickness of 213 meters. Immediately underlying acoustic basement is a layer 88 m thick with a velocity of 3.18 km/sec. Refractions suggest that layer 2 does not begin for another 370 m below this layer assuming 3.5 km/sec for the interval. These results are shown in Figure 41.

The velocity versus depth and spectral ratios relative to the bottom and to the layer above are shown in Figures 42 and 43. The maximum attenuation for the first sediment interval was estimated at 0.025 dB/m at 127 Hz and is plotted in Figure 40. This corresponds to a minimum Q-value of 80 for the interval. The peak in the spectral ratio at 107 Hz corresponds to an attenuation of .012 dB/m. Refer to Table 3 for values of parameters used in these estimates. Since linear extrapolation of high frequency attenuation results to low frequencies is acceptable for muds and clays (Figure 4), this
Figure 41. Time section and interval velocities for SB 46 over the Tufts abyssal plain near the west end of the Blanco fracture zone.
Figure 42. Velocity versus depth and average moduli of complex spectral ratios relative to arrival from sea floor for SB 46 over the Tufts abyssal plain. Spectral ratios are not corrected for geometric spreading.
Figure 43. Spectral ratios relative to layer above for SB 46. Spectral ratios are not corrected for geometric spreading.
attenuation suggests clayey silt or silty clay material (Hamilton, 1974) and would be in accordance with a low velocity (1.505-1.521 km/sec) for most of the interval. The spectral ratios of the first interval are greater than one which indicates a low bottom reflection coefficient relative to that at the base of the layer. Within the limits of error, spectral shaping due to non-resolvable structure is not apparent in the second and third layers, and the attenuation in the first layer is essentially accounted for by material absorption within the error of the spectral ratios.

**Sonobuoy 57**

Sonobuoy #57 was located west of Cascadia sea channel and north of the Blanco fracture zone at 44°15' N, 127°30' W. Two piston cores in the vicinity were 6604-31 at 44°14' N, 127°6' W (Griggs, 1969) and 6509-28 at 44°27' N, 127°20' W (Duncan, 1968). Both cores are described as lutite (silty clay or clayey silt) and have 0 to 1 percent sand. Core 6509-28 has a 10 cm thick surface layer of brown lutite (20% silt, 79% clay) overlying gray lutite (47% silt, 52% clay). Core 6604-31 has 85% silt, 15% clay down to 131 cm after which the composition shifts to predominantly clay for a few cm to the base of the core. Hamilton (1970b) reports the sound velocity of silty clay in a turbidite abyssal plain environment as 1.52 km/sec.

Deconvolution with the stacked bottom arrival of SB 46 resulted in greatly improved resolution of reflection events as shown in Appendix 9. Computation of the velocity of a well defined surface layer 0.028 sec of two way time from the sea floor gave 1.54 ± .12 km/sec.
This velocity is inaccurate because of the thickness limitation of the $T^2 - \chi^2$ method. The overall velocity of the interval from the surface to the next major interface was $1.46 \pm 0.03$ km/sec, giving a thickness of 172 m. This low velocity suggests that the material is predominantly a fine clay with a sediment mean diameter of .001 mm (Shumway, 1960). The lower unit has a velocity of 2.04 km/sec, a thickness of 267 m and overlies acoustic basement with a layer 2 velocity of 4.6 km/sec determined from refracted arrivals. Figure 44 shows these results in a time section.

The velocity versus depth and frequency dependent spectral ratio plots are shown in figures 45 and 46. Figure 47 is the power spectrum of a selected trace of streamer data. Notice that measurable power extends out to about 125 Hz. Using the parameters in Table 3 the maximum attenuation in the first and second interval is .05 to .07 dB/m at 100 Hz. This value suggests sand or sand-silt-clay mixtures (Hamilton, 1974). The discrepancy between the material type based on velocity and attenuation may be due to the presence of a stratigraphic component of attenuation, in which case the minimum in the spectral ratios at 100 Hz suggests an average non-resolvable bed thickness of about 5 meters. The calculation is from equation (2.29) with $\phi = 50\%$, an intrabed clayey silt-clay reflection coefficient of 0.06 and a velocity of 1535 m/sec (turbidite abyssal plain clayey silt) for the scatterer embedded in a clay medium.
Figure 44. Time section and interval velocities for SB 57 near the Cascadia sea channel north of the Blancofracture zone.
Figure 45. Velocity versus depth and average moduli of complex spectral ratios relative to arrival from sea floor for SB 57 near the Cascadia sea channel. Spectral ratios are not corrected for geometric spreading.

Figure 46. Spectral ratios relative to layer above for SB 57. Spectral ratios are not corrected for geometric spreading.
Sonobuoy 61

Sonobuoy #61 was located near the base of the Oregon continental slope in the southern end of the Astoria sea channel at 43°42' N, 125°40' W. Piston core 6511-5 at 43°40' N, 125°29' W penetrated about 4.5 m and revealed all light olive gray lutite (silty clay) (Duncan, 1968).

Deconvolution with the stacked bottom arrival of SB 46 again resulted in good resolution of fine structure, but the subsurface layers were too thin for accurate interval velocity computation. Figure 48 summarizes the WAR velocity results on the time section. Layer thicknesses vary from .121 to .082 seconds for the upper unit which has an average interval velocity of 1.69 km/sec. The lower unit
Figure 48. Time section and interval velocities for SB 61 at the base of the Oregon continental slope.
of velocity 2.59 km/sec is 102 meters thick. The four thin layers of the upper unit approach the thickness limitation of the $I^2 - X^2$ method. The interval velocity for the first layer was 1.56 km/sec with thickness of 74 meters and is reasonably reliable due to a long reflection hyperbola and good resolution. It was necessary to consolidate the next three intervals to compute a more reliable average velocity of 1.72 km/sec giving a total thickness of 267 meters.

Figures 49 and 50 show the velocity-depth structure and frequency dependent spectral ratio plots. The error bars are large over most of the band but several features can be noted. Figure 50 indicates a strong spectral shaping effect for the first and third intervals with minima at 48 Hz and 68 Hz respectively. The other layers, within the limits of error, tend to show a monotonic increase of attenuation with frequency. Assuming an intrabed reflection coefficient of 0.1 and $\phi = 50\%$, equation (2.29) predicts unresolved average layer thicknesses of 19 and 16 meters respectively for these intervals. Attenuation computed according to equation (2.28) for the first interval gave .012 nepers/m at 195 Hz and a Q value of 32. This agrees with the data compiled by Attewell and Ramana (1966) for sedimentary rocks but is too high for natural, saturated silty clay sediments (Hamilton, 1974). Again this value is only an upper limit estimate based on the values given in Table 3. The velocity of the first interval suggests an average composition of clayey silt with a mean grain diameter of .005 mm, and the relatively high velocity for the second interval is consistent with the presence of an additional fraction of sand.
Figure 49. Velocity versus depth and average moduli of complex spectral ratios relative to arrival from sea floor for SB 61 near the base of the Oregon continental slope. Spectral ratios are not corrected for geometric spreading.

Figure 50. Spectral ratios relative to layer above for SB 61. Spectral ratios are not corrected for geometric spreading.
CHAPTER 4 DISCUSSION AND CONCLUSIONS

Discussion of the Reliability of Solution Vectors for Field Data

Evaluation of the reliability of the solution vectors is summarized in Tables 1 and 4 through 6. Sonobuoys #3 and #46 were given a complete analysis (Tables 4 and 5) by the three tests described in Chapter 2: 1) perturbation bounds for the solution vector, 2) F-test for significance of regression and 3) F-test for extra sum of squares. Sonobuoys #5, #57, and #61 were given the first two tests as summarized in Tables 1 and 6. Sonobuoy #46 was used as a test case to optimize the parameters of the SVA subroutine. Table 5 shows that as the number of solution components increases, the condition number and therefore, the perturbation bounds for the solution vector increases. The F-test for extra sum of squares favors the full model with 22 unknowns over the reduced model with 8 unknowns, however the full model has such a large condition number that it must be rejected in favor of the optimum model with 8 unknowns. In addition the F-test for significance of regression favors the reduced model. For SB 3 however, the full model (16 unknowns) has a small condition number and the F-test for extra sum of squares strongly favors the full model over the reduced model (8 unknowns). Therefore, the full model is retained even though the F-test for significance of regression slightly favors the reduced model. The models for SB 5, 57 and 61 were chosen on the basis of their small condition numbers (Table 6) and generally acceptable significance of regression except for a few
Table 4. Complete error analysis for sonobuoy 3.

Perturbation Bounds for the Solution Vector

<table>
<thead>
<tr>
<th>Number of Unknowns, p</th>
<th>8</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Equations, n</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>Condition number, κ</td>
<td>1.14</td>
<td>1.38</td>
</tr>
<tr>
<td>Frequency (Hz)</td>
<td>29</td>
<td>49</td>
</tr>
<tr>
<td>[\frac{|dW|}{|W|} &lt; \kappa \frac{s}{|Y|}]</td>
<td>.11</td>
<td>.11</td>
</tr>
</tbody>
</table>

F-Test for Significance of Regression

| Frequency (Hz) | 29 | 49 | 69 | 29 | 49 | 69 |
| F-ratio | 5.38 | 4.57 | 2.41 | 4.87 | 3.32 | 1.93 |
| Compare with | F_{8,72}(.95) = 2.1 | F_{16,64}(.95) = 1.8 |

F-Test for Extra Sum of Squares

| p | 18 (full model, includes mean); 16 (reduced model, excludes mean) |
| n | 80 |
| Frequency (Hz) | 29 | 49 | 69 |
| F-ratio | 1.01 | 2.47 | 1.26 |
| Compare with | F_{2,62}(.95) = 3.15 |

| p | 16 (full model); 8 (reduced model) |
| n | 80 |
| Frequency (Hz) | 29 | 49 | 69 |
| F-ratio | 12.45 | 6.80 | 5.42 |
| Compare with | F_{8,64}(.95) = 2.1 |
Table 5. Complete error analysis for sonobuoy 46.

<table>
<thead>
<tr>
<th>Perturbation Bounds for the Solution Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Unknowns, p</td>
</tr>
<tr>
<td>Number of Equations, n</td>
</tr>
<tr>
<td>Condition number, $\kappa$</td>
</tr>
<tr>
<td>Frequency (Hz)</td>
</tr>
<tr>
<td>$\frac{|dW|}{\kappa} \frac{|s|}{|W| |Y|}$</td>
</tr>
</tbody>
</table>

F-Test for Significance of Regression

| p | 8 | 22 |
| n | 80 | 80 |
| Frequency (Hz) | 39 | 59 | 117 | 39 | 59 | 117 |
| F-ratio | 19.85 | 1.55 | 15.22 | 14.67 | .78 | 9.27 |
| Compare with | $F_{8,72}(.95) = 2.1$ | $F_{22,58}(.95) = 1.73$ |

F-Test for Extra Sum of Squares

| p | 10 (full model, includes mean); 8 (reduced model, excludes mean) |
| n | 80 |
| Frequency (Hz) | 39 | 78 | 117 |
| F-ratio | 2.34 | .84 | .64 |
| Compare with | $F_{2,70}(.95) = 3.1$ |

| p | 22 (full model); 8 (reduced model) |
| n | 80 |
| Frequency (Hz) | 39 | 78 | 117 |
| F-ratio | 4.34 | 9.24 | 3.07 |
| Compare with | $F_{14,58}(.95) = 1.84$ |
Table 6. Perturbation Bounds for the Solution Vectors.

| Sonobuoy Number | p | n | κ | Frequency (Hz) | $\frac{||dw||}{||w||} < \frac{\kappa}{s}$ |
|-----------------|---|---|---|----------------|----------------------------------|
| 5               | 10| 80| 1.38| 39             | .12                              |
|                 |   |   |     | 78             | .11                              |
|                 |   |   |     | 117            | .14                              |
| 57              | 8 | 80| 1.41| 39             | .15                              |
|                 |   |   |     | 78             | .13                              |
|                 |   |   |     | 117            | .10                              |
| 61              | 10| 80| 1.29| 39             | .13                              |
|                 |   |   |     | 78             | .11                              |
|                 |   |   |     | 117            | .12                              |

fraction of Nyquist

Model #3 | 8 | 32 | 1.27 | .13 | .05 |
|         |   |   |      | .19 | .04 |
|         |   |   |      | .38 | .07 |
|         |   |   |      | .44 | .15 |
frequencies. High frequency solutions for SB 5 were not acceptable and this is reflected in the low F-ratio at 117 Hz. The solution vector at 39 Hz for SB 57 is also suspect for this trace.

The F-test for extra sum of squares, where the full model included the mean (includes a peak at data point zero in the time domain) and the reduced model excluded the mean, favored the reduced model in all cases as expected (Tables 4 and 5).

The SVA subroutine outputs p trial solution vectors, where p is the number of unknowns. A choice must be made as to which solution vector to choose. The final solution was chosen for all regressions on the basis of the fact that the residual norm and the square root of the normalized cumulative sums of squares were almost always a minimum, thereby suggesting a minimum perturbation bound, given small condition numbers.

The number of equations per regression was optimized by a trade off between condition number and bandwidth. Increasing the number of equations decreases $\kappa$ but increases the bandwidth per solution vector. Best results were found with 40 equations giving a bandwidth of 20 Hertz per regression.

Additional tests were made to determine the effect of changing the number of solution components in the frequency domain regression. For example, increasing the number of peaks from 4 to 8 changed the 4 solution component values by an average of only 18% for a trace of SB 3. This tendency for solution vector components to be relatively insensitive to reduction in the number of unknowns was also found for model data.
Equation (2.28) indicates that it should not be necessary to divide the signal spectrum by the source spectrum prior to regression. A test of regression with and without division by the source spectrum indicated that the spectral ratios were essentially the same.

Finally, there will always be statistical errors in equalization by computing spectral ratios of solution vector components, especially in regions of the spectrum where the power is varying rapidly with frequency. Averaging redundant traces will smooth random oscillation, but will enhance coherent noise. The 20 Hertz bandwidth in each regression includes the complicated spectra of multiples in that band. Averaging may enhance rather than attenuate this noise, so that apparent attenuation due to intrabed multiples will affect the computed attenuation in that band and may not be equalized by taking spectral ratios since multiple interference is frequency and time or depth dependent. This may explain in part the oscillation in the spectral ratio plots. Further research on this problem is necessary.

Conclusions

Source signature deconvolution of marine seismic reflection data has been shown to be more effective for the resolution of fine structure if the source signature can be measured by stacking the signal reflected from a reasonably homogeneous bottom layer rather than stacking the near field direct wave. Inverse ghost filtering is then not necessary. Acceptable resolution also required judicious bandpass and notch filtering. Resolution was increased beyond the
capacity of the $T^2 - X^2$ method for accurate interval velocity computation. Reported interval velocities therefore generally included other discernable layers.

Model studies on the proposed least squares regression technique for computation of frequency dependent spectral ratios were successful in extracting the input amplitude attenuation functions. It is significant also that the model data suggests that multiple and primary events not included in the regression may be considered a part of the noise term without affecting the accuracy of the spectral ratios of the chosen primary events. Uncompensated phase distortion can affect the solution accuracy but not seriously if spectral ratios are taken relative to the arrival from the sea floor. The contribution of dispersion to phase distortion is negligible over the short paths considered in the upper sedimentary layers. Results with field data point out the need to average computed spectral ratios of many redundant traces. Comparison of F-values for significance of regression on model data and field data suggest that the solution vectors for field data are in most cases reliable. This F-test must be regarded with caution since the errors are not normally distributed, but on the basis of model studies, relative F-values are a good measure of reliability. Accuracy is not adequate to judge material type in most cases on the basis of attenuation but there are some encouraging results. The computed surface layer interval velocity and attenuation for sonobuoy #5 are in good agreement with the average material type found in the DSDP core at site 174 (sandy-silt with greater than 60% sand). The minimum Q value for the surface layer was 308 at 80 Hz.
for SB 5 and 80 at 127 Hz for SB 46. The maximum attenuation in the upper interval for SB 46 over the Tufts abyssal plain where fine-grained material (silts and clays) is expected was greater than the attenuation computed for the sandy-silt environment of the Astoria sea fan. This suggests that the cross-over frequency predicted by Stoll (1977) (figure 4) for attenuation in clays and sands is well above 130 Hz. Some maximum attenuation values for the surface layers of the study areas were high, suggesting the possibility of stratigraphic components. Spectral shaping occurred for some intervals and not for others, and gross estimates were made of the average non-resolvable layer thicknesses present in the intervals, assuming that short path multiples were responsible for the effect.

The proposed method for measuring attenuation from seismic reflection data appears feasible and warrants further study under more controlled conditions. Data should be taken in a different way for its successful application. Redundant traces should be stacked at a stationary source-receiver position arranged for near normal incidence to the subsurface strata. The fixed source-receiver geometry would then laterally scan the area of investigation. Accurate source signature measurements are critically important to its success. Lateral variations in attenuation characteristics of subsurface strata would become another variable to aid in judging subsurface structure and lithology. The measurement of frequency dependent amplitude information from seismic reflection data requires that deterministic methods be used as much as possible in the data processing leaving statistical averaging for the final step to improve the accuracy of spectral ratios.
BIBLIOGRAPHY


APPENDICES
APPENDIX 1

Formulas for Interval Velocity Computations with a Plane Dipping Layer Model (after LePichon et al., 1968)

The basic equation for the travel times of acoustic pulses in a single plane dipping layer is:

\[ T^2 = T_0^2 + \frac{x^2}{V^2} - \frac{2Tx}{V} \sin \omega \quad (A.1.1) \]

where \( \omega \) is negative if the bottom reflecting boundary is dipping downward in the direction from the origin to the shotpoint and \( V \) is the vertical water column velocity. Let \( VH \) be the horizontal surface velocity of sea water, then the distance, \( X = VH \cdot D \), where \( D \) is the direct surface acoustic wave travel time.

1. Compute horizontal surface water velocity and its standard deviation, the two way travel time, \( T_0 \), at the sonobuoy location (origin) and the water depth at the origin.
   a. Compute \( T_0 \) from a 4th order fit to the D-time, travel-time pairs. If the \( T/x \) curve starts at \( x = 0 \) or \( D = 0 \):
      \( T(i) = a_0 + a_1 D(i) + a_2 D(i)^2 + a_3 D(i)^3 + a_4 D(i)^4 \),
      or \( T_0 = a_0 \).
   b. Otherwise, if \( D(1) \neq 0 \), \( T_0 \) is found from a linear least squares fit of:
      \[ T_i^2 = T_0^2 + a_1 D_i^2 \]
with \( a_1 = \frac{\sum_{i=1}^{n} T_i^2 D_i - \sum_{i=1}^{n} D_i^2 \sum_{i=1}^{n} T_i}{\sum_{i=1}^{n} D_i^4 - \sum_{i=1}^{n} D_i^4 \sum_{i=1}^{n} D_i^2} \) \( \text{(A.1.2)} \)

\[ T_0 = \sqrt{\frac{\sum_{i=1}^{n} T_i^2 - a_1 \sum_{i=1}^{n} D_i^2}{n}} \] \( \text{(A.1.3)} \)

Carnahan et al. (1969)

The reason for the above two alternatives is, according to LePichon et al. (1968), that data obtained near the origin is usually not very accurate. It is therefore not safe to extrapolate the least squares fitted 4th order polynomial to the origin. Instead, it is best to obtain \( T_0 \) by a linear least squares fit to the \( T^2/D^2 \) or \( T^2/X^2 \) data, resulting in only a slight inaccuracy.

c. Compute for each \( T_i \) a correction term to reduce the travel times to a zero dip case. Equation (A.1.1) may be rewritten:

\[ T^2 = T_0^2 + \frac{D^2 \cdot VH^2}{V^2} - 2T \left( \frac{VH}{V} \right) D \sin \omega \]

or \( T^2 - T_{\text{CORR}} = D^2 \cdot \frac{VH^2}{V^2} = T_i^2 \)

where \( T_{\text{CORR}} = T_0^2 - 2TD \sin \omega \) and \( \frac{VH}{V} \approx 1 \) is neglected.

d. By linear least squares, fit \( T_i^2 = D_i^2 \left( \frac{VH}{V} \right)^2 \) according to the formulas (A.1.2) and (A.1.3).

Calculate, \( VH = V \sqrt{a_1} \)
e. Compute the standard deviation of the horizontal surface water velocity, $\sigma_{VH}$.

In practice there will be some finite intercept to the linear least squares fit:

$$T_{i}^{2} = a_{0} + a_{1} D_{i}^{2}$$

with

$$a_{0} = \frac{\sum_{i=1}^{n} T_{i}^{2} - a_{1} \sum_{i=1}^{n} D_{i}^{2}}{n}.$$

Compute the standard deviation of $T_{i}^{2}$ about the fitted line:

$$S_{T}^{2} = \sqrt{\frac{\sum_{i=1}^{n} (\delta T_{i}^{2})^{2}}{n-2}}$$

where

$$\delta T_{i}^{2} = T_{i}^{2} - (a_{0} + a_{1} D_{i}^{2}).$$

The standard deviation of the slope, $a_{1}$, is:

$$S_{a_{1}} = S_{T}^{2} \sqrt{\frac{n}{n \sum_{i=1}^{n} D_{i}^{4} - (\sum_{i=1}^{n} D_{i}^{2})^{2}}}$$

and

$$\sigma_{VH} = \frac{VH}{2} \left( \frac{S_{a_{1}}}{a_{1}} \right)$$

(Beers, 1953; Whittle and Yarwood, 1973).

f. Compute depth at origin, $h_{0}$:

$$h_{0} = \frac{VT_{0}}{2}$$
2. Compute interval velocities in subbottom layers, their standard deviations, round trip travel time at origin $T_{0i}$, corrected dip, layer thickness, and depth to bottom of layer by calling subroutine INTVEL.

**Formula derivations for algorithms in subroutine INTVEL using a 4-layer case as an example**

The acoustic ray paths must satisfy reciprocity, that is, the travel time at any given distance will be the same regardless of the direction of travel. In other words, the shotpoint and receiver can be interchanged. We think of the ray as traveling from the origin to the receiver even though the sonobuoy is actually at the origin. The moving point is considered that of the emergent ray, see figure A.1.1.

For the 4 layer case then, the angle of incidence of the emergent ray to the water surface is given by:

$$\sin i_1 = \frac{(dT)}{(dx)}_4$$

Since the $T/x$ data has been fitted to a 4th order polynomial,

$$T = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$$

$$\frac{dT}{dx} = a_1 + 2a_2x + 3a_3x^2 + 4a_4x^3$$

The ray trace for each shot is as follows:

$$r_1 = i_1 + \omega_1$$

$$r_1 = \sin^{-1} (V_1 (\frac{dT}{dx})_4) + \omega_1$$
\[ r_2 = i_2 + \omega_2 \quad \frac{\sin r_1}{V_1} = \frac{\sin i_2}{V_2} \quad \text{(Snell's Law)} \]
\[ \therefore r_2 = \sin^{-1}\left(\frac{V_1}{V_2} \sin r_1\right) + \omega_2 , \]

similarly,
\[ r_3 = \sin^{-1}\left(\frac{V_3}{V_2} \sin r_2\right) + \omega_3 \]
and \[ r_4 = \sin^{-1}\left(\frac{V_4}{V_3} \sin r_3\right) + \omega_4 \]

\[ i_3' = \sin^{-1}\left(\frac{V_3}{V_4} \sin (r_4 + \omega_4)\right) \]
\[ i_2' = \sin^{-1}\left(\frac{V_2}{V_3} \sin (i_3' + \omega_3)\right) \]
\[ i_1' = \sin^{-1}\left(\frac{V_1}{V_2} \sin (i_2' + \omega_2)\right) \]

\[ Z_{04} = \frac{T_{04} - T_{03}}{2} V_4 , \text{using trial values for } V_4 \]

\[ \theta_1 = \sin^{-1}\left(\frac{V_3}{V_4} \sin \omega_4\right), \quad \theta_2 = \sin^{-1}\left(\frac{V_2}{V_3} \sin (\theta_1 + \omega_3)\right) \]
\[ \theta_3 = \sin^{-1}\left(\frac{V_1}{V_2} \sin (\theta_2 + \omega_2)\right) . \]

The distances \( \Delta x_i \) are measured from the points of intersection with the layers of the ray reflecting perpendicularly from the lower interface of layer 4. This ray has the round trip travel time \( T_0 \).

\[ \Delta x_1 = Z_{01} (\tan i_1' - \tan \theta_3) , \quad BB' = Z_{02} - \Delta x_1 \sin \omega_2 = YYP(2,1) \quad \text{(in program)} \]
\[ \Delta x_2 = \Delta x_1 \cos \omega_2 - Z_{02} \tan \theta_2 , \quad \Delta Z_3 = \Delta x_2 \sin \omega_3 \]
\[ \Delta x_3 = \Delta x_2 \cos \omega_3 - Z_0 \tan \theta_1 + (BB') \tan i_2 \cos \omega_3 \]

\[ \Delta Z_4 = \Delta x_3 \sin \omega_4 \]

\[ CC' = Z_0 - (BB') \tan i_2 \sin \omega_3 - \Delta Z_3 = YYP(3, I) \text{ in program.} \]

\[ DD' = Z_0 - (CC') \tan i_3 \sin \omega_4 - \Delta Z_4 = YYP(4, I) \text{ in program.} \]

The travel time segments near the origin are:

\[ T_{AB} = \frac{Z_0}{V_1 \cos i_1}, \quad T_{BC} = \frac{BB'}{V_2 \cos i_2}, \quad T_{CD} = \frac{CC'}{V_3 \cos i_3}. \]

The layer thicknesses and travel time segments on the moving point side are:

\[ Z_0' = Z_0 - x \sin \omega_1, \quad T_{IH} = \frac{Z_0'}{V_1 \cos r_1} \]

where \( x \) is the distance from the origin to the moving point for this shot measured along the sea surface.

The reduced distance along the upper interface of layer 2 is:

\[ x_{r_2} = x \cos \omega_1 - Z_0 \tan i_1 - Z_0' \tan r_1. \]

Continuing,

\[ Z_0' = BB' - x_{r_2} \sin \omega_2, \quad T_{HG} = \frac{Z_0'}{V_2 \cos r_2} \]

\[ x_{R_3} = x_{R_2} \cos \omega_2 - BB' \tan i_2 - Z_0' \tan r_2. \]

\[ Z_0' = CC' - x_{R_3} \sin \omega_3, \quad T_{GF} = \frac{Z_0'}{V_3 \cos r_3} \]
\[ x_{R4} = x_{R3} \cos \omega_3 - CC' \tan i'_3 - z'_3 \tan r_3 \]

\[ T_{R4} = T - T_{AB} - T_{BC} - T_{CD} - T_{GF} - T_{HG} - T_{IH} \]

The round trip time of the ray reflecting perpendicularly from the lower interface of layer 4 is:

\[ T_{DD'} = \frac{2(DD')}{V_4} \]

The correction term to reduce the travel time to the zero dip case is:

\[ \text{CORR}_4 = T_{DD'}^2 - \frac{2 T_{DD'} x_{R4}}{V_4} \sin \omega_4 \]

The reduced time squared is:

\[ T_{R4}^2 = T_{R4}^2 - \text{CORR}_4 \]

A linear least squares fit is applied to the equation,

\[ T_{R4}^2(i) = \left( \frac{1}{V_4^2} \right) x_{R4}^2(i) \]

for all shots in the time-distance arrays to obtain the slope, \( a_1 = \frac{1}{V_4^2} \).

The least squares best slope is:

\[ a_1 = \frac{\sum_{i=1}^{N_5} x_{R}^2(i) T_R^2(i) - \sum_{i=1}^{N_5} x_{R}^2(i) \sum_{i=1}^{N_5} T_R^2(i)}{N_5 \sum_{i=1}^{N_5} x_{R}^2(i) - (\sum_{i=1}^{N_5} x_{R}^2(i))^2} \]
The trial velocity for the next iteration is:

\[ V'_{4} = \sqrt{\frac{1}{a_{1}}} \]

The dip correction is derived as follows:

![Dip Correction Geometry](image)

\[ \tan \omega = \frac{y}{x}, \quad y = t_{y}v, \quad v = \text{old velocity} \]

also \( \tan \omega' = \frac{y'}{x'}, \quad y' = t_{y}v', \quad v' = \text{new velocity} \)

if \( x' = x \)

\[ \frac{\tan \omega}{y} = \frac{\tan \omega'}{y'}, \quad \tan \omega' = \frac{y'}{y} \tan \omega = \frac{v'}{v} \tan \omega. \]

The program now checks for convergence by comparing the differences in successive velocities for each iteration. The difference must be decreasing. If the difference is greater than 0.001 the program resets the old trial velocity to the new one and the ray trace begins again.

When the difference is less than 0.001 the iterations are complete and the standard deviation of the final velocity is computed according to the formulas in 1(e) above.
Figure A.1.1 Ray Tracing for Formulas in Subroutine INTVEL
APPENDIX 2

Derivation of Normal Equations for a Least Squares 4th Order Polynomial Fit to Data

Fit the polynomial,

\[ y = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 \]

to a set of \( n \) pairs \((x_i, y_i)\) so that the sum of squares of the deviations of the data pairs about the fitted line is a minimum.

\[ SS = \text{sum of squares} = \sum_{i=1}^{n} (y_i - y)^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1x_i - a_2x_i^2 - a_3x_i^3 - a_4x_i^4)^2 \]

Set \[ \frac{\partial SS}{\partial a_0} = \frac{\partial SS}{\partial a_1} = \frac{\partial SS}{\partial a_2} = \frac{\partial SS}{\partial a_3} = \frac{\partial SS}{\partial a_4} = 0 \]

For example:

\[ \frac{\partial SS}{\partial a_0} = 2 \sum_{i=1}^{n} (y_i - a_0 - a_1x_i - a_2x_i^2 - a_3x_i^3 - a_4x_i^4)(-1) = 0 \]

\[ na_0 + a_1 \sum_{i=1}^{n} x_i + a_2 \sum_{i=1}^{n} x_i^2 + a_3 \sum_{i=1}^{n} x_i^3 + a_4 \sum_{i=1}^{n} x_i^4 = \sum_{i=1}^{n} y_i \]

\[ \frac{\partial SS}{\partial a_1} = 2 \sum_{i=1}^{n} (y_i - a_0 - a_1x_i - a_2x_i^2 - a_3x_i^3 - a_4x_i^4)(-x_i) = 0 \]

\[ a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 + a_2 \sum_{i=1}^{n} x_i^3 + a_3 \sum_{i=1}^{n} x_i^4 + a_4 \sum_{i=1}^{n} x_i^5 = \]

\[ \sum_{i=1}^{n} x_i y_i. \]
We thus obtain a system of 5 simultaneous equations in the 5 unknowns, \( a_i \):

\[
\begin{align*}
na_0 + a_1 \Sigma x_i + a_2 \Sigma x_i^2 + a_3 \Sigma x_i^3 + a_4 \Sigma x_i^4 &= \Sigma y_i \\
(a_0 \Sigma x_i + a_1 \Sigma x_i^2 + a_2 \Sigma x_i^3 + a_3 \Sigma x_i^4 + a_4 \Sigma x_i^5 &= \Sigma x_i y_i \\
(a_0 \Sigma x_i^2 + a_1 \Sigma x_i^3 + a_2 \Sigma x_i^4 + a_3 \Sigma x_i^5 + a_4 \Sigma x_i^6 &= \Sigma x_i^2 y_i \\
(a_0 \Sigma x_i^3 + a_1 \Sigma x_i^4 + a_2 \Sigma x_i^5 + a_3 \Sigma x_i^6 + a_4 \Sigma x_i^7 &= \Sigma x_i^3 y_i \\
(a_0 \Sigma x_i^4 + a_1 \Sigma x_i^5 + a_2 \Sigma x_i^6 + a_3 \Sigma x_i^7 + a_4 \Sigma x_i^8 &= \Sigma x_i^4 y_i \\
\end{align*}
\]

In matrix form:

\[
\begin{bmatrix}
  b_{11} & b_{12} & b_{13} & b_{14} & b_{15} \\
  b_{21} & b_{22} & b_{23} & b_{24} & b_{25} \\
  b_{31} & b_{32} & b_{33} & b_{34} & b_{35} \\
  b_{41} & b_{42} & b_{43} & b_{44} & b_{45} \\
  b_{51} & b_{52} & b_{53} & b_{54} & b_{55} \\
\end{bmatrix}
\begin{bmatrix}
  a_0 \\
  a_1 \\
  a_2 \\
  a_3 \\
  a_4 \\
\end{bmatrix}
=
\begin{bmatrix}
  Z_1 \\
  Z_2 \\
  Z_3 \\
  Z_4 \\
  Z_5 \\
\end{bmatrix}
\]

or \( \bar{B}\bar{a} = Z \)

The vector \( \bar{a} \) is then solved by Gauss-Jordan elimination with single pivot.
APPENDIX 3

Bandpass and Notch Filters

The low pass, high pass and notch filters are windows with cosine tapers in the frequency domain, symmetric about the Nyquist frequency. See programs for usage of labeled data points. These filters are zero phase since the imaginary part is zero.

1. Low Pass Filter (Subroutine LPFKW)

The window is constructed as in the diagram shown below:

Cosine taper according to:

\[
X_n = \frac{1 + \cos \left( \frac{\pi j}{L} \right)}{2}, \quad j = 0, 1, \ldots L.
\]

Figure A.3.1. Frequency domain window for low pass filter; LQ is low pass cutoff and LA is length of cosine taper.

The sampling frequency, \( \omega_s = \frac{1}{\Delta t} \) cycles/sec, where \( \Delta t \) is the digitizing interval. The frequency interval then is, \( \Delta f = \omega_s/N = \frac{1}{N \Delta t} \).
or \( \Delta f = \frac{1}{T} \), where \( T \) is the record length in the time domain.

2. High Pass Filter (Subroutine HPFKW)

Again a window is constructed with the same cosine taper.

![Diagram of high pass filter frequency domain window]

Figure A.3.2. Frequency domain window for high pass filter; LH is high pass cutoff.

3. Notch Filter (Subroutine SNOTCH)

![Diagram of notch filter frequency domain window]

Figure A.3.3. Frequency domain window for notch filter.
APPENDIX 4

Deghost Filter

1. Subroutine VGHOST

Deghost filter for seismic reflection data at near vertical incidence. The desired filter in the frequency domain is:

\[ F(\omega) = \frac{(1-e^{-i\omega t_1})}{(1-e^{-i\omega t_2})(1-e^{-i\omega t_3})} \]  \hspace{1cm} (A.4.1)

where: \( t_1 \) = Travel time difference between reflected and direct wave at streamer.
\( t_2 \) = ghost time delay at airgun for downgoing ray
\( t_3 \) = ghost time delay at streamer for upcoming ray.

The complex factor \( \frac{1}{1-e^{-i\omega t}} \) has a pole at \( \omega t = 2\pi n \). At these angles the cosine is +1 but the sin is zero and changing sign from - to +. The program truncates the sign at a predetermined level as shown:

\[ e^{i\theta} = \cos \theta - i \sin \theta \]

Figure A.4.1. Truncation of imaginary part of inverse ghost filter.

The modulus of the inverse ghost filter, \( G(\omega) = \frac{1}{1-e^{-i\omega t}} \) is plotted in figure A.4.2. At truncation points, the filter is given
the value,

\[ F'(\omega) = \text{CMPLX} \{ \text{RE}(G_S(\omega)), \text{SIGN}[\text{Im}(G(\omega))] \times \text{Im}(G_S(\omega)) \}. \]

Figure A.4.2 Amplitude spectrum of truncated inverse ghost filter.

The filter factor \((1-e^{-i\omega T})\) has an amplitude spectrum shown in figure A.4.3. It requires no truncation.

Figure A.4.3. Amplitude spectrum of forward ghost filter, \((1-e^{-i\omega T})\).
Multiplication is done up to Nyquist with all filter factors. Folding about Nyquist is the final step. Since this filter is not zero phase as is the case with the bandpass filters, folding must be done by reversing the sign of the imaginary part at the symmetrical folding frequency. Thus the filter is Hermitian and the inverse Fourier transform will then be real with a very small or zero imaginary part.
APPENDIX 5

Trigonometry of Ghost Delay Calculations

Compute the ghost delay times $\tau_1$, $\tau_2$, and $\tau_3$ given the depths and separation of the source and receiver (Figure A.5.1.)

![Source-receiver geometry and ray paths for ghosts and direct waves.](image)

$V_w =$ water acoustic velocity

From figure A.5.1.:

$$\cos \theta_s = \frac{h_s}{a}, \quad \cos 2\theta_s = \frac{b}{a},$$

so the source ghost delay is,

$$\tau_2 = \frac{a+b}{V_w} = \frac{h_s}{\cos \theta_s} + \frac{h_s}{\cos \theta_s} \cos 2\theta_s = \frac{h_s}{V_w \cos \theta_s} (1 + \cos 2\theta_s). \quad (A.5.1)$$

The ghost delay at the sonobuoy, $\tau_3$ is the same except substitute the angle of incidence of the ray with the normal to the surface at the receiver, $\theta_r$, and the depth $h_{son}$:

$$\tau_3 = \frac{h_{son}}{V_w \cos \theta_r} (1 + \cos 2\theta_r), \quad (A.5.2)$$
The ghost time delay at the streamer, $\tau_1$, is:

$$\tau_1 = \frac{\sqrt{x^2 + (h_s + h_r)^2} - \sqrt{x^2 + (h_s - h_r)^2}}{v_w}$$  \hspace{1cm} (A.5.3)$$

Derivation of the angle of incidence of the rays at the source and receiver.

Figure A.5.2 shows the geometry for the calculation of $\theta_s$ and $\theta_r$ for a plane dipping sea floor.

The angle $\omega$ is negative if the dip is down from the source to the receiver. Referring to the figure:

$$i = \theta_s - \omega, \quad r = \theta_r + \omega,$$

$$i = r \quad \text{or} \quad \theta_s - \omega = \theta_r + \omega$$

so $\theta_s = \theta_r + 2\omega$

and $\theta_r = \theta_s - 2\omega$  \hspace{1cm} (A.5.4)
The angle $\theta_S$ is derived from:

$$\begin{align*}
\tan i &= \frac{a}{Z_S}, \quad \tan r = \frac{b}{Z_R}, \\
\cos \omega &= \frac{a+b}{x} = \frac{Z_S \tan(\theta_S - \omega) + Z_R \tan(\theta_R + \omega)}{x}, \\
\end{align*}$$

but

$$Z_R = Z_S + x \sin \omega \quad \text{and} \quad \theta_R = \theta_S - 2\omega$$

therefore,

$$\cos \omega = \frac{Z_S [\tan(\theta_S - \omega) + \tan(\theta_S - \omega)] + x \sin \omega \tan(\theta_S - \omega)}{x}.$$ 

Simplifying,

$$\cos \omega = \frac{\tan(\theta_S - \omega) [2Z_S + x \sin \omega]}{x},$$

$$\tan(\theta_S - \omega) = \frac{x \cos \omega}{2Z_S + x \sin \omega},$$

or

$$\theta_S = \tan^{-1} \left[ \frac{x \cos \omega}{2Z_S + x \sin \omega} \right] + \omega \quad (A.5.5)$$

Using equations (A.5.1) through (A.5.5) all ghost time delays can be calculated from depths of the airgun, sonobuoy and streamer, the airgun-streamer separation and the bottom dip. These formulas are used in the programs STKDCN and WARDCN.
A reflected waveform, \( y(T) \), with frequency independent phase change, \( \varepsilon \), can be represented in terms of the inverse Fourier transform of the original waveform, \( f(t) \), as:

\[
y(T) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\tau) e^{-i[\omega(T-\tau)+\varepsilon]} \, d\tau.
\]

Officer (1958, p. 104-106) gives the solution for the phase shifted function, \( y(T) \), in terms of \( f(T) \) as:

\[
y(T) = \cos \varepsilon f(T) + \sin \varepsilon F(T)
\]

where, \( F(T) = \frac{1}{\pi} \int q(\omega) \cos(\omega T) d\omega - \frac{1}{\pi} \int p(\omega) \sin(\omega T) d\omega \), \( (A.6.1) \)

\[
q(\omega) = \int_{-\infty}^{\infty} f(\tau) \sin(\omega \tau) \, d\tau = \text{sin transform}
\]

and \( p(\omega) = \int_{-\infty}^{\infty} f(\tau) \cos(\omega \tau) \, d\tau = \text{cosine transform} \).

**Algorithm for synthetic seismogram of frequency independent phase shifted wavelets**

Program WVKW8 computes a synthetic seismogram of an input arbitrary source wavelet with a number of echoes of any desired phase change and reflection coefficient. Subroutines STFKW and CTFKW are
called from subroutine SCTKW to compute the function \( F(T) \) in equation (A.6.1). Subroutine STFKW\((N,SGNI)\) computes the sine transform, \( q(\omega) \). For discrete functions,

\[
q_j = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} f_n \sin \frac{2 \pi (j-1)(n-1)}{N} \quad \text{if } SGNI = 1.0 \quad (A.6.2)
\]

where \( N \) must be even.

On the second call, the integral

\[
\frac{1}{\pi} \int_{0}^{\infty} p(\omega) \sin(\omega T) d\omega
\]

becomes,

\[
q_j = \frac{1}{\sqrt{N}} \left\{ \frac{(N/2+1)-1}{2} \sum_{n=2}^{N} f_n \sin \frac{2 \pi (j-1)(n-1)}{N} + f_1 \cos(\omega) + f_{n/2+1} \sin \frac{2 \pi (j-1)N/w}{N} \right\} \quad \text{if } SGNI = 2.0 \quad (A.6.4).
\]

The integer \( j \) goes from 1 to \( N/2 + 1 \), the Nyquist frequency, and then folding is done. Since the sine transform is odd, the folded half is multiplied by -1.

Subroutine CTFKW\((N,SGNI)\) computes the cosine transform, \( p(\omega) \) and is identical to equations (A.6.2) to (A.6.4) except that cosine is exchanged for sine and since the cosine transform is even, folding is done without multiplying by -1.
Subroutine SCTKW(N) computes the function $F(T)$ and also rearranges the output of the sine and cosine transforms so that the folded portion corresponding to negative time after the second call, appears to the left of zero.

Program WVKW8 calls subroutine SCTKW once at the beginning to compute the function $F(T)$, then computes the phase shifted waveform according to equation (A.6.1) for each echo, multiplies each by the desired reflection coefficient and superimposes them at the appropriate echo delay. This is equivalent to convolution with a reflection coefficient series.

Analytical Function for the Impulse Response of the Frequency
Independent Phase Shifted Wavelet after Source Signature Deconvolution

From equation (2.1) the reflection seismogram is represented as the convolution of a source function $s_t$ with a reflection coefficient series, $r_t$:

$$x_t = s_t * r_t + n_t = \sum_{N-ti=1}^{N-t} s_{t-i} * r_{t-i} + n$$

The wavelet $s_t$ is now replaced by its phase shifted counterpart, $s'_t$:

$$s'_t = (\cos \epsilon)s_t + (\sin \epsilon)F_t$$

The function $F_t$ can be identified as the convolution of $s_t$ with a 90° phase shift or quadrature filter, $q_t$, since at $\epsilon = 90°$:

$$s'_t = F_t = s_t * q_t.$$
Neglecting the noise term and the frequency dependence of \( a_{j,n} \) (\( n \) are frequency data points) equation (2.9) becomes upon substituting \( S'_{t,1} \) for \( S_t \), where the phase changes are pulse delay dependent \((\epsilon \Rightarrow \epsilon_j)\) and consider only one echo or spike:

\[
x_t = S'_{t,j} * a_j \delta(t-j\Delta t).
\]

The Fourier transform of \( S'_t \) is:

\[
S'(\omega) = S(\omega) \cos \epsilon_j + S(\omega) Q(\omega) \sin \epsilon_j
\]

The Fourier transform of \( x_t \) is then modified from (2.10):

\[
X(\omega) = S(\omega) a_j (\cos \epsilon_j + Q(\omega) \sin \epsilon_j) e^{-i\omega t_j}.
\]

The lowest possible frequency for a digitized record of length, \( T \), is \( 2\pi/N\Delta t \), and the highest possible frequency, the sampling frequency is:

\[
\frac{2\pi N}{N\Delta t} = \frac{2\pi}{\Delta t}
\]

Let \( t = 1 \) time unit, then the full frequency range is 0 to \( 2\pi \) radians/sec or \( -\pi \) to \( \pi \). After source signature deconvolution, \( X(\omega) \) becomes:

\[
X'(\omega) = a_j \cos \epsilon_j e^{-i\epsilon_j} + a_j \sin \epsilon_j Q(\omega) e^{-i\omega t}
\]
Taking the inverse Fourier transform:

\[ x'_t = a_j \cos \epsilon_j \delta(t-j) + \frac{1}{2\pi} \int_{-\pi}^{\pi} (a_j \sin \epsilon_j Q(\omega)e^{-i\omega t})e^{i\omega \tau} d\omega. \]

If we let \( \tau = k \Delta t = k \), we have:

\[ x'_t = a_j \cos \epsilon_j \delta(t-j) + \frac{1}{2\pi} a_j \sin \epsilon_j \int_{-\pi}^{\pi} Q(\omega)e^{i\omega(k-j)} d\omega. \]

(A.6.5)

Now \( Q(\omega) \) may be represented as a weighted function of frequency:

\[ Q(\omega) = \frac{i\omega}{|\omega|} \quad \text{(Claerbout, 1976),} \]

which is plotted in figure A.6.1.

Figure A.6.1. Amplitude spectrum of the quadrature filter.

This may be seen by realizing that differentiation produces a phase shift at 90°, i.e.:

\[ f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega \tau} d\omega, \]
\[
\frac{df}{dt} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \omega F(\omega) e^{i\omega t} d\omega = \text{IFT}(Q(\omega)F(\omega))
\]

and where the correct \(Q(\omega)\) must be weighted by dividing by \(|\omega|\).

Evaluating the integral in equation (A.6.5):

\[
\frac{\sin c_{j}}{2\pi} a_{j} \left[ \int_{-\pi}^{\pi} (-i) e^{i\omega(k-j)} d\omega + \int_{0}^{\pi} ie^{i\omega(k-j)} d\omega \right]
\]

\[
= \frac{\sin c_{j}}{2\pi} a_{j} \left[ (-i) \sum_{k-j}^{0} e^{i\omega(k-j)} - \sum_{k-j}^{\pi} ie^{i\omega(k-j)} \right]
\]

\[
= \frac{\sin c_{j}}{2\pi} a_{j} \left[ -\frac{1}{(k-j)} + e^{-i\pi(k-j)} + e^{i\pi(k-j)} - \frac{1}{(k-j)} \right]
\]

\[
\frac{a_{j} \sin c_{j}}{2\pi(k-j)} \left[ e^{i\pi(k-j)} + e^{-i\pi(k-j)} - 2 \right] = \begin{cases} 
-\frac{2a_{j} \sin c_{j}}{\pi(k-j)} & (k-j)\text{odd} \\
0 & (k-j)\text{even}
\end{cases}
\]

\[
= (a_{j} \sin c_{j}) q(k-j) \quad \text{where } q(k-j) \text{ is the impulse response of the quadrature filter or Hilbert transformation centered at } \tau_{j}.
\]

The full representation is:

\[
x'_{t} = a_{j}[\delta(k-j)\cos \epsilon_{j} + q(k-j)\sin \epsilon_{j}].
\]

The impulse response of \(q_{n}\) is shown in figure A.6.2.
Figure A.6.2. Impulse response of the quadrature filter.

At delay $\tau_j$ is centered the time domain representation of the Hilbert transform scaled by $\sin \varepsilon_j$, superimposed on the delta function at $\tau_j$ scaled by $\cos \varepsilon_j$. 

$$q_n = \begin{cases} -2/\pi n & \text{n odd} \\ 0 & \text{n even} \end{cases}$$
APPENDIX 7

Derivation of the Normal Equations for Least Squares Regression
of a Linear Model with Complex Numbers.

Real Variables

We wish to solve the linear system of equations:

\[ Y = X \beta + \epsilon \]  \hspace{1cm} (A.7.1)

for the \((p \times 1)\) vector \(\beta\) where \(Y\) is an \((n \times 1)\) vector of observations,
\(X\) is an \((n \times p)\) vector of observations,
\(\epsilon\) is an \((n \times 1)\) vector of errors,
and where \(E(\epsilon) = 0, V(\epsilon) = I\sigma^2\), so the elements of \(\epsilon\) are uncorrelated.
The operator \(E\) is the expected value or mean, the operator \(V\) is the variance, \(I\) is the \((n \times n)\) identity matrix and \(\sigma^2\) is the variance of \(\epsilon\).

Since \(E(\epsilon) = 0\), the model may also be written,

\[ E(Y) = X \beta \]

The error sum of squares is:

\[ \epsilon^T \epsilon = (Y - X \beta)^T (Y - X \beta) \]
\[ = Y^T Y - X^T \beta^T Y - Y^T X \beta + X^T \beta^T X \beta \]
\[ = Y^T Y - 2 \beta^T X^T Y + \beta^T X X \beta. \]  \hspace{1cm} (A.7.2)

where \(T\) represents the transpose.
This follows since $\beta^T \chi T = \chi^T \beta^T$ and $\beta^T \chi T \psi$ is a scalar whose transpose $(\beta^T \chi T \psi)^T = \psi^T \chi T \beta$ must have the same value. Differentiating (A.7.2) with respect to the elements of $\beta$ and setting equal to zero we find that the best estimate $b$ of the solution vector $\beta$ which minimizes the error sum of squares irrespective of any distribution properties of the errors is given by the normal equations:

$$(X^T X)b = X^T Y$$

(Draper & Smith, 1966) (A.7.3)

Complex Variables

Consider the linear model with complex variables:

$$Y = AW + \Xi$$

where

- $Y$ is an $(n \times 1)$ vector of complex observation parameters.
- $A$ is an $(n \times p)$ matrix with complex elements.
- $W$ is a $(p \times 1)$ vector of complex solution vector components. Prove that the same normal equations will result whether we solve for them in the form given by (A.7.4) or in the form given by (2-30).

The error sum of squares for (A.7.4) is:

$$\Xi^* \Xi = (\Xi_1 + i\Xi_2)^* (\Xi_1 + i\Xi_2) = (\Xi_1 - i\Xi_2)^T (\Xi_1 + i\Xi_2)$$

$$= \Xi_1^T \Xi_1 - i\Xi_2^T \Xi_1 + i\Xi_1^T \Xi_2 + \Xi_2^T \Xi_2$$

$$= \Xi_1^T \Xi_1 + \Xi_2^T \Xi_2$$

(A.7.5)

since $\Xi_2^T \Xi_1 = \Xi_1^T \Xi_2$ is a scalar, and where * represents the conjugate transpose or the adjoint operation.
The second model to consider is that using real numbers given by:

\[
\begin{bmatrix}
(Y_1) \\
(Y_2)
\end{bmatrix} = \begin{bmatrix}
[A_1] & [-A_2] \\
[A_2] & [A_1]
\end{bmatrix} \begin{bmatrix}
(W_1) \\
(W_2)
\end{bmatrix} + \begin{bmatrix}
(\Xi_1) \\
(\Xi_2)
\end{bmatrix}
\]  

(2.30)

This model is the sum of two systems of equations:

\[
Y_1 = A_1 W_1 - A_2 W_2 + \Xi_1
\]

and \[
Y_2 = A_2 W_1 + A_1 W_2 + \Xi_2.
\]

The model is solved by minimizing with respect to \( W_1 \) and \( W_2 \) the error sum of squares:

\[
F = (Y_1 - A_1 W_1 + A_2 W_2)^T (Y_1 - A_1 W_1 + A_2 W_2) + (Y_2 - A_2 W_1 - A_1 W_2)^T (Y_2 - A_2 W_1 - A_1 W_2).
\]

By inspection, this is equal to:

\[
F = \Xi_1^T \Xi_1 + \Xi_2^T \Xi_2
\]

where the \( \Xi_1 \) and \( \Xi_2 \) come from,

\[
(Y_1 + iY_2) = (A_1 + iA_2)(W_1 + iW_2) + (\Xi_1 + i\Xi_2).
\]

The two models lead to the same error sum of squares and therefore the normal equations will be the same. The normal equations of the model:

\[
P = DR + \Xi
\]
are \((D^T D)r = D^T P\) or \(r = (D^T D)^{-1} D^T P\)

where \(P\) is a \((2n \times 1)\) vector of real observation parameters = 
\[
\begin{bmatrix}
(Y_1) \\
(Y_2)
\end{bmatrix}
\]

\(D\) is a \((2n \times 2p)\) matrix of reals = 
\[
\begin{bmatrix}
A_1 & -A_2 \\
A_1 & A_2
\end{bmatrix}
\]

and \(r\) is the \((2p \times 1)\) estimate of the vector \(R = \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}\)
APPENDIX 8

Error Propagation For Average Complex Moduli of Spectral Ratios

The complex solution vector is reconstructed from the real solution vector, $R$:

$$ R = \begin{bmatrix} (W_1) \\ (W_2) \end{bmatrix} $$

so that $W = W_1 + iW_2$. The options in Program FDPLOSS are:

a) no spectral ratio relative to other solution vector components
b) spectral ratio relative to solution vector component corresponding to reflected arrival from sea floor

c) spectral ratio relative to selected previous solution vector components.

Error propagation for option a).

Each complex solution vector component is $W_i$:

$$ W_i = a + ib $$

and its modulus $WM_i$ is:

$$ WM_i = \sqrt{a^2 + b^2}. $$

The standard deviations of $a$ and $b$, $S_a$ and $S_b$ respectively, are output from subroutine SVA. The standard error of $a^2 + b^2$ is:

$$ S_1 = \left[ 4a^2 S_a^2 + 4b^2 S_b^2 \right]^{1/2}. $$
The error of the square root is:

\[ x_2 = x_1^{1/2}, \quad S_{x_2} = \pm \frac{1}{2} \left( \frac{x_1}{x_1} \right) x_2, \text{ and therefore,} \]

\[ S_{WM_i} = \pm \left[ \frac{a^2 S_a^2 + b^2 S_b^2}{a^2 + b^2} \right]^{1/2} \sqrt{a^2 + b^2} \]
\[ = \pm \left[ \frac{a^2 S_a^2 + b^2 S_b^2}{a^2 + b^2} \right]^{1/2}. \quad (A.8.1) \]

The error of the average of N of these moduli is:

\[ S_{av} = \frac{1}{N} \left[ S_{WM_1}^2 + S_{WM_2}^2 + \ldots + S_{WN}^2 \right]^{1/2} \quad (A.8.2) \]

**Error propagation for operations a) or b)**

If each complex solution vector component is divided by another selected component of the same solution vector, the errors propagate as follows. Let the numerator be \((a + ib)\) and the denominator be the reference value, \((c + id)\). Then the ratio \(RL_j\) is:

\[ RL_j = \frac{(a+ib)}{(c+id)} = \frac{a+ib}{c+id} \frac{(c-id)}{(c-id)} \frac{1}{c^2 + d^2} [(ac + bd) + i(bc - ad)], \]

and the complex modulus is:

\[ |RL_j| = \frac{1}{c^2 + d^2} \left[ (ac+bd)^2 + (bc-ad)^2 \right]^{1/2} \quad (A.8.3) \]
The standard error of the various terms of (A.8.3) are:

\[ S(c^2 + d^2) = 2[c^2 S_{c}^2 + d^2 S_{d}^2]^{1/2} \]

\[ S(ac+bd) = \left\{ a^2 c^2 \frac{S_{c}^2}{a^2} + \frac{S_{c}^2}{c^2} \right\} + b^2 d^2 \left\{ \frac{S_{d}^2}{b^2} + \frac{S_{d}^2}{d^2} \right\}^{1/2} \]

\[ = \left\{ c^2 S_{a}^2 + a^2 S_{c}^2 + d^2 S_{b}^2 + b^2 S_{d}^2 \right\}^{1/2} \]

and \[ S(bc-ad) = \left\{ c^2 S_{b}^2 + b^2 S_{c}^2 + d^2 S_{a}^2 + a^2 S_{d}^2 \right\}^{1/2} \]

Let \[ Z = [(ac+bd)^2 + (bc-ad)^2]^{1/2} \], then:

\[ S_Z = \left\{ \frac{(ac+bd)^2 S_{ac+bd}^2 + (bc-ad)^2 S_{bc-ad}^2}{Z^2} \right\}^{1/2} \]

The final result for the standard error of the complex modulus of the spectral ratio is:

\[ S|RL_j| = |RL_j| \left[ \frac{S_{j}^2}{Z^2} + \frac{S(c^2 + d^2)}{(c^2 + d^2)} \right]^{1/2} \]

(A.8.4)

The average of N spectral ratios is then given by (A.8.2).

The formulas for error propagation are from Whittle and Yarwood (1973).
APPENDIX 9

Input and Processed Field Data
Figure A.9.1  Input unfiltered wide angle reflection (WAR) data for SB 3.
Figure A.9.2 Input unfiltered WAR data for SB 3, corrected for normal moveout.
Figure A.9.3  Input WAR data for SB 3 bandpass filtered from 5 Hz to 100 Hz.
Figure A.9.4  WAR data for SB 3 after source signature deconvolution, deghost filtering and bandpass filtering from 5 Hz to 100 Hz. Blackened peaks are picks for interval velocity computations.
Figure A.9.5  Vertical profiler data for SB 3; lower set is input unfiltered data and the upper set of traces were processed by source signature deconvolution, noise subtraction, and bandpass filtering from 10 Hz to 150 Hz. Blackened peaks are at data point locations used in least squares regression and numbered peaks separate layers with computed interval velocities.
Figure A.9.6  Input unfiltered WAR data for SB 5, corrected for normal moveout.
Figure A.9.7  WAR data for SB 5 after source signature deconvolution and bandpass filtering from 2.5 Hz to 29.5 Hz.
Figure A.9.8  Vertical profiler data for SB 5; lower set includes three of the traces chosen for processing and the upper set are source signature deconvolved and bandpass filtered from 50 Hz to 125 Hz.
Figure A.9.9  Input unfiltered WAR data for SB 46, corrected for normal moveout.
Figure A.9.10  WAR data for SB 46 after source signature deconvolution, bandpass filtering from 2.5 Hz to 112.5 Hz and notch filtering from 57.5 Hz to 62.5 Hz.
Figure A.9.11 Vertical profiler data for SB 46; lower set is input unfiltered data and upper set was processed with source signature deconvolution, noise subtraction, bandpass filtering from 15 to 150 Hz and notch filtering from 55 to 65 Hz.
Figure A.9.12  Input unfiltered WAR data for SB 57, corrected for normal moveout.
Figure A.9.13 WAR data for SB 57 after source signature deconvolution, bandpass filtering from 2.5 to 112.5 Hz and notch filtering from 57.5 to 62.5 Hz.
Figure A.9.14 Vertical profiler data for SB 57; lower set is input unfiltered data and upper set was processed with source signature deconvolution, noise subtraction, bandpass filtering from 15 to 150 Hz and notch filtering from 55 to 65 Hz.
Figure A.9.15 Input unfiltered WAR data for SB 61, corrected for normal moveout.
Figure A.9.16 WAR data for SB 61 after source signature deconvolution, bandpass filtering from 2.5 to 112.5 Hz and notch filtering from 57.5 to 62.5 Hz.
Figure A.9.17  Vertical profiler data for SB 61; lower set is input unfiltered data and upper set was processed with source signature deconvolution, noise subtraction, bandpass filtering from 15 to 150 Hz and notch filtering from 55 to 65 Hz.
# Appendix 10

## Program Listings and Flow Charts

### Program List

<table>
<thead>
<tr>
<th>Program*</th>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. A2DP1</td>
<td>Plots digitized data from mag tape.</td>
<td>188</td>
</tr>
<tr>
<td>2. DCN1OW</td>
<td>Source signature deconvolution, dehosting and bandpass filtering of seismic vertical reflection data recorded by streamer. Swaps with Program STKDCN.</td>
<td>271</td>
</tr>
<tr>
<td>3. DCN14W</td>
<td>Source signature deconvolution, dehosting and bandpass filtering of wide angle reflection data recorded by sonobuoy. Swaps with Program WARDCN.</td>
<td>306</td>
</tr>
<tr>
<td>4. FDPLOSS</td>
<td>Frequency dependent loss analysis. Reads from disk data file.</td>
<td>189</td>
</tr>
<tr>
<td>5. LAYRVEL</td>
<td>Computes interval velocities from wide angle reflection data.</td>
<td>219</td>
</tr>
<tr>
<td>6. SIGPSKW</td>
<td>Plots power spectrum of an input file which has been Fourier transformed.</td>
<td>239</td>
</tr>
</tbody>
</table>

* in alphabetical order, with subroutines called.
<table>
<thead>
<tr>
<th>Program</th>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7. SNPREPARE subroutine: WND8W FFT3W</td>
<td>Reads the output of STACKW and prepares it for Program STKDCN by removing mean and/or windowing source function and noise sample. It Fourier transforms both, writing first the noise then the source signature on file &quot;FFTSN&quot;.</td>
<td>240</td>
</tr>
<tr>
<td>8. STACKW subroutine: PLTW</td>
<td>Plots successive traces of digitized data from mag tape. Intended for plotting data detected by streamer. Stacks direct wave of airgun source for each plotted trace and stores file on disk. Reads a noise sample and stores on disk.</td>
<td>241</td>
</tr>
<tr>
<td>9. STKDCN</td>
<td>Reads digitized vertical seismic reflection data from mag tape and swaps with DCN10W for source signature deconvolution.</td>
<td>253</td>
</tr>
<tr>
<td>10. STKSB</td>
<td>Reads digitized wide angle reflection data and stacks a source signature from the sea bottom reflected arrival. Reads noise sample and stores.</td>
<td>278</td>
</tr>
<tr>
<td>11. STKSS</td>
<td>Reads digitized wide angle reflection data and stacks a source signature from the air gun direct wave received by the sonobuoy. Computes surface ghost delay times and writes results on a disk file. Reads noise sample and stores on disk.</td>
<td>283</td>
</tr>
<tr>
<td>12. WARDCN</td>
<td>Reads digitized wide angle reflection data from mag tape and swaps with DCN14W for source signature deconvolution</td>
<td>294</td>
</tr>
<tr>
<td>13. WARPLOT</td>
<td>Reads digitized wide angle reflection data from mag tape in the same way as WARDCN and plots.</td>
<td>310</td>
</tr>
<tr>
<td>14. WVKW8 subroutine: SCTKW CTFKW STFKW PLTKW</td>
<td>Creates a synthetic seismogram of delayed arbitrary wavelets each with a predetermined frequency independent phase shift.</td>
<td>313</td>
</tr>
</tbody>
</table>
Program       Function                                                                                     Page
15. WVW12    Creates a synthetic seismogram of delayed arbitrary wavelets each with a predetermined complex frequency dependent attenuation function. 317
           subroutines:
           FFTW
           WNDKW
           RANBW
           PLTWW

16. XSNSFW   Reads the output files of STKSS or STKSB and computes the FFT of the source function and the noise sample. Has the options of plotting amplitude spectrum of stacked source function and noise sample, stacked sonobuoy direct wave and stacking ghost filter amplitude spectrum. Writes on disk the files, "SONOSOURCE" and "SONONOISE" 322
           subroutines:
           FFT3W
           PLT3W

Subroutine List

| Subroutine* | Function                                                                                     Page |
|-------------|-----------------------------------------------------------------------------------------------|------|
| 1. CTFKW    | Cosine transform                                                                              | 315  |
| 2. FFTW     | Fast Fourier transform with complex input array dimensioned: CX(550).                        | 319  |
| 3. FFT3W    | Fast Fourier transform with complex input array dimensioned: COMMON/KWLBI/CX(1024)            | 274  |
| 4. HPFKW    | Frequency domain high pass filter                                                             | 277  |
| 5. LPFKW    | Frequency domain low pass filter                                                              | 276  |
| 6. OUTFILE  | Opens file with conversational file naming.                                                   | 276  |
| 7. PLTKW    | Plots a file passed to it through a common block: COMMON/KWLBI/XX(550), YY(550)              | 316  |

* in alphabetical order
<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8. PLTW</td>
<td>Plots a file passed to it through a common block: COMMON/KWLBL/YY(1024). X-axis label is LBLX(7).</td>
<td>252</td>
</tr>
<tr>
<td>9. PLTWW</td>
<td>Same as PLTKW except x-axis length is 8 inches instead of 10 inches.</td>
<td>321</td>
</tr>
<tr>
<td>10. PLT3W</td>
<td>Plots a file passed in through common: COMMON/KWLBL/YY(1024) EQUIVALENCE (YYC(1),ZZ(1)), x-axis is generated internally.</td>
<td>325</td>
</tr>
<tr>
<td>11. RANBW</td>
<td>Random number generator. Two types of random generators can be chosen, one with a variable starting parameter. Random series will have zero mean.</td>
<td>321</td>
</tr>
<tr>
<td>12. SCTKW</td>
<td>Calls CTFKW and STFKW and computes a specific sine-cosine function.</td>
<td>314</td>
</tr>
<tr>
<td>13. SNOTCH</td>
<td>Frequency domain notch filter.</td>
<td>277</td>
</tr>
<tr>
<td>14. STFKW</td>
<td>Sine transform</td>
<td>316</td>
</tr>
<tr>
<td>15. SVA</td>
<td>Univariate least squares regression by singular value analysis.</td>
<td>205</td>
</tr>
<tr>
<td>16. VGHOST</td>
<td>Frequency domain deghost filter for use in program STKDCN.</td>
<td>275</td>
</tr>
<tr>
<td>17. WGHOST</td>
<td>Frequency domain deghost filter for use in program WARDCN</td>
<td>309</td>
</tr>
<tr>
<td>18. WNDKW</td>
<td>Frequency domain window function. The attenuation function is symmetrical about the Nyquist frequency and can be either a linear ramp, an exponential or a rectangular window with a cosine taper.</td>
<td>320</td>
</tr>
<tr>
<td>19. WND8W</td>
<td>Time domain window with cosine taper.</td>
<td>274</td>
</tr>
</tbody>
</table>
PROGRAM A2DP1
DIMENSION IBUF(4,1024), KBUF(4,1024)
CALL MTOD (1, "MT0.0", 0, IER)
CALL INITIAL (6, 100, 0., 0.)
ACCEPT "WRITE DATA VALUES ON LINE PRINTER? YES=1, NO=0 ", NWRITE
IF(NWRITE.EQ.0)GO TO 7
ACCEPT "CHANNEL NO. TO BE WRITTEN=", NCH
DO 10 K=1,10
DO 15 I=1,1024,10
  KBUF(1,J)=IBUF(1,J)
  KBUF(2,J)=IBUF(2,J)
  KBUF(3,J)=IBUF(3,J)
  KBUF(4,J)=IBUF(4,J)
  J=J+1
CONTINUE
IF(NWRITE.EQ.0)GO TO 5
WRITE(12,100)(KBUF(NCH,J), J=1,1024)
FORMAT(1H ,10I10)
DO 30 J = 1, 4
  X = 0.0
  DO 20 I =1,1024
    Y = KBUF(J, I) / 1000.
    IF (ABS(Y) .GT. 2.) Y = SIGN(2., Y)
    X = X + 0.02
    CALL PLOT(X, Y, 2)
  20 CONTINUE
  CALL PLOT (0., 1., -3)
  CALL PLOT (0., 2., -3)
GO TO 7
END
Program FDPLOSS (Binary version - BFLSVA) computes the average complex moduli of spectral ratios for any number of input single channel reflection seismic traces. The program is conversational with the following input required:

1) number of data points in FFT of data sets
2) choice of computing spectral ratios relative to arrival from sea floor or relative to layer above
3) time domain digitizing interval
4) number of equations desired per regression
5) number of frequency intervals skipped per equation in regression
6) starting and ending frequencies of a complete scan within the range from zero to Nyquist.
7) (Data point locations - 1) of chosen peaks in inverse Fourier transform of a given data set
8) number of solution vectors to be stored for average
9) which solution vectors to be stored for average over all data sets
10) option of computing interval Q
11) a disk file which contains the velocities, standard deviations and thicknesses of selected intervals which correspond to the average solution vectors stored. The following is an example of the commands required for input data file preparations and for running the program. Included also is an example of line printer output for the analysis of one trace.
Example input for Program FDPLLOSS

Commands for file preparation

SB 61
Example conversational input

for Program FDPLLOSS

SB 61
Example output of Program FDPLOSS For SB 61

<table>
<thead>
<tr>
<th>FREQ</th>
<th>BNORM</th>
<th>R(NORM/3NORM)</th>
<th>19.7666</th>
<th>.339146E+02</th>
<th>.839174E+00</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>.540</td>
<td>.51</td>
<td>.554</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.540</td>
<td>.541</td>
<td>.534</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.551</td>
<td>.551</td>
<td>.540</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.540</td>
<td>.51</td>
<td>.554</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.1379E+01</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.1995E+01</td>
<td>STOEV =</td>
<td>.2576E+01</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.1606E+01</td>
<td>STOEV =</td>
<td>.2103E+01</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.1196E+01</td>
<td>STOEV =</td>
<td>.1616E+01</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.1027E+01</td>
<td>STOEV =</td>
<td>.1410E+01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQ</td>
<td>BNORM</td>
<td>R(NORM/3NORM)</td>
<td>39.2969</td>
<td>.240569E+02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.383</td>
<td>.385</td>
<td>.394</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.394</td>
<td>.391</td>
<td>.394</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.383</td>
<td>.385</td>
<td>.394</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.8334E+00</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.9926E+00</td>
<td>STOEV =</td>
<td>.9970E+00</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.7469E+00</td>
<td>STOEV =</td>
<td>.7072E+00</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.1164E+01</td>
<td>STOEV =</td>
<td>.1009E+01</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.8452E+00</td>
<td>STOEV =</td>
<td>.7714E+00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQ</td>
<td>BNORM</td>
<td>R(NORM/3NORM)</td>
<td>53.6231</td>
<td>.200146E+02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.315</td>
<td>.316</td>
<td>.323</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.323</td>
<td>.321</td>
<td>.315</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.315</td>
<td>.316</td>
<td>.323</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.5929E+00</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.2721E+00</td>
<td>STOEV =</td>
<td>.2360E+00</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.3631E+00</td>
<td>STOEV =</td>
<td>.3621E+00</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.7285E+00</td>
<td>STOEV =</td>
<td>.4653E+00</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.2964E+00</td>
<td>STOEV =</td>
<td>.2900E+00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQ</td>
<td>BNORM</td>
<td>R(NORM/3NORM)</td>
<td>79.3990</td>
<td>.140999E+02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.271</td>
<td>.271</td>
<td>.276</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.276</td>
<td>.271</td>
<td>.278</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.271</td>
<td>.271</td>
<td>.276</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.1163E+01</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.1339E+01</td>
<td>STOEV =</td>
<td>.1439E+01</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.1550E+01</td>
<td>STOEV =</td>
<td>.1716E+01</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.1891E+01</td>
<td>STOEV =</td>
<td>.2065E+01</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.2229E+01</td>
<td>STOEV =</td>
<td>.2414E+01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQ</td>
<td>BNORM</td>
<td>R(NORM/3NORM)</td>
<td>97.8306</td>
<td>.122029E+02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.188</td>
<td>.188</td>
<td>.193</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.193</td>
<td>.188</td>
<td>.193</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.188</td>
<td>.188</td>
<td>.193</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.2406E+01</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.3821E+01</td>
<td>STOEV =</td>
<td>.3849E+01</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.1312E+01</td>
<td>STOEV =</td>
<td>.3053E+01</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.2517E+01</td>
<td>STOEV =</td>
<td>.5616E+01</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.3662E+01</td>
<td>STOEV =</td>
<td>.8114E+01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQ</td>
<td>BNORM</td>
<td>R(NORM/3NORM)</td>
<td>117.4219</td>
<td>.306387E+01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.193</td>
<td>.193</td>
<td>.193</td>
</tr>
<tr>
<td></td>
<td></td>
<td>STD DEVIATIONS OF SOLUTION VECTORS</td>
<td>.193</td>
<td>.193</td>
<td>.193</td>
</tr>
<tr>
<td>RL (1)</td>
<td>=</td>
<td>.1000E+01</td>
<td>STOEV =</td>
<td>.1955E+01</td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>=</td>
<td>.2060E+01</td>
<td>STOEV =</td>
<td>.2033E+01</td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>=</td>
<td>.3946E+00</td>
<td>STOEV =</td>
<td>.1959E+01</td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>=</td>
<td>.6504E+01</td>
<td>STOEV =</td>
<td>.4446E+00</td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>=</td>
<td>.1816E+01</td>
<td>STOEV =</td>
<td>.1815E+01</td>
<td></td>
</tr>
<tr>
<td>FREQ</td>
<td>BNM</td>
<td>RNORM/BNM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>136.9331</td>
<td>8.82424E+01</td>
<td>8.29119E+00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**STANDARD DEVIATIONS OF SOLUTION VECTORS**

<table>
<thead>
<tr>
<th>.139</th>
<th>.139</th>
<th>.142</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
<th>.139</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL (1)</td>
<td>.1000E+01</td>
<td>STOEV = .1008E+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>.5329E+00</td>
<td>STOEV = .6412E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>.1804E+01</td>
<td>STOEV = .1714E+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>.6214E+00</td>
<td>STOEV = .797E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>.8779E+00</td>
<td>STOEV = .907E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FREQ | BNM | RNORM/BNM |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>156.4344</td>
<td>2.1691E+02</td>
<td>6.99351E+00</td>
</tr>
</tbody>
</table>

**STANDARD DEVIATIONS OF SOLUTION VECTORS**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RL (1)</td>
<td>.1000E+01</td>
<td>STOEV = .376E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>.1366E+01</td>
<td>STOEV = .756E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>.918E+00</td>
<td>STOEV = .570E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>.276E+00</td>
<td>STOEV = .2809E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>.239E+00</td>
<td>STOEV = .2794E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FREQ | BNM | RNORM/BNM |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>175.0156</td>
<td>7.79439E+01</td>
<td>6.03004E+00</td>
</tr>
</tbody>
</table>

**STANDARD DEVIATIONS OF SOLUTION VECTORS**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RL (1)</td>
<td>.1000E+01</td>
<td>STOEV = .3098E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>.5429E+00</td>
<td>STOEV = .2217E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>.195E+00</td>
<td>STOEV = .1759E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>.524E+00</td>
<td>STOEV = .1966E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>.326E+00</td>
<td>STOEV = .1587E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FREQ | BNM | RNORM/BNM |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>195.5469</td>
<td>8.68050E+01</td>
<td>9.12695E+00</td>
</tr>
</tbody>
</table>

**STANDARD DEVIATIONS OF SOLUTION VECTORS**

<table>
<thead>
<tr>
<th>.150</th>
<th>.151</th>
<th>.154</th>
<th>.153</th>
<th>.150</th>
<th>.151</th>
<th>.154</th>
<th>.153</th>
<th>.150</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL (1)</td>
<td>.1000E+01</td>
<td>STOEV = .7437E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (2)</td>
<td>.1937E+00</td>
<td>STOEV = .3166E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (3)</td>
<td>.1420E+00</td>
<td>STOEV = .3115E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (4)</td>
<td>.2035E+00</td>
<td>STOEV = .324E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL (5)</td>
<td>.1312E+00</td>
<td>STOEV = .3022E+00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FREQ | BNM | RNORM/BNM |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>199.766</td>
<td>19.766</td>
<td>58.828</td>
</tr>
</tbody>
</table>

**FREQ | BNM | RNORM/BNM |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>136.953</td>
<td>156.434</td>
<td>176.016</td>
</tr>
</tbody>
</table>

**STDEV (1) = 1.3793E+00 | 8.9337 | 5.9250 | 1.1624 | 2.4051 | 1.35471 |

**STDEV (2) = 1.1994E+00 | 7.0763 | 3.9985 | 2.0795 |

**STDEV (3) = 1.1352E+00 | 3.1662 | 1.6219 | 1.4932 |

**STDEV (4) = 1.0265E+00 | 7.5636 | 2.3401 | 1.4492 |

**STDEV (5) = 1.4309E+00 | 7.7114 | 2.5004 | 2.4137 |

**STDEV (6) = 1.0701E+00 | 2.7930 | 1.5070 | 3.0220 |

**STDEV (7) = 1.6214E+00 | 2.7640 | 5.2423 | 2.0530 |

**STDEV (8) = 1.6159E+00 | 1.0832 | 4.6331 | 2.0651 | 5.6179 |

**STDEV (9) = 1.7097E+00 | 2.3984 | 1.9662 | 3.2433 |

**STDEV (10) = 1.977E+00 | 2.9354 | 2.9404 | 2.2287 |

**STDEV (11) = 1.977E+00 | 2.9354 | 2.9404 | 2.2287 | 3.6624 | 1.5160 |

**STDEV (12) = 1.977E+00 | 2.9354 | 2.9404 | 2.2287 | 3.6624 | 1.5160 | 8.10357 | 1.61476 | 0.98701 | 2.7930 | 1.5070 | 3.0220 |
Flow Chart for Program FDPLOSS

START

Initialize Variables and Arrays to zero
NDATA=0

Conversational Input

1

Input NSKIP

SKIP DATA ?

yes

READ Data and DUMP

NDATA=NDATA+1

NDATA 1 ?

yes

NO

6
```plaintext
Input 6

Compute starting data point in frequency domain

READ ENTIRE DATA SET

Input

DETERMINE MAX NUMBER OF SOLUTION COMPONENTS TO BE STORED

Create an integer array which accumulates the number of values stored for each layer

Compute: # equations, number unknowns, Δf between bands and the starting frequency.

10

Compute last data point to be read in next regression, NII

5
```
11) NEND

Compute 1st data point in next regression

Increment # of regressions by one

Compute Integer Array of data points to be used in this regression

Rearrange complex Y-vector into 2(#equation) element Real vector

Rearrange Complex A-matrix into Real Array.

CALL SVA
Singular Value Analysis
Compute Average Moduli of uncorrected Spectral Ratios

WRITE on Line Printer the Spectral Ratios

Conversational Input

COMPUTE Q

Yes

Conversational Input, Correct for Geom. Spreading

READ VELOCITY, σ, DEPTH INFO

Correct for Geometric Spreading

WRITE CORRECTED SPECTRAL RATIOS ON LINE PRINTER
COMPUTE
INTERVAL Q
and
Error Estimates

WRITE ON
LINE PRINTER
the computed
Q values and
error estimates

END
PROGRAM CPLOCSS
6(INP,UT,OUTP,TAPE3,TAPE4,TAPE5=INP,TAPE6=OUTPUT,TAPE7,TAP8)
C THIS PROGRAM COMPUTES THE BAND-AVERAGED SPECTRAL RATIOS
C OF GIVEN ACOUSTIC ARRIVALS RELATIVE TO THE SOURCE FUNCTION,
C THE BOTTOM ARRIVAL (X(I)) OR TO CHOSEN SOLUTION VECTORS
C AS DESIRED. IT ALSO COMPUTES AN INTERVAL I FOR CHOSEN ARRIVALS
C AVERAGED OVER ANY GIVEN NUMBER OF DATA SETS. SOLUTION
C VECTORS CHOSEN TO BE STORED MUST BE CORRELATED ARRIVALS
C IN THE SAME ORDER FOR EACH DATA SET. ONLY FINAL
C VECTORS MAY BE MISSING.
C
DIENSION NHWT(100),NHJ(25),PM1(200),4.22000,4.22000,5.22129,205,129,23)
DIENSION SING(150),US(50),NAMES(50),STJER(30),NSORE(25)
DIENSION VEL(25),VSTO(25),DELTH(25),SMJ(25),SUMD(25)
DIENSION SMH(25),ALPHA(20),NACCUM(25)
COMPLEX PTM(513),CARG,AC1(100,25),CAI
DATA I3LANK/1H /
C
NAME$=1=BLANK
C
COMPUTE P AND THE A MATRIX OF P=AR, WHERE P=PM(I) AND R=RL(I)
NOATA=0
NSTMAX=0
DO 200 I=1,25
DATA (I)=0.
S=SM1=0.
SMQ(I,J)=0.
210 CONTINUE
WRITE(6,93)
READ(5,*) NT
NX=NT/2+1
1 READ(5,97) NSKP
IF(NSKP.EQ.0) GO TO 2
READ(4,120)(PTM[I],I=1,NT)
DO 3 I=1,NX
3 PTM[I]=0.
GO TO 1
2 IF(NOATA.EQ.11) GO TO 6
WRITE(6,114)
READ(5,*) NSFRA
IF(NSFRA.EQ.0) GO TO 7
NSPRB=0
GO TO 9
7 WRITE(6,99)
READ(5,99) NSPRA
8 IF(NSPRA.EQ.10) NSPRA.EQ.11) GO TO 11
WRITE(6,116)
READ(5,*) AMPFACT
11 WRITE(6,121)
READ(5,94)
WRITE(6,99)
READ(5,99) NSPRA
9 IF(NSPRA.EQ.10) NSPRA.EQ.11) GO TO 11
WRITE(6,116)
READ(5,*) AMPFACT
11 WRITE(6,121)
READ(5,94)
WRITE(6,99)
READ(5,99) NSPRA
8 IF(NSPRA.EQ.10) NSPRA.EQ.11) GO TO 11
WRITE(6,116)
READ(5,*) AMPFACT
11 WRITE(6,121)
READ(5,94)
WRITE(6,99)
READ(5,99) NSPRA
9 IF(NSPRA.EQ.10) NSPRA.EQ.11) GO TO 11
WRITE(6,116)
READ(5,*) AMPFACT
11 WRITE(6,121)
**REAL(5,*) FEND**

C
NJW(I) ARE DATA POINT LOCATIONS OF FREQUENCIES IN REGRESSION
NEND=IFIX(FEND*2.*PI/DELW)+1
6 NJW(I)=IFIX(FSTART*2.*PI/DELW)
READ(4,120) (PTW(I),I=1,144)
WRITE(6,104)
READ(5,*) NPEAK
C
READ DATA POINT LOCATIONS IN TIME OF CHOSEN ARRIVALS
WRITE(6,106)
READ(5,*) (NJT(I),I=1,NST)
60 IJ=0
DO 4 I=1,NPEAK
IF(NSTORE(I),EQ,0) GO TO 4
NJT(I)=NJT(I)+1
30 NJW(I)=NJW(J)+NSKIP
NJW(J)=NJW(J)+NSKIP
C
REARRANGE COMPLEX F-VECTOR INTO 2*NEQU ELEMENT REAL VECTOR
DO 40 I=1,NEQU
KII=NJW(I)+1
PW(I)=REAL(PTW(KII))
IA=I+NEQU
40 PW(I)=IAIMAG(PTW(KII))
DO 50 I=1,NEQU
DO 60 K=1,NPEAK
CARG=DELW*NJW(I)*NJT(K)*GTT*(0.,-1.,0.)
AC(K,I)=CEXP(CARG)
50 CONTINUE
C
REARRANGE COMPLEX AC MATRIX INTO REAL ARRAY
DO 70 I=1,NEQU
DO 75 J=1,NPEAK
A(I,J)=REAL(AC(I,J))
IB=I+NEQU
A(IB,J)=AIMAG(AC(I,J))
70 CONTINUE
C
CALL SVA(L,200,NEQZ,NPKZ,NEQZ,PM,JING,NAMES,1,DS+30),STDER,ROBN,BNRM)
WRITE(8,123) (STGER(I),I=1,NPK2)
C SPECTRAL RATIO RELATIVE TO SOURCE SIGNATURE
NP1=NPKEAK+1
REF=1
C=AI(I,NPK2)
D=A(NP1,NPK2)
CAI=CMPLX(C,D)
SG2D=4.*C*C*((STGER(I)**2)+4.*D*D*(STGER(NP1))**2)
DO 80 II=1,NPEAK
J=II+IPEAK
IF(NSPRA.EQ.1,4.,NSPRA.EQ.1)GO TO 84
RLMO=AI(I,NPK2)**2*(AI(J,NPK2))**2
SMRT=((AI(I,NPK2))**2)*(STGER(I))**2)
SMRT=SMRT*((AI(J,NPK2))**2)*(STGER(J))**2)
SMRT=SMRT/RLMOD
RLMOD=AMPFACT*SORT(RLMO)
GO TO 85
C SPECTRAL RATIO RELATIVE TO LAYER ABOVE OR BOTTOM ARRIVAL
54 RLMO=CABS(CMPLX(AI(I,NPK2),AI(J,NPK2)/CAI)
SAGD=((STGER(I)**2)**2)*(STGER(NREF))**2)**2*(AI(I,NPK2))**2)
SAGD=SCAGD*((STGER(J)**2)**2)*(STGER(NP1))**2)**2*(AI(J,NPK2))**2)
SAGD=((STGER(I)**2)**2)*(STGER(NREF))**2)**2*(AI(J,NPK2))**2)
SAGD=SCAGD*((STGER(I)**2)**2)*(STGER(NP1))**2)**2*(AI(J,NPK2))**2)
SMRT=RLMOD*SMRT*(S4Z((1+2*SCAGD)/(1+2))
SIC=AI(I,NPK2)*C+AI(J,NPK2)**2)
SIZ=(AI(I,NPK2)**2)*(SI/I(NPK2)**2)
SMT=SMRT/SMRT
RLMOD=AMPFACT*SORT(RLMO)
GO TO 85
C SPECTRAL RATIO RELATIVE TO LAYER ABOVE OR BOTTOM ARRIVAL
54 RLMO=CABS(CMPLX(AI(I,NPK2),AI(J,NPK2)/CAI)
SAGD=((STGER(I)**2)**2)*(STGER(NREF))**2)**2*(AI(I,NPK2))**2)
SAGD=SCAGD*((STGER(J)**2)**2)*(STGER(NP1))**2)**2*(AI(J,NPK2))**2)
SAGD=((STGER(I)**2)**2)*(STGER(NREF))**2)**2*(AI(J,NPK2))**2)
SAGD=SCAGD*((STGER(I)**2)**2)*(STGER(NP1))**2)**2*(AI(J,NPK2))**2)
SMRT=RLMOD*SMRT*(S4Z((1+2*SCAGD)/(1+2))
SIC=AI(I,NPK2)*C+AI(J,NPK2)**2)
SIZ=(AI(I,NPK2)**2)*(SI/I(NPK2)**2)
SMT=SMRT/SMRT
RLMOD=AMPFACT*SORT(RLMO)
GO TO 85
C SUM OF MODULI OF SPECTRAL RATIOS FOR AVERAGE
C IF THIS VALUE OF I IS A CHOSEN LAYER 8EST DENOMINATOR OF
C SPECTRAL RATIO: ALL SOLUTIONS BETWEEN CHOSEN COEFFICIENTS
C ARE RATIOED RELATIVE TO PREVIOUS CHOSEN COEFFICIENT
SMQII(NKR)=SMQII(NKR)*SMT
SII(NKR)=SII(NKR)+RLMOD
C CHANGE DENOMINATOR IF RATIOING RELATIVE TO LAYER ABOVE
IF(NSPRA.EQ.1)GO TO 83
NP1=NPNEAK+1
REF=1
C=AI(I,NPK2)
D=A(NP1,NPK2)
CAI=CMPLX(C,D)
SG2D=4.*C*C*((STGER(I)**2)+4.*D*D*(STGER(NP1))**2)
83 WRITE(8,135) I,RLMOD,SMT
GO TO 10
20 WRITE(6,105)
READ(5,*)NREF
IF(NREF.EQ.1)GO TO 110
GO TO 1
30 FORMAT(1H ,COMPUTE SPECTRAL RATIOS REL: TO X(1+ YES=1, NO=0))
7 FORMAT(1H ,#SKIP THIS DATA SET YES=1, NO=0)
98 FORMAT(1H ,26HNC. OF DATA POINTS IN FFT=)

FORMAT(1H ,20HO DIGITIZING INTERVAL=)
100 FORMAT(1H ,2TH STARTING FREQUENCY (HERTZ=)
101 FORMAT(1H ,17H ENDING FREQUENCY=)
102 FORMAT(1H ,16H NO. OF FREQ INT SKIPPED PER EQUATION IN REGRESSION=)
103 FORMAT(1H ,17H NO. OF EQUATIONS=)
104 FORMAT(1H ,31H NO. OF UNKNOWNs IN SOLN VECTOR=)
105 FORMAT(F8.5)
106 FORMAT(1H, TIMES OF CHOSEN ARRIVALS: (DATA POINT LOCATIONS=1)^)
107 FORMAT(1H, STORE HOW MANY SOLN. VECTORS=)
108 FORMAT(1H, STORE WHICH SOLN. VECTORS=)
109 FORMAT(1H, STOP READING DATA SETS: YES=1, NO=0)
110 FORMAT(1H, #COMPUTE Q: YES=1, NO=0)
111 FORMAT(1H, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
112 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
113 FORMAT(1F6.3)
114 FORMAT(1F6.3)
115 FORMAT(1F6.3)
116 FORMAT(1H, #APPL FACTOR OF SOURCE SIGN| USUALLY A FRACTION=)
117 FORMAT(IH, #WRITE STANDARD DEV. OF SOLUTION VECTORS: YES=1, NO=0)
118 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
119 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
120 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
121 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
122 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
123 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
124 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
125 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
126 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
127 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
128 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
129 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
130 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
131 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
132 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
133 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
134 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
135 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
136 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
137 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
138 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
139 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
140 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
141 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
142 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
143 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
144 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
145 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
146 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
147 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
148 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
149 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
150 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
151 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
152 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
153 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
154 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
155 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
156 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
157 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
158 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
159 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
160 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
161 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
162 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
163 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
164 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
165 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
166 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
167 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
168 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
169 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
170 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
171 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
172 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
173 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
174 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
175 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
176 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
177 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
178 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
179 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
180 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
181 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
182 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
183 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
184 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
185 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
186 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
187 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
188 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
189 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
190 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
191 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
192 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
193 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
194 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
195 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
196 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
197 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
198 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
199 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
200 FORMAT(IH, #AVERAGE MODULI OF UNCORRECTED SPECTRAL RATIOS=)
201 FORMAT(IH, #STANDARD DEVIATIONS OF SOLUTION VECTORS=)
202 FORMAT(IH, #NO. OF TIMES OF CHosen ARRIVALS=)
203 FORMAT(IH, #COMPUTE Q: YES=1, NO=0)
GO TO 235
96 G=DEPTH/RO
SGSUM=SGSUM+(SMCR(J)**2)
IF(NSPRA.EQ.1.OR.NSPRA.EQ.1)GO TO 236
SGM=SGSUM/RO
GO TO 235
236 SGM=SGM/(DEPTH*DEPTH)*(SMOR(J)**2/(RO*RO))/RO
SMG=SMG/DEPTH
C CORRECT FOR GEOMETRIC SPREADING
235 DO 240 J=1,NKR
YRAV=Q(I,J)
Q1(I,J)=Q(I,J)*G
SMQ=SQRT((SMQ(I,J)**2/(YRAV*YRAV)+SMG*SMG/(G*G))
240 IF(NSPRA.EQ.1)GO TO 310
RG=DEPTH
310 IF(NGEOM.EQ.1)GO TO 230
WRITE(8,113) I,(Q(I,J),J=1,NKR)
WRITE(8,116) I,(SMQ(I,J),J=1,NKR)
230 CONTINUE
JJA=1
330 FREQ=FSTART-OF/2
C COMPUTE INTERVAL 0
DO 250 I=1,NKR
IF(NSPRA.EQ.0.AND.NSPRA.EQ.3)GO TO 245
IF(JJA.GT.1)GO TO 245
Q(I,J)=1.0
SMQ(I,J)=Q(I,J)
GO TO 250
245 FREQ=FREQ+DF
XLN=ALOG(Q(JJA,J))
ALPHA(J)=CUMACP(I)+XLN/ETH(I)
SMA=SMQ(JJA,J)*(SMQ(JJA,I)*Q(I,J)*XLN*XLN)
SMA=SMQ(JJA,J)*(SMQ(JJA,I)*Q(I,J)*XLN*XLN)
IF(NSPRA.EQ.1)GO TO 255
SMTEMP=(((ALPHA(J)**2)*((DELTH(J)**2)
SMTEMP=(((ALPHA(J)**2)*((DELTH(J)**2)
SMSM(I)=SMSM(I)+SMTEMP
255 Q(JJA,I)=PI*FREQ/(ALPHA(I)*VEL(JJA))
250 CONTINUE
IF(NSPRA.EQ.0)GO TO 320
JJA=JJA+1
IF(JJA.GT.NSTMAX)GO TO 340
GO TO 330
220 DO 260 I=2,NSTMAX
FREQ=FSTART-DF/2
IM1=I-1
DO 270 J=1,NKR
FREQ=FREQ-DF
XLN=ALOG(Q(I,J))
SUMA=SUMA+Q(I,J)*ALPHA(I)*DELTH(I)
ALPHA(J)=(SUMA+Q(I,J)*XLN/DELTH(I)
SMA=SMQ(I,J)*(SMQ(I,J)**2)/(Q(I,J)**2)
SMA=SMQ(I,J)*(SMQ(I,J)**2)/(Q(I,J)**2)
SMTEMP=((SUMA(I,J)*XLN/DELTH(I)**2)*((SMQ(I,J)**2)/(DELTH(I)**2))
SMTEMP=((SUMA(I,J)*XLN/DELTH(I)**2)*((SMQ(I,J)**2)/(DELTH(I)**2))
SMSM(I)=SMSM(I)+SMTEMP
Q(I,J)=PI*FREQ/(ALPHA(I)*VEL(I))
SMQ(I,J)=Q(I,J)*SUMA+((VSTO(I)**2)/(VEL(I)**2))
270 CONTINUE
260 CONTINUE
SUBROUTINE SVA (A,M0A,M,N,M3ATA,3,SING, NAMES,ISCALE,C,
6STDER,R,HOBN,BNORM)
C resemblance of A,MOA,M,N,M3ATA,3,PHI,3,NAMES,ISCALE,C,
6STDER,R,HOBN,BNORM)
C ...
G TO 7U
60 CONTINUE
WRITE (7, 290)
70 CONTINUE

* OBTAIN SING. VALUE DECOM. OF SCALED MATRIX.
* *
* CALL SVORS (A, M, N, M, S, 1, SING)
* *
* PRINT THE V MATRIX.
* DO 400 IW = 1, N
* SUM = 0.
* DO 405 JM = 1, N
* SUM = SUM + A(IW, JM) * (SING(IW) * SING(JM))
* 400 SUM = SORT(SUM)
* CALL MFEOUT (A, M, N, N, NAMES, 1)
*C IF (ISCALE.EQ.1) GO TO 90
*C
*REPLACE V BY D*V IN THE ARRAY A(I)
* DO 80 IW = 1, N
* 80 A(IW, J) = D(I) * A(IW, J)
*C G NOW IN B ARRAY. V NOW IN A ARRAY.
*C
*OBTAIN SUMMARY OUTPUT.
* 90 CONTINUE
* WRITE (7, 220)
*C
*C COMPUTE CUMULATIVE SUMS OF SQUARES OF COMPONENTS OF
*C G AND STORE THEM IN SING(I), I = MINMN+1,...,2*MINMN+1
* SD = ZERO
* MINMN = MINMN+1
* MINMN = MINMN+1
* IF (M.EQ.MINMN) GO TO 110
* DO 100 I = MINMN, M
* 100 SB = SB + DELE(B(I)) ** 2
* 110 SING(2*MINMN+1) = SB
* DO 120 JJ = 1, MINMN
* 120 SING(J) = SING(MINMN+1)
*AJ = SING(MINMN+1)
*A4 = SORT(A3 / FLOAT(MAX(1, MCATA)))
* WRITE (7, 230) A3, A4
NSOL = 0

DO 150 K = 1, MINMN
IF (SING(K).EQ.ZERO) GO TO 130
NSOL = K
PI = B(K) / SING(K)
A1 = ONE / SING(K)
A2 = B(I)**2
A3 = SING(MINMN1 + K)
A4 = SORT(A3/FLOAT(MINMINA-K))

IF (K.NE.MINMN) GO TO 420
DO 140 IJK = 1, N

410 STERO(IJK) = A4*STERO(IJK)

420 TEST = SING(K).GE.100 OR SING(K).LT.0.1
IF (TEST) WRITE (7, 250) K, SING(K), PI, A13(K), A2, A3, A4
IF (.NOT. TEST) WRITE (7, 270) K, SING(K), PI, A13(K), A2, A3, A4
GO TO 140

130 WRITE (7, 340) K, SING(K)

140 DO 150 I = 1, N
A(I,K) = A(I,K)*D
150 IF (K.GT.1) A(I,K) = A(I,K)*A(I,K-1)
CONTINUE

500 COMPUTE AND PRINT VALUES OF YNORM AND RNORM.

WRITE (7, 300)
J = 0
YSQ = ZERO
GO TO 160

170 J = J + 1
YSQ = YSQ + (E(J)/SING(J))**2
180 YNORM = SORT(YSQ)
JS = MINMN1 + J
RNORM = SORT(YNORM)

YL = -1.000.
IF (YNORM.GT.0.) YL = ALG10(YNORM)
RL = -1.000.
IF (RNORM.GT.0.) RL = ALG10(RNORM)
IF (J.NE.NSOL) GO TO 430
RNOR = RNORM/SNOR
WRITE (7, 310)
190 YNOR, RNOR, YL, RL
IF (J.LT.NSOL) GO TO 170

500 COMPUTE VALUES OF XNORM AND RNORM FOR A SEQUENCE OF VALUES OF THE LEVENBERG-MARQUARDT PARAMETER.

IF (SING(I).EQ.ZERO) GO TO 210
EL = ALG10(SING(I)) + ONE
EL2 = ALG10(SING(NSOL)) + ONE
DEL = (EL2 - EL)/20.
TEN = 10.
ALNM = ALG10(TEN)
WRITE (7, 320)
DO 200 IE = 1, 21
500 COMPUTE
ALAMB = 10**EL
ALAMB = EXP(ALNM*EL)
YS = 0.
JS = MINMN1 + NSOL
RS = SING(1ST)
DO 130 I = 1, MINMN
SL = SING(I)**2 + ALAMB**2
YS = YS + (S(I)**2+SING(I)/SL)**2
RS = RS + (E(I)**2+ALAMB**2)/SL**2
CONTINUE

200 WRITE (7, 330)

500 COMPUTE
ALAMB = 10**EL
ALAMB = EXP(ALNM*EL)
YS = 0.
JS = MINMN1 + NSOL
RS = SING(1ST)
DO 130 I = 1, MINMN
SL = SING(I)**2 + ALAMB**2
YS = YS + (S(I)**2+SING(I)/SL)**2
RS = RS + (E(I)**2+ALAMB**2)/SL**2
CONTINUE

200 WRITE (7, 330)

500 COMPUTE
ALAMB = 10**EL
ALAMB = EXP(ALNM*EL)
YS = 0.
JS = MINMN1 + NSOL
RS = SING(1ST)
DO 130 I = 1, MINMN
SL = SING(I)**2 + ALAMB**2
YS = YS + (S(I)**2+SING(I)/SL)**2
RS = RS + (E(I)**2+ALAMB**2)/SL**2
CONTINUE

200 WRITE (7, 330)
IF (YNORM.GT.ZERO) YL=ALOG10(YNORM)
WHITE (7,330) ALE49,YNORM,RNORM,EL,YL,EL

CONTINUE

PRINT CANDIDATE SOLUTIONS.

IF (NSOL.GE.1.) CALL MFEOUI (A,MQA,N,NSOL,NAMES,2)
RETURN

FORMAT (42H INDEX SING. VALUE P COEF ,4H RECIPI. S.
IV. G COEF G**2 ,3H C.S.S.)

C FORMAT (80X, x1HO,8X,E15.4,1PE17.4)
C FORMAT (1H1,6E12.4,1PE15.4,E15.4,4X,E15.4,4X,E15.4,4X,E15.4,
12X,E15.4))

C FORMAT (1H1,6E12.4,1PE15.4,E15.4,4X,E15.4,4X,E15.4,4X,E15.4,
12X,E15.4))

C FORMAT (I5H0 INDEX 10H FORM 1X .5H CI.S.S.
18H FORM (10H NORMS OF THE LEAST SQUARES RESIDUALS FOR A RANGE OF
14H VALUES OF THE LEVBERG-Marquardt PARAMETER gamma, LAMDA/1100-A
14H LAMDA LAMDA/100 42H LOG10 (LAMDA)

C SUBROUTINE SVDS (A,M,Q,N,S)
C
C SPECIAL SINGULAR VALUE DECOMPOSITION SUBROUTINE.
C
C WE HAVE THE M X N MATRIX A AND THE SYSTEM A*X=B TO SOLVE.
C EITHER M .GE. N OR M .LT. N IS PERMITTED.
C THE SINGULAR VALUE DECOMPOSITION
A = U*S*V**T (1) IS MADE IN SUCH A WAY THAT ONE GETS
C (1) THE MATRIX V IN THE FIRST N ROWS AND COLUMNS OF A.
C (2) THE DIAGONAL MATRIX UP ORDERED SINGULAR VALUES IN
C THE FIRST N COLUMNS OF THE ARRAY S.
C (3) THE MATRIX PRODUCT U**T (1)**T IS PLACED BACK IN A.
C (4) THE USER MUST COMPLETE THE SOLUTION AND DO HIS OWN
C SINGULAR VALUE ANALYSIS.
C
C GIVE SPECIAL
C TREATMENT TO ROWS AND COLUMNS WHICH ARE ENTIRELY ZERO. THIS
C CAUSES CERTAIN ZERO SING. VALS. TO APPEAR AS EXACT ZEROS RATHER
C THAN AS SMALL SING. VALS. IT SIMILARLY
C CLEANS UP THE ASSOCIATED COLUMNS OF U AND V.
C
C METPOC
C 1. EXCHANGE COLS OF A TO PACK NONZERO COLS TO THE LEFT.
C SET N = NO. OF NONZERO COLS.
C USE LOCATIONS A(1,NX+1),A(1,NM+1),...,A(1,4N+1) TO RECORD THE
C COL PERMUTATIONS.
2. EXCHANGE ROWS OF A TO PACK NONZERO ROWS TO THE TOP.
QUIT PACKING IF FIND N NONZERO ROWS. MAKE SAME ROW EXCHANGES
IN B. SET M SO THAT ALL NONZERO ROWS OF THE PERMUTED A
ARE IN FIRST M ROWS. IF M .LE. N THEN ALL M ROWS ARE
NONZERO. IF M .GT. N THEN THE FIRST N ROWS ARE KNOWN
TO BE NONZERO, AND ROWS N+1 THRU M MAY BE ZERO OR NONZERO.
3. APPLY ORIGINAl ALGORITHM TO THE M BY M PROBLEM.
4. MOVE PERMUTATION RECORD FROM A() TO S(I), I=1, ..., NN.
5. BUILD V UP FROM N BY N TO NN BY NN BY PLACING ONES ON
THE DIAGONAL AND ZEROS ELSEWHERE. THIS IS ONLY PARTLY DONE
EXPLICITLY. IT IS COMPLETED DURING STEP 6.
6. EXCHANGE ROWS OF V TO COMPENSATE FOR COL EXCHANGES OF STEP 2.
7. PLACE ZEROS IN S(I), I=N+1, NN TO REPRESENT ZERO-SING-VALS.

SUBROUTINE SVDRS (A, M, A, MM, NN, B, M, B, NB, S, NN, M)
DIMENSION A(M, A), B(M, B), S(NN, M)

BEGIN. SPECIAL FOR ZERO ROWS AND COLS.
PACK THE NONZERO COLS TO THE LEFT.

N=NN
IF (N LE. 0. OR. M.LE. 0) RETURN
J=N
10 CONTINUE
DO 20 I=1, MM
20 CONTINUE
IF (A(I, J)) 50, 20, 53
CONTINUE

IF J IS ZERO, EXCHANGE IT WITH COL N.
IF (J.EQ.N) GO TO 40
DO 30 I=1, MM
30 A(I, J)=A(I, N)
40 CONTINUE
A(I, J)=J
N=N+I
50 CONTINUE
J=J+1
IF (J.GE.1) GO TO 10
IF N=0 THEN A IS ENTIRELY ZERO AND SVD
COMPUTATION CAN BE SKIPPED
NS=0
IF (N.LE.0) GO TO 240

PACK NONZERO ROWS TO THE TOP.
QUIT PACKING IF FIND N NONZERO ROWS.

I=1
M=NN
60 IF (I.GT.M.0 OR. I.GE.4) GO TO 150
IF (A(I, I)) 90, 70, 90
70 DO 80 J=I, M
80 CONTINUE
GO TO 100
90 I=I+1
GO TO 60

ROW I IS ZERO
EXCHANGE ROWS I AND N
100 IF (NS.LE.0) GO TO 110
DO 110 J=1, NS
T=B(I, J)
Bi(J) = 0 (M,J)

DO 120 J=1,N
A(I,J) = A(M,J)

IF (M.GT.N) GO TO 140
DO 130 J=1,N
A(I,J) = ZERO

140 CONTINUE

EXCHANGE IS FINISHED

GO TO 60

150 CONTINUE

END.. SPECIAL FOR ZERO ROWS AND COLUMNS

METHOD.. BEGIN.. SVD ALGORITHM

(1) REDUCE THE MATRIX TO UPPER BIDIAGONAL FCP* WITH HOUSEHOLDER TRANSFORMATIONS.

H(N) ... H(1)AQ(1) ... Q(N-2) = (D**T, 0)**T

WHERE D IS UPPER BIDIAGONAL.

(2) APPLY H(N) ... H(1) TO B. HERE H(N) ... H(1)*B REPLACES B IN STORAGE.

(3) THE MATRIX PRODUCT W = Q(1) ... Q(N-2) OVERWRITES THE FIRST N ROWS OF A IN STORAGE.

(4) AN SVD FOR D IS COMPUTED. HERE K ROTATIONS P0 AND PI ARE COMPUTED SO THAT

RK ... R*K*PI**(T) ... PK**(T) = DIAG(S1, ..., SM)

TO WORKING ACCURACY. THE SI ARE NONNEGATIVE AND NONINCREASING.

HERE RK ... R*K OVERWRITES B IN STORAGE WHILE

A*PI**(T) ... PK**(T) OVERWRITES A IN STORAGE.

(5) IT FOLLOWS THAT, WITH THE PROPER DEFINITIONS,

U**(T)*B OVERWRITES B, WHILE V OVERWRITES THE FIRST K ROW AND

COLUMNS OF A.

L = MIN0(M, N)

THE FOLLOWING LOOP REDUCES A TO UPPER BIDIAGONAL AND

ALSO APPLIES THE PREMULTIPLYING TRANSFORMATIONS TO B.

DO 170 J=1,L
IF (J.GE.M) GO TO 160
CALL H12 (1, J+1, J, A(I, J), 1, T, A(I, J+1), 1, MDA, N-J)
160 CONTINUE

CALL H12 (1, J+1, J+2, N, A(J, J), MDA, S(J, J), 1, MDA, 1, M-J)
170 CONTINUE

COPY THE BIDIAGONAL MATRIX INTO THE ARRAY S(I) FOR QRED.

IF IN.EQ.11 GO TO 190
DO 180 J=2, N
S(J,1) = A(J,J)
180 S(J,2) = A(J-1, J)
190 S(I,1) = A(I, 1)

NS = N
IF (M.GE.N) GO TO 200
NS = M+1
S(NS,1) = ZERO
S(NS,2) = A(M, M+1)
CONSTRUCT THE EXPLICIT N BY N PRODUCT MATRIX, W=G1*G2*...*Gm
IN THE ARRAY A(I,J).

DO 230 K=1,N
I=1+1-K
IF(I.GT.MIN0(M,N-2)) GO TO 210
CALL H2 0, I+1, I+2, W, A(I,1), IRA, S(I,1), A(I,1+1), I, MDA, N=I
210 CALL D2 0, J=1,N
A(I,J)=ZERO
230 A(I,I)=ONE

CALL XORO(IPASS, S(I,1), S(I,2), NS, A, MDA, N, B, MDA, NE)

GO TO (240, 31C), IPASS

CONTINUE
IF (NS.GE.N) GO TO 260
NS1=NS+1
DO 250 J=NS1,N
250 S(I,J)=ZERO
CONTINUE
IF (N.EQ.NN) RETURN
NP1=NP+1

DO 260 J=NP1,NN
S(I,J)=A(I,J)
DO 270 I=1,N
A(I,J)=ZERO
270 A(I,J)=ZERO
260 CONTINUE

PERMUTE ROWS AND SET ZERO SINGULAR VALUES.

DO 300 K=NP1,NN
I=S(I,1)
S(I,1)=ZERO
DO 290 J=1,NN
A(I,J)=A(I,J)
A(I,J)=ZERO
290 A(I,J)=ONE
300 CONTINUE

RETURN

END -- SPECIAL FOR ZERO ROWS AND COLUMNS --

WRITE (7, 320)

320 FORMAT (4WH CONVERGENCE FAILURE IN OR BIIDIAGONAL SV ROUTINE)

SUBROUTINE ORDO(IPASS, Q, N, F, NN, V, M, NV, G, MDV, NCC)

C LAWSON AND HANS, JET PROPULSION LABORATORY, 1973 JUN 12
TO APPEAR IN SOLVING LEAST SQUARES PROBLEMS, PRENICE-HALL, 1974
OR ALGORITHM FOR SINGULAR VALUES OF A BIIDIAGONAL MATRIX.

THE BIDIAGONAL MATRIX

D= [Qf, E2, 0, ...]
[0, Q2, E3, 0, ...] [0, 0, 0, ...]
ELEMENTARY ROTATION MATRICES
RI AND PI SO THAT
\[ RK_{i} \cdots R_{1} \ast P_{i} \cdots P_{k} = \text{DIAG}(31, \ldots, 3n) \]

TO WITHIN WORKING ACCURACY.

1. EI AND QI OCCUPY E(I) AND Q(I) AS INPUT.

2. RM_{i} \cdots R_{1} \ast C\# IN STORAGE AS OUTPUT.

3. V \ast P_{i} \cdots P_{k} \ast C\# IN STORAGE AS OUTPUT.

4. SI OCCUPIES Q(I) AS OUTPUT.

5. THE SIS ARE NONINCREASING AND NONNEGATIVE.

THIS CODE IS BASED ON THE PAPER AND \#ALGOL\# CODE.

1. REIMBUSH, C.H. AND GOLUB, G.H. \#SINGULAR VALUE DECOMPOSITION

SUBROUTINE QRED (IPASS, E, NN, V, CV, NR, G, MGC, NCC)
LOGICAL WNTV, HAVERS, FAIL
DIMENSION Q(NN), E(NN), IMOV(NN), G(MGC, NCC)
ZER0 = 0,
ONE = 1,
TWO = 2.

N = NN
IPASS = 1
IF (N.LE.0) RETURN
N10 = 10 * N
WNT = NRV.GT. 0
HAVERS = NCC.GT. 0
FAIL = .FALSE.
NQRS = 0
E(1) = ZERO
QNORM = ZERO

DO 10 J = 1, N

10
QNORM = AMAX1(ABS(Q(J)),ABS(E(1)))/QNORM
DO 20 K = 1, N
K = N + 1 - K

20
IF (K.EQ.1) GO TO 50
IF (DIFF(QNORM, Q(K)), CNORM) 50, 25, 50

C
TEST FOR SPLITTING OR RANK DEFICIENCIES.

C FIRST MAKE TEST FOR LAST DiAGONAL TERM, Q(K), BEING SMALL.

C
IF (K.EQ.1) GO TO 50
IF (DIFF(QNORM, Q(K)), CNORM) 50, 25, 50
C
SINCE Q(K) IS SMALL WE WILL MAKE A SPECIAL PASS TO
TRANSFORM E(K) TO ZERO.

C
CS = ZERO
SN = -ONE

DO 40 I = 2, K
I = K + 1 - I
F = -SN * E(I-1)
E(I-1) = CS * E(I-1)
CALL GI E(I), F, CS, SN, I(I)

C
TRANSFORMATION CONSTRUCTED TO ZERO POSITION (I, K).

C
IF (.NOT. WNTV) GO TO 40

DC 30 J = 1, NRV
CALL G2(CS,SN,V(J,I),V(J,K))

ACCUMULATE RT. TRANSFORMATIONS IN V.

CONTINUE

THE MATRIX IS NOW BIDAGONAL, AND OF LOWER ORDER

SINCE E(K) = 0.

DO 60 LL=1,K
L=K+1-LL

IF(DIFF(NORM+G(E(L),NORM)) .GT. 100,55)
IF(DIFF(NORM+G(1-L),NORM)) 60,70,60
CONTINUE

THIS LOOP CANT COMPLETE SINCE E(1) = ZERO.

GO TO 100

CANCELLATION OF E(L), L.GT.1.

CS=0
SN=-ONE

DO 90 I=L,K
F=-SN*E(I)
E(I)=CS*F(I)

IF(DIFF(NORM+F,NORM)) 75,107,75
CALL GI(Q(I),F,CS,SN,Q(I))
IF (.NOT.HAVE'S) GO TO 90

DO 100 J=1,NGC
CALL G2(CS,SN,G(I,J),G(L-1,J))

CONTINUE

TEST FOR CONVERGENCE..

Z=Q(K)

IF (L.EQ.K) GO TO 170

SHIFT FROM BOTTOM 2 BY 2 MINOR OF 3**(I)*3.

A=Q(I)
Y=Q(K-1)
G=E(K-1)
H=E(K)

F=((-Z)**2+G**2+H**2)/(G*H*Y)

IF (.LT.ZERO) GO TO 110

T=F/G

GO TO 120

110

T=F-G

120

F=((-Z)**2+(X-Z)**2+H**2)/(Y**2-H**2)*X

NEXT QR SWEEP..

CS=ONE
SN=ONE

DO 160 I=LP1,K

G=Q(I)
Y=Q(I)

H=SN*G

G=CS*G

CALL GI(F,H,CS,SN,E(I-1))

F=X*CS+G*SN

G=X*SN+G*CS

H=Y*SN

Y=Y*CS

IF (.NOT.WNTV) GO TO 140
ACUMULATF RTATIONS FROM THE RIGHT IN X'X.

DO 130 J=1,NRV

130 CALL G2 (CS,SN,V(J,I-1),V(J,I))

140 CALL G1 (F,CS,SN,Q(I-1))

F=CS*G+SN*Y
X=-SN*G+CS*Y

IF (.NOT.HAVES) GO TO 150

DO 150 J=1,NRV

150 CALL G2 (CS,SN,Q(I-1),J,C(I,J))

160 CONTINUE

E(I,J)=ZERO
E(I,J)=F
Q(I,J)=X

NORS=NORS+1
IF (NORS.LE.N) GO TO 20

RETURN TO TEST FOR SPLITTING.

FAIL=.TRUE.,

CUTOFF FOR CONVERGENCE FAILURE. NORS WILL BE 2*N. USUALLY.

170 IF (Z.GE.ZERO) GO TO 190

Q(K)=Z

IF (.NOT.HAVES) GO TO 190

DO 180 J=1,NRV

180 V(J,K)=V(J,I)

CONTINUE CONVERGENCE. Q(K) IS MADE NONNEGATIVE.

190 CONTINUE

CONVERGENCE. Q(K) IS MADE NONNEGATIVE.

200 CONTINUE

IF (N.EQ.1) RETURN

DO 210 I=2,N

210 CONTINUE

IF (Q(I,J).GT.0) GO TO 220

IF (.NOT.HAVES) RETURN

EVERY SINGULAR VALUE IS IN ORDER.

220 DO 270 I=2,N

230 CONTINUE

T=Q(I-1)

K=I-1

IF (T.GE.ZERO) GO TO 230

T=Q(J)

240 CONTINUE

IF (K.EQ.I-1) GO TO 270

Q(K)=Q(I-1)

Q(I-1)=T

IF (.NOT.HAVES) GO TO 250

DO 240 J=1,NRV

250 IF (.NOT.HAVES) GO TO 270

DO 250 J=1,NRV

260 CONTINUE

270 CONTINUE

END OF ORDERING ALGORITHM.
SUBROUTINE MFCUT (A, M, N, NAMES, NAME)

C. L. LAWSON AND R. J. HANSON, JET PROPULSION LABORATORY, 1973 JUN 12
TO APPEAR IN #SOLVING LEAST SQUARES PROBLEMS#, PRENTICE-HALL, 1974
SUBROUTINE FOR MATRIX OUTPUT WITH LABELING.

A( ) MATRIX TO BE OUTPUT

M0A FIRST DIMENSION OF A ARRAY
M NO. OF ROWS IN A MATRIX
N NO. OF COLS IN A MATRIX
NAMES(1) ARRAY OF NAMES. IF NAMES(1) = 1H , THE REST
OF THE NAMES() ARRAY WILL BE IGNORED.
MODE =1 FOR 4P815.0 FORMAT FOR V MATRIX,
=2 FOR 8E15.3 FORMAT FOR CANDIDATE SOLUTIONS.

DIMENSION A(M0A, 01)
INTEGER NAMES(M), IHEAD(2)
LOGICAL NOTBLK
DATA MAXCOL/4/, IBLANK/I1H /, IHEAD(1)/4H COL/, IHEAD(2)/4HSOLN/

C
NOTBLK=NAMES(1) .NE. IBLANK
IF (M.LE.0.OR. N.LE.0) RETURN
C
IF (MODE.EQ.2) GO TO 10
WRITE (7, 70)
GO TO 20
10 WRITE (7, 0)
20 CONTINUE
C
NBLOCK=N/ MAXCOL
LAST=N-NBLOCK*MAXCOL
NCOL=MAXCOL
J1=1
C
MAIN LOOP STARTS HERE

30 IF (NBLOCK.GT.0) GO TO 40
IF (LAST.LE.0) RETURN
LAST=0
C
J2=J1+NCOL-1
WRITE (7, 90) (IHEAD(MODE), J, J=J1, J2)
C
GO 60 I=I+1
NAME=IBLAN
IF (NOTBLK) NAME=NAMES(I)
C
IF (MODE.EQ.2) GO TO 50
WRITE (7, 100) I, NAME, (A(I,J), J=J1, J2)
GO TO 60
50 WRITE (7, 110) I, NAME, (A(I,J), J=J1, J2)
60 CONTINUE
C
J1=J1+MAXCOL
NBLOCK=NBLOCK-1
GO TO 30
C
70 FORMAT (45H0V-MATRIX OF THE SINGULAR VALUE DECOMPOSITION,
* 4H OF A*J, 47H (ELEMENTS OF V SCALED UP BY A FACTOR OF 10**4))
CONSTRUCTION AND/OR APPLICATION OF A SINGLE

MODE = 1 OR 2 TO SELECT ALGORITHM M1 OR M2.

LPIVOT IS THE INDEX OF THE PIVOT ELEMENT.

IF LI = 0, THE TRANSFORMATION WILL BE CONSTRUCTED TO
ZERO ELEMENTS INDEXED FROM LI THROUGH M. IF LI GT M,

THE SUBROUTINE APPLIES AN IDENTITM TRANSFORMATION.

U1, IUE, UP ON ENTRY TO H1 U1 CONTAINS THE PIVOT VECTOR.

IUE IS THE STORAGE INCREMENT BETWEEN ELEMENTS.

ON EXIT FROM H1 U1 AND UP

AND UP SHOULD CONTAIN QUANTITIES PREVIOUSLY COMPUTED
BY H1. THESE WILL NOT BE MODIFIED BY H2.

CL ON ENTRY TO H1 OR H2 C() CONTAINS A MATRIX WHICH WILL BE
REGARDED AS A SET OF VECTORS TO WHICH THE HOUSEHOLDER
TRANSFORMATION IS TO BE APPLIED. ON EXIT C() CONTAINS THE
SET OF TRANSFORMED VECTORS.

ICE STORAGE INCREMENT BETWEEN ELEMENTS IN C().

ICV STORAGE INCREMENT BETWEEN VECTORS IN C(I).

NCV NUMBER OF VECTORS IN C() TO BE TRANSFORMED. IF NCV = 0
NO OPERATIONS WILL BE DONE ON C().

SUBROUTINE M12 (MOOD.,LPIVOT,LI,M,U,ICV,Ncv)

DIMENSION (LIUE,M), C(I)
DOUBLE PRECISION SM,B
ONE = 1.

IF (G.E.,LPIVOT,CL,LPIVOT,GE,LI,DR,LT,GT,M) RETURN

IF (MODE.EQ.2) GO TO 61

****** CONSTRUCT THE TRANSFORMATION, ******

00 10 J=LI,1,M

CL=MAX1ABS(U1,1,J), CL)

IF (CL) 130,130,30

20 CLIHY=ONE/CL

S=SM clave uble(u1,1,LPIVOT)*CLIHY)**2

00 30 J=LI,1,M

S=SM+SM*CLAVE(u1,1,J)*CLIHY)**2

CONVERE ABLE. PREC. SM TO SGL. PREC. SM

S1=SM

CL=CL+SQRT(SM1)

IF (LI,1,LPIVOT) 50,50,40

CL=CL

30 U=U1,1,LPIVOT)*CL

40 U1,1,LPIVOT)=CL

GO TO 70

****** APPLY THE TRANSFORMATION I+U*(U**T)/B TO C. ******

50 IF (CL) 130,130,70

70 IF (NCV.LE.0) RETURN

9=ABLE(U1)*U1,1,LPIVOT)

B MUST BE NONPOSITIVE HERE. IF B .LE. 0, RETURN.
IF (B) 80,130,130
80 B=ONE/B
I2=I-ICV+ICE*(LPIVOT-1)
INC=ICE*(LI-LPIVOT)
DO 120 J=1,NCV
12 I2=I2+ICV
I3=I2+INC
I4=I3
SM=G(I2)*D BLE(U)
DO 90 I=LI,M
SM=SM+C(I3)*D BLE(U(1,1))
I3=I3+ICE
90 IF (SM) 100,120,100
SM=SM*B
C(I2)=C(I2)+SM*DBLE(U)
DO 110 I=LI,M
C(I4)=C(I4)+SM*DBLE(U(1,1))
110 I4=I4+ICE
120 CONTINUE
130 RETURN
SUBROUTINE G1 (T, C, COS, SIN, SIG)
CL. LAWSON AND R.J. HANSON, JET PROPULSION LABORATORY, 1973 JUN 12
TO APPEAR IN SCLING LEAST SQUARES PROBLEMS, PRENTICE-HALL, 1974
COMPUTE ORTHOGONAL ROTATION MATRIX...
COMPUTES MATRIX (C, 3) SO THAT (C, S) (4) = (SQR(A**2+B**2)
-C(3), -S(3))
COMPUTE SIG = SQR(A**2+B**2) SIG IS COMPUTE LAST TO ALLOW FOR THE POSSIBILITY THAT SIG MAY BE IN THE SAME LOCATION AS A OR B.
ZERO=0.
ONE=1.
IF (ABS(A).LE.ABS(B)) GO TO 10
XR=3/A
YR=SQR(T(ONE+X**2))
COS=SIGN(ONE/YR,A)
SIN=COS*XR
SIG=ABS(A)*YR
RETURN
10 IF (B) 20,30,20
XR=A/B
YR=SQR(T(ONE+X**2))
SIN=SIGN(ONE/YR,B)
COS=SIGN*XR
SIG=ABS(B)*YR
RETURN
30 SIG=ZERO
COS=ZERO
SIN=ONE
RETURN
SUBROUTINE G2 (COS, SIN, X, Y)
CL. LAWSON AND R.J. HANSON, JET PROPULSION LABORATORY, 1972-DEC-15
TO APPEAR IN SCLING LEAST SQUARES PROBLEMS, PRENTICE-HALL, 1974
APPLY THE ROTATION COMPUTED BY G1 TO X AND Y.
XR=COS*X+SIN*Y
Y=-SIN*X+COS*Y
X=XR
RETURN
FUNCTION DIFF(X,Y)
C C-L. LAWSON AND R.J. HANSON. JET PROPULSION LABORATORY, 1973 JUNE 7
C TO APPEAR IN SOLVING LEAST SQUARES PROBLEMS, PRENTICE-HALL, 1974
DIFF=X-Y
RETURN
END
Program LAYRVEL computes interval velocities between selected reflection hyperbolas of a plane dipping layer model. An example input data list and output is shown below.
Flow Chart for Program LAYRVEL

START

Read:
- LAYERS
- HEADER
- N_water, Dip
- Time-Dist pairs

CALL FIT
Apply 4th order
Least square polynomial fit to Time-dist pairs

IF
D(1)=0,

no

CALL FIT1
Apply 1st order polynomial fit to time-dist pairs

yes

CALL FIT
Apply 4th order
Least square polynomial fit to Time-dist pairs

Calculate
T_o at X=0
T_o = a_o of FIT or FIT1

Calculate
Z_1 = V(1)T_o(1)/2

Compute correction term to reduce to zero dip case and calculate reduced times.

Obtain least squares best fit slope of reduced time^2, dist^2 values.
Compute Horizontal water velocity = $V_{(1)} \sqrt{slopes}$ and $\phi$

**WRITE** LAYER, HORVEL, STD DEV, 2WAY-T, DIP, DEPTH

**READ**
N, DIP
T-D pairs for Layer 2

**CONVERT**
D-times to Distances and dip to radians relative to layer above

**CALL FIT**
Fit 4th order polynomial to T-D pairs of Layer 2

**CALL INTVEL**
Compute interval velocity for Layer 2

**READ**
N, DIP
T-D pairs for Layer 3

Convert D-times to distances and dips to radians relative to layer above

**WRITE**
LAYER #, VEL, $\phi$
T, DIP, Thicke
tess, Depth, # data pairs used in computation
CALL FIT for Layer 3

CALL INTVEL for Layer 3

WRITE LAYER #, VEL, \sigma\', T_o, DIP, THICKNESS, DEPTH, NS

READ N, DIP T-D pairs for Layer 4

Convert D-times to distances and dip to radians rel. to layer above.

CALL FIT for Layer 4

CALL INTVEL for Layer 4

WRITE LAYER #, VEL, \sigma\', T_o, DIP, THICKNESS, DEPTH, NS
READ
N, DIP
T-D pairs
for Layer 5

Convert
D-times to distance to radians
relative to layer above

CALL FIT
for Layer 5

CALL INTVEL
for Layer 5

WRITE
LAYER #, VEL, \( \phi \),
T_o, DIP, Thickness, Depth, NS

READ
N, DIP
T-D pairs
for Layer 6

Convert
D-times to distances and dip
to radians relative to layer above.

/READ
/N, DIP
T-D pairs
for Layer 6
CALL FIT for Layer 6

CALL INTVEL for Layer 6

WRITE LAYER #, VEL, \sigma, T_0, DIP, Thickness, DEPTH, NS

END
Flow Chart for Subroutine INTVEL

START

CALL FIT 1

IF X(1) = 0.

Yes

$T_o(i) = A_o$
of 4th order fit

No

Assume 1st
trial velocity
$V_i = V_{i-1}$

Compute Angles
of Refraction
in Layers
$\theta_i = \sin^{-1} \left[ V_i \frac{dT}{dx_i} \right]$,

Compute Reduced
Times and
Distances in
Layers

Correct to zero
dip and compute
reduced times
and distances.

Eliminate data pairs
if reduced $T^2$
or reduced $x^2$ or both
are negative
Calculate trial velocity $V'$ from linear least squares fit to reduced $T^2 - x^2$ data.

If all reduced $x^2$ and $T^2$ are eliminated:
- Yes: Write "ALL REDUCED $X$ AND $T$ ARE ZERO."

Correct dip angle according to:
$$\tan \theta' = \tan \theta \frac{V'}{V_i}$$

IF |$V_i - V'|$ is increasing:
- Yes: Write "NON-CONVERGENT"
- No:
  IF |$V_i - V'| < 0.001:
    - Yes: Set $V_i = V'$
    - Compute standard deviation,
  END

Set:
$$V_i = V'$$

RETURN

END
Flow Chart for Subroutine FIT

START

Compute elements of B matrix and Y vector of \( \bar{Y} = \bar{I}B \)

Solve system of equations for \( \bar{x} \) by Gauss-Jordan Elimination

RETURN

END
PROGRAM LAYRVEL

C PROGRAM LAYRVEL COMPUTES INTERVAL-SOUND VELOCITIES IN ONE PLANE
C DIPPING LAYERS GIVEN THE JIP AS COMPUTED FROM THE
C VERTICAL REFLECTION PROFILE AND THE TIME-DISTANCE
C DATA PAIRS FOR RAYS REFLECTING FROM THE INTERFACES.
C INPUT IS IN THE FOLLOWING FORMAT:
C REAL HEADER (II, A3)
C N+WHJ+WH1+13+2F8.5
C D-TIMES IN SECONDS=DISTANCE/HORIZONTAL WATER VELOCITY
C (MEASURED TO O-LINE) (10F6.3)
C ARRIVAL TIMES IN SECONDS (10F6.3)
C N+WHJ+13+2F8.5 (SAME FOR ALL SUCCESSIVE LAYERS)
C D-TIMES (10F6.3)
C ARRIVAL TIMES (10F6.3)
C
C MX=NUMBER OF REFLECTING INTERFACES (MAXIMUM OF 6)
C N=NUMBER OF DATA PAIRS (MAXIMUM OF 50)
C V=VERTICAL WATER COLUMN VELOCITY FROM MATTHEWS TABLES
C WH= DIP IN DEGREES OF REFLECTING INTERFACE RELATIVE TO
C HORIZONTAL (DIP IS + IF DIPPING UP FROM START TO FINISH
C COMPUTE HORIZONTAL SOUND VELOCITY IN UPPER WATER COLUMN
C
C DIMENSION COR(10),TSQR(10),T1(10),T2(10)
C TO(10),D(10),X2(10)
C REAL V(1),W1(1),WH(1),WX(1)+10,
C N+MX*5+2F8.5
C INTEGER 4,H,HX
C
C 1 READ(20,11)H,X,HEADER
C IF(EOF(20),11,11)
C CALL EXIT.
C 11 FORMAT(II,4B)
C WRITE(6,12)HEADER
C 12 FORMAT(1H,1A)
C
C H=1
C SUM2=0
C READ(20,13)O1(I),I=1,N
C READ(20,13)O1(I),I=1,N
C 9 FORMAT(13,2F5.5)
C E32=0.0
C 10 FORMAT(1F6.3)
C W(I)=WH(I)+(I+1)*7/453
C IF(W(I).LE.E1)GO TO 16
C CALL FIT1R(4,01,11,TP)
C T(I)=TP
C GO TO 17
C 16 CALL FIT1R(4,T1,01,04,A1,A2,A3,A4)
C T(I)=04
C 17 SUM1=SUM2=SUM3=SUM4=SUM5=2
C DO 1 I=1,N
C COR(I)=T(I)*T(I)-2*T(I)*O1(I)*SIN(W(I))
C TSQR(I)=T(I)*T(I)-COR(I)
C SUM1=SUM1+TSQR(I)*D(I)*O1(I)
C SUM2=SUM2+O1(I)*D(I)
C SUM3=SUM3+TSQR(I)
C SUM4=SUM4+O1(I)
C 1 CONTINUE
C RETURN
C END
C
C SLOPE1=(N*SUM1-SUM2*SUM3)/(N*SUM4-SUM2*SUM2)
C WH1=V(1)*SQR(SLOPE1)
C XINT=(SUM3-SLOPE1*SUM2)/N
C DO 2 I=1,N
C DELYSQ(I)=(TSQR(I)-XINT+SLOPE1*O1(I)*O1(I))*2
C SUM5=SUM5+DELYSQ(I)
<table>
<thead>
<tr>
<th>Layer</th>
<th>Velocity</th>
<th>Depth</th>
<th>DIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>0.65</td>
<td>10.5</td>
<td>15</td>
</tr>
<tr>
<td>L2</td>
<td>0.70</td>
<td>20.0</td>
<td>20</td>
</tr>
</tbody>
</table>

C: CALCULATE COEFFICIENTS OF FOURTH ORDER FIT FOR LAYER 2

```
WRITE(1,15) *THICK4E STS-N5-E1

15 FORMAT(18F8.5,15)
```

```
WRITE(1,15) *THICK4E STS-N5-E1

15 FORMAT(18F8.5,15)
```

```
WRITE(1,15) *THICK4E STS-N5-E1

15 FORMAT(18F8.5,15)
```
READ(23,13) (Q2(I),I=1,N)
READ(20,10) (T2(I),I=1,N)
K(2)=(WH2-WH1)*0.017453
IFM(W1)=EQ.0.0,1.0,W(2)=3.0001
DO 3 I=1,N
X2(I)=WH1*Q2(I)
3 CONTINUE
CALL FITIV(T2,X2,A22,A22,A12,A22,A32,A42)
C CALCULATE INTERVAL VELOCITY IN LAYER 2
K=2
CALL INTVEL(W,V,Z0,T0,A02,A12,A22,A32,A42,X2,T2,H,SIGMA2,N5)
SUM2=SUMZ+Z0(2)*Z0(1)
WRITE(61,L21)(V(2),SIGMA2,T0(2),H2,Z0(2),SUMZ,N5)
IF(Q2.EQ.0.G0) GO TO 101
Q2=Q2+1
C CALCULATE COEFFICIENTS OF FOURTH ORDER FIT FOR LAYER 3
READ(20,93)N,NH3
READ(20,10) (O3(I),I=1,N)
READ(20,10) (T3(I),I=1,N)
W(3)=(WH3-WH2)*0.017453
IFM(W3)=EQ.0.0,1.0,W(3)=0.0001
DO 4 I=1,N
X3(I)=WH1*O3(I)
4 CONTINUE
CALL FITIV(T3,X3,A03,A13,A23,A33,A43)
C CALCULATE INTERVAL VELOCITY IN LAYER 3
K=3
CALL INTVEL(W,V,Z0,T0,A03,A13,A23,A33,X3,T3,N,H,SIGMA3,N5)
SUM3=SUMZ+Z0(3)
WRITE(61,L23)(V(3),SIGMA3,T0(3),H3,Z0(3),SUMZ,N5)
IF(Q2.EQ.0.G0) GO TO 101
Q2=Q2+1
C CALCULATE COEFFICIENTS OF FOURTH ORDER FIT FOR LAYER 4
READ(20,93)N,NH4
READ(20,10) (O4(I),I=1,N)
READ(20,10) (T4(I),I=1,N)
W(4)=(WH4-WH3)*0.017453
IFM(W4)=EQ.0.0,1.0,W(4)=0.0001
DO 21 I=1,N
X4(I)=WH1*O4(I)
21 CONTINUE
CALL FITIV(T4,X4,A04,A14,A24,A34,A44)
C CALCULATE INTERVAL VELOCITY IN LAYER 4
K=4
CALL INTVEL(W,V,Z0,T0,A04,A14,A24,A34,X4,T4,N,H,SIGMA4,N5)
SUM4=SUMZ+Z0(4)
WRITE(61,L24)(V(4),SIGMA4,T0(4),H4,Z0(4),SUMZ,N5)
IF(Q2.EQ.0.G0) GO TO 101
Q2=Q2+1
C CALCULATE COEFFICIENTS OF FOURTH ORDER FIT FOR LAYER 5
READ(20,93)N,NH5
READ(20,10) (O5(I),I=1,N)
READ(20,10) (T5(I),I=1,N)
W(5)=(WH5-WH4)*0.017453
IFM(W5)=EQ.0.0,1.0,W(5)=0.0001
DO 22 I=1,N
X5(I)=WH1*O5(I)
22 CONTINUE
CALL FITIV(T5,X5,A05,A15,A25,A35,A45)
C CALCULATE INTERVAL VELOCITY IN LAYER 5
W5=W(5)/0.01753+W4
SUM2=SUM2+20(5)
WRITE(131,131,V(5),SIGMA5,T0(5),H5,Z0(5),SUM2,N5)
IF(Q2.EQ.1X)GO TO 101
Q2=Q2+1

C CALCULATE COEFFICIENTS OF FOURTH ORDER FIT FOR LAYER 6
READ(20,9)N,H6
READ(20,10)(D6(I),I=1,N)
READ(20,1)16(I),J=1,N
IF(W(6),E2,F=.W(6)=0.0301
W(6)=(H6-H5)*Q.01753
DO 23 I=1,N
W6(I)=W1*D6(I)
23 CONTINUE
CALL FIT(4,T6,X6,A06,A16,A26,A36,A46)
C CALCULATE INTERVAL VELOCITY IN LAYER 6
M=6
CALL INTEL(W,V,ZO,T0,A0,A1,A2,A3,A4,X,T,N,H,SIGMA,N5)
M=SUMZ+Z0(6)
WRITE(6,12)M,V(6),SIGMA,T0(6),W6,ZO(6),SUMZ,N5
GO TO 101
END
SUBROUTINE INTEL(W,V,Z0,T0,A0,A1,A2,A3,A4,X,T,N,H,SIGMA,N5)
DIMENSION Y((10,50)),T((10,50)),T0((10,50)),SUM((50)),SUM2((50))
REAL R(10,50),AIP(10,50),DIFF(50),ARG1(50),V(10),W(10)
DIMENSION ARG2(10,50),ARG3(10,50),DELX(10,50)
DIMENSION DELU((10,50)),THETA(10,50),DELX(10,50)
DIMENSION DELU(50),X(50),T(50),TRED(50),TRED(50)
DIMENSION TRTH(50)
INTEGER N,M,1,N,E,P,BL,CL,int,k,p
C CALCULATE MINIMUM REFLECTION TIME
M=H-1
IF(X(I),E2,0,I)GO TO 52
1 CONTINUE
DO 2 I=2,K
2 CONTINUE
DO 3 I=1,K
3 CONTINUE
DO 4 I=1,K
4 CONTINUE
DO 5 I=1,K
5 CONTINUE
DO 6 I=1,K
6 CONTINUE
DO 7 I=1,K
7 CONTINUE
DO 8 I=1,K
8 CONTINUE
DO 9 I=1,K
9 CONTINUE
DO 10 I=1,K
10 CONTINUE
DO 11 I=1,K
11 CONTINUE
DO 12 I=1,K
12 CONTINUE
DO 13 I=1,K
13 CONTINUE
DO 14 I=1,K
14 CONTINUE
DO 15 I=1,K
15 CONTINUE
DO 16 I=1,K
16 CONTINUE
DO 17 I=1,K
17 CONTINUE
DO 18 I=1,K
18 CONTINUE
DO 19 I=1,K
19 CONTINUE
DO 20 I=1,K
20 CONTINUE
DO 21 I=1,K
21 CONTINUE
DO 22 I=1,K
22 CONTINUE
DO 23 I=1,K
23 CONTINUE
DO 24 I=1,K
24 CONTINUE
DO 25 I=1,K
25 CONTINUE
DO 26 I=1,K
26 CONTINUE
DO 27 I=1,K
27 CONTINUE
DO 28 I=1,K
28 CONTINUE
DO 29 I=1,K
29 CONTINUE
DO 30 I=1,K
30 CONTINUE
DO 31 I=1,K
31 CONTINUE
DO 32 I=1,K
32 CONTINUE
DO 33 I=1,K
33 CONTINUE
DO 34 I=1,K
34 CONTINUE
DO 35 I=1,K
35 CONTINUE
DO 36 I=1,K
36 CONTINUE
DO 37 I=1,K
37 CONTINUE
DO 38 I=1,K
38 CONTINUE
DO 39 I=1,K
39 CONTINUE
DO 40 I=1,K
40 CONTINUE
DO 41 I=1,K
41 CONTINUE
DO 42 I=1,K
42 CONTINUE
DO 43 I=1,K
43 CONTINUE
DO 44 I=1,K
44 CONTINUE
DO 45 I=1,K
45 CONTINUE
DO 46 I=1,K
46 CONTINUE
DO 47 I=1,K
47 CONTINUE
DO 48 I=1,K
48 CONTINUE
DO 49 I=1,K
49 CONTINUE
DO 50 I=1,K
50 CONTINUE
DO 51 I=1,K
51 CONTINUE
DO 52 I=1,K
52 CONTINUE
DO 53 I=1,K
53 CONTINUE
DO 54 I=1,K
54 CONTINUE
DO 55 I=1,K
55 CONTINUE
DO 56 I=1,K
56 CONTINUE
DO 57 I=1,K
57 CONTINUE
DO 58 I=1,K
58 CONTINUE
DO 59 I=1,K
59 CONTINUE
DO 60 I=1,K
60 CONTINUE
DO 61 I=1,K
61 CONTINUE
DO 62 I=1,K
62 CONTINUE
DO 63 I=1,K
63 CONTINUE
DO 64 I=1,K
64 CONTINUE
DO 65 I=1,K
65 CONTINUE
DO 66 I=1,K
66 CONTINUE
DO 67 I=1,K
67 CONTINUE
DO 68 I=1,K
68 CONTINUE
DO 69 I=1,K
69 CONTINUE
DO 70 I=1,K
70 CONTINUE
DO 71 I=1,K
71 CONTINUE
DO 72 I=1,K
72 CONTINUE
DO 73 I=1,K
73 CONTINUE
DO 74 I=1,K
74 CONTINUE
DO 75 I=1,K
75 CONTINUE
DO 76 I=1,K
76 CONTINUE
DO 77 I=1,K
77 CONTINUE
DO 78 I=1,K
78 CONTINUE
DO 79 I=1,K
79 CONTINUE
DO 80 I=1,K
80 CONTINUE
DO 81 I=1,K
81 CONTINUE
DO 82 I=1,K
82 CONTINUE
DO 83 I=1,K
83 CONTINUE
DO 84 I=1,K
84 CONTINUE
DO 85 I=1,K
85 CONTINUE
DO 86 I=1,K
86 CONTINUE
DO 87 I=1,K
87 CONTINUE
DO 88 I=1,K
88 CONTINUE
DO 89 I=1,K
89 CONTINUE
DO 90 I=1,K
90 CONTINUE
DO 91 I=1,K
91 CONTINUE
DO 92 I=1,K
92 CONTINUE
DO 93 I=1,K
93 CONTINUE
DO 94 I=1,K
94 CONTINUE
DO 95 I=1,K
95 CONTINUE
DO 96 I=1,K
96 CONTINUE
DO 97 I=1,K
97 CONTINUE
DO 98 I=1,K
98 CONTINUE
DO 99 I=1,K
99 CONTINUE
DO 100 I=1,K
100 CONTINUE
GO TO 5
6 R(I,J)=ASINF(ARG2(I,J)*W(I))
5 CONTINUE
4 CONTINUE
DO 7 I=2,K
AIP(H,I)=R(H,I)
7 CONTINUE
DO 9 I=2,K
B=I+1
C=B+1
DO 10 A=1,K
IF(TREDQ(J).EQ.0.)GO TO 3
ARGI(B,J)={V(B)/V(C)}*SIN(AIP(C,J)*W(C))
10 CONTINUE
IF(ARG3(B,J).LE.1.)GO TO 3
TREDQ(J)=XRED(H,J)=0.
L=L+1
GO TO 8
9 AIP(J,J) = ASINF(ARG3(B,J))
9 CONTINUE
7 CONTINUE
10 IF(IPM.EQ.2) GO TO 12
11 I = 2, M
12 IF(TREOSQ(I).EQ.0.) GO TO 11
13 CONTINUE
14 IF(TREOSQ(I).EQ.0.) GO TO 14
15 R := I = 2, K
16 CONTINUE
17 IF(TREOSQ(J).EQ.0.) GO TO 17
18 YYP(1,J) = 0
19 CONTINUE
20 IF(TREOSQ(1).EQ.0.) GO TO 20
21 DELX(1,J) = DELX(1,J) + TANF(AIP(1,I) - TANF(THETA(M)))
22 CONTINUE
23 IF(TREOSQ(J).EQ.0.) GO TO 23
24 SUMTA(I) = SUMTA(I) + TA(1,I)
25 CONTINUE
26 IF(TREOSQ(I).EQ.0.) GO TO 26
27 SUMTA(J) = SUMTA(J) + TA(1,J)
28 CONTINUE
29 IF(TREOSQ(J).EQ.0.) GO TO 29
30 SUMTA(1) = SUMTA(1) + TA(1,1)
31 CONTINUE
32 IF(TREOSQ(I).EQ.0.) GO TO 32
33 SUMTA(1) = SUMTA(1) + TA(1,1)
34 CONTINUE
35 IF(TREOSQ(J).EQ.0.) GO TO 35
36 SUMTA(I) = SUMTA(I) + TA(1,1)
37 CONTINUE
38 IF(TREOSQ(I).EQ.0.) GO TO 38
39 SUMTA(1) = SUMTA(1) + TA(1,1)
40 CONTINUE
41 IF(TREOSQ(J).EQ.0.) GO TO 41
42 SUMTA(I) = SUMTA(I) + TA(1,1)
43 CONTINUE
44 IF(TREOSQ(I).EQ.0.) GO TO 44
45 SUMTA(1) = SUMTA(1) + TA(1,1)
46 CONTINUE
47 IF(TREOSQ(J).EQ.0.) GO TO 47
48 SUMTA(I) = SUMTA(I) + TA(1,1)
49 CONTINUE
50 IF(TREOSQ(I).EQ.0.) GO TO 50
51 SUMTA(1) = SUMTA(1) + TA(1,1)
52 CONTINUE
53 IF(TREOSQ(J).EQ.0.) GO TO 53
54 SUMTA(I) = SUMTA(I) + TA(1,1)
55 CONTINUE
56 IF(TREOSQ(I).EQ.0.) GO TO 56
57 SUMTA(1) = SUMTA(1) + TA(1,1)
58 CONTINUE
59 IF(TREOSQ(J).EQ.0.) GO TO 59
60 SUMTA(I) = SUMTA(I) + TA(1,1)
61 CONTINUE
62 IF(TREOSQ(I).EQ.0.) GO TO 62
63 SUMTA(1) = SUMTA(1) + TA(1,1)
64 CONTINUE
65 IF(TREOSQ(J).EQ.0.) GO TO 65
66 SUMTA(I) = SUMTA(I) + TA(1,1)
67 CONTINUE
68 IF(TREOSQ(I).EQ.0.) GO TO 68
69 SUMTA(1) = SUMTA(1) + TA(1,1)
70 CONTINUE
71 IF(TREOSQ(J).EQ.0.) GO TO 71
72 SUMTA(I) = SUMTA(I) + TA(1,1)
73 CONTINUE
74 IF(TREOSQ(I).EQ.0.) GO TO 74
75 SUMTA(1) = SUMTA(1) + TA(1,1)
76 CONTINUE
77 IF(TREOSQ(J).EQ.0.) GO TO 77
78 SUMTA(I) = SUMTA(I) + TA(1,1)
79 CONTINUE
80 IF(TREOSQ(I).EQ.0.) GO TO 80
81 SUMTA(1) = SUMTA(1) + TA(1,1)
82 CONTINUE
83 IF(TREOSQ(J).EQ.0.) GO TO 83
84 SUMTA(I) = SUMTA(I) + TA(1,1)
85 CONTINUE
86 IF(TREOSQ(I).EQ.0.) GO TO 86
87 SUMTA(1) = SUMTA(1) + TA(1,1)
88 CONTINUE
89 IF(TREOSQ(J).EQ.0.) GO TO 89
90 SUMTA(I) = SUMTA(I) + TA(1,1)
27  DO 28 I=2,X
     IF(TRED(I).EQ.J) GO TO 28
     TRED(I) = T(I) - SUMT(I) - SUM3(I)
     COR(I) = T4TH(I)*TRTH(I) - 2.*TRTH(I)*SIN(W(I))*XRED(H,I)/V(H)
     TREDQ(I) = TRED(I)*TRED(I) - COR(I)
     IF(TREDQ(I).LT.0.) TREDQ(I) = XRED(H,I) = 0.
     IF(XRED(H,I).LT.0.) XRED(H,I) = TREDQ(I) = 0.
     IF(TREDQ(I).EQ.J) J1 = J1 + 1
     CONTINUE
C CALCULATE TRIAL VELOCITY, STANDARD DEVIATION, DIP BY LINEAR LEAST SQUARES

C

SUM1 = SUM2 = SUM3 = SUM4 = SUM5 = 1
DO 29 I = 2, K
SUM1 = SUM1 + XRED(H, I) * XRED(H, I) * TREDSQ(I)
SUM2 = SUM2 + XRED(H, I) * TRED(I)
SUM3 = SUM3 + TREDSQ(I)
SUM4 = SUM4 + XRED(H, I) ** 4
29 CONTINUE
NS = N - 2 - J1 - J5
IF (NS.EQ.1) GO TO 67
TEMP = NS * SUM4 - SUM2 * SUM2
IF (TEMP .GE. 0) GO TO 65
67 X6 = 4 * XALL
X7 = XRED
X8 = XGEO
X9 = XH
XN = XMO T
X14 = XRED
X15 = XZERO
X16 = XRED
WRITE (61, 102) X6, X7, X8, X9, X10, X11, X12
102 FORMAT (1MO, 744)
VM1 = VM1
GO TO 59
65 SLOPEV = (NS * SUM1 - SUM2 * SUM3) / TEMP
VP = SQRT (1. / SLOPEV)
1111
IF (QL.GT.5) GO TO 30
IF (Q1.EQ.1) GO TO 50
W(H) = ATAN ((TANF (W(H)) * (VP / V(H)))
50 VDIFF = ABS (VP - V(H))
IF (QL.EQ.1) GO TO 55
IF (VDIFF .GE. GO) GO TO 37
56 VDIFF = VDIFF
IF (VDIFF .LE. 0.01) GO TO 33
VM1 = VP
GO TO 31
30 VM1 = VP
KINT = (SUM1 - SLOPEV * SUM2) / NS
DO 34 I = 2, K
DELYSQ(I) = (TREDSQ(I) - (XINT * SLOPEV * XRED(H, I) * XRED(H, I))) ** 2
34 CONTINUE
SY = SQRT (SUM5 / (NS - 2))
SB = SY / SQRT (NS / (NS * SUM4 = SUM2 * SUM2))
SIMAV = V(H) * (0.5) * (SB / SLOPEV)
GO TO 59
57 WRITE (61, 58)
58 FORMAT (1H + 28H NON CONVERGENT)
59 RETURN
END

SUBROUTINE FIT(N, T, X, A0, A1, A2, A3, A4)
DIMENSION B(5, 6, 1(5), X(5))
SUM1 = SUM2 = SUM3 = SUM4 = SUM5 = SUM6 = SUM7 = SUM8 = SUM9 = 0.
SUM10 = SUM11 = SUM12 = SUM13 = 0.
DO 1 I = 1, N
SUM1 = SUM1 + X(I)
1 CONTINUE
B(I, 1) = B(I, 1) = SUM1
JO 2 I = 1, 4
SUM2 = SUM2 + X(I) ** 2
2 CONTINUE
B(I, 2) = B(I, 2) = SUM2
B(I, 3) = B(I, 3) = X(I)
B(I, 4) = B(I, 4) = X(I) ** 3
2 CONTINUE
B(1,3)=B(2,2)=B(3,1)=SUM2
DO 3 I=1,4
SUM3=SUM3+X(I)**3
3 CONTINUE
B(1,4)=B(2,3)=B(3,2)=B(4,1)=SUM3
DO 4 I=1,4
SUM4=SUM4+X(I)**4
4 CONTINUE
B(1,5)=B(2,4)=B(3,3)=B(4,2)=B(5,1)=SUM4
DO 5 I=1,4
SUM5=SUM5+X(I)**5
5 CONTINUE
B(2,5)=B(3,4)=B(4,3)=B(5,2)=SUM5
DO 6 I=1,4
SUM6=SUM6+X(I)**6
6 CONTINUE
B1(3,5)=B(3,4)=B(5,3)=SUM6
00441
DO 7 I=1,N
00442
SUM7=SUM7*SUM7*T(I)**2
00443
7 CONTINUE
00444
B1(4,5)=B(4,4)=SUM7
00445
DO 8 I=1,N
00446
SUM8=SUM8*SUM8*T(I)**2
00447
8 CONTINUE
00448
d(5,5)=SUM8
00449
DO 9 I=1,N
00450
SUM9=SUM9+T(I)
00451
9 CONTINUE
00452
B1(1,6)=SUM9
00453
DO 10 I=1,N
00454
SUM10=SUM10+T(I)*T(I)
00455
10 CONTINUE
00456
B2(2,6)=SUM10
00457
DO 11 I=1,N
00458
SUM11=SUM11+T(I)*T(I)*T(I)
00459
11 CONTINUE
00460
B3(3,6)=SUM11
00461
DO 12 I=1,N
00462
SUM12=SUM12+T(I)*T(I)*T(I)*T(I)
00463
12 CONTINUE
00464
B4(4,6)=SUM12
00465
DO 13 I=1,N
00466
SUM13=SUM13+T(I)*T(I)*T(I)*T(I)*T(I)
00467
13 CONTINUE
00468
B5(5,6)=SUM13
00469
DO 14 I=1,N
00470
JBMAX=6+K
00471
DO 15 J=1,JBMAX
00472
J=7-J8
00473
15 CONTINUE
00474
B6(6,6)=B(6,6)
00475
DO 16 I=1,N
00476
JBMAX
00477
16 CONTINUE
00478
RETURN
00479
END
00480
SUBROUTINE FIT1(N,X,T,T0)
00481
DIMENSION X(50),T(50)
00482
SUM1=SUM2=SUM3=SUM4=0
00483
DO 50 I=1,N
00484
SUM1=SUM1*T(I)**4
00485
SUM2=SUM2*T(I)**4*T(I)**4
00486
SUM3=SUM3*T(I)**4*T(I)**4*T(I)**4
00487
SUM4=SUM4*T(I)**4*T(I)**4*T(I)**4
00488
SLOPE=(N*SUM1-SUM2*SUM3)/(N*SUM4-SUM2-SUM3)
00489
T3=SQR(SUM3+SLOPE*SUM2)/V1
00490
RETURN
00491
END
PROGRAM SIGPSKW
COMPUTES POWER SPECTRUM OF INPUT FILE
DIMENSION Z(1824)
COMPLEX YYC(1024)
COMMON/KULBI/YYC
COMMON/KULB4/LBLF(3),Lbla(3)
EQUIVALENCE (YYC(1),Z(1))
DATA LBLF/6H FREQ /
DATA LBLA/6H AMPL /
100 FORMAT(4E20.8)
CALL INFILE(1)
ACCEPT "NO. OF DATA POINTS IN FFT=",NT
NX=NT/2+1
ACCEPT "READ ? DATA POINTS ",NR
5 DO I=1,NX
YYC(I)=(0.,0.)
READ(1,100)(YYC(I),I=1,NR)
10 DO 10 I=1,NR
10 Z(I)=REAL(YYC(I)*CONJG(YYC(I)))
20 YMAX=0.
20 DO 20 I=1,NR
AMP=ABS(Z(I))
23 IF(AMP.LT.YMAX)GO TO 20
24 YMAX=AMP
25 CONTINUE
26 DO 30 I=1,NR
30 Z(I)=Z(I)/YMAX
28 NY=NR-1
29 DO 40 I=NY,NX
30 Z(I)=0.
31 YMAX=1.0
32 CALL PLT3W(NX,LBLF,LBLA,YMAX)
33 END
SNPREPARE

PROGRAM SNPREPARE

READS OUTPUT OF STACKW AND PREPARES IT FOR STACKCM
BY REMOVING MEAN AND/OR WINDOWING NOISE
THEN FOURIER TRANSFORMING BOTH, STORING NOISE FIRST
AND SOURCE FUNCTION SECOND ON DATA FILE "FFTSN"
READS OUTPUT OF STACKW AND PREPARES IT FOR STACKCM
BY REMOVING MEAN AND/OR WINDOWING NOISE
THEN FOURIER TRANSFORMING BOTH, STORING NOISE FIRST
AND SOURCE FUNCTION SECOND ON DATA FILE "FFTSN"

COMPLEX YYC(1024), YYN(5)
COMMON/KWLB1/YYC
COMMON/KWLB2/YYN
COMMON/KWLB4/LBLT(3), LBLA(3), LBLF(3)
EQUIVALENCE (YYC(313), YYC(1)), (YYC(1), ZZ(1)), (YYN(1), DMP(1))
DATA LBLT/6H TIME /
DATA LBLA/6H AMPL /
DATA LBLF/6H FREQ /

FORMAT(S10)
FORMAT(1E13.6)
FORMAT(IH.1OE13.6)
READ(*,100)FILE(1)
CALL FOPEN(1, FILE)
ACCEPT "NO. OF DATA POINTS IN NOISE (MULTIPLE OF 10)=", NN
ACCEPT "NO. OF DATA POINTS IN SOURCE FUNCTION=", NS
ACCEPT "REMOVE MEAN? YES=1, NO=0 ", NMEAN
ACCEPT "WINDOW? YES=1, NO=0 ", NWIN
I=0
NT=1024
NX=NT/2+1
CALL FOPEN(2, "NOISE")
CALL FOPEN(3, "FFTSN")
IN=0
5 IN=IN+1
IF(IN.EQ.2) NN=NS
DO 20 IN=1, NT
20 YYC(I)=<0.0,0.)
IF(IN.EQ.1)GO TO 40
READ(*,110)(YY(I), I=1, NS)
GO TO 41
40 READ(2,110)(YY(I), I=1, NN)
HDUMP=1024-NN
41 READ(2,110)(ZZ(I), I=1, HDUMP)
DO 10 I=1, 1024
10 ZZ(I)=0.
42 IF(NMEAN.EQ.0)GO TO 29
SUM=0.
43 DO 26 I=1, NN
44 SUM=SUM+YY(I)
45 YMEAN=SUM/NN
46 DO 28 I=1, NN
47 YY(I)=YY(I)-YMEAN
48 IF(NWIN.EQ.0)GO TO 31
49 CALL WHDBW(NH)
50 DO 30 I=1, NN
51 YYN(I)=YY(I)
52 IF(IN.EQ.2)GO TO 50

SNPREPARE 3/21/78 16:32:39 PAGE 2
GO TO 5
CALL RESET
END
Program STACKW plots successive traces of digitized streamer data from magnetic tape and stacks the direct arrival for use as a source signature. Reads A to D data that has been digitized at record lengths of 1024 points. A noise sample of 1024 points is stored on disk. Input is conversational. The output of program A2DPl is inspected to determine the starting point and the proper shotbreak search test level. All A to D tape reading programs run on the Data General Nova 1200 computer.
INIT MSG
RUN
STACK
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
FILE NAME OF STACKED SOURCE FUNCTION
SVE1
CHANNEL NUMBER TO BE PLOTTED AND NOISE STORED=3
CHANNEL NO. FOR STACKING=2
NO. OF TRACES DESIRED=3
NO. OF SECONDS SPACED OVER=52
STARTING ACCUMULATED TIME=0.
SHOT BREAK TEST AMPL.=630.
NO. OF SECONDS BETWEEN SHOTS=8
SKIP OVER # SHOTS BETWEEN TRACES=0
NO. OF POINTS IN SOURCE FUNCTION=430
NO. OF SECS OF DATA TO BE PLOTTED=1.
READ EVERY ? DATA POINT=2
ATTENUATION FACTOR=.0003
NOISE SAMPLE BEFORE START=3, AFTER END=1
CAUTION-AUTOMATICALLY READS 1324 POINTS
DELAY ORIGIN OF PLOT:
FROM START OF BUBBLE PULSE? YES=1, NO=0
AIRGUN-STREAMER LIST, (METERS)=99.2
TWO WAY TRAVEL TIME TO BOTTOM=4.1

3.80600E 3
3.95500E 3
3.85500E 3
3.77900E 3
3.88900E 3
3.92200E 3
3.83300E 3
3.99300E 3
3.94700E 3
3.86500E 3
3.96500E 3
3.13220E 4
3.79500E 3
3.77200E 3
3.89200E 3
3.75500E 3
3.86600E 3
3.85900E 3
3.91500E 3
3.83200E 3
3.96300E 3
3.98300E 3
3.61600E 3
3.79500E 3
3.65300E 3
3.77900E 3
3.93400E 3
3.94000E 3
3.65700E 3
3.85500E 3

TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
DIGITIZING INTERVAL=.992
LENGTH OF HORIZONTAL AXIS IN INCHES=13.
STOP
Flow Chart for Program STACKW

START

Type "TURN ON PLOTTER" on Teletype

TURN ON PLOTTER MANUALLY

Conversational Teletype Input

READ MAG TAPE SPACE OVER DATA

Conversational Teletype Input

STORE NOISE SAMPLE NOW?

NO

S

yes
STORE NOISE SAMPLE (1024 points)

Conv. Teletype Input

Activate Plotter:
Trace Scales and Label

Initialize Array:
Y(1) = 0.
Set NPASS = 0.

NPASS = NPASS + 1

IF

NPASS = 1
or NPASS > NTACE

READ MAG TAPE
AND SKIP DATA

YES

NO

Initialize Variables
20

STORE
NOISE
SAMPLE
ARRAY

Plot
Noise
Sample

IF
# points \( \not\geq \) 1024

NO

23

ACCUMULATE
TIME SINCE
SHOTBREAK

STACK
Source Signature
DATA VALUES

IF
ACCUMULATED TIME \( \geq \) DESIRED DELAY

yes

NO

PLOT
SIGNAL

50

NO
20

Yes

Accumulated time < desired plot length?

No

Yes

# traces < # traces desired?

NO

Average Stacked Source Signature

Store Stacked Source Sign on Disk

Determine max Value in Source Signature

CALL PLTW

PLOT DATA
IF Noise sample was taken at beginning

NO

STORE NOISE SAMPLE

S0

S1

CLOSE FILES

END
PROGRAM STACK

THIS PROGRAM PLOTS A SEQUENCE OF DATA FROM MAC TAPE AND STACKS THE SOURCE FUNCTION FROM CHANNEL 2.

DIMENSION IBUF(4, 1024), IFILE(6)
COMMON/KWL81/Y(1024)
COMMON/KWL84/LBLT(7), LBLA(3)
DATA LBLT/14HTIME (SECONDS)/
DATA LBLA/4H AMPL /

TYPE "TURN ON PLOTTER"
PAUSE
CALL MTODP1, "MT0. 10.0. IER"
CALL FOPEP6, "SPLT"
CALL INIT(6. 100. 0. 0. 0. 0. 0.)
100 FORMAT(1H16E13.G)
110 FORMAT(S10)

TO TO "FILE NAME OF STACKED SOURCE FUNCTION"
READ(11, 110)IFILE(1)
CALL FOPEP2, IFILE
CALL PLOT(1. 0. -3)

ACCEPT "CHANNEL NUMBER TO BE PLOTTED AND NOISE STORED =", NCH
ACCEPT "CHANNEL NO. FOR STACKING =", NS
ACCEPT "NO. OF TRACES DESIRED =", NSC
ACCEPT "STARTING ACCUMULATED TIME =", SAC
ACCEPT "SHOT BREAK TEST AMP. =", TST

CALL MTDIO(1. 0. 1824, IER)
IF(K.EQ.0.)GO TO 2
K=K+1
GO TO 1

2 ACCEPT "NO. OF SECONDS BETWEEN SHOTS =", NSHOT
ACCEPT "SKIP OVER ? SHOTS BETWEEN TRACES =", NSKIP
ACCEPT "NO. OF POINTS IN SOURCE FUNCTION =", NS
ACCEPT "NO. OF SECS OF DATA TO BE PLOTTED =", SED
ACCEPT "READ EVERY ? DATA POINT =", NDAT
ACCEPT "ATTENUATION FACTOR =", AFFE
ACCEPT "NOISE SAMPLE BEFORE START =", NSB
ACCEPT "READ EVERY ? DATA POINT =", NDAT

IF(NBE.EQ.0.)GO TO 9

40 TYPE "CAUTION-AUTOMATICALLY READS 1824 POINTS"
41 JJB=0
42 X=0.

9 CALL MTDIO(1. 0. 1824, IER)
DO 11 I=1, 1024, HDAT
JJB=JJB+1
Y(JJB)=IBUF(NCH, I)
X=X+.02
YY=Y(JJB)*AFFE
CALL PLOT(X, YY, 2)
60 IF(JJB.EQ.1824)GO TO 12
11 CONTINUE
GO TO 9
12 WRITE(3, 100)(Y(I), I=1, 1024)
13 CALL PLOT(0., 1. -.3)
8 TYPE "DELAY ORIGIN OF PLOT"
ACCEPT "FROM START OF BUBBLE PULSE? YES=1, NO=0 ", NBEL
7 IF(NBEL.EQ.0.)GO TO 3
ACCEPT "ARGUM-STREAMER DIST. (METERS) =", X
STACK

ACCEPT 'TWO WAY TRAVEL TIME TO BOTTOM=*',TWT

V=1500.

ZT=TWT+V/2.

SDELAY=2.*SGRT(X+X/4.+ZT*ZT)/V-.23*X/V

GO TO 4

SDELAY=0.

4 SXY=SECD+SDELAY

DX=SECD/20.

CALL AXIS(0.,0.,LBLT,-14,20,0.,0.,DX,3)

CALL PLOT(0.,1.,-3)

CALL SYMBOL(-.3,1.,0.,14,LBLT,.98,.8,14)

CALL PLOT(0.,0.,3)

NDMP=NSkip*NShot

DO 6 I=1,1024

6 Y(I)=0.

MPASS=0

7 IF(MPASS.EQ.0)GO TO 15

IF(MPASS.GT.NTAC)GO TO 15

IF(U4DMP.EQ.0)GO TO 1015

K=1

SACUM=SACCCUM+NDMP*1.024

5 CALL MDIO(1.0,IBUF,ISTS,IER,IVDCT)

IF(K.EQ.NDMP)GO TO 15

K=K+1

GO TO 5

15 IFLAG=1

8 X=9.8

9 J=1

10 IJK=0

SKP=0.

20 CALL MDIO(1.0,IBUF,ISTS,IER,IVDCT)

DO 20 I=1,1.1024,NDAT

IF(IFLAG.EQ.1)GO TO 25

SKP=SKP+0.801*NDAT

YY=IBUF(HSTK,1)

IF(ABS(YY).LT.TEST)GO TO 20

TYPE YY

IF(MPASS.EQ.0)GO TO 21

SNUM=SACCCUM+SDELAY

IF(MPASS.GT.NTAC)SNUM=SACCCUM

CALL NUMBER(0.,-3.,14.,SNUM,98,9.3)

CALL PLOT(0.,0.,3)

193 IFLAG=1

25 J=J+1

105 IF(NBE.EQ.0)GO TO 23

106 IF(MPASS.GT.NTAC)GO TO 23

107 IJK=IJK+1

108 YY(IJK)=IBUF(NCH,1)

109 YY=IBUF(NCH,1)*ATTEN

110 X=X+0.92

111 CALL PLOT(X,YY,2)

112 IF(IJK.GE.1024)GO TO 50

113 GO TO 20

114 J=J-(J-1)*NBAT

115 SECD=FLOAT(JJ)*0.001

116 IF(J.GT.NS)GO TO 26
Y(J)+Y(J)+FLOAT(IBUF(NSTK,I))

IF(SCD LT SDELAY) GO TO 20

YY=IBUF(NCH,I)*ATTEN

CALL PLOT(X,YY,2)

IF(SCD GE SXY) GO TO 18

CONTINUE GO TO 22

CALL PLOT(0.,1.,-3)

SACCUM=SACCUM+SCD+FLOAT(1825-I)*0.001

IF(NPASS LT NTRACE) GO TO 7

DO 30 I=1,NS

Y(I)=Y(I)+NPASS WRITE(2,100)(Y(I),I=1,NS)

YMAX=0.0 DO 40 I=1,NS

AMP=ABS(Y(I))

IF(AMP LT YMAX) GO TO 40

YMAX=AMP

CONTINUE

CALL PLTW(NS,LBLT,LBLA,YMAX)

IF(HBE.EQ.0)GO TO 51

TYPE "RESET PLOTTER* PAUSE

GO TO 7

CALL PLOT(0.,0.,3)

WRITE(3,100)(Y(I),I=1,1824)

CALL FCLOS(2)

CALL FCLOS(3)

END
SUBROUTINE PLTW(NN,LBLX,LBLY,YMAX)

DIMENSION LBLX(7),LBLY(3)

COMMON/KULB1/YY(1024)

TYPE "TURN ON PLOTTER"

CALL INITIAL(6.100.0.0.0.0)

CALL PLOT(1.,0.0.-3)

YMIN=-YMAX

DY=2.*YMAX/8.

CALL AXIS(0.,0.,LBLY,6.8.,90.,YMIN,DY,2)

ACCEPT "DIGITIZING INTERVAL=",DINTX

ACCEPT "LENGTH OF HORIZONTAL AXIS IN INCHES=",XLENGTH

XMAX=DINTX*NN

DX=XMAX/NN

XMIN=0.

CALL AXIS(0.,0.,LBLX,-14.,XLENGTH,0.,XMIN,DX,3)

XX=DINTX/DX

YX=YY(I)/BY+4.

CALL PLOT(XX,YX,3)

IPEN=2

DO 10 I=1,NN

XX=DINTX*I/DX

YX=YY(I)/BY+4.

CALL PLOT(XX,YX,IPEN)

10 CONTINUE

YLAST=4.

CALL PLOT(XX,YLAST,3)

CALL PLOT(0.,YLAST,2)

CALL PLOT(0.,0.,3)

RETURN

END
Program STKDCN reads digitized streamer or vertical profiler data from magnetic tape and swaps with Program DCNIOW for source signature deconvolution.

Conversational Input

Save transfer functions?: This option allows storage of the Fourier transform at various steps in the processing of selected traces.

% Nyquist for cos taper =: width of cosine taper on frequency domain window of bandpass filter.

Low-high frequency cutoffs in % Nyquist: location of the points in frequency at which amplification factor is zero for the bandpass or notch filters.

Window?: option for windowing time domain trace before processing.

Subtract noise?: option to subtract noise spectrum from signal spectrum in cases where large amplitude coherent noise is present.

No. of secs of data to be deconvolved =: must be less than 2.024 secs.

Attenuation factor =: amplification factor for output plotted trace; a number greater than 1 will give a peak to peak amplitude greater than 1 inch.

Apply water surface ghost filter?: inverse ghost filter option.

Read delay times?: If yes, ghost delay times remain fixed at input values; if no, delay times are computed internally.

COS taper on ends of filter = ? % of Nyquist: determines truncation point of inverse ghost filter; larger values
truncation at smaller filter amplitudes.

Delay origin of plot from start of bubble pulse?: If no is opted, data will be read from the onset of the bubble pulse; if yes, delay is computed and data is read from approximately .2 sec prior to bottom reflected arrival.

Peak scan?: option for line printer output of all peaks with amplitude greater than the % rejection level; gives peak amplitudes and data point and time locations.

```
INIT MT3
R
STKDCN
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
STREAMER CHANNEL NO. = 3
NO. OF traces DESIRED = 17
NO. OF SECONDS SPACED OVER = 52
NO. OF SECONDS BETWEEN SHOTS = 6
CHANNEL FOR SHOT BREAK SEARCH = 2
SHOT BREAK TEST AMPL. LEVEL = 63.
SKIP OVER ? SHOTS BETWEEN TRACES = 2
SAVE TRANSFER FUNCTIONS? YES=1, NO=0
BANDPASS FILTER? YES=1, NO=0
% OF NYQUIST FOR COS TAPER = 4.
LOW FREQUENCY CUTOFF (ZERO) IS ? % OF NYQUIST = 6.
HIGH FREQUENCY CUTOFF (ZERO) IS ? % OF NYQUIST = 60.
NOTCH FILTERS YER=1, NO=0
NO. OF NOTCHES = 2
LOW-HIGH LOC. (% NYQ.) MAX OF 5 = 14, 16, 22, 26.
REMOVE MEAN? YES=1, NO=0
WINDOW? YES=1, NO=0
SUBTRACT NOISE? YES=1, NO=0
NO. OF SECS OF DATA TO BE DECONVOLVED = 1.45
READ EVERY ? DATA POINT = 2
ATTENUATION FACTOR = 1.
BANDPASS FILTER ONLY, NO DECONV.? YES=1, NO=0
DEPTH OF AIRGUN (METERS) = 8.9
STREAMER DEPTH (METERS) = 3.
RMS WAVE HEIGHT = 1.
DIST. FROM AIRGUN TO STREAMER MIDPOINT (N) = 99.2
DIP (DEG.) OF BOTTOM (<- DOWN FROM SOURCE TO REC.) = 3.
APPLY WATER SURFACE GHOST FILTER? YES=1, NO=0
DELAY ORIGIN OF PLOT
FROM START OF BUBBLE PULSE? YES=1, NO=0
TWO WAY TRAVEL TIME TO BOTTOM = 4.1
PEAK SCAN? YES=1, NO=0
% REJECTION LEVEL = 29.
STOP
R
RELEASE MT3
R
APPEND FD61 RD1 RD2 RD3 RD4 RD5
R
```
Flow Chart for Program STKDCN

START

MANUALLY TURN ON PLOTTER

Initialize Plotter
Place Pen at Origin

Conversational INPUT ON TELETYPewriter

READ MAG TAPe SKIP DATA

Conversational TELETYPewriter INPUT

Ghost Filter ?

YES

READ DELAY TIMES

YES

NO

COMPUTE SURFACE GHOST DELAY TIMES

12
256

12

DELAY ORIGIN OF PLOT?

yes

COMPUTE TIME DELAY FROM SHOT BREAK TO START OF TRACE (SDELAY)

SDELAY=0.

Put Axes and LABELS ON PLOT

NPASS = 0

7

NPASS = NPASS + 1

Initialize INTEGER STORAGE ARRAY TO ZERO
SWAP TO DCN1OW. SV
CALL FSWAP "DCN1OW. SV"
RETURN FROM SWAP

19

yes
N*PASS <# TRACES DESIRED?

no
END.
PROGRAM STKDCH

THIS PROGRAM PLOTS A SEQUENCE OF DATA FROM MAG TAPE AND SWAPS WITH DCH9V FOR DECONVOLUTION OF VERTICAL PROFILER DATA.

DIMENSION IBUF(4, 1824), IFILE(6)
COMMON IY(1024), H, NFILE, XLLH, XLL, NMEAN, NWAV, DX, DY, DL,
6DLT1, DLT2, DLT3, NGST, CG, MSCAN, PRJ, MDAT, MNBDP, HW, THET,
7SHETR, AST, SDELAY, HWRITE, NCHT, NCHMAX, FNCHT(10), NBD
8COMMON /KWL/4/LBL(7), LBLA(3)
9DATA LBLT'I4HTIME (SECONDS)' DATA LSLA/6H AMPL /

CALL MTOPO(1, "MT8 0", 0, IER)
CALL FOPEN(G, "$PLT")
CALL INITIAL(6, 100, 0, 0)
CALL PLOT(1, 0, -3)
K = 0,
XLL=0,
XLH=0.
20 ACCEPT "STREAMER CHANNEL NO. =", NCH
21 ACCEPT "NO. OF TRACES DESIRED="", NTRACE
22 ACCEPT "NO. OF SECONDS SPACED OVER="", NSEC
23 SACCUN$.
24 IF(K.EQ.HSEC)GO TO 2
25 IF(K.EQ.HSEC)GO TO 2
26 IF(K.EQ.HSEC)GO TO 2
27 K = K + 1
28 GO TO 1
29 2 ACCEPT "NO. OF SECONDS BETWEEN SHOTS="", NSHOT
30 ACCEPT "CHANNEL FOR SHOT BREAK SEARCH="",NSBS
31 ACCEPT "SHOT BREAK TEST AMP. LEVEL="", TEST
32 ACCEPT "SKIP OVER SHOTS BETWEEN TRACES ", NSKIP
33 ACCEPT "SAVE TRANSFER FUNCTIONS? YES=1, NO=0 ", HWRITE
34 ACCEPT "BANDPASS FILTER? YES=1, NO=0 ", NFIL
35 IF(NFIL.EQ.0)GO TO 8
36 ACCEPT "% OF NYQUIST FOR COS TAPER="", C
37 ACCEPT "LOW FREQUENCY CUTOFF (ZERO) IS % OF NYQUIST ", XLH
38 ACCEPT "HIGH FREQUENCY CUTOFF (ZERO) IS % OF NYQUIST ", XLL
39 ACCEPT "NOTCH FILTER? YES=1, NO=0 ", NCHT
40 IF(NCHT.EQ.0)GO TO 8
41 ACCEPT "NO. OF NOTCHES="", NCHMAX
42 NCH=NCHMAX+2
43 ACCEPT "LOW-HIGH LOC. (% NYQ.) MAX OF 5 ",(FNCHT(I)), 1=1, NCHT)
44 ACCEPT "REMOVE MEAN? YES=1, NO=0 ", HMEAN
45 ACCEPT "WINDOW? YES=1, NO=0 ", NWAV
46 ACCEPT "SUBTRACT NOISE? YES=1, NO=0 ", NBND
47 ACCEPT "NO. OF SECS OF DATA TO BE DECOHOLVED="", SEC3
48 ACCEPT "READ EVERY DATA POINT ", MDAT
49 ACCEPT "ATTENUATION FACTOR=", ATTE
50 ACCEPT "BANDPASS FILTER ONLY, NO DECON? YES=1, NO=0 ", NHBDP
51 IF(NHBDP.EQ.0)GO TO 10
52 ACCEPT "DEPTH OF AIRGUN (METERS)="", H
53 ACCEPT "STREAMER DEPTH (METERS)="", SH
54 ACCEPT "RMS WAVE HEIGHT="", HW
55 ACCEPT "DIST. FROM AIRGUN TO STREAMER MIDPOINT (M)="", XST
56 ACCEPT "DIP (DEG.) OF BOTTOM (- DOWN FROM SOURCE TO REC.)="", D
57 ACCEPT "APPLY WATER SURFACE GHOST FILTER? YES=1, NO=0 ", HWST
58 IF(HWST.EQ.0)GO TO 9
STKDCN 5/16/78 9:30:52 PAGE 2

ACCEPT "READ DELAY TIMES? YES=1, NO=0 ".HRD

ACCEPT "RECORDED SOURCE DELAY=",DLT1
ACCEPT "BOWING RAY DELAY=",DLT2
ACCEPT "DELAY AT RECEIVER=",DLT3
ACCEPT "COS TAPER ON ENDS OF FILTER=, % OF HUQUEST ",CG

IF(NRD.EQ.1)GO TO 11

ACCEPT "FROM START OF BUBBLE PULSE? YES=1, NO=0 ".HRDEL
ACCEPT "TWO WAY TRAVEL TIME TO BOTTOM=",TWT
ACCEPT "PEAK SCAN? YES=1, NO=0 ".HRSCAN
IF(HRSCAN.EQ.0)GO TO 16

TYPE "DELAY ORIGIN OF PLOT"

ACCEPT "TWO WAY TRAVEL TIME TO BOTTOM=",TWT
ACCEPT "FROM START OF BUBBLE PULSE? YES=1, NO=0 ".HRDEL
ACCEPT "TWO WAY TRAVEL TIME TO BOTTOM=",TWT

THET = -1.024

IF(NRD.EQ.1)GO TO 12

Z = ZT+0.25+XST/W
GO TO 4

SDELAY = 2.081

CALL AXIS(0.0,.14,1.0,24)

DO 6 1=1,1024

IY(I)=1

CALL PLOT(8.,0.,-3)

IF(NPASS.EQ.0)GO TO 15

K=1

SACCUM = SACCUM+HMDP*1.024

IF(HMDP.EQ.0)GO TO 15

K=K+1

GO TO 5

IFLAG = 0

X = 0.0

J = 0
CALL MTIO(1.0, IBUF, ISTS, IER, IMBCHT)
DO 20 I=1, 1024, NDAT
IF(IFLAG.EQ.1) GO TO 25
SKP=SKP+0.001*NDAT
YY=BUF(NBBS, I)
IF(YY.LT.TEST) GO TO 20
IF(NPASS.EQ.1) GO TO 21
SACCUM=SACCUM+SKP-0.002*NDAT
CALL NUMBER(-1.1, 1.1, 14, SDELAY, 0.0, 3)
XNUM=FLOAT(NPASS)
CALL NUMBER(-2.0, -3.0, 14, XNUM, .0, -1)
CALL PLOT(0.0, 0.0, 3)
IFLAG=1
J=J+1
JJ=J+NDAT
SCD=FLOAT(JJ)*0.001
IF(SCD.LT.SDELAY) GO TO 20
IJK=IJK+1
STORE=(IJK-1)*NDAT*0.001
IV=(IJK-1)*IBUF(NCH, I)
IF(IJK.EQ.1024) GO TO 20
IF(STORE.EQ.SECSD) GO TO 15
IFLAG=1
CONTINUE
GO TO 22
SACCUM=SACCUM+SCD+FLOAT(1025-I)*0.001
N=IJK
IF(NWRITE.EQ.0) GO TO 19
ACCEPT "SAVE TRANSFER FUNCTIONS THIS TRACE? YES=1, NO=0 ", NTRANS
IF(NTRANS.EQ.0) GO TO 40
CALL FSAP("BCH10W.5V")
IF(NPASS.LT.NTRACE) GO TO 7
END
Flow Chart for DCN10W and DCN14W

START

OPEN INPUT FILES

Initialize Complex Array to zero

Equate Input INTEGER ARRAY to REAL ARRAY

REMOVE MEAN?

yes

REMOVE MEAN

no

WINDOW

yes

CALL WND5W WINDOW with COSINE TAPER

no
31

EQUATE REAL ARRAY TO COMPLEX ARRAY

CALL FFT3W
FAST FOURIER TRANSFORM

SAVE TRANSFER FUNCTIONS?

NO

WRITE FFT OF SIGNAL?

CALL OUTFILE
(GIVE OUTPUT FILE NAME)

WRITE FILE ON DISK

CLOSE FILE

BANDPASS FILTER ONLY?

yes

80

NO
SUBTRACT NOISE?

SUBTRACT NOISE IN FREQUENCY DOMAIN FROM SIGNAL

SAVE TRANSFER FUNCTION?

WRITE FFT OF SIGNAL-NOISE?

CALL OUTFILE

WRITE FILE ON DISK

CLOSE FILE

DECONVOLVE BY DIVISION BY FFT OF SOURCE SIGNATURE
SAVE TRANSFER FUNCTION?

WRITE FFT OF DE-GHOSTED TRACE?

CALL OUTFILE

WRITE FILE ON DISK

CLOSE FILE

BANDPASS FILTER?

IF HIGH FREQ. CUTOFF = 100% Nyquist

CALL LPFKW LOWPASS FILTER

34

85
CALL FFT3W
FAST INVERSE
FOURIER
TRANSFORM

EQUATE REAL
ARRAY TO
REAL PART OF
COMPLEX ARRAY

FIND
MAX VALUE
IN REAL ARRAY

NORMALIZE
REAL ARRAY

PEAK SCAN?

YES

WRITE VALUES
ON LINE
PRINTER

PEAK SCAN

NO

CALL FBACK
(Return to Main
Program)

PLOT REAL
ARRAY

Return to STKDCN if DCN10W
or WARDCN if DCN14W

END
PROGRAM DCN10W (SWAPS FROM STKDCN)

SPECTRAL DECONVOLUTION AND TRANSFORMS BACK TO TIME

USED TO DECONVOLVE VERTICAL PROFILER DATA

FFT OF NOISE SUBTRACTED FROM FFT OF SIGNAL

BANDPASS FILTERS, WINDOWS AND REMOVES MEAN OF SIGNAL

SOURCE FUNCTION AND NOISE READ FROM DISK, SIGNAL FROM MAG TAPE

REMOVES WATER SURFACE GHOST BY FILTERING SOURCE FUNCTION

MULTIPLES BY INSTRUMENT TRANSFER FUNCTION IN FREQUENCY DOMAIN

DIMENSION (FILE(6),YYC(1024),ZZ(1024),DMP(10))

COMMON IY(1024),N,FILE(6),XLL,XLY,MIN,DX,DY,DLT

6DLT1,DLT2,DLT3,NGHOST,CG,NSCAN,PRJ,NDAT,NBNDF,H,W,THETS

6THETR,AST,DDELAY,WRITE,NOTCH,NOTCHMAX,NOTCH(10),MN

COMMON/XL1L1,YYC

COMMON/XL1L2/YYC

EQUALYCE (YYC(513),YYC(1)),(YYC(1),ZZ(1)),(YYC(1),DMP(1))

TOTAL TWO-WAY TIME=".F6.3," SEC"

FOR NWRITE.EQ.0 GO TO 31

ACCEPT WRITE FFT OF SIGNAL? YES/NO

WRITE FFT OF SIGNAL.

SUBTRACT FFT OF NOISE FROM SIGNAL

READ FFT OF NOISE FROM SIGNAL

CALL DCN10W
59: READ(1,110)(YYM(K),K=1,3)
60: J=1
61: 53 YYC(I)=YYC(I)-YYM(J)
62: CONTINUE
63: C DECONVOLVE BY DIVISION BY FFT OF SOURCE
64: IF(NWRITE.EQ.0)GO TO 55
65: ACCEPT "WRITE FFT OF SIGNAL-NOISE? ",NWR
66: IF(NWR.EQ.0)GO TO 55
67: CALL OUTFILE(3)
68: WRITE(3,130)(YYC(I),I=1,NK)
69: CALL FCLOSE(3)
70: 55 J=0
71: READ(1,110)(YYM(I),I=1,5)
72: DO 60 I=1,N
73: J=J+1
74: IF(J.LE.5)GO TO 62
75: READ(1,110)(YYM(K),K=1,5)
76: J=1
77: 62 IF(DMPZJ-I).EQ.E0.0)GO TO 63
78: YYC(I)=YYC(I)-YYM(J)
79: GO TO 60
80: 63 YYC(I)=(0,0)
81: 60 CONTINUE
82: C APPLY WATER SURFACE GHOST ELIMINATING FILTER
83: IF(NWRITE.EQ.0)GO TO 65
84: ACCEPT "WRITE FFT OF DECONVOLVED TRACE? ",NWR
85: IF(NWR.EQ.0)GO TO 65
86: CALL OUTFILE(3)
87: WRITE(3,130)(YYC(I),I=1,NK)
88: CALL FCLOSE(3)
89: 65 IF(NGHOST.EQ.0)GO TO 80
90: CALL VHOSTD(NT,DLT1,DLT2,DLT3,DLT,CG,HU,THETS,THETR,AST)
91: C APPLY BANDPASS FILTER IN THE FREQUENCY DOMAIN
92: IF(NWRITE.EQ.0)GO TO 80
93: ACCEPT "WRITE FFT OF DEHOSTED TRACE? ",NWR
94: IF(NWR.EQ.0)GO TO 80
95: CALL OUTFILE(3)
96: WRITE(3,130)(YYC(I),I=1,NK)
97: CALL FCLOSE(3)
98: 80 IF(NFILT.EQ.0)GO TO 85
99: IF(NLH.EQ.100)GO TO 34
100: CALL LPFKW(NT,C,XL)
101: 34 IF(XLM.EQ.0.)GO TO 36
102: CALL HFPKW(NT,C,XL)
103: 36 IF(NOTCH.EQ.0.)GO TO 85
104: NOTCH=0
105: 35 NOTCH=NOTCH+1
106: IF(NITCH.GT.NOTCHMAX)GO TO 85
107: XL=NOTCH(2*NOTCH-1)
108: XH=NOTCH(2*NOTCH)
109: CALL SMOTCH(NT,C,XL,XH)
110: GO TO 35
111: 85 IF(NWRITE.EQ.0)GO TO 86
112: ACCEPT "WRITE FFT OF BANDPASS FILT. AND DEHOSTED TRACE? ",NWR
113: IF(NWR.EQ.0)GO TO 86
114: CALL OUTFILE(3)
115: WRITE(3,130)(YYC(I),I=1,NK)
116: C INVERSE FOURIER TRANSFORM DECONVOLVED SERIES
CALL FFT3VC((I, 0))
DO 99 I=1, NX
XX(I)=REAL(YYC(I))
YMAX=0.
DO 99 I=1, NX
AMP=ABS(ZZ(I))
IF(AMP.LT.YMAX)GO TO 90
YMAX=AMP
CONTINUE
DO 90 I=1, NX
ZZ(I)=ZZ(I)/YMAX
IF(NSCAN.EQ.0)GO TO 58

PEAK SCAN
JFLAG=0
ZMAX=0.
XPRJ=PRJ/100.
DO 78 I=1, NX
TPEAK=ABS(ZZ(I))
IF(TPEAK.LT.XPRJ.AND.JFLAG.EQ.0)GO TO 75
IF(TPEAK.LT.XPRJ)GO TO 71
IF(TPEAK.LT.ZMAX)GO TO 75
ZMAX=TPEAK
IP=0
XPRJ=0.
DO 40 I=1, NX
CALL PLOT(X,Y,2)
CALL PLOT(X,Y,3)
END
SUBROUTINE WNSWM(N)
!
WINDOWS WITH COS TAPER OF 2% OF N
!
DIMENSION YYC(1024)
!
COMMON/KWLB1/YYC
!
EQUIVALENCE (YYC(513),D(1))
!
PI=3.141593
!
C=.02
!
L=IFIX(C*N)
!
THX=PI/L

!
L=L+1
!
DO 38 I=1,LQ
!
J=LQ-I
!
ARG=THX*FLOAT(J)
!
30 D(I)=D(I)*(1.+COS(ARG))/2.
!
NR=N-L
!
DO 40 I=NR,N
!
J=I-WR
!
ARG=THX*J
!
D(I)=D(I)*(1.+COS(ARG))/2.
!
40 CONTINUE
!
RETURN
!
END

SUBROUTINE FFT3W(LX,SIGNI)
!
COMPLEX CX(1024),CARG,CW,CTEMP
!
COMMON/KWLB1/CX
!
J=1

SC=SQRT(1./LX)
!
DO 5 I=1,LX
!
IF(I.GT.J)GO TO 2
!
CTEMP= CX(J)*SC
!
CX(J)=CX(J)*SC

!
CX(I)=CTEMP
!
6 M=LX/2
!
IF(J.LE.M)GO TO 5
!
J=-J-M
!
M=M/2
!
IF(M.GE.I)GO TO 3
!
5 J=J+M
!
L=1
!
ISTEP=2*L
!
DO 8 M=1,L
!
CARG= (0.,1.)*(3.141593*SIGNI*(M-1))/L
!
CW=EXP(CARG)
!
DO 8 I=M,LX,ISTEP
!
IPL=1+L
!
CTEMP=CW*CX(IPL)
!
CX(IPL)=CX(I)-CTEMP
!
8 CX(I)=CX(I)+CTEMP
!
L=ISTEP
!
IF(L.LT.LX)GO TO 6
!
RETURN
!
END
SUBROUTINE VHOST (MT, DELT1, DELT2, DELT3, DLT, CG, NW, THETS, THETR, 
AST)
COMPLEX YYC(TWOD), CARG, SLT, GFS
COMMON /KWLB1/ YYC
MX=MT/2+1
P=1.141593
W=(1.00)
DEL=2.*PI/<HT*DLT>
ARG1=DEL+DELT1
IF(CG.EQ.0.)CG=1.
LX=IFIX(CG*(CN-1)/100.)
TB=DELT2
AN=PI/2-THETS
LOOP=0
1
LOOP=LOOP+1
ARG2=DEL+DELT2
CARG=ARG2*TY/<0.,-1.>
X=SLTB*NY/<DELT*LY>
R=HW*SIN(ANGLE)/XLAMBDA
FACT=EXP(R)
GFS=1./<( ... 0. )-FACT*CEXP(CARG))
GFMB=ABS(GFS)
XR=REAL(GFS)
XI=REAL(geh-CONJG(GFS))<(0.,-.5))
DO 10 I=1,MX
CARG=ARG2*(I-1)*<(0.,-1.>)
10 IF(I.GT.1)GO TO 11
GFLT=GFS
20 GO TO 60
11 XLAMBDA=TY/<DELT*(I-1))
12 R=HW*SIN(ANGLE)/XLAMBDA
13 FACT=EXP(R)
14 GFLT=1./<( ... 0. )-FACT*CEXP(CARG))
15 GFMD=ABS(GFLT)
16 IF(GFMD.LT.GFMD)GO TO 60
17 X=REAL(geh-CONJG(GFLT))<(0.,-.5))
18 XIP=SIGN(X,XIT)
19 GFLT=CPLX(XR,XIP)
20 YYC(I)=YYC(I)*GFLT
21 IF(LOOP.EQ.2)GO TO 50
22 IF(DELT1.EQ.0.)GO TO 10
23 CARG=ARG1*(I-1)*<(0.,-1.>)
24 IF(I.GT.1)GO TO 20
25 YYC(I)<(0.,0.)
26 GO TO 10
27 R=HW*SIN(ANGLES)/XLAMBDA
28 FACT=EXP(R)
29 GFLT=<( ... 0. )-FACT*CEXP(CARG))
30 Y=(YCC(I)-YYC(I)*GFLT
31 IF(I.EQ.1)GO TO 18
32 IF(I.EQ.NX)GO TO 10
33 J=MT+I+2
34 XRGX=REAL(YYC(I))
35 XIGF=REAL((YYC(I)-CONJG(YYC(I)))*<(0.,-.5)))
36 Y=(YCC(J)-CRPLX(XRGF,XIGF)
37 CONTINUE
50 IF(LOOP.EQ.2)GO TO 70
VHOST
CTINUE
275
SUBROUTINE OUTFILE(N)
  DIMENSION NFILE(8)
  WRITE(10,1) N
  1 FORMAT('OUTFILE FOR CH '12,': ',2)
  2 FORMAT(S1)
  READ(11,2) NFILE(1)
  DELETE FILENAME FIRST TO KEEP GARBAGE OFF END
  BUT FIRST CHECK FOR RESERVED FILE NAMES
  IF(AQDCNFILE(N),EQ.922999K) CO TO 1B
  LFILE(L).EQ.'PT'AND. NFILE(2).EQ.'B,') GO TO 19
  ALSO NOT A TAPE FILE
  CALL DELETE(HFILE>
  CALL FCOPEN(H,NFILE)
  RETURN
  END

SUBROUTINE LPFKW(N.C,XLL)
  COMPLEX D(1024)
  COMMON/KWLBI/D
  PI=3.141593
  NX=N/2+1
  LA=NX-IFIX(XLL*NX/180.)
  L=IFEX(C*NX/180.)
  THX=PI/L
  LX=NX-LA
  LP=LQ-L
  DO 35 I=LP, LQ
    J=I-LP
    ARG=THX*FLOAT(J)
    D(I)=D(I)*(1.+COS(ARG))/2.
  CONTINUE
  NR=NX+LA
  MS=NR+L
  LC=LQ+1
  LD=NR-1
  DO 60 I=LC, LD
    60 D(I)=D(I)*(1.+COS(ARG))/2.
  CONTINUE
  RETURN
  END
SUBROUTINE HPFK(N, C, XLH)
COMPLEX D(1024)
PI=3.141593
NX=N/2+1
LH=IFIX(XLH*NX/100.)+1
L=IFIX(C*NX/100.)
THX=PI/L
LB=LH+L
DO 10 I=LB,LO
J=LB+I
ARG=THX*FLOAT(J)
10 D(I)=D(I)*(1.+COS(ARG))/2.
DO 20 I=1,LH
20 D(I)=(0.,0.)
LD=N-LH+2
LC=LB-L
DO 30 I=LC,LD
J=I-LC
ARG=THX*J
30 D(I)=D(I)*(1.+COS(ARG))/2.
DO 40 I=LB,N
40 D(I)=(0.,0.)
RETURN
END

SUBROUTINE SHQTCH(N, C, XL, XLH)
COMPLEX D(1024)
PI=3.141593
NX=NX-IFIX(XL*NX/100.)
L=NX-L
LH=IFIX(XLH*NX/100.)+1
THX=PI/L
LP=LO-L
LB=LH+L
DO 10 I=LB,LO
J=LB-I
LD=LD+1
ARG=THX*FLOAT(J)
10 D(I)=D(I)*(1.+COS(ARG))/2.
DO 20 I=1,LH
20 D(I)=D(I)*(1.+COS(ARG))/2.
NS=NX+LA+L+1
DO 30 I=LP,LQ
30 D(I)=D(I)*(1.+COS(ARG))/2.
LZ1=LQ+1
LZ2=H-1
DO 30 I=LZ1,LZ2
30 D(I)=(0.,0.)
LZ=LH+L
L22=NX+LA-1
DO 40 I=LZ1,LZ2
40 D(I)=(0.,0.)
RETURN
END
Program STKSB reads digitized WAR data and stacks a desired portion of the bottom reflected arrival for use as a source signature. It has the option of rejecting certain traces which by experimentation are not detected properly. Teletype prints the amplitude on the sonobuoy channel which passed the test level. Additional input parameters requiring explanation are:

No. of secs in scale =: number of seconds in time axis for plot
Skip? secs after shot break: number of seconds to be skipped after shot break on streamer channel before beginning search for direct arrival on sonobuoy channel.

Program XSNSFW reads the output disk files of programs STKSS or STKSB and processes them for use in Program WARDCN. A sample conversational input is shown below that for Program STKSB.
**FORT/B STKSB**

**PROGRAM IS RELOCATABLE**

```
R
RLDX/M STKSB FORT.LB
R
INIT MT0
R
STKSB
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
```

**Example input**

```
IN!? MT
ST KS B
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
```

NO. OF TRACES DESIRED=25
NO. OF TRACES REJECTED=4
 WHICH TRACES REJECTED? (25 MAX)  4,6,11,15
NO. OF SECONDS SPACED OVER=76
NO. OF SECS IN SCALE=.5
SHOTBREAK CHANNEL NO.=2
SHOT BREAK TEST LEVEL=600.
SOURCE SIGNATURE CHANNEL NO.=4
SOURCE SIGN. TEST LEVEL=200.
SKIP ? SECS AFTER LAST READ TO NOISE SAMPLE 2
SKIP OVER ? SECONDS AFTER READS 24
SKIP ? SEC AFTER SHOT BREAK 3
NO. OF POINTS IN SOURCE FUNCTION=130
READ EVERY ? DATA POINT 2
ATTENUATION FACTOR=.3338
S?TEST= 0.331330E 3
S?TEST= 0.371330E 3
S?TEST= 0.395330E 3
S?TEST= 0.420330E 3
S?TEST= 0.246033E 3
S?TEST= 0.406330E 3
S?TEST= 0.357330E 3
S?TEST= 0.409330E 3
S?TEST= 0.372330E 3
S?TEST= 0.514003E 3
S?TEST= 0.340003E 3
S?TEST= 0.317003E 3
S?TEST= 0.532003E 3
S?TEST= 0.275330E 3
S?TEST= 0.377003E 3
S?TEST= 0.334003E 3
S?TEST= 0.365003E 3
S?TEST= 0.403003E 3
S?TEST= 0.439003E 3
S?TEST= 0.420003E 3
S?TEST= 0.418003E 3
STOP
R
RELEASE MT0
R
```

**XSNVF**

```
NO. OF POINTS IN SOURCE FUNCTION=130
NO. OF TRACES IN STACK=21
COMPUTE FFT OF NOISE SAMPLE? YES=1,NO=0  0
PLOT FILTER AMPL SPECTRUM ONLY? YES=1,NO=0  0
PLOT STACKED SOURCE FUNCTION? YES=1, NO=0  1
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
DIGITIZING INTERVAL=.302
LENGTH OF HORIZONTAL AXIS IN INCHES=8.
PLOT AMPL SPECTRUM OF SOURCE? YES=1,NO=0  1
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
DIGITIZING INTERVAL=.48823
LENGTH OF HORIZONTAL AXIS IN INCHES=8.
APPLY GHOST FILTER TO SOURCE? YES=1,NO=0  0
STOP
R
```
PROGRAM STKS

STACKS THE BOTTOM REFLECTED WAVE AND STORES IT ON FILE "ISFW"

STORES A SAMPLE OF NOISE OF LENGTH 1024 ON FILE "INW"

THIS DATA IS THE SOURCE SIGNATURE FOR INPUT TO PROGRAM XNSFW

DIMENSION IBUF(4,1024), Y(1024), NREJECT(25)

COMMON/KULB/LBLT(7)

DATA LBLT/14HTIME (SECONDS)/

100 FORMAT(1H, 10E13.6)

CALL FOPEN(2, "ISFW")

CALL FOPEN(3, "INW")

TYPE "TURN ON PLOTER"

PAUSE

CALL MTOPD(I, NT0, 0, IER)

CALL FOPEN(6, "*PLT")

CALL INITIAL(6, 100, 0, 0.)

CALL PLOT(1, 0, -3)

ACCEPT "NO. OF TRACES DESIRED=", NTRACE

ACCEPT "NO. OF TRACES REJECTED=", NREJECT

IF(NREJECT.EQ.0) GO TO 3

ACCEPT "WHICH TRACES REJECTED? (25 MAX) " , (NREJECT(K), K=1, NREJECT)

3 ACCEPT "NO. OF SECONDS SPACED OVER=". NSEC

ACCEPT "NO. OF SECS IN SCALE=". SECD

ACCEPT "SHOT BREAK CHANNEL NO.=". STR

ACCEPT "SHOT BREAK TEST LEVEL=". STRE

ACCEPT "SOURCE SIGNATURE CHANNEL NO.=". NSON

ACCEPT "SOURCE SIGNATURE TEST LEVEL=". STRT

ACCEPT "SKIP ? SECS AFTER LAST READ TO NOISE SAMPLE " , NWJ

K=1

1 CALL NTDIO(1, 8, IBUF, ISTAT, IER, IDs)

IF(K.EQ.1) GO TO 2

K+K+1

GO TO 1

2 ACCEPT "SKIP OVER ? SECONDS AFTER READS " , NDMP

ACCEPT "SKIP ? SEC AFTER SHOT BREAK " , HD2

ACCEPT "NO. OF POINTS IN SOURCE FUNCTION " , HS

ACCEPT "READ EVERY ? DATA POINT " , NDAT

ACCEPT "ATTENUATION FACTOR=" , ATTN

BXX=SECD/20.

BXX=0.2 * NDAT/SECD

CALL AXIS(0, 0, LBLT, -14, 20, 0, 0, BXX, 3)

CALL PLOT(0, 1, -3)

CALL SYMBOL(-3, 1, 0, 14, LBLT, 0, 0, 14)

CALL PLOT(0, 0, 3)

NPASS=0

DO 6 I=1, 1024

Y(I)=0

NPASS=NPASS+1

IF(NPASS.EQ.1) GO TO 15

IF(NDSP.EQ.0) GO TO 15

K=1

5 CALL NTDIO(1, 8, IBUF, ISTAT, IER, IDs)

IF(K.EQ.1) GO TO 15

K+K+1

GO TO 5

15 IFLAG1=0

IFLAG2=0

X=0.0
STKSB 3/25/78 13:17:6 PAGE 2

59; MNTR=0
60; IJK=5
61; 22 CALL MTDIO(1.0,IBUF,ISTS,IER,IMCDNT)
62; DO 20 I=1,1024,MDAT
63; IF(ILG1.EQ.1)GO TO 25
64; YY=IBUF(HSTR,1)
65; IF(YY.LT.SMTREST)GO TO 20
66; IFILG1=1
67; DO 21 J=1,25
68; IF(MREJCT(J).EQ.NPASS)GO TO 26
69; 21 CONTINUE
70; GO TO 27
71; 26 NB3=M2+HTEMP
72; K=1
73; 28 CALL MTDIO(1.0,IBUF,ISTS,IER,IMCDNT)
74; IF(K.EQ.NB3)GO TO 7
75; K=K+1
76; GO TO 29
77; 27 IF(ND2.EQ.0)GO TO 25
78; K=1
79; 23 CALL MTDIO(1.0,IBUF,ISTS,IER,IMCDNT)
80; IF(K.EQ.ND2)GO TO 25
81; K=K+1
82; GO TO 23
83; 25 IF(IFLG2.EQ.1)GO TO 30
84; YY=IBUF(NSON,1)
85; IF(ABS(YY).LT.SONTEST)GO TO 28
86; TYPE "SONTEST=".YY
87; IFLG2=1
88; IT1=1-5
89; IT2=1-4
90; IT3=1-3
91; IT4=1-2
92; IT5=1-1
93; Y(1)=FLOAT(IFBUF(NSON,IT1))*Y(I)
94; YY=IBUF(NSON,IT1)*ATTEN
95; X=X+DX
96; CALL PLOT(X,YY,2)
97; Y(2)=FLOAT(IFBUF(NSON,IT2))*Y(2)
98; YY=IBUF(NSON,IT2)*ATTEN
99; X=X+DX
100; CALL PLOT(X,YY,2)
101; Y(3)=FLOAT(IFBUF(NSON,IT3))*Y(3)
102; YY=IBUF(NSON,IT3)*ATTEN
103; X=X+DX
104; CALL PLOT(X,YY,2)
105; Y(4)=FLOAT(IFBUF(NSON,IT4))*Y(4)
106; YY=IBUF(NSON,IT4)*ATTEN
107; X=X+DX
108; CALL PLOT(X,YY,2)
109; Y(5)=FLOAT(IFBUF(NSON,IT5))*Y(5)
110; YY=IBUF(NSON,IT5)*ATTEN
111; X=X+DX
112; CALL PLOT(X,YY,2)
113; 30 IJK=IJK+1
114; IF(IJK.GT.NS)GO TO 30
115; YY=IBUF(NSON,1)+Y(IJK)
116; YY=IBUF(NSON,1)*ATTEN
117: \( \text{CALL PLOT}(x, y, 2) \)
119: 20 \text{CONTINUE}
120: \( \text{NMTR}=\text{NMTR+1} \)
121: \( \text{HTEMP} = \text{NMTR} \)
122: \text{GO TO 22}
123: 58 \text{CALL PLOT}(0, 1, -3)
124: \( \text{IF}(\text{NPASS}. \text{LT.} \text{NTRACE}) \text{GO TO 7} \)
125: \text{DO 55 I=1, NS}
126: 55 \text{Y(I)} = \text{Y(I)} / (\text{NTRACE} - \text{NTREJ})
127: \text{WRITE}(2, 100)(\text{Y(I), I}=1, \text{NS})
128: \text{IF}(\text{NNJ}. \text{EQ.} 0) \text{GO TO 70}
129: \text{K}=1
130: 60 \text{CALL MDIQ}(1, 0, \text{IBUF, ISTS, IER, IMBDNT})
131: \text{IF}(\text{K} \text{EQ.} \text{NNJ}) \text{GO TO 70}
132: \text{K}=\text{K}+1
133: \text{GO TO 60}
134: 70 \text{DO 75 I}=1, 1024
135: 75 \text{Y(I)} = 0.
136: \text{IKJ}=0
137: \text{K}=0. 8
138: 80 \text{CALL MDIQ}(1, 0, \text{IBUF, ISTS, IER, IMBDNT})
139: \text{DO 90 I}=1, 1024, 10
140: \text{IKJ}=\text{IKJ}+1
141: \text{IF}(\text{IKJ} \text{GT.} 1024) \text{GO TO 95}
142: \text{Y(IKJ)} = \text{IBUF(NSON, I)}
143: \text{YY} = \text{IBUF(NSON, I)* (ATTEN)}
144: \text{X} = \text{X+DX}
145: \text{IF}(\text{X} \text{LE.} 20.) \text{GO TO 92}
146: \text{X} = \text{DX}
147: \text{CALL PLOT}(0, 1, -3)
148: 92 \text{CALL PLOT}(x, y, 2)
149: 90 \text{CONTINUE}
150: \text{GO TO 90}
151: 95 \text{WRITE}(3, 100)(\text{Y(I), I}=1, 1024)
152: \text{CALL RESET}
153: \text{CALL PLOT}(0, 1, -3)
154: \text{END}
Program STKSS reads digitized WAR data from magnetic tape and stacks a desired number of direct wave arrivals for use as a source signature. The program searches the streamer channel for the shot-break and then searches the sonobuoy channel for the direct arrival. The teletype prints the travel time from airgun to sonobuoy in seconds, the computed ghost time delay and the argument which are stored on disk file "ARGFILE". The input is conversational and self explanatory except for the following parameters:

Skip over ? seconds after reads: the integer number of tape blocks (1 sec) to be skipped after storage of each direct arrival to streamer before beginning the search for the next direct arrival.

Attenuation factor: amplification factor for plot (usually approximately 0.0016)
Flow Chart for Program STKSS

START

OPEN
INPUT-OUTPUT
FILES
AND TAPE DRIVE

MANUALLY
TURN ON
PLOTTER

Conversational
Teletype Input

READ
MAG TAPE
SPACE OVER
DATA

Conversational
Teletype Input

Compute time
correction for
travel time from
Airgun to Streamer

Put Axes
and
Labels
on Plot
Initialize Array
Y(1) = 0,
and set
N Pass = 0

7

N Pass = N Pass + 1

IF
N Pass = 1
or NDMP = 0

yes

NO

READ
MAG TAPE
SKIP OVER
DATA

Initialize
Variables and
Integers
IJK = 5

22

READ
MAG TAPE
1 Block
IF IFLAG1 = 1 (Shot break has been found)

IF Y(1) (streamer channel) < STRTEST

SET IFLAG = 1

Plot Data from Sonobuoy Channel

IF IFLAG2 = 1 (Sonobuoy direct wave has been found)

ACCUMULATE TIME SINCE shot break

IF [Y] (Sonobuoy ch.) < SONTEST

20

30

yes

yes

NO

NO

NO

NO

yes
Set $I_{FLA}\cdot G_2 = 1$

Compute Travel Time from airgun to sonobuoy
$T_{AGSO} = T_{CORR} + S_{ACCUM}$

TYPE
"TAGSO" =
TAGSO on Teletype

Compute water surface ghost delay time and grazing angle.

Store 5 points prior to detection point of sonobuoy direct wave

$IJK = IJK + 1$
IF
\[ IJK > \#data \]
points of source sign. to be stored
NO

STACK THE SOURCE SIGNATURE
\[ Y(IJK) = Y(IJK) + \text{VALUE} \]

IF
\[ I \leq 1024, \text{NDAT} \]

RESET PEN TO ORIGIN OF NEXT PLOT

STORE GHOST TIME DELAY ARGUMENT AND GRAZING ANGLE ON DISK

TYPE GHOST TIME DELAY, AND ARG

IF not # traces desired

NO
Average stacked Source Signature

Store stacked source signature on disk

READ MAG TAPE SKIP DATA TO NOISE SAMPLE

Initialize Array $y(1) = 0$.

READ MAG TAPE

STORE NOISE SAMPLE
PLOT NOISE SAMPLE

WRITE NOISE SAMPLE ON DISK FILE

CLOSE FILES

RETURN PLOTTER PEN TO ORIGIN

END
PROGRAM STKSS

STACKS THE SONOBuoy DIRECT WAVE AND STORES ON FILE "ISFW"

STKSS STORES A SAMPLE OF NOISE OF LENGTH 1024 ON FILE "INU"

THIS DATA IS FOR INPUT OF PROGRAM XSNSFW

DIMENSION Ibuf(4,1024).y(1024)

COMMON/KWL64/LBLT(7)

DATA LSLT/48TIME (SECONDS)/

100 FORMAT(H1, 10E13.6)

110 FORMAT(1H, 2E13.6)

CALL FOPEN(2,"ISFW")

CALL FOPEN(3, "INU")

CALL FOPEN(4, "ARGFILE")

TYPE "TURN ON PLOTTER"

PAUSE

CALL MTOPD(1, "MTB: 0", 0, IER)

CALL FOPEN(6, "$PLT>")

CALL INI.TIAL(100.0, 0.

CALL PLOT(1

ACCEPT "NO. OF TRACES DESIRED=". NTRACE

ACCEPT "NO. OF SECONDS SPACED OVER=". NSEC

ACCEPT "NO. OF SECS OF SONO DATA PLOTTED=". SECD

ACCEPT "AIRGUN DEPTH (METERS)=". HAG

ACCEPT "STREAMER DEPTH (METERS)=". HSTR

ACCEPT "SONOBuoy HYDROPHONE DEPTH=". HSON

ACCEPT "AIRGUN TG STREAMER DIST. (METERS)=". XSTR

ACCEPT "STREAMER CHANNEL NO.=". MSTR

ACCEPT "SHOT BREAK TEST LEVEL=". STRTEST

ACCEPT "SONOBuoy CHANNEL NO.=". MSON

ACCEPT "SONOBuoy TEST LEVEL=". SNDTEST

ACCEPT "SKIF 7 SECS AFTER LAST READ TO NOISE SAMPLE ". NHJ

CALL MTBIO(I. 0, Ibuf, Ists, IER, IWBCHT)

IF(K.EQ.NSEC) GO TO 2

K=K+1

GO TO 1

2 IF(K.EQ.NSEC) GO TO 2

ACCEPT "SKIP OVER 7 SECONDS AFT РЕ READS ". NDMP

ACCEPT "NO. OF POINTS IN SOURCE FUNCTION=". NS

ACCEPT "READ EVERY DATA POINT ". NDAT

ACCEPT "ATTENUATION FACTOR=". ATTEN

W=1509.

41: DELF=2. *3. 141593/ .001*NDAT*1024)

42: TAGST=XSTR/WV

43: ANGLE=MTAN(ABS(HAG-HSTR)/XSTR)

44: TCGRR=TAGST*CO5(ANGLE)

45: DXX=SECD/20.

46: DXX = 82*NDAT/SECD

47: CALL AXIS(0., 0.. .0, .0... .0, .DXX, 3)

48: CALL PLOT(0, -1, -3)

49: CALL SYMBOL(-. 3, .0, 14. LBLT, 90. 0, 14)

50: CALL PLOT(0, .0, 3)

51: MPASS=0

52: IF(G 6 I=1.1024

53: Y(I)=0

54: 7 MPASS=MPASS+1

55: IF(MPASS.EQ.1)GO TO 15

56: IF(MPASS.EQ.0)GO TO 15

57: K=1

59: CALL MTBIO(I. 0, Ibuf, Ists, IER, IWBCHT)
S1S$
3/23/70 19.43.33  PAGE 2

59; IF(K.EQ.MMP)GO TO 15
60; K=K+1
61; GO TO 5
62; 15 IFLC1=0
63; IFLC2=0
64; SACCUM=.001*NDAT
65; X0=0
66; IJK=5
67; 22 CALL MDIOC(1,0,IBUF,ISTS,IER,IMBDCHT)
68; DO 20 I=1,1024,NDAT
69; IF(IFLG1.EQ.1)GO TO 25
70; YY=IBUF(NSTR,I)
71; IF(Y.LT.STEST)GO TO 20
72; IFLG1=1
73; 25 YY=IBUF(HSON,I)*ATTEN
74; X=X+9X
75; CALL PLOT(X,YY,2)
76; IF(IFLG2.EQ.1)GO TO 30
77; SACCUM=SACCUM+.001*NDAT
78; YY=IBUF(NSON,I)
79; IFKAS5(YY).LT.SONTEST)GO TO 20
80; TYPE "SONTEST",YY
81; IFLG2=1
82; TAGSO=TCORR+SACCUM
83; TYPE "TRAVEL TIME FROM AIRGUN TO SONOBUGY",TAGSO
84; TS0=TAGSO+TAGSO
85; DELT2=SQRT(TS0+(HSON+HAG)**2/(WY**WY))
86; DELT2=DELT2-SQRT(TS0+(HAG-HSON)**2/(WY**WY))
87; ARG=DELT+DELT2
88; XX2=TAGSO+WY/(1.+HAG/HSON)
89; ARA=HSON/XX2
90; ANGLE=ATAN(ARA)
91; IT1=1-5
92; IT2=I-4
93; IT3=1-3
94; IT4=1-2
95; IT5=1-1
96; Y(1)=FLOAT(IBUF(HSON,IT1))+Y(1)
97; Y(2)=FLOAT(IBUF(HSON,IT2))+Y(2)
98; Y(3)=FLOAT(IBUF(HSON,IT3))+Y(3)
99; Y(4)=FLOAT(IBUF(HSON,IT4))+Y(4)
100; Y(5)=FLOAT(IBUF(HSON,IT5))+Y(5)
101; 30 IJK=IJK+1
102; IF(IJK.GT.HS)GO TO 50
103; Y(IJK)=FLOAT(IBUF(HSON,I))+Y(IJK)
104; 20 CONTINUE
105; GO TO 22
106; 50 CALL PLOT(0.,1.,-3)
107; WRITE(4,110)ARG,ANGLE
108; TYPE "GHOST TIME DELAY",DELT2,"ARG=",ARG
109; IF(NPASS.LT.NTRACE)GO TO 7
110; DO 55 I=1,HS
111; 55 Y(I)=Y(I)/NTRACE
112; WRITE(2,100)(Y(I),I=1,NS)
113; IF(NWJ.EQ.0)GO TO 70
114; K=1
115; 60 CALL MDIOC(1,0,IBUF,ISTS,IER,IMBDCHT)
116; IF(K.EQ.NWJ)GO TO 70
117)  K=K+1
118)  GO TO 60
119)  70 DO 75 I=1,1024
120)  75 Y(I)=0.
121)  IJK=0
122)  X=0.0
123)  CALL MTDIO(1.0,IBUF,ISTS,IER,IVSCHT)
124)  GO 90 [*=1,1024,IBMAT
125)  IJK=IJK+1
126)  IF(IJK.GT.1024)GO TO 95
127)  Y(IJK)=IBUF(HSON,I)
128)  YY=IBUF(HSON,I)*(ATTEN)
129)  X*X=DA
130)  IF(X.LE.20.)GO TO 92
131)  X=DX
132)  CALL PLOT(0..1..-3)
133)  92 CALL PLOT(X,YY.2)
134)  90 CONTINUE
135)  GO TO 89
136)  55 WRITE(3,100)(Y(I),I=1,1024)
137)  CALL RESET
138)  CALL PLOT(0.,1.,-3)
139)  END
Program WARDCN reads digitized magnetic tape and swaps with program DCN14W for source signature deconvolution of wide angle reflection data. Prints on the plotter the two way travel times to the beginning of each processed trace and the direct wave travel time from the airgun to the sonobuoy for that shot. Conversational input and options are similar to program STKDCN except for the following parameters.

Sonobuoy depth: depth to midpoint of sonobuoy hydrophone array in meters.

Variable delay = 1, constant delay = 0: Option 1 will cause program to compute the time delay after the shot break to the beginning of the sequence to be processed taking into account the increasing moveout time for each shot. It will begin storing approximately 0.2 secs before the bottom arrival. Option 2 will compute the time delay only for the first shot, keeping this value fixed for all successive shots.
INIT MT0
R
WARDCN
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
SB 61
STREAMER CHANNEL NO. = 2
SONOBuoy CHANNEL NO. = 4
STREAMER (SHOTBREAK TEST) LEVEL = 500
SONOBuoy (DIRECT WAVE) TEST LEVEL = 300.
NO. OF DIRECT ARRIVALS SEARCHED FOR = 4
NO. OF TRACES DESIRED = 40
NO. OF SECONDS SPACED OVER = 55
NO. OF SECONDS BETWEEN SHOTS = 8
SKIP OVER ? SHOTS BETWEEN TRACES 3
SAVE TRANSFER FUNCTIONS? YES = 1, NO = 0 1
BANDPASS FILTER? YES = 1, NO = 0 1
% OF NYQUIST FOR COS TAPER = 2.
LOW FREQUENCY CUTOFF (ZERO) IS 1/2% OF NYQUIST 1.
HIGH FREQUENCY CUTOFF (ZERO) IS 40% OF NYQUIST 40.
NOTCH FILTER? YES = 1, NO = 0 1
NO. OF NOTCHES = 1
LOW-HIGH LOC. (% NYQ.) MAX OF 5 23, 25.
REMOVE MEAN? YES = 1, NO = 0 1
WINDOW? YES = 1, NO = 0 0
SUBTRACT NOISE? YES = 1, NO = 0 0
NO. OF SECS OF DATA TO BE DECONVOLVED = 1.
READ EVERY ? DATA POINT 2
ATTENUATION FACTOR = 1.
BANDPASS FILTER ONLY; NO DECON.? YES = 1, NO = 0 0
DEPTH OF AIRGUN (METERS) = 9.9
STREAMER DEPTH (METERS) = 3...
RMS WAVE HEIGHT = 1.
SONOBuoy DEPTH (METERS) = 3...
DIST. FROM AIRGUN TO STREAMER MIDPOINT (M) = 99.2
DIP (DEG.) OF BOTTOM (- DOWN FROM SOURCE TO REC.) = 0.
APPLY WATER SURFACE GHOST FILTER? YES = 1, NO = 0 0
DELAY ORIGIN OF PLOT FROM START OF BUBBLE PULSE? YES = 1, NO = 0 1
VARIABLE DELAY = 1, CONSTANT DELAY = 0 1
TWO WAY TRAVEL TIME TO BOTTOM = 4.1
PEAK SCAN? YES = 1, NO = 0 1
% REJECTION LEVEL = 20.
RELEASE MT0
R
Flow Chart for Program WARPLT and WARDCN

START

MANUALLY TURN ON PLOTTER

INITIALIZE MAG TAPE and PLOTTER

Conversational TELETYPETE INPUT

READ MAG TAPE
SKIP OVER DATA

Conversational TELETYPETE INPUT

COMPUTE TRAVEL TIME CORRECTION FROM AIRGUN TO STREAMER

PUT SCALE AND LABELS ON PLOT

PROGRAM WARDCN

Modified Conversational TELETYPETE INPUT
INITIALIZE VARIABLES
NPASS = 0

WARD CN

INITIALIZE INTEGER ARRAY TO ZERO

NPASS = NPASS + 1

SET PLOT TO ORIGIN

IF NPASS = 1 or # BLOCKS TO BE SKIPPED = 0

READ MAG TAPE SKIP DATA

INITIALIZE VARIABLES

READ 1 BLOCK OF MAG TAPE

I = 0
Compute Difference Between Previous and Present Direct Wave Travel Time; Accumulate sums of Differences

IFLAG3 = 1

Direct Wave Search for

NO

TAKA AVERAGE DIFFERENCE OF DIRECT WAVE TRAVEL TIMES

PLAC EPOTTER PEN AT ORIGIN

ACCUMULATE TIME SINCE SHOTBREAK STRT

STRT < SDELAY

NO
For WARDEN replace page with page

PLOT DATA TO BE DECONVOLVED

# data points to be deconvolved = 1024

NO

# data points to be decon. ? SECD

NO

1> 1024, NDAT

RETURN PLOTTER TO ORIGIN OF NEXT TRACE

yes

NPARS < # TRACES DESIRED ?

YES

END

NO

22

20
WARDCN
continued from page

20

22

STORE DATA IN INTEGER ARRAY

IF DATA POINTS TO BE DECONVOLVED = 1024

IF DATA POINTS TO BE DECONV. > DECON. SECD.

YES

YES

1 ≤ 1024, NDAT.

NO

WRITE TRANSFER FUNCTIONS ON DISK

YES

SAVE TRANSFER FUNCTIONS OF THIS TRACE?

YES

CALL FSAM "DCN14W. SV"

NO

SWAP TO PROGRAM

DCN14N. SV

RETURN FROM SWAP

YES

NPASS # TRACES DESIRED?

NO

END

YES

NO

7

NO
PROGRAM WARICH

This program plots a sequence of data from mag tape and swaps with WINRICH for deconvolution of war data. It has the option of bandpass filtering with no deconvolution. The program plots a sequence of data from mag tape and swaps with WINRICH for deconvolution, or it can be used as a bandpass filter with no deconvolution.

The program initializes various common variables and sets up the dimensions of the buffer. It then reads in time labels, data labels, and other parameters necessary for the plotting process. The program allows the user to accept various parameters such as the number of traces desired, the number of seconds spaced over, the number of seconds between shots, the number of shots between traces, and the type of transfer functions desired. It also allows the user to select the type of bandpass filter to be used and the number of notches in the filter. The program then accepts the number of seconds of data to be deconvolved and whether a bandpass filter is to be used with no deconvolution. Finally, it reads in the depth of airgun and the streamer depth and accepts the rms wave height.
IF(NCH.EQ.2)GO TO 10

ACCEPT "SONAR BUOY DEPTH (METERS)"=*, HSON

10 ACCEPT "DIST. FROM AIRGUN TO STREAMER MIDPOINT (M)"=*, XST

IF(NBNDP.EQ.1)GO TO 9

ACCEPT "DIP (DEG.) OF BOTTOM (< DOWN FROM SOURCE TO REC.>)"=*, W

ACCEPT "APPLY WATER SURFACE GHOST FILTER? YES=1, NO=0 "*, NGHFST

IF(NGHST.EQ.0)GO TO 9

ACCEPT "COS TAPER ON ENDS OF FILTER=? % OF NYQUIST ", CG

9 DX=0.0390625

DY=2.0 ATTEN

70 DLT=NBAT*.001

TYPE "DELAY ORIGIN OF PLOT"

71 ACCEPT "FROM START OF BUBBLE PULSE? YES=1, NO=0 ", NDEL

72 IF(NDEL.EQ.0)GO TO 3

73 ACCEPT "VARIABLE DELAY=1, CONSTANT DELAY=0 ", NVCD

74 ACCEPT "TWO WAY TRAVEL TIME TO BOTTOM="*, TWT

75 ACCEPT "PEAK SCAN? YES=1, NO=0 ", NSCAN

76 IF(NSCAN.EQ.0)GO TO 4

77 ACCEPT "% REJECTION LEVEL="*, PRJ

78 SDelay=5.0

W=W+.3.141593/180.

WV=1500.

51 TAGST=XST/WV

ANGLE=ATAN(ABS(H-SH)/XST)

TCoRR=TAGST*COS(ANGLE)

ZT=TWT+WV/2.

80 IF(NDEL.EQ.1)GO TO 13

86 SDelay=0.

87 DXX=512*HDAT*.001/20.

CALL AXIS(.0, .0.., LBLT, -14, 20, .0, .0.., DXX, 3)

89 CALL PLOT(.0, 5, 3)

90 CALL SYMBOL(-.7, 1, .14, LBLT, 90, 0, 14)

MNP=NMPK*NSHOT

92 IFLG3=0

93 CSUM=0

94 HPASS=0

95 HPASS=HPASS+1

96 IF(I=1)GO TO 1824

97 IV(1)=0

98 CALL PLOT(.0, 0, .0..3)

99 IF(HPASS.EQ.1)GO TO 15

100 IF(NDMP.EQ.1)GO TO 15

101 K=1

102 CALL MTBIO(.1, 0, IBUF, ISTS, IER, IWDCT)

103 IF(K.EQ.HNB)GO TO 15

104 K=K+1

105 GO TO 5

106 IFLG1=0

107 IFLG2=0

108 IF(HPASS.GT.NDIR)GO TO 16

109 SONT=0.

110 ST=0.

111 X=0.0

112 IJK=0

113 CALL MTBIO(.1, 0, IBUF, ISTS, IER, IWDCT)

114 DO 29 I=1, 1824, NDAT

115 IF(IFLG1.EQ.1)GO TO 25

116 YY=IBUF(ISTR, 1)
305

WARDCN 5/2/78 14:12:13 PAGE 3

117; IF(ABS(YY).LT.STRTE) GO TO 20
119; IFLG1 = 1
120; 25 IF(IFLG2.EQ.1) GO TO 26
121; STR=STR+ .001*MBAT
122; SONT=SONT+.001*MBAT
123; YY=IBUF(NSOM,1)
124; IF(ABS(YY).LT.STRTE) GO TO 20
125; IFLG2 = 1
126; GO TO 20
127; 27 IF(NPASS.LE.2) GO TO 58
128; SONT=SONT+DIFF
129; IFLG2=1
130; 28 IF(NDEL.EQ.0) GO TO 21
131; IF(NVCD.EQ.0.AND.NPASS.GT.1) GO TO 21
132; XT=YY+SONT+XST
133; ARG=XT*COS(W)/(TWT+XT+SIN(W))
134; THET=ATAN(ARG)+W
135; THET=THET-2.*W
136; THS=THET-2.*THET
137; THR=THS-2.*THR
138; DLT2=H*(+1.+.COS(THS2)) /(MV*COS(THET2))
139; DLT3=HSDN*(+1.+.COS(THR2)) /(MY*COS(THETR))
140; SDelay=2.*SQRT(XT*XT/4.+ZT*ZT)/MV-0.25*XT/MV
141; DTIME=SONT+TCORR
142; 21 CALL NUMBER(-6.0..14, SDELAY,0.0,3)
143; CALL Symbol(.2.-3.,14, LBLD,0.0,8)
144; CALL NUMBER(1.32.-3.,14, DTIME,0.0,3)
145; IF(NPASS.EQ.1) GO TO 30
146; IF(IFLG3.EQ.1) GO TO 32
147; DIFF=SONT-TSONT
148; SUM+SUM+DIFF
149; IF(NPASS.GE.HDIR) GO TO 31
150; 30 TSON=SONT
151; GO TO 32
152; 31 DIFF=SUM/(FLOAT(HDIR-1))
153; IFLG3=1
154; 32 CALL PLOT(0.,0.,3)
155; GO TO 29
156; 26 STR=STR+.001*MBAT
157; 29 IF(STR.LT.SDELAY) GO TO 20
159; IJK=IJK+1
160; STORE*(IJK-1)*MBAT=8.881
161; IY*(IJK)*IEUF(NCH,1)
162; IF(IJK.EQ.124) GO TO 18
163; IF(STORE.GE.SECD) GO TO 18
164; CONTINUE
165; GO TO 22
166; 10 =H=IJK
167; IF(NWRITE.EQ.0) GO TO 19
168; ACCEPT *SAVE TRANSFER FUNCTION OF THIS TRACE? YES=1, NO=0 **.NTRANS
169; IF(NTRANS.EQ.0) GO TO 65
170; 19 CALL FSAMF("DCN14W.SV")
171; 65 IF(NPASS.LT.NTRACE) GO TO 7
172; GO TO 68
173; 58 TYPE "SOMEOBY DIRECT WAVE NOT FOUND"
174; CONTINUE
174; END
PROGRAM DCM14W (SWAPS FROM WARDCH)
SPECTRAL DECONVOLUTION AND TRANSFORMS BACK TO TIME
USED TO DECONVOLVE WAR DATA
FFT OF NOISE SUBTRACTED FROM FFT OF SIGNAL
BANDPASS FILTERS, WINDOWS AND REMOVES MEAN OF SIGNAL
SOURCE FUNCTION AND NOISE READ FROM DISK, SIGNAL FROM MAG TAPE
REMOVES WATER SURFACE GHOST BY FILTERING SOURCE FUNCTION
MULTIPLIES BY INSTRUMENT TRANSFER FUNCTION IN FREQUENCY DOMAIN
DIMENSION IFILE(6), YY(1024), ZZ(1024), DMP(10)
COMMON IY(1024), MMENT, C, XLL, YLL, NMEAN, NWIN, BK, BY, DLT,
6DLT2, DTV3, MGSHS, CC, NSCAN, PRJ, NDATA, MBAND, HW, THETS, THER,
65536.L, NWRITE, NOTCH, NTMAX, FNUMAX(10), MBN
COMMON/KWLIB1/YYC
COMMON/KWLIB2/YYN
EQUIVALENCE (YYC(S12), YY(I)), (YYC(I), ZZ(I)), (YYN(I), DMP(I))
FORMAT(19E13.6)
610X, "TOTAL TWO WAY TIME="F6.3," SEC"
FORMAT(1H4E28.8)
CALL FOPENCI, 'SOHONOISE')
CALL FOPENC2, 'SONOSOURCE')
DO 29 I=1, NT
YYC(I)=YY(I)
DO 19 I=1, NT
IF(NMEAN.EQ.0)GO TO 29
SUM=0.
DO 26 I=1, NT
SUM=SUM+YY(I)
YMEAN=SUM/NT
DO 28 I=1, NT
YY(I)=YY(I)-YMEAN
DO 29 I=1, NT
YMEAN=SUM/NT
YY(I)=YY(I)
CALL WNBW(N)
DO 31 I=1, NT
YYC(I)=YY(I)
DO 32 I=1, NT
CALL FSBW(N)
DO 33 I=1, NT
CALL FCLOSE(3)
CALL IFWRITE(1)
CALL FCLOSE(3)
DO 35 I=1, NT
CALL FCLOSE(3)
DO 37 I=1, NT
CALL FCLOSE(3)
DO 39 I=1, NT
CALL FCLOSE(3)
DO 40 I=1, NT
CALL FCLOSE(3)
DO 42 I=1, NT
CALL FCLOSE(3)
DO 44 I=1, NT
CALL FCLOSE(3)
DO 46 I=1, NT
CALL FCLOSE(3)
DO 48 I=1, NT
CALL FCLOSE(3)
DO 50 I=1, NT
CALL FCLOSE(3)
DO 52 I=1, NT
CALL FCLOSE(3)
DO 54 I=1, NT
CALL FCLOSE(3)
DO 56 I=1, NT
CALL FCLOSE(3)
DO 58 I=1, NT
35) IF(J.LE.5) GO TO 53
36) READ(1,110)(YYN(K),K=1,N)
37) J=1
38) 53 YYC(I)=YYC(I)-YYN(J)
39) 54 CONTINUE
40) C DECONVOLVE BY DIVISION BY FFT OF SOURCE
41) IF(NWRITE.EQ.0) GO TO 55
42) CALL "WRITE FFT OF SIGNAL-NOISE? ", NWR
43) IF(NWR.EQ.0) GO TO 55
44) CALL OUTFILE(3)
45) WRITE(3,130)(YYC(I),I=1,NX)
46) CALL FCLOSE(3)
47) 55 J=0
48) READ(2,110)(YYN(I),I=1,N)
49) DO 80 I=1,N
50) J=I+1
51) IF(J.LE.5) GO TO 62
52) READ(2,110)(YYN(K),K=1,N)
53) J1
54) YYC(I)=YYC(I)*YYH(I)
55) CONTINUE
56) C APPLY WATER SURFACE GHOST ELIMINATING FILTER
57) IF(NWRITE.EQ.0) GO TO 65
58) CALL "WRITE FFT OF DECONVOLVED TRACE? ", NWR
59) IF(NWR.EQ.0) GO TO 65
60) CALL OUTFILE(3)
61) WRITE(3,130)(YYC(I),I=1,NX)
62) CALL FCLOSE(3)
63) 65 J=8
64) READ(2,110)(YYN(I),I=1,N)
65) DO 80 I=1,N
66) J=I+1
67) IF(J.LE.5) GO TO 62
68) READ(2,110)(YYN(K),K=1,N)
69) J1
70) YYC(I)=YYC(I)*YYH(I)
71) CONTINUE
72) C APPLY BANDPASS FILTER IN THE FREQUENCY DOMAIN
73) IF(NWRITE.EQ.0) GO TO 80
74) CALL "WRITE FFT OF DEHOSTED TRACE? ", NWR
75) IF(NWR.EQ.0) GO TO 80
76) CALL OUTFILE(3)
77) WRITE(3,130)(YYC(I),I=1,NX)
78) CALL FCLOSE(3)
79) 80 IF(NFIL.EQ.0) GO TO 85
80) IF(XLL.EQ.108.) GO TO 34
81) CALL LPPFKU(H,T.C,XLL)
82) IF(XHL.EQ.0) GO TO 95
83) IF(NCH.EQ.0) GO TO 110
84) IF(NCH>MAX) GO TO 85
85) CALL SNITCH(H.T.C,XL,XH)
86) GO TO 35
87) 85 IF(NWRITE.EQ.0) GO TO 86
88) ACCEPT "WRITE FFT OF BANDPASS FILT. AND DEHOSTED TRACE? ", NWR
89) IF(NWR.EQ.0) GO TO 86
90) CALL OUTFILE(3)
91) WRITE(3,130)(YYC(I),I=1,NX)
92) C INVERSE FOURIER TRANSFORM DECONVOLVED SERIES
93) CALL FFT3VCNT,1,8)
94) CALL FCLOSE(3)
95) 95 NCH=NCH+1
96) IF(NCH.CH.TCNCHMAX) GO TO 85
97) XL=SNITCH(2*NCH-1)
98) CALL HFFKW(H.T.C,XL)
99) 34 IF(XLL.EQ.0) GO TO 36
100) CALL HFFKW(H.T.C,XL)
101) 36 IF(NFIL.EQ.0) GO TO 95
102) NCH=0
103) 35 NCH=NCH+1
104) IF(NCH.CH.TCNCHMAX) GO TO 85
105) XL=SNITCH(2*NCH-1)
106) XH=SNITCH(2*NCH)
107) CALL SNOTCH(H.T.C,XL,XH)
108) GO TO 35
109) 86 IF(NWRITE.EQ.0) GO TO 86
110) ACCEPT "WRITE FFT OF BANDPASS FILT. AND DEHOSTED TRACE? ", NWR
111) IF(NWR.EQ.0) GO TO 96
112) CALL OUTFILE(3)
113) WRITE(3,130)(YYC(I),I=1,NX)
114) C INVERSE FOURIER TRANSFORM DECONVOLVED SERIES
115) 96 CALL FFT3VCNT(H.T.1,8)
ZZ(I)=REAL(YYC(I))
YMAX=0.
DO 90 I=1,NX
AMP=ABS(ZZ(I))
IF(AMP.LT.YMAX)GO TO 90
MAX=AMP
CONTINUE
DO 98 I=1,NX
ZZ(I)=ZZ(I)/YMAX
IF(MSCAN.EQ.0)GO TO 50
C PEAK SCAN
JFLAG=0
ZMAX=0.
XPRJ=PRJ/100.
DO 78 I=1,NX
TPEAK=ABS(ZZ(I))
IF(TPEAK.LT.XPRJ.AND.JFLAG.EQ.1)GO TO 75
IF(TPEAK.LT.XPRJ)GO TO 71
IF(TPEAK.LT.ZMAX)GO TO 75
ZMAX=TPEAK
IPOS=I
JFLAG=1
GO TO 70
ZMAX=TPEAK
IF(JFLAG.EQ.0)GO TO 70
JFLAG=0
TIME=(IPOS-1)*DST*.001
TOTT=TIME+SNDEL
WRITE(12,120)IPOS,ZZ(IPOS),TIME,TOTT
GO TO 78
ZMAX=XPRJ
CONTINUE
CALL FOPEN(G,"*PLT")
CALL INITIAL(G,100.,0.,0.)
CALL PLOT(0.,0.,-3)
X=0.
DO 40 I=1,NX
X=X+DX
Y=ZZ(I)/DY
CALL PLOT(X,Y,2)
CALL PLOT(0.,5.,-3)
CALL FBACK
END
SUBROUTINE VGHOST(NX, DELT2, DELT3, DLT, CG, MW, THETS, THETR)

CXG: MW = 0

NX = NT + 1

W = 1.03

DEL = 0.3 * PI / (NT * DLT).

IF (CG > 1.0) CG = 1.

LX = (FI) / (CM * (NX - 1) / 100.)

TDEL = DELT2

ANGLE = FI / 2 - THETS

LOGP = 0

1 LOOP = LOOP + 1

AP = DELF + TDEL

ARG = 4.2 + LX < (0., -1.)

KLAGED = 1. < DELF * LX

RH = -W * SIN(ANGLE) / XLAGDA

FACT = EXP(A)

GF = FACT < (1., 0., -FACT * EXP(ARG))

GFMD = CHBS(GF)

XR = REAL(GF)

XI = REAL(GF - CJNG(GF)) < (0., -5.)

DO 14 I = 1, NX

CM = ARG < (1., 0., -FACT * EXP(ARG))

GFMD = CHBS(GF)

GO TO 40

2 LOOP = LOOP + 1

IF (GFMD < LFLT) GO TO 80

XIT = REAL(GF - CJNG(GFLT)) < (0., 5.)

XIP = SIGN(XIT, XIT)

GFLT = CMLX(XIT, XIT)

YXC1 = YXC(I) * GFLT

IF (LOGF > Eq. 2) GO TO 50

3 GO TO 18

50 IF (I.EQ.1) GO TO 10

60 IF (I.EQ.NX) GO TO 10

70 J = NT - 1 + 2

XRGF = REAL(YXC(I))

80 XIGF = REAL(YXC1 - CJNG(YXC(I)) < (0., -5.))

90 YXC(I) = CMLX(XGRF, XIGF)

10 CONTINUE

11 IF (LOGF < Eq. 2) GO TO 70

12 TDEL = DELT2

13 ANGLE = FI / 2 - THETR

14 GO TO 1

15 CONTINUE

16 RETURN

17 END
Program WARPLOT reads digitized sonobuoy data from magnetic tape in the same way as program WARDCN and plots the time domain traces to be deconvolved. Input is conversational and an example run is shown below.

```
INIT MT0
R
WARPLOT
TURN ON PLOTTER
PAUSE
STRIKE ANY KEY TO CONTINUE.
STREAMER CHANNEL NO.=2
SONOBUOY CHANNEL NO.=4
STREAMER (SHOTBREAK) TEST LEVEL=500.
SONOBUOY (DIRECT WAVE) TEST LEVEL=300.
NO. OF DIRECT WAVES SEARCHED FOR=4
NO. OF TRACES DESIRED=40
NO. OF SECONDS SPACED OVER=55
NO. OF SECONDS BETWEEN SHOTS=8
SKIP OVER 1 SHOTS BETWEEN TRACES 3
READ EVERY 2 DATA POINT 2
ATTENUATION FACTOR=.0004
AIRGUN DEPTH=6.9
STREAMER DEPTH=3.
AIRGUN TO STREAMER MIDPOINT DIST. (METERS)=99.2
DELAY ORIGIN OF PLOT
FROM START OF BUBBLE PULSE? YES=1,NO=0 1
VARIABLE DELAY=1, CONSTANT DELAY=0 1
TWO WAY TRAVEL TIME TO BOTTOM=4.1
S_CNTEST= -.780000E 3
S_CNTEST= .513000E 3
S_CNTEST= .335000E 3
S_CNTEST= .418000E 3
STOP
R
RELEASE MT0
```
PROGRAM WARPLT

This program plots a sequence of data from Mac tape in the same way as WARICH assuming a 1024 point FFT in DECON.

COMMON/KWLB4/LBLT(7), LBLA(3), LBLD(4)
DATA LBLT/I4: TIME (SECONDS)/
DATA LBLD/I4: D-TIME/
DATA LBLA/I4: AMP /

TYPE "TURN ON PLOTTER"
PAUSE
CALL NTOPD(1, "MTB: 0", 0, IER)
CALL FOPENC(*, "SPLT")
CALL INITIAL(6, 100, 0, 0)
CALL PLOT(I, 0, -3)

STREAMER CHANNEL NO."1", NSTR
ACCEPT "SONOBODY CHANNEL NO."1", NSON
ACCEPT "STREAMER (SHOTBREAK) TEST LEVEL=", STRTEST
ACCEPT "SONOBODY (DIRECT WAVE) TEST LEVEL=", SONTST
ACCEPT "NO. OF DIRECT WAVES SEARCHED FOR=", NDIR
ACCEPT "NO. OF TRACES DESIRED=", NTRACE
ACCEPT "NO. OF SECONDS SPACED OVER=", NSEC
K=0.

KONMIO(1, 0, IBUF, ISTS, IER, IWBCNT)
IF(K.EQ.0) GO TO 2
K=K+1
GO TO 1
2

ACCEPT "NO. OF SECONDS BETWEEN SHOTS=", NSHOT
ACCEPT "SKIP OVER 7 SHOTS BETWEEN TRACES", NSKIP
ACCEPT "READ EVERY 7 DATA POINT", NDAT
ACCEPT "ATTENUATION FACTOR=", ATTEN
SEC=1024*NDAT*0.001/2.

D(B=0.02*NDAT/SEC

ACCEPT "AIRGUN DEPTH=", H
ACCEPT "STREAMER DEPTH=", SH

ACCEPT "AIRGUN TO STREAMER MIDPOINT DIST. (METERS)=" , XST

INPUT "DELAY ORIGIN OF PLOT"
ACCEPT "FROM START OF BUBBLE PULSE? YES=1, NO=0 ", HDEL
IF(HDEL EQ.0) GO TO 3
ACCEPT "VARIABLE DELAY=1, CONSTANT DEL-AY=0 ", NVCU

ACCEPT "TWO WAY TRAVEL TIME TO BOTTOM=", TWT
SDLAY=5.0

WD=1500.

TAGST=XST+WV
ANGLE=ATANABS(M-H)*XST)
TCORRH=TAGST*COS(ANGLE)
ZTT=WV/2.

IF(NDEL EQ.0) GO TO 13

DXX=312*NDAT*0.001/2.

CALL AXIS(0, 0, LBLI, -14, 20, 0, 0, DXX, 3)
CALL PLOT(0, 0, -3)

CALL SYMBOL(-7, 1, 0..14, LBLT, 98, 14)
NDMP=MSKIP*NSHOT
IF(CL)3=0
SUM=0.

NPASS=0
7
NPASS=NPASS+1

CALL PLOT(0, 0, -3)
59; IF(NPASS.EQ.1) GO TO 15
60; IF(NMAP.EQ.0) GO TO 15
61; K=1
62; 5. CALL MTIO(1.0, IBUF, ISTS, IER, IWDCHT)
63; IF(K.EQ.NMAP) GO TO 15
64; K=K+1
65; GO TO 5
66; 15 IFLG1=0
67; IFLG2=0
68; IF(NPASS.GT.NDIR) GO TO 16
69; SONT=0.
70; STRT=0.
71; X=0.0
72; IJK=0
73; 22 CALL MTIO(1.0, IBUF, ISTS, IER, IWDCHT)
74; DO 20 I=1,10,BAT
75; IF(IIFLG1.EQ.1) GO TO 25
76; YY=IBUF(NSTR,1)
77; IF(ABS(YY).LT.STRTST)GO TO 20
78; IFLG1=1
79; 25 IF(IIFLG2.EQ.1)GO TO 26
80; STRT=STRT+.001*BAT
81; IF(IIFLG2.EQ.1)GO TO 27
82; SONT=SONT+.001*BAT
83; YY=IBUF(NSTR,1)
84; IF(ABS(YY).LT.STRTST)GO TO 20
85; TYPE "SONT="YY
86; IFLG2=1
87; GO TO 20
88; 27 SONT=SONT+DIFF
89; IFLG2=1
90; 20 IF(NDEL.EQ.0)GO TO 21
91; IF(NYCD.EQ.0.AND.NPASS.GT.1)GO TO 21
92; XT=WY+SONT+XST
93; SDELAY=2.*SQRT(XT*XT/4.+ZT*ZT)/WY-0.25*XST/WY
94; DTIME=SONT+TCORR
95; 21 CALL NUMBER(-.8,.8,.14,SDELAY,.0,.3)
96; CALL SYMBOL(.2-.3,.14,LBD,.0,.8)
97; CALL NUMBER(.32-.3,.14,DTIME,.0,.3)
98; IF(NPASS.EQ.1)GO TO 30
99; IF(IIFLG3.EQ.1)GO TO 32
100; DIFF=SONT-TSONT
101; SUM=SUM+DIFF
102; IF(NPASS.GE.NDIR)GO TO 31
103; 30 TSONT=SONT
104; GO TO 32
105; 31 DIFF=SUM/(FLOAT(NDIR-1))
106; IIFLG3=1
107; 32 CALL PLOT(0.,0.,3)
108; GO TO 29
109; 26 STRT=STRT+.001*BAT
110; IF(STRT.LT.SDELAY)GO TO 20
111; IJK=IJK+1
112; STORE=(IJK-1)*BAT+.001
113; YY=IBUF(NSHN,1)*ATTEN
114; X=X+DX
115; CALL PLOT(X,YY,2)
116; IF(IJK.EQ.1024)GO TO 18

WARPLOT 5/2/78 14:12:13 PAGE 3

117; IF(STORE.GE.SEC2)GO TO 18
118; 20 CONTINUE
119; GO TO 22
120; 18 CALL PLOT(0.,5,-3)
121; IF(NPASS.LT.NTRACE)GO TO 7
122; END
INPUT WAVELET PLUS 4 ECHOES WITH PHASE CHANGE

COMMON:: KUBL=4/YP(200), YA(550)
COMMON:: KWLE2/YI(550), YD(550)
COMMON:: KUBL=Y(550), Y(550)
COMMON:: KUL=3/LBLT(3), LBLA(3), LBLF(3)
DATA : LST/6H TIME 2
DATA : LSL/6H AMPL /
DATA : LBF/6H FREQ /
CALL FOPEN(1,*KUDT7*)
CALL FOPEN(2,*WAVEL*)
READ(1, 100) NOLYM
FORMAT(100) CTKW
READ(2, 105) NOLF
FORMAT(105) CTKW
READ(3, 110) NOLT
FORMAT(110) CTKW
READ(1, 120) NOLR
FORMAT(120) PLTW
pi=3.141593
M=N+1
DO 5 I=1,N
Y(I)=0.
S=1.
4 YP(I)=3.
5 CONTINUE
READ(1, 150) YP(I), I=1,N
WRITE(2, 110) M
WRITE(2, 120) YP(I), I=1,M
CALL STKRM
DO 55 J=1,M
Y(J)=0.
55 CONTINUE
10 NX=N/2+1
11 N=NX
12 M=NX
13 DO 70 I=1,N
14 J=M+1
15 Y(I)=YP(I)
16 CONTINUE
30 READ(1, 120) END=50) NOLY, R, PHI
31 IF(Phi.EQ.0.) GO TO 60
32 IF(Phase=Phi/FI.EQ.180)
33 DO 53 I=1,N
34 Y(I)=Y(I)+COS(Phase)*YD(I)*SIN(Phase)
35 CONTINUE
40 CONTINUE
41 M=M-1
42 CONTINUE
150 DO 150 J=1,N
161 Y(J)=Y(J)
170 CONTINUE
50 CONTINUE
51 DO 170 I=1,N
52 J=I+M=1
53 Y(J)=Y(J)
54 CONTINUE
55 MLY=0
56 CONTINUE
57 CONTINUE
58 GO TO 10
SCTKW

1. SUBROUTINE SCTKW(N)
2. \( \text{COMPUTES SIN-COS OPERATION ON INPUT WAVELET} \)
3. COMMON..KHLBI/YC(550), YD(550)
4. COMMON..KMLE/Vh(200), Yh(550)
5. DO 10 I=1,N
6. YI(I)=Yh(I)
7. 10 CONTINUE
8. CALL CTFKW(N, SGNI)
9. \( \text{SGNI} = 1.0 \)
10. CALL STFWK(N, SGNI)
11. \( \text{DO 20 I=1,N} \)
12. YAC(I)=Yh(I)
13. 20 CONTINUE
14. \( \text{CALL STFWK(N, SGNI)} \)
15. \( \text{SGNI} = 2.0 \)
16. CALL CTFKW(N, SGNI)
17. \( \text{DO 30 I=1,N} \)
18. YAC(I)=Yh(I)
19. 30 CONTINUE
20. \( \text{CALL STFWK(N, SGNI)} \)
21. \( \text{SGNI} = 2.0 \)
22. CALL CTFKW(N, SGNI)
23. \( \text{DO 40 I=1,N} \)
24. YAC(I)=Yh(I)
25. 40 CONTINUE
26. \( \text{CALL STFWK(N, SGNI)} \)
27. \( \text{DO 50 I=1,N} \)
28. \( \text{CALL STFWK(N, SGNI)} \)
29. \( \text{DO 60 I=1,N} \)
30. \( \text{RETURN} \)
31. \( \text{END} \)
SUBROUTINE CTFK(W, SGNI)

COMMON /KWL2/ YI(550), YD(550)

DO 1 I=1, N
1 YI(I)=YD(I)

YD(I)=0.

10 CONTINUE

NX=N/2+1

SC=1./SORT(FLOAT(N))

D=2.*3.141593/FLOAT(N)

IF(SGNI.EQ.1.) GO TO 59

HA=2

NE=NX-1

GO TO 55

50 HA=1

55 DO 30 I=1, NX

YI(I)=0.

I=I+1

W=HI+FLOT(I)

DO 40 =HE, HA

J=J-1

TH=FLOT(J)*W

YD(I)=D(I)*YI(J)*COS(TH)

40 CONTINUE

GO TO 35

TH=FLOT(HE)*W

YI(I)=SCNI+YD(I)*YI(1)+YI(NX)*COS(TH)*SC

GO TO 70

35 YD(I)=SCNI*SC*YD(I)

30 CONTINUE

NX=NX-2

DO 20 =1, NX

J=J-1

35 K=NX+1

36 YD(K)=D(K)

20 CONTINUE

RETURN

END
SUBROUTINE STFKCN, SCIY

1. DO I = 1, N
2. YI(I) = ID(I)
3. YD(I) = SC
4. N = N/2
5. SC = 1./SQRT(FLOAT(N))
6. DU = 2.*3.141593/FLOAT(N)
7. IF(SONLEQ.1.) GO TO 50
8. HA = 2
9. NB = NX - 1
10. GO TO 55

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

RETURN

END.

SUBROUTINE PLTK(NN, LBLX, LBLY)

1. COMMON /KVLBI/X(550), YY(550)
2. TYPE "TURN ON PLOTTER"
3. CALL INITIAL(6.1.30, 0.0, 0.8)
4. CALL PLOT(1., 0.0, 0.0)
5. CALL SCALE(YY, M, 0., YMIN, DY)
6. CALL AXIS(0., 0., M, 0., YMIN, DY, 2)
7. CALL SCALE(X, M, 10., XMIN, DX)
8. CALL AXIS(0., 0., LBLX, -6.10., XMIN, DX, 0)
9. CALL PLOT(XX(1), YY(1), 3)
10. IPEN = 2
11. DO 10 I = 1, NN
12. CALL PLOT(XX(I), YY(I), IPEN)
13. CONTINUE
14. CALL PLOT(XX(M), YY(M), 3)
15. CALL PLOT(0., YY(M), 2)
16. CALL PLOT(0., 0., 3)
17. RETURN
18. END

END

END
WV112 1.29.77 7.52.18 PAGE 1

1.C  PROGRAM WV112
2.C  INPUT "WAVELET PLUS ECHOES WITH FREQUENCY DEPENDENT LOSS"
3.C  AND LINEAR FREQUENCY DEPENDENT PHASE SHIFT APPLIED ONLY TO LINEAR
4.C  RAMP AND EXPONENTIAL LOSS FUNCTIONS
5.C  INCLUDES ADDITIVE NOISE OPTION ON SIGNAL AND SOURCE FUNCTION
6.C  WRITE SIGNAL+NOISE, THEN SOURCE+NOISE, THEN SIGNAL+NOISE
7.C  NOISE MAY BE DIFFERENT IN EACH CASE
8.C  COMMON MODEL: YG(550)
9.C  COMMON KNLB/YPK(200), YA(550)
10.C  COMMON KNLB/YK(550)
11.C  COMMON KNLB/CLVL(3), LBLA(3), LSLF(3)
12.C  DATA LSLF/CH TINE /
13.C  DATA LSLF/CH AMP /
14.C  DATA LSLF/CH FREQ /
15.C  CALL FCOPEN(1, "MUDT7")
16.C  CALL FCOPEN(2, "WAVEL")
17.C  READ(1,100), N, NLAY
18.C  100 FORMAT(2(3))
19.C  105 FORMAT(7,F4.3,7.12,4F13.6,13)
20.C  120 FORMAT(10E13.5)
21.C  120 FORMAT(10F6.2)
22.C  J=0
23.C  P=J, 1415.3
24.C  K=NLAY
25.C  S=5,12.12
26.C  YP(I)=0
27.C  YP(I)=0
28.C  X(I)=0
29.C  CONTINUE
30.C  READ(1,100),(YP(I), I=1,N)
31.C  NLAY=NLAY+1
32.C  35 FORMAT(35, END=5) NOLY. P. I. H. R. S. C. E. I. L.A
33.C  IF(15, 105, 105) TE 50
34.C  DO 15 EN=1, N
35.C  YP(I)=YP(I)
36.C  15 CONTINUE
37.C  CALL FFT4(N,-1.0)
38.C  CALL UNNDKWN, I.W. G. B. C. E. L.A
39.C  CALL FFT4H(I)
40.C  DO 10 I=1,N
41.C  YC(I)=YFNL(YC(I))
42.C  CONTINUE
43.C  GO TO 50
44.C  60 DO 29 I=1,N
45.C  70 CONTINUE
46.C  80 CONTINUE
47.C  DO 38 I=1,N
48.C  CONTINUE
49.C  DO 48 I=1,N
50.C  J=NLAY
51.C  YC(J)=YC(J)+Y(I)*R
52.C  30 CONTINUE
53.C  GO TO 75
54.C  DO 53 I=1,N
55.C  53 CONTINUE
56.C  50 J=J+1
57.C  IF(CD.J.EQ.1) GO TO 160
58.C  IF(JJ.J.EQ.2) GO TO 165
DO 175 I=1,M
59: 170 X(I)=Y(I).
60:   GO TO 165
61: 165 DO 175 I=1,M
62: 160 ACCEPT "DO YOU WISH TO ADD NOISE. YES=1, NO=0 ". NO!
63:   IF(NO.EQ.0) GO TO 55
64: 55 S=1.
65:   ACCEPT "WISH RANDOM SERIES, 1 OR 2(VARIABLE) ". IR
66:   IF(IR.EQ.1) GO TO 56
67: 56 CALL R:NE(1,M,IR)
68:   SUM=Y
69:   I=1.
70:   SUM=SUM+Y(I)
71:   S=S+c
72:   DO 79 I=1,M
73: 79 SUM=SUM+Y(I)
74:   S=S+c
75:   DO 80 I=1,M
76: 80 SUM=SUM+(Y(I)-MEAN)**2
77:   MOV=SRT(SUM)
78:   DO 100 I=1,M
79: 100 X(I)=X(I)+MOV*Y(I)
80:   WRITE(2,120)<X(I),I=1,M>
81:   IF(J.EQ.0) GO TO 100
82: 100 J=J-1
83: 150 CALL FLO(S(E)
84: 140 ACCEPT "DO YOU WISH TO PLOT. YES=1, NO=0 ". PLOT
85:   IF(PLOT.EQ.0) GO TO 170
86: 170 J=J+1
87:   Y(I)=X(I)
88: 190 X(I)=J
89: 180 CONTINUE
90:   END
SUBROUTINE FFTW(LX, SIGNED)
COMMON/K4L3/CX
J = 1
SC = SQRT(1./LX)
DO 5 I = 1, LX
5 IF (I.GT.J) GO TO 2
CTEMP = CK(J) + SC
CX(J) = CX(I) + SC
CX(I) = CTEMP
2 M = LX / 2
3 IF (J.LT.M) GO TO 5
J = J - M
5 IF (M.GE.1) GO TO 3
J = J + M
L = 1
6 ISTEP = 2 * L
DO 8 M = 1, L
7 CARG = (0., 1.) * (3.141593 + SIGNED) * (M - 1) / L
8 CX = CEXF(CARG)
DO 8 I = M, LX, ISTEP
10 IPL = I + L
11 CX(IPL) = CX(I) + CTEMP
12 CX(I) = CX(I) + CTEMP
13 L = ISTEP
15 IF (L.LT.LX) GO TO 6
16 RETURN
END
SUBROUTINE WMBKV(N, IV, A, B, C, E, LA)

COMMON * E, FL, X

GO TO (15, 20, 30, 6)

10 J = 1

15 PHI = 0.1 * E * PI * FLOAT(J) / (NX - 1)

ARG = -1. * B / J / NX

D(I) = D(I) + EXP(ARG) * C * F (PHI)

CONTINUE

GO TO 20

20 J = 1

30 L = IF (X + CX) N

TH = PI / L

L = L + X

50 J = 1

60 ARG = TH * FLOAT(J)

70 D(I) = D(I) + (1 + COS(ARG)) / 2

CONTINUE

90 CONTINUE

100 J = M - 1

200 ARG = TH * J

300 D(I) = D(I) + (1 + COS(ARG)) / 2

CONTINUE

400 GO TO 50

500 DO 70 I = 1, N

600 IF (E(I, 1) * OF (1, 2, NX) 70 TO 70

700 J = M - 1

800 XGF = REAL (D(I))

900 KGF = REAL (D(I) - CONJG (D(I)) + (0, - 5))

1000 D(I) = XGF - KGF

CONTINUE

50 CONTINUE

RETURN

END
SUBROUTINE PAHEN(N, IR, S)
COMMON/XUL56/YP(290), Y(550)
GO TO(10, 20), IR
10 DO 30 I=1, N
30 X=(3.1.37+sin(.347*I))/((29.4651+sin(.5173*I))
Y(I)=Y(I)-I
CONTINUE
10 GO TO 20
20 DO 40 I=1, N
40 Q=(2498.*S+9981)/1981
CONTINUE
RETURN
END

SUBROUTINE PLTWY(NM, LELX, LELY)
DIMENSION LELY(3), LELY(3)
COMMON/XUL56/XX(550), YY(550)
TYPE "TURN ON PLOTTER"
PAUSE
CALL INITIAL(6.190, 0.0, 0.0)
CALL PLOT(1.0, 0.0, -3)
M=NM+1
CALL SCALE(YY, M, S, YMIN, DY)
CALL AXIS(0.0, 0.0, LELY, 5.8, 30.0, YMIN, DY, 2)
CALL SCALE(XX, M, S, XXMIN, DX)
CALL AXIS(0.0, 0.0, LELY, -5.8, 30.0, XXMIN, DX, 0)
CALL PLOT(XX(I), YY(I), 1)
IPEN=2
DO 10 I=1, NM
10 CALL PLOT(XX(I), YY(I), IPEN)
CONTINUE
CALL PLOT(XX(M), YY(M), 3)
CALL PLOT(0.0, YY(M), 2)
CALL PLOT(0.0, 0.0, 3)
RETURN
END

END
PROGRAM XSNSFW

READS NOISE FROM FILE "INW" REMOVES MEAN AND TAKES FFT WRITING ONE
FILE "SPECTRUM", HAS OPTION OF PLOTTING AMPLITUDE SPECTRUM
READ STACKED SOUNDBOY SOURCE FUNCTION FROM FILE "ISFW", REMOVES MEAN, PLOTS, TAKES FFT, PLOTS AMPLITUDE SPECTRUM, TAKES INVERSE SPECTRUM AND FINALLY IS APPLIED TO SOURCE FUNCTION WRITING ON FILE "SOUNDSOURCE"

DIMENSION Y(1024), Z(1024)

COMMON/XWLS/ YFT

COMMON/XWLS/LBLA(3), LBLF(3)

EQUIVALENCE (YFT(512), Y(I), (YFT(1), Z(I))

READ FILE "SOUNDNOISE" FILE SMOOTH NOISE. MRS OPTION OF PLOTTING AMPL SPECTRUM
READS STACKED SOUNDBOY SOURCE FUNCTION FROM FILE "ISFW" REMOVES MEAN, PLOTS, TAKES FFT, PLOTS AMPLITUDE SPECTRUM, TAKES INVERSE COMPUTES INVERSE STACKING GHOST FILTER, PLOTS AMPLITUDE AND FINALLY IS APPLIED TO SOURCE FUNCTION WRITING ON FILE "SOUNDSOURCE"

COMPLEX YFT(1024), YWS(5), CARX

COMMON/XWLS/ YFT

COMMON/XWLS/LBLA(3), LBLF(3)

EQUIVALENCE (YFT(512), Y(I), (YFT(1), Z(I))

DATA LELT, LBLA / 6H TIME I

DATA LELF / 6H FREQ /

FORMAT(1H, 10E13.6)

FORMAT(2E13.6)

CALL FOPEN(2, "SOUNDNOISE")

CALL FOPEN(3, "ISFW")

MT = 1024

NX = NT / 2 + 1

ACCEPT "NO. OF POINTS IN SOURCE FUNCTION="*, NS

ACCEPT "NO. OF TRACES IN STACK="*, NTRACE

MPASS = 0

MPASS = MPASS + 1

ACCEPT "COMPUTE FFT OF NOISE SAMPLE? YES=1, NO=0 ", NNOISE

IF(NNOISE.EQ.0) GO TO 14

CALL FOPEN(4, "INW")

CALL FOPEN(5, "SOUNDNOISE")

READ(4, 128) (Y(I), I = 1, NT)

SUM = 0

DO 3 I = 1, NT

3 SUM = SUM + Y(I)

YMEAN = SUM / NT

DO 4 I = 1, NT

4 Y(I) = Y(I) - YMEAN

DO 2 I = 1, NT

2 Y(I) = 0

CALL FFT3W(NX, -1.0)

WRITE(5, 180) (YFT(I), I = 1, NT)

CALL FCLOS(4)

CALL FCLOS(5)

ACCEPT "PLOT NOISE AMPL SPECTRUM? YES=1, NO=0 ", NNOISE

IF(NNOISE.EQ.0) GO TO 14

DO 5 I = 1, NX

5 Z(I) = CARBS(YFT(I))

GO TO 50

14 ACCEPT "PLOT FILTER AMPL SPECTRUM? YES=1, NO=0 ", NFO

IF(NFO.EQ.0) GO TO 16

NPASS = 5

GO TO 39

16 R = 1.0

17 YFT(I) = 0, 0,

READ(3, 128) (Y(K), K = 1, MS)
REMIND 3

NPASS=NPASS+1

ACCEPT *PLOT STACKED SOURCE FUNCTION? YES=1, NO=0 *

IF(NPLOT.EQ.0)GO TO 22

DO 10 I=1,NS

10 Z(I)=Y(I)

GO TO 60

20 I=1,NT

26 YFT(I)=(0.,0.)

DO 120 (Y(K),K=1,NS)

120 SUM=0.

20 I=1,NS

SUM=SUM+Y(I)

GO TO 0

10 Z(I)=Y(I)

90 NPASS=NPASS+1

ACCEPT *PLOT AMPL SPECTRUM OF SOURCE? YES=1, NO=0 *

IF(NPLOT.EQ.0)GO TO 30

WRITE(2,100)(YFT(I),I=1,NT)

REWO 2

IF(HPLOT.EQ.8)GO TO 1030

READ(3,120)(Y(I),I=1,NT)

DO 20 I=1,NS

20 S=SUM+Y(I)

DO 21 I=1,NS

21 Y(I)=Y(I)-S

DO 23 I=1,NS

23 Y(I)=Y(I)

DO 24 I=1,NT

24 WRITE(3,100)(Y(I),I1,WT)

REMIND 2

READ(2,100)(Y(I),I1,NT)

REWO 2

30 I=1,NT

YTM=CA85(YFT(I))

IF(YTM.EQ.0.)GO TO 36

YFT(I)=Y(I)

GO TO 35

35 CONTINUE

WRITE(2,100)(YFT(I),I=1,NT)

REWO 2

ACCEPT *APPLY GHOST FILTER TO SOURCE? YES=1, NO=0 *

IF(NHOST.EQ.0)GO TO 90

NPASS=NPASS+1

90 NPASS=NPASS+1

33 TX=FL0AT(NTRACE)

100 PI=3.141593

101 DELF=1./(102.4*.002)

102 ACCEPT *RMS WAVE HEIGHT=*, H

103 DO 37 I=1,NT

37 YFT(I)=(1.,0.)

104 CALL FOPEN(1,"ARGFILE")

105 DO 40 J=1,NTRACE

40 READ(1,110)(ARGX,J)

106 DO 45 I=1,NX

45 CARX*ARGX*(I-1)*(-0.,-1.0)

109 IF(I.GT.1)GO TO 41

110 IF(I.GT.1)GO TO 41

111 FACT=1.

112 GO TO 42

113 XLAMBDA=1500./((I-1)*DELF)

114 R=1.+2.*PI*H*5IN(ANGLE)/XLAMBDA

115 FACT=CEXP(R)

116 YFT(I)=YFT(I)-FACT*CEXP(CARGX)*TX
ACCEPT "PLOT AMPL RESP. OF FILTER? YES=1, NO=0 ", NPLT
IF(NPLT.EQ.0) GO TO 65
CALL FPENCH("WFLT")
WRITE(7,100)(YFT(I),I=1,NX)
REWIN D 7
GO TO 1
DO 60 I=1,NT
YFT(I)=(0.,0.)
READ(7,120)(YFT(I),I=1,NX)
REWIN D 7
J=0
READ(2,120)(YS(J),J=1,5)
DO 70 I=1,NX
J=J+1
READ(2,129)(Y(J),I=1.5)
DO 78 J=1,5
J1=J
YFT(I)=YFT(I)*Y(J)
DO 75 I=2,NXX
41HT-2+Z
XRGF=REAL((YFT(I))
XIGF=-REAL((YFT(I)-CONJ(YFT(I)))*(-0.,-.5))
YFT(J)=CMPLX(XRGF,XIGF)
CONTINUE
WRITE(2,130)(YFT(I),I=1,HT)
GO TO 5
DO 85 J=1,5
AMP=ABS(Z(J))
IF(AMP LT YMAX) GO TO 85
YMAX=AMP
CONTINUE
DO 93 I=1,NT
AMP=ABS(Z(I))
IF(AMP LT YMAX) GO TO 85
YMAX=AMP
CONTINUE
DO 96 I=1,NX
Z(I)=Z(I)/YMAX
YMAX=1.0
IF(NPAS S .NE. 2) GO TO 97
CALL PLT3W(NS,L8LF,L8LA,YMAX)
GO TO 88
CALL PLT3W(NX,L8LF,L8LA,YMAX)
GO TO 88
IF(NPL T.EQ.0) GO TO 90
CALL FCLOS(7)
CALL DELETE("WFLT")
CALL RESET
END
PLT3W

SUBROUTINE PLT3W(NN, LBLX, LBLY, YMAX)

DIMENSION LBLX(3), LBLY(3), ZZ(1024)

COMMON/MLBL/YYC(1824)

EQUIVALENCE (YYC(1), ZZ(1))

TYPE "TURN ON PLOTTER"

PAUSE

CALL FOPEN(6, "#PLT")

CALL INITIAL(6, 100.0, 0.0, 0.0)

CALL PLOT(1.0, 0.0, -3)

YMIN=-YMAX

DY=2.*YMAX/9.

CALL AXIS(0., 0., LBLX, 6., 8., 90., YMIN, DY, 2)

ACCEPT "DIGITIZING INTERVAL=", DINTX

ACCEPT "LENGTH OF HORIZONTAL AXIS IN INCHES=", XLENGTH

XMAX=DINTX*(NN-1)

DX=XMAX/XLENGTH

XMIN=0.

CALL AXIS(0., 0., LBLX, -6., XLENGTH, 0., XMIN, DX, 2)

XX=DINTX/DX

YY=ZZ(1)/DY+4.0

CALL PLOT(XX, YY, 3)

IPEN=2

DO 10 I=1, NN

XX=DINTX*I/DX

YY=ZZ(1)/DY+4.0

CALL PLOT(XX, YY, IPEN)

10 CONTINUE

YLAST=4.0

CALL PLOT(XX, YLAST, 3)

CALL PLOT(0., YLAST, 2)

CALL PLOT(0., 0., 3)

CALL FCLOS(6)

RETURN

END