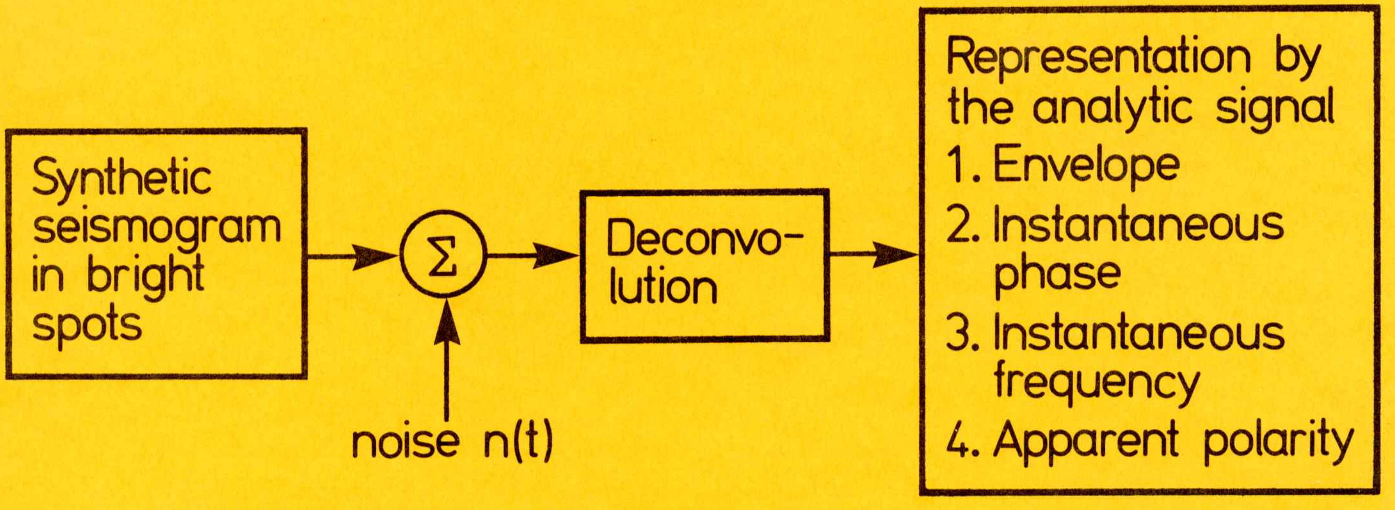
LARS Technical Report 081280
TR-EE 80-36

Analytic Signal Representation in the Synthetic Seismogram of Bright Spots

K. Y. Huang
C. D. McGillem
P. E. Anuta



Laboratory for Applications of Remote Sensing
Purdue University West Lafayette, Indiana 47906 USA
1980

LARS Technical Report 081280
TR-EE 80-36

ANALYTIC SIGNAL REPRESENTATION IN THE SYNTHETIC SEISMOGRAM OF BRIGHT SPOTS

By

K. Y. Huang
C. D. McGillem
P. E. Anuta

Laboratory for Applications of Remote Sensing
Purdue University
West Lafayette, Indiana 47907, U.S.A.

The research presented in this report was sponsored by the National Science Foundation under Grant ENG7820466.

TABLE OF CONTENTS

Page

APPENDICES

LIST OF FIGURES

ABSTRACT

Transformation of seismic signals into their analytic signal representation permits the unique separation of envelope, instantaneous phase, instantaneous frequency and apparent polarity. These parameters are useful in extracting the physical properties of a seismic signal and help in geophysical and geological interpretation. Deconvolution methods can improve the quality of the analytic seismic signal representation. From simulation studies it is found that time and space adaptive deconvolution is the best for preprocessing purposes. Minimum entropy deconvolution has polarity problems caused by noise. Removing the first and second order marine reverberations simplifies and reduces computation time in analytic signal processing.

CHAPTER ONE

INTRODUCTION TO SEISMIC SIGNAL PROCESSING

1.1 Introduction

Seismic petroleum exploration is more and more important in recent years and will continue so in the future because of the energy crisis. By 1972 oil companies had become successful in predicting the occurrence of offshore gas from seismic reflection data. These predictions were based on anomalies that would be expected in the amplitudes of reflections caused by differences between the reflectivity of surfaces bounding sands containing gas and those bounding water or oil-bearing portions of the sands.

The amplitude of a seismic wave reflected from an interface between two materials is governed by the reflection coefficient R which is expressed for normal incidence by the relation

$$R = \frac{\rho_2 V_2 - \rho_1 V_1}{\rho_2 V_2 + \rho_1 V_1} \qquad (1\text{-}1)$$

where ρ_1 and ρ_2 are the respective densities on the near (incident) and far sides of the boundary and V_1 and V_2 are the respective velocities for the two sides. The product of ρ and V is known as the acoustic impedance, and it is evident that the reflection coefficient and hence the reflection amplitude depend on the change in acoustic impedance across the reflecting interface. A high-amplitude portion of a seismic trace corresponding to a high reflection coefficient is referred to as a bright spot. The bright spots model which is employed here is shown in Figure 1.1 in an amplified form to show the variations more clearly. The gas sand zone has density D=1.9gm/cm^3, velocity V=1.7km/sec. The oil sand zone has density D=2.2gm/cm^3, velocity V=2.2km/sec.

When a dynamite explosion occurs at the bottom of a shot hole, this explosion creates a white signal (i.e., a signal containing all frequencies). However, since the earth is not a perfectly elastic medium, it distorts the signal, and selectively passes only a certain band of frequencies. This band-limited signal can be represented as an average wavelet. As this average wavelet travels into the earth, part of the energy is reflected back toward the surface at each layer interface, while the rest of the energy is transmitted deeper into the earth. We can represent this portion of reflected energy from each interface as a series of reflection coefficients at the vertical two-way travel time to each interface. Such a series of reflection coefficients we will term a reflectivity sequence. Because the incident wavelet produces components both up and down at each layer interface, there are reverberations and multiples also. All of these reflectivity sequence responses are called the subsurface response. The earth is like a filter. When the wavelet is convolved with the subsurface response, the output is called the synthetic seismogram (Claerbout, 1968; Treitel and Robinson, 1966). The synthetic seismogram of the bright spot model of Figure 1.1 is shown in Figure 1.2. The average wavelet used here consists of a single sinusoidal cycle. The duration of the average wavelet is 0.024 second and a sampling rate of 0.004 second/data interval is employed. The synthetic seismogram in Figure 1.2 corresponds to the situation where the explosion and the receiving instrument are at the same position for the 13 stations shown.

In the synthetic seismogram of Figure 1.2 we can look for some of the following geophysical properties of the subsurface structure. (1) As the reflection coefficient from the top or bottom surface of a gas sand is greater than that from the corresponding surface of an oil-bearing sand, the presence of the gas sand should be indicated on the section as a bright spot. (2) High amplitude is related to the high reflection coefficient at the top boundary of the gas sand and at the boundary of the gas sand and oil sand, because there occurs at these interfaces a large difference in acoustic impedance. The relation between reflection coefficient and acoustic impedance is given by Equation (1-1).

$$R = \frac{\rho_2 V_2 - \rho_1 V_1}{\rho_2 V_2 + \rho_1 V_1}$$

(3) Phase reversals are produced by a negative reflection coefficient at the top boundary of gas sand and again by the positive reflection coefficient at the bottom boundary of oil sand. (4) Velocity pulldown effects: The low velocity of gas sand compared to oil sand gives rise to lateral anomalies

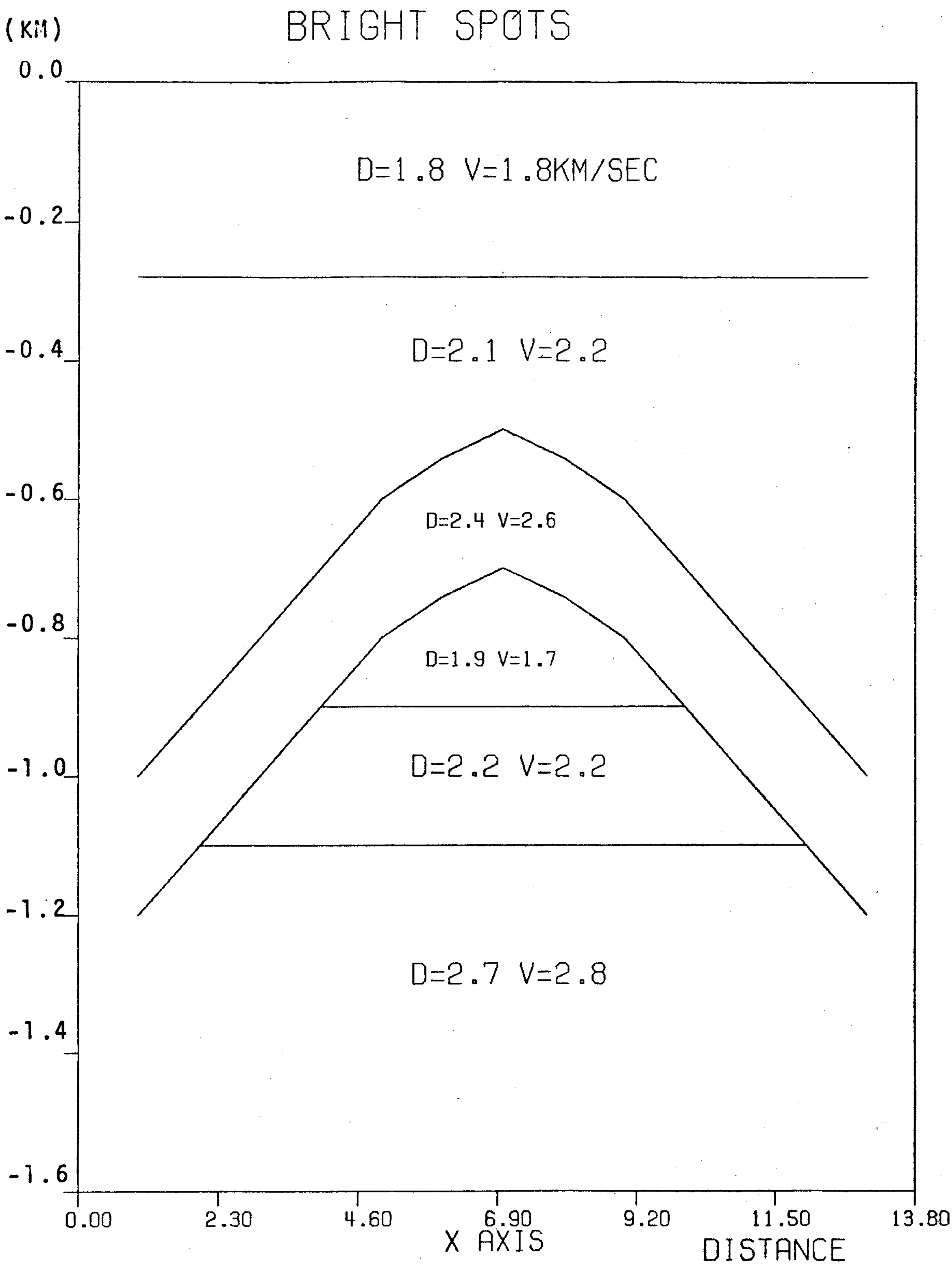


Figure 1.1 Bright spot model (amplified)

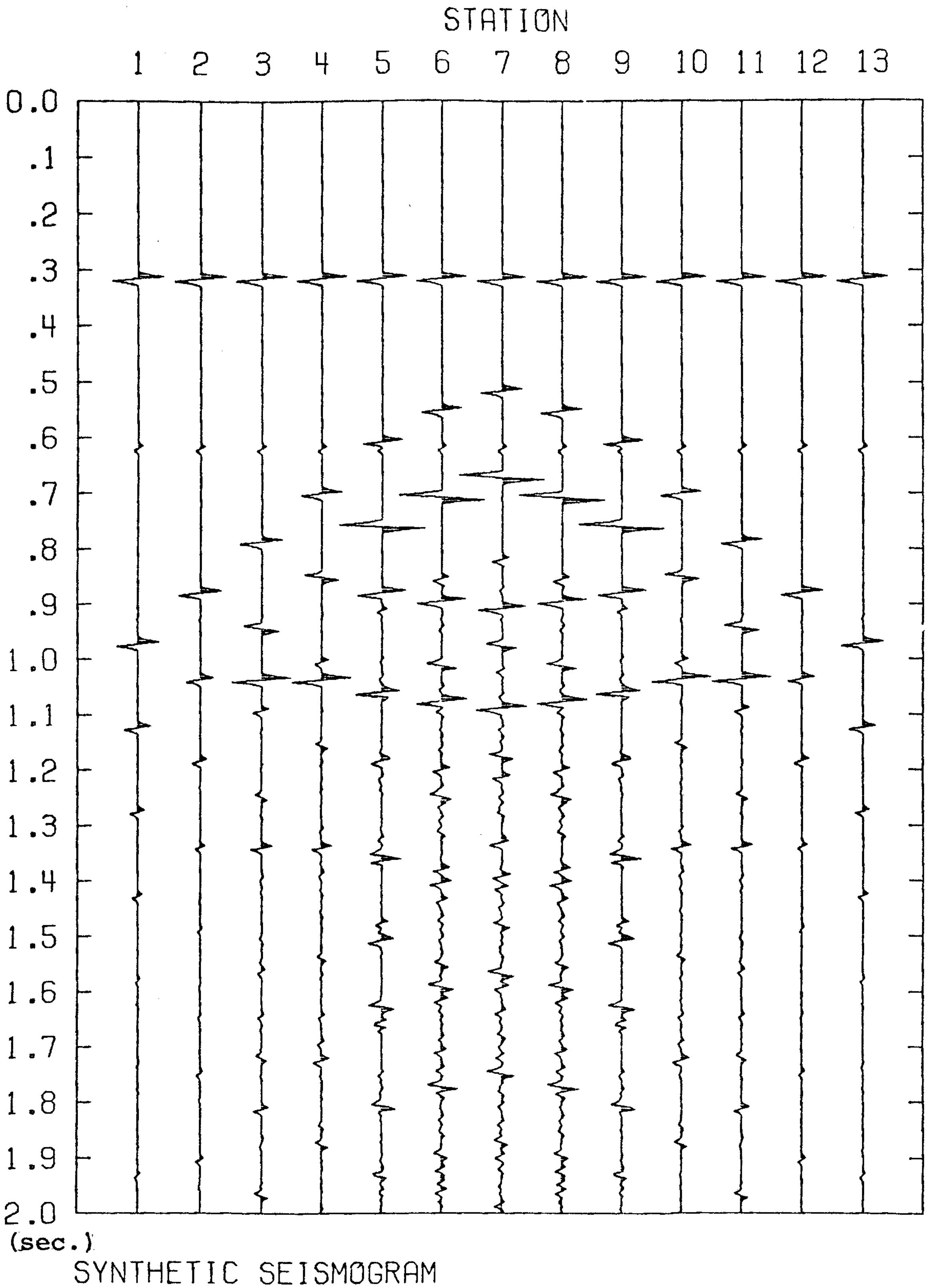


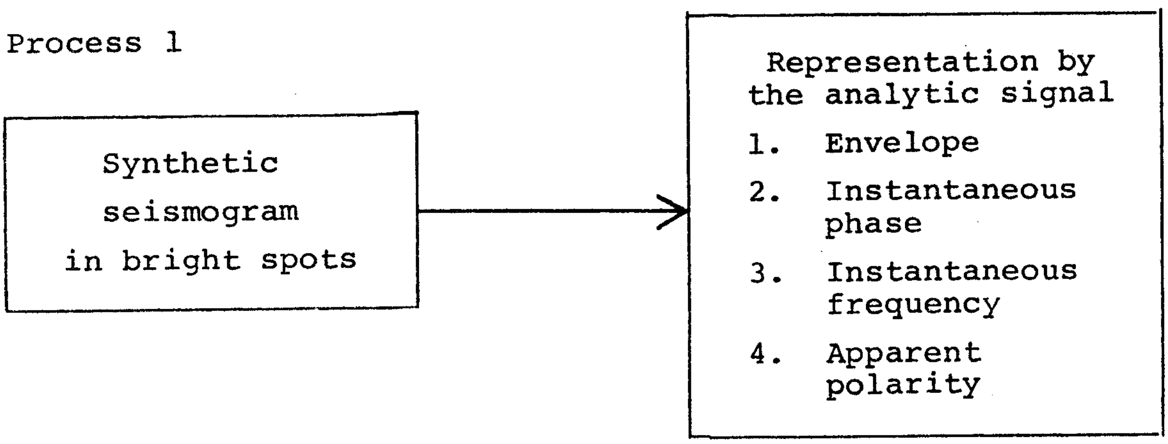
Figure 1.2 Synthetic seismogram of the bright spot model of Figure 1.1

not only in the reflection coefficients from surfaces bounding gas sand but also in the time required for seismic waves to pass through the sand. Figure 1.2 illustrates this effect. In Figure 1.1 the bottom of the gas sand and the oil sand are horizontal, but in Figure 1.2 they are pulled down because of low velocity in the gas sand zone. It requires the most travel time and affects the reflection shape of oil sand boundaries.

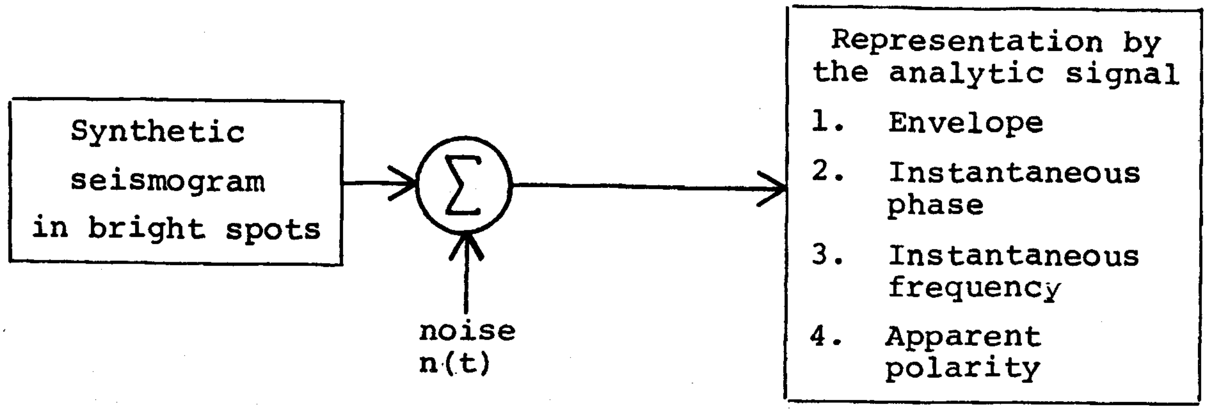
In seismic data processing, signal is defined as the P wave (longitudinal wave or compression wave) reflection of true geologic structure and it is used to interpret the subsurface geology. The noise includes coherent noise (reverberations, multiples, shear waves, surface waves, etc.) and random noise.

The seismic signal processing techniques to be considered in this thesis are the following:

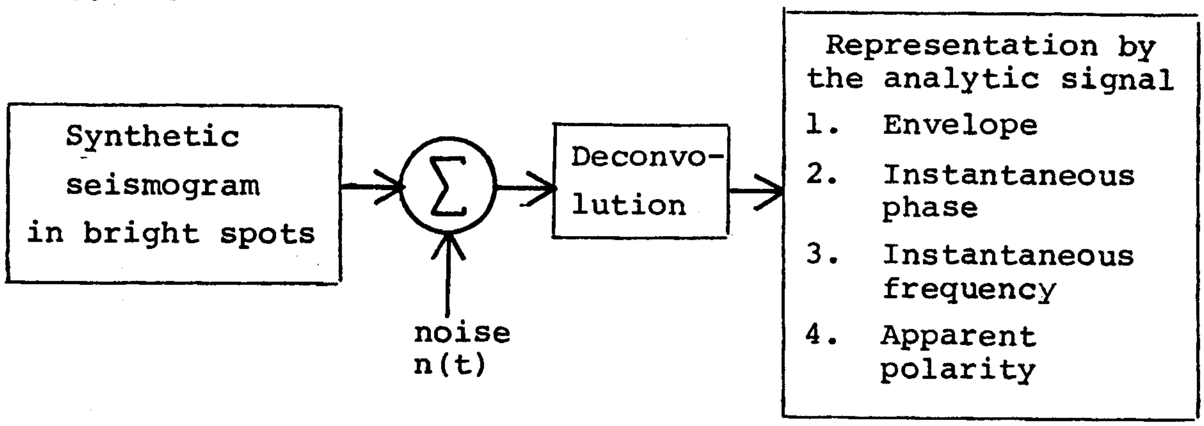
Process 1



Process 2



Process 3



where the synthetic seismogram is given by

$$G = W * R \qquad (1\text{-}2)$$

where $*$ = convolution

W = seismic wavelet

R = subsurface response

In the above processes, there are two objectives: (1) illustrate the importance of the deconvolution operation, and (2) to explain the application of analytic seismic signals in the detection of bright spots in petroleum exploration. The bright spot model used with these processes is shown in Figure 1.1, and its noise-free synthetic seismogram in Figure 1.2.

1.2 Deconvolution

The reflection signal can be represented as the convolution of the wavelet with the subsurface response. If we could recover the earth's response, we could relate it directly to earth's structure. This process of recovering the earth's response is called deconvolution. Obviously, if we could represent each layer interface by a sharp spike, it would be possible to map these interfaces very accurately, thus obtaining maximum resolution.

The deconvolution methods used here are time and space adaptive deconvolution, adaptive deconvolution using gradient descent, and minimum entropy deconvolution. The purposes of using deconvolution here are to spike the wavelet and to suppress the noise. Another purpose of the deconvolution process is to improve the quality of the analytic signal representation.

1.3 New technique in analytic signal processing

Farnbach (1975) used the analytic signal representation to separate the envelope information from angle (including instantaneous phase and frequency) information in teleseismic analysis. Sicking (1978) used the envelope and instantaneous frequency in the synthetic seismogram of the wedge model and interfingering model. Taner and Sheriff (1977) and Taner, Koeler and Sheriff (1979) used the representation of the seismic analytic signal, envelope, instantaneous phase, instantaneous frequency and apparent polarity to help in identifying the gas and oil accumulations in the bright spots of real seismic data.

In Chapter 3, a new technique in analytic signal processing designed to remove the first and second order marine reverberations will be presented. In Chapter 4, new results in connection with the instantaneous phase will also be presented.

CHAPTER TWO

MATHEMATICAL TECHNIQUE FOR DECONVOLUTION

2.1 Introduction

Conventional Wiener minimum mean square error filters are used in reflection seismic processing for wavelet shaping, wavelet spectrum whitening, and removal of multiple reflections, under the assumption of statistical stationarity of the waveforms used in the filter design. Due to dispersion, attenuation and lateral variation, the assumption of the stationarity of the seismic waveforms is seldom fully justified in reflection processing. Use of the adaptive deconvolution and minimum entropy deconvolution are employed in the analysis of non-stationary reflection signals.

Peacock and Treitel (1969) demonstrate that the Wiener minimum mean-square error deconvolution filter, which ideally transforms an unknown signal into an impulse at zero delay, is equivalent to the prediction error filter for which the prediction distance α is unity. It is used here in adaptive deconvolution for spiking the signal.

2.2 Time and space adaptive deconvolution

Riley (1973) described the adaptive deconvolution technique. This procedure is used here only for its spiking characteristics. Time and space adaptive deconvolution is derived from the Burg technique (Claerbout, 1976). The procedure is to sequentially process the entire length of data, continuously updating the filter at every time point. The system is self-optimizing in the face of a changing environment, i.e., in the presence of non-stationarity. To build an adaptive realization of the optimum prediction error operator, it is required to: (1) make the first filter weight be 1 (Peacock and Treitel, 1969); and (2) continuously adjust the parameters of the system to minimize the average squared-output.

Using Burg's method, the recursive solution of the prediction error filter is the following:

For the two point case

$$\begin{bmatrix} \phi(0) & \phi(1) \\ \phi(1) & \phi(0) \end{bmatrix} \begin{Bmatrix} 1 \\ C_1 \end{Bmatrix} = \begin{bmatrix} P_2 \\ 0 \end{bmatrix} \qquad (2\text{-}1)$$

For the three point case

$$\begin{bmatrix} \phi(0) & \phi(1) & \phi(2) \\ \phi(1) & \phi(0) & \phi(1) \\ \phi(2) & \phi(1) & \phi(0) \end{bmatrix} \left\{ \begin{bmatrix} 1 \\ C_1 \\ 0 \end{bmatrix} + C_2 \begin{bmatrix} 0 \\ C_1 \\ 1 \end{bmatrix}^* \right\} = \left\{ \begin{bmatrix} P_2 \\ 0 \\ \Delta_2 \end{bmatrix} + C_2 \begin{bmatrix} \Delta_2 \\ 0 \\ P_2 \end{bmatrix} \right\} = \begin{bmatrix} P_3 \\ 0 \\ 0 \end{bmatrix} \qquad (2\text{-}2)$$

where

$\phi(\cdot)$ - auto-correlation function

* - complex conjugate

$$\begin{bmatrix} 1 \\ C_1 \\ 0 \end{bmatrix} = \text{Forward prediction error filter}$$

$$\begin{bmatrix} 0 \\ C_1 \\ 1 \end{bmatrix}^* = \text{Backward prediction error filter}$$

$$\Delta_2 = \phi(2) + \phi(1)C_1^*, \quad C_2 = {}^{-\Delta_2}/_{p_2}, \quad p_3 = p_2(1-C_2^2)$$

Thus, the three-point prediction error filter is obtained from the known two-point filter by minimizing the output power, thereby giving C_2.

The filter operator is then $\begin{bmatrix} 1 \\ C_1 \\ 0 \end{bmatrix} + C_2 \begin{bmatrix} 0 \\ C_1 \\ 1 \end{bmatrix}^*$

The power output is

$$P_2\ (C_2) = \sum_{t=2}^{N} |\ x_t + C_1\ x_{t-1} + C_2(C_1^*\ x_{t-1} + x_{t-2})|^2$$

$$+ |\ x_{t-2}^* + C_1\ x_{t-1}^* + C_2(C_1^*\ x_{t-1}^* + x_t^*)|^2 \qquad (2\text{-}3)$$

Setting

$$F_2(t) = x_t + C_1\ x_{t-1}$$

$$B_2(t) = C_1^*\ x_{t-1} + x_{t-2}$$

It follows that

$$P_2(C_2) = \sum_t \mid F_2(t) + C_2\ B_2(t) \mid^2 + \mid B_2^*(t) + C_2\ F_2^*(t) \mid^2$$

$$= \sum_t (F_2(t) + C_2\ B_2(t))^* (F_2(t) + C_2\ B_2(t))$$

$$+ (B_2^*(t) + C_2\ F_2^*(t))^* (B_2^*(t) + C_2\ F_2^*(t)) \tag{2-4}$$

Setting the derivative with respect to C_2 equal to zero,

$$C_2 = - \frac{\sum_t 2F_2(t)\ B_2^*(t)}{\sum_t F_2^*(t)\ F_2(t) + B_2^*(t)\ B_2(t)} , \mid C_2 \mid < 1 \tag{2-5}$$

Gives the three-point prediction error filter, which can now be redefined as

$$F_3(t) \leftarrow F_2(t) + C_2\ B_2(t)$$

$$B_3(t) \leftarrow B_2(t) + C_2^*\ F_2(t) \tag{2-6}$$

From recursion, the power output of the j-th stage is

$$P_j = \sum_{k=1}^{N} [\ F_j(k) + C_j\ B_j(k)]^2 + [\ C_j\ F_j^*(k) + B_j^*(k)]^2 \tag{2-7}$$

Minimizing this expression leads to the following expression for C_j

$$C_j = \frac{-\Sigma 2F_j(k)\ B_j^*(k)}{\Sigma\ |\ F_j(k)\ |^2 + |\ B_j(k)\ |^2} \qquad |\ C_j\ | < 1 \qquad (2\text{-}8)$$

In the time and space adaptive deconvolution procedure, for every step the C_j is recomputed from the weighted average cross-power and auto-power of past values of the forward and backward errors, i.e., the $F_j(K)$ and $B_j(K)$ are multiplied by the time and space weighting function

$$W\ (k,m) = \exp\ (\ -\ k^{\Delta T}/_{T_T}\ -\ {}^{m}/_{T_X}\ ) \qquad (2\text{-}9)$$

where T_T = relaxation time (second)

T_X = relaxation distance (traces)

Doing the computer work in time and space adaptive deconvolution, we start from Figure 2.1, the synthetic seismogram plus random noise. The synthetic seismogram without random noise is the same as Figure 1.2. The random noise is uncorrelated in traces 1 through 7. Because the bright spot model is symmetrical in Figure 1.1, the synthetic seismogram is symmetrical also with respect to trace 7. Therefore it is reproduced as a mirror image in traces 8 through 13 to save computer processing time. We can compare the results from trace 1 to trace 7. Some parameters are explained as follows:

LCN = filter length

XTAU = T_X relaxation distance

ZTAU = T_T relaxation time

IFLGAP = whitening spectrum portion

S/N = the ratio of the root-mean-square of the signal to that of the noise in the time interval 0.3 to 0.324 second, 0.96 to 0.984 second, 1.108 to 1.132 second.

Some significant computer results of the time and space adaptive deconvolution are the following:

1. Effect of filter length. The length of the deconvolution filter has a significant effect on the noise present in the output. Comparing Figure 2.2 (LCN = 12) with Figure 2.3 (LCN = 3), it is seen that the noise between the spikes is much greater in the output of the filter with the shorter length. In general, the greater the length, the greater the noise reduction. However, as the filter length is increased, the processing time also increases and the cost of processing the data increases. There is, therefore, a tradeoff between quality and cost.

2. Effect of the relaxation time. There is no lateral variation in the synthetic seismogram plus noise. Set m = 0 in Equation (2-9). Then compare the time and space adaptive deconvolution outputs in Figure 2.4, relaxation time T_T = 0.08 sec.; Figure 2.5, T_T = 0.16 sec.; and Figure 2.6, T_T = 0.32 sec. At trace 7, in time interval 1.1-1.2 sec., Figure 2.6, T_T = 0.32 has less noise than traces in Figure 2.4 and 2.5.

3. Tolerance of time and space adaptive deconvolution. In Figure 2.7 S/N = 2.22; Figure 2.9, S/N = 1.48; Figure 2.11, S/N = 1.03; Figure 2.13, S/N = 0.74, and their respective deconvolution outputs in Figures 2.8, 2.10, 2.12, and 2.14, when S/N is below 1.48, there are distortions. When the magnitude of the random noise approaches that of the signal, there will be distortion.

2.3 Adaptive deconvolution using gradient descent

Griffiths *et al* published the technique of adaptive deconvolution using gradient descent in 1977. This technique uses a time-continuous adaptive procedure similar to that employed extensively in communications engineering and antenna array processing. In the present application, which is spiking deconvolution, the prediction distance is 1.

The normal equations for a Wiener minimum mean-square error deconvolution filter are given by

$$\begin{bmatrix} r_x(0) & r_x(1) & & r_x(L-1) \\ r_x(1) & r_x(0) & & \\ & & & \\ r_x(L-1) & & & r_x(0) \end{bmatrix} \begin{bmatrix} f_o(0) \\ f_o(1) \\ \\ f_o(L-1) \end{bmatrix} = \begin{bmatrix} r_x(T) \\ r_x(T+1) \\ \\ r_x(T+L-1) \end{bmatrix} \qquad (2\text{-}10)$$

and can be expressed in matrix notation as

$$\underline{R}_{xx}\,\underline{f}_o = \underline{p}_x(T) \qquad (2\text{-}11)$$

where $r_x(\ell)$ is the trace autocorrection at lag l, $f_o(\ell)$ is the l-th prediction coefficient and $T \geq 1$ is the prediction dis-

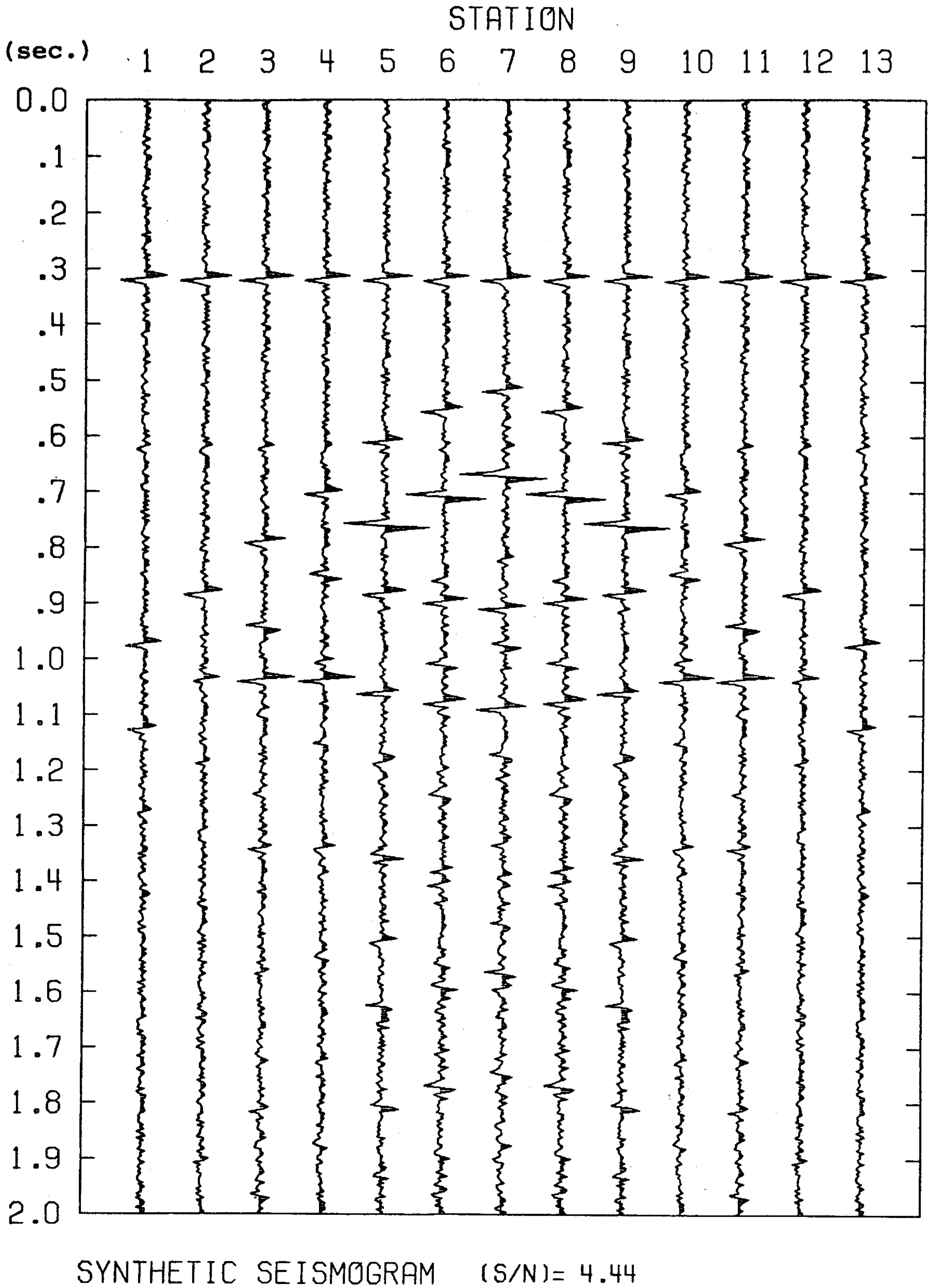


Figure 2.1 Synthetic seismogram plus random noise S/N=4.44

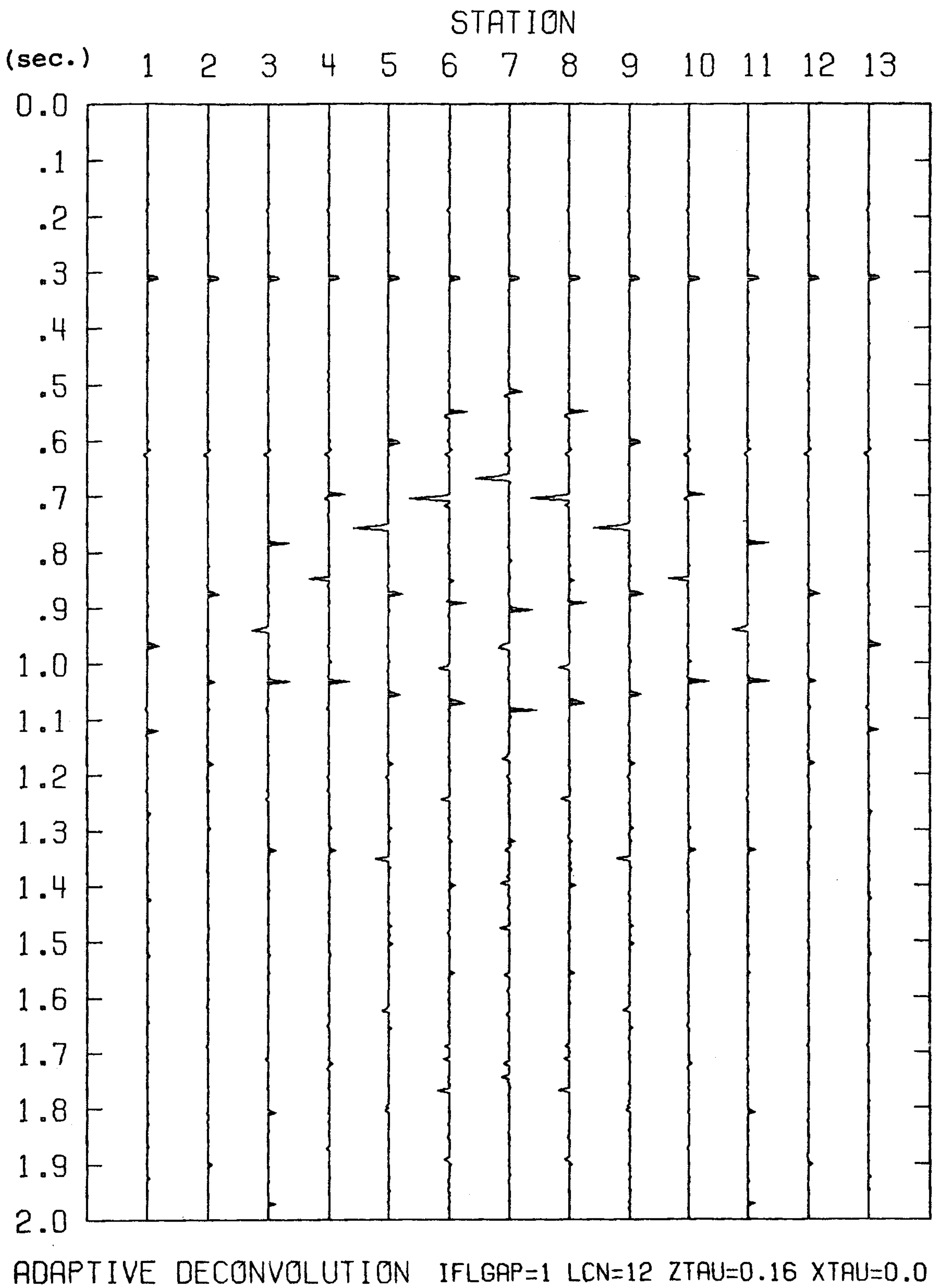


Figure 2.2 Effect of time and space adaptive deconvolution filter length LCN=12

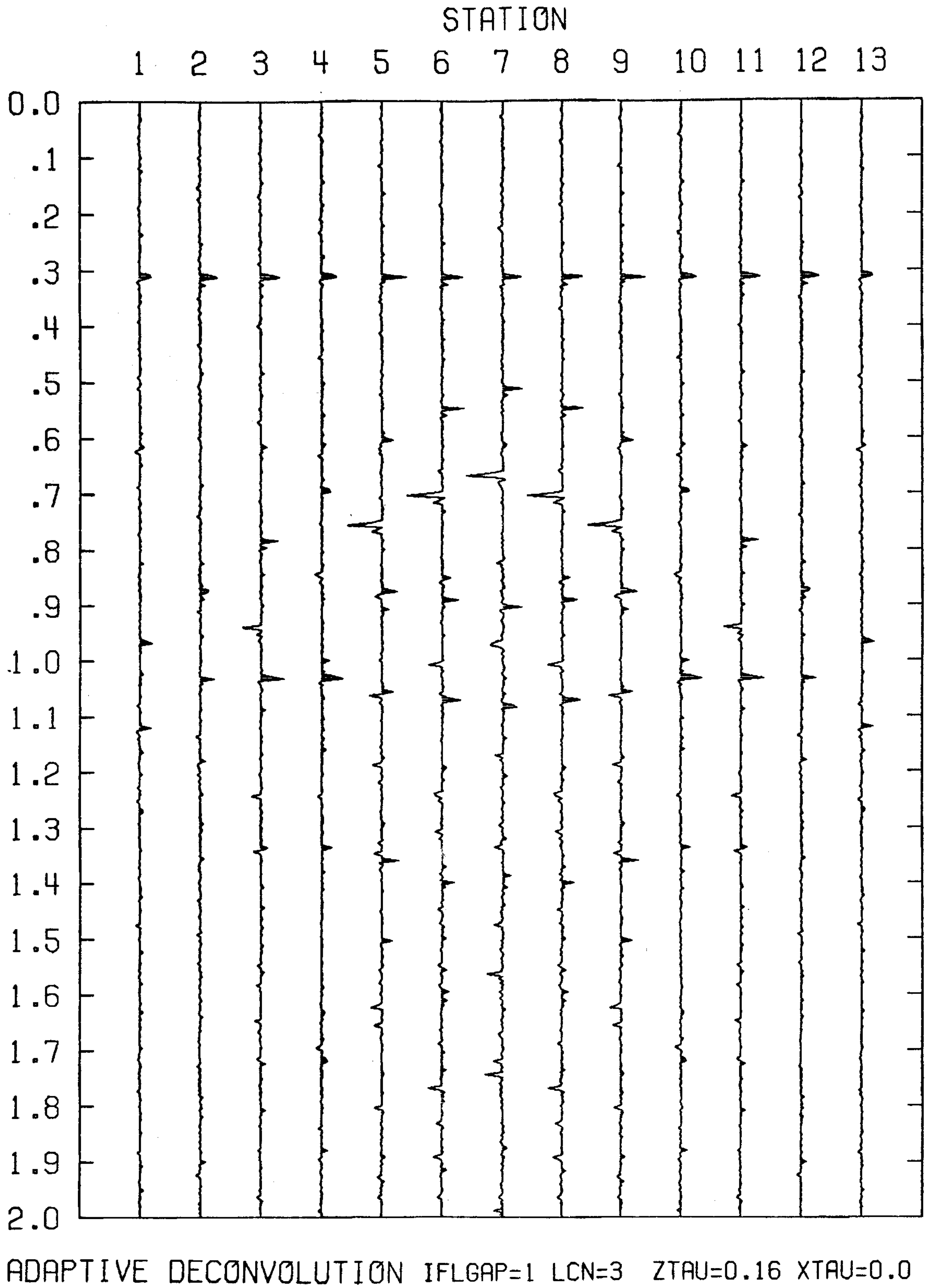


Figure 2.3 Effect of time and space adaptive deconvolution filter length LCN=3

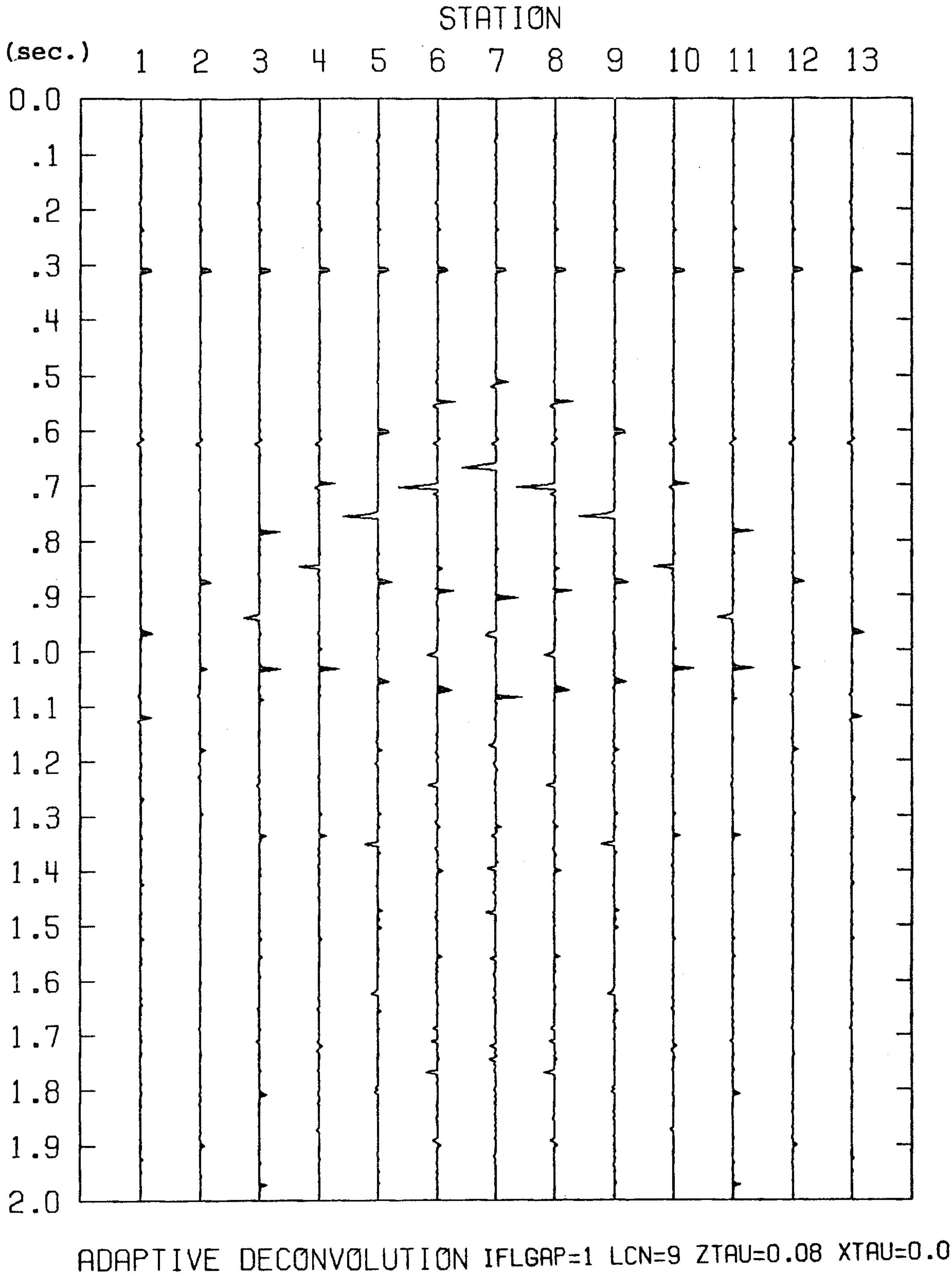


Figure 2.4 Effect of relaxation time T_T=0.08 second

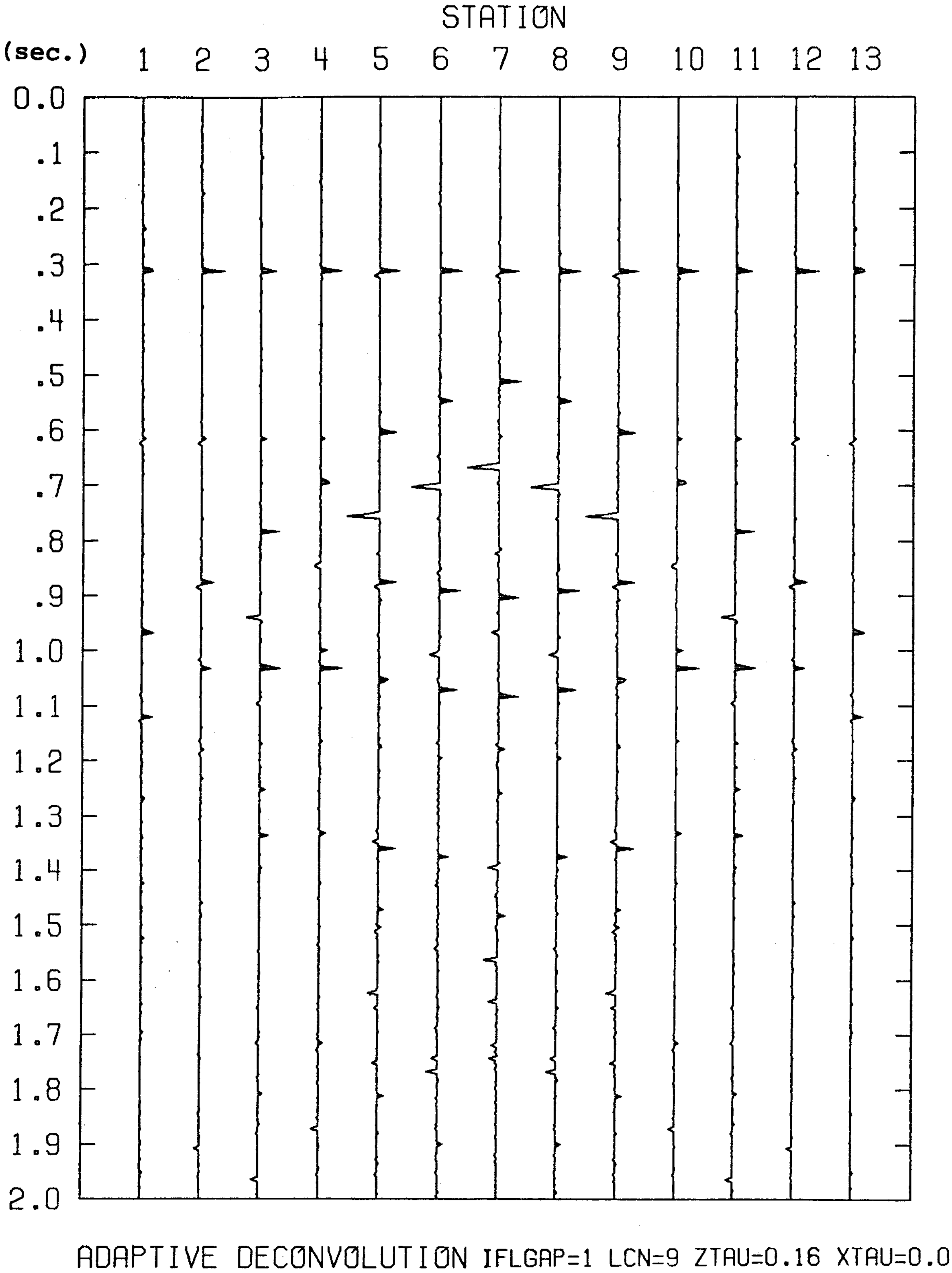


Figure 2.5 Effect of relaxation time T_T=0.16 second

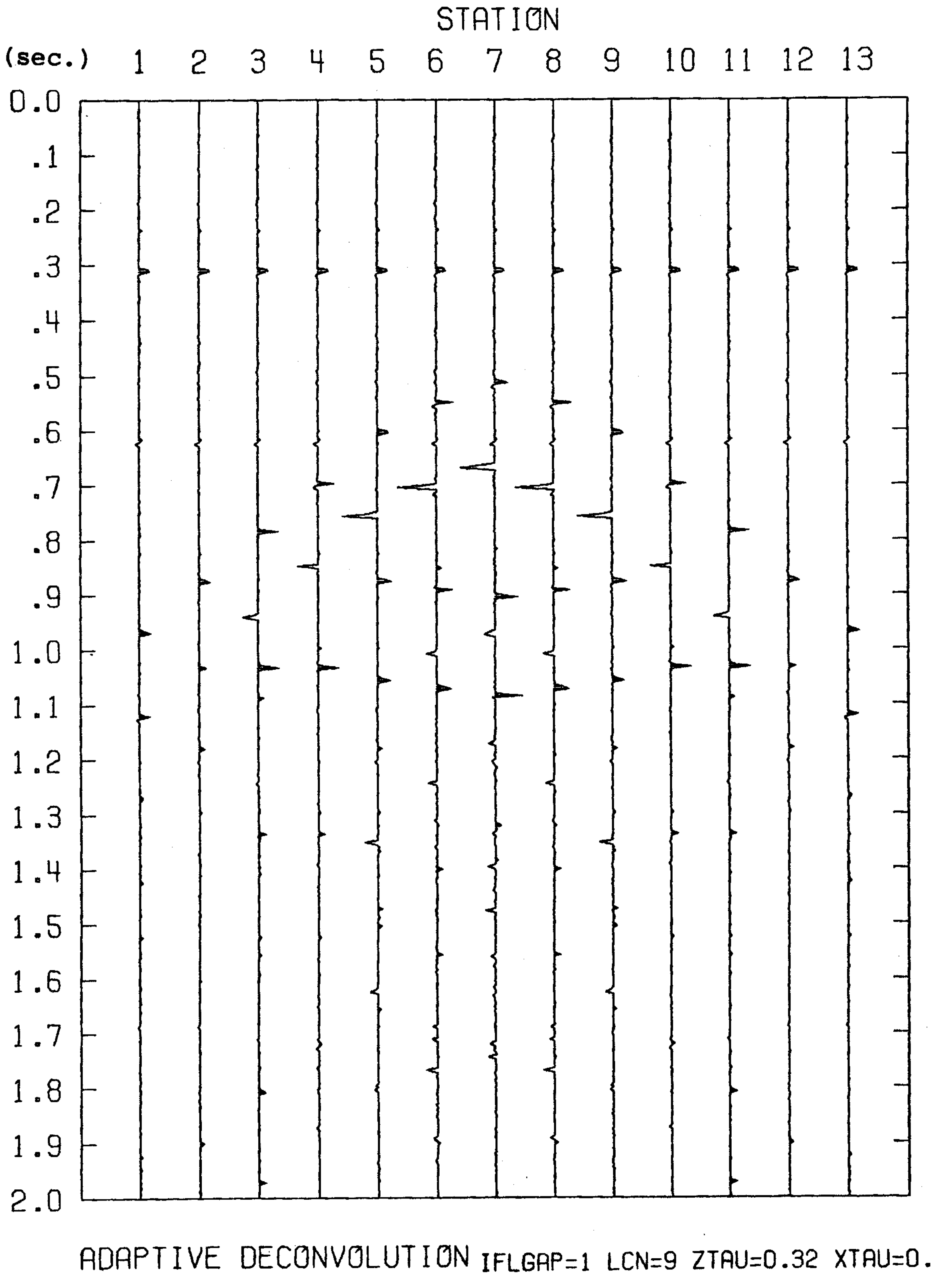


Figure 2.6 Effect of relaxation time T_T=0.32 second

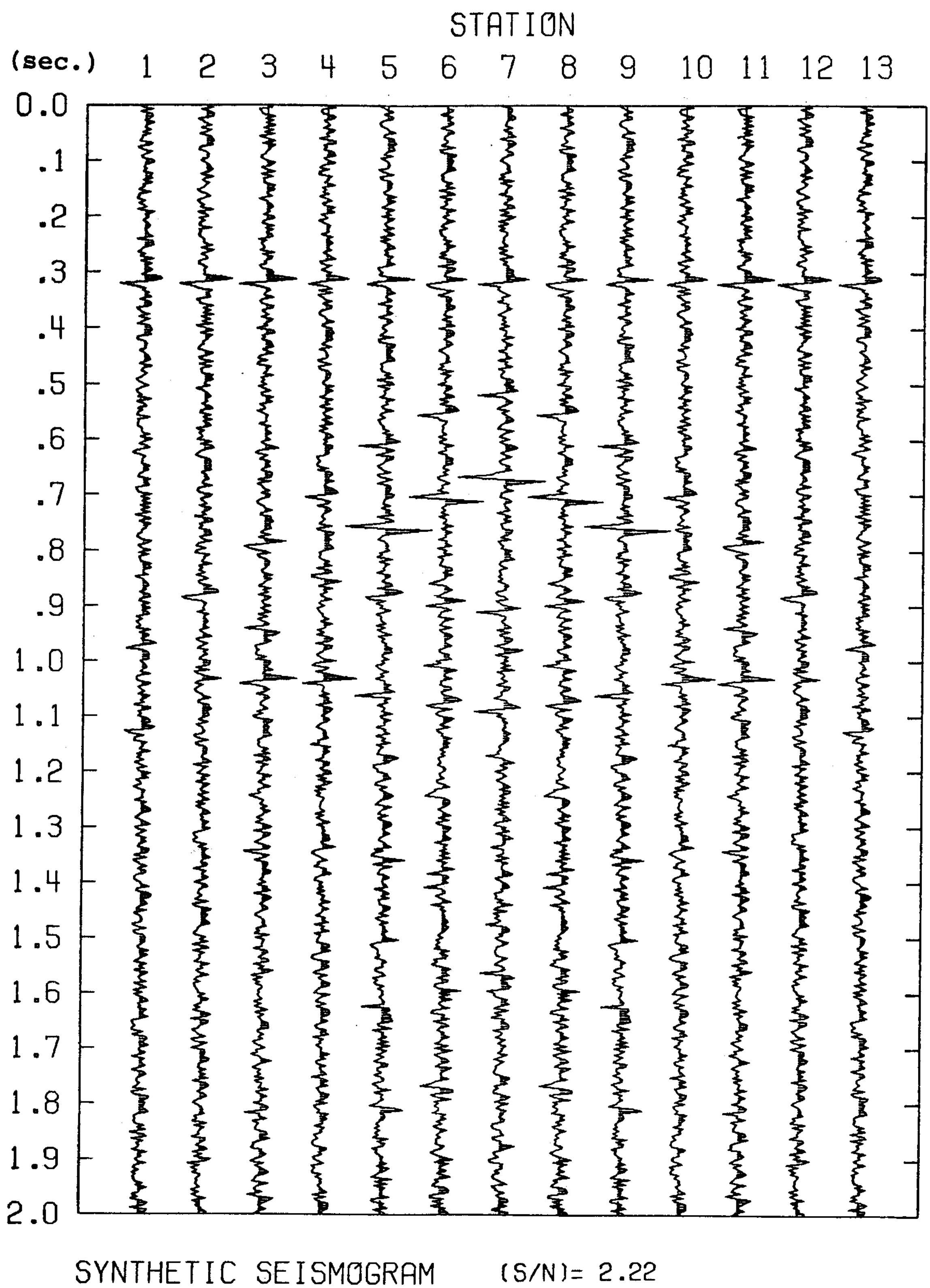


Figure 2.7 Synthetic seismogram plus random noise S/N=2.22

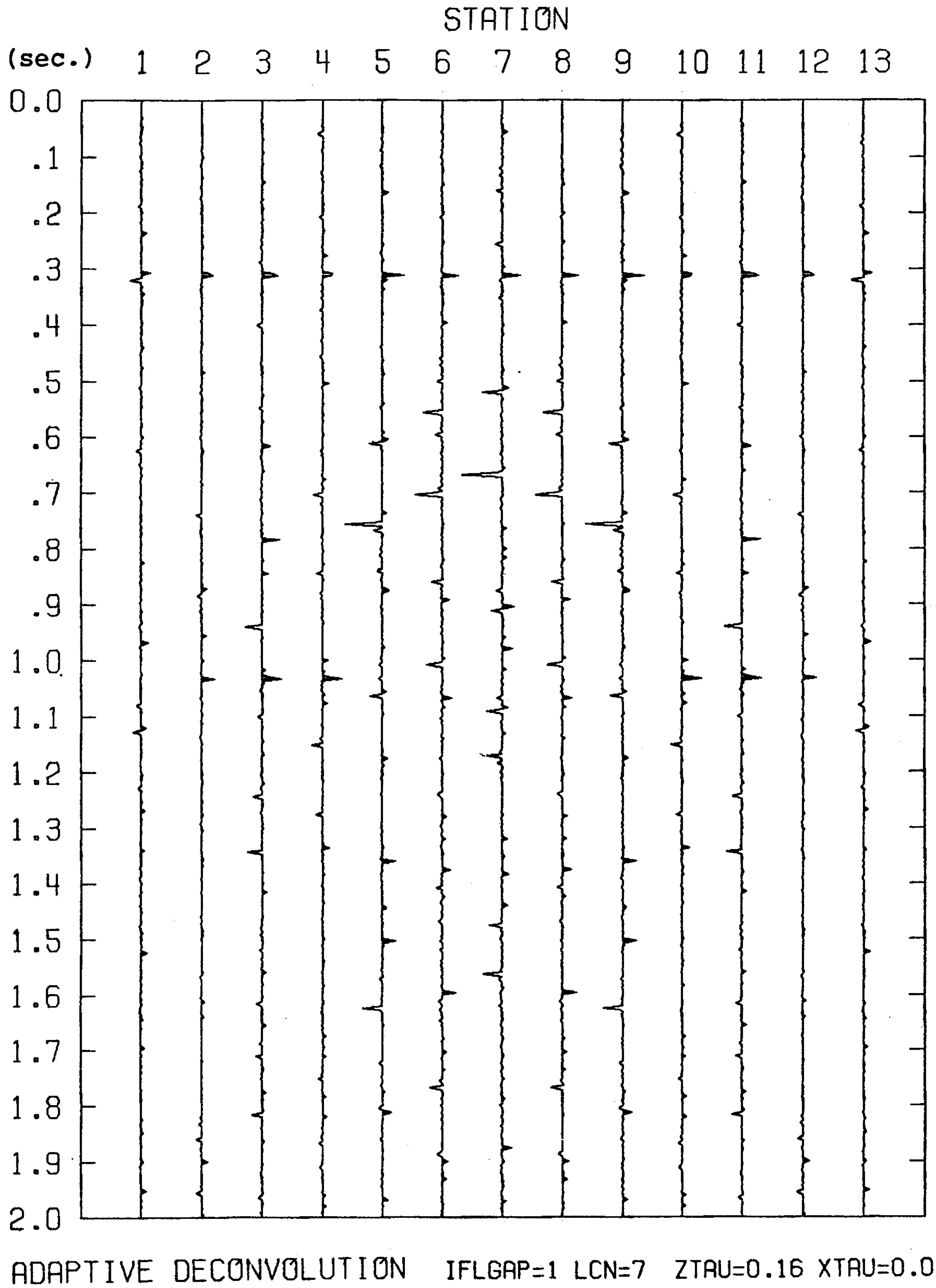


Figure 2.8 **Time and space adaptive deconvolution output of Figure 2.7 S/N=2.22**

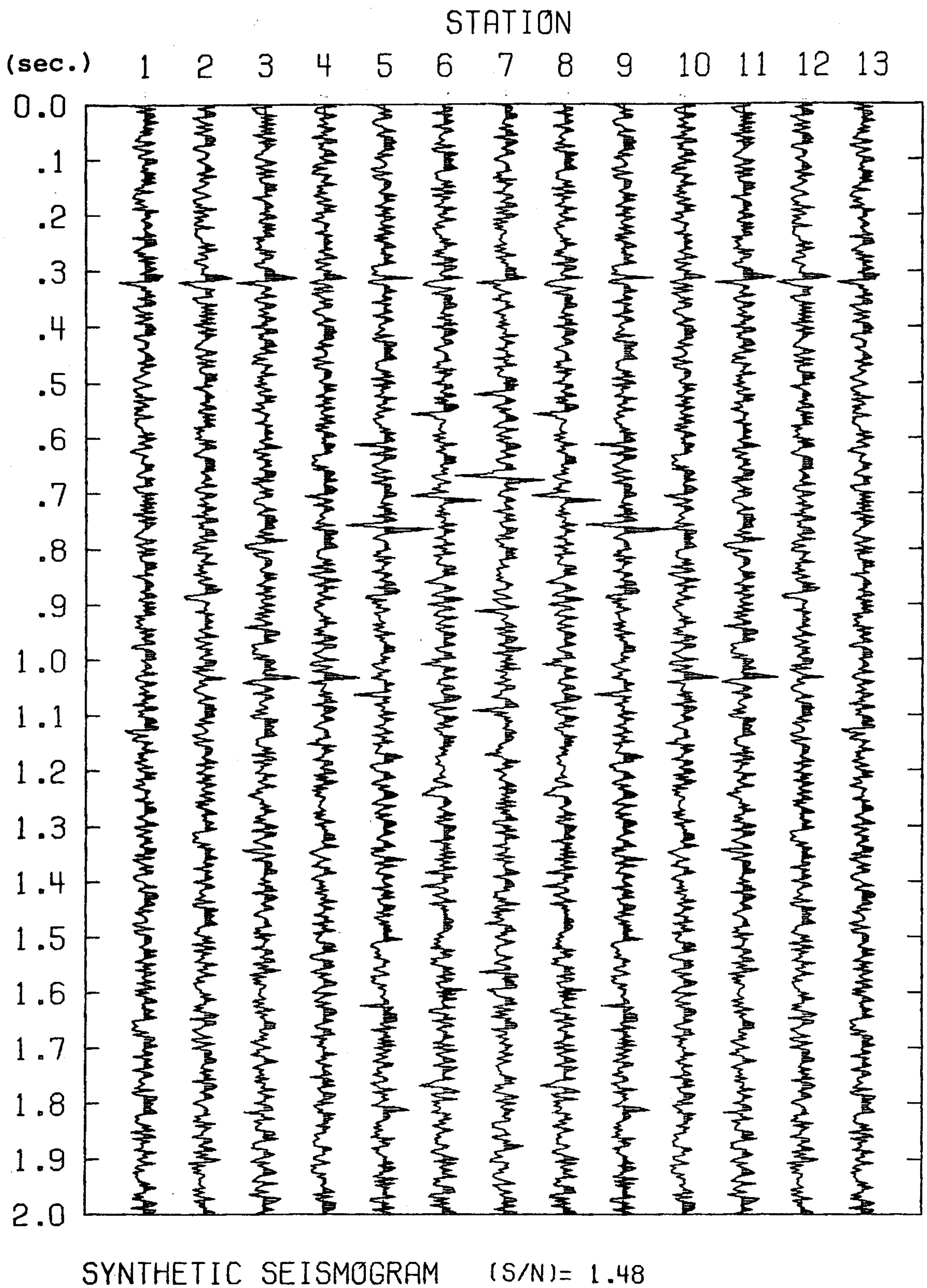


Figure 2.9 Synthetic seismogram plus random noise S/N=1.48

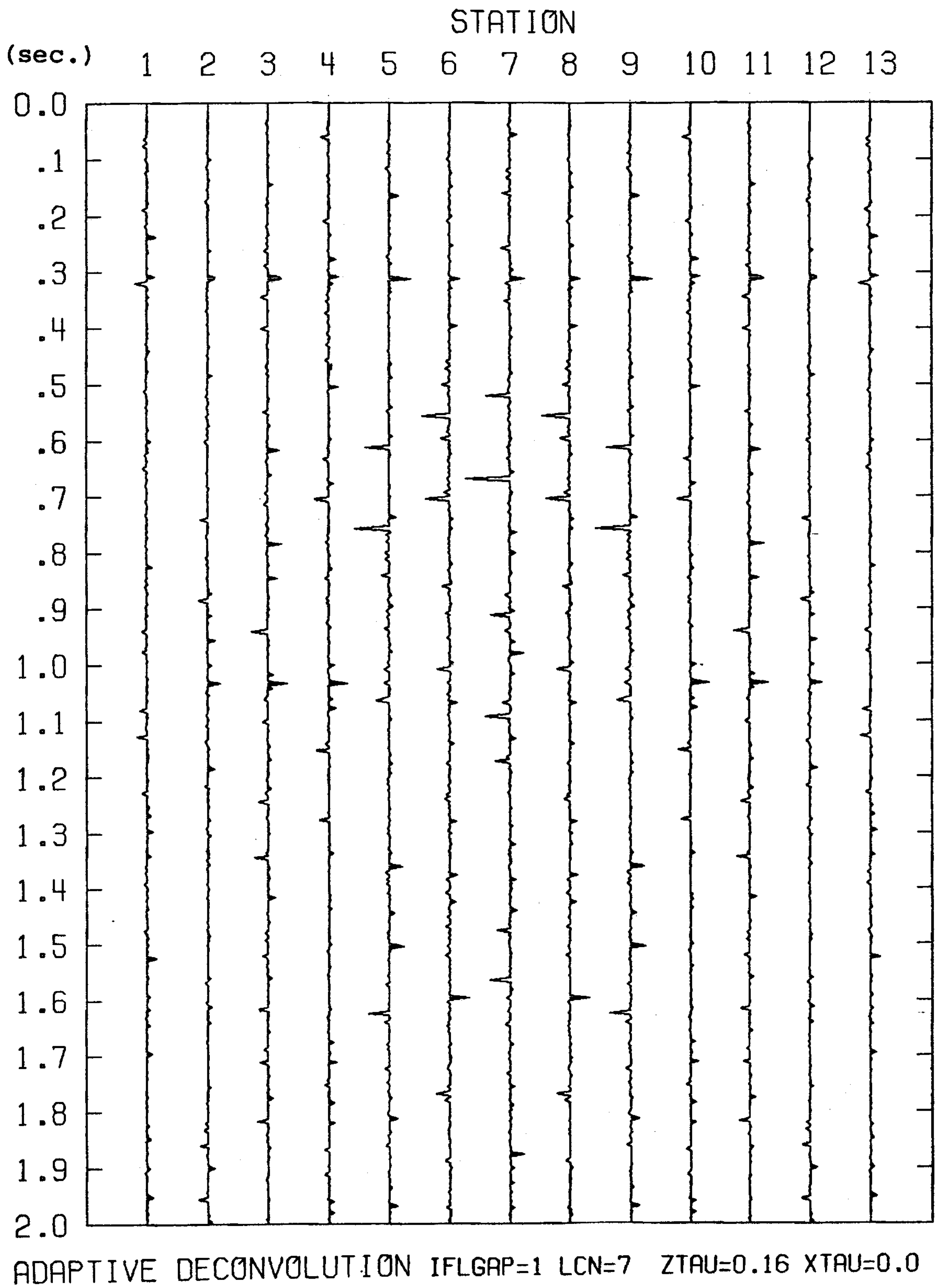


Figure 2.10 Time and space adaptive deconvolution output of Figure 2.9 S/N=1.48

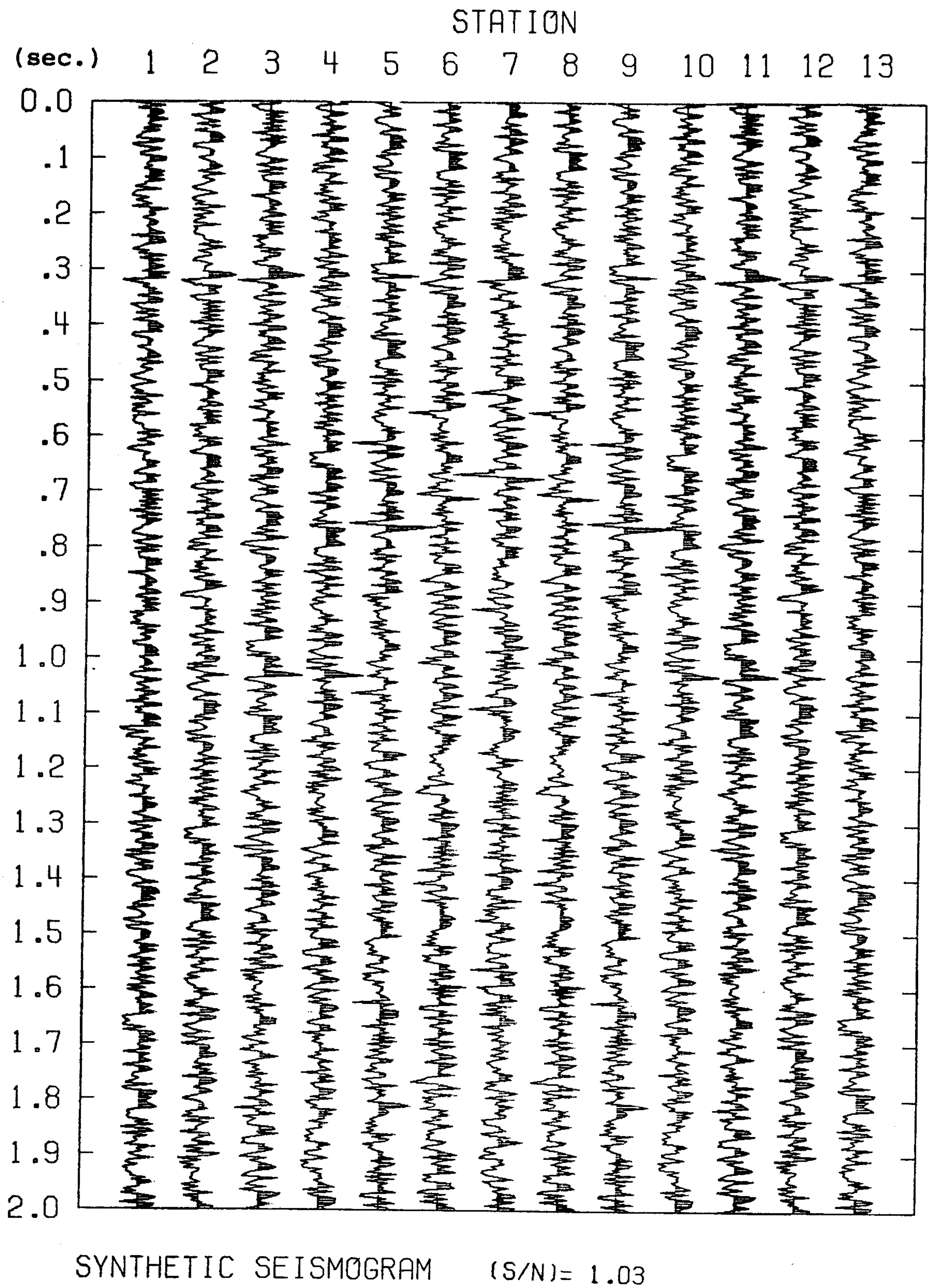


Figure 2.11 Synthetic seismogram plus random noise S/N=1.03

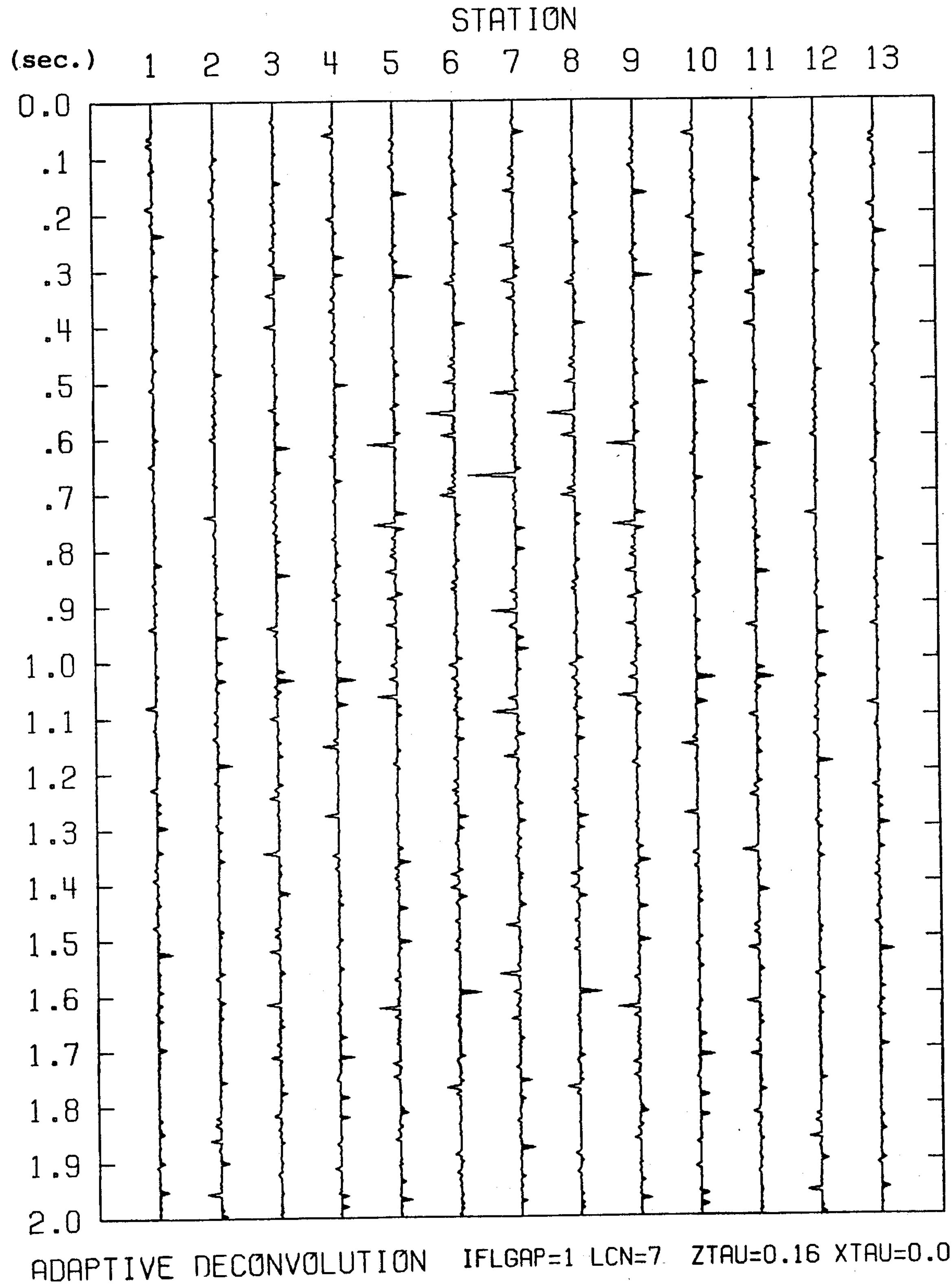


Figure 2.12 Time and space adaptive deconvolution output of Figure 2.11 S/N=1.03

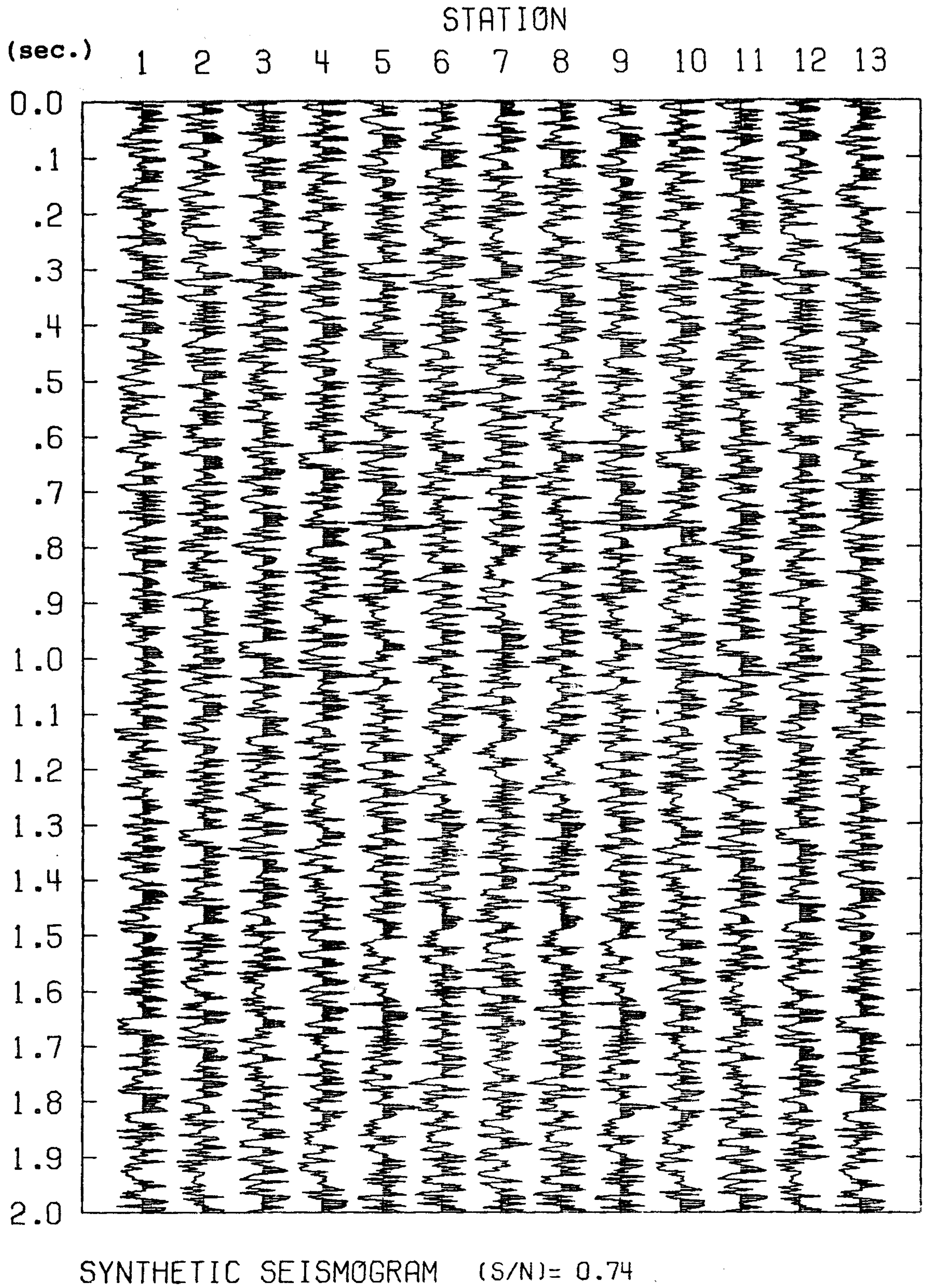


Figure 2.13 Synthetic seismogram plus random noise S/N=0.74

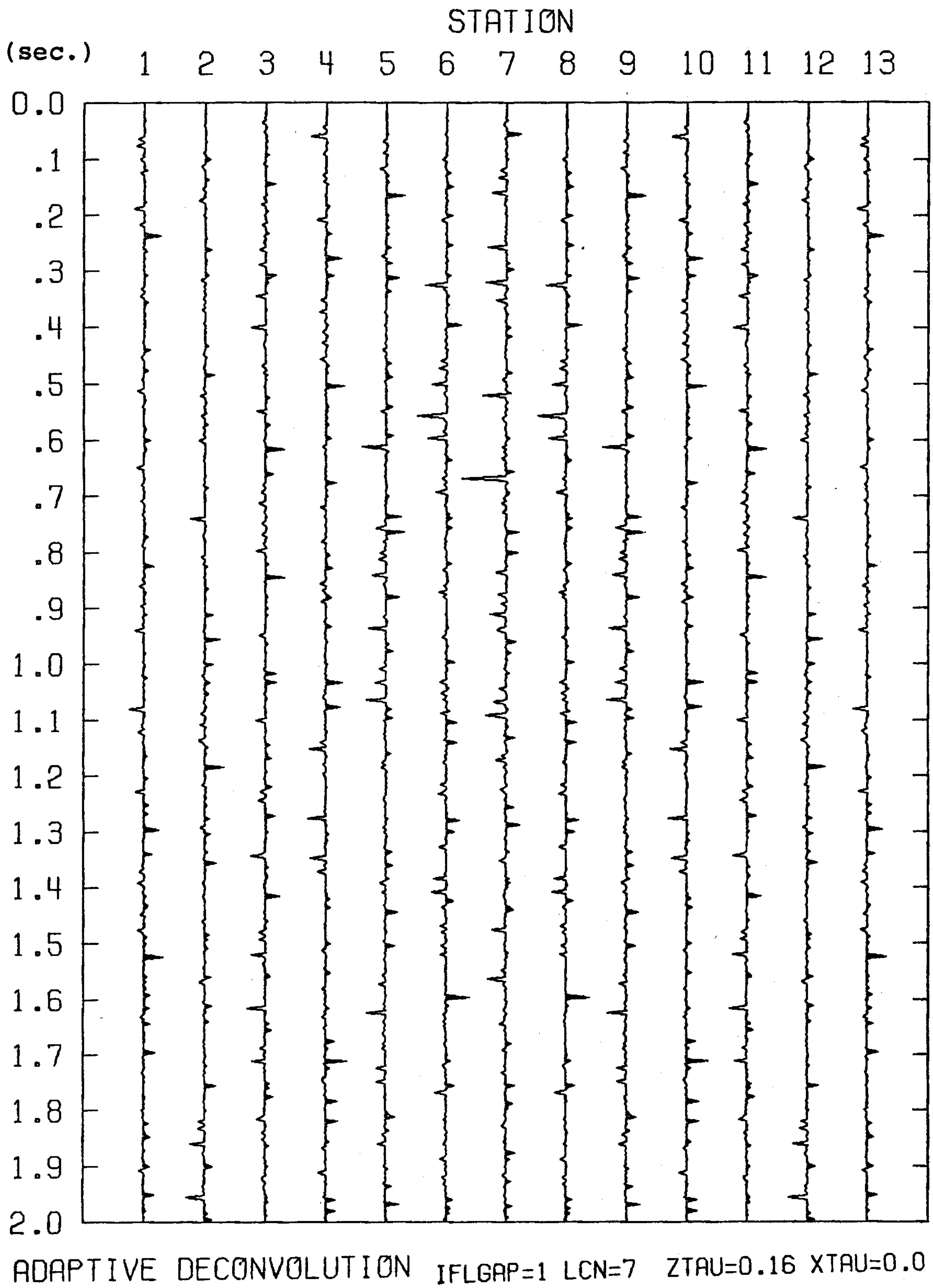


Figure 2.14 Time and space adaptive deconvolution output of Figure 2.13 S/N=0.74

tance, $\underline{R}_{xx}$ is the expected value of $\underline{x}(t)\underline{x}^T(t)$ and $\underline{p}_x(T)$ is the expected value of $x(t+T)\underline{x}(t)$. Using gradient descent, the procedure is to start with an arbitrary initial guess $\underline{f}(0)$ for the prediction coefficient and then update $\underline{f}(t)$ using the following algorithm:

$$\underline{f}(t+1) = \underline{f}(t) + \mu[\underline{p}_x(T) - \underline{R}_{xx}\ \underline{f}(t)] \quad (2\text{-}12)$$

where μ is the adaptive coefficient which determines the rate of convergence of $\underline{f}(t)$ toward $\underline{f}_o$, the ideal filter. $\underline{p}_x(T)$ and $\underline{R}_{xx}$ can be replaced by their instantaneous values

$$\underline{p}_x(T) \rightarrow x(t+T)\ \underline{x}(t) \qquad \underline{R}_{xx} \rightarrow \underline{x}(t)\ \underline{x}^T(t)$$

$$\rightarrow \underline{f}(t+1) = \underline{f}(t) + \mu[x(t+T) - \hat{x}(t+T)]\ \underline{x}(t) \quad (2\text{-}13)$$

where $\hat{x}(t+T) = \underline{x}^T(t)\ \underline{f}(t)$

Equation (2-13) is the forward-time adaptation and has been termed the LMS algorithm. The other equation is

$$\underline{f}(t-1) = \underline{f}(t) + \mu[x(t+T) - \hat{x}(t+T)]\ \underline{x}(t) \quad (2\text{-}14)$$

for reverse-time processing.

The deconvolved trace is $y(t) = x(t+T) - \hat{x}(t+T)$. Denoting the deconvolved reverse-time and forward-time traces by $y_r(t)$ and $y_f(t)$ respectively, the final deconvolved trace is

$$y(t) = \tfrac{1}{2}\ [y_f(t) + y_r(t)] \quad (2\text{-}15)$$

Griffiths found that the filter will converge to $\underline{F}_o$

under the condition $$\mu = \frac{\alpha}{L\sigma_x^2} \quad \text{and } 0 < \alpha < 2 \qquad (2\text{-}16)$$

where L is the filter length, σ_x^2 is the average power level determined from the input trace autocorrelation, and α is a parameter which gives the fractional increase in noise power output. Since the energy distribution varies greatly with time on seismic traces, instead of using a constant μ, Baker (1979) used σ_{t+1}^2 instead of σ_x^2, where

$$\sigma_{t+1}^2 = (1-b)\ x(t+1)^2 + b\ \sigma_t^2 \qquad b = \exp\left(-{}^{\Delta T}/_{T_a}\right) \qquad (2\text{-}17)$$

The time constant T_a has been verified experimentally by Griffiths to be

$$T_a = \frac{-\ \Delta T}{\ln\ (1-{}^{\alpha}/L)} \qquad (2\text{-}18)$$

where ΔT is the sampling interval. From these quantities μ is calculated continuously along the trace. For the synthetic seismogram plus random noise, as shown in Figure 2.1, the output of the adaptive deconvolution operation using gradient descent is shown in Figure 2.15. In this Figure

IOPT means $y = \frac{1}{2}[y_r(t) + y_f(t)]$

LS = number of datas LF=7, filter length

A=α, the fractional increase in noise power output

IPD=1, the prediction distance is 1.

The results are not impressive with regard to spiking or suppressing the noise.

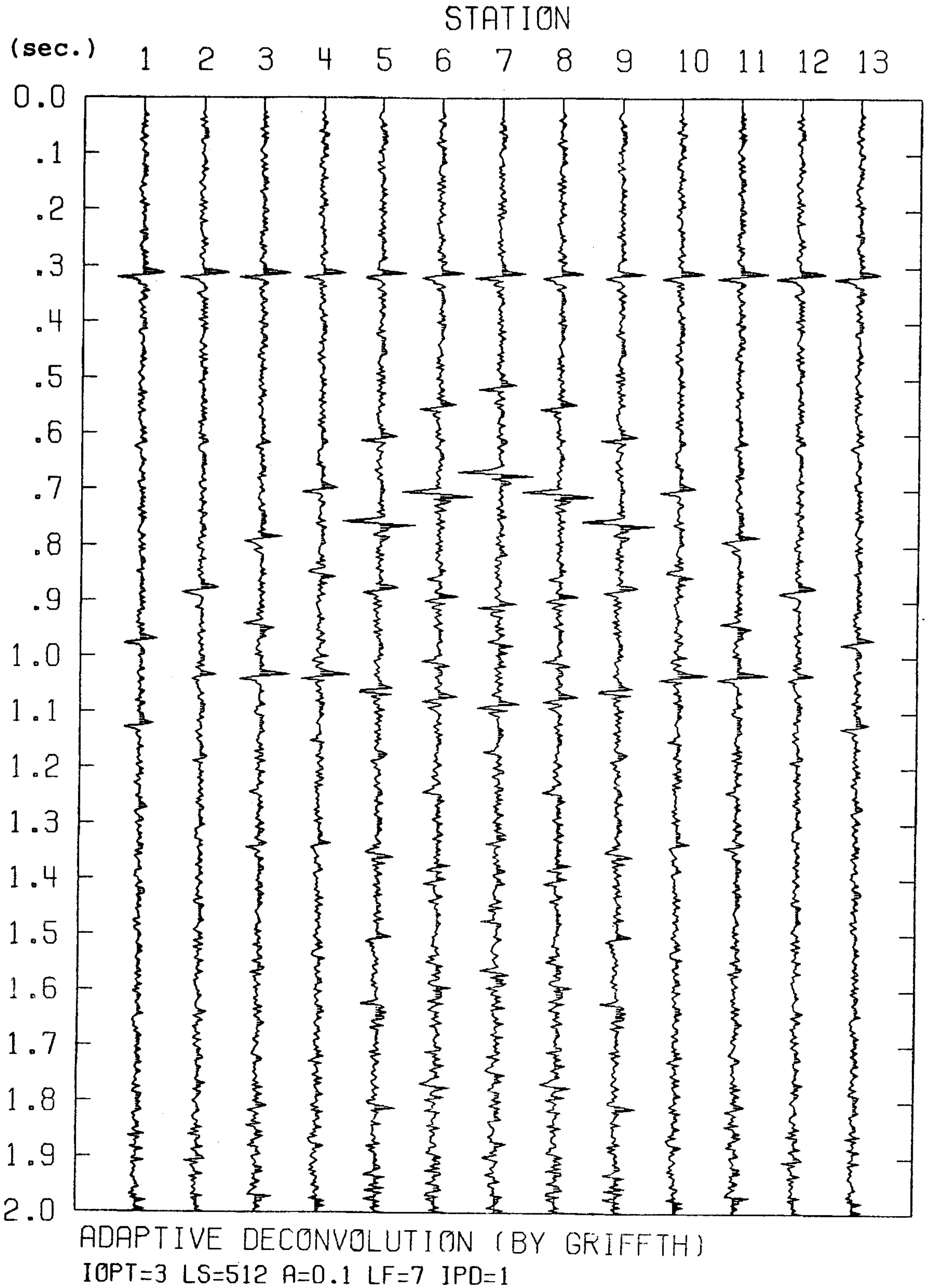


Figure 2.15 Gradient descent adaptive deconvolution output of Figure 2.1 S/N=4.44

2.4 Minimum entropy deconvolution

The minimum entropy deconvolution (MED) technique has been described by Wiggins (1978). MED employs a linear operator that maximizes the "spike-like" characteristics of a representative set of traces. Rather than seeking to whiten the data, the MED process seeks the smallest number of large spikes that is consistent with the data (i.e., it minimizes entropy or maximizes the order in the data), and it makes no assumptions regarding the phase characteristics of the wavelet. While maximizing the spikiness of the output traces, it selectively suppresses frequency bands over which the ratio of coherent signal-to-random noise is lowest and thereby emphasizes those bands in which coherent signals dominate.

The theory of MED processing is as follows. Suppose that the input signal samples are labeled:

$$x_{ij} \quad \begin{matrix} i = 1,2,\ldots N_s \\ j = 1,2,\ldots N_t \end{matrix}$$

where N_s is the number of trace segments or sample signals and N_t is the number of time samples per segment. The output that results from applying these inputs to the MED filter having impulse response samples f_k is

$$y_{ij} = \sum_{K=1}^{N_f} f_k \, x_{i,j-k} \qquad (2\text{-}19)$$

The varimax norm is defined as (Wiggins, 1978)

$$V = \Sigma V_i \qquad (2\text{-}20)$$

where $$V_i = \sum_j y_{ij}^4 \Big/ (\sum_j y_{ij}^2)^2 \tag{2-21}$$

A maximum of the varimax norm corresponds to a minimum number of individual spikes in the deconvolved signal. The maximum of V can be found from

$$\frac{\partial V}{\partial f_k} = 0 = \sum_i \frac{\partial V_i}{\partial f_k}$$

$$= \sum[4V_i U_i^{-1} \sum y_{ij} \frac{\partial y_{ij}}{\partial f_k} - 4U_i^{-2} \sum y_{ij}^3 \frac{\partial y_{ij}}{\partial f_k}] \tag{2-22}$$

where $U_i = \sum_j y_{ij}^2$

Since $\frac{\partial y_{ij}}{\partial f_k} = x_{i,j-k}$ we can write

$$\sum_i V_i U_i^{-1} \sum_j \sum_\ell f_\ell x_{i,j-\ell} x_{i,j-k} = \sum_i U_i^{-2} \sum_j y_{ij}^3 x_{i,j-k}$$

or $$\sum_\ell f_\ell \sum_i V_i U_i^{-1} \sum_j x_{i,j-\ell} x_{i,j-k}$$

$$= \sum_i U_i^{-2} \sum_j y_{ij}^3 x_{i,j-k} \tag{2-23}$$

This can be written in matrix notation as:

$$\underline{R}\, \underline{f} = \underline{g} \tag{2-24}$$

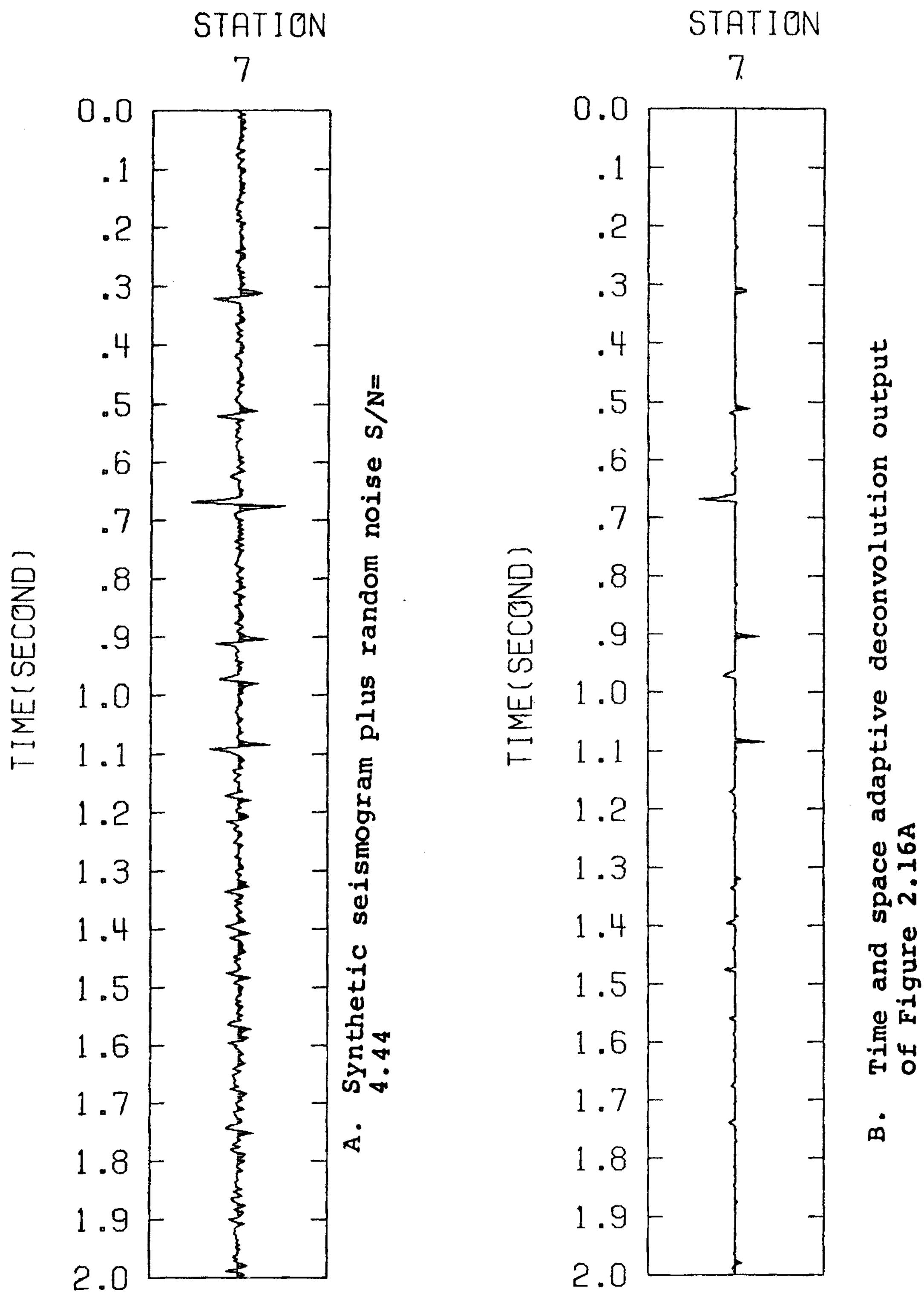
where $\underline{R}$ is an autocorrelation matrix that is a weighted summation of the autocorrelations of the input signals, and $\underline{g}$ is a cross-correlation vector that is a weighted summation of the cross-correlation of the cubed filter outputs with the inputs.

The equation is highly nonlinear so that it cannot be solved directly. However, it can be solved iteratively. The procedure consists of assuming a value of $\underline{f}$, computing $\underline{R}$ and $\underline{g}$, solving the Toeplitz normal equation (2-24) for $\underline{f}$, and recomputing $\underline{R}$ and $\underline{f}$.

Figure 2.16,A,D, shows the results of applying this procedure to the synthetic seismogram model. It can be seen from the figure that MED is a good process for spiking the signal. The disadvantage is in the polarity determination. That will adversely affect the analytic signal processing to be considered next.

2.5 Comparison and Summary

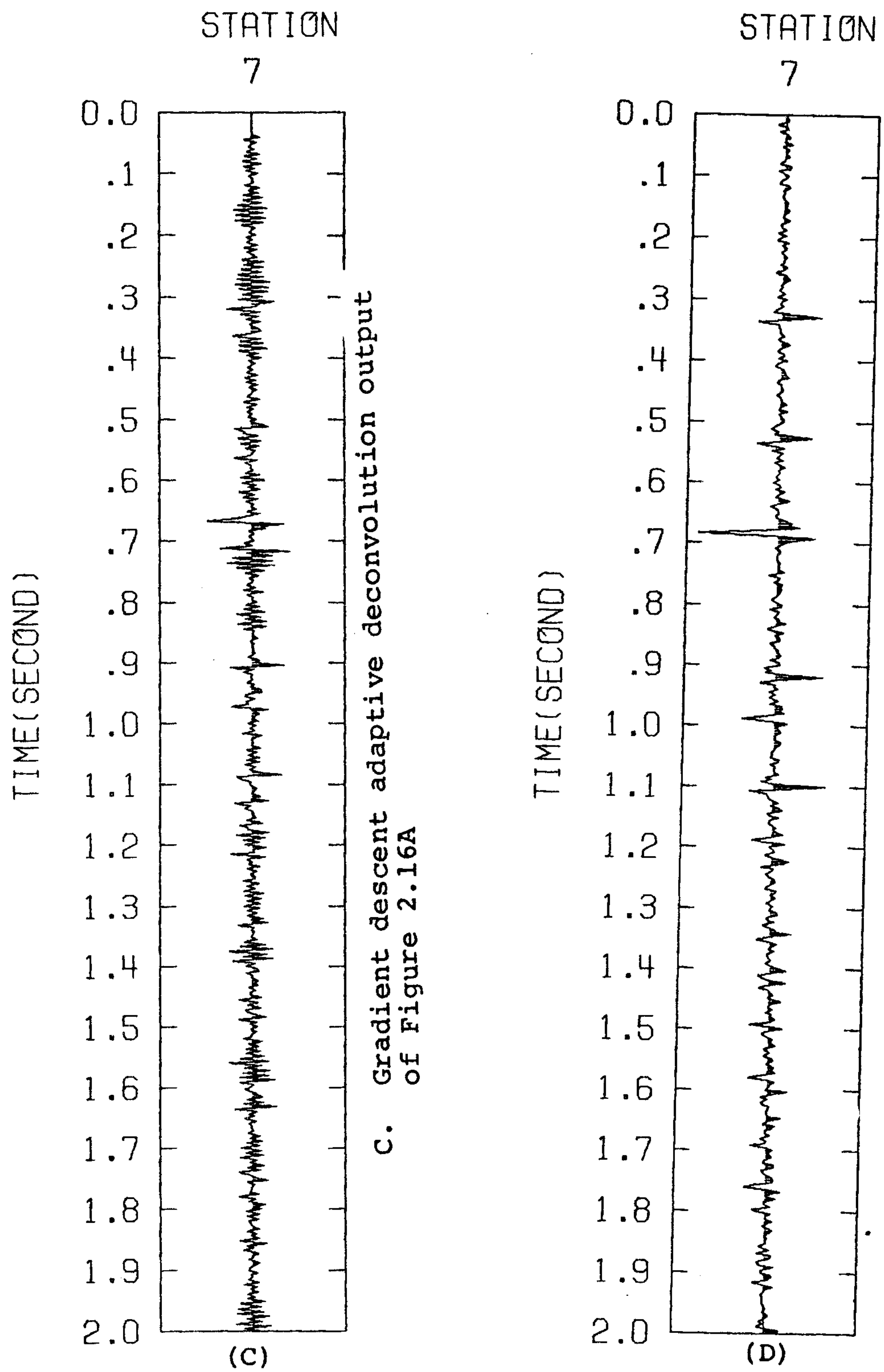
Consider Figure 2.1 the synthetic seismogram plus random noise, at station 7 for comparison of the deconvolution procedures. In Figure 2.16, A,B,C,D, there are three deconvolution outputs to compare in performance with regard to spiking the wavelet and suppressing the noise. It is evident from these figures that time and space adaptive deconvolution gives the best performance and it will be used in preprocessing the data for the analytic signal representation.



(A)
SYNTHETIC SEISMOGRAM S/N=4.44

(B)
TIME AND SPACE ADAPTIVE DECONVOLUTION

Figure 2.16 Comparison of deconvolution outputs



C. Gradient descent adaptive deconvolution output of Figure 2.16A

D. Minimum entropy deconvolution output of Figure 2.16A

GRADIENT DESCENT ADAPTIVE DECONVOLUTION MINIMUM ENTROPY DECONVOLUTION

Figure 2.16, continued

CHAPTER THREE

ANALYTIC SIGNAL

3.1 Introduction

Farnbach (1975) employed the analytic signal in teleseismic analysis. Taner and Sheriff *et al* (1977, 1979) used the analytic signal in seismic data analysis to help in the geological interpretation of bright spots in real seismic data. Sicking (1978) used the analytic signal in the synthetic seismogram of the wedge model and interfingering model analysis. The application of analytic signal processing to remove first and second order marine reverberation will be presented here. Some new computer results in the teleseismic pulse and signal analysis which are not in Farnbach (1975) will also be shown here to be of help in interpretation of seismograms.

3.2. Calculation of the analytic signal

The analytic signal representation $\psi(t)$ of a real time function $s(t)$ is a complex signal having the following properties:

(1) $s(t) = \text{Re}\ \psi(t)$ (3-1)

(2) $F\{\psi(t)\} = \Psi(f) = 2S(f) \quad f > 0$ (3-2)

$= 0 \quad f > 0$

$$\Psi(f) = 2S(f)\ U(f)$$

$$\psi(t) = F^{-1}\{2S(f)\ U(f)\}$$

$$= 2s(t) * \{\tfrac{1}{2}\delta(-t) - \frac{1}{j2\Pi t}\}$$

$$= s(t) + j\ s(t) * \frac{1}{\Pi t}$$

$$= s(t) + j\ \frac{1}{\Pi}\int_{-\infty}^{\infty} \frac{s(\lambda)}{t-\lambda}\ d\lambda$$

$$= s(t) + j\ \hat{s}(t) \qquad (3\text{-}3)$$

Thus the analytic signal consists of a real part s(t) equal to the original signal and an imaginary part $\hat{s}(t)$ given by the convolution of s(t) and $1/\Pi t$. This transformation of s(t) into $\hat{s}(t)$ is called the Hilbert transform and can be represented in the frequency domain as

$$F\{\hat{s}(t)\} = F\{\frac{1}{\Pi}\ s(t) * \frac{1}{t}\} = S(f) \cdot (-j\ \text{sgn}\ f) \qquad (3\text{-}4)$$

The generation of s(t) can be thought of as resulting from passing s(t) through a filter whose transfer function is given by

$$H(f) = -j\ \text{sgn}\ f \qquad \text{where} \quad \text{sgn}\ f = \begin{cases} 1 & f>0 \\ 0 & f=0 \\ -1 & f<0 \end{cases} \qquad (3\text{-}5)$$

$$\frac{1}{\Pi t} \Leftrightarrow -j\ \text{sgn}\ f$$

The Hilbert transform has the following properties: (1) The energy (or power) spectrum of a signal s(t) and its Hilbert transform $\hat{s}(t)$ are equal.

$$|\hat{S}(f)|^2 \triangleq |\mathcal{F}\{\hat{s}(t)\}|^2 = |-j \text{ sgn } f|^2 \; |S(f)|^2 = |S(f)|^2 \quad (3\text{-}6)$$

As a consequence, the total energy (or power) of a signal and its Hilbert transform are also equal.

(2) A signal and its Hilbert transform are orthogonal, that is,

$$\int_{-\infty}^{\infty} s(t)\;\hat{s}(t)\;dt = 0$$

the analytic signal can be put into a three-dimensional form, as shown in Figure 3.1.

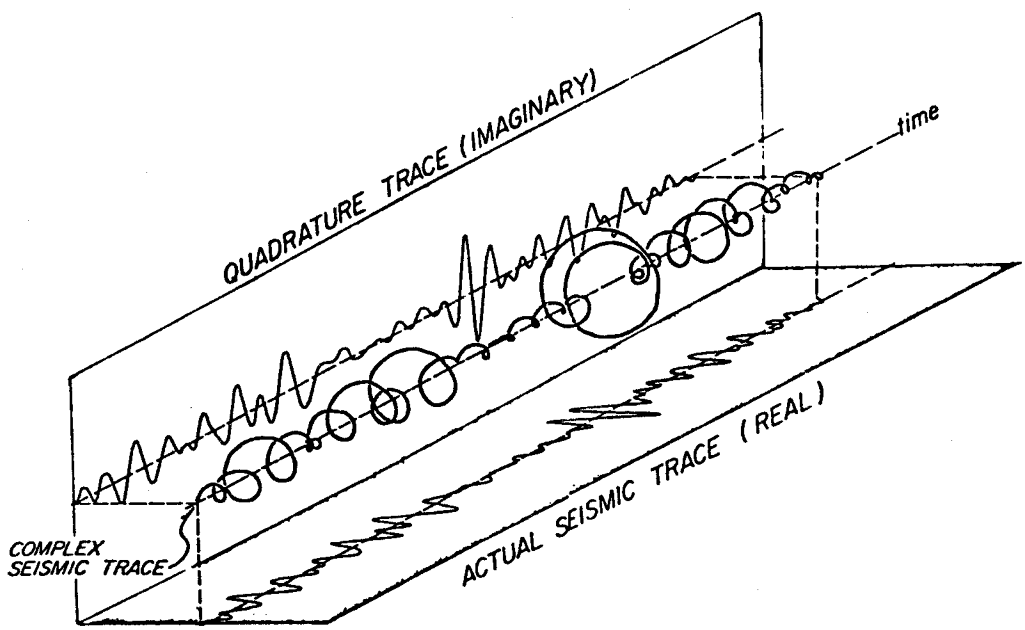


Figure 3.1 Isometric diagram of portion of an actual seismic trace (After Taner, Koehler and Sheriff, 1979)

(3) If c(t) and m(t) are signals with nonoverlapping spectra where m(t) is lowpass and c(t) is high pass, then $\widehat{m(t)c(t)} = m(t)\hat{c}(t)$.

The following parameters of a waveform can be obtained directly from its analytic signal representation.

Instantaneous amplitude (Envelope):

$$\text{Env}\{s(t)\} = A(t) = |\psi(t)| = \sqrt{s^2(t) + \hat{s}^2(t)} \tag{3-7}$$

Instantaneous phase:

$$\theta(t) = \angle\psi(t) = \tan^{-1} \frac{\hat{s}(t)}{s(t)} \tag{3-8}$$

or for $\psi(t) = |\psi(t)|\, e^{j\theta(t)}$

$$\ell n\psi(t) = \ell n\, |\psi(t)| + j\theta(t)$$

$$\therefore\ \theta(t) = \text{Im}\,[\ell n\psi(t)]$$

Instantaneous frequency $f_i(t)$:

$$f_i(t) = \frac{1}{2\Pi}\frac{d\theta(t)}{dt} = \frac{1}{2\Pi}\frac{d}{dt}\{\tan^{-1}\frac{\hat{s}(t)}{s(t)}\}$$

$$= \frac{1}{2\Pi}\, \frac{s(t)\,\frac{d\hat{s}(t)}{dt} - \hat{s}(t)\,\frac{ds(t)}{dt}}{s^2(t) + \hat{s}^2(t)} \tag{3-9}$$

$$\text{or } f_i(t) = \frac{1}{2\Pi}\frac{d\theta(t)}{dt} = \frac{1}{2\Pi}\frac{d}{dt}\,\text{Im}\{\ell n\psi(t)\} \tag{3-10}$$

$$= \frac{1}{2\Pi}\,\text{Im}\{\frac{d}{dt}\,\ell n\,\psi(t)\} = \frac{1}{2\Pi}\,\text{Im}\{\frac{\dot{\psi}(t)}{\psi(t)}\} \tag{3-11}$$

The <u>apparent polarity</u> is defined as the sign of s(t) when A(t) has a local maximum.

A computer program employing the following steps has been written to compute the analytic signal:

(1) Calculate the Fast Fourier Transform of s(t).

(2) Multiply this spectrum by 2 for indices $0 < K \leq \frac{N}{2}$, and by 0 for indices $\frac{N}{2} < K < N$.

(3) Compute the inverse Fast Fourier Transform, giving $\Psi(t)$.

3.3 Analytic signal representation of a teleseismic pulse

The pulse that will be studied is a form of the Berlage function often used to simulate teleseismic signals. This pulse is expressed mathematically as follows:

$$p(t) = \begin{cases} 0 & t<0 \\ t^2 \exp(2-2t) \sin(2\Pi t), & t \geq 0 \end{cases} \tag{3-12}$$

$$p(t) = \text{Re}\,\{t^2 \exp(2-2t) \exp[j\,(2\Pi t - \tfrac{\Pi}{2})]\}, \quad t \geq 0 \tag{3-13}$$

The angular frequency of sin $2\Pi t$ is 2Π. Now consider $t^2\, e^{-2t} U(t)$. The Fourier Transform of this function is

$$F\,\{t^2\, e^{-2t}\, U(t)\} = \frac{2}{(j\omega+2)^3}$$

The fraction of the total energy of $t^2\, e^{-2t}$ between 2Π to ∞ is given by

$$\int_{2\Pi}^{\infty} \frac{2}{(j\omega+2)^3} \frac{2}{(-j\omega+2)^3}\, dw \bigg/ \int_{0}^{\infty} \frac{2}{(j\omega+2)^3} \frac{2}{(-j\omega+2)}\, dw$$

$$= 4\int_{2\Pi}^{\infty} \frac{1}{(\omega^2+4)^3}\,dw \Bigg/ 4\int_{0}^{\infty} \frac{1}{(\omega^2+4)^3}\,dw$$

$$= \frac{\left[\dfrac{\omega}{4\cdot 4\cdot(\omega^2+4)^2} + \dfrac{3\omega}{8\cdot 2^4\cdot(\omega^2+4)} + \dfrac{3\tan^{-1}\frac{\omega}{2}}{8.2^5}\right]\Bigg|_{2\Pi}^{\infty}}{\dfrac{3}{8\cdot 2^5}\cdot\dfrac{\Pi}{2}}$$

$$= \frac{\dfrac{3}{8.2^5}\dfrac{\Pi}{2} - \left(\dfrac{2\Pi}{4.4(4\Pi^2+4)^2} + \dfrac{3.2\Pi}{8.2^4.(4\Pi^2+4)} + \dfrac{3}{8.2^5}\tan^{-1}\Pi\right)}{\dfrac{3}{8.2^5}\cdot\dfrac{\Pi}{2}}$$

$$= \frac{1.66 \times 10^{-5}}{0.0184077} = 9.01796 \times 10^{-4}$$

So outside the 0-2Π angular frequency interval, the energy of t^2e^{-2t} is very small and we can use the Hilbert transform property relating to the product of functions having nonoverlapping spectra, i.e.,

$$\widehat{t^2 \exp(2-2t)\ \sin 2\Pi t} \simeq t^2 \exp(2-2t)\ \widehat{\sin 2\Pi t}$$
$$= t^2 \exp(2-2t)\ \cos 2\Pi t \qquad (3\text{-}14)$$

$$\psi(t) = p(t) + j\,\hat{p}(t) \simeq t^2 \exp(2-2t) \exp[j(2\Pi t - \tfrac{\Pi}{2})]\ ,\ t \geq 0 \qquad (3\text{-}15)$$

$$\begin{cases} A(t) \simeq t^2 \exp(2-2t) & (3\text{-}16) \\ \theta(t) \simeq 2\Pi t - \dfrac{\Pi}{2} & (3\text{-}17) \end{cases}$$

The instantaneous frequency is

$$f_i(t) = \frac{1}{2\Pi} \frac{d\theta(t)}{dt} \simeq 1 \qquad (3\text{-}18)$$

or from

$$f_i(t) = \frac{1}{2\Pi} \frac{P(t)\frac{d\hat{P}(t)}{dt} - \hat{P}(t)\frac{dP(t)}{dt}}{P^2(t) + \hat{P}^2(t)} \simeq 1 \qquad (3\text{-}19)$$

Using these formulas and the computer program described previously, various properties of the signal in Equation (3-12) are computed and are shown in Figure 3.2-A,B,C,D,E,F,G, and H. These figures illustrate the pulse p(t), its quadrature component, envelope, instantaneous phase, and continuous phase that corresponds to the instantaneous phase curve and is obtained by the SUBROUTINE DRUM (Robinson, 1967). There are two ways to calculate the instantaneous frequency: One is to directly calculate the derivative of the continuous phase; the other is to calculate from the Equation (3-10). Apparent polarity is calculated from the definition given in Section 3-2.

The errors are most noticeable in the phase function (continuous), Figure 3.2.E, before t=0 and beyond t=5 sec. They occur at times when the pulse is very weak or nonexistent. This results from the fact that the phase function of a very small complex vector is ill-defined in a practical sense.

It is very useful for the analytic signal to be divided into its four elements when used in the interpretation of petroleum exploration data (bright spots).

If we define that the time of maximum of the envelope is the arrival time of a signal, then one application of the analytic signal is to enable us to find more precisely the relative arrival times of a common signal appearing on different traces in petroleum exploration.

3.4. Analytic signal representation of a simulated teleseismic signal.

A teleseismic signal can be simulated by the following representation:

$$s(t)=p(t)+\sum_n A_n p(t-t_n) \tag{3-20}$$

where the pulse $p(t)$ represents the main arrival at the recording station (the P phase) and the summation represents delayed pulses corresponding to minor waves arriving subsequently. These minor waves may be reverberations from inhomogeneities beneath the station or pulses from multiple paths in the layered transmission medium. Positive or negative values of A_n correspond to positive or negative reflection coefficients of the main wave at the discontinuities between layers. $s(t)$ has the Hilbert transform

$$\hat{s}(t) = \hat{p}(t) + \sum_n A_n \hat{p}(t-t_n) \tag{3-21}$$

and the analytic signal representation of $s(t)$ is

$$\psi(t) = s(t) + j\hat{s}(t) = p(t) + j\hat{p}(t) + \sum_n A_n [p(t-t_n) + j\hat{p}(t-t_n)] \tag{3-22}$$

This follows from the linearity of the Hilbert transform.

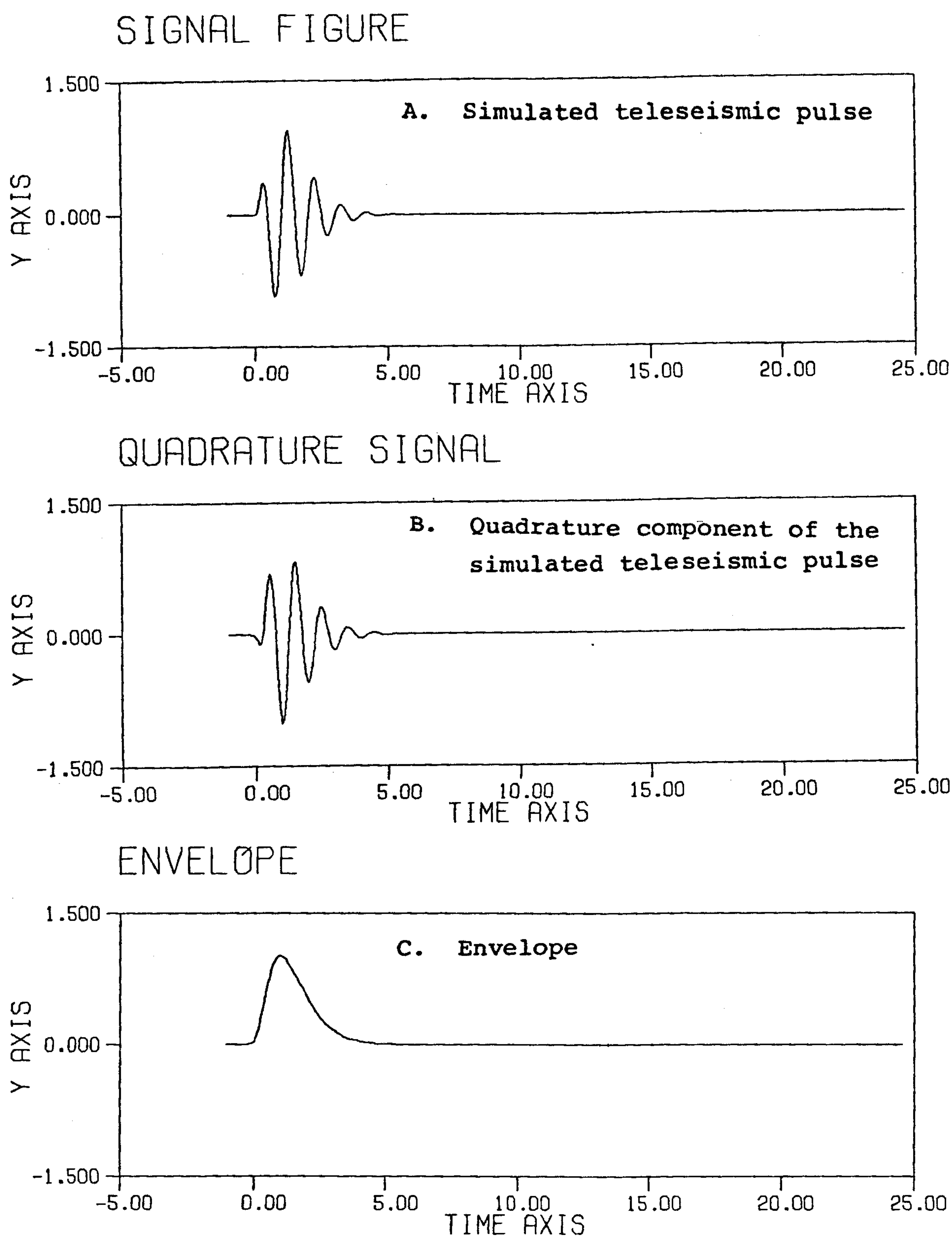
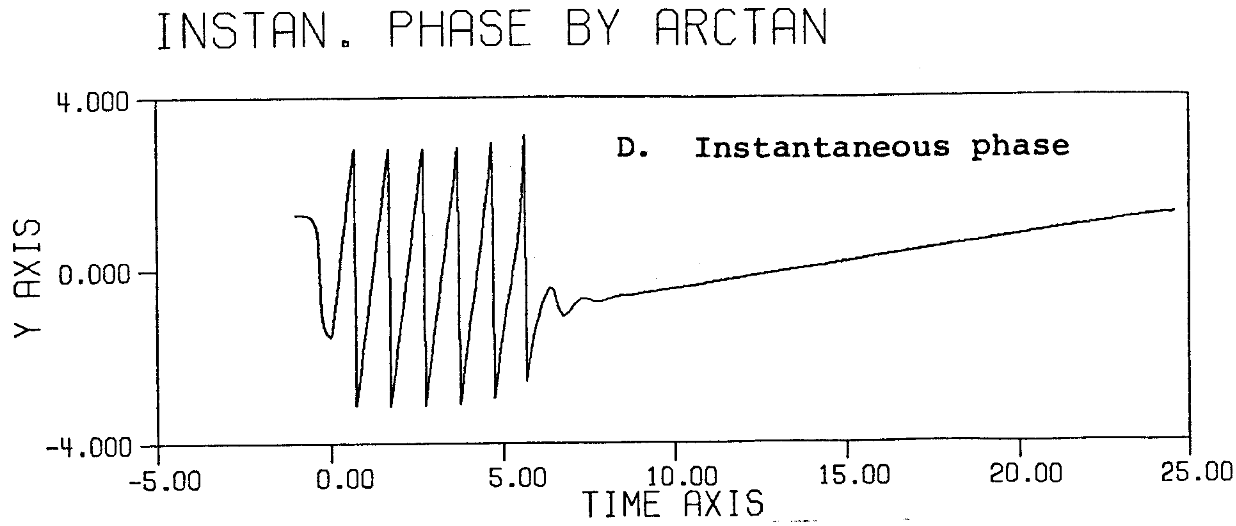
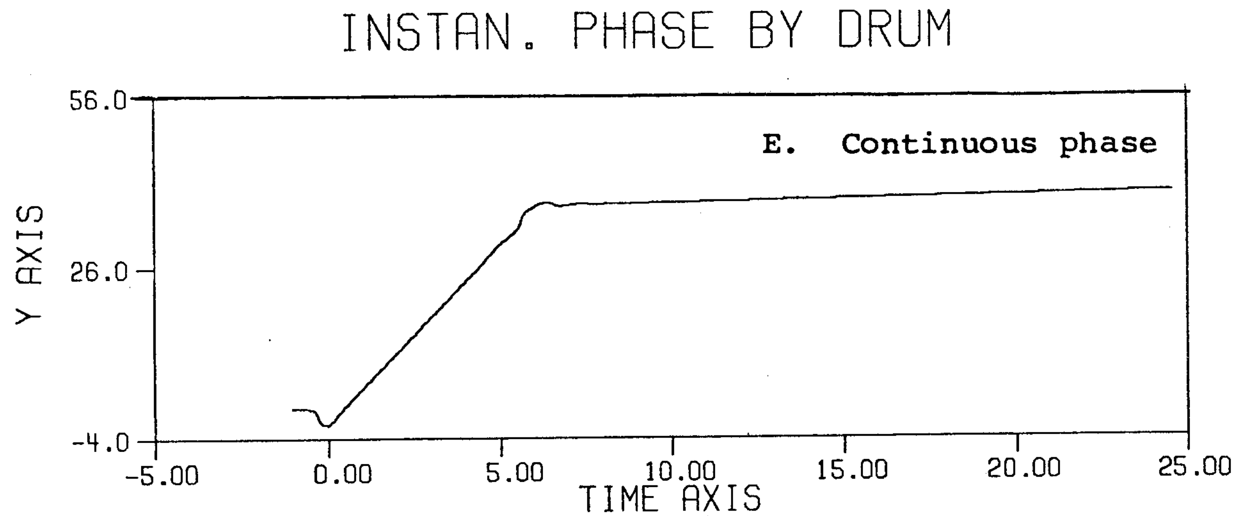


Figure 3.2 Analytic signal representation of a simulated teleseismic pulse





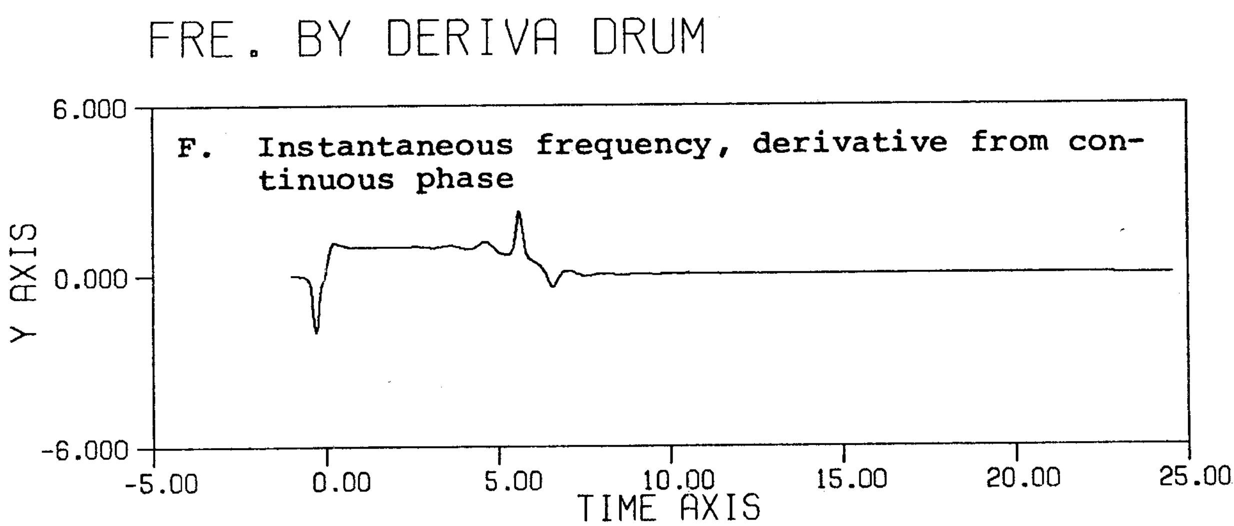
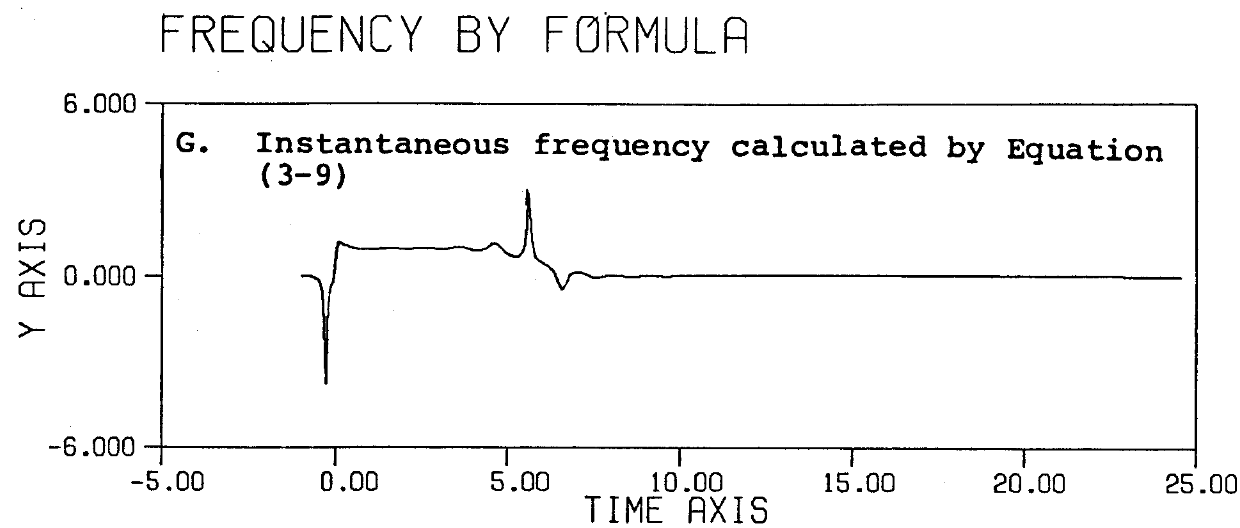


Figure 3.2, continued



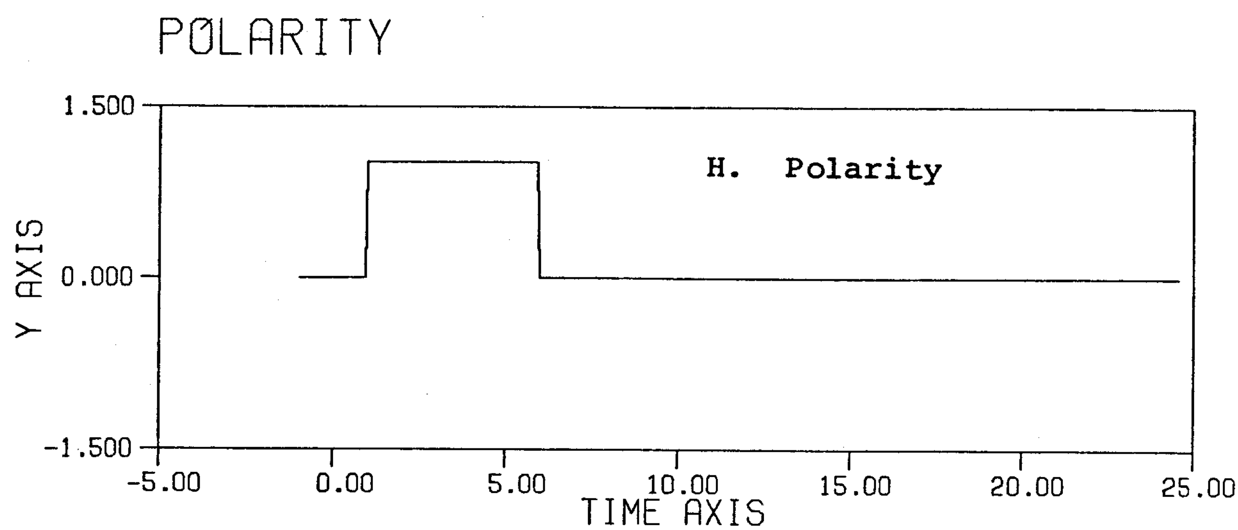


Figure 3.2, continued

A simulated signal consisting of the superposition of eight pulses of the shape shown by Equation (3-12) is analyzed using the analytic signal representation. The scale factor A_n and time delays t_n employed in the simulation are as follows:

n	1	2	3	4	5	6	7
Delay t_n (sec.)	1.6	2.3	3.9	5.5	8.5	9.8	11.3
Scale factor A_n	0.29	-0.24	0.36	-0.37	0.20	0.25	0.15

The resulting waveform is shown in Figure 3.3A.

Repeating the computer work as in the previous section, we obtain the data shown in Figure 3.3, A,B,C,D,E,F,G,H, and I. In Figure 3.3.F is shown the residual of the first order (straight line) least square fitting of the continuous phase, Figure 3.3.F over time interval 0.4 sec.-15.9 sec.

As an aid in interpreting the results, it is useful to consider the case of an amplitude modulated signal in the presence of noise. When the $(SNR)_T >> 1$, the envelope is linearly related to the signal. However, when $(SNR)_T << 1$, the principal component of the envelope is the Rayleigh distributed noise envelope, and no component is proportional to the signal. This severe loss of signal at low-input SNR is known as the threshold effect. These results can be used to aid in the interpretation of Figure 3.3. In Figure 3.3.C, the envelope shows no clear indication of pulse 1, and pulse 7 is greatly distorted by interference with pulse 6. This is because SNR is not very high.

To aid in interpretation of the instantaneous phase, we remove the linear trend and retain only residual phase of a least-square fitting of a straight line over the interval. In Figure 3.3.F, at pulse 2, 5, 7, phase reversals are seen to occur as the various component vectors increase or decrease in magnitude. At pulse 4, it is seen that there is a sharp discontinuity.

In Figure 3.3, G and H, there are jumps in the instantaneous frequency that indicate the arrival of pulses.

In Figure 3.3.I, polarity helps in the interpretation of phase reversal of the pulse.

3.5 Use of analytic signal processing to remove the first and second order reverberations.

It is possible to remove reverberations that are present in seismic traces by appropriate processing using the analytic signal. To see this, consider the signal s(t) consisting of the basic wavelet p(t) plus an infinite sequence of first order reverberations with reflection coefficient r ($|r|<1$),

$$s(t) = p(t) - rp(t-T) + r^2 p(t-2T) - r^3 p(t-3T) + \cdots \quad (3\text{-}23)$$

where T is the two-way travel time through the water layer. The corresponding analytic signal is

$$\begin{aligned}\psi(t) &= s(t) + j\,\hat{s}(t)\\ &= p(t) + j\,\hat{p}(t) - r\,[p(t-T) + j\,\hat{p}(t-T)] + r^2\,[p(t-2T)\\ &\quad + j\,\hat{p}(t-2T)] - r^3\,[p(t-3T) + j\,p\,(t-3T)] + \ldots \quad (3\text{-}24)\end{aligned}$$

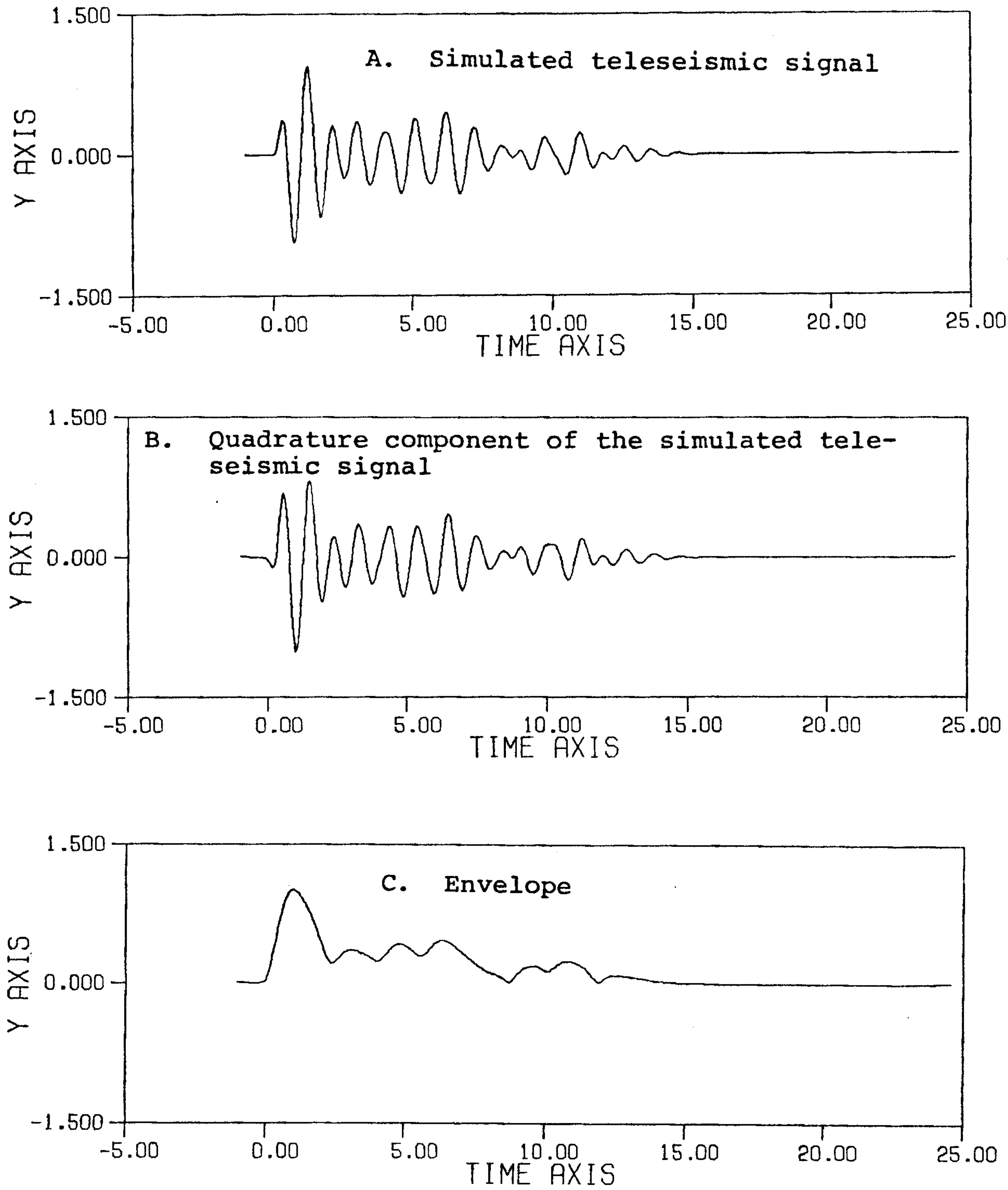


Figure 3.3 Analytic signal representation of a simulated teleseismic signal

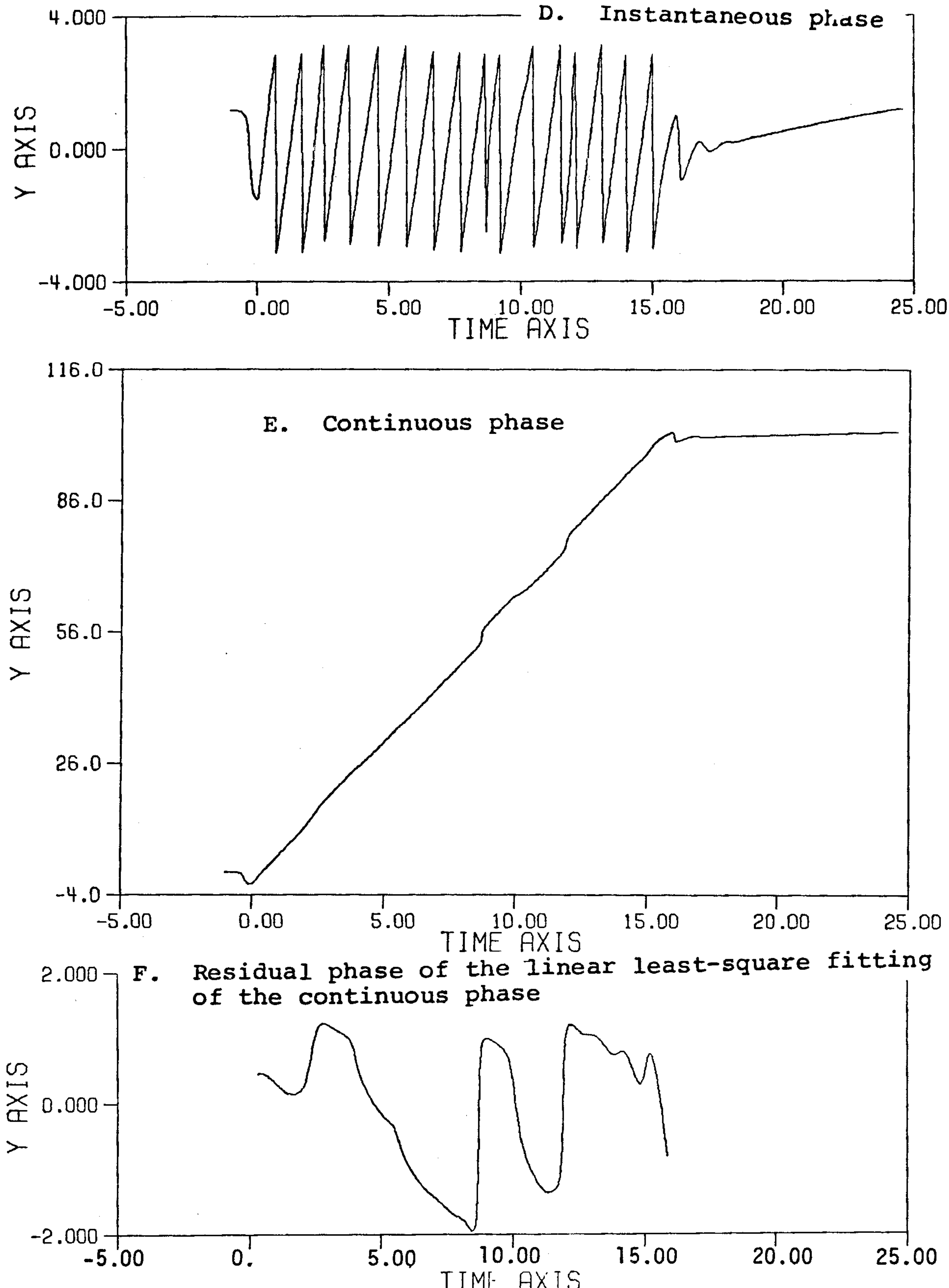


Figure 3.3, continued

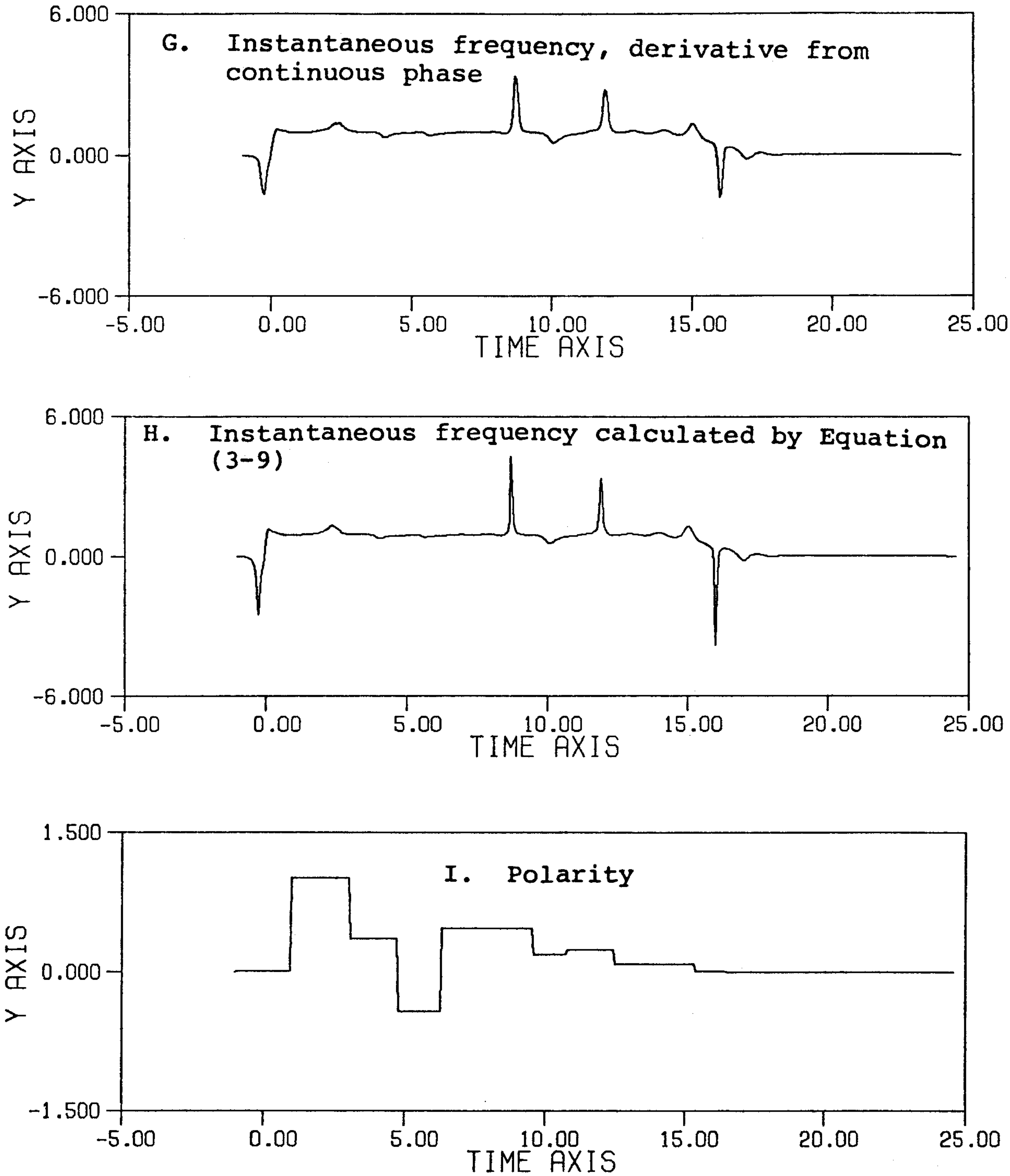


Figure 3.3, continued

Transforming to the frequency domain

$$\Psi(f) = 2P(f) - 2rP(f)e^{-j\omega T} + 2r^2P(f)e^{-j2\omega T}$$

$$- 2r^3P(f)e^{-j3\omega T} + ----, \quad \text{for } f > 0$$

$$= 2P(f)\,[1 - re^{-j\omega T} + r^2e^{-j2\omega T} - r^3e^{-j3\omega T} +---]\quad f > 0$$

$$= 2P(f)\,\frac{1}{1+re^{-j\omega T}} \qquad f > 0 \tag{3-25}$$

If $\Psi(f)$ is multiplied by the factor $\left(\frac{2}{1+re^{-j\omega T}}\right)^{-1}$, we obtain P(f), $f > 0$, and eliminate the reverberations.

To recover p(t), consider that for a real function the spectrum for negative frequencies is the complex conjugate of the spectrum for positive frequencies, i.e., the amplitude is an even function of frequency and the phase is an odd function of frequency. Therefore, we can determine P(f) for $f < 0$ and take the Inverse Fourier Transform to obtain p(t).

This procedure can be extended to include processing of second order reverberations also. The second order reverberation response corresponds to the impulse response of the water layer convolved with itself. The second order reverberation signal can be represented as

$$s(t) = p(t) - 2r\,p(t-T) + 3r^2\,p(t-2T) - 4r^3\,p(t-3T) + 5r^4\,p(t-rT)\ \dots. \tag{3-26}$$

The corresponding analytic signal is

$$\psi(t) = s(t) + j\,\hat{s}(t) = p(t) + j\,\hat{p}(t) - 2r\,[p(t-T) + j\,\hat{p}(t-T)]$$

$$+ 3r^2 [p(t-2T) + j \hat{p}(t-2T)] - 4r^3 [p(t-3T) + j \hat{p}(t-3T)] + 5r^4 [p(t-4T) + j \hat{p}(t-4T)] + \ldots. \quad (3\text{-}27)$$

In the frequency domain, this becomes

$$\Psi(f) = 2P(f) - 2r \cdot 2P(f)\, e^{-j\omega T} + 3r^2\, 2P(f)\, e^{-j\omega 2T} - 4r^3\, 2P(f)\, e^{-j\omega 3T} + 5r^4 \cdot 2P(f)\, e^{-j\omega 4T} \ldots, f > 0$$

$$= 2P(f)\, [1-2re^{-j\omega T} + 3r^2 e^{-j\omega 2T} - 4r^3 e^{-j\omega 3T} + 5r^4 e^{-j\omega 4T} ----]$$

$$= 2P(f)\, [1-re^{-j\omega T} + r^2 e^{-j2\omega T} ----] \cdot [1-re^{-j\omega T} + r^2 e^{-j2\omega T} ----]$$

$$= 2P(f) \frac{1}{(1+r\, e^{-j\omega T})^2}, \quad f > 0 \quad (3\text{-}28)$$

So if $\Psi(f)$ is multiplied by the factor $\frac{1}{2}(1+r\, e^{-j\omega T})^2$ we will obtain $P(f)$, for $f > 0$. Repeating the recovery process as in the previous case, we can obtain $p(t)$ and remove the second order reverberation. The factor $\frac{1}{2}(1+r\, e^{-j\omega T})^2$ in terms of the Z-transform representation is $\frac{1}{2}(1+r\, Z^T)^2 = \frac{1}{2}(1+2r\, Z^T + r^2\, Z^{2T})$.

In conventional second order dereverberation processing, the filter $1 + 2r\, Z^T + r^2\, Z^{2T}$ is normally called the three-point Backus filter (Backus, M. M., 1959). Theoretically after dereverberation processing, the quality of the analytic signal should be improved.

From the above, it is seen that the analytic signal provides an easy way to reduce the computation time and save computer costs. It seems probable that there are other processing operations that can be simplified using analytic signal processing techniques similar to those presented here.

CHAPTER FOUR

DECONVOLUTION AND ANALYTIC SIGNAL ANALYSIS OF A SYNTHETIC SEISMOGRAM

4.1 Introduction

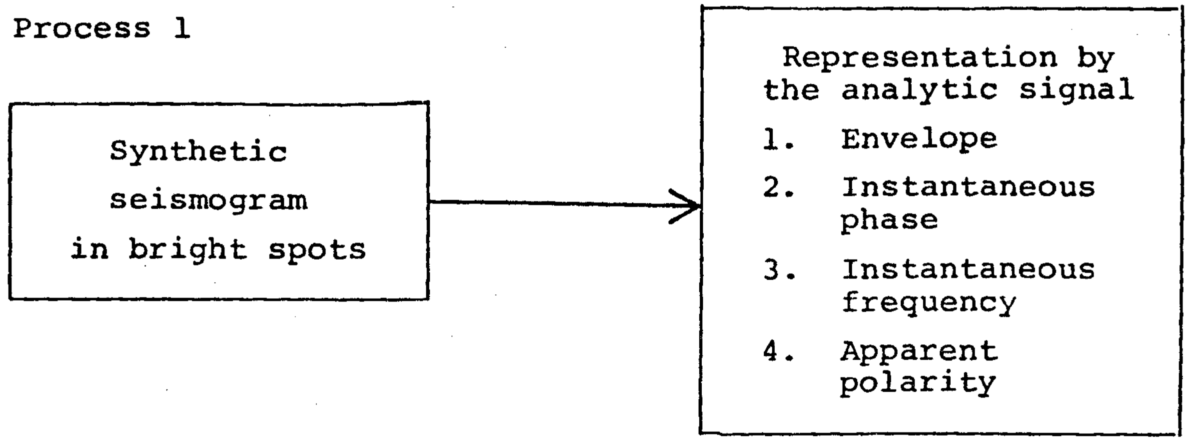
In this chapter we will investigate the use of the three processes discussed in Chapter One to determine their effects on the analytic signal of the synthetic seismogram of bright spots.

For Process 1 in the following section, 4.2, the synthetic seismogram of the bright spots model is the same as Figure 1.2. For Process 2 in Section 4.3, the synthetic seismogram plus random noise is the same as Figure 2.1. Random noise is uncorrelated from trace 1 to trace 7. Because of the known symmetry in the bright spot model, trace 8 to trace 13 are generated from the earlier traces to save computer processing time.

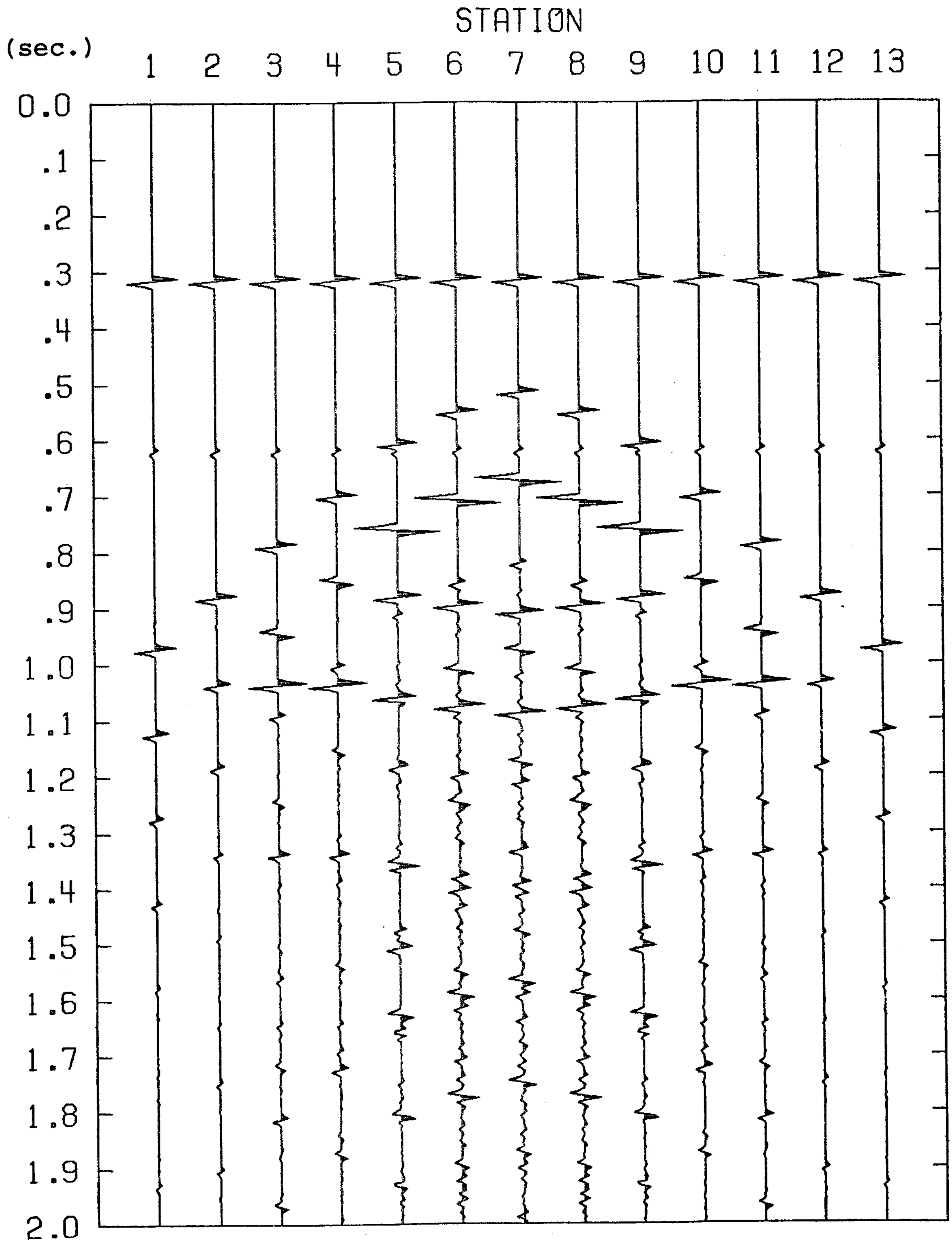
4.2 Effects of processing on the analytic signal representation of a synthetic seismogram.

This process is Process 1 of Section 1.1 and is as follows:

Process 1

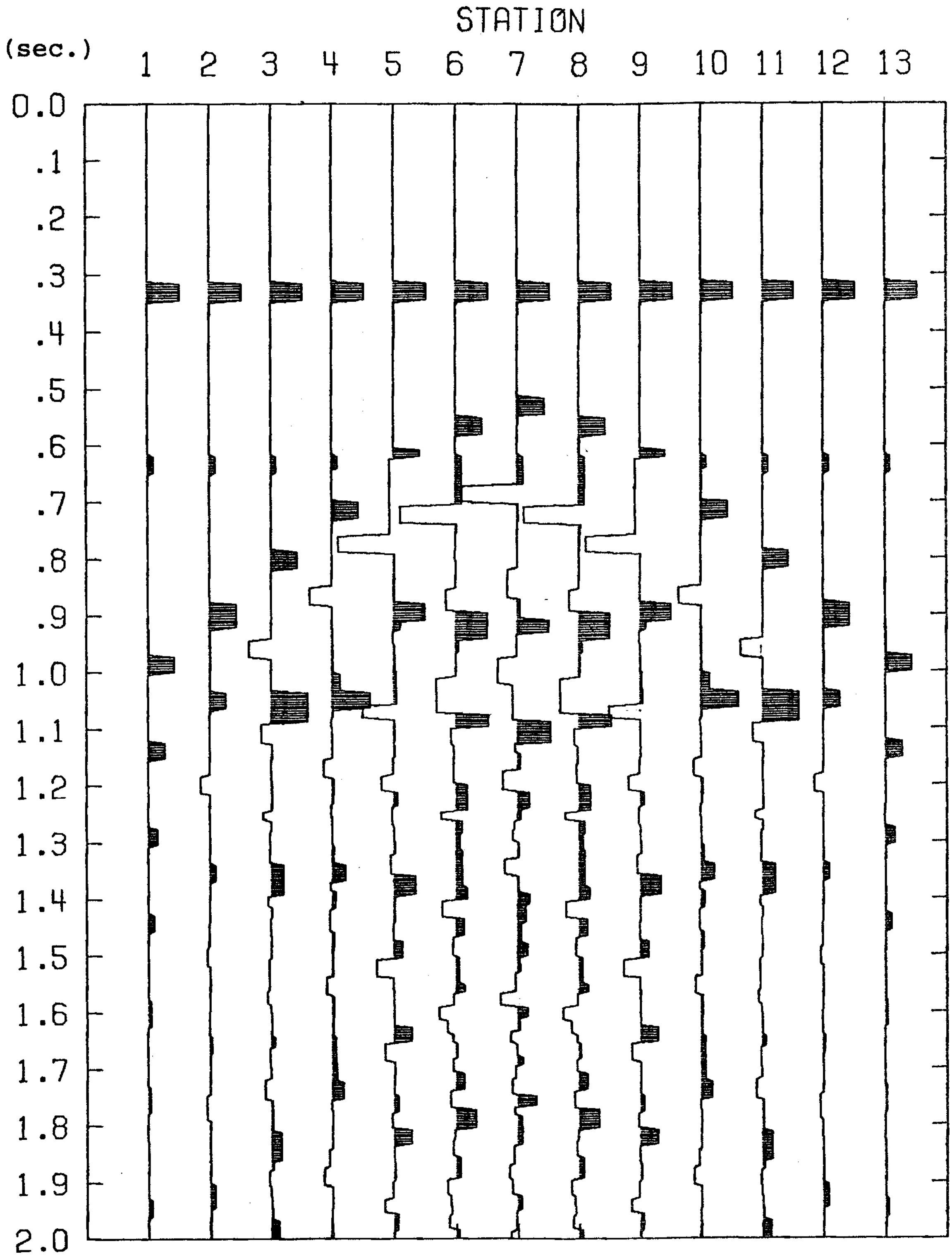


From calculation of the analytic signal representation, we obtain the envelope, instantaneous phase (phase between $-\Pi$ and Π, continuous phase, and residual phase of the linear least-square fitting), instantaneous frequency and apparent polarity of the synthetic seismogram. These quantities are shown in Figure 4.1, A, B, C, D, E, F, and G. It is noticeable in trace 5 of the apparent polarity plot, Figure 4.1.C, that the wavelet is disturbed by the multiple in such a manner that it affects the time of the maximum of the envelope and the polarity reversal. In the continuous phase plot of Figure 4.1.E, we notice that when the signal arrives, the phase will jump. It is interesting that in the residual phase plot, Figure 4.1.F reveals the parabolic curve in the gas and oil zone because of low velocity. In the frequency plot, Figure 4.1.G, it is difficult to classify the signal and the multiple, their frequencies being almost the same.



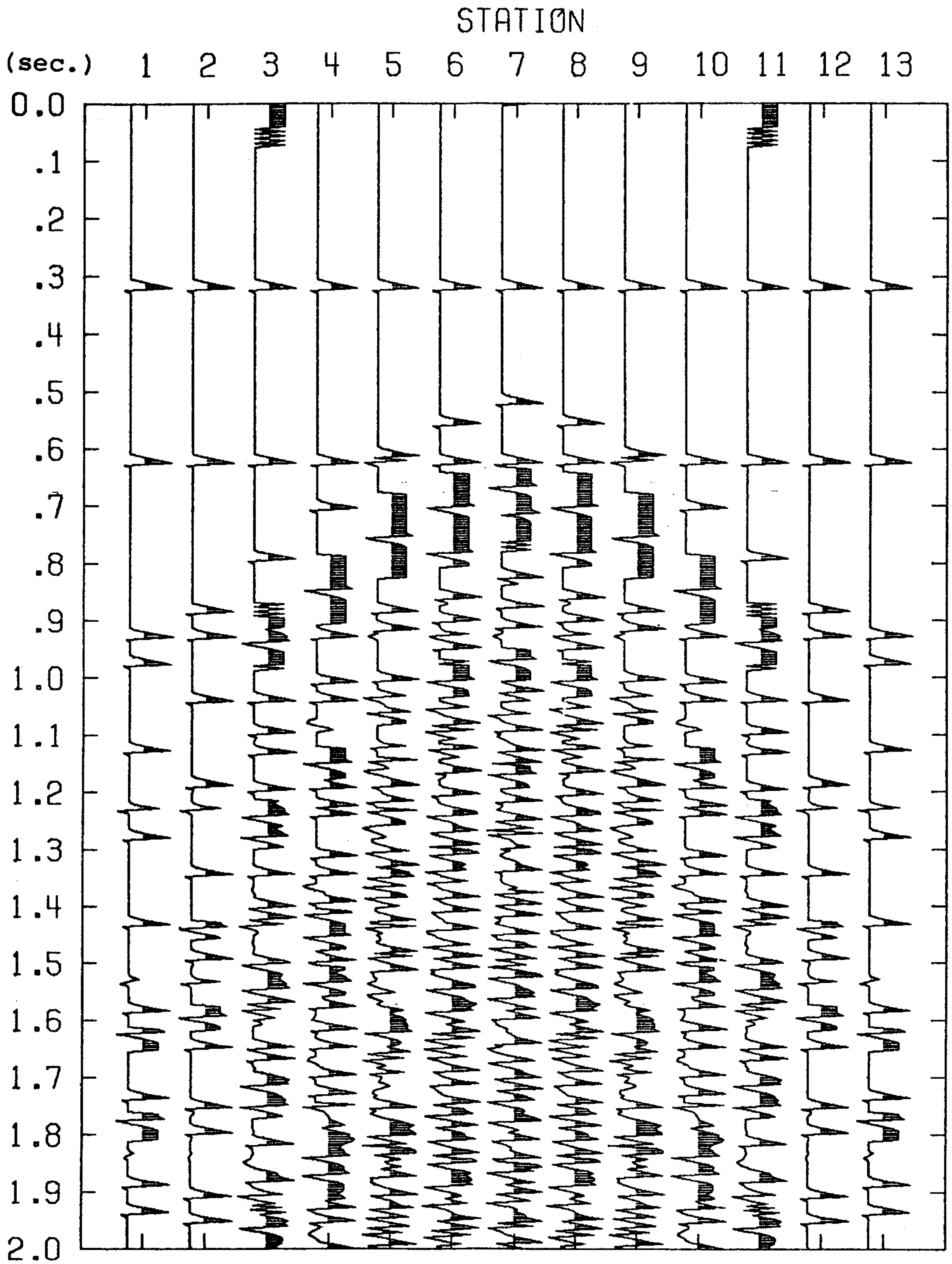
A. Synthetic seismogram of bright spots

Figure 4.1 Analytic signal representation of synthetic seismogram



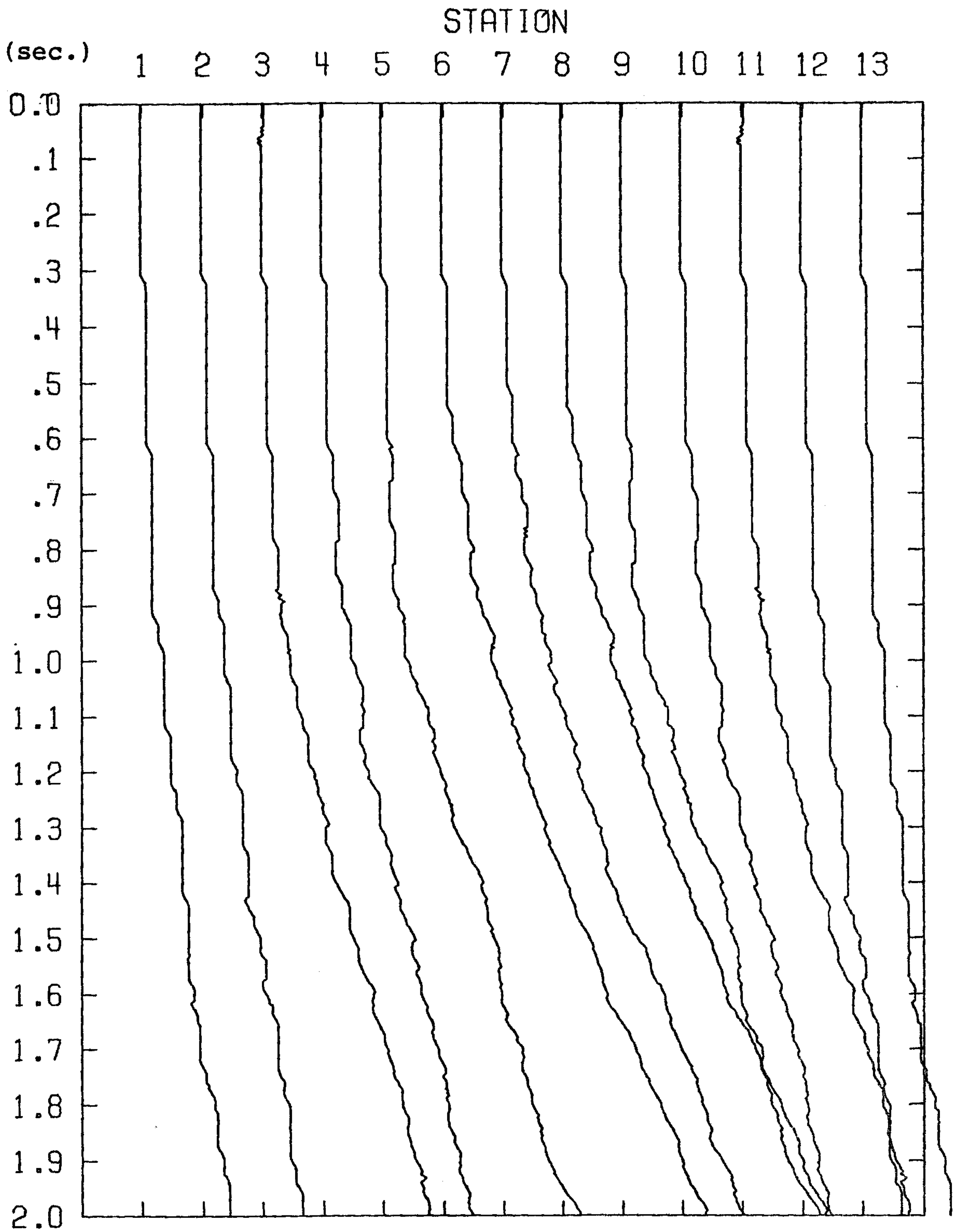
C. Apparent polarity

Figure 4.1, continued



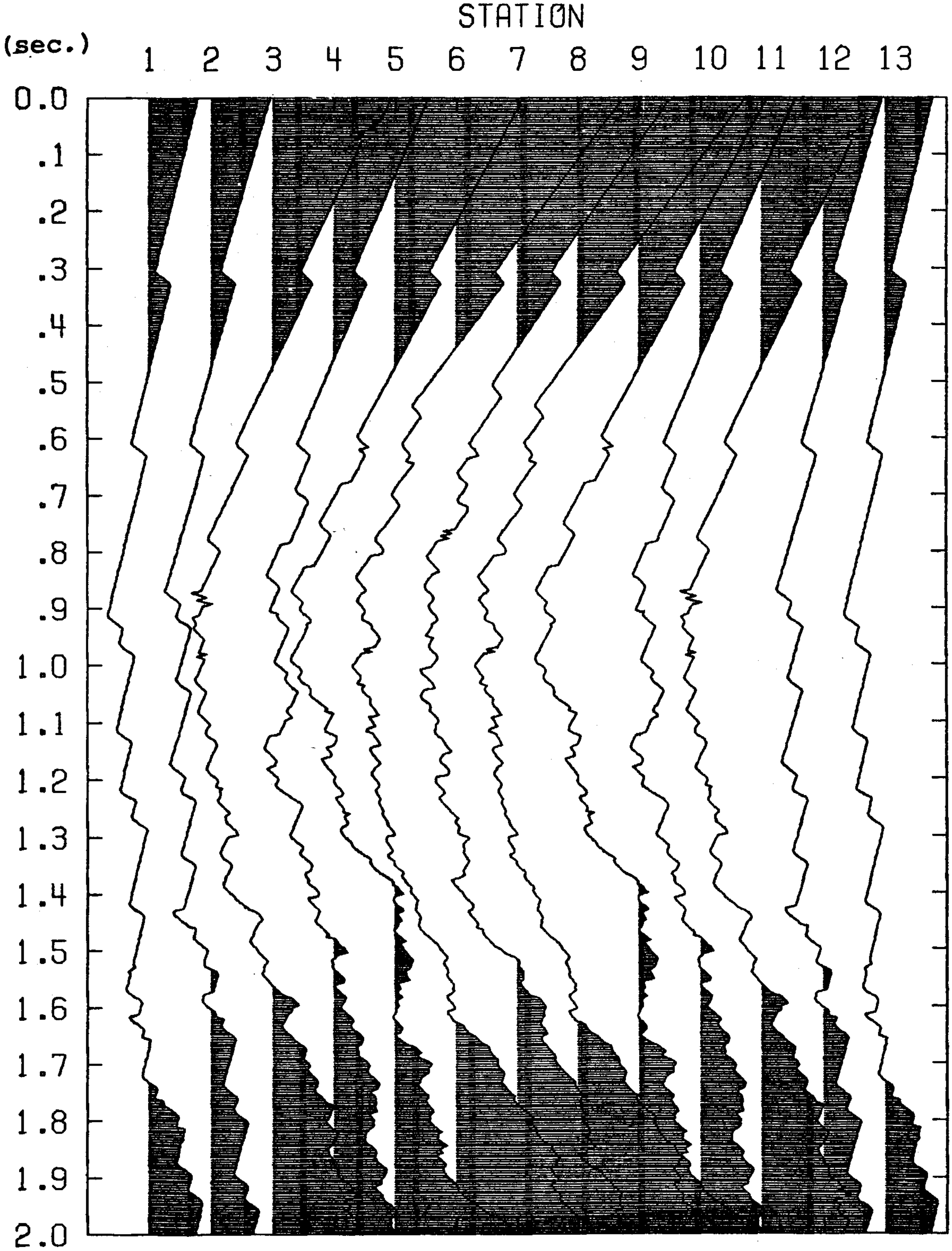
D. Instantaneous phase

Figure 4.1, continued



E. Continuous phase

Figure 4.1, continued



F. Residual phase of linear least-square fitting

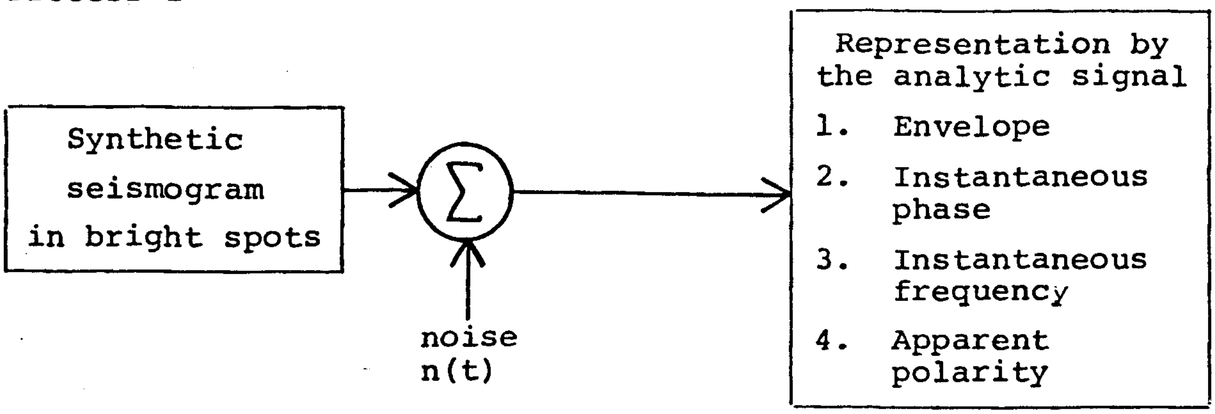
Figure 4.1, continued

This page is blank

4.3 <u>Processing of analytic signal representation of a synthetic seismogram plus noise</u>

This process is Process 2 of Section 1.1 and is as follows:

Process 2



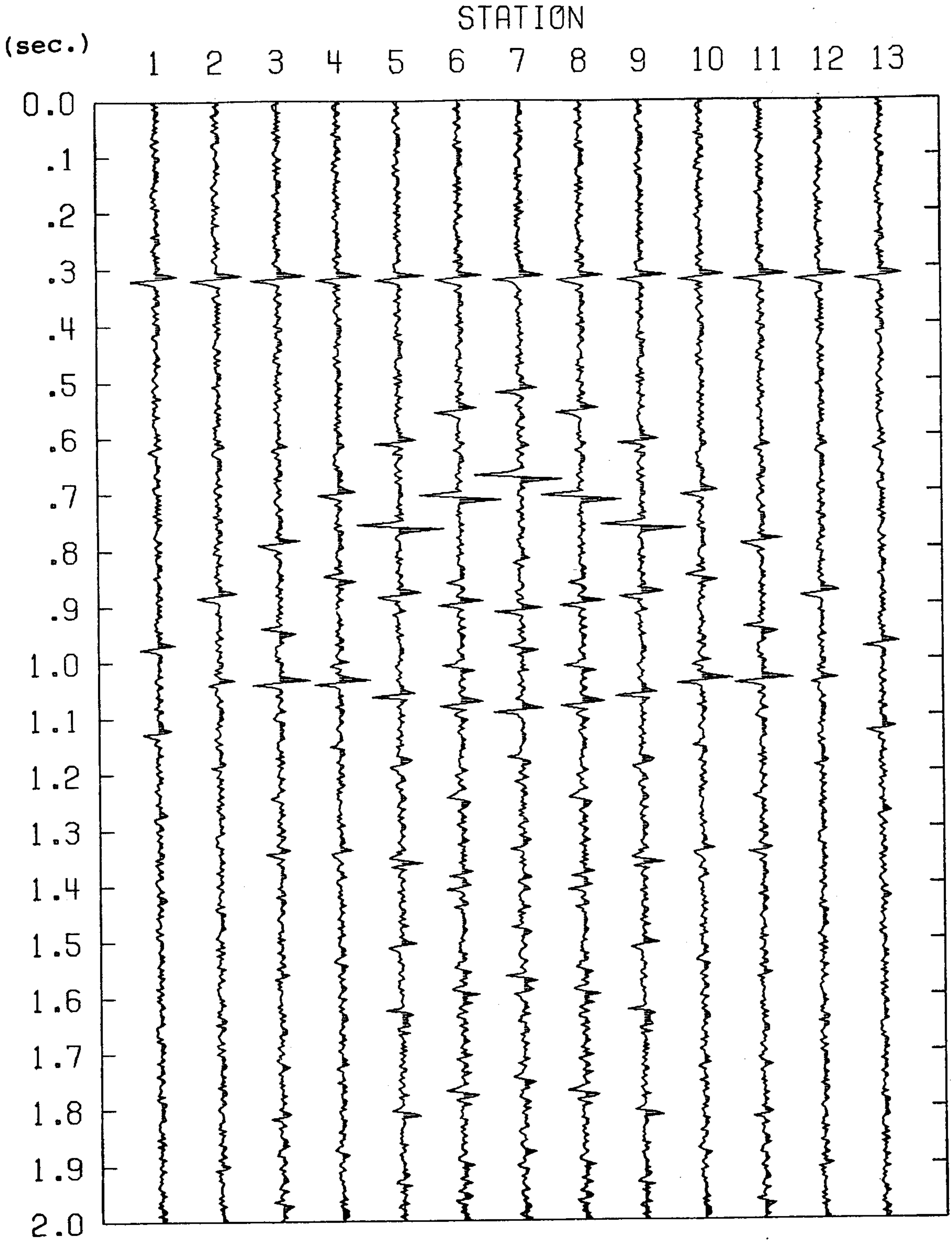
From the computed output shown in Figure 4.2, A, B, C, D, E, F, and G, the following can be seen:

(1) In the envelope plot, Figure 4.2.B, there is a large envelope for the multiple. If S/N is low, we will recognize the multiple envelope as a structure.

(2) In the apparent polarity plot, Figure 4.2.C. because of noise effects, the polarity appears random.

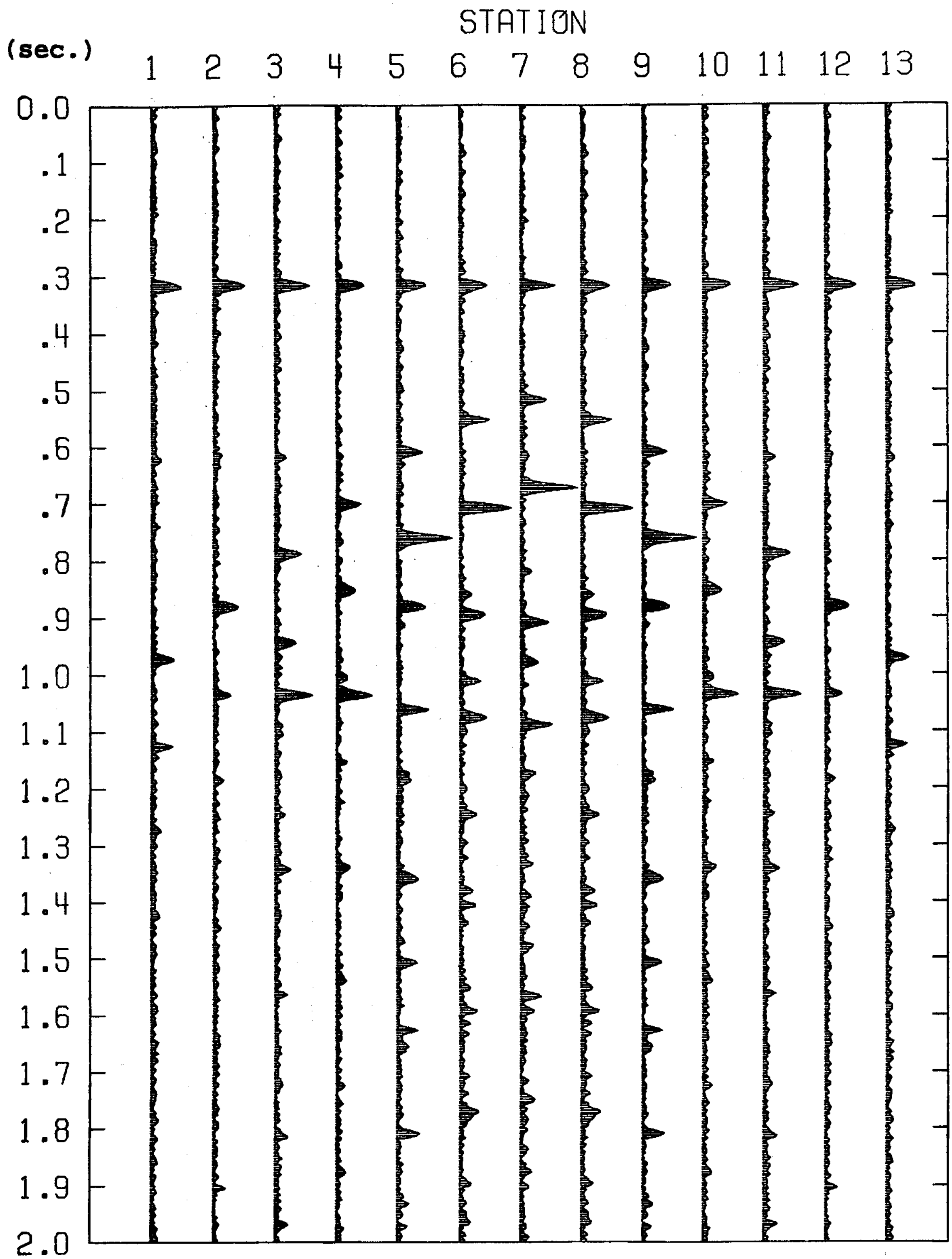
(3) In the continuous phase plot, Figure 4.2.E, because at every time there are noise components present, the continuous phase looks like a straight line. In the residual phase of the least-square fitting, Figure 4.2.F, there is no parabolic curve in the gas and oil zone.

(4) In the frequency plot, Figure 4.2.G, it is difficult to correlate the signal with the structure.



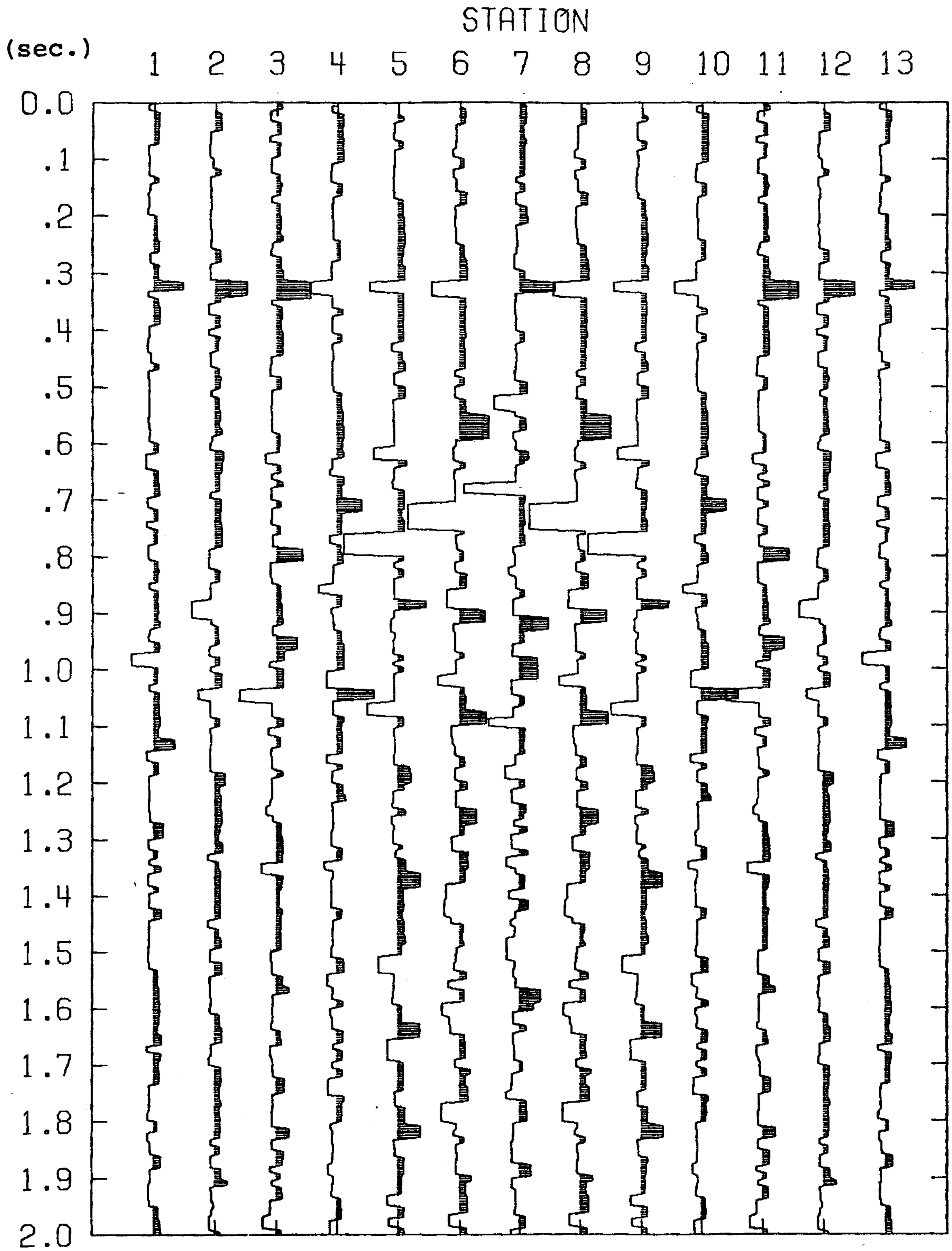
A. Synthetic seismogram plus random noise S/N= 4.44

Figure 4.2 Analytic signal representation of a synthetic seismogram plus noise



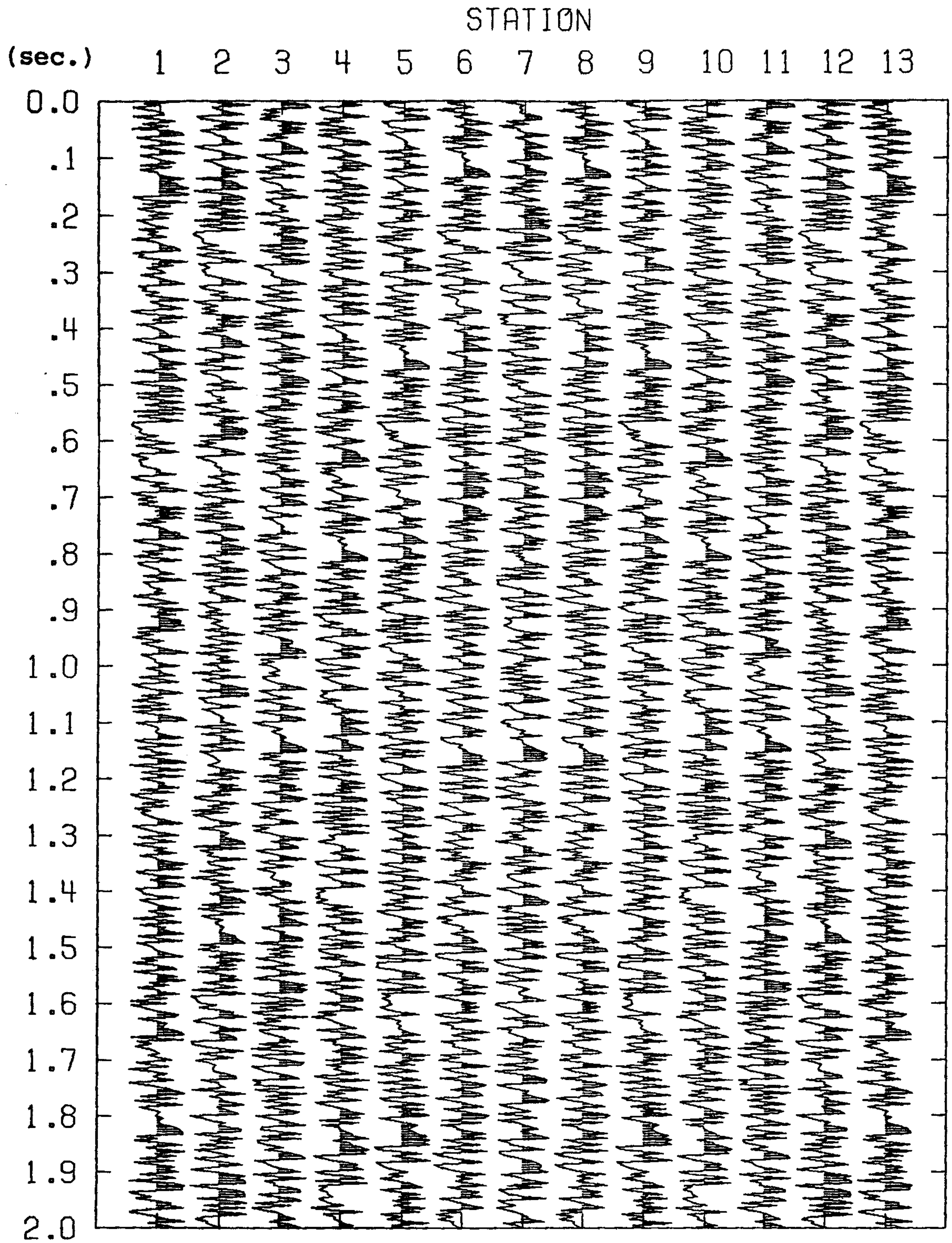
B. Envelope

Figure 4.2, continued



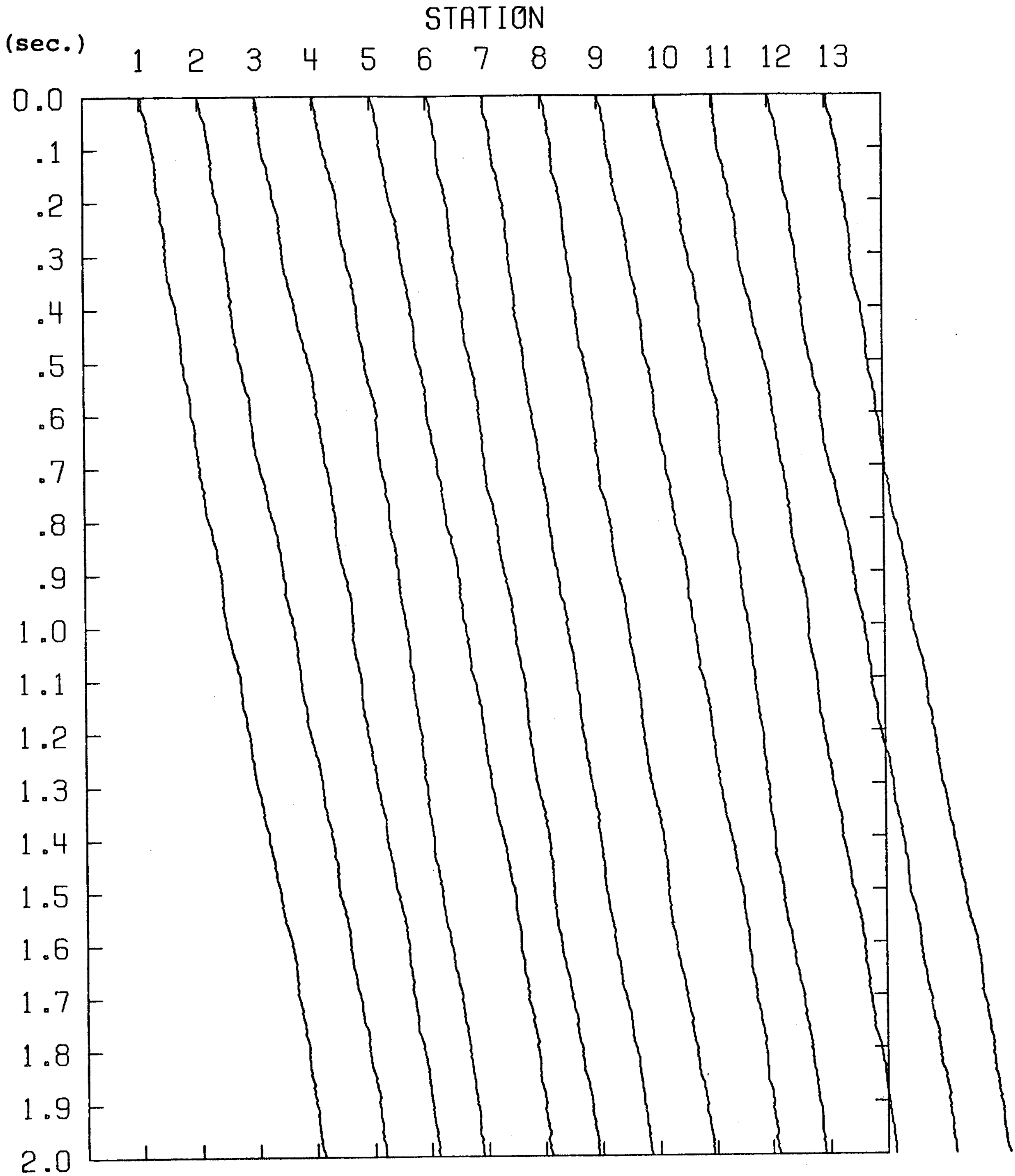
C. Apparent polarity

Figure 4.2, continued



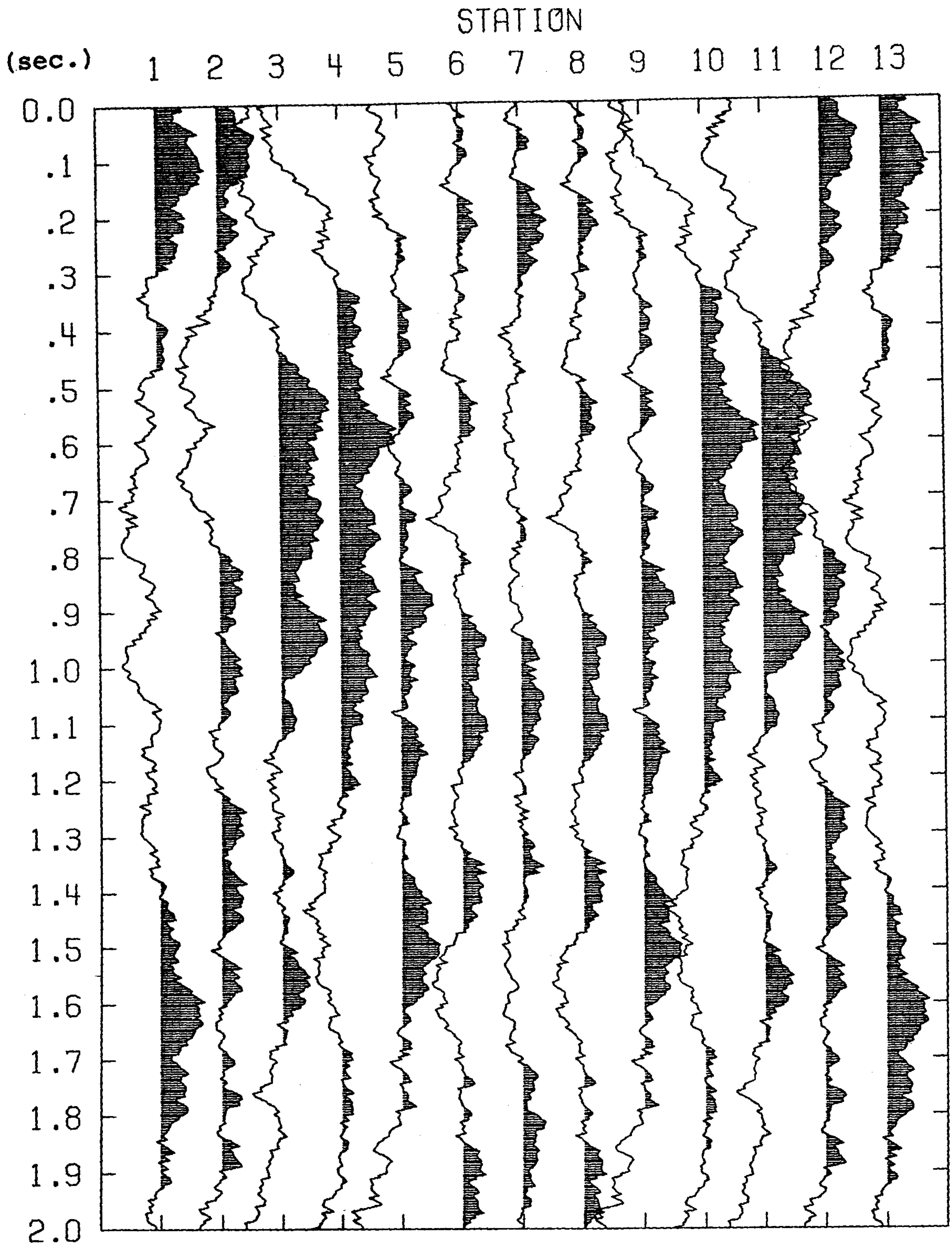
D. Instantaneous phase

Figure 4.2, continued



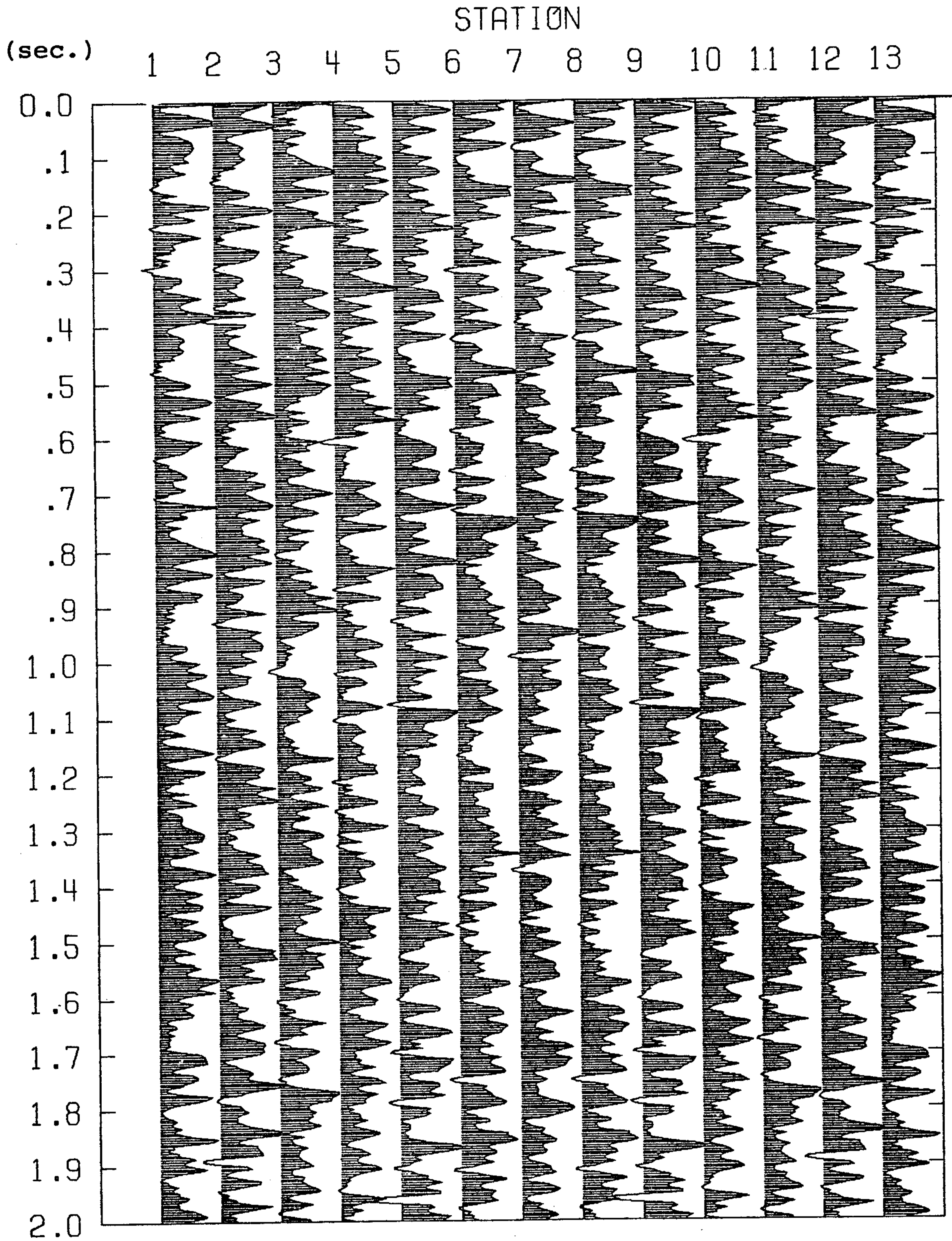
E. Continuous phase

Figure 4.2, continued



F. Residual phase of linear least-square fitting

Figure 4.2, continued

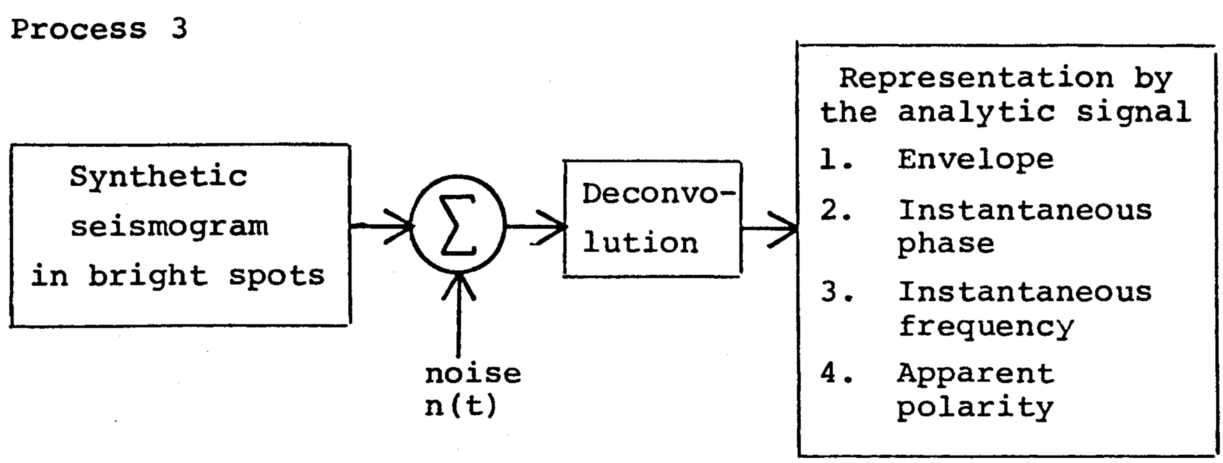


G. Instantaneous frequency

Figure 4.2, continued

4.4 Processing after deconvolution

This process is Process 3 of Section 1.1 and is as follows:

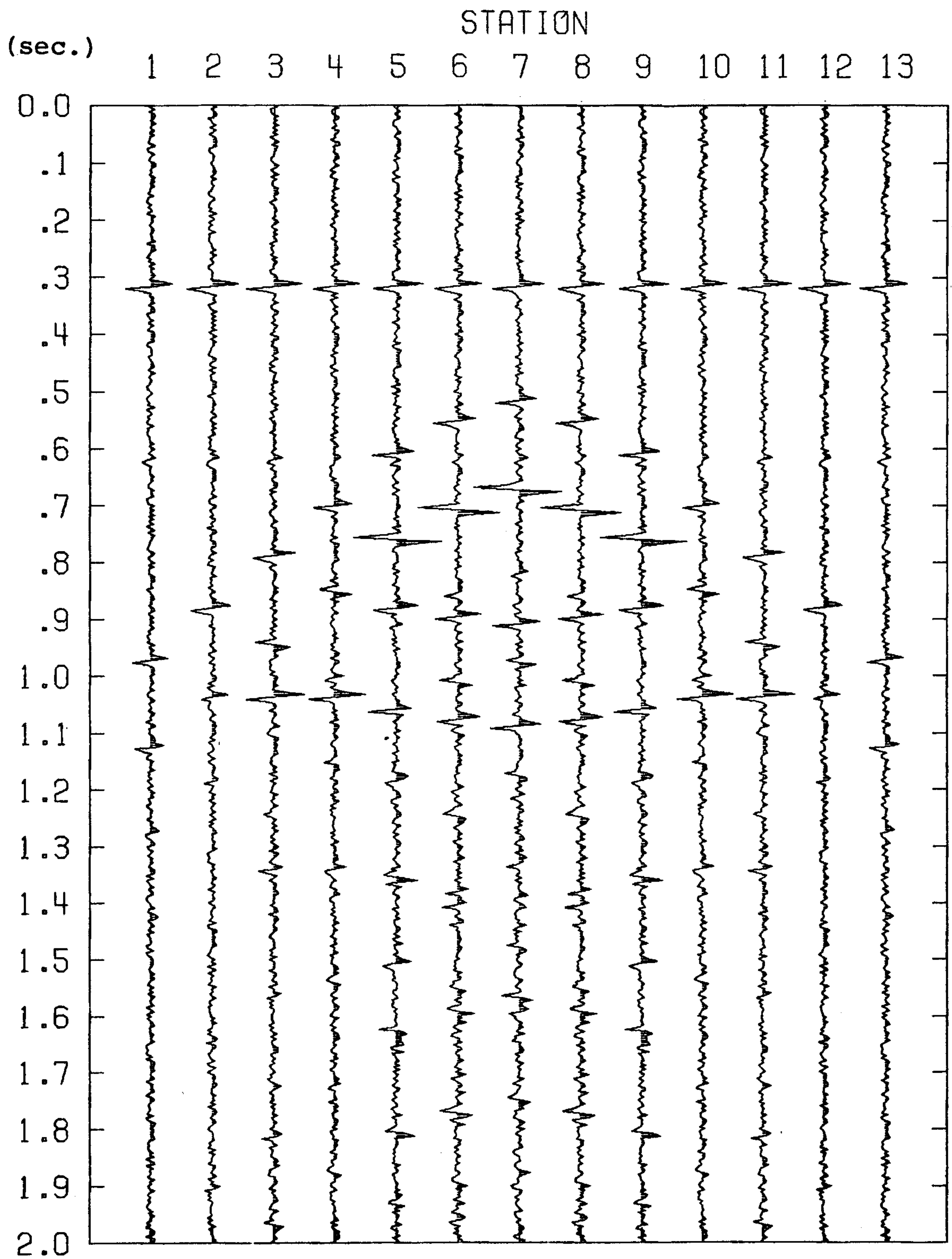


Time and space adaptive deconvolution preprocessing is used here. From the computed outputs shown in Figures 4.3.A, B, C, D, E, F, G, and H, the following is observed:

(1) The noise is reduced in the envelope plot, Figure 4.3.C.

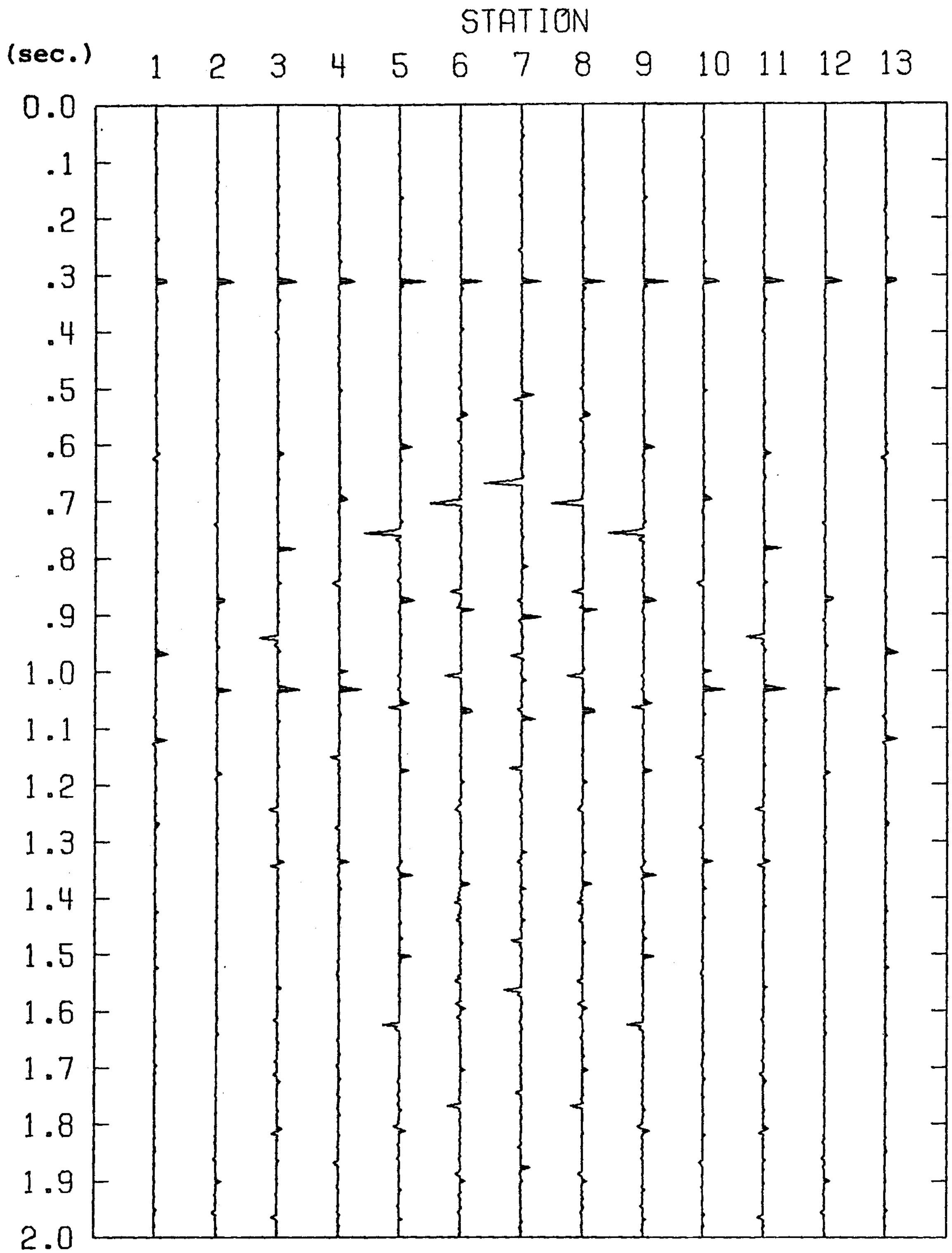
(2) The duration of the apparent polarity signals, Figure 4.3.D, has been reduced. On trace 5, at time about 1.05 seconds, the polarity is still negative, which is not the same as the synthetic seismogram.

(3) In the continuous phase plot, Figure 4.3.F, it appers that some jumps occur when signals are coming, but because of scaling the jumps do not become clear. In the residual phase of the linear least-square fitting, Figure 4.3.G, traces 5, 6, 7 reveal the parabolic curve in the oil and gas zones that occurs because of the reduced velocity.



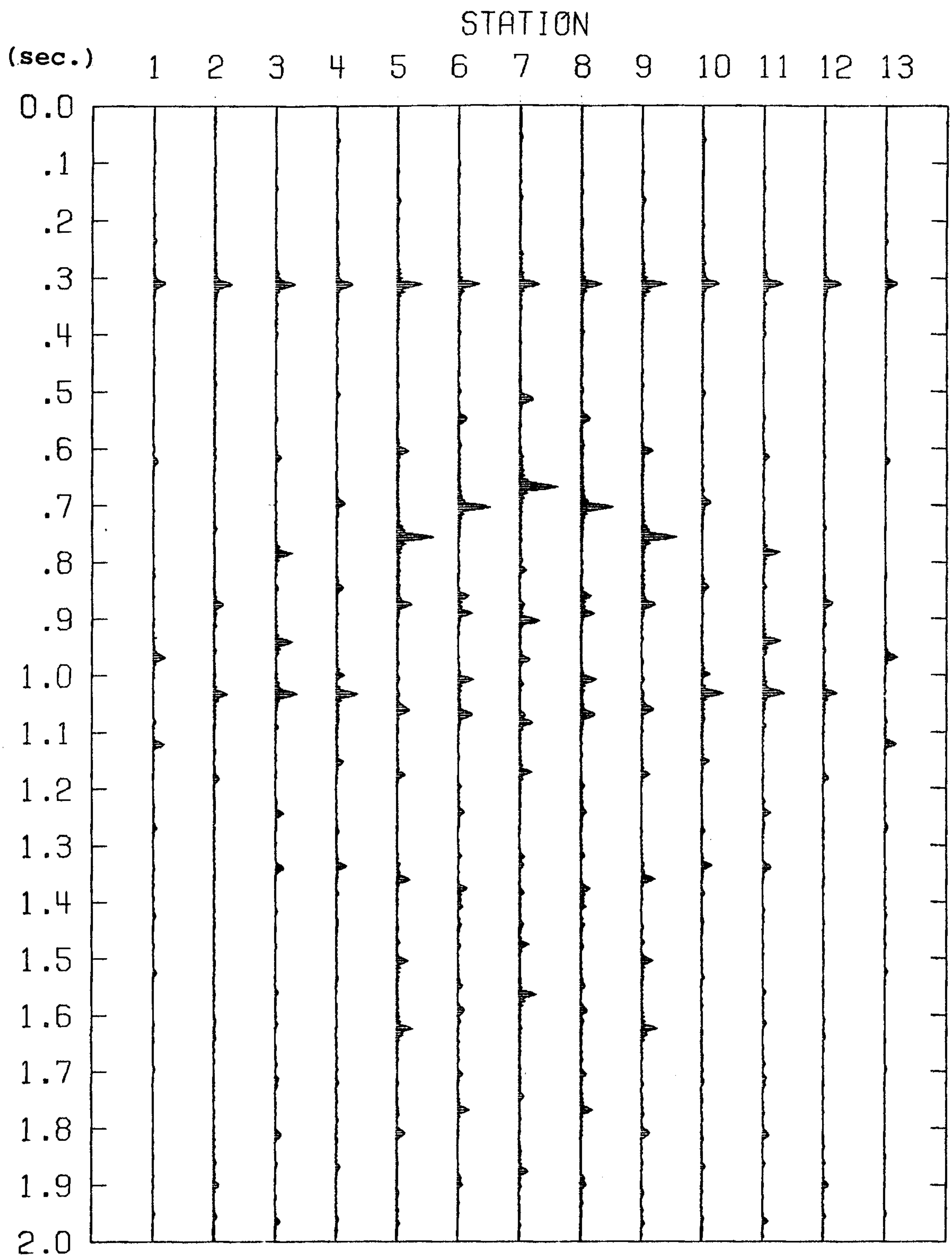
A. Synthetic seismogram plus random noise

Figure 4.3 Analytic signal representation after deconvolution



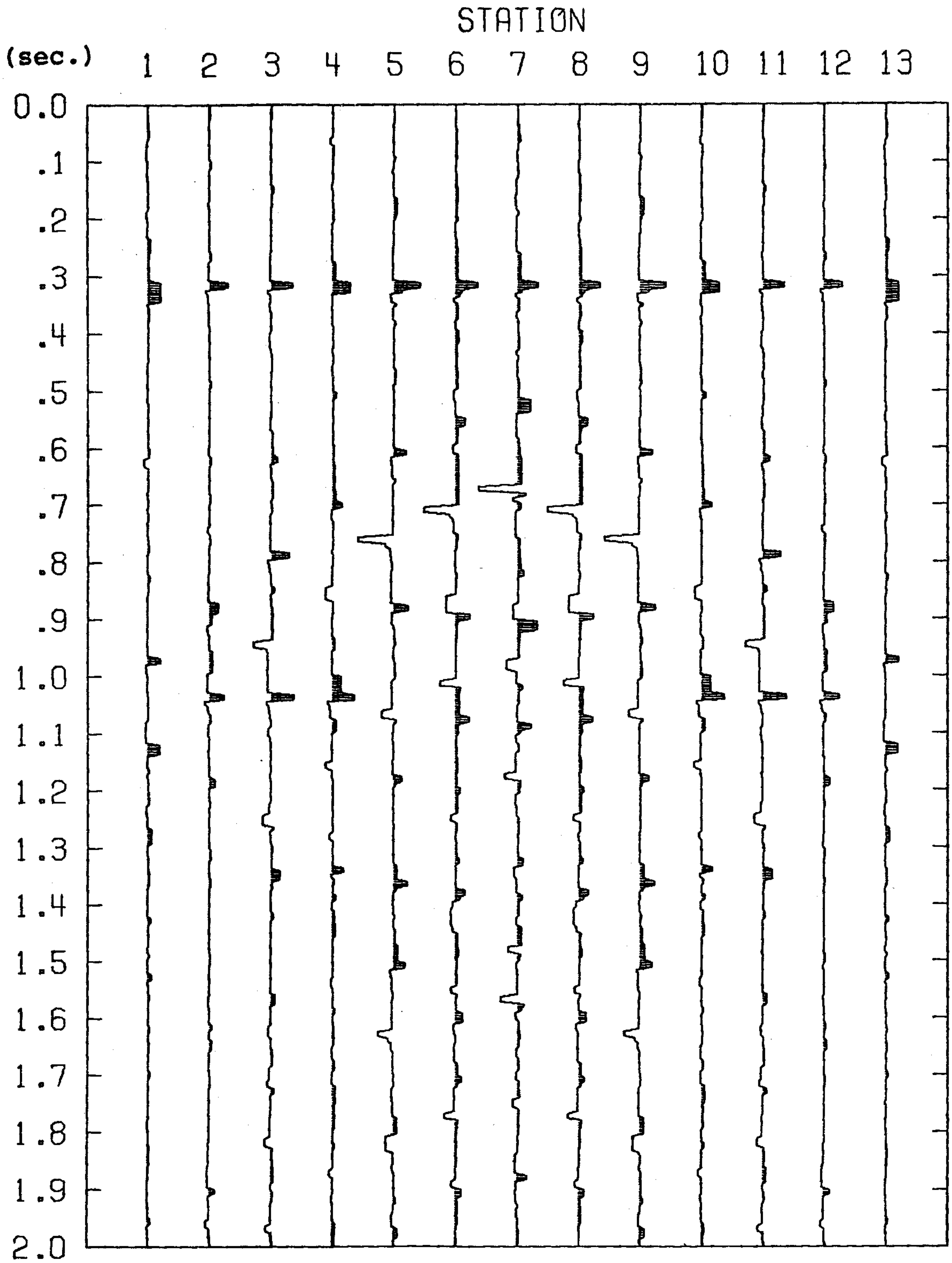
B. Time and space adaptive deconvolution output

Figure 4.3, continued



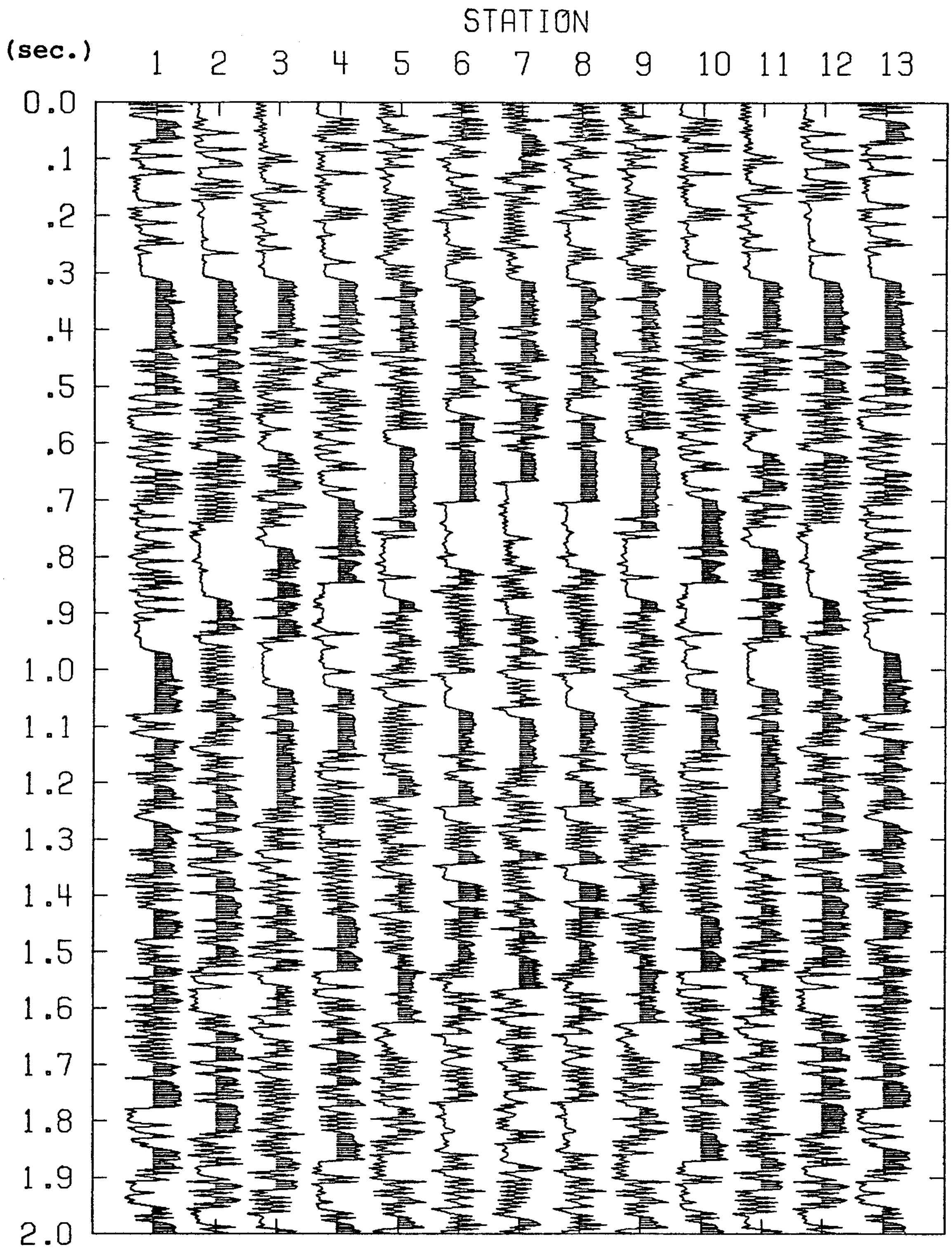
C. Envelope

Figure 4.3, continued



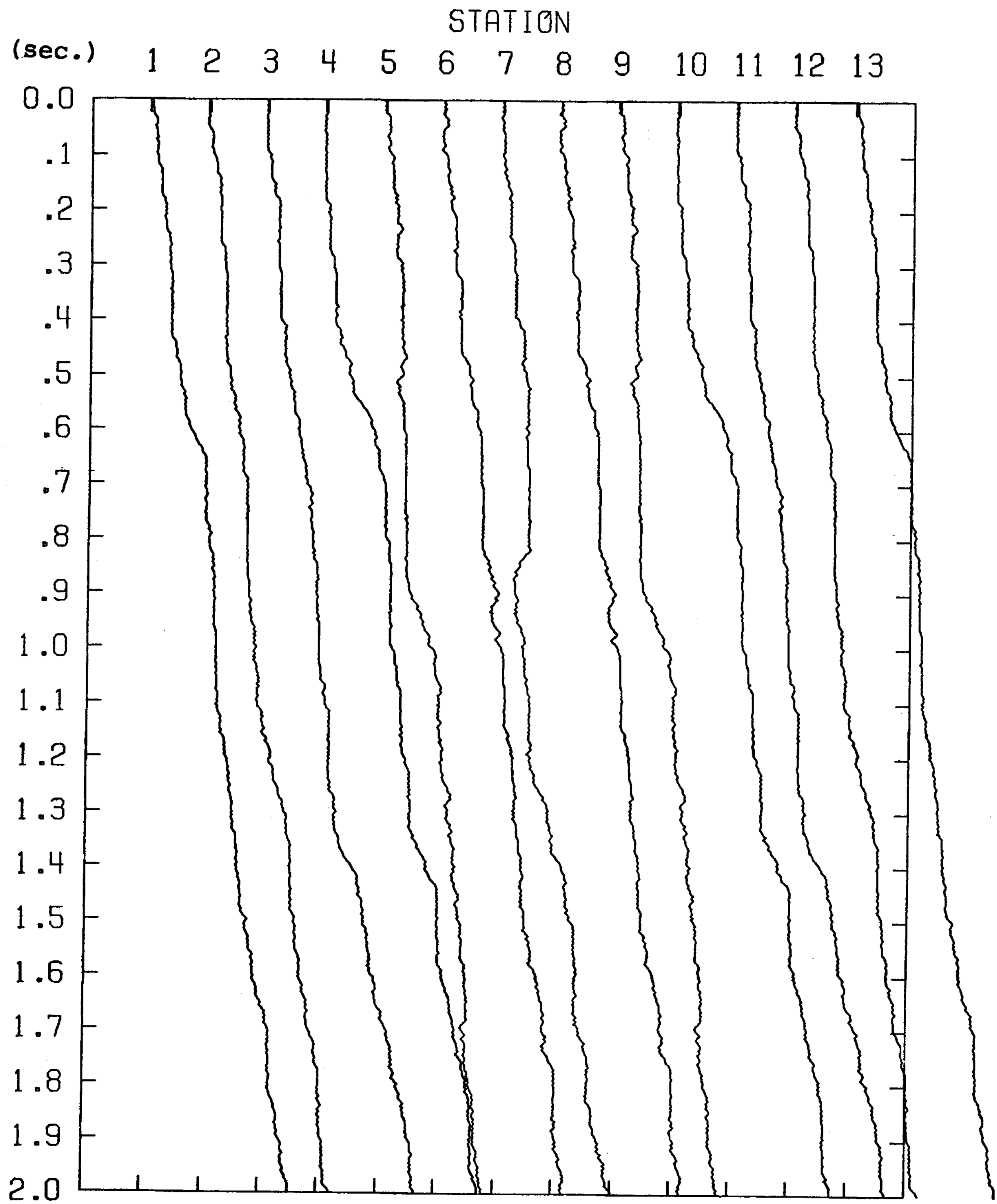
D. Apparent polarity

Figure 4.3, continued



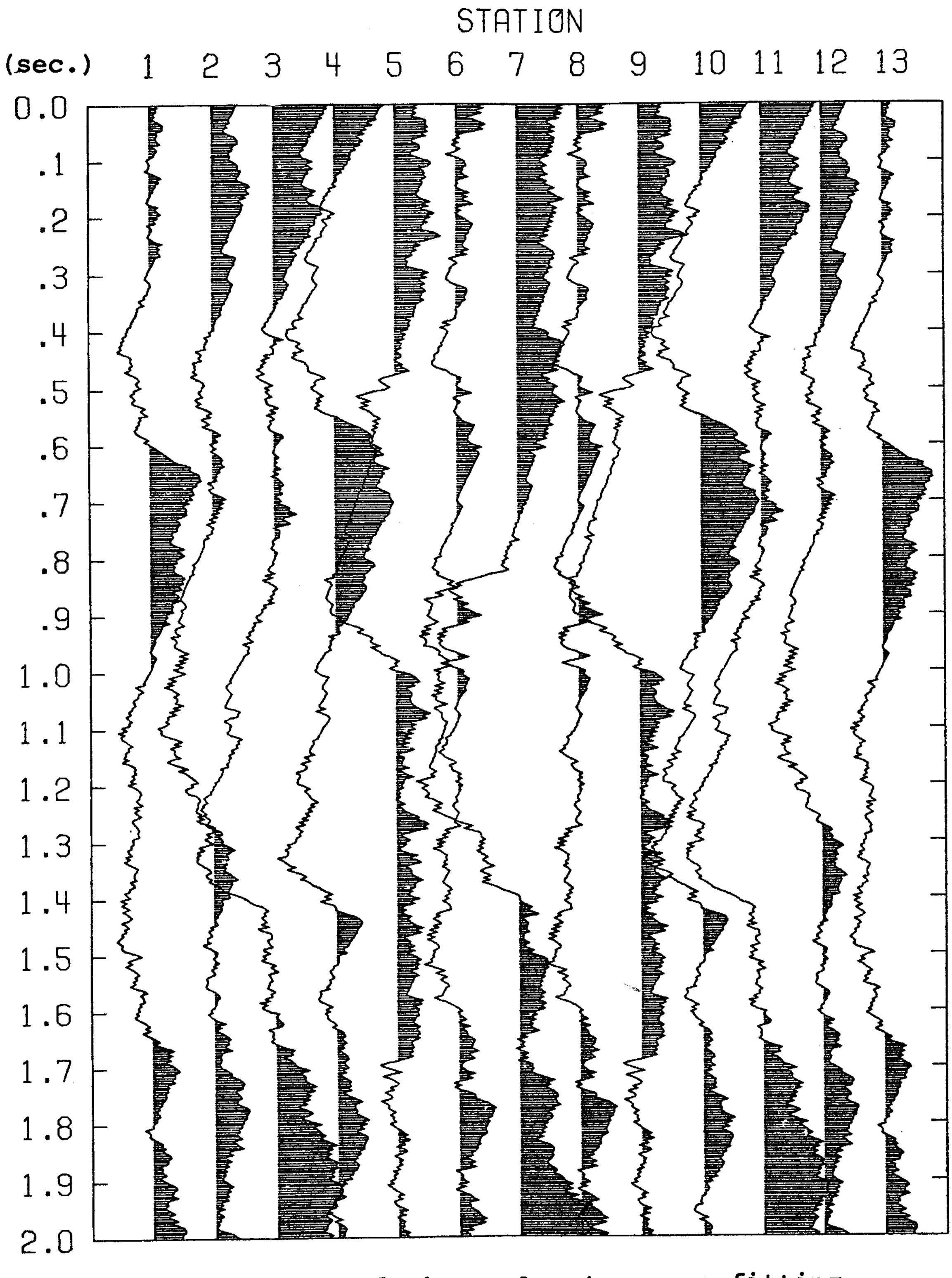
E. Instantaneous phase

Figure 4.3, continued



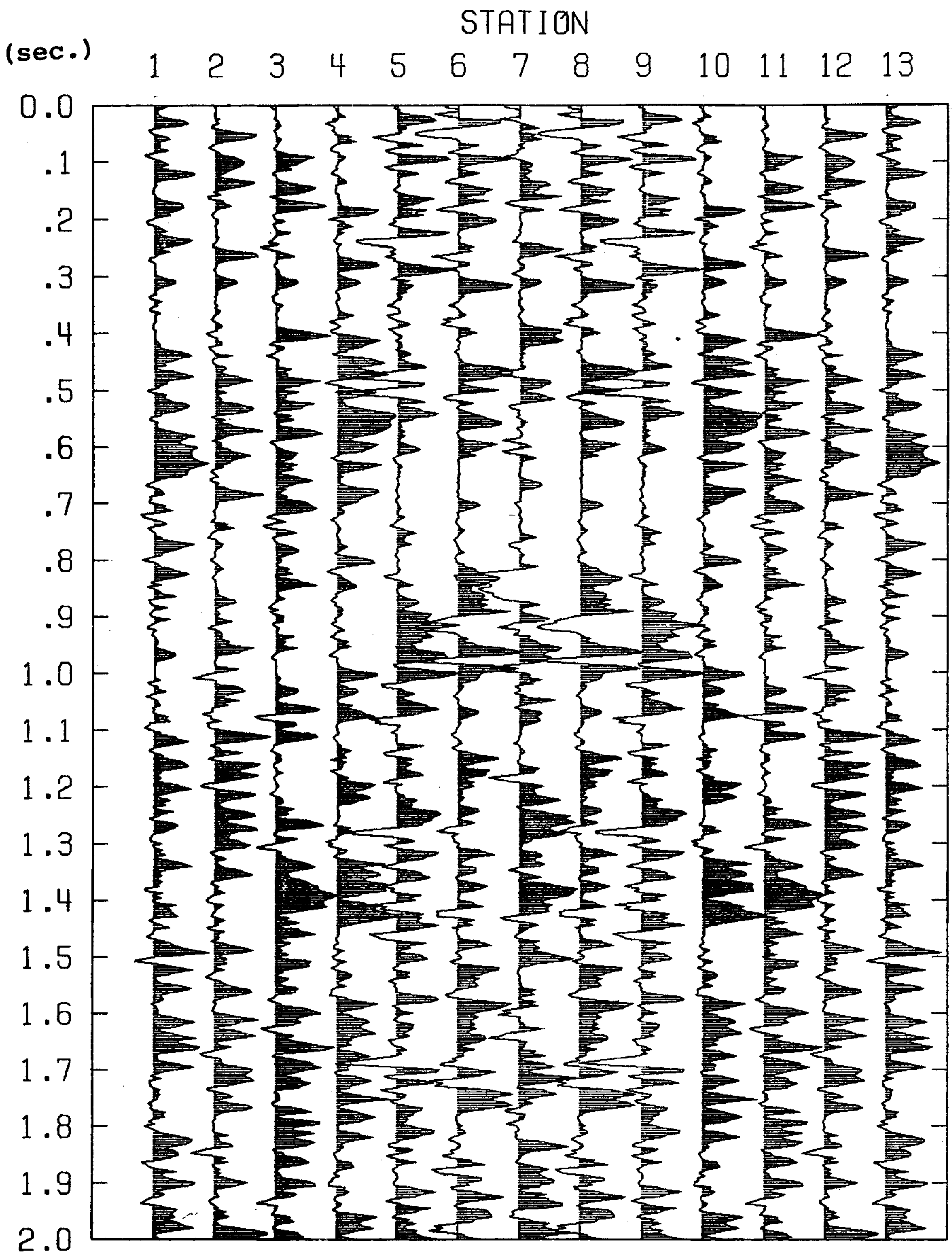
F. Continuous phase

Figure 4.3, continued



G. Residual phase of linear least-square fitting

Figure 4.3, continued



H. Instantaneous frequency

Figure 4.3, continued

(4) In the frequency plot, Figure 4.3.H, we can see some indications of correlation with the structure, but it is still difficult to make classifications, especially for the multiples.

4.5 Summary

From the above results, time and space adaptive deconvolution appears to improve the quality of the analytic signal representation. The analytic signal representation still retains the physical properties of the bright spots.

CHAPTER FIVE

SUMMARY AND CONCLUSIONS

5.1 Summary of the present study

(1) Preprocessing (deconvolution) provides a valuable enhancement in the analytic signal representation.

(2) Time and space adaptive deconvolution is effective for preprocessing of the analytic signal for spiking wavelets and suppressing noise.

(3) Minimum entropy deconvolution is a good preprocessing operation for spiking the wavelet but has polarity problems in the presence of noise.

(4) Large envelope magnitudes occur in the bright spots because of large difference in acoustic impedance that occurs there.

(5) Apparent polarity indicates the phase reversal of the arriving wave in the bright spots.

(6) In the continuous instantaneous phase plot, jumps indicate arriving wavelets.

(7) The parabolic curve in the residual phase of the linear least-square fitting indicates the gas and oil zones because of low velocity.

(8) Theoretically we can use the analytic signal to remove the first order and second order reverberations.

5.2 Suggestions for further investigation

(1) Develop a new technique for deconvolution similar to constrained image restoration processing (N. Y. Chu and C. D. McGillem, 1979) and carry out constrained deconvolution to improve the quality of the outputs.

(2) Use the theory of analytic signal processing to remove the first and second order reverberations in the synthetic seismogram and real data.

(3) Utilize non-linear deconvolution to improve the envelope plot.

(4) Use the representation by analytic signals in seismic pattern recognition.

(5) Use a color coding plot to improve the quality of the instantaneous phase and frequency plot to help the geophysical interpretation.

REFERENCES

[1]. Backus, M. M., "Water Reverberations -- Their Nature and Elimination," Geophysics, Vol. 24, 1959, pp. 233-261.

[2]. Baker, Mark R., "Application of Time Series Analysis to the enhancement of Seismic Refraction Data Interpretation," Master's thesis, Department of Geosciences, Purdue University, 1979.

[3]. Chu, N. Y. and McGillem, Clare D., "Methods and Performance Bounds for Constrained Image Restoration," LARS Technical Report 061678, Laboratory for Applications of Remote Sensing, Purdue University, 1978.

[4]. Claerbout, Jon F., "Synthesis of a Layered Medium From Its Acoustic Transmission Response," Geophysics, Vol. 33, 1968, pp. 264-269.

[5]. Calerbout, Jon F., "Fundamental of Geophysical Data Processing With Applications to Petroleum Prospecting," McGraw-Hill, 1976, New York.

[6]. Farnbach, J. S., "The Complex Envelope in Seismic Signal Analysis," Bulletin of the Seismological Society of America, Vol. 65, No. 4, 1975, pp. 951-962.

[7]. Griffiths, L. J., Smolka, F. R., and Trembly, L. D., "Adaptive Deconvolution: A New Technique for Processing Time-Varying Seismic Data," Geophysics, Vol. 42, 1977, pp. 742-759.

[8]. McGillem, Clare D. and Cooper, George R., "Continuous and Discrete Signal and System Analysis," Holt, Rinehart and Winston, 1971.

[9]. Peacock, K. L., and Treitel, Sven, "Predictive Deconvolution: Theory and Practice," Geophysics, Vol. 34, 1969, pp. 155-169.

[10]. Riley, Don C., "Computer Programs for Time and Space Adaptive Deconvolution of Single Channel Marine Seismic Reflection Data," U.S. Department of the Interior Geological Survey, 1973.

[11]. Robinson, E. A., "Multichannel Time Series Analysis With Digital Computer Program," Holden-Day, 1967, San Francisco.

[12]. Sicking, Charles J., "Modeling With the Complex Trace," presented at the 48th annual International SEG meeting, October 29, 1978, San Francisco.

[13]. Taner, M. T. and Sheriff, R. E., "Application of Amplitude, Frequency, and Other Attributes to Stratigraphic and Hydrocarbon Determination," AAPG, Memoir 26, 1977, pp. 301-327.

[14]. Taner, M. T., Koehler, F., and Sheriff, R. E., "Complex Seismic Trace Analysis," Geophysics, Vol. 44, No. 6, 1979, pp. 1041-1063.

[15]. Treitel, S. and Robinson, E. A., "Seismic Wave Propagation in a Layered Media in Terms of Communication Theory," Geophysics, Vol. 31, 1966, pp. 17-32.

[16]. Wiggins, Ralph A., "Minimum Entropy Deconvolution," Geoexploration, Vol. 16, 1978, pp. 21-35.

APPENDICES

Appendix A
Program SPOT

```
      PROGRAM SPOT(PLOT)
C     THIS PROGRAM GENERATE THE BRIGHT SPOTS MODEL
      DIMENSION AX(15),AY(15),BX(15),BY(15),CX(15),CY(15)
     * ,DX(9),DY(9),EX(13),EY(13)
      CALL PLOTS
      DO 1 I=1,13
      AX(I)=1.*I
      BX(I)=1.*I
      CX(I)=1.*I
      AY(I)=1.32
1     CONTINUE
      BY(1)=0.6 $ BY(2)=0.7 $ BY(3)=0.8 $ BY(4)=0.9 $ BY(5)=1.0
      BY(6)=1.06 $ BY(7)=1.1
      BY(8)=1.06
      BY(13)=0.6$ BY(12)=0.7$ BY(11)=0.8$ BY(10)=0.9$ BY(9)=1.0
      CY( 1)=0.4 $ CY(2)=0.5 $ CY(3)=0.6 $ CY(4)=0.7 $ CY( 5)=0.8
      CY(6)=0.86 $ CY(7)=0.9
      CY(8)=0.86
      CY(13)=0.4 $ CY(12)=0.5$ CY(11)=0.6$ CY(10)=0.7$ CY( 9)=0.8
      DO 2 I=1,7
      DX(I)=3+I
      DY(I)=0.7
2     CONTINUE
      DO 3 I=1,11
      EX(I)=1+I
      EY(I)=0.5
3     CONTINUE
      CALL SCALE(AX,6.0,13,1)
      CALL SCALE(AY,8.0,13,1)
      AX(14)=0.0
      AX(15)=2.3
      CALL AXIS(0.0,0.0,6HX AXIS,-6, 6.0,0.0,AX(14),AX(15),0)
      AY(14)=0.0
      AY(15)=0.2
      CALL AXIS(0.0,0.0,6HY AXIS,6, 8.0,90.0,AY(14),AY(15),-1)
      CALL SYMBOL(1.5,8.3 ,0.2,12HBRIGHT SPOTS,0.0,12)
      CALL SYMBOL(-0.7,5.0,0.2,9HDEPTH(KM) ,90.,9)
      CALL SYMBOL(4.5,-0.5,0.15,8HDISTANCE ,0.0,8)
      CALL PLOT(0.0,8.0,3)
      CALL PLOT(6.0,8.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(AX,AY,13,1,0,0)
      CALL SCALE(BX,6.,13,1)
      CALL SCALE(BY,8.,13,1)
      BX(14)=0.0
      BX(15)=2.3
      BY(14)=0.0
      BY(15)=0.2
      CALL LINE(BX,BY,13,1,0,0)
      CALL SCALE(CX,6.,13,1)
      CALL SCALE(CY,8.,13,1)
      CX(14)=0.0
      CX(15)=2.3
      CY(14)=0.0
      CY(15)=0.2
      CALL LINE(CX,CY,13,1,0,0)
      CALL SCALE(DX,6.,7,1)
      CALL SCALE(DY,8.,7,1)
      DX(8)=0.0
      DX(9)=2.3
      DY(8)=0.0
      DY(9)=0.2
      CALL LINE(DX,DY,7,1,0,0)
      CALL SCALE(EX,6.,11,1)
      CALL SCALE(EY,8.,11,1)
      EX(12)=0.0
```

```
EX(13)=2.3
EY(12)=0.0
EY(13)=0.2
CALL LINE(EX,EY,11,1,0,0)
CALL SYMBOL(2.0,7.3,0.15,17HD=1.8 V=1.8KM/SEC ,0.0,17)
CALL SYMBOL(2.4,6.0,0.15,11HD=2.1 V=2.2 ,0.0,11)
CALL SYMBOL(2.5,4.8,0.1 ,11HD=2.4 V=2.6 ,0.0,11)
CALL SYMBOL(2.5,3.8,0.1 ,11HD=1.9 V=1.7 ,0.0,11)
CALL SYMBOL(2.4,3.0,0.15,11HD=2.2 V=2.2 ,0.0,11)
CALL SYMBOL(2.4,1.5,0.15,11HD=2.7 V=2.8 ,0.0,11)
CALL PLOT(0,0,999)
STOP
END
```

Appendix B
Program THYSS

```
      PROGRAM THYSS(INPUT,OUTPUT,      PLOT,FILE3,FILE4,FILE5,FILE6,
     *  FILE7,FILE8,FILE9,FILE11,TAPE1=INPUT,
     *  TAPE2=OUTPUT,TAPE3=FILE3,TAPE4=FILE4,TAPE5=FILE5,
     *  TAPE6=FILE6,TAPE7=FILE7,TAPE8=FILE8,TAPE9=FILE9,TAPE11=FILE11)
C***********************************************************************
C
C     THIS PROGRAM CALCULATES THE DECONVOLUTION OUTPUTS.  FOUR
C METHODS ARE USED TO COMPARE THE OUTPUTS.  THEY ARE CONVENTIONAL
C DECONVOLUTIN,TIME AND SPACE ADAPTIVE DECONVOLUTION, MINIMUM ENTROPY
C DECONVOLUTION , AND GRADIENT DESCENT DECONVOLUTION.
C***********************************************************************
      DIMENSION TITLE(5),C( 520),CC(520),C2(520),R(520),S(50),T(520)
     *   ,VEL(520),DENS(50),THICK(50),TK(50),TZ(520),RCOEF(50),Z(520)
      DIMENSION X(520),CX(12,520),RNOISE(1024)
      DIMENSION         D(520),E(520),F(520),B(520),A(520)
      DIMENSION         G(50),Y(50),H(520),XX(520,1),SPACE(520)
      COMPLEX SPECT
      CALL PLOTS
      SF= 1.0
      ST=0.0
      SRATE=0.004
      DO 41 I=1,512
      T(I)=SRATE*(I-1)
41    CONTINUE
      READ(1,19) (RNOISE(I),I=1,512)
19    FORMAT(16F5.3)
      X1=0.   $    X2=2.    $    XINCH=1.2 $ NX=2
      T1=0.   $  T2=2.  $ TINCH=8.  $ NT=20
      NSEIS=1   $NDIST=0  $  NFRMT=0  $ NAMP=2  $NEXP=0
      NREV=1  $NPOS=0  $NSCALE=0  $ NFILL=1
      REDVEL=0.  $ TO=0.  $FACTOR=0.  $  AMP=1.  $  FDIST=0.
      DELX=0.  $  ALPHA=0.
      READ(1,10)    (TITLE(J),J=1,5)
      WRITE(3,10) (TITLE(J),J=1,5)
      WRITE(3,11)X1,X2,XINCH,NX
      WRITE(3,11)T1,T2,TINCH,NT
      WRITE(3,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(3,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)    (TITLE(J),J=1,5)
      WRITE(4,10) (TITLE(J),J=1,5)
      WRITE(4,11)X1,X2,XINCH,NX
      WRITE(4,11)T1,T2,TINCH,NT
      WRITE(4,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(4,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)    (TITLE(J),J=1,5)
      WRITE(5,10)(TITLE(I),I=1,5)
      WRITE(5,11)X1,X2,XINCH,NX
      WRITE(5,11) T1,T2,TINCH,NT
      WRITE(5,12) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE( 5,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)    (TITLE(J),J=1,5)
      WRITE(6,10) (TITLE(J),J=1,5)
      WRITE(6,11)X1,X2,XINCH,NX
      WRITE(6,11)T1,T2,TINCH,NT
      WRITE(6,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(6,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)    (TITLE(J),J=1,5)
      WRITE(7,10) (TITLE(J),J=1,5)
      WRITE(7,11)X1,X2,XINCH,NX
      WRITE(7,11)T1,T2,TINCH,NT
      WRITE(7,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(7,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      XTAU=10.  $ ZTAU=0.16
      NZC=0  $ LCN=10  $  IFLGAP=1
      DWARM=0.16
      WSTRT=0.2
      NWARM=DWARM/ SRATE
      NSTRT=WSTRT/ SRATE
```

```
      DIS =     1.0
       READ(1,100)  NOL,N,NEXT,SLNGTH,DT
      WRITE(2,100)  NOL,N,NEXT,SLNGTH,DT
      DO 5 I=1,NOL
       READ(1,26) VEL(I),DENS(I),THICK(I)
      IF(DENS(I).LT..001) DENS(I)=.252+.3788*VEL(I)
      WRITE(2,26) VEL(I),DENS(I),THICK(I)
    5 CONTINUE
C
C GENERATE THE SYNTHETIC SEISMOGRAM
C
      CALL        SYNTH(C2,TK,THICK,VEL,TZ,RCOEF ,DENS,Z,N,R,SLNGTH,
     *       DT,NEXT,S,C,NPTS,NP,NOL)
C
C SYNTHETIC SEISMOGRAM PLUS NOISE
C
      DO 22 I=1,512
22    C(I)=C(I)+0.03*RNOISE(I)
      WRITE(3,17) NP,DIS ,ST,DT   ,SF
      WRITE(3,18) (   C(I),I=1,512)
      DO 1 I=1,NP
      X(I)=C(I)
1     CONTINUE
      ST=0.  $  SF=1.   $  NP=512   $  DT=0.004
C
C CALCULATE  TIME AND SPACE ADAPTIVE DECONVOLUTION
C
      CALL BAFL(512,CX,IFLGAP,LCN,NWARM,NSTRT,ZTAU,XTAU,NZC,X)
7     DO 4 I=1,LCN
4     X(I)=0.0
      DO  88 I=1,512
88    D(I)=C(I)
C
C CALCULATE CONVENTIONAL DECONVOLUTION
C
      LC=10
      LD=512+1
      CALL DECON(LD,D,CC,F,LC,B,E,A)
      CALL FOLD(LD,D,LC,A,LR,B)
C
C CALCULATE MINIMUM ENTROPY DECONVOLUTION
C
      LSG=512
      LSMTH=1
      LF=10
      IOFDLY=0
      ROFCT=1.10
      LAMN=1
      NITER=6
      DO 75 I=1,LD
      XX(I,1)=D(I)
75    CONTINUE
      DO 79 I=1,LF
      G(I)=0.0
79    CONTINUE
      G(5)=1.0
      CALL WVINV(XX,LSG,NSG,IDMLSG,LTAPER,LSMTH,LF,IOFDLY,
     *  ROFCT,LAMN,NITER,G,VAR,Y,SPACE)
      WRITE(2,80) (Y(I),I=1,20)
80    FORMAT(1X,10F8.4)
      CALL FOLD(10,G,NP,D,LZ,H)
C
C CALCULATE GRADIENT DESCENT DECONVOLUTION
C
      CALL ADECON(C,3,512,0.01,10,1)
      WRITE(4,17) NP  ,DIS ,ST, DT    ,SF
      WRITE(5,17) NP  ,DIS ,ST, DT    ,SF
      WRITE(6,17) NP  ,DIS ,ST, DT    ,SF
```

```
      WRITE(7,17) NP  ,DIS ,ST, DT    ,SF
      WRITE( 4,18) (   B(I),I=1,512)
      WRITE(5,18)  ( X(I),I=1,512)
      WRITE(6,18) (   H(I),I=1,NP)
      WRITE(7,18)(C(I),I=1,512)
      REWIND 3
      CALL SECTION(3)
      REWIND 4
      CALL SECTION(4)
      REWIND 5
      CALL SECTION(5)
      REWIND 6
      CALL SECTION(6)
      REWIND 7
      CALL SECTION(7)
   26 FORMAT(3F10.3)
  100 FORMAT(3I5,2F10.5)
   10 FORMAT(5A10)
   11 FORMAT(3F10.3,I5)
   12 FORMAT(9I5)
   13 FORMAT(7F10.3)
   14 FORMAT(3I5,2F10.3)
   16 FORMAT(3F10.3)
   17 FORMAT(I5,3F10.4,F20.10)
   18 FORMAT(8F10.3)
10001 FORMAT(5A10)
10002 FORMAT(3F10.3,I5)
10003 FORMAT(9I5)
10004 FORMAT(7F10.3)
10005 FORMAT(I5,3F10.3,F20.10)
      CALL PLOT(0,0,999)
      STOP
      END
      SUBROUTINE DECON(LX,X,EP,EM,LC,C,A,S)
      DIMENSION X(LX),EP(LX),EM(LX),C(LC),A(LC),S(LX)
C           X(LX) INPUT
C           S(LC) SPIKING FILTER,OUTPUT
C           EP(LX),EM(LX),C(LC),A(LC)  SCRATCH ARRAYS
      DO 10 I=1,LX
10    S(I)=0.0
      A(1)=1.
      DO 20 I=1,LX
      EM(I)=X(I)
20    EP(I)=X(I)
      DO 60 J=2,LC
      TOP=0.0
      BOT=0.0
      DO 30 I=J,LX
      BOT=BOT+EP(I)*EP(I)+EM(I-J+1)*EM(I-J+1)
30    TOP=TOP+EP(I)*EM(I-J+1)
      C(J)=2.*TOP/BOT
      DO 40 I=J,LX
      EPI=EP(I)
      EP(I)=EP(I)-C(J)*EM(I-J+1)
40    EM(I-J+1)=EM(I-J+1)-C(J)*EPI
      A(J)=0.0
      DO 50 I=1,J
50    S(I)=A(I)-C(J)*A(J-I+1)
      DO 60 I=1,J
60    A(I)=S(I)
      RETURN
      END
      SUBROUTINE WVINV(X,LSG,NSG,IDMLSG,LTAPER,LSMTH,LF,IOFDLY,
     * ROFCT,LAMN,NITER,F,VAR,Y,SPACE)
C WVINV02   WVINV
C
C MINIMUM ENTROPY SOURCE WAVELET INVERSE
C
```

```
C                    ABSTRACT
C
C      COMPUTE THE VARIMAX SPIKING WAVELET FOR A SEQUENCE OF TRACE
C      SEGMENTS
C
C
C------------------------PROGRAM HISTORY-----------------------------
C
C AUTHOR     -R.A. WIGGINS                                  JAN 76
C
C---------------------USAGE------------------------------------------
C
C
C SAMPLE CALL STATEMENT
C      CALL WVINV(X,LSG,NSG,IDMLSG,LTAPER,LSMTH,LF,IOFDLY,ROFCT,LAMN,
C    *   NITER,F,VAR,Y,SPACE)
C
C----------PROGRAM INPUTS
C
C      X(I,J)     I=1...LSG, J=1...NSG  DIMENSIONED X(IDMLSG,NSG)
C                   CONTAINS THE TIME SERIES SEGMENTS OF LENGTH LSG FROM
C                    WHICH THE FILTER IS TO BE FOUND.
C
C      LSG,NSG,IDMLSG - SEE  X(I,J).
C
C      LTAPER     THE NUMBER OF SAMPLES ON BOTH ENDS OF EACH SEGMENT OF
C                    X(I,J) TO WHICH A LINEAR TAPER IS TO BE APPLIED.
C                 LTAPER=0 IMPLIES NO TAPERING.
C                 LTAPER .LT. 0 IMPLIES THAT THE TAPER IS NOT TO BE
C                   REMOVED BY WVINV AFTER THE FILTER IS FOUND.
C
C      LSMTH        IS THE LENGTH OF THE SMOOTHING OPERATOR THAT WILL BE
C                   CONVOLVED WITH THE INVERSE OF THE OUTPUT FILTER.
C                   IF LSMTH .LE. 0 NO INVERSE IS FOUND.
C
C      IF         LENGTH OF THE FILTER TO BE FOUND.
C
C      IOFDLY     BETWEEN 1 AND LF IF WVINV SHOULD SET UP AN INITIAL
C                   ESTIMATE OF F(I).  THE ESTIMATE USED IS ALL ZEROS
C                   WITH A UNIT SPIKE AT DELAY IOFDLY.
C                 =0  IF  F(I) ALREADY CONTAINS AN INITIAL ESTIMATE.
C      ROFCT      FACTOR BY WHICH THE CENTER TERMS OF THE AUTOCORRELATIONS
C                   OF X(I,J) ARE TO BE MULTIPLIED BY BEFORE THE FILTER
C                   IS FOUND.
C                 ROFCT=1.01 IS EQUIVALENT TO ADDING 1= WHITE NOISE.
C
C      LAMN       MINIMUM LENGTH OF THE PREDICTION ERROR OPERATOR
C                   TO BE COMPUTED BY WLLSEF.
C                 LAMN=1 IS SUFFICIENT FOR MANY APPLICATIONS.
C
C      NITER      NUMBER OF ITERATIONS TO PERFORM TO DETERMINE F(I).
C
C      F(I)     I=1...LF CONTAINS AN INITIAL ESTIMATE OF THE FILTER
C                   IF  FSETUP=F.
C
C      Y(I)       I=1...LSG+LF+LSMTH IS TEMPORARY COMPUTATION SPACE.
C
C      SPACE(I)   I=1...5*LF IS TEMPORARY COMPUTATION SPACE.
C
C------------PROGRAM OUTPUTS
C
C OUTPUT ARGUMENTS
C
C      F(I)       I=1..LF  CONTAINS THE NEW MED FILTER ESTIMATE.
C
C      Y(I)       I=1...2*LF  CONTAINS THE INVERSE OF THE FILTER
C                   IF LSMTH(2) .GE.1.
C
```

```
C     VAR       IS THE NORMALIZED VARIANCE OF THE SQUARES OF THE FILTERED
C                 X SEGMENTS.
C
C***********************************************************************
C
C
C
C MAXIMUM SPIKINESS WAVELET INVERSE.
C
      REAL X(IDMLSG,NSG),F(2),Y(2),SPACE(2)
C
C-------------------------------------------------------------
C COLLECT THE AUTOCORRELATIONS OF THE X SEGMENTS
C
      LT=IABS(LTAPER)
      DWT=1./(LT+1)
      LSEG=LSG
      LWTG=LF/10
      IF(LWTG.LT.1) LWTG=1
      IR1=LWTG+1
      IW1=IR1+LF
      IW=IW1-1
      SPACE(IW1)=DWT
      DO 111 I=1,LT
111   SPACE(I+IW1)=SPACE(I+IW) + DWT
      EWT=1.57/LWTG
      WT=EWT
      DO 112 I=1,LWTG
      SPACE(I)=SIN(WT)
112   WT=WT+EWT
      DO 113 I=1,LF
113   SPACE(I+IR1-1)=0.
C
      LCOR=MIN0(LF,LSG)
      DO 140 K=1,NSG
C
C SUBTRACT THE MEAN
C
      XMN=0.
      DO 1110 I=1,LSG
1110  XMN=XMN+X(I,K)
      XMN=XMN/LSG
      DO 1120 I=1,LSG
1120  X(I,K)=X(I,K)-XMN
C
C TAPER THE ENDS
C
      LSG1=LSG+1
      DO 1130 I=1,LT
      X(I,K)=X(I,K)*SPACE(I+IW)
1130  X(LSG1-I,K)=X(LSG1-I,K)*SPACE(I+IW)
C
C COMPUTE THE AUTOCORRELATION
C
130   PRO=0.
      DO 1140 I=1,LSG
1140  PRO=PRO+X(I,K)**2
      SCL=1./PRO
      SPACE(IR1)=SPACE(IR1)+1
      DO 1160 J=2,LCOR
      J1=J-1
      LXM=LSG-J1
      XC=0.
      DO 1150 I=1,LXM
1150  XC=XC+X(I,K)*X(I+J1,K)
1160  SPACE(IR1+J1)=SPACE(IR1+J1)+XC*SCL
C
140   CONTINUE
```

```
      SPACE(IR1)=SPACE(IR1)*ROFCT
C
C
      IA1=LF+IR1+1
      LAA=LF
      IG0=IA1+LAA-1
      IG1=IA1+LAA
C
C-----------------------------------------------------------
C SET UP THE INITIAL VALUE FOR F
C
      IF(IOFDLY*(IOFDLY-LF+1).GE.0) GO TO 260
      DO 250 I=1,LF
250   F(I)=0.
      F(IOFDLY)=1.
260    CONTINUE
C
C-----------------------------------------------------------------------------
C LOOP OVER THE ITERATIONS
C
      ITER=0
C
300   ITER=ITER+1
C
C-------------------------------------------------------------
C CONVOLVE F WITH X TO GET Y
C
C****
      Y2=0.
      Y4=0.
      DO 310 I=1,LF
310   SPACE(I+IG0) =0.
      LYT=LSG+LF-1
C
      DO 370 K=1,NSG
      DO 1310 I=1,LYT
1310   Y(I)=0.
      DO 1320 I=1,LSG
      DO 1320 J=1,LF
      IJ=I+J-1
      Y(IJ)=Y(IJ)+X(I,K)*F(J)
1320  CONTINUE
C
C------------------------------------------------------------
C CUBE Y
C
      Y2T=0.
      Y4T=0.
      DO 1330 I=1,LYT
      YI2=Y(I)**2
      Y2T=Y2T+YI2
      Y4T=Y4T+YI2**2
1330  Y(I)=Y(I)*YI2
      Y4=Y4+Y4T
      Y2=Y2+Y2T
C
      VT=Y4T/(Y2T*Y2T)
      SCL=1./Y4T
      IF(ITER.GT.NITER) GO TO 370
C
C-----------------------------------------------------------------------------
C CORRELATE X WITH Y**3
C
      DO 1360 I=1,LF
      I1=I-1
      GI=0.
      DO 1350 J=1,LSG
      JI=J+I1
```

```
1350  GI=GI+Y(JI)*X(J,K)
      IIG=I+IG0
1360  SPACE(IIG)=SPACE(IIG)+GI*SCL
C
370   CONTINUE
      VAR=Y4/(Y2*Y2)
      IF(ITER.GT.NITER) GO TO 500
C
C--------------------------------------------------------
C FIND THE FILTER AND NORMALIZE
C
      IGL=IG1+LF
      DO 1370 I=1,LWTG
      SPACE(I+IG0)=SPACE(I+IG0)*SPACE(I)
1370  SPACE(IGL-I)=SPACE(IGL-I)*SPACE(I)
C
      LAA=LAMN
      CALL WLLSEF(8*LSG,LCOR,SPACE(IR1),LF,SPACE(IG1),F,LAA,SPACE(IA1))
C
      FMX=0.
      DO 420 I=1,LF
420   IF(FMX.LT.ABS(F(I))) FMX=ABS(F(I))
      DO 430 I=1,LF
430    F(I)=F(I)/FMX
      GO TO 300
C
C-----------------------------------------------------------------
C REMOVE THE TAPER FROM X
C
500   IF(LTAPER.LT.1) GO TO 550
      WT=DWT
      DO 520 I=1,LT
      WTI=1./WT
      DO 510 K=1,NSG
      X(I,K)=X(I,K)*WTI
510   X(LSG1-I,K)=X(LSG1-I,K)*WTI
520   WT=WT+DWT
C
C-----------------------------------------------------------------
C INVERT F IF DESIRED.
C
550   LSM=LSMTH
      IF(LSM.LT.1) GO TO 600
      LINV=2*LF
      IG1=LF+1
      IA1=IG1+LINV
C
      LR=LF
      DO 570 I=1,LF
      I1=I-1
      RI=0.
      DO 560 J=1,LR
560    RI=RI+F(J)*F(J+I1)
      SPACE(I)=RI
570   LR=LR-1
      DO 571 I=1,LINV
571   SPACE(I+LF)=0.
      IGF=IG1+LF/2-1
      LF1=LF+1
      DO 572 I=1,LF
572    SPACE(I+IGF)=F(LF1-I)
C
      SPACE(1)=SPACE(1)*ROFCT
      LA=LINV
      CALL WLLSEF(4*LF,LF,SPACE,LINV,SPACE(IG1),Y,LA,SPACE(IA1))
C
      IF(LSM.LE.1) GO TO 600
      LINVS=LINV+3*(LSM-1)
```

```
      DO 575 I=LINV,LINVS
575   Y(I+1)=0.
C
      LSM1=LSM+1
      DO 585 IREP=1,3
      DO 580 I=2,LINVS
580    Y(I)=Y(I)+Y(I-1)
      I=LINVS
      DO 585 IR=LSM1,LINVS
      Y(I)=Y(I)-Y(I-LSM)
585   I=I-1
C
600   RETURN
      END
      SUBROUTINE WLLSEF(LX,LR,R,LGF,G,F,LA,A)
C
C WLLSEF
C        WIENER-LEVENSION LEAST-SQUARES ERROR FILTER OR  PREDICTOR
C
C                               ABSTRACT
C
C     THE STANDARD WIENER LEAST-SQUARES ERROR OPTIMUM FILTER DESIGN
C     EQUATIONS ARE SOLVED USING THE LEVENSON RECURSION ALGORITHM FOR
C     THE SOLUTION OF A TOEPLITZ MATRIX SYSTEM.  SPECIAL ERROR MONITOR-
C     ING HAS BEEN INCORPORATED TO ECONOMIZE THE FILTER COMPUTATION.
C
C
C-------------------------PROGRAM HISTORY------------------------------
C
C AUTHOR - R.A. WIGGINS    (WLLSEF) FORTRAN                 AUGUST,1976
C
C--------------------------------DISCUSSION----------------------------
C
C TECHNICAL REFERENCES-
C
C     THIS ROUTINE WAS DEVELOPED FROM
C       SUBROUTINE WLLSFP BY R.A. WIGGINS (1962) PRESENTED IN
C         S.M.SIMPSON JR, TIME SERIES COMPUTATIONS IN FORTRAN AND FAP,
C         ADDISON-WESLEY, 1966; AND
C       SUBROUTINE LSPOV1 BY J.C. ROSS (1970) WGC.
C
C     THE RECURSION ALGORITHM IS GIVEN BY N. LEVINSON IN
C       N.WIENER, EXTRAPOLATION, INTERPOLATION, AND SMOOTHING OF
C         STATIONARY TIME SERIES,JOHN WILEY $ SONS,1949,
C     AND MORE RECENTLY DISCUSSED AT SOME LENGTH IN
C       A.J.BERKHOUT AND P.R.ZAANEN, A COMPARISON BETWEEN WIENER
C         FILTERING,KALMAN FILTERING, AND DETERMINISTIC LEAST SQUARES
C         ESTIMATION, GEOPHYCICAL PROSPECTING 24,141-197,1976.
C
C     GIVEN AN AUTOCORRELATION VECTOR RR(I)  I=0,1,....,M
C     AND A CROSSCORRELATION VECTOR    GG(I)  I=0,1,....,M  (OF THE
C     INPUT WITH THE DESIRED OUTPUT)  THIS ROUTINE SOLVES THE
C     TOEPLITZ SYSTEM OF SIMULTANEOUS EQUATIONS
C
C
C          M
C         SUM (FF(N)*RR(K-N))  =  GG(K)   K=0,1,....,M
C         N=0
C
C     AND
C
C          M
C         SUM (AA(N)*RR(K-N))  =  EE(K)   K=0,1,....,M
C         N=0
C
C     WHERE EE(0) = ALPHA,  EE(I) = 0.  I=1,2,...,M.  THE TERMS
C     FF(I) I=0,1,....,M  ARE THE OPTIMUM FILTER COEFFICIENTS FOR
C     SHAPING THE INPUT TO THE DESIRED OUTPUT.  THE TERMS
```

```
C     AA(I) I=0,1,,...,M ARE THE PREDICTION ERROR COEFFICIENTS FOR A
C     UNIT PREDICTION LAG AND ALPHA IS THE EXPECTED PREDICTION ERROR.
C
C     AS THE LENGTH OF AA AND FF ARE INCREASED RECURSIVELY,
C     THE VALUE OF ALPHA DECREASES MONOTONICALLY.  BERKHOUT AND
C     ZAANEN POINT OUT THAT WHEN THE ERROR IN ESTIMATING THE PREDICTION
C     COEFFICIENTS IS CONSIDERED, THE TOTAL EXPECTED PREDICTION ERROR
C     IS ALPHA*(LX+1)/(LX-M)  WHERE LX IS THE LENGTH OF DATA SERIES
C     THAT WAS AUTOCORRELATED TO OBTAIN RR.  THIS MODIFIED ERROR TERM
C     WILL HAVE A LOCAL MINIMUM AT OR NEAR THE OPTIMUM LENGTH OF THE
C     PREDICTION ERROR OPERATOR.
C
C     WLLSEF ACHIEVES SPEED BY ECONOMIZING ON COMPUTATIONS IN THREE
C     WAYS
C
C       1.   FOLLOWING J.C.ROSS, A SPECIAL LOOP IS BRANCHED TO  IF A
C            UNIT LAG PREDICTION FILTER IS DESIRED.  THIS SAVES ABOUT
C            40 PER CENT FOR THIS SPECIAL CASE.
C
C       2.   IF THE AUTOCORRELATION IS SHORTER THAN THE DESIRED FILTER
C            LENGTH, THE CORRESPONDING DOT PRODUCTS ARE REDUCED
C            ACCORDINGLY.
C
C       3.   THE EXTENSION OF THE PREDICTION ERROR OPERATOR IS
C            TERMINATED AT LENGTH LA IF ALPHA(LA-1)/(LX-LA+1) IS
C            LESS THAN ALPHA(LA)/(LX-LA).  HENCE, THE LARGER LX THE
C            LARGER WILL BE THE TERMINAL LA.    IF LA = 0.75*M,  THE
C            RUNNING TIME IS DECREASED 19 PERCENT:  IF LA =0.50*M,
C            BY 38 PERCENT, AND IF LA=0.25*M, BY 53 PERCENT.
C
C
C-----------------------------USAGE-----------------------------------
C
C     NAMES BEGINNING WITH I-N ARE INTEGER, OTHERS ARE REAL.
C
C     TO PRESERVE COMPATIBILITY, THIS ROUTINE HAS TWO ENTRIES -
C       WLLSEF AND LSPO
C
C-----------------PROGRAM EVOCATION(WLLSEF)
C SAMPLE CALL STATEMENT
C
C     CALL WLLSEF (LX,LR,R,LGF,G,F,LA,A)
C
C----------------------PROGRAM INPUTS
C INPUT ARGUMENTS
C
C     LX        LENGTH OF THE DATA SERIES THAT WAS CORRELATED TO OBTAIN
C                 R(I). LX IS USED ONLY FOR JUDGING WHEN TO TERMINATE
C                 THE COMPUTATION OF A(I).  THE LARGER LX THE LONGER
C                 WILL BE A(I).
C               MUST BE GREATER THAN 2, SHOULD BE GREATER THAN LR.
C
C     LR        LENGTH OF R(I).
C               MUST BE GREATER THAN 2.
C
C     R(I)      I=1...LR CONTAINS THE AUTOCORRELATION RR(J)
C                 J=0,1,....LR-1.
C               IF R(1) IS LESS THAN OR EQUAL TO ZERO, CONTROL IS
C                 RETURNED WITH NO COMPUTATION.
C
C     LGF       LENGTH OF G(I) AND F(I),  THE DESIRED LENGTH OF THE
C                 FILTER.
C               MUST BE GREATER THAN 0.
C
C     G(I)      I=1...LGF  CONTAINS THE CROSSCORRELATION GG(J)
C                 J=0,1,...LGF-1.  IF G(1) IS EQUIVALENT TO R(2), A
C                 SPECIAL LOOP IS USED TO EVALUATE F(I).
C
```

```
C     LA       THE MINIMUM LENGTH TO WHICH THE PED WILL BE EXTENTED.
C
C--------------------PROGRAM OUTPUTS
C OUTPUT ARGUMENTS
C
C     F(I)     I=1...LGF CONTAINS THE OPTIMUM LEAST-SQUARES FILTER.
C                IF G IS NOT EQUIVALENT TO R, F MAY BE EQUIVALENT TO G
C
C     LA       MAXIMUM LENGTH OF A(I) COMPUTED.
C
C     A(I)     I=1...LA CONTZAINS THE UNIT-LAG PREDICTION ERROR FILTER
C                AA(J) J=0,1,...,LA-1.
C
C
C-------------------------PROGRAM EVOCATION (LSPO)
C
C     LSPO IS EQUIVALENT TO WGC ROUTINE LSPOV07 EXCEPT THAT THE
C     AUTOCORRELATION IS NOT NORMALIZED.  HENCE THE G VECTOR NEED
C     NOT HAVE ANY SPECIAL STORAGE RELATIONSHIP WITH THE R VECTOR.
C
C SAMPLE CALL STATEMENT
C
C     CALL LSPO(R,G,J1,J2,A,F)
C
C-----------------------PROGRAM INPUTS
C INPUT ARGUMENTS
C
C     R(I)     I=1...J2+1 CONTAINS THE AUTOCORRELATION RR(J)
C                J=0,1,...,J2.  THE ZERO LAG VALUE MUST BE POSITIVE.
C
C     G(I)     I=1,...,J2-J1+1 CONTAINS THE CROSSCORRELATION GG(J)
C                J=0,1,...,J2-J1.  FOR A PREDICTION OPERATOR, G IS A
C                LATER PORTION OF R, SO G(1) AND R(J1+1) SHOULD BE
C                EQUIVALENCED.
C
C     J1       PREDICTION DISTANCE IN SAMPLES.
C
C     J2       MAXIMUM LAG OF RR(J) TO BE PREDICTED.
C
C     THE VALUE OF LX IS ARBITRARILY SET TO 2*(J2-J1+1) WHEN THE
C     LSPO ENTRY IS USED.
C
C
C OUTPUT ARGUMENTS
C
C     A(I)     I=1...LA WHERE LA IS LESS THAN OR EQUAL TO J2-J1+1
C                CONTAINS THE UMIT-LAG PREDICTION ERROR FILTER
C                AA(J) J=0,1,...,LA-1.
C
C     F(I)     I=1....J2-J1+1 CONTAINS THE OPTIMUM LEAST-SQUARES FILTER.
C
C
C***********************************************************************
C
C
C WIENER-LEVENSON LEAST-SQUARES ERROR FILTER
C EQUIVALENT G-F ALLOWED
C
      REAL R(1),G(1),F(1),A(1)
      LOGICAL PEO,EXTAA
C
      GO TO 1
C
C***********************************************************
C SPECIAL ENTRY TO MIMIC LSPO
C
      ENTRY LSPO
      LGF=J2-J1+1
```

```
       LR=LGF
      LX=2*LR
C
1     CONTINUE
C
C
C
C REDEFINE CERTAIN INPUT PARAMETERS
C
      IF(R(1).LE.0.) RETURN
      LF=LGF
C
C CHECK IF ONLY A PED IS NEEDED
C
      G1=G(1)
      G(1)=2.*R(1)
      PEO=R(2).EQ.G(1)
      G(1)=G1
      IF(.NOT.PEO) GO TO 20
C
C-----------------------------------------------------------------
C G(1) IS EQUIVALENT TO R(2)- ONLY COMPUTE THE PED
C
      LF=LF+1
      A(1)=1.
      ALPHA=R(1)
      ALX=LX-2
C
      DO 1130 LL=2,LF
      L1=LL-1
      IF(L1.LT.LR) LD=L1
C
      ALPH =0.
      DO 1110 IR=1,LD
1110  ALPH =ALPH+A(LL-IR)*R(IR+1)
C
      AL=-ALPH/ALPHA
      A(LL)=AL
      IA2=L1
      LL2=LL/2
      IF(LL2.LT.2) GO TO 1125
      DO 1120 IA1=2,LL2
      AT=A(IA2)
      A(IA2)=A(IA2)+AL*A(IA1)
      A(IA1)=A(IA1)+AL*AT
1120  IA2=IA2-1
C
1125  IF(IA2.GT.LL2) A(IA2)=(1.+AL)*A(IA2)
C
      ALPHO=ALPHA
      ALPHA=ALPHA+AL*ALPH
C
      EXTAA=ALPHO/ALPHA .GT. (ALX+1.)/ALX
      IF(EXTAA) LA=LL
1130  CONTINUE
C
      DO 1140 IF=2,LF
1140  F(IF-1)=-A(IF)
      RETURN
C
20    CONTINUE
C
C
C-----------------------------------------------------------------
C GENERAL FILTER COMPUTATION - SET UP TWO LENGTH OPERATORS
C
      A(1)=1.
      F(1)=G(1)/R(1)
```

```
      ALPHA=R(1)
      LAMN=LA
      LA=1
      IF(LF.LE.1) RETURN
C
C-------------------------------------------------------------------
C LOOP OVER RECURSIONS
C
      EXTAA= .TRUE.
      ALX=LX-2
C
      DO 200 LL=2,LF
      L1=LL-1
      IF(L1.LT.LR) LD=L1
C
      IF(.NOT.EXTAA) GO TO 140
C
C-----------------------------------------------
C EXTEND THE PED AND FILTER
C
      LA=L1
      ALPH=0.
      BETA=G(LL)
      DO 110 IR=1,LD
      ALPH=ALPH + A(LL-IR)*R(IR+1)
110   BETA=BETA-F(LL-IR)*R(IR+1)
C
      AL=-ALPH/ALPHA
      ALPHO=ALPHA
      ALPHA=ALPHA+AL*ALPH
C
      A(LL)=AL
      FL=BETA/ALPHA
      F(LL)=FL
      F(1)=F(1)+FL*AL
      IA2=L1
      LL2=LL/2
      IF(LL2.LT. 2) GO TO 135
      DO 130 IA1=2,LL2
      AT=A(IA2)
      A(IA2)=A(IA2)+AL*A(IA1)
      A(IA1)=A(IA1)+AL*AT
      F(IA2)=F(IA2) +FL*A(IA1)
      F(IA1)=F(IA1) +FL*A(IA2)
130   IA2=IA2-1
C
135   IF(IA2.LE.LL2) GO TO 138
      A(IA2)=(1.+AL)*A(IA2)
      F(IA2)=F(IA2)+FL*A(IA2)
C
138   IF(LL.LE.LAMN) GO TO 200
      EXTAA=ALPHO/ALPHA .GT. (ALX+1.)/ALX
      GO TO 200
C
C-----------------------------------------------
C EXTEND THE FILTER
C
140    BETA=G(LL)
      DO 150 IR=1,LD
150   BETA=BETA-F(LL-IR)*R(IR+1)
C
      FL=BETA/ALPHA
      F(LL)=FL
      DO 160 IF=1,LA
160    F(LL-IF)=F(LL-IF)+FL*A(IF+1)
C
C END OF RECURSION
```

```
200   ALX=ALX-1.
C
      LA=LA+1
      RETURN
      END
      SUBROUTINE ADECON(S,IOPT,LS,A,LF,IPD)
C*************************************************************************
C
C     SUBROUTINE ADECON    ADAPTIVE DECONVOLUTION
C
C     SUBROUTINE ADECON IMPLEMENTS THE NOISY GRADIENT DESCENT ALGORITHM
C     AS SOLUTION TO THE WIENER MINIMUM MEAN SQUARE ERROR DECONVOLUTION
C     FILTER.
C
C     PARAMETERS PASSED AS SUBROUTINE ARGUMENTS ARE:
C       S IS THE INPUT TIME SERIES AND ALSO SERVES AS THE OUTPUT FOR
C       THE DECONVOLVED TIME SERIES.
C
C     IOPT IS AN OPTION FOR THE CHOICE OF WHICH DECONVOLVED OUTPUT IS
C       RETURNED IN S.
C       IF IOPT .EQ. 1 THE REVERSE FILTERED SERIES IS RETURNED.
C       IF IOPT .EQ. 2 THE FORWARD FILTERED SERIES IS RETURNED.
C       IF IOPT .EQ. 3 THE STACK OF FORWARD AND REVERSE FILTERING.
C       IS RETURNED.
C
C       LS IS THE NUMBER OF POINTS IN THE INPUT TIME SERIES.
C
C       A IS THE ADAPTION COEFFICIENT. THE USEFUL RANGE IS FROM 0.01 TO
C       0.4 AND REPRESENTS FRACTIONAL INCREASE IN NOISE ENERGY.
C
C       LF IS THE LENGTH OF FILTER
C
C       IPD IS THE PREDICTION DISTANCE WITH UNITS OF NUMBER OF POINTS.
C       IT IS NORMALLY ONE FOR SPECTRAL WHITENING AND IS APPROXIMATELY
C       THE MULTIPLE DISTANCE FOR PREDICTIVE DECONVOLUTION.
C
C     OTHER SIGNIFICANT VARIABLES USED IN ADECON ARE LISTED IN ORDER
C     OF APPEARANCE.
C
C       FIL(I) IS THE SET OF FILTER COEFFICIENTS GENERATED BY THE
C       ALGORITHM.
C
C       DS(I,J) IS THE STORAGE FOR THE DECONVOLED TRACE:
C       J.EQ.1 IS THE REVERSE FILTERED TRACE.
C       J.EQ.2 IS THE FORWARD FILTERED TRACE.
C
C       SIGMA IS THE ESTIMATE OF THE AVERAGE SIGNAL ENERGY LEVEL.
C
C       B IS THE FILTER COEFFICIENT FOR UPDATE OF SIGMA.
C
C       U IS AN ADAPTION COEFFICIENT DEPENDENT ON LF,SIGMA AND A.
C
C       SFI MAY BE USED TO SCALE THE TRACE TO A MAXIMUM OF ONE.
C
C*************************************************************************
      DIMENSION S(1100),DS(1100,2),FIL(50)
      NPTS=LS
C
C     INITIALIZE FILTER AND WORKSPACE ARRAYS.
C
      DO 101 I=1,LF
101   FIL(I)=(-1.)**I
      DO 102 J=1,2
      DO 102 I=1,LS
102   DS(I,J)=0.
      N2=LF+IPD
C
C     CALCULATE INITIAL SIGNAL ENERGY LEVEL AND FILTER COEFFICIENTS.
```

```
C
      SIGMA=0.
      DO 103 I=1,LF
       MR=NPTS-N2+I
103   SIGMA=SIGMA+S(MR)*S(MR)
      IF(SIGMA.LT.0.000001) SIGMA=0.001
C
      B=EXP(-1./FLOAT(LF))
C
C     BEGIN LOOP FOR TIME REVERSE FILTERING.
C
      DO 106 I=N2,NPTS
      M=NPTS-I+N2
      ESUM=0.
      N1=M-N2
C
C     CALCULATE PREDICTED SIGNAL VALUE.
C
      DO 104 J=1,LF
      N=N1+J
104   ESUM=ESUM+FIL(J)*S(N)
C
C     CALCULATE ACTUAL MINUS PREDICTED FOR ERROR VALUE.
C
      DS(M,1)=S(M)-ESUM
C
C     CALCULATE ADAPTION COEFFICIENT AND UPDATE INPUT ENERGY ESTIMATE.
C
      U=A/(FLOAT(LF)*SIGMA)
      IF(N1.LT.1) GO TO 105
      IF(ABS(S(N1)).LT.0.00001) GO TO 105
C
      SIGMA=(1.-B)*S(N1)*S(N1)+B*SIGMA
105   CONTINUE
C
C     UPDATE FILTER COEFFICIENTS.
C
      DO 106 J=1,LF
      N=N1+J
      FIL(J)=FIL(J)+U*DS(M,1)*S(N)
106   CONTINUE
C
C     BEGIN LOOP FOR TIME FORWARD FILTERING.
C
      SFI=0.
      MT=NPTS-N2+1
      DO 109 I=1,MT
      M=I+N2-1
      ESUM=0.
C
C     CALCULATE PREDICTED SIGNAL VALUE.
C
      DO 107 J=1,LF
      N=I+J-1
107   ESUM=ESUM+FIL(J)*S(N)
C
C     CALCULATE ACTUAL MINUS PREDICTED FOR ERROR VALUE.
C
      DS(M,2)=S(M)-ESUM
      IF(SFI .LT. ABS(DS(M,2))) SFI=ABS(DS(M,2))
      ES=S(M)-ESUM
C
C     CALCULATE ADAPTION COEFFICIENT AND UPDATE INPUT ENERGY ESTIMATE.
C
      U=A/(FLOAT(LF)*SIGMA)
      IF(ABS(S(M)).LT.0.0001) GO TO 108
C
      SIGMA=(1.-B)*S(M)*S(M)+B*SIGMA
```

```
108   CONTINUE
C
C     UPDATE FILTER COEFFICIENTS.
C
      DO 109 J=1,LF
      N=I+J-1
      FIL(J)=FIL(J)+U*ES*S(N)
109   CONTINUE
      SFI=1./SFI
C
C     ROUTE DESIRED TRACE FOR OUTPUT.
C
      IF(IOPT.EQ.3) GO TO 5
      DO 110 I=1,NPTS
110   S(I)=DS(I,IOPT)
      RETURN
C
C     STACK DECONVOLVED VALUES.
C
5     DO 111 I=1,NPTS
111   S(I)=0.5*(DS(I,1)+DS(I,2))
      RETURN
      END
C
C
      SUBROUTINE BAFL(LOUT,CX,IFLGAP,LCN,NWARM,ISTRT,ZTAU,XTAU,NZERO,X)
C               -----------------------------
C------------S U B R O U T I N E  B A F L--------------------
C               -----------------------------
C
C     THE BURG ADAPTIVE FILTER: AN ADAPTVIE OR TIME-VARYING
C     FIXED-LEAD PREDICTION ERROR PROCESSOR. ADJUSTMENT OF EACH
C     REFLECTION COEFFICIENT IS MADE EVERY JUMP STATE ATTEMPTING
C     TO MINIMIZE THE STAGE OUTPUT POWER.
C     INPUTS:
C     X(1)...X(LX)=INITIAL DATA
C     LCN= LAST NON-ZERO REFLECTION COEFFICIENT
C     IFLGAP= NUMBER OF GAPS BETWEEN FILTER COEFFICIENTS
C        SETTING IFLGAP=0 DOES NOT GAP THE F.C. AND
C        TRIES TO OPERATE ON THE ENTIRE SPECTRUM FROM
C         0 TO W WHERE IS THE FOLDING FREQUENCY.
C        SETTING IFLGAP=1 OPERATES ON THE PORTION OF
C        THE SPECTRUM FROM 0 TO W/2 , IFLGAP=2 FROM
C        0 TO W/3 FTC. THUS ALLOWING SPECIFICATION OF
C        WHAT PART OF THE SPECTRUM TO DECONVOLVE.
C
C        NWARN=DURATION OF STATIONARY C ESTIMATION CYCLE
C     ISTRT=START OF STATIONARY GATE
C     ZTAU=TEMPORAL RELAXATION TIME TO 1/E
C     XTAU=SPATIAL RELAXATION DISTANCE TO 1/E
C      OUTPUTS:
C       X(1)...X(LOUT)=FORWARD ERROR PREDICTION TRACE
C     OTHER VARIABLES:
C     F(1)...F(LCN)=FORWARD STATE VECTOR
C     B(1)...B(LCN)=BACKWARD STATE VECTOR
C     C(1)...C(LCN)=REFLECTOR COEFFICIENTS AT EACH STATE
C     CX=REFLECTION COEFF. INTEGRATED IN SPACE AND TIME
C     DEN(1)...DEN(LCN)= STAGE AUTOPOWER
C     NUM(1)...NUM(LCN)= STAGE CROSSPOWER
C
C     CALLING ≠BAFL≠ FIRST SETS UP THE LOOPING AND PASSING
C     ARRAYS FOT THE PARTICULAR PROBLEM AS SPECIFIED BY LCN AND
C     IFLGAP. THEN IT COMPUTES A SHORT (LENGTH=NWARM) ESTIMATE
C     OF THE REFLECTION COEFFICIENT SERIES IN ORDER TO START THE
C     ADAPTION OUT WITH SOME REASONABLE NUMBERS. THEN IT LOADS UP
C     THE CX ARRAY WITH THE INITIAL VALUES AND PASSED INTO ENTRY
```

```
C      ≠BAFLGO≠.
C
C      THE USAL ENTRY IS ≠BAFLGO≠ WHICH FIRST INITIALIZES THE
C      BACKWARD ARRAY THEN PASSES TO THE MAIN ALGORITHM.
C      THE CX SERIES IS UPDATED EVERY IFLGAP DATA POINTS AND IN
C      THE INTERMEDIATE STEPS THE OUTPUT ARE INTERPOLATED OR
C      PROCESSED AS THOUGH THE R.C. WERE STATIONARY.
C
C
C
       DIMENSION NUM(50),DEN(50),B(50),F(50),C(50),EM(1500),EP(1500),
      1CX(LCN,LOUT),A(50),X(530)
       REAL NUM
       DO 90 I=1,50
       A(I)=0.0
       B(I)=0.0
       C(I)=0.0
90     CONTINUE
       FTEST=1.00
       LCNP1=LCN+1
       LCNP2=LCN+2
       IFGM1=IFLGAP-1
       LBSP=LOUT-IFLGAP
       NZERP1=NZERO+1
       NEND=LCN-NZERP1
       DO 80 K=LBSP,LOUT
   80 EP(K)=0.
C      BEGIN STATIONARY WARM-UP
       DO 10 I=1,NWARM
       EM(I)=X(I*IFLGAP+ISTRT)
   10 EP(I)=X(I*IFLGAP+ISTRT)
C
       DO 11 J=2,LCNP1
       DEN(J)=0.
       NUM(J)=0.
       DO 12 I=J,NWARM
       DEN(J)=DEN(J)+EP(I)*EP(I)+EM(I-J+1)*EM(I-J+1)
   12 NUM(J)=NUM(J)+EP(I)*EM(I-J+1)
       C(J)=-2.*NUM(J)/DEN(J)
       DO 11 I=J,NWARM
       EPI=EP(I)
       EP(I)=EPI+C(J)*EM(I-J+1)
       KS=IFLGAP*LCN+1
   11 EM(I-J+1)=EM(I-J+1)+C(J)*EPI
       DO 8 J=1,LCN
       DO 8 K=1,KS
    8 CX(J,K)=C(J+1)
       DO 33 J=1,LCN
       DEN(J)=DEN(J+1)
   33 NUM(J)=NUM(J+1)
C      END WARM-UP CYCLE
C      SET RELAXATION TIMES
C      DLX=EXP(-1./XTAU)
       DLX=0.
       DL=EXP(-1./ZTAU)
       DRX=1.-DLX
       DR=1.-DL
C
C----------------USUAL--ENTRY-------------------------------------------
       ENTRY BAFLGO
C
C         INITIALIZE BACKWARD VECTOR
35     B(1)=X(KS-IFLGAP)
       A(1)=1.0
       DO 1000 J=2,LCN
       B(J)=X(KS-J*IFLGAP)
       A(J)=CX(J,KS)
       DO 1000 I=2,J
```

```
      A(I)=A(I)+CX(J,KS)*A(J-I+1)
 1000 B(J)=B(J)+A(I)*X(KS+(I-J-1)*IFLGAP)
C
C---------BEGIN--MAIN--LOOP-------------------------------------------------
C
      DO 5000 K=KS,LOUT,IFLGAP
      Z=X(K)
      DO 1010 J=1,NZERP1
 1010 F(J)=Z
      DO 2000 J=NZERP1,LCN
      DEN(J)=(F(J)**2+B(J)**2)*DR+DEN(J)*DL
      NUM(J)=F(J)*B(J)*DR+NUM(J)*DL
      IF(FTEST.LE.1.1) CX(J,K)=-2.*NUM(J)/DEN(J)
      CX(J,K)=-2.*DRX*NUM(J)/DEN(J)+CX(J,K)*DLX
 2000 F(J+1)=F(J)+CX(J,K)*B(J)
      X(K)=F(LCNP1)
      DO 3000 JR=1,NEND
      J=LCN-JR
 3000 B(J+1)=B(J)+CX(J,K)*F(J)
      IF(NZERO.EQ.0) GO TO 5000
      DO 4000 JR=1,NZERO
      J=NZERP1-JR
 4000 B(J+1)=B(J)
5000  B(1)=Z
C
C---------END--OF--MAIN--LOOP-----------------------------------------------
C
C     NCW GO BACK AND FILL IN THE GAPS...
      IF(IFLGAP.EQ.1) GO TO 9050
      DO 9000 L=1,IFGM1
      KSTART=KS+L
      DO 9000 K=KSTART,LOUT,IFLGAP
      KC=K-L
      Z=X(K)
      DO 6000 J=1,NZERP1
 6000 F(J)=Z
      DO 7000 J=NZERP1,LCN
 7000 F(J+1)=F(J)+CX(J,KC)*B(J)
      X(K)=F(LCNP1)
      DO 8000 JR=1,NEND
      J=LCN-JR
 8000 B(J+1)=B(J)+CX(J,KC)*F(J)
      IF(NZERO.EQ.0) GO TO 9000
      DO 8050 JR=1,NZERO
      J=NZERP1-JR
 8050 B(J+1)=B(J)
 9000 B(1)=Z
C
 9050 FTEST=FTEST+1.0
      RETURN
      END
      SUBROUTINE SECTION(IN)
C*************************************************************************
C
C     SUBROUTINE SCTION PLOTS A RECORD
C     SECTION CONSISTING OF NSEIS SEISMOGRAMS
C     AND PLOTS A DISTANCE AND TIME SCALE
C     SURROUNDING THE SEISMOGRAMS. THE RECORD
C     SECTION MAY USE NORMAL TIME SCALE OR
C     A REDUCED VELOCITY TIME SCALE (T-X/REDUCING
C     VELOCITY). DISTANCE ALWAYS INCREASES TO
C     THE RIGHT. TIME MAY BE SET TO INCREASE
C     UPWARDS OR DOWNWARDS. THE INPUT DATA
C     AND SEISMOGRAMS MUST BE WRITTEN
C     ON FILE IN (SPECIFIED IN THE ARGUMENT
C     OF THE CALL STATEMENT) AND IN THE
C     FORMATS SPECIFIED BELOW IF SCTION IS TO
C     BE CALLED FROM ANOTHER PROGRAM. FOR
```

```
C     PLOTTING A SECTION FROM SEISMOGRAMS ON
C     CARDS, IN MAY BE THE INPUT FILE AND
C     THE SUBROUTINE WILL READ THE DATA FROM CARDS.
C
C**********************************************************************
C
C
C     INPUTS ARE
C
C     1) TITLE(I), I=1,5 (5A10) 50 CHARACTER TITLE
C
C     2) X1,X2,XINCH,NX  (3F10.3,I5)
C
C        X1         BEGINNING DISTANCE
C        X2         ENDING DISTANCE
C        XINCH      LENGTH OF DISTANCE AXIS OF PLOT IN INCHES
C        NX         NUMBER OF INTERVALS ON X AXIS
C
C     3) T1,T2,TINCH,NT (3F10.3,I5)
C
C        SAME AS 2) EXCEPT FOR TIME AXIS
C
C     4) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL (9I5)
C
C        NSEIS      NUMBER OF SEISMOGRAMS
C        NDIST=1    SEISMOGRAMS EQUALLY SPACED BEGINNING WITH
C                   FDIST AND SPACING DELX
C        NDIST=0    DISTANCES READ IN SEPARATELY
C        NFRMT=0    SEISMOGRAMS IN (8F10.3) FORMAT
C        NFRMT=1    SEISMOGRAMS IN (16F5.2) FORMAT AND SCALED
C                   FROM 0. TO 9.99 SO THAT MEAN VALUE (5.0) IS
C                   REMOVED BEFORE PLOTTING
C        NAMP=0     ALL SEISMOGRAMS SCALED TO AMP INCHES MAXIMUM
C                   ZERO TO PEAK AMPLITUDE FOR PLOTTING
C        NAMP=1     ALL SEISMOGRAMS PLOTTED AT REAL AMPLITUDE*AMP
C                   *DISTANCE**ALPHA
C        NAMP=2     ALL SEISMOGRAMS PLOTTED AT REAL AMPLITUDE * AMP
C        NEXP=0     NO EXPONENTIAL TIME VARIABLE AMPLITUDE SCALING
C        NEXP=1     AMPLITUDES SCALED BY MULTIPLYING BY EXP(FACTOR
C                   *(TIME-T0))
C        NREV=0     TIME INCREASES UPWARD
C        NREV=1     TIME INCREASES DOWNWARD
C        NPOS=0     AMPLITUDES POSITIVE TO THE RIGHT
C        NPOS=1     AMPLITUDES POSITIVE TO THE LEFT
C        NSCALE=0   NO LINEAR TIME VARIABLE AMPLITUDE SCALING
C        NSCALE=1   AMPLITUDES SCALED BY MULTIPLYING BY
C                   FACT=(FACTOR*(TIME-T0)+1.)
C                   IF(FACT.LT.1.0) FACT=1.0
C        NFILL=0    NO SHADING OF POSITIVE CYCLES
C        NFILL=1    POSITIVE PORTION OF WAVELETS SHADED
C        NFILL=2    NEGATIVE PORTION OF WAVELETS SHADED
C
C     5) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA  (7F10.3)
C
C        REDVEL     REDUCING VELOCITY
C                   IF (REDVEL.EQ.0) NO REDUCING VELOCITY USED
C        T0         MINIMUM TIME FOR TIME VARIABLE AMPLITUDE SCALING
C                   T0 IS REFERENCED TO REDUCED TIME IF USED.
C                   (SEE 4) NEXP AND NSCALE)
C        FACTOR     AMPLITUDE SCALING FACTOR FOR TIME VARIABLE AMPLITUDE
C                   (SEE 4) NEXP AND NSCALE)
C        AMP        SCALE FACTOR FOR PLOTTING AMPLITUDES
C                   (SEE 4) NAMP)
C        FDIST      IF(NDIST.EQ.1) FIRST DISTANCE
C        DELX       IF(NDIST.EQ.1) DISTANCE INCREMENT
C        ALPHA      EXPONENT IN SCALING FACTOR FOR PARTIALLY REMOVING
C                   GEOMETRICAL SPREADING BY MULTIPLYING AMPLITUDES BY
C                   DISTANCE**ALPHA  (SEE 4) NAMP)
```

```
C
C *** INPUTS 6) AND 7) ARE REPEATED NSEIS TIMES
C
C     6) NPTS,DIST,ST,DELTAT,SF  (I5,3F10.3,F20.10)
C
C        NPTS      NUMBER OF POINTS IN SEISMOGRAM
C        DIST      DISTANCE
C        ST        STARTING TIME OF RECORD WITH RESPECT TO ORIGIN TIME
C        DELTAT    SAMPLING INTERVAL (SEC)
C        SF        SCALE FACTOR TO RECOVER REAL AMPLITUDES FROM
C                  FOLLOWING SEISMOGRAM SEQUENCE BY MULTIPLYING
C                  S(J)=S(J)*SF
C
C     7) S(J),J=1,NPTS  (8F10.3) OR (16F5.2) IF SCALED BETWEEN 0. AND 9.
C
C                  (SEE 4) NFRMT)
C        S(J)      SEISMOGRAM
C
C***********************************************************************
C
C
C
C
C *** OUTPUTS ARE PRINTOUTS OF INPUT DATA FOR CONFIRMATION AND PLOT
C
C***********************************************************************
      DIMENSION S(520),TITLE(5),TIT(6)
      READ (IN,126) (TITLE(I),I=1,5)
C     WRITE (2,127) (TITLE(I),I=1,5)
C     READ(IN,61) (TIT(I),I=1,6)
61    FORMAT(6A10)
C     CALL SYMBOL(1.0,0.0,.12,TIT,0.0,60)
      READ (IN,120) X1,X2,XINCH,NX
C     PRINT 130
C     WRITE (2,120) X1,X2,XINCH,NX
      READ (IN,120) T1,T2,TINCH,NT
C     PRINT 131
C     WRITE (2,120) T1,T2,TINCH,NT
      READ (IN,121) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
C     PRINT 132
C     WRITE (2,136) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      READ (IN,122) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA
C     PRINT 133
C     WRITE (2,122) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA
      IF(NFILL.EQ.1) GO TO 5
      IF(NFILL.EQ.2) GO TO 5
      NFLAG=NFILL
      NFILL=0
    5 CONTINUE
      SX=(X2-X1)/XINCH
      SZ=(T2-T1)/TINCH
      NX1=NX+1
      NT1=NT+1
      DX=XINCH/FLOAT(NX)
      DT=TINCH/FLOAT(NT)
      CALL SYMBOL (0.,.5,.10,TITLE,0.0,50)
      CALL PLOT (1.,1.,-3)
      DO 101 I=1,NX1
         X=(I-1)*DX
         CALL PLOT (X,0.,2)
         CALL PLOT (X,.1,2)
  101 CALL PLOT (X,0.,2)
      DO 102 I=1,NT1
         T=(I-1)*DT
         CALL PLOT (XINCH,T,2)
         CALL PLOT (XINCH-.1,T,2)
  102 CALL PLOT (XINCH,T,2)
      DO 103 I=1,NX1
```

```
          X=XINCH-(I-1)*DX
          CALL PLOT (X,TINCH,2)
          CALL PLOT (X,TINCH-.1,2)
  103 CALL PLOT (X,TINCH,2)
      DO 104 I=1,NT1
          T=TINCH-(I-1)*DT
          CALL PLOT (0.,T,2)
          CALL PLOT (.1,T,2)
  104 CALL PLOT (0.,T,2)
      DO 91 I=1,NSEIS
      DXX=DX*I-0.041
      IX=7
      IF(I .GT. 9) GO TO 1
      CALL NUMBER(DXX,8.2,.15,IX,0.0,2HI1)
      GO TO 91
1     CALL NUMBER(DXX,8.2,.15,IX,0.0,2HI2)
 91    CONTINUE
      CALL SYMBOL(.29,8.5,0.15,7HSTATION,0.0,7)
      DO 95 I=1,NT1
      DTT=    DT*(NT1-I)-0.075
      FY=(I-1)*0.1
      CALL  NUMBER(-0.5,DTT,.15,FY,0.0,4HF3.1)
95     CONTINUE
      CALL SYMBOL(-0.8,3.5,0.15,12HTIME(SECOND),90.,12)
      NR=1
      IF (NREV.EQ.1) CALL PLOT (0.,TINCH,-3)
      IF (NREV.EQ.1) NR=-1
      NP=1
      IF (NPOS.EQ.1) NP=-1
      DO 119 I=1,NSEIS
      IF(NFLAG.EQ.3.AND.I.GT.5) NFILL=1
      IF(NFLAG.EQ.4.AND.I.GT.5) NFILL=2
         READ (IN,123) NPTS,DIST,ST,DELTAT,SF
C     WRITE (2,137)
C        PRINT 134
         IF (NDIST.EQ.1) DIST=FDIST+(I-1)*DELX
C        WRITE (2,123) NPTS,DIST,ST,DELTAT,SF
         IF (NFRMT.EQ.1) GO TO 105
         READ (IN,124) (S(J),J=1,NPTS)
C        PRINT 135
C     WRITE(2,129) (S(J),J=1,NPTS)
         GO TO 107
  105    READ (IN,125) (S(J),J=1,NPTS)
C         PRINT 135
C     WRITE(2,129) (S(J),J=1,NPTS)
         DO 106 J=1,NPTS
  106    S(J)=(S(J)-5.001)/4.99
  107    DO 108 J=1,NPTS
  108    S(J)=S(J)*SF
         XD=(DIST-X1)/SX
         CALL PLOT (XD,0.,3)
         T=ST
         IF (REDVEL.LT..001) GO TO 109
         T=ST-DIST/REDVEL
  109    CONTINUE
         XMAX=1.0
         IF (NAMP.NE.0) GO TO 111
         XMAX=0.0
         DO 110 J=1,NPTS
  110    IF (ABS(S(J)).GT.XMAX) XMAX=ABS(S(J))
  111    IF (NAMP.NE.1) GO TO 113
         SC=DIST**ALPHA
         DO 112 J=1,NPTS
  112    S(J)=S(J)*SC
  113    CONTINUE
         DO 116 J=1,NPTS
            S(J)=(S(J)*AMP/XMAX)*NP
            TIME=T+(J-1)*DELTAT-T1
```

```
            IF (NEXP.EQ.0) GO TO 114
            FACT=EXP(FACTOR*(TIME-T0))
            GO TO 115
  114       IF (NSCALE.EQ.0) GO TO 116
            FACT=(FACTOR*(TIME-T0))+1.
  115       IF (TIME.LT.T0) FACT=1.0
            S(J)=S(J)*FACT
  116     CONTINUE
          XP=NP
      Y=(T-T1)/SZ
      IF(Y.LT.0.0) GO TO 219
      X=XD
      CALL PLOT(X,Y,2)
219       DO 118 J=1,NPTS
       Y=((T+(J-1)*DELTAT-T1)/SZ)
            X=XD+S(J)
      IF(X.GT.XINCH.OR.X.LT.-2.)  GO TO 118
            IF (Y.GT.TINCH.OR.Y.LT.0.0) GO TO 118
            Y=Y*NR
            CALL PLOT (X,Y,2)
            IF (NFILL.EQ.0) GO TO 118
            IF (NFILL.EQ.1) GO TO 117
            IF (XP*S(J).LT.0.) CALL PLOT (X-S(J),Y,2)
            CALL PLOT (X,Y,2)
            GO TO 118
  117       IF (XP*S(J).GT.0.) CALL PLOT (X-S(J),Y,2)
            CALL PLOT (X,Y,2)
  118     CONTINUE
  119 CONTINUE
      T=0.
      IF (NREV.EQ.1) T=-TINCH
      CALL PLOT (XINCH+1.,T-1.,-3)
      RETURN
C
  120 FORMAT (3F10.3,I5)
  121 FORMAT (9I5)
  122 FORMAT (7F10.3)
  123 FORMAT (I5,3F10.4,F20.10)
  124 FORMAT (8F10.3)
  125 FORMAT(16F5.3)
  126 FORMAT (5A10)
  127 FORMAT (10X,5A10)
  129 FORMAT(1X,16F6.3)
  130 FORMAT (/,1X,35H    X1         X2       XINCH      NX )
  131 FORMAT (/,1X,34H    T1         T2       TINCH      NT )
  132 FORMAT (/,1X,51H NSEIS NDIST NFRMT NAMP NEXP NREV NPOS NSCALE NFIL
     1L)
  133 FORMAT (/,1X,69H  REDVEL       TO       FACTOR      AMP       FDIST
     1  DELX       ALPHA   )
  134 FORMAT      (1X,48H NPTS   DIST        ST       DELTAT              SF
     1)
  135 FORMAT (/,1X,73H       SEISMOGRHM VALUES ARE LISTED FROM LEFT TO R
     1IGHT IN SUCCEEDING ROWS,/)
  136 FORMAT (1X,I5,I7,I6,4I5,2I6)
  137 FORMAT(1H1,///////)
      END
      SUBROUTINE SYNTH(C2,TK,THICK,VEL,TZ,RCOEF ,DENS,Z,N,R,SLNGTH,
     *    DELTAT,NEXT,S,C,NPTS,NP,NOL)
C
C GENERATE THE SYNTHETIC SEISMOGRAM
C
      DIMENSION C2(530),TK(50),THICK(50),TZ(530),RCOEF(50),DENS(50),
     *     Z(530),R(530),S(2000),C(530),VEL(530)
      INTEGER U,V
      DO 39 J=1,N
         C2(J)=0.0
   39 CONTINUE
      DO 7 I=1,NOL
```

```
C     COMPUTE TWO-WAY TRAVEL TIME
      TK(I)=2.0*(THICK(I)/VEL(I))
      IF(I.EQ.1) GO TO 9
      TZ(I)=TZ(I-1)+TK(I)
      GO TO 7
    9 TZ(I)=TK(I)
    7 CONTINUE
C     COMPUTE THE REFLECTION COEFFICIENTS
      DO 20 I=1,NOL
      V=I
      IF(I.EQ.NOL) GO TO 20
      U=I+1
      RCOEF(I)=((VEL(U)*DENS(U))-(VEL(V)*DENS(V)))/((VEL(U)*DENS(U))+
     *   (VEL(V)*DENS(V)))
   20 CONTINUE
C     SET UP TIME SERIES OF RECOEF(S)
      DO 40 I=1,NOL
      Z(I)=TZ(I)/DELTAT
      K=Z(I)
      C2(K)=RCOEF(I)
   40 CONTINUE
      IOPT=1
      CALL WIFLE(IOPT,N,C2,R,Z)
   42 NPTS=SLNGTH/DELTAT+1.
      R(1)=0.
   65 CALL SIGNAL(SLNGTH,DELTAT, NEXT, 1.,NPTS,0,0.0  ,S)
  107        NP=N+NPTS -1
      CALL FOLD(N,R,NPTS,S,NP,C)
      RETURN
      END
      SUBROUTINE WIFLE (IOPT,N,C,R,A)
C LET C(1).,C(2)...C(N)=REFLECTION COEFFICIENTS AT SUCCESIVE DEPTHS
C     A(1)=1.,A(2)...A(N)=INVERSE WAVELETOF THE TRANSMISION SEISMOGRAM
C     R(1)=1.,R(2)...R(N)=SURFACE INPULSE AND ITS REFLECTION SEISMOGRAM
C
C FOR  ; 1.   R(2) R(N); ; 1. ; ;V(N);                J
C N=3  ;R(2)  1.    R(I); YA(2);=; 0.;      V(J)=PROD (1-C(I)**2)
C      ;R(N) R(2)  1. Y ;A(N); ; 0.;,                I=I=2
C
C     IF(IOPT=1)   C=INPUT  A,R=OUTPUT
C     IF(IOPT=2)   R=INPUT  A,C=OUTPUT
C     IF(IOPT=3)   A=INPUT   C=OUTPUT
C TAKEN FROM GEOPHYSICS,VOL 33,NO.2,APRIL1968  BY J.F. CLAERBOUT
      DIMENSION C(N),A(N),R(N)
      A(1)=1.
      C(1)=-1.
      R(1)=1.
      V=1.
      GO TO (100,200,300),IOPT
  100 DO 120 J=2,N
      A(J)=0.
      R(J)=C(J)*V
      V=V*(1.-C(J)*C(J))
      DO 110 I=2,J
  110 R(J)=R(J)-A(I)*R(J-I+1)
      JH=(J+1)/2
      DO 120 I=1,JH
      BOT=A(J-I+1)-C(J)*A(I)
      A(I)=A(I)-C(J)*A(J-I+1)
  120 A(J-I+1)=BOT
      RETURN
  200 DO 220 J=2,N
      A(J)=0.
      E=0.
      DO 210 I=2,J
  210 E=E+R(I)*A(J-I+1)
      C(J)=E/V
      V=V-E*C(J)
```

```
      JH=(J+1)/2
      DO 220 I=1,JH
      BOT=A(J-I+1)-C(J)*A(I)
      A(I)=A(I)-C(J)*A(J-I+1)
  220 A(J-I+1)=BOT
      RETURN
  300 DO 310 I=1,N
  310 C(I)=A(I)
      DO 330 K=2,N
      J=N-K+2
      AL=1./(1.-C(J)*C(J))
      BE=C(J)*AL
      JH=(J+1)/2
      DO 320 I=1,JH
      TOP=AL*C(I)-BE*C(J-I+1)
      C(J-I+1)=AL*C(J-I+1)-BE*C(I)
  320 C(I)=TOP
  330 C(J)=-BE/AL
      RETURN
      END
      SUBROUTINE SIGNAL(T,DT,N,V1,NPTS,NA,THALF,S)
C     T=LENGTH OF WAVELET IN SEC
C     DT=SAMPLING INTERVAL IN SEC
C     N=NUMBER OF EXTREMA IN WAVELET
C        IF(N.LT.0) WAVELET IS MULTIPLIED BY -1.0
C     V1=VELOCITY OF UPPER LAYER OR 1.0
C     NPTS=NUMBER OF POINTS IN WAVELET TIME SERIES
C     NA=NUMBER OF ZEROES BEFORE WAVELET
C     THALF=IF(THALF.GT.0.0)  THE WAVELET IS WEIGHTED BY AN EXPONENTIAL
C        SUCH THAT AT THALF SECONDS THE AMPLITUDE IS MULTIPLIED BY 0.5
C        THALF USUALLY WILL BE EQAL TO APPROXIMATELY T OR T/2
C     S=SIGNAL
      DIMENSION S(500)
      IF(V1.LT..0001) V1=1.0
      FACTOR=1.0
      IF(N.LT.0) FACTOR=-1.0
      N=IABS(N)
      DO 100 I=1,NA
100   S(I)=0.0
      NS=T/DT+1.0
      FN=N
      FM=(FN+2.)/FN
      D=FN*3.14159265/T
      TT=-DT
      A=0.0
      DO 200 I=1,NS
      TT=TT+DT
      T1=D*TT
      T2=FM*T1
      C=    SIN(T1)-SIN(T2)/FM
      S(NA+I)=C
      IF(C.GT.A)A=C
200   CONTINUE
      A=V1/A
      DO 300 I=1,NS
      J=NA+I
300   S(J)=A*S(J)
      SE=0.0
      J=NA+NS+1
      DO 400 I=J,NPTS
400   S(I)=SE
      IF(THALF.LT..0001) GO TO 1000
      N2=NA+1
      DO 500 I=N2,NPTS
      TIME=(I-N2)*DT
      X=TIME*.696/THALF
500   S(I)=S(I)*EXP(-X)
1000  CONTINUE
```

```
      SMAX=0.0
      DO 600 I=1,NPTS
      IF(SMAX.LT.ABS(S(I))) SMAX=ABS(S(I))
600   CONTINUE
      DO 700 I=1,NPTS
700   S(I)=S(I)*FACTOR/SMAX
      RETURN
      END
      SUBROUTINE FOLD(LA,A,LB,B,LC,C)
C*********************************************************************C
C                                                                     C
C     SUBROUTINE FOLD CONVOLVES TWO SERIES                            C
C                                                                     C
C     A,B - SERIES TO BE CONVOLVED                                    C
C                                                                     C
C     LA,LB - LENGTH OF A AND B                                       C
C     C - CONVOLUTION SERIES                                          C
C                                                                     C
C     LC - LENGTH OF C                                                C
C                                                                     C
C*********************************************************************C
      DIMENSION A(LA),B(LB),C(LC)
      LC=LA+LB-1
      CALL ZERO(LC,C)
      DO 1 I=1,LA
      DO 1 J=1,LB
      K=I+J-1
1     C(K)=C(K)+A(I)*B(J)
      RETURN
      END
      SUBROUTINE ZERO(LX,X)
C*********************************************************************C
C
C     SUBROUTINE ZERO SETS ALL VALUES IN A SERIES EQUAL TO ZERO       C
C
C     X - SERIES TO BE FILLED WITH ZEROS
C
C     LX - LENGTH OF X
C
C*********************************************************************C
      DIMENSION X(LX)
      IF(LX.LE.0) RETURN
      DO 1 I=1,LX
1     X(I)=0.0
      RETURN
      END
-.864-.203 .812-.644 .621 .753 .359-.122 .971 .033-.118 .589-.370 .501-.435-.463
-.750 .766 .576-.982-.522 .643 .035-.506-.522 .861 .882-.106-.193 .140-.332 .800
 .472 .718-.264 .800-.261-.213-.811 .590-.710-.899-.969-.060-.746-.297 .118-.975
 .988 .076 .298 .286-.346 .145 .485 .361-.144 .094 .266 .845-.987 .744 .812 .913
-.230-.444-.869-.008-.553 .927-.494 .769 .082 .245 .864 .322-.364 .694-.918 .965
-.049-.223 .043-.318 .417 .101 .730-.576 .296 .789 .680-.982-.572-.308 .558-.666
 .550-.160-.831 .430 .623 .812 .399 .722-.840-.926-.051 .255 .141 .125-.591-.157
 .266 .456 .677 .265-.608 .829-.257-.402 .277-.057-.375-.964-.664-.004 .415-.447
-.942-.763-.261-.758 .848-.905 .026 .230-.818-.184-.775-.849 .646-.808-.644-.811
-.755-.286-.267-.213 .702-.509 .890 .031-.535 .064 .143-.782-.863-.241 .175-.211
-.179-.286 .667 .156 .505-.106-.542-.710 .720 .264 .412 .521-.988-.734-.609 .917
-.280 .649-.237 .859-.413 .631-.334 .793-.947-.058 .867-.523 .848 .372 .222-.222
-.252-.652-.796-.122 .911 .415-.698-.237 .865-.623 .699-.428-.393-.338 .895 .579
 .896-.003-.023 .540-.955-.132 .207 .483 .626 .083 .350-.371 .631 .873-.388-.515
 .755-.203 .810-.125 .480 .714-.888-.077-.640-.460-.276-.780 .842-.419 .633 .513
-.248-.070-.841-.683 .514 .741 .591 .730 .565 .791-.314 .106 .357 .460-.051-.298
 .924-.576-.792-.432 .913 .627-.427-.080 .270-.523 .819 .508-.245 .196-.984 .910
 .956-.488-.131-.497 .578 .717 .914-.389-.024 .917-.605-.920 .193-.845-.978 .571
 .189 .082-.274 .925-.575-.147-.532 .148-.828-.176 .550-.576-.389 .899 .137 .778
-.672-.194-.081-.964-.291 .348 .148-.035 .156-.748 .605-.020 .225 .635 .243 .000
 .562-.451-.085 .465 .998-.807-.569 .513-.111-.643 .533 .139 .544-.293-.409 .818
 .820 .641-.860 .683-.966-.405-.216 .621-.418 .586 .507-.566-.884-.791-.144-.010
```

```
 .879-.377-.546 .669 .672 .944 .476 .612 .748-.130-.471 .635-.771-.180 .078 .204
 .036-.963-.582 .653-.855-.707 .249 .531 .439 .420 .089-.397-.212 .957-.075 .869
 .085-.123-.667 .793-.143 .452-.690 .822 .371-.276-.175 .747-.422 .333 .784-.733
 .281-.719 .023 .616 .925-.213 .116-.196 .677 .332 .163 .923 .139-.977-.888-.567
-.949-.941 .316-.913 .690 .819 .426 .285 .994 .033-.107 .617 .473-.885 .548 .089
 .138 .662 .212-.875 .418 .715 .081-.346-.402 .002-.340-.735 .792-.934-.771 .989
 .916 .161 .688 .430 .408 .766 .979-.191-.676-.475-.669-.458 .535-.170 .800-.996
 .583 .865-.882 .452-.277-.051-.268-.752 .972-.108 .212-.519-.128 .866-.428-.692
-.036-.865-.719-.644-.234-.344 .441-.327 .923 .045 .084-.762-.090-.875 .297 .997
 .625-.982 .914 .818-.079-.302-.905-.539-.307 .584-.982-.621 .819 .086-.957 .484
SYNTHETIC SEISMOGRAM S/N=4.44
  CONVENTIONAL DECONVOLUTION
TIME AND SPACE ADAPTIVE DECONVOLUTION
   MINIMUM ENTROPY DECONVOLUTION
GRADIENT DESCENT ADAPTIVE DECONVOLUTION
    6  506    2   0.024    0.004
1.8         1.8         0.28
2.2         2.1       0.22
 2.6         2.4      0.20
1.7         1.9       0.2
2.2          2.2      0.2
2.8         2.7        5.
```

Appendix C
Program HUANG

```
      PROGRAM  HUANG(INPUT,OUTPUT      ,PLOT,TAPE1=INPUT,TAPE2=OUTPUT)
C
C THIS PROGRAM CALCULATES THE ENVELOPE,  INSTANTANEOUS PHASE,
C CONTINUOUS PHASE,INSTANTANEOUS FREQUENCY, AND POLARITY.
C
      DIMENSION AMP(520)
      DIMENSION SPECT(520),DSPEC(520)
      DIMENSION DD( 520),DQ( 520),W( 520),FI( 520)
      DIMENSION RE(520),XIM(520)
      DIMENSION T( 520),PHZ( 520),POLA(520)
      DIMENSION XM(520),RA(520),TT(320),PZ(320)
     *  ,V(520),RESPHZ(520),F(520)
      COMPLEX SPECT
      CALL PLOTS
      H=0.05
      DT=0.05
      NPT1=256
      NPTS=257
      NPTS1=258
      NPT2=512
C
C SIMULATED TELESEISMIC DATA
C
      CALL DATAIN2(    NPT2,DSPEC,H,T)
C     CALL DATAIN1(    NPT2,DSPEC,H,T)
C
C PLOT THE SIGNAL FIGURE
C
      CALL SCALE(T,6.0,512,1)
      CALL SCALE(DSPEC,2.0,512,1)
      T(513)=-5.0
      T(514)=5.0
      DSPEC(513)=-1.5
      DSPEC(514)=1.5
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,DSPEC(513),DSPEC(514),-1)
      CALL SYMBOL(0.0,2.3,0.2,16HSIGNAL FIGURE   ,0.0,16)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,DSPEC,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
      CALL       REMAV(NPT2,DSPEC,AVERAGE)
      DO 7003 I=1,NPT2
      SPECT(I)=CMPLX(DSPEC(I),0.0)
 7003 CONTINUE
      CALL FFT2D(SPECT,NPT2,1,-1)
      CALL       ANALSPE( SPECT,NPTS,NPTS1,NPT2)
      CALL FFT2D(SPECT,NPT2,1,1)
      DO 1 I=1,512
      RE(I)=REAL(SPECT(I))
      XIM(I)=AIMAG(SPECT(I))
    1 CONTINUE
      CALL POLAR(512,RE,XIM,AMP,PHZ)
C
C AMP IS THE ENVELOPE
C PHZ IS THE INSTANTANEOUS PHASE
C XIM IS THE QUADRATUE COMPONENT
C
C PLOT THE QUADRATURE COMPONENT
C
      CALL SCALE(XIM,2.0,512,1)
      XIM(513)=-1.5
      XIM(514)=1.5
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,  XIM(513),  XIM(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,17HQUADRATURE SIGNAL,0.0,17)
      CALL PLOT(0.0,2.0,3)
```

```
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,  XIM,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C PLOT THE ENVELOPE
C
      CALL SCALE(AMP,2.0,512,1)
      AMP(513)=-1.5
       AMP(514)=1.5
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,  AMP(513),  AMP(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,17HENVELOPE         ,0.0,17)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,  AMP,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C PLOT THE INSTANTANEOUS PHASE
C
      CALL SCALE(PHZ,2.0,512,1)
      PHZ(513)=-4.0
      PHZ(514)=4.0
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,  PHZ(513),  PHZ(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,25HINSTAN. PHASE BY ARCTAN   ,0.0,25)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,  PHZ,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C CALCULATE THE INSTANTANEOUS FREQUENCY FROM EQUATION (3-10),
C  I.E., DERIVATIVE OF THE ARC TAN.
C
      CALL DERIVA2(RE,512,DT,DD)
      CALL DERIVA2(XIM,512,DT,DQ)
      DO 19 I=1,512
      W(I)=(RE(I)*DQ(I)-DD(I)*XIM(I)) /(RE(I)**2+XIM(I)**2)
      W(I)=W(I)/(2.0*3.14159265)
19    CONTINUE
C
C PLOT THE INSTANTANEOUS FREQUENCY W(I)
C
      CALL SCALE(W,2.0,512,1)
      W(513)=-6.0
      W(514)=6.0
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,    W(513),    W(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,22HFREQUENCY BY FORMULA  ,0.0,22)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,    W,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C CALCULATE THE CONTINUOUS PHASE BY DRUM FROM E. A. ROBINSON
C PLOT  CONTINUOUS PHASE
C
      CALL DRUM(NPT2,PHZ)
      CALL SCALE(PHZ,4.0,512,1)
C     CALL SCALE(PHZ,2.0,512,1)
      PHZ(513)=-4.
      PHZ(514)=30.
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,4.0,90.0,  PHZ(513),  PHZ(514),-1)
      CALL SYMBOL(0.0,4.3 ,0.2,25HINSTAN. PHASE BY DRUM     ,0.0,25)
```

```
      CALL PLOT(0.0,4.0,3)
C     CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,4.0,2)
C     CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,  PHZ,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C CALCULATE THE RESIDUAL PHASE OF LINEAR LEAST SQUARE FITTING
C  AND PLOT THIS RESIDUAL PHASE
C
       DO 77 I=1,311
      PZ(I)=PHZ(I+27)
      TT(I)=T(I+27)
77    CONTINUE
      CALL FLEAST(311,TT,PZ)
      CALL SCALE(TT,6.0,311,1)
      TT(312)=-5.0
      TT(313)=5.0
      CALL SCALE(PZ,2.0,311,1)
      PZ(312)=-2.0
      PZ(313)=2.0
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,TT(513),TT(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0, PZ(312),PZ(313),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,33HLEAST SQUARE FITTING(FIRST ORDER) ,
     * 0.0,33)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(TT,PZ,311,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C CALCULATE THE INSTANTANEOUS FREQUENCY FROM DERIVATIVE OF THE
C THE CONTINUOUS PHASE
C
      CALL DERIVA2(PHZ,NPT2,DT ,FI)
      DO 30 I=1,512
      FI(I)=FI(I)/(2.0*3.14159265)
30    CONTINUE
C
C CALCULATE THE POLARITY
C
      CALL POLATY(DSPEC,AMP,NPT2,POLA)
C
C PLOT THE INSTANTANEOUS FREQUENCY F(I)
C
      CALL SCALE(FI,2.0,512,1)
      FI(513)=-6.0
      FI(514)=6.0
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0,   FI(513),   FI(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,20HFRE. BY DERIVA DRUM  ,0.0,20)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
      CALL PLOT(6.0,0.0,2)
      CALL LINE(T,   FI,512,1,0,0)
      CALL PLOT(7.5,0.0,-3)
C
C PLOT THE POLARITY
C
      CALL SCALE(POLA,2.0,512,1)
      POLA(513)=-1.5
      POLA(514)=1.5
      CALL AXIS(0.0,0.0,9HTIME AXIS ,-9,6.0,0.0,T(513),T(514),0)
      CALL AXIS(0.0,0.0,6HY AXIS,6,2.0,90.0, POLA(513), POLA(514),-1)
      CALL SYMBOL(0.0,2.3 ,0.2,10HPOLARITY      ,0.0,10)
      CALL PLOT(0.0,2.0,3)
      CALL PLOT(6.0,2.0,2)
```

```
      CALL PLOT(6.0,0:0,2)
      CALL LINE(T, POLA,512,1,0,0)
      CALL PLOT(0,0,999)
      STOP
      END
      SUBROUTINE DERIVA2(F,N,H,D)
C
C CALCULATE THE DERIVATIVE OF THE TABULATED FUNCTION
C
      DIMENSION F(N),D(N)
      D(1)=(-21.*F(1)+13.*F(2)+17.*F(3)-9.*F(4))/(20.*H)
      D(2)=(-11.*F(1)+3.*F(2)+7.*F(3)+F(4))/(20.*H)
      M=N-2
      DO 10 I=3,M
      D(I)=(-2.*F(I-2)-F(I-1)+F(I+1)+2.*F(I+2))/(10.*H)
   10 CONTINUE
      D(N-1)=-(-11.*F(N)+3.*F(N-1)+7.*F(N-2)+F(N-3))/(20.*H)
      D(N)=-(-21.*F(N)+13.*F(N-1)+17.*F(N-2)-9.*F(N-3))/(20.*H)
      RETURN
      END
      SUBROUTINE POLAR(L,RE,XIM,AMP,PHZ)
C SUBROUTINE POLAR COMES FROM ROBINSON P.66
      DIMENSION RE(L),XIM(L),AMP(L),PHZ(L)
      PI=3.14159265
      DO 110 I=1,L
      AMP(I)=SQRT(RE(I)**2+XIM(I)**2)
      IF(XIM(I)) 10,20,30
10    IF(RE(I)) 40,50,60
20    IF(RE(I)) 70,80,60
30    IF(RE(I)) 90,100,60
40    PHZ(I)=ATAN(XIM(I)/RE(I))-PI
      GO TO 110
50    PHZ(I)=-PI/2.0
      GO TO 110
60    PHZ(I)=ATAN(XIM(I)/RE(I))
      GO TO 110
70    PHZ(I)=-PI
      GO TO 110
80    PHZ(I)=0.0
      GO TO 110
90    PHZ(I)=ATAN(XIM(I)/RE(I))+PI
      GO TO 110
100   PHZ(I)=PI/2.0
110   CONTINUE
      RETURN
      END
      SUBROUTINE DRUM(LPHZ,PHZ)
      DIMENSION PHZ(LPHZ)
      PJ=0.
      DO 40 I=2,LPHZ
      IF(ABS(PHZ(I)+PJ-PHZ(I-1))-3.14159265) 40,40,10
   10 IF(PHZ(I)+PJ-PHZ(I-1)) 20,40,30
   20 PJ=PJ+3.14159265*2.
      GO TO 40
   30 PJ=PJ-3.14159265*2.
   40 PHZ(I)=PHZ(I)+PJ
      RETURN
      END
      SUBROUTINE REMAV(LY,Y,AVERAG)
C REMOVE AVERAGE
      DIMENSION Y(LY)
      S=0.
      DO 10 I=1,LY
10    S=S+Y(I)
      AVERAG=S/FLOAT(LY)
      DO 20 I=1,LY
20    Y(I)=Y(I)-AVERAG
      RETURN
```

```
      END
      SUBROUTINE ANALSPE( SPECT,NPTS,NPTS1,NPT2)
C
C THIS SUBROUTINE CALCULATE  THE ANALYTIC SIGNAL SPECTRAL I.E.,
C  DOUBLE THE POSITIVE FREQUENCY  SPECTRAL AND MAKE THE NEGATIVE
C FREQUENCY SPECTRAL ZERO
C
      DIMENSION SPECT(NPT2)
      COMPLEX SPECT
      DO 1 I=2,NPTS
      SPECT(I)=SPECT(I)*2.0
    1 CONTINUE
      DO 2 I=NPTS1,NPT2
      SPECT(I)=0.0
    2 CONTINUE
      RETURN
      END
      SUBROUTINE FFT2D(H,NX,NY,NSIGN)
C**********************************************************************C
C                                                                      C
C     SUBROUTINE FFT2D COMPUTES THE TWO DIMENSIONAL FOURIER TRANSFORM  C
C     OF A COMPLEX ARRAY H(NX,NY).  NX AND NY MUST BE A POWER OF 2.    C
C                                                                      C
C     NSIGN = +1  INVERSE TRANSFORM                                    C
C                                                                      C
C     NSIGN = -1  FORWARD TRANSFORM                                    C
C                                                                      C
C**********************************************************************C
      COMPLEX H(NX,NY),CTEMP(256)
      SIGNI=NSIGN
      DO 10 IY=1,NY
   10 CALL FORK(NX,H(1,IY),SIGNI)
      IF(NY.EQ.1) RETURN
      DO 20 IX=1,NX
      DO 30 IY=1,NY
   30 CTEMP(IY)=H(IX,IY)
      CALL FORK(NY,CTEMP,SIGNI)
      DO 40 IY=1,NY
   40 H(IX,IY)=CTEMP(IY)
   20 CONTINUE
      RETURN
      END
      SUBROUTINE FORK(LX,CX,SIGNI)
C   ******************************************************************
C   FAST FOURIER TRANSFORM, MODIFIED FROM CLAERBOUT, J.F., FUNDAMENTALS
C   OF GEOPHYSICAL DATA PROCESSING, MCGRAW-HILL, 1976
C         LX
C   CX(K)=SUM(CX(J)*EXP(2*PI*SIGNI*I*(J-1)*(K-1)/LX))
C         J=1
C        FOR K=1,2,....,(LX=2**INTEGER)
C   SIGNI=+1.   INVERSE TRANSFORM
C   SIGNI=-1.   FORWARD TRANSFORM
C   LX MUST BE A POWER OF 2 (LX=2**INTEGER)
C   NORMALIZATION PERFORMED BY DIVIDING BY
C   DATA LENGTH UPON THE FORWARD TRANSFORM
C   ******************************************************************
      COMPLEX CX(LX),CARG,CEXP,CW,CTEMP
      J=1
      SC=1./FLOAT(LX)
      DO 30 I=1,LX
      IF(I.GT.J) GO TO 10
      CTEMP=CX(J)
      CX(J)=CX(I)
      CX(I)=CTEMP
 10   M=LX/2
 20   IF(J.LE.M)GO TO 30
      J=J-M
      M=M/2
```

```
      IF(M.GE.1)GO TO 20
30    J=J+M
      L=1
40    ISTEP=2*L
      DO 50 M=1,L
      CARG=(0.,1.)*(3.14159265*SIGNI*FLOAT(M-1))/FLOAT(L)
      CW=CEXP(CARG)
      DO 50 I=M,LX,ISTEP
      CTEMP=CW*CX(I+L)
      CX(I+L)=CX(I)-CTEMP
50    CX(I)=CX(I)+CTEMP
      L=ISTEP
      IF(L.LT.LX)GO TO 40
      IF(SIGNI.GT.0.0) RETURN
      DO 60 I=1,LX
   60 CX(I)=CX(I)*SC
      RETURN
      END
      SUBROUTINE DATAIN1(     NPT2,D,H,T)
C
C SIMULATED TELESEISMIC PULSE
C
      DIMENSION D(NPT2),T(NPT2),TT(1000)
      B(N)=TT(N)**2* EXP(2.0-2.0*TT(N))*SIN(2.*3.14159265*TT(N))
      NS=21
      DO 1  I=1,NPT2
      T(I)=  H*(I-21)
1     CONTINUE
      DO 3 I=1,NPT2
      TT(I)=H*(I-1)
3     CONTINUE
      DO 2 I=1,NPT2
      IF(I.LT.NS) D(I)=0.0
      IF(I.GE.NS) D(I)=B(I- 20)
2     CONTINUE
      RETURN
      END
      SUBROUTINE DATAIN2(NPT2,D,H,T)
C
C SIMULATED TELESEISMIC SIGNAL
C
      DIMENSION D(NPT2),T(NPT2),TT(1000)
      B(N)=TT(N)**2* EXP(2.0-2.0*TT(N))*SIN(2.*3.14159265*TT(N))
      NS=21
      N1=53
      N2=67
      N3=67
      N4=131
      N5=191
      N6=217
      N7=247
      DO 1  I=1,NPT2
      T(I)=  H*(I-21)
      TT(I)=H*(I-1)
1     CONTINUE
      DO 2 I=1,NPT2
      IF(I.LT.NS) D(I)=0.0
      IF(I.GE.NS.AND.I.LT.N1) D(I)=B(I-20)
      IF(I.GE.N1.AND.I.LT.N2) D(I)=B(I-20)+0.29*B(I-52)
      IF(I.GE.N2.AND.I.LT.N3) D(I)=B(I-20)+0.29*B(I-52)-0.24*B(I-66)
      IF(I.GE.N3.AND.I.LT.N4) D(I)=B(I-20)+0.29*B(I-52)-0.24*B(I-66)
     *  +0.36*B(I-98)
      IF(I.GE.N4.AND.I.LT.N5) D(I)=B(I-20)+0.29*B(I-52)-0.24*B(I-66)
     *  +0.36*B(I-98)-0.37*B(I-130)
      IF(I.GE.N5.AND.I.LT.N6) D(I)=B(I-20)+0.29*B(I-52)-0.24*B(I-66)
     *  +0.36*B(I-98)-0.37*B(I-130)+0.2*B(I-190)
      IF(I.GE.N6.AND.I.LT.N7) D(I)=B(I-20)+0.29*B(I-52)-0.24*B(I-66)
     *  +0.36*B(I-98)-0.37*B(I-130)+0.2*B(I-190) +0.25*B(I-216)
```

```
      IF(I.GE.N7)        D(I)=B(I-20)  +0.29*B(I-52)-0.24*B(I-66)
     *  +0.36*B(I-98)-0.37*B(I-130)+0.2*B(I-190) +0.25*B(I-216)
     *  +0.15*B(I-246)
2     CONTINUE
      RETURN
      END
      SUBROUTINE POLATY(A,AMPL,NP,POLA)
C
C CALCULATE THE POLARITY BY THE COMPARISON METHOD
C
      DIMENSION A(520),AMPL(520),POLA(520),NS(520),XMAX(520)
      AMP1=AMPL(1)
      AMP2=AMPL(2)
      NP1=NP-1
      KK=0
      DO 500 I=3,NP1
      AMP3=AMPL(I)
      IF(AMP2.GT.AMP1.AND.AMP2.GE.AMP3)  GO TO 400
      GO TO 401
400   AMP4=AMPL(I+1)
      IF(AMP2.LT.AMP4)  GO TO 401
      KK=KK+1
      NS(KK)=I-1
      XMAX(KK)=AMP2
401   AMP1=AMP2
      AMP2=AMP3
500   CONTINUE
      DO 30 I=1,NP
      POLA(I)=0.0
30    CONTINUE
      K1=KK-1
      DO 601 K=1,K1
      N1=NS(K)
      N2=NS(K+1)-1
      DO 602 I=N1,N2
      IF(A(N1).LT.0.0)  GO TO 4
      POLA(I)=XMAX(K)
      GO TO 602
4     POLA(I)=-XMAX(K)
602   CONTINUE
601   CONTINUE
      J=NS(KK)
      DO 8 I=J,NP
      IF(A( J).LT.0.0)  GO TO 7
      POLA(I)=XMAX(KK)
      GO TO 8
7     POLA(I)=-XMAX(KK)
    8 CONTINUE
      RETURN
      END
      SUBROUTINE FLEAST(N,X,Y)
C
C LINEAR LEAST SQUARE FITTING
C
      DIMENSION X(N),Y(N)
      AN=N
      SUMX=0.0
      SUMY=0.0
      SUMXY=0.0
      SUMXX=0.0
      DO 2 I=1,N
      SUMX=SUMX+X(I)
      SUMY=SUMY+Y(I)
      SUMXY=SUMXY+X(I)*Y(I)
      SUMXX=SUMXX+X(I)**2
2     CONTINUE
      XAV=SUMX/AN
      YAV=SUMY/AN
```

```
      CSCP=SUMXY-SUMX*SUMY/AN
      CSS=SUMXX-SUMX**2/AN
      B=CSCP/CSS
      A=YAV-B*XAV
      DO 9 I=1,N
      Y(I)=Y(I)-(B*X(I)+A)
9     CONTINUE
      RETURN
      END
```

Appendix D
Program MCGIL

```
      PROGRAM MCGIL(INPUT,OUTPUT,      PLOT,FILE3,FILE4,FILE5,FILE6,
     *  FILE7,FILE8,FILE9,FILE11,TAPE1=INPUT,
     *  TAPE2=OUTPUT,TAPE3=FILE3,TAPE4=FILE4,TAPE5=FILE5,
     *  TAPE6=FILE6,TAPE7=FILE7,TAPE8=FILE8,TAPE9=FILE9,TAPE11=FILE11)
C
C THIS PROGRAM CALCULATES THE PROCESS 1, 2, 3, IN CHAPTER 1 AND CHAPTER
C 4. WE CALCULATE THE ENVELOPE,INSTANTANEOUS PHASE, CONTINUOUS PHASE
C  ,RESIDUAL PHASE OF LINEAR LEAST SQUARE FITTING, INSTANTANEOUS
C  FREQUENCY  AND POLARITY IN THE SYNTHETIC SEISMOGRAM.
C
      DIMENSION TITLE(5),C( 520),DSPEC( 520),SPECT( 520)
      DIMENSION C2( 520),R(520),S(50)
     *                ,VEL(520),DENS(50),B(50),THICK(50),TK(50),
     *  TZ(520),RCOEF(50),Z( 520)
      DIMENSION X(520),CX(12,520),RNOISE(1024),
     *  A(520),POLA(520),          PHZ(520),PHII(520),W(520),FI(520)
     *                             ,DD(520),DQ(520)
      DIMENSION AMPL(520)
      DIMENSION RE(520),XIM(520)
     *  ,T(520)
      DIMENSION PHZ1(520)  ,PHZD(520)
      COMPLEX SPECT
      CALL PLOTS
      SF= 1.0
      ST=0.0
      DIST=0.0
      DWARM=0.16
      SRATE=0.004
      WSTRT=0.2
      DO 41 I=1,512
      T(I)=SRATE*(I-1)
41    CONTINUE
      READ(1,19) (RNOISE(I),I=1,1022)
19    FORMAT(16F5.3)
      X1=0.   $    X2=14.   $    XINCH=6.$ NX=14
      T1=0.   $   T2=2.  $ TINCH=8.  $ NT=20
      NSEIS=13  $NDIST=0  $   NFRMT=0  $ NAMP=2  $NEXP=0
      NREV=1  $NPOS=0  $NSCALE=0  $ NFILL=1
      REDVEL=0.  $ TO=0.  $FACTOR=0.  $  AMP=1.  $  FDIST=0.
      DELX=0.  $  ALPHA=0.
      READ(1,10)   (TITLE(J),J=1,5)
      WRITE(3,10) (TITLE(J),J=1,5)
      WRITE(3,11)X1,X2,XINCH,NX
      WRITE(3,11)T1,T2,TINCH,NT
      WRITE(3,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(3,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)   (TITLE(J),J=1,5)
      NFILL=0
      WRITE(4,10) (TITLE(J),J=1,5)
      WRITE(4,11)X1,X2,XINCH,NX
      WRITE(4,11)T1,T2,TINCH,NT
      WRITE(4,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE(4,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)   (TITLE(J),J=1,5)
      NFILL=1
      WRITE(5,10)(TITLE(I),I=1,5)
      WRITE(5,11)X1,X2,XINCH,NX
      WRITE(5,11) T1,T2,TINCH,NT
      WRITE(5,12) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      WRITE( 5,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      READ(1,10)   (TITLE(J),J=1,5)
      WRITE(6,10) (TITLE(J),J=1,5)
      WRITE(6,11)X1,X2,XINCH,NX
      WRITE(6,11)T1,T2,TINCH,NT
      WRITE(6,12)NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
```

```
      WRITE(6,13)REDVEL,TO,FACTOR,AMP,FDIST,DELX,ALPHA
      XTAU=10.  $ ZTAU=0.16
      NZC=0  $ LCN=7   $  IFLGAP=1
      NWARM=DWARM/ SRATE
      NSTRT=WSTRT/ SRATE
      DO 3 JJ=1,7
      DIST=DIST+1.0
       READ(1,100)  NOL,N,NEXT,SLNGTH,DT
      WRITE(2,100)  NOL,N,NEXT,SLNGTH,DT
      DO 5 I=1,NOL
       READ(1,26) VEL(I),DENS(I),THICK(I)
      IF(DENS(I).LT..001) DENS(I)=.252+.3788*VEL(I)
      WRITE(2,26) VEL(I),DENS(I),THICK(I)
    5 CONTINUE
C
C GENERATE THE SYNTHETIC SEISMOGRAM
C
      CALL        SYNTH(C2,TK,THICK,VEL,TZ,RCOEF ,DENS,Z,N,R,SLNGTH,
     *        DT,NEXT,S,C,NPTS,NP,NOL)
C
C SYNTHETIC SEISMOGRAM PLUS NOISE
C
      DO 22 I=1,512
      IJJ=I+85.*(JJ-1)
22    C(I)=C(I)+0.03*RNOISE(IJJ)
C     DO 1 I=1,NP
C     X(I)=C(I)
C     CONTINUE
      ST=0.  $  SF=1.   $  NP=512   $  DT=0.004
C
C CALCULATE  TIME AND SPACE ADAPTIVE DECONVOLUTION
C
C     CALL BAFL(512,CX,IFLGAP,LCN,NWARM,NSTRT,ZTAU,XTAU,NZC,X)
7     DO 4 I=1,LCN
4     X(I)=0.0
   24 WRITE(3,17) NP,DIST,ST,DT   ,SF
      WRITE( 4,17) NP,DIST,ST,DT   ,SF
      WRITE(5,17) NP  ,DIST,ST, DT    ,SF
      WRITE(6,17) NP  ,DIST,ST, DT    ,SF
      NPT2=512
      DO 71 I=1,NP
71    DSPEC(I)=C(I)
      CALL REMAV(NPT2,DSPEC,AVERAG)
      DO 7003 I=1,NPT2
      SPECT(I)=CMPLX(DSPEC(I),0.0)
 7003 CONTINUE
      CALL FFT2D(SPECT,NPT2,1,-1)
      NPTS=257
      NPTS1=258
      CALL        ANALSPE( SPECT,NPTS,NPTS1,NPT2)
      CALL FFT2D(SPECT,NPT2,1,1)
      DO 38 I=1,512
      RE(I)=REAL(SPECT(I))
      XIM(I)=AIMAG(SPECT(I))
38    CONTINUE
      CALL POLAR(NPT2,RE ,XIM,AMPL,PHZ)
C
C AMPL IS THE ENVELOPE
C PHZ IS THE INSTANTANEOUS PHASE
C XIM IS THE QUADRATUE COMPONENT
C
C
C PHZ1 IS THE THE INSTANTANEOUS PHASE
C
      DO 51 I=1,512
      PHZ1(I)=PHZ(I)/(3.14159265*5)
51    CONTINUE
      CALL DRUM(NPT2,PHZ)
```

```
C
C PHZD IS THE CONTINUOUS PHASE
C
      DO 52 I=1,512
      PHZD(I)=PHZ(I)/400.
52    CONTINUE
C
C W IS THE INSTANTANEOUS FREQUENCY
C
      CALL DERIVA2(PHZ,512,DT,W)
C
C NOW PHZ IS THE RESIDUAL PHASE OF LINEAR LEAST SQUARE FITTING
C
      CALL FLEAST(NPT2,T,PHZ)
      DO 54 I=1,512
      W(I)=W(I)/1500.
      PHZ(I)=PHZ(I)/50.
54     CONTINUE
      WRITE(3,18) (PHZ1(I),I=1,512)
      WRITE( 4,18) (PHZD(I),I=1,512)
      WRITE(5,18)  ( W(I),I=1,512)
      WRITE(6,18) ( PHZ(I),I=1,NP)
      IF(JJ.EQ.7) GO TO 3
      DIS=DIST+(14.-2.*JJ)
      WRITE(3,17) NP,DIS ,ST,DT   ,SF
      WRITE(4,17) NP  ,DIS ,ST, DT    ,SF
      WRITE(5,17) NP  ,DIS ,ST, DT    ,SF
      WRITE(6,17) NP  ,DIS ,ST, DT    ,SF
      WRITE(3,18) (PHZ1(I),I=1,512)
      WRITE( 4,18) (PHZD(I),I=1,512)
      WRITE(5,18)  ( W(I),I=1,512)
      WRITE(6,18) ( PHZ(I),I=1,NP)
3     CONTINUE
      REWIND 3
      CALL SECTION(3)
      CALL PLOT(2.,0.,-3)
      REWIND 4
      CALL SECTION(4)
      REWIND 5
      CALL SECTION(5)
      REWIND 6
      CALL SECTION(6)
   26 FORMAT(3F10.3)
  100 FORMAT(3I5,2F10.5)
   10 FORMAT(5A10)
   11 FORMAT(3F10.3,I5)
   12 FORMAT(9I5)
   13 FORMAT(7F10.3)
   14 FORMAT(3I5,2F10.3)
   16 FORMAT(3F10.3)
   17 FORMAT(I5,3F10.4,F20.10)
   18 FORMAT(8F10.3)
10001 FORMAT(5A10)
10002 FORMAT(3F10.3,I5)
10003 FORMAT(9I5)
10004 FORMAT(7F10.3)
10005 FORMAT(I5,3F10.3,F20.10)
      CALL PLOT(0,0,999)
      STOP
      END
      SUBROUTINE SECTION(IN)
C************************************************************************
C
C     SUBROUTINE SCTION PLOTS A RECORD
C     SECTION CONSISTING OF NSEIS SEISMOGRAMS
C     AND PLOTS A DISTANCE AND TIME SCALE
C     SURROUNDING THE SEISMOGRAMS. THE RECORD
C     SECTION MAY USE NORMAL TIME SCALE OR
```

```
C     A REDUCED VELOCITY TIME SCALE (T-X/REDUCING
C     VELOCITY). DISTANCE ALWAYS INCREASES TO
C     THE RIGHT. TIME MAY BE SET TO INCREASE
C     UPWARDS OR DOWNWARDS. THE INPUT DATA
C     AND SEISMOGRAMS MUST BE WRITTEN
C     ON FILE IN (SPECIFIED IN THE ARGUMENT
C     OF THE CALL STATEMENT) AND IN THE
C     FORMATS SPECIFIED BELOW IF SCTION IS TO
C     BE CALLED FROM ANOTHER PROGRAM. FOR
C     PLOTTING A SECTION FROM SEISMOGRAMS ON
C     CARDS, IN MAY BE THE INPUT FILE AND
C     THE SUBROUTINE WILL READ THE DATA FROM CARDS.
C
C*******************************************************************
C
C
C     INPUTS ARE
C
C     1) TITLE(I), I=1,5 (5A10) 50 CHARACTER TITLE
C
C     2) X1,X2,XINCH,NX  (3F10.3,I5)
C
C        X1        BEGINNING DISTANCE
C        X2        ENDING DISTANCE
C        XINCH     LENGTH OF DISTANCE AXIS OF PLOT IN INCHES
C        NX        NUMBER OF INTERVALS ON X AXIS
C
C     3) T1,T2,TINCH,NT (3F10.3,I5)
C
C        SAME AS 2) EXCEPT FOR TIME AXIS
C
C     4) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL (9I5)
C
C        NSEIS     NUMBER OF SEISMOGRAMS
C        NDIST=1   SEISMOGRAMS EQUALLY SPACED BEGINNING WITH
C                  FDIST AND SPACING DELX
C        NDIST=0   DISTANCES READ IN SEPARATELY
C        NFRMT=0   SEISMOGRAMS IN (8F10.3) FORMAT
C        NFRMT=1   SEISMOGRAMS IN (16F5.2) FORMAT AND SCALED
C                  FROM 0. TO 9.99 SO THAT MEAN VALUE (5.0) IS
C                  REMOVED BEFORE PLOTTING
C        NAMP=0    ALL SEISMOGRAMS SCALED TO AMP INCHES MAXIMUM
C                  ZERO TO PEAK AMPLITUDE FOR PLOTTING
C        NAMP=1    ALL SEISMOGRAMS PLOTTED AT REAL AMPLITUDE*AMP
C                  *DISTANCE**ALPHA
C        NAMP=2    ALL SEISMOGRAMS PLOTTED AT REAL AMPLITUDE * AMP
C        NEXP=0    NO EXPONENTIAL TIME VARIABLE AMPLITUDE SCALING
C        NEXP=1    AMPLITUDES SCALED BY MULTIPLYING BY EXP(FACTOR
C                  *(TIME-T0))
C        NREV=0    TIME INCREASES UPWARD
C        NREV=1    TIME INCREASES DOWNWARD
C        NPOS=0    AMPLITUDES POSITIVE TO THE RIGHT
C        NPOS=1    AMPLITUDES POSITIVE TO THE LEFT
C        NSCALE=0  NO LINEAR TIME VARIABLE AMPLITUDE SCALING
C        NSCALE=1  AMPLITUDES SCALED BY MULTIPLYING BY
C                  FACT=(FACTOR*(TIME-T0)+1.)
C                  IF(FACT.LT.1.0) FACT=1.0
C        NFILL=0   NO SHADING OF POSITIVE CYCLES
C        NFILL=1   POSITIVE PORTION OF WAVELETS SHADED
C        NFILL=2   NEGATIVE PORTION OF WAVELETS SHADED
C
C     5) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA  (7F10.3)
C
C        REDVEL    REDUCING VELOCITY
C                  IF (REDVEL.EQ.0) NO REDUCING VELOCITY USED
C        T0        MINIMUM TIME FOR TIME VARIABLE AMPLITUDE SCALING
C                  T0 IS REFERENCED TO REDUCED TIME IF USED.
C                  (SEE 4) NEXP AND NSCALE)
```

```
C         FACTOR     AMPLITUDE SCALING FACTOR FOR TIME VARIABLE AMPLITUDE
C                    (SEE 4) NEXP AND NSCALE)
C         AMP        SCALE FACTOR FOR PLOTTING AMPLITUDES
C                    (SEE 4) NAMP)
C         FDIST      IF(NDIST.EQ.1) FIRST DISTANCE
C         DELX       IF(NDIST.EQ.1) DISTANCE INCREMENT
C         ALPHA      EXPONENT IN SCALING FACTOR FOR PARTIALLY REMOVING
C                    GEOMETRICAL SPREADING BY MULTIPLYING AMPLITUDES BY
C                    DISTANCE**ALPHA  (SEE 4) NAMP)
C
C *** INPUTS 6) AND 7) ARE REPEATED NSEIS TIMES
C
C     6) NPTS,DIST,ST,DELTAT,SF  (I5,3F10.3,F20.10)
C
C         NPTS       NUMBER OF POINTS IN SEISMOGRAM
C         DIST       DISTANCE
C         ST         STARTING TIME OF RECORD WITH RESPECT TO ORIGIN TIME
C         DELTAT     SAMPLING INTERVAL (SEC)
C         SF         SCALE FACTOR TO RECOVER REAL AMPLITUDES FROM
C                    FOLLOWING SEISMOGRAM SEQUENCE BY MULTIPLYING
C                    S(J)=S(J)*SF
C
C     7) S(J),J=1,NPTS  (8F10.3) OR (16F5.2) IF SCALED BETWEEN 0. AND 9.
C
C                    (SEE 4) NFRMT)
C         S(J)       SEISMOGRAM
C
C***********************************************************************
C
C
C
C
C *** OUTPUTS ARE PRINTOUTS OF INPUT DATA FOR CONFIRMATION AND PLOT
C
C***********************************************************************
      DIMENSION S(520 ), TITLE(5)
      READ (IN,126) (TITLE(I),I=1,5)
C     WRITE (2,127) (TITLE(I),I=1,5)
      READ (IN,120) X1,X2,XINCH,NX
C     PRINT 130
C     WRITE (2,120) X1,X2,XINCH,NX
      READ (IN,120) T1,T2,TINCH,NT
C     PRINT 131
C     WRITE (2,120) T1,T2,TINCH,NT
      READ (IN,121) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
C     PRINT 132
C     WRITE (2,136) NSEIS,NDIST,NFRMT,NAMP,NEXP,NREV,NPOS,NSCALE,NFILL
      READ (IN,122) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA
C     PRINT 133
C     WRITE (2,122) REDVEL,T0,FACTOR,AMP,FDIST,DELX,ALPHA
      IF(NFILL.EQ.1) GO TO 5
      IF(NFILL.EQ.2) GO TO 5
      NFLAG=NFILL
      NFILL=0
    5 CONTINUE
      SX=(X2-X1)/XINCH
      SZ=(T2-T1)/TINCH
      NX1=NX+1
      NT1=NT+1
      DX=XINCH/FLOAT(NX)
      DT=TINCH/FLOAT(NT)
      CALL SYMBOL (1.,.5,.15,TITLE,0.0,50)
      CALL PLOT (1.,1.,-3)
      DO 101 I=1,NX1
         X=(I-1)*DX
         CALL PLOT (X,0.,2)
         CALL PLOT (X,.1,2)
  101 CALL PLOT (X,0.,2)
```

```
      DO 102 I=1,NT1
         T=(I-1)*DT
         CALL PLOT (XINCH,T,2)
         CALL PLOT (XINCH-.1,T,2)
  102 CALL PLOT (XINCH,T,2)
      DO 103 I=1,NX1
         X=XINCH-(I-1)*DX
         CALL PLOT (X,TINCH,2)
         CALL PLOT (X,TINCH-.1,2)
  103 CALL PLOT (X,TINCH,2)
      DO 104 I=1,NT1
         T=TINCH-(I-1)*DT
         CALL PLOT (0.,T,2)
         CALL PLOT (.1,T,2)
  104 CALL PLOT (0.,T,2)
      DO 91 I=1,NSEIS
      DXX=DX*I-0.041
      IX=I
      IF(I .GT. 9) GO TO 1
      CALL NUMBER(DXX,8.2,.15,IX,0.0,2HI1)
      GO TO 91
1     CALL NUMBER(DXX,8.2,.15,IX,0.0,2HI2)
 91    CONTINUE
      CALL SYMBOL(2.6,8.5,0.15,7HSTATION,0.0,7)
      DO 95 I=1,NT1
      DTT=    DT*(NT1-I)-0.075
      FY=(I-1)*0.1
      CALL  NUMBER(-0.5,DTT,.15,FY,0.0,4HF3.1)
95     CONTINUE
      CALL SYMBOL(-0.8,3.5,0.15,12HTIME(SECOND),90.,12)
      NR=1
      IF (NREV.EQ.1) CALL PLOT (0.,TINCH,-3)
      IF (NREV.EQ.1) NR=-1
      NP=1
      IF (NPOS.EQ.1) NP=-1
      DO 119 I=1,NSEIS
      IF(NFLAG.EQ.3.AND.I.GT.5) NFILL=1
      IF(NFLAG.EQ.4.AND.I.GT.5) NFILL=2
         READ (IN,123) NPTS,DIST,ST,DELTAT,SF
C     WRITE (2,137)
C        PRINT 134
         IF (NDIST.EQ.1) DIST=FDIST+(I-1)*DELX
C        WRITE (2,123) NPTS,DIST,ST,DELTAT,SF
         IF (NFRMT.EQ.1) GO TO 105
         READ (IN,124) (S(J),J=1,NPTS)
C        PRINT 135
C     WRITE(2,129) (S(J),J=1,NPTS)
         GO TO 107
  105    READ (IN,125) (S(J),J=1,NPTS)
C         PRINT 135
C     WRITE(2,129) (S(J),J=1,NPTS)
         DO 106 J=1,NPTS
  106    S(J)=(S(J)-5.001)/4.99
  107    DO 108 J=1,NPTS
  108    S(J)=S(J)*SF
         XD=(DIST-X1)/SX
         CALL PLOT (XD,0.,3)
         T=ST
         IF (REDVEL.LT..001) GO TO 109
         T=ST-DIST/REDVEL
  109    CONTINUE
         XMAX=1.0
         IF (NAMP.NE.0) GO TO 111
         XMAX=0.0
         DO 110 J=1,NPTS
  110    IF (ABS(S(J)).GT.XMAX) XMAX=ABS(S(J))
  111    IF (NAMP.NE.1) GO TO 113
         SC=DIST**ALPHA
```

```
      DO 112 J=1,NPTS
112   S(J)=S(J)*SC
113   CONTINUE
      DO 116 J=1,NPTS
         S(J)=(S(J)*AMP/XMAX)*NP
         TIME=T+(J-1)*DELTAT-T1
         IF (NEXP.EQ.0) GO TO 114
         FACT=EXP(FACTOR*(TIME-T0))
         GO TO 115
114      IF (NSCALE.EQ.0) GO TO 116
         FACT=(FACTOR*(TIME-T0))+1.
115      IF (TIME.LT.T0) FACT=1.0
         S(J)=S(J)*FACT
116   CONTINUE
      XP=NP
   Y=(T-T1)/SZ
   IF(Y.LT.0.0) GO TO 219
   X=XD
   CALL PLOT(X,Y,2)
219      DO 118 J=1,NPTS
    Y=((T+(J-1)*DELTAT-T1)/SZ)
         X=XD+S(J)
   XXX=XINCH+1.5
   IF(X.GT.XXX  .OR.X.LT.-2.)  GO TO 118
         IF (Y.GT.TINCH.OR.Y.LT.0.0) GO TO 118
         Y=Y*NR
         CALL PLOT (X,Y,2)
         IF (NFILL.EQ.0) GO TO 118
         IF (NFILL.EQ.1) GO TO 117
         IF (XP*S(J).LT.0.) CALL PLOT (X-S(J),Y,2)
         CALL PLOT (X,Y,2)
         GO TO 118
117      IF (XP*S(J).GT.0.) CALL PLOT (X-S(J),Y,2)
         CALL PLOT (X,Y,2)
118   CONTINUE
119 CONTINUE
    T=0.
    IF (NREV.EQ.1) T=-TINCH
    CALL PLOT (XINCH+1.,T-1.,-3)
    RETURN
C
120 FORMAT (3F10.3,I5)
121 FORMAT (9I5)
122 FORMAT (7F10.3)
123 FORMAT (I5,3F10.4,F20.10)
124 FORMAT (8F10.3)
125 FORMAT(16F5.3)
126 FORMAT (5A10)
127 FORMAT (10X,5A10)
129 FORMAT(1X,16F6.3)
130 FORMAT (/,1X,35H      X1         X2        XINCH      NX )
131 FORMAT (/,1X,34H     T1         T2        TINCH      NT )
132 FORMAT (/,1X,51H NSEIS NDIST NFRMT NAMP NEXP NREV NPOS NSCALE NFIL
   1L)
133 FORMAT (/,1X,69H  REDVEL        TO        FACTOR       AMP        FDIST
   1  DELX       ALPHA   )
134 FORMAT      (1X,48H NPTS   DIST        ST        DELTAT             SF
   1)
135 FORMAT (/,1X,73H        SEISMOGRHM VALUES ARE LISTED FROM LEFT TO R
   1IGHT IN SUCCEEDING ROWS,/)
136 FORMAT (1X,I5,I7,I6,4I5,2I6)
137 FORMAT(1H1,///////)
    END
    SUBROUTINE BAFL(LOUT,CX,IFLGAP,LCN,NWARM,ISTRT,ZTAU,XTAU,NZERO,X)
C
C

C          ---------------------------
```

```
C-------------S U B R O U T I N E  B A F L---------------------
C               -----------------------------
C
C     THE BURG ADAPTIVE FILTER: AN ADAPTVIE OR TIME-VARYING
C     FIXED-LEAD PREDICTION ERROR PROCESSOR. ADJUSTMENT OF EACH
C     REFLECTION COEFFICIENT IS MADE EVERY JUMP STATE ATTEMPTING
C     TO MINIMIZE THE STAGE OUTPUT POWER.
C     INPUTS:
C     X(1)...X(LX)=INITIAL DATA
C     LCN= LAST NON-ZERO REFLECTION COEFFICIENT
C     IFLGAP= NUMBER OF GAPS BETWEEN FILTER COEFFICIENTS
C        SETTING IFLGAP=0 DOES NOT GAP THE F.C. AND
C        TRIES TO OPERATE ON THE ENTIRE SPECTRUM FROM
C         0 TO W WHERE IS THE FOLDING FREQUENCY.
C        SETTING IFLGAP=1 OPERATES ON THE PORTION OF
C        THE SPECTRUM FROM 0 TO W/2 , IFLGAP=2 FROM
C        0 TO W/3 FTC. THUS ALLOWING SPECIFICATION OF
C        WHAT PART OF THE SPECTRUM TO DECONVOLVE.
C
C        NWARN=DURATION OF STATIONARY C ESTIMATION CYCLE
C     ISTRT=START OF STATIONARY GATE
C     ZTAU=TEMPORAL RELAXATION TIME TO 1/E
C     XTAU=SPATIAL RELAXATION DISTANCE TO 1/E
C      OUTPUTS:
C       X(1)...X(LOUT)=FORWARD ERROR PREDICTION TRACE
C     OTHER VARIABLES:
C     F(1)...F(LCN)=FORWARD STATE VECTOR
C     B(1)...B(LCN)=BACKWARD STATE VECTOR
C     C(1)...C(LCN)=REFLECTOR COEFFICIENTS AT EACH STATE
C     CX=REFLECTION COEFF. INTEGRATED IN SPACE AND TIME
C     DEN(1)...DEN(LCN)= STAGE AUTOPOWER
C     NUM(1)...NUM(LCN)= STAGE CROSSPOWER
C
C     CALLING ≠BAFL≠ FIRST SETS UP THE LOOPING AND PASSING
C     ARRAYS FOT THE PARTICULAR PROBLEM AS SPECIFIED BY LCN AND
C     IFLGAP. THEN IT COMPUTES A SHORT (LENGTH=NWARM) ESTIMATE
C     OF THE REFLECTION COEFFICIENT SERIES IN ORDER TO START THE
C     ADAPTION OUT WITH SOME REASONABLE NUMBERS. THEN IT LOADS UP
C     THE CX ARRAY WITH THE INITIAL VALUES AND PASSED INTO ENTRY
C     ≠BAFLGO≠.
C
C     THE USAL ENTRY IS ≠BAFLGO≠ WHICH FIRST INITIALIZES THE
C     BACKWARD ARRAY THEN PASSES TO THE MAIN ALGORITHM.
C     THE CX SERIES IS UPDATED EVERY IFLGAP DATA POINTS AND IN
C     THE INTERMEDIATE STEPS THE OUTPUT ARE INTERPOLATED OR
C     PROCESSED AS THOUGH THE R.C. WERE STATIONARY.
C
C
C
      DIMENSION NUM(50),DEN(50),B(50),F(50),C(50),EM(1500),EP(1500),
     1CX(LCN,LOUT),A(50),X(520)
      REAL NUM
      DATA A,B,C/150*0.0/
      FTEST=1.00
      LCNP1=LCN+1
      LCNP2=LCN+2
      IFGM1=IFLGAP-1
      LBSP=LOUT-IFLGAP
      NZERP1=NZERO+1
      NEND=LCN-NZERP1
      DO 80 K=LBSP,LOUT
   80 EP(K)=0.
C     BEGIN STATIONARY WARM-UP
      DO 10 I=1,NWARM
      EM(I)=X(I*IFLGAP+ISTRT)
   10 EP(I)=X(I*IFLGAP+ISTRT)
C
      DO 11 J=2,LCNP1
```

```
      DEN(J)=0.
      NUM(J)=0.
      DO 12 I=J,NWARM
      DEN(J)=DEN(J)+EP(I)*EP(I)+EM(I-J+1)*EM(I-J+1)
   12 NUM(J)=NUM(J)+EP(I)*EM(I-J+1)
      C(J)=-2.*NUM(J)/DEN(J)
      DO 11 I=J,NWARM
      EPI=EP(I)
      EP(I)=EPI+C(J)*EM(I-J+1)
      KS=IFLGAP*LCN+1
   11 EM(I-J+1)=EM(I-J+1)+C(J)*EPI
      DO 8 J=1,LCN
      DO 8 K=1,KS
    8 CX(J,K)=C(J+1)
      DO 33 J=1,LCN
      DEN(J)=DEN(J+1)
   33 NUM(J)=NUM(J+1)
C     END WARM-UP CYCLE
C     SET RELAXATION TIMES
C     DLX=EXP(-1./XTAU)
      DLX=0.
      DL=EXP(-1./ZTAU)
      DRX=1.-DLX
      DR=1.-DL
C
C---------------USUAL--ENTRY-------------------------------------------
      ENTRY BAFLGO
C
C        INITIALIZE BACKWARD VECTOR
35    B(1)=X(KS-IFLGAP)
      A(1)=1.0
      DO 1000 J=2,LCN
      B(J)=X(KS-J*IFLGAP)
      A(J)=CX(J,KS)
      DO 1000 I=2,J
      A(I)=A(I)+CX(J,KS)*A(J-I+1)
 1000 B(J)=B(J)+A(I)*X(KS+(I-J-1)*IFLGAP)
C
C--------BEGIN--MAIN--LOOP---------------------------------------------
C
      DO 5000 K=KS,LOUT,IFLGAP
      Z=X(K)
      DO 1010 J=1,NZERP1
 1010 F(J)=Z
      DO 2000 J=NZERP1,LCN
      DEN(J)=(F(J)**2+B(J)**2)*DR+DEN(J)*DL
      NUM(J)=F(J)*B(J)*DR+NUM(J)*DL
      IF(FTEST.LE.1.1) CX(J,K)=-2.*NUM(J)/DEN(J)
      CX(J,K)=-2.*DRX*NUM(J)/DEN(J)+CX(J,K)*DLX
 2000 F(J+1)=F(J)+CX(J,K)*B(J)
      X(K)=F(LCNP1)
      DO 3000 JR=1,NEND
      J=LCN-JR
 3000 B(J+1)=B(J)+CX(J,K)*F(J)
      IF(NZERO.EQ.0) GO TO 5000
      DO 4000 JR=1,NZERO
      J=NZERP1-JR
 4000 B(J+1)=B(J)
5000  B(1)=Z
C
C--------END--OF--MAIN--LOOP-------------------------------------------
C
C     NOW GO BACK AND FILL IN THE GAPS...
      IF(IFLGAP.EQ.1) GO TO 9050
      DO 9000 L=1,IFGM1
      KSTART=KS+L
      DO 9000 K=KSTART,LOUT,IFLGAP
      KC=K-L
```

```
      Z=X(K)
      DO 6000 J=1,NZERP1
 6000 F(J)=Z
      DO 7000 J=NZERP1,LCN
 7000 F(J+1)=F(J)+CX(J,KC)*B(J)
      X(K)=F(LCNP1)
      DO 8000 JR=1,NEND
      J=LCN-JR
 8000 B(J+1)=B(J)+CX(J,KC)*F(J)
      IF(NZERO.EQ.0) GO TO 9000
      DO 8050 JR=1,NZERO
      J=NZERP1-JR
 8050 B(J+1)=B(J)
 9000 B(1)=Z
C
 9050 FTEST=FTEST+1.0
      RETURN
      END
      SUBROUTINE SYNTH(C2,TK,THICK,VEL,TZ,RCOEF ,DENS,Z,N,R,SLNGTH,
     *   DELTAT,NEXT,S,C,NPTS,NP,NOL)
C
C SYNTH GENERATES THE SYNTHETIC SEISMOGRAM
C
      DIMENSION C2(520),TK(50),THICK(50),TZ(520),RCOEF(50),DENS(50),
     *    Z(520),R(520),S(50  ),C(520),VEL(520)
      INTEGER U,V
      DO 39 J=1,N
         C2(J)=0.0
   39 CONTINUE
      DO 7 I=1,NOL
C     COMPUTE TWO-WAY TRAVEL TIME
      TK(I)=2.0*(THICK(I)/VEL(I))
      IF(I.EQ.1) GO TO 9
      TZ(I)=TZ(I-1)+TK(I)
      GO TO 7
    9 TZ(I)=TK(I)
    7 CONTINUE
      DO 115 I=1,N
      Z(I)=0.0
115   R(I)=0.
C     COMPUTE THE REFLECTION COEFFICIENTS
      DO 20 I=1,NOL
      V=I
      IF(I.EQ.NOL) GO TO 20
      U=I+1
      RCOEF(I)=((VEL(U)*DENS(U))-(VEL(V)*DENS(V)))/((VEL(U)*DENS(U))+
     *  (VEL(V)*DENS(V)))
   20 CONTINUE
C     SET UP TIME SERIES OF RECOEF(S)
      DO 40 I=1,NOL
      K   =TZ(I)/DELTAT
      C2(K)=RCOEF(I)
   40 CONTINUE
      IOPT=1
      CALL WIFLE(IOPT,N,C2,R,Z)
   42 NPTS=SLNGTH/DELTAT+1.
      R(1)=0.
   65 CALL SIGNAL(SLNGTH,DELTAT, NEXT, 1.,NPTS,0,0.0  ,S)
  107      NP=N+NPTS -1
      CALL FOLD(N,R,NPTS,S,NP,C)
      RETURN
      END
      SUBROUTINE WIFLE (IOPT,N,C,R,A)
C LET C(1).,C(2)...C(N)=REFLECTION COEFFICIENTS AT SUCCESIVE DEPTHS
C     A(1)=1.,A(2)...A(N)=INVERSE WAVELETOF THE TRANSMISION SEISMOGRAM
C     R(1)=1.,R(2)...R(N)=SURFACE INPULSE AND ITS REFLECTION SEISMOGRAM
C
C FOR  ; 1.   R(2) R(N); ; 1. ; ;V(N);              J
```

```
C N=3  ;R(2)  1.   R(I); YA(2);=; 0.;      V(J)=PROD (1-C(I)**2)
C       ;R(N) R(2)  1. Y ;A(N); ; 0.;,              I=I=2
C
C      IF(IOPT=1)   C=INPUT   A,R=OUTPUT
C      IF(IOPT=2)   R=INPUT   A,C=OUTPUT
C      IF(IOPT=3)   A=INPUT    C=OUTPUT
C TAKEN FROM GEOPHYSICS,VOL 33,NO.2,APRIL1968  BY J.F. CLAERBOUT
      DIMENSION C(N),A(N),R(N)
      A(1)=1.
      C(1)=-1.
      R(1)=1.
      V=1.
      GO TO (100,200,300),IOPT
  100 DO 120 J=2,N
      A(J)=0.
      R(J)=C(J)*V
      V=V*(1.-C(J)*C(J))
      DO 110 I=2,J
  110 R(J)=R(J)-A(I)*R(J-I+1)
      JH=(J+1)/2
      DO 120 I=1,JH
      BOT=A(J-I+1)-C(J)*A(I)
      A(I)=A(I)-C(J)*A(J-I+1)
  120 A(J-I+1)=BOT
      RETURN
  200 DO 220 J=2,N
      A(J)=0.
      E=0.
      DO 210 I=2,J
  210 E=E+R(I)*A(J-I+1)
      C(J)=E/V
      V=V-E*C(J)
      JH=(J+1)/2
      DO 220 I=1,JH
      BOT=A(J-I+1)-C(J)*A(I)
      A(I)=A(I)-C(J)*A(J-I+1)
  220 A(J-I+1)=BOT
      RETURN
  300 DO 310 I=1,N
  310 C(I)=A(I)
      DO 330 K=2,N
      J=N-K+2
      AL=1./(1.-C(J)*C(J))
      BE=C(J)*AL
      JH=(J+1)/2
      DO 320 I=1,JH
      TOP=AL*C(I)-BE*C(J-I+1)
      C(J-I+1)=AL*C(J-I+1)-BE*C(I)
  320 C(I)=TOP
  330 C(J)=-BE/AL
      RETURN
      END
      SUBROUTINE SIGNAL(T,DT,N,V1,NPTS,NA,THALF,S)
C     T=LENGTH OF WAVELET IN SEC
C     DT=SAMPLING INTERVAL IN SEC
C     N=NUMBER OF EXTREMA IN WAVELET
C        IF(N.LT.0) WAVELET IS MULTIPLIED BY -1.0
C     V1=VELOCITY OF UPPER LAYER OR 1.0
C     NPTS=NUMBER OF POINTS IN WAVELET TIME SERIES
C     NA=NUMBER OF ZEROES BEFORE WAVELET
C     THALF=IF(THALF.GT.0.0)  THE WAVELET IS WEIGHTED BY AN EXPONENTIAL
C        SUCH THAT AT THALF SECONDS THE AMPLITUDE IS MULTIPLIED BY 0.5
C        THALF USUALLY WILL BE EQAL TO APPROXIMATELY T OR T/2
C     S=SIGNAL
      DIMENSION S(50 )
      IF(V1.LT..0001) V1=1.0
      FACTOR=1.0
      IF(N.LT.0) FACTOR=-1.0
```

```
      N=IABS(N)
      DO 100 I=1,NA
100   S(I)=0.0
      NS=T/DT+1.0
      FN=N
      FM=(FN+2.)/FN
      D=FN*3.14159265/T
      TT=-DT
      A=0.0
      DO 200 I=1,NS
      TT=TT+DT
      T1=D*TT
      T2=FM*T1
      C=    SIN(T1)-SIN(T2)/FM
      S(NA+I)=C
      IF(C.GT.A)A=C
200   CONTINUE
      A=V1/A
      DO 300 I=1,NS
      J=NA+I
300   S(J)=A*S(J)
      SE=0.0
      J=NA+NS+1
      DO 400 I=J,NPTS
400   S(I)=SE
      IF(THALF.LT..0001) GO TO 1000
      N2=NA+1
      DO 500 I=N2,NPTS
      TIME=(I-N2)*DT
      X=TIME*.696/THALF
500   S(I)=S(I)*EXP(-X)
1000  CONTINUE
      SMAX=0.0
      DO 600 I=1,NPTS
      IF(SMAX.LT.ABS(S(I))) SMAX=ABS(S(I))
600   CONTINUE
      DO 700 I=1,NPTS
700   S(I)=S(I)*FACTOR/SMAX
      RETURN
      END
      SUBROUTINE FOLD(LA,A,LB,B,LC,C)
C***********************************************************************C
C                                                                       C
C     SUBROUTINE FOLD CONVOLVES TWO SERIES                              C
C                                                                       C
C     A,B - SERIES TO BE CONVOLVED                                      C
C                                                                       C
C     LA,LB - LENGTH OF A AND B                                         C
C     C - CONVOLUTION SERIES                                            C
C                                                                       C
C     LC - LENGTH OF C                                                  C
C                                                                       C
C***********************************************************************C
      DIMENSION A(LA),B(LB),C(LC)
      LC=LA+LB-1
      CALL ZERO(LC,C)
      DO 1 I=1,LA
      DO 1 J=1,LB
      K=I+J-1
1     C(K)=C(K)+A(I)*B(J)
      RETURN
      END
      SUBROUTINE ZERO(LX,X)
C***********************************************************************C
C
C     SUBROUTINE ZERO SETS ALL VALUES IN A SERIES EQUAL TO ZERO         C
C
C     X - SERIES TO BE FILLED WITH ZEROS
```

```
C
C     LX - LENGTH OF X
C
C***********************************************************************C
      DIMENSION X(LX)
      IF(LX.LE.0) RETURN
      DO 1 I=1,LX
1     X(I)=0.0
      RETURN
      END
      SUBROUTINE FFT2D(H,NX,NY,NSIGN)
C***********************************************************************C
C                                                                       C
C     SUBROUTINE FFT2D COMPUTES THE TWO DIMENSIONAL FOURIER TRANSFORM   C
C     OF A COMPLEX ARRAY H(NX,NY).  NX AND NY MUST BE A POWER OF 2.     C
C                                                                       C
C     NSIGN = +1  INVERSE TRANSFORM                                     C
C                                                                       C
C     NSIGN = -1  FORWARD TRANSFORM                                     C
C                                                                       C
C***********************************************************************C
      COMPLEX H(NX,NY),CTEMP(256)
      SIGNI=NSIGN
      DO 10 IY=1,NY
   10 CALL FORK(NX,H(1,IY),SIGNI)
      IF(NY.EQ.1) RETURN
      DO 20 IX=1,NX
      DO 30 IY=1,NY
   30 CTEMP(IY)=H(IX,IY)
      CALL FORK(NY,CTEMP,SIGNI)
      DO 40 IY=1,NY
   40 H(IX,IY)=CTEMP(IY)
   20 CONTINUE
      RETURN
      END
      SUBROUTINE FORK(LX,CX,SIGNI)
C   ********************************************************************
C   FAST FOURIER TRANSFORM, MODIFIED FROM CLAERBOUT, J.F., FUNDAMENTALS
C   OF GEOPHYSICAL DATA PROCESSING, MCGRAW-HILL, 1976
C         LX
C   CX(K)=SUM(CX(J)*EXP(2*PI*SIGNI*I*(J-1)*(K-1)/LX))
C         J=1
C        FOR K=1,2,...,(LX=2**INTEGER)
C   SIGNI=+1.   INVERSE TRANSFORM
C   SIGNI=-1.   FORWARD TRANSFORM
C   LX MUST BE A POWER OF 2 (LX=2**INTEGER)
C   NORMALIZATION PERFORMED BY DIVIDING BY
C   DATA LENGTH UPON THE FORWARD TRANSFORM
C   ********************************************************************
      COMPLEX CX(LX),CARG,CEXP,CW,CTEMP
      J=1
      SC=1./FLOAT(LX)
      DO 30 I=1,LX
      IF(I.GT.J) GO TO 10
      CTEMP=CX(J)
      CX(J)=CX(I)
      CX(I)=CTEMP
10    M=LX/2
20    IF(J.LE.M)GO TO 30
      J=J-M
      M=M/2
      IF(M.GE.1)GO TO 20
30    J=J+M
      L=1
40    ISTEP=2*L
      DO 50 M=1,L
      CARG=(0.,1.)*(3.14159265*SIGNI*FLOAT(M-1))/FLOAT(L)
      CW=CEXP(CARG)
```

```
      DO 50 I=M,LX,ISTEP
      CTEMP=CW*CX(I+L)
      CX(I+L)=CX(I)-CTEMP
50    CX(I)=CX(I)+CTEMP
      L=ISTEP
      IF(L.LT.LX)GO TO 40
      IF(SIGNI.GT.0.0) RETURN
      DO 60 I=1,LX
   60 CX(I)=CX(I)*SC
      RETURN
      END
      SUBROUTINE REMAV(LY,Y,AVERAG)
      DIMENSION Y(1)
      S=0.
      DO 10 I=1,LY
10    S=S+Y(I)
      AVERAG=S/FLOAT(LY)
      DO 20 I=1,LY
20    Y(I)=Y(I)-AVERAG
      RETURN
      END
      SUBROUTINE ANALSPE( SPECT,NPTS,NPTS1,NPT2)
C
C THIS SUBROUTINE CALCULATE  THE ANALYTIC SIGNAL SPECTRAL I.E.,
C  DOUBLE THE POSITIVE FREQUENCY  SPECTRAL AND MAKE THE NEGATIVE
C FREQUENCY SPECTRAL ZERO
C
      DIMENSION SPECT(NPT2)
      COMPLEX SPECT
      DO 1 I=2,NPTS
      SPECT(I)=SPECT(I)*2.0
    1 CONTINUE
      DO 2 I=NPTS1,NPT2
      SPECT(I)=0.0
    2 CONTINUE
      RETURN
      END
      SUBROUTINE DERIVA2(F,N,H,D)
C
C CALCULATE THE DERIVATIVE OF THE TABULATED FUNCTION
C
      DIMENSION F(N),D(N)
      D(1)=(-21.*F(1)+13.*F(2)+17.*F(3)-9.*F(4))/(20.*H)
      D(2)=(-11.*F(1)+3.*F(2)+7.*F(3)+F(4))/(20.*H)
      M=N-2
      DO 10 I=3,M
      D(I)=(-2.*F(I-2)-F(I-1)+F(I+1)+2.*F(I+2))/(10.*H)
   10 CONTINUE
      D(N-1)=-(-11.*F(N)+3.*F(N-1)+7.*F(N-2)+F(N-3))/(20.*H)
      D(N)=-(-21.*F(N)+13.*F(N-1)+17.*F(N-2)-9.*F(N-3))/(20.*H)
      RETURN
      END
      SUBROUTINE DRUM(LPHZ,PHZ)
C
C MAKE THE PHASE CONTINUOUS
C
      DIMENSION PHZ(LPHZ)
      PJ=0.
      DO 40 I=2,LPHZ
      IF(ABS(PHZ(I)+PJ-PHZ(I-1))-3.14159265) 40,40,10
   10 IF(PHZ(I)+PJ-PHZ(I-1)) 20,40,30
   20 PJ=PJ+3.14159265*2.
      GO TO 40
   30 PJ=PJ-3.14159265*2.
   40 PHZ(I)=PHZ(I)+PJ
      RETURN
      END
      SUBROUTINE POLATY(A,AMPL,NP,POLA)
```

```
C
C CALCULATE THE POLARITY BY THE COMPARISON METHOD
C
      DIMENSION A(520),AMPL(520),POLA(520),NS(200)
     *   ,XMAX(200)
      AMP1=AMPL(1)
      AMP2=AMPL(2)
      NP1=NP-1
      KK=0
      DO 500 I=3,NP1
      AMP3=AMPL(I)
      IF(AMP2.GT.AMP1.AND.AMP2.GE.AMP3)  GO TO 400
      GO TO 401
400   AMP4=AMPL(I+1)
      IF(AMP2.LT.AMP4)  GO TO 401
      II=I-1
      KK=KK+1
      NS(KK)=II
      XMAX(KK)=AMP2
401   AMP1=AMP2
      AMP2=AMP3
500   CONTINUE
      DO 30 I=1,NP
      POLA(I)=0.0
30    CONTINUE
      K1=KK-1
      DO 601 K=1,K1
      N1=NS(K)
      N2=NS(K+1)-1
      DO 602 I=N1,N2
      IF(A(N1).LT.0.) GO TO 4
      POLA(I)=XMAX(K)
      GO TO 602
4     POLA(I)=-XMAX(K)
602   CONTINUE
601   CONTINUE
      J=NS(KK)
      DO 8 I=J,NP
      IF(A(J).LT.0.)  GO TO 7
      POLA(I)=XMAX(KK)
      GO TO 8
7     POLA(I)=-XMAX(KK)
    8 CONTINUE
      RETURN
      END
      SUBROUTINE POLAR(L,RE,XIM,AMP,PHZ)
C SUBROUTINE POLAR COMES FROM ROBINSON P.65
      DIMENSION RE(L),XIM(L),AMP(L),PHZ(L)
      PI=3.14159265
      DO 110 I=1,L
      AMP(I)=SQRT(RE(I)**2+XIM(I)**2)
      IF(XIM(I)) 10,20,30
10    IF(RE(I)) 40,50,60
20    IF(RE(I)) 70,80,60
30    IF(RE(I)) 90,100,60
40    PHZ(I)=ATAN(XIM(I)/RE(I))-PI
      GO TO 110
50    PHZ(I)=-PI/2.0
      GO TO 110
60    PHZ(I)=ATAN(XIM(I)/RE(I))
      GO TO 110
70    PHZ(I)=-PI
      GO TO 110
80    PHZ(I)=0.0
      GO TO 110
90    PHZ(I)=ATAN(XIM(I)/RE(I))+PI
      GO TO 110
100   PHZ(I)=PI/2.0
```

```
110   CONTINUE
      RETURN
      END
      SUBROUTINE FLEAST(N,X,Y)
      DIMENSION X(N),Y(N)
      AN=N
      SUMX=0.0
      SUMY=0.0
      SUMXY=0.0
      SUMXX=0.0
      DO 2 I=1,N
      SUMX=SUMX+X(I)
      SUMY=SUMY+Y(I)
      SUMXY=SUMXY+X(I)*Y(I)
      SUMXX=SUMXX+X(I)**2
2     CONTINUE
      XAV=SUMX/AN
      YAV=SUMY/AN
      CSCP=SUMXY-SUMX*SUMY/AN
      CSS=SUMXX-SUMX**2/AN
      B=CSCP/CSS
      A=YAV-B*XAV
      DO 9 I=1,N
      Y(I)=Y(I)-(B*X(I)+A)
9     CONTINUE
      RETURN
      END
-.864-.203 .812-.644 .621 .753 .359-.122 .971 .033-.118 .589-.370 .501-.435-.463
-.750 .766 .576-.982-.522 .643 .035-.506-.522 .861 .882-.106-.193 .140-.332 .800
 .472 .718-.264 .800-.261-.213-.811 .590-.710-.899-.969-.060-.746-.297 .118-.975
 .988 .076 .298 .285-.346 .145 .485 .361-.144 .094 .266 .845-.987 .744 .812 .913
-.230-.444-.869-.008-.553 .927-.494 .769 .062 .245 .864 .322-.364 .694-.918 .965
-.049-.223 .043-.318 .417 .101 .730-.576 .296 .789 .680-.982-.572-.308 .558-.666
 .550-.160-.831 .430 .623 .812 .399 .722-.840-.926-.051 .255 .141 .125-.591-.157
 .266 .456 .677 .265-.608 .829-.257-.402 .277-.057-.375-.964-.664-.004 .415-.447
-.942-.763-.261-.758 .848-.905 .026 .230-.818-.184-.775-.849 .646-.808-.644-.811
-.755-.286-.267-.213 .702-.509 .890 .031-.535 .064 .143-.782-.863-.241 .175-.211
-.179-.286 .667 .156 .505-.106-.542-.710 .720 .264 .412 .521-.988-.734-.609 .917
-.280 .649-.237 .859-.413 .631-.334 .793-.947-.058 .867-.523 .848 .372 .222-.222
-.252-.652-.796-.122 .911 .415-.698-.237 .865-.623 .699-.428-.393-.338 .895 .579
 .896-.003-.023 .540-.955-.132 .207 .483 .626 .083 .350-.371 .631 .873-.388-.515
 .755-.203 .810-.125 .480 .714-.888-.077-.640-.460-.276-.780 .842-.419 .633 .513
-.248-.070-.841-.683 .514 .741 .591 .730 .565 .791-.314 .106 .357 .460-.051-.298
 .924-.576-.792-.432 .913 .627-.427-.080 .270-.523 .819 .508-.245 .196-.984 .910
 .956-.488-.131-.497 .578 .717 .914-.389-.024 .917-.605-.920 .193-.845-.978 .571
 .189 .082-.274 .925-.575-.147-.532 .148-.828-.176 .550-.576-.389 .899 .137 .778
-.672-.194-.081-.964-.291 .348 .148-.035 .156-.748 .605-.020 .225 .635 .243 .000
 .562-.451-.085 .465 .998-.807-.569 .513-.111-.643 .533 .139 .544-.293-.409 .818
 .820 .641-.860 .683-.966-.405-.216 .621-.418 .586 .507-.566-.884-.791-.141-.010
 .879-.377-.546 .669 .672 .944 .476 .612 .748-.130-.471 .635-.771-.180 .078 .204
 .036-.963-.582 .653-.855-.707 .249 .531 .439 .420 .089-.397-.212 .957-.075 .869
 .085-.123-.667 .793-.143 .452-.690 .822 .371-.276-.175 .747-.422 .333 .784-.733
 .281-.719 .023 .616 .925-.213 .116-.196 .677 .332 .163 .923 .139-.977-.888-.567
-.949-.941 .316-.913 .690 .819 .426 .285 .994 .033-.107 .617 .473-.885 .548 .089
 .138 .662 .212-.875 .418 .715 .081-.345-.402 .002-.340-.735 .792-.934-.771 .989
 .916 .161 .688 .430 .408 .766 .979-.191-.676-.475-.669-.458 .535-.170 .800-.996
 .583 .865-.882 .452-.277-.051-.268-.752 .972-.108 .212-.519-.128 .866-.428-.692
-.036-.865-.719-.644-.234-.344 .441-.327 .923 .045 .084-.762-.090-.875 .297 .997
 .625-.982 .914 .818-.079-.302-.905-.539-.307 .584-.982-.621 .819 .086-.957 .484
 .544 .494 .204 .947 .875-.040-.161-.245 .633-.475 .575 .339 .987-.859 .620 .974
 .734 .262-.254 .509 .708 .074-.539-.839-.149 .201 .280 .217 .491-.672-.313-.873
 .842-.674 .528 .149-.154-.073-.927 .559 .624-.006-.330 .707 .902-.774 .287-.681
 .990 .550-.874 .184-.937-.223-.768 .051 .005-.346-.289-.447-.289 .114-.865 .038
-.786-.165 .817-.734 .039-.325 .230 .104 .843 .685 .851-.231-.267 .335-.292 .237
-.966-.728-.214 .499-.524-.332-.964 .330 .123 .721-.048-.495-.857-.030-.378 .022
-.521 .393-.736 .043 .038-.242 .964-.454-.046 .771-.014 .083-.773 .190 .045 .636
-.887 .216-.175 .822-.835-.173-.438-.590-.561 .760 .887-.145 .983-.091-.904-.725
-.807 .330 .243 .798 .319-.539-.982 .472-.429 .095 .600 .995 .790-.344-.601-.895
```

```
 .993 .004-.369-.652 .560-.634-.046 .835 .913 .549-.050 .262 .820-.097-.698-.527
 .528 .457 .841-.962 .154-.597-.013-.253 .235-.084 .068 .991-.420-.411-.618 .432
 .825-.094-.081 .342 .634-.933 .256 .740-.123 .811-.492-.365-.234 .911 .783 .403
-.496 .705-.766-.567-.124-.446 .938 .215 .735 .219 .344-.337 .461-.177 .111-.719
-.516-.343-.651-.224-.718 .038 .479-.096 .128-.763-.762 .051 .461-.980-.575-.950
-.845 .192-.453 .144 .074 .116-.211 .111 .415 .837-.801 .713-.564-.239 .965 .880
 .890 .659 .143 .471 .574-.998 .520 .112 .532-.346 .852-.442 .970-.672 .576 .104
-.575 .349-.202-.817 .762 .640-.393-.540-.330-.839 .397 .399 .274-.939-.485-.500
 .917-.775 .438-.082 .441 .024 .133-.115 .663 .994 .163 .533 .545 .370-.142-.530
-.211-.731-.073-.669-.264-.942 .890-.326 .582 .371-.860-.339-.538-.258 .190 .155
 .398 .873-.432 .800 .519-.299 .611-.287 .759-.826 .925-.551 .453-.080 .394 .290
 .887-.948-.963 .447-.094 .922-.755 .438 .728-.297-.473 .284-.020 .636 .029 .089
-.728 .769-.947 .591 .029-.075 .006 .692 .967-.091-.843 .816 .719 .519 .159-.118
 .842-.615-.592-.801 .161-.576 .955-.555-.586 .838-.937-.302-.812-.123-.234-.972
 .474-.117 .658-.192 .217-.867-.405 .853 .136-.602-.390-.654-.448-.597 .470 .239
-.542-.771 .335 .438 .683-.870 .964 .030-.528 .786 .370 .301 .829-.844-.523-.783
-.471 .722 .381 .949-.528-.989 .393-.660 .714-.944 .478 .221 .906 .945 .754-.976
 .280 .449-.600 .027-.199-.439 .653 .511 .198-.433 .418 .080-.150 .522-.390-.688
-.491-.130 .845-.334-.599 .560 .258 .570-.703 .061-.536-.619 .172 .883 .382-.773
 .694-.073-.053 .683 .271 .320-.225 .410 .149-.853-.518-.759 .040 .145 .505-.923
-.594 .994-.033 .945 .475 .301-.781-.981-.075-.565-.979 .176 .040 .385-.230-.057
-.960-.933-.641 .134 .843-.793-.328-.359 .641-.170 .096 .892 .901-.452 .992 .430
-.613 .751 .628 .894-.375 .144 .085 .428 .200-.683-.715 .429 .468 .081
INSTANTANEOUS PHASE
CONTINUOUS PHASE
INSTANTANEOUS FREQUENCY
CONTINUOUS PHASE (FIRST ORDER LEAST SQUARE FITTING)
    4  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1        0.72
 2.6         2.4       0.2
2.8         2.7        5.
    4  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1        0.62
 2.6         2.4       0.2
2.8         2.7        5.
    5  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1          0.52
 2.6         2.4       0.2
2.2          2.2       0.1
2.8         2.7        5.
    5  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1        0.42
 2.6         2.4       0.2
2.2          2.2       0.2
2.8         2.7        5.
    6  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1        0.32
 2.6         2.4       0.2
1.7         1.9         0.1
2.2          2.2       0.2
2.8         2.7        5.
    6  506     2   0.024     0.004
1.8         1.8         0.28
2.2         2.1           0.26
 2.6         2.4       0.2
1.7         1.9         0.16
2.2          2.2       0.2
2.8         2.7        5.
    6  506     2   0.024     0.004
1.8         1.8         0.23
2.2         2.1        0.22
 2.6         2.4       0.20
1.7         1.9        0.2
2.2          2.2       0.2
2.8         2.7        5.
```