NORSAR

ROYAL NORWEGIAN COUNCIL FOR SCIENTIFIC AND INDUSTRIAL RESEARCH

Norsar Technical Report No. 3/81

QUADRATIC VERSUS LINEAR ENVELOPE BEAMFORMING FOR SEISMIC EVENT DETECTION

by

Asgeir Nysæter

December 1981

NORSAR Contribution No. 309

QUADRATIC VERSUS LINEAR ENVELOPE BEAMFORMING

FOR SEISMIC EVENT DETECTION

A Thesis in Informatics

by

Asgeir Nysæter

University of Oslo

December 1981

CONTENTS

The work of this thesis has been conducted at NORSAR Data Processing Center (NDPC) at Kjeller, and I should like to express my gratitude to the whole NORSAR staff for their contribution to this thesis, and for some instructive years. Special thanks to Eystein S. Husebye and Frode Ringdal for skilled and stimulating guidance and help with the thesis work, to Jan Fyen for help with programs, and to Bernt Hokland for help with interpretation of detected events.

Kjeller, 1 December 1981

Asgeir Nysæter

Asgeir Nysæter

1. INTRODUCTION

Seismology has experienced major advances in the last twenty years, which have resulted in a considerable increase in knowledge concerning the structure and properties of the earth's interior. One reason for this is the large investments made in the subject in order to detect and identify nuclear explosions due to the political importance of monitoring a comprehensive test ban treaty. Some of the associated funding has gone into large seismic array construction, and one of these is NORSAR (Norwegian Seismic Array), completed in 1971 and located around Lake Mjøsa (Fig. 1.1). The present-day NORSAR configuration comprises 42 short periodic vertical seismometers, and 21 long periodic seismometers directed vertical, horizontal north-south and horizontal east-west. They are organized into 7 subarrays, with 6 short periodic and 3 long periodic seismometers in each, and with a subarray and array diameter of about 7 km and 60 km, respectively.

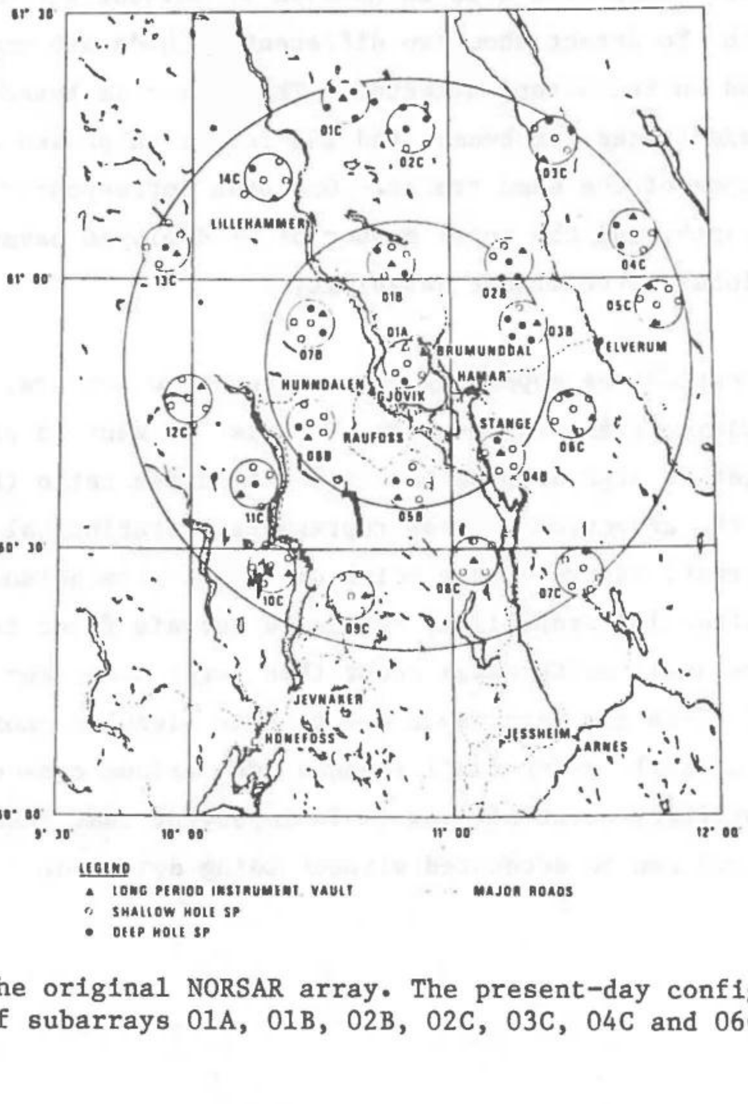


Fig. 1.1 The original NORSAR array. The present-day configuration consists of subarrays 01A, 01B, 02B, 02C, 03C, 04C and 06C.

From these seismometers the earth motion is recorded with 20 samples per second for the short periodic and 1 for the long periodic and transmitted via telephone lines to NORSAR data processing center (NDPC) at Kjeller. At NDPC the data is recorded on magnetic tapes by the Detection Processor (DP), which also processes the short periodic data in real time and decides possible signal arrivals in the ambient noise environments. In such cases relevant detector parameters are calculated and saved. Later on an off-line program, the Event Processor (EP) reads the recorded data tapes together with the detector reportings and checks more carefully the results from DP. The last stage in the process is an analyst checking the EP output before a seismological bulletin is published. For more information about the NORSAR system, see Bungum et al (1971).

The signals arriving at the array have been generated either by an earthquake or a man-made explosion. The source area is unknown in advance and can be everywhere on the earth. To detect them two different methods are employed, a coherent detector and an incoherent detector. The former is based on phased sums of amplitude traces, or beams, and the latter on phased sums of approximated envelopes of the same traces. One beam corresponds to a specific area on the earth, and the total number of DP deployed beams ensures an adequate global surveillance capability.

Apart from very local explosions appearing only on some few sensors, the goal is to detect as many signals as possible. That is, we want to process the incoming data to get as high as possible signal-to-noise ratio (SNR). In other words, since the detection process represents a statistical decision problem where the alternatives are either noise or signal plus noise, we want to maximize the detection probability against a certain fixed false alarm rate. As far more small earthquakes occur than large ones (see Fig. 1.2) and only a fraction of these are detectable due to poor signal-to-noise ratios, the lack of detection capability for small signals has various consequences. From a political and military point of view it is important that weak underground nuclear explosions can be detonated without being detected.

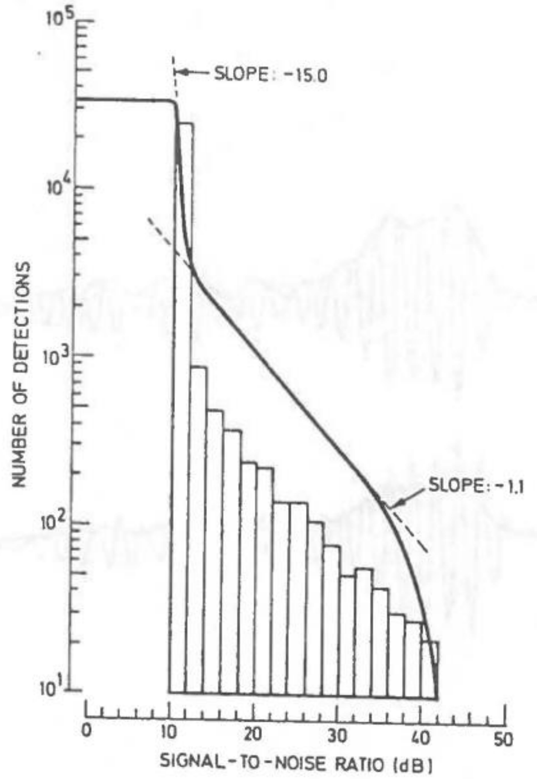


Fig. 1.2 Number of detections as a function of SNR reported by DP during the period July-December 1972. Around 10 dB the noise detections begin to influence more strongly, indicated by the turn-point in the slope.

The work of this thesis has been devoted to the development of an alternative real time incoherent detector. The essential feature of this is that the beams are based on weighted sums of square envelopes, which are obtained from complex data traces where the imaginary part has been generated by a Hilbert transformer. It is known (Ringdal et al, 1974) that the incoherent detector is generally superior to the coherent one in detecting regional events, which are characterized by relatively strong high frequency signal components of low coherency across the array.

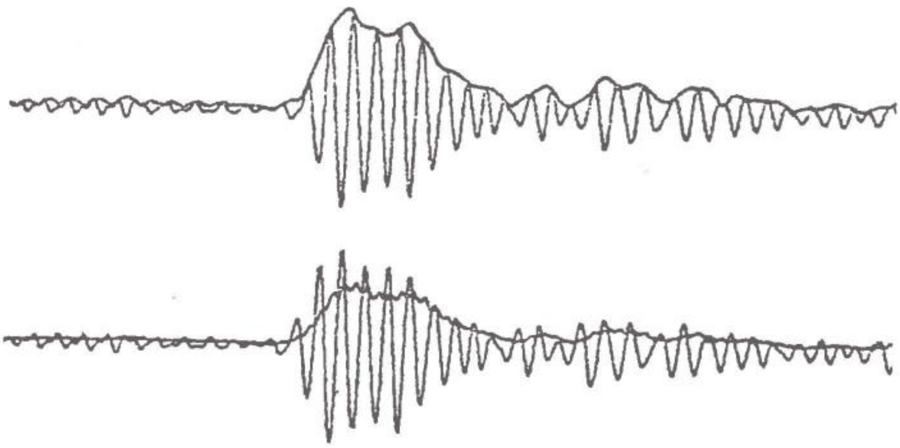


Fig. 1.3 Hilbert type envelope and STA type envelope of a seismic signal.

These regions include the Mediterranean area, Western Russia, the Pakistan-Afghanistan region, and the local area around the array, i.e., with an epicentral distance less than 30 degrees (1 degree ca 111 km). The starting hypothesis was that square envelope beamforming should prove to give better detectability than the currently used method, although requiring a higher threshold due to increased detector output variability in noise environments. Another reason to try an alternative lies in the fact that the present-day envelopes are generated in an approximate manner, by a running short term average (STA) of rectified amplitude samples, partly due to computer limitations when it was implemented. Experiments by Wen-Wu-Shen (1974) indicate that beamforming of Hilbert-generated envelopes gives better detection performance than STA-envelopes in the 1.5-2.5 and 3.0-4.0 Hz passband, and new generations of more powerful computers and decreased hardware prices opens up the possibility for real time processing with more complex algorithms. At NORSAR this will be a reality when DP has been converted from the old IBM 360/40 IBM SPS system to an IBM 4331 and MODCOMP Classic computer.

The two incoherent detectors, called the square envelope detector and the STA detector, have been programmed and tested on continuous short periodic data. Chapters 2 and 3 contain theoretical considerations concerning seismology, seismic arrays, and statistical methods and models underlying detector design. The effort of making the algorithms real time applicable was concentrated towards performing the digital filtering as fast as possible, and material about this is contained in chapter 4. The necessity of such an optimization before a performance evaluation was also partly justified by the great amount of data to be processed, requiring hours of CPU-time. Questions concerning the background noise are treated in chapter 5, and data results and performance comparisons in chapter 6. An elementary knowledge of probability theory, hypothesis testing and digital signal processing will be assumed. The same notation will be used both for a stochastic variable and its value at a particular time instant, and seismometer, sensor and instrument are used interchangeably.

2. FUNDAMENTALS OF SEISMIC ARRAYS

Some of the energy released by earthquakes or underground nuclear explosions are converted to elastic waves which travel outwards from the source area, following different paths according to the laws of wave propagation and conveying information about the seismic source and the medium through which they pass. In this chapter a basic presentation of seismic wave propagation, array beamforming technique and array capability will be given.

2.1 Seismic wave propagation

An infinite homogeneous elastic solid can support two wave types, called body waves. P waves propagate energy by means of motion normal to the wavefront, and S waves have motion in the plane containing the wavefront. Their nondispersive velocities are determined by the density and elastic properties of the medium, and are given by

$$V_p = \left(\frac{K+4n/3}{\rho}\right)^{\frac{1}{2}}$$

$$V_s = \left(\frac{n}{\rho}\right)^{\frac{1}{2}} \qquad (2.1)$$

where K and n are the incompressibility and rigidity, and ρ the density. The conditions in the earth's interior are a good approximation to such a medium. In many solid rocks $K = 5/3\ n$, which gives a ratio between V_p and V_s equal to $\sqrt{3}$. Thus P waves reach a seismic station before S waves from the same event. They are usually the first to arrive at the array when an event has occurred, and have typical maximum energy (at NORSAR) in the 1.0-3.0 Hz band for teleseismic events. The P and S phase, and other secondary phases generated from these by reflection and refraction in the earth are best recorded by the short periodic seismometers. In addition seismic events produce surface waves travelling along the surface of the earth. These are Rayleigh and Love waves, and have longer periods than the body waves. Surface waves are useful together with body waves for discriminating between earthquakes and explosions. In this thesis, however, we will be concerned with P-waves, as it is most efficient for detection of seismic events.

For P waves propagating through a set of plane stratified layers, the seismic ray, or normal to the wavefront, is subject to Snell's law, i.e.,

$$\frac{\sin i_1}{V_1} = \dots = \frac{\sin i_n}{V_n} = \text{constant}$$

where i_k, $k=1,\dots,n$ is the angle between the ray and normal to the k-th layering and V_k the k-th layer velocity. If we assume the layers in the earth to be spherically symmetric, a somewhat modified law comes out.

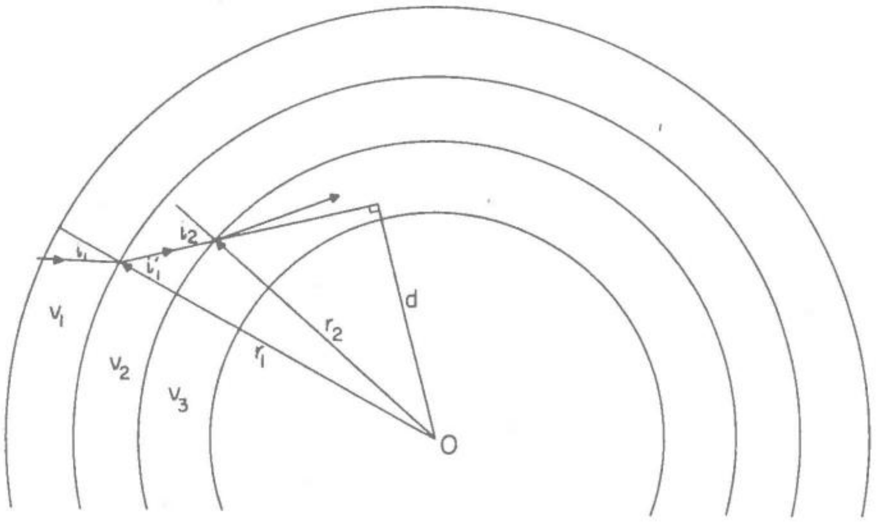


Fig 2.1 Successive refraction of a wave in a layered sphere.

Referring to Fig. 2.1, from Snell's law

$$\frac{\sin i_1}{V_1} = \frac{\sin i_1'}{V_2} \quad \text{or} \quad \frac{r_1 \sin i_1}{V_1} = \frac{r_1 \sin i_1'}{V_2}$$

Since $\dfrac{r_2}{\sin i_1'} = \dfrac{r_1}{\sin(\pi - i_2)} = \dfrac{r_1}{\sin i_2}$

we get $$\frac{r_1 \sin i_1}{V_1} = \frac{r_2 \sin i_2}{V_2} = p \qquad (2.2)$$

The constant p is called the ray parameter. Since the velocity usually increases with depth down to the core of the earth, the ray will follow a curve with the convex side outwards. p can be determined through another relationship.

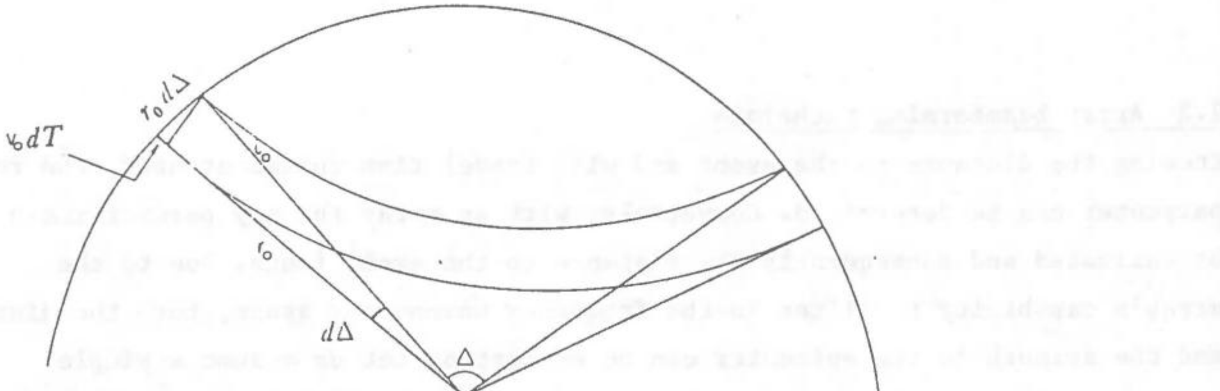


Fig. 2.2 Change in parameters for a small increase in Δ.

From Fig. 2.2

$$dT = \frac{r_o d\Delta \sin i_o}{v_o} = p\, d\Delta \quad \text{or} \quad p = \frac{dT}{d\Delta} \tag{2.3}$$

where r_o is the radius of the earth. In degrees ($\Delta^o = \frac{360}{2\pi}\Delta$)

$$\frac{dT}{d\Delta^o} = \frac{dT}{d\Delta}\frac{d\Delta}{d\Delta_o} = p\,\frac{2\pi}{360} \tag{2.4}$$

Thus p can be found as the slope of a curve giving the travel time as function of the angle distance between the seismometer station and surface point above the event (epicenter). Such curves are experimentally obtainable from a large number of events with known epicenters when the arrival time is observed at different seismic stations, and they give important information about the earth's inner structure. From Fig. 2.3 there are no observations beyond about 100^o, indicating that at such distances the waves travel through media with other velocity properties, the core of the earth. For our purposes it suffices to consider the earth as consisting of a crust, mantle and core (see Fig. 2.4).

2.2 Array beamforming technique

Knowing the distance to the event and with travel time curves at hand, the ray parameter can be determined. Conversely, with an array the ray parameter can be estimated and subsequently the distance to the event found. Due to the array's capability to filter in the frequency wavenumber space, both the distance and the azimuth to the epicenter can be estimated. Let us assume a simple harmonic plane wave is crossing the array. This wave model is useful to explain some principles and problems related to beamforming. At seismometer location $\underline{r}_\ell$ the wave can be expressed in complex notation as

$$s(\underline{r}_\ell,t) = A\, e^{j(\underline{k}\cdot\underline{r}_\ell - \Omega t)} \tag{2.5}$$

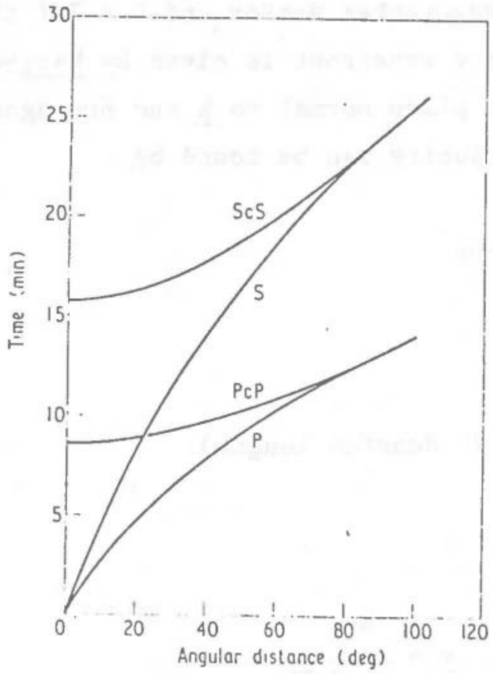


Fig. 2.3 Travel time diagram for P and S waves and their reflections from the core-mantle boundary, PcP and ScS.

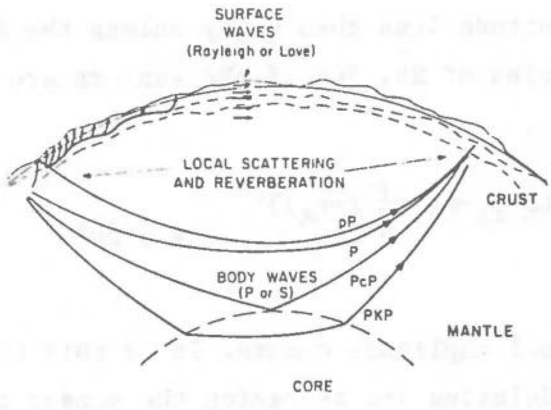


Fig. 2.4 A simple model of the interior of the earth with modes of propagation of seismic waves.

where $\underline{k} = (k_x, k_y, k_z)$ is the wavenumber vector and $\Omega = 2\pi f$ the angular frequency. The equation of the wavefront is given by $\underline{k}\cdot\underline{r}_\ell - \Omega t = c$, thus consisting of the points on a plane normal to $\underline{k}$ and propagating in the in the direction of $\underline{k}$. The velocity can be found by

$$\underline{k}\cdot(\underline{r}+\Delta\underline{r})-\Omega(t+\Delta t) = \underline{k}\cdot\underline{r}-\Omega t$$

$$\underline{k}\cdot\Delta\underline{r} = \Omega\Delta t$$

Since $\Delta\underline{r}$ is parallel to $\underline{k}$ ($\|\ \|$ denotes length)

$$\|\underline{k}\|\ \|\Delta\underline{r}\| = \Omega\Delta t$$

$$\|\underline{v}\| = \frac{\|\Delta\underline{r}\|}{\Delta t} = \frac{\Omega}{\|\underline{k}\|} \quad \text{and} \quad \underline{v} = \frac{\Omega}{\|\underline{k}\|^2}\underline{k} \qquad (2.6)$$

If the wave is simultaneously recorded at a number of locations, $\underline{r}_\ell$, $\ell = 1,\ldots,L$ and the outputs averaged, the resulting function will be

$$b(t) = A\ e^{-j\Omega t} \sum_{\ell=1}^{L} \frac{1}{L}\ e^{j\underline{k}\cdot\underline{r}_\ell} \qquad (2.7)$$

The sum will have a magnitude less than unity unless the difference between the $\underline{k}\cdot\underline{r}_\ell$ values are multiples of 2π. But if the sensors are delayed with $-(1/\Omega)\underline{k}\cdot\underline{r}_\ell$

$$b(t) = A \sum_{\ell=1}^{L} \frac{1}{L}\ e^{j(\underline{k}\cdot\underline{r}_\ell-\Omega(t+\frac{1}{\Omega}\underline{k}\cdot\underline{r}_\ell))} = A\ e^{-j\Omega t} \qquad (2.8)$$

and no reduction in signal amplitude occurs. It is this technique which is called beamforming. By delaying and averaging the sensor outputs as described, the array is steered for angular frequency Ω and wavenumber $\underline{k}$.

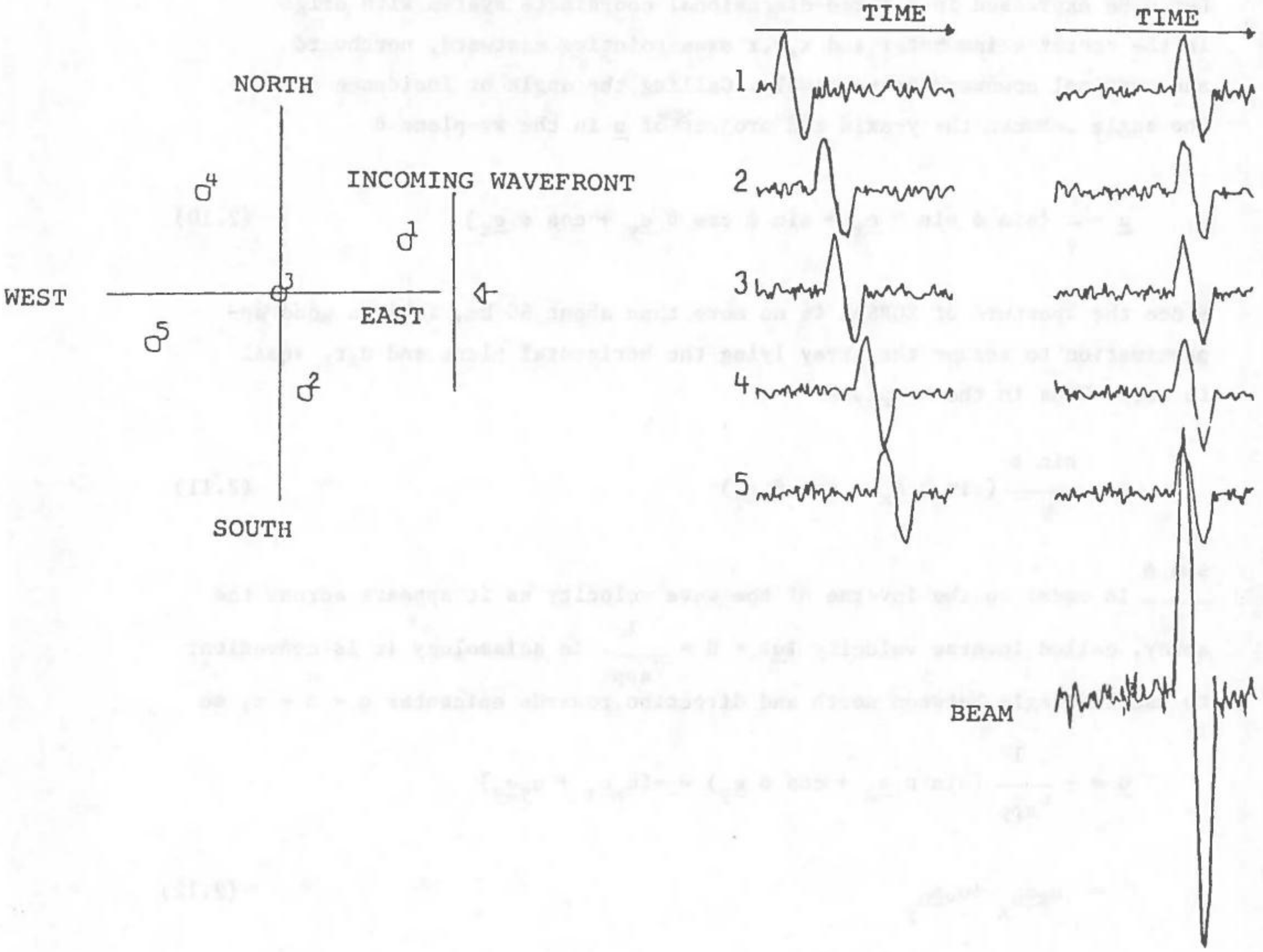


Fig. 2.5 Illustration of the beamforming. The beam is not averaged on this picture.

$1/\Omega\ \underline{k}$ points in the direction of the incoming wave, and the length of $1/\Omega\ \underline{k}$ is related to the ray parameter. From (2.6)

$$\frac{\|\underline{k}\|}{\Omega} = \frac{1}{\|\underline{v}\|} \quad \text{or} \quad u = \frac{1}{\Omega}\underline{k} = \frac{1}{V^2}\underline{v} \text{ where } V^2 = \|\underline{v}\|^2 \qquad (2.9)$$

Let $\underline{u}$ be expressed in a three-dimensional coordinate system with origo in the center seismometer and x,y,z axes pointing eastward, northward and vertical downward respectively. Calling the angle of incidence ϕ and the angle between the y-axis and projection of $\underline{u}$ in the xy-plane θ

$$\underline{u} = \frac{1}{V} (\sin \phi \sin \theta \, \underline{e}_x + \sin \phi \cos \theta \, \underline{e}_y + \cos \phi \, \underline{e}_z) \tag{2.10}$$

Since the aperture of NORSAR is no more than about 60 km, it is a good approximation to assume the array lying the horizontal plane and $u_z r_z$ equal to zero. Thus in the xy-plane

$$\underline{u} = \frac{\sin \phi}{V} (\sin \theta \, \underline{e}_x + \cos \theta \, \underline{e}_y) \tag{2.11}$$

$\frac{\sin \phi}{V}$ is equal to the inverse of the wave velocity as it appears across the array, called inverse velocity $\|\underline{u}\| = U = \frac{1}{V_{app}}$. In seismology it is convenient to use the angle between north and direction towards epicenter $\rho = \theta + \pi$, so

$$\underline{u} = - \frac{1}{V_{app}} (\sin \rho \, \underline{e}_x + \cos \rho \, \underline{e}_y) = -(u_x \underline{e}_x + u_y \underline{e}_y)$$

$$= u_x \underline{e}_{u_x} + u_y \underline{e}_{u_y} \tag{2.12}$$

What we have done is to move from a three-dimensional frequency wavenumber space to a two-dimensional inverse velocity space, corresponding to that of a nondispersive wave with velocity $V_{app} = \|\underline{v}_{xy}\|$ will be represented by the line

$$\underline{v}_{xy} = \frac{\Omega}{\|\underline{k}\|^2} \underline{k}_{xy}$$

in the frequency wavenumber space. Now the relation to the ray parameter should be clear, since U can be expressed as

$$U = \frac{dT}{dQ} = \frac{dT}{d\Delta}\frac{d\Delta}{dQ} = \frac{p}{r_o} \qquad (2.13)$$

where r_o is the radius of the earth. Thus the ray parameter can be found from the length of the $\underline{u}$ vector when the earth radius is known, and $\underline{u}$ points in the direction to the epicenter. A preselected set of such $\underline{u}$-space points with corresponding delays is employed in DP to cover the most interesting seismic regions.

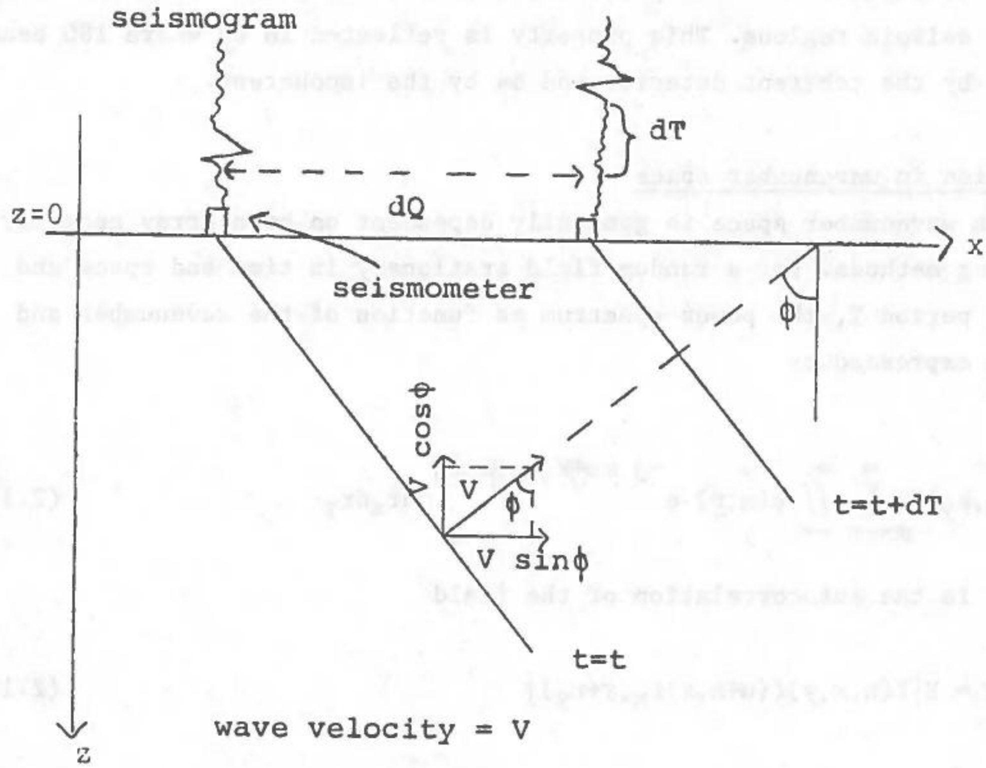


Fig. 2.6 Illustration of how the horizontal component of the wave velocity can be measured.

From (2.8) and Fig. 2.5 it can be observed that the array beam also acts as a signal estimate. Factors which reduce the estimate besides noise may be different signal amplitudes and frequency at the different sensors or deviations from a plane wavefront, due to inhomogeneities in the earth. In addition, signals may arrive from areas not covered by beams in DP. Sensitivity to the latter will depend on the lobe pattern or resolution in the frequency wavenumber space. Deviation from a plane wavefront is in practice eliminated by including regional corrections obtained from empiric accumulation of data. Variation in frequency can be compensated by discarding the phase information and instead summing the envelopes or square envelopes, i.e., incoherent beamforming. Intuitively we then lose resolution in the frequency wavenumber space due to slower variation of the envelope. However, the beamforming will be less sensitive to deviations from the preselected areas, and fewer beams are needed to cover the seismic regions. This property is reflected in DP where 180 beams are employed by the coherent detector and 64 by the incoherent.

2.3 Resolution in wavenumber space

Resolution in wavenumber space is generally dependent on both array geometry and processing methods. For a random field stationary in time and space and sampled with period T, the power spectrum as function of the wavenumber and frequency is expressed by

$$P(\omega,k_x,k_y) = \sum_{m=-\infty}^{\infty} \iint_{-\infty}^{\infty} \phi(m,\underline{r})\, e^{-j\, m\omega + jk_x r_x + jk_y r_y}\, dr_x dr_y \qquad (2.14)$$

where $\phi(m,\underline{r})$ is the autocorrelation of the field

$$\phi(m,\underline{r}) = E\left[f(n,x,y)f(n+m,x+r_x,y+r_y)\right] \qquad (2.15)$$

and ω is the discrete angle frequency, $\omega = 2\pi fT$. U sing a very narrow bandpass filter centered in $\omega_o = 2\pi f_o T = \Omega_o T$ and inverse velocity $\underline{u}_o = 1/\Omega_o\ \underline{k}_o$, we get for the autocorrelation of the beam

$$\phi_{bb}(m) = E[b(n)b(n+m)]$$

$$= E\Big[\sum_{\ell=1}^{L} \frac{1}{L} y_\ell\big(n + \frac{1}{T}\underline{u}_o \cdot \underline{r}_\ell\big) \sum_{i=1}^{L} \frac{1}{L} y_i\big(n+m + \frac{1}{T}\underline{u}_o \cdot \underline{r}_i\big)\Big]$$

$$= \frac{1}{L^2} \sum_{\ell,i=1}^{L} E\Big[y_\ell\big(n + \frac{1}{T}\underline{u}_o \cdot \underline{r}_\ell\big) y_i\big(n+m + \frac{1}{T}\underline{u}_o \cdot \underline{r}_i\big)\Big]$$

$$= \frac{1}{L^2} \sum_{\ell,i=1}^{L} \phi_{\ell i}\big(m + \frac{1}{T}\underline{u}_o \cdot (\underline{r}_i - \underline{r}_\ell)\big) \qquad (2.16)$$

where y_ℓ is the ℓ'th sensor output and $\phi_{\ell i}$ is the cross correlation between sensor ℓ and sensor i. The power spectrum is equal to

$$P_{bb}(\omega_o, \underline{k}_o) = \sum_{m=-\infty}^{\infty} \phi_{bb}(m)\, e^{-j\omega_o m}$$

$$= \frac{1}{L^2} \sum_{\ell,i=1}^{L} \sum_{m=-\infty}^{\infty} \phi_{\ell i}\big(m + \frac{1}{T}\underline{u}_o \cdot (\underline{r}_i - \underline{r}_\ell)\big)\, e^{-j\omega_o m}$$

$$= \frac{1}{L^2} \sum_{\ell,i=1}^{L} P_{\ell i}(\omega_o)\, e^{j\underline{k}_o \cdot (\underline{r}_i - \underline{r}_\ell)}$$

$$= \frac{1}{L^2} \sum_{\ell,i=1}^{L} e^{j\underline{k}_o \cdot (\underline{r}_i - \underline{r}_\ell)} \iint_{-\infty}^{\infty} P(\omega_o, \underline{k})\, e^{j\underline{k} \cdot (\underline{r}_\ell - \underline{r}_i)}\, d\underline{k}\, \frac{1}{4\pi^2}$$

$$= \iint_{-\infty}^{\infty} P(\omega_o, \underline{k}) \frac{1}{L^2} \sum_{\ell,i=1}^{L} e^{-j(\underline{k}_o - \underline{k}) \cdot (\underline{r}_\ell - \underline{r}_i)}\, d\underline{k}\, \frac{1}{4\pi^2}$$

$$= \iint_{-\infty}^{\infty} P(\omega_o, \underline{k})\, |B(\underline{k}_o - \underline{k})|^2\, d\underline{k}\, \frac{1}{4\pi^2} \qquad (2.17)$$

where $B(\underline{k}) = \frac{1}{L} \sum_{\ell=1}^{L} e^{-j\underline{k} \cdot \underline{r}_\ell}$ is defined as the beamforming array response pattern, and $P(\omega_0,\underline{k})$ is the wavenumber spectrum of the input field at frequency ω_0 before processing. In analogy with time domain, since $B(k)$ is a discrete Fourier transform (DFT) in space domain, more sensors and larger spacing between them gives better resolution due to decreased width of the main lobe. This also gives better noise suppression due to uncorrelated noise in space domain. On the other hand, large spacing reduces the signal coherency and introduces aliasing in space domain. At NORSAR the separation of instruments within a subarray ranges from 3.5 to 8 kilometers, and distances between the subarrays from 15 to 50 kilometers. For this separation the spatial coherency of the noise field is essentially zero for frequencies above 0.5 Hz (Ringdal & Bungum, 1977), and can be disregarded due to bandpassfiltering before beamforming. The signal coherency will vary depending on region, but is in general good for teleseismic regions. From (2.17) it should be noted that the lobe pattern is the same regardless of $\underline{k}_0$. Another method exists (the maximum likelihood method (Capon, 1973)) where the lobe pattern also depends on the spectral properties of the noise and thus gives different lobe patterns for different $\underline{k}_0$. This method gives better results if the noise is concentrated in the frequency wavenumber space. Note that the bulk of the noise energy originates from outside the array and propagates across the array as waves, in contrast to antennas where the noise is generated mostly within the receiver. But due to the spacing between the sensors, the noise shows a diffuse frequency wavenumber spectrum and in practice the conventional beamforming gives about the same signal-to-noise ratio as the maximum likelihood method.

Fig. 2.7 shows the beam pattern for the sensors used in this thesis, namely, the center seismometers from 01A, 02B, 02C, 03C, 04C and one additional seismometer from the 02B subarray to get 6 channels. The 01B and 06C subarrays were masked out at the time the experiments were done. From Fig. 1.1 it can be seen that this sensor geometry has largest extension in the northeast/southwest direction, and this is reflected in the beam pattern with best resolution in the same direction. Since the P waves are nondispersive, the signal envelope will travel with a velocity equal to the phase velocity. Due to lower frequency content in the envelope, the wavenumber spectrum will be

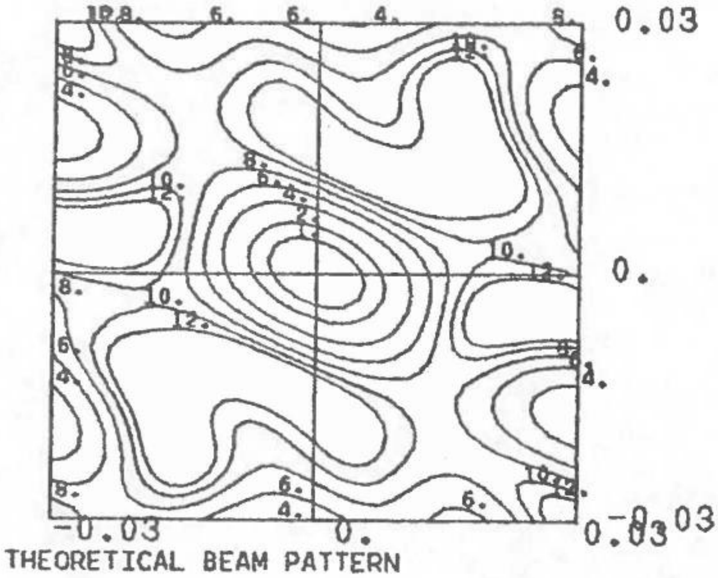


Fig. 2.7 The beam pattern for the sensors 01A00, 02B00, 02B01, 02C00, 03C00, 04C00. The numbers on the contours are suppresion in dB down from the maximum power.

much more concentrated towards the center in Fig. 2.7. In other words, the beam pattern will exhibit a much broader main lobe. This explains why incoherent beamforming is relatively insensitive to signals arriving from outside the preselected areas and that fewer beams are needed to cover the world. In addition the sample rate can be reduced and the location of the sensors more scattered without introducing aliasing in time or space.

3. DETECTOR DESIGN

As mentioned our problem is a statistical decision problem between the alternatives noise or signal plus noise, and we want to get the best possible signal detection probability against a certain fixed noise detection rate. Specially we want to detect more weak signals, as the strong ones are detected by most methods. The signal detection problem is not simple in seismology as little is known in advance about the signal. When an event occurs, energy is radiated over a broad frequency range in all directions from the source, and the signal shape in the different directions depends on both the earth structure at and around the source and the generating mechanism. The signal appearing at the sensors may be considered as a convolution between the original signal and a filter with impulse response determined from the path through the earth. In addition effects from local inhomogeneities are added. So instead of making assumptions on the signal shape, signal coherence is assumed across the sensors. This assumption leads to the coherent detector, and that of coherent signal envelope to the incoherent detector. It is known (Van Trees, 2.2) that both the minimum cost criterion (Bayes criterion) and a maximum probability criterion (Neyman-Pearson criterion) give likelihood ratio tests, so we want to base our detector statistic on such a ratio. We will assume the source area known in advance, and the traces delayed accordingly.

3.1 Likelihood ratio testing

Assume a test statistic η has been calculated as a function of the measurement values, $\eta = f(y_1,\ldots,y_n) = f(\underline{y})$, and denote by $p(\eta|H_o)$ the probability density of η when only noise is present, and by $p(\eta|H_1)$ the probability density of η when signal and noise are present. If η_o is the threshold value, the probability of a false alarm is given by

$$P_F = P(\eta>\eta_o|H_o) = \int_{\eta_o}^{\infty} p(\eta|H_o)\, d\eta \tag{3.1}$$

the probability of missed detection,

$$P_M = P(\eta<\eta_o|H_1) = \int_{-\infty}^{\eta_o} p(\eta|H_1)d\eta \tag{3.2}$$

and the detection probability or power of the test by

$$P_D = 1-P_M = P(\eta>\eta_o|H_1) = \int_{\eta_o}^{\infty} p(\eta|H_1)\,d\eta \tag{3.3}$$

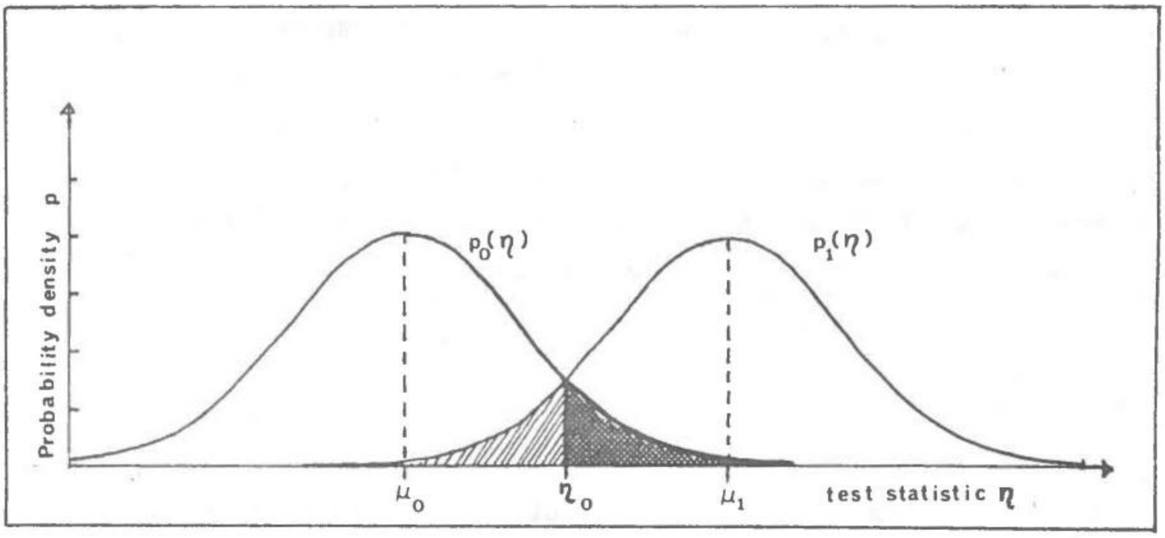


Fig. 3.1 The probability density of η when noise is true, $p_o(\eta)=p(\eta|H_o)$ and when signal plus noise is true, $p_1(\eta)=p(\eta|H_1)$, assuming that η under H_o is normal (μ_o,σ) and under H_1 normal (μ_1,σ). η_o denotes the threshold, the hatched field to the right of η_o the probability of false alarm, and the hatched field to the left of η_o the probability of missed detection.

Referring to Fig. 3.1 our desire is to construct a test where the densities under H_o and H_1 have the largest possible separation in order to make both P_F and P_M as small as possible. These are conflicting objectives, so instead we constrain P_F to be below a certain level and try to maximize P_D. Performing the optimization, we get (Van Trees, pp. 33-34)

$$L(\underline{y}) = \frac{p(\underline{y}|H_1)}{p(\underline{y}|H_o)} \tag{3.4}$$

where $p(\underline{y}|H_i)$ is the joint probability density of the measurement variables under hypothesis H_i. $\eta > \eta_o$ corresponds to $\underline{y}$ being in a subspace $S_1 \in \mathbb{R}^n$, and for $\eta < \eta_o$ $\underline{y} \in S_o = \mathbb{R}^n - S_1$. Intuitively the probability that $\underline{y} \in S_1$ is high when H_1 is true, and conversely when H_o is true.

When $p(\underline{y})$ in addition contains unknown paramters $\underline{\theta} = (\theta_1, \ldots, \theta_m)$, a logical extension is to make two likelihood estimates of $\underline{\theta}$, one for $\underline{\theta}$ constrained to the parameter subspace $\omega_1 \in \mathbb{R}^m$ corresponding to H_1, and one for $\underline{\theta}$ constrained to $\omega_o = \mathbb{R}^m - \omega_1$ corresponding to H_o. This results in the generalized likelihood rato test

$$L(\underline{y}) = \frac{\max\limits_{\underline{\theta} \in \omega_1} p(\underline{y}|\underline{\theta})}{\max\limits_{\underline{\theta} \in \omega_o} p(\underline{y}|\underline{\theta})} \tag{3.5}$$

3.2 Coherent detection

For the coherent signal model we may set up the following two alternatives:

$$\left.\begin{aligned} H_o &: y_\ell(n) = v_\ell(n) \\ H_1 &: y_\ell(n) = s(n) + v_\ell(n) \end{aligned}\right\} \quad \ell = 1, \ldots, L \quad n = 0, \ldots, N-1 \tag{3.6}$$

Here $y_\ell(n)$ and $v_\ell(n)$ denote the n'th input and noise sample at sensor ℓ respectively, and s(n) the n'th signal sample assumed equal across the sensors. The noise field samples are assumed stationary, normal $(0,\sigma)$, and uncorrelated in time and space.

$$E[v_\ell(n)v_i(m)] = \begin{cases} \sigma^2 & \ell = i \text{ and } n = m \\ 0 & \ell \neq i \text{ or } n \neq m \end{cases}$$

The signal amplitude is treated as the unknown parameter vector $\underline{s}$ = $(s(0),\dots,s(N-1))$. Under H_0 $\underline{s}$ will be the zero vector, $\omega_0 = \underline{0}$, and under H_1 $\underline{s} \in \omega_1 = \mathbb{R}^N - \underline{0}$, thus only one maximum likelihood estimate is necessary. Given the measurement vector $\underline{y}$, the function $\ell n\ p(\underline{y}|\underline{s})$ viewed as a function of $\underline{s}$ is called the log likelihood function, and the maximum likelihood estimate $\hat{\underline{s}}$ is found by

$$\frac{\partial}{\partial \underline{s}} \ell n\ p\ (\underline{y}|\underline{s}) = 0 \tag{3.7}$$

assuming the function differentiable and a global maximum in the interior of ω_1 exists. Thus inserting in the joint probability density function (PDF) for the noise samples and noting that uncorrelated Gaussian variables are independent, we get

$$\ell n\ p(\underline{y}|\underline{s}) = \ell n \prod_{\ell=1}^{L} \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi\sigma^2}}\ e^{-1/2\sigma^2 (y_\ell(n)-s(n))^2}$$

$$= -\frac{NL}{2} \ell n\ 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{\ell=1}^{L} \sum_{n=o}^{N-1} (y_\ell(n)-s(n))^2 \tag{3.8}$$

and

$$\frac{\partial}{\partial s(n)} \ell n\ p(\underline{y}|\underline{s}) = 0 \implies \sum_{\ell=1}^{L} (y_\ell(n)-s(n)) = 0 \quad n=0,\dots,N-1$$

$$\hat{s}(n) = \frac{1}{L} \sum_{\ell=1}^{L} y_\ell(n) \tag{3.9}$$

We see that simple beamforming is a maximum likelihood estimate of the signal when the above model is true. For the log likelihood ratio, we get

$$\ell n L(\underline{y}) = \ell n \frac{(2\pi\sigma^2)^{-NL/2} e^{-\frac{1}{2\sigma^2}\sum_{\ell=1}^{L}\sum_{n=o}^{N-1}(y_\ell(n) - \frac{1}{L}\sum_{i=1}^{L} y_i(n))^2}}{(2\pi\sigma^2)^{-NL/2} e^{-\frac{1}{2\sigma^2}\sum_{\ell=1}^{L}\sum_{n=o}^{N-1}(y_\ell(n))^2}}$$

$$= \frac{1}{2\sigma^2}\sum_{\ell=1}^{L}\sum_{n=o}^{N-1}\left[\frac{2}{L} y_\ell(n) \sum_{i=1}^{L} y_i(n) - (\frac{1}{L}\sum_{i=1}^{L} y_i(n))^2\right]$$

$$= \frac{1}{2\sigma^2}\sum_{n=o}^{N-1}(\frac{1}{L}\sum_{\ell=1}^{L} y_\ell(n))^2 \tag{3.10}$$

and for the test

$$\eta = \sum_{n=o}^{N-1}(\sum_{\ell=1}^{L} y_\ell(n))^2 \underset{H_o}{\overset{H_1}{\gtrless}} \eta_o \tag{3.11}$$

where η_o is the threshold determined from the distribution of the test statistic when only noise is present to get the desired false alarm rate. Note from (3.10) that the log likelihood ratio is proportional to an estimate of the power from the point in the u-space corresponding to the selected beam.

For the coherent detector running in DP the squaring in (3.10) is replaced by rectifying. This is faster to implement, while simulation of the detection performance showed the difference to be insignificant (Berteussen, 1972). N is 15 samples, and the sum is repeated each 5th sample for every beam, thus giving a running short term average (STA) for each preselected seismic area. Another sum is computed from an exponential weighted sum of STA's with decreased weights backwards in time, to get a running long term average of the noise (LTA).

$$\mathrm{STA}_c(n) = \sum_{i=o}^{N-1} \left| \sum_{\ell=1}^{L} y_\ell(n-i) \right|$$

$$\mathrm{LTA}_c(n) = \frac{1}{2}\,\mathrm{STA}_c(n-N) + (1-2^{-5})\mathrm{LTA}_c(n-N)$$

$$= \frac{1}{2} \sum_{i=1}^{\infty} (1-2^{-5})^{i-1}\mathrm{STA}_c(n-iN) \tag{3.12}$$

for each STA calculation, the test statistic

$$\eta = \frac{\mathrm{STA}_c(n)}{\mathrm{LTA}_c(n)} \tag{3.13}$$

is calculated and compared to a threshold. The noise field can be considered as wide sense stationary for short time intervals, but show both seasonal and diurnal fluctuations (Ringdal et al, 1977). The test statistic is insensitive to noise level fluctuations, but not to noise variance fluctuations, therefore the threshold is floating in order to attain an approximated constant false alarm rate (CFAR) receiver (Steinert et al, 1975). When η exceeds this threshold on one of the beams, a time window is set up, and the area giving the beam with the largest η value in the window is selected as signal source area. Prior to beamforming all channels are bandpass filtered with a digital third order Butterworth filter in the 1.2-3.2 Hz band. This frequency band has been found on average to give the best signal-to-noise ratio for the beam (Bungum et al, 1971).

The model in (3.6) is only an approximation to the real conditions across the array for the coherent signal case. Signal amplitude variations have been observed above 20 dB within the array (Berteussen, 1975), the noise variance varies with the sensor locations, and the noise is not uncorrelated in time. Nevertheless the STA/LTA test based on simple beams (DS processing)

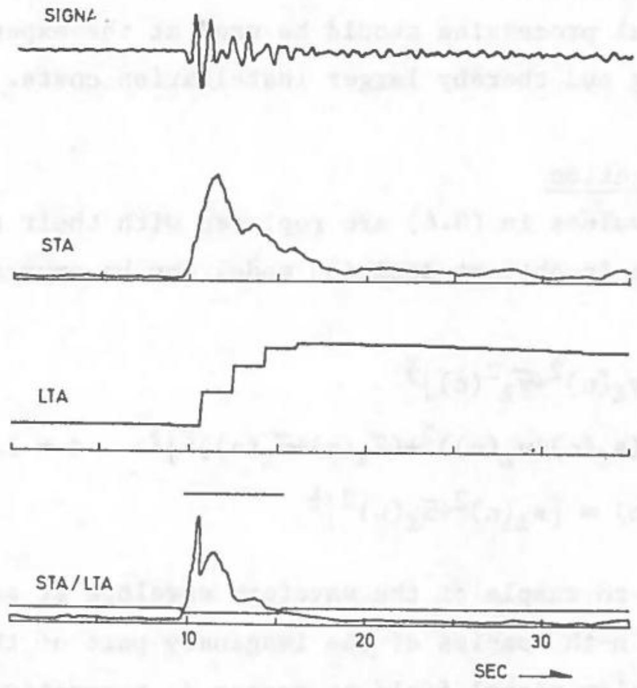


Fig. 3.2 The principle of the STA/LTA detector. The short line above the STA/LTA curve indicates detection state, and the line crossing the curve is the threshold. Note that the time window for the significant contributing part of the LTA estimate is decreased when the threshold is exceeded.

have been kept in operation because other tests based on more realistic models either have required too much computing time or have not proved to be any better. Fyen (1978) obtained with the model $y_\ell(n) = h_\ell s(n)+v_\ell(n)$ better signal estimates, but also higher noise variance, so SNR was not enhanced for an STA/LTA test based on this model. Other tests based on the same model gave better performance, but are too time critical for real-time operations as the pattern of the amplitude weights will be altered when the signal power is about equal to or lower than the noise power. Berteussen (1972) tried with a prewhitening filter, but the increase in detection performance was not significant. A likelihood test based on correlated noise certainly gives better SNR when the sensor separation is small, but at a sensor separation of 3 km the SNR gain with simple beamforming is about equal to the more complicated filter and sum (FS) processing (Capon, 1973). Both NORSAR and LASA (Large Aperture Seismic Array) in the USA were designed so that simple signal processing should be used at the expense of larger interelement spacing and thereby larger installation costs.

3.3 Incoherent detection

When the amplitude values in (3.6) are replaced with their respective envelope values, the incoherent decision model can be expressed as

$$\begin{aligned} H_o &: z_\ell(n) = \left[v_\ell(n)^2+\tilde{v}_\ell{}^2(n)\right]^{\frac{1}{2}} \\ H_1 &: z_\ell(n) = \left[(s_\ell(n)+v_\ell(n))^2+(\tilde{s}_\ell(n)+\tilde{v}_\ell(n))^2\right]^{\frac{1}{2}} \qquad \ell = 1,\ldots,L \\ & \qquad r(n) = \left[s_\ell(n)^2+\tilde{s}_\ell(n)^2\right]^{\frac{1}{2}} \end{aligned} \tag{3.14}$$

$z_\ell(n)$ denotes the n-th sample of the waveform envelope at sensor ℓ, $\tilde{v}_\ell(n)$ and $\tilde{s}_\ell(n)$ the n-th samples of the imaginary part of the complex noise field and complex signal field at sensor ℓ, respectively, and $r(n)$ the n-th sample of the signal envelope assumed equal across the sensors. If the signal is represented in terms of its instantaneous envelope and phase as $s(n) = r(n)\cos\phi(n)$, (3.14) means that the signal phase is not longer considered to be equal across the array and therefore discarded for detection purposes. It is natural to extend to the complex domain by representing the imaginary part of the signal as $\tilde{s}(n) = r(n)\sin\phi(n)$, and the envelope is then obtained by $r(n) = \left[s(n)^2+\tilde{s}(n)^2\right]^{\frac{1}{2}}$. How to generate the complex samples from the real ones is treated in chapter 4 and appendix A. The observation interval in (3.14) has been decreased to one sample or to the instantaneous value. From Fig. 1.3 and Fig. 3.2 the STA operator can be viewed as giving a time delayed instantaneous envelope value, and our desire is to try an alternative to the STA operator. From appendix C with the same assumptions for the noise as in the coherent case, $z_\ell(n)$ is Rayleigh distributed under H_o and Rice distributed under H_1. So for the log likelihood ratio ($\underline{z}(n) = (z_1(n),\ldots,z_L(n))$:

$$\ell n L(\underline{z}(n)) = \ell n \frac{p(\underline{z}(n)|H_1)}{p(\underline{z}(n)|H_o)}$$

$$= \ell n \frac{\prod_{\ell=1}^{L} \frac{z_\ell(n)}{\sigma^2} e^{-\frac{z_\ell(n)^2+r(n)^2}{2\sigma^2}} I_o\left(\frac{z_\ell(n)r(n)}{\sigma^2}\right)}{\prod_{\ell=1}^{L} \frac{z_\ell(n)}{\sigma^2} e^{-\frac{z_\ell(n)^2}{2\sigma^2}}}$$

$$\sum_{\ell=1}^{L} \ell n\, I_o\left(\frac{z_\ell(n)r(n)}{\sigma^2}\right) - \frac{L}{2\sigma^2} r(n)^2 \qquad (3.15)$$

$I_o(x)$ is the modified Bessel function of first kind and zero order

$$I_o(x) = \frac{1}{2\pi}\int_o^{2\pi} e^{x\cos\theta}\, d\theta = \frac{1}{2\pi}\int_o^{2\pi} \sum_{k=o}^{\infty} \frac{(x\cos\theta)^k}{k!}\, d\theta$$

$$= 1 + \frac{x^2}{4} + \frac{x^4}{64} + \ldots \qquad (3.16)$$

Since $r(n)$ and σ^2 are unknown and the calculation of $\ell n I_o(\ )$ requires too much computer time, an approximation must be done to (3.15). For $r(n) \ll \sigma$, $z_\ell(n)$ wil have a distribution near to the Rayleigh distribution, and

$$E\left[\frac{z_\ell(n)r(n)}{\sigma^2}\right] \approx \frac{r(n)}{\sigma^2}\sqrt{\pi/2}\,\sigma \approx 1.25\,\frac{r(n)}{\sigma}$$

$$\mathrm{Var}\left[\frac{z_\ell(n)r(n)}{\sigma^2}\right] \approx \frac{r(n)^2}{\sigma^4}\,\frac{4-\pi}{2}\sigma^2 \approx 0.43\,\frac{r(n)^2}{\sigma^2} \qquad (3.17)$$

The argument to $\ell n\ I_o(\ )$ will be on the quadratic part of the curve in Fig. 3.3

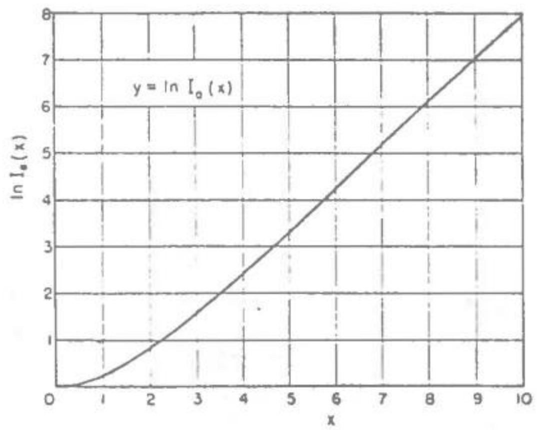


Fig. 3.3 Plot of $\ell n\ I_o(x)$.

$$\ell n\ I_o(x) \approx \ell n(1 + \frac{x^2}{4}) \approx \frac{x^2}{4} \qquad (3.18)$$

Since our main interest is to improve the weak signal detectability, we do the approximation in (3.18) and get

$$\ell n\ L(\underline{z}(n)) \approx \sum_{\ell=1}^{L} \frac{z_\ell(n)^2 r(n)^2}{4\sigma^4} - \frac{L}{2\sigma^2}\, r(n)^2 \qquad (3.19)$$

The test statistic becomes

$$\eta = \sum_{\ell=1}^{L} z_\ell(n)^2 \underset{H_o}{\overset{H_1}{\gtrless}} \eta_o \qquad (3.20)$$

Thus our best detector in the weak signal case when the model in (3.14) is true and the memory reduced to the instantaneous value consists of beamforming the square envelopes from the individual sensor waveforms. The value of the signal envelope has been included in the threshold which can be determined independently from the wanted false alarm rate and detector distribution for noise input. For stronger signals it can be seen from Fig. 3.3 that a linear envelope detector will be a better approximation to the optimum detector. The square envelope detector was preferred in this thesis due to less computing time and theoretically better small signal detectability.

The incoherent model is not strictly valid either. As in the coherent case, the noise variance and the signal envelope strength will vary across the sensors. Nevertheless it is more correct to assume equal signal envelopes across the array than equal signal amplitudes. Ringdal et al (1975) found a significantly higher average correlation for the former, 0.90-0.95 against 0.50-0.75 units, respectively, and these authors also give a theoretical explanation why a high signal envelope similarity can be expected. It could certainly be possible to utilize a possible distribution of the signal envelope values across the sensors to improve the detection performance (the signal amplitudes seem to be lognormally distributed (Ringdal et al, 1974)), or try signal envelope weights similar to the coherent case, but it will not be considered.

As mentioned in Chapter 2, the spatial uncorrelatedness of the noise field will be assured with a bandpass filter with lower cutoff frequency above 0.5 Hz, and the seismometers are regularly calibrated from a console at NDPC to give zero mean value. The correlation in time is no problem since only instantaneous values are used. The assumption of normality of the noise is quite natural in the seismic case, as it is composed of many independent sources and therefore can be expected to behave according to the Central Limit Theorem. In chapter 5 some selected intervals of noise samples have been tested to investigate the question of normality more completely.

The combination of an LTA operator and a floating threshold to control the noise variability will not be employed in this thesis. Instead the square envelope samples are divided by a time-delayed running variance estimate of the noise from the same channel to get equal, normalized noise distributions across the channels and stationary detector output for noise input. Since the Fast Fourier Transform (FFT) are used for block filtering of the data, it is easy to establish a finite time window, and no weighting is necessary. The normalized square envelope values can be expressed as

$$t_{\ell}(n) = z_{\ell}(n)^2 \omega_{\ell}(n) \tag{3.21}$$

where $\frac{1}{\omega_{\ell}(n)} = \hat{\sigma}_{\ell}^2(n) = \frac{1}{N}\sum_{i=o}^{N-1} y_{\ell}(n-k-i)^2$ is the variance estimate, k the estimate delay, and N the number of samples in the estimate. Since the same estimate is used for all samples in one block, k will be different for the different samples in the block and decrease backwards in time. If the noise is stationary and normal $(0,\sigma_{\ell})$ for the time interval used

$$E\left[\hat{\sigma}_{\ell}{}^2(n)\right] = \frac{1}{N}\sum_{i=o}^{N-1} E\left[y_{\ell}(n-i-k)^2\right] = \frac{1}{N} N\sigma_{\ell}{}^2$$

so $\hat{\sigma}_{\ell}{}^2(n)$ is an unbiased estimator of $\sigma_{\ell}{}^2$. From Oppenheim, Schaeffer pp. 540, $\hat{\sigma}_{\ell}{}^2(n)$ is also a consistent estimator when max correlation lag is finite, so with N actually used equal to 1808, we will set $\hat{\sigma}_\ell^2(n) \approx \sigma_\ell^2$. Thus for the probability density of the normalized noise square envelope

$$t_\ell \approx \frac{z_\ell^2}{\sigma_\ell^2}$$

$$z_\ell \approx \psi(t_\ell) = \sigma_\ell\sqrt{t_\ell} \qquad z_\ell' \approx \psi_\ell'(t_\ell) = \frac{\sigma_\ell}{2}\frac{1}{\sqrt{t_\ell}}.$$

$$p_{T_\ell}(t_\ell) \approx p_{Z_\ell}(\psi(t_\ell))\psi'(t_\ell) = \frac{\sigma_\ell\sqrt{t_\ell}}{\sigma_\ell^2}\, e^{-\frac{\sigma_\ell^2 t_\ell}{2\sigma_\ell^2}}\, \frac{\sigma_\ell}{2}\frac{1}{\sqrt{t_\ell}}$$

$$= \begin{cases} \frac{1}{2} \quad e^{-t_\ell/2} & t_\ell > 0 \\ 0 & t_\ell \leq 0 \end{cases} \tag{3.22}$$

We see that t_ℓ is approximately chi-squared distributed with 2 degrees of freedom when the above assumptions are satisfied. If the shape of the noise probability density also changes, the question of robustness will be of importance. We will, however, as a starting hypothesis consider our detector as an approximate CFAR-receiver. Experiments recently performed by Unger (1981) with linear envelope samples divided by a running root mean square (RMS) estimate of the noise support this hypothesis, and the method from DP cannot be used here since the threshold values are precalculated from an experimentally established regression line (Steinert et al, 1975).

We define the signal-to-noise ratio for the square envelope detector as the maximum detector value within the signal window divided by the average of the detector output in noise, when the source area has been found. Since the detector output in noise is expected to be approximately chi-square distributed with degrees of freedom equal to the double of the numbers of sensors used, we get

$$SNR_{SE} = \max_{n} \sum_{\ell=1}^{L} z_\ell(n)^2 \omega_\ell(n)/2L \tag{3.23}$$

This definition is used since it corresponds to the SNR definitions for other detectors used for detection of seismic signals. A better measure of SNR is called the detectability of the detector, defined as the maximum detector output within the signal window subtracted from the average of the detector output in noise and divided by the standard deviation of the detector output innoise. A chi-squared distributed variable with 2L degrees of freedom has variance 4L, so the detectability for the square envelope detector can be expressed as

$$D_{SE} = \max_{n} \frac{\sum_{\ell=1}^{L} z_\ell(n)^2 \omega_\ell(n) - 2L}{2\sqrt{L}} \tag{3.24}$$

In the incoherent detector running in DP the beamforming and rectification is exchanged, so for the incoherent STA

$$STA_I(n) = \sum_{\ell=1}^{L} \sum_{i=o}^{N-1} |y_\ell(n-i)| = \sum_{\ell=1}^{L} STA_\ell(n) \tag{3.25}$$

Since we want as much as possible equal detection methods, the LTA-operator has been replaced by independent running channel noise standard deviation estimates on our off-line version to get the following detector statistic

$$\eta = \sum_{\ell=1}^{L} \omega_\ell(n) \sum_{i=o}^{N-1} |y_\ell(n-i)| \tag{3.26}$$

where $\frac{1}{\omega_\ell(n)} = N\hat{\sigma}_\ell(n) = \frac{N}{M} \sum_{m=o}^{M-1} |y_\ell(n-k-m)|$, N is the length of the envelope window, M the length of the standard deviation estimate window, and k the estimate delay. Note that actually two standard deviation estimates are performed on each channel, one with a short window and the other with a long window. Since M also was 1808, we will assume $\hat{\sigma}_\ell(n) \approx \sigma_\ell$. From elementary probability calculation, the expectation of the absolute value of a normal $(0,\sigma)$ variable is $\sqrt{2/\pi}\,\sigma$, thus for the expectation of the detector statistic in noise

$$E\left[\sum_{\ell=1}^{L} \omega_\ell(n) \sum_{i=o}^{N-1} |y_\ell(n-i)|\right] \approx E\,\frac{\frac{1}{N}\sum_{i=o}^{N-1} |y_\ell(n-i)|}{\sqrt{\frac{2}{\pi}}\,\sigma_\ell}$$

$$= \frac{\sqrt{\pi/2}\,\sigma_\ell}{\sqrt{\pi/2}\,\sigma_\ell} = 1 \tag{3.27}$$

SNR can therefore be defined by

$$SNR_I = \max_n \sum_{\ell=1}^{L} \omega_\ell(n) \sum_{i=o}^{N-1} |y_\ell(n-i)|/L \tag{3.28}$$

Since theoretical distributions for sums of absolute values of dependent normal variables cannot be calculated (Berteussen, 1972), we do not known the theoretical distribution for this detector in noise and cannot establish a detectability measure equal to (3.24). This explains the above definition of SNR since a running standard deviation estimate for the detector output in noise is not usually calculated for an on-line detector. However, in off-line analysis denoting the standard deviation estimate of the detector output in noise by $\hat{\sigma}_{I,noise}$

$$D_I = \max_n \frac{\sum_{\ell=1}^{L} \omega_\ell(n) \sum_{i=o}^{N-1} |y_\ell(n-i)| - L}{\hat{\sigma}_{I,noise}} \tag{3.29}$$

Essentially two different methods are used to compare the detectors. If thresholds can be established giving equal false alarm rates and the detectors have stationary output in noise, the number of signals detected by each can be counted and compared. The other method is more indirect and consists of comparing the detectability of the detectors on signals detected by both. This however requires the detectors to have equal statistical distributions for noise input. Note that it is not possible to do any theoretical comparison due to lack of theoretical knowledge about the STA detector's statistical behavior. We have assumed that the weighting gives stationary detector output in noise, but the answer to this and the performance comparisons will be answered from data analysis in chapter 6. First, however, we will digress to say more about digital filtering and the statistical properties of the complex noise field.

4. DIGITAL FILTERING

Different filters will generally be present in seismic detection systems. The seismometer itself acts as a filter, with a characteristic response, and the seismometer output is analog lowpass filtered before sampling to avoid aliasing (see Fig. 4.1). The digital filter used together with STA-calculation, Hilbert transforming, beamforming, etc., in the detection process will have as aim to enhance the signal-to-noise ratio (SNR). This is possible because the signal and noise spectra will differ from each other. It is known from the literature (Helstrom) that a noise prewhitening filter in cascade with a signal matching filter maximizes SNR when the signal and noise spectra are known. The matching filter has a response identical to the 'whitened' signal spectrum. Unfortunately, it is a well known fact for NORSAR and other seismic arrays that the noise spectrum is not stationary, and signal spectra may vary considerably from one event to another. Thus practical experiments with different signal and noise conditions must be carried out to find the best choice.

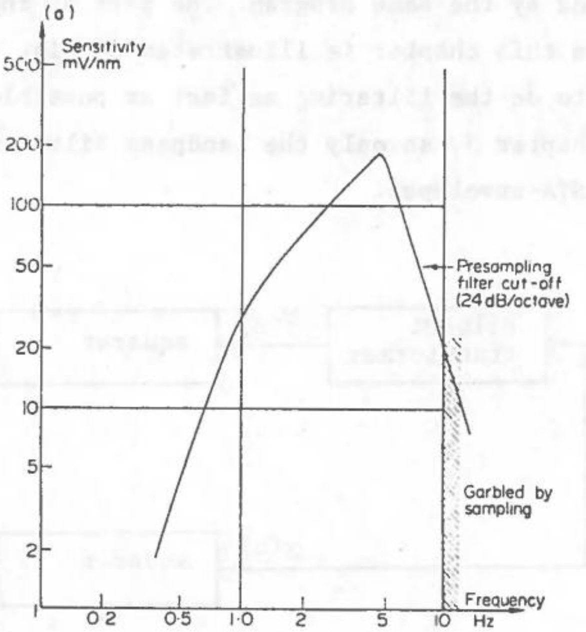


Fig. 4.1 Short period system response. The Nyquist frequency is 10 Hz. The 1A00, 2B00, 2C00 and 4C00 seismometers use nonstandard 8.0 Hz analog lowpass filters, while the 2B01 and 3C00 seismometers use standard 4.75 Hz lowpass filters.

Bandpass filters have been found to on average give highest detectability (IBM Seismic Array Design Handbook). In DP a third order Butterworth filter with passband 1.2-3.2 Hz and 1.6-3.6 Hz are used for the coherent and incoherent detector, respectively, so we will use the 1.6-3.6 Hz band in the bulk of the experiments. It is known that the signal coherency across the sensors decreases with increasing frequency, while SNR on a single trace increases with increasing frequency. Since signal coherency is not critical in incoherent beamforming, the passband is moved upwards to take advantage of higher single trace SNR.

The imaginary part of the trace has been generated by convolving the real part with a Hilbert transformer. To calculate the filter coefficients, a program made by McClellan, Parks and Rabiner (1973) has been used. The program calculates coefficients for linear phase finite impulse response (FIR) filters, and is based on Chebyshev approximation which is optimum in the sense that it minimizes the maximum error in the specified bands. It is capable of designing many types of FIR filters, therefore the bandpass filter coefficients have been calculated by the same program. The part of the detector algorithm which is treated in this chapter is illustrated by Fig. 4.2, and the main part is devoted to how to do the filtering as fast as possible. The STA-type envelope was described in Chapter 3, so only the bandpass filtering has sense for the detector based on STA-envelopes.

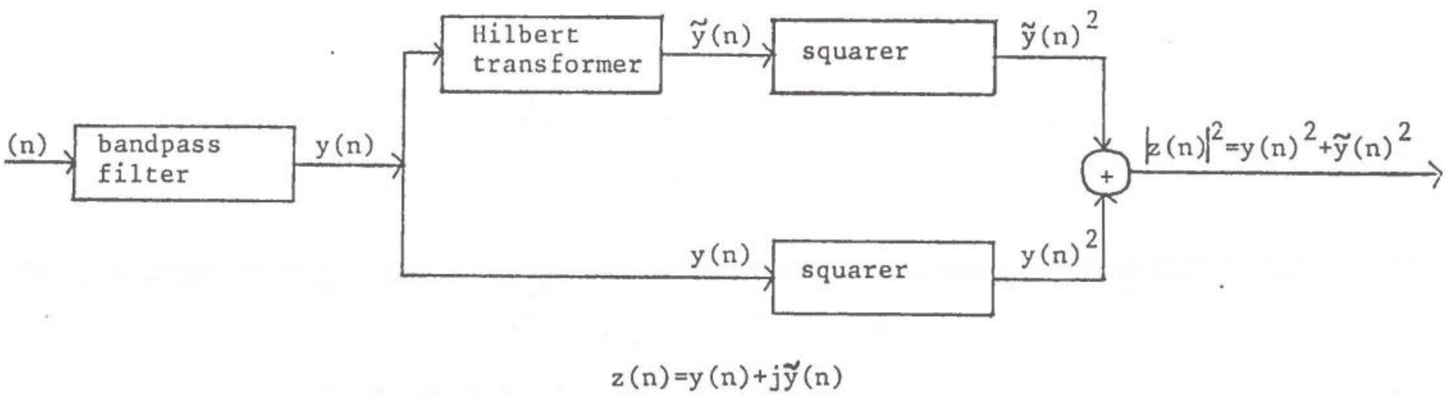


Fig. 4.2 Block diagram of the system that transforms an unfiltered input sequence into its filtered square envelope sequence.

4.1 The discrete Hilbert transformer, and discrete representation of the real and imaginary bandpass filtered data trace

The Hilbert transformer for discrete data sequences is defined to have the same values in the frequency domain as in the continuous case (see Appendix A):

$$H(e^{j\omega}) = \begin{cases} -j & 0 < \omega \leq \pi \\ 0 & \omega = 0 \\ j & -\pi < \omega < 0 \end{cases} \qquad H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h(n)\, e^{-j\omega n} \tag{4.1}$$

ω = digital angle frequency = $2\pi fT$.

If we denote by $\tilde{y}(n)$ the convolution between the Hilbert transformer and a bandpass filtered sequence $y(n)$, the representation

$$z(n)=y(n)+j\tilde{y}(n)=y(n)+j\sum_{k=-\infty}^{\infty} h(k)y(n-k)$$

will due to the discrete convolution theorem have the following frequency properties:

$$Z(e^{j\omega}) = \begin{cases} 2Y(e^{j\omega}) & 0 < \omega < \pi \\ Y(e^{j\omega}) & \omega = 0 \\ 0 & -\pi < \omega < 0 \end{cases} \qquad Y(e^{j\omega}) = \sum_{n=-\infty}^{\infty} y(n)\, e^{-j\omega n} \tag{4.2}$$

With those properties, the results presented in Appendix A also apply in the discrete case. Thus if the bandpass filter pass frequencies in the band ω_o to $\omega_o + \omega_1$, a general expression for $y(n)$ is given by

$$y(n) = r(n)\cos(\omega_o n + \phi(n)) \tag{4.3}$$

and for $\tilde{y}(n)$

$$\tilde{y}(n) = r(n)\sin(\omega_o n + \phi(n)) \tag{4.4}$$

where $r(n)\, e^{j\phi(n)}$ is a lowpass signal.

We see that the Hilbert transformer gives a natural extension into the complex domain, since

$$z(n) = y(n) + j\tilde{y}(n) = r(n)\, e^{j(\omega_o n+\phi(n))} \tag{4.5}$$

The square envelope and instantaneous phase can be calculated as

$$r(n)^2 = y(n)^2 + \tilde{y}(n)^2$$

$$\omega_o n+\phi(n) = tg^{-1}\,\frac{\tilde{y}(n)}{y(n)} \qquad\qquad \omega_o n+\phi(n) \in (-\pi,\pi] \tag{4.6}$$

4.2 Linear phase FIR filters designed with Chebyshev approximation

Because of the discontinuity at $\omega = 0$, the ideal Hilbert transformer is an unrealizable filter. Likewise from Appendix A the discrete Hilbert transformer (DHT) is not suited for block filtering of infinite data sequences. Thus, both the bandpass filter and Hilbert transformer must have frequency responses which are approximations to an ideal response. The Hilbert transformer will have transition bands around $\omega = 0$ and $\omega = \pi$. If the lower and upper transition widths of the Hilbert transformer are set equal, and equal to the shortest stopband width of the bandpass filter, the approximation will still have the same effect as a full band Hilbert transformer. This is possible with the design technique used in this thesis. In addition, depending on the width and location of the passband of the bandpass filter, it is possible to design transformers requiring very few mulitiplications per input sample. It is due both to an inverse proportionality relation between transition width and filter length for fixed maximum deviation from the ideal response (Rabiner & Schafer, 1974), and other properties which will be demonstrated.

The impulse response coefficients h(n) for FIR linear phase filters satisfy the following symmetry condition

$$h(n) = \pm\, h(N-1-n) \qquad\qquad n = 0,\ldots,N-1 \tag{4.7}$$

where N is the filter length. Bandpass filters have equal symmetry since a real ideal frequency response is approximated, and a pure imaginary ideal frequency response gives odd symmetry Hilbert transformers. From a computational point of view the symmetry property is valuable since it reduces the number of multiplications in time domain filtering with one half. The linear phase property implies no signal distortion, and exactly known delay for the filter output. The finite impulse response length further implies stability and the option for frequency domain filtering without introducing aliasing effects. In addition, efficient algorithms are available to design these filters with very good frequency response characteristics.

These properties also make them interesting from a seismological viewpoint. An exactly known filter output delay will eliminate one of the uncertainties in the process of determining precise signal arrival times across the array. This is a very important task for off-line source localization. Parameters as dominant period, velocity, etc. are best determined with an undistorted waveform, and a good fit to an ideal frequency response give better control over the frequency content in the filtered waveform.

The method of Chebyshev approximation for linear phase FIR filters is explained in Appendix B. Here we will demonstrate it on Hilbert transformer design and show that every other filter coefficient can be designed equal to zero.

Only odd length transformers are considered, since the calculation of the square envelope requires the output sequence to be delayed an integer number of samples. For $N_H = 2M_H+1$ the Hilbert transformer frequency response can be expressed as ($h(M_H) = 0$):

$$H(e^{j\omega}) = \sum_{n=o}^{N_H-1} h(n)e^{-j\omega n}$$

$$= e^{-jM_H\omega} \left[\sum_{n=0}^{M_H-1} h(n)(e^{-j\omega(n-M_H)} - e^{-j\omega(M_H-n)}) \right]$$

$$= je^{-jM_H\omega} \sum_{n=1}^{M_H} a(n) \sin(\omega n)$$

$$= je^{-jM_H\omega} G(\omega) \qquad (4.8)$$

$$a(n) = 2h(M_H-n) \qquad n=1,\ldots,M_H \qquad (4.9)$$

Assume $G(\omega) = G(\pi-\omega)$. Then

$$\sum_{n=1}^{M_H} a(n)\sin(\omega n) - \sum_{n=1}^{M_H} a(n)\sin[(\pi-\omega)n] =$$

$$\sum_{n=1}^{M_H} a(n)\sin(\omega n)[1+(-1)^n] = 0 \qquad (4.10)$$

This implies $a(n) = 0$, $n=2,4,\ldots$, and

$$h(n) = 0 \begin{cases} n = 0,2,\ldots,N_H-1 & M_H \text{ even} \\ n = 1,3,\ldots,N_H-2 & M_H \text{ odd} \end{cases} \qquad (4.11)$$

For M_H even, the actual filter length will be N-2, so these lengths are not considered further.

It is possible to compute Chebychev approximations with the property that $G(\omega) = G(\pi-\omega)$. If the lower passband frequency is ω_L, choose the upper ω_u to be $\pi-\omega_L$, and minimize

$$\max|D(\omega)-G(\omega)| \qquad (4.12)$$

where $\omega \in [\omega_L, \pi-\omega_L]$ and $D(\omega)= -1$.

Then $G(\omega) = G(\pi-\omega)$. Observe that due to the sine functions, $G(0) = G(\pi)$, and $G(\omega) = -G(-\omega)$. If $G(\omega) \neq G(\pi-\omega)$, both $G(\omega)$ and $G(\pi-\omega)$ would satisfy the condition for optimality, since $\omega_u = \pi-\omega_L$, contradicting the uniqueness of the optimum approximation (Appendix B).

Thus with N_H and M_N odd integers, and $\omega_u=\pi-\omega_L$, the number of multiplications are reduced with one more half for the Hilbert transformer.

4.3 Implementation of bandpass filtering and Hilbert transformation

The bandpass filtering (BP) and Hilbert transformation (HT) are a time-consuming part of the square envelope detector algorithm, but also a part where it is possible to use different algorithms in order to reduce the amount of computation. It is important to investigate how long these reductions can be carried out, both for off-line CPU economy and in later valuation of the square envelope detector's usefulness in real time processing. Therefore three different manners of performing BF/HT have been compared with regard to the number of floating point multiplications and additions involved, and locations needed for the filter coefficients. Further, an attempt was made to find a fast FFT algorithm for the cases where frequency domain filtering was applied.

4.3.1 Time domain BF/HT

The most straightforward method is time domain BF/HT. Denoting b(k) and h(k) the impulse response coefficients of the bandpass filter and Hilbert transformer respectively, the expression for the BF are ($N_B = 2M_B+1$)

$$y(n) = \sum_{k=o}^{N_B-1} b(k)x(n-k) = \sum_{k=o}^{M_B-1} b(k)x(n-k)$$

$$+ b(M_B)x(n-M_B) + \sum_{k=o}^{M_B-1} b(2M_B-k)x(n-2M_B+k) \qquad (4.13)$$

$$= \sum_{k=o}^{M_B-1} b(k)\left[x(n-k)+x(n-2M_B+k)\right]+b(M_B)x(n-M_B)$$

This gives $\frac{N_B-1}{2} + 1 = \frac{N_B+1}{2}$ multiplications, and $\frac{N_B-1}{2} + \frac{N_B-3}{2} + 1 = N_B-1$ additions per sample.

For the HT, we have

$$y(n) = \sum_{k=o,2,\ldots}^{M_H-1} h(k)\left[y(n-k)-y(n-2M_H+k)\right] \qquad (4.14)$$

which gives $\frac{N_H+1}{4}$ multiplications, and $\frac{N_H+1}{4} + \frac{N_H+1}{4} - 1 = \frac{N_H-1}{2}$ additions per sample.

The required number of real storage locations for the filter coefficients are $\frac{N_B+1}{2}$ for the bandpass filter, and $\frac{N_H+1}{4}$ for the Hilbert transformer.

4.3.2 FFT algorithms

The remaining methods apply frequency domain filtering, thus the speed of these methods will depend on the speed of the FFT used in the forward and inverse Discrete Fourier Transform (DFT) calculations. The fastest software FFT found in the literature for N a power of 2 and complex input sequences, was Bergland and Dolan's base 8-4-2 routine FFT842 (Programs for Digital Signal Processing, IEEE Press).

The FFT842 performs as many base 8 iterations as possible, and then performs one base 4 or base 2 iteration, if necessary. From Bergland (1968) this FFT is probably the best choice when N is a power of 2, since the computations are nearly minimized while the DFT is still relatively easy to compute. The complex input option makes it possible to combine two real data sequences into one, thereby saving computations.

From Bergland again, the base 8-4-2 algorithm requires $4(Nm/3-(N-1))$ real multiplications, and $2(11/8\ Nm - (N-1))$ real additions when $N = 8^{m/3} = 2^m$, m=3,6,9... and the cosine, sine calculations not accounted for. For N equal to other integral powers of 2, the formulas are good approximations. The base 8 FFT are explained in Appendix B.

Before the FFT842 was known, two base 4-2 FFT's were programmed. The bit reversing was avoided by letting the inverse FFT routine have bit reversed input with normal ordered output. Further stored values of the complex exponentials were used with the number of storage locations reduced to N/2, utilizing the symmetries in cos(x) and sin(x). We have $\sin(2\pi/N)kn = \cos(2\pi/N)(kn-N/4)$, $\cos-(2\pi/N)kn = \cos(2\pi/N)kn$, and $\cos(2\pi/N)(kn+N/2) = -\cos(2\pi/N)kn$. The two routines were compared in speed with the FFT842 on a PDP 11/34 computer, to see the effect of avoiding bit reversing and complex exponential computation. The mean elapsed CPU time in seconds for one DFT and inverse DFT computation is presented in Table 4.1. It can be seen that the percentage gain for FFT42/ IFFT42 decreases for increasing N, due to the increased number of base 8 computations in FFT842. For N = 2048 FFT842 is faster. It will generally depend on both storage capacity of the computer, how time critical the computations are, and N which choice will be preferable.

N	FFT842	FFT42 IFFT42	% GAIN
4	0.01	0.00	
8	0.02	0.00	
16	0.03	0.01	66.7
32	0.06	0.04	33.3
64	0.13	0.10	23.1
128	0.28	0.24	14.3
256	0.60	0.55	8.3
512	1.30	1.24	4.6
1024	2.86	2.80	2.1
2048	6.04	6.18	-2.3

TABLE 4.1

Mean elapsed CPU time in seconds for one DFT and inverse DFT computation.

If the sequence to be multiplied by in frequency domain has real impulse response, the storage space for this sequence can be reduced utilizing symmetries in the frequency response. But this implies a more complicated array indexing in the frequency domain filtering routine for the FFT42/IFFT42 combination, thereby slowing down the computation compared to when FFT842 is used. Due to this, the extra data storage requirement for the FFT42/IFFT42 combination, and the relatively small time differences for n > 128, the FFT842 was preferred. The program storage was found to be approximately equal. All three routines are listed in Appendix E.

4.3.3 Mixed mode BF/HT

We call frequency domain BF/time domain HT mixed mode BF/HT. Time domain BF/frequency domain HT have not been considered, since shorter filter lengths and fewer multiplications are needed for time domain HT than time domain BF.

Let two real sequences $x_1(k)$ and $x_2(k)$ be represented as one complex: $x(k) = x_1(k)+jx_2(k)$. If N_B and $B(k)$ are the bandpass filter length and frequency response, respectively, $N = 2^m$, and $N-N_B+1$ the block length of the data samples, the frequency domain BF can be performed according to the following expressions (nulls are inserted in $x_1(n)$ and $x_2(n)$ from $N-N_B+1$ to $N-1$).

$$X(k) = \sum_{n=o}^{N-1} (x_1(n)+jx_2(n))W_N^{kn} = X_1(k)+jX_2(k) \qquad k = 0,1,\ldots N-1$$

$$y(n) = y_1(n)+jy_2(n) = \sum_{k=o}^{N-1} [X_1(k)S(k)W_N^{-kn}+jX_2(k)S(k)W_N^{-kn}] \tag{4.15}$$

$$= \sum_{k=o}^{N-1} X(k)S(k)W_N^{-kn} \qquad n = 0,1,\ldots,N-1$$

where

$$W_N = e^{-j(2\pi/N)}, \; B(k) = \sum_{n=0}^{N-1} b(k){W_N}^{kn}, \quad S(k) = B(k)/N, \text{ and } b(n) = 0$$

for $n = N_B,\ldots,N-1$. Since the filtered output sequences $y_1(k)$ and $y_2(k)$ are real, two FFT computations will suffice for the BF of the input sequences. This gives $8(1/3\ Nm-(N-1))+4N$ multiplications, and $4(11/8\ Nm-(N-1))+2N$ additions for one block BF. Per sample the number of multiplications are $[4(1/3\ Nm-(N-1))+2N]/[N-N_B+1]$, and additions $[2(11/8\ Nm-(N-1))+N]/[N-N_B+1]$. The filter coefficients $b(n)$ are real. Thus the frequency response $B(k)$ has the property $B(k) = B^*(N-k)$, $k = 0,\ldots,N$, $B(0) = B(N)$, and the storage requirement for the $B(k)$'s is N locations.

4.3.4 Frequency domain BF/HT

When frequency domain BF/HT are used, an unfiltered input sequence x(n) can directly be transformed to a filtered complex sequency $y(n)+j\tilde{y}(n)$. Expressed mathematically

$$
\begin{aligned}
y(n)+j\tilde{y}(n) &= \frac{1}{N}\sum_{k=0}^{N-1} X(k)B(k)W_N^{-kn}+j\frac{1}{N}\sum_{k=0}^{N-1} X(k)B(k)H(k)W_N^{-kn} \\
&= \frac{1}{N}\sum_{k=0}^{N-1} X(k)B(k)\left[1+jH(k)\right]W_N^{-kn} \qquad (4.16) \\
&= \frac{1}{N}\sum_{k=0}^{N-1} X(k)U(k)W_N^{-kn} \qquad n=0,\ldots,N-1
\end{aligned}
$$

where $U(k) = B(k)\left[1+jH(k)\right]$, $H(k) = \sum_{n=0}^{N-1} h(k)W_N^{kn}$, $X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}$ $x(n)=0$ for $n=N-N_H-N_B+2,\ldots,N-1$, and $h(n)=0$ for $n=N-N_H+1,\ldots,N-1$.

Due to the term $1+jH(k)$, it is not sufficient with two FFT computations when two real sequences are combined into one complex. Let $y_1(n)$ and $y_2(n)$ in (4.14) be replaced with $y_1(n)+jy_1(n)$ and $y_2(n)+jy_2(n)$. Then

$$
\left[y_1(n)+jy_1(n)\right]+j\left[y_2(n)+jy_2(n)\right] = \left[y_1(n)-y_2(n)\right]+j\left[y_1(n)+y_2(n)\right]
$$

and there are no means to distinguish the two sequences. But they can be commonly transformed to frequency domain, and separated there. Let $X_1(k) = X_{1e}(k)+jX_{1o}(k)$ and $X_2(k) = X_{2e}(k)+jX_{2o}(k)$, where $X_{ie}(k)$ and $X_{io}(k)$, $i=1,2$, are the even and odd DFT parts of $x_1(n)$ and $x_2(n)$. Then

$$
X(k) = X_1(k)+jX_2(k) = \left[X_{1e}(k)-X_{2o}(k)\right] + j\left[X_{1o}(k)+X_{2e}(k)\right]
$$

and by utilizing the symmetry properties of $X_1(k)$ and $X_2(k)$

$$X_1(k) = \tfrac{1}{2}[X(k)+X^*(N-k)] \qquad k=0,\ldots,N-1$$

$$= \tfrac{1}{2}([X_{1e}(k)-X_{2o}(k)+X_{1e}(N-k)-X_{2o}(N-k)]$$

$$+ j[X_{1o}(k)+X_{2e}(k)-X_{1o}(N-k)-X_{2e}(N-k)] \qquad (4.17)$$

$$= X_{1e}(k)+jX_{1o}(k)$$

$$X_2(k) = -j/2[X(k)-X^*(N-k)] \qquad k=0,\ldots,N-1$$

$$= -j/2[-2X_{2o}(k)+j2X_{2e}(k)] = X_{2e}(k)+jX_{2o}(k) \qquad (4.18)$$

For frequency domain BF/HT, this gives 12(Nm/3-(N-1))+8N multiplications, and 6(11/8 Nm-(N-1))+6N additions per block. Per sample the number of multiplications are $[6(Nm/3-(N-1))+4N]/\ [N-N_B-N_H+2]$, and additions $3[11/8Nm-(N-1)+N]/[N-N_B-N_H+2]$

The term 1+jH(k) also destroys any symmetry in the bandpassfilter Hilbert transformer frequency response [1+jH(k)]B(k), so the storage requirements are 2N locations for the coefficients of this frequency response.

4.3.5 Comparison between time domain, mixed mode and frequency domain BF/HT

In Table 4.3 the multiplications and additions per sample for the three methods are presented for four different bandpass filter lengths, and twelve different Hilbert transformer lengths, averaging over N=256, 512 and 1024. The least integers greater than or equal to the real numbers are substituted. The unknown number of multiplications and additions included in the W_N^{kn} computations in the FFT842 were not accounted for in Tables 4.2 and 4.3, so actually there will be a few more multiplications and additions per sample for mixed mode and frequency domain BF/HT. The computation errors due to finite register length of the computers were not considered either. On the PDP 11/34 with 32 bits floating point arithmetic, the errors were maximum around 10^{-4} for N = 1024. This is very little compared to the seismic noise.

It is apparent from Table 4.3 that the multiplication and addition savings increase with increasing N_B for mixed mode and frequency domain BF/HT, compared to time domain BF/HT, due to the logarithmic effect of the FFT. Only for N_B=21 are the computation efforts about equal.

$N=2^m$	Multiplications per sample	Additions per sample	Storage Requirement
Time domain BF/HT	$\frac{2N_B+N_H+3}{4}$	$\frac{2N_B+N_H-3}{2}$	$\frac{2N_B+N_H+3}{4}$
Mixed mode BF/HT	$[4/3\ Nm-2N+4]/[N-N_B+1]+\frac{N_H+1}{4}$	$[11/4\ Nm-N+2]/[N-N_B+1]+\frac{N_H-1}{2}$	$\frac{4N+N_H+1}{4}$
Frequency domain BF/NT	$[2N(m-1)+6]/[N-N_B-N_H+2]$	$[33/8\ Nm+3]/[N-N_B-N_H+2]$	2N

TABLE 4.2

For the same reason, the multiplication and addition increase for increasing N_H is slower for frequency domain BF/HT than mixed mode BF/HT. Thus for large N_H frequency domain BF/HT is faster than mixed mode BF/HT. From Table 4.2 time domain BF/HT needs fewer filter coefficient locations than mixed mode BF/HT, which again needs fewer than frequency domain BF/HT. The storage requirement of the two latter depends on N, which is not the case with the former, thus the difference increases with incresing N.

The advantages of time domain BF/HT are little program and data space requirements and ease of programming. The other two methods require considerably more space and more complex algorithms, but the profit in computation is considerable for longer filter lengths. In addition, hardware FFT's on VLSI chips

Bandpass filter length N_B	Hilbert transformer length N_H	Multiplications per sample: time domain BF/HT	mixed mode BF/HT	frequency domain BF/HT	Additions per sample: time domain BF/HT	mixed mode BF/HT	frequency domain BF/HT
	7	13	13	17	23	28	40
	11	14	14	18	25	30	40
	15	15	15	18	27	32	41
	19	16	16	18	29	34	41
	23	17	17	18	31	36	41
21	27	18	18	18	33	38	42
	31	19	19	18	35	40	42
	35	20	20	19	37	42	43
	39	21	21	19	39	44	43
	43	22	22	19	41	46	44
	47	23	23	19	43	48	44
	51	24	24	20	45	50	45
	7	23	13	18	43	30	42
	11	24	14	18	45	32	42
	15	25	15	19	47	34	43
	19	26	16	19	49	36	43
	23	27	17	19	51	38	44
41	27	28	18	19	53	40	44
	31	29	19	20	55	42	45
	35	30	20	20	57	44	45
	39	31	21	20	59	46	46
	43	32	22	20	61	48	46
	47	33	23	20	63	50	47
	51	34	24	21	65	52	48
	7	33	14	19	63	31	44
	11	34	15	20	65	33	45
	15	35	16	20	67	35	45
	19	36	17	20	69	37	46
	23	37	18	20	71	39	46
61	27	38	19	20	73	41	47
	31	39	20	21	75	43	48
	35	40	21	21	77	45	48
	39	41	22	21	79	47	49
	43	42	23	22	81	49	49
	47	43	24	22	83	51	50
	51	44	25	22	85	53	51
	7	43	15	20	83	33	47
	11	44	16	21	85	35	48
	15	45	17	21	87	37	48
	19	46	18	21	89	39	49
	23	47	19	22	91	41	49
81	27	48	20	22	93	43	50
	31	49	21	22	95	45	51
	35	50	22	22	97	47	52
	39	51	23	23	99	49	52
	43	52	24	23	101	51	53
	47	53	25	24	103	53	54
	51	54	26		105	55	

TABLE 4.3

are available making it possible to do extremely fast frequency domain filtering. In the experiments N_B=61 were chosen, and for the 1.6-3.6 Hz band passband cutoff frequencies were set at 1.7 and 3.5 Hz to match with cutoff frequencies in recursive design, and transition widths were set equal to 0.7 Hz. With a stopband/passband weighting ratio of 10/1, this gives -49 dB max. deviation in the stopband and -29 dB in the passband. The max. deviation for the Hilbert transformer is -26 dB with N_H = 15 and cutoff frequencies at 1.0 and 9.0 Hz. Mixed mode BF/HT have been applied, since from Table 4.3 this method needs less storage space and is faster than frequency domain BF/HT for moderate and short length Hilbert transformers. In the current detection system running at NORSAR, the bandpass filter algorithm is implemented in microcode, and a third order recursive Butterworth filter is employed in time domain requiring 10 multiplications per sample (IBM Seismic Array Design Handbook). On the new system this extra microcode option is not available, so frequency domain BF with FIR linear phase Chebyshev approximation filters could be a possible alternative to time domain Butterworth filtering, both due to about comparable computer load (Table 4.3) and the possibilities for 'better' filtering. If Hilbert transforming were added, a third order Butterworth filter has longer transition widths than the above-mentioned 61 length bandpass filter, so shorter transition widths and more filter coefficients would be needed for the Hilbert transformer in combination with the Butterworth filter than in combination with the FIR filter.

4.4 Square envelope calculation

The output from the Hilbert transformer is delayed by $M_H = \frac{N_H-1}{2}$ samples, so the complex samples must be calculated from

$$z(n) = y(n) + j\tilde{y}(n+M_H) \tag{4.19}$$

and the square envelopes

$$r(n)^2 = y(n)^2 + \tilde{y}(n+M_H)^2 = z(n)z^*(n) = |z(n)|^2 \tag{4.20}$$

The sample rate of $r(n)^2$ can be reduced without introducing aliasing due to its lower frequency content. If y(n) is bandlimited to $[\omega_o,\omega_o+\omega_1]$, then

$$|Z(e^{j\omega})| = \left|\sum_{n=o}^{N-1} z(n)e^{-j\omega n}\right| \quad \begin{aligned} &\approx 2\,|Y(e^{j\omega})|,\ \omega \in [\omega_o, \omega_o+\omega_1] \\ &\approx 0 \text{ elsewhere in } [-\pi,\pi] \end{aligned} \tag{4.21}$$

Since $w(n)=z^*(n)$ implies $W(e^{j\omega}) = Z^*(e^{-j\omega})$,

$$R_s(e^{j\omega}) = \sum_{n=o}^{N-1} r(n)^2 e^{-j\omega n} = \frac{1}{2\pi}\int_{-\pi}^{\pi} Z(e^{j\theta})Z^*(e^{j(\theta-\omega)})\, d\theta$$

$$= \frac{1}{2\pi}\int_{\omega_o}^{\omega_o+\omega_1} Z(e^{j\theta})Z^*(e^{j(\theta-\omega)})\, d\theta \tag{4.22}$$

$$\neq 0 \text{ when } \omega \in [-\omega_1, \omega_1]$$

$$= 0 \text{ elsewhere in } [-\pi,\pi]$$

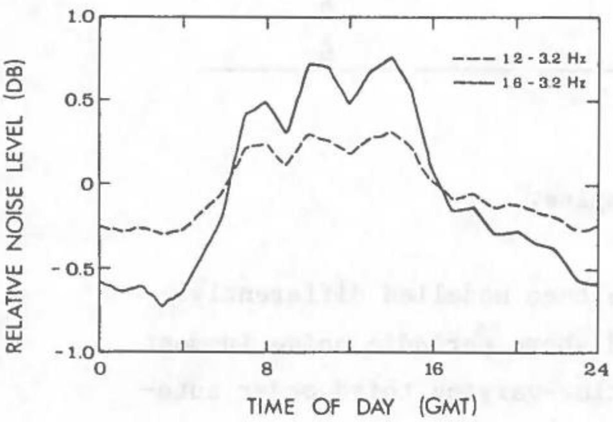
Thus $r(n)^2$ will have equal passband width to the bandpass filter shifted down in frequency ω_o. ω_1 is about 2.5 Hz with the bandpass filter mentioned in 4.3.5. From Steinert et al (1975) the main lobe of the frequency response for the STA-operator is down 20 dB at 0.5 Hz, so this explains the higher detector variability of the square envelope detector than the STA-detector. If T is the original sample interval, and it is assumed that x(n) have been sampled at high enough rate, i.e., at least twice the highest cutoff frequency of the analogous lowpass filter, $\omega_1 = \Omega_1 T < \pi$. Then the new sample interval T' can be chosen:

$$\Omega_1 T' = \pi \qquad T' = \frac{\pi}{\Omega_1} = \frac{\pi}{2\pi f_1} = \frac{1}{2f_1} \qquad f_1 = \frac{\omega_1}{2\pi T} \tag{4.23}$$

Since FIR Chebyshev approximation filters have ripples in both passband and stopband, and the designer has precise control over the transition widths, they can be chosen small, and depending on the bandpass width, the sample rate can be considerably reduced. Small transition widths will imply longer filter lengths, but from Table 4.3 that disadvantage is eliminated by using fast FFT-algorithms and frequency domain filtering. In the detection experiments 5 Hz beamforming sample rate has been used for both methods.

5. NOISE PROPERTIES

The recordings to be processed by a seismic detector include noise of many types. Depending on source, it can be divided into different categories. The so-called cultural noise is generated by traffic, machinary, local wind and so on. It has its major effect for frequencies greater than 1 Hz, and since it is mostly manmade it is expected to show diurnal variations. Fig. 5.1, taken from Ringdal and Bungum (1977) shows this variability. It can be seen that the short period noise level is higher during the day than night, and also higher during workdays than holidays. Noise reductions are achieved by moving the recording sites from industrial areas and burying the seismometers a few meters below the surface. Rayleigh waves generated by ocean surfs are called microseisms. The dominant energy is around 3-10 seconds period and thus outside the peak responses for the short periodic seismometers (Fig. 4.1). Ringdal and Bungum have found that this noise shows strong seasonal trends with higher levels during winter than summer. System noise comes from seismometer amplifiers, transmission lines, etc., and is many dB lower than the seismic noise.



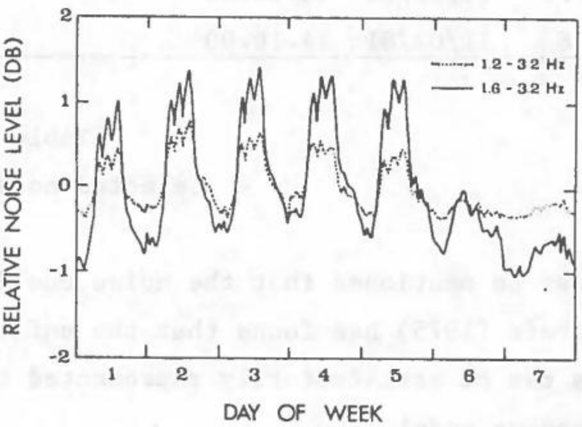


Fig. 5.1 Diurnal variations of short period noise level by time of day and day of week.

As mentioned the signal frequency also increases with decreasing earthquake magnitude (Fig. 1.2), and there will always be weak signals from small earthquakes completely unidentifiable from the noise.

In this chapter some properties assumed about the complex noise field in chapter 3 and appendix C are tested. The question about normality has also been examined. Table 5.1 describes the noise samples used in the experiments. They were selected from time intervals without detections and plotted to ensure that no spikes or signals were present. The noise can be assumed stationary and ergodic for the interval considered.

	Date and Start Time		Sample Length	Channels
1	4/11/79	02.16.00	4700	6
2	4/11/79	13.04.00	4700	6
3	5/08/79	03.42.00	4700	6
4	5/08/79	15.42.00	4700	6
5	13/05/81	03.25.00	4700	6
6	13/05/81	11.10.00	4700	6
7	11/02/81	00.10.00	4700	6
8	11/02/81	14.16.00	4700	6

Table 5.1
Selected noise samples.

It must be mentioned that the noise could have been modelled differently. Tjøstheim (1975) has found that the unfiltered short periodic noise in most cases can be satisfactorily represented by a time-varying third order autoregressive model, i.e.,

$$y(n) - a_1(n)y(n-1) - a_2(n)y(n-2) - a_3(n)y(n-3) = z(n) \tag{5.1}$$

where z(n) is a white noise process. This model acts as a prediction error filter, as the residual z(n) will deviate from the expected behavior when a signal is present. The method has been tried (Gjøystdal & Husebye, 1974), but found to give lower SNR than bandpass filtering. However, it is excellent for estimating noise characteristics such as power spectral density.

5.1 Properties following from the Hilbert transformer

The most important results in this section are that the output sequence from the Hilbert transformer has autocorrelation similar to the input sequence, and that corresponding real and imaginary samples are uncorrelated. From chapter 3 the real noise field has zero mean, and since the expectation is a linear operator the same will be true for the imaginary part.

Using (4.8), we get for a sequence being Hilbert transformed twice

$$\tilde{\tilde{y}}(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k)H(k)Y(k)W_N^{-kn}$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} -W_N^{2M_Hk} G(k)^2 Y(k) W_N^{-kn} \tag{5.2}$$

At those k's where Y(k) is not approximately zero, $G(k) \approx \pm 1$, hence

$$\tilde{\tilde{y}}(n) \approx \frac{1}{N} \sum_{k=0}^{N-1} -W_N^{2M_Hk} Y(k) W_N^{-kn} = -y(n-2M_H) \tag{5.3}$$

Thus $\tilde{\tilde{y}}(n)$ is approximately equal to $-y(n)$, delayed $2M_H$ samples. Further the cross covariance between input and output is the Hilbert transform of the autocovariance of the input sequence. Let $v(n) = \tilde{y}(n+M_H)$, $\mu_y = E[y(0)] = 0$, $\mu_v = E[v(n)] = 0$. Then

$$\gamma_{yv}(m) = E[(y(n)-\mu_y)(v(n+m)-\mu_v)] = E[y(n)v(n+m)]$$

$$= E[y(n)\sum_{k=0}^{N_H-1} h(k)y(n+M_H+m-k)] = \sum_{k=0}^{N_H-1} h(k)\gamma_{yy}(M_H+m-k) \tag{5.4}$$

$$= \tilde{\gamma}_{yy}(M_H+m)$$

Due to the antisymmetry of the Hilbert transformer, the following property for the cross covariance is valid

$$\gamma_{yv}(-m) = \sum_{k=0}^{N_H-1} h(k)\gamma_{yy}(M_H-m-k) = \sum_{k=0}^{N_H-1} h(k)\gamma_{yy}(m+k-M_H)$$

$$= \sum_{\ell=0}^{N_H-1} h(2M_H-\ell)\gamma_{yy}(m+M_H-\ell) = -\sum_{\ell=0}^{N_H-1} h(\ell)\gamma_{yy}(m+M_H-\ell)$$

$$= -\gamma_{yv}(m) \tag{5.5}$$

From (5.5) y(n) and v(n) are uncorrelated at the same time instant, since $\gamma_{yv}(0) = -\gamma_{yv}(0) = 0$. Since a 1.6-3.6 Hz band is used, the complex noise samples from the different sensors are therefore spatially uncorrelated.

Now, utilizing (5.3), (5.4), (5.5), the fact that $\gamma_{yv}(m) = \gamma_{vy}(-m)$, and $E[x(n)(-x(n+m))] = -E[x(n)x(n+m)]$,

$$\tilde{\gamma}_{vv}(m) = \gamma_{v\tilde{v}}(m) \approx -\gamma_{vy}(m-M_H) = \gamma_{vy}(M_H-m) = \gamma_{yv}(m-M_H)$$

$$= \tilde{\gamma}_{yy}(m) \tag{5.6}$$

We see that the Hilbert transforms of the autocovariances of the imaginary and real sequences are approximately equal. Hence the autocovariances themselves and particularly the variances are approximately equal. This property is not difficult to imagine, because the Hilbert transformer only shifts the phase on each frequency component $-(\pi/2)$.

Fig. 5.2 shows autocorrelation estimates of the real and imaginary samples from the first channel and time in table 5.1. As can be seen the curves are almost indistinguishable and show a decaying sinusoidal shape. From ca. lag 30 the fluctuations are more or less constant. Assuming zero correlation at this lag and calculating the standard deviation for larger lags (large lag standard error), the curves are within twice this value from lag 30. The same was seen from autocorelation listings of the other samples, thus the correlation length for the filtered noise samples was set to 1.5 seconds.

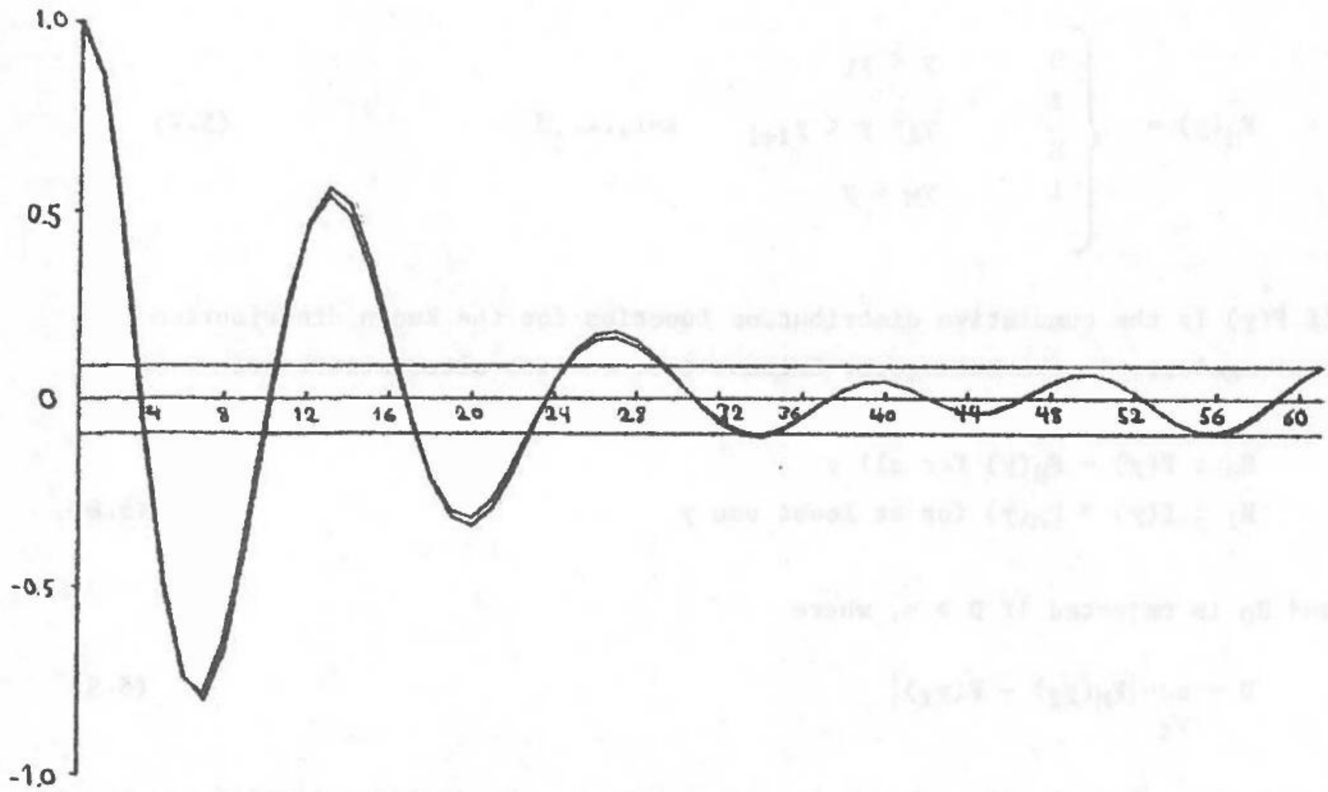


Fig. 5.2 Autocorrelation estimates of the real and imaginary part of the noise samples from sensor 1A00, starting at 4/11/79 02.16.00. The horizontal lines at each side of the time axis indicate ± 2 large lag standard error.

5.2 The probability density of noise

In chapter 3 Gaussian distributed noise samples were assumed in the derivation of the square envelope detector. If the real samples are Gaussian, the imaginary samples will also be Gaussian since the Hilbert transformer is a linear operator (Van Trees, chapter 3.3.3). To check the Gaussian assumption, the data were therefore resampled at 2/3 Hz rate to get uncorrelated samples, and then tested with two different tests, the Kolmogorov-Smirnov test and the skewness test.

Let $\{y(i)\}$ be a resampled sequence, and denote by $\{y_\ell\}$ this sequence sorted in nondecreasing order. Then the empiric cumulative distribution function of y(i) is equal to

$$F_M(y) = \begin{cases} 0 & y < y_1 \\ \frac{\ell}{M} & y_\ell \leq y < y_{\ell+1} \qquad \ell=1,\ldots,M \\ 1 & y_M \leq y \end{cases} \tag{5.7}$$

If F(y) is the cumulative distribution function for the known distribution to test against, in the Kolmogorov-Smirnov test the two alternatives are

$$\begin{aligned} H_0 &: F(y) = F_M(y) \text{ for all } y \\ H_1 &: F(y) \neq F_M(y) \text{ for at least one } y \end{aligned} \tag{5.8}$$

and H_0 is rejected if $D > k$, where

$$D = \sup_{y_\ell} |F_M(y_\ell) - F(y_\ell)| \tag{5.9}$$

and k the critical value chosen to given the test the desired significance level. Since F(y) in this case is a normal cumulative distribution function, it depends on the parameters μ and σ^2. As σ^2 is unknown, estimated values must be used. Then the standard tables will not give correct values. The mean and variance were therefore estimated from each of the resampled sequences, using

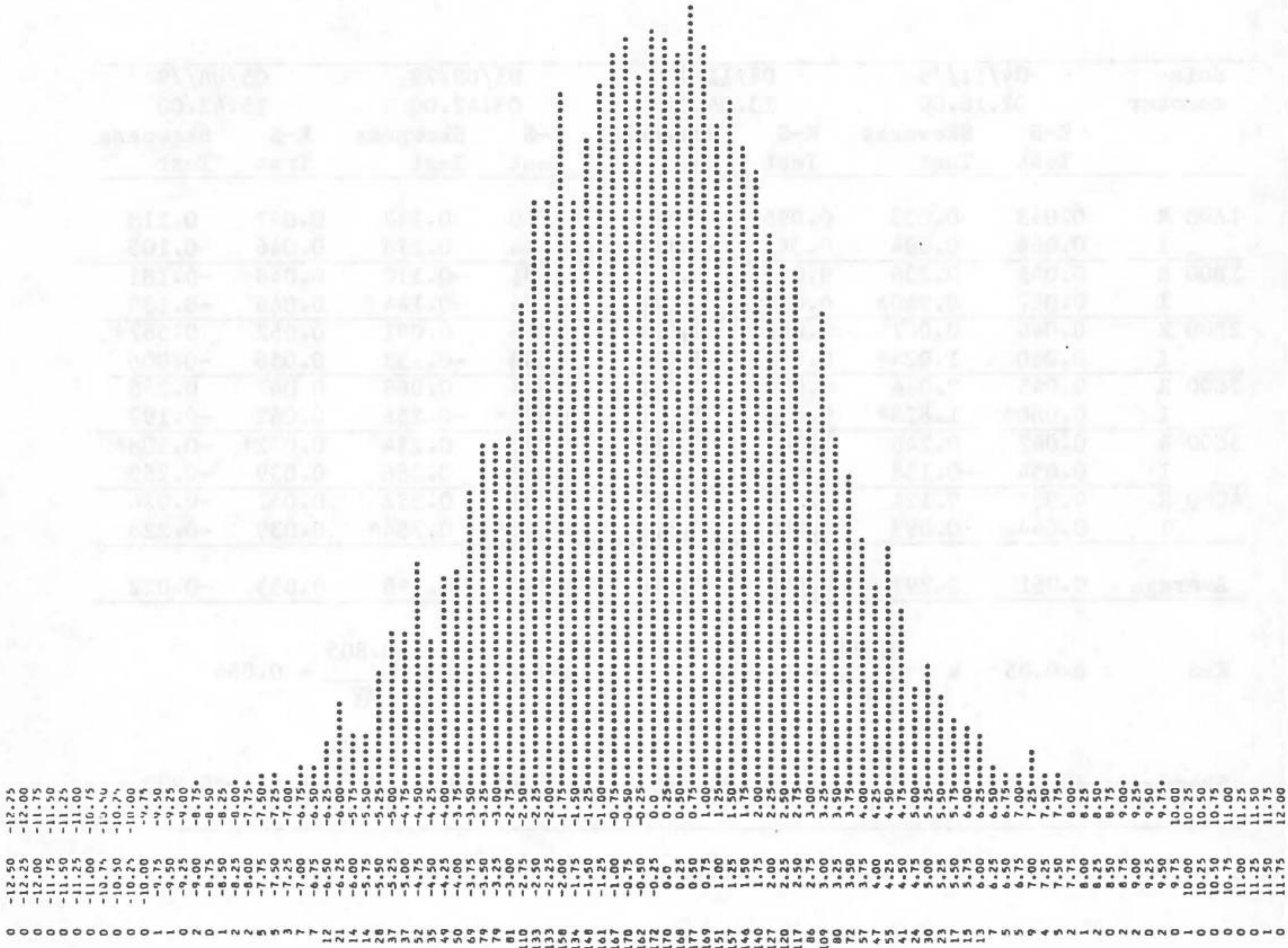


Fig. 5.3 Probability histogram of the real part of the noise samples from sensor 1A00, starting at 4/11/79 02.16.00. The samples have been scaled by 10, and the numbers are frequency values and pocket limits.

$$\bar{y} = \frac{1}{M} \sum_{i=1}^{M} y(i) \qquad \text{and} \qquad \hat{\sigma}^2 = \frac{1}{M-1} \sum_{i=1}^{M} (y(i)-\bar{y})^2$$

and substituted as parameters in F(y) to be able to use simulated values from Lilliefors (1967).

Seismometer		04/11/79 02.16.00		04/11/79 13.04.00		05/08/79 03.42.00		05/08/79 15.42.00	
		K-S Test	Skewness Test	K-S Test	Skewness Test	K-S Test	Skewness Test	K-S Test	Skewness Test
1A00	R	0.043	0.003	0.095*	0.581*	0.040	0.117	0.047	0.118
	I	0.064	0.094	0.041	-0.165	0.064	0.278	0.046	0.105
1B00	R	0.045	0.236	0.048	0.334	0.031	-0.110	0.048	-0.181
	I	0.067	0.940*	0.048	-0.044	0.054	-0.144	0.044	-0.139
2B00	R	0.046	0.007	0.081*	0.014	0.066	0.091	0.062	0.587*
	I	0.060	1.024*	0.058	-0.235	0.046	-0.193	0.056	-0.006
2C00	R	0.045	0.044	0.039	0.112	0.044	0.068	0.047	0.265
	I	0.080*	1.823*	0.048	-0.341	0.072*	-0.256	0.062	-0.197
3C00	R	0.042	0.246	0.043	0.180	0.053	0.234	0.072*	-0.508*
	I	0.054	-0.138	0.067	-0.024	0.067	0.266	0.059	-0.259
4C00	R	0.031	0.124	0.040	0.041	0.041	0.332	0.051	-0.076
	I	0.044	0.093	0.099*	0.523*	0.085*	0.754*	0.039	-0.226
Average		0.051	0.297	0.058	-0.024	0.055	0.168	0.053	-0.022

K-S : $\alpha=0.05 \quad k = \dfrac{0.886}{\sqrt{M}} = 0.071,\qquad \alpha=0.10 \quad k = \dfrac{0.805}{\sqrt{M}} = 0.064$

Skewness: $\alpha= 0.05 \quad k=\pm\sqrt{\dfrac{6}{M}}\,u_{0.025} =\pm 0.383,\qquad \alpha=0.10 \quad k =\pm\sqrt{\dfrac{6}{M}}\,u_{0.05} =\pm 0.322$

Table 5.2

The amount of skewness in a population is given by the average of $(x-\mu)^3$, divided by σ^3 to get a measure independent of scale. Since a normal distribution is symmetric about its mean, it will have a skewness equal to zero. The coefficient of skewness was estimated by (Snedecor & Cochran, 3.13)

$$\sqrt{b_1} = \frac{1}{M}\sum_{i=1}^{M} (y(i)-\bar{y})^3/(\frac{1}{M}\sum_{i=1}^{M} (y(i)-\bar{y})^2)^{3/2} \qquad (5.10)$$

Seis-mometer	13/05/81 03.25.00		13/05/81 11.10.00		11/02/81 00.10.00		11/02/81 14.16.00	
	K-S Test	Skewness Test	K-S Test	Skewness Test	K-S Test	Skewness Test	K-S Test	Skewness Test
1A00 R	0.050	-0.043	0.041	-0.027	0.060	-0.459*	0.048	0.115
I	0.039	0.120	0.061	-0.039	0.052	0.181	0.031	0.140
1B00 R	0.043	0.235	0.047	0.394*	0.057	0.066	0.060	-0.113
I	0.043	-0.078	0.044	-0.160	0.047	-0.218	0.056	-0.109
1B01 R	0.043	0.000	0.040	0.043	0.057	1.340*	0.066	-0.533*
I	0.042	0.000	0.037	-0.137	0.061	0.251	0.073*	-0.611*
2C00 R	0.073*	0.282	0.065	0.065	0.051	-0.147	0.052	0.119
I	0.052	0.064	0.045	-0.129	0.040	0.092	0.050	0.143
3C00 R	0.036	0.179	0.033	-0.184	0.036	-0.070	0.053	0.088
I	0.074*	0.037	0.062	-0.303	0.068	-0.320	0.050	-0.088
4C00 R	0.054	-0.025	0.046	-0.078	0.049	0.189	0.053	0.375
I	0.040	-0.088	0.036	-0.039	0.042	-0.953*	0.051	-0.212
Average	0.049	0.057	0.046	-0.109	0.052	-0.004	0.054	-0.072

Table 5.3

If the samples are from a normal population, and M exceeds 150, $\sqrt{b_1}$ will be approximately normally distributed with zero mean and standard deviation $\sqrt{6/M}$. Since M=157, this was used to get critical values equal to $\pm \sqrt{6/M}\, u_{\alpha/2}$ for an α-level test, where $u_{\alpha/2}$ is the $\alpha/2$ quantile in N(0,1).

The results from the K-S and skewness testing are shown in Tables 5.2 and 5.3. The values exceeding the $\alpha = 0.05$ level are marked with a *. With this level, there should be about 5 ($96 \cdot 0.05 = 4.8$) significant values for each test if the samples were from normal populations. The numbers are 10 and 14, so the data does not satisfy the zero hypothesis completely. Nevertheless the Gaussian assumption is quite good, which is also confirmed from probability histograms (Fig. 5.2). The reason for non-normality due to too heavy tails can for instance well be hidden signals in the noise.

6. DETECTOR RESULTS

In chapter 5 the assumption of normality of noise was shown to be acceptable, and the uncorrelatedness and equal statistical properties of corresponding real and imaginary samples followed from Hilbert transformer properties. The further things to be tested are the statistical properties of the detectors for noise input and possible differences in detection performance. The square envelope detector is expected to be chi-squared distributed with degrees of freedom equal to twice the number of sensors in noise. For the STA-detector no theoretical distribution can be calculated, but according to the Central Limit Theorem the detector output will approach a normal distribution when the number of sensors increase. Nor is it possible to draw theoretic performance curves, but since the STA-detector approximates a linear envelope detector, the square envelope detector is expected to show a better small signal detectability due to the argumentation behind (3.18). First, however, program implementation of the detectors will be commented.

6.1 Programming considerations

Due to use of the Fast Fourier Transform, it is convenient to process the incoming data stream in blocks. In a real time environment three data buffers can be used in a cyclic manner, as illustrated in Fig. 6.1. One buffer is occupied by the A/D converter, while data from the other two are processed by the detector. Since time displaced samples are required in the beamforming, the last half in the one buffer and the first half in the other buffer are used by the detector routine. To be real time applicable, the data processing of these two halves must be finished within the A/D-converter has filled up the next buffer. The program was tested offline, therefore no third buffer was used.

The weights are calculated from buffers of previous data indicated by the broken lines in Fig. 6.1. Only buffers processed without giving any detection triggerings are used, therefore data from the nearest previous buffer are skipped.

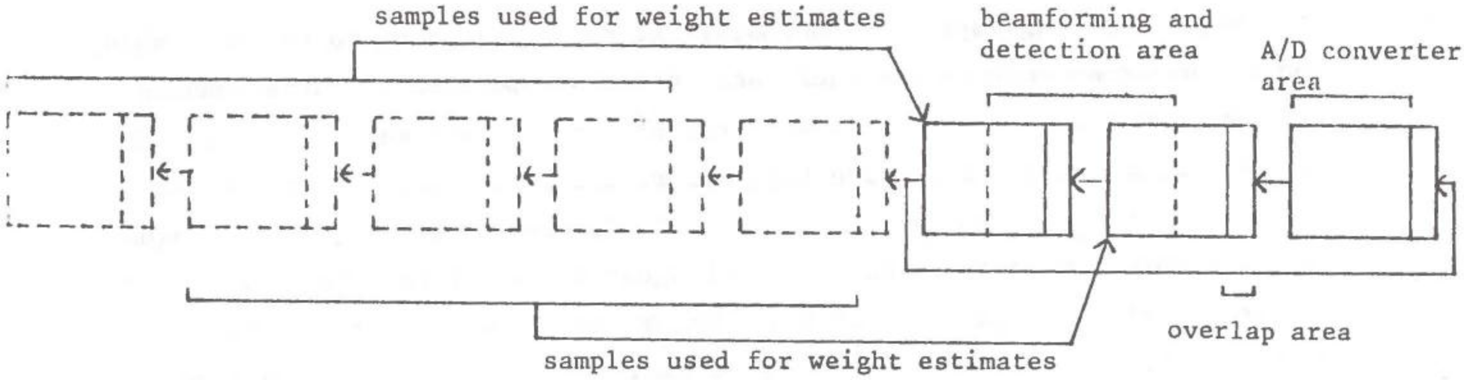


Fig. 6.1 Illustration of data buffering of the detector program in a possible real time environment. The overlap area is the temporary storage area for filter output samples to be added with the first filter output samples in the next buffer. A buffer length of 512 and a bandpass filter length of 61 gives an overlap area length of 60 and data area length of 452.

Four data buffers are used in the weight calculation, and with a buffer length of 452 every weight value is calculated from 1808 samples. The number of four was conveniently determined due to giving small fluctuations in the updating of the weights for the square envelope detector. When the threshold is exceeded, the weight updating is stopped and not continued before the detector has left detection state and a new buffer processed without giving any detections. In detection state a time interval is set up and a sliding window of 2 seconds duration moved through the interval. The beam having highest energy within this window is selected as the beam pointing at the source area for the event. The same beam set was employed as for the incoherent detector running in DP, but extended with 24 new beams to take better care of local events. In detection state also the energy for each channel is calculated. These energy values are compared to check whether the detection was caused by spikes and not real seismic events. The spikes come from communication errors due to electrical discharges or breaks in the data transmission. Fig. 6.3 shows an example where the test failed since spikes occurred at all the channels.

When detection state is left, a certain time interval must be processed without giving any detections before an eventual new event can be declared. Large events can have a duration of many minutes, so the interval is necessary to prevent multiple detections of the same event. Also for every buffer read the array status block must be checked because channels can be masked out. The square envelope detector program is listed in Appendix E.

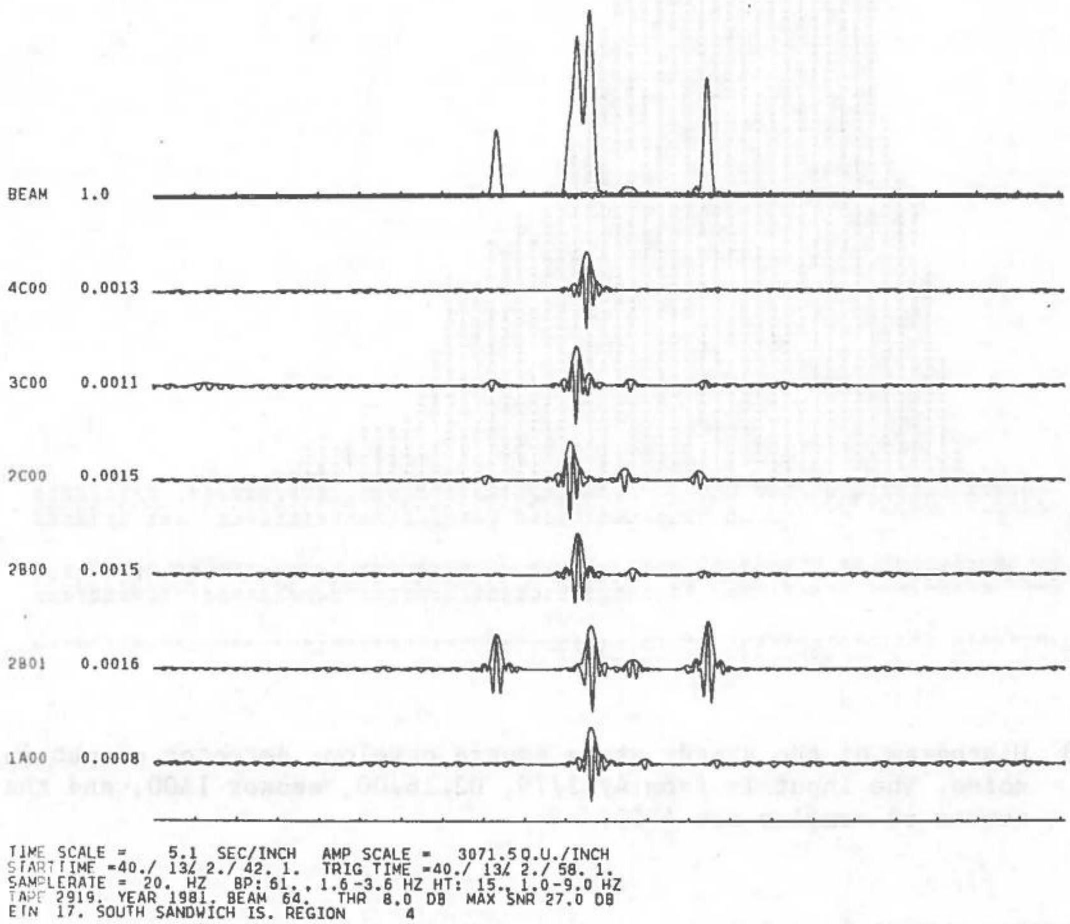


Fig. 6.2 Spikes detected by the square envelope detector. The detector output is plotted on the top and the unweighted data channels below. Channel codes and weight values are plotted to the left.

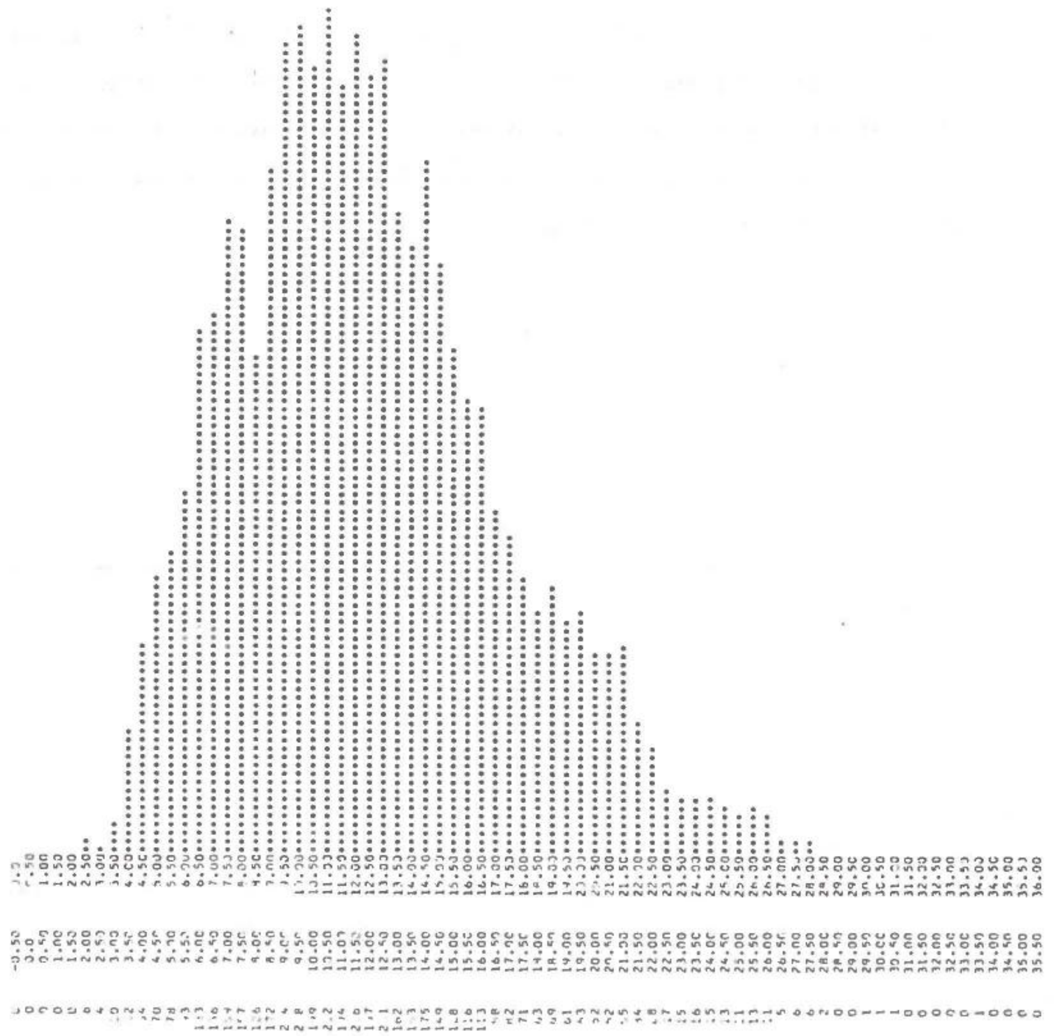


Fig. 6.3 Histogram of the steady state square envelope detector output in noise. The input is from 4/11/79, 02.16.00, sensor 1A00, and the number of samples are 4700.

6.2 Detector output in noise

To check the statistical behavior of the detector outputs in noise data from the same times as in chapter 5 were used, but longer time intervals were selected to permit the detectors to get out of the initiation phase. Since the noise across the channels is uncorrelated, the channels were not beamformed but straight-summed. The detector output was sampled at 2/3 Hz rate and tested for equality in distribution by the Kolmogorov-Smirnov test to check the assumption of stationarity. The samples are assumed independent, since the weight estimates have little variability compared to the square envelope samples and can therefore be treated as approximately constant. To check the assumption of chi-square distribution for the square envelope detector, independent chi-square distributed samples with 12 degrees of freedom were generated from a pseudo-random generator existing in the Nag-library at the IBM 4341 computer and tested with the same test. When D is calculated as in (5.9) and the number of samples to be compared are equal and large, an approximate expression for the significance probability can be calculated (Høyland, 18.3.3):

$$P\left(\sqrt{\frac{N}{2}}\, D \geq z \,|\, H_o\right) \approx 2 \sum_{k=1}^{\infty} (-1)^{k-1} e^{-2(kz)^2} \qquad (6.1)$$

The results from the tests are presented in Tables 6.1 and 6.2. All the data sets are compared to each other, and to the chi-squared generated samples when processed by the square envelope detector. As seen from the tables, only three significance probability values for the square envelope detector and one for the STA detector were below 0.2. Thus according to the Kolmogorov-Smirnov test, the assumption of stationary output in noise for both detectors seems to be valid. Also the square envelope detector output in noise seems to fit a chi-square distribution with degrees of freedom equal to twice the number of sensors. Again histogram checks supported the calculated results.

6.3 Threshold setting

To compare the performance of the detectors, thresholds were first tried to be established giving equal false alarm rates. The histograms from Figs. 6.3 and 6.4 are not adequate for such a task since 64 beams are performed and the detector triggers if one of the beams exceeds the threshold. Therefore it is correct to use the maximum value of these 64 values, and from Høyland (Chap. 7) the max. value will have another statistical distribution than indicated in Figs. 6.3 and 6.4. It is possible to draw histograms for the

	2	3	4	5	6	7	8	χ^2_{12}
1	0.072 0.200	0.508 0.959	0.451 0.987	0.734 0.655	0.508 0.959	0.734 0.655	1.185 0.121*	0.621 0.836
2		0.903 0.389	0.846 0.471	0.734 0.655	0.959 0.316	0.790 0.560	0.790 0.560	0.677 0.749
3			0.564 0.908	0.621 0.836	0.564 0.908	0.790 0.560	1.411 0.037*	0.734 0.655
4				0.734 0.655	0.395 0.998	0.677 0.749	1.072 0.200	0.621 0.836
5					0.508 0.959	0.903 0.389	1.072 0.200	0.959 0.316
6						0.621 0.836	1.242 0.092*	0.734 0.655
7							0.959 0.316	0.508 0.959
8								0.790 0.560

Table 6.1

Square envelope detector output of the 8 noise sets compared to each other and to the chi-squared distribution by the Kolmogorov-Smirnov test. The number pairs are $y = \sqrt{\frac{157}{2}}\, D$ and $P(\sqrt{\frac{157}{2}}\, D > y \mid H_o)$, and the asterisk denotes significance probabilities below 0.2.

	2	3	4	5	6	7	8
1	0.734 0.655	0.621 0.836	0.790 0.560	0.564 0.908	0.903 0.389	0.734 0.655	0.903 0.389
2		0.621 0.836	1.016 0.253	0.846 0.471	0.564 0.908	0.903 0.389	0.564 0.908
3			0.677 0.749	0.451 0.987	0.903 0.389	0.564 0.908	0.846 0.471
4				0.451 0.987	1.016 0.253	0.734 0.655	1.185 0.121*
5					1.072 0.200	0.734 0.655	1.072 0.200
6						0.677 0.749	0.564 0.908
7							1.016 0.253

Table 6.2

STA-detector output of the 8 noise sets compared by the Kolmogorov-Smirnov test.

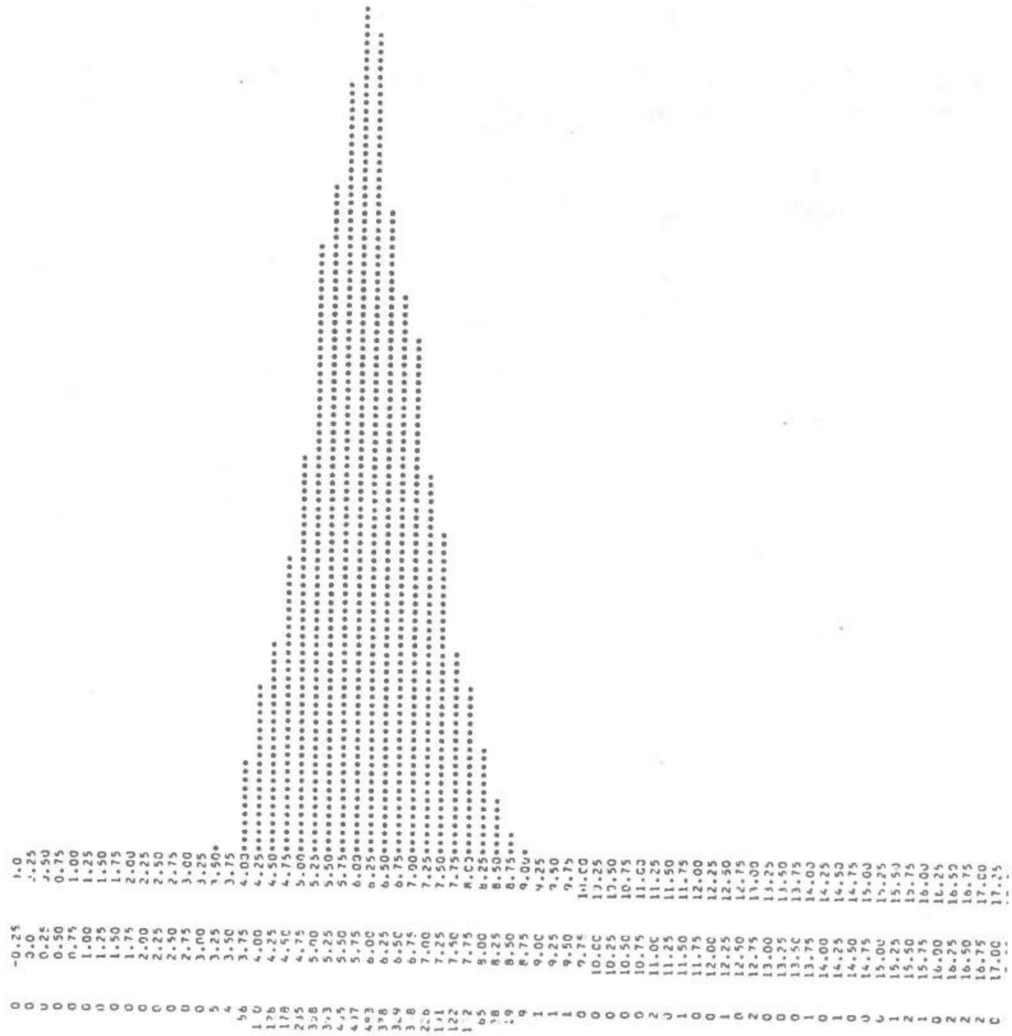


Fig. 6.4 Histogram of the STA-detector output for the same data set as in Fig. 6.3.

max. values, but it is not easy to set thresholds because they must be far out on the tail in order not to give too many false alarms.

Instead the detectors were tested on a tape with 4 hours continuous recording, and the max. beam values within certain SNR windows where the noise was assumed dominating accumulated. Beam values exceeding the upper limit of these windows were processed as signal detections in usual manner to prevent beam values

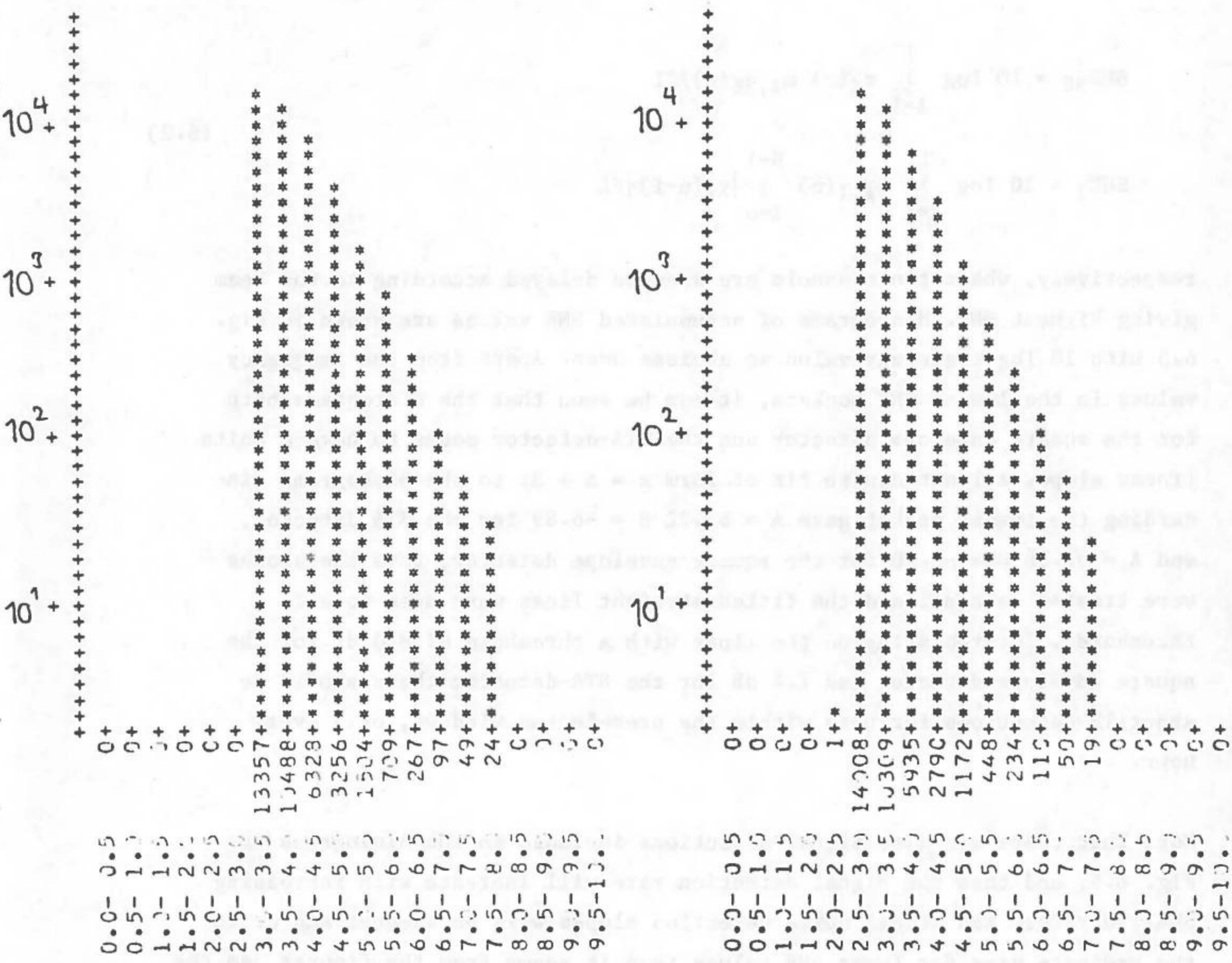


Fig. 6.5 No. of square envelope detections for SNR between 3.0 and 8.0 dB to the left, and no. of STA-detections for SNR between 2.5 and 7.5 to the right.

from the signals to occur. To get comparable SNR values, the SNR definitions for the square envelope detector and the STA detector from chapter 3 were expressed in dB

$$
\begin{aligned}
SNR_{SE} &= 10 \log \sum_{\ell=1}^{L} z_{\ell}(n)\ \omega_{\ell,SE}(n)/2L \\
SNR_{I} &= 20 \log \sum_{\ell=1}^{L} \omega_{\ell,I}(n) \sum_{i=o}^{N-1} |y_{\ell}(n-i)|/L
\end{aligned}
\tag{6.2}
$$

respectively, where the channels are assumed delayed according to the beam giving highest SNR. Histograms of accumulated SNR values are shown in Fig. 6.5 with 10 log frequency value as abcissa axes. Apart from the frequency values in the lowest SNR pockets, it can be seen that the histograms both for the square envelope detector and the STA-detector seems to have a quite linear slope. A least square fit of form $x = A + \beta t$ to the histograms discarding the lowest pocket gave $A = 63.22$ $\beta = -6.89$ for the STA detector, and $A = 67.28$ $\beta = -6.90$ for the square envelope detector, thus the slopes were treated as equal and the fitted straight lines were used to set thresholds. Interpolating on the lines with a threshold of 8.0 dB for the square envelope detector and 7.4 dB for the STA-detector there should be about 12 detections for both within the preselected windows, or 3 every hour.

Note that there are also signal detections included in the histograms in Fig. 6.5, and that the signal detection rate will increase with increasing SNR. Therefore the actual noise detection slopes will be steeper and cross the ordinate axes for lower SNR values than it seems from the figures, so the method above is an approximation for determining equal false alarm rates. Yet it was the best approximation found, since the selected noise data had too few samples to be able to obtain proper slopes. The SNR windows in Fig. 6.5 were determined through a trial and error procedure. Fig. 6.6 shows a histogram of STA-detections within a larger window, and it is obviously influenced by signal detections.

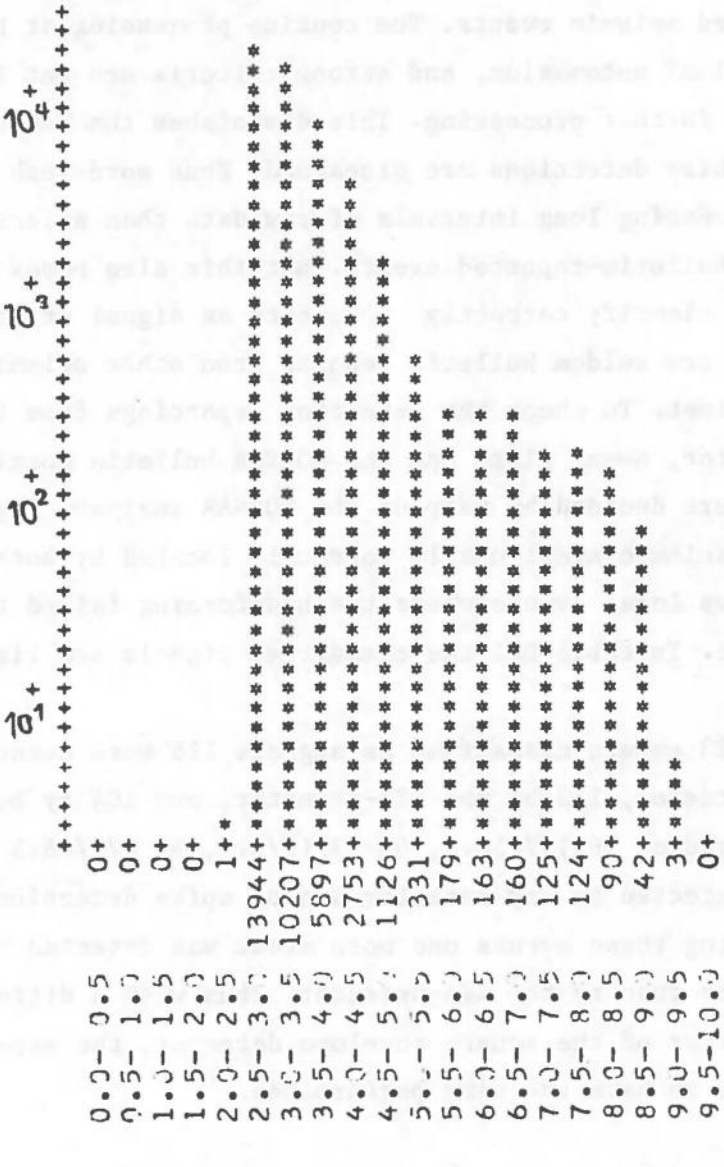


Fig. 6.6 No. of STA-detections for SNR between 2.5 and 9.0 dB.

6.4 Detector performance comparison

Using the tresholds determined from the last section, the detectors were tested on some selected data tapes with 4 hours continuous recording for each. Such recordings exist 9 months backwards in time, and tapes were selected from time intervals where the NORSAR bulletin showed a relatively high density of detected seismic events. The routine processing at NORSAR is kept on a high level of automation, and strong criteria are set by EP to accept an event for further processing. This diminishes the human interaction, but also non-noise detections are discarded. Thus more weak events can be detected by processing long intervals of raw data than selecting intervals only around bulletin-reported events. But this also poses a problem, it may be difficult to classify correctly events as signal or noise. Also for weak events, there are seldom bulletin reports from other seismic installations to compare against. To check the detection reportings from the square envelope and STA detector, event plots and the NORSAR bulletin mostly were used. Doubtful cases were decided by help of the NORSAR analyst. Fig. 6.7 shows a plot of a teleseismic event nearly correctly located by both detectors, and Fig. 6.8 shows local events where the beamforming failed to give proper signal alignment. In table D.1 the classified signals are listed.

Of a total number of 123 events classified as signals 115 were detected by the square envelope detector, 112 by the STA-detector, and 104 by both. The events which occurred at 56 1/7/58.5, 63 23/12/0.2, 64 0/7/46.5 and 640/55/51.6 were not detected by any detector due to spike detections or tape read problems. Discarding these events one more event was detected by the square envelope detector than by the STA-detector. Thus with a difference of only 0.9 per cent in favor of the square envelope detector, the experiment indicates the detectors to have the same performance.

Equal detection slopes with difference between zero crossings equal to 0.6 were obtained by defining SNR according to (6.2). This suggests another method to compare the detectors. Plotting (SNR_{SE}, SNR_I) according to (6.2) for events detected by both detectors, any performance difference between them should appear in deviations from the straight line $y = -0.6 + x$. Fig. 6.9 shows this plot where max. SNR obtained within the signal window is used.

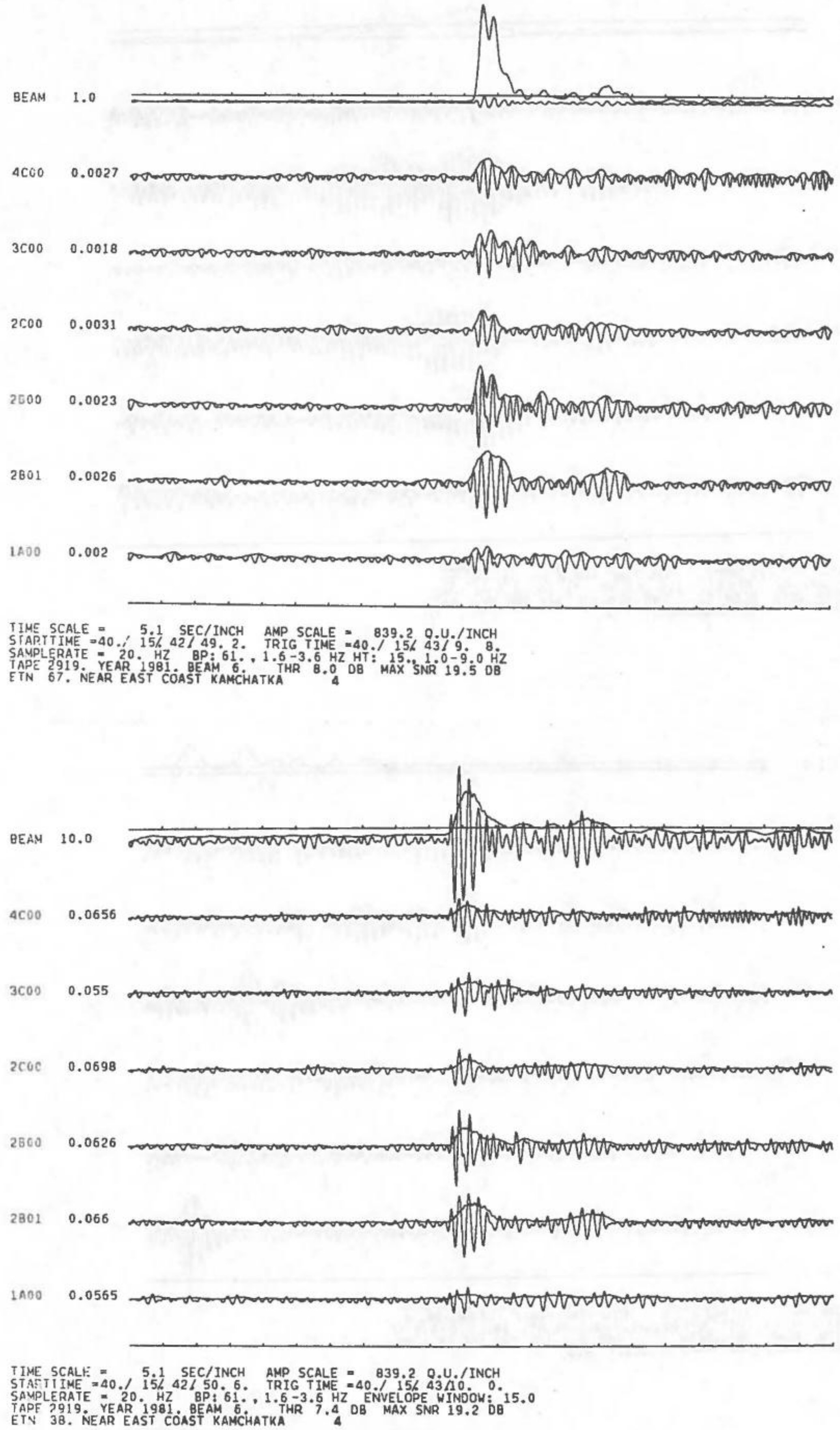
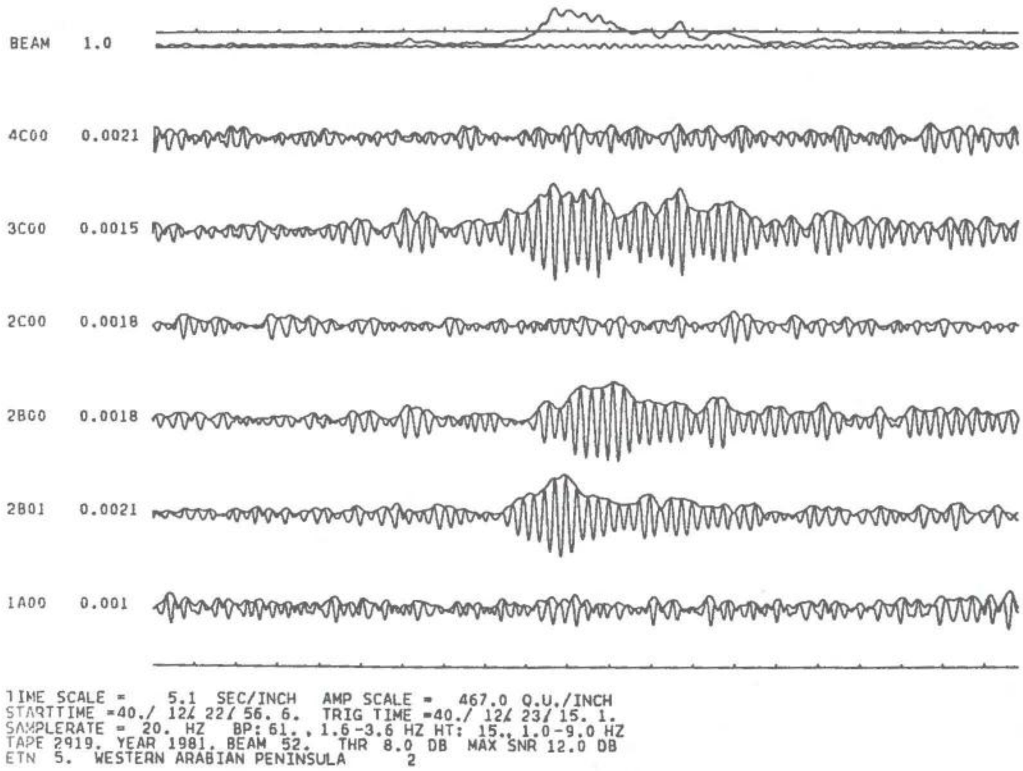


Fig. 6.7 A teleseismic event determined to Near East Coast Kamchatka by both the square envelope detector (top) and STA detector (bottom). According to the NORSAR bulletin the right source location is Komandorsky Islands in the Aleutians.



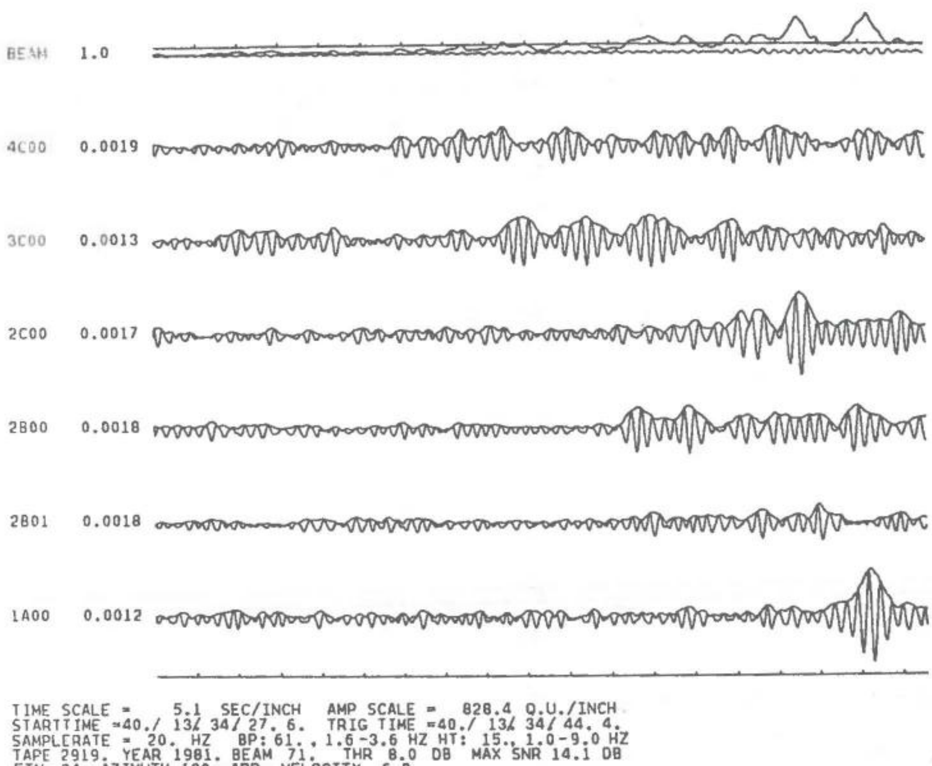


Fig. 6.8 Cases where the beamforming failed to give proper signal alignment. The top picture shows signals coming from a local explosion, possibly from Glomma since the 2B and 3C subarrays are located near to this river (Fig. 1.1).

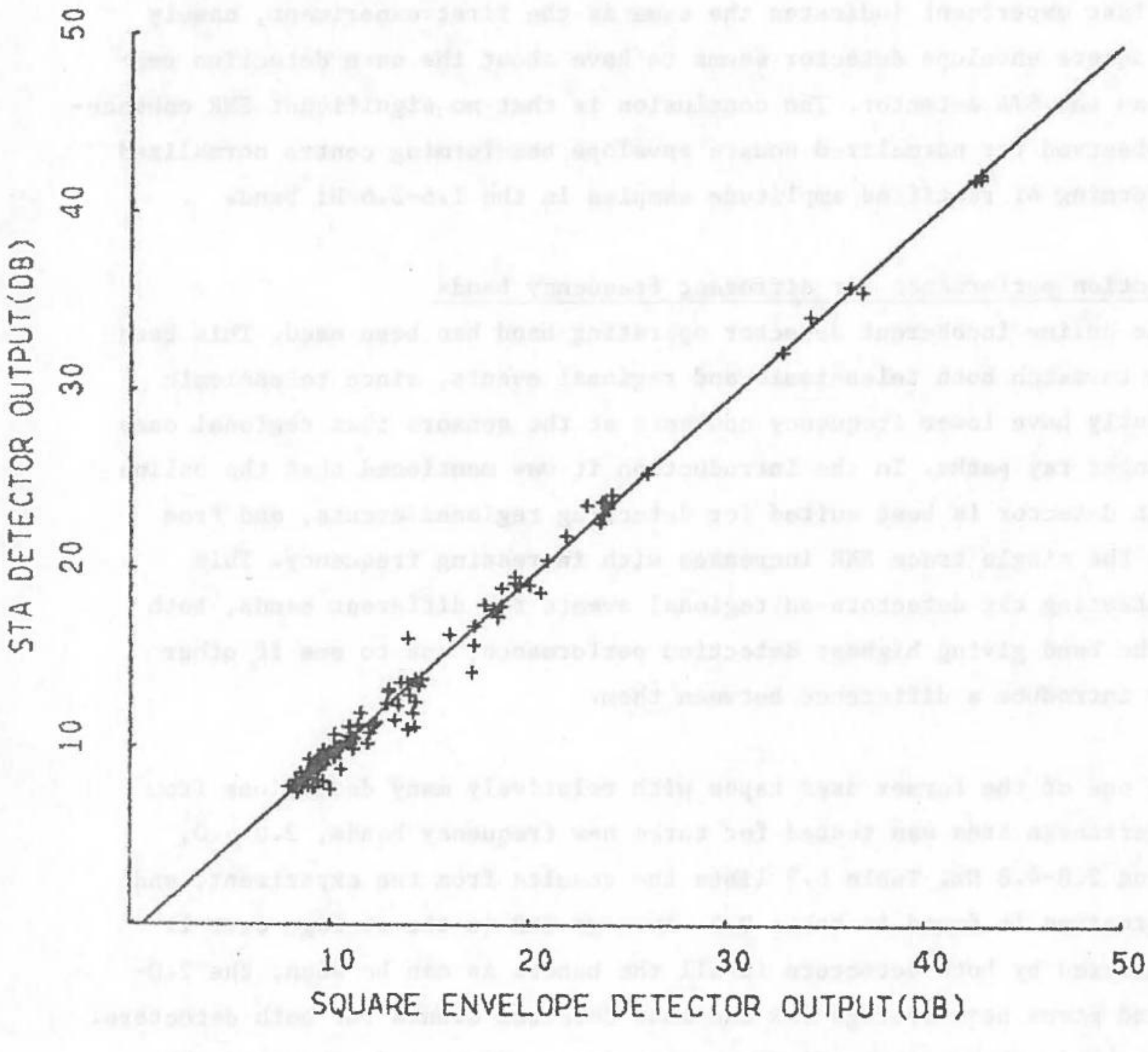


Fig. 6.9 Max. SNR from the square envelope detector versus max. SNR from the STA-detector for signals detected by both. The equation for the straight line is $y = -0.6 + x$.

No trend away from the straight line is observed. Nor can any expected theoretical better small signal detectability for the square envelope detector be inferred from Fig. 6.9. The average SNR for the square envelope detector was 15.03 and for the STA detector 14.24, subtracting 0.6 from 15.03

gives 14.43 or 1.3 per cent in favor of the square envelope detector. Thus the last experiment indicates the same as the first experiment, namely that the square envelope detector seems to have about the same detection performance as the STA detector. The conclusion is that no significant SNR enhancement is observed for normalized square envelope beamforming contra normalized STA beamforming of rectified amplitude samples in the 1.6-3.6 Hz band.

6.5 Detection performance for different frequency bands

So far the online incoherent detector operating band has been used. This band is chosen to match both teleseismic and regional events, since teleseismic events mostly have lower frequency contents at the sensors than regional ones due to longer ray paths. In the introduction it was mentioned that the online incoherent detector is best suited for detecting regional events, and from chapter 4 the single trace SNR increases with increasing frequency. This suggests testing the detectors on regional events for different bands, both to find the band giving highest detection performance, and to see if other bands may introduce a difference between them.

Therefore one of the former used tapes with relatively many detections from the Mediterranean area was tested for three new frequency bands, 2.0-4.0, 2.4-4.4 and 2.8-4.8 Hz. Table 6.3 lists the results from the experiment, and more information is found in table D.2. Average SNR is the average over 11 events detected by both detectors in all the bands. As can be seen, the 2.0-4.0 Hz band gives best average SNR and most detected events for both detectors. For higher bands the average SNR decreases, due to decreased energy in the corresponding frequency components of the detected signals, but it seems to decrease faster for the STA-detector. This indicates that the square envelope detector is superior to the STA-detector for very high frequency bands, however, for such bands the many detections due to local disturbances are troublesome. So from the experiment the 2.0-4.0 Hz band seems to be the best one for detecting regional events, and the detection performance in this band is about equal for the two detectors.

Frequency	1.6-3.6 Hz		2.0-4.0 Hz		2.4-4.4 Hz		2.8-4.8 Hz	
Bands	S-E	STA	S-E	STA	S-E	STA	S-E	STA
No. of detected events	19	19	22	22	19	16	17	16
Average SNR	19.13	18.58	20.90	20.14	20.78	19.69	19.70	18.45
SNR difference	0.55		0.76		1.09		1.25	

Table 6.3

7. DISCUSSION

To summarize, the channel normalization seems to give stationary noise detector output for both the square envelope detector and STA-detector. The former is more statistically tractable, while the latter has faster processing time. The detectors seems to have about the same detection capability. The explanation to this is that the gain in signal enhancement obtained with square envelope calculation relative to STA-calculation has been lost in a corresponding increase in noise variability. Compared to Hilbert-generated envelopes, square envelope beamforming was preferred to linear envelope beamforming due to better small signal detectability and lesser computing time. With 6 channels and 8.0 dB threshold, the beam value must exceed 75.7 for the square envelope detector to be triggered. Assuming equal signal envelopes across the sensor outputs, each envelope amplitude will be about 3.6 and the argument to ln $I_o(x)$ is probably out on the linear part of the curve in Fig. 3.3 when the threshold is exceeded. Thus a linear envelope detector is a better optimum approximation for the threshold used, and also for other realistic thresholds to be set, and this explains why no better small signal detectability was observed. However, theoretical work on pulsed radar detection has shown a quadratic detector to be roughly as efficient as a linear one (Helstrom, VII, 2c), and the relatively slow operation of square root extraction makes a linear Hilbert-generated envelope detector less interesting for real time applications.

The square envelope detector and STA-detector were compared from a detection point of view. But in a real time detection processing environment also other potentials included in the methods must be valued, since automatic parameter measurements are performed for detected events. The square envelope calculation involves Hilbert transforming to get complex waveforms, and thus makes it possible to separate the amplitude and angle information, while from the STA-method only approximated envelopes are obtained. The square envelopes and instantaneous frequencies may serve as a valuable tool for automatic determination of arrival time and secondary phases (see

Farnbach, 1975), and thus indirectly for source location. In all the cases the square envelopes may be used for a first rough determination, and the corresponding spikes in the instantaneous frequency spectra, due to inclusion of a new component in the rotating complex waveforms, as a second refinement.

Of particular interest in this connection is the proposal of a global seismic network for surveillance of a comprehensive nuclear test ban treaty (see Ringdal, 1981). In this proposal international data centers are thought to process data from a world network of seismic stations, and envelope beamforming is specially suited since the broad main lobe makes the method more robust than coherent beamforming against explosions occurring at unpredictable sites, and also against spatial aliasing due to large sensor separation. One of the remaining problems for such a system is the need for better algorithms for automatic parameter measurement, and complex signal processing may represent a potential improvement.

This thesis is the result of one of the first explorations on complex signal processing performed at NORSAR, and it was natural to employ it on detection processing. Much research about the method remains, but it seems clear that a possible extended potential of the method does not lie particularly within detection processing, but rather with detection processing combined with automatic parameter measurement.

REFERENCES

Bergland, G.D. (1968): A fast fourier transform algorithm using base 8 iterations. Math. Computation, 22, 275-279.

Bergland, G.D., & M.T. Dolan (1979): Fast fourier transform algorithms. Programs for digital signal processing, IEEE Press, John Wiley & Sons, New York, 1.2-1 - 1.2-18.

Berteussen, K.-A. (1972): Seismic event detection problems with special reference to the NORSAR array. A Thesis in Geophysics. University of Bergen.

Berteussen, K.-A. (1975): P-wave amplitude variability at NORSAR. J. Geophys., 41, 595-613.

Berteussen, K.-A. (1979): GF 280 bølgebevegelser.

Bungum, H., E.S. Husebye & F. Ringdal (1971): The NORSAR array and preliminary results of data analysis. Geophys. J.R. astr. Soc., 25, 115-126.

Capon, J. (1973): Signal processing and frequency-wavenumber spectrum analysis for a large aperture seismic array. Methods in Computational Physics, 13, Academic Press, New York, 1-59.

Cizek, V. (1970): Discrete Hilbert transform. IEEE Trans. on Audio and Electroac., AU-18, No. 4, 340-343.

Davies, D. (1973): Seismology with large arrays. Reports on progress in physics, 36, The Institute of Physics, Bristol, England, 1233-1283.

Dugundji, J. (1958): Envelopes and preenvelopes of real waveforms. IRE Trans. on Information Theory, IT-4, 53-57.

Erdelyi, A., et al (1954): Tables of Integral Transforms, Vol. 1, McGraw-Hill Book Co., New York.

Farnbach, J.S. (1975): The complex envelope in seismic signal analysis. Bull. Seism. Soc. Am., 65, 951-962.

Fuller, W.A. (1976): Introduction to Statistical Time Series. John Wiley & Sons, New York.

Fyen, J. (1978): On the problem of false alarm reduction in real time processing of seismic data. A Thesis in Geophysics, University of Oslo.

Gjøystdal, H., & E.S. Husebye (1974): A comparison of performance between prediction error and bandpass filters. Geological Institute, Tech. & Econ. Studies, D Series, 10, Bucharest.

Helstrom, C.W. (1968): Statistical Theory of Signal Detection. Pergamon Press, Oxford.

Høyland, A. (1977): Sannsynlighetsregning og statistisk metodelære, I & II, Tapir, Trondheim.

IBM (1972): Seismic Array Design Handbook. ARPA Order No. 800, Contract No. F19628-68-C-0400.

Kapittel III og IV: Forelesninger i GF 180 Den faste jords fysikk.

Lacoss, R.T. (1975): Review of some techniques for array processing. Proc. NATO ASI Exploitation of Seismograph Networks, Series E, 11, Nordhoff-Leiden, 11-21.

Lilliefors, H.W. (1967): On the Kolmogorov-Smirnov test for normality with mean and variance unknown, J.A.S.A., 62, 399-402.

McClelland, J.H., & T.W. Parks (1973): A unified approach to the design of optimum FIR linear phase digital filters. IEEE Trans. on Circuit Theory, CT-20, No. 6, 697-701.

McClelland, J.H., T.W. Parks & L.R. Rabiner (1973): A computer program for designing optimum FIR linear phase digital filters. IEEE Trans. on Audio and Electroac., AU-21, No. 6, 506-526.

Oppenheim, A.V. & R.W. Schafer (1975): Digital Signal Processing. Prentice-Hall, Englewood Cliffs, New Jersey.

Parks, T.W., & J.H. McClelland (1972): Chebyshev approximation for nonrecursive digital filters with linear phase. IEEE Trans. on Circuit Theory, CT-19, No. 2, 189-194.

Rabiner, L.R. & R.W. Schafer (1974): On the behavior of minimax FIR digital Hilbert transformers. Bell Syst. Tech. J., 53, No. 2, 363-390.

Ringdal, F., E.S. Husebye & A. Dahle (1974): Event problems using a partially coherent seismic array. Geological Institute Tech. & Econ. Studies, D series, Bucharest.

Ringdal, F., E.S. Husebye & A. Dahle (1975): P-wave envelope representation in event detection using array data. Proc. NATO ASI Exploitation of Seismograph Networks, Series E, 11, Nordhoff-Leiden, 353-372.

Ringdal, F. & H. Bungum (1977): Noise level variation at NORSAR and its effect on detectability. Bull. Seism. Soc. Am., 67, 479-492.

Ringdal, F. (1981): Automatic processing methods in the analysis of data from a global seismic network. Proc. NATO ASI Identification of Seismic Sources - Earthquake or Underground Explosions, D. Reidel Publ. Co., Holland, 787-810.

Rubin, W.L., & J.V. DiFranco (1963): Analytic representation of wide band radio frequency signals. J. of the Franklin Institute, 275, 197-204.

Shen, W.W. (1974): Comparison of coherent and incoherent beamforming envelope detectors for NORSAR regional seismic events. Technical Report No. 6, Texas Instruments Report No. ALEX(01)-TR-74-06, AFTAC Contract No. F08606-74-C-0033, Texas Instruments Inc., Dallas, Texas.

Snedecor, G.W. & W.G. Cochran (1967): Statistical Methods. The Iowa State University Press, Ames, Iowa.

Steinert, O., E.S. Husebye & H. Gjøystdal (1975): Noise variance fluctuations and earthquake detectability. J. Geophys., 41, 289-302.

Sverdrup, E. (1964): Lov og Tilfeldighet, I & II, Universitetsforlaget.

Tjøstheim, D. (1975): Some autoregressive models for short-period seismic noise. Bull. Seism. Soc. Am., 65, 677-691.

Unger, R. (1981): The instantaneous amplitude, phase and frequency in seismic event detection, timing and identification. Proc. NATO ASI Identification of Seismic Sources - Earthquake or Underground Nuclear Explosions, D. Reidel Publ. Co., Holland, 649-662.

Van Trees, H.L. (1968): Detection, Estimation, and Modulation Theory, Part 1, John Wiley & Sons, New York.

The Hilbert transform

We want to transfer a real waveform $x(t)$ to complex representation, in such a way that $x(t)$ becomes the real part: $z(t) = x(t)+j\tilde{x}(t)$. Generally we can represent $x(t)$ in terms of a time varying envelope and angle as $x(t) = R(t)\cos\Theta(t)$, and by $\tilde{x}(t)$ we will think an expression of form $\tilde{x}(t) = R(t)\sin\Theta(t)$. We then get $z(t) = R(t)\cos\Theta(t)+j\,R(t)\sin\theta(t) = R(t)e^{j\Theta(t)}$ where $R(t) = \left[x(t)^2+\tilde{x}(t)^2\right]^{\frac{1}{2}}$ and $\Theta(t) = tg^{-1}\,\tilde{x}(t)/x(t) \qquad \Theta(t) \in (-\pi,\pi]$

Mathematical $\tilde{x}(t)$ is obtained from $x(t)$ by means of the Hilbert transform. It is defined by:

$$\tilde{x}(t) = H\{x(t)\} = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{x(\tau)}{t-\tau}\,d\tau = \frac{1}{\pi t} * x(t) \tag{A.1}$$

where the Cauchy principal value is used. All waveforms considered here are assumed to have both Fourier and Hilbert transforms. We see that $x(t)$ is convolved with $h(t) = 1/\pi t$. The frequency response of $h(t)$ can be evaluated by help of a comp contour integral. Let Γ_z be the contour in the upper complex half plane drawn in Fig. A.1. Then ($z = t+iy$, $y > 0$, $\Omega > 0$, $\Omega=2\pi f$: analogous angle frequency)

$$\int_{\Gamma_z} \frac{e^{j\Omega z}}{\pi z}\,dz = \lim_{\substack{R\to\infty\\ \varepsilon\to 0}}\Big[\int_{-R}^{-\varepsilon} \frac{e^{j\Omega t}}{\pi t}\,dt + \int_{\Gamma_\varepsilon} \frac{e^{j\Omega z}}{\pi z}\,dz +$$

$$\int_{\varepsilon}^{R} \frac{e^{j\Omega t}}{\pi t}\,dt + \int_{\Gamma_R} \frac{e^{j\Omega z}}{\pi z}\,dz\Big] = 0$$

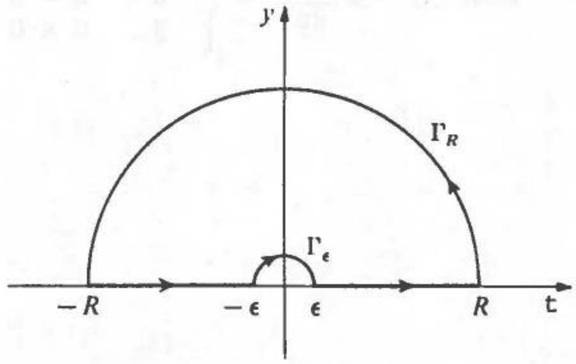


Fig. A.1 Integration path

$$\lim_{R\to\infty} \int_{\Gamma_R} \frac{e^{j\Omega z}}{\pi z}\, dz = 0$$

since $\lim_{R\to\infty} \left|\int_{\Gamma_R} \frac{e^{j\Omega z}}{\pi z}\, dz\right| < \lim_{R\to\infty} \int_{\Gamma_R} \frac{e^{-\Omega y}}{\pi R}\, dR = 0$

$$\lim_{\varepsilon\to 0}\left[\int_{\Gamma_\varepsilon} \frac{e^{j\Omega z}}{\pi z}\, dz\right] = \lim_{\varepsilon\to 0}\left[-\int_{-\Gamma_\varepsilon} \frac{e^{j\Omega z}}{\pi z}\, dz\right] = \lim_{\varepsilon\to 0}\left[-\tfrac{1}{2}\, 2j\cdot 1\right] = -j$$

Since when $\Omega > 0$

$$F\{\frac{1}{\pi t}\} = \int_{-\infty}^{\infty} \frac{e^{-j\Omega t}}{\pi t}\, dt = -\int_{-\infty}^{\infty} \frac{e^{j\Omega t}}{\pi t}\, dt = \lim_{\substack{R\to\infty \\ \varepsilon\to o}}\left[-\int_{-R}^{-\varepsilon} \frac{e^{j\Omega t}}{\pi t}\, dt\right.$$

$$\left. - \int_{\varepsilon}^{R} \frac{e^{j\Omega t}}{\pi t}\, dt\right] = -j$$

and when $\Omega = 0$,

$$F\{\frac{1}{\pi t}\} = \int_{-\infty}^{\infty} \frac{1}{\pi t}\, dt = \lim_{\substack{R\to\infty \\ \varepsilon\to o}}\left[\int_{-R}^{\varepsilon} \frac{1}{\pi t}\, dt + \int_{\varepsilon}^{R} \frac{1}{\pi t}\, dt\right] = \lim_{\substack{R\to\infty \\ \varepsilon\to o}}[0] = 0$$

and when $\Omega < 0$

$$\int_{-\infty}^{\infty} \frac{e^{-j\Omega t}}{\pi t}\, dt = \int_{-\infty}^{\infty} \frac{e^{j|\Omega| t}}{\pi t}\, dt = j$$

we get:

$$F\{h(t)\} = F\{\frac{1}{\pi t}\} = \begin{cases} -j & \Omega > 0 \\ 0 & \Omega = 0 \\ j & \Omega < 0 \end{cases} \qquad \text{(A.2)}$$

$$F\{\tilde{x}(t)\} = \begin{cases} -jX(j\Omega) = |X(j\Omega)|e^{j(\arg X(j\Omega)-\pi/2)} & \Omega > 0 \\ 0 & \Omega = 0 \\ jX(j\Omega) = |X(j\Omega)|e^{j(\arg X(j\Omega)+\pi/2)} & \Omega < 0 \end{cases} \quad \text{(A.3)}$$

where $X(j\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt$, and $z(t)$:

$$F\{z(t)\} = \begin{cases} 2X(j\Omega) & \Omega > 0 \\ X(j\Omega) & \Omega = 0 \\ 0 & \Omega < 0 \end{cases} \quad \text{(A.4)}$$

We see that the effect of Hilbert transforming is to change the argument values $-\pi/2$ for positive frequencies, and $\pi/2$ for negative frequencies. Specially for $x(t) = \cos(\Omega_o t+\phi)(\ \Omega_o > 0,\ \phi \in (-\pi,\pi]\ )$:

$$F\{x(t)\} = \pi[\delta(\Omega-\Omega_o) + \delta(\Omega+\Omega_o)]\ e^{j\Omega\phi/\Omega_o}$$

where $\delta()$ is the Dirac delta function.

$$\tilde{x}(t) = F^{-1}\{\tilde{x}(j\Omega)\} = \frac{\pi}{2\pi}\left[\int_{-\infty}^{0} j\delta(\Omega+\Omega_o)e^{j\Omega(t+\phi/\Omega_o)}d\Omega +\right.$$

$$\left.\int_{0}^{\infty} -j\delta(\Omega-\Omega_o)e^{j\Omega(t+\phi/\Omega_o)}d\Omega\right] = \frac{1}{2}\left[\int_{-\infty}^{0}\delta(\Omega+\Omega_o)e^{j\Omega(t+\phi/\Omega_o)+\pi/2}d\Omega +\right.$$

$$\left.\int_{0}^{\infty}\delta(\Omega-\Omega_o)e^{j\Omega(t+\phi/\Omega_o)-\pi/2}d\Omega\right] = \frac{1}{2}\left[e^{-j(\Omega_o t+\phi-\pi/2)} + e^{j(\Omega_o t+\phi-\pi/2)}\right]$$

$$= \cos(\Omega_o t+\phi-\pi/2) = \sin(\Omega_o t+\phi)$$

We are now in a position to give a general expression for any real band-limited waveform $x(t)$, in terms of a lowpass waveform $y(t)$. Let $y(t)$ be limited to $[-\Omega_1,\Omega_1]$, $\Omega_1>0$. Then $w(t) = y(t)+j\tilde{y}(t)$ will be limited to $[o,\Omega_1]$, and $w(t)e^{j\Omega_o}$ to $[\Omega_o,\Omega_o+\Omega_1]$, $\Omega_o>0$. If $y(t)$ has the same frequency shape from 0 to Ω_1 as $x(t)$ from Ω_o to $\Omega_o+\Omega_1$, and the Fourier transform of $x(t)$ is 0 elsewhere on the positive frequency axis, $x(t)$ can be expressed by:

$$x(t) = \mathrm{Re}\left[w(t)e^{j\Omega_o t}\right] = y(t)\cos\Omega_o t - \tilde{y}(t)\sin\Omega_o t = r(t)\cos(\Omega_o t+\phi(t)) \tag{A.5}$$

$$r(t) = \left[y^2(t)+\tilde{y}^2(t)\right]^{\frac{1}{2}} \tag{A.6}$$

$$\phi(t) = tg^{-1}\frac{\tilde{y}(t)}{y(t)} \qquad \phi(t) \in (-\pi,\pi] \tag{A.7}$$

This corresponds to single sideband modulation (SSB) in communication theory context.

The question to ask now is if $\tilde{x}(t)$ is equal to $r(t)\sin(\Omega_o t + \phi(t))$. We will show that the answer is yes, because $x(t)$ is band-limited. Our desired complex representation is:

$$u(t) = r(t)\cos(\Omega_o t+\phi(t)) + jr(t)\sin(\Omega_o t+\phi(t)) \tag{A.8}$$

and any differences $d(t)$ between $z(t)$ and $u(t)$ must be in the imaginary part:

$$d(t) = \tilde{x}(t) - r(t)\sin(\Omega_o t+\phi(t)) \tag{A.9}$$

We have

$$r(t)\sin(\Omega_o t+\phi(t)) = r(t)\sin\Omega_o t\cos\phi(t) + r(t)\cos\Omega_o t\sin\phi(t) =$$

$$y(t)\sin\Omega_o t + \tilde{y}(t)\cos\Omega_o t = \mathrm{Im}\left[w(t)e^{j\Omega_o t}\right]$$

and

$$F\{Im[w(t)e^{j\Omega_o t}]\} = \frac{-j}{2}[M(j\Omega)-M^*(-j\Omega)]$$

$$F\{Re[w(t)e^{j\Omega_o t}]\} = \frac{1}{2}[M(j\Omega)+M^*(-j\Omega)]$$

where

$$M(j\Omega) = F\{w(t)e^{j\Omega_o t}\}$$

Then if $D(j\Omega) = F\{d(t)\}$:

$$D(j\Omega) = \begin{cases} -jM^*(-j\Omega) & \Omega > 0 \\ 0 & \Omega = 0 \\ jM(j\Omega) & \Omega < 0 \end{cases} \tag{A.10}$$

$$d(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} D(j\Omega)\, e^{j\Omega t}\, d\Omega =$$

$$\frac{j}{2\pi}\left[\int_{-\infty}^{0} M(j\Omega)e^{j\Omega t}d\Omega - \int_{0}^{\infty} M^*(-j\Omega)e^{j\Omega t}d\Omega\right] \tag{A.11}$$

Since $M(j\Omega) = 0$ for $\Omega \notin [\Omega_o, \Omega_o+\Omega_1]$, $\Omega_o>0$, $\Omega_1>0$, we see that the above expression is equal to zero. Further, since all waveforms considered in this thesis will be band-limited, the Hilbert transform will give the 'right' expression for the imaginary part.

A discrete version of the Hilbert transform is needed, in order to be able to implement it on a computer. One algorithm has been given by Cizek (1970), and it has been used in some seismological applications. It is the so-called discrete Hilbert transform (DHT), and has the following values in the frequency domain:

$$H(k) = \begin{cases} -j & k=1,2,\ldots\ldots, \frac{N}{2}-1 \\ 0 & k=0,\frac{N}{2} \qquad\qquad (N \text{ even}) \\ j & k=\frac{N}{2}+1,\ldots\ldots, N-1 \end{cases} \tag{A.12}$$

Note we must require $H(k) = H^*(N-k)$, $k=0,..,N$, and $H(0) = H(N)$ to get a real filter. We evaluate what this becomes in the time domain:

$$h(h) = \frac{1}{N} \sum_{k=1}^{N-1} H(k) e^{j(2\pi/N)kn} =$$

$$\frac{1}{N} \left[\sum_{k=0}^{N/2-1} -je^{j(2\pi/N)kn} + \sum_{k=N/2+1}^{N-1} je^{j(2\pi/N)kn} \right] =$$

$$\frac{j}{N} \left[\sum_{k=1}^{N/2-1} -e^{j(2\pi/N)kn} + \sum_{k=1}^{N/2-1} e^{j(2\pi/N)(k+N/2)n} \right] =$$

$$\frac{j}{N} \left[e^{j\pi n} - 1 \right] \sum_{k=1}^{N/2-1} e^{j(2\pi/N)kn} =$$

$$\frac{j}{N} \left[e^{j\pi n} - 1 \right] \frac{e^{j(2\pi/N)n} - e^{j(2\pi/N)N/2}}{1 - e^{j(2\pi/N)n}} =$$

$$\frac{-j}{N} \left[e^{j\pi n} - 1 \right] \frac{1 - e^{j(2\pi/N)(N/2-n)}}{1 - e^{-j(2\pi/N)n}} =$$

$$\frac{-j}{N} \left[e^{j\pi n} - 1 \right] \frac{1 + e^{-j(2\pi/N)n}}{1 - e^{-j(2\pi/N)n}} = \frac{-1}{N} \left[e^{j\pi n} - 1 \right] \frac{\cos \pi/N\, n}{\sin \pi/N\, n}$$

$$= \begin{cases} 0 & n=0,2,\ldots,N-2 \\ \frac{2}{N} \quad \operatorname{cotg} \pi/N\, n & n=1,3,\ldots,N-1 \end{cases} \qquad \text{(A.13)}$$

There are some disadvantages in using this filter for block filtering of big data sequences. To get the frequency response above, the number of filter coefficients must be equal to the number of samples used in the FFT algorithm to calculate the Discrete Fourier Transform (DFT). Shorter lengths will give unsatisfactory oscillations between the ideal values, and these oscillations will increase with decreasing lengths. When the ideal DHT is used in frequency domain filtering, the convolution will then be circular, not linear, and the imaginary sequence will vary depending on the length used in the FFT. So we conclude that the DHT is not suited for block filtering, and an approximation must be tried. This will imply a smoother crossing between j and -j, but that does not necessarily matter, since we will be occupied by band-limited waveforms.

FIR linear phase Chebyshev approximation and the base 8 fast Fourier transform

B.1 FIR linear phase Chebyshev approximation

We want to find the unit-sample response for a causal FIR system with linear phase, i.e., h(n)=0,...,N-1 such that

$$h(n) = h(N-1-n) \tag{B.1}$$

Assume we want a lowpass filter of odd length, N=2M+1. Applying the Chebyshev approximation method, the passband and stopband cutoff frequencies can be specified in advance, $0<\omega_p<\omega_s<\pi$, and the approximation error to the wanted frequency response is spread out uniformly in frequency with a maximum defviation given as a function of N, ω_s, and ω_p. The frequency response of the filter is

$$H(e^{j\omega}) = \sum_{n=0}^{2M} h(n)e^{-j\omega n} = \sum_{n=0}^{M-1} h(n)e^{-j\omega n} + \sum_{n=0}^{M-1} h(2M-n)e^{-j\omega(2M-n)} + h(M)e^{-j\omega M}$$

$$= e^{-j\omega M}\Big[\sum_{n=0}^{M-1} h(n)(e^{-j\omega(n-M)} + e^{-j\omega(M-n)}) + h(M)\Big]$$

$$= e^{-j\omega M} \sum_{n=0}^{M} d(n) \cos \omega n = e^{-j\omega M}\, G(\omega) \tag{B.2}$$

where $G(\omega)$ is a real function and

$$d(n) = \begin{cases} 2h(M-n) & n=1,\ldots,M \\ h(M) & n=0 \end{cases} \tag{B.3}$$

We see that the symmetry property in (B.1) implies linear phase. To determine the d(n)'s, we define an error function $E(\omega)$:

$$E(\omega) = W(\omega)\left[D(\omega)-G(\omega)\right] \qquad \omega \in F$$

$$= W(\omega)\left[D(\omega)-\sum_{n=0}^{M} d(n)\cos\omega n\right] \tag{B.4}$$

$$F = \{\omega|\omega \in [0,\omega_p] \cup [\omega_s,\pi]\}$$

$W(\omega)$ is a weighting function and $D(\omega)$ the desired frequency response

$$D(\omega) = \begin{matrix} 1 & 0 \leq \omega \leq \omega_p \\ 0 & \omega_s \leq \omega \leq \pi \end{matrix} \tag{B.5}$$

The design procedure then requires an algorithm for minimizing:

$$\max_{\omega \in F} |E(\omega)| \tag{B.6}$$

When $G(\omega)$ is expressed as a finite linear sum of cosine functions as above, a necessary and sufficient condition for this is given in a theorem formulated by Parks and McClellan (1972):

$E(\omega)$ must exhibit on F at least M+2 alternations, i.e.,

$E(\omega_i) = -E(\omega_{i-1}) = \pm\max_{\omega \in F}|E(\omega)|$, $i=1,\ldots,M+1$

and $0 \leq \omega_o < \ldots < \omega_{M+1} \leq \pi$. Then $G(\omega)$ will be the unique best weighted Chebyshev approximation to $D(\omega)$ on F for a given choice of N, ω_p and ω_s.

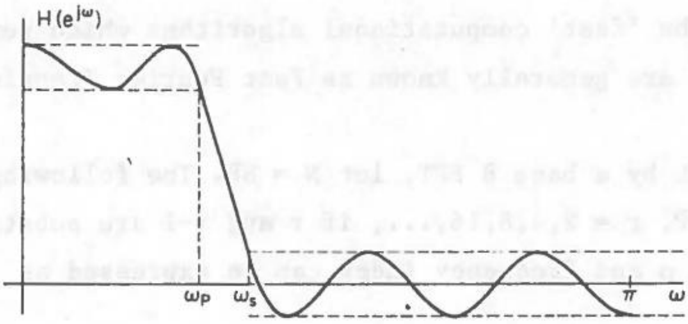


Fig. B.1 Possible optimum lowpass filter approximation for M=7.

If a computer program can determine values of the d(n)'s so the condition above is satisfied, the problem will be solved. The minimizing problem can be solved either by linear programming technique, or by using an iterative Remez exchange algorithm. The McClellan, Parks and Rabiner program uses the Remez method, which is far more efficient than the former (McClellan and Parks, 1973).

In the cases we want an odd symmetry filter (Hilbert transformer), or a filter with N an even integer, it is possible (McClellan and Parks, 1973) to express $G(\omega)$ as $G(\omega) = Q(\omega)\,P(\omega)$ where $P(\omega)$ is a linear combination of cosine functions. The error function can then be rewritten in the form:

$$E(\omega) = W(\omega)\left[D(\omega)-G(\omega)\right] = W(\omega)Q(\omega)\left[\frac{D(\omega)}{Q(\omega)} - P(\omega)\right] \qquad \omega \in F' \qquad \text{(B.7)}$$
$$F' = F - \{\omega \mid Q(\omega)=0\}$$

and the new problem is solved as in the first case.

B.2 The base 8 fast Fourier transform

The expression for the discrete Fourier Transform (DFT) is

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn} \qquad k = 0,\dots,N-1 \tag{B.6}$$

where $W_N = e^{-j(2\pi/N)}$. The 'fast' computational algorithms which perform the DFT and inverse DFT are generally known as Fast Fourier Transform or FFT.

To explain what is meant by a base 8 FFT, let $N = 8^p$. The following formulas will be valid for $N = r^p$, $r = 2,4,8,16,\dots$, if r and r-1 are substituted for 8 and 7. The time index n and frequency index can be expressed as

$$n = n_{p-1}8^{p-1} + n_{p-2}8^{p-2} + \dots + n_o = (n_{p-1}, n_{p-2},\dots,n_o)$$

$$k = k_{p-1}8^{p-1} + k_{p-2}8^{p-2} + \dots + k_o = (k_{p-1}, k_{p-2},\dots,k_o) \tag{B.7}$$

$$n,k \in [0,1,\dots N-1], \text{ and } n_j,k_j \in [0,1,\dots,7], \quad j=0,\dots,p-1$$

Then

$$X(k) = X(k_{p-1},\dots,k_o) = \sum_{n=o}^{N-1} x(n)W_N^{kn}$$

$$= \sum_{n_o=o}^{7} \dots \sum_{n_{p-2}=o}^{7} \sum_{n_{p-1}=o}^{7} x(n_{p-1}, n_{p-2},\dots,n_o)W_N^{(k_{p-1},\dots,k_o)(n_{p-1},n_{p-2},\dots,n_o)}$$

$$= \sum_{n_o=o}^{7} \dots \sum_{n_{p-2}=o}^{7} \left[\sum_{n_{p-1}=o}^{7} x(n_{p-1},\dots,n_o)W_N^{k_o(n_{p-1},\dots,n_o)} \right]$$

$$W_N^{(k_{p-1},\dots,k_1)(n_{p-2},\dots,n_o)} \tag{B.8}$$

Remark that

$$W_N^{(k_{p-1},\ldots,k_1)(n_{p-1},\ldots,n_o)} = W_N^{n_{p-1}8^p(k_{p-1}8^{p-2}+\ldots+k_1)}\ W_N^{(k_{p-1},\ldots,k_1)(n_{p-2},\ldots,n}$$

$$= W_N^{(k_{p-1},\ldots,k_1)(n_{p-2},\ldots,n_o)}$$

Let us denote

$$\hat{A}_1(k_o,n_{p-2},\ldots,n_o) = \sum_{n_{p-1}=o}^{7} x(n_{p-1},\ldots,n_o)\ W_N^{k_o(n_{p-1},\ldots,n_o)}$$

Then for the next stage of computation

$$X(k_{p-1},\ldots,k_o) = \sum_{n_o=o}^{7} \cdots \sum_{n_{p-3}=0}^{7} \left[\sum_{n_{p-2}=o}^{7} \hat{A}_1(k_o,n_{p-2},\ldots,n_o) W_N^{k_1 8(n_{p-2},\ldots,n_o)} \right]$$

$$W_N^{(k_{p-1},\ldots,k_2)(n_{p-3},\ldots,n_o)}$$

or

$$\hat{A}_2(k_o,k_1,n_{p-3},\ldots,n_o) = \sum_{n_{p-2}=o}^{7} \hat{A}_1(k_o,n_{p-2},\ldots,n_o) W_N^{k_1 8(n_{p-2},\ldots,n_o)}$$

Generally for the ℓ-th stage

$$\hat{A}_\ell(k_o,\ldots,k_{\ell-1},n_{p-\ell-1},\ldots,n_o) =$$

$$\sum_{n_{p-\ell}=o}^{7} \hat{A}_{\ell-1}(k_o,\ldots,k_{\ell-2},n_{p-\ell},\ldots,n_o) W_N^{k_{\ell-1}8^{\ell-1}(n_{p-\ell},\ldots,n_o)} \qquad \text{(B.9)}$$

$$\ell = 1,\ldots,p$$

$$\hat{A}_o(n_{p-1},\ldots,n_o) = x(n_{p-1},\ldots,n_o) \qquad \text{(B.10)}$$

$$\hat{A}_p(k_o,k_1,\ldots,k_{p-1}) = X(k_{p-1},\ldots,k_o) \qquad \text{(B.11)}$$

The ℓ-th stage can be written as an 8-term Fourier transform multiplied with a twiddle factor.

$$\hat{A}_\ell(k_o,\dots,k_{\ell-1},n_{p-\ell-1},\dots,n_o) =$$

$$W_N^{k_{\ell-1}8^{\ell-1}(n_{p-\ell-1},\dots,n_o)} \sum_{n_{p-\ell}=o}^{7} \hat{A}_{\ell-1}(k_o,\dots,k_{\ell-2},n_{p-\ell},\dots,n_o)$$

$$W_N^{k_{\ell-1}8^{\ell-1}n_{p-\ell}8^{p-\ell}} =$$

$$W_N^{k_{\ell-1}8^{\ell-1}(n_{p-\ell-1},\dots,n_o)} \sum_{n_{p-\ell}=o}^{7} \hat{A}_{\ell-1}(k_o,\dots,k_{\ell-2},n_{p-\ell},\dots,n_o)W_8^{k_{\ell-1}n_{p-\ell}} \quad \text{(B.12)}$$

Now it is easy to understand why the base 8 FFT is so effective. Let the values of $W_8^{k_{\ell-1}n_{p-\ell}}$ be drawn in the complex plane. From Fig. B.2 the only numbers to multiply with in the sum are $\pm\sqrt{0.5}$. When programming the algorithm, there will be 4 multiplications in each 8 term Fourier transform. In addition $W_N^{k_{\ell-1}8^{\ell-1}(n_{p-\ell-1},\dots,n_o)} = 1$ for $k_{\ell-1}=o$, or $(n_{p-\ell-1},\dots,n_o) = o$.

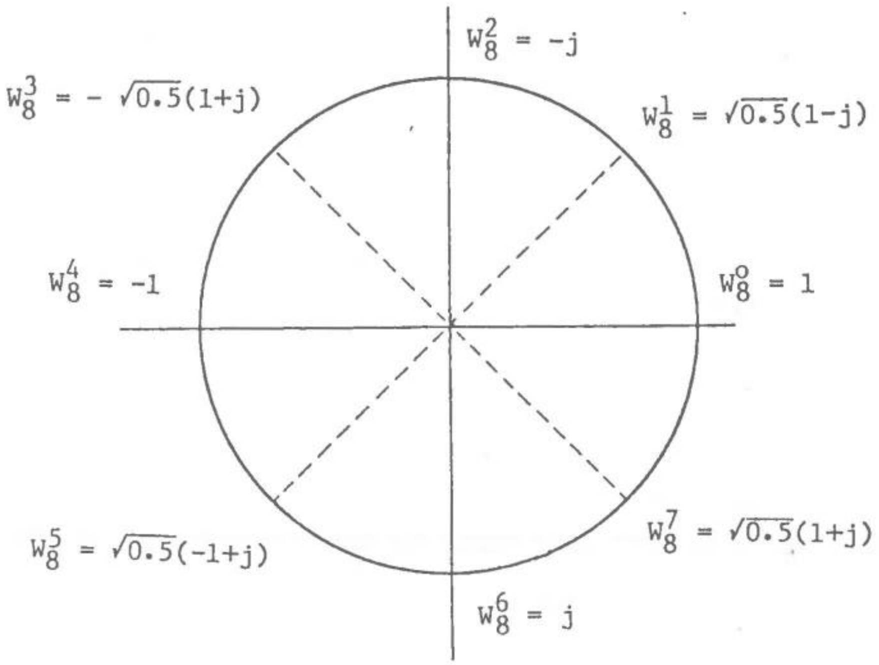


Fig. B.2 Illustration of the data points arranged on the unit circle in the complex plane.

Some more comments must be given to (B.12). Since the $k_{\ell-1}$ term in $\hat{A}_\ell$ has the same position as $n_{p-\ell}$ in $\hat{A}_{\ell-1}$, the computations can be done 'in place'. When the recursion is finished, X(k) where $k = k_{p-1}8^{p-1}+ \ldots +k_o$ will be in the relative position $k_o 8^{p-1}+\ldots+k_{p-1}$ in the array. If X(k) is desired in the k-th position, the X(k)'s must be reordered, called generalized bit reversing. The twiddle factor

$$W_N^{k_{\ell-1}8^{\ell-1}(n_{p-\ell-1},\ldots,n_o)} = W_{8^{p-\ell+1}}^{k_{\ell-1}(n_{p-\ell-1},\ldots,n_o)}$$

can be interpreted as a frequency rotation due to a time shift, in analogy with with that the DFT of x(n-m) is $W_N^{km}X(k)$ when the DFT of x(n) is X(k). If $N = 2^M = 8 \ldots 8{\cdot}4$ or $8 \ldots 8{\cdot}2$, either a base 4 or base 2 iteration must finish the DFT computation.

Detection probability of the square envelope detector

From (3.14) the noise and signal and noise alternatives for the incoherent signal model can be expressed, leaving out the time indication, as

$$H_o : z_\ell = \left[v_\ell^2 + \tilde{v}_\ell^2\right]^{\frac{1}{2}}$$

$$H_1 : z_\ell = \left[(s_\ell + v_\ell)^2 + (\tilde{s}_\ell + \tilde{v}_\ell)^2\right]^{\frac{1}{2}} \qquad \ell=1,\ldots,L \tag{C.1}$$

$$r_\ell = \left[s_\ell + \tilde{s}_\ell\right]^{\frac{1}{2}}$$

v_ℓ, $\tilde{v}_\ell$, and s_ℓ, $\tilde{s}_\ell$ denote the real and imaginary part of the complex noise field and complex signal field at sensor ℓ respectively, and z_ℓ the waveform envelope at sensor ℓ, all sampled at the same time instant. L is the number of sensors and r_ℓ the instantaneous value of the signal envelope at senor ℓ. The noise samples v_ℓ are assumed uncorrelated and identical Gaussian distributed across the array, i.e.,

$$v_\ell \sim N(0,\sigma) \quad \ell=1,\ldots,L \qquad E\left[v_\ell v_i\right] = \begin{matrix} \sigma^2 & i = \ell \\ 0 & i \neq 1 \end{matrix} \tag{C.2}$$

From chapter 5 $\tilde{v}_\ell$ is uncorrelated with v_ℓ and also Gaussian $(0,\sigma)$ when $\tilde{v}_\ell$ is generated from the real samples with a Hilbert transformer. Since Gaussian distributed variables are independent when they are uncorrelated, the real and imaginary noise samples can be considered as independent and identically distributed at the same time instant.

We will first find the probability distribution for z_ℓ when a signal is present. From the above the joint probability distribution of v_ℓ and $\tilde{v}_\ell$ is given by

$$p_{v_\ell \tilde{v}_\ell}(v_\ell,\tilde{v}_\ell) = p_{v_\ell}(v_\ell)p_{\tilde{v}_\ell}(\tilde{v}_\ell)$$

$$= \frac{1}{2\pi\sigma^2} e^{-1/2\sigma^2 (v_\ell{}^2+\tilde{v}_\ell{}^2)} \qquad \text{(C.3)}$$

By changing to new coordinates

$$v_\ell = \psi_1(z_\ell,\theta_\ell)$$

$$\tilde{v}_\ell = \psi_2(z_\ell,\theta_\ell)$$

where the variables are related by

$$z_\ell \cos\theta_\ell = s_\ell + v_\ell \qquad \theta \in (0,2\pi]$$

$$z_\ell \sin\theta_\ell = \tilde{s}_\ell + \tilde{v}_\ell \qquad z_\ell \in [0,\infty]$$

the joint probability distribution for z_ℓ and θ_ℓ is given by (Sverdrup, II)

$$p_{z_\ell \theta_\ell}(z_\ell,\theta_\ell) = p_{v_\ell \tilde{v}_\ell}(\psi_1(z_\ell,\theta_\ell),\psi_2(z_\ell,\theta_\ell)) \begin{vmatrix} \frac{\partial\psi_1}{\partial z_\ell} & \frac{\partial\psi_1}{\partial\theta_\ell} \\ \frac{\partial\psi_2}{\partial z_\ell} & \frac{\partial\psi_2}{\partial\theta_\ell} \end{vmatrix} \qquad \text{(C.4)}$$

Since

$$v_\ell{}^2+\tilde{v}_\ell{}^2 = z_\ell{}^2+s_\ell{}^2+\tilde{s}_\ell{}^2-2z_\ell(s_\ell \cos\theta_\ell+\tilde{s}_\ell \sin\theta_\ell)$$

we get

$$p_{z_\ell\theta_\ell}(z_\ell,\theta_\ell) = \frac{1}{2\pi\sigma^2} e^{-1/2\sigma^2\left[z_\ell^2+r_\ell^2-2z_\ell(s_\ell \cos\theta_\ell+\tilde{s}_\ell \sin\theta_\ell)\right]}$$

$$\cdot \begin{vmatrix} \cos\theta_\ell & -z_\ell \sin\theta_\ell \\ \sin\theta_\ell & z_\ell \cos\theta_\ell \end{vmatrix}$$

$$= \frac{z_\ell}{2\pi\sigma^2} e^{-1/2\sigma^2\left[z_\ell^2+r_\ell^2\right]} e^{z_\ell/\sigma^2\left[s_\ell \cos\theta_\ell+\tilde{s}_\ell \sin\theta_\ell\right]} \tag{C.5}$$

Setting

$$s_\ell = r_\ell \cos\mu_\ell$$

$$\tilde{s}_\ell = r_\ell \sin\mu_\ell$$

and noting that

$$s_\ell \cos\theta_\ell + \tilde{s}_\ell \sin\theta_\ell = r_\ell \cos(\theta_\ell-\mu_\ell)$$

the probability density for z_ℓ can be found by integrating over θ_ℓ

$$p_{z_\ell}(z_\ell) = \frac{z_\ell}{\sigma^2} e^{-1/2\sigma^2\left[z_\ell^2+r_\ell^2\right]} \frac{1}{2\pi}\int_o^{2\pi} e^{\frac{z_\ell r_\ell}{\sigma^2}\cos(\theta_\ell-\mu_\ell)} d\theta_\ell$$

$$= \frac{z_\ell}{\sigma^2} e^{-1/2\sigma^2\left[z_\ell^2+r_\ell^2\right]} \frac{1}{2\pi}\int_o^{2\pi} e^{\frac{z_\ell r_\ell}{\sigma^2}\cos\theta_\ell} d\theta_\ell$$

$$= \begin{cases} \dfrac{z_\ell}{\sigma^2} e^{-1/2\sigma^2\left[z_\ell^2+r_\ell^2\right]} I_o\left(\dfrac{z_\ell r_\ell}{\sigma^2}\right) & z_\ell > 0,\ r_\ell > 0 \\ 0 & \text{Otherwise} \end{cases} \tag{C.6}$$

The function $I_o(\alpha) = \frac{1}{2\pi} \int_o^{2\pi} e^{\alpha\cos\theta}\, d\theta$ is called the modified Bessel function of the first kind and zero order, and the probability density for z_ℓ the Rician distribution. Setting r = 0, it reduces to a Rayleigh distribution

$$p_{z_\ell}(z_\ell) = \begin{cases} \frac{z_\ell}{\sigma^2} e^{-\frac{z_\ell^2}{2\sigma^2}} & z_\ell > 0 \\ 0 & \text{Otherwise} \end{cases} \qquad (C.7)$$

Our test statistic is given by

$$\eta = \sum_{\ell=1}^{L} z_\ell^2 \qquad (C.8)$$

where the source area is assumed known and the sensor waveforms delayed accordingly. We will further assume the variance normalized, i.e., $\sigma = 1$. A convenient method of finding the probability density function (PDF) of η is to use the characteristic function. For a random variable x with probability density p(x), it is defined by

$$h_x(z) = E\left[e^{jzx}\right] = \int_{-\infty}^{\infty} p(x)\, e^{jzx}\, dx \qquad (C.9)$$

for all values of z where the integral exists. It is simply the Fourier transform of the PDF, and if it is known, the PDF can be found by the inverse transformation

$$p(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} h_x(z)\, e^{-jzx}\, dz \qquad (C.10)$$

Since the sum of two independent random variables x and y is equal to the convolution of their PDF's, the characteristic function is by common Fourier transform theory equal to the product of the individual characteristic function $h_{x+y}(z) = h_x(z)h_y(z)$.

The square envelope detector resembles the quadratic threshold detector used in pulsed radars. We will therefore closely follow the derivation for this detector described in Helstrom (1968). Under H_1 the test statistic is

$$\eta = \sum_{\ell=1}^{L} z_\ell^2 = \sum_{\ell=1}^{L} [y_\ell^2+\tilde{y}_\ell^2] = \sum_{\ell=1}^{L} [(s_\ell+v_\ell)^2+(\tilde{s}_\ell+\tilde{v}_\ell)^2]$$

$$= \sum_{\ell=1}^{L} [(r_\ell \cos \mu_\ell+v_\ell)^2 + (r_\ell \sin \mu_\ell+\tilde{v}_\ell)^2]$$

where $y_\ell+j\tilde{y}_\ell$ is the complex input sample at sensor ℓ. The PDF of y_ℓ and $\tilde{y}_\ell$ is given by

$$p_{Y_\ell}(y_\ell) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y_\ell-r_\ell \cos \mu_\ell)^2}$$

$$p_{\tilde{Y}_\ell}(y_\ell) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\tilde{y}_\ell-r_\ell \sin \mu_\ell)^2}$$

and the characteristic function of η is

$$h_\eta(z) = \prod_{\ell=1}^{L} h_{Y_\ell^2}(z) \prod_{\ell=1}^{L} h_{\tilde{Y}_\ell^2}(z)$$

$$= \prod_{\ell=1}^{L} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{[-\frac{1}{2}(x-r_\ell\cos \mu_\ell)^2+izx^2]} dx \quad \cdot$$

$$\prod_{\ell=1}^{L} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{[-\frac{1}{2}(x-r_\ell\sin \mu_\ell)^2+izx^2]} dx$$

$$= \prod_{\ell=1}^{L} (1-2iz)^{-\frac{1}{2}} e^{\left(\frac{izr_\ell^2\cos^2 \mu_\ell}{1-2iz}\right)} \prod_{\ell=1}^{L} (1-2iz)^{-\frac{1}{2}} e^{\left(\frac{izr_\ell^2\sin^2 \mu_\ell}{1-2iz}\right)}$$

$$= (1-2iz)^{-L} e^{\left(\frac{iz\sum_{\ell=1}^{L} r_\ell^2}{1-2iz}\right)} \qquad (C.13)$$

Note that the μ_ℓ's have disappeared in the calculation $\sum_{\ell=1}^{L} r_\ell^2 = S_p$ is equal to the sum of the instantaneou square envelope values of the signal across the sensors. Looking up in a transform table (Erdelyi, 1954), the PDF for η under H_1 is

$$p(\eta|H_1) = \tfrac{1}{2}\left(\frac{\eta}{S_p}\right)^{(L-1)/2} e^{-\left(\frac{\eta+S_p}{2}\right)} I_{L-1}(\sqrt{\eta S_p}) \qquad (C.14)$$

$I_{L-1}(x)$ is the modified Bessel function of order L-1, given in series expansion by

$$I_{L-1}(x) = \sum_{k=o}^{\infty} \frac{(x/2)^{L-1+2k}}{k!(L-1+k)!} \qquad (C.15)$$

Inserting (C.15) in (C.14) and setting S_p equal to 0, the PDF for η under H_o is

$$p(\eta|\ H_o) = \tfrac{1}{2}\left(\frac{\eta}{S_p}\right)^{(L-1)/2} e^{-\left(\frac{\eta+S_p}{2}\right)} \sum_{k=o}^{\infty} \left.\frac{\left(\frac{\eta S_p}{4}\right)^{(L-1)/2+k}}{k!(L-1+k)!}\right|_{S_p=o}$$

$$= \frac{1}{2}\, e^{-\left(\frac{\eta+S_p}{2}\right)} \sum_{k=o}^{\infty} \frac{(\frac{1}{2})^{L-1+2k}\, \eta^{L-1+k}\, S_p{}^{k}}{k!\,(L-1+k)!} \Bigg|_{S_p = 0}$$

$$= \frac{1}{2^L (L-1)!}\, \eta^{L-1}\, e^{-\eta/2} \tag{C.16}$$

We see that η is chi-squared distributed with 2L degrees of freedom. That could have been predicted in advance, since x^2 is chi-squared distributed with 1 degree of freedom when x is normal (0,1). The mean and variance of η under H_o is then

$$E[\eta|H_o] = L$$
$$Var[\eta|H_o] = 2L \tag{C.17}$$

and the false alarm probability for a threshold $\eta > 0$

$$p_F = \int_{\eta_o}^{\infty} p(\eta|H_o)\, d\eta = \int_{\eta_o}^{\infty} \frac{1}{2^L(L-1)!}\, \eta^{L-1} e^{-\eta/2}\, d\eta \tag{C.18}$$

We see that p_F is a function of the number of sensors and η_o.

Similarly for the detection probability

$$p_D = \int_{\eta_o}^{\infty} p(\eta|H_1)\, d\eta = \int_{\eta_o}^{\infty} \frac{1}{2}\left(\frac{\eta}{S_p}\right)^{(L-1)/2} e^{-\left(\frac{\eta+S_p}{2}\right)}\, I_{L-1}(\sqrt{\eta S_p})\, d\eta \tag{C.19}$$

p_D is a function of $S_p = \sum_{\ell=1}^{L} r_\ell{}^2$, L, and η_o.

S_p can approximately be estimated from the data by

$$S_p \approx \max_n \sum_{\ell=1}^{L} (z_\ell(n)^2-2) = \max_n \sum_{\ell=1}^{L} z_\ell(n)^2-2L \tag{C.19}$$

where 2 is the expectation of $z_\ell(n)^2$ when only noise is available. max means the sample time which gives highest SNR within the signal window when the correct beam has been determined. Note that the PDF's for the test statistic in (C.8) have been derived completely independently of the assumption made in chapter 3 of equal signal envelopes. Our only requirement is independent, normalized Gaussian distributed noise samples.

Event listing

The table D.1 lists the events detected by the square envelope detector or STA detector in the 1.6-3.6 Hz band, and classified as signals. In the cases where corresponding entries existed in the NORSAR bulletin, NORSAR estimated arrival time, magnitude and source area have been inserted. The SNR values are max. SNR within the signal window taken over all beams. Table D.2 lists events detected from tape 9218 in the bands 1.6-3.6, 2.0-4.0, 2.4-4.4, and 2.8-4.8 Hz.

TAPE	YEAR	DAY	TIME	S-E SNR	STA SNR	MAGN	SOURCE LOCATION
1374	1981	30	7/31/55.4		7.7		LOCAL
1374	1981	30	8/22/29.1	14.1	11.0	4.3	ALEUTIAN ISLANDS REGION
1374	1981	30	9/ 3/33.7	43.0	42.0	6.3	RAT ISLANDS
1374	1981	30	9/14/12.9	11.2	10.6	4.4	RAT ISLANDS
1374	1981	30	9/23/12.5	24.0	23.1	5.7	ALEUTIAN ISLANDS
1374	1981	30	9/31/52.6	10.5	9.8	6.3	RAT ISLANDS PKPPKPB
1374	1981	30	9/37/29.2	9.6	9.4		UNKNOWN SOURCE
1374	1981	30	9/42/ 4.6	14.5	13.8	4.4	RAT ISLANDS
1374	1981	30	9/45/35.3	8.8		4.0	RAT ISLANDS
1374	1981	30	10/22/36.1	10.7	8.8		UNKNOWN SOURCE
1374	1981	30	10/48/44.8	9.6	9.2	4.3	RAT ISLANDS
1374	1981	30	11/ 9/ 5.9	20.8	18.7	4.2	OFF EAST COAST HONSHU JAPAN
2919	1981	40	12/10/30.8	9.4	7.8		UNKNOWN SOURCE
2919	1981	40	12/20/ 0.0	14.1	13.6		LOCAL
2919	1981	40	12/23/15.1	12.5	11.4		LOCAL
2919	1981	40	12/33/59.8	24.1	23.8	4.8	NEAR EAST COAST KAMCHATKA
2919	1981	40	12/43/55.6	9.8	8.1	4.4	KOMANDORSKY ISLANDS
2919	1981	40	12/52/52.1	8.6	8.0		LOCAL
2919	1981	40	12/58/21.2	36.4	35.9	5.7	KOMANDORSKY ISLANDS
2919	1981	40	13/ 9/53.3	12.1	10.8	3.9	NEAR EAST COAST KAMCHATKA
2919	1981	40	13/34/44.4	14.1	16.1		LOCAL
2919	1981	40	13/36/29.9	11.2	10.2	4.2	NEAR EAST COAST KAMCHATKA
2919	1981	40	13/39/53.9	22.1	21.9	4.6	KOMANDORSKY ISLANDS
2919	1981	40	13/40/35.1	11.1	10.6		KOMANDORSKY ISLANDS PCP
2919	1981	40	13/41/ 9.9	10.4	10.7		KOMANDORSKY ISLANDS
2919	1981	40	13/45/42.1		7.6		LOCAL
2919	1981	40	14/ 0/40.4	11.7	11.9	4.3	NEAR EAST COAST KAMCHATKA
2919	1981	40	14/11/13.5	10.0	9.4	4.0	KOMANDORSKY ISLANDS
2919	1981	40	15/ 3/49.1	10.4	10.0		LOCAL
2919	1981	40	15/ 6/28.8	34.4	34.2	5.2	NEAR EAST COAST KAMCHATKA
2919	1981	40	15/10/21.8	17.4	15.7	4.1	NEAR EAST COAST KAMCHATKA
2919	1981	40	15/26/17.0	14.4	11.1	4.5	TONGA ISLANDS
2919	1981	40	15/43/ 8.3	19.5	19.2	4.6	KOMANDORSKY ISLANDS
2919	1981	40	15/54/ 3.3	9.6	8.9		LOCAL
9069	1981	55	21/58/40.7	8.5	7.5	3.1	GREECE
9069	1981	55	22/20/56.9	8.5		3.3	YUOGOSLAVIA
9069	1981	55	22/31/32.5	8.9	8.3	3.5	GREECE
9069	1981	55	22/34/54.5	18.8	17.9	3.9	AEGAN SEA
9069	1981	55	22/39/12.0	11.4	10.5	3.4	GREECE-ALBANIA BORDER REG.
9069	1981	55	22/47/ 7.5	9.8	9.1	3.5	SOUTHERN GREECE
9069	1981	55	22/55/55.1	11.6	11.2	3.3	ROMANIA
9069	1981	55	23/ 7/ 5.7	14.8	13.8	3.7	GREECE
9069	1981	55	23/15/17.1	14.2	12.9	4.4	SOUTH OF FIJI ISLANDS PKPB
9069	1981	55	23/22/29.1	16.2	16.3	3.9	YUOGOSLAVIA
9069	1981	55	23/59/15.4		7.4	3.2	GREECE
9069	1981	56	0/ 2/42.0	9.4	8.9		LOCAL
9069	1981	56	0/43/16.6		7.6		MEDITERANEAN AREA
9069	1981	56	1/ 5/34.4	9.0	8.6	3.3	GREECE
9069	1981	56	1/ 7/58.5		19.4	4.1	SOUTH OF KERMADEC ISL. PKPB
9069	1981	56	1/20/44.2	13.1	13.1	4.0	GREECE
9069	1981	56	1/30/53.3	8.1	7.8		MEDITERANEAN AREA
9069	1981	56	1/36/28.8		7.8		MEDITERANEAN AREA
9069	1981	56	1/45/ 2.2	18.0	18.0	4.5	EASTERN KASHMIR
9218	1981	56	2/ 3/ 8.3	24.2	23.5	4.8	GREECE
9218	1981	56	2/12/35.2	9.8	9.8		LOCAL
9218	1981	56	2/20/33.5	10.1	9.7	3.3	GREECE
9218	1981	56	2/35/13.9	24.4	24.2	4.3	AEGAN SEA
9218	1981	56	2/41/ 6.3	42.7	41.9	5.8	AEGAN SEA
9218	1981	56	2/56/55.9	10.3	9.4	3.7	GREECE
9218	1981	56	3/ 1/52.8	10.3	9.6	3.4	YUOGOSLAVIA
9218	1981	56	3/23/37.6	11.0	10.3	3.7	GREECE
9218	1981	56	3/37/47.0	8.7	7.7	3.9	DODECANESE ISLANDS
9218	1981	56	3/44/10.0	19.7	19.1	3.8	GREECE
9218	1981	56	3/51/31.8		7.7		UNKNOWN SOURCE
9218	1981	56	4/18/35.4	11.3	9.9		LOCAL
9218	1981	56	4/30/ 2.0	9.5	8.7		LOCAL
9218	1981	56	4/35/29.2	23.9	23.8	4.0	YUOGOSLAVIA
9218	1981	56	4/43/32.3	8.2		3.2	GREECE
9218	1981	56	4/52/37.3	8.7	8.5		UNKNOWN
9218	1981	56	5/ 0/55.7	10.7	10.0	3.5	GREECE
9218	1981	56	5/ 7/16.5	9.9	9.0		LOCAL
9218	1981	56	5/ 7/39.5	9.3	7.7	3.4	TURKEY
9218	1981	56	5/13/26.9	33.0	32.2	4.4	YUOGOSLAVIA
12511	1981	63	20/49/19.7	10.1	7.6		LOCAL
12511	1981	63	22/ 3/18.3	43.0	42.2	6.1	AEGAN SEA
12511	1981	63	22/19/41.5	17.4	16.8	4.0	GREECE
12511	1981	63	22/29/52.2	8.5	7.7	3.9	SOUTHERN GREECE
12511	1981	63	22/36/20.8	19.5	19.6	4.1	YUOGOSLAVIA
12511	1981	63	22/39/43.9	12.3	11.2	3.9	GREECE
12511	1981	63	22/43/28.8	9.3	8.4	3.5	YUOGOSLAVIA
12511	1981	63	22/48/23.4	9.5	8.8	3.6	YUOGOSLAVIA
12511	1981	63	22/52/42.0	18.8	18.9	4.1	GREECE
12511	1981	63	23/ 1/57.5	10.5	9.9	4.0	AEGAN SEA
12511	1981	63	23/ 4/46.9	13.6	12.3	3.9	YUOGOSLAVIA
12511	1981	63	23/10/ 0.7	21.1	20.5	4.5	YUOGOSLAVIA
12511	1981	63	23/12/ 0.2	13.0			MEDITERANEAN AREA
12511	1981	63	23/15/45.3	8.5	7.8	3.0	GREECE
12511	1981	63	23/23/ 2.8	9.1	9.3	3.6	YUOGOSLAVIA
12511	1981	63	23/27/23.6	13.0	12.4	4.0	GREECE
12511	1981	64	0/ 4/33.1	13.7	13.6	3.9	GREECE BULGARIA BORDER
12511	1981	64	0/ 7/46.5		8.4	3.5	YUOGOSLAVIA
12517	1981	64	0/24/59.7	8.3		3.5	GREECE
12517	1981	64	0/40/36.4	11.0	10.3	3.7	YUOGOSLAVIA
12517	1981	64	0/55/10.3	13.4	11.5	4.1	KURILE ISLANDS
12517	1981	64	0/55/51.6	13.5		4.1	GREECE
12517	1981	64	1/15/27.6	23.8	22.6	4.1	GREECE
12517	1981	64	1/27/ 0.4	23.1	23.6	4.9	WESTERN MEDITERANEAN SEA
12517	1981	64	2/10/38.9	8.4	8.1		MEDITERANEAN AREA
12517	1981	64	2/58/59.7	11.1	11.1	4.0	GREECE-BULGARIA BORDER
12517	1981	64	3/10/50.4	9.3	8.7		LOCAL
12242	1981	115	5/32/53.3	17.3	14.2	4.2	LOYALTY ISLANDS REGION PKPD
12242	1981	115	5/49/21.4	20.2	19.1	5.0	VANUATU ISLANDS PKPD
12242	1981	115	5/51/ 4.0	12.0	10.2	4.4	SOUTH OF FIJI ISLANDS PKPB
12242	1981	115	5/54/58.2	37.0	35.6	5.3	CAROLINE ISLANDS PKPD
12242	1981	115	6/ 6/17.6	9.6	9.4	5.3	CAROLINE ISLANDS PKKPA
12242	1981	115	6/17/34.5	14.4	12.5	3.6	IRAN
12242	1981	115	6/31/16.2	8.3	7.8	5.3	CAROLINE ISLANDS P5KP
12242	1981	115	6/33/54.3	9.5	8.1	3.9	COOK STRAIT NEW ZEALAND PKPB
12242	1981	115	6/38/33.2	9.3	9.0		LOCAL
12242	1981	115	6/50/25.3	9.5	8.8	4.7	VANUATU ISLANDS PKPD
1600	1981	123	13/ 7/46.5	26.2	25.4	4.6	SOUTH OF FIJI ISLANDS PKPB
1600	1981	123	13/10/ 8.4	9.0	7.8		UNKNOWN SOURCE
1600	1981	123	13/10/45.0	14.3	11.9	4.6	SOUTH OF FIJI ISLANDS SKPB
1600	1981	123	13/35/16.8	18.6	17.3	3.9	CRETE
1600	1981	123	13/56/49.8	13.0	12.4	4.3	OAXACA MEXICO
1600	1981	123	14/ 3/31.6	18.4	17.7	3.9	CRETE
1600	1981	123	14/ 5/12.4	8.9			UNKNOWN SOURCE
1600	1981	123	14/35/16.0	13.1	13.2	3.9	OAXACA MEXICO
1600	1981	123	14/40/37.2	8.5			UNKNOWN SOURCE
1600	1981	123	14/49/ 0.4	8.4			UNKNOWN SOURCE
1600	1981	123	14/53/30.7	8.0			UNKNOWN SOURCE
1600	1981	123	14/56/ 1.1	8.2			UNKNOWN SOURCE
1600	1981	123	15/17/55.6	14.6	13.4	3.5	CRETE

Table D.1

YEAR DAY TIME	1.6-3.6 S-E SNR	1.6-3.6 STA SNR	2.0-4.0 S-E SNR	2.0-4.0 STA SNR	2.4-4.4 S-E SNR	2.4-4.4 STA SNR	2.8-4.8 S-E SNR	2.8-4.8 STA SNR	MAGN	SOURCE
81 56/1/49/42.5			8.3	7.4	8.9	8.1	9.6	8.7	3.6	KIRGHIZ SSR
81 56/2/ 3/ 8.3	24.2	23.5	27.6	26.4	26.1	24.9	23.7	22.9	4.8	GREECE
81 56/2/11/58.1					9.4		12.2	9.2		LOCAL
81 56/2/12/35.2	9.8	9.8	14.7	13.3	17.8	14.8	18.6	16.8		LOCAL
81 56/2/20/33.5	10.1	9.7	10.9	10.5	10.8	10.6	9.8	9.2	3.3	GREECE
81 56/2/35/13.9	24.4	24.2	25.7	25.6	24.0	23.5	22.2	21.0	4.3	AEGAN SEA
81 56/2/41/ 6.3	42.7	41.9	44.7	43.8	46.5	45.4	44.9	43.5	5.8	AEGAN SEA
81 56/2/56/55.9	10.3	9.4	11.7	10.6	11.3	10.3	10.8	10.4	3.7	GREECE
81 56/3/ 1/52.8	10.3	9.6	12.7	11.5	13.0	12.1	11.9	10.6	3.4	YUOGOSLAVIA
81 56/3/ 8/41.1			10.4	10.4	12.7	12.3	12.7	12.2	3.0	GREECE
81 56/3/23/37.6	11.0	10.3	13.5	12.2	13.4	12.4	12.4	10.6	3.7	GREECE
81 56/3/37/47.0	8.7	7.7	9.8	9.8	10.0	9.6	8.1		3.9	DODECANESE ISLAND
81 56/3/44/10.0	19.7	19.1	22.6	22.0	24.3	22.7		20.0	3.8	GREECE
81 56/3/51/31.8		7.7	10.3	7.7	10.0		11.3	7.7		LOCAL
81 56/4/18/35.4	11.3	9.9	12.6	11.2						LOCAL
81 56/4/21/12.5			8.9	8.0						LOCAL
81 56/4/30/ 2.0	9.5	8.7	11.3	9.8						LOCAL
81 56/4/35/29.2	23.9	23.8	25.8	25.7	26.5	25.1	25.1	23.1	4.0	YUOGOSLAVIA
81 56/4/43/32.3	8.2		9.1	9.5	8.8	8.0	8.4	7.5	3.2	GREECE
81 56/4/52/37.3	8.7	8.5	10.5	9.7	8.7	8.0	8.0			UNKNOWN
81 56/5/ 0/55.7	10.7	10.0	11.7	11.5	10.7	10.0	11.0	9.7	3.5	GREECE
81 56/5/ 7/16.5	9.9	9.0	10.9	9.4						LOCAL
81 56/5/ 7/39.5	9.3	7.7	9.9	8.7	8.3				3.4	TURKEY
81 56/5/13/26.9	33.0	32.2	30.9	30.4	28.5	27.5	26.3	25.2	4.4	YUOGOSLAVIA

Table D.2

APPENDIX E

Program listing

This appendix lists the source code of the square envelope detector program, the routines FFT42 and IFFT42, and the routine which generates the coefficients to FFT42 and IFFT42. The STA-detector program is not listed, as it is quite analogous to the square envelope detector program, apart from the square envelope routine and the Hilbert transformation routine.

```
      REAL X1(512,7),Y1(512,7),X2(512,7),Y2(512,7),
     *BCOFR(257),BCOFI(257),HCOF(43)
      INTEGER TAPCOD
      LOGICAL*1 LSENS(49)/49*.FALSE./
C
      COMMON/DTP/ ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(2,4,7),ATAB(8,204),IBST,INSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
      INTEGER BMTAB
      LOGICAL TRIG,GROUP,PRINT/.TRUE./
C
      COMMON /TYPE/ NAME(132),CHAN(132),COMP(132),IRATE,OPTION,NCH,
     *EXPECT,FOUND,IND,RAD,ETIME,TAPEOP,NTCH,STIME,FLAG,LIMDTC,
     *NEWLAS,NSAMP,NCHAN,FRSTIM,UX,UY,TAU,DELAY,BEAM,IFILT,JSAMP,NTAPE
      REAL*6 NAME
      INTEGER CHAN,COMP,OPTION,EXPECT,FOUND,ETIME,TAPEOP,STIME,FRSTIM
      INTEGER*2 TAU(132)
      LOGICAL FLAG,NEWLAS,DELAY,BEAM
      EQUIVALENCE(NSAMP,IRL),(JN,NCH)
C
      COMMON/WORKER/ INPUT,BLOCK
      INTEGER*2 INPUT(2744),BLOCK
C
      COMMON/SAMPLE/SAMP(132)
      INTEGER*4 SAMP
C
      COMMON/STATUS/ SENS
      LOGICAL*1 SENS(138)
C
      COMMON /CMSTAP/ TAPX,TAPY,TAPZ
      INTEGER*4 TAPX,TAPY,TAPZ
C
      DATA IAL/512/,IA2/256/,JJ/7/
C
C FI  3 PRINTER (BLOCK 132
C FI  8 DISK HILB COF
C FI  9 DISK RANDP COF
C FI 10 DISK DLYTAB DATA
C FI 11 TAP1
C FI 12 DISK DETEC DATA
C FI 13 DISK EVENT DATA (RECFM VS LRECL 28 BLOCK 36
C FI 14 DISK SNR DATA (RECFM VS LRECL 4 BLOCK 12
C
C READ FILTER COEFFICIENTS AND OTHER DATA VALUES
C
      DO 2 I=1,15
      N2POW=I
      IF(IAL.EQ.2**N2POW) GO TO 4
  2   CONTINUE
      STOP 1000
  4   CALL GENCOF(X1(1,1),Y1(1,1),BCOFR,BCOFI,HCOF,
     *IAL,N2POW,IRL,NDEL,NOL,NHC)
      WRITE(3,110) IAL,N2POW,IRL,NDEL,NOL,NHC
      WRITE(6,110) IAL,N2POW,IRL,NDEL,NOL,NHC
      IF(NOL.GT.IRL.OR.NHC.GT.NOL) STOP 1020
      CALL SETPAR(KTAPE,TAPCOD,INC,JJ)
      NOB=2*IRL
      NTSMP=NOB
      SMI=10.0/FLOAT(IRATE)
      IFTD=(NOL/2)*SMI
      WRITE(12,170) KTAPE,IYR,NOB,SMI,DBTHR
C
C READ DATA AND SET DATA DEPENDENT PARAMETERS
C
 10   CALL PRESET(X1,IAL,JJ,1,IAL,1,JJ,0.0)
      CALL PRESET(X2,IAL,JJ,1,IAL,1,JJ,0.0)
      CALL PRESET(Y1,IAL,JJ,1,IAL,1,JJ,0.0)
      CALL PRESET(Y2,IAL,JJ,1,IAL,1,JJ,0.0)
      CALL PRESET(WT,JJ,1,1,JJ,1,1,1.0)
      CALL SPOLES(X1,IAL,JJ,KTAPE,IYR,TAPCOD,&200)
      CALL BMINP(PRINT)
      WRITE(6,130) IAL,NCH,NTCH,IFTD,NBM
      WRITE(3,130) IAL,NCH,NTCH,IFTD,NBM
      IF(.NOT.PRINT) GO TO 7
      WRITE(6,160)
      WRITE(3,160)
      CALL OUTP(6,1,1,X1,X1,IAL,JJ,1,5,1,JJ)
      CALL OUTP(3,1,1,X1,X1,IAL,JJ,1,5,1,JJ)
      PRINT=.FALSE.
  7   IYC=IRL/2+1
      NBL=1
      LTT=0
      TRIG=.FALSE.
      GROUP=.FALSE.
      IBT1=STIME-IFTD
      EVAR=NCH*4.0
      ENL=NCH*2.0
      THR=(10.0**(DBTHR/10.0))*ENL
      WRITE(6,140) ENL,EVAR,THR,DBTHR
      WRITE(3,140) ENL,EVAR,THR,DBTHR
C
C TOP OF THE LOOP
C
 15   DO 16 J=1,JN,2
 16   CALL MFILT(X1(1,J),Y1(1,J),X1(1,J+1),Y1(1,J+1),X2(1,J),Y2(1,J),
     *X2(1,J+1),Y2(1,J+1),BCOFR,BCOFI,HCOF,IAL,N2POW,IRL,NOL,NHC,2)
      IF(MOD(JN,2).NE.0)
     *CALL MFILT(X1(1,JN),Y1(1,JN),Y1(1,JN),Y1(1,JN),X2(1,JN),Y2(1,JN),
     *Y2(1,JN),Y2(1,JN),BCOFR,BCOFI,HCOF,IAL,N2POW,IRL,NOL,NHC,1)
      IF(NBL.GE.3) GO TO 19
      CALL SENV(X2,Y2,X1,Y1,WT,VES1,IAL,JN,IRL,NDEL)
      DO 17 J=1,JN
      VES3(J)=VES1(J)
      VES4(J)=VES1(J)
      VES5(J)=VES1(J)
 17   VES2(J)=VES1(J)
      CALL WTCALC(IRL,NCH)
      GO TO 22
 19   IF(.NOT.TRIG) CALL WTCALC(IRL,NCH)
      CALL SENV(X2,Y2,X1,Y1,WT,VES1,IAL,JN,IRL,NDEL)
      CALL DETECT(Y2,Y1,X2,X1,COMP,IAL,JN,IRL,IYC,INC,IBT2,IBT1)
 22   IF(.NOT.TRIG) CALL ACHECK(X1,X2,Y1,Y2,CHAN,COMP,SENS,LSENS,
     *IAL,JJ,JN,&200,&10)
C
 23   CALL SPOLES(X2,IAL,JJ,KTAPE,IYR,TAPCOD,&10)
      IBT2=STIME-IFTD
      NBL=NBL+1
C
 25   DO 30 J=1,JN,2
 30   CALL MFILT(X2(1,J),Y2(1,J),X2(1,J+1),Y2(1,J+1),X1(1,J),Y1(1,J),
     *X1(1,J+1),Y1(1,J+1),BCOFR,BCOFI,HCOF,IAL,N2POW,IRL,NOL,NHC,2)
      IF(MOD(JN,2).NE.0)
     *CALL MFILT(X2(1,JN),Y2(1,JN),Y2(1,JN),Y2(1,JN),X1(1,JN),Y1(1,JN),
     *X1(1,JN),Y1(1,JN),BCOFR,BCOFI,HCOF,IAL,N2POW,IRL,NOL,NHC,1)
      IF(NBL.GT.2) GO TO 34
      CALL SENV(X1,Y1,X2,Y2,WT,VES1,IAL,JN,IRL,NDEL)
      GO TO 37
 34   IF(.NOT.TRIG) CALL WTCALC(IRL,NCH)
      CALL SENV(X1,Y1,X2,Y2,WT,VES1,IAL,JN,IRL,NDEL)
      CALL DETECT(Y1,Y2,X1,X2,COMP,IAL,JN,IRL,IYC,INC,IBT1,IBT2)
 37   IF(.NOT.TRIG) CALL ACHECK(X1,X2,Y1,Y2,CHAN,COMP,SENS,LSENS,
     *IAL,JJ,JN,&200,&10)
C
 38   CALL SPOLES(X1,IAL,JJ,KTAPE,IYR,TAPCOD,&10)
      IBT1=STIME-IFTD
      NBL=NBL+1
      GO TO 15
C
 110  FORMAT(1X/' IAL,N2POW,IRL,NDEL,NOL,NHC=',6I5)
 130  FORMAT(1X/' IAL,NCH,NTCH,IFTD,NBM=',5I5)
 140  FORMAT(1X/' EX(BM NOISE),VAR(BM NOISE),THR,DBTHR',4F9.2)
 160  FORMAT(1X/' TEST OF THE DATA SAMPLES')
 170  FORMAT(3I10,2F10.3)
 200  CONTINUE
      END
C
      SUBROUTINE MFILT(/X1/,/Y1/,/X2/,/Y2/,/SX1/,/SY1/,/SX2/,/SY2/,
     */BCOFR/,/BCOFI/,/HCOF/,/IAL/,/N2POW/,/IRL/,/NOL/,/NHC/,/IS/)
      REAL X1(1),Y1(1),X2(1),Y2(1),SX1(1),SY1(1),
     *SX2(1),SY2(1),BCOFR(1),BCOFI(1),HCOF(1)
C
C THIS ROUTINE PERFORMS BANDPASS FILTERING IN FREQUENCY DOMAIN
C AND HILBERT TRANSFORMING IN TIME DOMAIN WITH ONE(IS=1) OR TWO
C (IS=2) INPUT TRACES
C
      K=IRL+1
      IAL2=IAL/2
      IALP=IAL+2
      IAL3=IAL2+1
      DO 10 I=K,IAL
 10   X1(I)=0.0
      IF(IS.EQ.1) K=1
      DO 20 I=K,IAL
 20   X2(I)=0.0
C
      CALL FFT842(0,N2POW,X1,X2)
      X1(1)=X1(1)*BCOFR(1)
      X2(1)=X2(1)*BCOFR(1)
      DO 30 I=2,IAL2
      J=IALP-I
      STR=X1(I)
      X1(I)=STR*BCOFR(I)-X2(I)*BCOFI(I)
      X2(I)=STR*BCOFI(I)+X2(I)*BCOFR(I)
      STR=X1(J)
      X1(J)=STR*BCOFR(I)+X2(J)*BCOFI(I)
 30   X2(J)=X2(J)*BCOFR(I)-STR*BCOFI(I)
      X1(IAL3)=X1(IAL3)*BCOFR(IAL3)
      X2(IAL3)=X2(IAL3)*BCOFR(IAL3)
      CALL FFT842(1,N2POW,X1,X2)
C
      DO 80 I=1,NOL
 80   X1(I)=X1(I)+SX1(I+IRL)
      CALL HFILT(X1,Y1,SY1(IRL+1),HCOF,IRL,NHC)
      IF(IS.EQ.1) RETURN
C
      DO 110 I=1,NOL
 110  X2(I)=X2(I)+SX2(I+IRL)
      CALL HFILT(X2,Y2,SY2(IRL+1),HCOF,IRL,NHC)
      RETURN
      END
C
C
      SUBROUTINE HFILT(/X/,/Y/,/S/,/HCOF/,/IRL/,/NHC/)
      REAL X(1),Y(1),S(1),HCOF(1)
C
C THIS ROUTINE PERFORMS HILBERT TRANSFORMING IN TIME
C DOMAIN WITH A NEGATIVE SYMMETRI HILBERT FILTER
C
      NC1=NHC-1
      NC2=(NHC+1)/2
      IR1=IRL+1
      MMM=IRL+NC1
C
      DO 20 L=1,NC1
      Y(L)=S(L)
      DO 10 K=1,NC2,2
      I=L-K+1
      IF(I.LT.1) GO TO 20
      Z=X(I)
      I=L-NHC+K
      IF(I.GT.0) Z=Z-X(I)
 10   Y(L)=Y(L)+HCOF(K)*Z
 20   CONTINUE
C
      DO 30 L=NHC,IRL
      Y(L)=HCOF(1)*(X(L)-X(L-NHC+1))
      DO 30 K=3,NC2,2
 30   Y(L)=Y(L)+HCOF(K)*(X(L-K+1)-X(L-NHC+K))
C
      DO 50 L=IR1,MMM
      Y(L)=-HCOF(1)*X(L-NHC+1)
      DO 40 K=3,NC2,2
      I=L-NHC+K
      IF(I.GT.IRL) GO TO 50
      Z=-X(I)
      I=L-K+1
      IF(I.LE.IRL) Z=Z+X(I)
 40   Y(L)=Y(L)+HCOF(K)*Z
 50   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE GENCOF(/X/,/Y/,/BCOFR/,/BCOFI/,/HCOF/,
     */IAL/,/N2POW/,/IRL/,/NDEL/,/NOL/,/NHC/)
C
C THIS ROUTINE GENERATES THE COEFFICIENTS TO BE MULTIPLIED
C WITH IN SUBROUTINE MFILT,FFT,IFFT
C
      REAL BCOFR(1),BCOFI(1),HCOF(1),
     *X(1),Y(1),STR,STI
C
      DO 20 I=1,IAL
      X(I)=0.
 20   Y(I)=0.0
C
      WRITE(3,100)
      CALL COFLES(HCOF,8,NHC)
      NDEL=(NHC-1)/2
C
      WRITE(3,110)
      CALL COFLES(X,9,NBC)
      CALL FFT842(0,N2POW,X,Y)
      IRL=IAL-NBC+1
      NOL=IAL-IRL
      S=1.0/FLOAT(IAL)
      IAL2=IAL/2+1
      DO 50 I=1,IAL2
      BCOFR(I)=X(I)*S
 50   BCOFI(I)=Y(I)*S
      RETURN
 100  FORMAT(1X/' HILBERT TRANSFORMER IMPULSE RESPONSE')
 110  FORMAT(1X/' BANDPASS FILTER IMPULSE RESPONSE')
      END
C
C
      SUBROUTINE COFLES(/X/,/NO/,/NFLT/)
C
C THIS ROUTINE READS FILTER COEFFICIENTS
C
      REAL X(1)
      REAL*8 S
      READ(NO) NFLT
      WRITE(3,100) NFLT
      DO 10 I=1,NFLT
      READ(NO) S
      WRITE(3,100) I,S
 10   X(I)=SNGL(S)
 100  FORMAT(1X,I4,F15.7)
      RETURN
      END
C
C
```

```
C
C
      SUBROUTINE DETECT(/Y1/,/Y2/,/X1/,/X2/,/COMP/,
     */IAL/,/JJ/,/IRL/,/I/,/INC/,/IBT1/,/IBT2/)
      REAL Y1(IAL,JJ),Y2(IAL,JJ),X1(IAL,JJ),X2(IAL,JJ)
C
      COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
      INTEGER BMTAB,FTT(5),COMP(1),BLT(5)
      LOGICAL TRIG,GROUP
C
C THIS ROUTINE COMPARES THE NBM BEAMS AGAINST THE THRESHOLD. FOR
C TRIGS IBTL SAMPLES AFTER LAST PRECEEDING TRIG ,MXBEAM ARE CALLED
C
      IR2=IRL/2
      IBTL=40
      IBL=0
      ICST=IBT1
      IF(NTSMP.EQ.NOB) GO TO 4
      L=NOB-NTSMP
      CALL OUTP(13,0,1,X2,X2,IAL,JJ,1,L,1,JJ)
      NTSMP=NOB
 4    IF(GROUP)
     *CALL MXBEAM(Y1,Y2,IAL,JJ,IRL,I,1,IBL,IBT1,IBT2,&50,&100,&110)
      TRIG=.FALSE.
C
 10   DO 40 K=1,NBM
      BM=0.0
      DO 35 J=1,JJ
      II=I+BMTAB(K,J)
      IF(II.GT.IRL) GO TO 30
      BM=BM+Y1(II,J)
      GO TO 35
 30   BM=BM+Y2(II-IRL,J)
 35   CONTINUE
      IF(BM.LT.THR) GO TO 40
      TRIG=.TRUE.
      ITI=(I-1)*SMI+0.5
      NTT=IBT1+ITI
      IF(NTT.GE.IBTL+LTT) GO TO 120
      LTT=NTT
      GO TO 50
 40   CONTINUE
 50   I=I+INC
      IF(I.LE.IRL) GO TO 10
      I=I-IRL
C
      IBL=1
      ICST=IBT2
 70   DO 90 K=1,NBM
      BM=0.0
      DO 85 J=1,JJ
      II=I+BMTAB(K,J)
      IF(II.LT.1) GO TO 80
      BM=BM+Y2(II,J)
      GO TO 85
 80   BM=BM+Y1(II+IRL,J)
 85   CONTINUE
      IF(BM.LT.THR) GO TO 90
      TRIG=.TRUE.
      ITI=(I-1)*SMI+0.5
      NTT=IBT2+ITI
      IF(NTT.GE.IBTL+LTT) GO TO 120
      LTT=NTT
      GO TO 100
 90   CONTINUE
 100  I=I+INC
      IF(I.LE.IR2) GO TO 70
 110  RETURN
C
C A POSSIBLE EVENT IS DETECTED
C MOVE DATA TO DISK AND CALL MXBEAM AND SPIKET
 120  GROUP=.TRUE.
      ITRC=ITRC+1
      LTT=NTT
      K1M=1
      E1(K1M)=0.0
      INS=40
      NWS=400
      IESC=0
      NTRIG=0
      IPC=1
      CALL PRESET(TME,JJ,1,1,JJ,1,1,0.0)
      CALL PRESET(EST,4,NBM,1,4,1,NBM,0.0)
      CALL PRESET(SNR,NBM,1,1,NBM,1,1,0.0)
      DO 130 J=1,NBM
 130  ITRT(J)=0
      S=10.0*ALOG10(BM/ENL)
      CALL CHANGE(ICST,BLT)
      CALL CHANGE( LTT,FTT)
      WRITE( 3,200) ITRC,K,IBL,I,S,DBTHR,BLT,FTT
      WRITE( 6,200) ITRC,K,IBL,I,S,DBTHR,BLT,FTT
      WRITE(12,210) ITRC,JJ
      WRITE(12,210) (COMP(J),J=1,JJ)
      WRITE(12,220) (WT(J),J=1,JJ)
      I=I-IEPL
      II=I-IR2
C
 140  IF(IBL.EQ.1.AND.I.GT.0) GO TO 150
      IF(IBL.EQ.1) GO TO 142
      IF(II.LT.1) II=1
      GO TO 145
 142  I=I+IRL
      II=II+IRL
 145  CALL OUTP(13,0,1,X1,X1,IAL,JJ,II,IRL,1,JJ)
      NTSMP=IRL-II+1
      ITI=(II-1)*SMI+0.5
      IBST=IBT1+ITI
      IWSS=I-II+1
      CALL OUTP(13,0,1,X2,X2,IAL,JJ,1,IRL,1,JJ)
      NTSMP=NTSMP+IRL
      IBL=0
      CALL MXBEAM(Y1,Y2,IAL,JJ,IRL,I,1,IBL,IBT1,IBT2,&50,&100,&110)
C
 150  II=IRL+II
      CALL OUTP(13,0,1,X1,X1,IAL,JJ,II,IRL,1,JJ)
      NTSMP=IRL-II+1
      IWSS=IR2+1
      ITI=(II-1)*SMI+0.5
      IBST=IBT1+ITI
      CALL OUTP(13,0,1,X2,X2,IAL,JJ,1,IRL,1,JJ)
      NTSMP=NTSMP+IRL
      CALL MXBEAM(Y1,Y2,IAL,JJ,IRL,I,1,IBL,IBT1,IBT2,&50,&100,&110)
C
 200  FORMAT(1X/' TRIG,BEAM,BLOCK,SAMPLE,SNR(DB),THR(DB)',
     *4I5,2F9.2/' BLOCK START TIME                         ',
     *5I5        /' TRIG  START TIME                         ',5I5)
 210  FORMAT(16I5)
 220  FORMAT(7F10.6)
      END
C
C
C
      SUBROUTINE SPIKET(/TME/,/NWS/,/JJ/,/SPIKE/)
      REAL TME(JJ),TMP(7)
C
C THIS ROUTINE CHECKS FOR POSSIBLE SPIKE DETECTIONS
C
      SPIKE=0
      DO 100 J=1,JJ
      TME(J)=TME(J)/NWS
 100  TMP(J)=TME(J)
C
      JM=JJ-1
      DO 120 J=1,JM
      T=TMP(J)
      L=J
      JP=J+1
      DO 110 K=JP,JJ
      IF(TMP(K).LE.T) GO TO 110
      T=TMP(K)
      L=K
 110  CONTINUE
      TMP(L)=TMP(J)
 120  TMP(J)=T
C
      T=0.0
      DO 130 J=3,JJ
 130  T=T+TMP(J)
      T=T*8.0/FLOAT(JJ-2)
      WRITE(3,240) (TME(J),J=1,JJ)
      WRITE(6,240) (TME(J),J=1,JJ)
      IF(TMP(1).LT.T) RETURN
C
      SPIKE=1
      WRITE(3,250)
 250  FORMAT(' TRIG DECIDED TO BE SPIKE'/)
 240  FORMAT(' MEAN POWER ',7F8.1)
      RETURN
      END
C
C
      SUBROUTINE MXBEAM(/Y1/,/Y2/,/IAL/,/JJ/,
     */IRL/,/I/,/INC/,/IBL/,/IBT1/,/IBT2/,*,*,*)
      REAL Y1(IAL,JJ),Y2(IAL,JJ)
      COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
      INTEGER BMTAB,TIMTAB(5)
      LOGICAL TRIG,GROUP
C
C THIS ROUTINE DETERMINES THE DIRECTION OF THE POSSIBLE EVENT
C
      IR2=IRL/2
      ISBL=IBL
      ISSS=I
      IBTL=40
      IF(IBL.EQ.1) GO TO 70
C
 10   DO 20 J=1,JJ
 20   TME(J)=TME(J)+Y1(I,J)
      DO 40 K=1,NBM
      BM=0.0
      DO 35 J=1,JJ
      II=I+BMTAB(K,J)
      IF(II.GT.IRL) GO TO 30
      BM=BM+Y1(II,J)
      GO TO 35
 30   BM=BM+Y2(II-IRL,J)
 35   CONTINUE
      EST(IPC,K)=EST(IPC,K)+BM
      IF(BM.GT.SNR(K)) SNR(K)=BM
      IF(ITRT(K).NE.0) GO TO 40
      IF(BM.LT.THR) GO TO 40
      ITI=(I-1)*SMI+0.5
      ITRT(K)=IBT1+ITI
      NTRIG=NTRIG+1
 40   CONTINUE
      IESC=IESC+1
      IF(MOD(IESC,IEPL).EQ.0) IPC=IPC+1-(IPC/4)*4
      IF(IESC.LT.INS) GO TO 60
      INS=INS+IEPL
      K2M=1
      DO 50 K=1,NBM
      E2(K)=0.0
      DO 45 J=1,4
 45   E2(K)=E2(K)+EST(J,K)
      EST(IPC,K)=0.0
 50   IF(E2(K).GT.E2(K2M)) K2M=K
      IF(E2(K2M).LE.E1(K1M)) GO TO 58
      K1M=K2M
      IWST=IESC
      DO 55 K=1,NBM
 55   E1(K)=E2(K)
 58   IF(IESC.EQ.NWS) GO TO 200
 60   I=I+INC
      IF(I.LE.IRL) GO TO 10
      I=I-IRL
      IBL=1
C
 70   DO 75 J=1,JJ
 75   TME(J)=TME(J)+Y2(I,J)
      DO 90 K=1,NBM
      BM=0.0
      DO 85 J=1,JJ
      II=I+BMTAB(K,J)
      IF(II.LT.1) GO TO 80
      BM=BM+Y2(II,J)
      GO TO 85
 80   BM=BM+Y1(II+IRL,J)
 85   CONTINUE
      EST(IPC,K)=EST(IPC,K)+BM
      IF(BM.GT.SNR(K)) SNR(K)=BM
      IF(ITRT(K).NE.0) GO TO 90
      IF(BM.LT.THR) GO TO 90
      ITI=(I-1)*SMI+0.5
      ITRT(K)=IBT2+ITI
      NTRIG=NTRIG+1
 90   CONTINUE
      IESC=IESC+1
      IF(MOD(IESC,IEPL).EQ. ) IPC=IPC+1-(IPC/4)*4
      IF(IESC.LT.INS) GO TO 150
      INS=INS+IEPL
      K2M=1
      DO 100 K=1,NBM
      E2(K)=0.0
      DO 95 J=1,4
 95   E2(K)=E2(K)+EST(J,K)
      EST(IPC,K)=0.0
 100  IF(E2(K).GT.E2(K2M)) K2M=K
      IF(E2(K2M).LE.E1(K1M)) GO TO 120
      K1M=K2M
      IWST=IESC
      DO 105 K=1,NBM
 105  E1(K)=E2(K)
 120  IF(IESC.EQ.NWS) GO TO 200
 150  I=I+INC
      IF(I.LE.IR2) GO TO 70
      RETURN 3
```

```
 200  CALL SPIKET(TME,NWS,JJ,SPIKE)
      WRITE(12,300) KIM,IBST,IWSS,NWS,SPIKE
      IF(SPIKE.EQ.0) GO TO 210
      IBL=ISBL
      I=ISSS
      LTT=LTT+IBTL
      GO TO 260
 210  WRITE(3,310)
      WRITE(6,310)
 220  DO 250 L=1,5
      LL=1
      DO 240 K=2,NBM
 240  IF(E1(K).GT.E1(LL)) LL=K
      S=10.0*ALOG10(SNR(LL)/ENL)
      CALL CHANGE(ITRT(LL),TIMTAB)
      WRITE(3,320) L,LL,E1(LL),S,TIMTAB,(ATAB(K,LL),K=1,8)
      WRITE(6,320) L,LL,E1(LL),S,TIMTAB,(ATAB(K,LL),K=1,8)
 250  E1(LL)=0.0
      LL=1
      DO 255 K=2,NBM
 255  IF(SNR(K).GT.SNR(LL)) LL=K
      S=10.0*ALOG10(SNR(LL)/ENL)
      ITI=(IWST-NES-IEPL)*SMI
      IWST=LTT+ITI
      CALL CHANGE(IWST,TIMTAB)
      WRITE(3,330) NWS,NES,TIMTAB,NTRIG,S,LL
      WRITE(6,330) NWS,NES,TIMTAB,NTRIG,S,LL
      ITI=(NWS-IEPL)*SMI+0.5
      LTT=LTT+ITI
 260  GROUP=.FALSE.
      IF(IBL.EQ.0) RETURN 1
      RETURN 2
C
 300  FORMAT(10I10)
 310  FORMAT(1X/' THE 10 BEAMS WITH MAX ENERGY'/
     *' NO,BEAM,ENERGY,MAX SNR(DB),TRIG TIME,AREA')
 320  FORMAT(1X,I2,I4,F11.1,F6.1,I4,4I3,1X,8A4)
 330  FORMAT(' WINDOW AND ENERGY LENGTH,WINDOW START,TRIGS,MAX SNR,BEAM'
     */1X,8I5,F6.1,I5/)
      END
C
C
      SUBROUTINE SENV(/X1/,/Y1/,/X2/,/Y2/,/WT/,/VES/,
     */NN/,/JJ/,/IRL/,/NDEL/)
      REAL X1(NN,JJ),Y1(NN,JJ),X2(NN,JJ),Y2(NN,JJ),WT(JJ),VES(JJ)
      REAL ZZ(7)
C
C THIS ROUTINE CALCULATES THE NORMALIZED SQUARED ENVELOPE
C
      IU1=IRL-NDEL
      IU2=IU1+1
      KDEL=NDEL-IRL
C
      DO 100 J=1,JJ
      VES(J)=0.0
      DO 40 I=IU2,IRL
      S=X1(I,J)**2
      VES(J)=VES(J)+S
 40   Y1(I,J)=(S+Y2(I+KDEL,J)**2)*WT(J)
C
      DO 60 I=1,IU1
      S=X2(I,J)**2
      VES(J)=VES(J)+S
 60   Y2(I,J)=(S+Y2(I+NDEL,J)**2)*WT(J)
 100  ZZ(J)=1.0/WT(J)
C
      WRITE(3,300) (ZZ(J),J=1,JJ)
      WRITE(6,300) (ZZ(J),J=1,JJ)
 300  FORMAT(' RUN EST',14F8.1)
      RETURN
      END
C
C
      SUBROUTINE PRESET(/A/,/N/,/M/,/N1/,/N2/,/M1/,/M2/,/V/)
      REAL A(N,M)
C
C THIS ROUTINE SETS SPECIFIED ELEMENTS IN A EQUAL TO V
C
      DO 10 J=M1,M2
      DO 10 I=N1,N2
 10   A(I,J)=V
      RETURN
      END
C
C
      SUBROUTINE FFT842(/IN/,/N2POW/,/X/,/Y/)
C
C THIS PROGRAM REPLACES THE VECTOR Z=X+IY BY ITS FINITE
C DISCRETE,COMPLEX FOURIER TRANSFORM IF N=0. THE INVERSE
C TRANSFORM IS CALCULATED FOR IN=1. IT PERFORMS AS MANY
C BASE 8 ITERATIONS AS POSSIBLE AND THEN FINISHES WITH A
C BASE 4 OR A BASE 2 ITERATION IF NEEDED. N MUST BE A
C POWER OF 2.
C
      DIMENSION X(2),Y(2),L(15)
      EQUIVALENCE (L15,L(1)),(L14,L(2)),(L13,L(3)),(L12,L(4)),
     *(L11,L(5)),(L10,L(6)),(L9,L(7)),(L8,L(8)),(L7,L(9)),
     *(L6,L(10)),(L5,L(11)),(L4,L(12)),(L3,L(13)),(L2,L(14)),
     *(L1,L(15))
C
      NTHPO=2**N2POW
      IF(IN.EQ.1) GO TO 40
      DO 30 I=1,NTHPO
      Y(I)=-Y(I)
 30   CONTINUE
 40   N8POW=N2POW/3
      IF(N8POW.EQ.0) GO TO 60
C
C RADIX 8 PASSES,IF ANY
      DO 50 IPASS=1,N8POW
      NXTLT=2**(N2POW-3*IPASS)
      LENGT=8*NXTLT
      CALL R8TX(NXTLT,NTHPO,LENGT,X(1),X(NXTLT+1),X(2*NXTLT+1),
     *X(3*NXTLT+1),X(4*NXTLT+1),X(5*NXTLT+1),X(6*NXTLT+1),
     *X(7*NXTLT+1),Y(1),Y(NXTLT+1),Y(2*NXTLT+1),Y(3*NXTLT+1),
     *Y(4*NXTLT+1),Y(5*NXTLT+1),Y(6*NXTLT+1),Y(7*NXTLT+1))
 50   CONTINUE
C
C IS THERE A FOUR FACTOR LEFT
 60   IF(N2POW-3*N8POW-1) 90,70,80
C
C GO THROUGH THE BASE 2 ITERATION
 70   CALL R2TX(NTHPO,X(1),X(2),Y(1),Y(2))
      GO TO 90
C
C GO THROUGH THE BASE 4 ITERATION
 80   CALL R4TX(NTHPO,X(1),X(2),X(3),X(4),Y(1),Y(2),Y(3),Y(4))
C
 90   DO 110 J=1,15
      L(J)=1
      IF(J-N2POW) 100,100,110
 100  L(J)=2**(N2POW+1-J)
 110  CONTINUE
      IJ=1
      DO 130 J1=1,L1
      DO 130 J2=J1,L2,L1
      DO 130 J3=J2,L3,L2
      DO 130 J4=J3,L4,L3
      DO 130 J5=J4,L5,L4
      DO 130 J6=J5,L6,L5
      DO 130 J7=J6,L7,L6
      DO 130 J8=J7,L8,L7
      DO 130 J9=J8,L9,L8
      DO 130 J10=J9,L10,L9
      DO 130 J11=J10,L11,L10
      DO 130 J12=J11,L12,L11
      DO 130 J13=J12,L13,L12
      DO 130 J14=J13,L14,L13
      DO 130 JI=J14,L15,L14
      IF(IJ-JI) 120,130,130
 120  R=X(IJ)
      X(IJ)=X(JI)
      X(JI)=R
      FI=Y(IJ)
      Y(IJ)=Y(JI)
      Y(JI)=FI
 130  IJ=IJ+1
      IF(IN.EQ.1) RETURN
      DO 140 I=1,NTHPO
      Y(I)=-Y(I)
 140  CONTINUE
      RETURN
      END

      SUBROUTINE R2TX(/NTHPO/,/CR0/,/CR1/,/CI0/,/CI1/)

 RADIX 2 ITERATION ROUTINE

      DIMENSION CR0(2),CR1(2),CI0(2),CI1(2)
      DO 10 K=1,NTHPO,2
      R1=CR0(K)+CR1(K)
      CR1(K)=CR0(K)-CR1(K)
      CR0(K)=R1
      FI1=CI0(K)+CI1(K)
      CI1(K)=CI0(K)-CI1(K)
      CI0(K)=FI1
 10   CONTINUE
      RETURN
      END

      SUBROUTINE R4TX(/NTHPO/,
     */CR0/,/CR1/,/CR2/,/CR3/,/CI0/,/CI1/,/CI2/,/CI3/)

 RADIX 4 ITERATION SUBROUTINE

      DIMENSION CR0(2),CR1(2),CR2(2),CR3(2),CI0(2),CI1(2),CI2(2),
     *CI3(2)
      DO 10 K=1,NTHPO,4
      R1=CR0(K)+CR2(K)
      R2=CR0(K)-CR2(K)
      R3=CR1(K)+CR3(K)
      R4=CR1(K)-CR3(K)
      FI1=CI0(K)+CI2(K)
      FI2=CI0(K)-CI2(K)
      FI3=CI1(K)+CI3(K)
      FI4=CI1(K)-CI3(K)
      CR0(K)=R1+R3
      CI0(K)=FI1+FI3
      CR1(K)=R1-R3
      CI1(K)=FI1-FI3
      CR2(K)=R2-FI4
      CI2(K)=FI2+R4
      CR3(K)=R2+FI4
      CI3(K)=FI2-R4
 10   CONTINUE
      RETURN
      END

      SUBROUTINE R8TX(/NXTLT/,/NTHPO/,/LENGT/,/CR0/,/CR1/,/CR2/,/CR3/,
     */CR4/,/CR5/,/CR6/,/CR7/,/CI0/,/CI1/,/CI2/,/CI3/,/CI4/,/CI5/,
     */CI6/,/CI7/)

 RADIX 8 ITERATION SUBROUTINE

      DIMENSION CR0(2),CR1(2),CR2(2),CR3(2),CR4(2),CR5(2),CR6(2),
     *CR7(2),CI0(2),CI1(2),CI2(2),CI3(2),CI4(2),CI5(2),CI6(2),
     *CI7(2)
      PI2=6.2831853072
      P7=0.7071067812

      SCALE=PI2/FLOAT(LENGT)
      DO 30 J=1,NXTLT
      ARG=FLOAT(J-1)*SCALE
      C1=COS(ARG)
      S1=SIN(ARG)
      C2=C1**2-S1**2
      S2=C1*S1+C1*S1
      C3=C1*C2-S1*S2
      S3=C2*S1+S2*C1
      C4=C2**2-S2**2
      S4=C2*S2+C2*S2
      C5=C2*C3-S2*S3
      S5=C3*S2+S3*C2
      C6=C3**2-S3**2
      S6=C3*S3+C3*S3
      C7=C3*C4-S3*S4
      S7=C4*S3+S4*C3
      DO 20 K=J,NTHPO,LENGT
      AR0=CR0(K)+CR4(K)
      AR1=CR1(K)+CR5(K)
      AR2=CR2(K)+CR6(K)
      AR3=CR3(K)+CR7(K)
      AR4=CR0(K)-CR4(K)
      AR5=CR1(K)-CR5(K)
      AR6=CR2(K)-CR6(K)
      AR7=CR3(K)-CR7(K)
      AI0=CI0(K)+CI4(K)
      AI1=CI1(K)+CI5(K)
      AI2=CI2(K)+CI6(K)
      AI3=CI3(K)+CI7(K)
      AI4=CI0(K)-CI4(K)
      AI5=CI1(K)-CI5(K)
      AI6=CI2(K)-CI6(K)
      AI7=CI3(K)-CI7(K)
      BR0=AR0+AR2
      BR1=AR1+AR3
      BR2=AR0-AR2
      BR3=AR1-AR3
      BR4=AR4-AI6
      BR5=AR5-AI7
      BR6=AR4+AI6
      BR7=AR5+AI7
      BI0=AI0+AI2
      BI1=AI1+AI3
      BI2=AI0-AI2
      BI3=AI1-AI3
      BI4=AI4+AR6
      BI5=AI5+AR7
      BI6=AI4-AR6
      BI7=AI5-AR7
      CR0(K)=BR0+BR1
      CI0(K)=BI0+BI1
      IF(J.LE.1) GO TO 10
```

```
      IF(J.LE.1) GO TO 10
      CR1(K)=C4*(BR0-BR1)-S4*(BI0-BI1)
      CI1(K)=C4*(BI0-BI1)+S4*(BR0-BR1)
      CR2(K)=C2*(BR2-BI3)-S2*(BI2+BR3)
      CI2(K)=C2*(BI2+BR3)+S2*(BR2-BI3)
      CR3(K)=C6*(BR2+BI3)-S6*(BI2-BR3)
      CI3(K)=C6*(BI2-BR3)+S6*(BR2+BI3)
      TR=P7*(BR5-BI5)
      TI=P7*(BR5+BI5)
      CR4(K)=C1*(BR4+TR)-S1*(BI4+TI)
      CI4(K)=C1*(BI4+TI)+S1*(BR4+TR)
      CR5(K)=C5*(BR4-TR)-S5*(BI4-TI)
      CI5(K)=C5*(BI4-TI)+S5*(BR4-TR)
      TR=-P7*(BR7+BI7)
      TI=P7*(BR7-BI7)
      CR6(K)=C3*(BR6+TR)-S3*(BI6+TI)
      CI6(K)=C3*(BI6+TI)+S3*(BR6+TR)
      CR7(K)=C7*(BR6-TR)-S7*(BI6-TI)
      CI7(K)=C7*(BI6-TI)+S7*(BR6-TR)
      GO TO 20
10    CR1(K)=BR0-BR1
      CI1(K)=BI0-BI1
      CR2(K)=BR2-BI3
      CI2(K)=BI2+BR3
      CR3(K)=BR2+BI3
      CI3(K)=BI2-BR3
      TR=P7*(BR5-BI5)
      TI=P7*(BR5+BI5)
      CR4(K)=BR4+TR
      CI4(K)=BI4+TI
      CR5(K)=BR4-TR
      CI5(K)=BI4-TI
      TR=-P7*(BR7+BI7)
      TI=P7*(BR7-BI7)
      CR6(K)=BR6+TR
      CI6(K)=BI6+TI
      CR7(K)=BR6-TR
      CI7(K)=BI6-TI
20    CONTINUE
30    CONTINUE
      RETURN
      END


      SUBROUTINE OUTP(/NO/,/IS/,/IN/,/X/,/Y/,
     */NN/,/MM/,/N1/,/N2/,/M1/,/M2/)
      REAL X(NN,MM),Y(NN,MM)

C THIS ROUTINE PERFORMS DISK OR PRINTER OUTPUT
C IS=0 UNFORMATTED OUTPUT
C IS=1 FORMATTED OUTPUT
C IN=1 ONLY X OUTPUT
C IN=2 BOTH X AND Y OUTPUT

      I=N2-N1+1
      WRITE(3,*) I
      IF(IS.NE.0) GO TO 40
      IF(IN.EQ.2) GO TO 20
      DO 10 I=N1,N2
10    WRITE(NO) (X(I,J),J=M1,M2)
      RETURN
20    DO 30 I=N1,N2
30    WRITE(NO) (X(I,J),Y(I,J),J=M1,M2)
      RETURN

40    IF(IN.EQ.2) GO TO 60
      DO 50 I=N1,N2
50    WRITE(NO,100) I,(X(I,J),J=M1,M2)
      RETURN
60    DO 70 I=N1,N2
70    WRITE(NO,100) I,(X(I,J),Y(I,J),J=M1,M2)
      RETURN
100   FORMAT(1X,I4,7F8.2)
      END


      SUBROUTINE CHANGE(/IDSEC/,/NTIME/)
      DIMENSION NTIME(1)

C CONVERTS TIME FROM DECISECONDS

      NTIME(1)=IDSEC/864000
      NTIME(2)=(IDSEC-864000*NTIME(1))/36000
      NTIME(3)=(IDSEC-864000*NTIME(1)-36000*NTIME(2))/600
      NTIME(4)=(IDSEC-864000*NTIME(1)-36000*NTIME(2)-600*NTIME(3))/10
      NTIME(5)= IDSEC-864000*NTIME(1)-36000*NTIME(2)-600*NTIME(3)
     * -10*NTIME(4)
      RETURN
      END


      SUBROUTINE SETPAR(/KTAPE/,/TAPCOD/,/INC/,/JJ/)

      COMMON /TYPE/ NAME(132),CHAN(132),COMP(132),IRATE,OPTION,NCH,
     *EXPECT,FOUND,IND,RAD,ETIME,TAPEOP,NTCH,STIME,FLAG,LIMDTC,
     *NEWLAS,NSAMP,NCHAN,FRSTIM,UX,UY,TAU,DELAY,BEAM,IFILT,JSAMP,NTAPE
      REAL*8 NAME
      INTEGER CHAN,COMP,OPTION,EXPECT,FOUND,ETIME,TAPEOP,STIME,FRSTIM,
     *TAPCOD,IT(6),HLP(7)
      INTEGER*2 TAU(132)
      LOGICAL FLAG,NEWLAS,DELAY,BEAM

      COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
      INTEGER BMTAB
      LOGICAL TRIG,GROUP

      ITRC=0
      NES=40
      IEPL=10
      INC=4
      IRATE=20
      LIMDTC=5000
      OPTION=5
      NEWLAS=.TRUE.
      NTAPE=0
      TAPEOP=1
      FRSTIM=0
      FLAG=.FALSE.
      TAPCOD=1

      WRITE(6,400)
      READ(5,*) JJ
      NCH=JJ
      WRITE(6,500)
      READ(5,*) (CHAN(J),J=1,NCH)
      DO 10 J=1,NCH
10    COMP(J)=CHAN(J)
      WRITE(3,300) (CHAN(J),J=1,NCH)
      WRITE(6,600)
      READ(5,*) DBTHR
      WRITE(6,200)
      READ(5,*) KTAPE,IT
      WRITE(3,100) IT
      IYR=IT(1)
      STIME=IT(2)*864000+IT(3)*36000+IT(4)*600+IT(5)*10+IT(6)
      RETURN
100   FORMAT(' START TIME IS',2I5,'/',I2,'.',I2,'.',I2,'.',I1)
200   FORMAT(' GIVE TAPE NUMBER AND START(YEAR,DAY,TIME,MIN,SEC,DES)')
300   FORMAT(1X/' CHANNELS WANTED ',7I5)
400   FORMAT(' GIVE NUMBER OF CHANNELS TO BE PROCESSED')
500   FORMAT(' GIVE CHANNEL IDENTIFICATION NUMBERS')
600   FORMAT(' GIVE DETECTION THRESHOLD IN DB')
      END


      SUBROUTINE BMINP(PRINT)
      COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
      INTEGER BMTAB,MM(7)
      LOGICAL TRIG,GROUP,PRINT

      COMMON /TYPE/ NAME(132),CHAN(132),COMP(132),IRATE,OPTION,NCH,
     *EXPECT,FOUND,IND,RAD,ETIME,TAPEOP,NTCH,STIME,FLAG,LIMDTC,
     *NEWLAS,NSAMP,NCHAN,FRSTIM,UX,UY,TAU,DELAY,BEAM,IFILT,JSAMP,NTAPE
      REAL*8 NAME
      INTEGER CHAN,COMP,OPTION,EXPECT,FOUND,ETIME,TAPEOP,STIME,FRSTIM
      INTEGER*2 TAU(132)
      LOGICAL FLAG,NEWLAS,DELAY,BEAM

C THIS ROUTINE READS THE BEAM DELAY TABLE

      READ(10,200) NBM
      IF(PRINT) WRITE(3,100)
      DO 5 I=1,NBM
      READ (10,200) MM,(ATAB(J,I),J=1,8)
      DO 4 J=1,NCH
4     BMTAB(I,J)=MM((COMP(J)+5)/6)
      IF(PRINT) WRITE(3,300) I,(ATAB(J,I),J=1,8),(BMTAB(I,J),J=1,NCH)
5     CONTINUE
      REWIND 10
100   FORMAT(1X/' BEAM DELAY DATA TABLE')
200   FORMAT(7I5,1X,8A4)
300   FORMAT(1X,I5,2X,8A4,1X,7I5)
      RETURN
      END

      SUBROUTINE WTCALC(/IRL/,/JJ/)

C THIS ROUTINE CALCULATES THE WEIGHTING FACTORS FOR EACH CHANNEL

      COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
     *BMTAB(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
     *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
     *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)

      R4=4*IRL
      DO 10 J=1,JJ
      WT(J)=R4/(VES2(J)+VES3(J)+VES4(J)+VES5(J))
      VES5(J)=VES4(J)
      VES4(J)=VES3(J)
      VES3(J)=VES2(J)
10    VES2(J)=VES1(J)
      RETURN
      END


      SUBROUTINE SPOLES(/X1/,/IAL/,/JJ/,/KTAPE/,/IYR/,/TAPCOD/,*)
      REAL X1(IAL,JJ)
      INTEGER TAPCOD
      LOGICAL INERR

C ROUTINE FOR READING NORSAR SHORT PERIODIC DATA
C AFTER 1. OCT. 1976 (ASGEIR NYSAETER MAI 1981)

      COMMON/WORKER/ INPUT,BLOCK
      INTEGER*2 INPUT(2744),BLOCK

      COMMON /TYPE/ NAME(132),CHAN(132),COMP(132),IRATE,OPTION,NCH,
     *EXPECT,FOUND,IND,RAD,ETIME,TAPEOP,NTCH,STIME,FLAG,LIMDTC,
     *NEWLAS,NSAMP,NCHAN,FRSTIM,UX,UY,TAU,DELAY,BEAM,IFILT,JSAMP,NTAPE

      REAL*8 NAME
      LOGICAL*1 NAMEL1(1056)
      EQUIVALENCE (NAME,NAMEL1)
      INTEGER CHAN,COMP,OPTION,EXPECT,FOUND,ETIME,TAPEOP,STIME,FRSTIM
      INTEGER*2 TAU(132)
      LOGICAL FLAG,NEWLAS,DELAY,BEAM

      COMMON/SAMPLE/SAMP(132)
      INTEGER*4 SAMP

      COMMON/STATUS/ SENS
      LOGICAL*1 SENS(138)

      COMMON /CMSTAP/ TAPX,TAPY,TAPZ
      INTEGER*4 TAPX,TAPY,TAPZ,CTAPE1/'TAP1'/,CTAPE2/'TAP2'/,
     *CTAPE3/'TAP3'/
      INTEGER TIME(5)
      LOGICAL*4 CONTIN

      INERR=.FALSE.
      CONTIN = KTAPE.EQ.NTAPE
      IND    = 0
      IDT     = 10.0/FLOAT(IRATE)
      IF(IDT.LE.0) IDT=1

  400 IF(CONTIN) GO TO 500
      NTAPE=KTAPE
      TAPX=CTAPE1
      TAPY=CTAPE2
      TAPZ=CTAPE3

405   CALL S7OPEN( ITAPE, IDOY, IYEAR )
      WRITE(3,3) ITAPE, IDOY, IYEAR
      WRITE(6,3) ITAPE, IDOY, IYEAR

      IF( IND .NE.-10.AND.IND.NE.-9 ) GO TO 420
415   WRITE(3,4) IND
      WRITE(6,4) IND
      NSAMP=0
      INERR=.TRUE.
      GO TO 1350

420   EXPECT = 367*864000
      CALL S7READ(&415)
      CALL BACKSP(TAPX,IERR)
      BLOCK=0
425   CALL CHANGE( FOUND,TIME )
      FRSTIM = FOUND
      WRITE(3,5) TIME
      WRITE(6,5) TIME

430   EXPECT = STIME
      CALL S7FIND( &415 )
      CALL CHANGE( FOUND,TIME )
      WRITE(6,6) TIME
      EXPECT = FOUND

500   STIME  = EXPECT
      ISAMP=0
      JSAMP=1
1000 CALL S7READ(&1100)
      IF(IND) 1100,1200,1200
1100 INERR=.TRUE.
      NSAMP=ISAMP
      WRITE(3,14) EXPECT,FOUND,NSAMP
      WRITE(6,14) EXPECT,FOUND,NSAMP
      GO TO 1300

1200 ISAMP=ISAMP+1
      DO 1210 J=1,NCH
1210 X1(ISAMP,J)=SAMP(COMP(J))
      EXPECT=EXPECT+IDT
      IF(ISAMP.LT.NSAMP) GO TO 1000
```

```
1300 ETIME=FOUND
     CALL CHANGE( ETIME,TIME )
     WRITE(6,9) ETIME,TIME,NSAMP
     IF(.NOT.INERR) RETURN
1350 WRITE(3,15)
     WRITE(6,15)
     RETURN 1

3    FORMAT(' TAPE',I7,' DAY',I4,' YEAR',I5,' WAS MOUNTED')
4    FORMAT(1X,'**** MOUNTED TAPE IN ERRROR ****IND=',I5)
5    FORMAT(' FIRST DATA TIME IS',I5,'/',I2,'.',I2,'.',I2,'.',I1)
6    FORMAT(' FOUND START TIME IS',I5,'/',I2,'.',I2,'.',I2,'.',I1)
7    FORMAT(' READ END',I10,'/',5I4,'/ NSAMP ',I4)
14   FORMAT('0** EXPECTED',I10,'  FOUND',I10,' NSAMP SET TO',I8)
15   FORMAT(1X/' SPOLES INERR=.TRUE.')
     END


     SUBROUTINE ACHECK(/X1/,/X2/,/Y1/,/Y2/,/CHAN/,/COMP/,
    */SENS/,/LSENS/,/IAL/,/JJ/,/NCH/,*,*)
     REAL X1(IAL,JJ),X2(IAL,JJ),Y1(IAL,JJ),Y2(IAL,JJ)
     INTEGER CHAN(1),COMP(1)
     LOGICAL*1 SENS(138),LSENS(48)
     LOGICAL INERR,NEWCNF

     COMMON/DTP/ITRT(204),E1(204),E2(204),EST(4,204),SNR(204),K1M,K2M,
    *BMTAR(204,7),ATAB(8,204),IBST,IWSS,SPIKE,INS,NWS,NES,LTT,IPC,
    *BM,NOB,TRIG,GROUP,THR,DBTHR,IEPL,NBM,IESC,ITRC,ENL,SMI,IYR,NTSMP,
    *NTRIG,WT(7),VES1(7),VES2(7),VES3(7),VES4(7),VES5(7),TME(7)
     INTEGER BMTAB
     LOGICAL TRIG,GROUP

 THIS ROUTINE CHECKS THE ARRAY STATUS BLOCK SENS

     J=0
     INERR=.FALSE.
     NEWCNF=.FALSE.
     DO 260 LL=1,JJ
     J1=CHAN(LL)
     IF(SENS(J1).OR.LSENS(J1)) GO TO 220
     J=J+1
     LSENS(J1)=.FALSE.
     COMP(J)=J1
     GO TO 260
220  IF(SENS(J1).AND.LSENS(J1)) GO TO 260
     IF(.NOT.SENS(J1).AND.LSENS(J1)) GO TO 230
     WRITE(6,11) J1
     WRITE(3,11) J1
     IF(NCH.LT.3) INERR=.TRUE.
     NCH=NCH-1
     DO 226 J2=LL,NCH
     WT(J2)=WT(J2+1)
     VES1(J2)=VES1(J2+1)
     VES2(J2)=VES2(J2+1)
     VES3(J2)=VES3(J2+1)
     VES4(J2)=VES4(J2+1)
     VES5(J2)=VES5(J2+1)
     DO 222 J3=1,IAL
     X1(J3,J2)=X1(J3,J2+1)
     X2(J3,J2)=X2(J3,J2+1)
     Y1(J3,J2)=Y1(J3,J2+1)
222  Y2(J3,J2)=Y2(J3,J2+1)
     DO 226 J3=1,NBM
226  BMTAB(J3,J2)=BMTAB(J3,J2+1)
     ENL= NCH*2.0
     EVAR=NCH*4.0
     THR=(10.0**(DBTHR/10.0))*ENL
     WRITE(6,16) ENL,EVAR,THR,DBTHR
     WRITE(3,16) ENL,EVAR,THR,DBTHR
     LSENS(J1)=.TRUE.
     GO TO 260
230  WRITE(6,12) J1
     WRITE(3,12) J1
     NCH=NCH+1
     NEWCNF=.TRUE.
     J=J+1
     LSENS(J1)=.FALSE.
     COMP(J)=J1
260  CONTINUE
     IF(J.EQ.NCH) GO TO 280
     WRITE(6,13) J,NCH
     WRITE(3,13) J,NCH
     INERR=.TRUE.
280  IF(.NOT.INERR) GO TO 290
     WRITE(3,15)
     WRITE(6,15)
     RETURN 1
290  IF(.NOT.NEWCNF) RETURN
     WRITE(3,14)
     WRITE(6,14)
     RETURN 2

11   FORMAT(1X/' ACHECK CHANNEL',I5,' MASKED OUT')
12   FORMAT(1X/' ACHECK CHANNEL',I5,' AVAILABLE')
13   FORMAT(1X/' ACHECK ERROR IN INDEX COUNTING,J,NCH=',2I5)
14   FORMAT(1X/' ACHECK NEWCNF=.TRUE.')
15   FORMAT(1X/' ACHECK INERR=.TRUE.')
16   FORMAT(1X/' EX(BM NOISE),VAR(BM NOISE),THR,DBTHR',4F9.2)
     END
```

```
C
C
      SUBROUTINE GENCOF(WWR,N)
      REAL WWR(1)
      REAL*8 TSR,TSI
C
C THIS ROUTINE GENERATES COEFFICIENTS TO FFT42 AND IFFT42
C
      NCOF=N/2
      TSR=6.28318530717958600/DFLOAT(N)
      TSI=0.0D0
      DO 10 I=1,NCOF
      TSI=TSI+TSR
      WWR(I)=SNGL(DCOS(TSI))
 10   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE IFFT42(X,Y,WWR,N)
      REAL X(1),Y(1),WWR(1)
C
C THIS ROUTINE PERFORMS THE INVERSE FOURIER TRANSFORM
C FOR N = 2**M. IT USES DECIMATICN IN TIME TECHNIQUE,
C AND IN PLACE COMPUTATIONS FOR A RADIX OF 4.
C
      I1=1
      N4=N/4
      K1=N4
      N2=N/2
      IC1=1
      IC2=4
      SQ2=0.7071067812
 10   I1=I1*4
      IF(I1.LT.N) GO TO 10
      IF(I1.EQ.N) GO TO 30
C
      IC1=2
      IC2=2
      DO 20 IO=1,N,IC2
      I1=IO+1
      A0=X(IO)
      B0=Y(IO)
      X(IO)=A0+X(I1)
      Y(IO)=B0+Y(I1)
      X(I1)=A0-X(I1)
 20   Y(I1)=B0-Y(I1)
      IC2=IC2*4
      K1=K1/2
C
 30   DO 40 IO=1,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      A0=X(IO)+X(I1)
      B0=Y(IO)+Y(I1)
      A1=X(I2)+X(I3)
      B1=Y(I2)+Y(I3)
      A2=X(IO)-X(I1)
      B2=Y(IO)-Y(I1)
      A3=X(I2)-X(I3)
      B3=Y(I2)-Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      X(I1)=A2-B3
      Y(I1)=B2+A3
      X(I2)=A0-A1
      Y(I2)=B0-B1
      X(I3)=A2+B3
 40   Y(I3)=B2-A3
      IF(IC1.EQ.1) GO TO 100
C
      I=2
      KR2=K1
      KR1=KR2+KR2
      IT=N4
      ITT=(IC1-1)*KR1+1
      IF(KR1.EQ.N4) GO TO 80
 50   KI1=IABS(N4-KR1)
      KI2=N4-KR2
      K33=KR1+KR2
      KI3=IABS(N4-K33)
      KR3=IABS(N2-K33)
      DO 60 IO=I,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      S1   =X(I1)*WWR(KR1)-Y(I1)*WWR(KI1)
      Y(I1)=Y(I1)*WWR(KR1)+X(I1)*WWR(KI1)
      S2   =X(I2)*WWR(KR2)-Y(I2)*WWR(KI2)
      Y(I2)=Y(I2)*WWR(KR2)+X(I2)*WWR(KI2)
      S3   =X(I3)*WWR(KR3)+Y(I3)*WWR(KI3)
      Y(I3)=Y(I3)*WWR(KR3)-X(I3)*WWR(KI3)
      A0=X(IO)+S1
      B0=Y(IO)+Y(I1)
      A1=S2-S3
      B1=Y(I2)-Y(I3)
      A2=X(IO)-S1
      B2=Y(IO)-Y(I1)
      A3=S2+S3
      B3=Y(I2)+Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      X(I1)=A2-B3
      Y(I1)=B2+A3
      X(I2)=A0-A1
      Y(I2)=B0-B1
      X(I3)=A2+B3
 60   Y(I3)=B2-A3
 70   I=I+1
      KR2=KR2+K1
      KR1=KR2+KR2
      IF((KR1-IT)) 50,80,100
C
 80   DO 90 IO=I,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      S2   =(X(I2)-Y(I2))*SQ2
      Y(I2)=(X(I2)+Y(I2))*SQ2
      S3   =(X(I3)+Y(I3))*SQ2
      Y(I3)=(X(I3)-Y(I3))*SQ2
      A0=X(IO)-Y(I1)
      B0=Y(IO)+X(I1)
      A1=S2-S3
      B1=Y(I2)+Y(I3)
      A2=X(IO)+Y(I1)
      B2=Y(IO)-X(I1)
      A3=S2+S3
      B3=Y(I2)-Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      X(I1)=A2-B3
      Y(I1)=B2+A3
      X(I2)=A0-A1
      Y(I2)=B0-B1
      X(I3)=A2+B3
 90   Y(I3)=B2-A3
      IT=ITT
      GO TO 70
C
 100  IC1=IC2
      IC2=IC2*4
      K1=K1/4
      IF(IC2.LE.N) GO TO 30
      RETURN
      END
C
```

```
C
C
      SUBROUTINE FFT42(X,Y,WWR,N)
      REAL X(1),Y(1),WWR(1)
C
C THIS ROUTINE PERFORMS THE DISCRETE FOURIER TRANSFORM
C FCR N = 2**M. IT USES DECIMATICN IN FREQUENCY TECHNIQUE,
C AND IN PLACE COMPUTATIONS FOR A RADIX OF 4.
C
      IC2=N
      N2=N/2
      N4=N/4
      IC1=N4
      SQ2=0.7071067812
      K1=1
C
 30   DO 40 IO=1,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      A0=X(IO)+X(I2)
      B0=Y(IO)+Y(I2)
      A1=X(I1)+X(I3)
      B1=Y(I1)+Y(I3)
      A2=X(IO)-X(I2)
      B2=Y(IO)-Y(I2)
      A3=X(I1)-X(I3)
      B3=Y(I1)-Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      X(I1)=A0-A1
      Y(I1)=B0-B1
      X(I2)=A2+B3
      Y(I2)=B2-A3
      X(I3)=A2-B3
 40   Y(I3)=B2+A3
      IF(IC1.EQ.1) RETURN
C
      I=2
      KR2=K1
      KR1=KR2+KR2
      IT=N4
      ITT=(IC1-1)*KR1+1
      IF(KR1.EQ.N4) GO TO 80
 50   KI1=IABS(N4-KR1)
      KI2=N4-KR2
      K33=KR1+KR2
      KI3=IABS(N4-K33)
      KR3=IABS(N2-K33)
      DO 60 IO=I,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      A0=X(IO)+X(I2)
      B0=Y(IO)+Y(I2)
      A1=X(I1)+X(I3)
      B1=Y(I1)+Y(I3)
      A2=X(IO)-X(I2)
      B2=Y(IO)-Y(I2)
      A3=X(I1)-X(I3)
      B3=Y(I1)-Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      S1   =A0-A1
      Y(I1)=B0-B1
      S2   =A2+B3
      Y(I2)=B2-A3
      S3   =B3-A2
      Y(I3)=B2+A3
      X(I1)=S1*WWR(KR1)+Y(I1)*WWR(KI1)
      Y(I1)=Y(I1)*WWR(KR1)-S1*WWR(KI1)
      X(I2)=S2*WWR(KR2)+Y(I2)*WWR(KI2)
      Y(I2)=Y(I2)*WWR(KR2)-S2*WWR(KI2)
      X(I3)=S3*WWR(KR3)+Y(I3)*WWR(KI3)
 60   Y(I3)=S3*WWR(KI3)-Y(I3)*WWR(KR3)
 70   I=I+1
      KR2=KR2+K1
      KR1=KR2+KR2
      IF((KR1-IT)) 50,80,100
C
 80   DO 90 IO=I,N,IC2
      I1=IO+IC1
      I2=I1+IC1
      I3=I2+IC1
      A0=X(IO)+X(I2)
      B0=Y(IO)+Y(I2)
      A1=X(I1)+X(I3)
      B1=Y(I1)+Y(I3)
      A2=X(IO)-X(I2)
      B2=Y(IO)-Y(I2)
      A3=X(I1)-X(I3)
      B3=Y(I1)-Y(I3)
      X(IO)=A0+A1
      Y(IO)=B0+B1
      X(I1)=B0-B1
      Y(I1)=A1-A0
      A0   =A2+B3
      B0   =B2-A3
      A1   =B3-A2
      B1   =B2+A3
      X(I2)=(A0+B0)*SQ2
      Y(I2)=(B0-A0)*SQ2
      X(I3)=(B1+A1)*SQ2
 90   Y(I3)=(A1-B1)*SQ2
      IT=ITT
      GO TO 70
C
 100  IC2=IC1
      IC1=IC1/4
      K1=K1*4
      IF(IC2.GT.2) GO TO 30
C
      DO 110 IO=1,N,IC2
      I1=IO+1
      A0=X(IO)
      B0=Y(IO)
      X(IO)=A0+X(I1)
      Y(IO)=B0+Y(I1)
      X(I1)=A0-X(I1)
 110  Y(I1)=B0-Y(I1)
      RETURN
      END
C
```