COMPUTER STUDIES OF MICROSEISM STATISTICS WITH APPLICATIONS TO PREDICTION AND DETECTION

by
JAMES NELSON GALBRAITH, Jr.
S.B., Massachusetts Institute of Technology
(1958)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF
PHILOSOPHY
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June, 1963

Signature of Author ______________________________
Department of Geology and Geophysics

Certified by ______________________________
Thesis Supervisor

Accepted by ______________________________
Chairman, Departmental Committee on Graduate Students

COMPUTER STUDIES OF MICROSEISM STATISTICS WITH APPLICATIONS TO PREDICTION AND DETECTION

by
James Nelson Galbraith, Jr.

Submitted to the Department of Geology and Geophysics on 17 May 1963 in partial fulfillment of the requirement for the Degree of Doctor of Philosophy

ABSTRACT

Computational experiments have been performed on seismic data digitized from the records obtained by the Air Force during the Logan and Blanca underground nuclear shots, by Dr. Bruce Bogert in New Jersey and by the Wichita Mountain Seismic Observatory.

The experiments indicate that microseismic noise of about .3 cps frequency is associated with the oceans but the higher frequencies are not. Attempts to identify definite wave types, such as Rayleigh and Love waves, and to follow wave packets from station to station failed, but the failure illustrated the complexity of the microseisms and points out the necessity of a statistical study.

For the statistical studies the microseisms were considered to be stochastic time series. It was found that the probability densities of the amplitudes were Gaussian and were not independent. Spectral analysis showed the typical microseism spectrum to have a maximum at about .3 cps and often other strong bands at 1.4 and 2 cps.

The microseism time series are approximately stationary and can be described as a moving average operation. Thus they can be generated by a convolution of a minimum phase wavelet with a white light series. The wavelet is found for typical data by factorization of the power spectrum and the white light series is obtained by convolution of the inverse minimum phase wavelet with the noise data. Tests on the white light series indicate that its probability density is approximately Gaussian and that it is approximately independent. Hence non-linear operators or filters are not particularly useful in microseism studies.

Cross correlation and cross spectra between different components of data at the same station, like components at different stations and array data have been computed. It was not possible to identify individual wave types or directions of travel with any degree of certainty.

Prediction studies of microseisms have been done to try to improve the signal to noise ratio during the first motion interval. The mean squared error technique and the spectrum factorization technique have been used. The spectrum factorization is found to be superior because long operators can be more readily obtained. However, one can predict at best about 50% of the energy which is not sufficient to produce a significant improvement in the signal to noise ratio. Indications are that other prediction techniques will not give much better results.

Artificial microseisms generated by convolution of a typical microseism wavelet with Gaussian white has been used in a computer simulation of a detection system. The system is an energy detector which detects events in microseismic noise. The system is studied in terms of false alarm rate and failure to detect rate. Overall system effectiveness is given in terms of false alarms per hour as function of signal to noise ratio for a 95% probability of detection success. The system characteristics are found to be essentially invariant when the inputs are band pass filtered. The simple band pass filter can in some cases give significant signal to noise ratio improvement.

Details of the statistical tests and computer programs are given along with an approximate solution to a non-linear water wave problem related to microseism generation. The solution, which uses DeVorkin's representation scheme, is for arbitrary initial conditions and shows that sum and difference frequencies of all the frequencies present initially will be generated.

Thesis Supervisor: Stephen M. Simpson, Jr.
Title: Associate Professor of Geophysics

ACKNOWLEDGEMENTS

I wish to express my sincere thanks to Professor S. M. Simpson, Jr., for his help, ideas and computer programs which were so necessary in the preparation of this thesis. Valuable aid and criticism were also freely given by Professor Theodore R. Madden, Dr. Enders A. Robinson and Dr. Donald DeVorkin.

I am also grateful to R. A. Wiggins, R. Greenfield, J. F. Claerbout and Mrs. Jacqueline Simpson for their assistance and the use of their computer programs.

I wish to thank Mrs. Elizabeth Studer, Mrs. Irene Hawkins, Joseph Procito and Karl Gentili for their assistance in performing innumerable tasks necessary for the completion of this thesis. I am grateful to Mrs. Myrna Kasser and my wife, Joan, for typing the preliminary copy of the thesis, and owe especial thanks to Mrs. Jane McNabb for the preparation of the final copy.

The data was digitized with the aid of Wolf Research and Development Corporation and Research Calculations. The computation was done in part at the M.I.T. Computation Center with the help and cooperation of Michael Saxton of the IBM Liasion Office and in part at the M.I.T. Cooperative Computing Laboratory with the valuable assistance of Anthony Sacco.

Acknowledgement is extended to Geoscience, Inc. for the use of computer programs for detection simulation studies. I wish to acknowledge the Advanced Research Projects Agency who sponsored this work as part of contract AF 19(604)7378.

TABLE OF CONTENTS

INTRODUCTION

Need to Study Noise

The disarmament talks at Geneva and the need for a surveillance network to detect and report the testing of nuclear devices, particularly underground testing, have put new emphasis on the field of Seismology. Government support in this area has made possible much research into the nature of seismic disturbances and instrumentation for detecting them. The present thesis was supported by the Advanced Research Projects Agency under the Vela Uniform Project contract AF 19(604)7378. The contract covers the digitization of the paper records from the Logan and Blanca shots of the 1958 Hardtack series, investigation of ways to improve the signal to noise ratio, particularly in the first motion interval, and investigation of the properties of bomb and earthquake signals.

Definition of Microseisms

Essential to the problem of signal detection and signal to noise ratio improvement is an understanding of the natures of both the signal and the noise. This thesis will deal mainly with the properties of the noise. A definition of what is meant by noise is necessary since in many cases what is noise to one man is signal to another. In the context of this thesis any ground motion not associated with definite bomb or earthquake signals, motion which is present at all times, will be considered noise and will be called microseisms or microseismic noise.

The study of microseisms dates back about 100 years to the pendulum measurements of an Italian monk, Bertelli (Haq, 1954). Only very qualitative conclusions which generalized the data could be made, but it was obvious from study of Bertelli and others that the surface of the earth was in a state of oscillation. This "sea" of elastic waves came under the scrutiny of other observers who were interested in the causes of the disturbances. Wiechert (1905) suggested that microseisms were generated by the impact of surf on a steep coast. Gutenburg (1912) noted a correlation of microseisms with 4 to 8 second periods with surf and wind direction. Ramirez (1940) studied the physical properties of microseismic waves, the velocity, direction of travel and particle motion, with a tripartate or triangular arrangement of three component instruments. He found that the properties of these waves were fairly consistent with those of Rayleigh and Love waves.

Sources of Microseisms

Observers noted that the microseisms and sea waves seemed to be connected, and, in some cases, the periods of the sea waves were twice the period of the microseisms. However, the idea that sea waves produced microseisms was hard to justify theoretically since pressure variations due to travelling water waves die out exponentually with depth and are nearly zero within a wave length. Miche (1944) showed that there is a pressure fluctuation under a standing wave which is unattenuated with depth (for incompressible fluids), and its frequency is twice that of the sea wave. Longuet-Higgins (1950) realized that this was what was needed to explain the observations. He also showed that the mechanism could account for the energy of the observed microseisms. The presence of an unattenuated double frequency variation is demonstrated by Longuet-

Higgins in a small parameter expansion approximation to the solution of the non-linear equations for the pressure variations at the bottom of a layer of water with a rigid lower boundary and a standing wave on the top. Another method of approximation for this type of problem using a representation scheme for the solution of non-linear equations worked out by DeVorkin (1963) is given in Appendix A. It illustrates that the sum and difference frequencies of all frequencies present initially are expected to develop.

The microseisms with periods from 4 to 12 seconds are generally attributed to ocean waves and recourse to the theory of Longuet-Higgins can be made for their explanation although there is still controversy on the matter. The data which has been used in this thesis was recorded with a Benioff short period instrument so that only the shortest period oceanic microseisms come through. Microseisms of higher frequency than the oceanic band are usually attributed to wind and meteorological factors or are thought to be cultural noise. Typical noise sources are swaying trees and buildings, storms, city traffic, heavy machinery, power plants, trains etc.

This brief allusion to the history of the study of microseisms does not give a feeling for the enormous amount of work which has been done in this area. (See Haq, 1954, for a fuller account and references.) A great deal of the work has been concerned with microseism generation mechanisms, surface wave propagation and particle motion, and studies of the direction of propagations and their relation to storms. Nearly all of these studies consider microseisms as a signal. This thesis for the most part considers microseisms as noise. The main object is to treat the microseisms from a statistical point of view and try to describe them so that something can be done about them rather than with them. To this goal, the tools of statistical analysis have been brought forward and applied with the aid of high speed digital computers.

We shall see that a few examples which treat the microseisms as signals will suffice to point out the need for a more general description of the noise. It is obvious that that time series analysis can be applied to the study of microseisms, but stronger and more useful statements can be made about the time series if it can be shown that they are stationary or, better still, ergodic. We must therefore test the microseisms to see if they fall into one or more of these special categories of time series. Spectral analysis, probability studies and independence tests are some of the techniques which aid in the classification of microseisms.

The proper mathematical description of microseisms can also be the key to the optimum prediction problem, and will permit the study of the predictability of microseisms. We shall see that prediction can be used in some cases to reduce the noise level and therefore, if a signal is also present, improve the signal to noise ratio. The amount of improvement is of course dependent on the predictability of the noise.

A good mathematical model of microseismic noise will also permit us to generate the noise artificially. This artificial noise is extremely useful when long sections of continuous noise are required, and is therefore necessary when we simulate by computer a system to detect events in microseismic noise.

Outline

The thesis is divided into four chapters. The first deals with the basic statistics of the data on which the present studies are based. It includes a description of the data and how it was recorded as well as amplitude studies, auto and cross spectra, empirical probability density functions, and a mathematical model for noise generation.

Chapter two discusses the prediction of the noise by different methods and then applies this to the problem of the determination of the direction of first motion of a signal in the noise. Improvement with non-linear predictors is also considered.

In chapter three an automatic system for the detection of signals in microseismic noise is proposed and the results of a computer simulation of this system are given in terms of detection probabilities and false alarm rates for filtered and unfiltered inputs.

Chapter four is a summary which restates the major conclusions.

Details of some analyses and the computer programs used are left for the Appendices.

1. BASIC STATISTICAL STUDIES

1.1 Empirical Data

Data Sources - Noise before and Noise after Events

The data which forms the basis for most of the computational studies described in this thesis are the seismic records of the Logan (5 KT) and Blanca (19KT) underground nuclear shots of the 1958 Hardtack series (Romney, 1959). These were recorded by the U. S. Air Force at 28 temporary stations set up across the United States as shown in Figure 1.1.1. The instruments used were short period Benioffs with galvanometer periods (T_g) of .20 seconds. Most stations were equipped with a vertical instrument (up-down) and two horizontals, a "toward-away" and a "right-left". These designations are with respect to an observer standing at the shot point looking at the station. The vertical and horizontal instrument responses are the same and are shown in Figures 1.1.2 and 1.1.3 (Geotechnical Corp., 1961). The paper records from these shots were provided by the Air Force and were digitized at 20 samples per second. In no case were the paper records for an entire drum revolution provided so that the greatest time interval of continuous record available was on the order of a few minutes. For this reason the noise records which have been digitized are labeled "Noise Before" and "Noise After" with the appropriate shot, distance from shot and component. Noise before refers to the trace on the paper record which is just above the signal trace, and is therefore one drum revolution time before the shot. Noise after is the trace just below the signal trace. A copy of one of the original paper records which was digitized is shown in Figure 1.1.4, and a plot of the corresponding digitized record is shown in Figures 1.1.5 to 1.1.7. Figures 1.1.5 to 1.1.7 have been plotted by computer program using the oscilloscope attached to the IBM 7090 computer at the M.I.T. Computation Center. These graphs, and many of the others appearing in later sections, have been plotted as histograms. In several cases, particularly the spectral computations, the values plotted are averages or estimates over some range so that there is no justification for interpolation and the histogram is the predered method of presentation.

Logan and Blanca Digitization Procedure

The records were broken up into sections and each section was digitized separately. This procedure can lead to some error since each section could have a linear trend. This was compensated for by removing the best fitting (in the least squares sense) segmented line from the entire record, where each segment is the length of a section.

The digitization accuracy is good to a few percent, and the gain values supplied with the original records are quite good, but the actual ground motion values may be off by as much as 15 percent.

Other digitized data has been provided by Dr. Bruce Bogert of the Bell Telephone Laboratories, who has a short period vertical Benioff at Cherry Hill Park, New Jersey, and by United Electro Dynamics, Inc., who have digitized the records from the WMSO station in Oklahoma. Dr. Bogert's Benioff has a response similar to that of the Hardtack instruments, but its low frequency cut off is somewhat higher (Bogert, 1961), Figure 1.1.8. The WMSO station is a linear array of vertical Benioffs with the same response as the Hardtack instruments.

A list of our record numbers appropos to this thesis and the event and station to which they correspond, is given in Table 1.1.1.

TABLE 1.1.1

RECORD NUMBER	DESCRIPTION	SAMPLES/SEC.
1000	NOISE BEFORE LOGAN 1902 KM., LEFT	20
1001	NOISE AFTER LOGAN 1902 KM., LEFT	20
1002	NOISE BEFORE LOGAN 1902 KM., UP	20
1003	NOISE AFTER LOGAN 1902 KM., UP	20
1004	NOISE BEFORE LOGAN 1902 KM., TOWARD	20
1005	NOISE AFTER LOGAN 1902 KM., TOWARD	20
1006	NOISE BEFORE LOGAN 2111 KM., LEFT	20
1007	NOISE AFTER LOGAN 2111 KM., LEFT	20
1008	NOISE BEFORE LOGAN 2111 KM., UP	20
1009	NOISE AFTER LOGAN 2111 KM., UP	20
1010	NOISE BEFORE LOGAN 2111 KM., TOWARD	20
1011	NOISE AFTER LOGAN 2111 KM., TOWARD	20
1026	NOISE BEFORE BLANCA 1610 KM., LEFT	20
1027	NOISE AFTER BLANCA 1610 KM., LEFT	20
1028	NOISE BEFORE BLANCA 1610 KM., UP	20
1029	NOISE AFTER BLANCA 1610 KM., UP	20
1030	NOISE BEFORE BLANCA 1610 KM., AWAY	20
1031	NOISE AFTER BLANCA 1610 KM., AWAY	20
204	CHERRY HILL PARK 4, NOISE	9.0909
233	CHERRY HILL PARK 31, NOISE	9.0909
301	WMSO L9, NOISE BEFORE CALIF. E.Q. JUNE 20, 1962	20
303	WMSO L7, NOISE BEFORE CALIF. E.Q. JUNE 20, 1962	20
305	WMSO L5, NOISE BEFORE CALIF. E.Q. JUNE 20, 1962	20
307	WMSO L3, NOISE BEFORE CALIF. E.Q. JUNE 20, 1962	20
309	WMSO L1, NOISE BEFORE CALIF. E.Q. JUNE 20, 1962	20

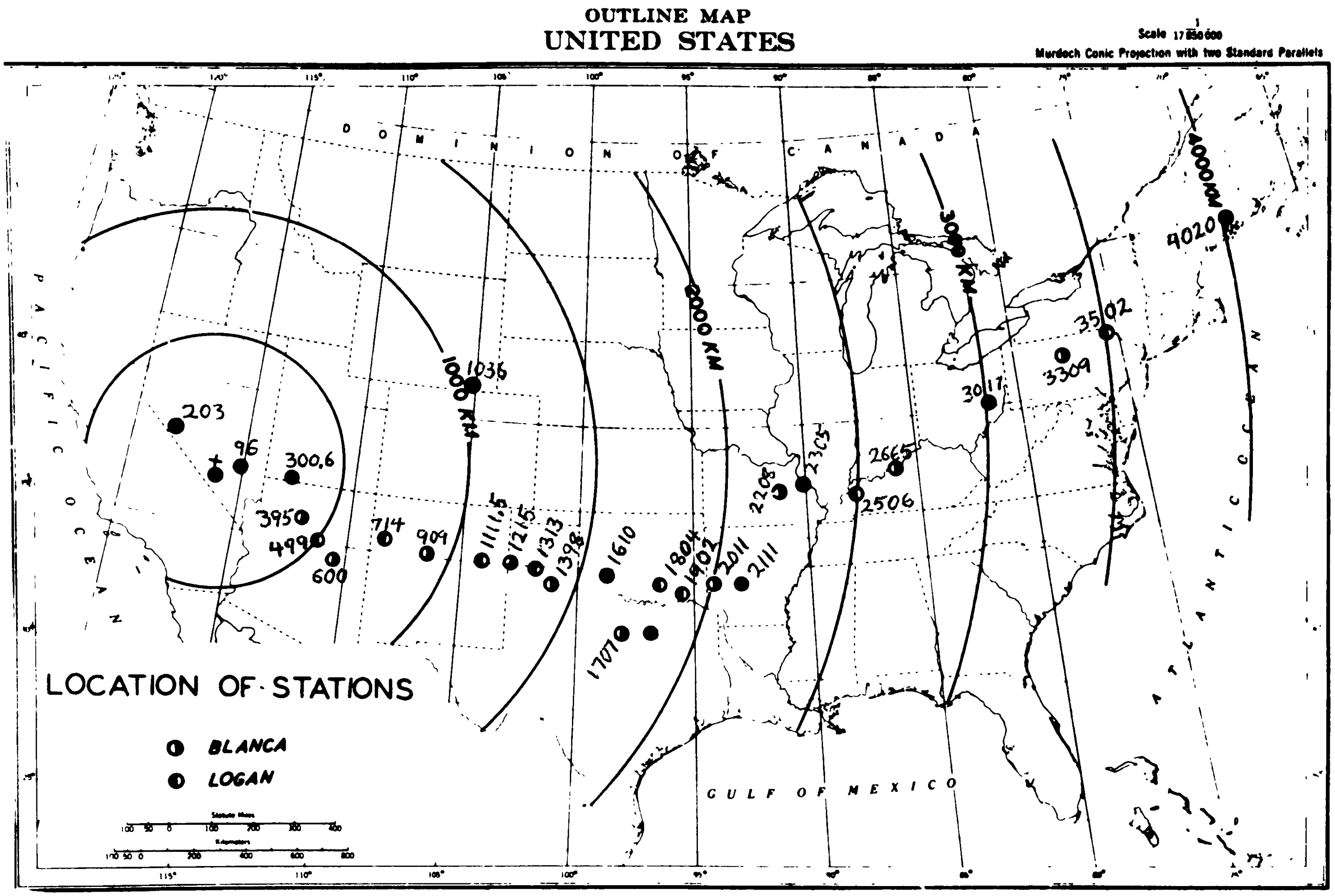

OUTLINE MAP
UNITED STATES
Scale 1/17,850,000
Murdoch Conic Projection with two Standard Parallels
DOMINION OF CANADA
PACIFIC OCEAN
ATLANTIC OCEAN
GULF OF MEXICO
1000 KM
2000 KM
3000 KM
4000 KM
203
96
300.6
3950
4990
600
714
909
1036
1111.5
1215
1313
1398
1610
1707
1804
1902
2011
2111
2208
2305
2506
2665
3017
3309
3502
4020
LOCATION OF STATIONS
BLANCA
LOGAN
Statute Miles
Kilometers

Figure 1.1.1

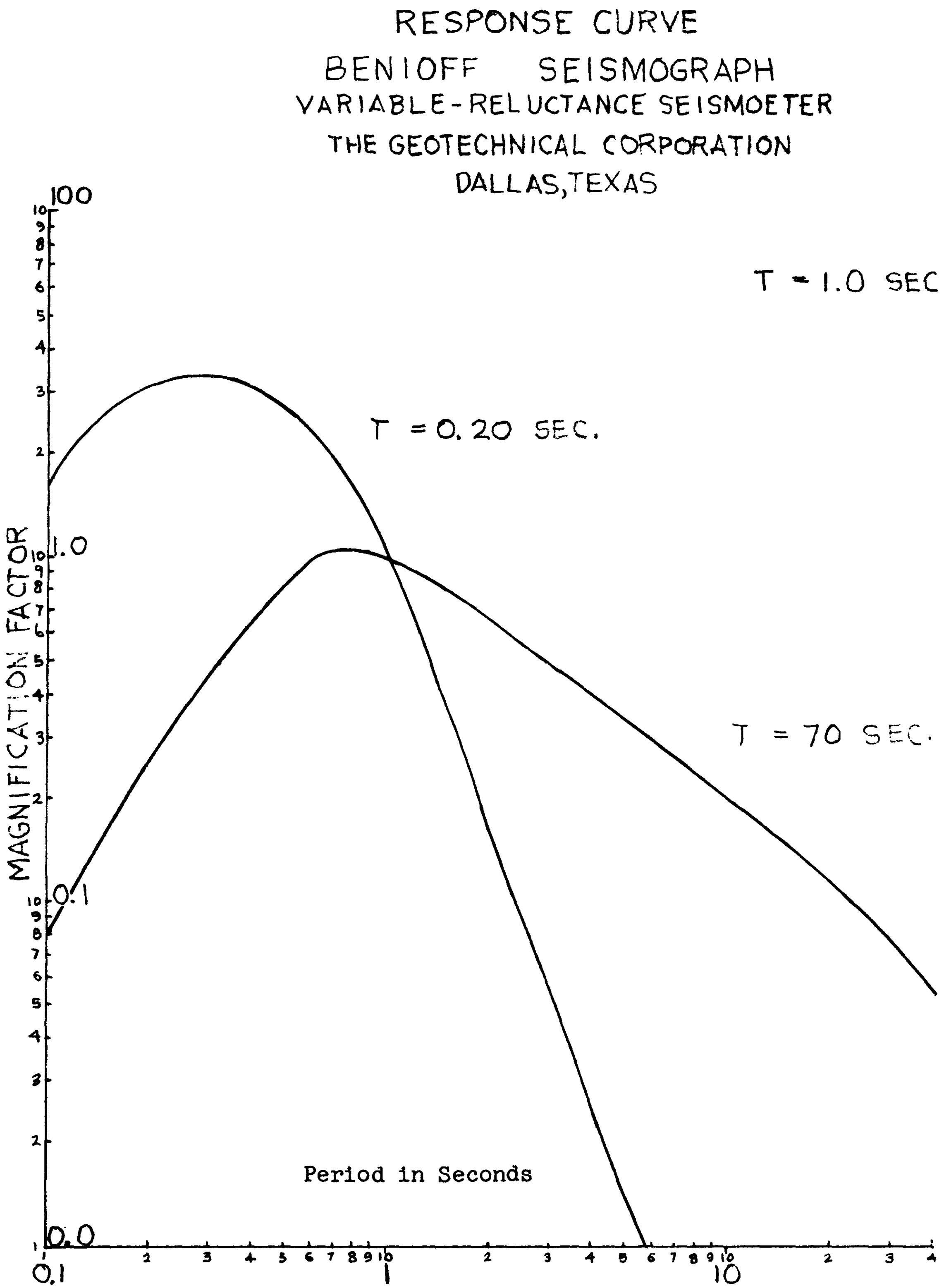
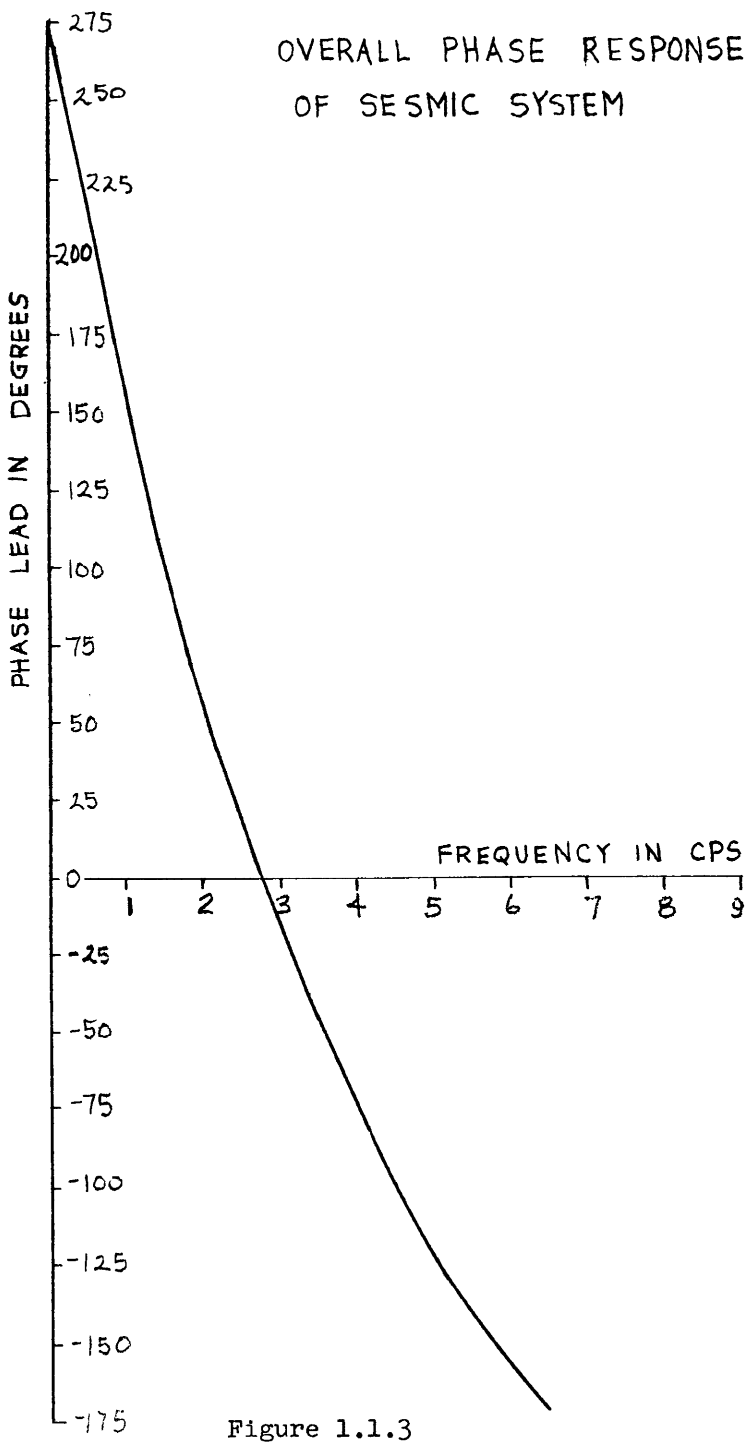

RESPONSE CURVE
BENIOFF SEISMOGRAPH
VARIABLE-RELUCTANCE SEISMOETER
THE GEOTECHNICAL CORPORATION
DALLAS,TEXAS
100
T = 1.0 SEC.
T = 0.20 SEC.
1.0
T = 70 SEC.
MAGNIFICATION FACTOR
0.1
Period in Seconds
0.0
0.1
1
10

Figure 1.1.2



OVERALL PHASE RESPONSE
OF SESMIC SYSTEM
PHASE LEAD IN DEGREES
FREQUENCY IN CPS
275
250
225
200
175
150
125
100
75
50
25
0
-25
-50
-75
-100
-125
-150
-175
1
2
3
4
5
6
7
8
9

Figure 1.1.3

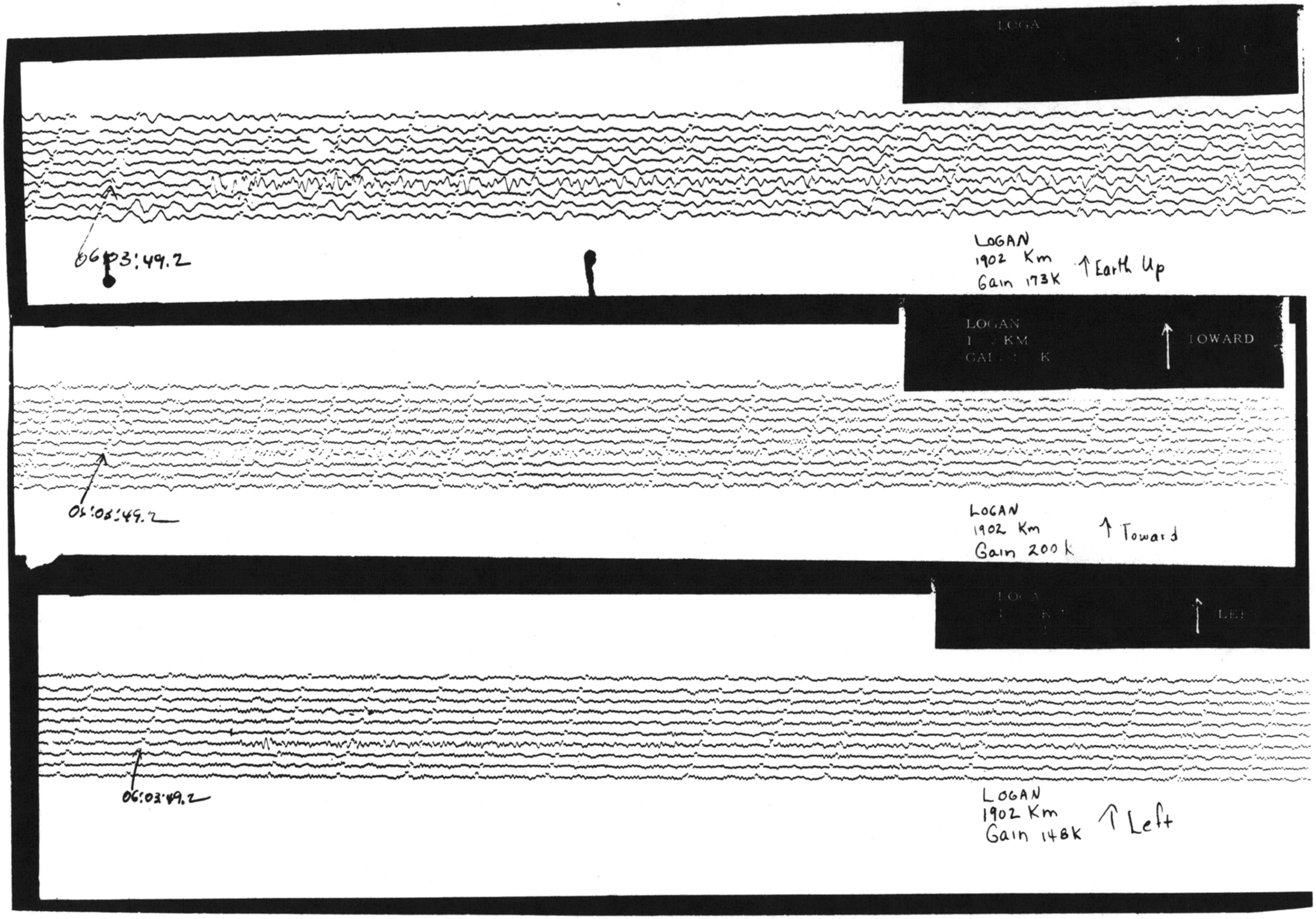

06:03:49.2
LOGAN
1902 Km
Gain 173K
↑ Earth Up
TOWARD
06:03:49.2
LOGAN
1902 Km
Gain 200 k
↑ Toward
06:03:49.2
LOGAN
1902 Km
Gain 148K
↑ Left

Figure 1.1.4

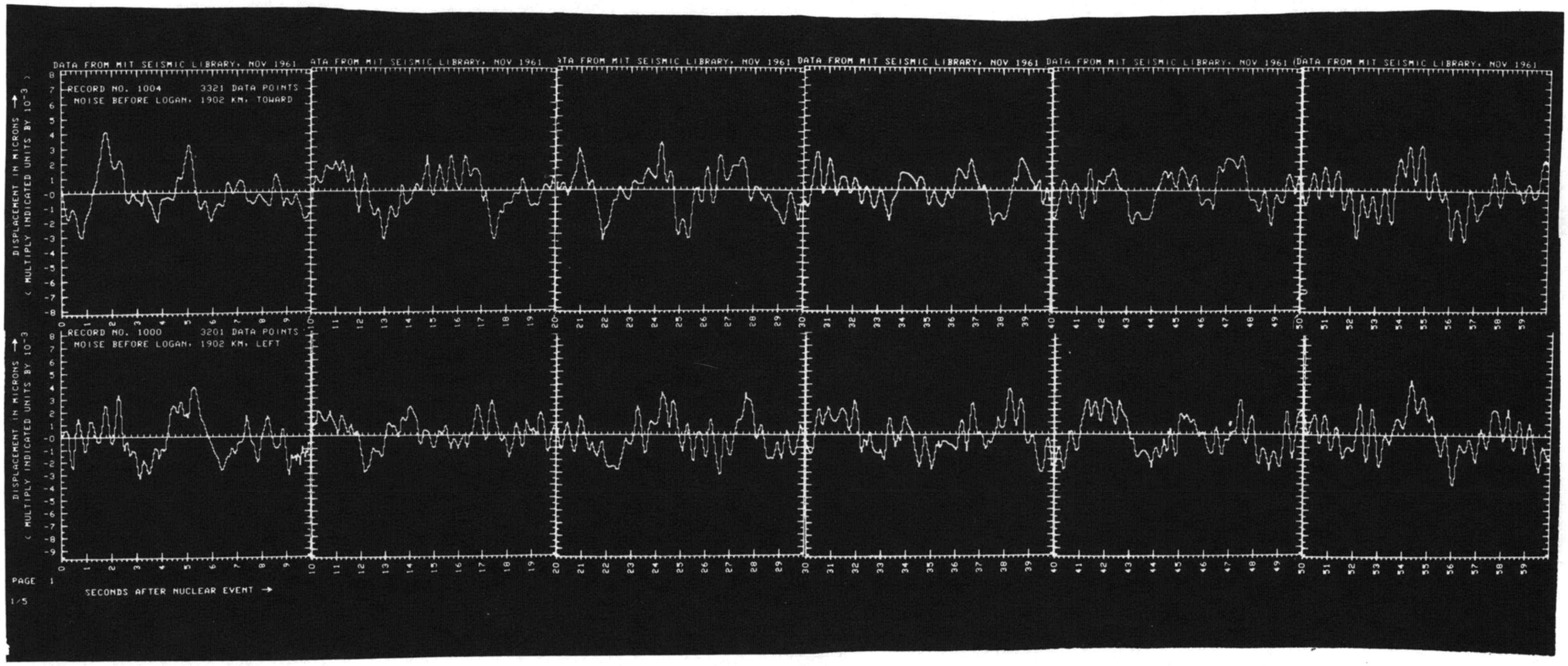

DATA FROM MIT SEISMIC LIBRARY, NOV 1961
RECORD NO. 1004 3321 DATA POINTS
NOISE BEFORE LOGAN, 1902 KM, TOWARD
RECORD NO. 1000 3201 DATA POINTS
NOISE BEFORE LOGAN, 1902 KM, LEFT
DISPLACEMENT IN MICRONS
(MULTIPLY INDICATED UNITS BY 10-3)
PAGE 1
1/5
SECONDS AFTER NUCLEAR EVENT →

FIG. 1.1.5

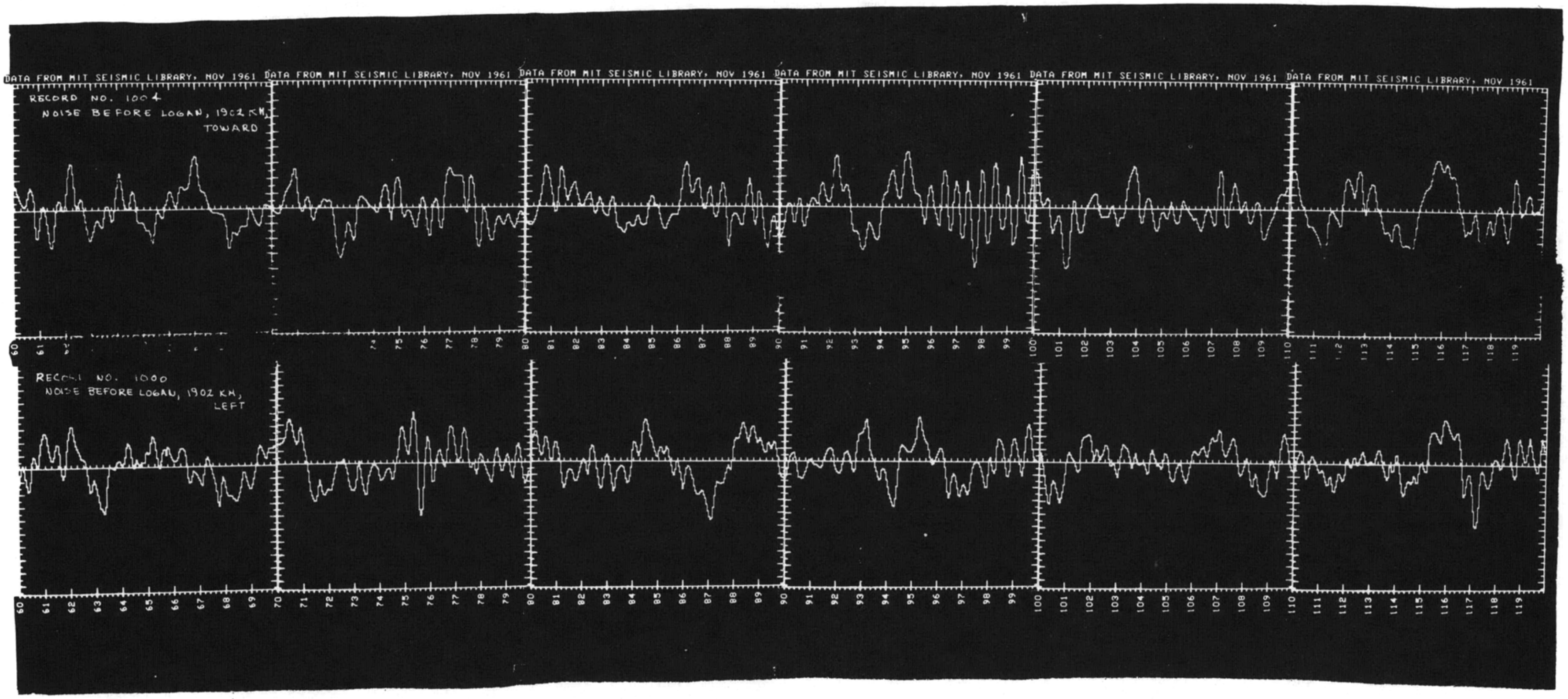

DATA FROM MIT SEISMIC LIBRARY, NOV 1961
RECORD NO. 1004
NOISE BEFORE LOGAN, 1902 KM, TOWARD
RECORD NO. 1000
NOISE BEFORE LOGAN, 1902 KM, LEFT

FIG. 1.1.6

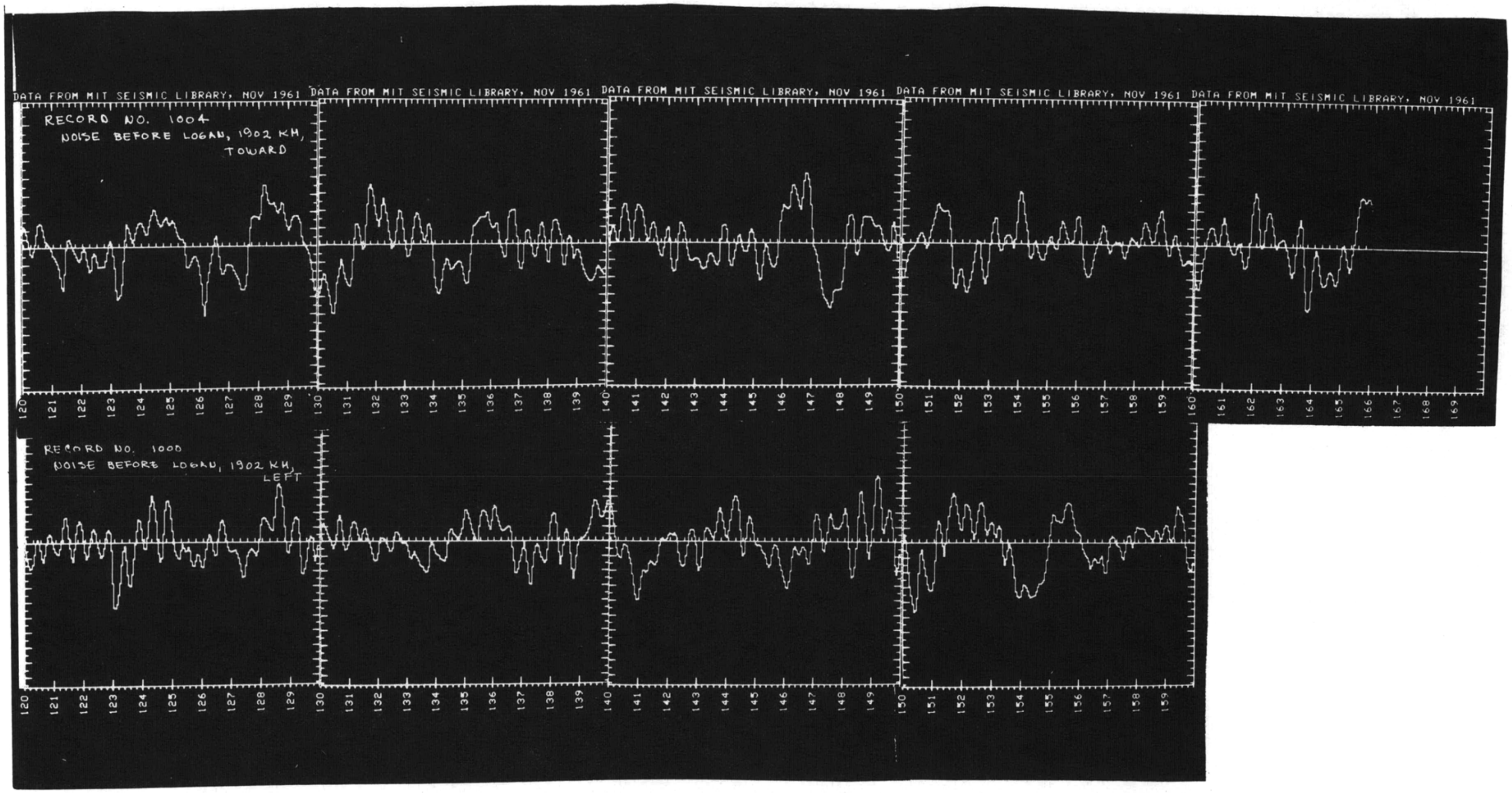

DATA FROM MIT SEISMIC LIBRARY, NOV 1961
RECORD NO. 1004
NOISE BEFORE LOGAN, 1902 KM, TOWARD
RECORD NO. 1005
NOISE BEFORE LOGAN, 1902 KM, LEFT

FIG. 1.1.7

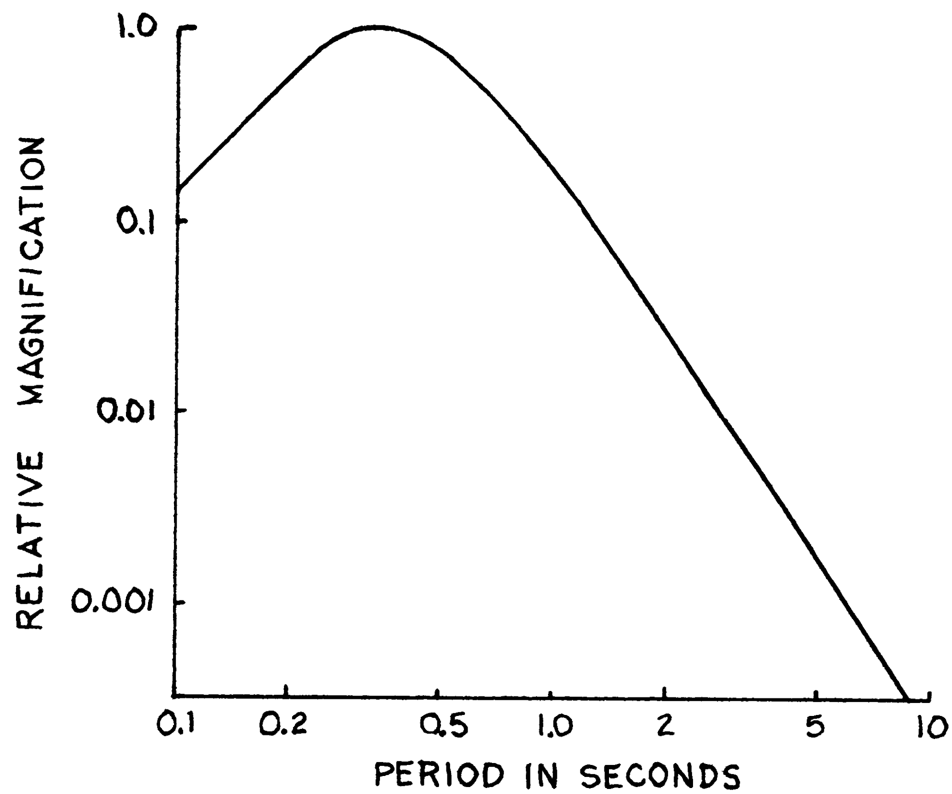

1.0
0.1
0.01
0.001
RELATIVE MAGNIFICATION
0.1
0.2
0.5
1.0
2
5
10
PERIOD IN SECONDS

Figure 1.1.8

1.2 Elementary Properties

We shall briefly consider the microseisms as a signal in a few somewhat naive computational experiments which will suffice to make apparent the need for a more general approach to the study of microseisms which can be provided by statistical techniques.

The first experiment, which is concerned with microseism amplitudes, has some bearing on microseism sources and the results are in agreement with those obtained by others. The second set of experiments deals with the identification of wave types, specifically Rayleigh and Love waves, in the microseisms. As we shall see this set of experiments failed badly because of the simplicity of the model which is used and the complexity of the microseisms themselves.

Microseism Amplitude Studies

Some studies have been made on the amplitudes at two frequencies of the noise from the Logan and Blanca records to determine the change in amplitude with distance from an ocean. If the microseisms, at the frequencies in question, are of oceanic origin, there should be a definite decrease in amplitude with distance from the coast. The frequencies and amplitudes were estimated directly from the paper records. The approximate frequency values were obtained by counting peaks over a minute or more of record. On almost all the records, the noise appeared to have two distinct frequencies, one at about .3 cycles/second, and the other near 2 cycles/second. Approximate peak amplitudes were measured on the records and averaged over several cycles of the frequency of interest. An attempt was made to choose an average noise trace before the shot.

A plot was then made of amplitude versus distance from the Atlantic <u>or</u> Pacific coast (whichever was closer) for both frequencies. These graphs appear in Figures 1.2.1 and 1.2.2 for Logan and Blanca respectively.

We can see from these figures that for low frequency the noise decreases for inland stations, but for the higher frequency there is no systematic trend. The increase in amplitude of the low frequency component at about 1400 km from a coast may be due to microseisms from the Gulf of Mexico. These rather rough quantitative results are as expected, since the low frequencies are usually assumed to be caused by ocean waves and the high frequencies are attributed to local sources, and are not correlated with the distance from the coast.

It is interesting to note that the rough computation of the frequencies involved is supported by detailed spectral analysis. Figures 1.3.6 to 1.3.9 show spectra of some of the noise and it can be seen that the important frequencies are at about .3 cps, 1.4 cps and 2 cps for the Logan and Blanca records.

<u>Rayleigh and Love Wave Experiments</u>

Much of the energy in microseismic noise has been attributed to surface waves of the Rayleigh and Love wave types. Studies by several observers mentioned in the introduction have indicated the presence of these waves in the 4 to 8 second period range. The spectrum of noise from Logan, Blanca and Cherry Hill Park records which appear in Figures 1.3.6 to 1.3.9 show spectral lines with most of the energy concentrated in fairly narrow bands. The low frequency peak, as was mentioned before, is a bit artificial, since it is the high frequency end of the oceanic microseism band with the low end cut off by the Benioff response. We might well suppose that this peak is composed of Rayleigh waves. The higher frequency lines may also be Rayleigh waves but of a non-oceanic origin. The Cherry Hill Park records in Figure 1.3.9 are remarkably similar, with rather narrow bands, even though they were taken three months apart, and one would like to investigate the important frequencies to identify wave types. Unfortunately, there are no horizontal recordings available and thus no study of this nature can be done. However, the Logan and Blanca records are three component and some attempt has been made at wave type identification. The spectra of these records, Figures 1.3.6 to 1.3.9, show in general more energy in the horizontal components at high frequency than in the vertical component. This suggests that the higher frequency noise, 1.4 cps and 2 cps, may be Love waves, and the possibility that the lower frequency energy is due to oceanic microseisms is still present.

Rayleigh waves are a special combination of P waves and S-V waves which confine all particle motion to a plane defined by the vertical and the direction of travel of the waves. For a single frequency the partical motion is retrograde elliptical. Assuming, therefore, that we have a single Rayleigh wave of a single frequency, we can resolve the horizontal components of motion into a new coordinate system which is rotated with respect to the original seismometer coordinate systems such that all horizontal motion is along one axis, the X" axis. This axis then determines the direction of travel of the wave, but not the sense of the direction. The sense can be determined from the resolved horizontal, X", and the vertical, Z", components. Since the partical motion is retrograde elliptical, X" must lead Z" by 90° for the wave to be travelling in the positive X" direction. A plot of X" against Z" should be an ellipse with its X" intercept almost 2/3 of its Z" intercept.

Records 2000, 1002 and 1004, the noise before the Logan shot 1902 km from the shot point, form a three component set and therefore can be checked in the manner described for a Rayleigh wave component. All three records were band pass filtered with a filter of width .08 cps centered at .255 cps. This frequency corresponds to the maximum of the spectrum and is possibly attributable to Rayleigh waves from oceanic sources. The two horizontal components were plotted against each other and a line fitted to the plot. The plot was fairly scattered so that the fit of the line was quite poor. The horizontal to vertical component power ratio after rotation was only 5 which is not correct for Rayleigh waves. If the plot fell exactly on a straight line the ratio after rotation would be zero. The indication is that the plot was not even close to a straight line. The resolved horizontal component was then plotted against the vertical and an ellipse was fitted to the resulting curve. This plot was the best fitting ellipse superposed is shown in Figure 1.2.7. The ellipse in this figure is a very poor fit and it is not possible to reconcile these results with the single Rayleigh wave hypothesis. This does not mean that the low frequency peaks are not Rayleigh waves. Presence of two or more Rayleigh waves from different sources could explain the lack of a linear relationship between the horizontal components and the poorly fitting ellipse to the horizontal versus vertical plot. We might note, however, that some of the motions shown in Figure 4.2.1 are relatively elliptical, but with tilted axes. Examination of the spectra (Figures 1.3.6 to 1.3.8) shows relatively more power in the vertical at .255 cps than we would expect on the Rayleigh wave hypothesis, but this could be explained by a mismatch of seismometer characteristics.

A test for the presence of Love waves was also performed on this data. The peak at about 2 cps was of interest here, since there was relatively more power in the horizontal than in the vertical. For a single Love wave we would again expect that a plot of the horizontal components would fall on a straight line. This was not the case, however, for a band width of about .08 cps centered at 2.05 cps. It is most probable that either Love or Rayleigh waves from a single source do not occur, or the band width used is too wide to see them. Cross correlation experiments could be most useful here, since the equivalent band width is the Daniell window width and the phase at each window width may be easily checked. For Rayleigh waves, we expect the horizontal to be in phase, but 90° out of phase with the vertical. For Love waves the horizontal should again be in phase, but there should be very little energy in the vertical component.

The failure of these two experiments does not eliminate the possibility of the existence of Rayleigh and Love waves at the frequencies considered, but it does illustrate the complicated nature of the noise. The suggestion is, therefore, that the structure of the microseisms is too complex to be handled by simple deterministic models. Rather than introduce more complicated models which require an enormous amount of labor to fit to the data, we shall consider the microseisms as stochastic time series and treat them from the statistical point of view.

Apparent Stationarity

The majority of the results of time series analysis are applicable to stationary time series, that is, series whose probability densities are not dependent on absolute time. If in a time series the probability, $P_{\xi_1}(x_1;t_1)\;dx_1$, that ξ_1 is in the interval (x_1, x_1+dx_1) at time t_1 is the same for all t, and if the probability $P_{\xi_1\xi_2}(x_1,x_2;t_1,t_2)$ that at time t_1, ξ_1 is in the interval (x_1, x_1+dx_1) and at time t_2, ξ_2 is in the interval (x_2, x_2+dx_2) is dependent only on the time separation $\tau = t_2 - t_1$ and not on absolute time, the time series is said to be wide sense stationary. If all higher densities $P_{\xi_1,\xi_2,\ldots\xi_n}(x_1,x_2\ldots x_n; t_1,t_2,\ldots,t_n)$ are also independent of absolute time and dependent only on $\tau_k = t_j - t_i$ the series is strictly stationary.

It is obvious that microseism records are not stationary over long periods of time since microseism activity is strongly influenced by meteorological conditions. Over short periods of time, however, when there have been no great changes in the generating mechanisms for microseisms, the records can be considered stationary. For our purposes we need only be concerned with stationarity over the few hours necessary to record the shot signal and noise before and after the signal. We now consider an ensemble or group of time series lined up one beneath the other each with the same first and second probability densities. We arbitrarily label time on these series so that a vertical line strikes each time series at the same time. The ensemble can be constructed by breaking up a long time series into smaller pieces and considering each piece as a member of the ensemble. In the case of microseismic noise, the noise before and the noise after the event can be considered as two members of the ensemble. We wish then to see if the probability densities are approximately the same for these ensemble members. We can do this computing directly the probability densities, but this becomes a lengthy process for the second density, $P_{\xi_1,\xi_2}(x_1, x_2; t_1, t_2)$ and it is worse for the higher densities. If we are only interested in wide sense stationarity we can consider time and ensemble averages and, assuming that the ensemble is ergodic, equate these averages. The ensemble average of ξ_1 at time t_1 and ξ_2 at time t_2 is

$$Ave = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1 x_2 P_{\xi_1,\xi_2}(x_1, x_2; \tau)\, dx_1\, dx_2 \quad , \tau = t_2 - t_1$$

The time average is

$$Ave = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} f(t)\, f(t+\tau)\, dt$$

We note that the time average is the autocorrelation and that the Fourier transform of the autocorrelation is the power density spectrum (see section 1.3). Hence, under the ergodic hypothesis, the constancy of the spectral density in time reflects the wide-sense stationarity of the time series. Spectral density computations have been performed on the noise before and noise after the shot and the results are shown in Figures 1.3.6 to 1.3.8. One can easily see that the general character of the spectrum does not change much over a period of time representing two drum revolutions of the Benioff. This strongly suggests that the microseisms are, for our purposes wide-sense stationary.

Mean and Variance

Time series analysis simplifies to some extent if the series have zero mean and unit variance. The digitized records had the best least squares fitting segmented mean line removed, but this does not guarantee that the mean is zero. The mean is, however, quite small and can usually be considered zero. It can easily be computed and subtracted off if necessary. The variance of the records is not unity and no scaling has been done to make it so.

Amplitude Distribution and Normality Test

The amplitude distribution of the records can easily be computed and, given the mean and standard deviation (square root of the variance), the corresponding normal distribution can be found and compared with the empirical amplitude distribution. Appendix B gives a flow graph of the necessary steps in the comparison of the distributions and the programs necessary. Appendix G contains listings of the programs. The comparison is done by finding the values along the x axis which divide the appropriate normal density (given mean and standard deviation) into sections of equal area (equal probability). A count is then made of the number of amplitude values which fall into each section. The chi square comparison measure is then

$$\chi^2 = \sum_{i=1}^{L} \frac{(N_i - pN)^2}{pN}$$

where there are L sections and N amplitude data points, $p = 1/L$, and N_i is the number of points which fall in the i-th section. There are $L-3$

degrees of freedom since the mean and standard deviation are used to determine the appropriate Gaussian. The chi square measure thus defined is chi square distributed and its expected value depends only on (Cramer, 1946). The probability $P(\chi^2)$ of exceeding χ^2 is the quantity of importance in comparison. Acceptance regions for X^2 are generally set so that $P(X)^2 \geq .1$ or .01. Comparisons were made between empirical and normal probability densities for all the Logan and Blanca noise records listed in Table 1.1.1. The chi square test was used as a measure of goodness of fit and the results are shown in Table 1.4.1 in section 1.4. The probability of exceeding X^2 varies considerably and for the records shown only six or seven can be considered normally distributed for this test. Figures 1.2.3 and 1.2.4 show some of the empirical frequency ratio plots and Figures 1.2.5 and 1.2.6 show typical computer output from the normalcy and independence tests. It can be seen from these figures that even though some of the densities fail the X^2 test, they look fairly Gaussian and to a rough approximation may be considered normal.

(Note: If the alternate method of test for normality which is given in section 1.4 is used, all records are found to be Gaussian.)

The independence tests are discussed further in section 1.4 and in Appendix C. It is sufficient to say here that the amplitudes are not independent.

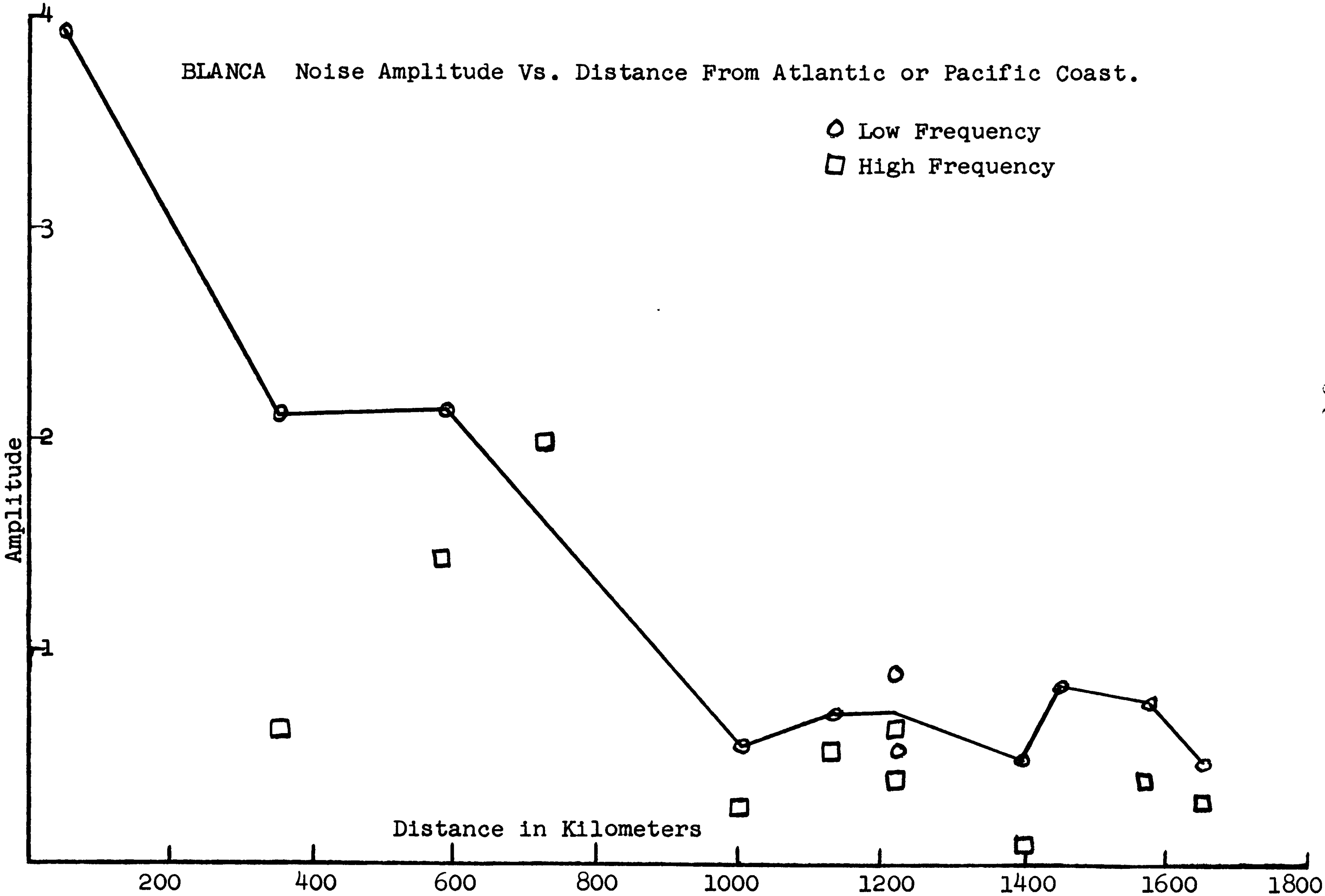

BLANCA Noise Amplitude Vs. Distance From Atlantic or Pacific Coast.
Low Frequency
High Frequency
Amplitude
1
2
3
4
Distance in Kilometers
200
400
600
800
1000
1200
1400
1600
1800

Figure 1.2.1

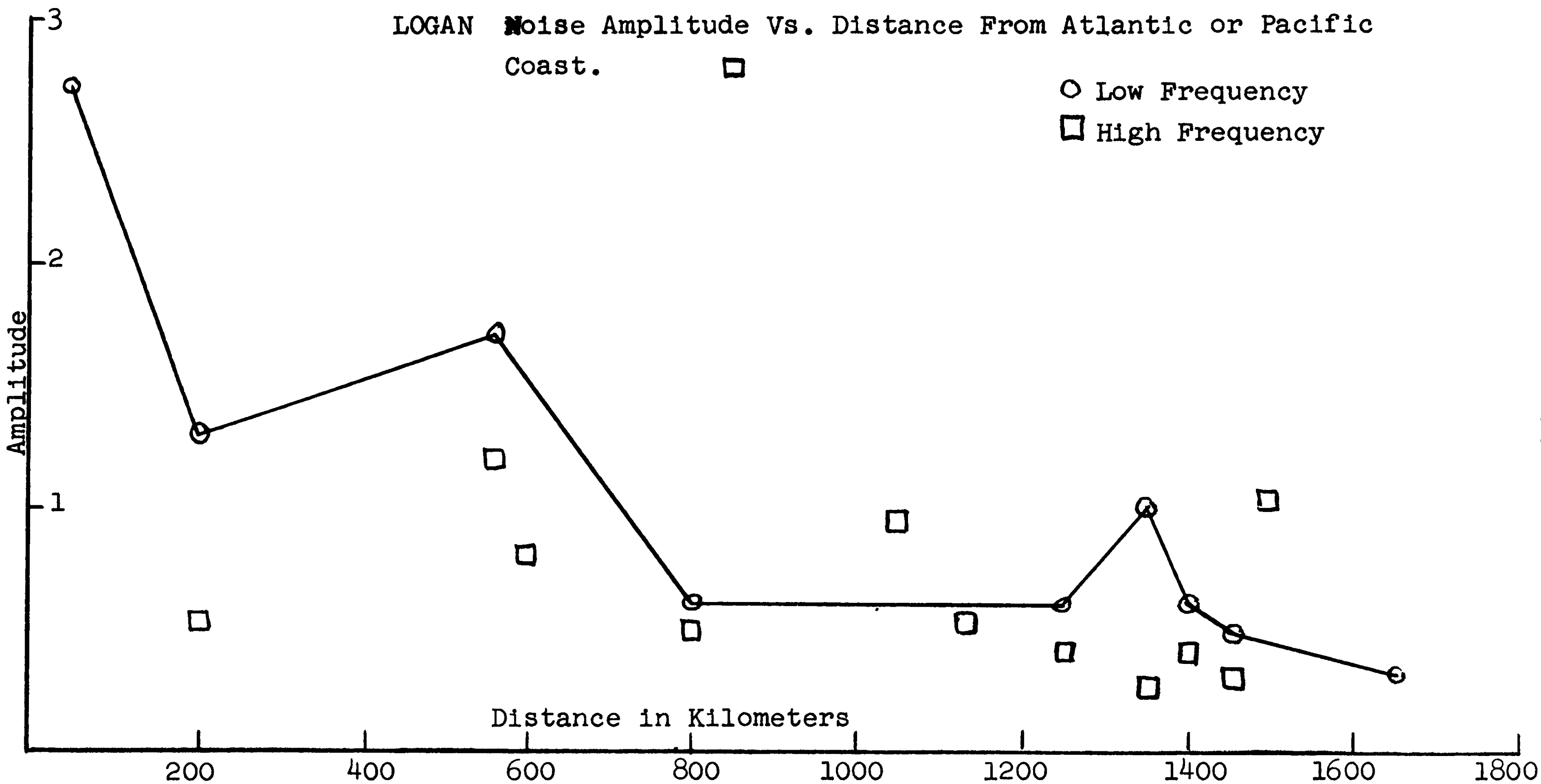
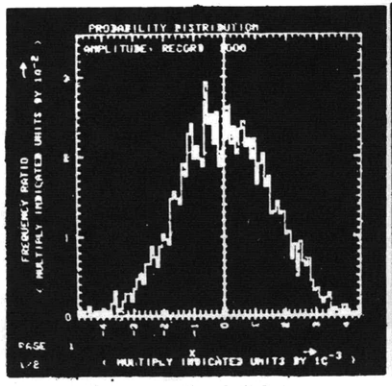
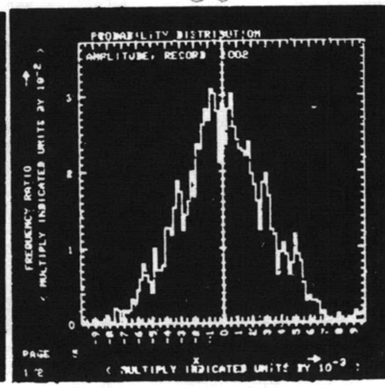

LOGAN Noise Amplitude Vs. Distance From Atlantic or Pacific Coast.
Low Frequency
High Frequency
Amplitude
3
2
1
Distance in Kilometers
200
400
600
800
1000
1200
1400
1600
1800

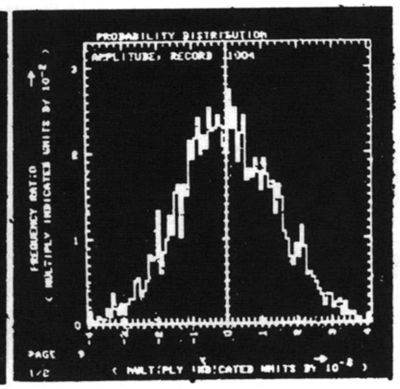
Figure 1.2.2



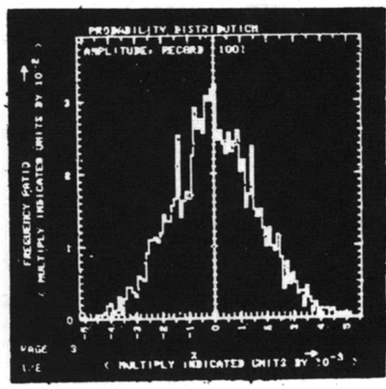
Record 1000



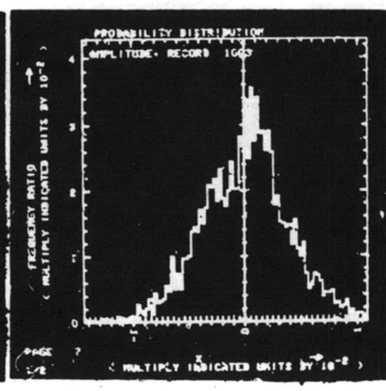
Record 1002



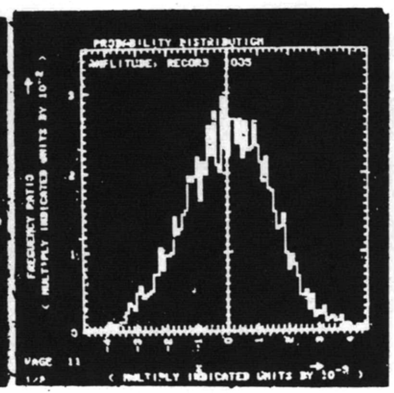
Record 1004



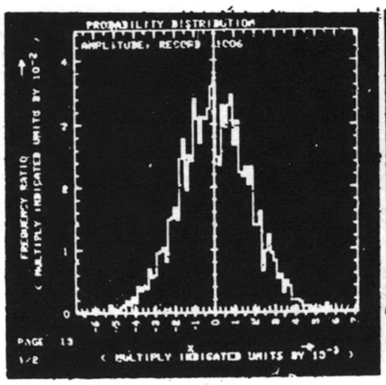
Record 1001



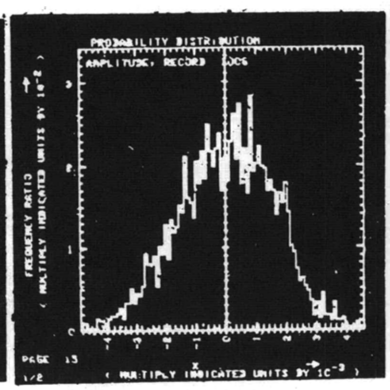
Record 1003



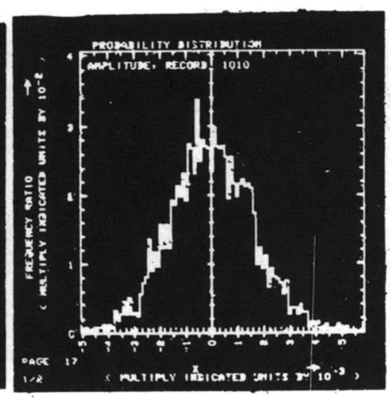
Record 1005



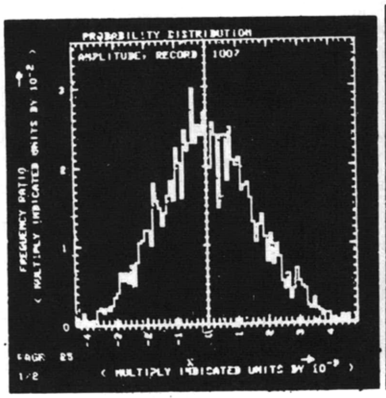
Record 1006



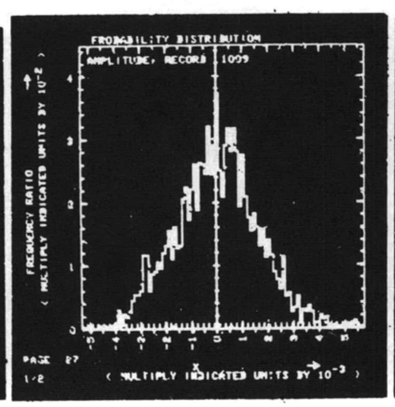
Record 1008



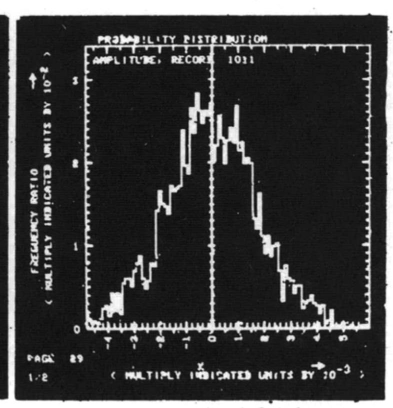
Record 1010



Record 1007

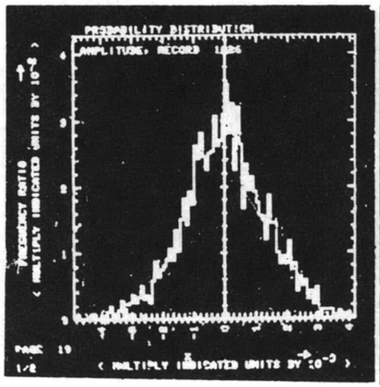


Record 1009

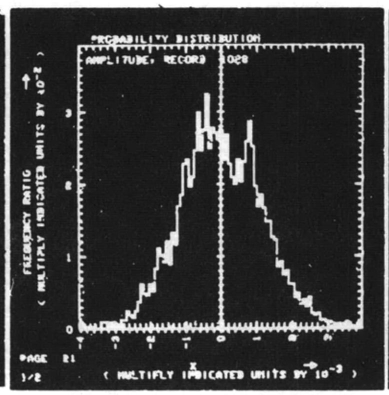


Record 1011

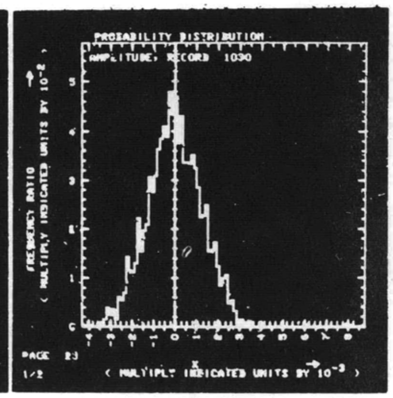
Figure 1.2.3 Frequency Ratios of Microseism Amplitudes



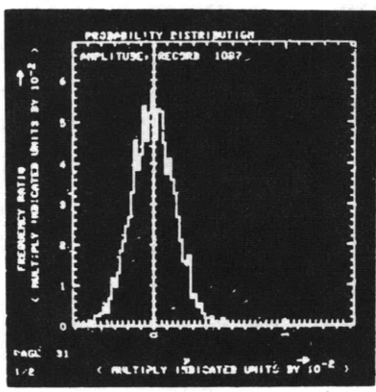
Record 1026



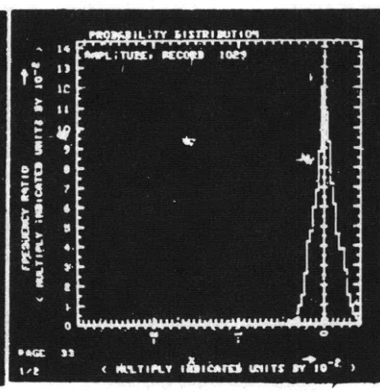
Record 1028



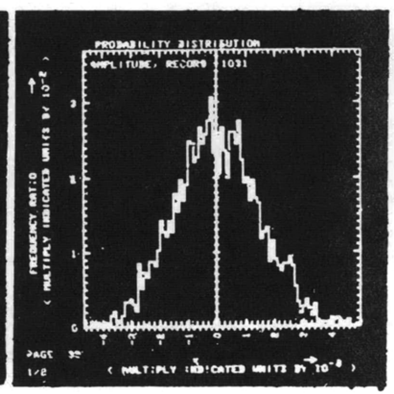
Record 1030



Record 1027



Record 1029



Record 1031

Figure 1.2.4 Frequency Ratios of Microseism Amplitudes

ANALYSIS OF AMPLITUDE DISTRIBUTION FOR RECORD 1005

COMPARISON OF ACTUAL DISTRIBUTION AND NORMAL DISTRIBUTION

NUMBER OF RANGES= 57
LENGTH OF SERIES= 3321
DEGREES OF FREEDOM= 54
MEAN OF SERIES= -0.22500189E-05
STANDARD DEVIATION= 0.14274400E-02

HIGHER CENTRAL MOMENTS
THIRD MOMENT= -0.19685886E-09
FOURTH MOMENT= 0.12106580E-10
FIFTH MOMENT= -0.12533012E-14
SIXTH MOMENT= 0.11494952E-15

EXPECTED COUNT= 58.2632

CHI-SQUARE= 0.62046965E 02
PROBABILITY OF EXCEEDING CHI-SQUARE= 0.21316E-00

POKER COUNT TEST RESULTS

HAND TYPE	ACTUAL COUNT	EXPECTED COUNT
BUST	35	196.01280
1 PAIR	138	334.65599
2 PAIR	81	71.71200
3 OF A KIND	117	47.80800
FULL HOUSE	20	5.97600
STRAIGHT	95	4.78080
4 OF A KIND	105	2.98800
5 OF A KIND	73	0.06640

MEAN SQUARE CONTINGENCY= 0.27838460E 01

DEPENDENCY MEASURE= 0.30931623E-00

PROBABILITY DISTRIBUTION

NUMBER OF VALUES IN EACH OF 100 EQUALLY SPACED RANGES FROM
-0.47553504E-02 TO 0.45647645E-02. 3321 VALUES IN ALL.

1.	1.	0.	0.	1.	0.	1.	1.	2.	4.
3.	1.	3.	4.	4.	8.	12.	11.	9.	16.
19.	14.	15.	16.	17.	24.	24.	35.	26.	32.
32.	33.	48.	41.	43.	49.	51.	65.	63.	65.
73.	55.	71.	66.	86.	74.	92.	70.	67.	98.
77.	74.	89.	89.	79.	89.	73.	88.	76.	77.
88.	78.	71.	63.	73.	60.	59.	50.	43.	44.
49.	33.	26.	32.	28.	23.	15.	15.	17.	9.
15.	6.	8.	7.	10.	4.	5.	6.	4.	4.
6.	1.	3.	1.	1.	0.	1.	2.	1.	3.

Figure 1.2.5

ANALYSIS OF AMPLITUDE DISTRIBUTION FOR RECORD 1026

COMPARISON OF ACTUAL DISTRIBUTION AND NORMAL DISTRIBUTION

NUMBER OF RANGES= 59
LENGTH OF SERIES= 3581
DEGREES OF FREEDOM= 56
MEAN OF SERIES= -0.37916552E-07
STANDARD DEVIATION= 0.13271835E-02

HIGHER CENTRAL MOMENTS
THIRD MOMENT= -0.84812047E-10
FOURTH MOMENT= 0.97164132E-11
FIFTH MOMENT= -0.29763772E-14
SIXTH MOMENT= 0.86117256E-16

EXPECTED COUNT= 60.6949

CHI-SQUARE= 0.10001674E 03
PROBABILITY OF EXCEEDING CHI-SQUARE= 0.15617E-03

POKER COUNT TEST RESULTS

HAND TYPE	ACTUAL COUNT	EXPECTED COUNT
BUST	38	211.36320
1 PAIR	159	360.86399
2 PAIR	133	77.32800
3 OF A KIND	111	51.55200
FULL HOUSE	8	6.44400
STRAIGHT	84	5.15520
4 OF A KIND	112	3.22200
5 OF A KIND	71	0.07160

MEAN SQUARE CONTINGENCY= 0.23302333E 01

DEPENDENCY MEASURE= 0.25891481E-00

PROBABILITY DISTRIBUTION

NUMBER OF VALUES IN EACH OF 100 EQUALLY SPACED RANGES FROM -0.48722361E-02 TO 0.41697387E-02. 3581 VALUES IN ALL.

1.	2.	0.	0.	0.	0.	0.	3.	0.	0.
1.	2.	4.	7.	1.	3.	5.	10.	1.	8.
9.	13.	9.	12.	18.	11.	13.	9.	21.	31.
23.	27.	29.	32.	38.	32.	48.	37.	54.	65.
51.	69.	62.	94.	87.	101.	88.	81.	90.	91.
110.	94.	97.	111.	127.	101.	117.	81.	115.	95.
60.	84.	70.	77.	69.	63.	56.	54.	43.	67.
52.	52.	36.	30.	34.	30.	42.	27.	30.	15.
23.	18.	21.	11.	15.	8.	17.	2.	5.	5.
5.	5.	1.	2.	4.	1.	1.	3.	1.	1.

Figure 1.2.6

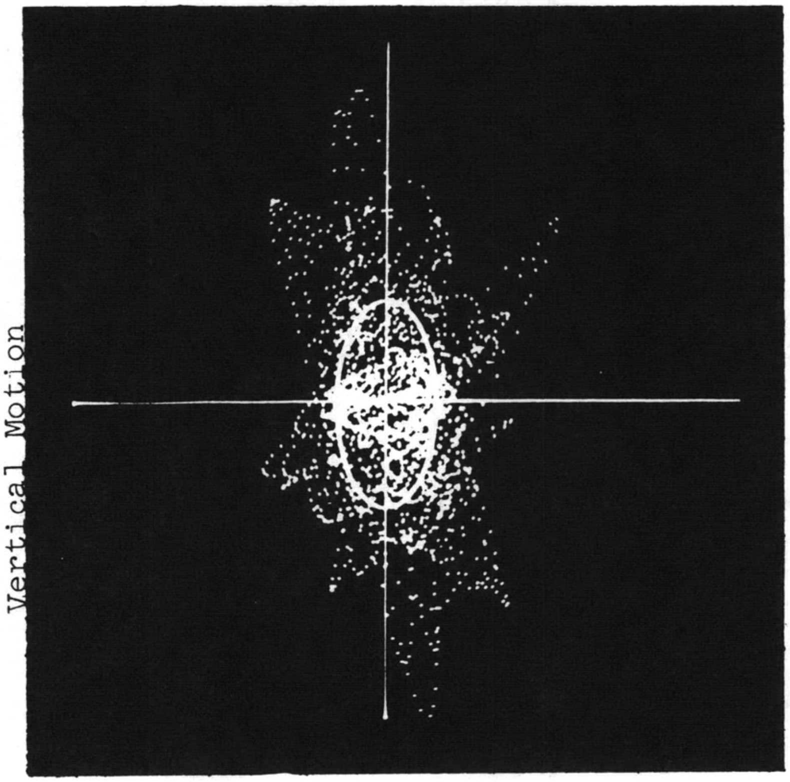


Figure 1.2.7 Results of Rayleigh Wave Experiment on Records 1000, 1002 and 1004 with Best Fitting Ellipse.

1.3 Correlation and Spectral Properties

Description of Random Functions - Correlation and Spectrum

The description of the spectrum of a random function, such as microseismic noise as recorded on a seismogram, cannot be adequately done by simple Fourier transformation since the Fourier transform specifies the phase spectrum and immediately particularizes the function thus setting it aside from all the other possible realizations of the random process. In order to treat all the members of the ensemble simultaneously we must make use of the Wiener theorem for autocorrelation. The autocorrelation, $\varphi(\tau)$, of a continuous time function $f(t)$ is defined as

$$\varphi(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{T}^{-T} f(t) f(t+\tau)\, dt$$

With a change of variables $r = t - \tau$ we can see that $\varphi(\tau) = \varphi(-\tau)$. The Wiener theorem then states that the power density spectrum $\Phi(\omega)$ of $f(t)$ is the cosine transform of $\varphi(\tau)$ (Lee, 1960).

$$\Phi(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \varphi(\tau) \cos\omega\tau \, d\tau$$

We see that the autocorrelation has the effect of bringing all the phases down to zero thus throwing away the phase information which pins down a particular member of the ensemble.

The continuous infinite theory has its counterpart in discrete finite time, but with some modification and some problems.

Digitization and Aliasing

Digitization or division into discrete time puts some restriction on the description in the frequency domain. One must pay the price for throwing away the information between the digitized points and that price, as specified by the sampling theorem, is that one can only see frequencies which are less than or equal to half the sampling rate. If there are n samples per second we can only distinguish up to $n/2$ cycles per second, the Nyquist frequency, which corresponds to a radian frequency of $\omega = \pi$. If the data actually contain a frequency higher than $n/2$ cps., say $n/2 + \Delta$, this frequency will be folded down to $n/2 - \Delta$, since $\cos(\pi + \delta) = \cos(\pi - \delta)$, and this process is called aliasing. Thus if there are frequencies present higher than $n/2$ cps, the spectral estimate at frequency f, $(0 < f < n/2)$, is made up of frequencies f, $2(n/2) \pm f$, $4(n/2) \pm f$, $\ldots M(n/2) \pm f$, M even, and the spectrum loses meaning. One can avoid this problem by sampling often enough to include all frequencies or by low pass filtering before digitization.

Spectral Estimation - Daniell Window and Variance of Estimate

The fact that the data is known for a finite length of time requires an assumption about the data outside of the interval in which it is known since the autocorrelation $\varphi(\tau)$ involves this time. One usually assumes that the data is zero outside this interval and the autocorrelation must therefore go to zero when τ equals the interval length. This is the complete transient (Wiener) autocorrelation

$$\varphi(\tau) = \frac{1}{N} \sum_{i=0}^{N-|\tau|} x_i \, x_{i+\tau} \,, \quad \tau = 0, \pm 1, \pm 2, \ldots, \pm(N-1)$$

where there are N data points, $x_1, x_2, \ldots x_N$. Some methods of estimating the autocorrelation such as the Tukey estimation try to compensate for the fact that the data is zero outside $i = 1, \ldots N$ by adding weighting factors

$$\varphi(\tau) = \frac{1}{N-|\tau|} \sum_{i=0}^{M} x_i \, x_{i+\tau} \,, \qquad \tau = 0, \ldots \pm M$$

where M is less than N (e.g. $M = N/5$). The higher lag terms (τ large) are thus given more weight to compensate for the smaller number of terms in the summation. This will, of course, result in a biased estimate.

In any case the computed spectrum, $\Phi_c(\omega)$, is an estimate of the true $\Phi(\omega)$ and can be though of as a convolution of some weighing function $W(\omega)$ with the true spectrum

$$\Phi_c(\omega) = \Phi_T(\omega) * W(\omega)$$

where the asterisk denotes convolution. $W(\omega)$ is then called the spectral window (Blackman and Tukey, 1958). Ideally the spectral window is rectangular and the convolution process will then move it along the true spectrum and the estimate at ω_K, $\Phi_c(\omega_K)$ will be an unweighted average of the true spectrum $\Phi_T(\omega)$ from ω_{K+h} to ω_{K-h}

where $2h$ is the window width. Since convolution in one domain is multiplication in the other, the Fourier transform of $\Phi_T(\omega) * W(\omega)$ is $\varphi_T(\tau)\, W(\tau)$ where $\varphi_T(\tau)$ is the true autocorrelation.

The spectral estimate which has been used to compute the spectra and cross spectra shown in this thesis is the Daniell estimate. The Daniell method uses the complete transient (Wiener) autocorrelation of the time function X_t, $t = 1, \ldots, N$

$$\varphi(\tau) = \frac{1}{N} \sum_{t=1}^{N-|\tau|} X_t X_{t+\tau} , \qquad \tau = 0, \pm 1, \ldots \pm (N-1)$$

The Daniell spectral estimate $\Phi_D(\omega)$ is then

$$\Phi_D(\omega) = \frac{1}{2\pi} \sum_{\tau=-(N-1)}^{N-1} \varphi(\tau) \frac{\sin \frac{\pi \tau}{M}}{\pi \tau / M} \cos \omega \tau \qquad (1.3.1)$$

where $\dfrac{\sin(\pi \tau / M)}{\pi \tau / M}$ is the Daniell weighting function.

We note that the spectral window is not simply the Fourier transform of the Daniell weight since $\varphi(\tau)$ is not the true autocorrelation. We can, however, compute the spectral window if we choose a time function X_t for which we know $\Phi_T(\omega)$ (Simpson et al, 1961b). If the time function X_t is N points of a sine wave $\sin \omega_r t$ we know that $\Phi_T(\omega)$ is a delta function $\delta(\omega_r)$ so that the spectral estimate becomes

$$\Phi_D(\omega) = \Phi_T(\omega) * W(\omega)$$

$$\Phi_D(\omega) = \delta(\omega_r) * W(\omega) = W(\omega - \omega_r)$$

Hence we compute the transient autocorrelation $\varphi(\tau)$ from the N points of the sine wave, weight this with the Daniell weighting function and take the cosine transform as indicated in equation (1.3.1) to obtain the overall spectral window for the computational process. This has been done (Simpson et al 1961b, Appendix K) for $\omega_r = \pi/2$ which leads to an X_t of $X_t = \ldots, 1, 0, -1, 0, 1, \ldots\ldots,$ and a correspondingly simple autocorrelation function. It can be seen that the Daniell estimate has parameters M and N, and therefore spectral windows were computed for several different M and N values. A few examples of the windows have been included in Figure 1.3.1 to 1.3.4 (Simpson et al, 1961b). These figures show that the windows are always non-negative, they tend to get squarer as the M/N ratio decreases and they are essentially non-oscillatory. The variance, σ_D^2, of the Daniell estimate has been worked out by E. A. Robinson (Simpson et al, 1961b, 1962a) and is

$$\sigma_D^2 = \frac{\pi}{2Nh^2} \int_{\omega_0-h}^{\omega_0+h} \Phi_T^2(\omega)\, d\omega$$

where $h = \pi/M$ and N is the number of data points. As an approximation to this we have used

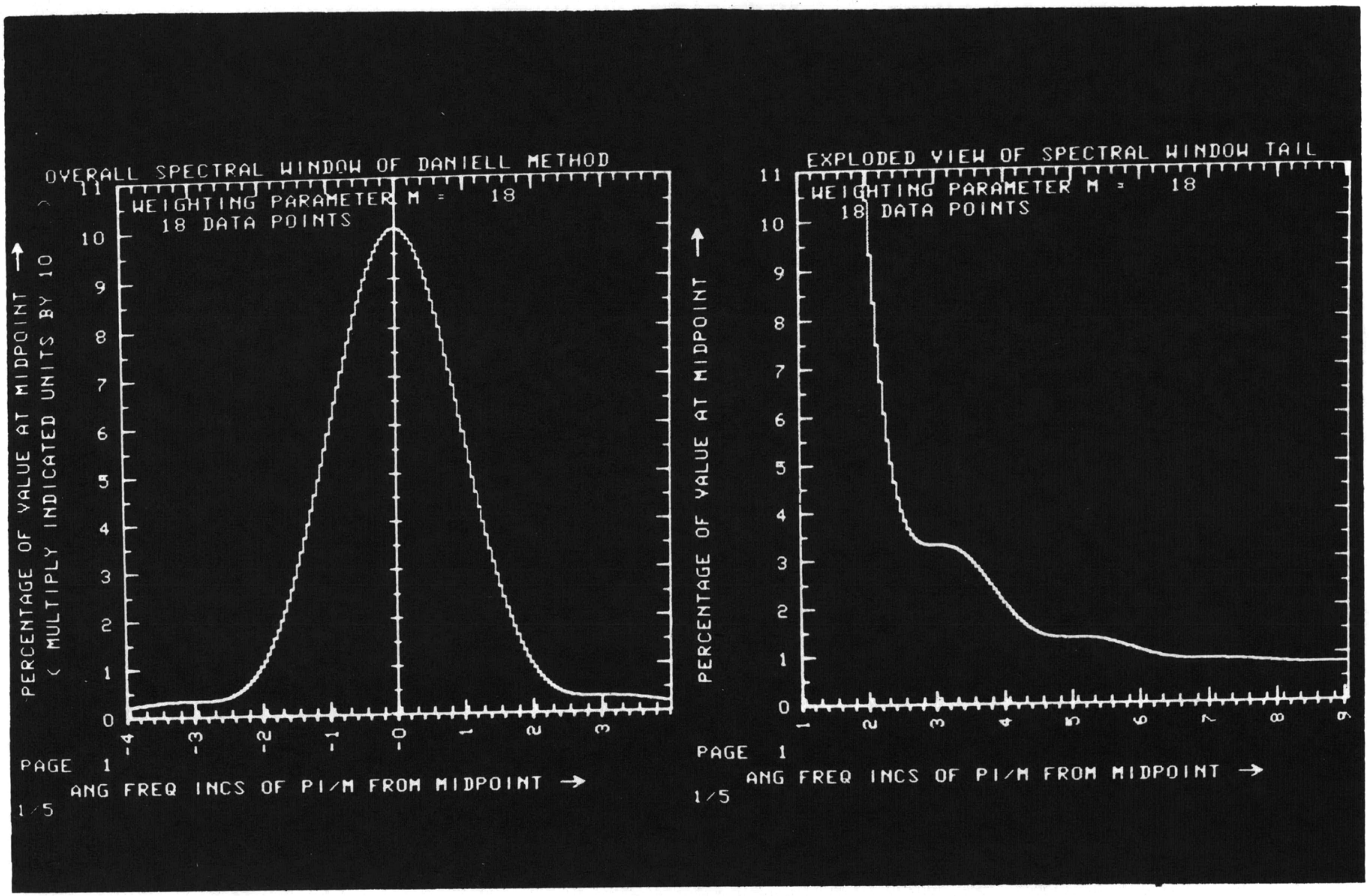
$$\sigma_A^2 = \frac{\pi}{2Nh^2} \left[\Phi_D^2(\omega)\ 2h \right]$$

$$\sigma_A = \sqrt{\frac{M}{N}}\ \Phi_D(\omega).$$

Figure 1.3.5 shows a plot of the Daniell spectrum (solid line) of a typical noise record with dotted line denoting the approximate standard deviation, σ_λ , plotted above and below the solid line. The spectra are plotted as histograms since the value at any one frequency is an estimate averaged over the spectral window width. We note that M is the number of spectral estimates between $\omega = 0$ and π . One can then see that the N/M ratio is an estimate of the number of cycles of a sine wave which the data affords and therefore an increase in N/M ratio (decrease in M/N) means that one is looking at more cycles and can therefore make a better estimate of the frequency. This is, of course, just the uncertainty principle.

Spectrum and Benioff Response

It is important to remember that the data was recorded on a Benioff seismometer and that the spectrum we see is observed through the eye of the Benioff. The apparent spike at low frequency, .25 cps, is artificial since the Benioff cuts off the lows. The sharp cut off on the low frequency side of the major low frequency feature in the spectrum of Figure 1.3.5 and other spectra in Figures 1.3.6 to 1.3.9 is a result of the seismometer response and is not a real phenomenon. We notice from Figure 1.3.2 that there is essentially no energy at frequencies greater than 2.5 cps so that, with our sampline rate of 20 samples per second, there is no problem with aliasing of frequencies.



OVERALL SPECTRAL WINDOW OF DANIELL METHOD
WEIGHTING PARAMETER M = 18
18 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
(MULTIPLY INDICATED UNITS BY 10
0 1 2 3 4 5 6 7 8 9 10 11
-4 -3 -2 -1 -0 1 2 3
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
1/5
EXPLODED VIEW OF SPECTRAL WINDOW TAIL
WEIGHTING PARAMETER M = 18
18 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
0 1 2 3 4 5 6 7 8 9 10 11
1 2 3 4 5 6 7 8 9
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
1/5

Figure 1.3.1

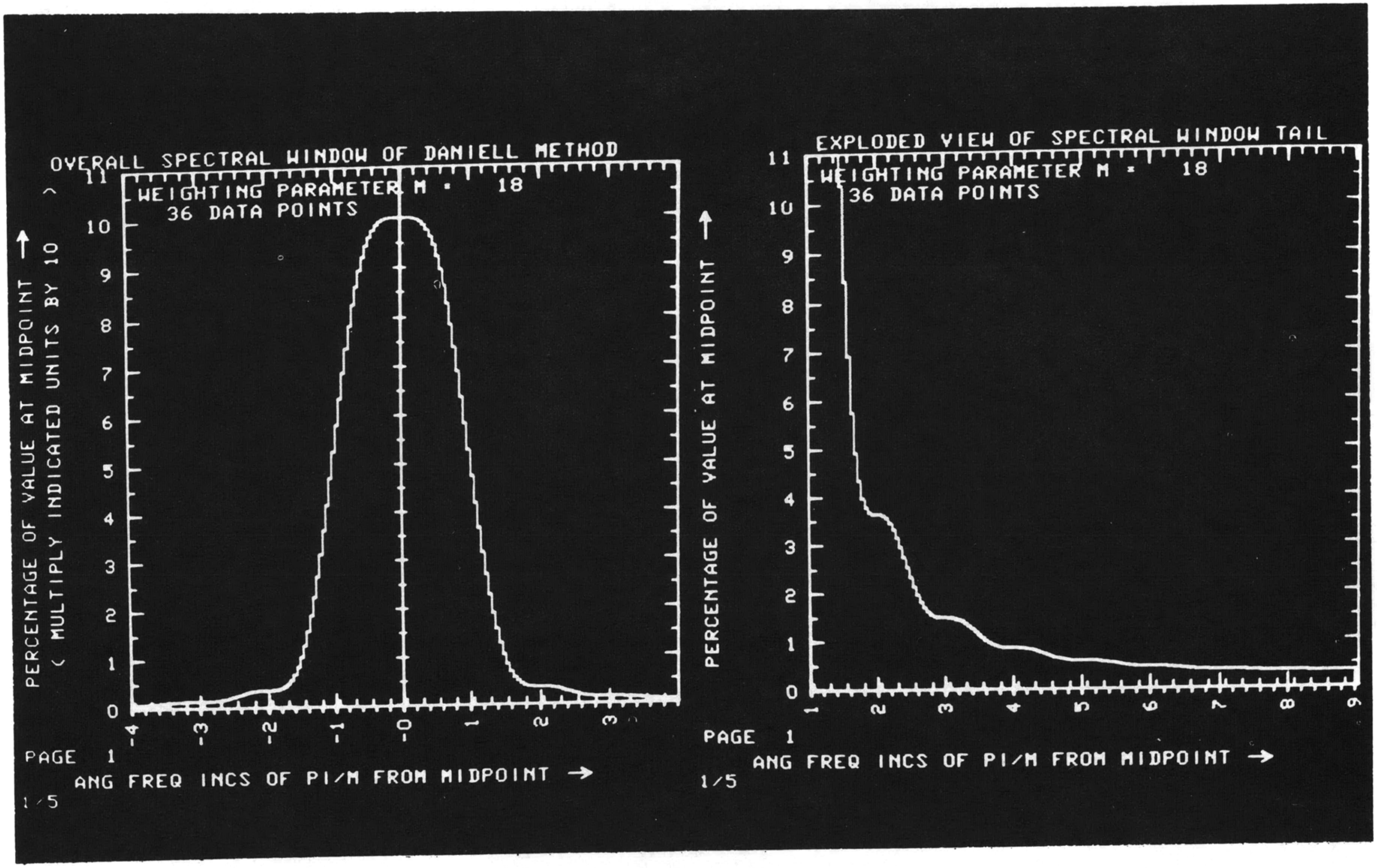
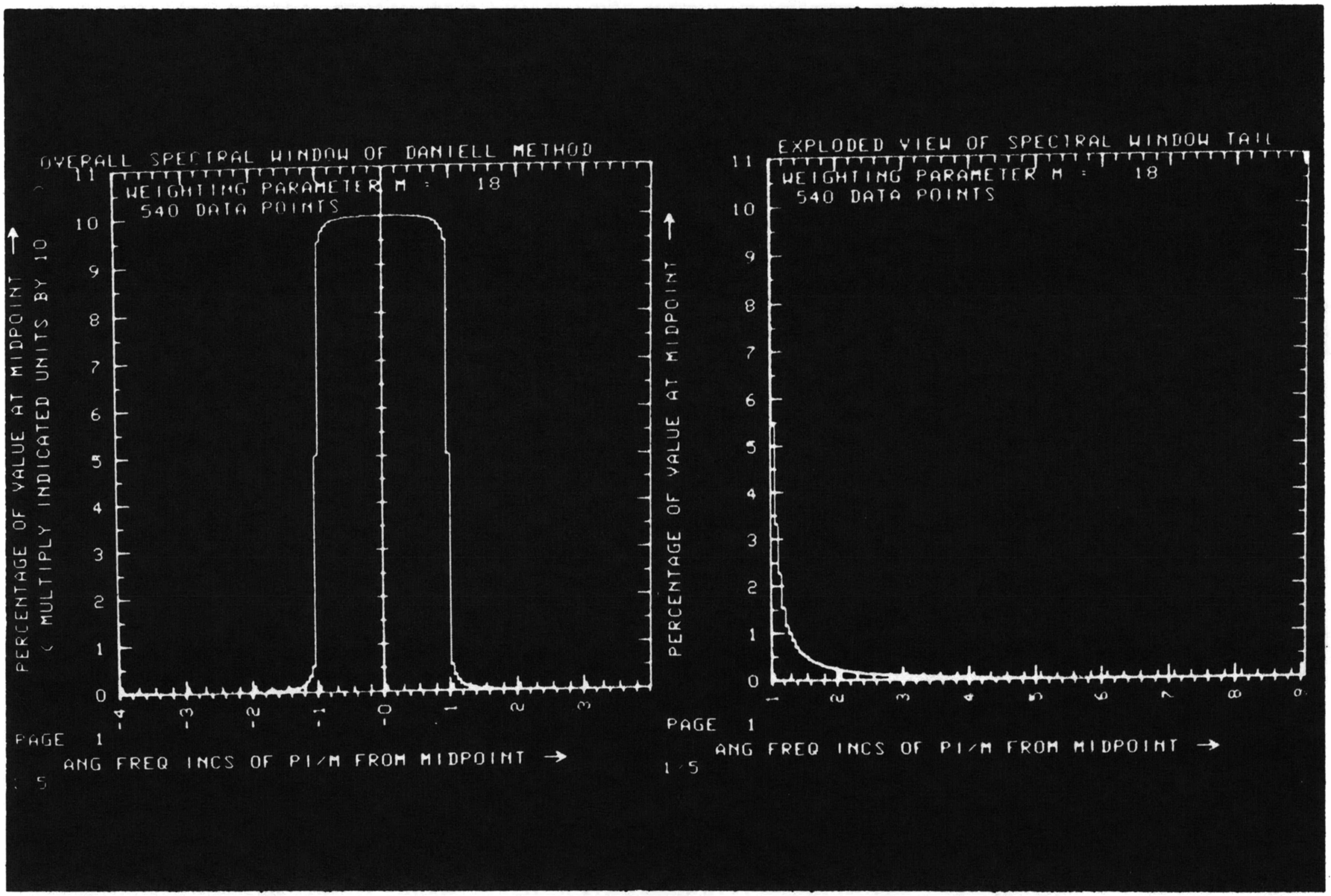

OVERALL SPECTRAL WINDOW OF DANIELL METHOD
WEIGHTING PARAMETER M = 18
36 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
(MULTIPLY INDICATED UNITS BY 10)
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT
1/5
EXPLODED VIEW OF SPECTRAL WINDOW TAIL
WEIGHTING PARAMETER M = 18
36 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT
1/5

Figure 1.3.2



OVERALL SPECTRAL WINDOW OF DANIELL METHOD
WEIGHTING PARAMETER M = 18
540 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
(MULTIPLY INDICATED UNITS BY 10
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
EXPLODED VIEW OF SPECTRAL WINDOW TAIL
WEIGHTING PARAMETER M = 18
540 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
1/5

Figure 1.3.3

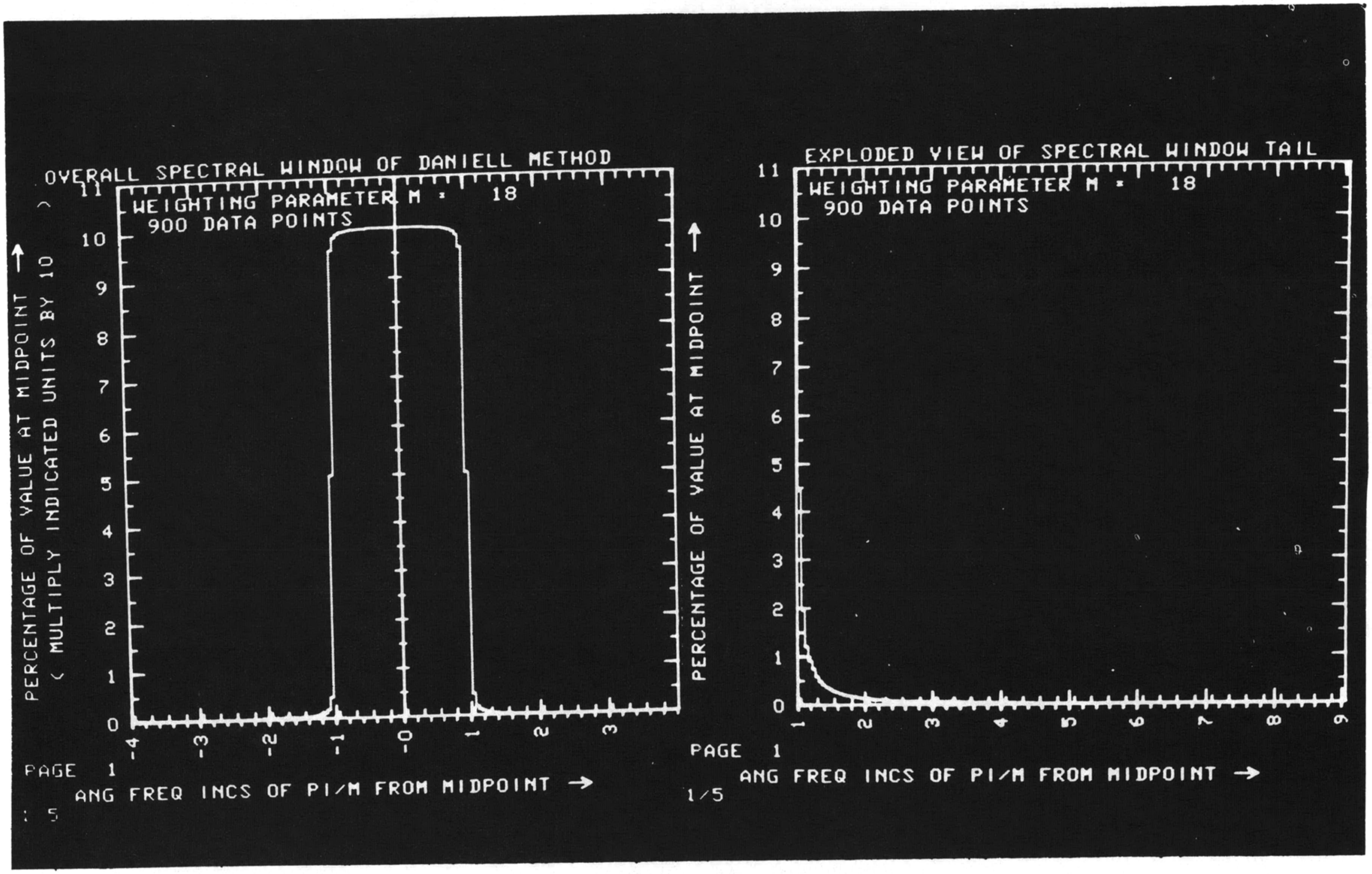
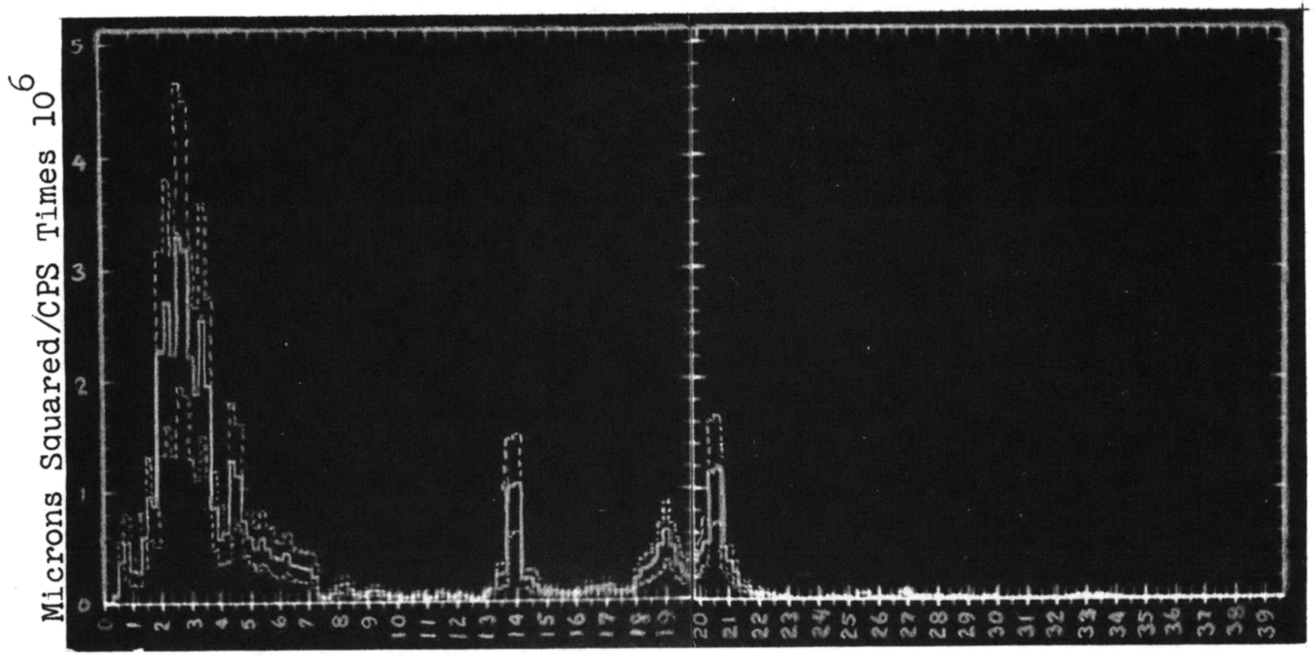

OVERALL SPECTRAL WINDOW OF DANIELL METHOD
WEIGHTING PARAMETER M = 18
900 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT →
(MULTIPLY INDICATED UNITS BY 10)
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
EXPLODED VIEW OF SPECTRAL WINDOW TAIL
WEIGHTING PARAMETER M = 18
900 DATA POINTS
PERCENTAGE OF VALUE AT MIDPOINT →
PAGE 1
ANG FREQ INCS OF PI/M FROM MIDPOINT →
1/5

Figure 1.3.4

POWER DENSITY SPECTRUM OF RECORD 1000



Cycles Per Second Times 10

Figure 1.3.5

Spectrum of Record 1000 with standard deviation plotted above and below the spectral estimate.

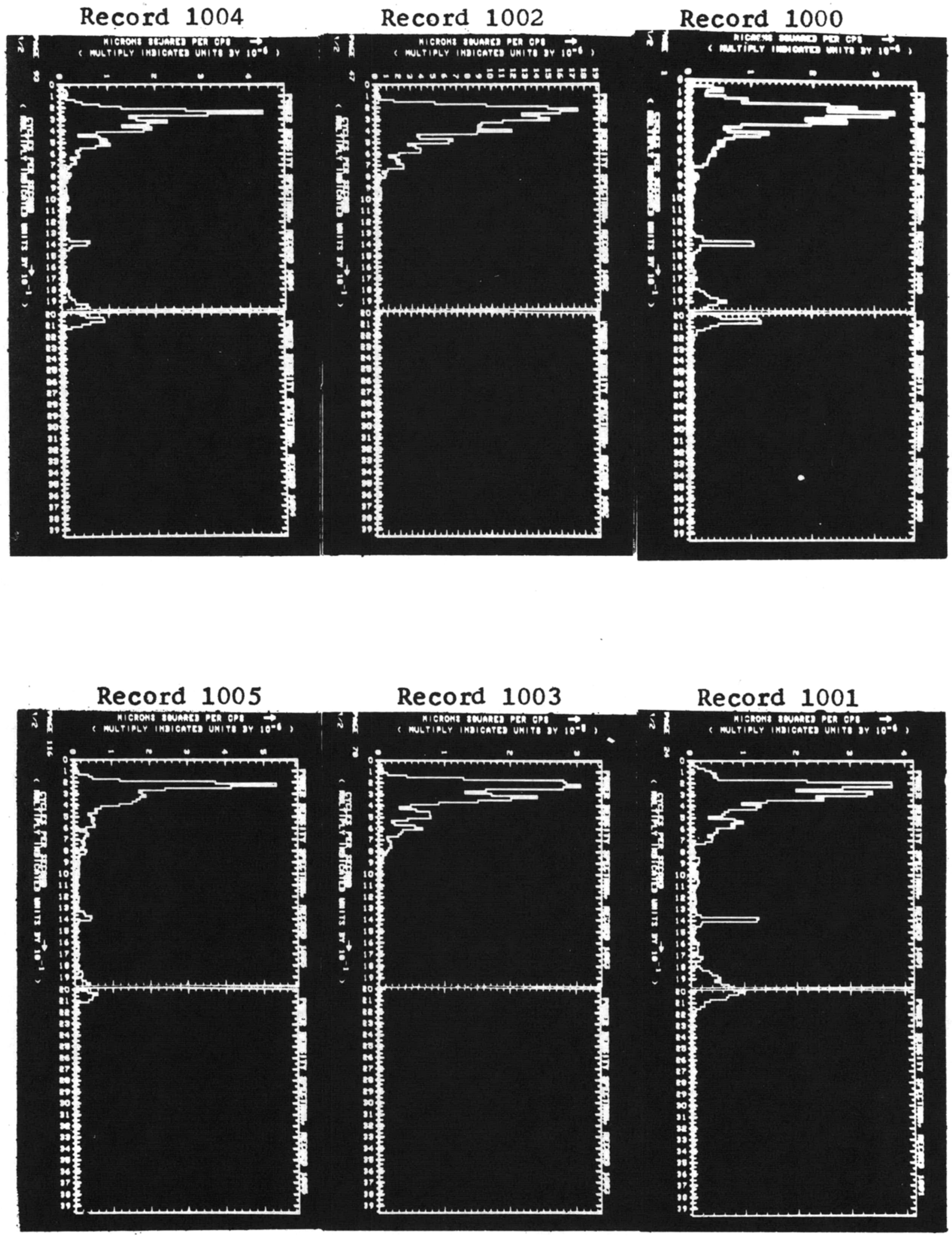


Figure 1.3.6 Power Density Spectra of Records 1000 to 1005

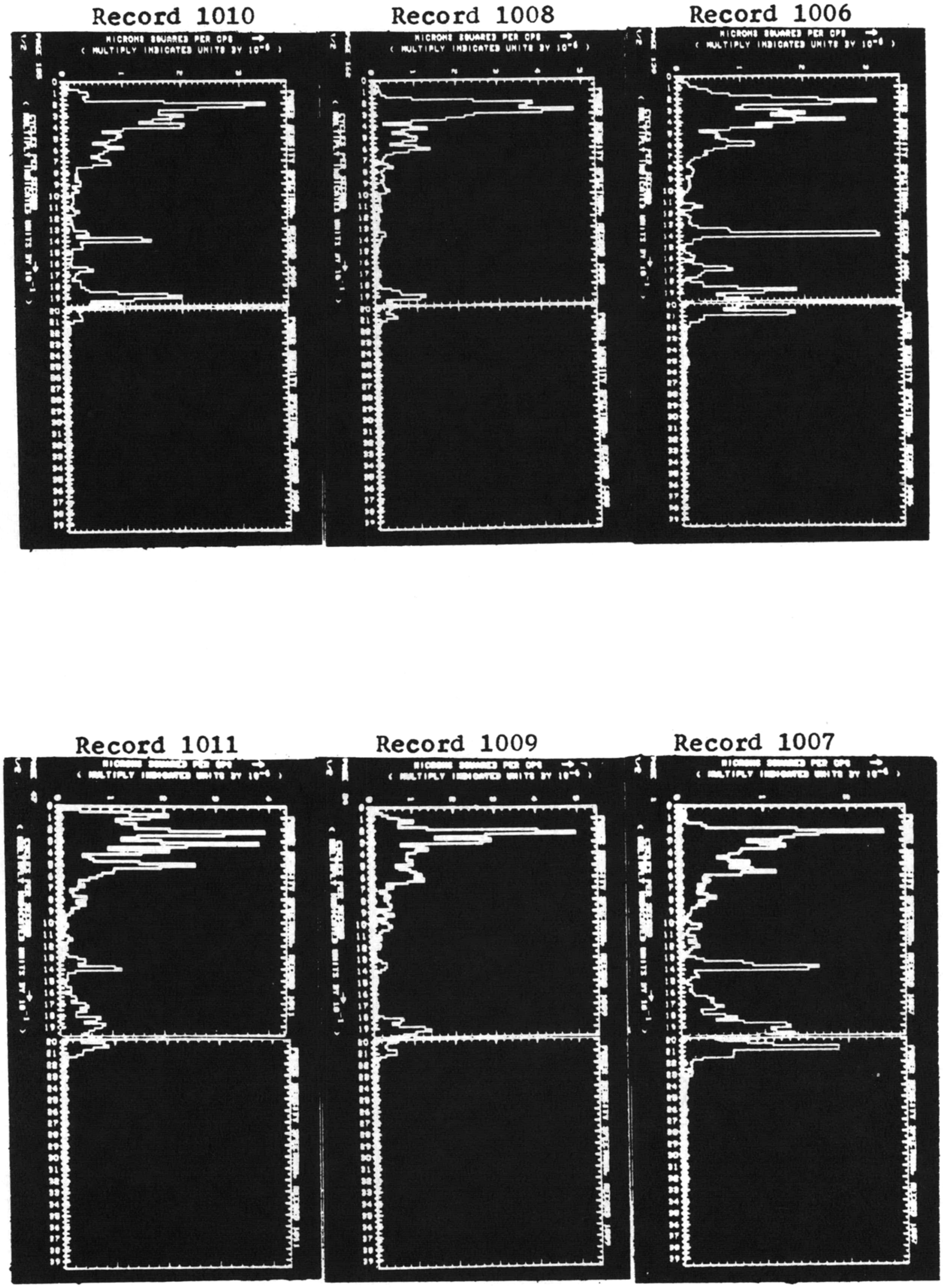


Figure 1.3.7 Power Density Spectra of Records 1006 to 1011

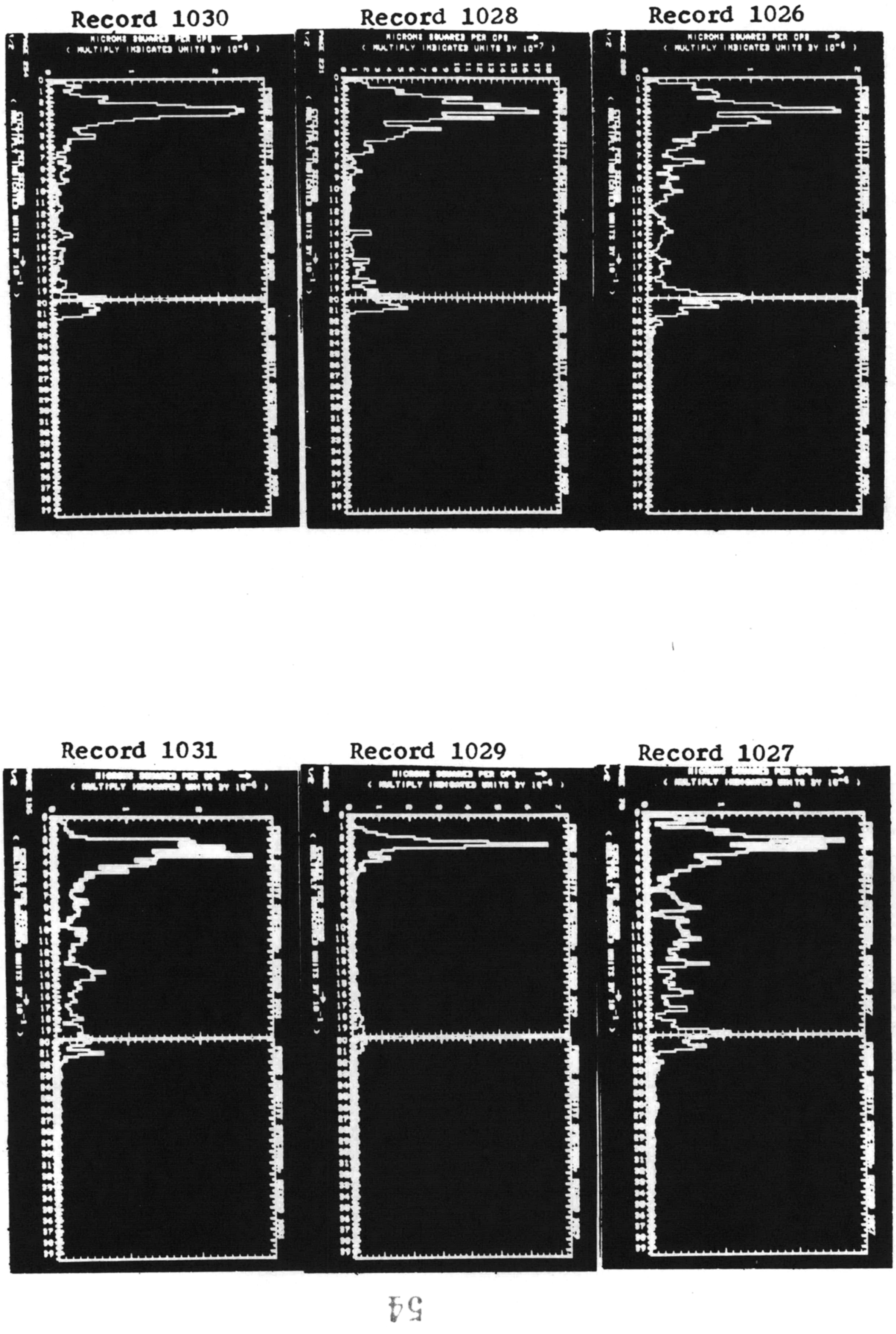


Figure 1.3.8 Power Density Spectra of Records 1026 to 1031

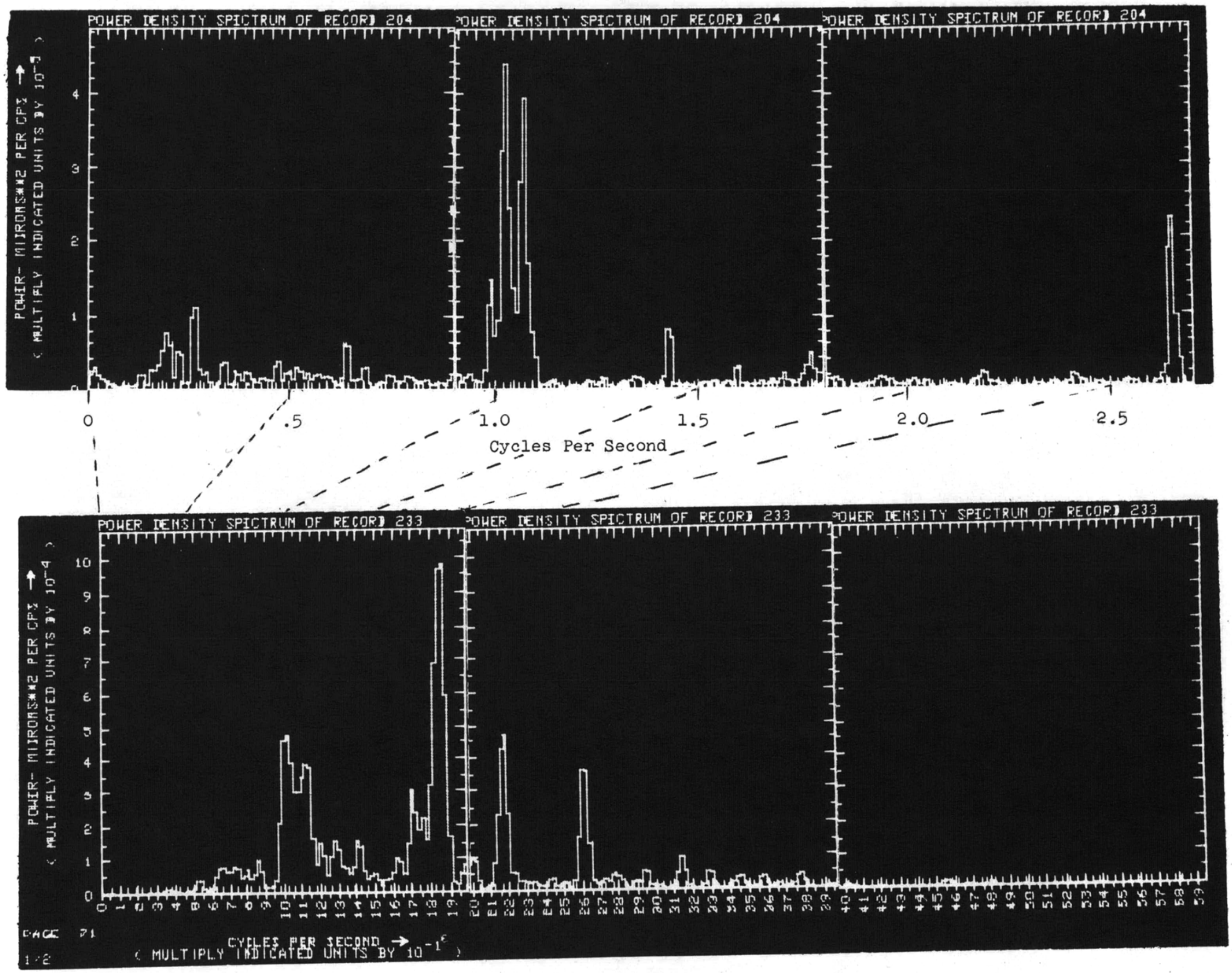


Figure 1.3.9 Power Density Spectra of Records 204 and 233 (CHP 4 and CHP 31).
(Note: The spectra have different frequency scales.)

1.4 Mathematical Generating Model for Microseisms

Stationary Time Series - Moving Summation and Decomposition

We have seen that microseismic noise can be considered at least as a wide sense stationary time series. With an additional assumption of an absolutely continuous spectral distribution (Doob, 1953) we can consider that the time series is generated by a moving average or moving summation which is written as a convolution. That is, the time series x_t can be generated by convolution of an uncorrelated or purely random series, ξ_t, with a weighting function W_i.

$$X_t = \sum_{i=-\infty}^{\infty} W_i \, \xi_{t-i}$$

Since ξ_t is at least uncorrelated and may be purely random, it is obvious that the autocorrelation of X_t will simply be the autocorrelation of W_i. Hence the spectral properties of X_t are defined by the wavelet W_i. If the power density spectrum, $\Phi(\omega)$, of the time series or, equivalently, of W_i can be factored

$$\Phi(\omega) = B(\omega)\overline{B(\omega)}$$

and $B(\omega)$ has no poles or zeros in the lower half plane then

$$B(\omega) = \sum_{K=0}^{\infty} b_K e^{i\omega K}$$

and

$$W_K = b_K, \quad W_K = 0 \text{ for } K < 0$$

(See Appendix E, Spectrum Factorization) b_K is one sided and invertable and is called the minimum phase wavelet. The considerations

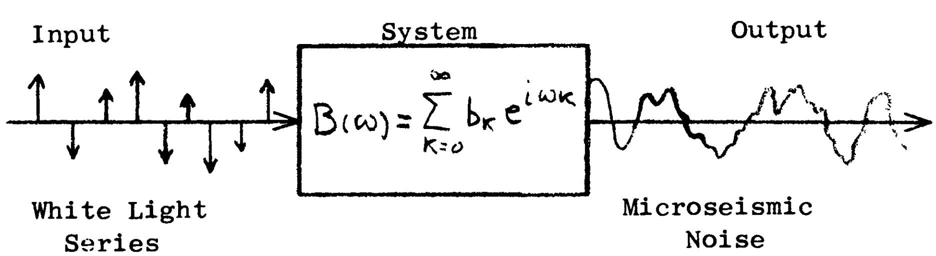
1. $\Phi(\omega) = 0$ almost nowhere

2. $\int_{-\pi}^{\pi} \log \Phi(\omega)\, d\omega > -\infty$

3. $\int_{-\pi}^{\pi} \Phi(\omega)\, d\omega < \infty$

must be met for b_K to exist (Robinson, 1956). These conditions are discussed further in Appendix E.

If we assume that the above conditions are met for microseismic noise, we can choose a simple mathematical model for microseism generation. We can consider that microseisms can be produced by passing a train of white light (uncorrelated) impulses through a system whose transfer function is $B(\omega)$. In block diagram form:



$B(\omega)$ corresponds to a realizable system since b_K is a one sided wavelet.

Spectrum factorization computations using the method of Kolmogorov as described in Appendix E have been carried out on real microseismic noise. Figures 1.3.6 to 1.3.8 show the spectra and Figures 1.4.1 to 1.4.5 show some of the minimum phase wavelets and inverse minimum phase wavelets for several of the Logan and Blanca noise records.

Autoregression, Probability Density and Edgeworth Series

Since the inverse minimum phase wavelet, a_K, exists, we can represent the noise X_t as the autoregressive process

$$\xi_t = \sum_{K=0}^{\infty} a_K X_{t-K}$$

where ξ_t is the white light series, and a_K can be found from b_K by polynomial division (See POLYDV in Appendix G).

$$A(\omega) = \sum_{K=0}^{\infty} a_K e^{i\omega K} = B^{-1}(\omega) = \frac{1}{\sum_{K=0}^{\infty} b_K e^{i\omega K}}$$

Taking the Z transform, $Z = e^{i\omega}$

$$\sum_{K=0}^{\infty} a_K Z^K = \frac{1}{\sum_{K=0}^{\infty} b_K Z^K}$$

Hence the white light series ξ_t for the process can be found by convolution of a_K with X_t. This computation has been done for most of the Logan and Blanca noise records and statistical tests have been made on the resulting white light series, ξ_t. The probability density of ξ_t for these records has been compared to the normal density using the steps outlined in Appendix B. In most cases the comparison measure resulted in the probability of exceeding chi-squared being so small that it was very unlikely the density of ξ_t was exactly normal. The numerical results summerized in Table 1.4.1 show that only four of the records pass the χ^2 test. The empirical densities, however, look so very nearly Gaussian (see Figures 1.4.6 to 1.4.12) that it seems likely that they can be expressed in terms of the Gaussian density with only small correction terms. (Note that we use the terms "Gaussian" and "normal" interchangeably throughout this section. Cramer (1951) gives the Edgeworth series expansion for the probability density $f(x)$

$$f(x) = C_0 \varphi(x) + \frac{C_1}{1!} \varphi^{(1)}(x) + \frac{C_2}{2!} \varphi^{(2)}(x) + \dots + \frac{C_n}{n!} \varphi^{(n)}(x) + \dots$$

where $\varphi(x)$ is the Gaussian, $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$, and the superscripts denote differentiation. The C_n depend on the moments. The details of the applicability of the expansion and the computation of the moments and the C_n appear in Appendix C. The first seven C's, C_0 to C_6 have been computed and the corresponding densities have been compared with the empirical density using the chi-squared measure of goodness of fit.

Normality - Chi-Squared Test

Table 1.4.2 shows the results of the Chi-squared test of the comparison of the probability density of the white light series with the normal density and the higher approximations given by the Edgeworth series. The method of computation of the Chi squared value used here differs somewhat from the method mentioned in Appendix B. In Appendix B we ignore the fact that the series undergoing the test is bounded and, after dividing up the normal density into N regions of equal area (probability), we count the number of data points which fall into each region. The approximation involving the terms in the Edgeworth series, including the normal approximations were compared directly to the empirical density, computed for r subregions over the interval in which the data fell. There was not attempt at division into regions of equal probability. For this case, where the chi squared value is computed directly from the probabilities, chi squared is

$$\chi^2 = \sum_{i=1}^{r} \frac{(P_{A_i} - P_{E_i})^2}{P_{A_i}} N$$

where P_{A_i} is the probability that a value falls in the ith range using the approximation given by the Edgeworth series, P_{E_i} is the empirical probability density for the same range, N is the number of data points which were used to compute the empirical density, and r is the number of sub-regions used in forming the empirical density. There may be some bias in this method of computation if P_{A_i} and P_{E_i} are very small. For this reason the sub-regions are grouped together so that for every grouping the quantities $P_{A_i} N$ and $P_{E_i} N$ are both at least five. (This rule of thumb is given in Wadsworth and Bryan, 1961). The grouping will reduce the number of degrees of freedom so that it becomes

$$NDF = S - 1 - m$$

where m is the highest moment used in the Edgeworth series and S is the total number of sub-groupings. S is in general less than r. We note that this method compares the empirical density and the approximation about the normal density only over the region where the data actually exists and does not assume that the data is unbounded.

In computing P_{A_i} it was necessary to calculate at least five equally spaced points across the sub-region and integrate using Simpson's Rule. The estimate of the integral using just the center point was not accurate enough. (We note here that P_{E_i} is a probability density and thus must be normalized such that its integral is equal to one.)

We see from Table 1.4.2 that, using the above method of comparison, most of the white light series are actually Gaussian (first approximation of Edgeworth series), and all can be fitted quite well using the third approximation or less. It is not disturbing that the fit gets poorer in some cases for higher approximations, since the series used is asymptotic and may oscillate.

Figures 1.4.6 to 1.4.12 show the empirical density as a solid line histogram and the Edgeworth approximation as a dotted line. The first approximation is the normal, the second approximation involves the third moment since $C_0 = 1$, $C_1 = C_2 = 0$, the third involves up to the fourth moment, etc. We can therefore say that the probability density of is,in most cases, Gaussian.

Independence Tests

The ξ_t are necessarily uncorrelated since the convolution of X_t with has removed all the linear dependence. It is not necessary that the ξ_t series be purely random or, equivalently, independent (unless the ξ_t are normally distributed, see section 2.3). Independence tests are somewhat difficult because one has to show that the joint probability density for all ξ_t factors in order to prove independence.

$$P_{\xi_1 \xi_2 \cdots \xi_n}(x_1, x_2, \ldots, x_n) = P_{\xi_1}(x_1)\, P_{\xi_2}(x_2) \ldots P_{\xi_n}(x_n)$$

Two tests for independence have been used on the ξ_t from microseismic noise. The poker count test (Appendix D) is based on the fact that we can compute the a priori probabilities of occurrance of poker hands of various values from the assumption of independence of the series from which the hands are drawn. In this case the hands are assumed drawn from an infinite supply of integers with values 0 to 9 and hence the removal of a number does not change the probability of its occurrance. In the performance of the poker count test, the ξ_t must be integers from 0 to 9 with equal probability, so the series with nearly Gaussian density must be mapped into a series with rectangular density. This mapping will not make the series dependent if it is independent and vice versa. Proof of this statement and the steps necessary for the poker count test are given in Appendix D. We may note that the poker count test is concerned with the joint density of up to five variables. The other test, the dependence measure related to the mean square contingence test, is also treated in Appendix D. It is simply a numerical measure of the factorization of the joint density of two random variables.

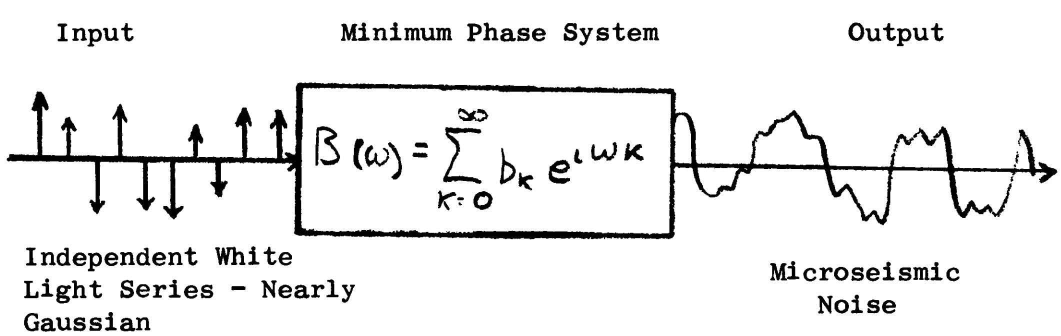
The measure, which we call the dependency, is zero is the variables are independent, and non-zero otherwise. Tests of numerical data are somewhat difficult since in almost no case will the dependency actually come out zero although it may be quite small. In order to see how small the dependency measure must be to indicate dependence, the test was run on the Rand random digits (Rand Corporation, 1955). These digits were generated by an independent process and are therefore suitable for testing purposes. A graph of the result of this test for different series lengths appears in Appendix D. For a length of 2500 the average dependency was about .0035. For dependent series such as the amplitude of the microseisms the dependency was about .25. The dependency value for the white light series, were between .0907 and .0039 and are tabulated along with the tests on the amplitudes in Table 1.4.1. Some output from the tests is shown in Figures 1.4.13 and 1.4.15. In some cases the dependency value was as low as that of the Rand digits and in others it was somewhat higher but not orders of magnitude higher. The figures mentioned above also show the results of the poker count test. In most cases a chi-squared comparison of the results is in the .1 or .05 acceptance region. The poker count test was also run on the Rand random digits. For these the chi-squared value was quite low and well within the accptance region.

Mathematical Model

The independence tests performed on are certainly not exhaustive since the poker test treats up to fifth joint density and the mean square contingency treats only the second joint density. The results are surprisingly good, however, particularly when we consider the error in the computation of the ξ_t series introduced by the spectral estimation procedure, spectrum factorization, polynomial division and convolution. It is therefore claimed that the ξ_t series is essentially independent and the microseism generating model is now an independent white light series into a minimum phase system. '

A purely random series ξ_t is ergodic and stationary. Further, the process of moving summation (convolution) is ergodic (Robinson, 1956, p. 116). Ergodicity, for our purposes, means that the time averages and ensemble averages are equal with probability one (see also Section 1.2). Hence the estimation of the moments of the series by time averages for the expansion of the density in terms of the Gaussian is justified.

In summary, we have shown that microseismic noise can be considered stationary and ergodic with a nearly Gaussian probability distribution, The model for the generation is an independent white light series convolved with a minimum phase wavelet.



$$X_t = \sum_{k=0}^{\infty} b_k \xi_{t-k}$$

Generation of Artificial Microseisms

We are now in a position to generate microseismic noise artificially. The Rand random digits which are independent and equally likely were summed in groups of ten and the mean subtracted out to give, by the central limit theorem, zero mean normal variates. These variates are the Gaussian white light input to the minimum phase system. They are Gaussian because of the central limit theorm as mentioned above, and white because the independence of the variates guarentees that only the zero lag of the autocorrelation has a non-zero value and hence insures that all frequencies will be present in the same amount. The minimum phase system response, can be computed from real data by spectrum factorization (Appendix E). The artificial noise is then generated by convolution of the minimum phase wavelet with the Gaussian white light series. Figure 1.4.16 shows real and artificial microseismic noise with the same r.m.s amplitude plotted one above the other. It is difficult, if not impossible, to tell the difference between the two with the eye alone. The identification of the two traces has been deliberately omitted from the figure. The upper trace is actually the artificial noise. Since we have been able to show that microseismic noise can be decomposed into a white light series and a wavelet, and that the white light is fairly indpenedent and nearly Gaussian, our mathematical model is quite good, and thus our artificial microseisms are quite representative. In order to tell the difference between real and artificial microseisms we would have to decompose the series into a wavelet and white light and test the probability density against the normal density. If it is normal and not just "nearly" normal, the noise is artificial. It is possible to overcome this difficulty by mapping the Gaussian series into a series with a probability density representative of the real noise, but this labor does not seem justified by the slight variation of the probability density from the Gaussian.

The chief use of the generating model is in the detection simulation studies in Chapter 3. Several hours of consecutive noise are needed for these studies and only a few minutes of it is available from our records. Using the model discussed above we can generate the necessary amount of noise artificially and it will be typical of microseisms and nearly indistinguishable from them.

It is also possible to generate three component artificial noise. The bind here would appear to be in simulating the coherency between the various components. However it has been shown (Simpson et al, 1962) that one can generate pairs of white light series with controlled coherency at zero phase. A simple extension of this to three series with controlled coherencies is given in Appendix F. One can therefore specify the coherencies between pairs of the three series, generate three white light series with these coherencies, and convolve each of the series with a different wavelet to obtain three component simulated coherent microseismic noise.

TABLE 1.4.1

SUMMARY OF RESULTS OF NORMALITY AND DEPENDENCY TESTS
ON AMPLITUDE SERIES AND WHITE LIGHT SERIES.

	PROB. EXCEED. CHI SQUARE		DEPENDENCY		LENGTH OF SERIES	
RECORD	AMPLITUDE	WHITE LIGHT	AMPLITUDE	WHITE LIGHT	AMPL.	WHITE LIGHT
1000	.66435	.0000	.25336	.00976	3201	2702
1001	.01293	.0000	.26546	.00935	3201	2702
1002	.0000	.01522	.47489	.03863	3401	2902
1003	.0000	.00305	.50919	.05031	3401	2902
1004	.28699	.0000	.28226	.01525	3321	2822
1005	.21316	.00004	.30931	.01378	3321	2822
1006	.01426	.09632	.22233	.00820	3181	2682
1007	.00289	.32880	.20035	.00397	3181	2682
1008	.0000	.00004	.27856	.00830	3361	2862
1009	.0000	.01919	.28603	.01051	3351	2852
1010	.0000	.00350	.24385	.01144	3321	2822
1011	.00113	.00048	.27526	.00731	3321	2822
1026	.00015	.0000	.25891	.00483	3581	3082
1027	.0000	.0000	.25699	.00677	3581	3082
1028	.00051	.0000	.24425	.00520	3241	2742
1029	.0000	.0000	.27333	.09075	3241	2742
1030	.00252	.00197	.25838	.02333	3301	2802
1031	.12048	.0000	.24759	.00618	3301	2802

PROBABILITY OF EXCEEDING CHI SQUARE LISTED AS
.0000 IS ACTUALLY LESS THAN .000032, BUT NOT ZERO.

TABLE 1.4.2

EDGEWORTH SERIES RESULTS

RECORD	PROBABILITY OF EXCEEDING CHI-SQUARED FOR APPROXIMATION					
	ONE	TWO	THREE	FOUR	FIVE	DEGREES
1000	.00063	.44294	.99999	.99999	.0	39
1001	.0	.0	.43359	.80852	.0	37
1002	.0	.52057	.98030	.99999	.99999	46
1003	.87704	.99999	.51583	.99999	.94568	57
1004	.0	.0	.99999	.99999	.02469	52
1005	.0	.02302	.99999	.99999	.08298	53
1006	.93772	.04635	.0	.0	.0	30
1007	.23902	.95413	.99999	.99999	.99999	56
1008	.99949	.34555	.99999	.99999	.99999	59
1009	.0	.09997	.99999	.99999	.99999	54
1010	.99999	.32270	.99999	.99999	.99999	63
1011	.99999	.81863	.0	.99986	.0	44
1026	.0	.00043	.99999	.0	.0	40
1027	.99995	.0	.0	.0	.0	9
1028	.02309	.04340	.99996	.0	.0	50
1029	.28383	.0	.0	.0	.0	17
1030	.77600	.99999	.0	.0	.0	43
1031	.31825	.0	.0	.0	.0	31

DEGREES REFERS TO THE NUMBER OF DEGREES OF FREEDOM FOR THE LOWEST APPROXIMATION NUMBER FOR WHICH THE PROBABILITY OF EXCEEDING CHI-SQUARED IS GREATER THAN .01.

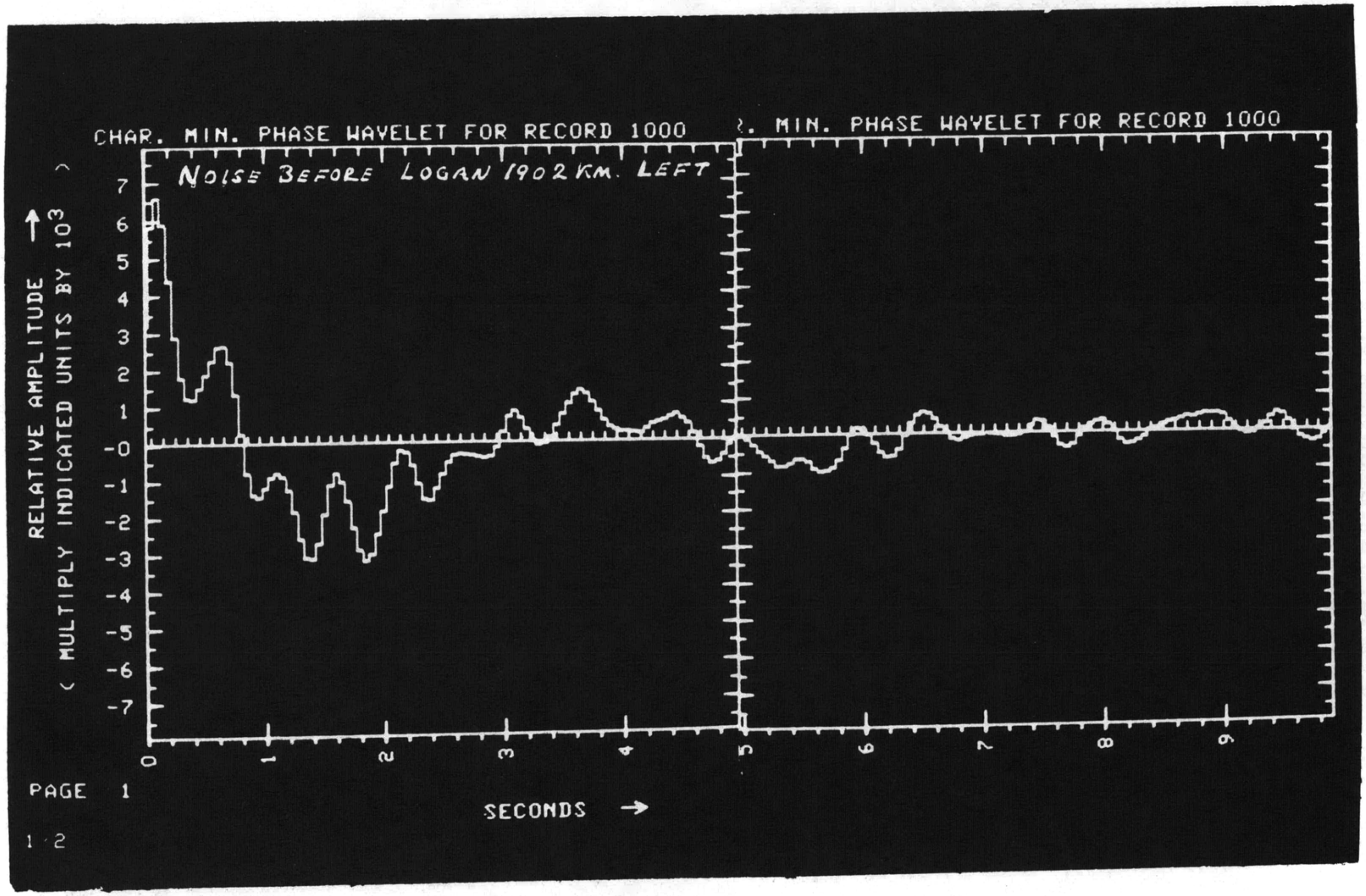

CHAR. MIN. PHASE WAVELET FOR RECORD 1000
?. MIN. PHASE WAVELET FOR RECORD 1000
NOISE BEFORE LOGAN 1902 KM. LEFT
RELATIVE AMPLITUDE
(MULTIPLY INDICATED UNITS BY 10³)
7
6
5
4
3
2
1
-0
-1
-2
-3
-4
-5
-6
-7
0
1
2
3
4
5
6
7
8
9
PAGE 1
SECONDS
1 2

Figure 1.4.1

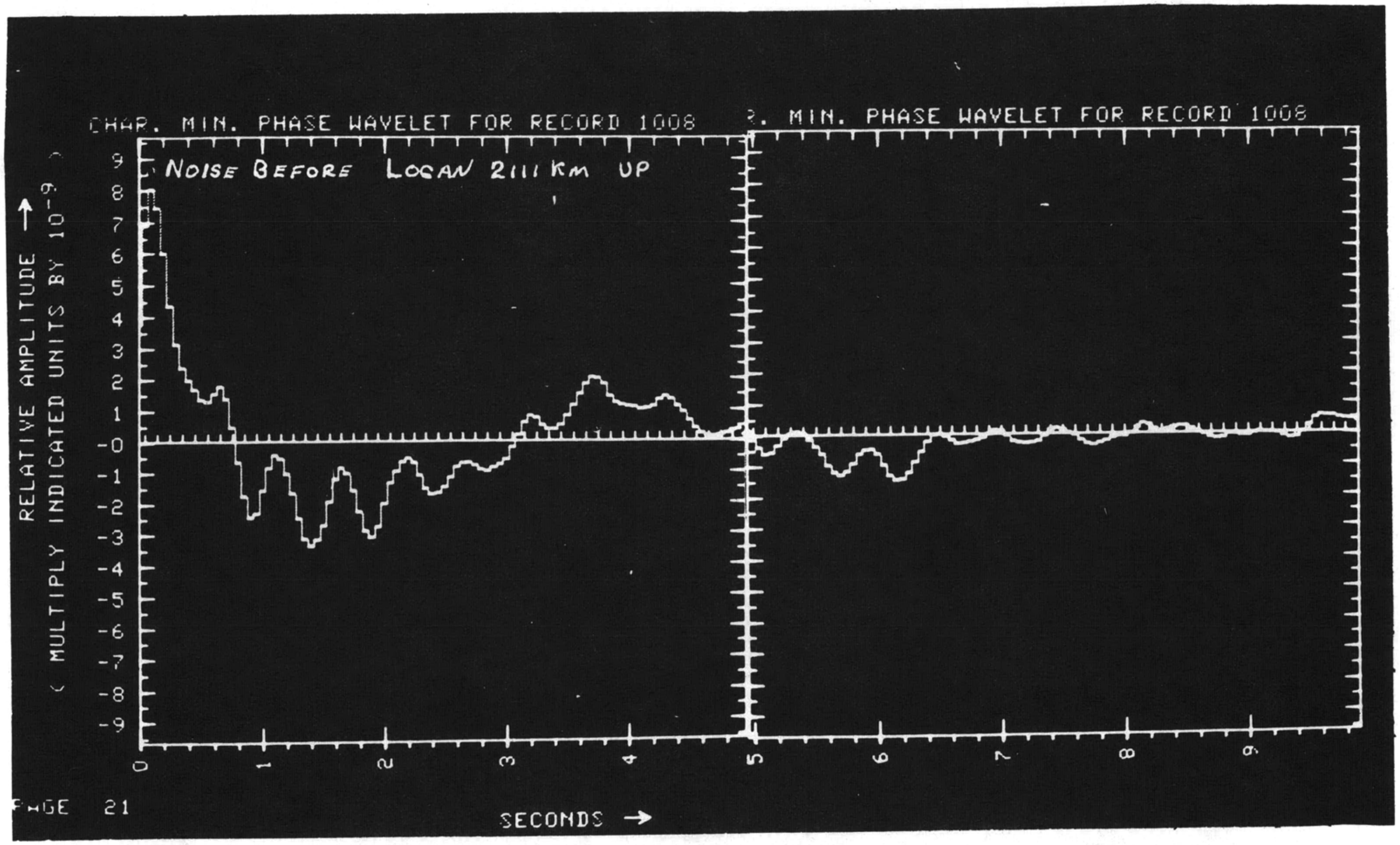

CHAR. MIN. PHASE WAVELET FOR RECORD 1008
MIN. PHASE WAVELET FOR RECORD 1008
NOISE BEFORE LOGAN 2111 KM UP
RELATIVE AMPLITUDE →
(MULTIPLY INDICATED UNITS BY 10^-9)
9 8 7 6 5 4 3 2 1 -0 -1 -2 -3 -4 -5 -6 -7 -8 -9
0 1 2 3 4 5 6 7 8 9
SECONDS →
PAGE 21

Figure 1.4.2

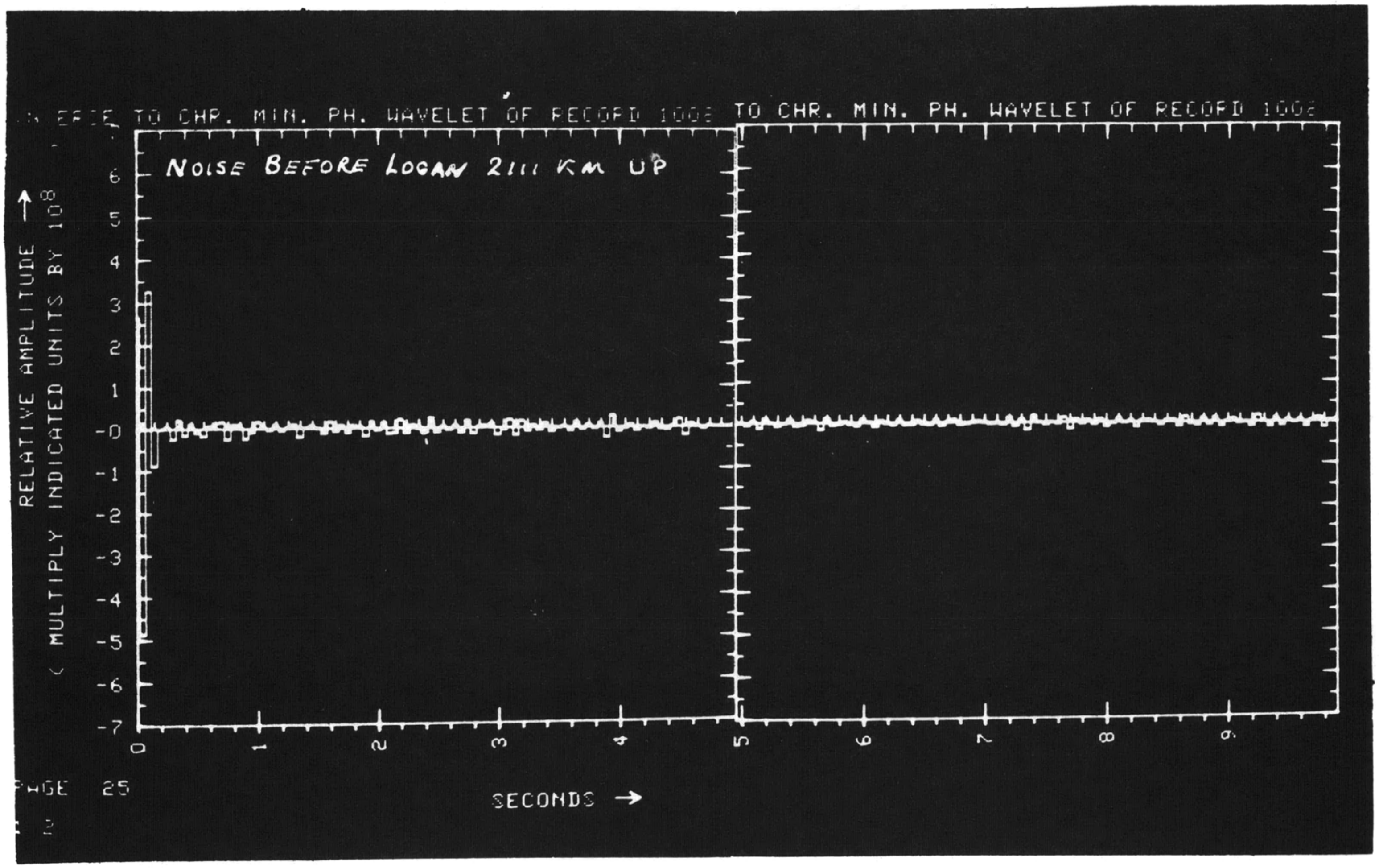

TO CHR. MIN. PH. WAVELET OF RECORD 1002
NOISE BEFORE LOGAN 2111 KM UP
RELATIVE AMPLITUDE →
(MULTIPLY INDICATED UNITS BY 10^8
SECONDS →
PAGE 25

Figure 1.4.3

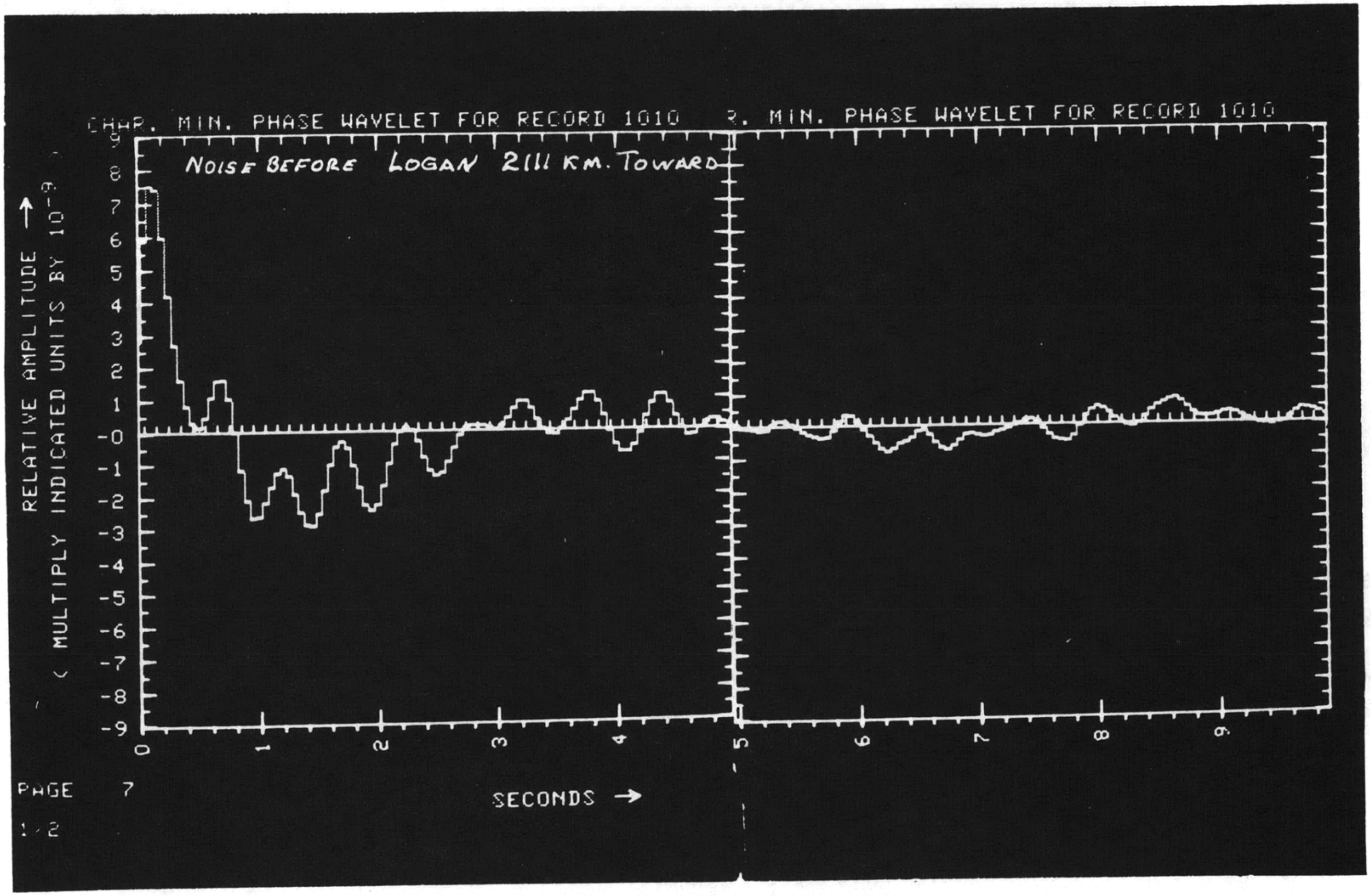
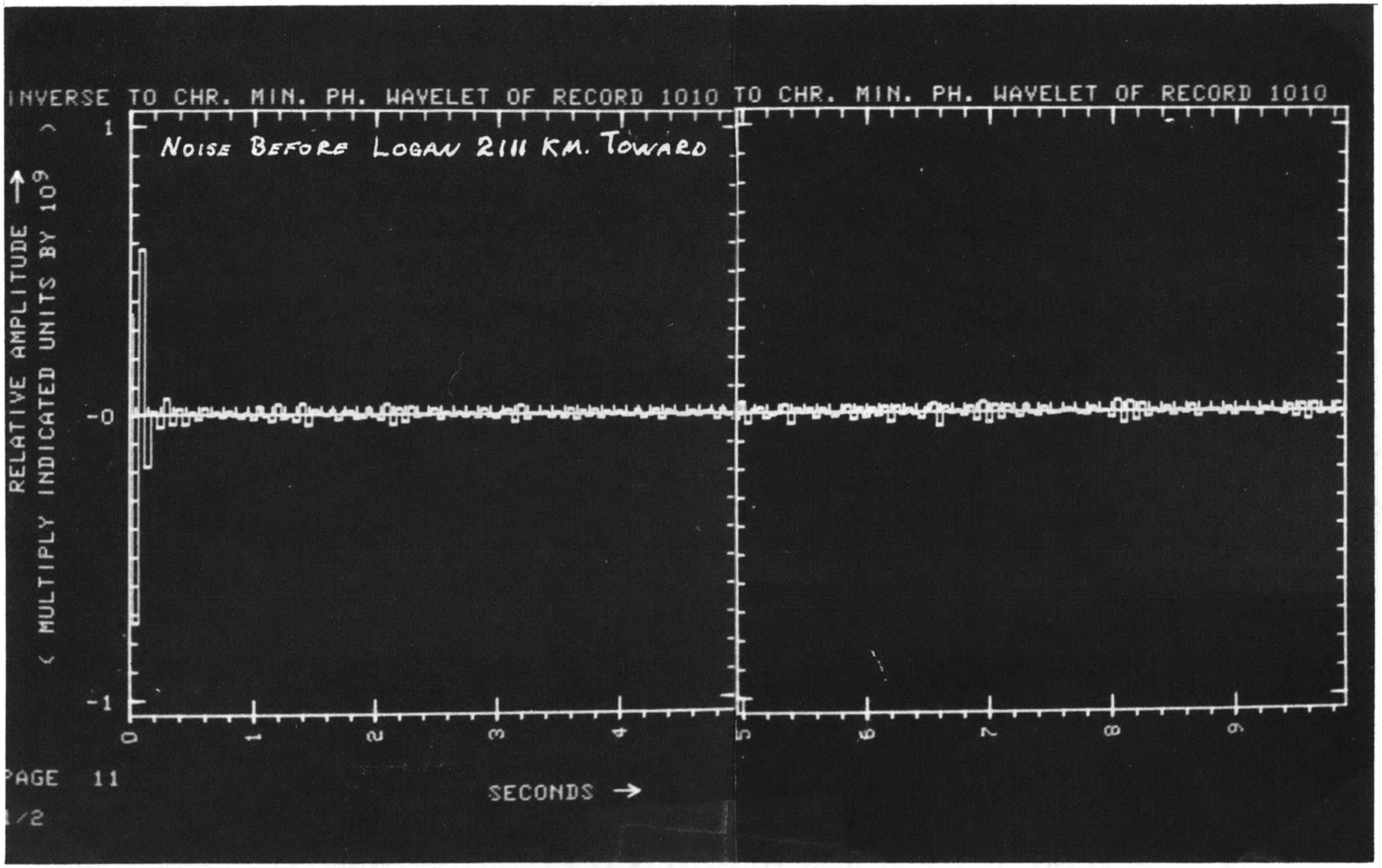

CHAR. MIN. PHASE WAVELET FOR RECORD 1010
MIN. PHASE WAVELET FOR RECORD 1010
NOISE BEFORE LOGAN 2111 KM. TOWARD
RELATIVE AMPLITUDE →
(MULTIPLY INDICATED UNITS BY 10^-9)
9 8 7 6 5 4 3 2 1 -0 -1 -2 -3 -4 -5 -6 -7 -8 -9
0 1 2 3 4 5 6 7 8 9
SECONDS →
PAGE 7
1-2

Figure 1.4.4



INVERSE TO CHR. MIN. PH. WAVELET OF RECORD 1010 TO CHR. MIN. PH. WAVELET OF RECORD 1010
NOISE BEFORE LOGAN 2111 KM. TOWARD
RELATIVE AMPLITUDE →
< MULTIPLY INDICATED UNITS BY 10⁹ >
1
-0
-1
0
1
2
3
4
5
6
7
8
9
PAGE 11
1/2
SECONDS →

Figure 1.4.5

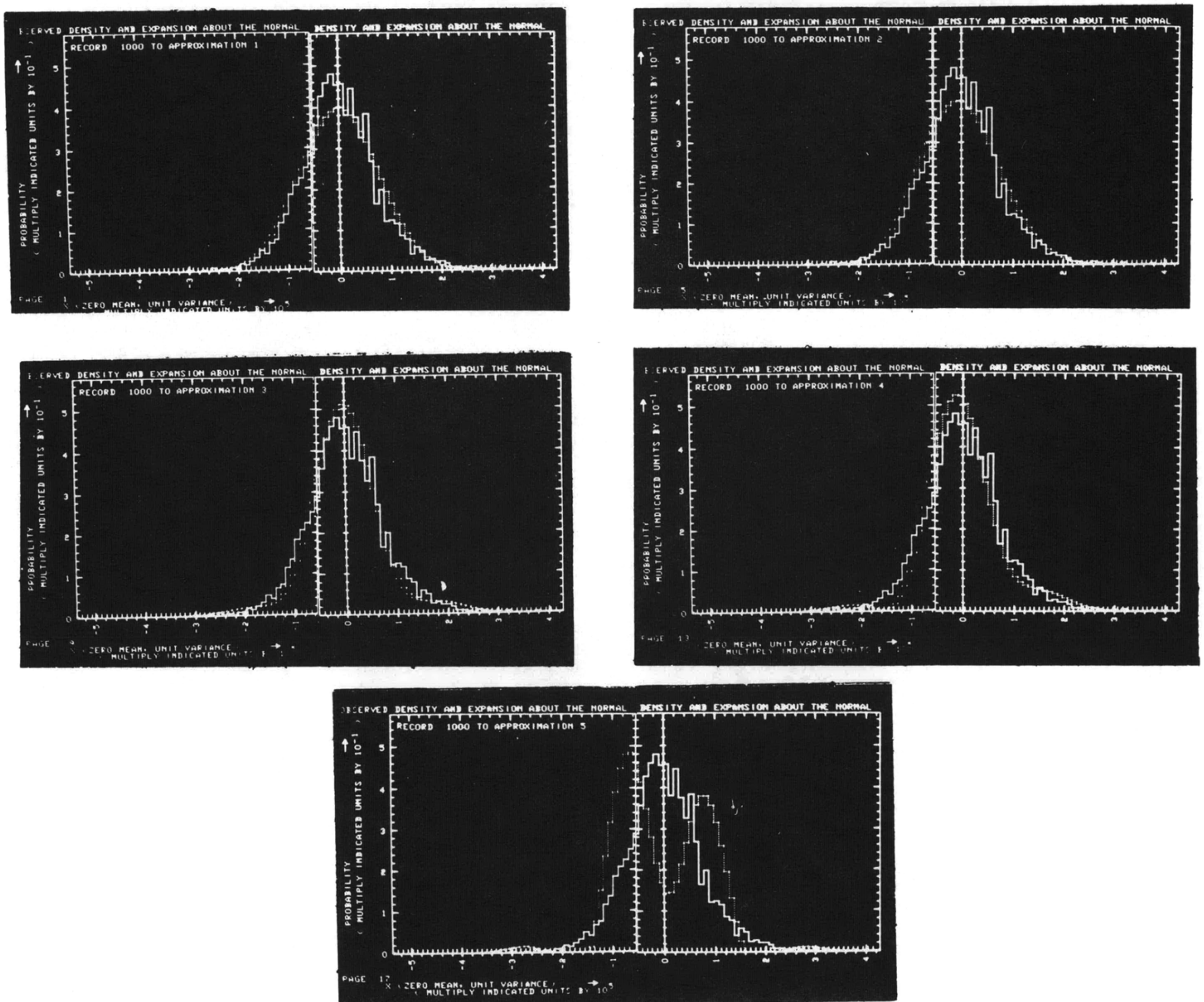


Figure 1.4.6 Empirical Probability Density of White Light Series of Record 1000 With First Five Edgeworth Series Approximations.

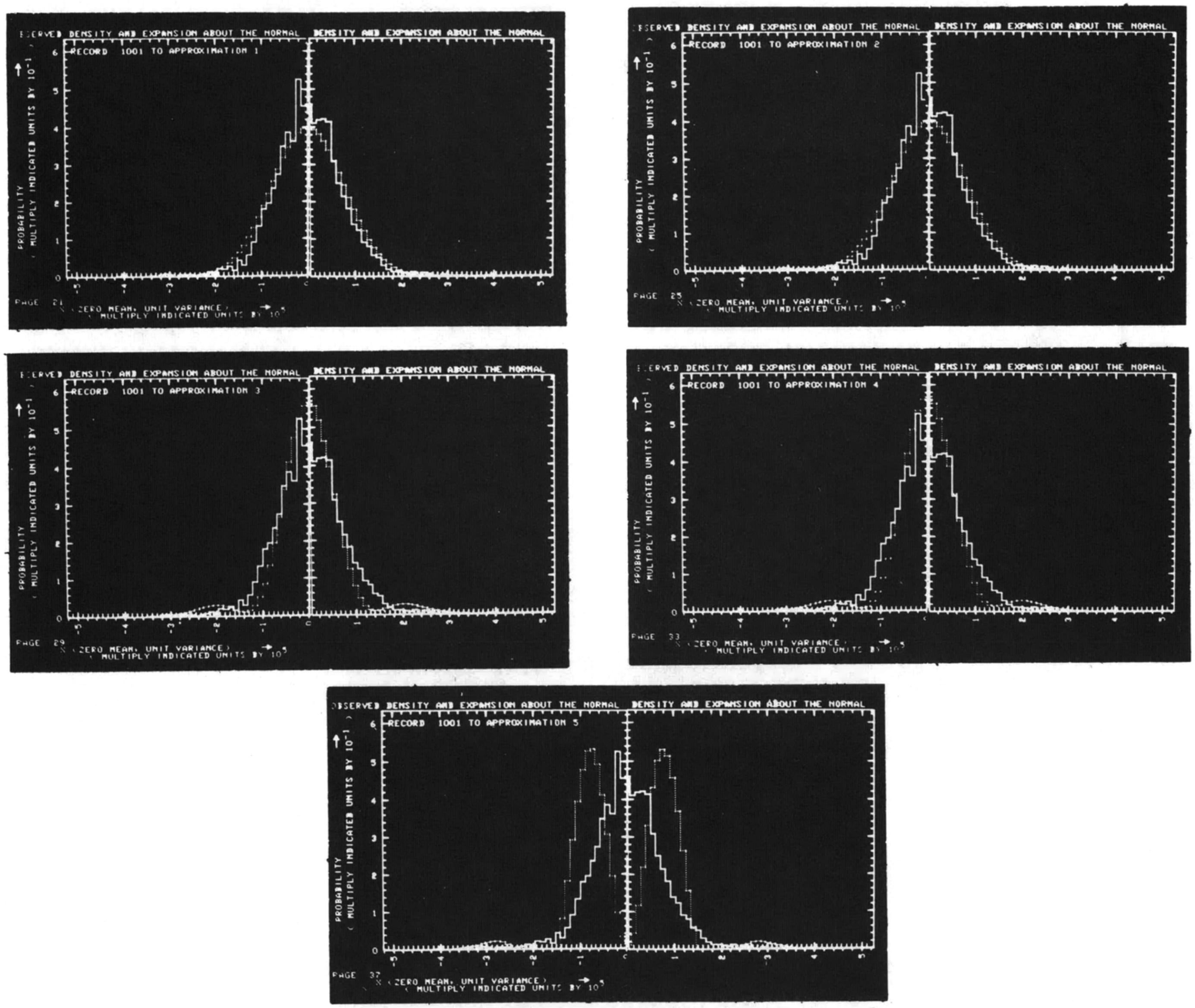


Figure 1.4.7 Empirical Probability Density of White Light Series of Record 1001 With First Five Edgeworth Series Approximations.

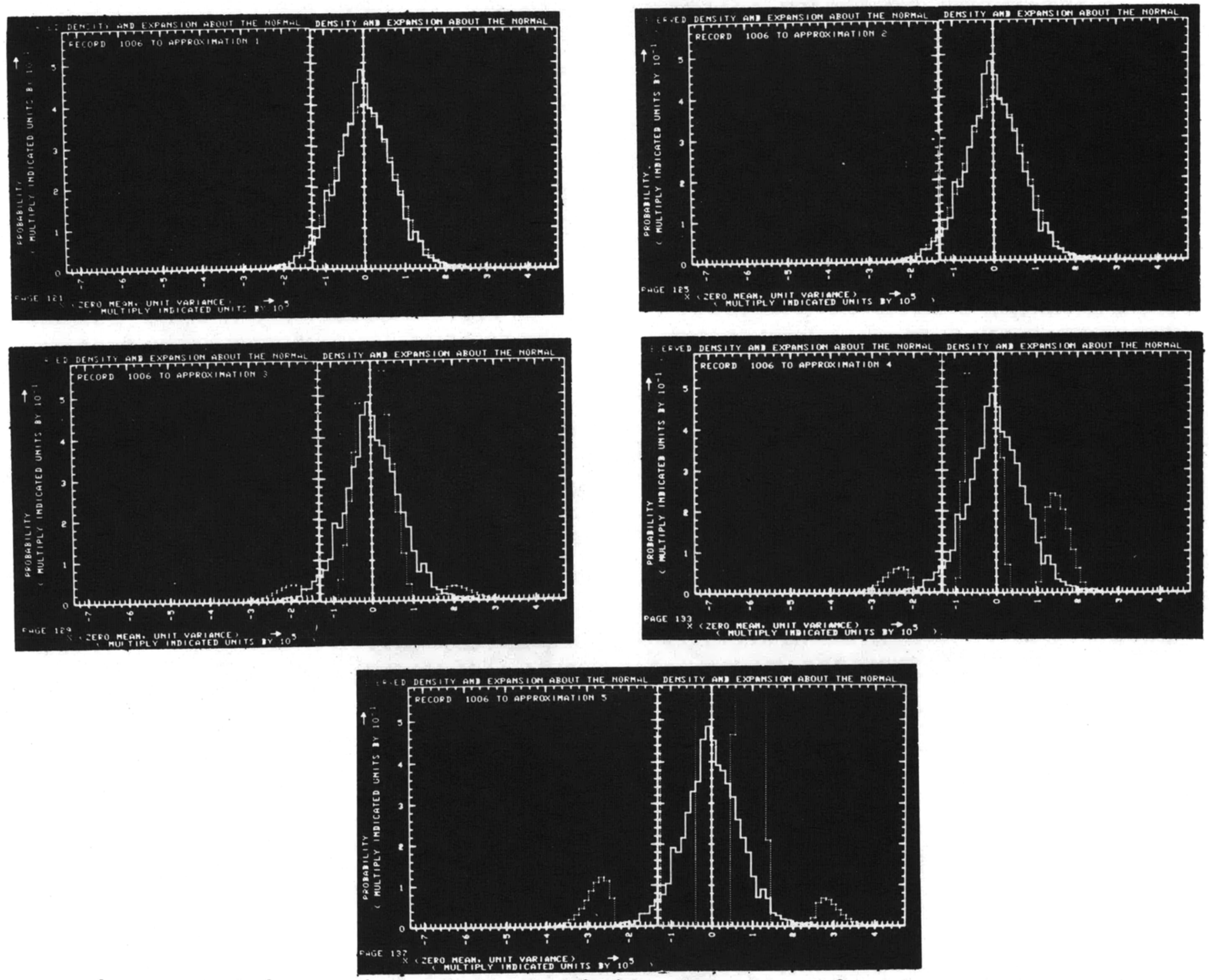


Figure 1.4.8 Empirical Probability Density of White Light Series of Record 1006 With First Five Edgeworth Series Approximations.

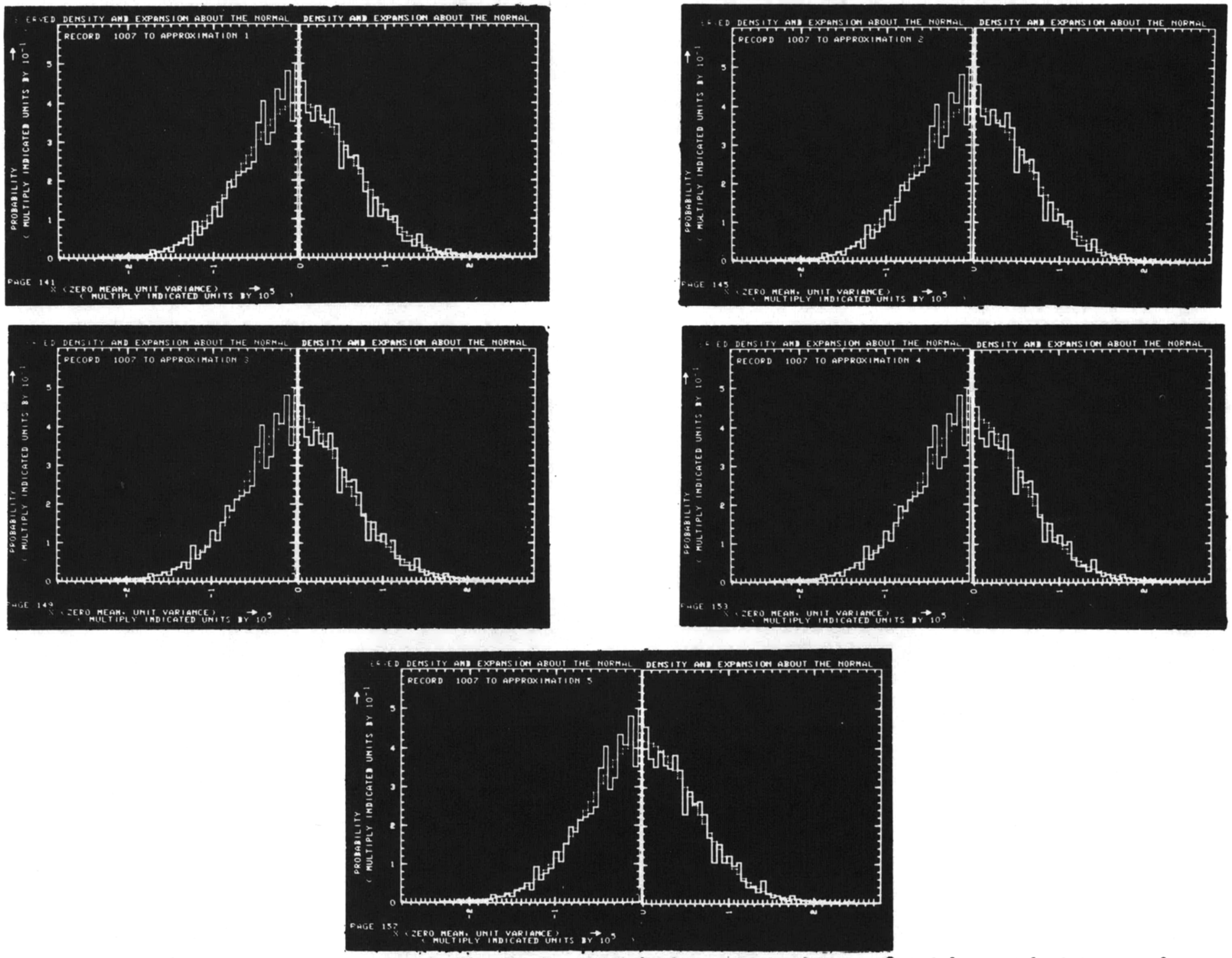


Figure 1.4.9 Empirical Probability Density of White Light Series of Record 1007 With First Five Edgeworth Series Approximations.

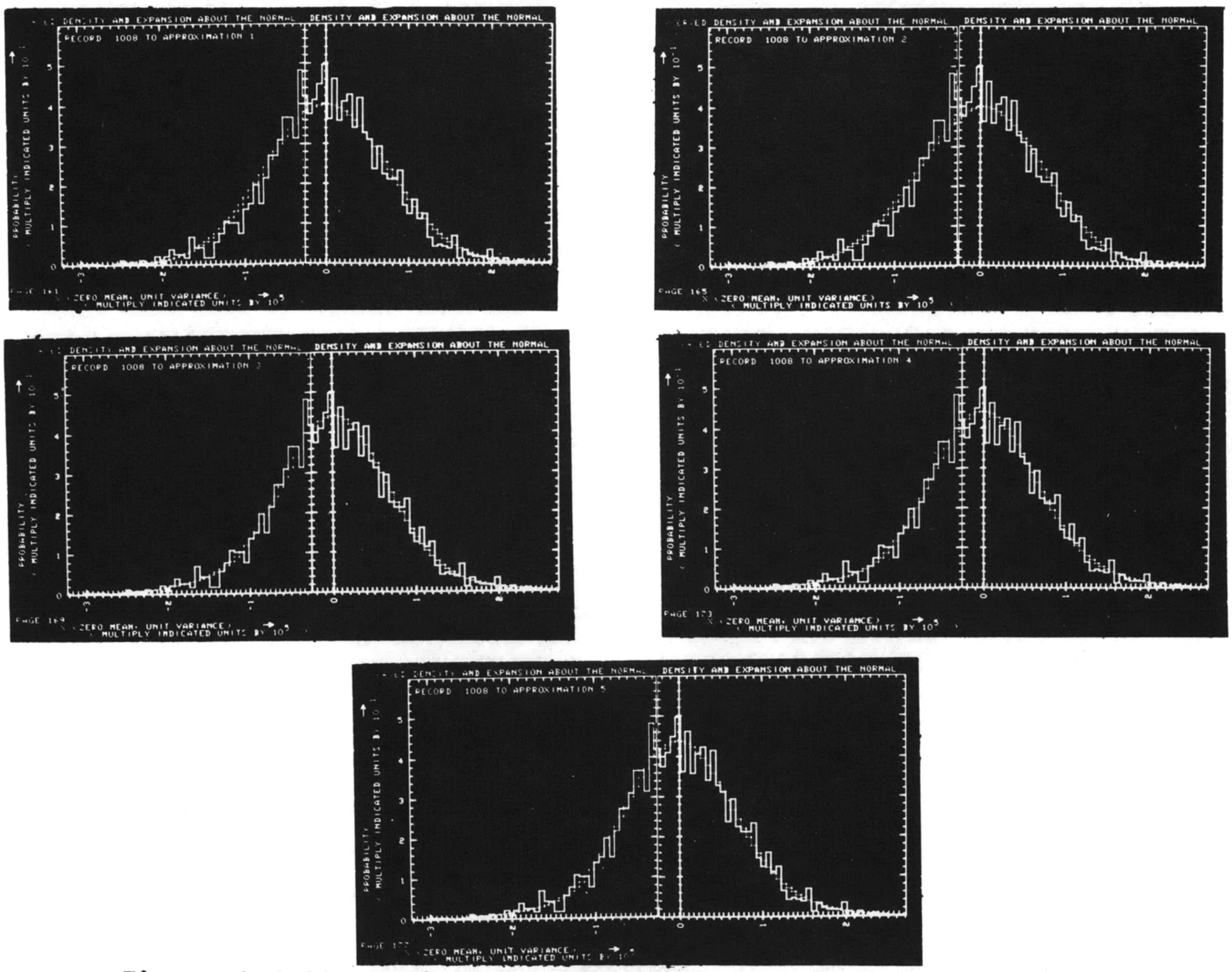


Figure 1.4.10 Empirical Probability Density of White Light Series Of Record 1008 With First Five Edgeworth Series Approximations.

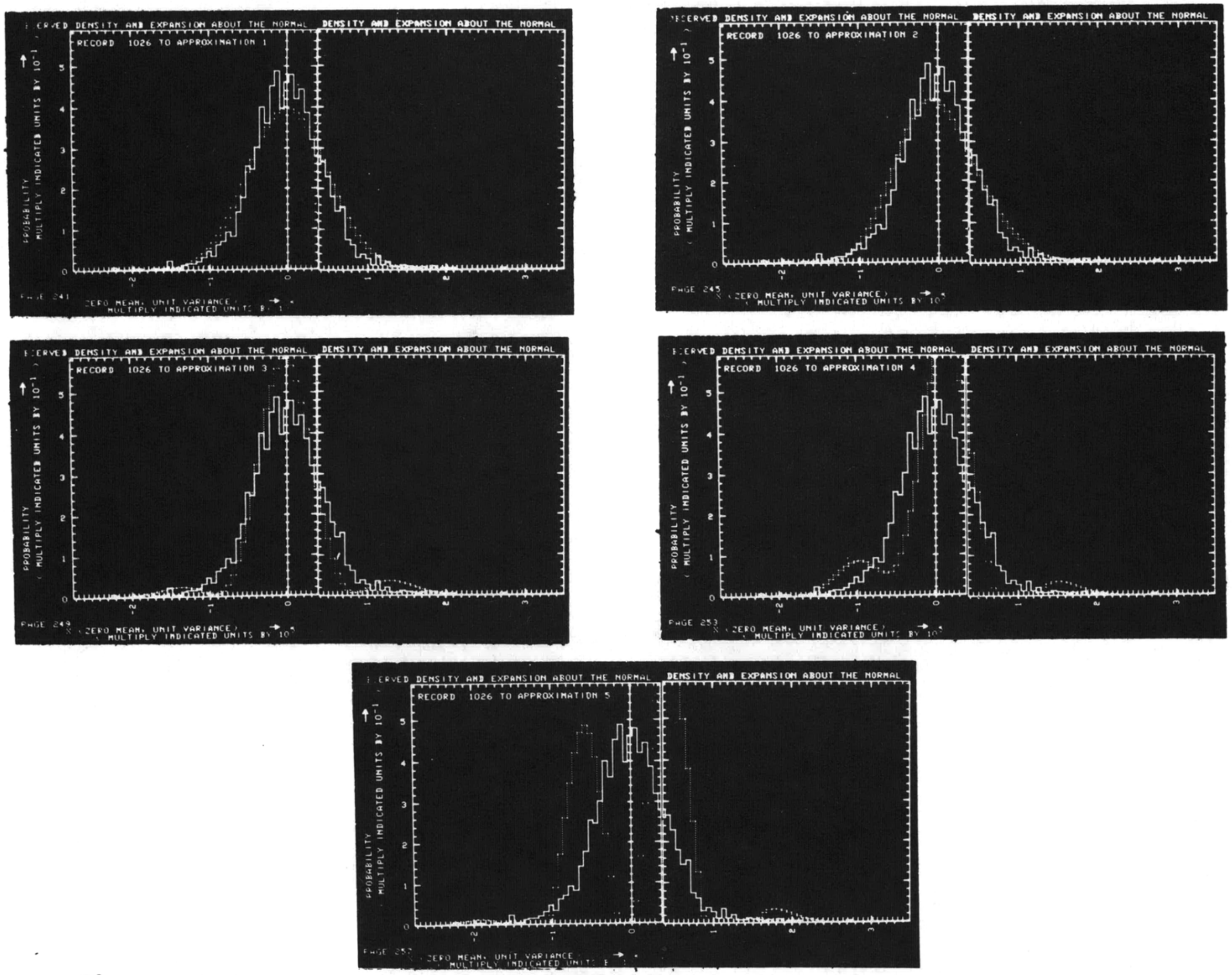


Figure 1.4.11 Empirical Probability Density of White Light Series of Record 1026 With First Five Edgeworth Series Approximations.

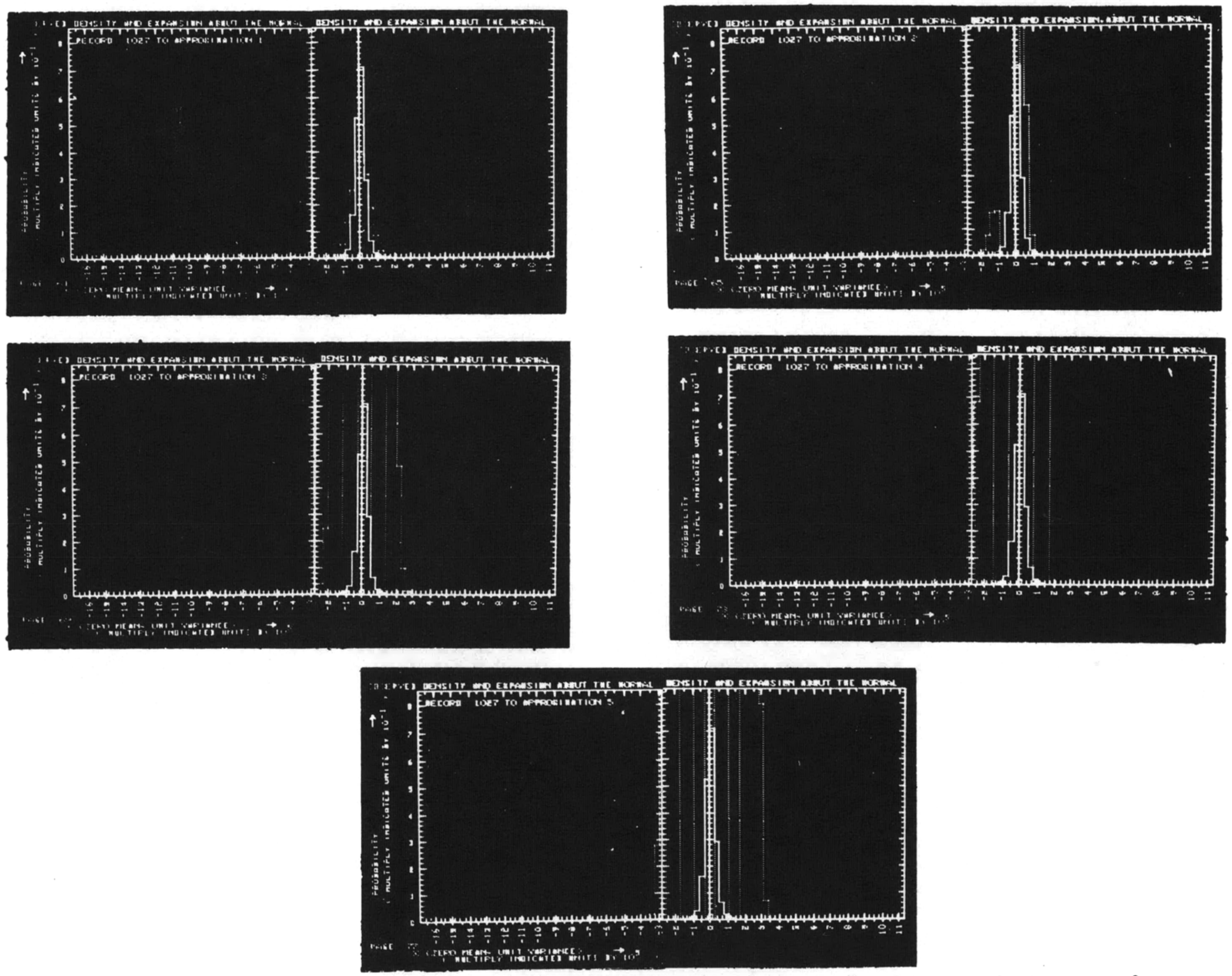


Figure 1.4.12 Empirical Probability Density of White Light Series of Record 1027 With First Five Edgeworth Series Approximations.

Figure 1.4.13

ANALYSIS OF WHITE LIGHT SERIES OBTAINED BY CONVOLVING THE INVERSE OF THE MINIMUM PHASE WAVELET OF RECORD 1000 WITH THE ORIGINAL RECORD

COMPARISON OF ACTUAL DISTRIBUTION AND NORMAL DISTRIBUTION

NUMBER OF RANGES= 51
LENGTH OF SERIES= 2702
DEGREES OF FREEDOM= 48
MEAN OF SERIES= -0.10384890E 03
STANDARD DEVIATION= 0.75864953E 05

HIGHER CENTRAL MOMENTS
THIRD MOMENT= 0.91304071E 14
FOURTH MOMENT= 0.17391028E 21
FIFTH MOMENT= -0.10809396E 25
SIXTH MOMENT= 0.17594533E 32

EXPECTED COUNT= 52.9804

CHI-SQUARE= 0.11462693E 03
PROBABILITY OF EXCEEDING CHI-SQUARE IS LESS THAN 0.00032

POKER COUNT TEST RESULTS

HAND TYPE	ACTUAL COUNT	EXPECTED COUNT
BUST	146	159.40800
1 PAIR	240	272.16000
2 PAIR	66	58.32000
3 OF A KIND	73	38.88000
FULL HOUSE	5	4.86000
STRAIGHT	7	3.88800
4 OF A KIND	3	2.43000
5 OF A KIND	0	0.05400

MEAN SQUARE CONTINGENCY= 0.88167071E-01

DEPENDENCY MEASURE= 0.97963411E-02

PROBABILITY DISTRIBUTION

NUMBER OF VALUES IN EACH OF 100 EQUALLY SPACED RANGES FROM -0.53663570E 06 TO 0.43644589E 06. 2702 VALUES IN ALL.

1.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	1.	0.	0.	1.	0.	2.
2.	2.	2.	3.	3.	6.	3.	9.	12.	18.
11.	24.	29.	37.	54.	60.	72.	80.	90.	95.
129.	145.	164.	164.	159.	148.	141.	145.	131.	119.
130.	87.	65.	68.	38.	44.	38.	30.	27.	13.
21.	17.	7.	11.	5.	12.	7.	4.	2.	1.
1.	3.	2.	1.	1.	1.	0.	2.	0.	0.
0.	0.	0.	1.	0.	0.	0.	0.	0.	1.

Figure 1.4.14

ANALYSIS OF WHITE LIGHT SERIES OBTAINED BY CONVOLVING THE INVERSE OF THE MINIMUM PHASE WAVELET OF RECORD 1006 WITH THE ORIGINAL RECORD

COMPARISON OF ACTUAL DISTRIBUTION AND NORMAL DISTRIBUTION

NUMBER OF RANGES= 51
LENGTH OF SERIES= 2682
DEGREES OF FREEDOM= 48
MEAN OF SERIES= 0.17902389E 03
STANDARD DEVIATION= 0.71888679E 05

HIGHER CENTRAL MOMENTS
THIRD MOMENT= -0.47103929E 14
FOURTH MOMENT= 0.22192675E 21
FIFTH MOMENT= -0.62127688E 26
SIXTH MOMENT= 0.67908355E 32

EXPECTED COUNT= 52.5882

CHI-SQUARE= 0.61046970E 02
PROBABILITY OF EXCEEDING CHI-SQUARE= 0.96320E-01

POKER COUNT TEST RESULTS

HAND TYPE	ACTUAL COUNT	EXPECTED COUNT
BUST	130	158.22720
1 PAIR	263	270.14399
2 PAIR	69	57.88800
3 OF A KIND	46	38.59200
FULL HOUSE	8	4.82400
STRAIGHT	13	3.85920
4 OF A KIND	7	2.41200
5 OF A KIND	0	0.05360

MEAN SQUARE CONTINGENCY= 0.73803157E-01

DEPENDENCY MEASURE= 0.82003506E-02

PROBABILITY DISTRIBUTION

NUMBER OF VALUES IN EACH OF 100 EQUALLY SPACED RANGES FROM -0.73412665E 06 TO 0.48402021E 06. 2682 VALUES IN ALL.

1.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	2.	4.	5.	2.	6.	14.	14.	22.	29.
35.	47.	87.	82.	98.	126.	149.	160.	206.	220.
205.	178.	172.	158.	143.	118.	98.	82.	55.	32.
41.	31.	15.	12.	8.	7.	2.	4.	3.	4.
2.	0.	0.	0.	0.	0.	1.	0.	0.	0.
0.	0.	0.	0.	0.	1.	0.	0.	0.	1.

Figure 1.4.15

ANALYSIS OF WHITE LIGHT SERIES OBTAINED BY CONVOLVING THE INVERSE OF THE MINIMUM PHASE WAVELET OF RECORD 1026 WITH THE ORIGINAL RECORD

COMPARISON OF ACTUAL DISTRIBUTION AND NORMAL DISTRIBUTION

NUMBER OF RANGES= 55
LENGTH OF SERIES= 3082
DEGREES OF FREEDOM= 52
MEAN OF SERIES= 0.29668643E 02
STANDARD DEVIATION= 0.49980906E 05

HIGHER CENTRAL MOMENTS
THIRD MOMENT= 0.36927477E 14
FOURTH MOMENT= 0.41691343E 20
FIFTH MOMENT= 0.39579482E 25
SIXTH MOMENT= 0.22342489E 31

EXPECTED COUNT= 56.0364

CHI-SQUARE= 0.15871704E 03
PROBABILITY OF EXCEEDING CHI-SQUARE IS LESS THAN 0.00032

POKER COUNT TEST RESULTS

HAND TYPE	ACTUAL COUNT	EXPECTED COUNT
BUST	143	181.84320
1 PAIR	307	310.46399
2 PAIR	90	66.52800
3 OF A KIND	53	44.35200
FULL HOUSE	4	5.54400
STRAIGHT	12	4.43520
4 OF A KIND	6	2.77200
5 OF A KIND	1	0.06160

MEAN SQUARE CONTINGENCY= 0.43508112E-01

DEPENDENCY MEASURE= 0.48342347E-02

PROBABILITY DISTRIBUTION

NUMBER OF VALUES IN EACH OF 100 EQUALLY SPACED RANGES FROM -0.27321346E 06 TO 0.35513622E 06. 3082 VALUES IN ALL.

1.	0.	0.	0.	0.	0.	0.	0.	3.	1.
0.	1.	0.	2.	0.	0.	2.	1.	2.	9.
2.	2.	5.	6.	6.	8.	12.	18.	13.	24.
27.	36.	32.	55.	69.	99.	96.	117.	155.	140.
175.	189.	154.	179.	186.	163.	172.	150.	124.	108.
103.	89.	70.	56.	60.	29.	24.	12.	15.	11.
5.	14.	5.	8.	3.	3.	5.	4.	3.	2.
3.	0.	4.	0.	2.	2.	2.	0.	0.	0.
0.	0.	0.	0.	0.	0.	2.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	1.	1.

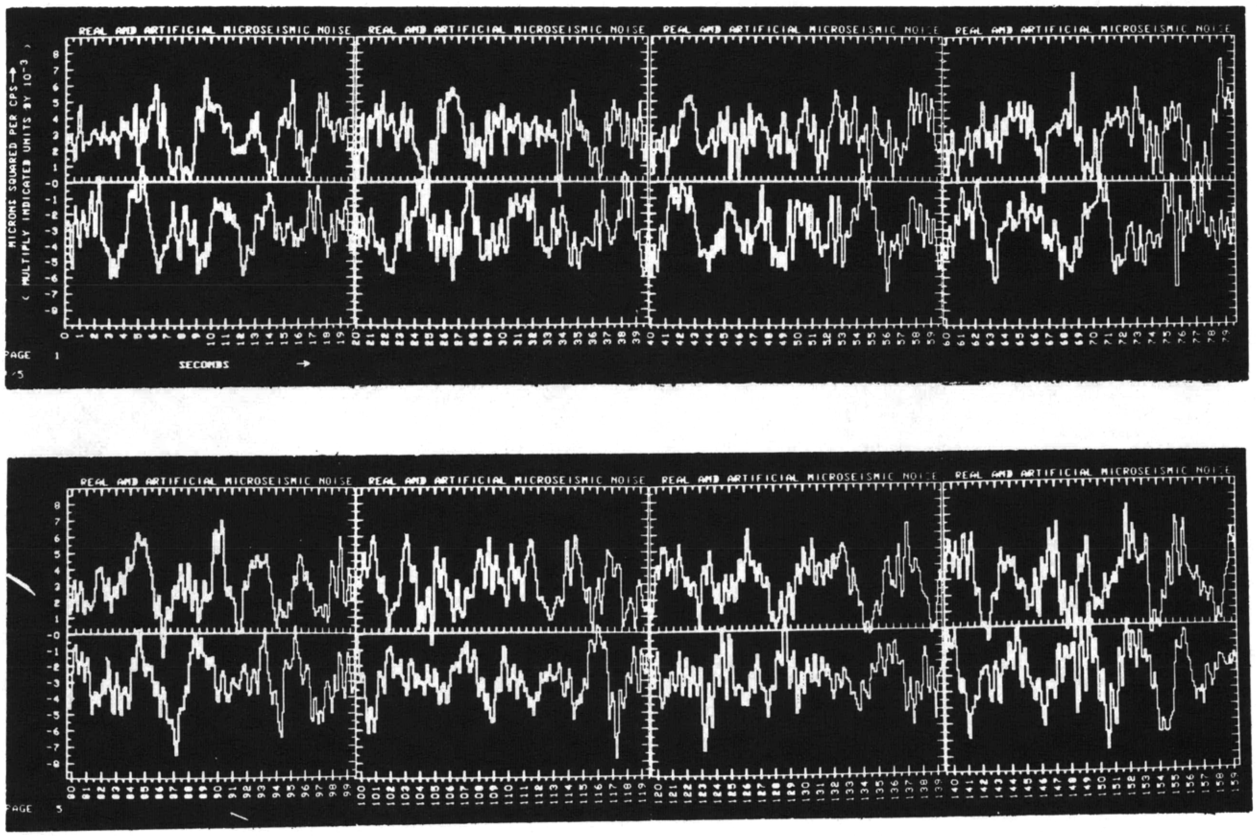


Figure 1.4.16

Real and Artificial Microseismic Noise

1.5 Cross-Series Properties

The availability of simultaneous three component seismic noise records from different stations affords opportunity for cross correlation and cross-spectral analyses. Techniques similar to those of autospectral analysis have been worked out and programmed for high speed digital computers. The major computational difference is the need for a sine transform in addition to the cosine transform since the cross correlation is not in general an even function. Knowing the sine and cosine transforms of the cross correlation it is easy to compute the magnitude cross power and phase spectra, and it is also useful to compute the coherency. The development of the usual expression for coherency can be done quickly for transients and then carried over to discrete time for our case.

Cross Correlation, Cross Power and Coherency

For two transients $x(t)$ and $y(t)$ the cross correlation is

$$\varphi_{xy}(\tau) = \int_{-\infty}^{\infty} x(t)\, y(t+\tau)\, dt$$

The cross power spectrum is then the Fourier transform

$$\Phi_{xy}(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \varphi_{xy}(\tau)\, e^{i\omega\tau}\, d\tau = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(t)\, y(t-\tau)\, e^{i\omega\tau}\, dt\, d\tau$$

with the change of variables $r = t+\tau$ this becomes

$$\Phi_{xy}(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} x(t)\, e^{-i\omega t}\, dt \int_{-\infty}^{\infty} y(r)\, e^{i\omega r}\, dr$$

hence

$$\Phi_{xy}(\omega) = 2\pi \overline{F_x(\omega)} F_y(\omega) \tag{1.5.1}$$

where $F_x(\omega)$ is the Fourier transform of $x(t)$, $F_y(\omega)$ the Fourier transform of $y(t)$, and the bar denotes complex conjugation. The auto-power spectra are found to be, by similar treatment,

$$\Phi_{xx}(\omega) = 2\pi F_x(\omega) \overline{F_x(\omega)}$$

$$\Phi_{yy}(\omega) = 2\pi F_y(\omega) \overline{F_y(\omega)}$$

The coherency is then usually defined as

$$Coh_{xy}(\omega) = \frac{|\Phi_{xy}(\omega)|}{\sqrt{\Phi_{xx}(\omega)\,\Phi_{yy}(\omega)}}$$

$$= \frac{|\overline{F_x(\omega)}\,F_y(\omega)|}{\sqrt{F_x(\omega)\overline{F_x(\omega)}\;F_y(\omega)\overline{F_y(\omega)}}} = 1$$

This definition is not particularly useful since $Coh_{xy}(\omega)$ is always one. If the cross-correlation is weighted by some function, such as the Daniell weighting function (Section 1.3), the coherency is not necessarily one and has some meaning as a measure.

We define the normalized cross power vector $N(\omega)$

$$N(\omega) = \frac{\Phi'_{xy}(\omega)}{\sqrt{\Phi_{xx}(\omega)\,\Phi_{yy}(\omega)}}$$

where $\Phi'_{xy}(\omega)$ now takes into consideration the weighting function $W(\tau)$.

$$\Phi'_{xy}(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(t)\, y(t+\tau)\, dt\, e^{i\omega\tau}\, W(\tau)\, d\tau$$

$$\Phi'_{xy}(\omega) = F_x(\omega)\,\overline{F_y(\omega)} * W(\omega)$$

where $W(\omega)$ is the Fourier transform of $W(\tau)$ and the asterisk denotes convolution. $\Phi'_{xy}(\omega)$ is in general complex, hence $N(\omega)$ is truly a vector. The coherency is then defined

$$coh_{xy}(\omega) = |N(\omega)|$$

Daniell Window and M/N Ratio

The treatment is almost identical for discrete time. The complete transient cross correlation for the two series x_t and y_t each of points is

$$\varphi_{xy}(\tau) = \frac{1}{2N-1} \sum_{t=-(N-|\tau|)}^{N-|\tau|} x_t \, y_{t+\tau} \quad , \quad \tau = 0, \pm 1, \ldots, \pm(N-1)$$

and the cross power spectrum with the Daniell weighting function is

$$\Phi'_{xy}(\omega) = \frac{1}{2\pi} \sum_{\tau=-(N-1)}^{N-1} \varphi(\tau) \left(\frac{\sin \frac{\pi\tau}{M}}{\frac{\pi\tau}{M}} \right) e^{i\omega\tau}$$

We shall take $\omega = n\omega_0$ with $\omega_0 = \pi/M$ where M is the Daniell parameter, and $n = 0, 1, 2, \ldots, M$. We have seen in Section 1.3 that, for N/M large, the Daniell window is nearly rectangular. With $\omega_0 = \pi/M$ the windows for neighboring spectral estimates $K\omega_0$ and $(K+1)\omega_0$ overlap by about 50%. The Daniell window averages the sine and cosine transforms over the window width and consequently averages the cross power vector, $N(\omega)$. We see, therefore, that $|N(\omega)|$, the coherency, is less than or equal to one. If the $N(\omega)$ vector changes direction rapidly over the band $\omega \pm \frac{\pi}{M}$ the vector averaging will tend to cancel out and the coherency will be low, and if the vector direction is not changing or changing only slightly, the coherency will be high. Thus the coherency as we use it is a measure of how rapidly the cross power phase is changing. If the records being cross correlated are identical, the phase spectrum is zero and the coherency is one. (Actually the coherency may be slightly less than one since the Daniell window is not quite rectangular.) If the records are different, the coherencies will be low unless there are some bands of frequencies where the phase remains relatively constant.

Cross Spectra of Different Components at the Same Station

Figures 1.5.1 to 1.5.3 show the results of the cross spectral computations between different components at the same station. The graphs in the figures are identified individually with the two record numbers of the data used, the indices of the first and last points of the data for each record and the Daniell parameter, M . In most cases, no computation has been done for frequencies above five cps. The recordings at any one station were made within a fraction of a wavelength of any wave of interest so that no compensation need be made for linear phase shifts due to spatial separation.

Figure 1.3.1 shows the cross-spectra of the components of the noise recorded before the Logan shot 1902 km from the shot (records 1000, 1002 and 1004). The only really prominent feature of this set of computations is the low frequency spike which is the tail end of the well-known oceanic microseisms. The Benioff instrument cuts off fairly sharply at low frequencies so that this spike is somewhat artificial in that its low frequency side is simply instrument cutoff, but that sharpness of the higher frequency side must be a real phenomenon. The phase spectrum does not show the expected 90° phase shift for Rayleigh waves, but this may be explained by the fact that the instrument characteristics are changing rapidly here and are hence possibly non-uniform from instrument to instrument. None of the frequencies with fairly high coherency seem to have phases corresponding to any known wave type. We note that the phases have been plotted to fall between $+\pi$ and $-\pi$.

Figure 1.5.2 shows the cross-spectra of the components of the noise before the Logan shot 2111 km from the shot point (Records 1006, 1008 and 1010). The 1008-1010 set of graphs have high coherence and power at 1.9 cps, but the phase is $-\pi$ which does not pin down any wave type. The peak at 2.1 cps has a phase closer to -90° which could conceivably be a Rayleigh wave. The 1006-1010 set of graphs has reasonably coherent peaks at .6, 1.4 and 1.9 cps. The .6 and 1.4 cps peaks are nearly in phase and could, therefore, be Love waves. The 1.9 cps peak is another of the many bands which are fairly coherent but have phase relationships which are not indicative of any particular wave type.

Figure 1.5.3 shows the cross spectra of the noise recorded before the Blanca shot 1610 km from the shot (records 1026, 1028 and 1030). There are possible Rayleigh waves at 1 and 2 cycles per second, but the coherencies are somewhat low.

Figure 1.5.4 shows the auto spectra of the records used in the cross spectral computations. They are included for convenient reference.

It seems that, in view of the above results, the model of a single band of surface waves from one direction is entirely too simple. It is much more likely that there are many surface waves of several frequencies coming from several sources. For a few stations quite close to the coast it may be possible to complicate the model to take care of surface waves from a few directions, and produce some believable results. However, the stations for which we have good noise data are very far inland, nearly equi-distant from the Atlantic and Pacific coasts. Thus, sources from the

Atlantic, Pacific, Gulf and Great Lakes may produce microseisms which will be recorded with nearly the same amplitude at these inland stations. On top of this we have local sources which confuse the issue considerably. The higher frequency bands at 1.4 and 2.0 cps were seen in the last two sections to have no particular directional properties and to have no simple amplitude dependence on distance from water wave sources. We conclude that there are of local origin and may be isotropic. Even a fairly complicated model taking into account many sources may not fit the data too well, and would certainly require a lot of labor to use.

Cross Spectra of Like Components at Different Stations - Linear Phase Shifts

The coherency measure used causes some difficulty if the two series are shifted in time, since a time shift will result in a linear phase shift. For example, $e^{i\omega t}$ has zero phase at time $t=0$ but at a later time the phase is ωt. If the time shift is large, the phase changes over the small band of frequencies $\omega \pm \pi/m$ will be large and will tend to reduce the coherency estimate. If meaningful coherency values are to be obtained one must line up the records properly in time before computing the cross correlation. This procedure assumes that the relative time shift is known and this is not always the case. For three component records at one station there is no difficulty since a line up in absolute time is all that is necessary. However, if one is trying to follow a wave packet across considerable distance by cross correlation and coherency measures, difficulties arise. If the records are lined up in absolute time, the relative time of the maximum of the cross correlation may give an idea of the arrival time differences, but the coherency will not necessarily be large in the range of the frequencies which comprise the wave packet. If the records are shifted the amount, τ , indicated by the maximum of the cross correlation and then cross correlated, the coherency in the frequency region which caused the maximum will certainly become larger, but there may have been features in the original record other than the wave packet which caused the maximum. Hence we have still not identified the wave packet or its relative time shift. The magnitude of the time shift for any particular wave packet will of course depend on the velocity, V, of the packet, on the distance between the stations, X , and on the direction of travel of the wave relative to a line between the stations. The time shift can therefore vary from $t=0$, if the waves are travelling perpendicular to the line between the stations, to $t = x/v$, if the waves are parallel to the line. The problem is complicated by the existence of many waves of different frequency of waves of the same frequency travelling in different directions. In even the simple case of a single wave packet dispersion may disrupt the coherence.

There is another scheme to find the appropriate time shifts which is a bit more promising than the cross correlation method. If the cross correlation is computed and not weighted by the Daniell factor, the sine and cosine transforms will not average the cross power vector over the Daniell window width. The cross power vectors can then be rotated by phase shifts corresponding to known time shifts in the frequency range of interest and averaged in this range. This is done for several time shifts and one looks for the time shift corresponding to the largest resultant of the averaged vectors. This should be close to the shift necessary to maximize the coherency in the band of frequencies when the Daniell window is used.

Some time shifting experiments have been done using data from two different stations. Cross correlation and cross spectral computations have been carried out on like components at different stations using the methods described above. Figure 1.5.5 shows the complete cross correlation of records 1000, the noise before the Logan shot 1902 km from the shot point, and record 1006, the noise before the Logan shot 2111 km from the shot point. The two records were lined up in absolute time before the computation. If most of the energy was travelling in one direction we would expect the cross correlation to have a pronounced maximum, but not necessarily for zero lag. There is no such maximum in Figure 1.5.5. (The correlation is the transient cross correlation and so dies off to zero at the ends.) If the energy were coming directly from one station to the other at about 3 km/sec it would take about 70 seconds or 1400 data points. The correlation covers from minus to plus 2999 lags and should show a maximum if one were present. It is, of course, possible that a maximum occurs for one frequency and that it is masked by the presence of other frequencies. To check this for the more energetic bands, the data was band pass filtered before correlation. Figures 1.5.6 and 1.5.7 show the cross correlation for pass bands centered at 1.4 cps and 2.0 cps. The results are perhaps a bit disappointing but not totally unexpected. The cross correlation for the 1.4 cps band is exceedingly sinusoidal. This can, of course, happen if the band is too narrow, but we expect something more like the figure for the 2 cps pass band which shows a beating between the frequencies present. It is not possible to pick a maximum on either of these figures with any certainty. If the energy is contained in such a narrow band as the 1.4 cps correlation indicates, the signal is not random enough for coherency to have any meaning.

Some time shifting was also done to maximize the coherency by looking for a linear trend in the phase. Figures 1.5.8 and 1.5.9 show cross spectral results for records 1000 and 1006 for several different time shifts. The frequencies about 1.4 and 2.0 cps were checked for a linear trend and appropriate shift made. The coherency was increased at these frequencies for the time shift indicated. The shifts were +1.5 seconds (that is, record 1000 has been shifted such that its absolute time origin, τ , lines up with absolute time $\tau + 1.5$ seconds on record 1006) and -2.5 seconds. In view of the cross correlation results, it does not seem that these time shifts, even though they increase the coherency, have any particular physical interpretation in terms of velocity and direction of travel of particular waves. If the 1.4 and 2.0 cps are from local sources (and there must be many of these local sources across the country to explain the occurrance of the spectral lines at different stations) we would not expect the time shifts to have any significance since the lines are narrow and the sources isotropic. With such narrow band signals we can expect the coherency to be high for shifts which are integer multiples of the wave period. We can see that time shifting experiments are not particularly fruitful for the narrow band signals or for the bands when the instrument characteristics change so rapidly with frequency that a mismatch between instruments is probable. The experiments are more suitable for long period records where local sources play a smaller part.

Some cross spectral computations were also done on some data from the WMSO linear array. Simultaneous sections of noise were used with no time shifting. The noise from the first instrument in the array was cross correlated with the noise from several other instruments in the array. The results are shown in Figures 1.5.10 to 1.5.15. Again we see that at the frequencies with high coherence the phase is not changing rapidly. Figures 1.5.10 and 1.5.11 have a Daniell parameter of 400 and a slightly different frequency scale from Figures 1.5.13 and 1.5.14 which have a Daniell parameter of 200. The smaller Daniell parameter will take averages over wider bands and the resulting coherencies and phases will not be quite as jagged as those for a Daniell parameter of 400. Auto spectra are shown in Figures 1.5.12 and 1.5.15. When the coherency is high, we tend to say that the waves at that frequency are travelling at right angles to the array and there is no linear phase shift to disrupt the coherency computation. The phase spectra also show in some cases linear trends over bands of frequencies which are of course accompanied by low coherencies. A time shift would bring up the coherency and indicate the direction of travel of the source waves for these bands.

A much more sophisticated analysis of array data is needed before any reliable results can be stated. Simulation studies of the sort described in Chapter 3 would be of interest with the array recordings time shifted (delayed) to minimize the noise and thus utilize the directional properties of the array. Similar studies could also be done with data from a two dimensional array.

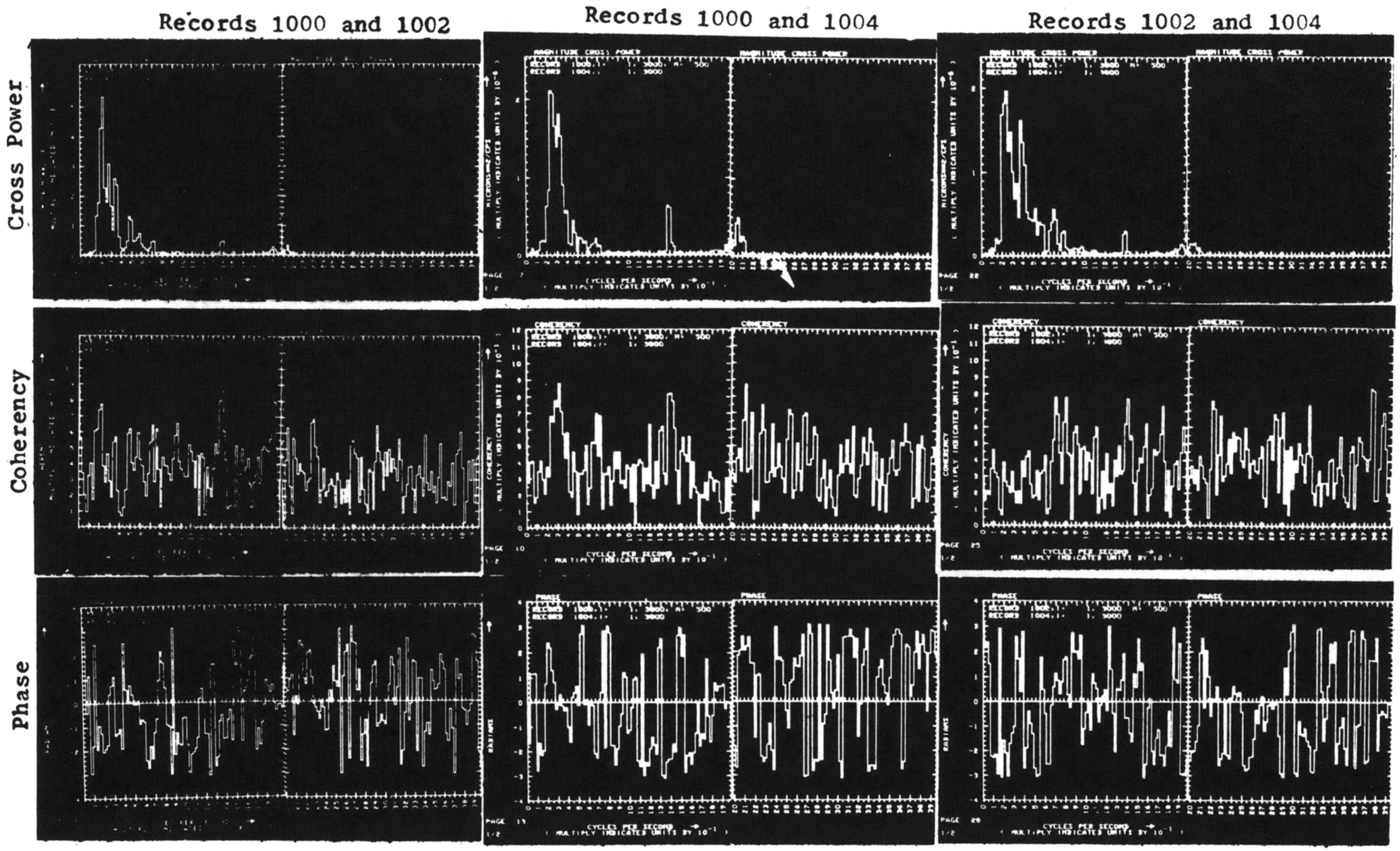


Figure 1.5.1 Cross Spectra of Different Components at the Same Station

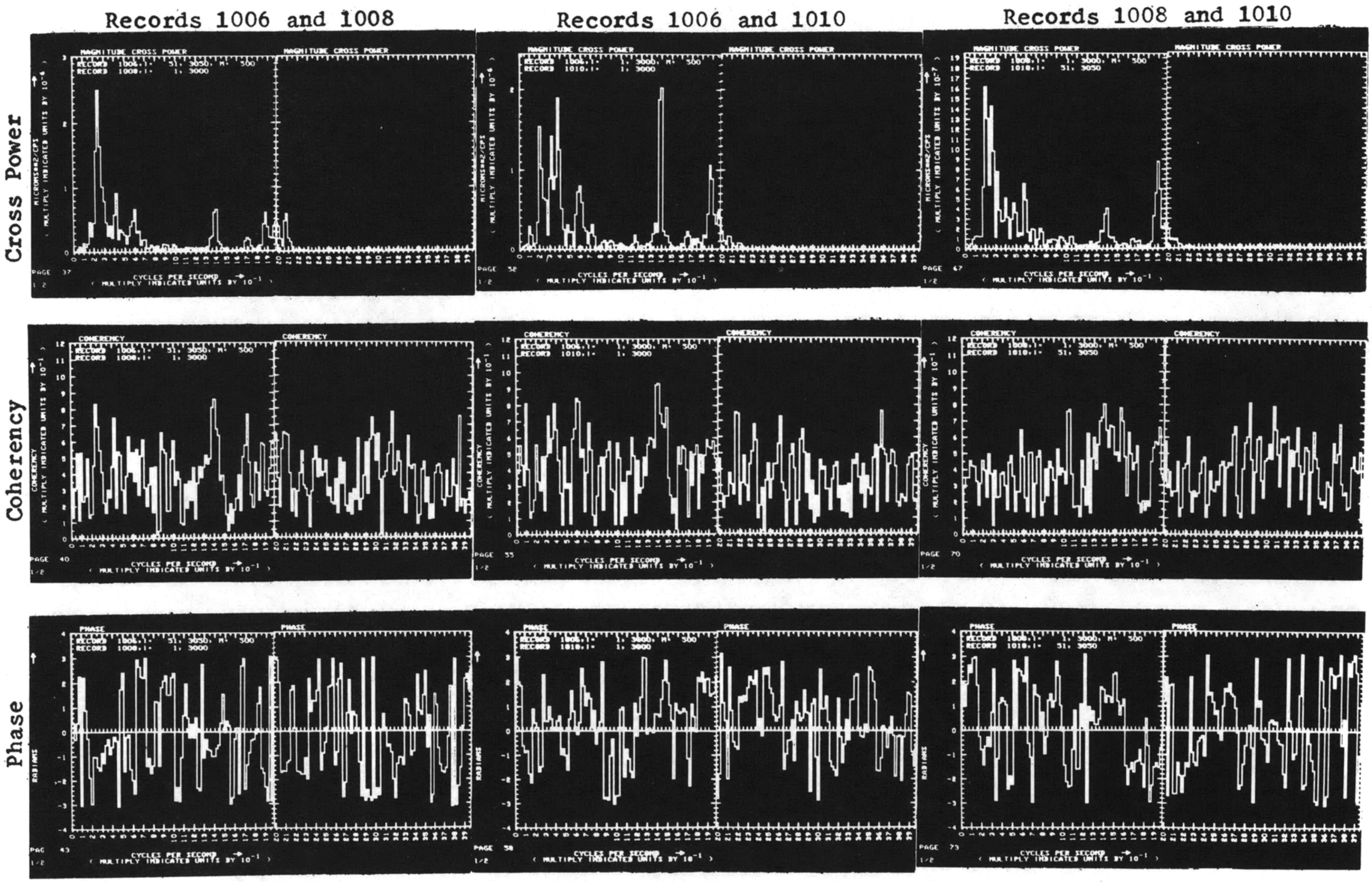


Figure 1.5.2 Cross Spectra of Different Components at the Same Station

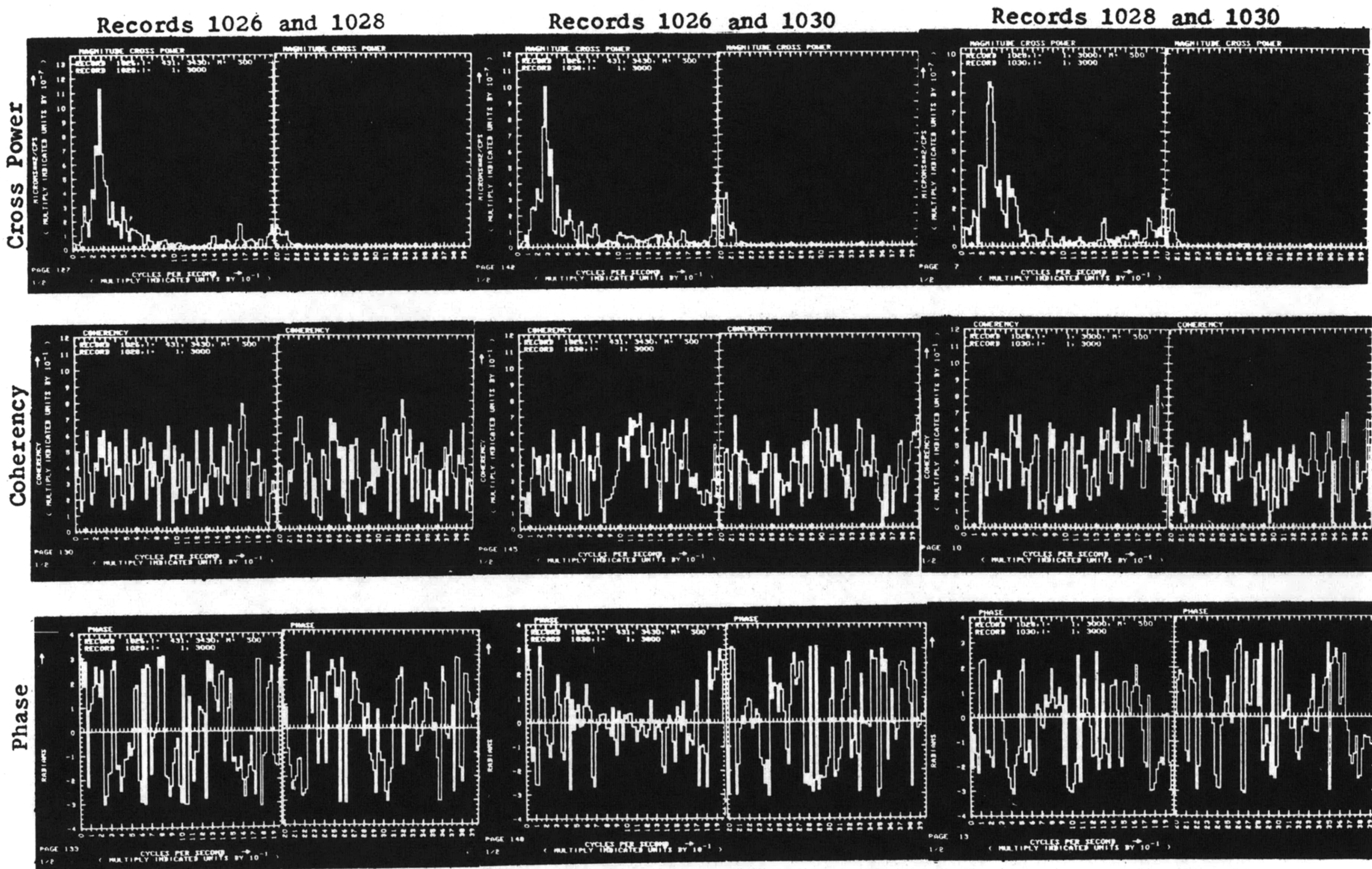
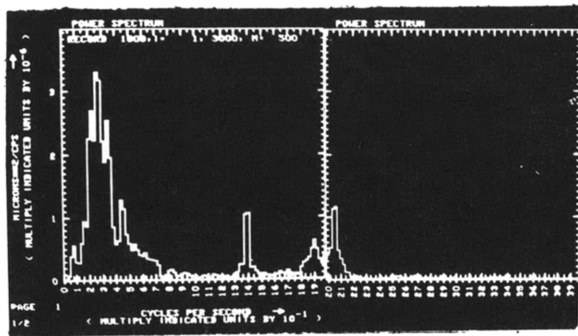
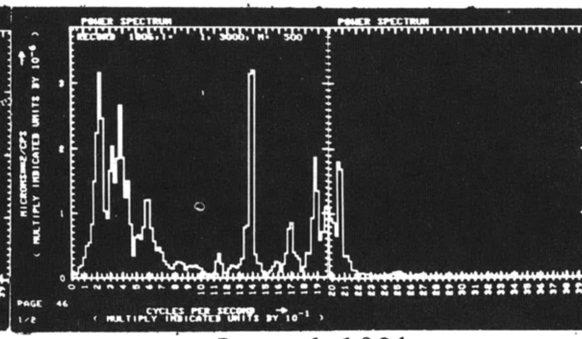


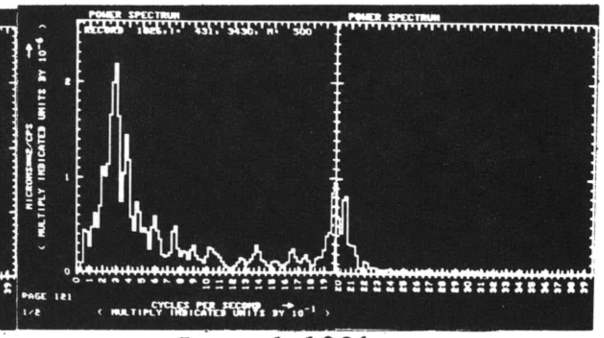
Figure 1.5.3 Cross Spectra of Different Components at the Same Station



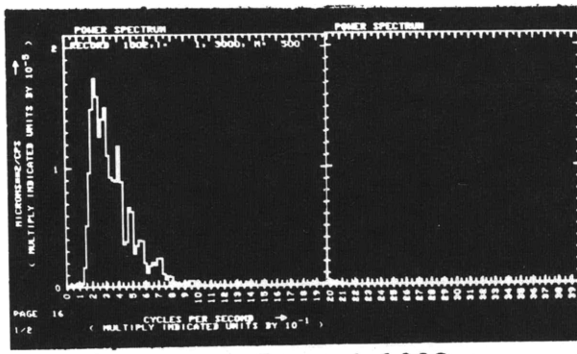
Record 1000



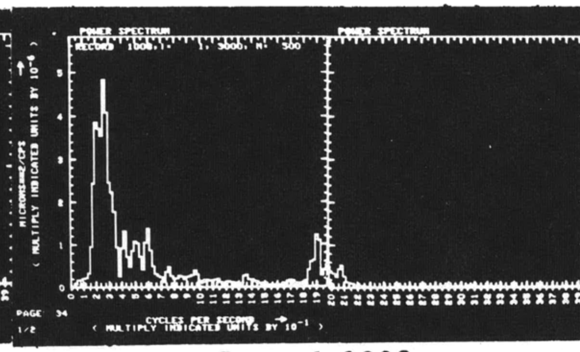
Record 1006



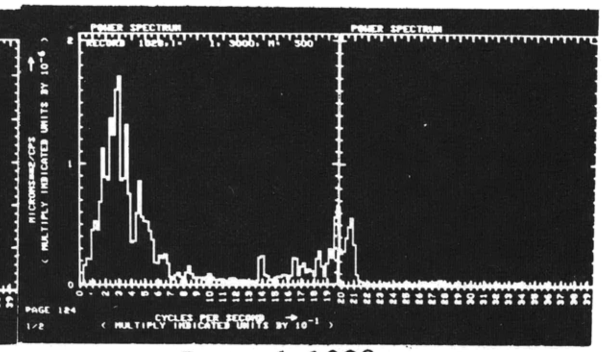
Record 1026



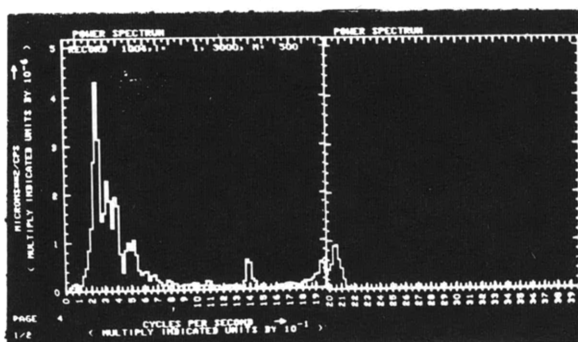
Record 1002



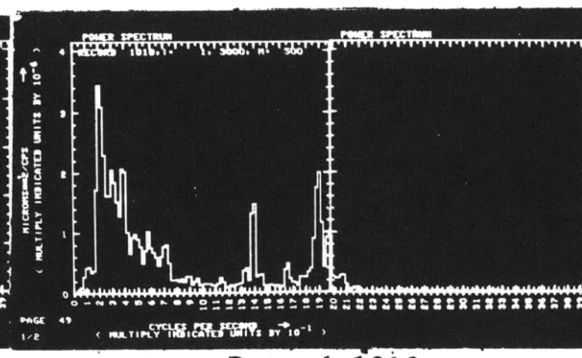
Record 1008



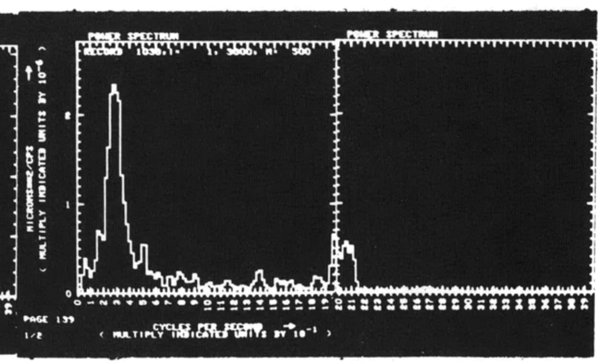
Record 1028



Record 1004



Record 1010



Record 1030

Figure 1.5.4 Auto Spectra

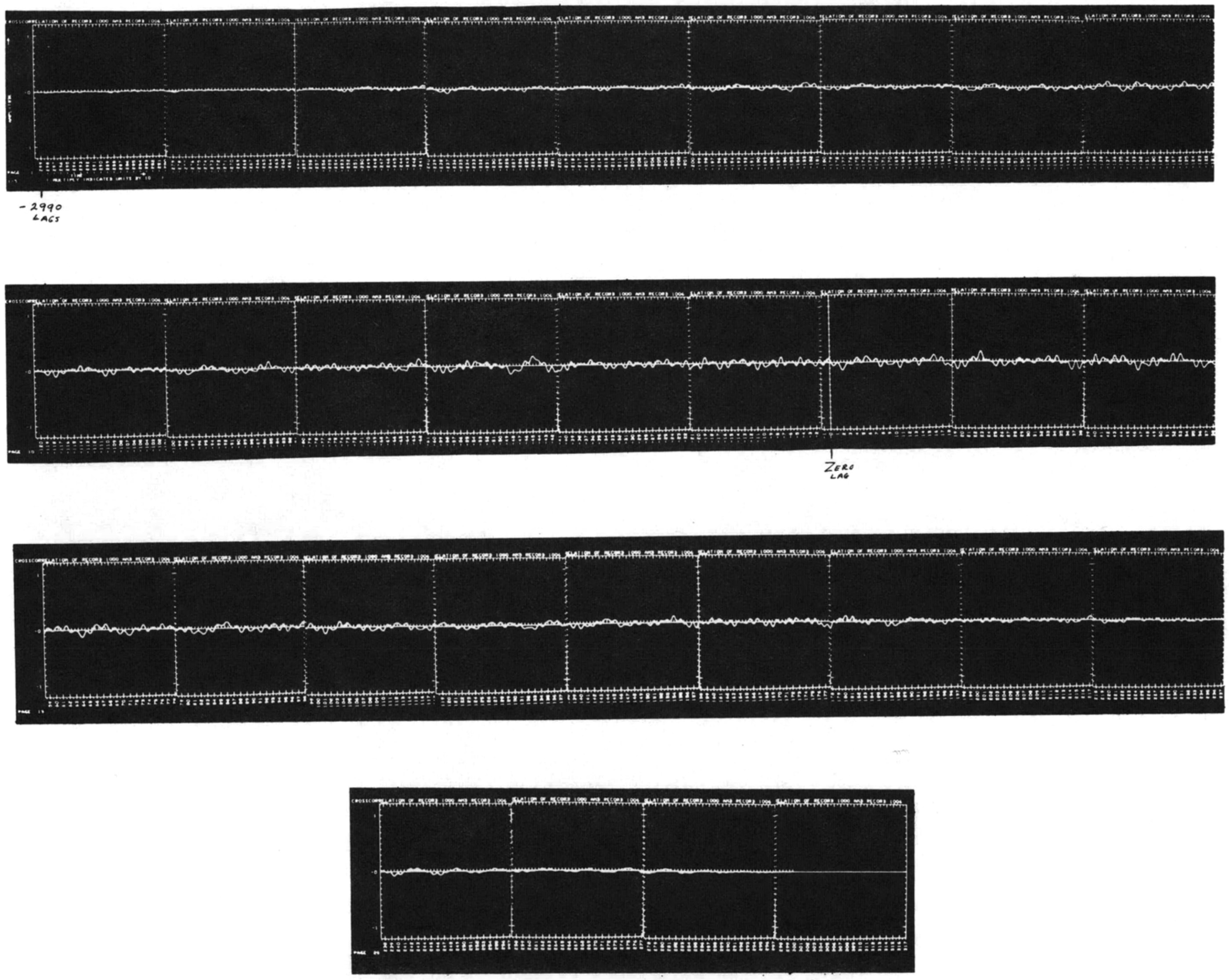


Figure 1.5.5 Complete Transient Cross Correlation of Records 1000 and 1006

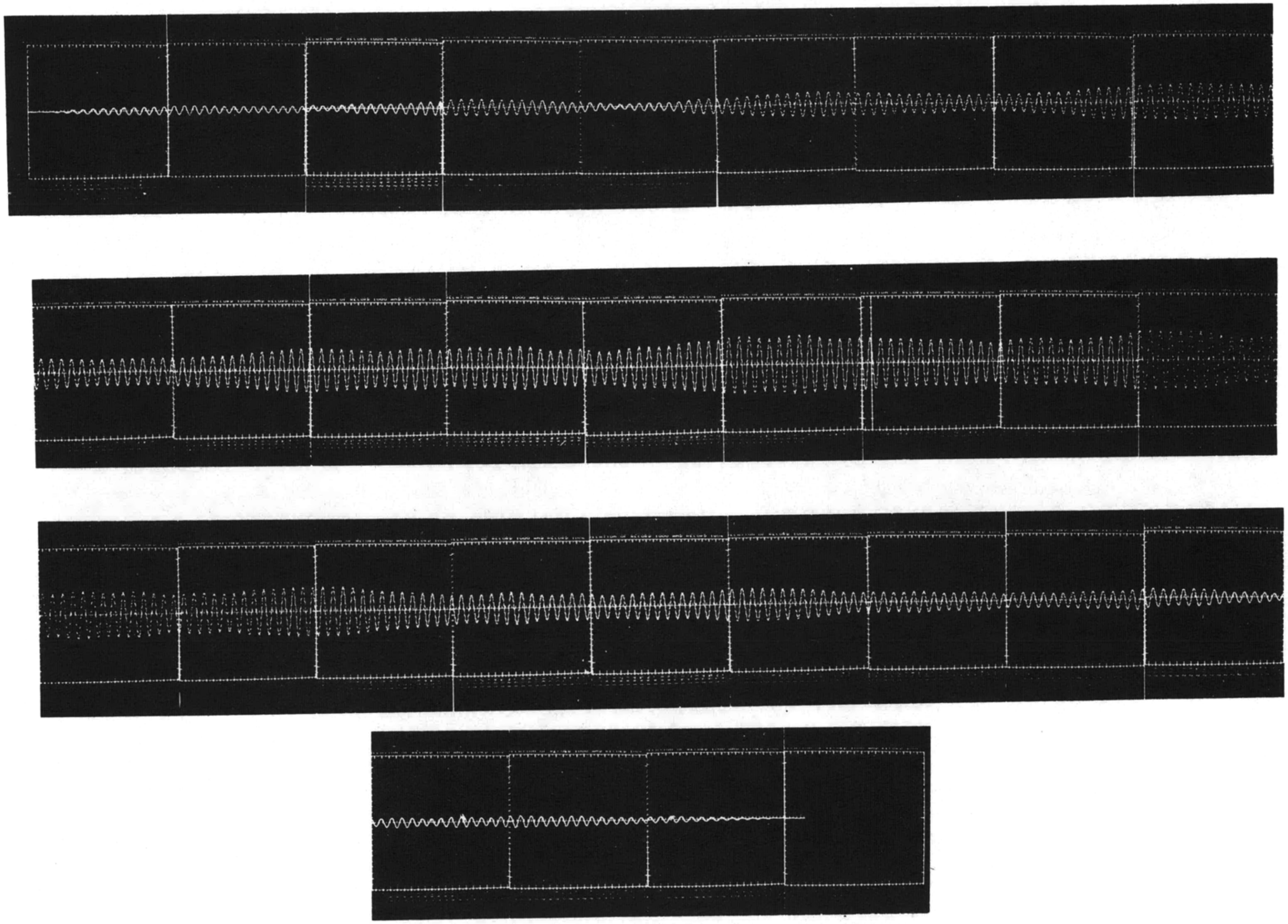


Figure 1.5.6 Complete Transient Cross Correlation of Records 1000 and 1006
Band Pass Filtered at 1.4 Cycles Per Second

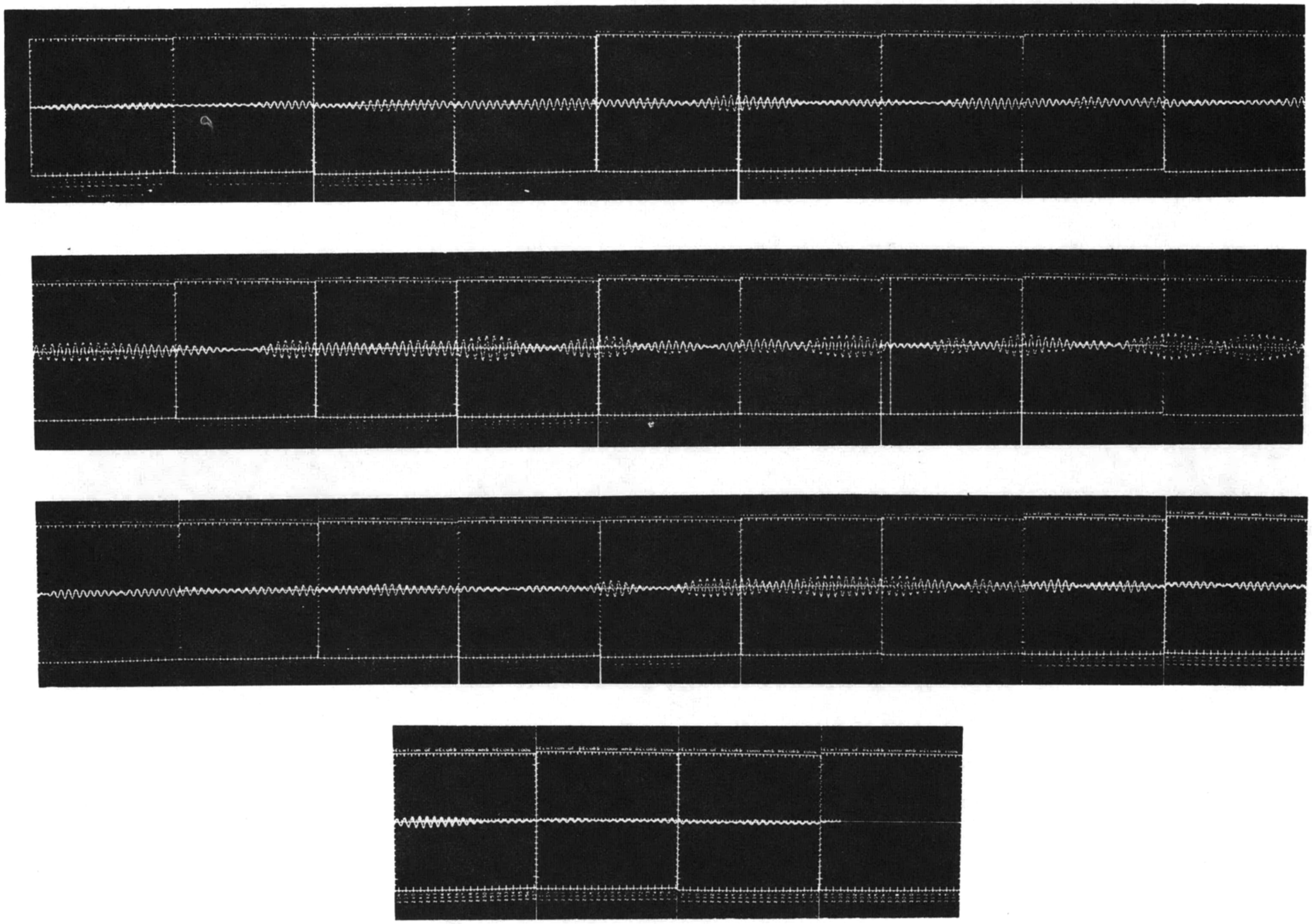


Figure 1.5.7 Complete Transient Cross Correlation of Records 1000 and 1006 Band Pass Filtered at 2.0 Cycles Per Second

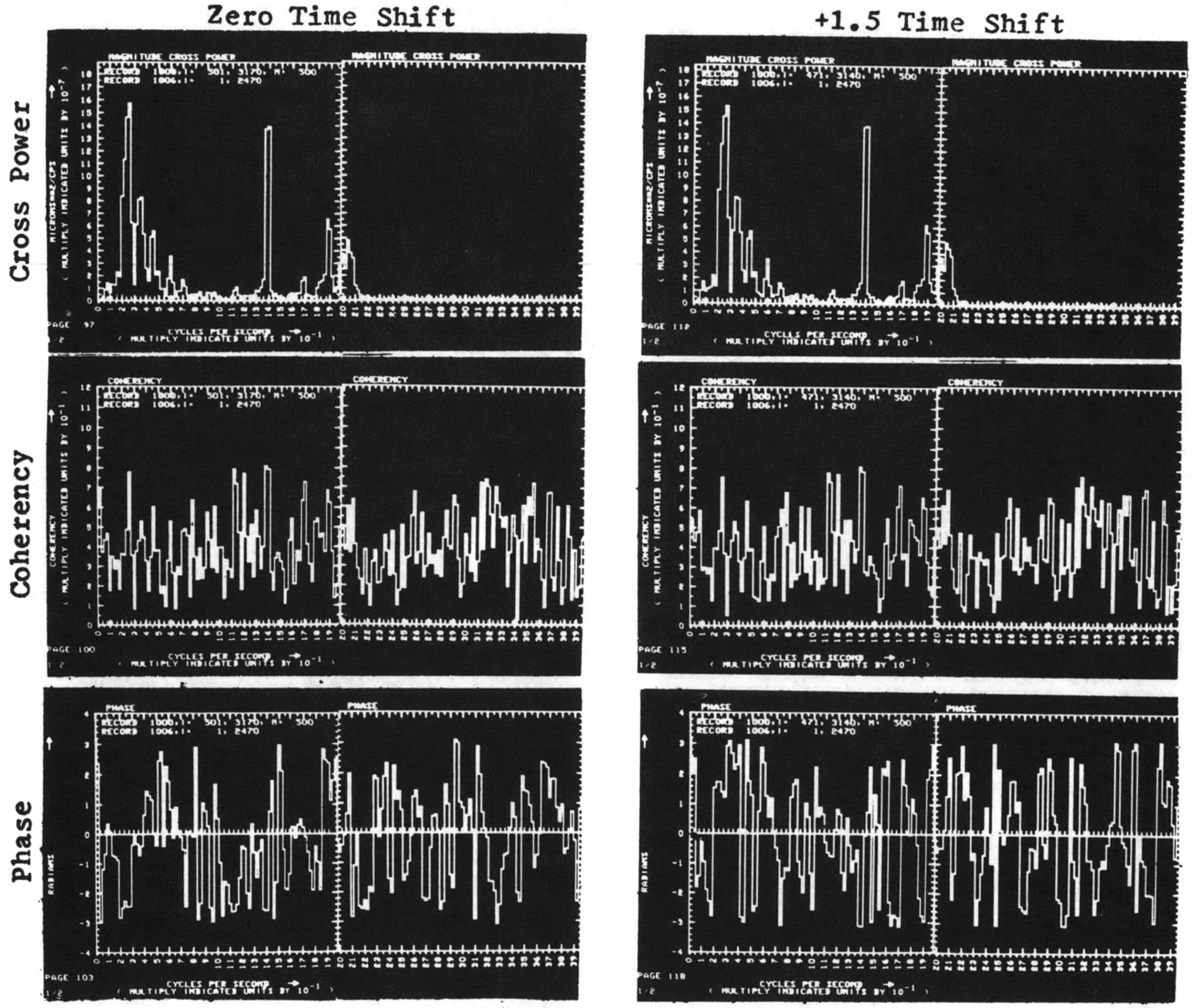


Figure 1.5.8 Cross Spectra of Records 1000 and 1006 For Indicated Time Shifts

-1.5 Time Shift

-2.5 Time Shift

Cross Power

Coherency

Phase

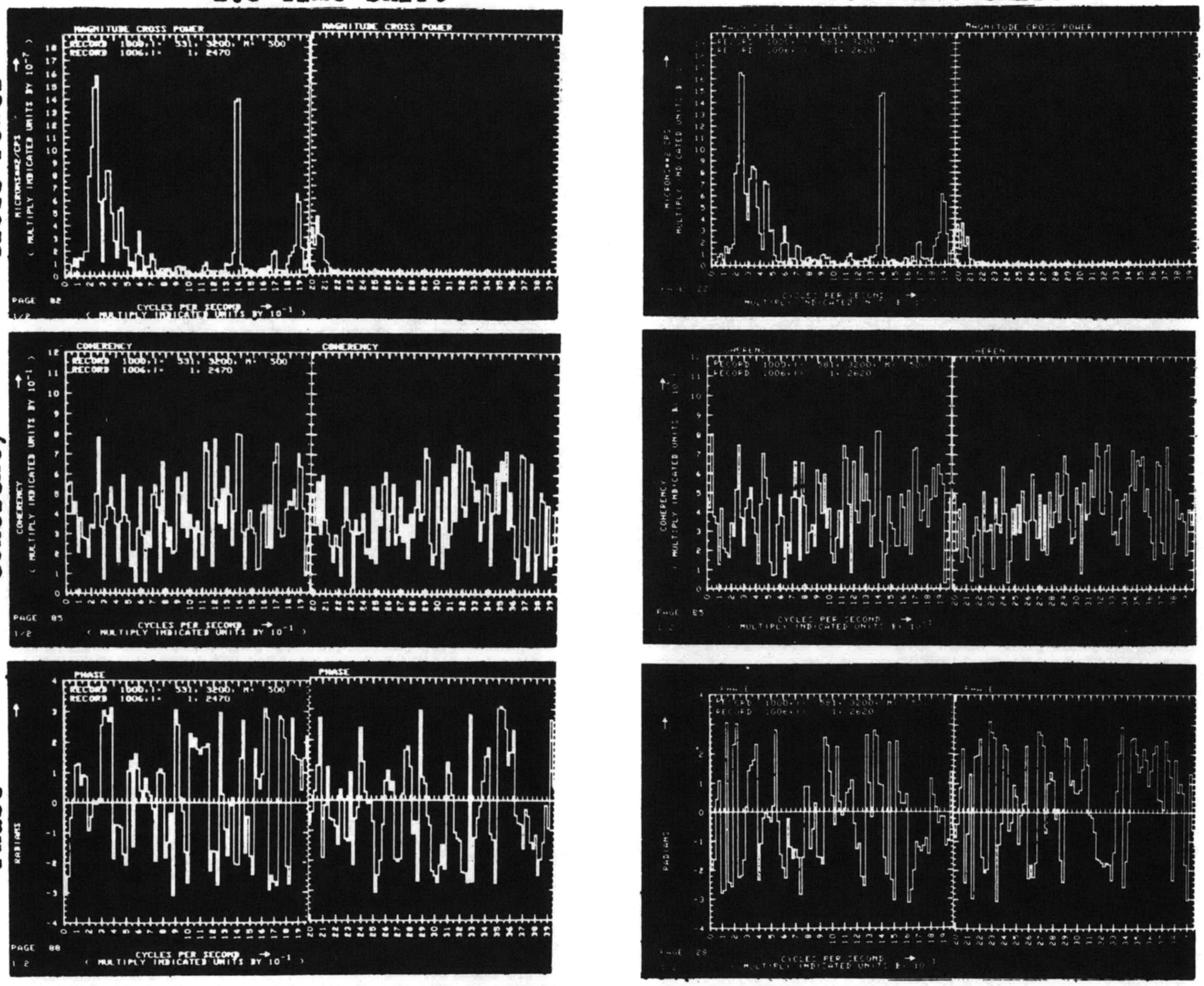


Figure 1.5.9 Cross Spectra of Records 1000 and 1006 For Indicated Time Shifts

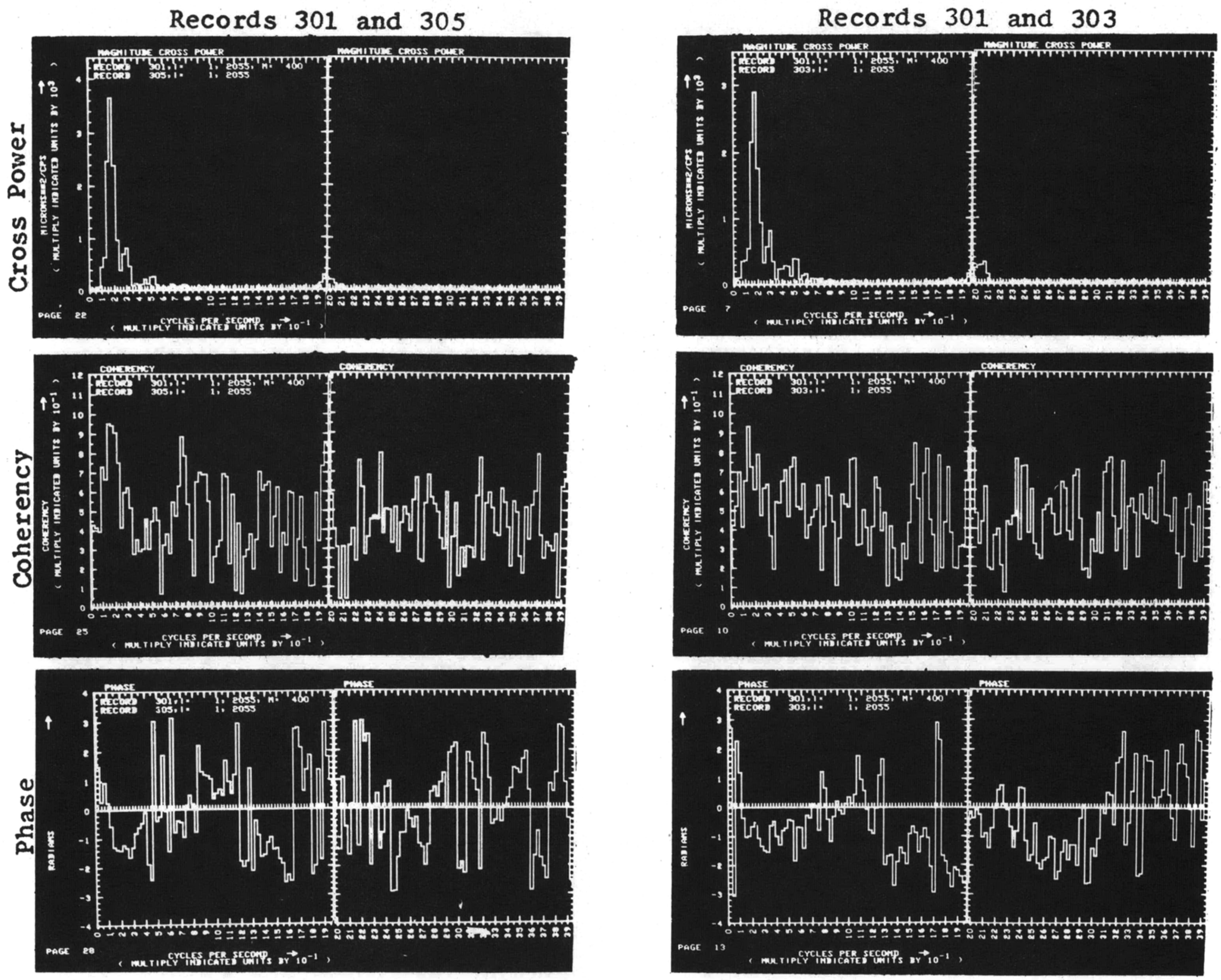


Figure 1.5.10 Cross Spectra of Array Elements

Records 301 and 307

Records 301 and 309

Cross Power

Coherency

Phase

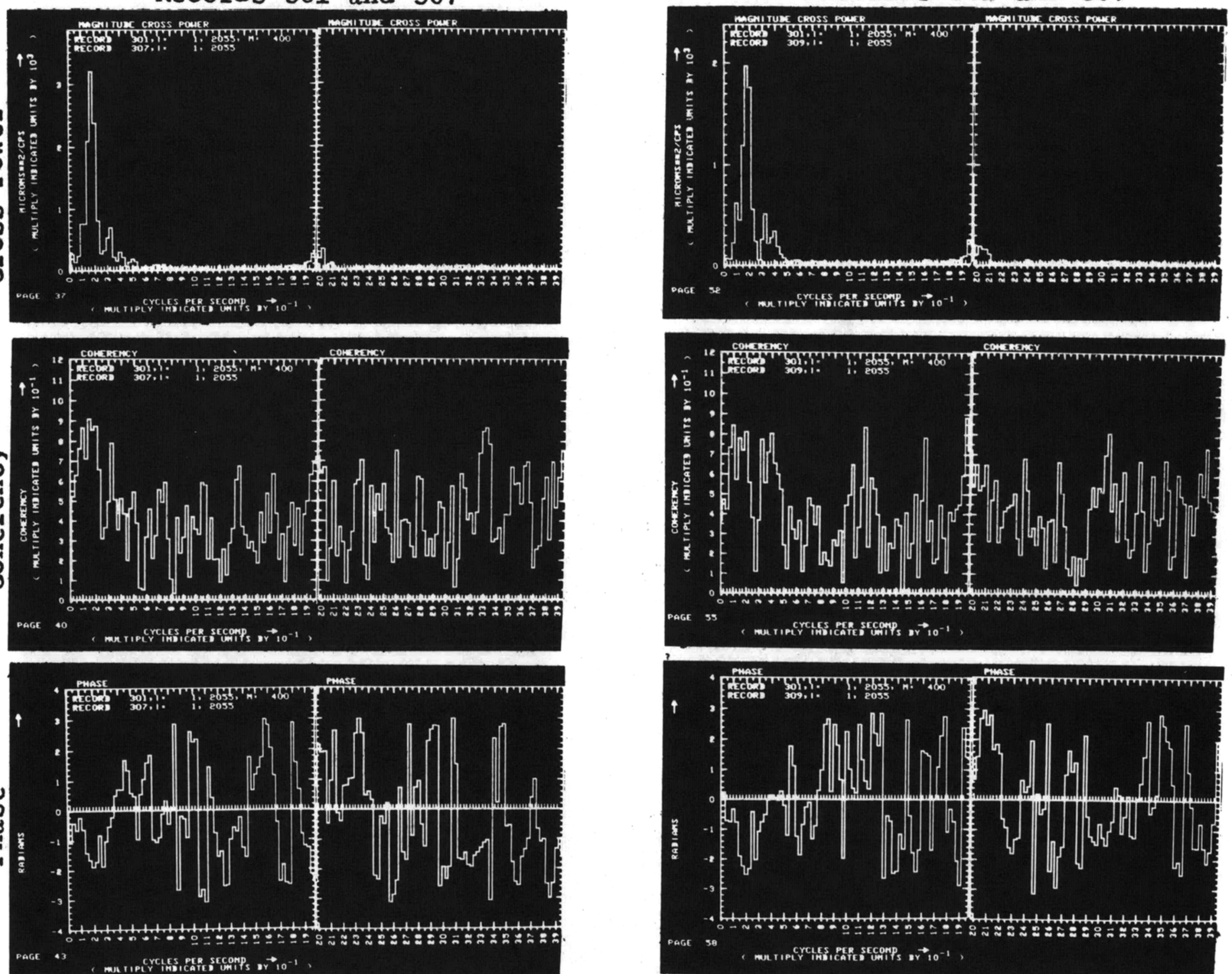


Figure 1.5.11 Cross Spectra of Array Elements

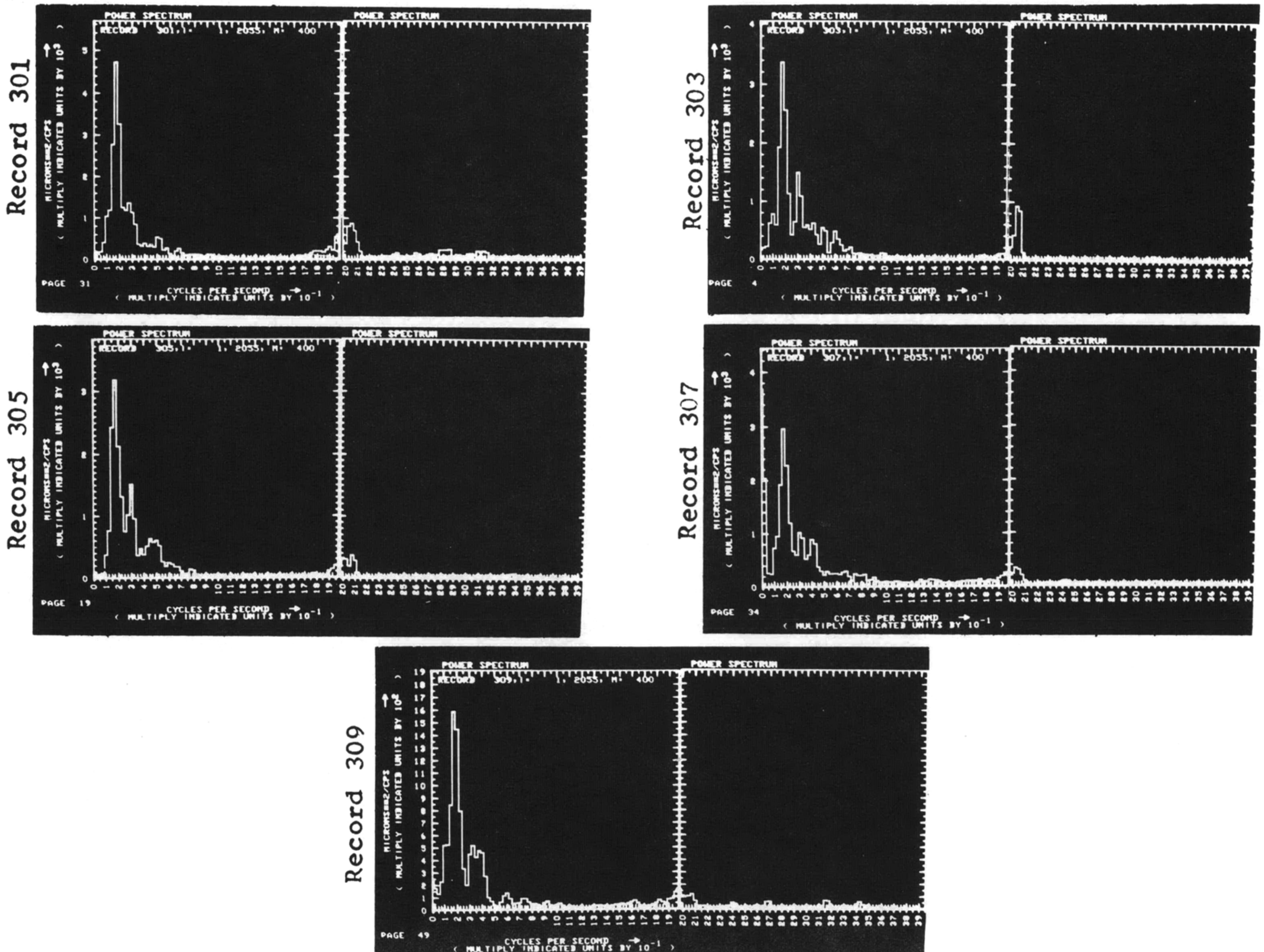


Figure 1.5.12 Auto Spectra of Array Elements

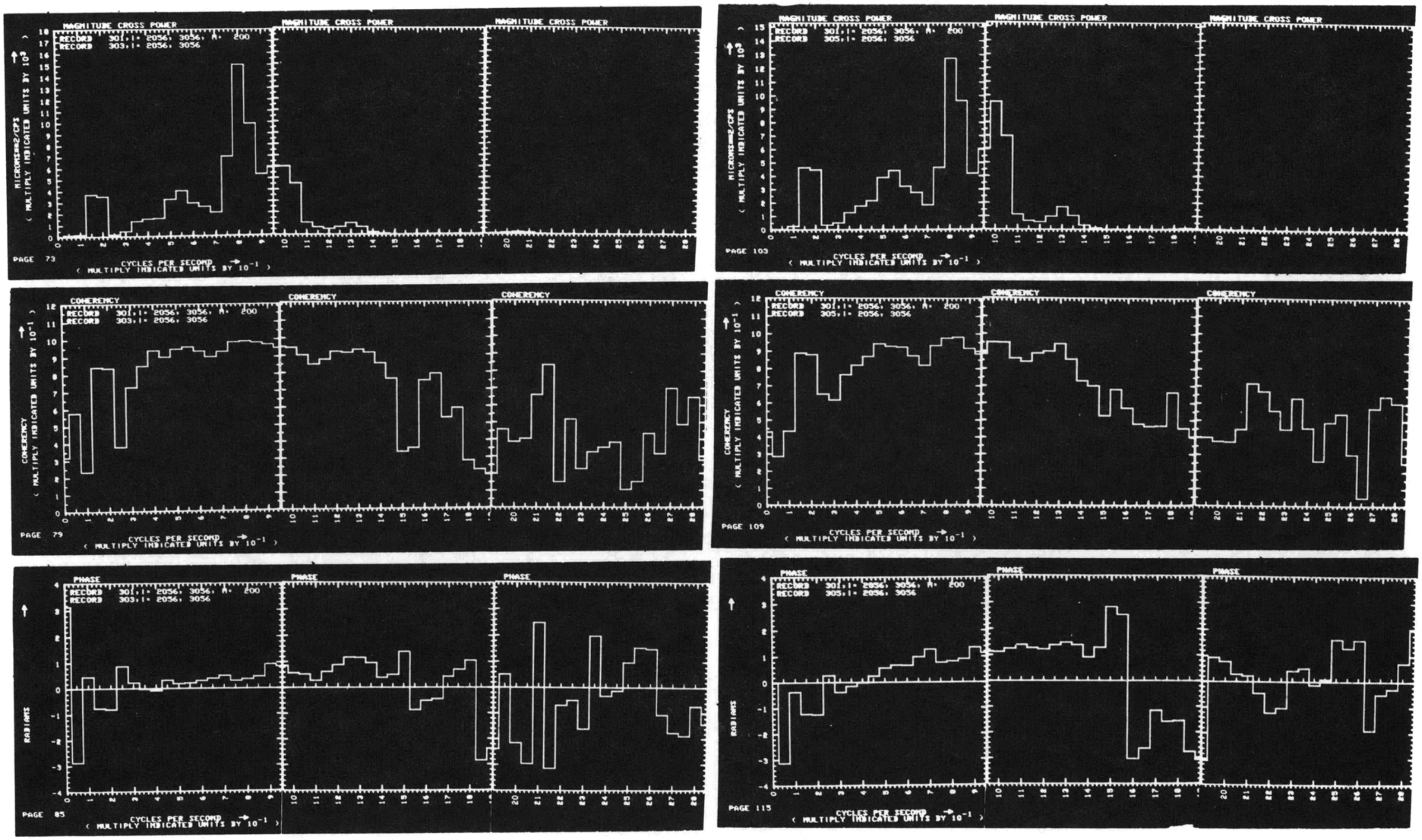


Figure 1.5.13 Cross Spectra of Array Elements

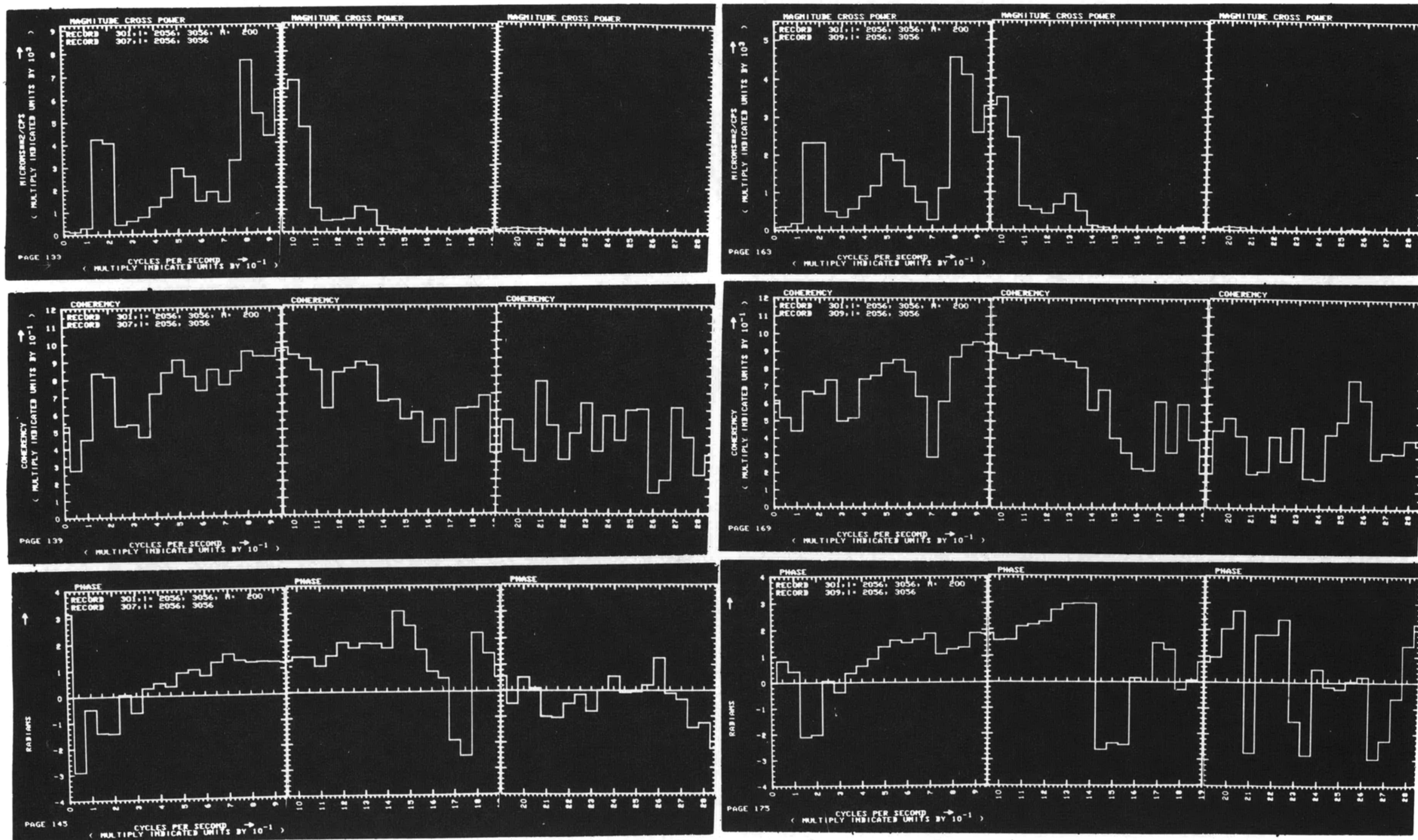


Figure 1.5.14 Cross Spectra of Array Elements

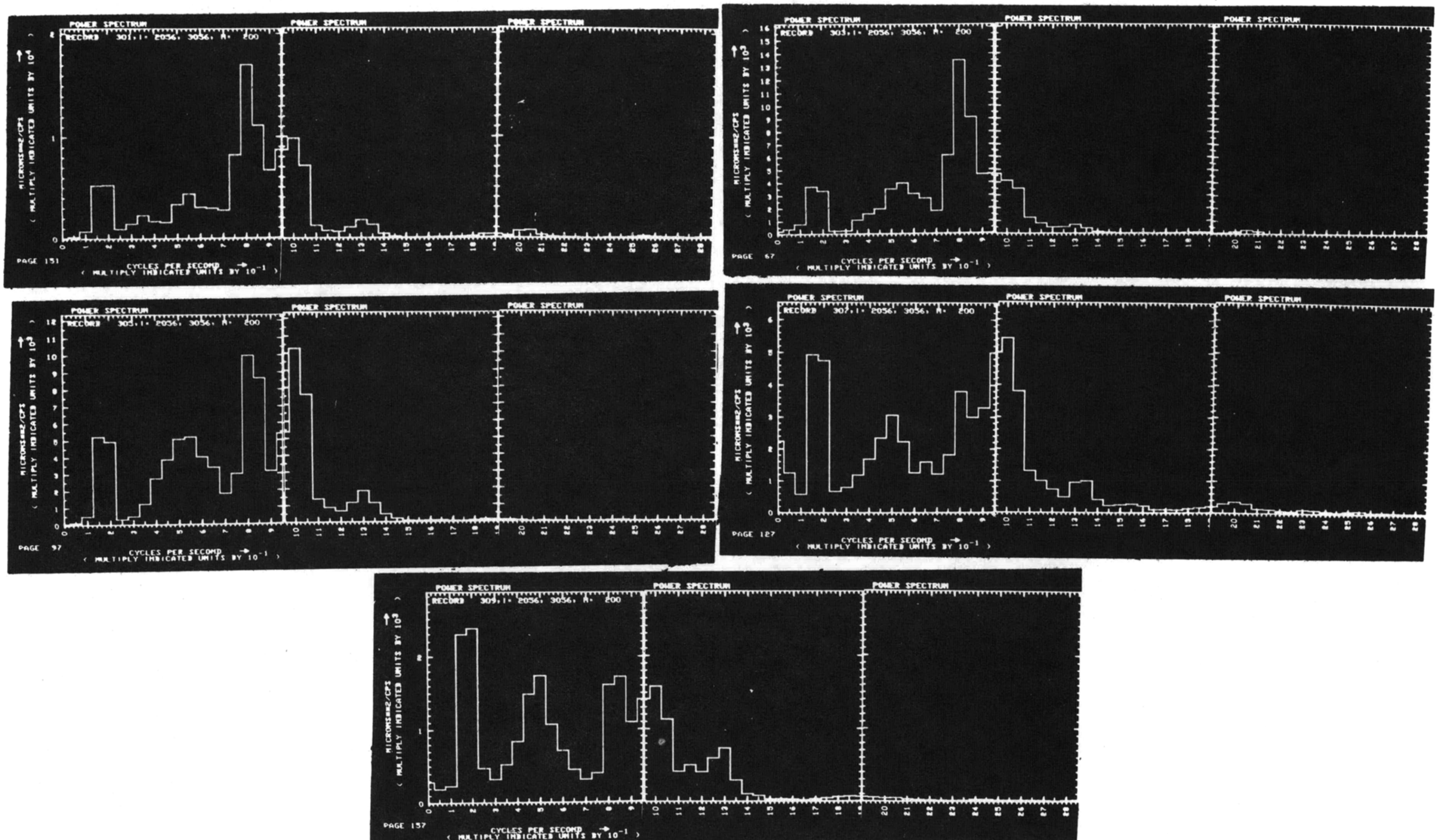


Figure 1.5.15 Auto Spectra of Array Elements

2. PREDICTION OF MICROSEISMS

2.1 Prediction by Minimization of Mean Squared Error

Prediction and the First Motion Interval

Elementary considerations of the possible differences between the signals from earthquakes and the signals from underground explosions were based on the obvious differences in the source mechanisms. An explosion should give an initial compression whereas an earthquake, being a shearing source, should give compressions or rarefactions depending on the position of the observer relative to the fault plane and the direction of slip along the plane. A group of recording stations around a source should therefore all record initial compressive first motion for an explosion, but would vary if an earthquake were the source. Granting the first motion criterian is legitimate, there is still the problem of identifying the first motion on the record when the signal is corrupted by noise. The problem is somewhat simplified by the fact that, even though its pulse may be small, the first motion is followed by stronger P waves which are easily discernible in the noise. These P waves therefore allow us to say approximately where in time the first motion pulse arrived. If we could by some means predict what the noise would be in a small interval preceeding the strong P waves and subtracted the predicted noise from the signal plus noise, we would be left with the uncorrupted signal and could make definite statements concerning the direction of first motion. Figure 2.1.1 illustrates this idea with the assumption of perfect prediction of the noise.

In general, of course, we cannot predict perfectly, but a good prediction could possibly increase the signal to noise ratio to a point where there would no difficulty in picking out the first motion direction. We will therefore wish to express the predictability of the noise in terms of signal to noise ratio imprevement. Evaluation of the effectiveness of the scheme can be done by prediction studies of the noise alone without reference to any particular signal. The only parameter we need is time length over which we must predict. This will be called the prediction distance and it will be denoted by K in the following analysis.

We wish to form a linear operator which will predict the "future" of a record, x_i , from its "past" and possibly from the past of other related records (e.g. three components at one station). We note that even though we are not necessarily operating in real time it is necessary that we use only the past as a basis for prediction since the past is noise alone and the future is signal plus noise. We shall present the analysis for the formation of a linear operator operating on three records to predict one of the three. The expressions will reduce simply to the case of self prediction, the prediction of one record from itself. The analysis has been done (Wadsworth et al, 1953) for the two dimensional case and the simple extension to three dimensions is given here.

The requirement that the record x_i be predicted from itself and from y_i and z_i can be stated by the regression function (Wadsworth et al, 1953).

$$\hat{x}_{i+K} = d + \sum_{s=0}^{M} a_s x_{i-s} + \sum_{s=0}^{M} b_s y_{i-s} + \sum_{s=0}^{M} c_s z_{i-s}$$

where $\hat{X}_{i+k}$ is the predicted value of the X_i time series k time units ahead. One time unit is simply the sampling period and is .05 seconds for the Logan and Blanca records. The X_i are the actual noise values and d, a_s, b_s and c_s constitute the linear operator which must be determined. The criterion used in this determination is the Wiener mean squared error criterion where we wish to minimize the sum of the mean squared error between the actual and predicted X_i series. This means, of course, that we have to know what the future is of the noise above. Hence a long series of pure noise is arbitrarily divided into past and future and the operator formed. The operator, under the assumption of stationarity of the time series, can then be used on the portion of the noise preceding the first motion to predict the noise in the first motion interval.

Mean Squared Error Techniques for Three-Dimensional Case

The sum of the squared error is taken over the operator interval length from $i+k=N$ to $i+k=N+n-1$ a duration of n time units. Thus we minimize I where

$$I = \sum_{i=N-k}^{N+n-1-k} \left(X_{i+k} - \hat{X}_{i+k} \right)^2$$

$$I = \sum_{i=N-k}^{N+n-1-k} \left[X_{i+k} - \left(d + \sum_{s=0}^{M} a_s X_{i-s} + \sum_{s=0}^{M} b_s y_{i-s} + \sum_{s=0}^{M} c_s z_{i-s} \right) \right]^2$$

with respect to d, a_s, b_s and c_s. This is done by setting the partial derivatives with respect to d, a_s, b_s and c_s equal to zero for all s. The resulting set of $3M+4$ equations for the $3M+4$ operation coefficients is

$$nd + \sum_s \left[a_s \sum_i x_{i-s} + b_s \sum_i y_{i-s} + c_s \sum_i z_{i-s} \right] = \sum_i x_{i+k}$$

$$d \sum_i x_{i-r} + \sum_s \left[a_s \sum_i x_{i-s} x_{i-r} + b_s \sum_i y_{i-s} x_{i-r} + c_s \sum_i z_{i-s} x_{i-r} \right] = \sum_i x_{i-r} x_{i+k}$$

$$d \sum_i y_{i-r} + \sum_s \left[a_s \sum_i x_{i-s} y_{i-r} + b_s \sum_i y_{i-s} y_{i-r} + c_s \sum_i z_{i-s} y_{i-r} \right] = \sum_i y_{i-r} x_{i+k}$$

$$d \sum_i z_{i-r} + \sum_s \left[a_s \sum_i x_{i-s} z_{i-r} + b_s \sum_i y_{i-s} z_{i-r} + c_s \sum_i z_{i-s} z_{i-r} \right] = \sum_i z_{i-r} x_{i+k}$$

for $r = 0$ to M.

where summations over i are from $i = N-k$ to $i = N+n-1-k$, and summations over s are from $s=0$ to $s=M$. We write this as the matrix equation

$$RA = B \tag{2.1.1}$$

where R is a $3M+4$ by $3M+4$ symmetric correlation matrix, each element depending essentially on different lags of the auto and cross correlations of x_i, y_i and z_i. A is the $3M+4$ by L solution matrix where each column of A is the prediction operator $(a_0^k, \dots a_M^k, b_0^k, \dots, b_M^k, c_0^k, \dots, c_M^k, d^k)$ for different prediction distance k, and k takes on L different values. A is obtained by inversion of the R matrix.

$$A = R^{-1} B$$

B is an L by $3M+4$ matrix, where each column of B is the right hand side of the equation for a different k. The matrix equation can be partitioned as shown below

$$
\left[\begin{array}{c|c|c|c}
R^{11}_{rs} = \sum_i x_{i-r} x_{i-s} & R^{12}_{rs} = \sum_i x_{i-r} y_{r-s} & R^{13}_{rs} = \sum_i x_{i-r} z_{r-s} & R^{14}_{rs} = \sum_i x_{i-r} \\
\hline
 & R^{22}_{rs} = \sum_i y_{i-r} y_{r-s} & R^{23}_{rs} = \sum_i y_{i-r} z_{r-s} & R^{24}_{rs} = \sum_i y_{i-r} \\
\hline
 & & R^{33}_{rs} = \sum_i z_{i-r} z_{r-s} & R^{34}_{rs} = \sum_i z_{i-r} \\
\hline
 & & & R^{44}_{rs} = n
\end{array}\right]
\left[\begin{array}{c}
a_0^k \cdots \\ \vdots \\ a_M^k \cdots \\
\hline
b_0^k \cdots \\ \vdots \\ b_M^k \cdots \\
\hline
c_0^k \cdots \\ \vdots \\ c_M^k \cdots \\
\hline
d^k \cdots
\end{array}\right]
=
\left[\begin{array}{c}
B^{11}_{rk} = \sum_i x_{i-r} x_{i+k} \\
\hline
B^{21}_{rk} = \sum_i y_{i-r} x_{i+k} \\
\hline
B^{31}_{rk} = \sum_i z_{i-r} x_{i+k} \\
\hline
\sum_i x_{i+k}
\end{array}\right]
$$

If we donote the auto correlation or Toeplitz matrix by

$$\begin{bmatrix} r_0 & r_1 & r_2 & \cdots & r_n \\ r_n & r_0 & r_1 & \cdots & r_{n-1} \\ r_{n-1} & r_n & r_0 & r_1 & \ddots \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

where r_j is the auto correlation for the j^{th} lag we see that the diagonal submatrices of R in equation (2.1.1) are not quite auto correlation matrices because the terms along diagonals of the submatrices are summed over different intervals. If the operator interval length, n, is large, the diagonal submatrices are only very slightly different from auto correlation matrices and approach this as $n \rightarrow \infty$ If we take the one dimensional zero mean case $(b_s = c_s = \alpha = 0)$ with n large, the problem becomes the same as that treated by Levinson (1949).

Predictability and the Percent Reduction

A measure of how well the prediction operator performs its task is the percent reduction, R_p. This quantity is defined (Wadsworth et al, 1953) as

$$R_p = 100\left(1 - \frac{I_m}{I_o}\right)$$

where I_m is the value for I for the operator used and I_o is a measure of the sample variance over the same interval.

$$I_o = \sum_i (x_{i+k} - \bar{x})^2$$

If we think of $I_0 - I_M$ as a measure of the variance of the prediction we can see that the percent reduction is a measure of the amount of power which can be predicted. In terms of the signal to noise ratio, if we take S as a general signal and N the noise, then before filtering we have

$$\left(\frac{S}{N}\right)_{BEFORE} = \frac{S}{\sqrt{\frac{1}{n} I_0}}$$

and after filtering

$$\left(\frac{S}{N}\right)_{AFTER} = \frac{S}{\sqrt{\frac{1}{n} I_m}}$$

Hence

$$\left(\frac{S}{N}\right)_{AFTER} = \frac{1}{\sqrt{1 - \frac{R_p}{100}}} \left(\frac{S}{N}\right)_{BEFORE}$$

<u>Prediction Computations</u>

In order to test the predictability, then, one must take a section of noise record, divide it into past and future and form the R and B matrices given in equation (2.1.1). The R matrix is inverted and R^{-1} is multiplied by B . The columns of the resulting A matrix are the operators or filters for different prediction distance K . n predictions for a given K are made by moving the operator along the real data for successive points. The prediction error, I_m for this K can then be formed and, with I_0 for the same n points, the percent reduction can be computed. This is done for each operator so that the percent reduction as a function of K can be obtained.

This procedure has been programmed for the IBM 709-7090 computers. Computation has been done for one dimension with several M values with $K = 1$ to 30 and for three dimensions with $M = 30$ also for $K = 1$ to 30. The results of the one dimensional experiments are shown in Figures 2.1.2 to 2.1.4. The percent reduction should increase with increasing length of operator (M value) and does in all cases computed. For an infinite length operator the percent reduction must decrease monotonely with K (Robinson 1954, p. 148) which does not occur in the cases shown. This is obviously due to the short operator lengths used in the computations, and we can be sure that higher percent reduction would be obtained with longer operators. The spectra of the records (Figures 1.3.6 to 1.3.9) show that most of the energy is crowded into a few narrow bands, the lowest frequency being about 1 cps. It would be best to have operator lengths covering a few wave lengths of the major frequency components which in this case would be about three seconds or at least 60 terms. The method of solution for the operators then involves inversion of a 60 by 60 matrix which starts to suffer from round off error.

We note that in all cases the percent reduction falls off rapidly at first and then has one or more plateaus. The Cherry Hill Park records remain fairly predictable out to three seconds, maintaining a percent reduction of about 50. This is attributed to the narrowness of prominent spectral lines of these records. (A spike in the frequency domain represents a sine wave and can be predicted exactly with a two term operator.)

If a typical wave length of the first motion is established at 1 second the corresponding prediction distance for the C.H.P. records would be 10 units. This would give a signal to noise ratio improvements of 1.4 and 1.3 for C.H.P. 31 (record 237) and C.H.P. 4 (record 204) which is not significant.

The Logan 1902 records show a plateau effect in the percent reductions but the initial fall is more pronounced than in the C.H.P. records. The vertical is the most predictable component and a 20 term operator gives a signal to noise improvement of only about 1.3 for 1 second (20 units).

We have seen that the predictability in the one dimension or self prediction case is not particularly significant. However, one might expect that the use of information from more than one component would do somewhat better if the components used are related. The analysis for three components has been shown and was programmed for the IBM 709-7090 computers.

The precent reduction for M values of 5, 10, 15 and 20 (corresponding to operator lengths of 16, 31, 46 and 61) for the prediction of the vertical component, Logan 1902 km, record 1002 from itself and the two horizontals is shown in Figure 2.1.5. Comparison of this figure with Figure 2.1.3, the self prediction results, shows an almost imperceptable improvement by using all components.

As mentioned above, the predictability is almost certain to be better if longer operators are used. With the above method of solution the increase of operator length becomes impossible because the machine core is rapidly used up and significant additional time is needed for the computation. Therefore another method must be applied to obtain the longer operators or the idea of prediction must be discarded as impractical. Such a method does, however, exist and is treated in the next section, 2.2.

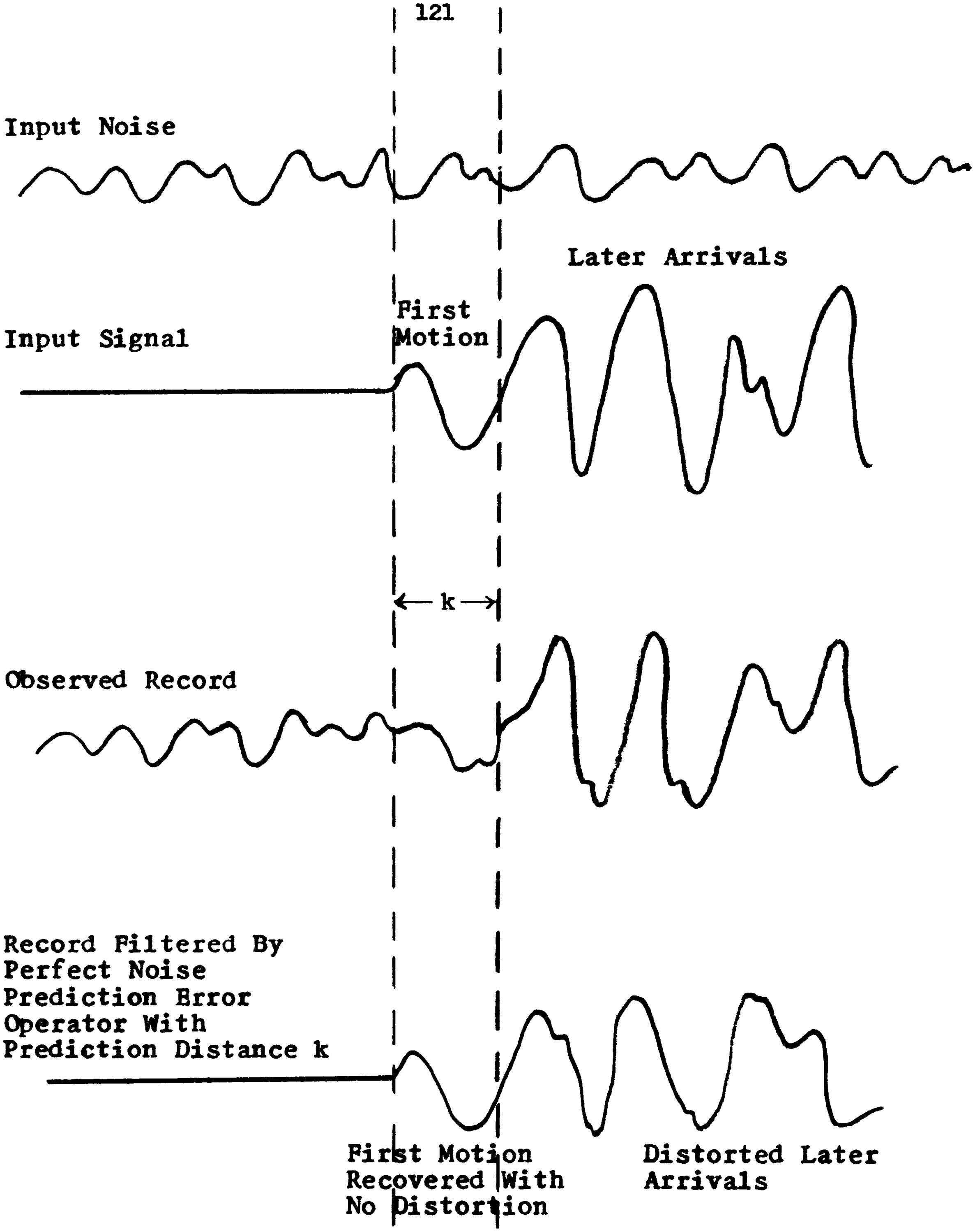


Figure 2.1.1 Concept Behind Least Squares Prediction Operator Experiments.

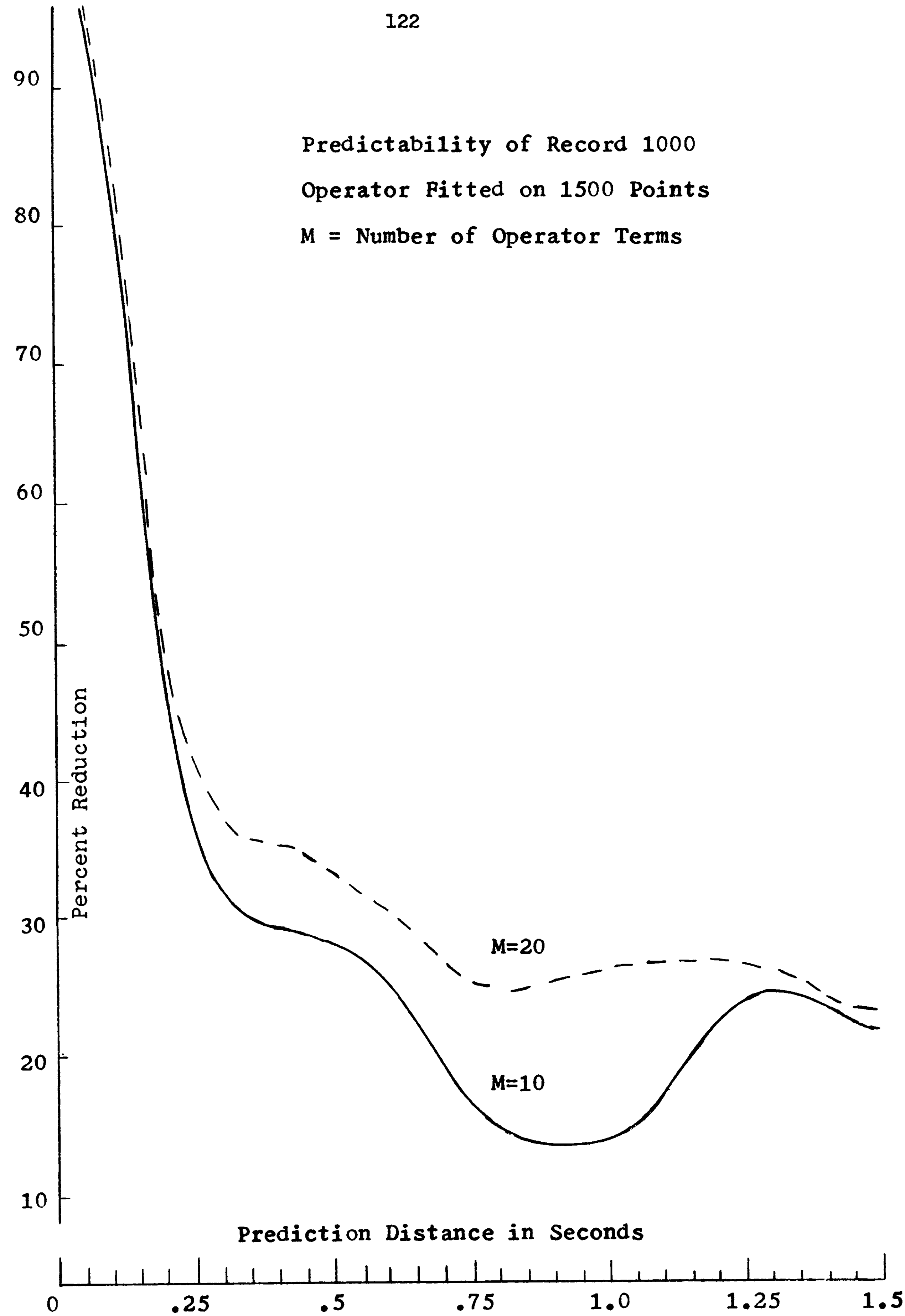

Predictability of Record 1000
Operator Fitted on 1500 Points
M = Number of Operator Terms
Percent Reduction
90
80
70
60
50
40
30
20
10
M=20
M=10
Prediction Distance in Seconds
0
.25
.5
.75
1.0
1.25
1.5

Figure 2.1.2

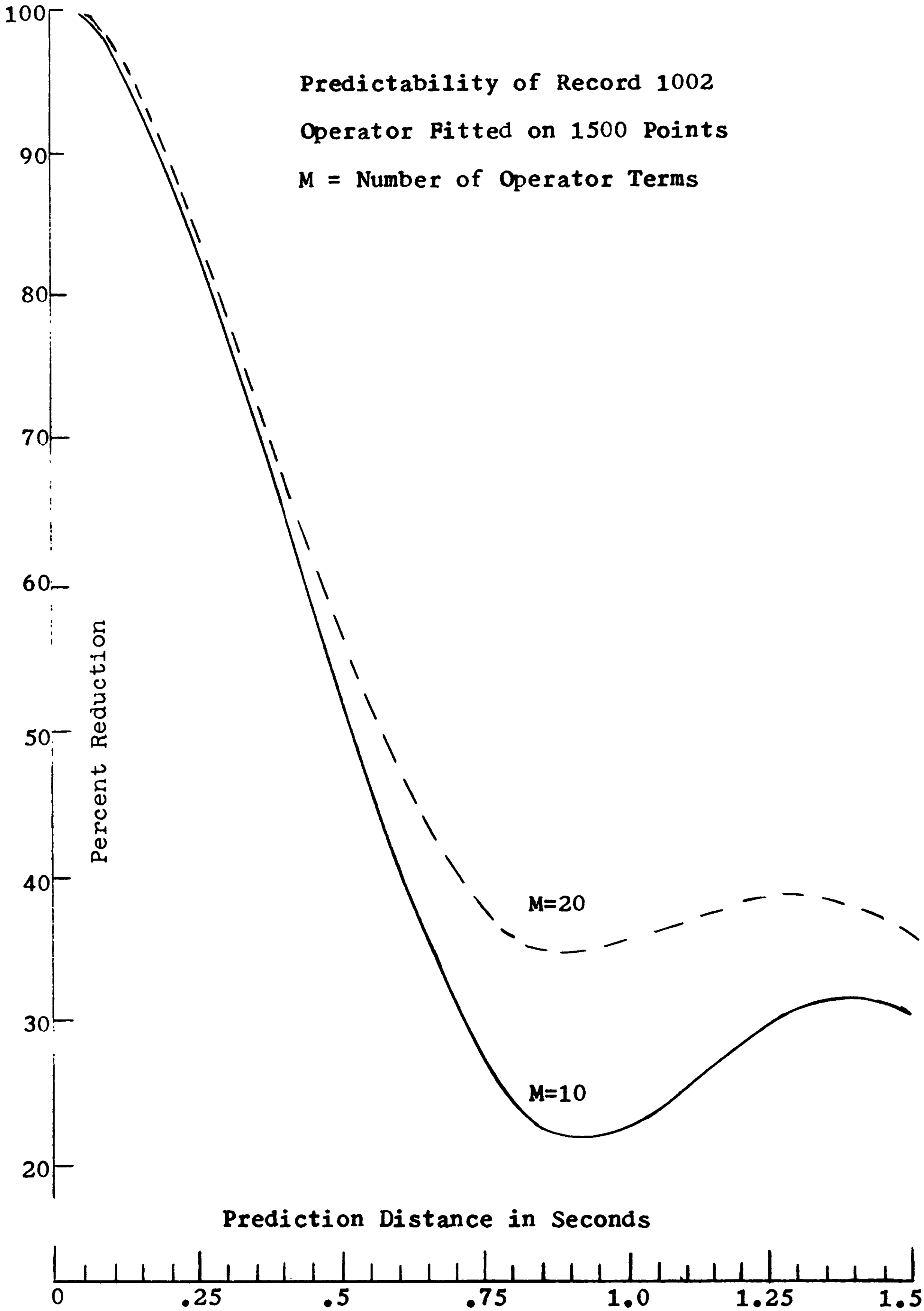

Predictability of Record 1002
Operator Fitted on 1500 Points
M = Number of Operator Terms
100
90
80
70
60
50
40
30
20
Percent Reduction
M=20
M=10
Prediction Distance in Seconds
0
.25
.5
.75
1.0
1.25
1.5

Figure 2.1.3

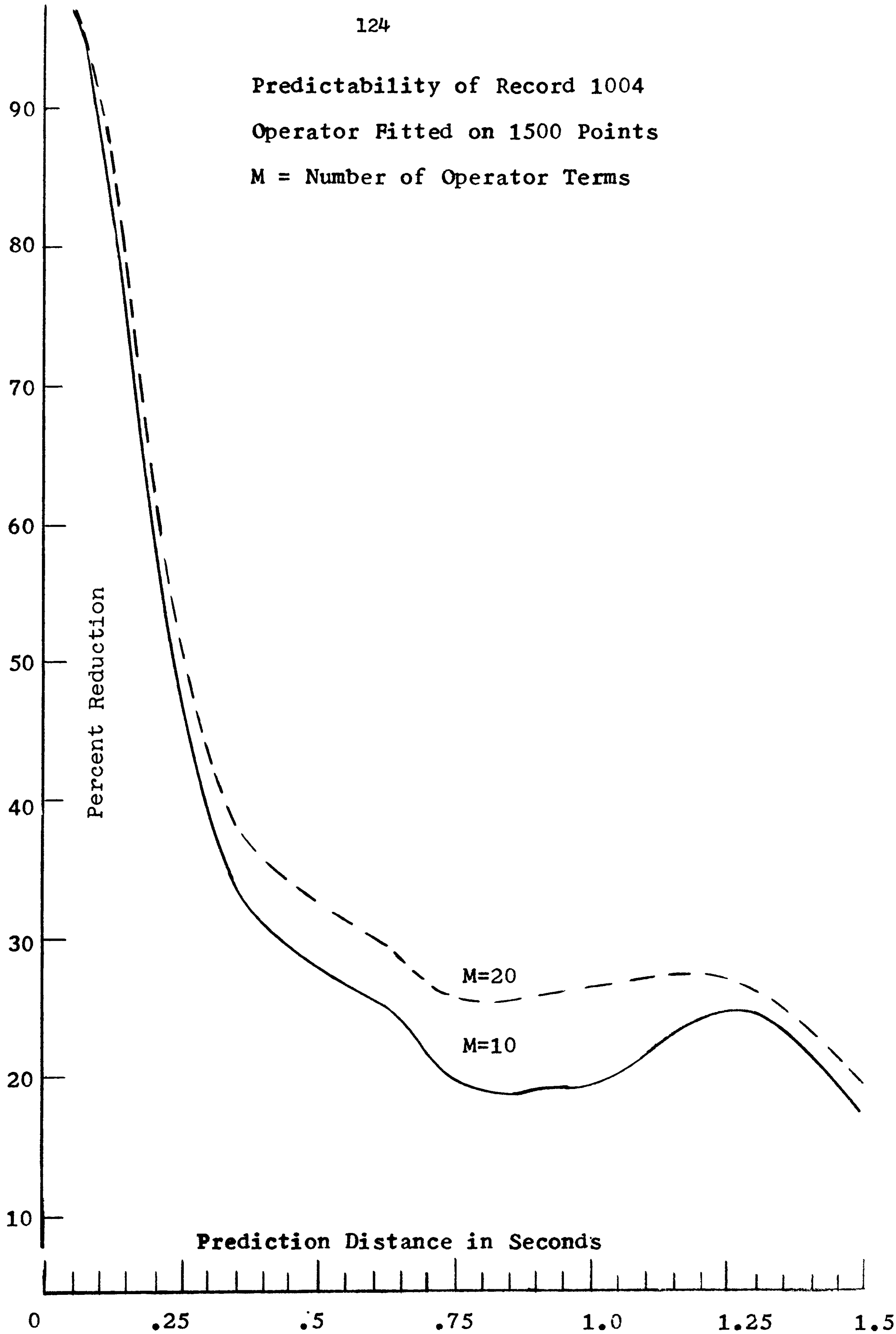
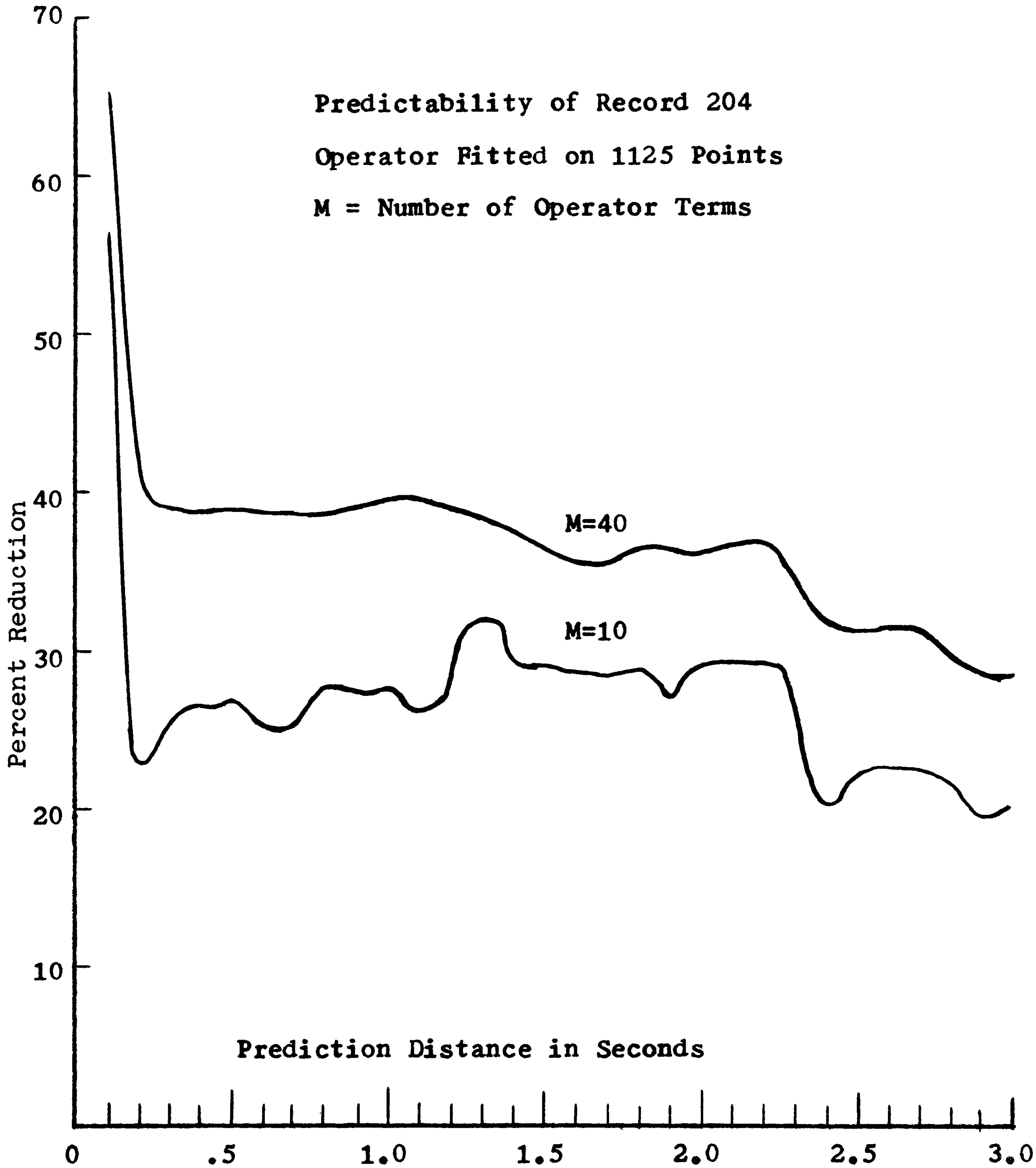

Predictability of Record 1004
Operator Fitted on 1500 Points
M = Number of Operator Terms
90
80
70
60
50
40
30
20
10
Percent Reduction
M=20
M=10
Prediction Distance in Seconds
0
.25
.5
.75
1.0
1.25
1.5

Figure 2.1.4



Predictability of Record 204
Operator Fitted on 1125 Points
M = Number of Operator Terms
M=40
M=10
Percent Reduction
70
60
50
40
30
20
10
Prediction Distance in Seconds
0
.5
1.0
1.5
2.0
2.5
3.0

Figure 2.1.5

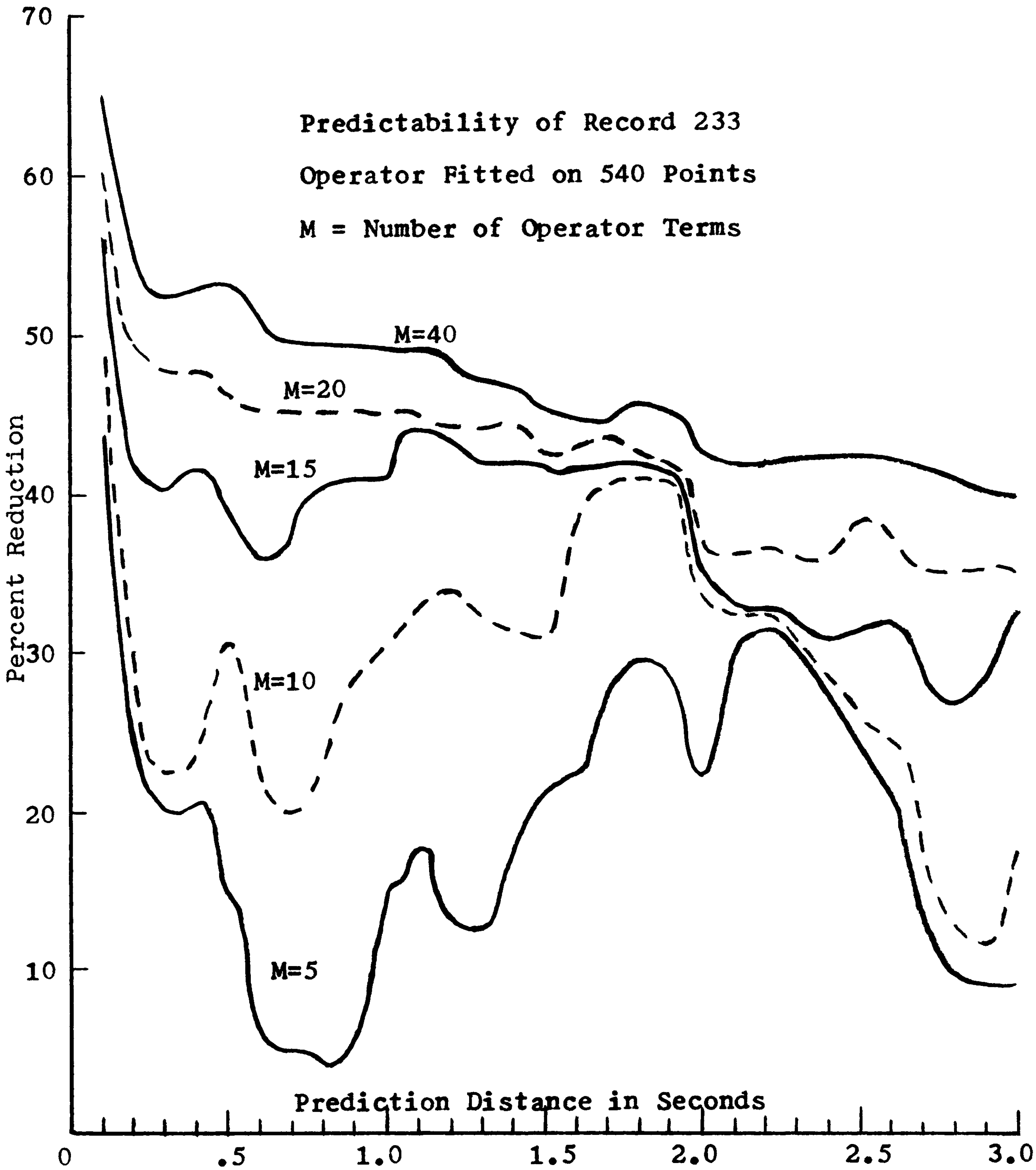
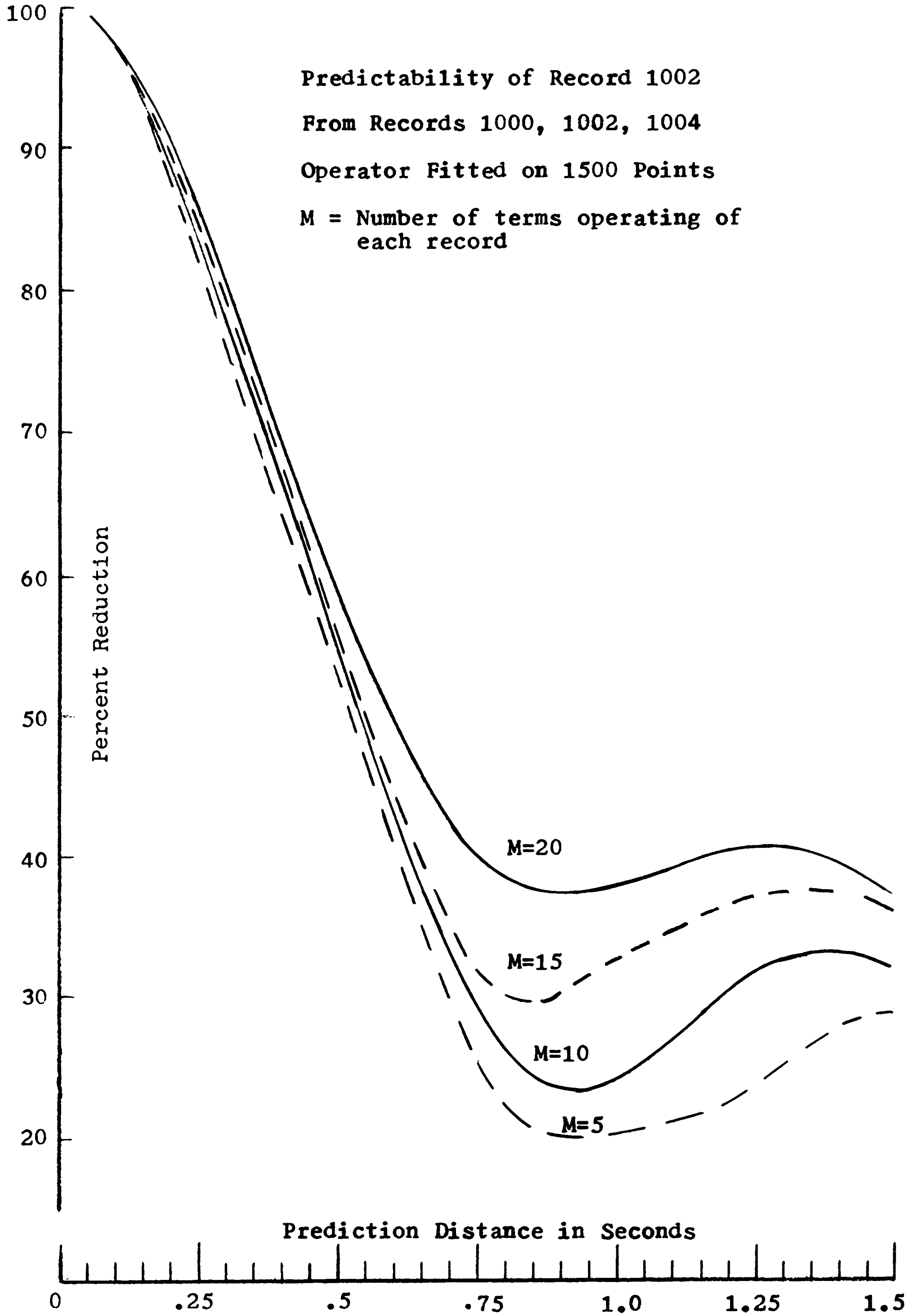

70
60
50
40
30
20
10
Percent Reduction
Predictability of Record 233
Operator Fitted on 540 Points
M = Number of Operator Terms
M=40
M=20
M=15
M=10
M=5
Prediction Distance in Seconds
0
.5
1.0
1.5
2.0
2.5
3.0

Figure 2.1.6



Predictability of Record 1002
From Records 1000, 1002, 1004
Operator Fitted on 1500 Points
M = Number of terms operating of each record
Percent Reduction
100
90
80
70
60
50
40
30
20
M=20
M=15
M=10
M=5
Prediction Distance in Seconds
0
.25
.5
.75
1.0
1.25
1.5

Figure 2.1.7

2.2 Prediction and Spectrum Factorization

Comparison of Prediction Techniques

We have seen in the last section that the mean squared error technique was not a practical method of prediction in the form in which it was used because of the large amount of computer space and time required. The program for prediction using the mean squared error technique was written almost entirely in FORTRAN and, due somewhat to the inefficiency of FORTRAN, the time required to obtain a 60 term self-prediction operator was about 10 minutes on the IBM 7090. The spectrum factorization method requires the spectrum as an input but the time needed to compute a 500 term wavelet is only 2 minutes on the 7090. Since the timing of both methods increases as the cube of the operator length, it is easy to see that there are tremendous advantages to the spectrum factorization method. The computation of the complete transient autocorrelation of 3000 data points and Daniell spectrum of 500 terms takes only about 2 minutes if high speed techniques are used (Simpson et al, 1961b). The Levenson (1949) technique has been programmed for the 709-7090 computers by Ralph Wiggins, but the work presented here was done before this program was available. The timing of the Levenson technique program increases as the square of the operator length but is about the same as the spectrum factorization program for a 500 term operator. The factorization method yields the minimum phase wavelet from which, as we shall see, the percent reduction can be obtained directly. The Levenson technique, on the other hand, gives the prediction operator directly, and we must compute this operator for unit prediction distance and invert it to obtain the wavelet. The choice between the two methods might well depend on whether one wants to actually do prediction or just find the percent reduction. An iteration technique for the multi-dimensional problem has been worked out by E. A. Robinson (personal communication), and it will be quite a bit faster than the three-dimensional technique described in the last section. The program for this has not been completed at the time of this publication.

Decomposition

The spectrum factorization method is much more fruitful than the mean squared error technique and the theory behind it is intimately related to the contents of section 1.4. In that section we showed that we could consider microseismic noise as a stationary ergodic time series and that, with a few additional considerations, we could assume that microseisms were generated by a white light (essentially independent) series convolved with a minimum phase wavelet. The importance of the minimum phase wavelet is that it is one sided, and therefore the expression for the present value of X_t , the microseismic noise, involves only the past values of ξ_t, the white light series. That is

$$X_t = \sum_{i=0}^{\infty} b_i \xi_{t-i}$$

where b_i is the minimum phase wavelet. We have seen that if b_i is known we can easily find a_i , the inverse minimum phase wavelet and can therefore write

$$\xi_t = \sum_{i=0}^{\infty} a_i X_{t-i} \qquad (2.2.1)$$

so that all the past ξ_t can be found from all the past X_t . We can therefore evaluate the expression for the minimum error for the mean squared error criterion (Robinson, 1954).

The minimum error is

$$I_{MIN} = E\left(X_{t+k} - \hat{X}_{t+k}\right)$$

where X_{t+k} is the true value of the series at time $t+k$, $\hat{X}_{t+k}$ is the predicted value, and the E means expected value. The true value is, from the above considerations,

$$X_{t+k} = \sum_{i=0}^{\infty} b_i \, \xi_{t+k-i} \tag{2.2.2}$$

But we know ξ_{t-i} from equation (2.2.1), so that the error in prediction must result from our lack of knowledge of ξ_{t+j} from $j=0$ to k . Since ξ_t are uncorrelated the best prediction we can do for them is to predict their mean, which is zero. Hence, our best prediction of X_{t+k} , $\hat{X}_{t+k}$, is given by equation (2.2.2) with $\xi_{t+k-i}=0$ for $t+k-i>t$. That is

$$\hat{X}_{t+k} = \sum_{i=k}^{\infty} b_i \, \xi_{t+k-i}$$

This has been shown to be true by Wold (1938), (Robinson, 1954).

<u>Minimum Error and Percent Reduction in Terms of the Wavelet</u>

The minimum error is, therefore,

$$I_{MIN} = E\left[\sum_{i=0}^{\infty} b_i \xi_{t+K-i} - \sum_{i=K}^{\infty} b_{i-K} \xi_{t+K-i}\right]^2$$

$$= E\left[\sum_{i=0}^{K-1} b_i \xi_{t+K-i}\right]^2$$

$$= \sum_{i=0}^{K-1} b_i^2 E\left[\xi_t\right]^2$$

If the expected value of ξ_t^2 is one

$$I_{MIN} = \sum_{i=0}^{K} b_i^2$$

and we see that the minimum error and hence the percent reduction decreases monotonely with increasing prediction distance K. We can now easily obtain an expression for the percent reduction, R_p, in terms of b_i. We recall that

$$R_p = 100\left(1 - \frac{I_{MIN}}{I_0}\right)$$

where I_0 is the variance of the sample,

$$I_0 = E\left[X_t\right]^2 = E\left[\sum_{i=0}^{\infty} b_i \xi_{t-i}\right]^2$$

$$= \sum_{i=0}^{\infty} b_i^2 E\left[\xi_t\right]^2$$

Hence

$$R = 100\left(1 - \frac{\sum_{i=0}^{K-1} b_i^2}{\sum_{i=0}^{\infty} b_i^2}\right)$$

where we have made no assumptions regarding the value of $E(\xi_t)^2$

Thus we see that if b_i is known we can find the value of R_p for all K without actually computing the prediction, or even the prediction operator. We saw in section 1.4 that it is possible to find b_i, and the process is called spectrum factorization. The derivation of the b_i from the power spectrum is given in Appendix E. We see also in Appendix E that it is possible to find the first M terms exactly. This procedure has been programmed for the IBM 709 and 7090 computers, and the program listing, FACTOR, appears in Appendix G. Appendix E also explains most of the program logic.

We note that the expression for I_0 requires all of the b_i and the program will only give us the first M. For long operators this is not troublesome since the wavelet dies off fairly rapidly. However, the estimate of I_0 using just M terms will be a bit small, and therefore the value of R_p will be a bit small. We could, of course, estimate I_0 from the data without using the b_i since I_0 is just the variance,

$$I_0 = \frac{1}{N}\sum_{i=0}^{N-1}(x_i - \bar{x})^2$$

where the mean is zero.

The computation of the minimum phase wavelet, b_i, has been done for 500 terms and the corresponding percent reductions are shown in Figures 2.2.1 to 2.2.6. Included also are some of the minimum phase wavelets and some of the inverse wavelets (Figures 1.4.1 to 1.4.5). The minimum phase wavelets for all the records are quite similar, so it is not necessary to include all of the graphs.

The percent reductions are now, of course monotonely decreasing and are forced to zero at t = 25 seconds (not shown in graphs) because is computed from the first 500 terms (25 seconds). Comparison of these figures with the self-prediction of section 2.1 (Figures 2.1.2 to 2.1.4) shows a marked increase in predictability using this technique, as much as 10 in the percent reduction, but the increase is still not large enough to improve the signal to noise ratio in the first motion interval by a significant amount. Comparison of the estimate of I_0 from the 500 term wavelet with the sample variance estimated from 3000 data points indicates that the percent reductions obtained are off by less than one.

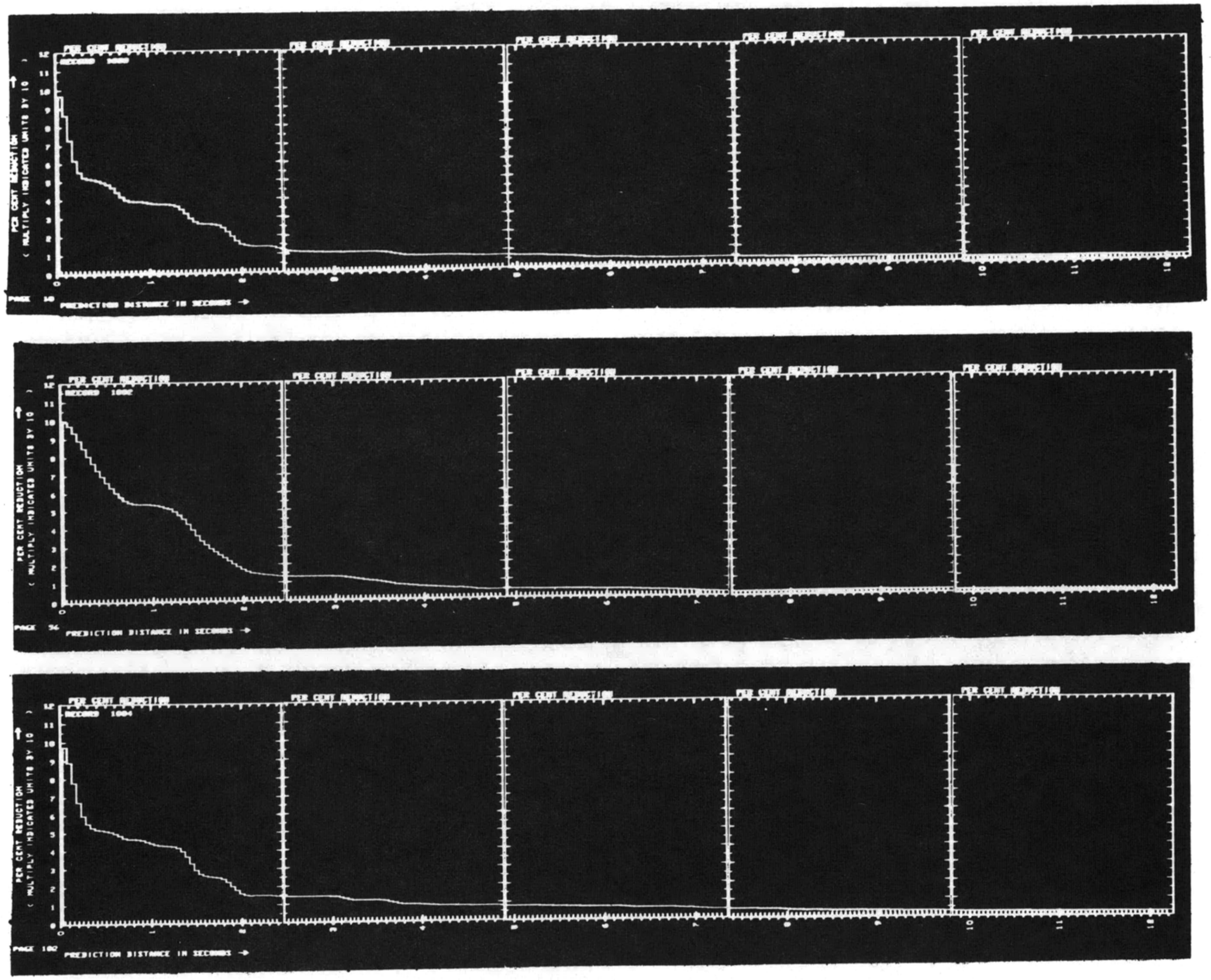


Figure 2.2.1 Percent reductions for prediction distances up to 12 seconds for records 1000, 1002, 1004.

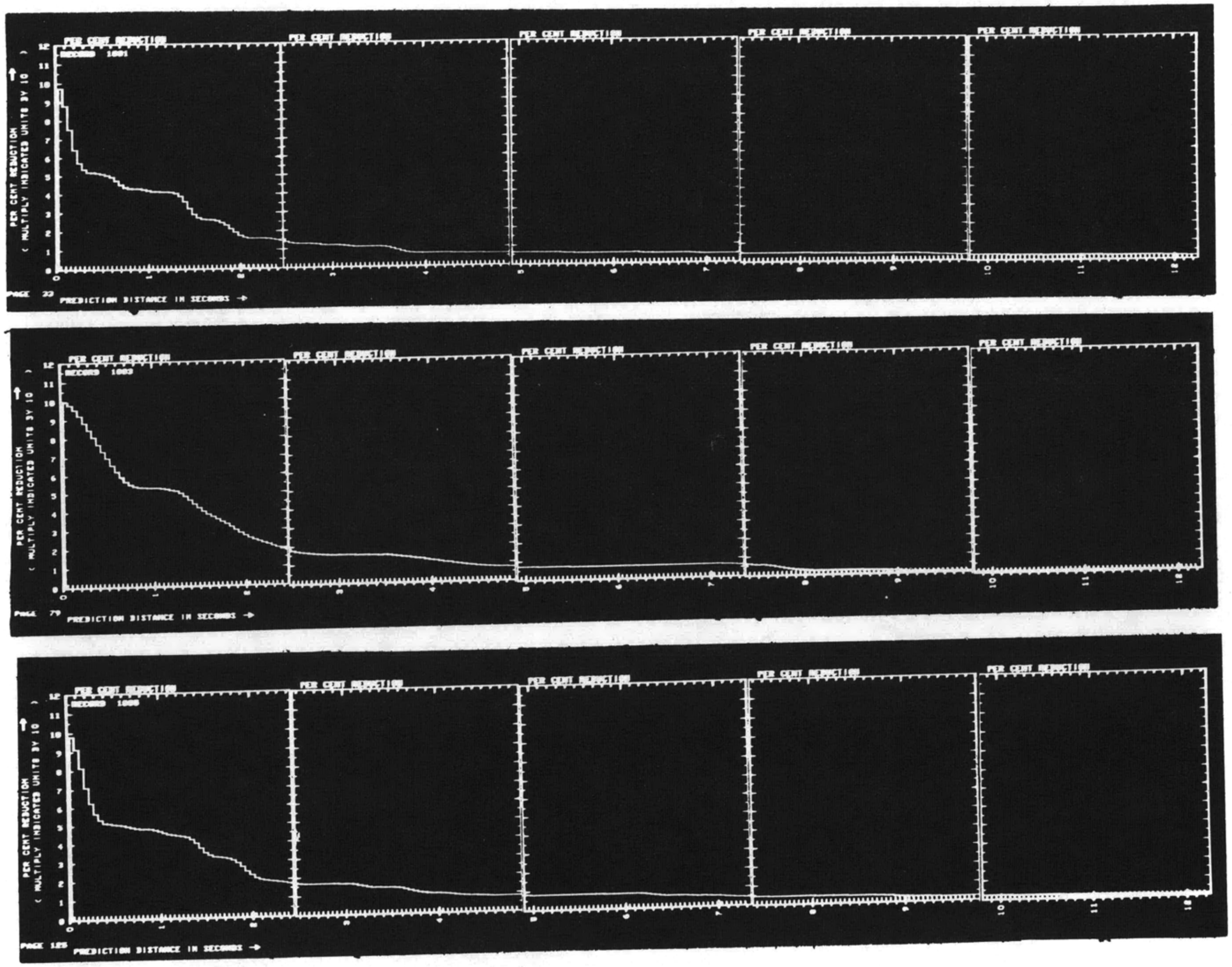


Figure 2.2.2 Percent reductions for prediction distances up to 12 seconds for records 1001, 1003, 1005.

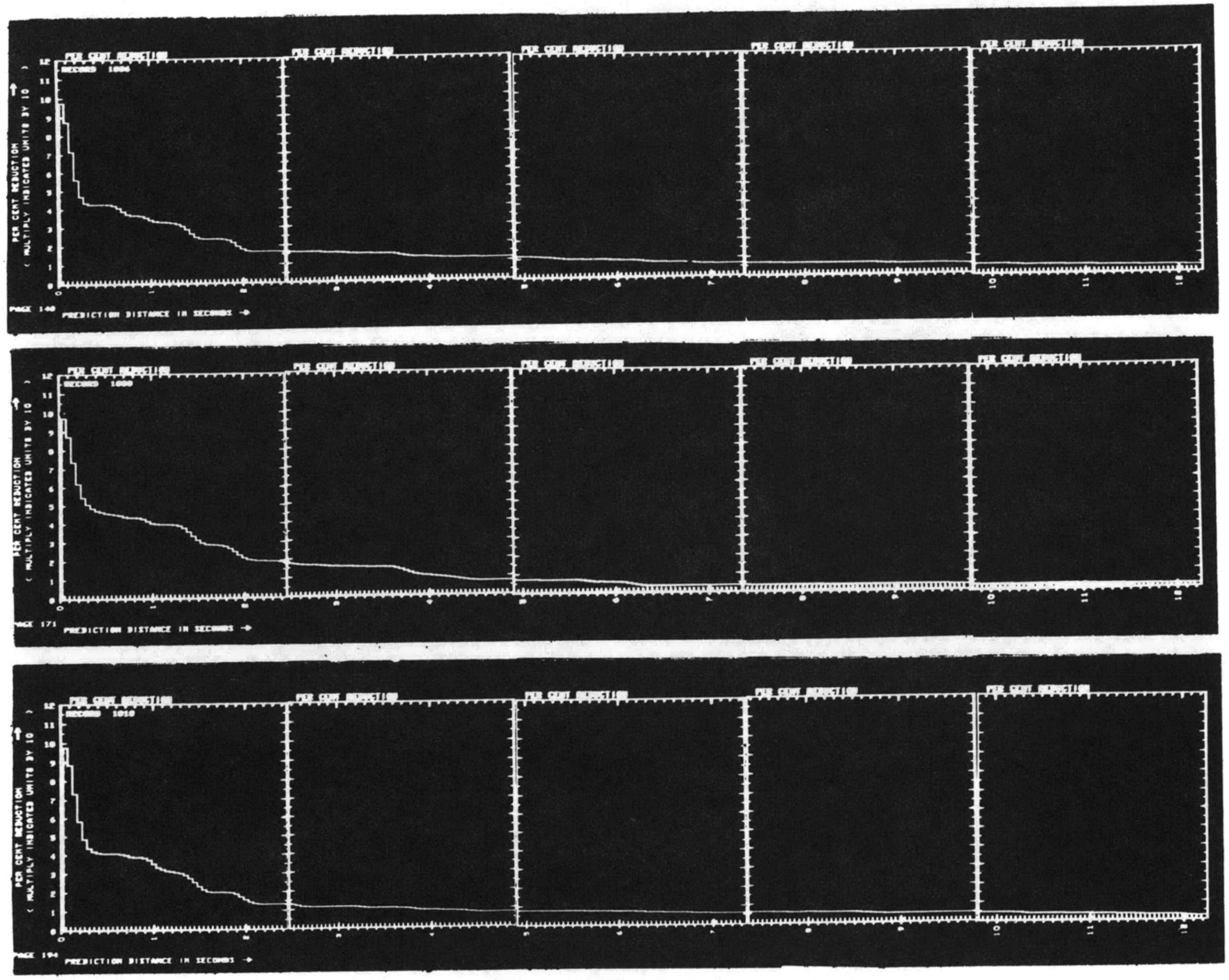


Figure 2.2.3 Percent reductions for prediction distances up to 12 seconds for records 1006, 1008, 1010.

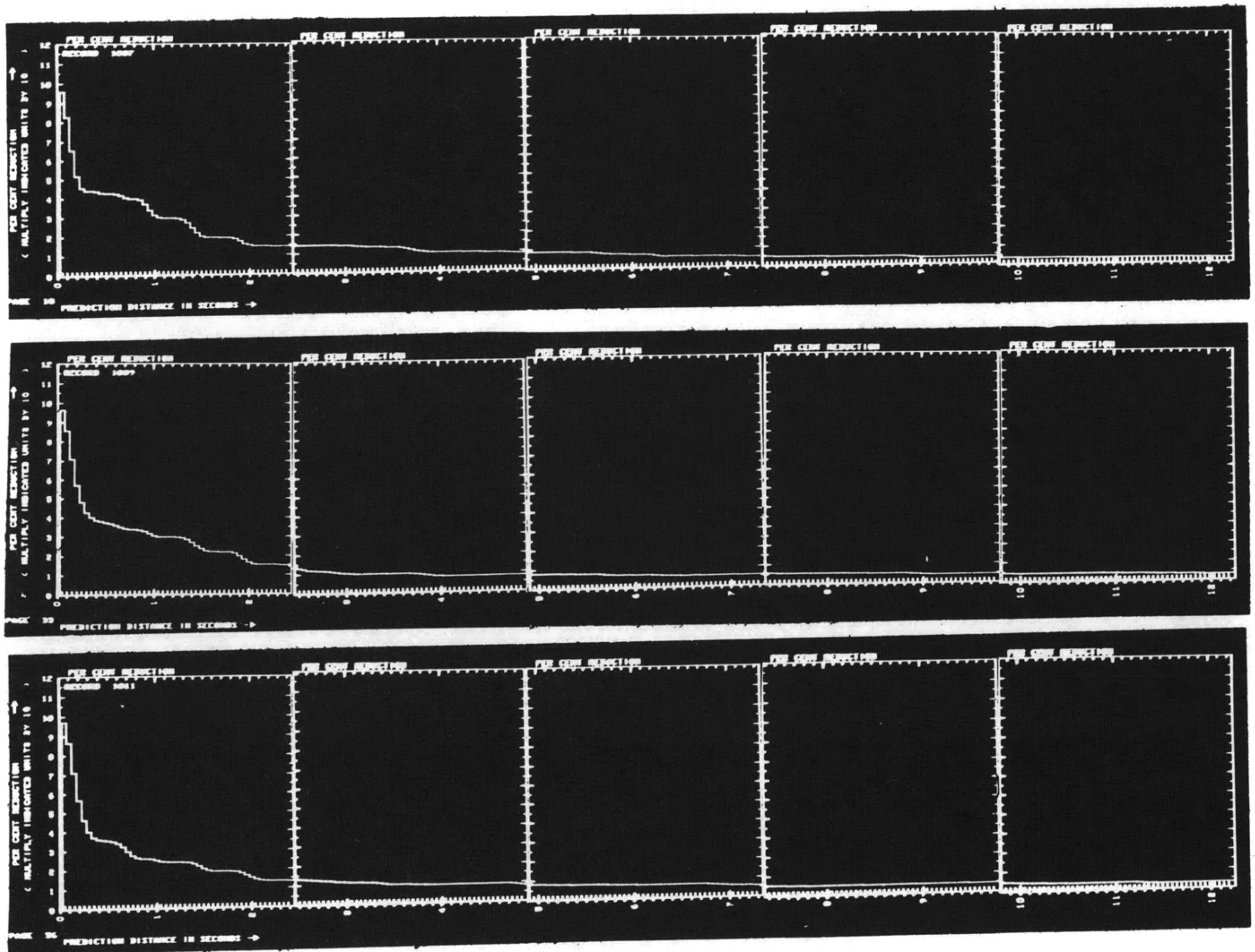


Figure 2.2.4   percent reductions for prediction distances up to 12 seconds for records 1007, 1009, 1011.

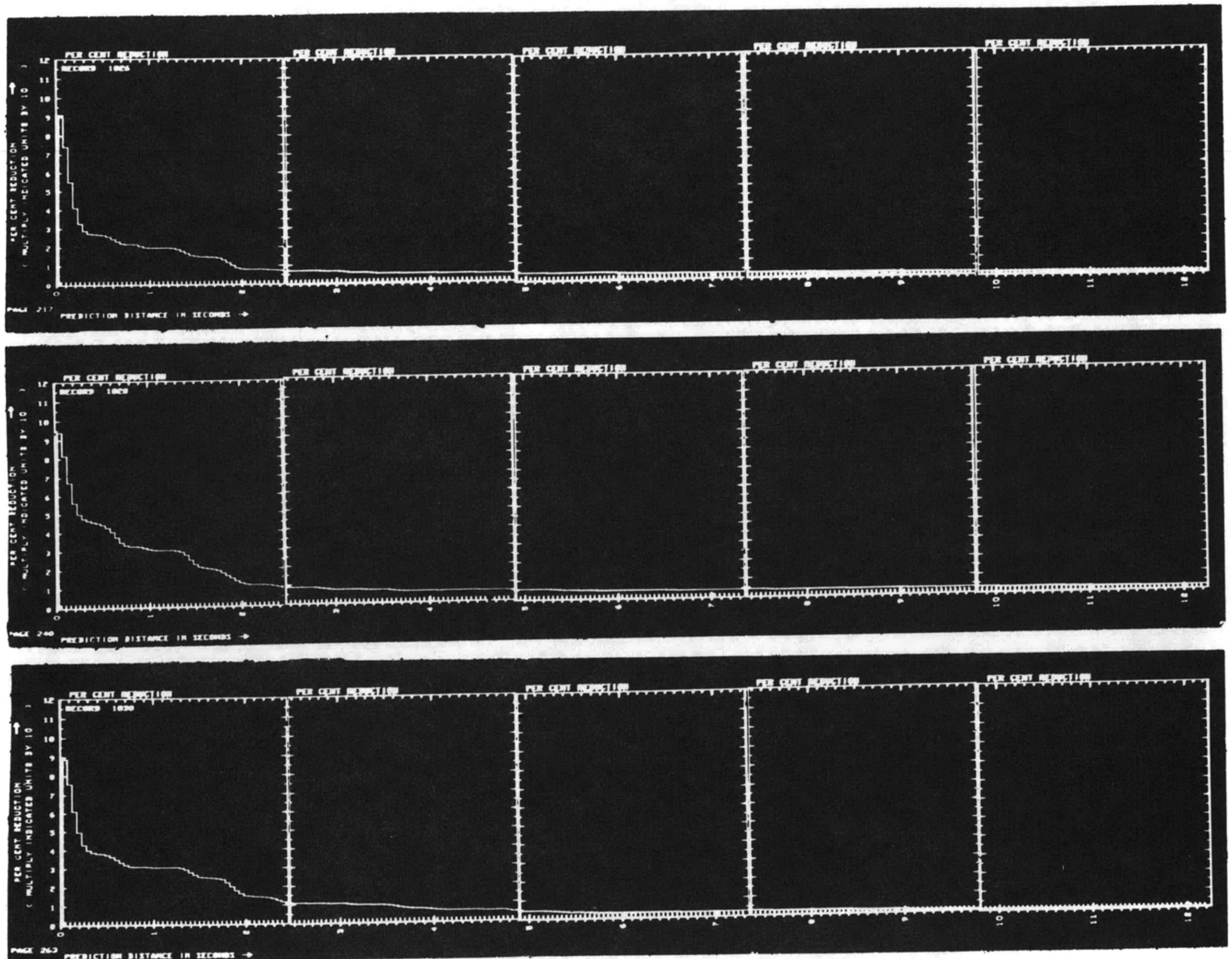


Figure 2.2.5 Percent reductions for prediction distances up to 12 seconds for records 1026, 1028, 1030.

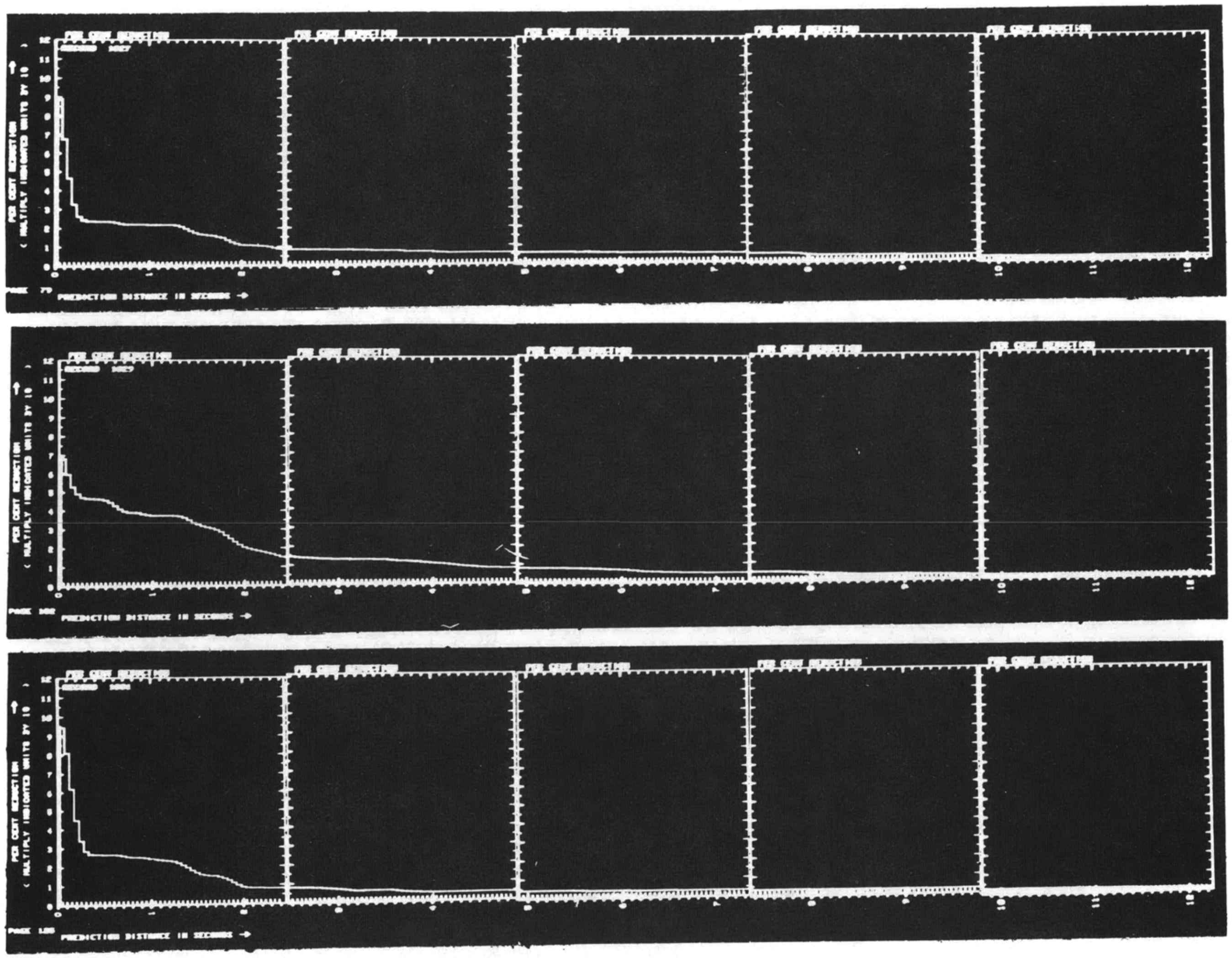


Figure 2.2.6 Percent reductions for prediction distances up to 12 seconds for records 1027, 1029, 1031.

2.3 Summary Comments on Prediction

We have seen in the last two sections that the optimum least squares prediction for short operators and for one and three dimensions are not good enough to improve the signal to noise ratio significantly. Further, we saw that the best predictions possible using the wavelet obtained by spectrum factorization did not yield results of any consequence. The fact that we only had 500 terms of the infinite wavelet is not important since the estimate of the standard deviation using the 500 terms was quite good (within 0.1 percent). We have alternatives of increasing the operator length of the three dimensional prediction, of going to non-linear prediction models, or, of course, of rejecting the technique of prediction of the microseisms in the first motion interval as a useful method of improving the signal to noise ratio. The first alternative, increasing the operator length for the three-dimensional case, does not seem worth trying. The improvement in predictability of the three-dimensional case, over self prediction was seen to be minescule. Further, the improvement of predictability of long operators over short was not significant. We therefore reject the first alternative.

Independence of White Light Series

It is possible, also, to reject the second alternative, that of non-linear prediction models. We saw, in section 1.4, in the decomposition of the microseisms to a white light series and a minimum phase wavelet, that the white light series could be considered purely random. That is, the ζ_t were not only uncorrelated, but also statistically independent.

From elementary probability considerations we have

$$P_{\xi_1 \xi_2}(x_1, x_2) = P_{\xi_1}(x_1) P_{\xi_2|\xi_1}(x_2|x_1)$$

The joint probability of ξ_1 and ξ_2 is equal to the marginal probability of ξ_1 times the conditional probability of ξ_2 given ξ_1. If ξ_1 and ξ_2 are independent

$$P_{\xi_1 \xi_2}(x_1, x_2) = P_{\xi_1}(x_1) P_{\xi_2}(x_2) \quad ; \quad P_{\xi_2|\xi_1}(x_2|x_1) = P_{\xi_2}(x_2)$$

We can repeat this for many ξ_i and obtain

$$P_{\xi_{n+1}|\xi_1 \xi_2 \dots \xi_n}(x_{n+1}|x_1, x_2, \dots x_n) = P_{\xi_{n+1}}(x_{n+1})$$

Thus from the definition of independence we see that the knowledge of $\xi_1, \xi_2, \dots \xi_n$ give no information about ξ_{n+1}. In a prediction problem where $\xi_1, \xi_2, \dots \xi_n$ are the past values and ξ_{n+1} the future values of a time series and the ξ_i, $i = 1$ to n are independent, we have no information about ξ_{n+1} except its probability density $P_{\xi_{n+1}}(x_{n+1})$ which we know from the assumption of stationarity. Any prediction scheme using any of the ξ_i, $i = 1$ to n will avail us nought, but $P_{\xi_{n+1}}(x_{n+1})$. The best least squares prediction which one can do in the case of independence is to predict the expected value of ξ_{n+1}, the mean, which a linear predictor can do. Therefore, if random noise can be considered as an independent white light series convolved with a minimum phase wavelet, the best prediction one can do is linear prediction, since the non-linear predictor will only bring in higher order correlations which give no new information.

Weiner (1946) states that linear prediction is optimum in the case where the noise series can be reduced to a Gaussian white light series by convolution with a operator. The reason for this can be seen from the following analysis of the joint probability density for independent and dependent variables.

Independence and Gaussian White Light - Example

Let ξ_1 and ξ_2 be normally distributed independent random variables. Then the joint density of ξ_1 and ξ_2 is

$$P_{\xi_1 \xi_2}(x_1, x_2) = P_{\xi_1}(x_1)\, P_{\xi_2}(x_2) = \frac{1}{2\pi \sigma_1 \sigma_2} \exp\left[-\frac{x_1^2}{2\sigma_1^2} - \frac{x_2^2}{2\sigma_2^2}\right]$$

where σ_i is the standard deviation of ξ_i. Now we define y_1 and y_2 as a linear combination of x_1 and x_2

$$\begin{aligned} y_1 &= a x_1 + b x_2 \\ y_2 &= c x_1 + d x_2 \end{aligned} \qquad (2.3.1)$$

and therefore

$$P_{\eta_1 \eta_2}(y_1, y_2)\, dy_1\, dy_2 = P_{\xi_1 \xi_2}(x_1, x_2)\, dx_1\, dx_2$$

or

$$P_{\eta_1 \eta_2}(y_1, y_2) = |J|\, P_{\xi_1 \xi_2}(x_1, x_2)$$

where $|J|$, the magnitude of the Jacobian for this transformation, is

$$J = ad - bc$$

Solving (2.3.1) for x_1 and x_2:

$$x_1 = \frac{d}{J} y_1 - \frac{b}{J} y_2$$

$$x_2 = \frac{a}{J} y_2 - \frac{c}{J} y_1$$

Hence joint density for the dependent variables η_1 and η_2 is

$$P_{\eta_1 \eta_2}(y_1, y_2) = \frac{|J|}{2\pi\sigma_1\sigma_2} \exp\left[-\left(\frac{\sigma_1^2 d^2 + \sigma_2^2 c^2}{2\sigma_1^2\sigma_2^2 J^2}\right) y_1^2 - \left(\frac{\sigma_1^2 a^2 + \sigma_2^2 b^2}{2\sigma_1^2\sigma_2^2 J^2}\right) y_2^2 + \left(\frac{\sigma_2^2 bd + \sigma_1^2 ac}{\sigma_1^2\sigma_2^2 J^2}\right) y_1 y_2\right]$$

We note the expected values of the following quantities.

$$\mu_1 = E(y_1^2) = a^2\sigma_1^2 + b^2\sigma_2^2$$

$$\mu_2 = E(y_2^2) = c^2\sigma_1^2 + d^2\sigma_2^2$$

$$\mu_{12} = E(y_1 y_2) = ac\,\sigma_1^2 + bd\,\sigma_2^2$$

Thus

$$P_{\eta_1 \eta_2}(y_1, y_2) = \frac{|J|}{2\pi\sigma_1\sigma_2} \exp\left[\frac{-\mu_1 y_1^2 - \mu_2 y_2^2 + 2\mu_{12} y_1 y_2}{2\sigma_1^2\sigma_2^2 J^2}\right]$$

If μ_{12}, the correlation of y_1 and y_2, is zero, the cross term in the exponential is zero and $P_{\eta_1\eta_2}(y_1, y_2)$ factors. This can be extended for $P_{\eta_1\eta_2\cdots\eta_n}(y_1, y_2 \ldots y_n)$ and we see that in general if the correlation coefficients are zero the joint density of n variables factors. Hence for the Gaussian, linear independence implies statistical independence. (Davenport and Root, 1950).

Non-Linear Operators

We thus see the reason behind Wiener's statement that linear prediction is optimum if it reduces the series to Gaussian White light. We need actually only show, therefore, that the white light series, ξ_t is Gaussian in order to reject the adoption of a non-linear predictor. We saw in section 1.4 that, for microseisms, ξ_t was Gaussian in many cases, and was in general nearly Gaussian. We can fall back on the independence tests for these non-Gaussian cases which showed that we could consider ξ_t independent. The independence of ξ_t forces us to drop the notion of non-linear prediction and hence forces us to reject the technique of prediction for signal to noise ratio improvement in the first motion interval.

3. AUTOMATIC DETECTION OF SIGNALS IN MICROSEISMIC NOISE

3.1 Detection System

Description - Inputs and Outputs

A detection system to automatically detect signals in microseismic noise has been designed and a computer program has been written to simulate the system. The system and programs have been developed by S. M. Simpson, Jr., for Geoscience, Inc. A flow chart of the computer simulation of the system appears in Figure 3.1.1. The signal plus noise input is rectified by squaring or by taking the absolute value and this rectified waveform is averaged. The averaged rectified wave form then enters a network which decides if there is a signal present or not, and sets an alarm if there is a signal. The system variables are the type of rectification, the averaging time, the hesitation time and the alarm level. The averaging time is the length of time over which the rectified waveform is averaged before going to the decision network. Averaging over some length of time is necessary to reduce false alarms due to an occasional high noise amplitude, but the length must not be much greater than the expected length of the signal, since the average would be too small to trigger the alarm. The hesitation time is the length of time that the rectified averaged input must remain above the alarm level before an alarm is sounded. This also tends to cut down alarms which might be caused by noise spikes. The alarm level is the ratio of the value which averaged rectified wave must reach for an alarm to the r.m.s. amplutide of the noise.

It is, therefore, the signal to noise ratio at which the system can operate. For example, if the alarm level is 1.75, an alarm will not be sounded until the average rectified waveform reaches 1.75 times the r.m.s. noise amplitude.

The system as it stands is an event detector. It tells whether or not an event has occurred, but makes no statement as to the nature of the signal which triggered the alarm. Such a system could be used in an automatic nuclear surveilance network to control the collection of data. Only data near the time of an alarm would be recorded, and these alarms could be studied for source type. An alternate procedure would be to collect all data and just study the portions corresponding to alarms.

In order to rate the effectiveness of this system, it is necessary to study the false alarm rate and failure to detect rate as a function of the system parameters. The next few sections give the results of false alarm and failure rate studies on the computer simulated system for raw and filtered signals and noise.

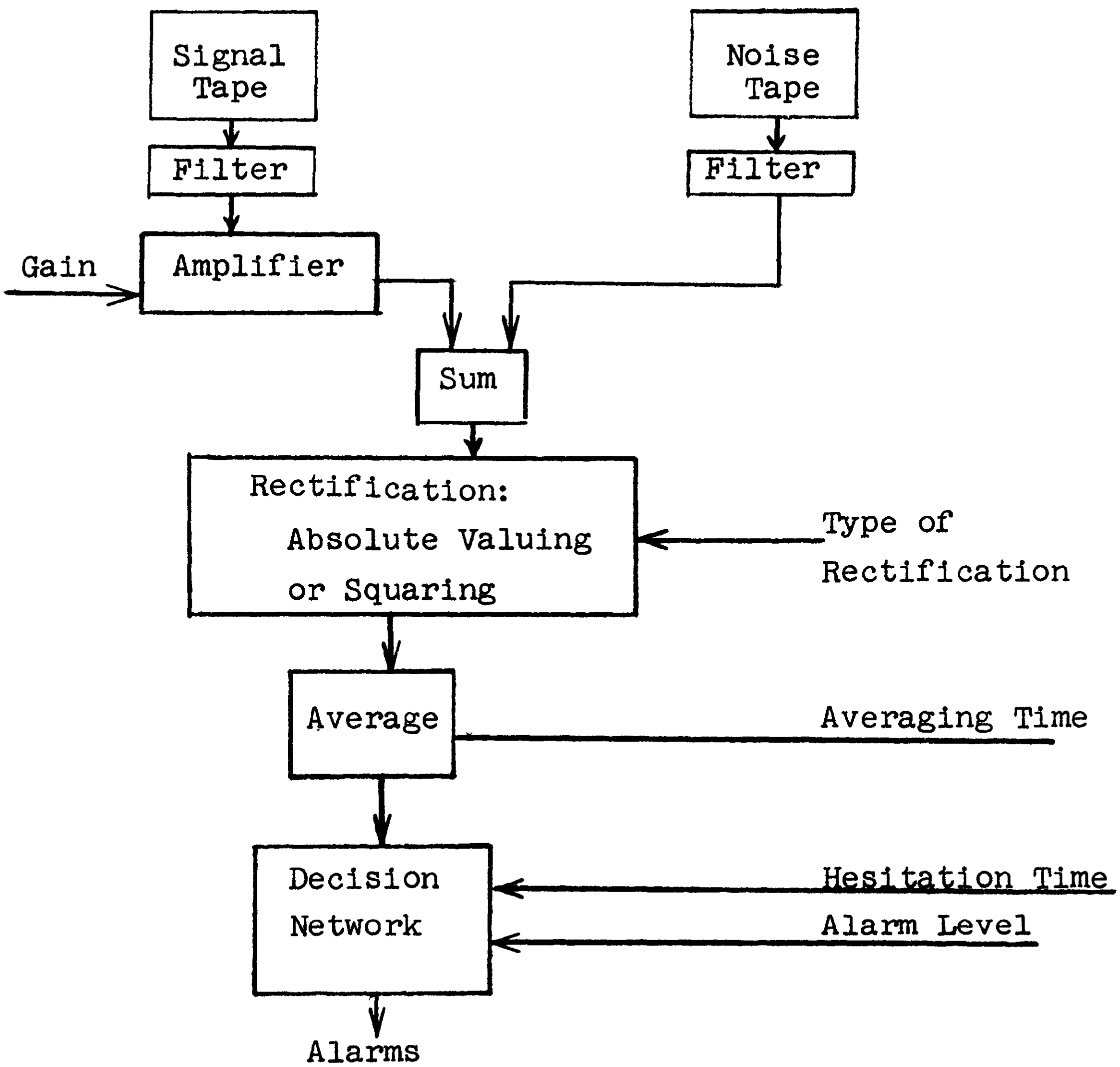


Figure 3.1.1 Computer Simulation Flow Chart

3.2 False Alarm Rate - FALARA

Generation of Input Noise

The false alarm rate of the detection system can be obtained by using a pure noise input rather than a signal plus noise input and counting the number of times an alarm is sounded as a function of the system parameters. A large amount of noise representing many hours of sequencial microseisms is necessary to carry out the study. Since only a few minutes of consecutive microseismic noise is available from our digitized noise library, the microseisms must be generated artificially. We have seen in section 1.4 that this could be done to a good approximation using a minimum phase wavelet from real data and Gaussian white noise. Thus, the artificial microseisms, X_t , shown in the upper trace of Figure 1.4.16, are generated by the convolution

$$X_t = \sum w_i \xi_{t-i}$$

where w_i is the wavelet and ξ_t is the Gaussian white noise. The wavelet used in these studies was computed from record 1002, the vertical component of the noise before the Logan shot 1902 km from the shot point. The Gaussian white noise is generated from the Rand random digits by summing non-overlapping groups of ten digits. The central limit theorem tells us that the resulting sequence will have an approximately normal distribution.

A 500 term minimum phase wavelet was computed and every other point was then deleted. This left a 250 point wavelet with an equivalent digitization rate of 10 points per second. The deletion is not unreasonable since there is almost no power above 5 cps. This wavelet was then convolved with 85,249 points of Gaussian white noise to yield 85,000 points of artificial microseisms which correspond to 2.22 hours of noise.

False Alarm Rate Studies

The computer program FALARA (FAlse Alaram RAte) has been written by S. M. Simpson to simulate the detection system with pure microseismic noise input. For each set of system parameters the simulation was continued until either 100 alarms were sounded or all 85,000 points of noise were used. A flow chart of the simulation for the false alarm rate is shown in Figure 3.2.1 along with the system parameters used. As can be seen from this figure, two different types of rectification were used with five averaging times, ten alarm levels and five hesitation times. The false alarm rate is computed in units of alarms per hour. The results are shown in Figures 3.2.2 and 3.2.3 where the false alarm rate is plotted against the alarm level for several averaging times and for both types of rectification. Each figure is for a different hesitation time. Curves are included for only part of the results, but these are sufficient to indicate over-all trends in the system.

It is obvious that a desirable system should have very few false alarms for a low alarm level. We see from the figures that the curves with both low false alarm rate and low alarm level are relatively insensitive to hesitation time. For a given hesitation time the curves show that a long averaging time is desirable. These qualitative results are just as expected. The noise amplitudes change fairly rapidly and the high noise values, which are of short duration, are what trigger the alarm. Consequently the curves for short averaging time are affected by the hesitation time whereas the curves for long averaging time are only slightly changed. We note that for given averaging and hesitation times the curves for rectification by squaring are always better. We also see that the curves for high averaging times are fairly close together, which indicates that very little improvement will be obtained with averaging times greater than 10 seconds.

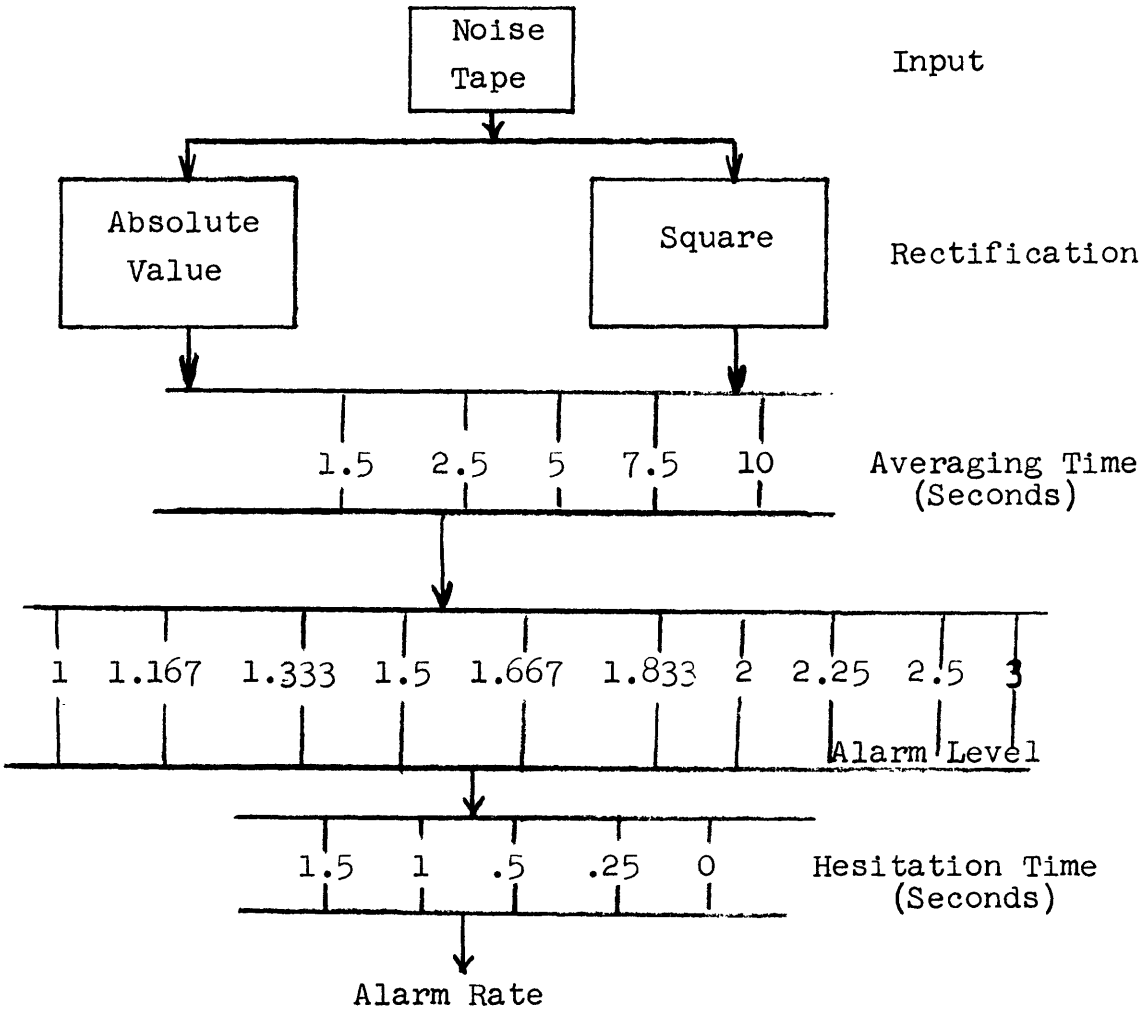


Figure 3.2.1 False Alarm Rate Flow Chart

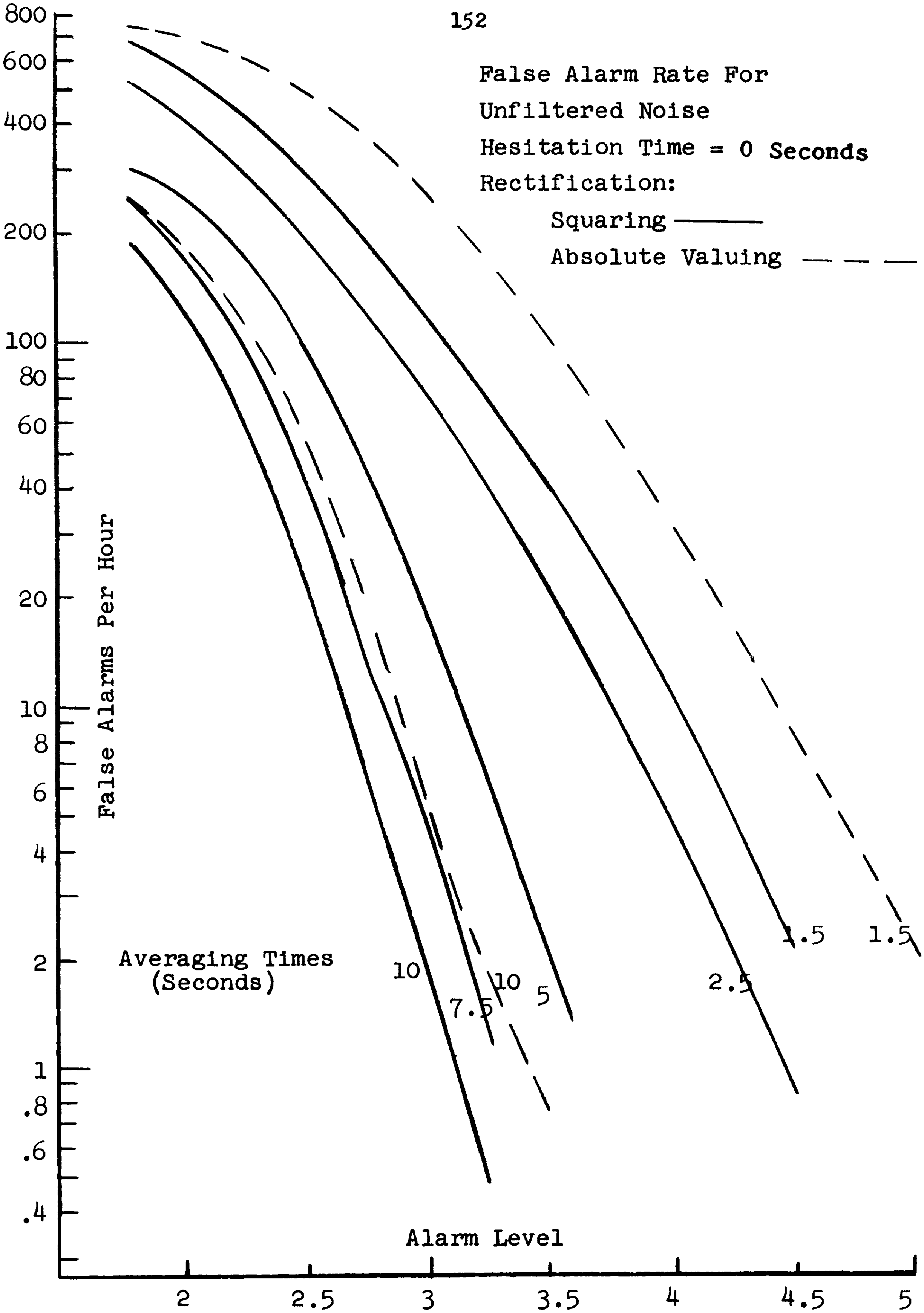
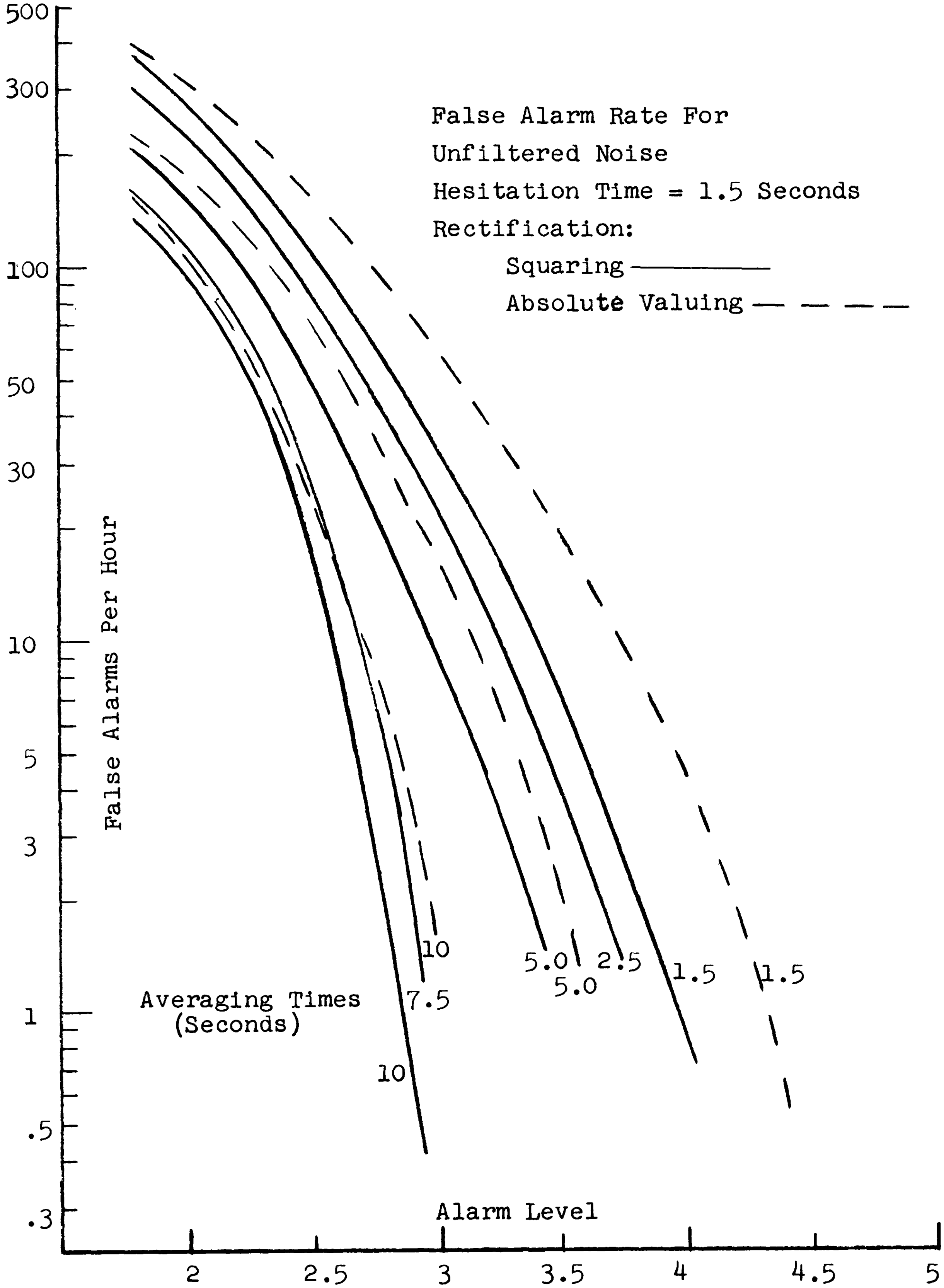

False Alarm Rate For
Unfiltered Noise
Hesitation Time = 0 Seconds
Rectification:
Squaring
Absolute Valuing
False Alarms Per Hour
800
600
400
200
100
80
60
40
20
10
8
6
4
2
1
.8
.6
.4
Averaging Times
(Seconds)
10
7.5
10
5
2.5
1.5
1.5
Alarm Level
2
2.5
3
3.5
4
4.5
5

Figure 3.2.2



500
300
100
50
30
10
5
3
1
.5
.3
False Alarms Per Hour
False Alarm Rate For
Unfiltered Noise
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
10
7.5
10
5.0
5.0
2.5
1.5
1.5
Averaging Times
(Seconds)
Alarm Level
2
2.5
3
3.5
4
4.5
5

Figure 3.2.3

3.3 Failure Rate - FAILRA

Description of System

The failure rate of the detection system is somewhat more difficult to obtain than the false alarm rate. Both signal and noise are required along with several signal to noise ratios. In the simulation of the system, the signal, scaled to give the required r.m.s. signal to noise ratio, and a block of noise are added together to give the input waveform. This is rectified and averaged and sent to the decision network where the alarm is announced if triggered. Figure 3.3.1 shows a flow chart of the computer program FAILRA (FAILure RAte), written by S. M. Simpson, with the system parameters used to obtain the failure rate.

The artificial microseismic noise used for the false alarm rate determination was used for the failure rate studies. For the signal it was necessary to pick out a representative bomb record with a fairly high signal to noise ratio so that the noise occurring with the signal was negligible compared to the microseismic noise added later. The record chosen was the vertical component of the signal from the Blanca shot recorded at 1398 km from the shot point (record 58, see Figure 3.3.2). Every other point of the first 600 points of this record were used thus giving 30 seconds of signal. The signal to noise ratios used were 1.78, 2.07, 2.37, 2.67, 2.97, 3.26, 3.56, 4.0, 4.45 and 5.34.

Failure Rate Studies

The system simulation was carried out for a hesitation time 1.5 seconds, both types of rectification, five averaging times, ten alarm levels and all above signal to noise ratios. For each set of system parameters the detection was tried 101 times and the number of successes and failures noted. In graphs showing the results, Figures 3.3.2 and 3.3.3, the success probability is plotted against alarm level for different averaging times. Each figure gives the curves for a different signal to noise ratio. The complete set of results is not given since the success probabilities for signal to noise ratios greater than 3.26 are nearly all equal to one.

The curves show that the long averaging times are successful over a smaller range of alarm levels than the short averaging times for a given signal to noise ratio, and they stop being successful at an alarm level approximately equal to the signal to noise ratio. This is not surprising since the long averaging time will average the signal alarm but the short averaging time will permit high amplitude pulses to trigger an alarm.

The wider range of success for short averaging times is offset by the unavoidably large false alarm rate which was noted in the last section. The most generally effective system parameters must balance the false alarm rate and the failure rate. In Figure 3.3.4 the overall system effectiveness, taking into account both false alarms and failures, is shown as a graph of signal to noise ratio versus false alarm rate for .95 success probability. The curves were obtained, for a given averaging time, by picking off the alarm levels for .95 probability of success for all signal to noise ratiosand then turning to the false alarm rate curves and picking the false alarm rates for the previously obtained alarm levels. The hesitation time was kept at 1.5 for these curves. We see that, for smaller signal to noise ratios, rectification by squaring and use of long averaging times are best. For a signal to noise ratio of 1.78 and 10 second averaging time gives about 10 false alarms per hour, and as the signal to noise ratio increases the false alarm rate drops sharply so that the system is quite good at high signal to noise ratios. The large number of false alarms make the system relatively ineffective for signal to noise ratios less than 1.78.

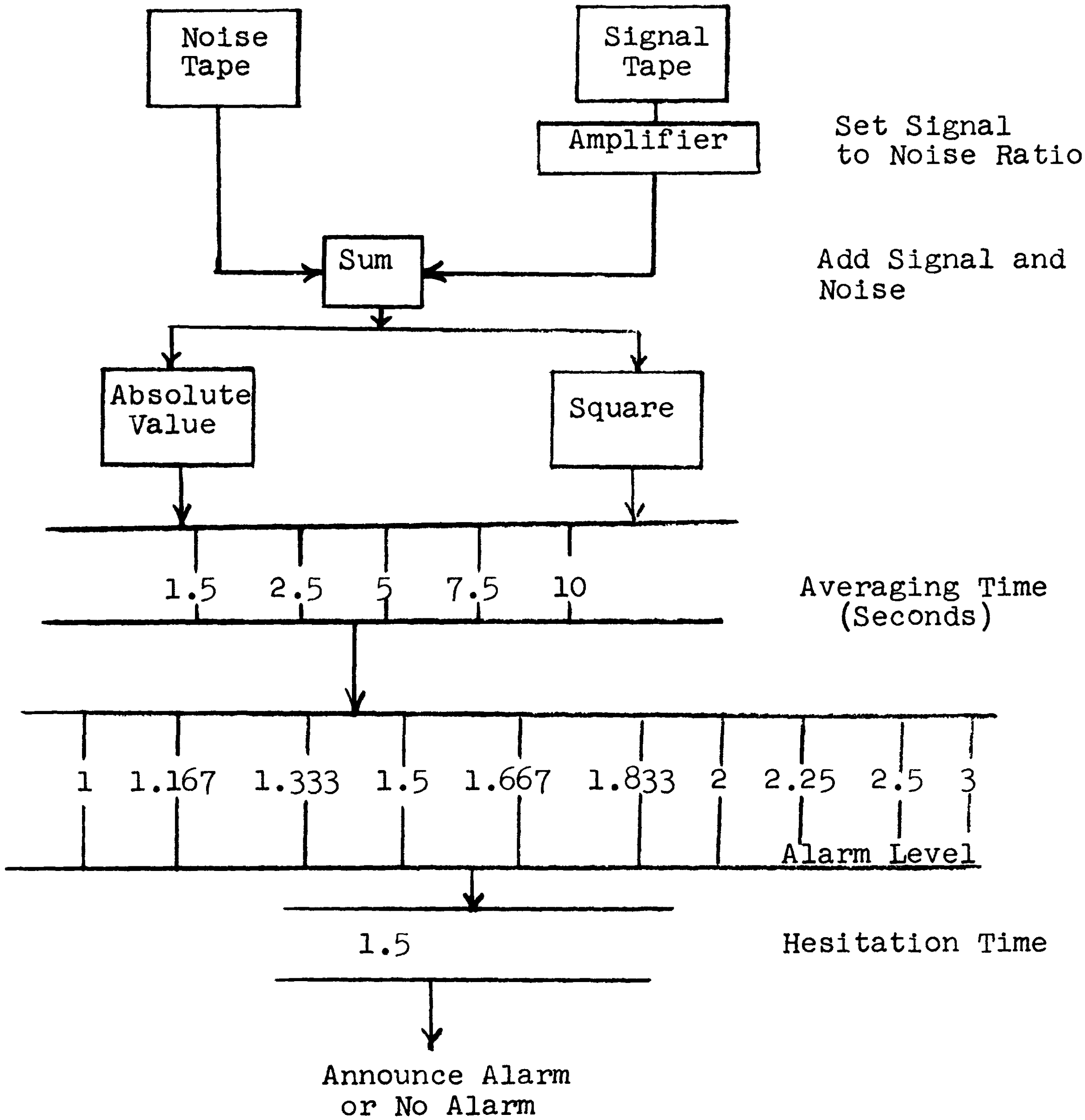


Figure 3.3.1 Failure to Detect Flow Chart

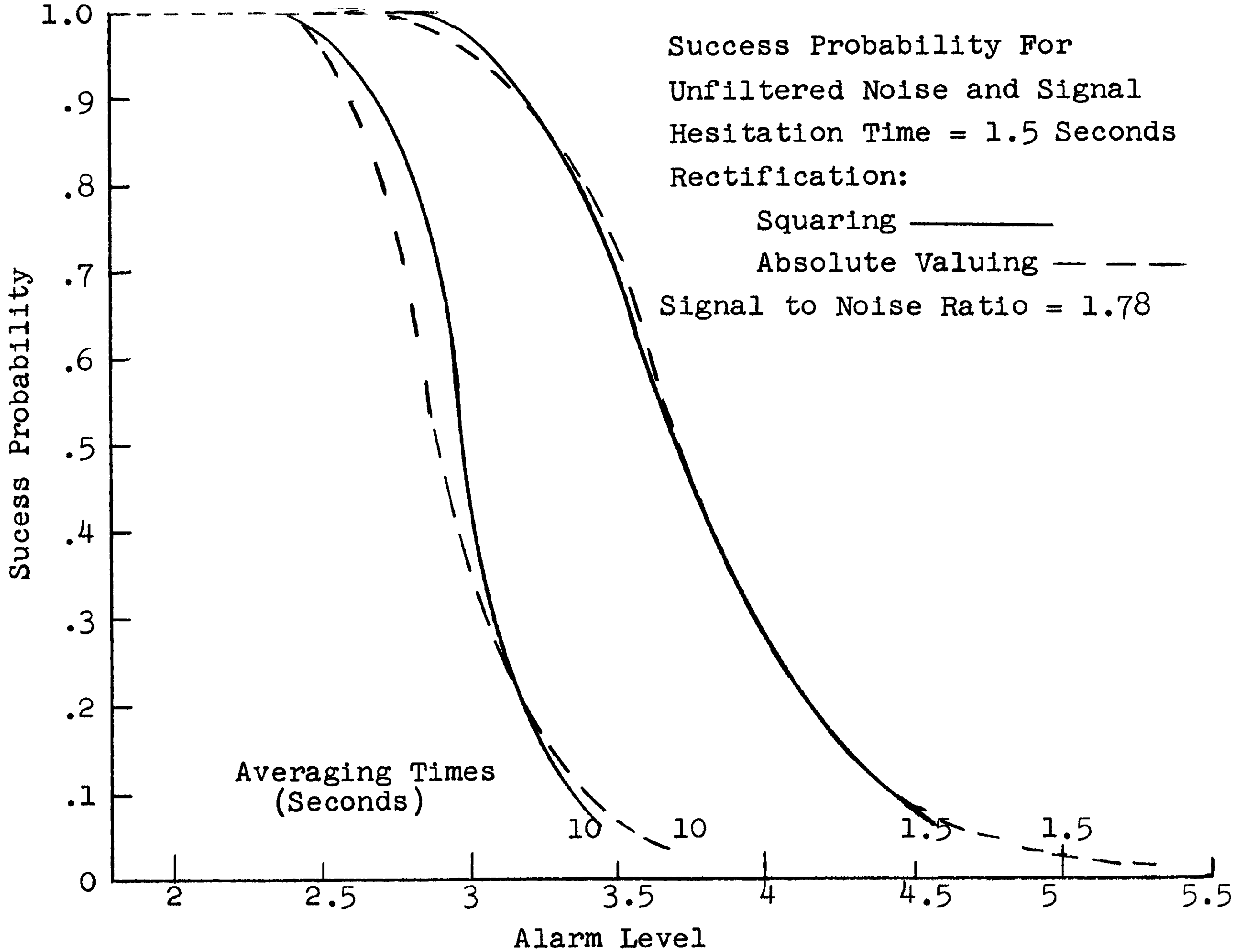
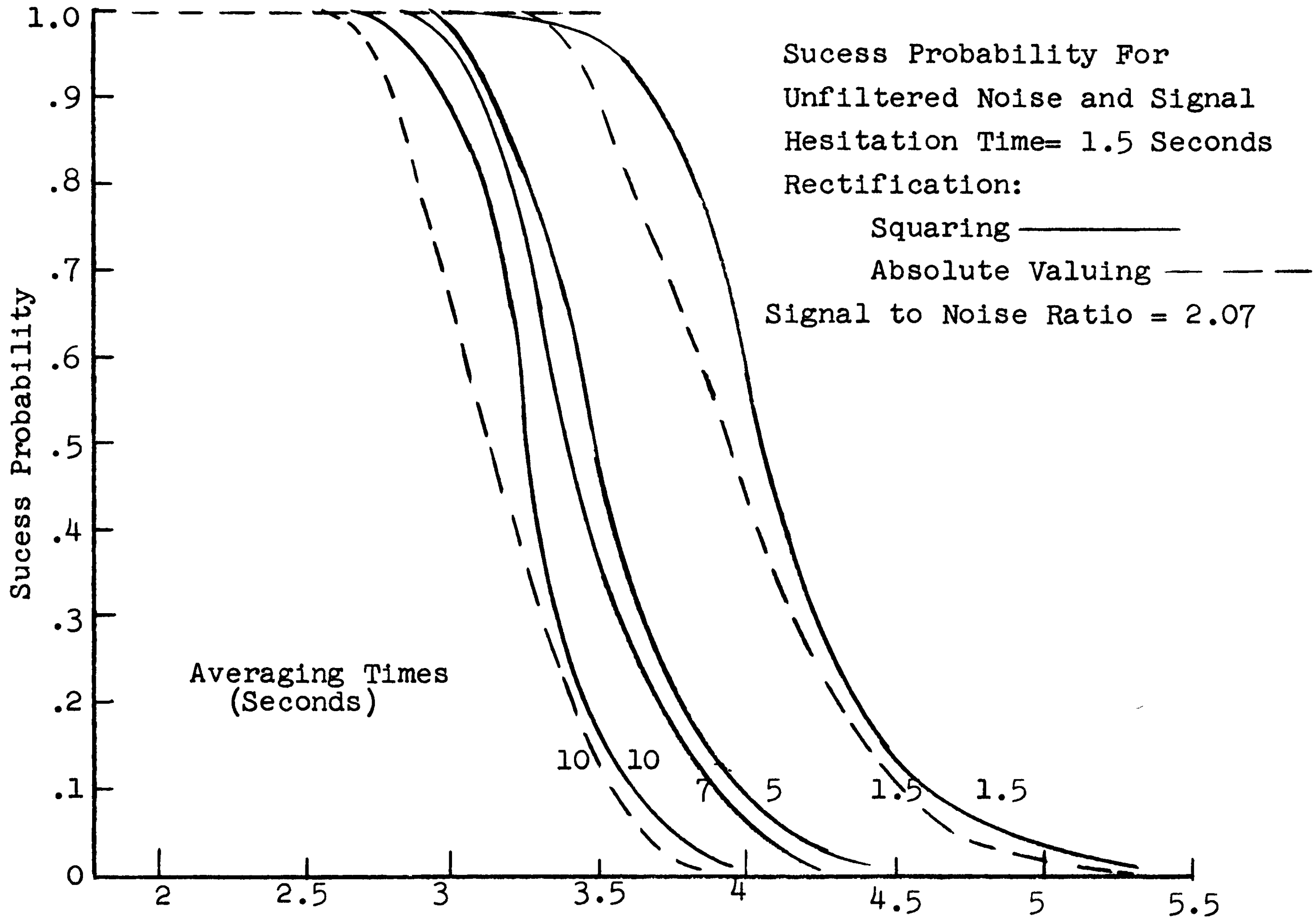

Success Probability For
Unfiltered Noise and Signal
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
Signal to Noise Ratio = 1.78
Sucess Probability
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
0
2
2.5
3
3.5
4
4.5
5
5.5
Alarm Level
Averaging Times
(Seconds)
10
10
1.5
1.5

Figure 3.3.2



Sucess Probability For
Unfiltered Noise and Signal
Hesitation Time= 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
Signal to Noise Ratio = 2.07
Sucess Probability
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
0
Averaging Times
(Seconds)
10
10
7
5
1.5
1.5
2
2.5
3
3.5
4
4.5
5
5.5
Alarm Level

Figure 3.3.3

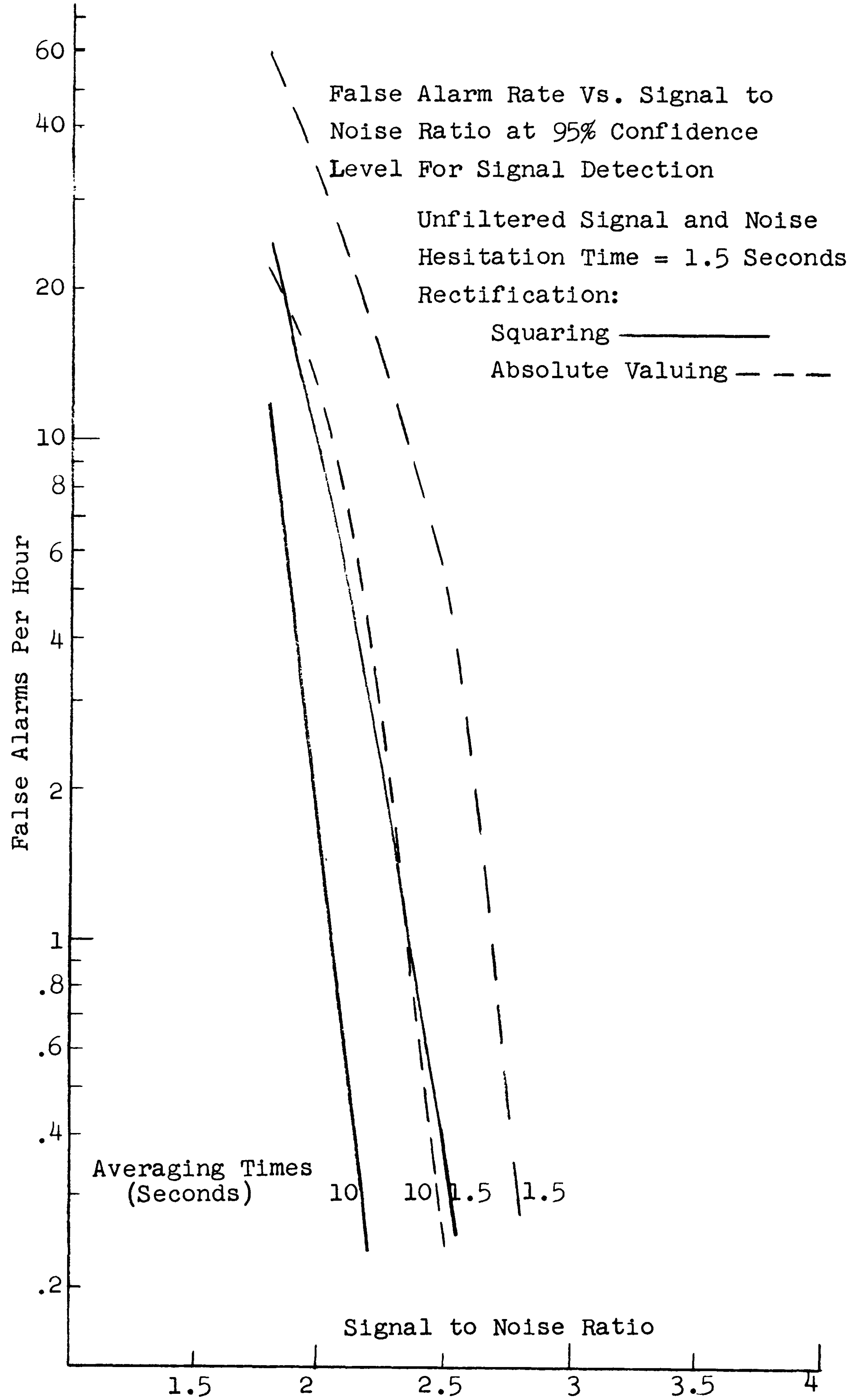

False Alarm Rate Vs. Signal to Noise Ratio at 95% Confidence Level For Signal Detection
Unfiltered Signal and Noise
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
60
40
20
10
8
6
4
2
1
.8
.6
.4
.2
False Alarms Per Hour
Averaging Times (Seconds)
10
10
1.5
1.5
Signal to Noise Ratio
1.5
2
2.5
3
3.5
4

Figure 3.3.4

3.4 Automatic Detection with Filtering

Band Pass Filters and the Signal to Noise Ratio

The last section showed the overall effect of the detection system and indicated that it was not particularly good for signal to noise ratios less than 1.78. If, however, the signal to noise ratio of the raw data can be improved by filtering, the usefulness of the detection system may be increased enormously. Examination of the spectra of the noise records (Figures 1.3.6 to 1.3.9) show that most of the power is between 0 and about .7 cps with a few spikes around 1.4 and 2.0 cps. The vertical records have less energy at the higher frequencies than do the horizontals. If we look at the noise spectra through a window from .7 to 1.8 cps we see only a very small percentage of the total power. The signal, on the other hand, has energy all through this band. If a reasonable percentage of the total signal power appears in this range of frequencies, a simple band pass filter will improve the signal to noise ratio quite a bit.

The programs FAILRA and FALARA can be used again to study the failure and false alarm rates by pre-filtering the signal and noise and the proceeding as in the last two sections. The flow charts in Figures 3.2.1 and 3.3.1 are applicable if "Noise Tape" is changed to "Filtered Noise Tape", and "Signal Tape" changed to "Filtered Signal Tape."

The signal to noise ratio improvement obtained by band pass filtering can be estimated from the spectra of the signal and the noise which are shown in Figure 3.4.1. If the signal and noise were initially scaled to have a one-to-one ratio, and were then band pass filtered to pass .8 to 1.7 cps we see that nearly all the signal would remain and nearly all the noise would be removed. The signal to noise ratio improvement for this case would be a factor of about 5.

Effect of Filter on System Characteristics

It is important to see if the detection system characteristics change significantly when the filtered signal and noise both have band widths which are narrow compared to the band widths of the raw signal and noise. If the characteristics are relatively invariant with band width, the system can be said to be an energy detector and its effectiveness can be measured in terms of the signal to noise ratio improvement brought about by the filtering, and the system response to unfiltered signals.

The constancy of the system to change in band width was studied by band pass filtering the signal and noise separately and using the programs FAILRA and FALARA to obtain the false alarm rates and failure rates. The signal to noise ratios and alarm levels were computed from the amplitudes of the filtered noise and signal. The results of the study are shown in Figures 3.4.2 to 3.4.6. As in the last two sections, the false alarm rate is shown as a graph of the number of false alarms per hour against alarm level, the failure rate is given by the success probability as a function of alarm level, and the system's effectiveness is shown in a graph of the false alarm rate versus signal to noise ratio. In comparing these graphs to the ones for unfiltered data we see only slight differences. The trends are all the same and the actual curves, particularly those for longer averaging time, are approximately the same. The overall system effectiveness is also about the same for the filtered and unfiltered cases.

In view of the findings from the filtered and unfiltered cases we can say that the system is essentially an energy detector and that the curves obtained for the unfiltered case can be used for the filtered case if we can compute the signal to noise ratio improvement due to filtering. We have seen that for the particular signal and noise used this improvement was enormous and results in an extremely low false alarm rate. With the use of the curves which have been presented one can easily compute the range of signal amplitudes which can be detected reliably if the level of the background noise is known.

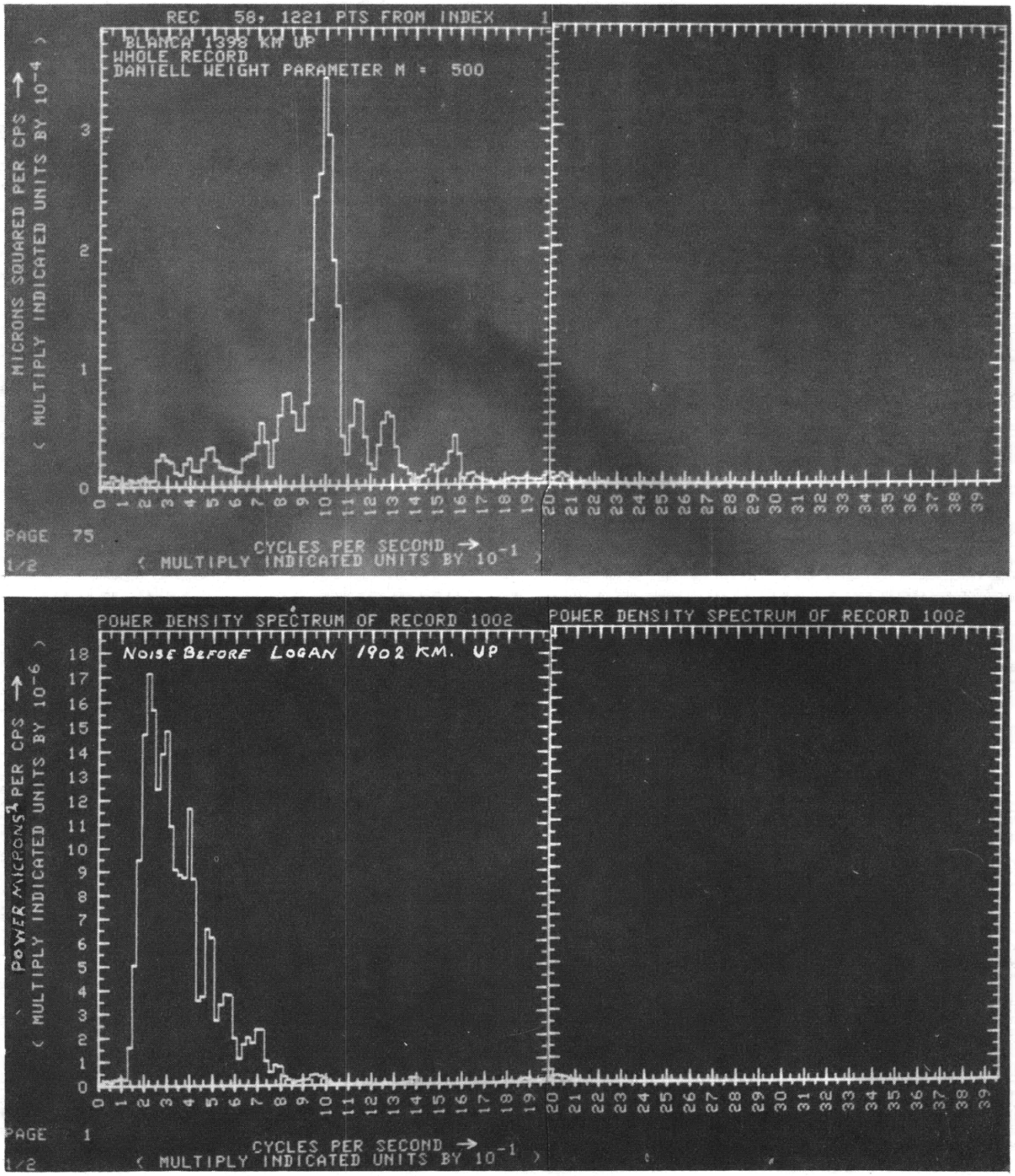


Figure 3.4.1 Signal and Noise Auto Spectra

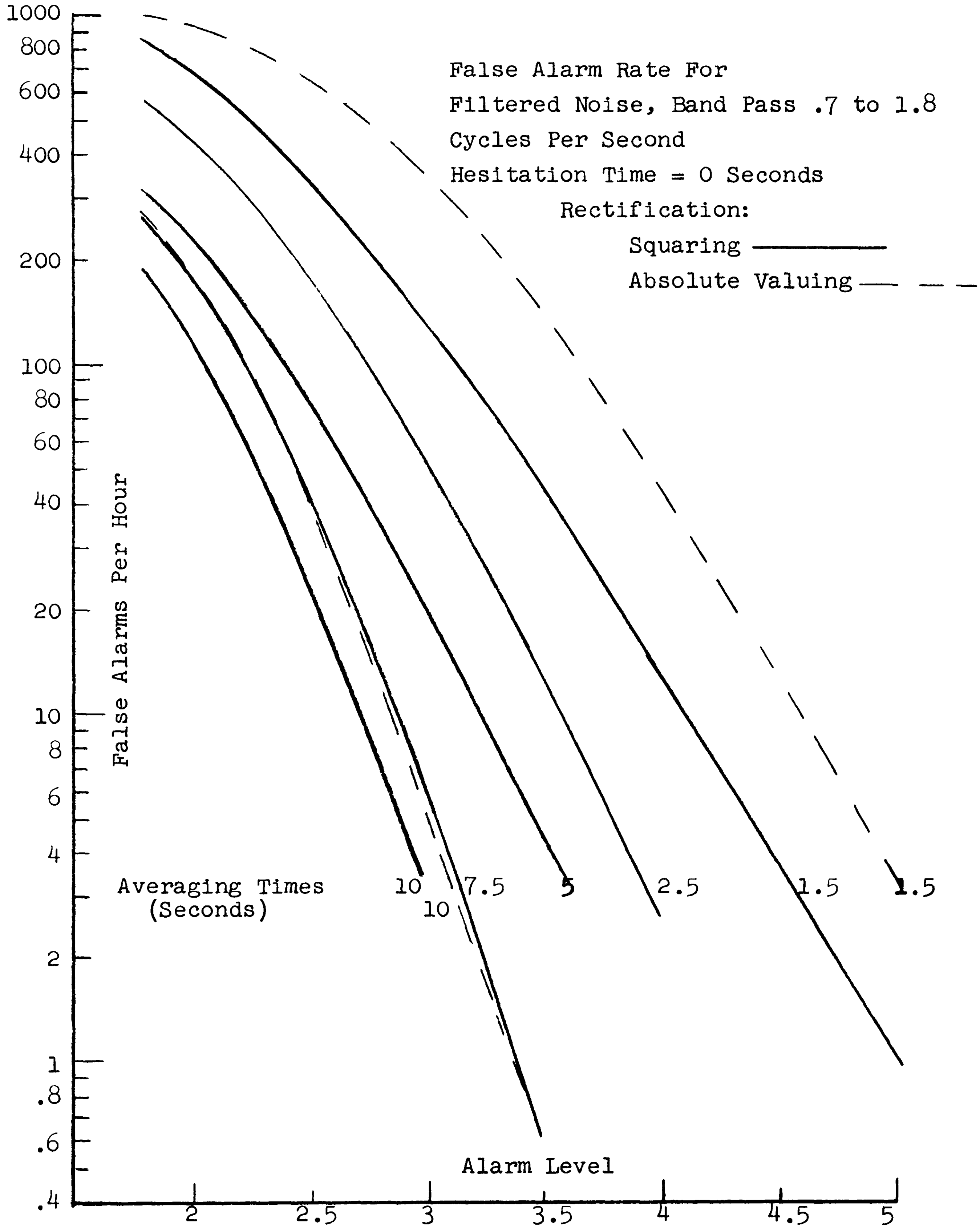
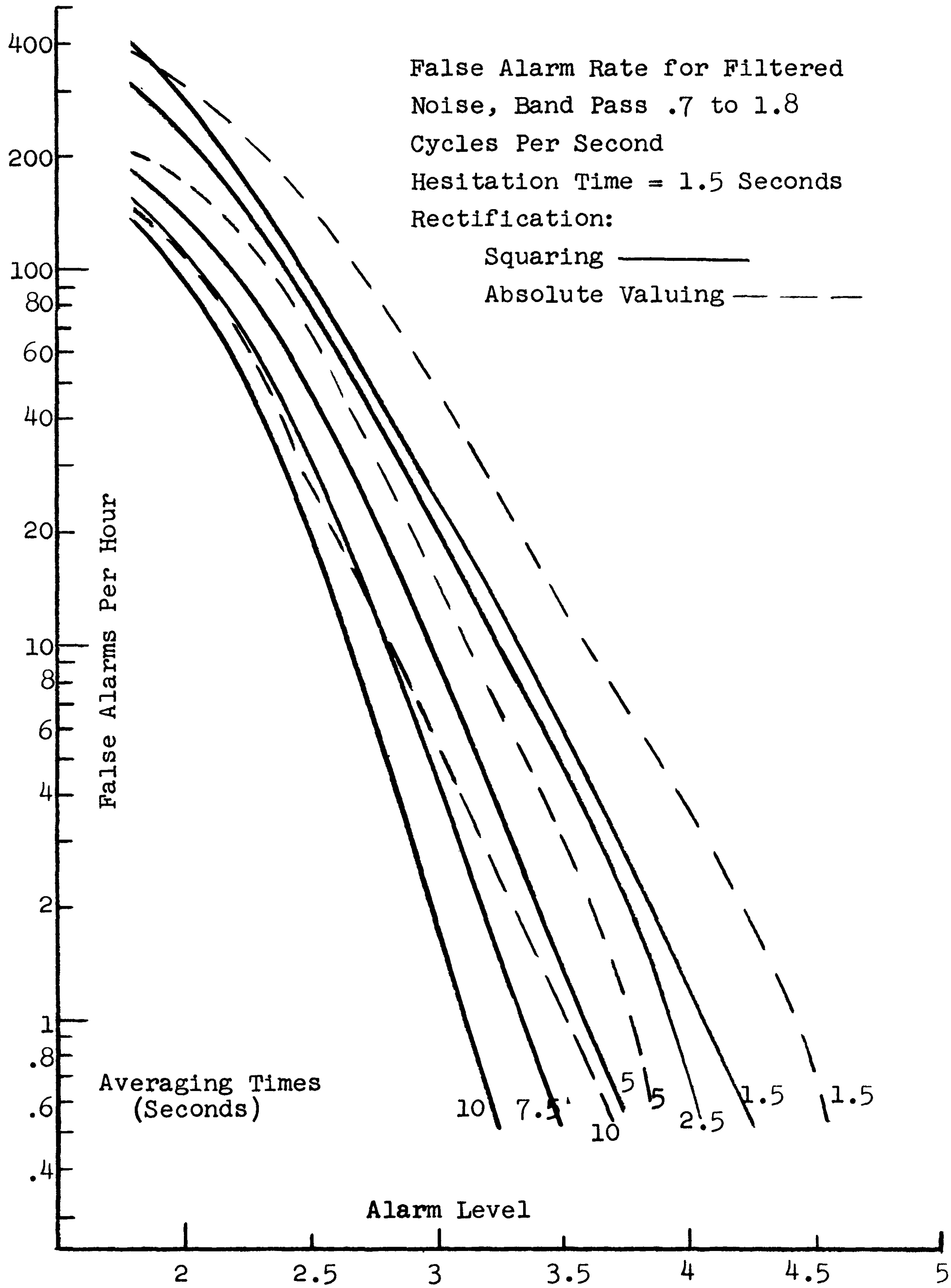

False Alarm Rate For
Filtered Noise, Band Pass .7 to 1.8
Cycles Per Second
Hesitation Time = 0 Seconds
Rectification:
Squaring
Absolute Valuing
False Alarms Per Hour
1000
800
600
400
200
100
80
60
40
20
10
8
6
4
2
1
.8
.6
.4
Averaging Times
(Seconds)
10
10
7.5
5
2.5
1.5
1.5
Alarm Level
2
2.5
3
3.5
4
4.5
5

Figure 3.4.2



400
200
100
80
60
40
20
10
8
6
4
2
1
.8
.6
.4
False Alarms Per Hour
False Alarm Rate for Filtered
Noise, Band Pass .7 to 1.8
Cycles Per Second
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
Averaging Times
(Seconds)
10
7.5
10
5
5
2.5
1.5
1.5
Alarm Level
2
2.5
3
3.5
4
4.5
5

Figure 3.4.3

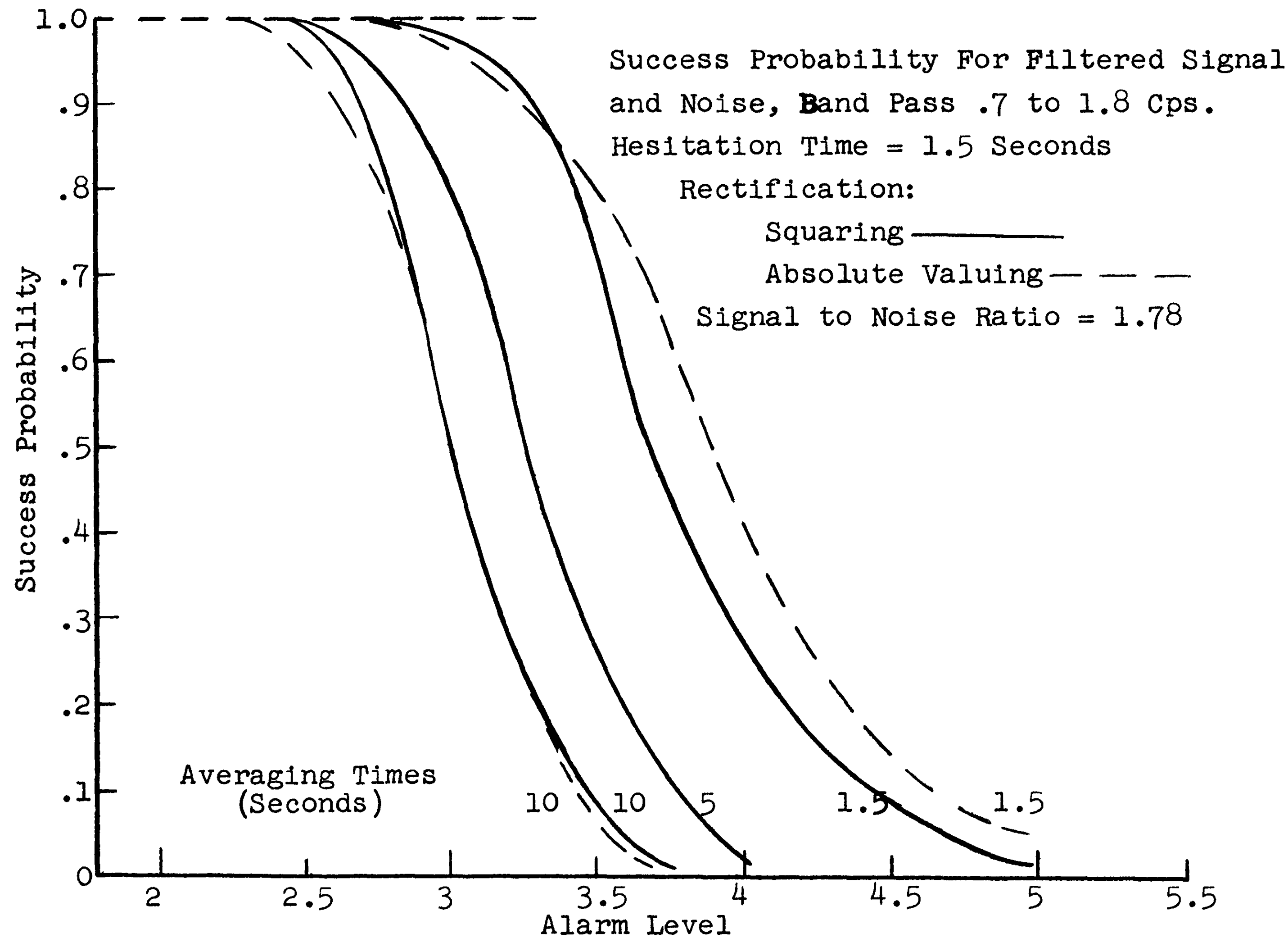

Success Probability For Filtered Signal and Noise, Band Pass .7 to 1.8 Cps.
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
Signal to Noise Ratio = 1.78
Success Probability
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
0
Averaging Times (Seconds)
10
10
5
1.5
1.5
2
2.5
3
3.5
4
4.5
5
5.5
Alarm Level

Figure 3.4.4

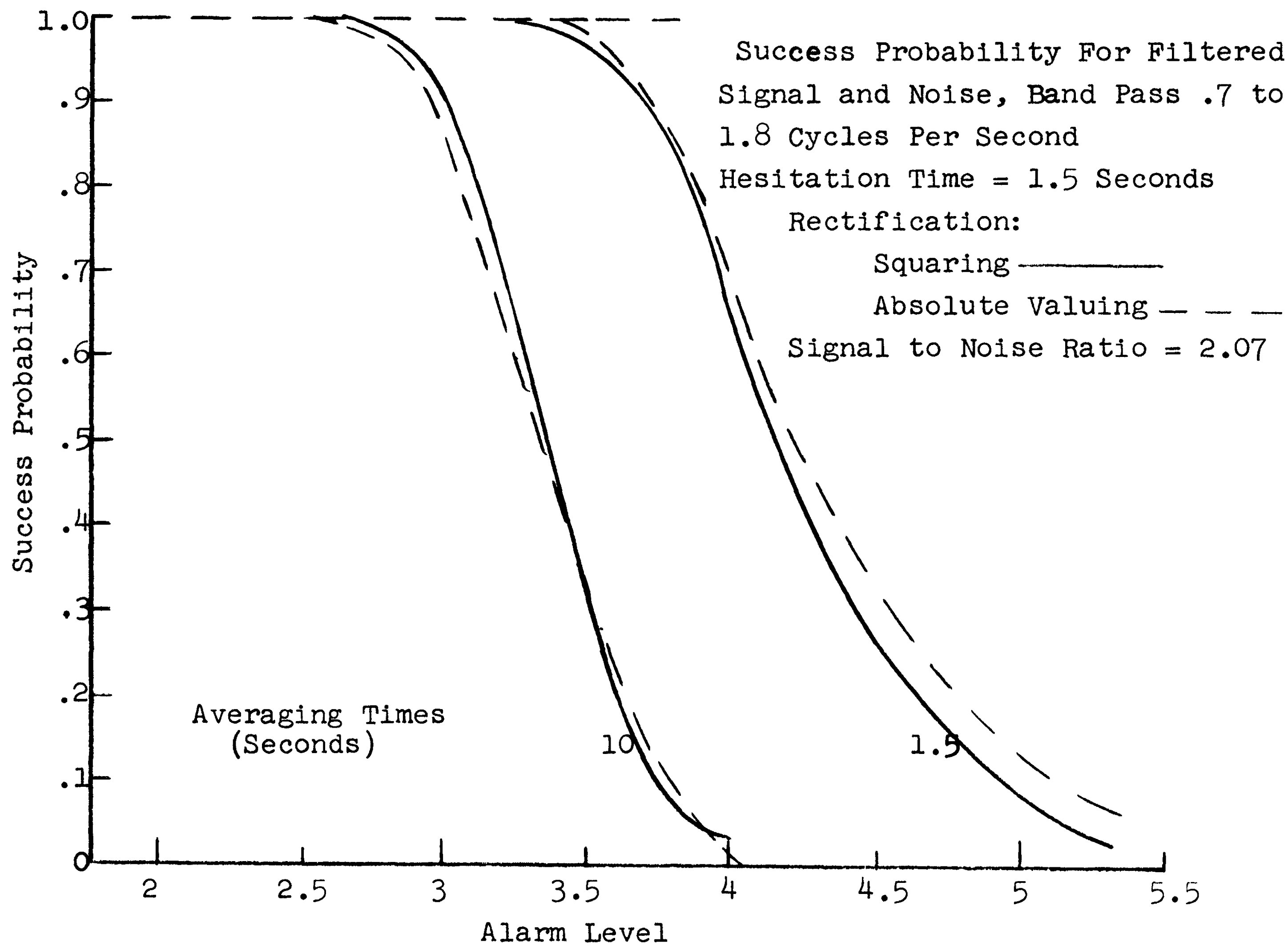

Success Probability For Filtered Signal and Noise, Band Pass .7 to 1.8 Cycles Per Second
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
Signal to Noise Ratio = 2.07
Success Probability
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
0
Averaging Times (Seconds)
10
1.5
2
2.5
3
3.5
4
4.5
5
5.5
Alarm Level

Figure 3.4.5

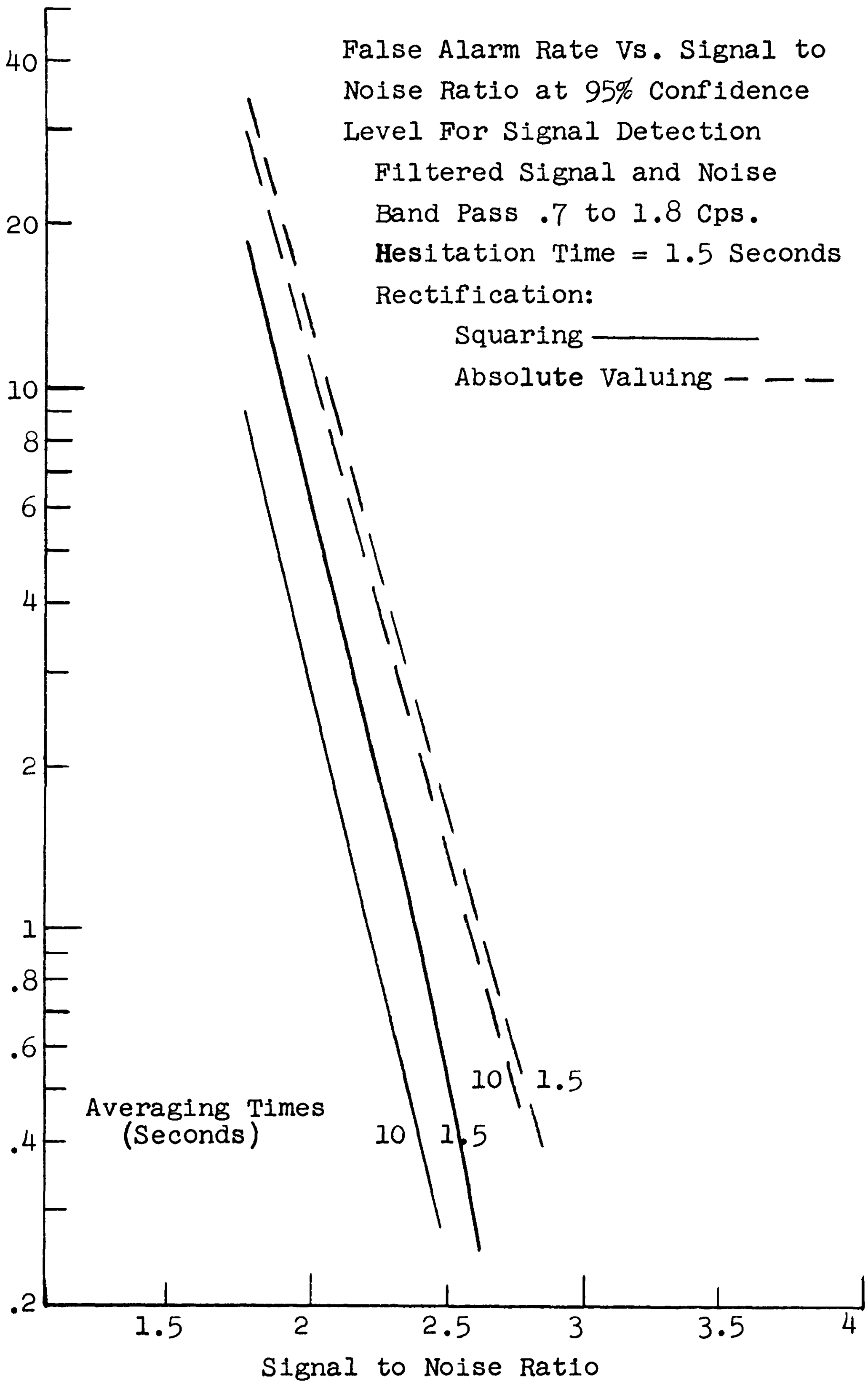

40
20
10
8
6
4
2
1
.8
.6
.4
.2
False Alarm Rate Vs. Signal to Noise Ratio at 95% Confidence Level For Signal Detection
Filtered Signal and Noise
Band Pass .7 to 1.8 Cps.
Hesitation Time = 1.5 Seconds
Rectification:
Squaring
Absolute Valuing
10
1.5
Averaging Times (Seconds)
10
1.5
1.5
2
2.5
3
3.5
4
Signal to Noise Ratio

Figure 3.4.6

4. SUMMARY

The seismic data from the Logan and Blanca underground nuclear shots, which was provided by the Air Force, has been digitized and, along with other data contributed by Dr. Bruce Bogert and by United Electro Dynamics, Inc., has been subjected to many computational experiments. In the first of these the microseism data was considered as a signal and the object was to infer the nature of the sources and the wave types involved. We saw that the amplitude of the microseisms at about .3 cps decreased with increasing distance from the coast, but the higher frequency did not display any regular trend. The suggestion is that the low frequency noise is of oceanic origin whereas the higher frequencies are more likely of local origin. It was not possible to pin down Rayleigh and Love waves with any degree of certainty, but their presence was not disproved. The failure of the wave type experiments is attributed to the complex nature of the microseisms. The model used cannot deal with many waves of the same frequency but different directions of travel.

The inadequacy of a simple deterministic model motivated a statistical treatment of microseismic noise. The microseisms are considered as a time series and, under the ergodic hypothesis, the relative constancy of the power density spectrum suggests that the time series is at least wide sense stationary. Studies on the microseism amplitudes show that their probability distribution is Gaussian and that they are dependent.

The power density spectra have been computed using the Daniell technique. The spectra are quite similar in structure over distances of several hundred kilometers. There is a prominent peak at about .3 cps and in some cases there are peaks at 1.4 and 2 cps. The low frequency peak is interpreted as the high end of the oceanic microseism band which is cut off on the low end by the seismometer response. The higher frequencies are attributed to local causes.

Cross spectra of different components at the same station, like components from different stations, and array data have been computed. Again it is difficult to pick out individual wave types and it is not possible to follow waves from one station to another. This is again attributed to the complex structure of the noise.

Since the microseisms can be considered as a wide sense stationary time series, a mathematical description is possible. The moving summation and autoregressive representations are valid. With the assumption of an absolutely continuous spectral density the spectra can be factored and a minimum phase wavelet found for the moving average representation. The generating model for microseisms is then a white light series into a minimum phase system. Probability studies on the white light series obtained by convolving the inverse minimum phase wavelet with the original data show that the white light is essentially Gaussian and independent.

The minimum phase wavelet is also the predictive decomposition and can be used to compute the predictability of the microseisms. This technique of prediction is found to be faster and easier to handle than the mean aquare error method, although the Levinson technique is quite good. The predictability of the microseisms is not very great. About half the energy (50 percent reduction) can be predicted for one or two seconds and then the decrease is fairly rapid. Multidimensional prediction does not give appreciably better results than the one dimensional or self prediction. Thus prediction as a method of noise reduction in the first motion interval is not particularly promising. We can say, however, that our linear prediction is the best we can do, and that non-linear operators will not help. This is because the microseisms can be considered to be generated by Gaussian white noise into a minimum phase system. In this case the white noise is independent and higher correlations give no information about the noise.

The mathematical model enables us to generate artificial microseisms so that long periods of continuous noise are available. These long noise series are required by the computer program which simulates a detection system. Noise above is needed to compute the false alarm rate and signal plus noise is needed for the failure rate. The system effectiveness is plotted on a graph of false alarms per hour as a function of signal to noise ratio for 95% detection probability (5% failure rate). The system characteristics are found to remain approximately constant when a band pass filter is introduced at the input. Thus the system will function as an energy detector and band pass filters can be used to improve the signal to noise ratio. Improvement of a factor of five was found for the particular signal, noise, and filter used.

The emphasis has been on the statistical approach throughout this thesis. There is, of course, plenty of room for additional work of both statistical and deterministic nature on the available data in the same general area as the present work. More complicated models which take into account several wave types and many directions of travel may be introduced and fitted to the data. New techniques will enable multi-dimensional prediction studies with long operator lengths, and it would be interesting to compare results of this sort of study with the long operator studies of section 2.2.

The cross correlation results on the array data certainly do not represent exhaustive study. Multi-dimensional prediction experiments as well as summation of records with variable time lags would be quite interesting. Three component and array detection system studies by computer simulation would also prove useful.

APPENDIX A

WATER WAVE PROBLEM

Longuet-Higgins (1950) has shown that a standing wave can produce a second order pressure fluctuation which is unattenuated with depth and which has twice the time frequency of the standing wave. Hence it is possible to show that microseisms could be produced in deep water even though the linear theory tells us that the pressure fluctuations die off exponentially with depth. In order that there be enough energy transmitted to the bottom, there must be a "patch" of standing waves which is coherent over a fairly large area and the patch must not move because the motion will cause the pressure oscillations to average out to zero. Therefore the standing waves must meet nearly head on. In fact, it has been shown (Kenyon, 1961) that if the travelling waves meet at an angle θ ($\theta = 0$, head on), the average pressure on the bottom must be multiplied by $\exp(-2h\alpha\sin\theta)$ where h is the depth of the water, α the wave number and θ the angle between the travelling wave fronts.

There is a special case of interest when the waves meet at such an angle that the "patch" of standing waves moves with a velocity, V_S , equal to the velocity of propagation of Rayleigh waves, V_r , in the medium. The travelling waves, with velocity V_t , must meet at an angle θ such that

$$V_t = V_r / \sin(\theta/2)$$

In this case there is essentially a resonance and strong microseisms could build up if the "patch" of water waves remains coherent for a long enough time.

One of the problems considered by Longuet-Higgins was the two dimensional compressible case of a layer of water with a rigid lower boundary and a standing wave at the surface. His solution requires the small parameter expansion technique of handling non-linear problems and illustrates the frequency doubling effect as well as organ pipe resonance. The problem which will be treated here is a good deal simpler in that it considers the incompressible transient problem. This is done to illustrate the energy swapping to the sum and difference frequencies of all frequencies present and uses a representation for non-linear problems devised by DeVorkin (1963). DeVorkin's scheme is particularly useful in that the solution is in terms of kernels which do not depend on the initial conditions. Therefore once the kernels have been found for a given geometry the solution of many problems with different initial conditions can readily be found. The method is also useful for statistical initial conditions.

We consider the two dimensional transient problem of an incompressible irrotational fluid layer of constant thickness, h , over a rigid half space with arbitrary initial conditions on the velocity and surface shape. We assume a velocity potential φ . The velocity is therefore $\vec{V} = -\vec{\nabla}\varphi$. The continuity equation is then $\nabla^2\varphi = 0$ and the equation of motion is

$$\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \vec{\nabla})\vec{V} + g\vec{\Omega} + \frac{1}{\rho}\vec{\nabla}p = 0$$

where Ω is the gravitational potential, ρ is the density (assumed constant) and p the pressure. We factor out a ∇ and obtain Bernoulli's equation

$$-\frac{\partial \varphi}{\partial t} + \nabla \varphi \cdot \nabla \varphi + g z + \frac{p}{\rho} = 0$$

where z is negative downward and $p = 0$ at the surface $z = \eta$. The free surface condition is

$$\frac{\partial \varphi}{\partial x} \frac{\partial \eta}{\partial x} - \frac{\partial \varphi}{\partial z} + \frac{\partial \eta}{\partial t} = 0 \quad \text{at} \quad z = \eta(x,t) \qquad \text{(A-1)}$$

Bernoulli's equation becomes at $z = 0$

$$-\frac{\partial \varphi}{\partial t}\Big|_{z=\eta} + \left(\frac{\partial \varphi}{\partial x}\right)^2\Big|_{z=\eta} + \left(\frac{\partial \varphi}{\partial z}\right)^2\Big|_{z=\eta} + g\eta = 0 \qquad \text{(A-2)}$$

The solution to the continuity equation which satisfies the condition $\frac{\partial \varphi}{\partial z} = 0$ at $z = -h$ is

$$\varphi(x,z,t) = \sum_{m=-M}^{M} \Phi_m(t)\left[e^{-mz} + e^{2mh} e^{mz}\right] e^{-imx} \qquad \text{(A-3)}$$

where we have assumed a discrete set of frequencies. DeVorkin's representation scheme applies to total differential equations and hence to the Fourier transform over the spacial frequencies of the boundary equations.

The initial conditions are

for φ: $F(m,0)$, $m = -M$ to M

for η: $N(m,0)$, $m = -M$ to M

where $F(m,t)$ and $N(m,t)$ are the Fourier transforms of $\varphi(t)$ and $\eta(t)$.

We combine these into a single variable

$$\psi_\alpha\ , \quad \alpha = 1, \ldots, 4m+2 \qquad \text{where}$$

$$\psi_1 = F(-m,0),\ \psi_2 = N(-m,0),\ \psi_3 = F(-m+1,0),\ \psi_4 = N(-m+1,0),\ \text{etc.}$$

The representation scheme is then:

$$F(m,t) = \sum_\alpha K^m_\alpha \psi_\alpha + \sum_{\alpha\beta} K^m_{\alpha\beta} \psi_\alpha \psi_\beta + \sum_{\alpha\beta\gamma} K^m_{\alpha\beta\gamma} \psi_\alpha \psi_\beta \psi_\gamma + \ldots$$

$$N(m,t) = \sum_\alpha L^m_\alpha \psi_\alpha + \sum_{\alpha\beta} L^m_{\alpha\beta} \psi_\alpha \psi_\beta + \sum_{\alpha\beta\gamma} L^m_{\alpha\beta\gamma} \psi_\alpha \psi_\beta \psi_\gamma + \ldots$$

which can be combined to

$$\psi_n(t) = \sum_\alpha R^n_\alpha \psi_\alpha + \sum_{\alpha\beta} R^n_{\alpha\beta} \psi_\alpha \psi_\beta + \sum_{\alpha\beta\gamma} R^n_{\alpha\beta\gamma} \psi_\alpha \psi_\beta \psi_\gamma + \ldots \qquad \text{(A-4)}$$

where

$$\psi_n(t) = F\left(\frac{n-2m-1}{2}, t\right) \qquad \text{for } n \text{ odd}, \geq 1$$

$$\psi_n(t) = N\left(\frac{n-2m-2}{2}, t\right) \qquad \text{for } n \text{ even}, \geq 2\ . \qquad \text{(A-5)}$$

The R's are thus system functions independent of initial conditions.

The boundary equations (A-1) and (A-2) apply at $z = \eta$ but since

η is unknown the equations must be expanded in a Taylor series about $z = 0$ in powers of η. Expanding to second order only

$$-\varphi_t - \frac{\partial \varphi_t}{\partial z}\eta + \varphi_x^2 + \varphi_z^2 + g\eta = 0 \qquad \text{(A-6)}$$

$$\varphi_x \eta_x - \varphi_z - \frac{\partial \varphi_z}{\partial z}\eta - \eta_t = 0 \qquad \text{(A-7)}$$

where the subscripts denote differentiation.

We take the Fourier transform of these equations to obtain

$$\dot{F}(m) = i\sum_p p\, C(p)\, \dot{F}(p)\, N(m-p) - \sum_p p\, F(p)\,(m-p)\, F(m-p) +$$

$$-\sum_p p\, C(p)\, F(p)\,(m-p)\, C(m-p)\, F(m-p) + g N(m) = 0 \qquad \text{(A-8)}$$

for equation (A-6) and a similar expression for equation (A-7). In this transformation we have used the fact that multiplication in one domain is convolution in the other, and have set the transform of $\frac{\partial \varphi}{\partial z}$ equal to $C(m)\,F(m)$. The dots represent time differentiation. We note that equation (A-8) contains more than one term with a time derivative. Poincare's theorem on small parameter expansions does not guarantee a solution unless the right-hand side contains not time derivatives. We can, however, consider all the time derivative terms as an operator, H, operating on $F(m)$ and then show that the operator $H = I - a$ can be inverted if a is small. That is, if the operator H cannot in general be inverted, we must demand that it can be expressed as $I - a$ where a is small enough that the Neumann series resulting from the inversion converges. Hence, for many cases we must impose the restriction that the non-linear terms be small compared to the linear ones.

Since H can be inverted we go ahead and use the representation scheme equating terms of like order and remembering that the equations must hold for arbitrary initial conditions, ψ_α.

The first order equations are from equations (A-6), (A-7) and (A-8), using the notation introduced in equations (A-4) and (A-5),

$$\dot{R}_\alpha^n - g R_\alpha^{n+1} = 0$$

$$\dot{R}_\alpha^{n+1} + \left(\frac{n-J}{2}\right) C\left(\frac{n-J}{2}\right) R_\alpha^n = 0 \qquad ; \; J = 2M+1$$

These can be solved to give

$$R_\alpha^n = a_+ \exp[i\gamma(n,J)t] + a_- \exp[-i\gamma(n,J)t]$$

for n odd, where

$$\gamma(n,J) = \sqrt{g\left(\frac{n-J}{2}\right) C\left(\frac{n-J}{2}\right)}$$

$$a_+ = \frac{g + i\gamma(n,J)}{2\gamma(n,J)} \delta_{n\alpha}$$

$$a_- = \frac{-g - i\gamma(n,J)}{2\gamma(n,J)} \delta_{n\alpha}$$

where $\delta_{n\alpha}$ is the Kronecker delta, and

$$R_\alpha^{n+1} = b_+ \exp[i\gamma(n,J)t] + b_- \exp[-i\gamma(n,J)t]$$

for n odd, where

$$b_+ = \frac{-\frac{n-J}{2}C\left(\frac{n-J}{2}\right) - \gamma(n,J)}{2\gamma(n,J)}\,\delta_{n+1,\alpha}$$

$$b_- = \frac{\frac{n-J}{2}C\left(\frac{n-J}{2}\right) + \gamma(n,J)}{2\gamma(n,J)}\,\delta_{n+1,\alpha}$$

The above equations for R_α^n and R_α^{n+1} are correct for $n \neq J$. For $n = J$, R_α^n and R_α^{n+1} are zero for all t.

The second order equations, obtained by equating the second order terms in equations (A-6), (A-7) and (A-8) containing the second order kernels and convolutions of the first order terms. The convolutions may easily be performed and the $\dot{R}^n_{\alpha\beta}$, $\dot{R}^{n+1}_{\alpha\beta}$ equations can be considered as a matrix equation. However, due to the simple coupling of the equations only a 2 × 2 matrix need be considered. The zero spacial frequency, $n = J$, must again be considered as a special case.

The second order equations are

$$\sum_{K\ell} \dot{R}^n_{K\ell}\,\psi_K\,\psi_\ell - g\sum R^{n+1}_{K\ell}\,\psi_K\,\psi_\ell = \sum_{p=1}^{N-1} \frac{p-J}{2}\,C\left(\frac{p-J}{2}\right) \sum \dot{R}^p_K\psi_K \sum R^{n-p}_\alpha \psi_K$$
$$+ \sum_{p=1}^{N-1}\left(\frac{n-J}{2}\right)\left(\frac{n-p}{2}\right)\left(1 - C\left(\frac{n-p}{2}\right)\right) C\left(\frac{n-J}{2}\right) \sum_K R^p_K\,\psi_K \sum_\alpha R^{n-p+1}_\alpha\,\psi_K \qquad (A-9)$$

where $N = 4M+2$ and n and p are odd,

$$\sum_{\kappa\ell} \dot{R}^{n+1}_{\kappa\ell} \psi_\kappa \psi_\ell + \frac{n-J}{2} C\left(\frac{n-J}{2}\right) \sum_{\kappa\ell} R^n_{\kappa\ell} \psi_\kappa \psi_\ell =$$

$$-\sum_{p=1}^{N-1} \left(\frac{p-J}{2}\right)\left(\frac{n-J}{2}\right) \sum_\alpha R^p_\alpha \psi_\alpha \sum_\kappa R^{n-p}_\kappa \psi_\kappa$$

The equations must hold for arbitrary ψ_α so that

$$\dot{R}^n_{\kappa\ell} - q R^{n+1}_{\kappa\ell} = \sum_{p=1}^{N-1} \left(\frac{p-J}{2}\right) C\left(\frac{p-J}{2}\right) \left[\dot{R}^p_\kappa R^{n-p}_\ell + \dot{R}^p_\ell R^{n-p}_\kappa\right] +$$

$$+ \sum_{p=1}^{N-1} \left(\frac{n-J}{2}\right)\left(\frac{n-p}{2}\right)\left(1 - C\left(\frac{n-p}{2}\right)\right) C\left(\frac{p-J}{2}\right) \left[R^p_\kappa R^{n-p+1}_\ell + R^p_\ell R^{n-p+1}_\kappa\right] \qquad \text{(A-10)}$$

and

$$\dot{R}^{n+1}_{\kappa\ell} + \left(\frac{n-J}{2}\right) C\left(\frac{n-J}{2}\right) R^n_{\kappa\ell} = -\sum_{p=1}^{N-1} \left(\frac{p-J}{2}\right)\left(\frac{n-J}{2}\right) \left[\dot{R}^p_\kappa R^{n-p}_p + \dot{R}^p_\ell R^{n-p}_\kappa\right] \qquad \text{(A-11)}$$

The convolutions are not hard since R^n_α is diagonal. The last two equations may be written

$$\dot{R}^n_{\kappa\ell} - q R^{n+1}_{\kappa\ell} = T^n_{\kappa\ell}$$

$$\dot{R}^{n+1}_{\kappa\ell} + \left(\frac{n-J}{2}\right) C\left(\frac{n-J}{2}\right) R^n_{\kappa\ell} = T^{n+1}_{\kappa\ell} \qquad n \text{ odd}$$

We write this as a matrix equation

$$\begin{bmatrix} \dot{R}^n_{\kappa\ell} \\ \dot{R}^{n+1}_{\kappa\ell} \end{bmatrix} + A \begin{bmatrix} R^n_{\kappa\ell} \\ R^{n+1}_{\kappa\ell} \end{bmatrix} = \begin{bmatrix} T^n_{\kappa\ell} \\ T^{n+1}_{\kappa\ell} \end{bmatrix}$$

where A is the matrix

$$A = \begin{bmatrix} 0 & -g \\ \left(\frac{n-J}{2}\right) C \left(\frac{n-J}{2}\right) & 0 \end{bmatrix}$$

The solution to the equation is, then,

$$\begin{bmatrix} R^{n}_{K\ell} \\ R^{n+1}_{K\ell} \end{bmatrix} = \int_0^t e^{-A(t-\tau)} \begin{bmatrix} T^{n}_{K\ell} \\ T^{n+1}_{K\ell} \end{bmatrix} d\tau$$

Since $R^{n}_{K\ell}, R^{n+1}_{K\ell} = 0$ at $t = 0$. This is simplified considerably if A can be diagonalized. If U is the transformation matrix for this diagonalization then $R^{n}_{K\ell} = U S^{n}_{K\ell}$ and

$$U \begin{bmatrix} \dot{S}^{n}_{K\ell} \\ \dot{S}^{n+1}_{K\ell} \end{bmatrix} + A U \begin{bmatrix} S^{n}_{K\ell} \\ S^{n+1}_{K\ell} \end{bmatrix} = \begin{bmatrix} T^{n}_{K\ell} \\ T^{n+1}_{K\ell} \end{bmatrix}$$

multiplying by U^{-1}

$$\begin{bmatrix} \dot{S}^{n}_{K\ell} \\ \dot{S}^{n+1}_{K\ell} \end{bmatrix} + U^{-1} A U \begin{bmatrix} S^{n}_{K\ell} \\ S^{n+1}_{K\ell} \end{bmatrix} = U^{-1} \begin{bmatrix} T^{n}_{K\ell} \\ T^{n+1}_{K\ell} \end{bmatrix}$$

where $U^{-1} A U = D$ is diagonal.

Then

$$\begin{bmatrix} S^{n}_{K\ell} \\ S^{n+1}_{K\ell} \end{bmatrix} = \int_0^t e^{-D(t-\tau)} U^{-1} \begin{bmatrix} T^{n}_{K\ell} \\ T^{n+1}_{K\ell} \end{bmatrix} d\tau$$

and

$$\begin{bmatrix} R^{n}_{K\ell} \\ R^{n+1}_{K\ell} \end{bmatrix} = \int_0^t U e^{-D(t-\tau)} U^{-1} \begin{bmatrix} T^{n}_{K\ell} \\ T^{n+1}_{K\ell} \end{bmatrix} d\tau$$

For the matrix AU and U^{-1} are

$$U = \begin{bmatrix} 1 & 1 \\ -i\sqrt{\dfrac{(n-J)C\left(\frac{n-J}{2}\right)}{g}} & i\sqrt{\dfrac{(n-J)C\left(\frac{n-J}{2}\right)}{g}} \end{bmatrix}$$

$$U^{-1} = \begin{bmatrix} 1 & -i\sqrt{\dfrac{g}{(n-J)C\left(\frac{n-J}{2}\right)}} \\ 1 & i\sqrt{\dfrac{g}{(n-J)C\left(\frac{n-J}{2}\right)}} \end{bmatrix}$$

the term $e^{-D(t-\tau)}$ becomes

$$\begin{bmatrix} \exp(-D_{nn}(t-\tau)) & 0 \\ 0 & \exp(-D_{n+1,n+1}(t-\tau)) \end{bmatrix}$$

and the solution for $R^n_{k\ell}$, $n \neq J, J+1$ is then

$$\begin{bmatrix} R^n_{k\ell} \\ R^{n+1}_{k\ell} \end{bmatrix} = \int_0^t \begin{bmatrix} X+Y & -i\sqrt{\dfrac{g}{(n-J)C\left(\frac{n-J}{2}\right)}}\,(X-Y) \\ -i\sqrt{\dfrac{(n-J)C\left(\frac{n-J}{2}\right)}{g}}\,(X+Y) & -X-Y \end{bmatrix} \begin{bmatrix} T^n_{k\ell} \\ T^{n+1}_{k\ell} \end{bmatrix} d\tau$$

where

$$X = \exp(i\,\gamma(n)(t-\tau))$$

$$Y = \exp(-i\,\gamma(n)(t-\tau))$$

For the zero spacial frequency, which is the frequency of interest for deep water microseism generation, $n = J, J+1$, we have from equation (A-11)

$$\dot{R}^{J+1}_{k\ell} = 0 \quad , \quad R^{J+1}_{k\ell} = 0$$

In equation (A-10) we note a symmetry in k and ℓ so that we need only consider half of the right-hand side from which we determine half the solution for $R^J_{k\ell}$. We call this half of the solution $R'^J_{k\ell}$ and the entire solution is thus

$$R^J_{k\ell} = R'^J_{k\ell} + R'^J_{\ell k}$$

We can determine $C(m)$ from equation (A-3) by setting $z = 0$ after differentiation.

$$C(m) = \tanh(mh)$$

The solution $R'^J_{k\ell}$ is then

$$R'^J_{k\ell} = \int_0^{\tau} \left\{ \frac{k-J}{2} \tanh\left(\frac{k-J}{2}h\right) \left[\dot{R}^k_k R^{J-k}_{\ell} \right] \delta_{\ell, J-k} + \left(\frac{k-J}{2}\right)\left(\frac{J-k}{2}\right) \left[\left(1+\tanh\left(\frac{k-J}{2}h\right)\right) \tanh\left(\frac{J-k}{2}h\right) \right] \left[R^k_k R^{J-k+1}_{\ell} \right] \delta_{\ell, J-k+1} \right\} d\tau$$

where the R^s_{α} are functions of τ . We substitute in for the and integrate to obtain terms of the form:

$$R'^{J}_{\kappa\ell} = \frac{\kappa-J}{2}\tanh\left(\frac{\kappa-J}{2}h\right)\delta_{\ell,\,J-\kappa}\,\gamma(\kappa)\left\{\frac{a_+ b_- \exp[i(\gamma(\kappa)+\gamma(J-\kappa-1))t]}{\gamma(\kappa)+\gamma(J-\kappa-1)}\right.$$

$$+\frac{a_+ b_- \exp[i(\gamma(\kappa)-\gamma(J-\kappa-1))t]}{\gamma(\kappa)-\gamma(J-\kappa-1)}+\frac{a_- b_+ \exp[-i(\gamma(\kappa)+\gamma(J-\kappa-1))t]}{\gamma(\kappa)-\gamma(J-\kappa-1)}$$

$$\left.+\frac{a_- b_- \exp[-i(\gamma(\kappa)+\gamma(J-\kappa-1))t]}{\gamma(\kappa)-\gamma(J-\kappa-1)}\right\} + \text{const.} + \text{other}$$

$$\text{terms in} \quad \exp[\pm i(\gamma(\kappa)\pm\gamma(J-\kappa+1))t]$$

To see what frequencies are present we look at the frequency of one term, e.g. the first term above. This term, T_1 is

$$T_1 = f_1 \exp[i(\gamma(\kappa)-\gamma(J-\kappa-1))]t$$

where $\gamma(\kappa)$ is

$$\gamma(\kappa) = \sqrt{\frac{\kappa-J}{2}\,g\tanh\left(\frac{\kappa-J}{2}h\right)}$$

We assume that h is large (deep water) and we have

$$\gamma(\kappa) \simeq \pm\frac{\kappa-J}{2}\sqrt{gh}$$

and

$$\gamma(J-\kappa-1) \simeq \pm\frac{\kappa+1}{2}\sqrt{gh}$$

The frequencies present are then

$$\omega_K = \gamma(K) - \gamma(J-K-1) = \left(\pm \frac{K-J}{2} \pm \frac{K+1}{2}\right)\sqrt{gh}$$

which are the sum and difference frequencies of all frequencies present. If we start with just a few frequencies we generate many more due to the nonlinearity of the problem. A study of the energy flow from one frequency to another is possible with the representation scheme used, but is quite tedious. We have shown here only part of the second order kernel, $R^{n}_{\alpha\beta}$ which is itself quite cumbersome, and the higher order kernels are even worse. The only saving grace is that once the kernels are found the problem is solved for arbitrary initial conditions.

APPENDIX B

NORMALITY TEST FLOW GRAPH

Input - X(I) series, I=1,LX

Compute mean

$$XMEAN = \sum_{I=1}^{LX} X(I)/LX$$

Compute standard deviation

$$STDEV = \left[\sum_{I=1}^{LX} (X(I) - XMEAN)^2 / LX \right]^{1/2}$$

Define NRANGE

$NRANGE = \sqrt{LX}$

(This is an arbitrary definition. NRANGE should be small enough so that at least 5 values of X(I) fall in each range.)

Find the X values which divide the normal density with mean XMEAN and standard deviation STDEV into NRANGE ranges of equal probability. Use SUBROUTINE NOINT2.

Returns LRANGE(=NRANGE-1) values for range limits, RANGE(1).

First range is $(-\infty$, RANGE(1)), 1st range is (RANGE(LRANGE),∞).

Count number of values falling in each range. Use SUBROUTINE FRQCT2.

Returns fixed point count of number in each range in vector ICOUNT(I).

Chi Square test

P=1/NRANGE=probability of falling in any range.

$$\chi^2 = \sum_{I=1}^{NRANGE} (ICOUNT(I) - P*LX)^2 / (P*LX)$$

Number of degrees of freedom=NRANGE-3. Use SUBROUTINE CHISQR.

Compute probability of exceeding χ^2 . Use SUBROUTINE KIINT1.

See APPENDIX G for program listings

APPENDIX C

EXPANSION OF EMPIRICAL PROBABILITY DENSITY FUNCTIONS ABOUT THE NORMAL DENSITY IN TERMS OF MOMENTS

It is possible to expand a probability density about the normal density if the moments higher than the mean and variance are known. It is not, however, guaranteed that the expansion will converge in all cases. If $F(x)$ is the probability distribution, and $f(x)$,

$$f(x) = \frac{dF(x)}{dx}$$

is the density and $\varphi(x)$ is the normal density,

$$\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

then the expansion in terms of the derivatives of the normal density, the Edgeworth series, is

$$f(x) = c_0 \varphi(x) + \frac{c_1}{1!} \varphi'(x) + \frac{c_2}{2!} \varphi''(x) + \ldots \qquad (C-1)$$

and will converge if the integral

$$\int_{-\infty}^{\infty} e^{-x^2/4} dF(x)$$

converges and if $f(x)$ is of bounded variation in $(-\infty, \infty)$ (Cramer, 1946). For our purposes we need not worry too much about the convergence. We only wish to see if we can approximate the distribution fairly well with just a few terms of the expansion.

It is now possible to obtain the coefficient C_n in terms of the moments. Remembering that the normal density, $\varphi(x)$ is the "generating function" for Hermite polynomials

$$\left(\frac{d}{dx}\right)^n e^{-x^2/2} = (-1)^n H_n(x)\, e^{-x^2/2} \tag{C-2}$$

where $H_n(x)$ is the nth order Hermite polynomial, and that the Hermite polynomials are orthagonal with respect to $\varphi(x)$

$$\int_{-\infty}^{\infty} H_m(x) H_n(x) \varphi(x) dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} H_m(x) H_n(x) e^{-x^2/2} dx$$

$$= \begin{cases} n! & \text{for } m = n \\ 0 & \text{for } m \neq n \end{cases} \tag{C-3}$$

we can now solve for the C_n .

Substituting $\varphi^{(n)}_{(x)} = (-1)^n H_n(x) \varphi(x)$ into equation (C-1) we have

$$f(x) = c_0 H_0(x) + c_1 \frac{(-1)}{1!} H_1(x) + c_2 \frac{(-1)^2}{2!} H_2(x) + \dots + \frac{c_n (-1)^n}{n!} H_n(x) \tag{C-4}$$

Multiplying both sides by $H_m(x)$ and integrating we have, because of (C-3),

$$c_m = (-1)^m \int_{-\infty}^{\infty} H_m(x) f(x)\, dx \tag{C-5}$$

Since $H_m(x)$ is a polynomial and $f(x)$ is a probability density the integral is simply a sum of moments. The moments (central moments) are μ_K where

$$\mu_K = \int_{-\infty}^{\infty} (\xi - m)^K f(\xi)\, d\xi \qquad \text{(C-6)}$$

and m is the mean. The unit normal density (zero mean, unit standard deviation) was assumed in this derivation so that $f(x)$ must be the function of the standardized variable $\frac{\xi - m}{\sigma}$ where σ is the standard deviation. This means that the r-th moment of the standardized variable is $\frac{\mu_r}{\sigma^r}$. Hence $C_0 = 1$, $C_1 = C_2 = 0$, $H_3(x) = x^3 - 3x$, and so from (C-5), $C_3 = -\frac{\mu_3}{\sigma^3}$ The rest of the C_n may be obtained from the $H_n(x)$ in the same manner. Thus

$$C_4 = \frac{\mu_4}{\sigma^4} - 3$$

$$C_5 = -\frac{\mu_5}{\sigma^5} + 10\frac{\mu_3}{\sigma^3}$$

$$C_6 = \frac{\mu_6}{\sigma^6} - 15\frac{\mu_4}{\sigma^4} + 30$$

The moments may be estimated from the data by averaging so that the integral (A-6) need not be performed.

The computation of the approximations using up to C_6 has been programmed by Roy Greenfield. (See SUBROUTINE PRBFIT in APPENDIX G.) The expressions for the approximations which must be evaluated are

$$f_1(x) = \left[1 + \frac{\mu_3}{6\sigma^3}(x^3 - 3x)\right] \varphi(x)$$

$$f_2(x) = f_1(x) + \left[\left(\frac{\mu_4}{24\sigma^4} + \frac{1}{8}\right)(x^4 - 6x^2 + 3)\right] \varphi(x)$$

$$f_n(x) = f_{n-1}(x) + \left[\frac{c_n}{n!}(-1)^n H_n(x)\right] \varphi(x)$$

Care must be taken that the X's are the values of the standardized variables.

APPENDIX D

INDEPENDENCE AND DEPENDENCE MEASURES

Poker Count Test for Independence

Given a series of equally likely integers from zero to nine it is possible, under the assumption that the numbers are independent, to compute the probable number of non-overlapping groups of five numbers which fall into each of eight categories. These categories are similar to those of a poker game where each group of five is considered a hand and each hand has a certain value. The analogy to the poker game is not completely accurate since the "card" values are 0 to 9 rather than ace to king, and it is possible to have five of a kind. Also the series, which takes the place of the card deck, has many more than 52 numbers in it, and removal of a number does not decrease its later probability of occurrence. The eight categories or hand types with their respective probabilities are (Durand, 1962, personal communication):

Hand	Probability
Bust	.2952
1 pair	.5040
2 pair	.1080
3 of a kind	.0720
Full house	.0090
Straight	.0072
4 of a kind	.0045
5 of a kind	.0001

These probabilities are exact. The decimals terminate at the fourth place. In assigning a hand to one of the categories the order of the digits within the group of five does not matter.

If the series of numbers is independent, then it is expected that the number of each type of hand will be approximately the probability for that hand times the total number of hands. Both this test and the mean square contingency test require a mapping of the given series into an integer series. The poker count test requires that the ten digits have equal probability. Hence the probability density of the original series is transformed into a rectangular density and the original series is mapped into an integer series with values from zero to nine with each integer having probability .1. Figure D-4 shows the steps necessary in the poker count test and APPENDIX G contains program listings.

Transformation of Probability Densities

Suppose $P_\xi(x) = f(x)$ is the probability density (frequency function) of a random variable ξ. The distribution function is then

$$Q(x) = \int_{-\infty}^{x} f(y)\,dy = F(x)$$

The change of variable, $y = F(x)$ is known as the "probability transformation" (Wadsworth and Bryan, 1960).

The probability density $P_\eta(y)$ can be found as follows:

$$P_\eta(y)\,dy = P_\xi(x)\,dx$$

$$P_\eta(y) = P_\xi(x)\,\frac{dx}{dy} = \frac{f(x)}{f(x)} = 1$$

The variable ξ is thus rectangularly distributed and, since $F(x)$ is defined from 0 to 1, $0 \leq y \leq 1$.

For the joint distribution, $P_{\xi_1 \xi_2}(x_1, x_2)$, using the same transformation, we have

$$P_{\xi_1 \xi_2}(x_1, x_2) = P_{\xi_1}(x_1)\, P_{\xi_2|\xi_1}(x_1, x_2)$$

where $P_{\xi_2|\xi_1}(x_2|x_1)\,dx_1\,dx_2$ denotes the compound probability that $x_2 < \xi_2 < x_2 + dx_2$ given that $x_1 < \xi_1 < x_1 + dx_1$.
Using the same transformation, $y = F(x)$, we have

$$P_{\eta_1 \eta_2}(y_1, y_2)\,dy_1\,dy_2 = P_{\xi_1}(x_1)\, P_{\xi_2|\xi_1}(x_2|x_1)\,dx_1\,dx_2$$

The Jacobian for this transformation, J, gives

$$dx_1\,dx_2 = |J|\,dy_1\,dy_2$$

$$J = \begin{bmatrix} \dfrac{\partial x_1}{\partial y_1} & \dfrac{\partial x_2}{\partial y_2} \\[2ex] \dfrac{\partial x_2}{\partial y_1} & \dfrac{\partial x_2}{\partial y_2} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{f(x)} & 0 \\[2ex] 0 & \dfrac{1}{f(x)} \end{bmatrix}$$

$$J = \frac{1}{[f(x)]^2}$$

$$P_{\eta_1 \eta_2}(y_1, y_2) = \frac{P_{\xi_1}(x_1)\, P_{\xi_2|\xi_1}(x_2|x_1)}{[f(x)]^2}$$

$$P_{\eta_1 \eta_2}(y_1, y_2) = \frac{P_{\xi_2|\xi_1}(x_2|x_1)}{f(x)}$$

If ξ_1 and ξ_2 are independent then

$$P_{\xi_2|\xi_1}(x_2|x_1) = P_{\xi_2}(x_2) = f(x)$$

and

$$P_{\eta_1\eta_2}(y_1, y_2) = 1$$

The result is that if ξ_1 and ξ_2 are independent, then η_1 and η_2 are also independent, and if ξ_1 and ξ_2 are dependent, then η_1 and η_2 are also dependent. The compound probabilities will differ by a factor equal to $\left|\frac{1}{f(x)}\right|$

$$P_{\eta_2|\eta_1}(y_2|y_1) = P_{\xi_2|\xi_1}(x_2|x_1)\left|\frac{1}{f(x)}\right|$$

If ξ_1 and ξ_2 are independent, then all of the higher probability densities for η are rectangular. An extension of this can easily be made for any number of random variables, and in particular for five variables as is necessary for the poker count test.

Mean Square Contingency and Dependency Measure

The measure of the degree of dependence of two variables which has been used is related to the mean square contingency (Cramer, 1951).

Suppose that two variables, ξ and η have densities $P_\xi(x_i)$ and $P_\eta(y_j)$ and a joint density $P_{\xi\eta}(x_i, y_j)$ where x_i and y_j are discrete and $i = 1, \ldots, N;\ j = 1, \ldots, M$.

Hence

$$\sum_i P_{\xi\eta}(x_i, y_j) = P_\eta(y_j)$$

$$\sum_j P_{\xi\eta}(x_i, y_j) = P_\xi(x_i)$$

The mean square contingency, φ^2 is defined as

$$\varphi^2 = \sum_i \sum_j \frac{\left(P_{\xi\eta}(x_i, y_j) - P_\xi(x_i) P_\eta(y_j)\right)^2}{P_\xi(x_i)\, P_\eta(y_j)}$$

$$= \sum_i \sum_j \frac{\left[P_{\xi\eta}(x_i, y_j)\right]^2}{P_\xi(x_i)\, P_\eta(y_j)} - 1$$

If and only if the variables are independent

$$P_{\xi\eta}(x_i, y_j) = P_\xi(x_i)\, P_\eta(y_j)$$

and $\varphi^2 = 0$.

Since

$$P_{\xi\eta}(x_i, y_j) = P_\xi(x_i)\, P_{\eta|\xi}(y_j|x_i) = P_\eta(y_j)\, P_{\xi|\eta}(x_i|y_j)$$

and all probabilities are less than or equal to one,

$$P_{\xi\eta}(x_i, y_j) \leq \begin{cases} P_\eta(y_j) \\ P_\xi(x_i) \end{cases}$$

thus

$$\sum_{i,j}^{N,M} \frac{P^2_{\xi\eta}(x_i, y_j)}{P_\xi(x_i)\, P_\eta(y_j)} \leq q$$

and

$$\varphi^2 \leq q-1$$

where q is the smaller of N and M , the limits of the sumation. Therefore the quantity $\varphi^2/(q-1)$, which we will call the dependency, may be used as a standard measure of dependence since

$$0 \leq \frac{\varphi^2}{q-1} \leq 1$$

There is, of course, some difficulty in using this or any measured dependence on numerical data. Numbers generated by independent random processes will not in general give a zero value for the dependency. The question arises, therefore, as to the interpretation of the number resulting from the dependency test. Since it is uncertain how large the dependency can be and the series still remain independent, a number of tests were run on independent random numbers. The numbers were obtained from the Rand Corporation on punched cards and are the same as the numbers which appear in the book, 1,000,000 Random Digits (Rand Corporation 1958). These numbers were generated by an independent process.

The numbers were run through both the poker count test and the dependency test. Three different lengths of series were used, 3000, 2500 and 2000, and each was repeated 8 times so that a mean and cariance could be computed. The results of the dependency test are shown in Figure D-1. Straight lines have been dotted in to indicate the mean and standard deviation changes with series length. There is no reason to suspect that their values actually fall on a straight line, in fact one would suspect that the line would curve off concave upward on the right and concave downward to the left. These tests were carried out for a lag of one, that is the random variables took on values of X_n and X_{n+1} of the series of digits.

Since it is important that the denominator not be zero, the series of real data were mapped into integer series from 1 to 10 with rectangular densities. This was, of course, not necessary with the Rand random digits, since they were already equally likely integers. However, one was added to each Rand digit so that the series would be from 1 to 10 rather than 0 to 9. This was necessary only for ease and speed of computation of the second probability density. Figures D-2 to D-5 show flow graphs of the steps necessary to compute the empirical probability density and perform the probability transformation, the poker count test and the mean square contingency test. APPENDIX G contains the listings of the programs used in these operations.

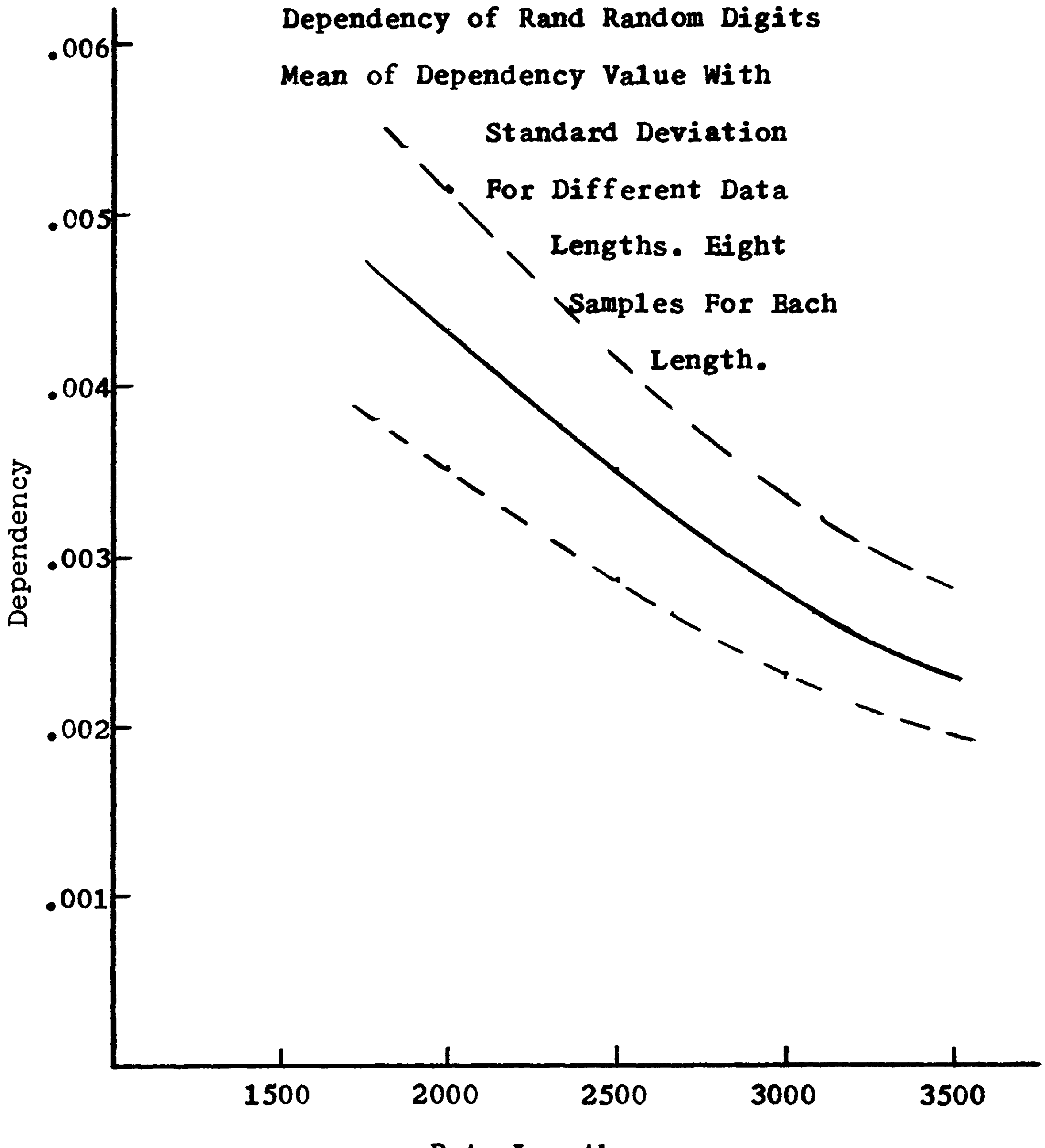

Dependency of Rand Random Digits
Mean of Dependency Value With
Standard Deviation
For Different Data
Lengths. Eight
Samples For Each
Length.
.006
.005
.004
.003
.002
.001
Dependency
1500
2000
2500
3000
3500
Data Length

Figure D-1

Empirical Probability Density Flow Graph

Inputs - X(I) series, I=1,LX

NDIV number of ranges

Find maximum, XMAX, and minimum, XMIN, of X series.

Compute range limits for NDIV equally spaced ranges from XMIN to XMAX

RANGE(I)=XMIN+(I-1) (XMAX-XMIN)/NDIV, I=1, NDIV+1

NDIV is somewhat arbitrary. It should be much smaller than LX, the length of the X series. We have used NDIV=100 with LX 2500.

Count number of values of X(I) falling in each of the NDIV ranges.

Use SUBROUTINE FRQCT2.

NOTE - FRQCT2 assumes that the NDIV+1 range limits define NDIV+2 ranges. The count vector, ICOUNT(I), I=1,NDIV+2, must therefore be altered such that ICOUNT(2)=ICOUNT(2)+ICOUNT(1), and ICOUNT(NDIV+1)=ICOUNT(NDIV+1)+ICOUNT(NDIV+2). The correct counts are then in ICOUNT(2) to ICOUNT(NDIV+1). This may then be normalized to give the frequency ratio or probability density, PROB(I).

PROB(I)=ICOUNT(I) NDIV/(LX (XMAX-XMIN))

Figure D-2

Probability Transformation Flow Graph

Rectangularize Probability Density

Inputs - PROB(I), I=1,NDIV, The probability density normalized such that

$$\sum_{I=1}^{NDIV} PROB(I)\,\Delta X = 1 ; \quad \Delta X = (XMAX - XMIN)/LX$$

XMIN = Minimum value of original time series

XMAX = Maximum value of original time series

NPROB = Number of ranges of equal probability desired.

Need not equal NDIV

X(I),I=1,LX, the time series

Find X limits which divide the empirical density into NPROB ranges of equal probability, XLIMIT(I), I=1,NPROB+1.

(Linear interpolation where necessary) Use SUBROUTINE GRUP2

Map X(I) series into IX(I) series (integer series such that for XLIMIT(J) X(I) XLIMIT(J+1), IX(I)=J-1+IXLO

where IXLO can be adjusted to give desired d.c. level.

Use SUBROUTINE MPSEQ1

Result is interger series IX(I), I=1,LX with NPROB different values from IXLO to IXLO+NPROB-1 with equal probability, 1/NPROB

Figure D-3

Poker Count Test Flow Graph

Inputs - X(I),I=1,LX time series

LX length of series

Compute empirical probability density. See Figure D-2 for flow graph of this procedure

Perform probability density transformation to map X(I) series into IX(I) series with

$$0 \leq IX(I) \leq 9$$

See Figure D-3 for flow graph of this procedure with IXLO=0.

Take IX(I) series in non-overlapping groups of 5,IX(I), I=1, ... 5,IX(I), I=6,.....10, etc and consider these as poker hands. Evaluate the poker hands and count number of each type. (Types - bust, 1 pair, 2 pair, 3 of a kind etc.) Total number of hands is LX/5 rounded down. USE SUBROUTINE POKCT1.

Compare with theoretical count for independent series.

(See a priori probabilities on first page of this APPENDIX.)

Figure D-4

Mean Square Contingency and Dependency Test Flow Graph

Inputs - X(I), I=1,LX time series

LX length of series

Compute empirical probability density. See Figure D-2 for flow graph of this procedure.

Perform probability density transformation to map X(I) series into IX(I) series with $1 \leq IX(I) \leq JHIGH$, where $JHIGH \leq 25$. (Requirement of SUBROUTINE PROB2 used below.)

Note - If poker count test is also done the mapped series used there can be used here if one is added to every IX value. JHIGH will be 10 for this case.

(See Figure D-3 for transformation and mapping flow graph.)

Compute second probability density, P(I,J) for lag of one.

Use SUBROUTINE PROB2. (Gives joint probability that IX(I)=L and IX(I+1)=M for I=1, LX-1, and M and $L \geq 1, \leq JHIGH$.)

Compute mean square contingency and dependency.

$$M.S.C. = \sum_{I=1}^{JHIGH} \sum_{J=1}^{JHIGH} \left[(P(I,J))^2 / (P(I) * P(J)) \right] - 1$$

where

$$P(I) = \sum_{J=1}^{JHIGH} P(I,J) \neq 0 \ , \ P(J) = \sum_{I=1}^{JHIGH} P(I,J) \neq 0$$

DEPENDENCY=M.S.C./ (JHIGH-1)

USE SUBROUTINE MSCON1.

Figure D-5

APPENDIX E

FACTORIZATION OF THE POWER SPECTRUM

The problem of spectrum factorization in the frequency domain was solved by Kolmogorov (1941). The treatment here is similar to Robinson (1956).

Given a power density spectrum, $\Phi(\omega)$, it is possible to factor it such that

$$\Phi(\omega) = B(\omega)\,\overline{B(\omega)}$$

where

$$B(\omega) = \sqrt{\Phi(\omega)}\,e^{i\theta(\omega)}$$

That this factorization is possible is quite obvious and, in fact, an infinite number of such factorization exist. The trivial case is $\theta(\omega) = 0$. There is, however, one important case, and that is when $B(\omega)$ has no poles or zeros in the lower half of the λ plane ($\lambda = \omega + i\sigma$ (Lee, 1960). In this case $B(\omega)$ corresponds to the transfer function of a physically realizable system, that is, a system which does not have output before it has input. A pole in the lower half of the λ plane transforms to the negative time axis and can therefore be considered a "source" for negative time. If $B(\omega)$ has poles in the lower half plane, its Fourier transform $B(t)$ will only be non-zero for $t \geq 0$, and $B(t)$ then said to be one-sided in positive time. If $B(\omega)$ also has no zeros in the lower half plane, then its inverse $1/B(\omega)$ will have no poles in the lower half plane and its Fourier transform will also be one-sided. $B(t)$ is then called the minimum phase wavelet. The factorization problem is the problem of finding $B(t)$ from $\Phi(\omega)$ and can be solved as follows.

If we take the z transform, i.e. $z = e^{-i\omega}$, of $B(\omega)$ to obtain $B(z)$, we have mapped the lower half of the place into the interior of the unit circle and we now consider $B(z)$ a polynomial in z. That is $B(\omega)$ is the Fourier transform of some time function $B(t)$ and as such has the form

$$B(\omega) = \sum_{s=-\infty}^{\infty} b_s e^{-i\omega s}$$

and the z transform becomes

$$B(z) = \sum_{s=-\infty}^{\infty} b_s z^s$$

and $B(z)$ must have no poles or zeros inside the unit circle, There are certain restrictions on $\Phi(\omega)$, namely

1. $\Phi(\omega) = 0$

2. $\int_{-\pi}^{\pi} \log \Phi(\omega)\, d\omega > -\infty$

3. $\int_{-\pi}^{\pi} \Phi(\omega)\, d\omega < \infty$

which must be met if $B(z)$ is to exist. If condition (1) is not met, then the integral (2) will not converge. Condition (2) is equivalent to the Paley-Wiener criterion (Robinson, 1954, p. 149) and is a requirement for the existence of a moving average and an autoregressive representation of the time series. Condition (3) states that the power must be finite and is just a stability requirement.

If these requirements are fulfilled, then the logarithm of $B(z)$ will be analytic for $|z| \leq 1$.

$$\log B(\omega) = \frac{1}{2} \log \Phi(\omega) + i\,\theta(\omega)$$

or

$$\log B(z) = u(z) + i\,v(z)$$

Hence the problem of obtaining the minimum phase wavelet is now one of finding the imaginary part, $v(z)$, of a function analytic inside the unit circle given the real part, $u(z)$, on the circle. This is also the potential theory problem of finding the field inside of a region given the sources on the boundary. The function $\log B(z)$ can be expressed as a power (Taylor) series in its region of analyticity

$$\log B(z) = \sum_{r=-\infty}^{\infty} \alpha_r z^r$$

Expanding $\log B(z) = \log B(re^{i\omega})$ in a Fourier series

$$\log B(re^{i\omega}) = u(re^{i\omega}) + i\,v(re^{i\omega})$$

$$= \sum r^k \alpha_k e^{i\omega k}, \quad \alpha_k = c_k + i d_k$$

$$u(re^{i\omega}) = \mathrm{Re}\left[\sum (c_k + i d_k) r^k e^{i\omega k}\right]$$

$$= \mathrm{Re}\left[\sum c_k \cos k\omega + i d_k \cos k\omega + i c_k \sin k\omega - d_k \sin k\omega\right] r^k$$

$$= \sum (c_k \cos k\omega - d_k \sin k\omega)\, r^k$$

However

$$u(re^{i\omega}) = \frac{1}{2}\log\Phi(\omega) \quad \text{at} \quad r = 1$$

and $\Phi(\omega)$ is an even function, i.e.

$$\Phi(\omega) = \Phi(-\omega)$$

since

$$\Phi(\omega) = \sum_s \varphi_s' \cos \omega s$$

Therefore 1/2 log $\Phi(\omega)$ is also even

and $d_K = 0$

Hence $\alpha_K = c_K$

and $\frac{1}{2}\log\Phi(\omega) = \sum \alpha_K \cos K\omega$

and $\alpha_K = \frac{1}{\pi}\int_{-\pi}^{\pi}\frac{1}{2}\log\Phi(\omega)\cos K\omega \, d\omega$

The wavelet b_S is then determined from

$$B(z) = \sum_{S=0}^{\infty} b_S z^S = \exp\left[\sum_{K=-\infty}^{\infty}\alpha_K z^K\right] = \exp\left[\sum_{K=-\infty}^{\infty}\frac{1}{4\pi}\int_0^{\pi}\log\Phi(\omega)\cdot \cos K\omega \, d\omega \, z^K\right]$$

The following method, suitable for programming purposes, for getting the b_S was first given in MIT G.A.G. Report 9 (1956) and was repeated in Simpson et al (1962a).

The b_S will have to be cut off after some S value, say $S = m$. It is shown below that the first $m+1$ terms of b_S (the first $m+1$ points in the wavelet) may be obtained exactly from the first $m+1$ α'S.

Expanding

$$\sum_{s=0}^{\infty} b_s z^s = e^{\alpha_0}\left[1+\frac{2\alpha_1}{1!}z+\left(\frac{2\alpha_1}{2!}\right)^2 z^2+\cdots\right]\left[1+\frac{2\alpha_2}{1!}z^2+\left(\frac{2\alpha_2}{2!}\right)^2 z^4\cdots\right]$$

$$\times\left[1+\frac{2\alpha_3}{1!}z^3+\left(\frac{2\alpha_3}{2!}\right)^2 z^6+\cdots\right]\left[\cdots\cdots\right]\cdots\cdots$$

$$\times\left[1+\frac{2\alpha_m}{1!}z^m+\left(\frac{2\alpha_m}{2!}\right)^2 z^{2m}+\cdots\cdots\right]\left[\cdots\cdots\right]\cdots$$

Matching like powers of z we find

$$b_0 = e^{\alpha_0}$$
$$b_1 = e^{\alpha_0}(2\alpha_1)$$
$$b_2 = e^{\alpha_0}\left[\left(\frac{2\alpha_1}{2!}\right)^2+\frac{2\alpha_2}{1!}\right]$$

etc.

In general, if we are interested in obtaining $b_0, \ldots b_m$, we may drop terms in any polynomial with exponents $> m$ and we may drop all polynomials whose first power of z is $> m$. We also do not care about any cross terms whose z exponents are $> m$.

We disregard e^{α_0} for the time being and consider the problem as follows:

$$\sum_{s=0}^{m} b_s z^s \quad = \text{(First } m+1 \text{ terms of) } P_1(z)\,P_2(z)\ldots P_m(z)$$

(this is just another way of grouping the terms).

Where $P_i = 1 + C_{i1} z + C_{i2} z^2 + \ldots + C_{im} z^m$

and

$$C_{ij} = \begin{cases} \left[\left(\frac{2\alpha_i}{1}\right)\left(\frac{2\alpha_i}{2}\right)\left(\frac{2\alpha_i}{3}\right)\cdots\left(\frac{2\alpha_i}{j/i}\right) & \text{for } j = k_i \\ 0 & \text{for } j \neq k_i \end{cases}$$

$$C_{i0} = 1$$

K is a positive integer. Considering b_s and C_{is} as time functions we may now consider the problem as one of partial convolution. Let F stand for "First $m+1$ terms of." Then

$$b = F(c_1 * c_2 * c_3 \ldots * c_m)$$

and

$$b = F(c_1 * F(c_2 * F(c_3 * \ldots F(c_{m-1} * c_m))) \ldots)$$

Let $b^{(m)} = C_m$

$$b^{(m-1)} = F(C_{m-1} * C_m) = F(C_{m-1} * b^{(m)})$$

$$b^{m-2} = F(C_{m-2} * F(C_{m-1} * C_m)) = F(C_{m-2} * b^{m-1})$$

$$b^{(1)} = F(c_1 * b^2) = b$$

Examination shows that $b^{(\ell-1)}$ may be obtained from $b^{(\ell)}$ by the following formula representing partial convolution

$$b_s^{(\ell-1)} = \sum_{i=0}^{s} C_{\ell-1,s-i}\, b_i^{(\ell)}$$

$$s = 0, 1, 2, \ldots, m$$

Further examination shows that $b^{(M)}$, where $M =$ 1+ integral part of $m/2$, may be written down by inspection

$$b_0^{(M)} = 1$$

$$b_1^{(M)} = 0$$

$$b_2^{(M)} = 0$$

$$\vdots$$

$$b_M^{(M)} = C_{M,M}$$

$$b_{M+1}^{(M)} = C_{M+1,M+1}$$

$$\vdots$$

$$b_m^{(M)} = C_{m,m}$$

This can be seen by noting first that $b_0^{(L)} = 1$ for all L and $b_s^{(L)} = 0$ for $1 < s < L$ and that the C_{LS} for $m/2 < L \leq m$ have only two terms in them. As the partial convolution proceeds, the b_0 terms pickup the diagonal terms in the C_{ij} matrix, and there are no other contributions to the next $b_s^{(L)}$ until $L \geq m/2$. It can be seen that only one column of the C_{ij} matrix is needed at a time.

A program has been written for the spectrum factorization problem for 709 or 7090 computers. The program makes sure that $\Phi(\omega) > 0$ by setting any value of $\Phi(\omega)$ which is less than 10^{-6} of the maximum value of $\Phi(\omega)$ equal to 10^{-6} of the maximum. The Daniell method of spectral estimation guarantees $\Phi(\omega) > 0$ but other spectral window such as the Turkey-Hamming window do not have the guarantee. The computation of the α's in the computation of the cosine expansion of $\frac{1}{2} \log \Phi(\omega)$ was done by trigonometric interpolation (Lanczos, 1956) so that the integral need not be computed. The program FACTOR is listed in APPENDIX G.

APPENDIX F

CONSTRUCTION OF THREE WHITE LIGHT SERIES WITH SPECIFIED COHERENCES

We wish to construct three unit variance white light series X_t^1, X_t^2, X_t^3 with controlled coherences

$$Coh_{12}(\omega) = \frac{|\Phi_{12}(\omega)|}{\sqrt{\Phi_{11}(\omega)\,\Phi_{22}(\omega)}} = \alpha_{12}(\omega)$$

$$Coh_{13}(\omega) = \frac{|\Phi_{13}(\omega)|}{\sqrt{\Phi_{11}(\omega)\,\Phi_{33}(\omega)}} = \alpha_{13}(\omega) \qquad \text{(F-1)}$$

$$Coh_{23}(\omega) = \frac{|\Phi_{23}(\omega)|}{\sqrt{\Phi_{22}(\omega)\,\Phi_{33}(\omega)}} = \alpha_{23}(\omega)$$

The solution is an obvious extension of the Simpson et al (1962) treatment of constructing a pair of series with controlled coherence. Since X_t^1, X_t^2, X_t^3 are unit variance white light their spectra are

$$\Phi_{11}(\omega) = \Phi_{22}(\omega) = \Phi_{33}(\omega) = \frac{1}{2\pi}$$

hence

$$\left| \Phi_{ij}(\omega) \right| = \frac{\alpha_{ij}(\omega)}{2\pi} \quad , \quad 1 \le i < j \le 3$$

or for zero phase shift

$$\Phi_{ij}(\omega) = \frac{\alpha_{ij}}{2\pi}$$

We assume that X_t^1 , X_t^2 and X_t^3 are broken up to have common and uncorrelated parts

$$\begin{aligned} X_t^1 &= X_t^{C_1} + X_t^{C_3} + X_t^{R_1} \\ X_t^2 &= X_t^{C_1} + X_t^{C_2} + X_t^{R_2} \\ X_t^3 &= X_t^{C_1} + X_t^{C_2} + X_t^{C_3} + X_t^{R_3} \end{aligned} \qquad \text{(F-2)}$$

where all cross correlations

$$\varphi_{C_i C_j} \ , \ \varphi_{R_i R_j} \ ; \ i \neq j$$

$$\varphi_{C_i R_j} \ ; \ i = 1, 2, 3 \ ; \ j = 1, 2, 3$$

are zero. The autospectra of the X_t^i series are then

$$\Phi_{11}(\omega) = \Phi_{c_1}(\omega) + \Phi_{c_3}(\omega) + \Phi_{R_1}(\omega) = \frac{1}{2\pi}$$

$$\Phi_{22}(\omega) = \Phi_{c_1}(\omega) + \Phi_{c_2}(\omega) + \Phi_{R_2}(\omega) = \frac{1}{2}\pi$$

$$\Phi_{33}(\omega) = \Phi_{c_1}(\omega) + \Phi_{c_2}(\omega) + \Phi_{c_3}(\omega) + \Phi_{R_3}(\omega) = \frac{1}{2}\pi$$

The cross-spectra are

$$\Phi_{12}(\omega) = \Phi_{c_1}(\omega) = \frac{\alpha_{12}(\omega)}{2\pi}$$

$$\Phi_{13}(\omega) = \Phi_{c_1}(\omega) + \Phi_{c_3}(\omega) = \frac{\alpha_{13}(\omega)}{2\pi}$$

$$\Phi_{23}(\omega) = \Phi_{c_1}(\omega) + \Phi_{c_2}(\omega) = \frac{\alpha_{23}(\omega)}{2\pi}$$

We therefore have

$$\Phi_{C_2}(\omega) = \frac{\alpha_{23}(\omega) - \alpha_{12}(\omega)}{2\pi}$$

$$\Phi_{C_3}(\omega) = \frac{\alpha_{13}(\omega) - \alpha_{12}(\omega)}{2\pi}$$

$$\Phi_{R_1}(\omega) = \frac{1 - \alpha_{13}(\omega)}{2\pi}$$

$$\Phi_{R_2}(\omega) = \frac{1 - \alpha_{23}(\omega)}{2\pi}$$

$$\Phi_{R_3}(\omega) = \frac{1 + \alpha_{12}(\omega) - \alpha_{23}(\omega) - \alpha_{13}(\omega)}{2\pi}$$

We must first construct the six mutually independent series $X_t^{C_i}$, $X_t^{R_i}$, i=1,2,3 with the power spectra Φ_{C_i}, Φ_{R_i} given above. We then construct the X_t^i series with equations F-2. These series have the coherences $\alpha_{ij}(\omega)$ as shown in equations F-1.

APPENDIX G

PROGRAM LISTINGS

Listings, with descriptions and examples, of some of the more important programs used in the computations in this thesis. The listings are in alphabetical order and include all subroutines appearing in the transfer vectors with the exception of the FORTRAN System routines. An index of these programs and other programs useful in time series analysis appears in Scientific Report Number 4 of Contract AF 19(604)7378 (Simpson et al, 1962b) and complete listings will appear (Simpson, 1963, in press) in book form in the near future. All the programs appearing here are designed to operate under the FORTRAN-II system for the IBM 709-7090 computers.

Throughout the listings the terms FORTRAN INTEGER, FORTRAN II INTEGER, and INTEGER are synonomous and refer to a fixed point integer in the decrement. The terms MACHINE LANGUAGE INTEGER, MACHINE INTEGER and MLI refer to a fixed point integer in the decrement. The terms LSTHN and LSTHN = are equivalent to $<$ and $\leq$ while GRTHN and GRTHN = are equivalent to $>$ amd $\geq$. It should be noted that expressions which appear in the "ABSTRACT" section of the writeup may deviate from the usual FORTRAN conventions.

```
*     CHISQR (SUBROUTINE)                2/18/63   LAST CARD IN DECK IS NO. 0084
*     LABEL
CCHISQR
      SUBROUTINE CHISQR(NBLOCS,ICOUNT,N,CHISQ,IANS)
C
C                      ----ABSTRACT----
C
C TITLE - CHISQR
C         COMPUTES CHI-SQUARE FOR EQUALLY LIKELY PROBABILITY CASE.
C
C              CHISQR COMPUTES CHI SQUARE WHEN GIVEN THE DISTRIBUTION
C              COUNT AND THE NUMBER OF EQUALLY LIKELY BLOCKS INTO WHICH
C              THE DATA IS PUT. NUMBER OF BLOCKS = NBLOCKS, N = TOTAL
C              NUMBER OF OBSERVATIONS, ICOUNT = DISTRIBUTION COUNT.
C
C              CHISQ=SUM((ICOUNT(I)-N/NBLOCKS)**2/(N/NBLOCKS))
C
C              SUMMED OVER NBLOCKS, WHERE FLOATING OPERATIONS ARE ASSUMED
C              RATHER THAN THE INDICATED INTEGER OPERATIONS.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
C STORAGE   - 1C5 REGISTERS
C SPEED     -
C AUTHOR    - J.N. GALBRAITH
C
C                      ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C        AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C      CALL CHISQR(NBLOCS,ICOUNT,N,CHISQ,IANS)
C
C INPUTS
C
C    NBLOCKS   IS NUMBER OF EQUALLY LIKELY BLOCKS.
C              MUST BE GRTHN 1.
C
C    ICOUNT(I) I=1...NBLOCKS IS THE DISTRIBUTION COUNT. I.E. THE NUMBER
C              OF VALUES IN I-TH EQUALLY LIKELY BLOCK.
C              MUST BE NON-NEGATIVE
C
C    N         IS TOTAL NUMBER OF OBSERVATIONS (=SUM(ICOUNT(I))).
C              MUST BE GRTHN=1.
C
C OUTPUTS
C
C    CHISQ     IS THE CHI-SQUARE VALUE
C
C    IANS      =0  NORMAL
C              =1  ILLEGAL NBLOCS
C              =2  ILLEGAL N
C
C EXAMPLES
C
C 1. INPUTS  - NBLOCS=3   ICOUNT(1...3)=1,3,5   N=9
C    OUTPUTS - CHISQ=2.666667   IANS=0
C
C 2. INPUTS  - NBLOCS=1   ICOUNT(1)=1   N=9
C    OUTPUTS - ERROR   IANS=1
C
C 3. INPUTS  - NBLOCS=3   ICOUNT(1...3)=1,3,5   N=0
C    OUTPUTS - ERROR   IANS=2
C
C 4. INPUTS  - NBLOCS=5   ICOUNT(1...5)=1,2,3,4,5   N=15
C    OUTPUTS - CHISQ=3.333333   IANS=0
C
      DIMENSION ICOUNT(100)
      IANS=0
      IF(NBLOCS-1) 990,990,5
 5    IF(N) 992,992,10
 10   P=1./FLOATF(NBLOCS)
      EXPNO=P*FLOATF(N)
      CHISQ=0
```

CHISQR (PAGE 2)

```
      DO 25  I=1,NBLCCS
      DIF=FLOATF(ICOUNT(I))-EXPNO
   25 CHISQ=CHISQ+DIF*DIF
      CHISQ=CHISQ/EXPNO
   26 RETURN
  990 IANS=1
      GO TO 26
  992 IANS=2
      GO TO 26
      END
```

```
*       COSP (SUBRCUTINE)               2/18/63   LAST CARD IN DECK IS NO. 0844
*       FAP
*COSP
        CCUNT   1CC0
        LBL     COSP
        ENTRY   COSP (SSX,ASX,L,COSTAB,M,JMIN,JMAX,TYPE,COSTR)
        ENTRY   SISP (SAX,AAX,L,SINTAB,M,JMIN,JMAX,TYPE,SINTR)
        ENTRY   CCSISP (SSX,ASX,SAX,AAX,L,COSTAB,SINTAB,M,JMIN,JMAX,TYPE,
*                       CCSTR,SINTR)
*
*                    ----ABSTRACT----
*
* TITLE - COSP  WITH SECCNDARY ENTRY POINTS SISP AND COSISP
*        FAST COSINE AND/OR SINE TRANSFORMS FROM 2 OR 4 EVEN-ODD PARTS
*
*              COSP CCMPUTES COSINE SUMS, CT(J) J=JMIN,...,JMAX , ON
*              TWC INPUT SERIES, SS(I) AND AS(I) I=0,1,...,L , ACCORDING
*              TO             L
*                             SUM ( SS(I)*COS(I*J*(PI/M)) )     J EVEN
*                             I=0
*                 CT(J) =
*                             L
*                             SUM ( AS(I)*COS(I*J*(PI/M)) )     J ODD
*                             I=0
*
*                   FCR J =JMIN,JMIN+1,...,JMAX
*                WHERE
*                     PI = 3.14159265
*                     M = INPUT PARAMETER
*                     COS(I*(PI/M))  I=C,1,...,M  IS AN INPUT TABLE
*                     SS(I),AS(I), MAY BE EITHER FIXED OR FLOATING POINT
*                         (THE COSINE TABLE MUST CORRESPOND IN TYPE)
*                     C LSTHN= JMIN LSTHN JMAX LSTHN= M
*
*            . SISP CCMPUTES SINE SUMS, ST(J)
*                             L
*                             SUM ( AA(I)*SIN(I*J*(PI/M)) )      J EVEN
*                             I=0
*                 ST(J) =
*                             L
*                             SUM ( SA(I)*SIN(I*J*(PI/M)) )      J ODD
*                             I=0
*
*                   FCR J = JMIN,JMIN+1,...,JMAX
*                WHERE
*                     SIN(I*(PI/M))  I=0,1,...,M  IS AN INPUT TABLE
*                     AA,SA, AND THE SINE TABLE ARE FIXED OR FLOATING
*
*              COSISP COMPUTES BOTH CT(J) AND ST(J) AS DEFINED ABOVE
*
*              NOTE THAT THE FUNDAMENTAL FREQUENCY AS DEFINED BY THE
*              INPUT TABLES HAS PERIOD = EVEN NO. OF POINTS = 2M
*
* LANGUAGE  - FAP SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709 CR 709C (MAIN FRAME ONLY)
* STORAGE   - 492 REGISTERS
* SPEED     -        709-FIXED PT    709-FLOATING PT
*              COSP    34*K*(L+1)     37*K*(L+1)          MACHINE CYCLES
*              SISP    39*K*(L+1)     43*K*(L+1)          MACHINE CYCLES
*              CCSISP  67*K*(L+1)     72*K*(L+1)          MACHINE CYCLES
*                 WHERE  K = JMAX-JMIN+1
*                   (REDUCE ESTIMATES ABOUT 1C PERCENT FOR 7090)
* AUTHOR    - S.M. SIMPSON,  OCT 26, 61
*
*                    ----USAGE----
*
* TRANSFER VECTOR CCNTAINS ROUTINES -   NONE
*       AND FORTRAN SYSTEM ROUTINES -   NONE
*
* FORTRAN USAGE OF COSP
*     CALL COSP  (SSX,ASX,L,COSTAB,M,JMIN,JMAX,TYPE,COSTR)
*
* INPUTS TO COSP
*
```

```
*    SSX(I)     I=1...L+1  CONTAINS SS(J) J=0,1,...,L  FIXED OR FLOATING
*
*    ASX(I)     I=1...L+1  CONTAINS AS(J) J=0,1,...,L  FIXED OR FLOATING
*               EQUIVALENCE (SSX,ASX) IS PERMITTED
*
*    L          MUST EXCEED 0
*
*    COSTAB(I) I=1...M+1  CONTAINS COS(J*PI/M)  J= 0,1,...,M
*               COSTAB IS FIXED OR FLOATING
*                  FOR FIXED POINT IT IS ASSUMED THAT THE BINARY POINT
*                  IS BETWEEN THE SIGN BIT AND BIT 1 SO THAT VALUES
*                  1.0 AND -1.0 SHOULD BE ENTERED AS OCTAL 377777777777
*                  AND 777777777777 RESPECTIVELY. THE BINARY POINT OF
*                  SSX AND ASX IS IMMATERIAL, BUT OVERFLOW MAY ARISE.
*
*    M          MUST EXCEED 0
*
*    JMIN       DEFINES  LOWEST MULTIPLE OF FUNDAMENTAL DESIRED
*               MUST BE GRTHN= 0  AND LSTHN= JMAX
*
*    JMAX       DEFINES  HIGHEST MULTIPLE OF FUNDAMENTAL DESIRED
*               MUST BE GRTHN JMIN AND LSTHN= M
*
*    TYPE       = 0.0  SIGNIFIES  SS,AS, AND COSTAB ARE FIXED PT.
*               NOT= 0.0  MEANS   SS,AS, AND COSTAB ARE FLTG. PT.
*
* OUTPUTS FROM COSP
*
*    COSTR(I)   I=1...JMAX-JMIN+1 CONTAINS CT(J) J=JMIN...JMAX  AS
*                 DEFINED IN ABSTRACT.
*
*               (PROGRAM EXITS WITHOUT COMPUTATION IF L,M,JMIN,
*                 OR JMAX ILLEGAL)
*
* FORTRAN USAGE OF SISP
*      CALL SISP  (SAX,AAX,L,SINTAB,M,JMIN,JMAX,TYPE,SINTR)
*
* INPUTS TO SISP
*
*    SAX(I)     I=1...L+1  CONTAINS  SA(J) J=0,1,...,L
*
*    AAX(I)     I=1...L+1  CONTAINS  AA(J) J=0,1,...,L
*               EQUIVALENCE (SAX,AAX) IS PERMITTED.
*
*    L          SAME MEANING AS FOR COSP
*
*    SINTAB(I) I=1...M+1 CONTAINS SIN(J*PI/M)  J=0,1,...,M
*
*    M          SAME MEANING AS FOR COSP
*
*    JMIN       SAME MEANING AS FOR COSP
*
*    JMAX       SAME MEANING AS FOR COSP
*
*    TYPE       SAME MEANING AS FOR COSP
*
* OUTPUTS FROM SISP
*
*    SINTR(I)   I=1...JMAX-JMIN+1 CONTAINS ST(J) J=JMIN...JMAX AS
*                 DEFINED IN ABSTRACT
*
* FORTRAN USAGE OF COSISP
*      CALL COSISP(SSX,ASX,SAX,AAX,L,COSTAB,SINTAB,M,JMIN,JMAX,
*    1              TYPE,COSTR,SINTR)
*
*               WHERE ARGUMENTS ARE THE SAME AS FOR COSP AND SISP
*               EQUIVALENCE (SSX,ASX,SAX,AAX) IS PERMITTED.
*
* EXAMPLES
*
* 1. USE OF COSP, SISP, COSISP  WHEN ALL INPUTS EQUATED, FIXED AND
*               FLOATING, ALL FREQUENCIES
*    INPUTS  - X(1...4) = 1.,2.,3.,4.   IX(1...4) = 100,200,300,400  L=3
*               COSTAB(1...3)=1.0,0.0,-1.0  SINTAB(1...3)=0.0,1.0,0.0 M=2
*               ICOSTB(1...3)=OCT377777777777,000000000000,777777777777
```

```
*              ISINTB(1...3)=OCT000000000000,377777777777,000000000000
*              JMIN = 0    JMAX = 2
*     USAGE   -        CALL COSP (X,X,L,COSTAB,M,JMIN,JMAX,1.,C1)
*                      CALL COSP (IX,IX,L,ICOSTB,M,JMIN,JMAX,0.,IC1)
*                      CALL SISP (X,X,L,SINTAB,M,JMIN,JMAX,1.,S1)
*                      CALL SISP (IX,IX,L,ISINTB,M,JMIN,JMAX,0.,IS1)
*                      CALL COSISP (X,X,X,X,L,COSTAB,SINTAB,M,JMIN,JMAX,
*                                  1.,C2,S2)
*                      CALL COSISP (IX,IX,IX,IX,L,ICOSTB,ISINTB,M,JMIN,
*                                  JMAX,C.,IC2,IS2)
*     OUTPUTS - C1(1...3) = C2(1...3) = 10.,-2.,-2.
*              S1(1...3) = S2(1...3) = 0.,-2.,0.
*              IC1(1...3) = IC2(1...3) = 1000,-200,-200
*              IS1(1...3) = IS2(1...3) = 0,-200,0
*
* 2. PARTIAL FREQUENCY COVERAGE
*     INPUTS  - SAME AS EXAMPLE 1. EXCEPT JMIN = 1
*     USAGE   - SAME AS EXAMPLE 1.
*     OUTPUTS - C1(1...2) = C2(1...2) = -2.,-2.
*              S1(1...2) = S2(1...2) = -2.,0.
*              IC1(1...2) = IC2(1...2) = -200,-200
*              IS1(1...2) = IS2(1...2) = -200,0
*
* 3. USE OF COSISP TO FIND COEFFICIENTS OF TRIGONOMETRICAL SERIES FOR
*              AN EVEN-LENGTH VECTOR
*       (SEE CARSLAW, 1930, FOURIER SERIES AND INTEGRALS, P324,325)
*              GIVEN XX(I)  I=1...2*M  CONTAINING X(J) J=0,1,...,2*M-1
*              FIND A(0),A(1),...A(M) AND B(1),B(2),...,B(M-1) SUCH THAT
*
*              X(J)=A(0)+A(1)COS(J*D)+...+A(M-1)COS((J-1)*D)+A(M)COS(PI)
*                       +B(1)SIN(J*D)+...+B(M-1)SIN((J-1)*D)
*                   WHERE D=PI/M   J=0,1,...,2*M-1
*              SOLUTION
*     INPUTS  - COSTAB(1...M+1) = COS(J*PI/M)  J = 0,1,...,M
*              SINTAB(1...M+1) = SIN(J*PI/M)  J = 0,1,...,M
*              L = 2*M-1
*     USAGE   -        CALL COSISP(X,X,X,X,L,COSTAB,SINTAB,M,0,M,1.,AA,BB)
*                      AA(1) = AA(1)/FLOATF(2*M)
*                      AA(M+1) = AA(M+1)/FLOATF(2*M)
*                      DO 10  I=2,M
*                      AA(I)=AA(I)/FLOATF(M)
*                   10 BB(I)=BB(I)/FLOATF(M)
*     OUTPUTS - AA(1...M+1) WILL CONTAIN  A(0),A(1),...A(M)  AS REQUIRED
*              BB(2...M) WILL CONTAIN  B(1),...B(M-1)  AS REQUIRED
*              (BB(1)=BB(M+1)=0.)
*
* 4. USE OF COSISP TO INVERT COEFFICIENTS OF TRIG SERIES FOR AN EVEN-
*              LENGTH VECTOR
*              GIVEN A(0),...A(M)  B(1)...B(M-1)  AS DEFINED ABOVE
*              FIND X(J) = TRIG SERIES ABOVE    J = 0,1,...,2*M-1
*              SOLUTION
*     INPUTS  - AA(I) AND BB(I) ARE SAME AS OUTPUTS OF EXAMPLE 3.
*     USAGE   -        CALL COSISP(AA,AA,BB,BB,M,COSTAB,SINTAB,
*                     1            M,0,M,1.,XS,XA)
*                      I2M=2*M
*                      DO 20 I=2,M
*                      J=I2M+2-I
*                      XS(J)=XS(I)
*              20      XA(J)=-XA(I)
*                      DO 30 I=1,I2M
*              30      XBAC(I)=XA(I)+XS(I)
*     OUTPUTS - XBAC(1...2*M) WILL CONTAIN  X(0,1,...,2*M-1)  AS REQUIRED
*
* 5. ILLUSTRATION OF FINDING TRIG SERIES
*     INPUTS  - SAME AS EXAMPLE 1.
*     USAGE   - SAME AS EXAMPLE 3.
*     OUTPUTS - AA(1...3) = 2.5,-1.,-.5
*              BB(1...3) = 0.,-1.,0.
*
* 6. ILLUSTRATION OF INVERTING TRIG SERIES
*     INPUTS  - SAME AS EXAMPLE 5. WITH AA,BB, SAME AS OUTPUTS FROM EX 5.
*     USAGE   - SAME AS EXAMPLE 4.
*     OUTPUTS - XBAC(1...4) = 1.,2.,3.,4.
*
* 7. USE OF SYMMETRIES TO REDUCE TIME IN COMPUTING TRANSFORMS ABOUT
```

```
*              MIDPOINT OF AN ODD-LENGTH SERIES
*              GENERAL FORM
*                                  I=M
*                        C(J) = SUM ( X(I)*COS(I*J*PI/M) )
*                                  I=-M
*                AND
*                                  I=M
*                        S(J) = SUM ( X(I)*SIN(I*J*PI/M) )
*                                  I=-M
*                            J = JMIN...JMAX
*              SUPPOSE  X(-6...6)=1.,3.,1.,2.,1.,1.,5.,4.,3.,3.,5.,4.,1.
*              FIRST SPLIT X ABOUT ITS MIDPOINT INTO ITS SYMMETRIC AND
*                ANTISYMMETRIC PARTS
*                     SX(1...7) = 5.,5.,4.,5.,6.,7.,7.
*                     AX(1...7) = 0.,3.,2.,1.,4.,1.,0.
*              THEN SPLIT EACH OF THESE ABOUT THEIR MIDPOINTS
*                SSX(1...4) = 5.,10.,12.,7.  ASX(1...4) = 0.,2.,2.,-3.
*                SAX(1...4) = 1.,6.,4.,0.  AAX(1...4) = 0.,2.,-2.,0.
*    INPUTS  - THEN REVERSE ALL THE VECTORS AND CHANGE SIGNS OF ASX
*              AAX TO GIVE
*                SSX(1...4) = 7.,12.,10.,5.  ASX(1...4) = 3.,-2.,-2.,0.
*                SAX(1...4) = 0.,4.,6.,1.  AAX(1...4) = 0.,2.,-2.,0.
*                L=3  M=6  COSTAB(1...7)=COS(J*PI/6)
*                          SINTAB(1...7)=SIN(J*PI/6)  J = 0...6
*    USAGE   -          CALL COSISP (SSX,ASX,SAX,AAX,3,COSTAB,SINTAB,M,0,M,
*                                 1.,COSTR,SINTR)
*    OUTPUTS - COSTR(1...7) = C(0...6) = 34.,.26795,3.,5.,1.,3.73205,0.
*              SINTR(1...7) = S(0...6) = 0.,8.19615,0.,3.,3.46410,
*                                             -2.19615,0.
*
* PROGRAM FOLLOWS BELOW
*      NOTATION DIFFERENCES IN PROGRAM NOTES ARE
*      RSS=SSX       PAS=ASX       KAA=AAX       RSA=SAX
*      P=L
*
*
        HTR     0
        BCI     1,COSP
 COSP   SXD     *-2,4          SET UP EXIT
        SXA     LV+1,1
        SXA     LV+2,2
        CLA     K10
        STA     EXIT
*SET ARGUMENT TABLE
        CLA     1,4
        STA     T1
        CLA     2,4
        STA     T2
        CLA*    3,4
        STD     T5
        CLA     4,4
        STA     T6
        CLA*    5,4
        STD     T8
        CLA*    6,4
        STD     T9
        CLA*    7,4
        STD     T10
        CLA*    8,4
        STO     T11
        CLA     9,4
        STA     T12
*SET COSP SWITCHES
        CLA     KA18           KA6
        STA     Z30
        CLA     KA6            Z90
        STA     Z33
        CLA     KA15           Z107
        STA     Z106
        CLA     KA19           Z130
        STA     Z109B
        CLA     KT1            TRA  Z104
        STO     Z114
        STO     Z112
        CLA     KT2            TRA  Z102
```

```
************************
*    COSP              *
************************
                (PAGE  5)
```


```
          STO     Z121A
          STO     Z122A
          TRA     Z14
 *SET EXIT
  SISP    SXD     COSP-2,4
          SXA     LV+1,1
          SXA     LV+2,2
          CLA     K10
          STA     EXIT
 *SET ARGUMENT TABLE
          CLA     1,4
          STA     T3
          CLA     2,4
          STA     T4
          CLA*    3,4
          STD     T5
          CLA     4,4
          STA     T7
          CLA*    5,4
          STD     T8
          CLA*    6,4
          STD     T9
          CLA*    7,4
          STD     T10
          CLA*    8,4
          STO     T11
          CLA     9,4
          STA     T13
 *SET SISP SWITCHES
          CLA     KA14           KA9
          STA     Z30
          CLA     KA9            Z50
          STA     Z33
          CLA     KA7            Z100
          STA     Z56
          STA     Z66
          STA     Z76
          STA     Z86
          CLA     KA16           Z115
          STA     Z106
          CLA     KZ1            ZET  SWE
          STO     Z114
          STO     Z112
          CLA     KZ2            ZET  SWO
          STO     Z121A
          STO     Z122A
          TRA     Z14
 *SET EXIT
 COSISP   SXD     COSP-2,4       SET UP EXIT
          SXA     LV+1,1
          SXA     LV+2,2
          CLA     K14
          STA     EXIT
 *SET UP ARGUMENT TABLE
          CLA     1,4
          STA     T1
          CLA     2,4
          STA     T2
          CLA     3,4
          STA     T3
          CLA     4,4
          STA     T4
          CLA*    5,4
          STD     T5
          CLA     6,4
          STA     T6
          CLA     7,4
          STA     T7
          CLA*    8,4
          STD     T8
          CLA*    9,4
          STD     T9
          CLA*    10,4
          STD     T10
          CLA*    11,4
```

```
        STO     T11
        CLA     12,4
        STA     T12
        CLA     13,4
        STA     T13
*SET COSISP SWITCHES
        CLA     KA14          KA9
        STA     Z30
        CLA     KA9           Z50
        STA     Z33
        CLA     KA6           Z90
        STA     Z56
        STA     Z66
        STA     Z76
        STA     Z86
        CLA     KA15          Z107
        STA     Z106
        CLA     KZ1           ZET  SWE
        STO     Z114
        STO     Z112
        CLA     KZ2           ZET  SWO
        STO     Z121A
        STO     Z122A
        CLA     KA16          Z115
        STA     Z109B
        TRA     Z14
*MAKE COMMON SETTINGS FOR COSP, SISP, COSISP AS IF IT WERE COSISP
*FIRST FOR FIXED POINT OR FLOATING POINT
 Z14    ZET     T11
        TRA     Z15           FLOATING
        CLA     MPY           FIXED
        LDQ     ADD
        TRA     Z16
 Z15    CLA     FMP           FLOATING
        LDQ     FAD
 Z16    STO     Z51
        STO     Z61
        STO     Z71
        STO     Z81
        STO     Z91
        STQ     Z52
        STQ     Z62
        STQ     Z72
        STQ     Z82
        STQ     Z92
        STO     Z54
        STO     Z64
        STO     Z74
        STO     Z84
        STO     Z94
        STQ     Z55
        STQ     Z65
        STQ     Z75
        STQ     Z85
        STQ     Z95
        CLA     KA2           SMSE
        STA     Z52
        STA     Z62
        STA     Z72
        STA     Z82
        CLA     KA3           SMSO
        STA     Z55
        STA     Z65
        STA     Z75
        STA     Z85
        CLA     KA4           SMCE
        STA     Z92
        CLA     KA5           SMCO
        STA     Z95
*THEN ADDRESSES
        CLA     T7            SINTAB (OR HASH)
        STA     Z50
        STA     Z53
        STA     Z60
        STA     Z63
```

```
          STA     Z70
          STA     Z73
          STA     Z80
          STA     Z83
          CLA     T4           RAA  (OR HASH)
          STA     Z51
          STA     Z61
          STA     Z71
          STA     Z81
          CLA     T3           RSA  (OR HASH)
          STA     Z54
          STA     Z64
          STA     Z74
          STA     Z84
          CLA     T6           COSTAB  (OR HASH)
          STA     Z90
          STA     Z93
          CLA     T1           RSS  (OR HASH)
          STA     Z91
          CLA     T2           RAS  (OR HASH)
          STA     Z94
          CLA     T8           M
          TMI     LV
          TZE     LV
          STD     Z101
          STD     Z103
          ADD     T8           2M
          STD     2M
          CLA     T5           P
          TMI     LV
          TZE     LV
          STD     Z105
          CLA     T12          COSTR  (OR HASH)
          STA     Z108
          STA     Z109A
          CLA     T13          SINTR  (OR HASH)
          STA     Z116
          STA     Z118
*FOR JMIN EVEN SET JE=JMIN+1,JO=JMIN+1,ESTOR=0,OSTOR=1
*    JMIN ODD  SET JO=JMIN,JE=JMIN+1),OSTOR=0,ESTOR=1
 Z20      CLA     T9           JMIN
          TMI     LV
          CAS     T10
          TRA     LV
          TRA     LV
          ARS     18
          LBT
          TRA     Z21          IS EVEN
          ALS     18           IS ODD
          STD     JO
          ADD     KD1
          STD     JE
          STZ     OSTOR
          CLA     K1
          STA     ESTOR
          TRA     Z23
 Z21      ALS     18           IS EVEN
          STD     JE
          ADD     KD1
          STD     JO
          STZ     ESTOR
          CLA     K1
          STA     OSTOR
*CLEAR DUMMY SWITCHES
 Z23      STZ     DUME
          STZ     DUMO
*NOW BEGIN LOOPING
*INITIALIZE Z105 SWITCH, CLEAR SUM REGISTERS, SET TRAVEL SWITCHES
*  FORWARD
 Z30      CLA     **           (**=KA6 COSP,  **=KA9 OTHERWISE)
          STA     Z105
          STZ     SMSE
          STZ     SMSO
          STZ     SMCE
          STZ     SMCO
```

```
        STZ     SWE
        STZ     SWO
        CLA     JE
        STD     Z100
        CLA     JO
        STD     Z102
*SET MINUS JE,JO
        LDC     JE,1
        SXD     MJE,1
        LDC     JO,1
        SXD     MJO,1
*XR4 WILL CONTROL MOTION FOR EVEN HARMONIC INDEX
*XR2 WILL CONTROL MOTION FOR ODD  HARMONIC INDEX
*XR1 WILL CONTROL MOTION FOR DATA INDEX
*DATA INDEX=SINE INDEX=COSINE INDEX=0
        AXT     0,7
 Z33    TRA     **              (**=Z90 FOR COSP,  =Z50  OTHERWISE)
*LOOP FOR FORWARD MOTION ON SINE WAVE FOR BOTH HARMONICS
*    THIS PART IS FOR EVEN HARMONICS (XR4) SUMMED IN SMSE
 Z50    LDQ     **,4            (**=SINTAB)
 Z51    NOP                     (MPY OR FMP $$,1  WITH ** = RAA)
 Z52    NOP                     (ADD OR FAD SMSE)
        STO     SMSE
*   THIS PART IS FOR ODD HARMONICS (XR2),  SUMMED IN SMSO
 Z53    LDQ     **,2            (**=SINTAB)
 Z54    NOP                     (MPY OR FMP  **,1  WITH **=RSA)
 Z55    NOP                     (ADD OR FAD SMSO
        STO     SMSO
*NOW GO TO COSINE SUMS IF COSISP, OR AVOID IF SISP
 Z56    TRA     **              (**=Z90 FOR COSISP,  **=Z100 FOR SISP)
*LOOP FOR FORWARD MOTION ON SINE WAVE OF EVEN HARMONIC AND
*    REVERSE MOTION ON SINE WAVE OF ODD HARMONIC
*   FOR EVEN
 Z60    LDQ     **,4            (**=SINTAB)
 Z61    NOP                     (MPY OR FMP **,1  WITH **=RAA)
 Z62    NOP                     (ADD OR FAD  SMSE)
        STO     SMSE
*   FOR ODD
 Z63    CLS     **,2            (**=SINTAB)
        XCA
 Z64    NOP                     (MPY OR FMP  **,1  WITH **=RSA)
 Z65    NOP                     (ADD OR FAD SMSO)
        STO     SMSO
 Z66    TRA     **              (**=Z90 IF COSISP, **=Z100 IF SISP)
*LOOP FOR REVERSE MOTION ON SINE WAVE OF EVEN HARMONIC AND
*     FORWARD MOTION ON SINE WAVE OF ODD HARMONIC
*   FOR EVEN
 Z70    CLS     **,4            (**=SINTAB)
        XCA
 Z71    NOP                     (MPY OR FMP  **,1  WITH **=RAA)
 Z72    NOP                     (ADD OR FAD SMSE)
        STO     SMSE
*   FOR ODD
 Z73    LDQ     **,2            (**=SINTAB)
 Z74    NOP                     (MPY OR FMP **,1  WITH **=RSA)
 Z75    NOP                     (ADD OR FAD SMSO)
        STO     SMSO
  Z76   TRA     **              (**=Z90 COSISP,  **=Z100 IF SISP)
*LOOP FOR REVERSE MOTION ON SINE WAVE FOR BOTH HARMONICS
*    THIS PART IS FOR EVEN HARMONICS
 Z80    CLS     **,4            (**=SINTAB)
        XCA
 Z81    NOP                     (MPY OR FMP  **,1  WITH **=RAA)
 Z82    NOP                     (ADD OR FAD SMSE)
        STO     SMSE
*    THIS PART IS FOR ODD HARMONICS
 Z83    CLS     **,2            (**=SINTAB)
        XCA
 Z84    NOP                     (MPY OR FMP  **,1    WITH **=RSA)
 Z85    NOP                     (ADD OR FAD SMSO)
        STO     SMSO
*NOW GO TO COSINE SUMS IF COSISP, OR AVOID IF SISP
 Z86    TRA     **              (**=Z90 FOR COSISP, **=Z100 FOR SISP)
*LOOP FOR FORWARD OR BACKWARD MOTION ON COSINE WAVE
*    THIS PART FOR EVEN HARMONICS   SUMMED IN SMCE
```

```
   Z90   LDQ     **,4          (**=COSTAB)
   Z91   NOP                   (MPY OR FMP  **,1  WITH **=RSS)
   Z92   NOP                   (ADD OR FAD SMCE)
         STO     SMCE
*    THIS PART IS FOR ODD HARMONICS   SUMMED IN SMCO
   Z93   LDQ     **,2          (**=COSTAB)
   Z94   NOP                   (MPY OR FMP  **,1  WITH **=RAS)
   Z95   NOP                   (ADD OR FAD SMCO)
         STO     SMCO
*INCREMENT INDEX FOR EVEN HARMONICS (BY +JE FOR FORWARD
*    TRAVEL, BY -JE FOR REVERSE TRAVEL)
  Z100   TXI     *+1,4,**      (**=JE  FORWARD)    (**=-JE  REVERSE)
*CHECK IF INDEX HAS RUN OFF END (GREATER THAN M FOR
*    FORWARD TRAVEL, LESS THAN ZERO FOR REVERSE)
*    (HOWEVER FOR REVERSE TRAVEL XR4 GOING NEGATIVE MEANS
*     XR4 GETS GREATER THAN M, SO SAME TEST APPLIES)
  Z101   TXH     Z120,4,**     **=M
*INCREMENT INDEX FOR ODD HARMONICS (BY+JO OR -(JO))
*     AND MAKE SAME KIND OF END TEST
  Z102   TXI     *+1,2,**      (**=JO FORWARD)     (**=-JO REVERSE)
  Z103   TXH     Z110,2,**     (**=M)
*INCREMENT DATA INDEX BY 1 AND CHECK FOR END OF DATA
*    LOOPING BACK TO PLACE DETERMINED BY WHETHER COSP OR
*    SISP OR COSISP AND FORWARD OR BACKWARD AND EVEN OR ODD
  Z104   TXI     *+1,1,1
  Z105   TXL     **,1,**       (TXL  **A,1,**B     **B=P)
*                              **A=Z90 FOR COSP
*                                FOR SISP OR COSISP  (INITIAL = Z50)
*                              **A=Z50  EVEN AND ODD HARMONICS FORWARD
*                              **A=Z60  EVEN FORWARD, ODD REVERSE
*                              **A=Z70  EVEN REVERSE, ODD FORWARD
*                              **A=Z80  EVEN AND ODD REVERSE
  Z106   TRA     **            (**=Z107 FOR COSP OR COSISP,
*                                                **=Z115 FOR SISP)
*READJUSTMENTS WHEN ODD HARMONIC INDEX RUNS OFF END
*FORWARD OR BACKWARD
  Z110   ZET     SWC
         TRA     Z113          BACKWARD
         CLA     K1
         STO     SWC
*IF FORWARD SET TO GO BACKWARD ON ODD
  Z111   SXD     TEMP,2
         CLA     2M
         SUB     TEMP
         PDX     0,2
         CLA     MJC
         STD     Z102
*IF COSP GO BACK, IF NOT REMAKE FORK AT Z105
*                              COSP                SISP OR COSISP
  Z112   NOP                   (TRA  Z104   OR     ZET SWE)
         TRA     Z112A
         CLA     KA10          (KA10 = PZE Z60)
         STA     Z105
         TRA     Z104
  Z112A  CLA     KA12          (KA12=PZE Z80)
         STA     Z105
         TRA     Z104
*IF BACKWARDS SET TO GO FORWARDS ON ODD
  Z113   STZ     SWC
         PXA     0,2
         PAC     0,2
         CLA     JC
         STD     Z102
*IF COSP GO BACK, IF NOT REMAKE FORK AT Z105
*                              COSP                SISP OR COSISP
  Z114   NOP                   (TRA  Z104      OR  ZET  SWE)
         TRA     Z114A
         CLA     KA9           (KA9=PZE  Z50)
         STA     Z105
         TRA     Z104
  Z114A  CLA     KA11          (KA11=PZE Z70)
         STA     Z105
         TRA     Z104
*READJUSTMENT WHEN EVEN HARMONIC INDEX RUNS OFF END
*WHICH WAY WERE WE GOING
```

```
 Z120  ZET     SWE
       TRA     Z122          BACKWARDS
*IF FORWARD, REVERSE SWE, READJUST IR4 AND DECREM OF TXI
 Z121  CLA     K1
       STO     SWE
       SXD     TEMP,4        RESET I*JE TO 2M-I*JE
       CLA     2M
       SUB     TEMP
       PDX     0,4
       CLA     MJE
       STD     Z100
*IS COSP GO BACK, IF NOT REMAKE FORK AT Z105
 Z121A NOP                   (TRA Z102(COSP) ZET SWO (SISP,COSISP))
       TRA     Z121B
       CLA     KA11          (KA11=Z70)
       STA     Z105
       TRA     Z102
 Z121B CLA     KA12          (KA12=Z80)
       STA     Z105
       TRA     Z102
*  IF BACKWARDS
 Z122  STZ     SWE
       PXA     0,4
       PAC     0,4
       CLA     JE
       STD     Z100
*IF COSP GO BACK, IF NOT REMAKE FORK AT Z105
 Z122A NOP                   (TRA Z102 (COSP),ZET SWO (SISP,COSISP))
       TRA     Z122B
       CLA     KA9           (KA9=Z50)
       STA     Z105
       TRA     Z102
 Z122B CLA     KA10          (KA10=Z60)
       STA     Z105
       TRA     Z102
*COSP OR COSISP RESULT STORAGE FOR COSINE TRANSFORMS
*WAS LAST EVEN HARMONIC A DUMMY
 Z107  ZET     DUME
       TRA     Z109          YES
*IF NOT STORE SMCE IN COSTR BLOCK
       LXA     ESTOR,4
       CLA     SMCE
 Z108  STO     **,4          (**=COSTR)
*WAS LAST ODD HARMONIC A DUMMY
 Z109  ZET     DUMO
       TRA     Z109B         YES
*IF NOT STORE SMCO IN COSTR BLOCK
       LXA     OSTOR,4
       CLA     SMCO
 Z109A STO     **,4          (**=COSTR)
 Z109B TRA     **            (**=Z115 COSISP, **=Z130 COSP)
*COSISP OR SISP RESULT STORAGE FOR SINE TRANSFORMS
*WAS LAST EVEN HARMONIC A DUMMY
 Z115  ZET     DUME
       TRA     Z117          YES
*IF NOT STORE SMSE IN SINTR BLOCK
       LXA     ESTOR,4
       CLA     SMSE
 Z116  STO     **,4          (**=SINTR)
*WAS LAST ODD HARMONIC A DUMMY
 Z117  ZET     DUMO
       TRA     Z130          YES
*IF NOT STORE SMSO IN SINTR BLOCK
       LXA     OSTOR,4
       CLA     SMSO
 Z118  STO     **,4          (**=SINTR)
*RESET FOR NEXT LOOP STORAGE
  Z130 CLA     ESTOR
       ADD     K2
       STO     ESTOR
       CLA     OSTOR
       ADD     K2
       STO     OSTOR
*INDEX JE BY TWO AND CHECK IF TOO BIG
       CLA     JE
```

```
        ADD     KD2
        STD     JE
        CAS     T1C          COMPARE WITH JMAX
        TRA     Z135         TOO BIG
        NOP                  OK
*IF NEW JE OK, INDEX JO BY TWO AND CHECK ITS SIZE
  Z131 CLA      JO
        ADD     KD2
        STD     JC
        CAS     T10
        TRA     Z133         TOO BIG
        NCP                  OK
*RETURN TO BEGINNING OF LOOP
  Z132 TRA      Z30
*IF JO TOO BIG SET SWITCH
 Z133  CLA      K1
        STO     DUMC
*IS JE ALSO TOO BIG
        ZET     DUME
        TRA     LV           YES - ALL FINISHED
        TRA     Z132         NO - ONE MORE TO GO
*IF JE TOO BIG SET SWITCH
 Z135  CLA      K1
        STO     DUME
        TRA     Z131         GO CHECK JO
*FINAL EXIT
  LV   LXD      COSP-2,4
        AXT     **,1         (**=IR1)
        AXT     **,2         (**=IR2)
 EXIT  TRA      **,4         (**=10 FOR COSP OR SISP, **=14.FOR COSISP)
*CONSTANTS, TEMPORARIES, ETC
  SWE  PZE      **           (**=0 WHILE EVEN HARMONIC GOING FORWARDS)
*                            (**=1 WHILE EVEN HARMONIC GOING BACKWARD)
  SWO  PZE      **           (**=0 WHILE ODD HARMONIC FORWARDS)
*                            (**=1 WHILE ODD HARMONIC BACKWARDS)
  JE   PZE      0,0,**       **=JE
 MJE   PZE      0,0,**       **=25 COMP OF JE
  JO   PZE      0,0,**       **=JO
 MJO   PZE      0,0,**       **=25 COMP OF JO
  DUME PZE      **           (**=0 FOR REAL EVEN,**=1 FOR DUMMY EVEN)
  DUMO PZE      **           (**=0 FOR REAL ODD,**=1 FOR DUMMY ODD)
 ESTOR PZE      **      (**=ZERO INDEX OF INITIAL EVEN HARMONIC STORAGE)
 OSTOR PZE      **      (**=ZERO INDEX OF INITIAL ODD HARMONIC STORAGE)
  MPY  MPY      **,1
  FMP  FMP      **,1
  ADD  ADD      **
  FAD  FAD      **
  SMSE PZE      **           SUM FOR EVEN HARMONIC SINE TRANSFORM
  SMSO PZE      **           SUM FOR ODD HARMONIC SINE TRANSFORM
  SMCE PZE      **           SUM FOR EVEN HARMONIC COSINE TRANSFORM
  SMCO PZE      **           SUM FOR ODD HARMONIC COSINE TRANSFORM
  2M   PZE      0,0,**       (**=2M)
  TEMP PZE      **
  T1   PZE      **           (**=RSS)
  T2   PZE      **           (**=RAS)
  T3   PZE      **           (**=RSA)
  T4   PZE      **           (**=RAA)
  T5   PZE      0,0,**       (**=P)
  T6   PZE      **           (**=COSTAB)
  T7   PZE      **           (**=SINTAB)
  T8   PZE      0,0,**       (**=M)
  T9   PZE      0,0,**       (**=JMIN)
  T10  PZE      0,0,**       (**=JMAX)
  T11  PZE      **           (**=TYPE)
  T12  PZE      **           (**=COSTR)
  T13  PZE      **           (**=SINTR)
  K0   PZE      0
  K1   PZE      1
  K2   PZE      2
  K10  PZE      1C
  K14  PZE      14
  KT1  TRA      Z104
 KT2   TRA      Z102
  KZ1  ZET      SWE
 KZ2   ZET      SWO
```

```
***********************
*    COSP             *
***********************
(PAGE 12)

      KD1   PZE     0,0,1
      KD2   PZE     0,0,2
      KA2   PZE     SMSE
      KA3   PZE     SMSO
      KA4   PZE     SMCE
      KA5   PZE     SMCO
      KA6   PZE     Z90
      KA7   PZE     Z100
      KA8   PZE     Z30
      KA9   PZE     Z50
     KA10   PZE     Z60
      KA11  PZE     Z70
      KA12  PZE     Z80
      KA13  PZE     KA8
      KA14  PZE     KA9
      KA15  PZE     Z107
      KA16  PZE     Z115
      KA17  PZE     Z120
      KA18  PZE     KA6
      KA19  PZE     Z130
            END
```

****
*   COSP   *
****
(PAGE 12)

```
*      COSTBL (SUBROUTINE)             2/15/63   LAST CARD IN DECK IS NO. 0199
*      FAP
*COSTBL
       COUNT   200
       LBL     COSTBL
       ENTRY   COSTBL (N,COSTAB)
       ENTRY   SINTBL (N,SINTAB)
       ENTRY   COSTBX (N,ICOSTB)
       ENTRY   SINTBX (N,ISINTB)
*
*                   ----ABSTRACT----
*
* TITLE - COSTBL WITH SECONDARY ENTRY POINTS SINTBL, COSTBX, SINTBX
*       GENERATE COSINE OR SINE HALF-WAVE TABLES, FIXED OR FLOATING
*
*             COSTBL GENERATES A HALF-WAVE COSINE TABLE FLOATING POINT
*             SINTBL GENERATES A HALF-WAVE SINE TABLE FLOATING POINT
*             COSTBX GENERATES A HALF-WAVE COSINE TABLE FIXED POINT
*             SINTBX GENERATES A HALF-WAVE SINE TABLE FIXED POINT
*                WHERE
*                   THE HALF-WAVE LENGTH IS AN INPUT PARAMETER.
*                   FOR FIXED POINT TABLES THE BINARY POINT IS BETWEEN
*                       THE SIGN BIT AND BIT 1.
*
* LANGUAGE  - FAP SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
* STORAGE   - 128 REGISTERS
* SPEED     - ABOUT 2N MILLISEC ON 709, WHERE N = HALF-WAVE LENGTH
* AUTHOR    - JON CLAERBOUT
*
*                   ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - (NONE)
*      AND FORTRAN SYSTEM  ROUTINES - COS,SIN
*
* FORTRAN USAGE OF COSTBL
*     CALL COSTBL(N,COSTAB)
*
* INPUTS TO COSTBL
*    N         DEFINES THE HALF-WAVE LENGTH TO BE N+1
*              MUST EXCEED ZERO (PROGRAM EXITS IF N IS NEGATIVE OR ZERO)
*
* OUTPUTS FROM COSTBL
*    COSTAB(I) I=1...N+1  CONTAINS TABLE(J) = COS(J*PI/N)  J=0,1,...,N
*                   I.E. COSTAB(I) CONTAINS TABLE(I-1)
*
* FORTRAN USAGE OF SINTBL
*     CALL SINTBL(N,SINTAB)
* INPUTS TO SINTBL
*    N         SAME MEANING AS FOR COSTBL
* OUTPUTS FROM SINTBL
*    SINTAB(I) I=1...N+1 CONTAINS  TABLE(J) = SIN(J*PI/N) FOR J=0,1...N
*
* FORTRAN USAGE OF COSTBX
*     CALL COSTBX(N,ICOSTB)
* INPUTS TO COSTBX
*    N         SAME MEANING AS FOR COSTBL
* OUTPUTS FROM COSTBX
*    ICOSTB(I) I=1...N+1 IS SAME AS FOR COSTBL BUT DATA IS FIXED POINT
*
* FORTRAN USAGE OF SINTBX
*     CALL SINTBX(N,ISINTB)
* INPUTS TO SINTBX
*    N         SAME MEANING AS FOR COSTBL
* OUTPUTS FROM SINTBX
*    ISINTB(I) I=1...N+1 IS SAME AS FOR SINTBL BUT DATA IS FIXED POINT
*
* EXAMPLES
* 1. GENERAL BEHAVIOR FOR N=4
*    INPUTS  - N=4
*    USAGE   -        CALL COSTBL(N,COSTAB)
*                     CALL SINTBL(N,SINTAB)
*                     CALL COSTBX(N,ICOSTB)
*                     CALL SINTBX(N,ISINTB)
*    OUTPUTS - NOTE - THESE NUMBERS ARE GOOD TO 8 OCTAL PLACES.
```

```
*               COSTAB(1...5) = 1.0,.70711,0.0,-.70711,-1.0
*               SINTAB(1...5) = 0.0,.70711,1.0,.70711,0.0
*               ICCSTB(1...5) = OCT 377777777777,265011714000,
*                                   000000000000,665011714000,777777777777
*               ISINTB(1...5) = OCT 000000000000,265011714000,
*                                   377777777777,265011714000,000000000000
*
        HTR     0
        BCI     1,CCSTBL
COSTBL  CLA     *
        STO     FL
        TRA     *+3
COSTBX  STZ     FL
        STZ     CCRS
        SXD     COSTBL-2,4
        SXA     SV,1
        CLA     KCCS            (TSX  $COS,4)
        STO     AL
        CLA     2,4             GET COSINS
        STA     B3
        ADD     =1              COSINS+1
        STA     A
        STA     B
        STA     B1
        STA     B2
        STA     B4
        TRA     D
SINTBL  CLA     *
        STO     FL
        TRA     *+4
SINTBX  STZ     FL
        CLA     *
        STO     CCRS
        SXD     CCSTBL-2,4
        SXA     SV,1
        CLA     KSIN            (TSX  $SIN,4)
        STO     AL
* SET UP FIXING LCOP
        CLA     2,4             GET SINS
        ADD     =1              SINS+1
        STA     A
        STA     B
        STA     B1
        STA     B2
        STA     L2
* SET UP COMPUTATION LOOP
 D      CLA*    1,4             GET N
        TZE     SV
        TMI     SV
        STD     N
        ADD     KD1             FORM N+1
        STD     AN
        STD     BN
        CLA     N               FLOAT N
        ARS     18
        ORA     ORF
        FAD     ORF
        STO     NFL
        CLA     =3.14159265         FORM PI/N
        FDP     NFL
        STQ     INCR
        STZ     ARG
* LOOP
        AXT     1,1             COS                 SIN
        CLA     ARG
 AL     NOP     **              TSX $COS,4          TSX $SIN,4
 A      STO     **,1            **=COSINS+1         **=SINS+1
        CLA     ARG
        FAD     INCR
        STO     ARG
        TXI     *+1,1,1
 AN     TXL     AL,1,**         **=N+1
        ZET     FL              FIX IF ZERO
        TRA     SV              EXIT - NOT ZERO
        AXT     1,1
```

COSTBL (PAGE 3)

```
BC      CLM
B       LDQ     **,1            **=COSINS+1
        LLS     8
        SSP
        SUB     =0200
        STA     RTSH
B1      CLA     **,1            **=COSINS+1
        LRS
        ANA     =0000777777777
        ALS     8
        LLS
RTSH    ARS     **              ** FROM B+4
B2      STO     **,1            **=COSINS+1
        TXI     *+1,1,1
BN      TXL     BC,1,**         **=N+1
        CLA     CORS
        TNZ     L1
        CLA     =0377777777777                      SET FIRST AND
B3      STO     **              **=COSINS           LAST VALUES
        SSM                                         IN TABLE = 1
        LXD     BN,1
B4      STO     **,1            **=COSINS+1
        TRA     SV
L1      CLA     N
        ARS     18
        LBT                     IF = 0, N EVEN - EXIT
        TRA     *+2
        TRA     SV
        CLA     N               N ODD - SET MDPT = 1
        ARS     1               GET (N+1)/2
        ADD     KD1
        STD     MD
        CLA     =0377777777777
        LXD     MD,1
L2      STO     **,1            ** = SINS+1
SV      AXT     **,1
        LXD     COSTBL-2,4
        TRA     3,4
 N      PZE     **              **=N IN DECR
FL      PZE     **              **=0,FXD
INCR    PZE     **              **=PI/N.
ARG     PZE     **              **=I*PI/N,  I=0,1,....,N
ORF     OCT     233000000000
NFL     PZE     **              **=FLOATF(N)
KD1     PZE     0,0,1
KCOS    TSX     $COS,4
KSIN    TSX     $SIN,4
CORS    PZE     **              **=0 IF COS
MD      PZE     0,0,**          **=(N+1)/2
        END
```

```
*       FACTOR (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0480
*       FAP
*FACTOR
        COUNT   450
        LBL     FACTOR
        ENTRY   FACTOR (SPECT,N,L,WAVE,B1,B2,C,TRAN,WORK,COST)
*
*                    ----ABSTRACT----
*
* TITLE - FACTOR
*        FACTOR POWER SPECTRUM TO FIND MINIMUM PHASE WAVELET
*
*              FACTOR USES THE METHOD OF KOLMOGOROV (REF.- 1. ROBINSON,E.
*              A., M.I.T. PH.D. THESIS,GEOPHYSICAL ANALYSIS GROUP REPORT
*              7,1954. 2. SIMPSON ET AL., SCIENTIFIC REPORT NO. 2 OF
*              CONTRACT AF 19(604)7378.) TO FACTOR THE POWER SPECTRUM
*              AND THUS PRODUCE THE MINIMUM PHASE WAVELET.
*              THE RESTRICTIONS ON APPLICABILITY OF THE METHOD REQUIRE
*              THAT THE INPUT SPECTRUM BE NON-NEGATIVE AND NON-ZERO.
*              HENCE SPECT(I), THE INPUT SPECTRUM, IS CHECKED AND ANY
*              VALUES WHICH ARE LESS THAN 10**(-6) OF THE MAXIMUM VALUE
*              OF SPECT(I) ARE SET EQUAL TO 10**(-6) OF THE MAXIMUM.(THIS
*              FEATURE MAY EASILY BE REMOVED FROM THE SYMBOLIC DECK).
*
*              ONE HALF OF THE NATURAL LOG OF THE SPECTRUM IS COMPUTED
*              AND EXPANDED IN A COSINE SERIES. THE COEFFICIENTS OF THE
*              EXPANSIONSION ARE COMPUTED BY TRIGONOMETRIC INTERPOLATION
*              (REF. LANCZOS, APPLIED ANALYSIS) RATHER THAN BY INTEGRA-
*              TION. SUBROUTINE COSP IS USED FOR THE CALCULATION, BUT THE
*              FIRST AND LAST TERMS OF THE SPECTRUM MUST BE WEIGHTED BY
*              1/2 SO THAT THE COSINE PRODUCTS PRODUCED BY COSP WILL BE
*              ORTHOGONAL UNDER SUMMATION. THE COEFFICIENTS OF THE COSINE
*              EXPANSION ARE TRAN(I),I=1,L. THE EXPONENTIAL
*
*                                      L
*                   EXP**(TRAN(1)+ SUM(TRAN(I)*(Z**(I-1))))
*                                     I=2
*
*              MUST BE EXPANDED IN A CONTINUED PRODUCT OF POLYNOMIALS IN
*              Z. THE POLYNOMIALS ARE THEN MULTIPLIED OUT AND GROUPED IN
*               THE FORM
*
*                                        L
*                                  P = SUM (W(I)*(Z**(I-1)))
*                                       I=1
*
*              WHERE L IS THE LENGTH OF THE WAVELET, AND W(I) IS THE
*              DESIRED WAVELET.
*
*    PROGRAM NOTES -
*              THE EXPANSION OF THE EXPONENTIAL AND MULTIPLICATION OF
*              THE RESULTING POLYNOMIALS MAY BE SIMPLIFIED BY THE
*              FOLLOWING CONSIDERATIONS - THE EXPONENTIAL MAY BE
*              REPRESENTED AS A CONTINUED PRODUCT OF POLYNOMIALS
*              WHERE THE ITH POLYNOMIAL IS OF THE FORM
*
*                               L-1
*                     P(I)=(SUM( C(I,J)*(Z**I))+ 1)*EXP**(TRAN(1))
*                               I=1
*              WHERE
*                   C(I,J)= (TRAN(1)/1)*(TRAN(2)/2)*.....*(TRAN(I)/(J/I))
*                                                       FOR J=K*I
*                   C(I,J)= 0 FOR  J NOT =K*I
*               THE C(I,0) TERMS ARE 1 FOR ALL I.
*
*              WE ARE ONLY INTERESTED IN THE FIRST L TERMS OF THE WAVELET
*              SO WE NEED ONLY CONSIDER TERMS IN THE POLYNOMIALS WITH
*              EXPONENTS LESS THAN OR =M,M=L-1. WE CAN THEN COMPUTE THE
*              WAVELET COEFFICIENTS BY PARTIAL CONVOLUTION OF THE
*              POLYNOMIAL COEFFICIENTS. THAT IS,
*
*                 WAVE(I)= C(1,J)*C(2,J)*...C(M,J)
*              WHERE WAVE(I) IS THE WAVELET, M=L-1, AND THE * SYMBOL
*              DENOTES CONVOLUTION.
*                IT WILL BE NOTED THAT IF THE CONVOLUTION IS REPRESENTED
```

```
*              IN STEPS BY
*                       B(M-1)= C(M-1,J)*C(M,J),  B(K)=C(K,J)*B(K+1)
*              BY CAREFUL INSPECTION OF THE FORM OF THE C(I,J) ONE CAN
*              WRITE DOWN THE B(N) BY INSPECTION FOR N=L/2 (ROUNDED DOWN)
*               +1. THIS CUTS DOWN THE TOTAL LABOR BY NEARLY 1/2.
*                   B(N)= 1,0,0,......,0,C(N,N),C(N+1,N+1),.....,C(M,M)
*                FACTOR SETS UP B(N) AND THEN USES AN INTERNAL SUBROUTINE
*              TO SET UP C(N-1,J) FOR J=0,M. THE INTERNAL SUBROUTINE
*              PARCON COMPUTES THE PARTIAL CONVOLUTION WHICH IS B(N-1).
*                THE NEXT C(I,J) IS SET UP BY CCOM AND THE NEXT B(I-1)
*              COMPUTED BY PARCON. THIS IS REPEATED UNTIL ALL THE PARTIAL
*              CONVOLUTIONS HAVE BEEN DONE. THE RESULTING WAVELET IS THEN
*              SCALED BY EXP**(TRAN(1)).
*                THE OUTPUT OF PARCON FOR ONE STAGE IS THE INPUT FOR THE
*              NEXT STAGE SO THAT THE ADDRESSES B1 AND B2 IN THE PARCON
*              ROUTINE ARE REVERSED BETWEEN STAGES.
*
* LANGUAGE  - FAP, SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709,7090 (MAIN FRAME ONLY)
* STORAGE   - 303 DECIMAL REGISTERS
* SPEED     - 2200+94L+16L**2+3L**3+270N+37L*N MACHINE CYCLES
* AUTHOR    - J.N. GALBRAITH    NOV. 1, 1961
*
*                    ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - MAXAB, COSTBL, COSP
*       AND FORTRAN SYSTEM ROUTINES - LOG, EXP
*
* FORTRAN USAGE
*     CALL FACTOR(SPECT,N,L,WAVE,B1,B2,C,TRAN,WORK,COST)
*
* INPUTS
*
*    SPECT(I)  I=1,N  SPECTRUM FROM ZERO TO PI
*
*    N         NUMBER OF POINTS IN SPECTRUM
*              MUST BE GRTHN 0.
*
*    L         LENGTH OF DESIRED WAVELET.
*              MUST BE GRTHN 0, LSTHN= N.
*
*    B1(I)     I=1,L  SPACE FOR PARTIAL CONVOLUTION
*
*    B2(I)     I=1,L  SPACE FOR PARTIAL CONVOLUTION
*
*    C(I)      I=1,L  SPACE FOR COLUMN OF C(I,J) MATRIX
*
*    TRAN(I)   I=1,L  SPACE FOR COSINE TRANSFORM
*
*    WORK(I)   I=1,N  SPACE FOR COMPUTATION OF 1/2*LOG(SPECT).MAY BE THE
*                SAME AS SPECT IF SPECT CAN BE DESTROYED.
*
*    COST(I)   I=1,L  SPACE FOR COSINE TABLE FOR COSINE SERIES EXPAN-
*                SION.
*    NOTE-
*     COST MAY BE THE SAME AS EITHER B1,B2,OR C IF THE LENGTH IS L+1
*     INSTEAD OF L AS NOTED ABOVE.
*     THE OUTPUT WAVELET MAY ALSO BE THE SAME AS B1,B2,OR C. HENCE
*     THE MINIMUM STORAGE FOR DATA USING ALL POSSIBLE EQUIVALENCES IS
*                N+4*L+1 , AND FACTOR COULD BE CALLED BY
*     CALL FACTOR(SPECT,N,L,B1,B1,B2,C,TRAN,SPECT,B1)
*     WHERE B1 IS OF LENGTH L+1 SINCE IT MUST DO DOUBLE DUTY FOR COST.
*     NO CHECKS ARE MADE ON THE VALUES OF N AND L. BOTH MUST BE GREATER
*     THAN 0, AND L MUST BE LESS THAN OR =N. ILLEGAL VALUES MAY RESULT
*     IN INCORRECT WAVELETS OR PROGRAM LOOPS.
*
* OUTPUTS
*
*    WAVE(I)   I=1,L OUTPUT MINIMUM PHASE WAVELET
*
*    SEE NOTE ABOVE FOR EQUIVALENCE ALLOWANCES.
*    IF THE COSINE TABLE CAN BE USED LATER BY THE CALLING PROGRAM,
*    FACTOR CAN BE CALLED WITH SEPARATE SPACE FOR COST, AND THE TABLE
*    WILL BE RETURNED ALSO.
*
```

FACTOR (PAGE 3)

```
* EXAMPLES
*
* 1. INPUTS  -
*              FOR A CONTINUOUS SPECTRUM
*                  SPECT= 1.25+COS(W), W=0,PI
*              THE WAVELET IS
*                  WAVE= 1.,.5,0.,0.,......,0.
*              FOR THE DISCRETE CASE THE NUMBERS WILL NOT COME OUT
*              EXACTLY THE SAME DUE TO ROUND OFF AND APPROXIMATION.
*              FOR A TEST CASE THE INPUT SPECTRUM CAN BE SET UP WITH A
*              FORTRAN LOOP.   SPECT(I)=1.25 +COSF(FLOATF(I-1)*W) ,I=1,N
*                                    W =PI/FLOATF(N-1)
*              WHERE N IS THE LENGTH OF THE SPECTRUM.
*              RESULTS ARE GIVEN BELOW FOR N=500
*
*    OUTPUTS - WAVE(1...6)= 1.0,0.4999,-0.00025,0.0004,-0.00001,0.000003
*
*              THE HIGHER TERMS ARE EVEN SMALLER WITH WAVE(20) LESS THAN
*               10**(-8)
*
       PZE
       BCI    1,FACTOR
FACTOR SXA    RETURN,1                SAVE IR1
       SXA    RETURN+1,2              SAVE IR2
       SXA    RETURN+2,4              SAVE IR4
       SXD    FACTOR-2,4
       CLA    5,4                      GET LOCATION OF B1
       STA    PAR+1
       STA    BFST
       STA    LOOP2
       STA    LOOP3+1
       CLA    6,4                     GET LOCATION OF B2
       STA    PAR+2
       CLA    1,4                     GET LOCATION OF SPECTRUM
       STA    MAX+2
       ADD    ONE
       STA    LOOP1
       CLA    2,4                     GET LOCATION OF N
       STA    MAX+1
       CLA    9,4                     GET WORK SPACE FOR SPECTRUM
       STA    WGT+3
       STA    WGT+5
       STA    CSP+1
       STA    CSP+2
       ADD    ONE
       STA    END1-2
       STA    WGT
       STA    WGT+2
 MAX   TSX    $MAXAB,4                FIND MAXIMUM OF SPECTUM
       PZE    **                      LOCATION OF N
       PZE    **                      LOCATION OF SPECTUM
       PZE    BIGSP
       PZE    INDEX
       LDQ    BIGSP                   MAX. OF SPECTUM
       FMP    DEC                     10**(-6) OF MAX
       STO    BIGSP
       LXA    RETURN+2,4              RESET IR4
       CLA    1,4
       CLA*   2,4                     GET N (IN DECREMENT)
       STD    END1
       STO    N
       LRS    18           N IN ADDRESS
       ORA    CONST
       FAD    CONST
       STO    NF           FLOATING N
       AXT    1,1
 LOOP1 CLA    **,1         **=SPECT+1
       CAS    BIGSP
       TRA    *+3                     SPECT LARGER
       TRA    *+2                     SPECT EQUAL
       CLA    BIGSP                   SPECT LESS
       TSX    $LOG,4                  LOG(SPECT)
       FDP    NF           1/2 LOG(SPECT)(WEIGHTED)
       STQ    **,1         **=WORK+1
       TXI    *+1,1,1
```

```
************************
*    FACTOR            *
************************
(PAGE  4)
```


```
      END1  TXL     LOOP1,1,**      **=N
            TXI     *+1,1,-1
      WGT   CLA     **,1            **=WORK+1. WEIGHT LAST
            FDP     TWOD            TERM IN SPECTRUM BY 1/2
            STQ     **,1            **=WORK+1
            CLA     **              **=WORK. WEIGHT FIRST
            FDP     TWOD            TERM IN SPECTRUM BY 1/2
            STQ     **              **=WORK
            LXA     RETURN+2,4
            CLA*    3,4             GET L
            STO     L
            SUB     DONE            L-1
            STO     LL
            CLA     10,4
            STA     CST+2
            STA     CSP+4
            CLA     N
            SUB     DONE            N-1
            STO     NN
            CLA     8,4             LOCATION OF TRAN
            STA     CSP+9
      CST   TSX     $COSTBL,4       GO TO COSINE TABLE
            PZE     NN
            PZE     **              COST
      CSP   TSX     $COSP,4         GO TO COSINE TRANSFORM
            PZE     **              WORK SPACE FOR SPECTRUM
            PZE     **              WORK SPACE FOR SPECTRUM
            PZE     NN              N-1
            PZE     **              COST
            PZE     NN              N-1
            PZE     ZERO            JMIN=0
            PZE     LL              JMAX=L-1
            PZE     ONED            1.0
            PZE     **              TRAN(COSTR)
            LXA     RETURN+2,4
            CLA     L
            ARS     1                         L/2
            ANA     MASK
            ADD     DONE                      L/2+1
            STO     M                         M=L/2+1
            CLA     ONED                      1.0
      BFST  STO     **              **=B1.    B1(0)=1.0
            AXT     1,1
            CLA     M                         M
            SUB     DONE                      M-1
            STD     END2
      LOOP2 STZ     **,1                      CLEAR B1
            TXI     *+1,1,1
      END2  TXL     *-2,1,**        **=M-1
            CLA     8,4                       GET LOC. OF TRAN.
            STA     LOOP3
            STA     COM+2
            CLA     L
            STD     END3
            LXD     M,1                       IR1=M
      LOOP3 CLA     **,1                      TRAN
            STO     **,1                      B1
            TXI     *+1,1,1
      END3  TXL     LOOP3,1,**      L IN DECREMENT
            AXT     1,2
            CLA     M
            STO     P
            SUB     DONE
            STD     END23
            AXT     1,1
            CLA     7,4                       GET LOCATION OF C
            STA     PAR+3
            STA     COM+1
      CONV  CLA     P
            SUB     DONE
            STO     P
            SXD     K,2
      COM   TSX     CCOM,4
            PZE     **                        C
            PZE     **              TRAN
```

FACTOR (PAGE 5)

```
PAR    TSX    PARCON,4
       PZE    **                         LOCATION OF B1
       PZE    **                         LOCATION OF B2
       PZE    **                         LOCATION OF C
       CLA    PAR+1                      EXCHANGE
       LDQ    PAR+2                      LOCATIONS
       STO    PAR+2                      OF B1
       STQ    PAR+1                      AND B2
       TXI    *+1,2,1
       TXI    *+1,1,1
END23  TXL    CONV,1,**      **=M-1
       LXA    RETURN+2,4                 RESET IR4
       CLA    M                          GET M
       ARS    18                         M IN ADDRESS
       LBT                               LOW BIT TEST
       TRA    *+4                        M EVEN, B2 CONTAINS WAVELET
       CLA    5,4                        M ODD, B1 CONTAINS WAVELET
       STA    LOOP4
       TRA    *+3
       CLA    6,4
       STA    LOOP4
       CLA    4,4            GET ADDRESS OF A (STORAGE FOR WAVELET)
       STA    LOOP4+2
       LDQ*   8,4            TRAN(1)
       FMP    =.5
       TSX    $EXP,4
       STO    NORM           SCALE FOR WAVELET
       CLA    LL
       STD    END4
       AXT    0,1
LOOP4  LDQ    **,1                       B2 OR B1
       FMP    NORM           SCALE FOR WAVELET
       STO    **,1                       WAVELET
       TXI    *+1,1,1
END4   TXL    LOOP4,1,**     **=L-1
RETURN AXT    **,1                       RESTORE IR1
       AXT    **,2                       RESTORE IR2
       AXT    **,4                       RESTORE IR4
       TRA    11,4
L      PZE    0
LL     PZE    0              L-1
K      PZE    0
N      PZE    0
NN     PZE    0              N-1
M      PZE    0
P      PZE    0
NF     PZE    0
NORM   PZE    0
BIGSP  PZE    0
INDEX  PZE    0
CONST  OCT    +233000000000
MASK   OCT    777777000000
ZERO   PZE    0
ONE    PZE    1,0,0
DONE   PZE    0,0,1
ONED   DEC    1.0
TWOD   DEC    2.0
DEC    DEC    .000001
*CCOM -COMPUTES C(P,J) FOR J=0 TO L-1
*CALLING SEQUENCE
*      TSX    CCOM,4
*      PZE    LOCATION OF C(P,0)
*      PZE    LOCATION OF TRAN
*      RETURN
CCOM   SXA    BACK,1         SAVE IR1
       SXA    BACK+1,2       SAVE IR2
       SXA    BACK+2,4       SAVE IR4
       CLA    L              GET L
       STD    ADDR2+2
       CLA    P              GET P
       ARS    18             L IN ADDRESS
       CHS
       ADD    1,4            ADDRESS OF C(P,P)
       STA    ADDR3
       STA    ADDR4
       CLA    1,4            LOCATION OF C(0)
```

```
        STA    ADDR1
        ADD    ONE
        STA    ADDR2
        CLS    P
        ARS    18
        ADD    2,4          TRAN
        STA    STO1
        CLA    ONED         1.0
ADDR1   STO    **           C(0)
        AXT    2,1          CLEAR
ADDR2   STZ    **,1         C(1) TO
        TXI    *+1,1,1      C(L)
        TXL    ADDR2,1,**   **=L
STO1    CLA    **           TRAN(P)
ADDR3   STO    **           C(P,P)
        STO    TEMP1
        STO    TEMP2
        CLA    LL
        LRS    35           INTO MQ
        DVP    P            (L-1)/P
        LLS    53           INTO AC
        SUB    DONE         (L-1)/P-1
        TZE    BACK         IF ZERO,NO MORE TO DO
        STD    END          NOT ZERO, SET TO DO (L-1)/P-1 TIMES
        CLA    P
        PDX    ,2           P IN IR2
        SXD    END-2,2
        AXT    1,1
        CLA    TWOD         GET 2.0
        STO    R            INITIALIZE R
LOOP    LDQ    TEMP1
        FMP    TEMP2        TRAN(1)
        FDP    R
ADDR4   STQ    **,2         **=C. C(R+1) COMPUTED.
        STO    TEMP1        SAVE FOR NEXT C
        CLA    R            GET R
        FAD    ONED         INCREMENT BY 1.0
        STO    R            RE-SET R
        TXI    *+1,2,**     **=P. INCREMENT C STORAGE INDEX
        TXI    *+1,1,1      INCREMENT LOOP COUNTER
END     TXL    LOOP,1,**    **=L-1/P-1. END LOOP CHECK.
BACK    AXT    **,1         RESTORE IR1
        AXT    **,2         RESTORE IR2
        AXT    **,4         RESTORE IR4
        TRA    3,4          RETURN
TEMP1   PZE    0,0,0        WILL CONTAIN PARTIAL SUM FOR C(P)
TEMP2   PZE    0,0,0        WILL CONTAIN TRAN(P)
R       PZE
*PARCON COMPUTES A PARTIAL CONVOLUTION OF C AND B1
*CALLING SEQUENCE
*       TSX    PARCON,4
*       PZE    LOCATION OF B1
*       PZE    LOCATION OF B2
*       PZE    LOCATION OF C(X,0)
PARCON  SXA    EXT,1        SAVE IR1
        SXA    EXT+1,2      SAVE IR2
        SXA    EXT+2,4      SAVE IR4
        CLA    2,4          GET LOCATION OF B2
        STA    REG1
        STA    REG3
        STA    REG3+1
        ADD    ONE
        STA    REG2
        CLA    3,4          LOCATION OF C
        STA    REG5
        CLA    ONED         1.0
REG1    STO    **           B2(0)=1.0
        AXT    2,1
        CLA    L            GET L
        STD    REG2+2
        SUB    DONE
        STD    REG8
REG2    STZ    **,1         CLEAR B2(1) TO B2(L)
        TXI    *+1,1,1
        TXL    REG2,1,**    DECREMENT=L
```

****
*   FACTOR   *
****
(PAGE 7)

```
         CLA    M
         SUB    K             K GOES FROM 1 TO M-1. SET BY CALLING LOOP.
         PDX    ,1            IR1=M-K
         SXD    REG3+2,1
         PDC    ,2
         SXD    REG3+3,2
         SXD    S,1           S=IR1=M-K
REG7     AXT    0,2           ZERO IR2
         LXA    EXT+2,4       RESET IR4
         CLA    S             GET S
         STD    REG6
         CLS    S
         ARS    18
         ADD    1,4           LOCATION OF B1(S)
         STA    REG4
         AXT    0,4
REG5     LDQ    **,4          C(0)
REG4     FMP    **,2          B1(S)
REG3     FAD    **,1          B2
         STO    **,1          B2
         TXI    *+1,4,**      (M-K) IN DECREMENT
         TXI    *+1,2,**      -(M-K) IN DECREMENT
REG6     TXL    REG5,4,**     **=S
         TXI    *+1,1,1
REG8     TXL    REG7-1,1,**   **=L-1
EXT      AXT    **,1          RESTORE IR1
         AXT    **,2          RESTORE IR2
         AXT    **,4          RESTORE IR4
         TRA    4,4           RETURN
S        PZE    0
         END
```

```
*     FRQCT1 (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0094
*     LABEL
CFRQCT1
      SUBROUTINE FRQCT1(IX,NX,IXLO,IXHI,ICT,IANS)
C
C                    ----ABSTRACT----
C
C TITLE - FRQCT1
C       FREQUENCY DISTRIBUTION OF A FIXED POINT VECTOR
C
C             FRQCT1 MAKES A FREQUENCY COUNT OF AN INTEGER SEQUENCE WITH
C             VALUES IN A SPECIFIED RANGE.  FOR EACH INTEGER VALUE IN
C             THE INCLUSIVE RANGE IXLO TO IXHI, THE NUMBER OF
C             OCCURRENCES OF THIS VALUE IN THE INTEGER SEQUENCE IS
C             COUNTED.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 709C (MAIN FRAME ONLY)
C STORAGE   - 117 REGISTERS
C SPEED     -
C AUTHOR    - S. M. SIMPSON
C
C                    ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C      CALL FRQCT1(IX,NX,IXLO,IXHI,ICT,IANS)
C
C INPUTS
C
C    IX(I)      I=1...NX IS THE GIVEN INTEGER SEQUENCE
C               IXLO LSTHN OR = IX(I) LSTHN OR = IXHI.
C
C    NX         IS THE NUMBER OF IX VALUES IN THE SEQUENCE.
C               MUST BE GRTHN 0.
C
C    IXLO       IS AN INTEGER
C               LSTHN OR = ALL IX(I)
C               IXLO MAY BE NEG.
C
C    IXHI       IS AN INTEGER
C               GRTHN OR = ALL IX(I)
C               IXHI MAY BE NEG.
C
C OUTPUTS
C
C    ICT(I)     I=1...NCT IS THE FREQUENCY COUNT WHERE
C                    ICT(1) = NUMBER OF MEMBERS OF THE INPUT SEQ = IXLO
C                    ICT(2) = NUMBER OF MEMBERS OF THE INPUT SEQ = IXLO+1
C                                   ETC.
C                    ICT(NCT) = NUMBER OF MEMBERS OF THE INPUT SEQ = IXHI
C                    WHERE NCT = IXHI-IXLO+1
C
C    IANS       = 0  NORMAL
C               = 1  ILLEGAL NX
C               = 2  ILLEGAL IXLO
C
C EXAMPLES OF FRQCT1
C
C 1. INPUTS  - IXLO=3     IXHI=10     NX=3     IX(1...3)=4,4,4
C    OUTPUTS - ICT(1...8) = 0,3,0,0,0,0,0,0   IANS=0
C
C 2. INPUTS  - IXLO=5     IXHI=12     NX=7     IX(1...7)=5,6,7,8,9,10,11
C    CUTPUTS - ICT(1...8) = 1,1,1,1,1,1,1,0   IANS=0
C
C 3. INPUTS  - IXLO=5     IXHI=12     NX=0
C    CUTPUTS - ERROR     IANS=1
C
C 4. INPUTS  - IXLO=13     IXHI=12     NX=7
C    CUTPUTS - ERROR     IANS=2
C
      DIMENSION  IX(2),ICT(2)
C SET UP AND CLEAR ICT(I).
```

```
      IANS=0
      NCT=IXHI-IXLO+1
      NSHIFT=IXLO-1
      IF (NX)  9991,9991,10
   10 IF (NCT)  9992,9992,15
   15 DO 20  I=1,NCT
   20 ICT(I)=0
C SCAN IX(I) TO MAKE COUNTS (PUT EACH IX IN RANGE 1 TO NCT FIRST).
      DO 35  I=1,NX
      IXI=IX(I)-NSHIFT
      IF (IXI)  9992,9992,30
   30 IF (IXI-NCT)  35,35,9992
   35 ICT(IXI)=ICT(IXI)+1
      GO TO 9999
 9999 RETURN
 9991 IANS=1
      GO TO 9999
 9992 IANS=2
      GO TO 9999
      END
```

```
*      FRQCT2 (SUBROUTINE)            2/18/63   LAST CARD IN DECK IS NO. 0211
*      FAP
*FRQCT2
       COUNT   200
       LBL     FRQCT2
       ENTRY   FRQCT2 (X,LX,B,LB,ICOUNT,IANS)
*
*                   ----ABSTRACT----
*
* TITLE - FRQCT2
*       FREQUENCY COUNT OF NUMBER OF VALUES OF A SERIES IN GIVEN RANGES.
*
*             FRQCT2 MAKES A FREQUENCY COUNT OF A FLOATING POINT,
*             FORTRAN INTEGER, OR MACHINE LANGUAGE INTERGER SERIES FOR
*             THE NUMBER OF VALUES LYING IN SPECIFIED RANGES. IT IS
*             USEFUL IN COMPUTING EMPIRICAL PROBABILITY DENSITIES.
*
*             THERE ARE LB RANGE LIMITS, B(I), I=1, LB, AND HENCE LB+1
*             RANGES. A NUMBER, X(J), IS SAID TO BE IN THE I-TH RANGE
*             IF B(I-1) LSTHN OR EQUAL X(J) LSTHN B(I). A NUMBER IS IN
*             THE FIRST RANGE IF IT IS LSTHN B(1), AND IN THE LB+1
*             RANGE IF GRTHN OR EQUAL B(LB). THE INPUT SERIES X(I) MUST
*             BE THE SAME MODE (FLOATING, INTEGER, ETC.) AS THE RANGE
*             LIMITS BECAUSE THE METHOD USES CAS INSTRUCTIONS.
*
* LANGUAGE  - FAP SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
* STORAGE   - 117 REGISTERS
* SPEED     -
* AUTHOR    - J. N. GALBRAITH
*
*                   ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - NONE
*       AND FORTRAN SYSTEM ROUTINES - NONE
*
* FORTRAN USAGE
*     CALL FRQCT2(X,LX,B,LB,ICOUNT,IANS)
*
* INPUTS
*
*    X(I)       I=1...LX IS THE GIVEN SERIES.
*               MAY BE FLOATING, FORTRAN INTEGER, OR MACHINE INTEGER.
*
*    LX         IS THE LENGTH OF THE X SERIES.
*               MUST BE GRTHN 0.
*
*    B(I)       I=1...LB IS VECTOR OF RANGE LIMITS. B(I) LSTHN B(I+1).
*               RANGES INTO WHICH THE SERIES IS DIVIDED ARE (-INFINITY,
*               LSTHN B(1)),(GRTHN OR =B(1),LSTHN B(2)) ETC.
*               MAY BE FLOATING, FORTRAN INTEGER, OR MACHINE INTEGER,
*                 BUT MUST BE THE SAME AS X(I)
*
*    LB         NUMBER OF RANGE LIMITS.
*               MUST BE GRTHN 0.
*               NOTE - NUMBER OF RANGES =1+ NUMBER OF RANGE LIMITS.
*
* OUTPUTS
*
*    ICOUNT(I)  I=1...LB+1=NUMBER OF X VALUES IN EACH RANGE OF B.
*               ICOUNT(1)=NO. X LSTHN B(1). ICOUNT(2)=NO. X LSTHN B(2),
*                 GRTHN OR =B(1).
*               ICOUNT(LB)=NO. X LSTHN B(LB),GRTHN OR=B(LB-1).
*               ICOUNT(LB+1)=NO. X GRTHN OR =B(LB).
*
*    IANS       IANS=0, NORMAL
*               IANS=1, ILLEGAL LX
*               IANS=2, ILLEGAL LB
*               IANS=3, WEIRD ERROR
*
* EXAMPLES
*
* 1. INPUTS   - X(1...15) = -21.,-20.,-15.,-14.,-12.,-11.,-8.,-7.,0.,1.,
*               2.1,3.,4.,5.,6.   LX=15  B(1...5)= -20.,-16.,-7.5,0.,.9
*               LB=5
```

```
*     OUTPUTS - ICCUNT(1...6) = 1,1,5,1,1,6,    IANS=0
*
*     INPUTS  - SAME AS EXAMPLE 1. EXCEPT B(1...5)=-21.,-11.5,0.,4.5,6.
*     OUTPUTS - ICCUNT(1...6) =0,5,3,5,1,1   IANS=0
*
* 3. INPUTS  - SAME AS EXAMPLE 1. EXCEPT B(1...5)=-21.,-11.5,0,4.5,6.1
*     OUTPUTS - ICCUNT(1...6) =0,5,3,5,2,0   IANS=0
*
* 4. INPUTS  - SAME AS EXAMPLE 1. EXCEPT B(1)=0. B(2)=.5  LB=2
*     OUTPUTS - ICCUNT(1...3) =8,1,6   IANS=0
*
* 5. INPUTS  - SAME AS EXAMPLE 4. EXCEPT LB=0
*     OUTPUTS - ERROR IANS =2
*
* 6. INPUTS  - SAME AS EXAMPLE 4. EXCEPT  LX=0  LB=2
*     OUTPUTS - ERROR IANS = 1
*
* SAVE IRS AND CHECK FOR ILLEGAL PARAMETERS
        PZE     0
        BCI     1,FRQCT2
FRQCT2  SXA     RETURN,1
        SXA     RETURN+1,2
        SXA     RETURN+2,4
        SXD     FRQCT2-2,4
        STZ*    6,4             IANS=0
        CLA*    2,4             GET LX
        TZE     ERR1
        TMI     ERR1
        STO     END
        CLA*    4,4             GET LB
        TZE     ERR2
        TMI     ERR2
        ARS     18              LB IN ADDRESS
        STO     LB
        ARS     1               LB/2 (IN ADDRESS)
        STO     LBHALF
        CLA     1,4             ADDRESS OF X
        ADD     K1MLI           A(X+1)
        STA     XADD
        STA     TESTLO
        CLA     3,4             ADDRESS OF B
        ADD     K1MLI           A(B+1)
        STA     BTEST1
        STA     BADD
        SUB     LB
        STA     TESTHI
        CLA     5,4             ADDRESS OF ICOUNT
        ADD     K1MLI           A(ICOUNT+1)
        STA     STZCNT
        STA     EQUAL
        STA     STCCNT
        LXA     LB,1
        TXI     *+1,1,1
        SXD     END1,1
        AXT     1,4
        AXT     1,1
STZCNT  STZ     **,1            ZERO ICOUNT(I),I=1,LB+1
        TXI     *+1,1,1
END1    TXL     STZCNT,1,**     **=LB+1
        AXT     1,1
LOOP    CLA     K1MLI
        STO     LBLO            INITIAL LBLO=1
        CLA     LB
        STO     LBHI            INITIAL LBHI=LB
        CLA     LBHALF
        STO     LBCOM           INITIAL LBCOM=LB/2
        AXT     1,2
TESTLO  CLA     **,1            GET X.  (**=A(X+1))
BTEST1  CAS     **,4            B(1)  SEE IF IN LOWEST RANGE
        TRA     TESTHI
        TRA     NEXIND
        TRA     EQUAL
TESTHI  CAS     **              **=A(B(LB)). SEE IF IN HIGHEST RANGE
        TRA     HIEST
        TRA     HIEST
```

```
SEARCH LXA    LBCOM,2
 XADD  CLA    **,1        GET X(IR1)
 BADD  CAS    **,2        COMPARE WITH B(LBCOM)
       TRA    GRATER      X GREATER, NEW LBLO (=LBCOM)
       TRA    NEXIND      GOT IT, INDEX ICOUNT(IR2+1)
 LESS  PXA    0,2         X LESS, NEW LBHI (=LBCOM)
       SUB    LBLO        LBCOM-LBLO=DIF
       CAS    K1MLI
       TRA    *+3         DIF GREATER THAN ONE
       TRA    EQUAL       DIF=1, GOT IT, INDEX ICOUNT(IR2)
       TRA    ERROR          IMPOSSIBLE
       ARS    1           DIF/2
       ADD    LBLO        NEW LBCOM
       LDQ    LBCOM
       STQ    LBHI
       STO    LBCOM
       TRA    SEARCH
GRATER PXA    0,2
       SUB    LBHI        LBCOM-LBHI=-DIF
       SSP                DIF
       CAS    K1MLI
       TRA    *+3
       TRA    NEXIND      GOT IT, INDEX ICOUNT(IR2+1)
       TRA    ERROR       IMPOSSIBLE
       ARS    1
       ADD    LBCOM
       LDQ    LBCOM
       STO    LBCOM
       STQ    LBLO
       TRA    SEARCH
NEXIND TXI    *+1,2,1
 EQUAL CLA    **,2        **=A(ICOUNT+1)
       ADD    K1FX
STCCNT STO    **,2        **=A(ICOUNT+1)
       TXI    *+1,1,1
 END   TXL    LOOP,1,**   **=LX
RETURN AXT    **,1
       AXT    **,2
       AXT    **,4
       TRA    7,4
 HIEST LXA    LB,2
       TRA    NEXIND
 ERR1  CLA    K1FX
       STO*   6,4
       TRA    7,4
 ERR2  CLA    K2FX
       STO*   6,4
       TRA    7,4
 ERROR CLA    K3FX
       STO*   6,4
       TRA    7,4
* CONSTANTS AND TEMPORARIES
 K1FX  PZE    0,0,1
 K2FX  PZE    0,0,2
 K3FX  PZE    0,0,3
 K1MLI PZE    1,0,0
 LB    PZE    0
LBHALF PZE    0
LBLO   PZE    0
LBCOM  PZE    0
LBHI   PZE    0
       END
```

```
*     GETRD1 (SUBROUTINE)              2/15/63   LAST CARD IN DECK IS NO. 0172
*     LABEL
CGETRD1
      SUBROUTINE GETRD1(ITAPE,NX,IX,IANS)
C
C                   ----ABSTRACT----
C
C TITLE - GETRD1
C       ACCESS ROUTINE FOR RAND CORP. MILLION RANDOM DIGITS FROM TAPE
C
C             GETRD1 FURNISHES THE NEXT NX SEQUENTIAL RANDOM DIGITS
C             AS FIXED POINT INTEGERS FROM A SPECIFIED TAPE UNIT.
C
C             THE TAPE UNIT CONTAINS THE MILLION DIGITS IN BCD FORM
C             AS LOADED OFF-LINE FROM THE 20000 CARDS CONTAINING THEM,
C             EACH CARD WITH FORMAT(50I1).  GETRD1 KEEPS A BUFFER OF
C             LENGTH 50 TO PREVENT MISSING ANY DIGITS, BUT DOES NOT
C             CHECK FOR THE POSSIBILITY THAT THE SUPPLY IS EXHAUSTED.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 7090 (MAIN FRAME PLUS 1 TAPE UNIT)
* STORAGE   - 229 REGISTERS
C SPEED     - SLOW, SINCE TAPE IS BCD
C AUTHOR    - S.M.SIMPSON JR.
C
C                   ----USAGE----
C
C TRANSVER VECTOR CONTAINS ROUTINES - (NONE)
C       AND FORTRAN SYSTEM ROUTINES - (TSH), (RTN)
C
C FORTRAN USAGE
C     CALL GETRD1(ITAPE,NX,IX,IANS)
C
C INPUTS
C
C    ITAPE      IS THE LOGICAL TAPE NO. OF THE RANDOM DIGITS TAPE
C               MUST LIE BETWEEN 1 AND 12 INCLUSIVE
C
C    NX         IS THE DESIRED NO. OF DIGITS
C               MUST EXCEED ZERO
C
C OUTPUTS
C
C    IX(I)      I=1...NX  WILL CONTAIN THE NEXT NX DIGITS AS FORTRAN
C                  FIXED POINT INTEGERS
C
C    IANS       =  0  NORMAL
C               = -1  FOR ILLEGAL ITAPE
C               =  2              NX
C
C EXAMPLES
C
C 1. ILLUSTRATING EFFECTS OF SUCCESSIVE CALLS
C    INPUTS  - THE FIRST THREE RAND DIGITS CARDS ARE AS FOLLOWS
C
C             C   CCLUMN NUMBERS
C             A
C             R   00000000011111111112222222222333333333344444444445
C             D   12345678901234567890123456789012345678901234567890
C
C             1   10097325337652013586346735487680959091173929274945
C             2   37542048056489474296248052403720636104020082291665
C             3   08422689531964509303232090256015953347643508033606
C             ASSUME THE CARDS ARE LOADED ON LOGICAL TAPE 9
C
C    USAGE   -        REWIND 9
C                     CALL GETRD1(9,10,IX1,IANS1)
C                     CALL GETRD1(9,10,IX2,IANS2)
C                     CALL GETRD1(9, 1,IX3,IANS3)
C                     CALL GETRD1(9,29,IX4,IANS4)
C                     CALL GETRD1(9, 1,IX5,IANS5)
C                     CALL GETRD1(9,55,IX6,IANS6)
C                     REWIND 9
C                     CALL GETRD1(9, 3,IX7,IANS7)
C
```

```
C    OUTPUTS - IANS1=IANS2 = ETC = IANS7 = 0    (NO ILLEGALITIES)
C              IX1(1...10) = 1,0,0,9,7,3,2,5,3,3
C              IX2(1...10) = 7,6,5,2,0,1,3,5,8,6
C              IX3(1...1)  = 3
C              IX4(1...29) = 4,6,7,3,5,4,8,7,6,8,0,9,5,9,0,9,1,1,7,3,
C                            9,2,9,2,7,4,9,4,5
C              IX5(1...1)  = 3
C              IX6(1...55) = 7,5,4,2,0,4,8,0,5,6,4,8,9,4,7,4,2,9,6,2,
C                            4,8,0,5,2,4,0,3,7,2,0,6,3,6,1,0,4,0,2,0,
C                            0,8,2,2,9,1,6,6,5,0,8,4,2,2,6
C              IX7(1...3)  = 8,9,5  (NOT = 1,0,0  SINCE GETRD1 STILL
C                                    HAS 44 DIGITS IN ITS BUFFER TO
C                                    USE UP BEFORE READING FROM TAPE
C                                    AGAIN)
C
C 2. ILLUSTRATING ILLEGAL USAGE
C
C    USAGE   -        CALL GETRD1(0,1,IX,IANS1)
C                     CALL GETRD1(13,1,IX,IANS2)
C                     CALL GETRD1(9,-3,IX,IANS3)
C
C    OUTPUTS - IANS1 = IANS2 = -1    (ILLEGAL ITAPE)
C              IANS3 = -2            (ILLEGAL NX)
C
C PROGRAM FOLLOWS BELOW
C
C DUMMY DIMENSION STATEMENT
      DIMENSION IX(2)
C TRUE DIMENSION STATEMENT
      DIMENSION INP(50)
C CHECK LEGALITIES OF ITAPE,NX
      IANS=-1
      IF (ITAPE)  9999,9999,2
    2 IF (ITAPE-12)  4,4,9999
    4 IANS=-2
      IF (NX)  9999,9999,10
   10 IOUT=0
      IANS=0
      MORE=NX
C
C ANY DIGITS LEFT IN BUFFER FROM PREVIOUS CALL (IF NO, GO READ
C 50 DIGITS).
C
      IF (NBUF)  20,40,20
C
C IF YES, CHECK IF REQUEST CAN BE FILLED FROM BUFFER.
C
   20 IF (NX-NBUF)  30,30,24
C
C IT CANT.  EMPTY BUFFER AND THEN GO READ MORE DIGITS.
C
   24 DO 26  I=1,NBUF
   26 IX(I)=INP(I)
      IOUT=NBUF
      MORE=MORE-NBUF
      GO TO 40
C
C IT CAN BE FILLED FROM BUFFER.  SET UP TO DO SO AND EXIT.
C
   30 NBLOK=NBUF
      GO TO 66
C
C READ 50 DIGITS
C
   40 READ INPUT TAPE ITAPE,42,(INP(I),I=1,50)
   42 FORMAT(50I1)
C
C CHECK IF THIS IS LAST BLOCK OF 50 NEEDED.
C
      IF (MORE-50)  60,60,50
C
C NO. MOVE BLOCK OF 50 AND GO BACK FOR ANOTHER.
C
   50 DO 54  I=1,50
      II=I+IOUT
```

```
   54 IX(II)=INP(I)
      IOUT=IOUT+50
      MORE=MORE-50
      GO TO 40
C
C YES.  SET FOR FINAL MOVE.
C
   60 NBLOK=50
C
C MOVE FINAL BLOCK AND SET UP BUFFER FOR NEXT CALL
C
   66 DO 68  I=1,MORE
      II=I+IOUT
   68 IX(II)=INP(I)
      NBUF=NBLOK-MORE
      IF (NBUF)  70,9999,70
   70 MRP1=MORE+1
      DO 74  I=MRP1,NBLOK
      II=I-MORE
   74 INP(II)=INP(I)
      GO TO 9999
 9999 RETURN
      END
```

```
*     GRUP2 (SUBROUTINE)               2/18/63   LAST CARD IN DECK IS NO. 0139
*     LABEL
CGRUP2
      SUBROUTINE GRUP2 (P,NDELX,DELX,XLO,YLIM,NWANT,IANS)
C
C                     ----ABSTRACT----
C
C TITLE - GRUP2
C       DIVIDES THE X AXIS INTO EQUALLY PROBABLE RANGES
C
C              GRUP1 PERFORMS A PROCESS KNOWN AS THE PROBABILITY
C              TRANSFORMATION WHEREBY A GIVEN PROBABILITY DENSITY IS
C              TRANSFORMED INTO A RECTANGULAR DENSITY.
C
C              THE PRINCIPAL INPUT IS A HISTOGRAM-TYPE PROBABILITY
C              DISTRIBUTION FUNCTION P(I),I=1...NDELX, WHERE P(I) =
C              PROBABILITY DENSITY FOR THE RANDOM VARIABLE X FALLING IN
C              THE ITH RANGE OF X VALUES, WHERE ALL RANGES ARE OF EQUAL
C              LENGTH DELX, AND THE LOWEST RANGE IS FROM XLO TO XLO+DELX.
C
C              GRUP2 DIVIDES THE X AXIS INTO NWANT RANGES FROM XLO TO
C              NDELX*DELX+XLO, EACH RANGE HAVING EQUAL PROBABILITY DELP.
C              DELP=1./FLOATF(NWANT). GRUP2 RETURNS THE X VALUES
C              CORRESPONDING TO THE RANGES. THE DIVISION IS MADE BY
C              INTEGRATING THE PROBABILITY DISTRIBUTION ALONG THE X AXIS.
C              LINEAR INTERPOLATION IS MADE WHEN AN INTEGER MULTIPLE OF
C              1/NWANT LIES BETWEEN SUM UP TO J AND J+1 OF (P(I)*DELX).
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
C STORAGE   - 198 REGISTERS
C SPEED     -
C AUTHOR    - J.N. GALBRAITH
C
C                     ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C      CALL GRUP2 (P,NDELX,DELX,XLO,YLIM,NWANT,IANS)
C
C INPUTS
C
C    P(I)       I=1...NDELX IS THE PROBABILITY DISTRIBUTION DEFINED
C               FROM XLO TO NDELX*DELX+XLO AND NORMALIZED SUCH THAT
C               THE SUM FROM I=1 TO NDELX OF P(I)*DELX =1.  IF P(I)
C               IS NORMALIZED SUCH THAT SUM (P(I)) LESS THAN 1., AN ERROR
C               MAY OCCUR WITH IANS=-4.  IF P(I) IS NORMALIZED SUCH THAT
C               SUM (P(I)) GRTHN 1., THE YLIM WILL BE COMPUTED IN THE
C               USUAL MANNER WITH NORMALIZATION ASSUMED =1.
C
C    XLO        IS LOWEST VALUE OF X FOR WHICH P(I) IS DEFINED.
C
C    DELX       IS THE INCREMENT IN X.
C               MUST BE GRTHN 0.
C
C    NDELX      IS THE NUMBER OF INCREMENTS.
C               MUST BE GRTHN 1.
C
C    NWANT      IS THE NUMBER OF EQUALLY LIKELY DIVISIONS WANTED.
C               MUST BE GRTHN 1.
C
C OUTPUTS
C
C    YLIM(I)    I=1...NWANT+1 IS THE VECTOR OF X VALUES WHICH
C               CORRESPOND TO EQUALLY LIKELY PROBABILITY DIVISIONS.
C               (YLIM(1)=XLO), (YLIM(NWANT+1)=XLO+FLOATF(NDELX)*DELX).
C
C    IANS       =  0  NORMAL
C               = -1  ILLEGAL NDELX
C               = -2  ILLEGAL DELX
C               = -3  ILLEGAL NWANT
C               = -4  WEIRD ERROR (P PROBABLY NOT PROPERLY NORMALIZED)
C
```

```
C EXAMPLES
C
C 1. INPUTS  - ALL P=0.     NDELX=1     DELX=0.     XLO=0.     NWANT=0
C    OUTPUTS - ERROR  IANS = -1
C
C 2. INPUTS  - SAME AS EXAMPLE 1. EXCEPT NDELX=20
C    OUTPUTS - ERROR  IANS= -2
C
C 3. INPUTS  - SAME AS EXAMPLE 2 EXCEPT DELX=.05  NWANT=1
C    OUTPUTS - ERROR  IANS= -3
C
C 4. INPUTS  - P(1...20) = 1.,.7,.5,1.3,2.,1.9,.6,.5,.4,.3,.2,.1,1.5,
C              1.5,1.5,1.5,1.5,.5,.5,2.     NDELX=20     DELX=.05
C              XLO=0.     NWANT=5
C    OUTPUTS - YLIM(1,...,6) = 0.,.2125,.35,.68333,.81666,1.     IANS=0
C
C 5. INPUTS  - SAME AS EXAMPLE 4. EXCEPT XLO=20.
C    OUTPUTS - YLIM(1,...,6) = 20.,20.2125,20.35,20.68333,20.86666.,21.
C              IANS=0
C
C 6. INPUTS  - SAME AS EXAMPLE 5. EXCEPT DELX=.0005
C    OUTPUT  - ERROR  IANS=-4
C
C 7. INPUTS  - SAME AS EXAMPLE 5. EXCEPT DELX=100.
C    OUTPUTS - YLIM(1,...,6) = 20.,20.2,20.4,20.6,20.8,20.20   IANS=0
C
      DIMENSION P(200),YLIM(201)
C     CHECK NDELX
      IANS=-1
      IF(NDELX-1)  9999,9999,5
C     CHECK DELX
 5    IANS=-2
      IF(DELX)  9999,9999,10
 10   NUM1=NWANT-1
      IANS=-3
      IF(NUM1)  9999,9999,20
 20   YLIM(1)=XLO
      YLIM(NWANT+1)=XLO+FLOATF(NDELX)*DELX
      DELP=1./FLOATF(NWANT)
      PTEST=DELP
      ISTART=1
      SUM=0
      IANS=0
      DO 100  J=1,NUM1
      DO 50  I=ISTART,NDELX
      DELTA=P(I)*DELX
      SUM=SUM+DELTA
      IF(SUM-PTEST)  50,60,70
 50   CONTINUE
C     ERROR- USED ALL P WITHOUT FINDING ALL YLIM.
      GO TO 9777
 60   YLIM(J+1)=FLOATF(I)*DELX+XLO
      ISTART=I+1
      GO TO 90
C     INTERPOLATE
 70   SUM=SUM-DELTA
      FRACTX=(PTEST-SUM)/DELTA
      YLIM(J+1)=(FLOATF(I-1)+FRACTX)*DELX+XLO
      ISTART=I
 90   PTEST=PTEST+DELP
 100  CONTINUE
 9999 RETURN
 9777 IANS=-4
      GO TO 9999
      END
```

* KIINT1 *

```
*     KIINT1 (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0128
*     LABEL
CKIINT1
      SUBROUTINE KIINT1 (CHISQ,NDF,PROB,IANS)
C
C                    ----ABSTRACT----
C
C TITLE - KIINT1
C       PROBABILITY THAT A CHI-SQUARED VARIATE EXCEEDS A VALUE.
C
C             KIINT1 PRODUCES THE PROBABILITY THAT A CHI-SQUARED VARIATE
C             WILL EXCEED A GIVEN VALUE. THIS PROBABILITY IS COMPUTED BY
C             EQUATIONS GIVEN BY YULE AND KENDALL, 1950, THEORY OF
C             STATISTICS, PAGE 464 (FOOTNOTE) FOR NDF LESS THAN 31,
C             WHERE NDF = NO. DEGREES OF FREEDOM.
C             FOR HIGHER NDF THE NORMAL APPROXIMATION IS USED.
C             WHEN THE NORMAL APPROXIMATION IS USED A TABLE OF THE
C             NORMAL DISTRIBUTION WHICH APPEARS IN SUBROUTINE NOINT1 IS
C             USED AND, SINCE THIS TABLE HAS ONLY 201 VALUES
C             CORRESPONDING TO VALUES OF X (UNIT NORMAL) FROM
C             0.0 TO 4.0, PROBABILITIES LESS THAN .00032 ARE SET TO ZERO
C             AND THOSE GREATER THAN 99968 ARE SET EQUAL TO ONE. THIS
C             DOES NOT OCCUR IF THE EQUATIONS ARE USED.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
C STORAGE   - 191 REGISTERS
C SPEED     -
C AUTHOR    - S.M. SIMPSON
C
C                    ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NOINT1
C       AND FORTRAN SYSTEM ROUTINES - SQRT, EXP(3
C
C FORTRAN USAGE
C      CALL KIINT1(CHISQ,NDF,PROB,IANS)
C
C INPUTS
C
C    CHISQ     IS THE PARTICULAR VALUE OF A CHI-SQUARED VARIATE.
C              MUST BE GRTHN=0.
C
C    NDF       IS THE NUMBER OF DEGREES OF FREEDOM OF THE VARIATE.
C              MUST BE GRTHN 0.
C
C OUTPUTS
C
C    PROB      IS THE PROBABILITY THAT THE VARIATE GRTHN=CHISQ.
C
C    IANS      =0  NORMAL
C              =1  ILLEGAL CHISQ
C              =2  ILLEGAL NDF
C
C EXAMPLES
C
C      THE AGREEMENT BETWEEN THE PROB VALUE IN THE EXAMPLES AND THE
C     COMPUTED PROB VALUE IS TO 3 OR FOUR PLACES SINCE 4 PLACE TABLES
C     WERE USED TO MAKE UP THE EXAMPLES.
C
C 1. INPUTS  - NDF=1   CHISQ=-1.
C    OUTPUTS - ERROR   IANS=1
C
C 2. INPUTS  - NDF=0   CHISQ=1.
C    OUTPUTS - ERROR   IANS=2
C
C 3. INPUTS  - NDF=1   CHISQ=1.
C    OUTPUTS - PROB=.3179   IANS=0
C
C 4. INPUTS  - NDF=8   CHISQ=2.7330
C    OUTPUTS - PROB=.95   IANS=0
C
C 5. INPUTS  - NDF=21   CHISQ=38.932
C    OUTPUTS - PROB=.01   IANS=0
C
```

```
C 6. INPUTS  - NDF=30   CHISQ=43.773
C     OUTPUTS - PROB=.05   IANS=0
C
C 7. INPUTS  - NDF=31   CHISQ=17.
C     OUTPUTS - PROB=.98   IANS=0
C
C 8. INPUTS  - NDF=3   CHISQ=2.366
C     OUTPUTS - PROB=.50   IANS=0
C
C
C INITIALIZE AND CHECK IF NORMAL CURVE APPROXIMATION IS TO BE USED.
      IANS=1
      IF(CHISQ)9999,10,10
 10   IANS=2
      IF(NDF) 9999,9999,12
 12   IANS=0
   15 CHI=SQRTF(CHISQ)
      IF (NDF-30)  20,20,70
C PROB IS COMPUTED IN THE FORM PROB = P1+P2*P3. CHECK NDF FOR EVEN, ODD.
   20 P2=(2.71828183)**(-CHISQ/2.0)
      NDFH=NDF/2
      IF (NDF-2*NDFH)  25,25,30
C EVEN.  SET P1=0, AND P3=1.0 IF NDF=2.
   25 P1=0.0
      IF (NDF-2)  27,27,50
   27 P3=1.0
      GO TO 60
C ODD.  COMPUTE P1, MODIFY P2 AND SET P3=0.0 IF NDF=1.
   30 CALL NOINT1(CHI,P1)
      P1=2.0*(1.0-P1)
      P2=CHI*P2*.79788480
      IF (NDF-1)  35,35,50
   35 P3=0.0
      GO TO 60
C EVALUATE P3 AS A POLYNOMIAL FOR NDF GREATER THAN 2.
   50 NLOOPS=NDFH-1
      P3=1.0
C IF NDF=3 (NLOOPS=0), P3=1.
      IF(NLOOPS)  60,60,52
 52   DIV=NDF-2
      DO 55  I=1,NLOOPS
      P3=P3*CHISQ/DIV+1.0
   55 DIV=DIV-2.0
      GO TO 60
C COMBINE PIECES TO FORM PROB.
   60 PROB=P1+P2*P3
      GO TO 9999
C USE NORMAL APPROXIMATION FOR NDF GREATER THAN 30.
   70 CHIMOD=CHI*1.414214-SQRTF(FLOATF(NDF)*2.0-1.0)
      CALL NOINT1(CHIMOD,P1)
      PROB=1.0-P1
      GO TO 9999
 9999 RETURN
      END
```

```
*     LINTR1 (SUBROUTINE)               2/18/63   LAST CARD IN DECK IS NO. 0092
*     LABEL
CLINTR1
      SUBROUTINE LINTR1(X,XLO,DELX,TABLE,NTABLE,YOFX)
C
C                   ----ABSTRACT----
C
C TITLE - LINTR1
C        LINEAR INTERPOLATION IN A TABLE
C
C              LINTR1 INTERPOLATES LINEARLY IN A TABLE TO FIND A VALUE
C              WHICH LIES BETWEEN THE TABULATED VALUES. XLO IS THE
C              ARGUMENT CORRESPONDING TO THE LOWEST TABULATED VALUE. DELX
C              IS THE ARGUMENT DIFFERENCE BETWEEN TABULAR VALUES.
C              THE TABLE IS LOCATED IN TABLE(I). X IS THE ARGUMENT AND
C              YOFX IS THE INTERPOLATED VALUE. HENCE
C
C                                                       XTRA
C              YOFX = TABLE(L) + (TABLE(L+1) - TABLE(L)) * ----
C                                                       DELX
C
C                   WHERE L IS SUCH THAT
C                        XLO+(L-1)*DELX LSTHN= X LSTHN XLO+L*DELX
C                   AND  XTRA = X-XLO-(L-1)*DELX
C
C              DELX IS CONSTRAINED TO BE POSITIVE
C              X MUST LIE IN THE ARGUMENT RANGE OF THE TABLE.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 709C (MAIN FRAME ONLY)
C STORAGE   - 96 REGISTERS
C SPEED     -
C AUTHOR    - S. M. SIMPSON
C
C                   ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C     CALL LINTR1(X,XLO,DELX,TABLE,NTABLE,YOFX)
C
C INPUTS
C
C    X          IS ARGUMENT FOR WHICH INTERPOLATION IS DESIRED.
C               XLC LSTHN OR = X LSTHN OR = XLO+(NTABLE-1)*DELX.
C
C    XLC        IS THE ARGUMENT CORRESPONDING TO THE FIRST TABULAR
C               ENTRY.
C
C    DELX       IS THE ARGUMENT DIFFERENCE BETWEEN TWO SUCCESSIVE
C               TABULAR ENTRIES.
C               MUST EXCEED C.0, BUT THIS CONSTRAINT IS NOT CHECKED.
C
C    TABLE(I)   I=1...NTABLE IS A GIVEN ARRAY IN WHICH TABLE(J)
C                 CONTAINS Y(XLO+DELX*(J-1)).
C
C    NTABLE     IS THE LENGTH OF THE TABLE.
C
C OUTPUTS
C
C    YOFX       WILL CONTAIN THE LINEARLY INTERPOLATED VALUE
C
C EXAMPLES
C
C 1. INPUTS  - X=7.5     XLO=5.     DELX=2.5     TABLE(1...9)=1.,4.,9.,
C              16.,25.,36.,49.,64.,81.     NTABLE=9
C    CUTPUTS - YOFX=4.
C
C 2. INPUTS  - SAME AS EXAMPLE 1. EXCEPT X=21.3
C    OUTPUTS - YOFX=56.8
C
C 3. INPUTS  - SAME AS EXAMPLE 1. EXCEPT X=25.
C    CUTPUTS - YOFX=81.
C
```

LINTR1 (PAGE 2)

```
C 4. INPUTS  - SAME AS EXAMPLE 1. EXCEPT X=13.
C     OUTPUTS - YOFX=17.8
C
      DIMENSION  TABLE(2)
C SET UP.
      XMXLO=X-XLO
   20 ILO=XMXLO/DELX+1.0
C INTERPOLATE ONLY IF ILO DOESNT CORRESPOND TO LAST TABULAR ENTRY.
      IF (ILO-NTABLE)  30,40,30
   30 FLILO=ILO-1
      DIFX=XMXLO-FLILO*DELX
      IHI=ILO+1
      YOFX=TABLE(ILO)+(TABLE(IHI)-TABLE(ILO))*DIFX/DELX
      GO TO 9999
   40 YOFX=TABLE(NTABLE)
      GO TO 9999
 9999 RETURN
      END
```

```
*      MAXSN (SUBROUTINE)               2/18/63   LAST CARD IN DECK IS NO. 0169
*      FAP
*MAXSN
       COUNT   150
       LBL     MAXSN
       ENTRY   MAXSN (LX,X,XMAX1,I)
       ENTRY   MINSN (LX,X,XMIN1,I)
       ENTRY   MAXAB (LX,X,XMAX2,I)
       ENTRY   MINAB (LX,X,XMIN2,I)
*
*                  ----ABSTRACT----
*
* TITLE - MAXSN , WITH SECONDARY ENTRY POINTS MINSN, MAXAB, AND MINAB
*       FIND SIGNED OR UNSIGNED EXTREMAL VALUES OF A VECTOR.
*
*             MAXSN FINDS THE MAXIMUM SIGNED NUMBER, AND ITS INDEX, IN
*               A VECTOR OF NUMBERS (EITHER FIXED OR FLOATING POINT).
*
*             MINSN FINDS THE MINIMUM SIGNED NUMBER.
*
*             MAXAB FINDS THE MAXIMUM OF THE ABSOLUTE VALUES.
*
*             MINAB FINDS THE MINIMUM OF THE ABSOLUTE VALUES.
*
* LANGUAGE  - FAP SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
* STORAGE   - 54 REGISTERS
* SPEED     - APPROX. 14N MACHINE CYCLES, N = LENGTH OF VECTOR
* AUTHOR    - J.F. CLAERBOUT
*
*                  ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - NONE
*       AND FORTRAN SYSTEM ROUTINES - NONE
*
* FORTRAN USAGE FOR MAXSN
*     CALL MAXSN (LX,X,XMAX1,I)
*
* INPUTS
*
*   X(I)       I=1...LX IS A VECTOR OF NUMBERS.
*              MAY BE FIXED OR FLOATING POINT.
*
*   LX         IS FORTRAN II INTEGER.
*              MUST BE GRTHN=1.
*
* OUTPUTS
*
*   XMAX1      IS THE MAXIMUM SIGNED VALUE IN THE X VECTOR.
*
*   I          IS THE INDEX OF THE MAXIMUM SIGNED VALUE.
*                I.E.   X(I) = XMAX1
*
* FORTRAN USAGE FOR MINSN
*     CALL MINSN (LX,X,XMIN1,I)
*
* INPUTS       SAME AS FOR MAXSN
*
* OUTPUTS
*
*   XMIN1      IS THE MINIMUM SIGNED VALUE IN THE X VECTOR
```

```
*
*    I          IS THE INDEX OF THE MINIMUM SIGNED VALUE.
*
* FORTRAN USAGE FOR MAXAB
*      CALL MAXAB (LX,X,XMAX2,I)
*
* INPUTS        SAME AS FOR MAXSN
*
* OUTPUTS
*
*    XMAX2      IS THE MAXIMUM ABSOLUTE VALUE IN THE X VECTOR.
*               NOTE THAT XMAX2 MAY BE NEGATIVE.
*
*    I          IS THE INDEX OF THE MAXIMUM ABSOLUTE VALUE.
*
* FORTRAN USAGE FOR MINAB
*      CALL MINAB (LX,X,XMIN2,I)
*
* INPUTS        SAME AS FOR MAXSN
*
* OUTPUTS
*
*    XMIN2      IS THE MINIMUM ABSOLUTE VALUE IN THE X VECTOR.
*               NOTE THAT XMIN2 MAY BE NEGATIVE.
*
*    I          IS THE INDEX OF THE MINIMUM ABSOLUTE VALUE.
*
* EXAMPLES
*
* 1. INPUTS   - X(1...10) = -11.,-8.,-5.,-2.,1.,4., 7.,10.,13.,16.
*               LX = 10
*    USAGE    -       CALL MAXSN (LX,X,XMAX1,I1)
*                     CALL MINSN (LX,X,XMIN1,I2)
*                     CALL MAXAB (LX,X,XMAX2,I3)
*                     CALL MINAB (LX,X,XMIN2,I4)
*    OUTPUTS -    XMAX1 = 16.     I1 = 10
*                 XMIN1 =-11.     I2 =  1
*                 XMAX2 = 16.     I3 = 10
*                 XMIN2 =  1.     I4 =  5
*
* 2. INPUTS   - X(1...10) = -16.,-13.,-10.,-7.,-4.,-1.,2.,5.,8.,11.
*               LX = 10
*    USAGE    - SAME AS EXAMPLE 1.
*    OUTPUTS -    XMAX1 = 11.     I1 = 10
*                 XMIN1 =-16.     I2 =  1
*                 XMAX2 =-16.     I3 =  1
*                 XMIN2 = -1.     I4 =  6
*
* 3. INPUTS   - X(1...10) = -16,-13,-10,-7,-4,-1,2,5,8,11   LX = 10
*    USAGE    - SAME AS EXAMPLE 1.
*    OUTPUTS      XMAX1 = 11     I1 = 10
*                 XMIN1 =-16     I2 =  1
*                 XMAX2 =-16     I3 =  1
*                 XMIN2 = -1     I4 =  6
*
        HTR     0
        BCI     1,MAXSN
 MAXSN  CLA     MX
        STO     USE
        TRA     *+3
 MINSN  CLA     MN
```

```
       STO    USE
       CLA    NOP
       STO    A-1
       CLA    SUB
       STO    A
       TRA    START
MAXAB  CLA    MX
       STO    USE
       TRA    *+3
MINAB  CLA    MN
       STO    USE
       CLA    SSP
       STO    A-1
       CLA    SBM
       STO    A
START  SXA    SV,1
       SXD    MAXSN-2,4
       CLA*   1,4
       PDX    ,1             ARRAY LENGTH TO IR1
       CLA    2,4
       ADD    =1
       STA    A+2
       STA    A
       CLA*   2,4            GET TRIAL
       STO*   3,4            EXTREMUM
       CLA    =1             SET CORRECT INDEX FOR TRIAL EXTREMUN
       ALS    18
       STO    INDEX
LOOP   CLA*   3,4
       HTR    0              EITHER NOP OR SSP
 A     HTR    **,1           EITHER SUB OR SBM
 USE   HTR    B              EITHERTPL OR TMI
       CLA    **,1
       STO*   3,4
       SXD    INDEX,1
 B     TIX    LOOP,1,1
       CLA    INDEX
       STO*   4,4
 SV    AXT    **,1
       TRA    5,4
NOP    NOP
SUB    SUB    0,1
SSP    SSP
SBM    SBM    0,1
MX     TPL    B
MN     TMI    B
INDEX  BSS    1
       END
```

```
*      MPSEQ1 (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0196
*      FAP
*MPSEQ1
       COUNT   200
       LBL     MPSEQ1
       ENTRY   MPSEQ1 (X,LX,B,LB,IX,IXLO,IANS)
*
*                   ----ABSTRACT----
*
* TITLE - MPSEQ1
*        MAPS A SEQUENCE OF NUMBERS INTO AN INTEGER SERIES
*
*              MPSEQ1 MAPS A SEQUENCE X(I), I=1,...,LX INTO AN INTEGER
*              SEQUENCE IX(I), I=1,...,LX.  THE MAPPING IS CONTROLLED BY
*              A GIVEN VECTOR OF RANGE LIMITS B(I), I=1,...,LB, WHERE
*              B(I) IS MONOTONELY INCREASING FROM B(1) TO B(LB), THUS
*              SPECIFYING LB-1 SEPARATE RANGES.  EACH RANGE IS CONSIDERED
*              CLOSED ON THE LOWER END, OPEN ON THE HIGH END AND THE
*              RANGES ARE INDEXED FROM IXLO+1 TO IXLO+LB-1, WHERE IXLO
*              IS A PARAMETER.  IX(I) IS THEN SET EQUAL TO THE INDEX OF
*              THE RANGE TO WHICH X(I) BELONGS, WITH THE FOLLOWING
*              TREATMENT OF EXTREMAL X VALUES
*                IF X(I) IS LSTHN  B( 1),  IX(I) = IXLO+1
*                IF X(I) IS GRTHN= B(LB),  IX(I) = IXLO+LB-1
*              NOTE- THE LOGIC USED IS ALMOST IDENTICAL TO THAT OF FRQCT2
*
* LANGUAGE  - FAP SUBROUTINE WITH FORTRAN II CALLING SEQUENCE
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
* STORAGE   - 110 REGISTERS
* SPEED     -
* AUTHOR    - J. N. GALBRAITH
*
*                   ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - NONE
*        AND FORTRAN SYSTEM ROUTINES - NONE
*
* FORTRAN USAGE
*     CALL MPSEQ1(X,LX,B,LB,IX,IXLO,IANS)
*
* INPUTS
*
*    X(I)      I=1...LX IS THE INPUT SERIES TO BE MAPPED.
*                MAY BE FLOATING, FORTRAN INTEGER, OR MACHINE LANGUAGE
*                INTEGER, BUT MUST BE THE SAME MODE AS B(J).
*
*    LX        IS LENGTH OF X VECTOR.
*              MUST BE GRTHN=1.
*
*    B(I)      I=1...LB GIVES INPUT RANGES OF MAPPING INTERVALS.
*              MUST BE SAME MODE AS X(I).
*              B(I) MUST INCREASE MONOTONELY, IE B(I+1) GRTHN B(I)
*
*    LB        IS LENGTH OF RANGE VECTOR.
*              MUST BE GRTHN=1.
*
*    IXLO      IS LOWER LIMIT OF OUTPUT MAPPING.  IXLO+1 = INDEX OF
*              LOWEST RANGE.
*
* OUTPUTS
*
*    IX(I)     I=1...LX IS THE INTEGER MAPPING OF X(I).
*
*    IANS      =0  NORMAL
*              =1  ILLEGAL LX
*              =2  ILLEGAL LB
*              =3  WEIRD ERROR
*
* EXAMPLES
*
* 1. INPUTS  - LX=0   X(1...16)=-5.,-4.,-3.2,-3.1,-2.,-2.1,0.,-1.1,
*              -.5,5.,4.,3.5,3.,2.9,1.1,1.   LB=16   B(1...9)=-4.,-3.,
*              -2.,-1.,0.,1.,2.,3.,4.,    IXLO=0
*    OUTPUTS - ERROR  IANS=1
*
```

```
* 2. INPUTS  - X AND B SAME AS EXAMPLE 1  LX=16  LB=0  IXLO=0
*    OUTPUTS - ERROR  IANS=2
*
* 3. INPUTS  - X AND B SAME AS EXAMPLE 1  LX=16  LB=9  IXLO=0
*    OUTPUTS - IX(1,...,16)=0,0,0,0,2,1,4,2,3,7,7,7,7,6,5,5   IANS=0
*
* 4. INPUTS  - X, B, LX, AND LB SAME AS EXAMPLE 3  IXLO=12
*    OUTPUTS - IX(1,...,16)=12,12,12,12,14,13,16,14,15,19,19,19,19,18,
*              17,17   IANS=0
*
        PZE     0
        BCI     1,MPSEQ1
MPSEQ1  SXA     RETURN,1
        SXA     RETURN+1,2
        SXA     RETURN+2,4
        SXD     MPSEQ1-2,4
        STZ*    7,4            IANS=0
        CLA*    2,4            GET LX
        TZE     ERR1
        TMI     ERR1
        STD     END
        CLA*    4,4            GET LB
        TZE     ERR2
        TMI     ERR2
        ARS     18             LB IN ADDRESS
        STO     LB
        ARS     1              LB/2 (IN ADDRESS)
        STO     LBHALF
        CLA     1,4            ADDRESS OF X
        ADD     K1MLI          A(X+1)
        STA     XADD
        STA     TESTLO
        CLA     3,4            ADDRESS OF B
        ADD     K1MLI          A(B+1)
        STA     BTEST1
        STA     BADD
        SUB     LB
        STA     TESTHI
        CLA*    6,4            GET IXLO
        SUB     K2FX           IXLO-2
        STO     XLOW
        CLA     5,4            ADDRESS OF IX
        ADD     K1MLI          A(IX+1)
        STA     IXSTO
        AXT     1,1
        AXT     1,4
 LOOP   CLA     K1MLI
        STO     LBLO           INITIAL LBLO=1
        CLA     LB
        STO     LBHI           INITIAL LBHI=LB
        CLA     LBHALF
        STO     LBCOM          INITIAL LBCOM=LB/2
        AXT     1,2
TESTLO  CLA     **,1           GET X.  (**=A(X+1))
BTEST1  CAS     **,4           B(1)  SEE IF IN LOWEST RANGE
        TRA     TESTHI
        TRA     NEXIND
        TRA     NEXIND
TESTHI  CAS     **             **=A(B(LB)). SEE IF IN HIGHEST RANGE
        TRA     HIEST
        TRA     HIEST
SEARCH  LXA     LBCOM,2
 XADD   CLA     **,1           GET X(IR1)
 BADD   CAS     **,2           COMPARE WITH B(LBCOM)
        TRA     GRATER         X GREATER, NEW LBLO (=LBCOM)
        TRA     NEXIND         GOT IT, SET IX(IR1+1)
 LESS   PXA     0,2            X LESS, NEW LBHI (=LBCOM)
        SUB     LBLO           LBCOM-LBLO=DIF
        CAS     K1MLI
        TRA     *+3            DIF GREATER THAN ONE
        TRA     EQUAL          DIF=1, GOT IT, SET IX(IR1+1)
        TRA     ERROR             IMPOSSIBLE
        ARS     1              DIF/2
        ADD     LBLO           NEW LBCOM
        LDQ     LBCOM
```

```
           STQ    LBHI
           STO    LBCOM
           TRA    SEARCH
    GRATER PXA    0,2
           SUB    LBHI         LBCOM-LBHI=-DIF
           SSP                 DIF
           CAS    K1MLI
           TRA    *+3
           TRA    NEXIND       DIF=1, GOT IT, SET IX(IR1+1)
           TRA    ERROR        IMPOSSIBLE
           ARS    1
           ADD    LBCOM
           LDQ    LBCOM
           STO    LBCOM
           STQ    LBLO
           TRA    SEARCH
    NEXIND TXI    *+1,2,1
     EQUAL PXD    ,2
           ADD    XLOW
     IXSTO STO    **,1         **= ADDRESS OF IX+1
           TXI    *+1,1,1
     END   TXL    LOOP,1,**    **=LX
    RETURN AXT    **,1
           AXT    **,2
           AXT    **,4
           TRA    8,4
     HIEST LXA    LB,2
           TRA    EQUAL
     ERR1  CLA    K1FX
           STO*   7,4          STORE IANS
           TRA    8,4          RETURN
     ERR2  CLA    K2FX
           TRA    ERR1+1
     ERROR CLA    K3FX
           TRA    ERR1+1
    * CONSTANTS AND TEMPORARIES
     K1FX  PZE    0,0,1
     K2FX  PZE    0,0,2
     K3FX  PZE    0,0,3
     K1MLI PZE    1,0,0
     LB    PZE    0
    LBHALF PZE    0
    LBLO   PZE    0
    LBCOM  PZE    0
    LBHI   PZE    0
     XLOW  PZE
           END
```

```
*     MSCON1 (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0107
*     LABEL
CMSCON1
      SUBROUTINE MSCON1 (NORDER,P,PHI,DEPEND,IANS)
C
C                    ----ABSTRACT----
C
C TITLE - MSCON1
C       MEAN SQUARE CONTINGENCY AND DEPENDENCY FROM PROBABILITY DENSITY.
C
C             MSCON1 COMPUTES THE MEAN SQUARE CONTINGENCY AND A
C             DEPENDENCY MEASURE AS DEFINED ON PAGE 282 OF CRAMER,
C             MATHEMATICAL METHODS OF STATISTICS, PRINCTON UNIV. PRESS,
C             1951. THE COMPUTATION REQUIRES THE SECOND PROBABILITY
C             DENSITY WHICH CAN BE COMPUTED WITH SUBROUTINE PROB2 (SEE
C             WRITE-UP OF PROB2). IF PHI IS THE MEAN SQUARE CONTINGENCY,
C             DEPEND IS THE DEPENDENCY MEASURE, AND NORDER IS THE ORDER
C             OF THE SECOND PROBABILITY MATRIX, P(I,J), THEN
C
C                       DEPEND = PHI/(NORDER-1)
C
C LANGUAGE   - FORTRAN II SUBROUTINE
C EQUIPMENT  - 709, 7090 (MAIN FRAME ONLY)
C STORAGE    - 238 REGISTERS
C SPEED      -
C AUTHOR     - J.N. GALBRAITH
C
C                    ----USAGE---
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C     CALL MSCON1(NORDER,P,PHI,DEPEND,IANS)
C
C INPUTS
C
C    NORDER    INTEGER. THE ORDER OF THE P(I,J) PROBABILITY DENSITY
C              MATRIX. GRTHN ONE, LSTHN OR EQUAL 25.
C
C    P(I,J)    I=1,..,NORDER, J=1,..,NORDER. PROBABILITY DENSITY MATRIX
C              NORMALIZED SUCH THAT THE SUM OVER I AND J IS = TO 1.
C              P(I,J) HAS DIMENSION (25,25), P(I,J) MUST NOT HAVE AN
C              ENTIRE ROW OR COLUMN SUM EQUAL TO ZERO, OR NEGATIVE.
C
C OUTPUTS
C
C    PHI       THE MEAN SQUARE CONTINGENCY.
C
C    DEPEND    THE DEPENDENCY MEASURE.
C
C    IANS      ERROR INDICATOR
C              =0  NORMAL
C              =-1 ILLEGAL NORDER. LSTHN 1 OR GRTHN 25
C              =-2 ILLEGAL P MATRIX. ROW OR COLUMN SUM ZERO OR NEGATIVE.
C
C EXAMPLES
C
C 1. INPUTS  - P(1,1)=.2 ,P(I,I),I=2,5 =.1, P(1,I),I=2,5 =.1
C              ALL OTHER P(I,J)=0.
C              NORDER=0
C    OUTPUTS - PHI=0.  DEPEND=0.  IANS=-1
C
C 2. INPUTS  - SAME AS EXAMPLE 1 EXCEPT
C              NORDER=26
C    OUTPUTS - PHI=0.  DEPEND=0.  IANS=-1
C
C 3. INPUTS  - SAME AS EXAMPLE 1 EXCEPT
C              NORDER=5
C    OUTPUTS - PHI=1.6666666  DEPEND=.41666666  IANS=0
C
C 4. INPUTS  - SAME AS EXAMPLE 1 EXCEPT
C              P(1,5)=0., P(5,1)=.1  NORDER=5
C    OUTPUTS - PHI=1.7333333   DEPEND=.43333333   IANS=0
C
```

```
C 5. INPUTS  - SAME AS EXAMPLE 4 EXCEPT
C                 P(5,5)=0.
C     OUTPUTS - IANS=-2
C
      DIMENSION P(25,25),PSROW(25),PSCOL(25)
C     CHECK NORDER
      IANS=-1
      IF(NORDER-1) 9999,9999,5
    5 IF(NORDER-26) 6,9999,9999
C     FIND ROW AND COLUMN SUMS
    6 DO 10 J=1,NORDER
      PSROW(J)=0.
      PSCOL(J)=0.
      DO 10 I=1,NORDER
      PSROW(J)=PSROW(J)+P(J,I)
   10 PSCOL(J)=PSCOL(J)+P(I,J)
C     CHECK ROW AND COLUMN SUMS
      IANS=-2
      DO 15 I=1,NORDER
      IF(PSROW(I)) 9999,9999,12
   12 IF(PSCOL(I)) 9999,9999,15
   15 CONTINUE
C     COMPUTE MEAN SQUARE CONTINGENCY
      PHI=0.
      DO 20 I=1,NORDER
      DO 20 J=1,NORDER
   20 PHI=PHI+P(I,J)*P(I,J)/(PSROW(I)*PSCOL(J))
      PHI=PHI-1.
C     COMPUTE DEPENDENCY MEASURE
      DEPEND=PHI/(FLOATF(NORDER-1))
      IANS=0
 9999 RETURN
      END
```

```
*      NOINT1 (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0374
*      FAP
*NOINT1
       COUNT   370
       LBL     NOINT1
       ENTRY   NOINT1 (X,PROB)
       ENTRY   NOINT2 (XMEAN,XSD,NDIV,XDIV,IANS)
*
*                    ----ABSTRACT----
*
* TITLE - NOINT1 WITH SECONDARY ENTRY NOINT2
*       NORMAL DISTRIBUTION AND DIVISION INTO EQUALLY LIKELY SECTIONS
*
*             NOINT1 FINDS THE INTEGRAL OF THE ZERO MEAN, UNIT VARIANCE,
*             NORMAL PROBABILITY DENSITY FUNCTION FROM MINUS INFINITY
*             TO X. THIS IS DONE BY TABLE LOOK UP IN A TABLE OF 201
*             VALUES OF THE NORMAL DISTRIBUTION WHICH CORRESPOND
*             TO VALUES OF X FROM 0.0 TO 4.0 IN INCREMENTS OF .02
*             LINEAR INTERPOLATION IS USED FOR VALUES OF X LYING
*             BETWEEN TABULATED VALUES. THE PROGRAM RETURNS ZERO FOR X
*             VALUES LESS THAN -4.0, AND RETURNS 1.0 FOR X VALUES
*             GREATER THAN 4.0.
*
*             NOINT2 DIVIDES UP THE ENTIRE X AXIS INTO AN ARBITRARY
*             NUMBER, NDIV, OF RANGES WHICH ARE EQUALLY LIKELY WITH
*             RESPECT TO A GIVEN NORMAL DISTRIBUTION SPECIFIED BY
*             ITS MEAN AND STANDARD DEVIATION.
*
*             THE INTEGRAL OF THE NORMAL DISTRIBUTION GIVES THE
*             PROBABILITY THAT X LIES IN A CERTAIN RANGE.  NOINT2
*             REVERSES THE PROCESS BY FINDING THE X RANGES WITH
*             A GIVEN PROBABILITY.  1/NDIV = PROBABILITY FOR EACH
*             DIVISION.  FOR K-TH DIVISION, XAXIS LIMITS CORRESPOND
*             TO THE PROBABILITIES (K-1)/NDIV, K/NDIV.  STORED VALUES
*             OF THE ANTISYMMETRIC INTEGRAL OF THE UNIT NORMAL
*             DISTRIBUTION FOR X VALUES ZERO TO 4 IN INCREMENTS OF .02
*             ARE SEARCHED FOR PROBABILITY VALUES GIVEN BY K/NDIV.
*             INTERPOLATION WHERE NECESSARY IS LINEAR. I.E. FIND NEAREST
*             VALUE OF X TO CORRESPONDING TO P WHEN P DOES NOT APPEAR
*             IN TABLE EXACTLY.  IF R-TH VALUE IN TABLE IS LESS THAN P,
*             AND (R+1) TH VALUE IS GREATER, THEN X VALUE = ((P-RTH
*             VALUE)/((R+1)TH-RTH VALUE))*.02+R*.02.  THIS VALUE IS
*             THEN SCALED FOR THE PARTICULAR NORMAL DISTRIBUTION SUCH
*             THAT THE OUTPUT X = X*XSD+MEAN.  SINCE ONLY HALF OF THE
*             NORMAL INTEGRAL IS STORED, THE X VALUES CORRESPONDING TO
*             P1 GREATER THAN .5 ARE COMPUTED FIRST AND THE VALUES
*             FOR P2 LESS THAN .5 ARE SYMMETRIC AND EQUAL TO 1-P1.
*
*             NOTE - NOINT1 AND NOINT 2 ARE INDEPENDENT EXCEPT FOR
*             THEIR MUTUAL NEED OF THE DISTRIBUTION FUNCTION TABLE.
*
* LANGUAGE  - FAP SUBROUTINE (FORTRAN II COMPATIBLE)
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
* STORAGE   - 369 REGISTERS
* SPEED     -
* AUTHOR    - S.M. SIMPSON AND J.N. GALBRAITH
*
*                    ----USAGE----
*
* TRANSFER VECTOR CONTAINS ROUTINES - LINTR1
*       AND FORTRAN SYSTEM ROUTINES - NONE
*
* FORTRAN USAGE OF NOINT1
*     CALL NOINT1(X,PROB)
*
* INPUTS TO NOINT1
*
*    X          = UPPER LIMIT OF THE INTEGRAL (FLT PT.).
*
* OUTPUTS FROM NOINT1
*
*                     1         X          2
*    PROB       = ---------  INTEGRAL (EXP(-X /2)DX).
*                 SQRT(2PI) -INFINITY
*
```

NOINT1 (PAGE 2)

```
*               IS FLOATING POINT
*
* FORTRAN USAGE OF NOINT2
*      CALL NOINT2(XMEAN,XSD,NDIV,XDIV,IANS)
*
* INPUTS TO NOINT2
*
*     XMEAN      = MEAN OF X SERIES
*
*     XSD        = STANDARD DEVIATION OF X SERIES.
*                  MUST BE GRTHN 0.
*
*     NDIV       = NUMBER OF EQUALLY LIKELY DIVISIONS INTO WHICH XSERIES
*                    IS TO BE PLACED.
*                  MUST BE GRTHN 1
*
* OUTPUTS FROM NOINT2
*
*     XDIV(I)    I=1...NDIV-1 ARE THE X VALUES FOR EQUALLY LIKELY
*                  DIVISIONS. FIRST DIVISION IS FROM -INFINITY TO XDIV(1),
*                  THE SECOND IS FROM XDIV(1) TO XDIV(2) ETC. THE LAST
*                  DIVISION IS FROM XDIV(NDIV-1) TO +INFINITY.
*
*     IANS       =0  NORMAL
*                =1  ILLEGAL XSD
*                =2  ILLEGAL NDIV
*
* EXAMPLES OF NOINT1
*
* 1. INPUTS  - X=-5.
*    OUTPUTS - PROB=0.
*
* 2. INPUTS  - X=-4.
*    OUTPUTS - PROB=.32 E-04
*
* 3. INPUTS  - X=.013
*    OUTPUTS - PROB=.5052
*
* 4. INPUTS  - X=4.
*    OUTPUTS - PROB=.999968
*
* 5. INPUTS  - X=4.1
*    OUTPUTS - PROB=1.
*
* EXAMPLES OF NOINT2
*
* 1. INPUTS  - XMEAN=0.       XSD=1.       NDIV=3
*    OUTPUTS - XDIV(1)=-.430722     XDIV(2)=.430722      IANS=0
*
* 2. INPUTS  - XMEAN=0.       XSD=2.       NDIV=3
*    OUTPUTS - XDIV(1)=-.861444     XDIV(2)=.861444      IANS=0
*
* 3. INPUTS  - XMEAN=1.       XSD=2.       NDIV=3
*    OUTPUTS - XDIV(1)=.1385185     XDIV(2)=1.861444     IANS=0
*
* 4. INPUTS  - XMEAN=0.       XSD=1.       NDIV=2
*    OUTPUTS - XDIV(1)=0.          IANS=0
*
* 5. INPUTS  - XMEAN=3.5      XSD=1.       NDIV=2
*    OUTPUTS - XDIV(1)=3.5         IANS=0
*
* 6. INPUTS  - XMEAN=3.5      XSD=1.       NDIV=1
*    OUTPUTS - ERROR  IANS=2
*
* 7. INPUTS  - XMEAN=3.5      XSD=0.       NDIV=2
*    OUTPUTS - ERROR  IANS=1
*
* 8. INPUTS  - XMEAN=0.       XSD=1.       NDIV=4
*    OUTPUTS - XDIV(1...3)=-.674602,0.,+.674602     IANS=0
*
* 9. INPUTS  - XMEAN=0.       XSD=1.       NDIV=5
*    OUTPUTS - XDIV(1...4)=-.8417856,-.253334,.253334,.8417856    IANS=0
*
*INITIALIZE.
       PZE     0
```

```
         BCI     1,NOINT1
  NOINT1 SXA     LV,4
         SXD     NOINT1-2,4
         CLA     1,4
         STA     GETX
         CLA     2,4
         STA     STORE
  *GET,STORE X AND ITS SIZE. COMPARE SIZE WITH 4.0.
   GETX  CLA     **            **=ADDRESS OF X
         STO     XX
         SSP
         STO     SX
         CAS     K4FL
         TRA     BIGGER
         TRA     INTRP
         TRA     INTRP
  *(OR ZERO FOR NEG X).
  BIGGER CLA     K1FL
         STO     TEMP
         TRA     CHECK
  *INTERPOLATE IF SIZE LESS THAN OR = 4.0.
  *NOTE LINTR1 MUST BE USED BACKWARDS SINCE OUR
  *TABLE IS FORWARDS.
   INTRP CLA     K4FL
         FSB     SX
         STO     SXMOD
         TSX     $LINTR1,4
         TSX     SXMOD         SXMOD=4.0-SX
         TSX     K0            XLO=0.0
         TSX     KDELX         KDELX=0.02
         TSX     Y+200         TABLE IS FORTRAN VECTOR
         TSX     KD201         NTABLE=201
         TSX     TEMP          ANSWER
  *IF X WAS MINUS WE NEED 1.0 MINUS THE INTERPOLATED
  *VALUE.
   CHECK CLA     XX
         TPL     STORE-1
         CLA     K1FL
         FSB     TEMP
         TRA     STORE
         CLA     TEMP
   STORE STO     **            **=ADDRESS OF PROB
      LV AXT     **,4          **=XR4
         TRA     3,4
  *TEMPORARIES
      XX PZE     **            **=X
      SX PZE     **            **=MAGNITUDE OF X
   SXMOD PZE     **            **=4.0-SX
    TEMP PZE     **            **=OUTPUT FROM LINTR1
  *CONSTANTS
      K0 PZE     0
   KD201 PZE     0,0,201
    K1FL DEC     1.0
    K4FL DEC     4.0
   KDELX DEC     0.02
  *      ENTRY   NOINT2 (XMEAN,XSD,NDIV,XDIV,IANS)
  *      SAVE IRS AND INITIALIZE IANS
         PZE     0
         BCI     1,NOINT2
  NOINT2 SXA     RETURN,1
         SXA     RETURN+1,2
         SXA     RETURN+2,4
         SXD     NOINT2-2,4
         STZ*    5,4              IANS=0
  *      CHECK XSD AND NDIV.
         CLA*    2,4              GET XSD
         TZE     ERR1             TRANSFER IF ILLEGAL
         TMI     ERR1             TRANSFER IF ILLEGAL
         CLA*    3,4              GET NDIV
         SUB     K1FX             NDIV-1
         TZE     ERR2             TRANSFER IF ILLEGAL
         TMI     ERR2             TRANSFER IF ILLEGAL
  *      PARAMETERS OK. SET UP MEAN LOOP AND GET XSD AND XMEAN ADDRESSES.
         STD     END2             SET UP MEAN LOOP
         CLA     4,4              ADDRESS OF XDIV
```

* NOINT1 * (PAGE 4)

```
         ADD    KMLI1
         STA    LOOP2
         STA    MEAN+1
         CLA    1,4            ADDRESS OF XMEAN
         STA    MEAN
         LDQ*   2,4
         FMP    KDELX
         STO    SCALE
         CLA    4,4            A(XDIV)
         CLA*   3,4            GET NDIV
         LRS    18             FLOAT IT
         ORA    CONST
         FAD    CONST
         STO    NDIVFL         NDIVFL=FLOATF(NDIV)
         CLA    K1FL
         FDP    NDIVFL
         STQ    DELP
         CLA*   3,4            GET NDIV
         LGR    19
*        NDIV/2 WITH REMAINDER IN SIGN OF MQ
         PAX    ,1
         SXD    END,1
         SSM
         ADD    4,4            (ADDRESS OF XDIV)-NDIV/2
         ADD    KMLI1          ADDRESS OF XDIV(NDIV/2)
         STA    STO1
         STA    STO2
         TQP    EVEN           TRANSFER IF NDIV EVEN
         CLA    DELP
         FDP    K2FL
         XCA
         FAD    Y              P=(.5+DELP/2)
         STO    P
         AXT    0,1
         AXT    1,2
         AXT    0,4
         TRA    SEARCH
EVEN     AXT    0,2
         CLA    Y              .5
         STO    P
         STZ*   STO1
         AXT    1,2
         AXT    -1,4
         AXT    0,1
LOOP     CLA    P
         FAD    DELP
         STO    P
SEARCH   CAS    Y,1            P IS IN AC
         TXI    SEARCH,1,-1     TRY AGAIN
         TRA    SKINT          GOT IT. SKIP INTERPOLATION
         FSB    Y-1,1          INTERPOLATE. P-RTH VALUE
         STO    XTEMP1
         CLA    Y,1            (R+1)TH
         FSB    Y-1,1          RTH
         STO    XTEMP2
         CLA    XTEMP1
         FDP    XTEMP2
         FMP    SCALE
         STO    XTEMP1
         TRA    SKINT+1
SKINT    STZ    XTEMP1         ZERO INTERPOLATION
         TXI    *+1,1,1        COMPLEMENT OF INDEX OF RTH VALUE IN IR1
         SXA    XTEMP2,1
         PXA    ,1             GET IR1
         PAC    ,1             2 COMPLEMENT
         PXA    ,1             INDEX FOR RTH VALUE =N
         ORA    CONST          FLOAT
         FAD    CONST
         XCA                   FLOATF(N)=FLN IN MQ
         FMP    SCALE          FLN*.02*XSD=X
         FAD    XTEMP1
STO1     STO    **,2           **=A(XDIV)-NDIV/2+1
         SSM
STO2     STO    **,4           **=A(XDIV)-NDIV/2+1
         LXA    XTEMP2,1
```

NOINT1 (PAGE 5)

```
         TXI    *+1,4,-1
         TXI    *+1,2,1
   END   TXL    LOCP,2,**     **=NDIV/2  ROUNDED DOWN
  *      FINISHED SEARCH AND SCALING FOR ALL BLOCKS. ADD MEAN
         AXT    1,2
   LOOP2 CLA    **,2          **=A(XDIV)+1
   MEAN  FAD    **            XMEAN
         STO    **,2
         TXI    *+1,2,1
   END2  TXL    LOOP2,2,**    **=NDIV-1
  RETURN AXT    **,1
         AXT    **,2
         AXT    **,4
         TRA    6,4
   ERR1  CLA    K1FX
         STO*   5,4
         TRA    6,4
   ERR2  CLA    K2FX
         STO*   5,4
         TRA    6,4
   CONST OCT    233000000000
   K1FX  PZE    0,0,1
   K2FX  PZE    0,0,2
   KMLI1 PZE    1
   K2FL  DEC    2.0
  XTEMP1 PZE    0
  XTEMP2 PZE    0
    P    PZE    0
   DELP  PZE    0
  NDIVFL PZE
   SCALE PZE    0
  *TABLE (YULE AND KENDALL, THEORY OF STATISTICS,
  *1950, PAGE 664.)
       Y DEC .5000,.5080,.5160,.5239,.5319
         DEC .5398,.5478,.5557,.5636,.5714
         DEC .5793,.5871,.5948,.6026,.6103
         DEC .6179,.6255,.6331,.6406,.6480
         DEC .6554,.6628,.6700,.6772,.6844
         DEC .6915,.6985,.7054,.7123,.7190
         DEC .7257,.7324,.7389,.7454,.7517
         DEC .7580,.7642,.7704,.7764,.7823
         DEC .7881,.7939,.7995,.8051,.8106
         DEC .8159,.8212,.8264,.8315,.8365
         DEC .8413,.8461,.8508,.8554,.8599
         DEC .8643,.8686,.8729,.8770,.8810
         DEC .8849,.8888,.8925,.8962,.8997
         DEC .9032,.9066,.9099,.9131,.9162
         DEC .9192,.9222,.9251,.9279,.9306
         DEC .9332,.9357,.9382,.9406,.9429
         DEC .9452,.9474,.9495,.9515,.9535
         DEC .9554,.9573,.9591,.9608,.9625
         DEC .9641,.9656,.9671,.9686,.9699
         DEC .9713,.9726,.9738,.9750,.9761
         DEC .9772,.9783,.9793,.9803,.9812
         DEC .9821,.9830,.9838,.9846,.9854
         DEC .9861,.9868,.9875,.9881,.9887
         DEC .9893,.9898,.9904,.9909,.9913
         DEC .9918,.9922,.9927,.9931,.9934
         DEC .99379,.99413,.99446,.99477,.99506
         DEC .99534,.99560,.99585,.99609,.99632
         DEC .99653,.99674,.99693,.99711,.99728
         DEC .99744,.99760,.99774,.99788,.99801
         DEC .99813,.99825,.99836,.99846,.99856
         DEC .99865,.99874,.99882,.99889,.99897
         DEC .99903,.99910,.99916,.99921,.99926
         DEC .99931,.99936,.99940,.99944,.99948
         DEC .99952,.99955,.99958,.99961,.99964
         DEC .99966,.99969,.99971,.99973,.99975
         DEC .99977,.99978,.99980,.99981,.99983
         DEC .99984,.99985,.99986,.99987,.99988
         DEC .99989,.99990,.999908,.999915,.999922
         DEC .999928,.999933,.999939,.999943,.999948
         DEC .999952,.999956,.999959,.999963,.999966
         DEC .999968
         END
```

```
************************
*    NOINT2            *
************************
 REFER TO
     NOINT1
```


```
************************
*    NOINT2            *
************************
 REFER TO
     NOINT1
```

```
*     PCKCT1 (SUBROUTINE)             2/18/63   LAST CARD IN DECK IS NO. 0131
*     LABEL
CPOKCT1
      SUBROUTINE POKCT1 (IX,NHANDS,ICT,IANS)
C
C                     ----ABSTRACT----
C
C TITLE - POKCT1
C       EVALUATION OF INTEGER SEQUENCE IN GROUPS OF FIVE AS POKER HANDS.
C
C              POKCT1 BREAKS UP A FORTRAN II INTEGER SEQUENCE INTO NON-
C              OVERLAPPING GROUPS OF FIVE DIGITS WHICH IT TREATS AS POKER
C              HANDS. THE HANDS ARE EVALUATED AND A TABULATION OF THE
C              NUMBER OF DIFFERENT TYPES OF HANDS IS PRODUCED. THE A
C              PRICRI PROBABILITIES OF DIFFERENT HAND TYPES ARE KNOWN FOR
C              THE CASE OF INDEPENDENT EQUALLY LIKELY DIGITS FROM ZERO TO
C              NINE. HENCE A POKER COUNT IS USEFUL IN DETERMINING THE
C              INCEPENDENCE OF A SEQUENCE. THE A PRIORI PROBABILITIES
C              ARE GIVEN BELOW AND ARE EXACT. THE DECIMALS TERMINATE AT
C              THE FOURTH PLACE.
C                  BUST                            .2952
C                  1 PAIR                          .5040
C                  2 PAIR                          .1080
C                  3 CF A KIND                     .0720
C                  FULL HOUSE                      .0090
C                  STRAIGHT                        .0072
C                  4 CF A KIND                     .0045
C                  5 OF A KIND                     .0001
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)
C STORAGE   - 219 REGISTERS
C SPEED     -
C AUTHOR    - S.M. SIMPSCN
C
C                    ----USAGE----
C
C TRANSFER VECTOR CCNTAINS ROUTINES - FRQCT1
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C      CALL PCKCT1(IX,NHANDS,ICT,IANS)
C
C INPUTS
C
C    IX(I)      I=1...5*NHANDS IS THE DIGIT SEQUENCE
C               ZERO LESS THAN OR = IX LESS THAN OR = 9
C
C    NHANDS     IS THE NUMBER OF HANDS TO BE FORMED FROM THE IX SEQUENCE.
C               NHANDS MUST BE GREATER THAN ZERO.
C
C CUTPUTS
C
C    ICT(I)     I=1...8 IS THE COUNT OF TYPES OF HANDS FOUND WHERE
C               ICT(1) = NO. OF HANDS OF NO VALUE
C               ICT(2) = NO. OF HANDS WITH 1 PAIR
C               ICT(3) = NO. OF HANDS WITH 2 PAIRS
C               ICT(4) = NO. OF HANDS WITH 3 OF A KIND
C               ICT(5) = NO. OF STRAIGHTS
C               ICT(6) = NO. OF FULL HOUSES
C               ICT(7) = NO. OF HANDS WITH 4 OF A KIND
C               ICT(8) = NO. OF HANDS WITH 5 OF A KIND
C               WHERE HAND NO. 1 =(IX(1),IX(2),IX(3),IX(4),IX(5))
C                     HAND NO. 2 =(IX(6),IX(7),IX(8),IX(9),IX(10))
C                            ETC.
C               AND SUM OF ICT(I) = NHANDS.
C
C    IANS       =0  NCRMAL
C               =1  ILLEGAL HANDS
C               =3  ERROR RETURN FROM FRQCT1
C
C EXAMPLES
C
C 1. INPUTS   - NHANDS = 0
C    IX(I)      I=1,280 BROKEN INTO GROUPS OF FIVE FOR EASY CHECKING.
```

```
C                 40123  43125  23456  52643  76543  87654  95867
C                 97654  02345  98762  14327  02678  86430  63142
C                 01230  18741  32024  99413  08628  54531  07499
C                 01220  42246  45999  94977  82238  77335  55060
C                 10020  23334  06033  88381  74877  06006  15113
C                 11222  21212  80808  94449  55454  61116  06006
C                 90000  66866  44644  88883  21111  00700  09999
C                 99999  00000  11111  22222  66666  33333  36410
C     OUTPUTS - ICT(1...8) = 0,0,0,0,0,0,0,0   IANS=1
C
C 2. INPUTS  - SAME AS EXAMPLE 1. EXCEPT NHANDS=56
C     OUTPUTS - ICT(1...8) = 8,7,7,6,7,8,7,6  IANS=0
C
      DIMENSION  IX(2),ICT(2),IC1(10),IC2(6)
C CLEAR THE OUTPUT VECTOR.  THEN WORK THRU DATA HAND BY HAND.
      IANS=1
      IF(NHANDS) 9999,9999,10
 10   IANS=0
      DO 15 I=1,8
   15 ICT(I)=0
      DO 90  II=1,NHANDS
C FOR EACH HAND FIRST MAKE A FREQUENCY COUNT OF THE DIGITS (VALUES 0-9).
C NOTE RESTRICTION 1 VIOLATION IS CAUGHT BY FRQCT1.
      J=(II-1)*5+1
      CALL FRQCT1(IX(J),5,0,9,IC1,IANS)
      IF(IANS)  9991,21,9991
C AND THEN MAKE A FREQUENCY COUNT OF THE FREQUENCY COUNT (VALUES 0 TO 5)
 21   CALL FRQCT1(IC1,10,0,5,IC2,IANS)
      IF(IANS)  9991,22,9991
C THE HAND VALUE, IVAL (1 TO 8), IS DETERMINABLE FROM IC2(1),IC2(3),
C IC2(2) EXCEPT FOR STRAIGHTS.
 22   IVAL=1
      IF (IC2(1)-6)  60,92,50
   50 IF (IC2(3)-1)  55,96,93
   55 IF (IC2(2)-1)  98,97,94
C CHECK FOR POSSIBLE STRAIGHT WHEN ALL DIGITS ARE DIFFERENT.
   60 I=0
   62 I=I+1
      IF (IC1(I))  70,62,70
   70 IF (IC1(I+1))  71,91,71
   71 IF (IC1(I+2))  72,91,72
   72 IF (IC1(I+3))  73,91,73
   73 IF (IC1(I+4))  95,91,95
C SET THE HAND VALUE.
   98 IVAL=IVAL+1
   97 IVAL=IVAL+1
   96 IVAL=IVAL+1
   95 IVAL=IVAL+1
   94 IVAL=IVAL+1
   93 IVAL=IVAL+1
   92 IVAL=IVAL+1
   91 ICT(IVAL)=ICT(IVAL)+1
   90 CONTINUE
 9999 RETURN
 9991 IANS=3
      GO TO 9999
      END
```

```
*     POLYDV (SUBROUTINE)              2/18/63   LAST CARD IN DECK IS NO. 0100
*     LABEL
CPOLYDV
      SUBROUTINE POLYDV (N,DVS,M,DVD,L,Q)
C
C                    ----ABSTRACT----
C
C TITLE - POLYDV
C      PERFORM LONG DIVISION OF TWO POLYNOMIALS
C
C            POLYDV COMPUTES THE FIRST L COEFFICIENTS OF THE QUOTIENT
C            OF TWO POLYNOMIALS. THE POLYNOMIALS ARE SPECIFIED BY THEIR
C            COEFFICIENTS.SOME OF THE LAST COEFFICIENTS MAY TURN OUT TO
C            BE ZERO IF THE QUOTIENT IS AN EXACT POLYNOMIAL OF ORDER
C            LESS THAN L. THE REMAINDER IS NOT COMPUTED. AN EXPLAN-
C            ATION AS TO HOW THE SYMBOLIC DECK MAY BE ALTERED SO THAT
C            THE REMAINDER WILL BE COMPUTED IS GIVEN IN THE SYMBOLIC
C            DECK. THE COMPUTATION IS...
C
C                             2       3          (L-1)
C            Q(1)+Q(2)*X+Q(3)*X +Q(4)*X +...+Q(L)*X     +REMAINDER =
C
C                                            (M+1)                 N-1
C                 =DVD(1)+DVD(2)*X+...DVD(M)*X    /DVS(1)+...DVS(N)*X
C
C            WHERE X IS UNSPECIFIED SINCE ALL OPERATIONS ARE ON THE
C                     COEFFICIENTS,
C                 Q IS THE QUOTIENT VECTOR,
C                 DVD IS THE DIVIDEND VECTOR,
C                 DVS IS THE DIVISOR VECTOR.
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - IBM 709, 7090 (MAIN FRAME ONLY)
C STORAGE   - 135 REGISTERS
C SPEED     -
C AUTHOR    - J. CLAERBOUT
C
C                    ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C      AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C     CALL POLYDV(N,DVS,M,DVD,L,Q)
C
C INPUTS
C
C   N         NUMBER OF COEFFICIENTS IN DIVISOR POLYNOMIAL
C             MUST BE GRTHN=1.
C
C   DVS(I)    I=1,N     COEFFICIENTS OF DIVISOR POLYNOMIAL
C             DVS(1) MUST BE NON ZERO
C
C   M         NUMBER OF COEFFICIENTS IN DIVIDEND POLYNOMIAL
C             MUST BE GRTHN=1.
C
C   DVD(I)    I=1,M     COEFFICIENTS OF DIVIDEND POLYNOMIAL
C
C   L         NUMBER OF COEFFICIENTS IN QUOTIENT POLYNOMIAL
C             MUST BE GRTHN=1.
C
C OUTPUTS
C
C   Q(I)      I=1,L     COEFFICIENTS IN QUOTIENT POLYNOMIAL
C
C EXAMPLES
C
C 1. INPUTS  - M=1    DVD(1)=1.
C             N=2    DVS(1...2)=1.,-.5
C             L=4
C   OUTPUTS - Q(1...4)=1.,.5,.25,.125
C
C 2. INPUTS  - M=3 , DVD(1...3)= 1.,2.,1.
C             N=2 , DVS(1...2)= 1.,1.
C             L=10
```

```
C     OUTPUTS - Q(1...10)=1.,1.,0.,0.,0.,0.,0.,0.,0.,0.
C
C       THIS COULD BE REPROGRAMMED TO ALLOW EQUIVALENCE(DVD,Q), NOT ALLOW
      DIMENSION DVS(10), DVD(10), Q(10)
      NM = N-1
    5 DO 8  I=1,L
  8   Q(I) = 0.
C     MOVE THE USED PORTION OF DVD TO Q
      MML=XMINOF(M,L)
      DO 10  I=1,MML
 10   Q(I) = DVD(I)
      DO 50  I = 1,L
      Q(I) = Q  (I)/DVS(1)
      IF (I-L)30,20,30
 20   RETURN
 30   K = I
C     IF THE FOLLOWING CARD IS CHANGED TO (ISUB=NM) THEN THE REMAINDER
C      WILL BE COMPUTED AND STORED AT Q(L+1) TO Q(L+N).
      ISUB = XMINOF(NM,L-I)
      DO 40  J = 1,ISUB
      K = K+1
      Q(K)=Q(K)-Q(I)*DVS(J+1)
 40   CONTINUE
 50   CONTINUE
C     PROGRAM NEVER GETS HERE
      END
```

```
*     PRBFIT (SUBROUTINE)              2/15/63   LAST CARD IN DECK IS NO. 0186
*     LABEL
CPRBFIT
      SUBROUTINE PRBFIT(NOR,XMOM,NOUT,X,F,PHI,IANS)
C
C                   ----ABSTRACT----
C
C TITLE - PRBFIT
C        GENERATE PROBABILITY DISTRIBUTION WITH SPECIFIED MOMENTS
C
C              PRBFIT GENERATES A ZERO-MEAN DISTRIBUTION FUNCTION, F(X),
C              WHOSE HIGHER MOMENTS (2ND,3RD,...,NTH  WHERE N IS LESS
C              THAN OR EQUAL 6) ASSUME GIVEN VALUES.  F(X) HAS THE FORM
C              OF A NORMAL DISTRIBUTION TIMES A POLYNOMIAL IN X, AND
C              CONSEQUENTLY IS USEFUL FOR APPROXIMATING EMPIRICAL
C              DISTRIBUTIONS WHICH ARE ROUGHLY NORMAL IN APPEARANCE,
C              BUT FOR WHICH THE NORMAL APPROXIMATION IS INADEQUATE.
C              IT SHOULD BE NOTED THAT THE PROCEDURE CAN YIELD NEGATIVE
C              VALUES FOR THE DISTRIBUTION IN CASES WHERE THE DEVIATION
C              FROM NORMALITY IS SEVERE.
C              AN ANALYSIS OF THE PROCEDURE USED MAY BE FOUND IN
C              CRAMER, H., 1951, MATHEMATICAL METHODS OF STATISTICS,
C              PRINCETON UNIVERSITY PRESS, PRINCETON, PAGE 222.
C
C              THE FORM OF THE CALCULATION IS
C
C                                  C(3)      D  D  D(PHI(U))
C                 F(X) = PHI(U) + ----- * (--*--*---------)
C                                  1*2*3    DU DU DU
C
C                          C(4)       D  D  D  D(PHI(U))
C                       + ------- * (--*--*--*---------) +...+
C                         1*2*3*4    DU DU DU DU
C
C                        C(NOR)          D      D(PHI(U))
C                       +----------- * (--*...*---------)
C                        1*2*...*NOR    DU     DU
C
C              EVALUATED FOR A GIVEN SET OF X VALUES
C                  X=X(1),X(2),...,X(NOUT)
C              WHERE
C                       D
C                       --    DENOTES DIFFERENTIATION WITH RESPECT TO U
C                       DU
C
C                        U = X/SIG
C
C                   PHI(U) = EXP(-.5*U*U)/(SQUARE ROOT(2*PI))
C                            (I.E. NORMAL CURVE)
C
C                       PI = 3.14159265
C
C                             K     XMOM(L)
C                     C(K) = SUM ( ------- * A(K,L) )
C                            L=0      SIG
C
C                   A(K,L) = COEFFICIENT OF LTH POWER OF X IN THE KTH
C                            HERMITE POLYNOMIAL (X)
C
C                  XMOM(L) = LTH PROBABILITY MOMENT
C                            (INPUT PARAMETER VECTOR)
C
C                      SIG = SQUARE ROOT(XMOM(2))
C                            I.E. STANDARD DEVIATION
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709, 7090 (MAIN FRAME ONLY)
C STORAGE   - 366 REGISTERS
C SPEED     -
C AUTHOR    - R.J. GREENFIELD,  JAN 1963
C
C                   ----USAGE----
C
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
```

```
C       AND FORTRAN SYSTEM ROUTINES - SQRT, EXP(2, EXP
C
C FORTRAN USAGE
C     CALL PRBFIT(NOR,XMOM,NOUT,X,F,PHI,IANS)
C
C INPUTS
C
C     NOR        IS THE ORDER OF THE HIGHEST ORDER MOMENT GIVEN
C                MUST BE GRTHN= 2 AND LSTHN = 6
C
C     XMOM(I)    I=1...NOR  CONTAINS THE MOMENTS WHICH WILL BE USED TO
C                  DEVELOP THE EXPANSION.  THE FIRST MOMENT, XMOM(1),
C                  IS NOT ACTUALLY USED, BUT IS ASSUMED TO BE =0.
C                  (I.E. ZERO MEAN ASSUMPTION).
C
C     NOUT       IS THE NUMBER OF X VALUES AT WHICH THE EXPANSION WILL BE
C                  EVALUATED
C
C     X(I)       I=1...NOUT IS THE LIST OF VALUES AT WHICH THE EXPANSION
C                  WILL BE EVALUATED
C
C     PHI(I)     USED FOR STORAGE
C                MUST BE DIMENSIONED AT LEAST AS LARGE AS NOUT
C
C OUTPUTS
C
C     F(I)       I=1...NOUT  ARE THE VALUES OF THE EXPANSION FOR THE
C                  NOUT VALUES OF X, I.E. F(I) = F(X(I)) AS DEFINED
C                  IN ABSTRACT
C
C     IANS       = 0  NORMAL
C                = 1  ILLEGAL NOR
C
C
C EXAMPLES
C
C 1. (NORMAL APPROXIMATION)
C     INPUTS  - NOR = 2    XMUD(1...4) = 0.,4.,8.,10.     NOUT = 4
C                X(1...4)= 0.,5.,.8,-.8
C     OUTPUTS - F(1...4)= .39894,.017528,.36828,.36828     IANS= 0
C
C 2. INPUTS    SAME AS IN EXAMPLE 1. EXCEPT NOR= 3
C     OUTPUTS - F(1...4)= .39894,.041265,.29854,.43800     IANS= 0
C
C 3. INPUTS  - SAME AS IN EXAMPLE 1. EXCEPT NOR= 4
C     OUTPUTS - F(1...4)= .28051,.0333501,.22328,.36272    IANS= 0
C
C 4. INPUTS  - SAME AS EXAMPLE 1. EXCEPT NOR= 0
C     OUTPUTS - ERROR  IANS= 1
C
C 5. INPUTS  - SAME AS IN EXAMPLE 1. EXCEPT NOR=10
C     OUTPUTS - ERROR   IANS = 1
C
      DIMENSION A(7,7),C(7),PHI(100),XMOM(7),X(100),XMUD(7)
      DIMENSION XMU(7),F(2)
      NORDER = NOR +1
C  TEST INPUT DATA
      IF (NORDER-2) 31,31,32
 31   IANS=1
      RETURN
 32   IF(NORDER-7)  33,33,31
 33   IANS=0
      XMU(1)= 1.
      XMU(2)= 0.
      DO 50 K=2,NOR
 50   XMU(K+1)=XMOM(K)
C  SET UP A TABLE
      DO 1  J=1,7
 1    A(J,J)=1.
      A(3,1)=-1.
      A(4,2)=-3.
      A(5,1)=3.
      A(5,3)=-6.
      A(6,2)=15.
      A(6,4)=-10.
```

PRBFIT (PAGE 3)

```
      A(7,1)=-15.
      A(7,3)=45.
      A(7,5)=-15.
C  ALL SUBSCRIPTS ADVANCED BY 1
C  X(I) INPUT NORMALIZED BY CALLING PROG (ZERO MEAN)
C  XMU ARE NOT NORMALIZED BUT ARE FOR ZERO MEAN
C  SEC TO COMP C
      SIG= SQRTF(XMU(3))
      DO 51 I=1,NCUT
 51   X(I)= X(I)/SIG
      FACT=1.
      DO 5 K=1,NORDER
      C(K)=0.
      IF(K-1) 41,41,40
 40   FACT=FACT*FLOATF(K-1)
 41   DO 4  L=1,K
 4    C(K)=C(K)+(XMU(L)/(SIG**(L-1)))*A(K,L)
 5    C(K)=C(K)/FACT
C  SET UP TABLE OF PHI
      DO 6  I=1,NCUT
 6    PHI(I)=EXPF(-X(I)*X(I)*.5)*.3989423
C  COMPUTE F(I) FOR NORMAL DISTRIBUTION
      DO 7  I=1,NCUT
 7    F(I)=C(1)*PHI(I)
      IF(NORDER-4)  99,8,8
C  COMPUTES OTHER ORDER F
 8    DO 19  K=4,NORDER
      DO 12  I=1,NCUT
      HER=A(K,1)
      DO 10  L=2,K
 10   HER=HER+A(K,L)*X(I)**(L-1)
 12   F(I)=F(I)+PHI(I)*C(K)*HER
19    CONTINUE
 99   DO 98 I=1,NCUT
 98   X(I)= X(I)*SIG
      RETURN
      END
```

```
*     PROB2 (SUBROUTINE)               2/18/63   LAST CARD IN DECK IS NO. 0174
*     LABEL
CPROB2
      SUBROUTINE PROB2 (IX,LX,N,IP,P,IXHI,IANS)
C
C                     ----ABSTRACT----
C
C TITLE - PROB2
C         SECOND PROBABILITY DENSITY OF INTEGER SERIES AT GIVEN LAG.
C
C              PROB2 COMPUTES THE SECOND PROBABILITY DENSITY FOR AN
C              INTEGER SERIES BY A FREQUENCY COUNT METHOD. THE SECOND
C              PROBABILITY DENSITY, P(M,L), OF A SERIES IX(K) IS THE
C              PROBABILITY THAT X(K) = M AND X(K+N)=L, WHERE N IS THE
C              LAG. PROB2 COMPUTES THIS QUANTITY FOR A GIVEN N. THE
C              INTEGER SERIES MUST BE SCALED SUCH THAT THE LOWEST VALUE
C              OF IX(K) =1 AND THE HIGHEST VALUE IS IXHI. IXHI MUST BE
C              LESS THAN OR EQUAL TO THE DIMENSION OF THE P(I,J) MATRIX.
C              THE PROGRAM BELOW DIMENSIONS P(I,J) TO P(25,25).
C
C              PROB2 COUNTS INTO AN INTEGER MATRIX, IP(I,J), THE NUMBER
C              OF TIMES IX(K)=M AND IX(K+N)=L OVER ALL INDEX PAIRS
C              K, K+N  SUCH THAT BOTH K AND K+N LIE IN THE INCLUSIVE
C              RANGE 1 TO LX WHERE LX IS THE SERIES LENGTH.  N MAY
C              BE NEGATIVE.
C
C              THE INTEGER FREQUENCY COUNT MATRIX IS FLOATED INTO P(I,J)
C              AND NORMALIZED SUCH THAT SUM OVER I AND J OF P(I,J) IS 1.
C              THIS IS DONE BY DIVIDING EACH ELEMENT BY R, WHERE
C              R=LX-XABSF(N). P(I,J) AND IP(I,J) MAY BE EQUIVALENT IF THE
C              FREQUENCY COUNT IS NOT NEEDED. (THIS CAN BE RECONSTRUCTED
C              SINCE LX AND N ARE KNOWN.)
C
C LANGUAGE  - FORTRAN II SUBROUTINE
C EQUIPMENT - 709,7090 (MAIN FRAME ONLY)
C STORAGE   - 229 DECIMAL REGISTERS
C SPEED     -
C AUTHOR    - J.N. GALBRAITH
C
C                     ----USAGE----
C
C TRANSFER VECTOR CONTAINS ROUTINES - NONE
C       AND FORTRAN SYSTEM ROUTINES - NONE
C
C FORTRAN USAGE
C      CALL PROB2 (IX,LX,N,IP,P,IXHI,IANS)
C
C INPUTS
C
C    IX(I)      I=1,..,LX  INTEGER SERIES. IX(I) GRTHN 0, LSTHN OR = IXHI
C
C    LX         INTEGER. LENGTH OF IX SERIES. GRTHN ZERO
C
C    N          INTEGER. LAG OR SEPARATION FOR COUNT. CAN BE +,-, OR 0.
C               XABS(N) LSTHN OR = IXHI
C
C    IP(I,J)    I=1,..,IXHI,J=1,..,IXHI  SPACE FOR COMPUTATION OF
C               FREQUENCY RATIOS. MAY BE EQUIVALENT TO P(I,J). WILL
C               CONTAIN FREQUENCY RATIOS WHEN RETURN IS MADE IF NO
C               EQUIVALENCE HAS BEEN MADE.
C
C    IXHI       INTEGER. LARGEST VALUE IX TAKES ON. PROGRAM ASSUMES
C               IXHI LSTHN OR = 25. MUST BE LSTHN OR EQUAL DIMENSION OF
C               P(I,J) MATRIX.
C
C OUTPUTS
C
C    P(I,J)     I=1,..,IXHI,J=1,..,IXHI. PROBABILITY DENSITY FOR LAG OF N
C               NORMALIZED SUCH THAT  SUM OVER I AND J OF P(I,J) IS 1.
C
C    IANS       INTEGER.  ERROR INDICATOR
C               =0 NORMAL
C               =-1 ILLEGAL IX VALUE. SOME IX LSTHN 1 OR GRTHN IXHI.
C               =-2 ILLEGAL LX. LX LSTHN 1
C               =-3 ILLEGAL N.  XABSF(N) GRTHN LX.
```

```
C                =-6 ILLEGAL IXHI. IXHI GRTHN 26 OR LSTHN 1.
C                =3  JOB DONE BUT N=0 AND ONLY CONTRIBUTIONS TO P(I,J) ARE
C                    ON THE DIAGONAL.
C
C EXAMPLES
C
C 1. INPUTS  - IX(I)=0, LX=5, N=1, IXHI=5
C    OUTPUTS - IP(I,J)=0 , P(I,J)=0 , IANS=-1
C
C 2. INPUTS  - SAME AS EXAMPLE 1  EXCEPT  IX(I)=1,2,3,4,6
C    OUTPUTS - SAME AS EXAMPLE 1
C
C 3, INPUTS  - SAME AS EXAMPLE 2 EXCEPT LX=0
C    OUTPUTS - IANS=-2
C
C 4. INPUTS  - SAME AS EXAMPLE 2 EXCEPT IXHI=0
C    OUTPUTS - IANS=-6
C
C 5. INPUTS  - SAME AS EXAMPLE 4 EXCEPT IXHI=26
C    OUTPUTS - IANS=-6
C
C 6. INPUTS  - SAME AS EXAMPLE 2 EXCEPT IX(5)=5, N=-6
C    OUTPUTS - IANS=-3
C
C 7. INPUTS  - IX(I)=1,1,2,2,3,3,4,4,5,5,1,2,2,3,4,5,5,1,1,1,1,1,1,1
C              IXHI=5, LX=21, N=1
C    OUTPUTS - IANS=0
C              4  2  0  0  0                .2  .1  .0   .0   .0
C              0  2  2  0  0                .0  .1  .1   .0   .0
C     IP(I,J)= 0  0  1  2  0       P(I,J)= .0  .0  .05  .1   .0
C              0  0  0  1  2                .0  .0  .0   .05  .1
C              2  0  0  0  2                .1  .0  .0   .0   .1
C
C 8. INPUTS  - SAME AS EXAMPLE 7 EXCEPT N=-1
C    OUTPUTS - IANS=0
C
C              4  0  0  0  2                .2  .0  .0   .0   .1
C              2  2  0  0  0                .1  .1  .0   .0   .0
C     IP(I,J)= 0  2  1  0  0       P(I,J)= .0  .1  .05  .0   .0
C              0  0  2  1  0                .0  .0  .1   .05  .0
C              0  0  0  2  2                .0  .0  .0   .1   .1
C
C 9. INPUTS  - SAME AS EXAMPLE 7 EXCEPT LX=24, N=3
C    OUTPUTS - IANS=0
C              3  1  2  0  0                .15  .05  .1   .0   .0
C              0  0  1  2  1                .0   .0   .05  .1   .05
C     IP(I,J)= 0  0  0  1  2       P(I,J)= .0   .0   .0   .05  .1
C              2  0  0  0  1                .1   .0   .0   .0   .05
C              2  2  0  0  0                .1   .1   .0   .0   .0
C
C10. INPUTS  - SAME AS EXAMPLE 7 EXCEPT LX=20, N=0
C    OUTPUTS - IANS=3
C
C             6  0  0  0  0                 .3  .0  .0   .0   .0
C             0  4  0  0  0                 .0  .2  .0   .0   .0
C     IP(I,J)= 0  0  3  0  0       P(I,J)= .0  .0  .15  .0   .0
C             0  0  0  3  0                 .0  .0  .0   .15  .0
C             0  0  0  0  4                 .0  .0  .0   .0   .2
C
      DIMENSION IX(1000),IP(25,25),P(25,25)
C     CHECK LX
      IANS=-2
      IF(LX)  9999,9999,2
 2    IANS=-6
C     CHECK IXHI
      IF(IXHI)  9999,9999,3
 3    IF(IXHI-25)  4,4,9999
C     CHECK IX SERIES
 4    IANS=-1
      DO 1  I=1,LX
      IF(IX(I)) 9999,9999,11
 11   IF(IX(I)-IXHI) 1,1,9999
 1    CONTINUE
      IANS=-3
C     CHECK N
```

```
      IF(XABSF(N)-LX) 41,9999,9999
   41 IANS=0
C     CLEAR IP(I,J)
      DO 5  I=1,25
      DO 5 J=1,25
    5 IP(I,J)=0
      IF(N) 6,7,8
    6 LFRST=-N+1
      LLAST=LX
      GO TO 9
    7 IANS=3
    8 LFRST=1
      LLAST=LX-N
    9 DO 10 I=LFRST,LLAST
      J=IX(I)
      KK=I+N
      K=IX(KK)
   10 IP(J,K)=IP(J,K)+1
      L=LLAST-LFRST+1
      TOTAL=L
      DO 15  I=1,IXHI
      DO 15 J=1,IXHI
   15 P(I,J)=FLOATF(IP(I,J))/TOTAL
 9999 RETURN
      END
```

BIBLIOGRAPHY

Blackman, R. B., and Tukey, J. W., 1958, The measurement of power spectra: Dover Publications Inc., New York

Bogert, B. P., 1961, The transfer function of a short-period vertical seismograph: Bull. Seis. Soc. Am., 51:503-513.

Bullen, K. E., 1953, An Introduction to the Theory of Seismology: Cambridge University Press, Cambridge.

Claerbout, J. F., 1963, Digital Filters and Applications to Seismic Detection and Discrimination: S.M. Thesis, Mass. Inst. of Tech.

Cramer, H., 1946, Mathematical Methods of Statistics: Princeton Univ. Press.

Davenport, W. B., Jr., and Root, W. L., 1958, An Introduction to the Theory of Random Signals and Noise: McGraw-Hill Book Company, Inc., New York.

DeVorkin, D., 1963, Representation Schemes for Investigating Non-Linear Processes: Ph.D. Thesis, Mass. Inst. of Tech.

Doob, J. L., 1953, Stochastic Processes: John Wiley and Sons, New York.

Geotechnical Corporation, 1961, Personal communication.

Haq, K. E., 1954, The nature and origin of microseisms: Sc.D. Thesis, Mass. Inst. of Tech.

Kenyon, K., 1961, Microseisms and Water Waves: S.M. Thesis, Mass. Inst. of Tech.

Kolmogorov, A., 1939, Sur l'interpolation et extrapolation des suites stationnaires: C. R. Acad. Sci., V. 208, Paris, 2043-2045.

Lamb, H., 1932, Hydrodynamics: Dover Publications, New York.

Lanczos, C., 1956, Applied Analysis: Prentice Hall, Inc., New Jersey.

Lee, Y. W., 1960, Statistical Theory of Communication: John Wiley and Sons, Inc., New York.

Levinson, N., 1949, The Wiener RMS (Root Mean Square) Error Criterion in Filter Design and Prediction: Appendix B of Wiener (1949).

Longuet-Higgins, M. S., 1950, A theory of the origin of microseisms: Phil. Trans. Roy. Soc. London (A).

Longuet-Higgins, M. S., and Ursell, F., Sea waves and microseisms: Nature.

Ramirez, J. E., 1940, An experimental investigation of the nature and origin of microseisms at St. Louis, Missouri: Bull. Seis. Soc. Am.

Rand Corporation, 1955, A million random digits with 100,000 normal deviates: Glencoe, Ill., Free Press.

Robinson, E. A., 1954, Predictive Decomposition of Time Series with Applications to Seismic Exploration: Ph.D. Thesis, Mass. Inst. of Tech.

Robinson, E. A., 1962, Random Wavelets and Cybernetic Systems: No. Nine of Griffin's Statistical Momographs and Courses, Charles Griffin and Company Limited, London.

Romney, C., 1959, Amplitudes of seismic body waves from nuclear explosions. J. Geophys. Res. V. 64, 1489-1498.

Simpson, S. M., Robinson, E. A., Claerbout, J. F., Galbraith, J. N., Clark, J., 1961a, Initial studies on underground nuclear detection with seismic data prepared by a novel digitization system; Annual Report No. 1 of Contract AF 19(604)7378 at M.I.T., prepared for Geophysics Research Directorate, Air Force Cambridge Research Laboratory, Office of Aerospace Research, United States Air Force Bedford, Mass.

Simpson, S. M., Robinson, E. A., Claerbout, J. F., Galbraith, J. N., Ross, W. P., Clark, J., 1961b: Time series techniques applied to underground nuclear detection and further digitized seismic data: Scientific Report No. 2 of Contract AF 19(604)7378 continued as above.

Simpson, S. M., Robinson, E. A. Claerbout, J. F., Clark, J., Galbraith, J. N., Pan, C., Wiggins, R., 1962a, Continued numerical studies on underground nuclear detection and further digitized seismic data; Scientific Report No. 3 of Contract AF 19(604)7378 continued as above.

Simpson, S. M., Robinson, E. A., Claerbout, J. F., Clark, J., Galbraith, J. N., Greenfield, R. J., Wiggins, R. A., 1962b, Magnetic tape copies of M.I.T. Geophysics Program Set I, (Time Series Programs for the IBM 709, 7090); Scientific Report No. 4 of Contract AF 19(604)7378 continued as above.

Simpson, S. M., Jr., (Director M.I.T. Geophysical Analysis Group), 1955, Linear Operators and Seismic Noise: MIT GAG Report No. 9

Simpson, S. M., Jr., 1956, Properties, Origin and Treatment of Certain Types of Seismic Noise: MIT GAG Report No. 10a.

Stoker, J. J., 1957, Water Waves: Interscience Publishers, Inc., New York.

Vesiac Staff, 1962, Problems in Seismic Background Noise: VESIAC Advisory Report 4410-32-X, Acoustics and Seismics Laboratory, Institute of Science and Technology, The University of Michigan.

Wadsworth, G. P., and Bryan, J. G., 1960, Introduction to Probability and Random Variables: McGraw-Hill Book Company, Inc., New York

Wadsworth, G. P., Robinson, E. A., Bryan, J. G., and Hurley, P. M., 1953, Detection of Reflections on Seismic Records by Linear Operators: Geophysics, V. 18, No. 3, July 1953.

Whittle, P., 1963, Prediction and Regulation; (in press).

Wiener, N., 1949, Extrapolation, Interpolation, and Smoothing of Stationary Time Series: The Technology Press of the Mass. Inst. of Tech., and John Wiley and Sons, Inc., New York.

Wold, H., 1938, A study in the analysis of stationary time series: Uppsala. Almqvist and Wiksells.

BIOGRAPHICAL NOTE

The author was born in Philadelphia, Pennsylvania on April 26, 1936. He attended the Germantown Friends School in Philadelphia from 1941 until his graduation in 1954. He entered the Massachusetts Institute of Technology in 1954 and obtained a Bachelor of Science degree in Physics in 1958. He entered the Graduate School at M.I.T. in the Geology and Geophysics Department in 1958 and was a research assistant under Professor W. F. Brace until 1960. He then held a tuition scholarship for a year and research assistantships for two years while working for Professor S. M. Simpson, Jr on this thesis. He was married in 1960 to the former Miss Joan Blumenstiel of Alliance, Ohio.