ESD-TR-66-203

Technical Note 1966-29

A Frequency-Domain Synthesis Procedure for Multidimensional Maximum-Likelihood Processing of Seismic Arrays

J. Capon
R. J. Greenfield
R. J. Kolker

6 May 1966

Prepared for the Advanced Research Projects Agency
under Electronic Systems Division Contract AF 19(628)-5167 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Lexington, Massachusetts

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This research is a part of Project Vela Uniform, which is sponsored by the U.S. Advanced Research Projects Agency of the Department of Defense; it is supported by ARPA under Air Force Contract AF 19(628)-5167 (ARPA Order 512).

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LINCOLN LABORATORY

A FREQUENCY-DOMAIN SYNTHESIS PROCEDURE FOR MULTIDIMENSIONAL MAXIMUM-LIKELIHOOD PROCESSING OF SEISMIC ARRAYS

J. CAPON

R. J. GREENFIELD

R. J. KOLKER

Group 64

TECHNICAL NOTE 1966-29

6 MAY 1966

LEXINGTON MASSACHUSETTS

ABSTRACT

A frequency-domain synthesis method for multidimensional maximum-likelihood filtering of sampled data obtained from seismic arrays is presented. This procedure is shown to possess several advantages relative to the time-domain synthesis technique. The primary advantage is that the frequency-domain method requires approximately ten times less computer time to synthesize the filter than does the time-domain technique.

The details of a direct segment method for the spectral matrix estimation required in the frequency-domain approach are presented. In addition, the bias, variance, mean square error, limiting distribution, and other properties of the spectral estimates are discussed. The details of a Fortran IV computer program implementation of the frequency-domain method are given. The experimental results obtained by processing two events recorded at the Large Aperture Seismic Array are presented as well as a comparison of the performance of the frequency-domain method relative to the time-domain synthesis technique. It is found that the processed noise power reduction for the frequency-domain method is typically about two out of a total of 20 db worse than that of the time-domain technique.

Accepted for the Air Force
Franklin C. Hudson
Chief, Lincoln Laboratory Office

TABLE OF CONTENTS

I. INTRODUCTION

A considerable reduction of seismic noise is possible by employing multidimensional filtering for seismic arrays.[1] One of the more important approaches to seismic array processing is the maximum-likelihood method which is equivalent to the minimum-variance unbiased estimator technique.[1] This multidimensional filtering method forms a single output waveform which serves as an estimator of the unknown signal which comes from a fixed direction.

The basic assumption in our analysis of multidimensional filters is that the output, $X_k(t)$, of the k^{th} seismometer may be written as

$$X_k(t) = S(t) + N_k(t) \tag{1}$$

where $S(t)$ is the signal waveform which is assumed to be the same in each seismometer and $N_k(t)$ is the noise present in seismometer k, $k = 1, \ldots, K$. In writing Eq. (1) it is assumed that the azimuth and horizontal velocity of the event, or signal, have already been determined with sufficient accuracy to allow the signal waveforms from each seismometer to be shifted to bring them into time coincidence. In most applications, the outputs of the seismometers are given in sampled form in which case Eq. (1) becomes

$$X_{km} = S_m + N_{km} \qquad \begin{array}{l} k = 1, \ldots, K \\ m = 0, \pm 1, \pm 2, \ldots \end{array} \tag{2}$$

Only the sampled-data multidimensional filtering problem for seismic arrays will be considered.

A time-domain technique can be used to design the maximum-likelihood filter. In this method the crosscorrelation matrix of the noise is measured in what is called a fitting interval, usually three minutes long, immediately preceding the event and using this measured matrix the filter is synthesized by employing a recursive synthesis procedure.[1] The filter designed in this manner is optimum in the sense that it provides a maximum-likelihood estimate of the signal, provided the noise is a multidimensional Gaussian process. It is also optimum in the sense that the output noise power is minimized subject to the constraint that the signal be undistorted by the filter.

However, in spite of these advantages, the time domain synthesis method is subject to several severe criticisms. The most important criticism is that the method requires a large amount of computer time, typically about 30 minutes of 7094 computing time to synthesize a filter for a 25-channel subarray of seismometers. In addition, this method tends to develop a "supergain" at certain frequencies, inside the fitting interval which is not maintained outside the fitting interval where the filtering action is most important. Thus, there is a considerable effort expended in this method in achieving a large noise reduction in the fitting interval which cannot be maintained outside the fitting interval. Perhaps another way of saying this is that the technique is too sensitive to the assumption that the noise is stationary. Still another disadvantage is that the technique is at times sensitive to the assumption that the signal be identical across the array of sensors. This drawback manifests itself in the form of a precursor, when two-sided or symmetric filters are used, which precedes the event. This can be troublesome if first motion is to be preserved for use as a discrimination criterion between natural seismic events and nuclear explosions.

The purpose of this report is to present a frequency-domain synthesis procedure for symmetric multidimensional maximum-likelihood filters which does not have the disadvantages of the time-domain synthesis method just mentioned. The theory for the method has been presented previously.[1] In brief, the frequency-domain method is optimum in the same sense as the time-domain method when the length, or memory, of the filter is large. The greatest advantage of the frequency-domain method is that it requires about one order of magnitude less computing time than the time-domain technique in the synthesis of a filter for a 25-channel subarray of seismometers. In addition, the frequency-domain method is less sensitive to the assumption of noise stationarity and identical signals across the array than the time-domain technique. However, the output noise power reduction for the frequency-domain method is typically about two out of a total of 20 db worse than that of the time-domain technique. This disadvantage, however, would seem to be offset by the advantages cited previously.

We begin our discussion of the frequency-domain synthesis method by assuming, for simplicity, that the noise components have zero mean and covariance matrix

$$E\{N_{jm} N_{kn}\} = \rho_{jk}(m,n) = \rho_{kj}(n,m), \qquad \begin{matrix} 1 \le j,k \le K \\ -\nu \le m,n \le \nu \end{matrix} \tag{3}$$

where E denotes expectation, and it is assumed that the estimator, or filter, is to use

$2\nu+1$ samples extending in time from $-\nu$ to ν. It will be assumed that the noise is stationary, so that

$$\rho_{jk}(m,n) = \rho_{jk}(m-n) = \int_{-\pi}^{\pi} f_{jk}(x)\, \epsilon^{-i(m-n)x} \frac{dx}{2\pi} \tag{4}$$

and

$$f_{jk}(x) = \sum_{m=-\infty}^{\infty} \rho_{jk}(m)\, \epsilon^{imx} \qquad j,k = 1,\ldots,K \tag{5}$$

is the sampled cross power spectral density function, $x = \omega T$, T is the sampling interval, and ω is the frequency in radians/second.

The minimum-variance unbiased estimator, or, equivalently, the maximum-likelihood estimate of S_n, denoted by $\hat{S}_{\nu n}$, can be written as

$$\hat{S}_{\nu n} = \sum_{m=-\nu}^{\nu} \sum_{k=1}^{K} \theta_{km} X_{k,m+n} \tag{6}$$

where the filter weights satisfy the constraint

$$\sum_{k=1}^{K} \theta_{km} = \delta_{mn}, \qquad m = -\nu,\ldots,\nu \tag{7}$$

where

$$\begin{aligned} \delta_{mn} &= 1, && m = n \\ &= 0, && \text{otherwise} \end{aligned} \tag{8}$$

and it is assumed that a symmetric filter is to be used.

It has been shown that for symmetric filters, asymptotically optimum filter weights are given by, cf. reference 1, p. 16,

$$\theta_{km} = \frac{1}{2\nu} [A_k(0) + (-1)^m A_k(\pi) + 2 \sum_{n=1}^{\nu-1} \operatorname{Re} A_k(n\frac{\pi}{\nu}) \cos mn \frac{\pi}{\nu} +$$

$$\operatorname{Im} A_k (n\frac{\pi}{\nu}) \sin mn \frac{\pi}{\nu}], \qquad \begin{array}{l} k = 1,\ldots, K \\ m = -\nu,\ldots,\nu \end{array} \tag{9}$$

where

$$A_k(x) = \frac{\sum_{j=1}^{K} q_{kj}(x)}{\sum_{j,k=1}^{K} q_{jk}(x)}, \qquad k = 1,\ldots,K, \tag{10}$$

and $\{q_{jk}(x)\}$ is the inverse of the spectral matrix $\{f_{jk}(x)\}$, $j,k = 1,\ldots,K$. If ν is large, the spectral density of $(\hat{S}_{\nu n} - S_n)$ is given approximately by

$$\Lambda(x) = [\sum_{j,k=1}^{K} q_{jk}(x)]^{-1} \tag{11}$$

We note the following symmetry properties

$$q_{jk}(x) = q^*_{kj}(x),$$

since

$$f_{jk}(x) = f^*_{kj}(x),$$

and

$$q_{jk}(x) = q^*_{jk}(-x),$$

since

$$f_{jk}(x) = f^*_{jk}(-x) .$$

In addition, the constraint equations are satisfied since

$$\sum_{k=1}^{K} \theta_{km} = \frac{1}{2\nu} [(-1)^m + 2(\frac{1}{2} + \sum_{n=1}^{\nu-1} \cos mn \frac{\pi}{\nu})]$$

$$= \frac{1}{2\nu} [(-1)^m + \frac{\sin (\nu - \frac{1}{2}) m \frac{\pi}{\nu}}{\sin \frac{1}{2} m \frac{\pi}{\nu}}]$$

$$= \frac{1}{2\nu} [1 + 2\nu - 1] = 1 , \qquad m = 0 ,$$

$$= \frac{1}{2\nu} [(-1)^m - \frac{\sin (\frac{1}{2} m \frac{\pi}{\nu} - m\pi)}{\sin \frac{1}{2} m \frac{\pi}{\nu}}]$$

$$= \frac{1}{2\nu} [(-1)^m - (-1)^m] = 0, \qquad m \neq 0 ,$$

where we have used the identity

$$\frac{1}{2} + \sum_{k=1}^{n} \cos ku = \frac{1}{2} \frac{\sin (n + \frac{1}{2}) u}{\sin \frac{1}{2} u}$$

In essence, the frequency-domain synthesis procedure operates as follows. The spectral matrices $\{f_{jk}(x)\}$ are inverted at $\nu + 1$ points in frequency, namely $n \frac{\pi}{\nu}$, $n = 0, 1, \ldots, \nu$, to yield the matrices $\{q_{jk}(n \frac{\pi}{\nu})\}$. The filter frequency functions $A_k(n \frac{\pi}{\nu})$

are then obtained according to Eq. (10). At this point the filter frequency functions have been sampled and are thus specified at only $\nu + 1$ frequencies. In order to obtain the filter functions for all $-\pi \le x \le \pi$, $\dfrac{\sin \frac{N}{2} x}{\sin \frac{1}{2} x}$ — type sampling functions are used as follows

$$A_k(x) = \frac{1}{2\nu} \sum_{n=-\nu+1}^{\nu} A_k\left(n \frac{\pi}{\nu}\right) \frac{\sin \nu \left(x - \frac{n\pi}{\nu}\right)}{\sin \frac{1}{2}\left(x - \frac{n\pi}{\nu}\right)}, \qquad k = 1, \ldots, K .$$

It is easily seen that the Fourier transform of these frequency filter functions lead to the filter coefficients given in Eq. (9). Thus, the first step in the frequency-domain synthesis procedure must be to estimate the spectral matrices of the noise $\{f_{jk}(n \frac{\pi}{\nu})\}$, $n = 0, \ldots, \nu$ inside a fitting interval known to contain only noise. The design of the spectral matrix estimation procedure will be discussed in the next section.

II. THE SPECTRAL MATRIX ESTIMATION PROCEDURE

There is a large amount of literature available on power spectral density estimation techniques. A book by Grenander and Rosenblatt[2] treats many of the theoretical problems encountered in spectral estimation. Two books which are motivated more by the practical considerations in spectral estimation are due to Blackman and Tukey[3] and Blackman.[4] These contributions deal mainly with the estimation of the power spectrum of a single random process. A discussion of the problems involved in the estimation of the spectral matrix of a multidimensional random process has been given by Goodman[5] and Rosenblatt.[6] Thus, there is available a large number of techniques which can be used to estimate the spectra of random processes.

It is possible to divide the spectral estimation procedures into two broad categories, namely direct and indirect methods. In the direct method the data are transformed immediately into the frequency domain and then the spectrum is measured using the transformed data. The indirect method first estimates the correlation function of the data and transforms this to obtain an estimate of the spectrum. This latter method has been discussed extensively by Blackman and Tukey.[3] Some of the criteria which have been used to judge the merits of a spectral estimation procedure are the bias and variance of the estimate, which will be discussed in detail subsequently, and the amount of computer time required to obtain the estimate. In the one-dimensional case it is largely a matter of taste as to which method one might use, as both the direct and indirect methods yield estimates with roughly the same bias and variance and the computation time is comparable for both methods. However, in the multidimensional case it has been recognized that much less computation time is required by the direct method than the indirect method, and both methods can be made to yield estimates with approximately the same bias and variance, cf. e.g. Jones.[7] In the present application data from the Large Aperture Seismic Array is to be filtered in which there are 25 sensors in each of the 21 subarrays. Thus, the spectral matrix of a 25-dimensional random process is to be estimated, since a single subarray is processed at a time, and in order to keep the computation time within reasonable bounds a direct method of spectral matrix estimation is necessary.

The present method of spectral matrix estimation may be termed a direct segment method. The number of data points in each channel which is to be used in the estimation, namely L, is divided into M segments of $N = 2\nu + 1$ data points. It should be noted that $2\nu + 1$ is equal to the number of filter points of Eq. (9). The data in each segment and each channel are transformed into the frequency domain and these transforms are used to obtain an estimate of the cross-spectra in the segment. The stability of the estimate is then increased by averaging over the M segments. We now describe the method in some detail.

The transform of the noise data in the n^{th} segment, j^{th} channel, and at the ℓ^{th} frequency is

$$S_{jn}(\ell) = (N)^{-1/2} \sum_{m=1}^{N} w_m N_{j,m+(n-1)N} \, \epsilon^{im\ell(2\pi/N-1)}, \qquad (12)$$

$$j = 1, \ldots, K,$$
$$\ell = 0, \ldots, \frac{N-1}{2},$$
$$n = 1, \ldots, M,$$

where w_m, $m = 1, \ldots, N$ are the coefficients of the weighting function. Let

$$\lambda(\ell) = \ell \frac{2\pi}{N-1}, \qquad \ell = 0, \ldots, \frac{N-1}{2}, \tag{13}$$

so that

$$S_{jn}(\lambda) = (N)^{-1/2} \sum_{m=1}^{N} w_m N_{j,m+(n-1)N} \epsilon^{im\lambda} \tag{14}$$

As an estimate for $f_{jk}(\lambda)$ we take

$$\hat{f}_{jk}(\lambda) = \frac{1}{M} \sum_{n=1}^{M} S_{jn}(\lambda) S^*_{kn}(\lambda), \qquad j,k = 1, \ldots, K. \tag{15}$$

The mean value of this estimate is

$$E\{\hat{f}_{jk}(\lambda)\} = E \frac{1}{M} \sum_{n=1}^{M} S_{jn}(\lambda) S^*_{kn}(\lambda)$$

$$= E \frac{1}{MN} \sum_{n=1}^{M} \sum_{m,m'=1}^{N} w_m w_{m'}$$

$$N_{j,m+(n-1)N} N_{k,m'+(n-1)N} \epsilon^{i(m-m')\lambda}$$

$$= \frac{1}{N} \sum_{m,m'=1}^{N} w_m w_{m'} \rho_{jk}(m-m') \epsilon^{i(m-m')\lambda}$$

$$= \int_{-\pi}^{\pi} f_{jk}(x) \, | W_N(x-\lambda) |^2 \frac{dx}{2\pi}, \tag{16}$$

where the frequency window $W_N(x)$ is defined as

$$W_N(x) = (N)^{-1/2} \sum_{m=1}^{N} w_m \epsilon^{-imx} , \tag{17}$$

and thus

$$w_m = N^{1/2} \int_{-\pi}^{\pi} W_N(x) \epsilon^{imx} \frac{dx}{2\pi} . \tag{18}$$

As usual, we require that $W_N(x)$ be a reasonably good window in the sense that $|W_N(x)|^2$ approaches a delta function in such a manner that

$$\int_{-\pi}^{\pi} |W_N(x)|^2 \frac{dx}{2\pi} = 1 . \tag{19}$$

In this case $\hat{f}_{jk}(\lambda)$ is a reasonably good estimate for $f_{jk}(\lambda)$.

Since $f_{jk}(\lambda)$ is, in general, complex it is convenient to define

$$f_{jk}(\lambda) = c_{jk}(\lambda) + iq_{jk}(\lambda) \qquad j,k = 1,\ldots,K , \tag{20}$$

where $c_{jk}(\lambda)$, $q_{jk}(\lambda)$ are real-valued functions and are known as the cospectrum and quadrature spectrum, respectively. Similarly,

$$\hat{f}_{jk}(\lambda) = \hat{c}_{jk}(\lambda) + i\,\hat{q}_{jk}(\lambda) . \tag{21}$$

It thus follows that $\hat{c}_{jk}(\lambda)$, $\hat{q}_{jk}(\lambda)$, are reasonable estimates for $c_{jk}(\lambda)$, $q_{jk}(\lambda)$, respectively

The mean-square value of $\hat{c}_{jk}(\lambda)$ is

$$E\{\hat{c}_{jk}(\lambda)\}^2 = E\,\frac{1}{M^2N}\sum_{n,n'=1}^{M}\ \sum_{\substack{m,m',m'',\\ m'''=1}}^{N} w_m\, w_{m'}\, w_{m''}\, w_{m'''}$$

$$N_{j,m+(n-1)N}\,N_{k,m'+(n-1)N}\,N_{j,m''+(n'-1)N}\,N_{k,m'''+(n'-1)N}\cos(m-m')\lambda\cos(m''-m''')\lambda. \tag{22}$$

In order to proceed with the analysis, we must at this point introduce the assumption that $\{N_{km}\}$ is a multidimensional Gaussian process, so that

$$E\{N_{j,m+(n-1)N}\,N_{k,m'+(n-1)N}\,N_{j,m''+(n'-1)N}\,N_{k,m'''+(n'-1)N}\} =$$

$$\rho_{jk}(m-m')\ \rho_{jk}(m''-m''') + \rho_{jj}[m-m'' + (n-n')N]\cdot$$

$$\rho_{kk}(m'-m''' + (n-n')N) + \rho_{jk}[m-m''' + (n-n')N]\cdot$$

$$\rho_{kj}[m'-m'' + (n-n')N]\ . \tag{23}$$

Using the above result in Eq. (22), we obtain, after some manipulations,

$$\mathrm{VAR}\{\hat{c}_{jk}(\lambda)\} = \frac{1}{2}\,\mathrm{Re}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\{f_{jj}(x)\,f_{kk}(x') +$$

$$f_{jk}(x)\,f_{jk}(x')\}\ \left(\frac{\sin\frac{M}{2}N(x-x')}{M\sin\frac{N}{2}(x-x')}\right)^2\ [\,|W_N(x-\lambda)\,W_N(x'+\lambda)|^2 +$$

$$W_N(x-\lambda)\,W_N^*(x+\lambda)\,W_N(x'+\lambda)\,W_N^*(x'-\lambda)]\ \frac{dx}{2\pi}\ \frac{dx'}{2\pi}\ . \tag{24}$$

At this point it is necessary to place some regularity conditions on $W_N(x)$ which guarantee that $|W_N(x)|^2$ behave like a delta function at $x = 0$, as $N \to \infty$, i.e., in addition to Eq. (19) we require,

$$\int_{|x|<\epsilon} |W_N(x)|^2 \frac{dx}{2\pi} \to 1 \ , \tag{25}$$

for any ϵ, as $N \to \infty$,

$$|W_N(x)| \to 0 \ , \tag{26}$$

uniformly when $|x| \geq \epsilon$, as $N \to \infty$, and

$$\underset{|x_o|<\frac{A}{N}}{\mathrm{MAX}} \left| \frac{\int_{-\pi}^{\pi} |W_N(x)|^2 \, |W_N(x+x_o)|^2 \frac{dx}{2\pi}}{\int_{-\pi}^{\pi} |W_N(x)|^4 \frac{dx}{2\pi}} - 1 \right| \to 0 \ , \tag{27}$$

as $N \to \infty$, for any $A > 0$. Under these conditions it may be shown, in a manner similar to that given by Grenander and Rosenblatt[2], pp. 137-145, that Eq. (24) becomes, as M, $N \to \infty$,

$$\begin{aligned} \mathrm{VAR}\{\hat{c}_{jk}(\lambda)\} &\cong \frac{1}{2MN} \int_{-\pi}^{\pi} \{f_{jj}(x)\, f_{kk}(x) + c_{jk}^2(x) - q_{jk}^2(x)\} \cdot \\ &\quad |W_N(x-\lambda)|^4 \frac{dx}{2\pi} \ , \quad \lambda \neq 0, \pi \\ &\cong \frac{1}{MN} \int_{-\pi}^{\pi} \{f_{jj}(x)\, f_{kk}(x) + c_{jk}^2(x) - q_{jk}^2(x)\} \cdot \\ &\quad |W_N(x-\lambda)|^4 \frac{dx}{2\pi} \ , \quad \lambda = 0, \pi \ . \end{aligned} \tag{28}$$

Similarly,

$$\mathrm{VAR}\,\{\hat{q}_{jk}(\lambda)\} \cong \frac{1}{2MN} \int_{-\pi}^{\pi} \{f_{jj}(x)\, f_{kk}(x) + q_{jk}^2(x) - c_{jk}^2(x)\} \cdot$$

$$|W_N(x-\lambda)|^4 \;\frac{dx}{2\pi}\,, \quad \lambda \neq 0,\, \pi$$

$$= 0, \quad \lambda = 0,\, \pi\,, \tag{29}$$

since $q_{jk}(0) = \hat{q}_{jk}(0) = 0$, and $q_{jk}(\pi) = \hat{q}_{jk}(\pi) = 0$. If $f_{jj}(x)$, $f_{kk}(x)$, $c_{jk}(x)$, $q_{jk}(x)$, are reasonably constant in the vicinity of $x = \lambda$, then (28) and (29) can be rewritten as

$$\mathrm{VAR}\,\{\hat{c}_{jk}(\lambda)\} \cong \frac{1}{2MN} \;\{f_{jj}(\lambda)\, f_{kk}(\lambda) + c_{jk}^2(\lambda) - q_{jk}^2(\lambda)\} \cdot$$

$$\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \;\frac{dx}{2\pi}\,, \quad \lambda \neq 0,\, \pi$$

$$\cong \frac{1}{MN} \;\{f_{jj}(\lambda)\, f_{kk}(\lambda) + c_{jk}^2(\lambda) - q_{jk}^2(\lambda)\} \cdot$$

$$\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \;\frac{dx}{2\pi}\,, \quad \lambda = 0,\, \pi \tag{30}$$

$$\mathrm{VAR}\,\{\hat{q}_{jk}(\lambda)\} \cong \frac{1}{2MN} \;\{f_{jj}(\lambda)\, f_{kk}(\lambda) + q_{jk}^2(\lambda) - c_{jk}^2(\lambda)\} \cdot$$

$$\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \;\frac{dx}{2\pi}\,, \quad \lambda \neq 0,\, \pi$$

$$= 0, \quad \lambda = 0,\, \pi\,. \tag{31}$$

In the important case of uniform weighting, $w_m = 1$, $m = 1, \ldots, N$, and

$$|W_N(x)|^2 = \frac{1}{N} \left| \sum_{m=1}^{N} \epsilon^{-imx} \right|^2$$

$$= \frac{1}{N} \left| \frac{\sin \frac{N}{2} x}{\sin \frac{1}{2} x} \right|^2 \tag{32}$$

It is easily seen that this weighting function satisfies the conditions in Eqs. (19), (25), (26), (27). We note that

$$|W_N(x)|^2 = \frac{1}{N} \sum_{m, m'=1}^{N} \epsilon^{-i(m-m')x}$$

$$= \sum_{m=-N}^{N} \left(1 - \frac{|m|}{N}\right) \epsilon^{-imx} \tag{33}$$

so that

$$\int_{-\pi}^{\pi} |W_N(x)|^4 \frac{dx}{2\pi} = \sum_{m=-N}^{N} \left(1 - \frac{|m|}{N}\right)^2$$

$$= 2N + 1 - \frac{4}{N} \frac{N(N+1)}{2} + \frac{2}{N^2} \left(\frac{N}{6}\right) (2N^2 + 3N + 1)$$

$$= \frac{2N^2 + 1}{3N} \cong \frac{2N}{3} \tag{34}$$

Thus, using (34) in (30), (31),

$$\mathrm{VAR}\,\{\hat{c}_{jk}(\lambda)\} \cong \frac{1}{3M}\,\{f_{jj}(\lambda)\,f_{kk}(\lambda) + c^2_{jk}(\lambda) - q^2_{jk}(\lambda)\} \qquad \lambda \neq 0,\ \pi$$

$$\cong \frac{2}{3M}\,\{f_{jj}(\lambda)\,f_{kk}(\lambda) + c^2_{jk}(\lambda) - q^2_{jk}(\lambda)\}\ , \qquad \lambda = 0,\ \pi \tag{35}$$

$$\mathrm{VAR}\,\{\hat{q}_{jk}(\lambda)\} \cong \frac{1}{3M}\,\{f_{jj}(\lambda)\,f_{kk}(\lambda) + q^2_{jk}(\lambda) - c^2_{jk}(\lambda)\} \qquad \lambda \neq 0,\ \pi$$

$$= 0, \quad \lambda = 0,\ \pi\ . \tag{36}$$

It follows from (35), (36) that $\hat{c}_{jk}(\lambda)$ and $\hat{q}_{jk}(\lambda)$ are consistent estimates for $c_{jk}(\lambda)$, $q_{jk}(\lambda)$, respectively, and that the variances of these estimators go to zero as $M \to \infty$.

III. BIAS AND MEAN SQUARE ERROR OF SPECTRAL ESTIMATES

The bias of the estimate of the cospectrum is defined as

$$b_{Njk}(\lambda) = E\{\hat{c}_{jk}(\lambda)\} - c_{jk}(\lambda)\,, \tag{37}$$

and provides an indication of the error between the mean value of the estimate and the true value, at each frequency. We may evaluate the bias for the uniform weighting function by using Eqs. (16) and (21)

$$b_{Njk}(\lambda) = \frac{1}{N}\sum_{m,m'=1}^{N} \rho_{jk}(m-m')\cos(m-m')\lambda - \sum_{m=-\infty}^{\infty} \rho_{jk}(m)\cos m\lambda$$

$$= \sum_{m=-N+1}^{N-1} \left(1-\frac{|m|}{N}\right) \rho_{jk}(m)\cos m\lambda - \sum_{m=-\infty}^{\infty} \rho_{jk}(m)\cos m\lambda$$

$$\cong -\sum_{m=-N+1}^{N-1} \frac{|m|}{N} \rho_{jk}(m)\cos m\lambda$$

$$= -\,2\sum_{m=0}^{N-1} \frac{|m|}{N} \rho_{jk}(m)\cos m\lambda$$

$$= \frac{2}{N}\sum_{m=0}^{N-1} \cos m\lambda \int_{-\pi}^{\pi} c'_{jk}(x)\sin mx\,\frac{dx}{2\pi}$$

$$= \frac{2}{N}\sum_{m=0}^{N-1} \int_{-\pi}^{\pi} \frac{\sin m(x-\lambda)+\sin m(x+\lambda)}{2}\, c'_{jk}(x)\,\frac{dx}{2\pi}$$

$$= \frac{2}{N}\int_{-\pi}^{\pi} \frac{1}{2}\,\mathrm{Im}\left\{\frac{1-\epsilon^{im(x-\lambda)}}{1-\epsilon^{i(x-\lambda)}} + \frac{1-\epsilon^{im(x+\lambda)}}{1-\epsilon^{i(x+\lambda)}}\right\} c'_{jk}(x)\,\frac{dx}{2\pi}$$

$$= \frac{2}{N} \int_{-\pi}^{\pi} \left\{ \frac{\sin(x-\lambda) - \sin m(x-\lambda) + \sin(m-1)(x-\lambda)}{4[1-\cos(x-\lambda)]} + \right.$$

$$\left. \frac{\sin(x+\lambda) - \sin m(x+\lambda) + \sin(m-1)(x+\lambda)}{4[1-\cos(x+\lambda)]} \right\} c'_{jk}(x) \frac{dx}{2\pi}$$

$$\cong \frac{1}{N} P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} c'_{jk}(x) \frac{dx}{2\pi}, \tag{38}$$

where P denotes that the principal part of the last integral is to be taken. Similarly,

$$b'_{Njk}(\lambda) = E\{\hat{q}_{jk}(\lambda)\} - q_{jk}(\lambda)$$

$$\cong \frac{1}{N} P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} q'_{jk}(x) \frac{dx}{2\pi}. \tag{39}$$

Thus, the bias of the estimators of the cospectrum and quadrature spectrum decrease to zero at the rate of N^{-1}.

The mean square error of the cospectral estimate is defined as

$$E\{\hat{c}_{jk}(\lambda) - c_{jk}(\lambda)\}^2 = b^2_{Njk}(\lambda) + VAR\{\hat{c}_{jk}(\lambda)\}, \tag{40}$$

which may be evaluated for the uniform weighting function by using (35), (38)

$$\cong \frac{1}{3M} [f_{jj}(\lambda) f_{kk}(\lambda) + c^2_{jk}(\lambda) - q^2_{jk}(\lambda)] + [\frac{1}{N} P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} c'_{jk}(x) \frac{dx}{2\pi}]^2, \; \lambda \neq 0, \pi. \tag{41}$$

We assume that M is of the form $M = cL^{\alpha}$, where c is a constant, and it will be recalled that L is the total number of data points used in the spectral matrix estimation and is equal to MN. We will determine the c and α which ensure that the mean square error is of smallest order as M, $N \to \infty$. It is easy to see from (41) that $\alpha = 2/3$ and

$$c = \left\{ \frac{f_{jj}(\lambda) f_{kk}(\lambda) + c_{jk}^2(\lambda) - q_{jk}^2(\lambda)}{6 \left[P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} c'_{jk}(x) \frac{dx}{2\pi} \right]^2} \right\}^{1/3}, \qquad \lambda \neq 0, \pi,$$

$$= \left\{ \frac{f_{jj}(\lambda) f_{kk}(\lambda) + c_{jk}^2(\lambda) - q_{jk}^2(\lambda)}{3 \left[P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} c'_{jk}(x) \frac{dx}{2\pi} \right]^2} \right\}^{1/3}, \qquad \lambda = 0, \pi. \tag{42}$$

Similarly, for the quadrature spectral estimate $\alpha = 2/3$ and

$$c = \left\{ \frac{f_{jj}(\lambda) f_{kk}(\lambda) + q_{jk}^2(\lambda) - c_{jk}^2(\lambda)}{6 \left[P \int_{-\pi}^{\pi} \frac{\sin(x-\lambda)}{1-\cos(x-\lambda)} q'_{jk}(x) \frac{dx}{2\pi} \right]^2} \right\}^{1/3}, \quad \lambda \neq 0, \pi. \tag{43}$$

Thus, the mean square error approaches zero at the rate $L^{-2/3}$ and $M = cL^{2/3}$, $N = c^{-1} L^{1/3}$. The above formulas are not too useful in designing the spectral matrix estimation procedure since the spectra are, of course, unknown. In addition, as far as designing a multidimensional maximum-likelihood filter is concerned, the mean square error may, or may not, be the important criterion. The merits of a spectral matrix estimation procedure must be determined by using digital computer experimentation.

However, the above results are useful in the sense that they indicate that, as L is increased, M should be increased at a faster rate than N if the mean square error of the estimate is to be minimized.

IV. LIMITING DISTRIBUTION OF SPECTRAL ESTIMATES

If the regularity conditions for $W_N(x)$ given in (19), (25), (26), (27) are satisfied, and if we assume that $\{N_{jm}\}$ is a multidimensional Gaussian process in order to avoid some complicated regularity conditions which make the following result valid for non-Gaussian processes, then a straightforward extension of a limit theorem of Rosenblatt[6] shows that the $\hat{c}_{jk}(\lambda)$, $\hat{q}_{jk}(\lambda)$, $j,k = 1,\ldots,K$ are jointly asymptotically normally distribut[e]
with variances and covariances given by

$$\lim_{M,N\to\infty} (2MN)\left\{\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \frac{dx}{2\pi} \int_{-\pi}^{\pi} |W_N(x-\mu)|^4 \frac{dx}{2\pi}\right\}^{-1/2} .$$

$$\mathrm{COV}\{\hat{c}_{\alpha\beta}(\lambda), \hat{c}_{\gamma\delta}(\mu)\} = \delta_{\lambda\mu} \,\mathrm{Re}\,[f^*_{\alpha\gamma}(\lambda) f_{\beta\delta}(\lambda) + f^*_{\alpha\delta}(\lambda) f_{\beta\gamma}(\lambda)], \quad \lambda \neq 0, \pi$$

$$= 2\delta_{\lambda\mu}[f_{\alpha\gamma}(\lambda) f_{\beta\delta}(\lambda) + f_{\alpha\delta}(\lambda) f_{\beta\gamma}(\lambda)], \quad \lambda = 0, \pi, \tag{44}$$

$$\lim_{M,N\to\infty} (2MN)\left\{\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \frac{dx}{2\pi} \int_{-\pi}^{\pi} |W_N(x-\mu)|^4 \frac{dx}{2\pi}\right\}^{-1/2} \mathrm{COV}\{\hat{c}_{\alpha\beta}(\lambda), \hat{q}_{\gamma\delta}(\lambda)\}$$

$$= \delta_{\lambda\mu} \,\mathrm{Im}\,[f^*_{\alpha\gamma}(\lambda) f_{\beta\delta}(\lambda) - f^*_{\alpha\delta}(\lambda) f_{\beta\gamma}(\lambda)], \quad \lambda \neq 0, \pi$$

$$= 0, \quad \lambda = 0, \pi \tag{45}$$

$$\lim_{M,N\to\infty} (2MN)\left\{\int_{-\pi}^{\pi} |W_N(x-\lambda)|^4 \frac{dx}{2\pi} \int_{-\pi}^{\pi} |W_N(x-\mu)|^4 \frac{dx}{2\pi}\right\}^{-1/2} \mathrm{COV}\,\{\hat{q}_{\alpha\beta}(\lambda),\ \hat{q}_{\gamma\delta}(\lambda)\}$$

$$= \delta_{\lambda\mu}\,\mathrm{Re}\,[f_{\alpha\gamma}(\lambda)\, f^*_{\beta\delta}(\lambda) - f_{\alpha\delta}(\lambda)\, f^*_{\beta\gamma}(\lambda)]\ , \qquad \lambda \neq 0,\ \pi$$

$$= 0\ , \qquad \lambda = 0,\ \pi\ . \tag{46}$$

It is possible to use these formulas to establish asymptotic confidence intervals for the spectral estimates. A different approximation for the asymptotic distribution of the spectral estimates has been given by Goodman[5] in terms of the complex Wishart distribution.

V. NONNEGATIVE-DEFINITENESS AND INVERSION OF ESTIMATED SPECTRAL MATRICES

We obtain from Eqs. (14) and (15) that

$$\sum_{j,k=1}^{K} a_j a_k^* \hat{f}_{jk}(x) = \frac{1}{L} \sum_{n=1}^{M} \left| \sum_{j=1}^{K} \sum_{m=1}^{N} w_m \, a_j \, N_{j,m+(n-1)N} \, \epsilon^{imx} \right|^2 \geq 0, \quad -\pi \leq x \leq \pi, \tag{47}$$

so that the spectral matrix is nonnegative-definite at all frequencies. This is a highly desirable property since inverses of spectral matrices must be computed in the frequency-domain synthesis procedure. If the spectral matrix is singular at a particular frequency then, physically speaking, this means that infinite suppression of the noise at that frequency is possible. As a practical matter, it is not possible to achieve infinite suppression, so that the spectral matrix will not be singular. This conclusion has been borne out by computer experimentation, as no singular spectral matrices have ever been encountered.

In general, the estimated spectral matrix will be nonnegative-definite and Hermitian. Let us denote such a matrix by $C = A + iB$, where A, B are real matrices. It is easily seen that A is symmetric, B is skew symmetric and that neither A or B need be nonnegative-definite if C is nonnegative-definite. Let the inverse of C be denoted by $D + iE$. It is easily seen that

$$\left[\begin{array}{c|c} D & E \\ \hline -E & D \end{array}\right] = \left[\begin{array}{c|c} A & B \\ \hline -B & A \end{array}\right]^{-1} \tag{48}$$

Thus, the inverse of the matrix C may be obtained by inverting the matrix $\left[\begin{array}{c|c} A & B \\ \hline -B & A \end{array}\right]$.
The advantage of performing the matrix inversion in this manner is that this matrix is invertible if and only if the matrix C is invertible, cf. Goodman.[8] It is for this reason that this method of inverting the estimated spectral matrices is used in the computer program implementation of the frequency-domain synthesis procedure.

VI. COMPARISON OF DIRECT SEGMENT METHOD WITH OTHER METHODS OF SPECTRAL MATRIX ESTIMATION

The amount of computer time required to estimate the crosscorrelation matrix for N lags is given approximately by

$$T_c = L K^2 N (\mu + \alpha) \text{ seconds} , \tag{49}$$

where μ, α are the multiply and addition times, respectively, in seconds. It is seen from Section II that the amount of computer time required to estimate the spectral matrix at $(N + 1)/2$ frequencies is approximately

$$T_s = L K (K + N) (\mu + \alpha) \text{ seconds} , \tag{50}$$

and

$$\frac{T_c}{T_s} = \frac{K N}{K + N} . \tag{51}$$

If K and N are large, the ratio T_c/T_s can be very large. If $K = 25$, $N = 21$, then $T_c/T_s = 11.4$. Thus, we see that the amount of computer time required to estimate the spectral matrix is inherently much smaller than that required to estimate the crosscorrelation matrix when K and N are large. An even greater saving is possible by using the Cooley-Tukey[9] algorithm to transform the data into the frequency domain. In this case, the amount of computer time becomes

$$T_s' = L K (K + F_N) (\mu + \alpha) \text{ seconds}, \tag{52}$$

and

$$\frac{T_c}{T'_s} = \frac{K\,N}{K + F_N} \quad , \tag{53}$$

where F_N is the sum of the factors of $N - 1$. If $N - 1 = 20 = 5 \cdot 2 \cdot 2$, then $F_N = 2 + 2 + 5 = 9$ and $T_c/T'_s = 15.5$.

If an indirect spectral matrix estimation procedure is used in which the estimates of the crosscorrelation function are transformed into the frequency domain, the amount of computer time becomes

$$\begin{aligned} T_I &= (L\,K^2\,N + K^2\,N^2)\,(\mu + \alpha) \\ &\cong L\,K^2\,N\,(\mu + \alpha) \text{ seconds} \\ &= T_c, \quad L \gg N\,. \end{aligned} \tag{54}$$

It has been assumed that the Cooley-Tukey algorithm is not employed since the saving would be incurred in the second term above which is already negligible compared to the first term. Thus, T_I/T_s is quite large and T_I/T'_s is even larger. This disadvantage makes the indirect method undesirable for spectral matrix estimation.

The amount of computer time required to synthesize the maximum-likelihood filter in the time domain is $2.5\,N^2\,K^3\,(\mu + \alpha)$ seconds and in the frequency domain is $2N\,K^3$, where it is assumed that $8\,K^3$ operations are required to invert a matrix with complex elements, according to the procedure given in Section V, cf. reference 1, pp. 18-19. Thus, the total computing time required in the time domain is

$$T_T = (L K^2 N + 2.5 N^2 K^3)(\mu + \alpha) \text{ seconds} \tag{55}$$

and in the frequency domain is

$$T_F = (L K (K + N) + 2 N K^3)(\mu + \alpha) \quad , \tag{56}$$

assuming the Cooley-Tukey algorithm is not used. Thus

$$\frac{T_T}{T_F} = \frac{L K N + 2.5 N^2 K^2}{L (K + N) + 2 N K^2} \quad , \tag{57}$$

which, if L is 1800, corresponding to 3 minutes of noise, 20 data points per second and every other data point skipped, and K = 25, N = 21, becomes $T_T/T_F = 15$. In practice this large saving is not achieved, of course, due to indexing operations and tape reading which are common to both methods, but savings of approximately a factor of ten have been realized.

VII. COMPUTATION OF INPUT AND OUTPUT NOISE POWER INSIDE THE FITTING INTERVAL

The input noise power at the ℓ^{th} frequency, inside the fitting interval, is obtained from the spectral estimation program as

$$P_{IN}(\ell) = \frac{1}{(N-1)K} \sum_{j=1}^{NS} \hat{f}_{jj}(\ell) , \tag{58}$$

and the total noise power is

$$P_{IN} = 2 \sum_{\ell=1}^{\frac{N-3}{2}} P_{IN}(\ell) + P_{IN}(0) + P_{IN}\left(\frac{N-1}{2}\right) . \tag{59}$$

The justification for this definition of $P_{IN}(\ell)$ is apparent and the justification for defining P_{IN} in Eq. (59) is that P_{IN} may be written as, after some manipulation,

$$P_{IN} = \frac{1}{L\,K} \sum_{j=1}^{K} \sum_{m=1}^{L} N_{jm}^2 , \tag{60}$$

which is numerically equal to the average input noise power.

Let us consider the power measurement which is made by summing the squares of the processed noise samples at the midpoints of the segments used in the spectral matrix measurement, i.e.,

$$P_{OUT} = \frac{1}{M} \sum_{n=1}^{M} \left(\sum_{j=1}^{NS} \sum_{m=-\left(\frac{N-1}{2}\right)}^{\frac{N-1}{2}} \theta_{jm} N_{j,m+(n-1)N} \right)^2 . \tag{61}$$

We may show, after some manipulation, that

$$P_{OUT} = \frac{N}{N-1}\,\frac{1}{N-1}\left\{\sum_{\ell=-\left(\frac{N-3}{2}\right)}^{\frac{N-1}{2}}\left[\sum_{j,k=1}^{K}\hat{q}_{jk}(\ell)\right]^{-1}\right\}, \tag{62}$$

where $\{\hat{q}_{jk}(\ell)\}$ is the inverse of the estimated spectral matrix $\{\hat{f}_{jk}(\ell)\}$. The frequency-domain computer program uses the inverse spectral matrices to compute P_{OUT} as indicated in Eq. (62). It should be noted that this estimate is based on M out of a possible L data samples in the fitting interval and must be regarded as an approximation for the true output power in the fitting interval which is equal to the sum of the squares of the L data points in the filtered output trace. In a similar manner, the output power at the ℓ^{th} frequency is computed as

$$P_{OUT}(\ell) = \frac{N}{N-1}\,\frac{1}{N-1}\left[\sum_{j,k=1}^{NS}\hat{q}_{jk}(\ell)\right]^{-1}, \tag{63}$$

which can be shown to be equal to the following

$$= \frac{1}{M(N-1)^2}\sum_{n=1}^{M}\left|\sum_{k=1}^{K}\sum_{m=1}^{N} A_k(\ell)\, N_{k,m+(n-1)N}\, \epsilon^{-im\ell(2\pi/N-1)}\right|^2. \tag{64}$$

This latter quantity is what is obtained by transforming the noise data in the segments used in the spectral matrix estimation, applying the filter $A_k(\ell)$, and then summing the squares of the magnitudes of results in each segment. This, then, is the justification for the definition of $P_{OUT}(\ell)$. In addition, we have

$$P_{OUT} = \sum_{\ell=-\left(\frac{N-3}{2}\right)}^{\frac{N-1}{2}} P_{OUT}(\ell) \ , \tag{65}$$

which is a desirable property for power estimates. It is interesting to compare Eqs. (62), (63) with (11), as this provides further justification for the definitions used.

VIII. DESCRIPTION OF COMPUTER PROGRAM

The Fortran IV computer program implements the synthesis of the maximum-likelihood filters in the frequency domain according to the algorithm given previously. The essential outputs of this program are the above filters and SC-4020 plots of the processed traces. The program works with data duplicated from Large Aperture Seismic Array tapes on a FORTRAN compatible tape known as EDIT-3 tape. A description of the program is as follows.

INPUT TO PROGRAM

Card 1 Format (8I5)
NSKP, WARMUP, BLKSZ, NSAMB, NIM, NS, NFP, POCTR
where
NSKP is the number of EDIT-3 data records or number of data samples skipped initially.
WARMUP is the number of EDIT-3 records used to determine average D.C. levels for each seismometer. The block over which the average D.C. is estimated immediately preceeds the fitting interval.
BLKSZ is the number of points of data used in each sample block.
NSAMB is the number of sample blocks.
NIM: Every NIMth point is used as data, in the fitting interval, the others are skipped.

It follows that the total number of EDIT-3 records spanning the fitting interval is BLKSZ * NSAMB * NIM which amounts to (BLKSZ * NSAMB * NIM − 1)/20 seconds of sampled data.

NS is the number of seismometers.
NFP is the number of filter points and is an odd integer. At present we make NFP and BLKSZ equal.
POCTR is a printout control parameter. If it is not zero, the cross power spectral density matrices will be printed out. If it is zero or left blank, these matrices will not be printed out.

Cards 2 to 2 + NS-1 Format (I5, 5X, A6, 4X, 2F10.3) contain IICH(i), ISEIS(i), XC(i), YC(i), i = 1, NS where
IICH is the LASA seismometer number.
ISEIS is the alphabetic name of the seismometer (such as AØ, B3).
XC(i) is the N.S. subarray projection.

YC(i) is the E.W. subarray projection of the seismometer numbered IICH(i).

Card 2 + NS   Format (2F10.3)
AZ = azimuth
HVEL = horizontal phase velocity

Card 3 + NS   Format (2I5)
NJUMP = number of EDIT-3 data records skipped before plotting. NSQSM = No. of records processed.

Card 4 + NS   Format (F10.3, I5)
TRCMX. The positive scaling for plot.
Total scaling range is 2 TRCMX.
NSITE   LASA Site Number.

The input tape is an EDIT-3 tape of the event. The following constraint must be obeyed by the data:

$$BLKSZ * NIM \leq 66$$
$$NFP * NIM \leq 66$$

and BLKSZ (and NFP) must be odd.

OUTPUT

SC-4020 hard copy plots of the center seismometer, WDS, DS, FS, and measured variances of FS, WDS and average power of all seismometers taken over blocks of 1ØØ consecutive data points. FS denotes filter and sum and is the designation for the maximum-likelihood filter which employs NFP filter points, WDS denotes weighted delay and sum and designates a one-point maximum-likelihood filter, i.e., NFP = 1, and DS denotes the sum of the delayed data.

TAPES

EDIT-3 hang on B8
Buffer on A7.

However, this may be changed at load times by changing $ ATTACH and $ AS card contents.

RUNNING TIME ON 7094

For WARMUP = 600, NSAMB = 90, NIM = 2, BLKSZ = NFP = 21, NSQSM = 7200, 6 minutes of data, the 7094 computer running time is about 10 minutes, consisting of 4 minutes to synthesize the filter and 6 minutes to obtain all the processed traces.

JOB STACKING

If several sets of data cards are run with one program deck, then the jobs will run consecutively without intervention and there is but one setup for tapes.

A listing of the program is given in Appendix A.

A detailed description of the flow of the program is as follows. Note that the uniform weighting function was used and that the Cooley-Tukey algorithm was not employed. However, a later version of the program will use this algorithm.

1. Compute the delays for the NS channels corresponding to the azimuth and horizontal velocity of the event.

2. Apply the computed delays to the data, and let the delayed data sample in the j^{th} channel at time point m be denoted by X_{jm}, to be consistent with previous notation. Let m = 1 designate the beginning of the warmup period.

3. Compute $\overline{X}_j = \frac{1}{L'} \sum_{m=1}^{L'} X_{jm}$, $j = 1, \ldots, NS$, where L' = WARMUP, which usually consists of 600 data points immediately preceding the fitting interval. No data samples are skipped in this computation.

4. Let $X'_{jm} = X_{jm} - \overline{X}_j$ and compute

$$S_{jn}(\ell) = (NFP)^{-1/2} \sum_{m=1}^{NFP} X'_{j,(m)(NIM)+L'+(n-1)(NFP)(NIM)} \epsilon^{im\ell(2\pi/NFP-1)}$$

$\ell = 0, \ldots, \frac{NFP-1}{2}$, $j = 1, \ldots, NS$, $n = 1, \ldots, M$, $M = NSAMB$.

5. Compute $\hat{f}_{jk}(\ell) = 1/M \sum_{n=1}^{M} S_{jn}(\ell) S^*_{kn}(\ell)$

$\ell = 0, \ldots, \frac{NFP-1}{2}$, $j = 1, \ldots, NS$, $k \geq j$.

Set $\hat{f}_{jk}(\ell) = \hat{f}^*_{kj}(\ell)$, $k < j$, $j = 1, \ldots, NS$.

6. Compute the inverse of $\{\hat{f}_{jk}(\ell)\}$, which we denote $\{\hat{q}_{jk}(\ell)\}$, $\ell = 0, \ldots, \frac{NFP-1}{2}$.

7. Compute

$$A_k(\ell) = \frac{\sum_{j=1}^{NS} \hat{q}_{kj}(\ell)}{\sum_{j,k=1}^{NS} \hat{q}_{kj}(\ell)} , \quad \ell = 0, \ldots, \frac{NFP-1}{2} \quad k = 1, \ldots, NS.$$

Note that, for each ℓ, the denominator need be computed only once for all k.

8. Compute

$$\theta_{jm} = \frac{1}{NFP-1} \left[A_k(0) + (-1)^m A_k\left(\frac{NFP-1}{2}\right) + 2 \sum_{\ell=1}^{\frac{NFP-3}{2}} \mathrm{Re}\{A_k(\ell)\} \cos\left(m\ell \frac{2\pi}{NFP-1}\right) + \mathrm{Im}\{A_k(\ell)\} \sin\left(m\ell \frac{2\pi}{NFP-1}\right)\right] , \quad j = 1, \ldots, NS \quad m = 0, \pm 1, \ldots, \pm\left(\frac{NFP-1}{2}\right) .$$

9. Compute the zero-lag correlation coefficients

$$\hat{\rho}_{jk}(0) = (NFP-1)^{-1} \sum_{\ell=-\left(\frac{NFP-3}{2}\right)}^{\frac{NFP-1}{2}} \hat{f}_{jk}(\ell) , \quad j = 1, \ldots, NS \quad k \geq j .$$

Set $\hat{\rho}_{jk}(0) = \hat{\rho}_{kj}(0)$, $k < j$, $j = 1, \ldots, NS$. Note that $\rho_{jk}(0)$ is numerically equal to the estimate obtained by correlating data directly, namely

$$\hat{\rho}_{jk}(0) = \frac{1}{L} \sum_{m=1}^{L} X'_{j,m(NIM)+L'} \; X'_{k,m(NIM)+L'} .$$

10. The filter coefficients for WDS, θ_k, $k = 1, \ldots, NS$, are obtained by inverting the zero-lag correlation matrix bordered by zeroes and ones, just as in time-domain method. A Lagrangian multiplier is also obtained in this computation

which is numerically the same as the output power of WDS in fitting interval, P_{WDS}.

11. Compute

$$P_{IN}(\ell) = \frac{1}{NFP - 1} \frac{1}{NS} \sum_{j=1}^{NS} \hat{f}_{jj}(\ell) , \qquad \ell = 0, \ldots, \frac{NFP-1}{2} ,$$

and

$$P_{IN} = P_{IN}(0) + P_{IN}\left(\frac{NFP-1}{2}\right) + 2 \sum_{\ell=1}^{\frac{NFP-3}{2}} P_{IN}(\ell) .$$

12. Compute

$$P_{OUT}(\ell) = \frac{NFP}{(NFP-1)^2} \left[\sum_{j,k=1}^{NS} \hat{q}_{jk}(\ell) \right]^{-1}, \qquad \ell = 0, \ldots, \frac{NFP-1}{2}$$

and

$$P_{OUT} = P_{OUT}(0) + P_{OUT}\left(\frac{NFP-1}{2}\right) + 2 \sum_{\ell=1}^{\frac{NFP-3}{2}} P_{OUT}(\ell) .$$

13. Compute signal-to-noise ratio gain of FS at ℓ^{th} frequency

$$G_{FS}(\ell) = 10 \log_{10} \frac{P_{IN}(\ell)}{P_{OUT}(\ell)}$$

and total gain

$$G_{FS} = 10 \log_{10} \frac{P_{IN}}{P_{OUT}} .$$

14. Compute signal-to-noise ratio gain of WDS

$$G_{WDS} = 10 \log_{10} \frac{P_{IN}}{P_{WDS}} .$$

15. Compute the output power of DS

$$P_{DS} = (NS)^{-2} \sum_{j,k=1}^{NS} \hat{\rho}_{jk}(0)$$

and signal-to-noise ratio gain of DS

$$G_{DS} = 10 \log_{10} \frac{P_{IN}}{P_{DS}} \quad .$$

16. Apply the filter coefficients to obtain the FS trace

$$\hat{S}_{NFP,n} = \sum_{m=-\left(\frac{NFP-1}{2}\right)}^{\frac{NFP-1}{2}} \sum_{j=1}^{NS} \theta_{jm} \, X'_{j,m(NIM)+n}, \qquad n = 1,2,3,\ldots$$

17. The DS and WDS traces are obtained in a manner similar to the computation in (16), with the appropriate weights replacing the θ_{jm}'s. For DS the weights are $\theta_j = 1/NS$, $j = 1,\ldots,NS$, and for WDS the θ_j's are as computed in step 10.

IX. EXPERIMENTAL RESULTS AND PRELIMINARY EVALUATION OF FREQUENCY-DOMAIN SYNTHESIS PROCEDURE

We will now present some of the experimental results obtained using the frequency-domain synthesis computer program. Figures 1, 4, 7, 10, 13 show the variation of $P_{OUT}(\ell), P_{IN}(\ell)$ and signal-to-noise ratio gains with frequency. The original data was sampled every 1/20 second but every other sample was used in the fitting interval so that the sampling rate was 10 cps corresponding to the foldover frequency of 5 cps shown in the figures. Although the filters are designed as if the sampling interval were 1/10 second, the filters are applied to the data with the sampling interval of 1/10 second but in steps of 1/20 second. In Figs. 2, 5, 8, 11, 14 are shown the results of the frequency-domain processing. The top trace is the center seismometer of the 25-sensor subarray, the next trace is WDS, the next is DS, and the bottom trace represents FS, all defined previously. The corresponding results for the time-domain synthesis are shown in Figs. 3, 6, 9, 12, 15. It should be noted that only the FS trace changes between these two sets of figures. Similar results for the frequency-domain synthesis and for a different event are given in Figs. 16-24. The manner in which the signal-to-noise ratio gain drops outside the fitting interval for the time and frequency-domain methods is given in Fig. 25, for various NFP and fitting interval lengths. The fitting interval estimates for P_{OUT} were compared with measured values and were found to be in agreement within ± 1 db. A plot of the frequency window $|W_N(x)|^2$, and $-10 \log_{10} |W_N(x)/W_N(0)|^2$ N=21, used in the spectral estimation program are shown in Figs. 25 and 27, respectively. The signal-to-noise ratio gain vs. frequency was obtained with somewhat better frequency resolution by using NFP = 33, as indicated in Fig. 28. The behavior of this gain at low frequencies is interesting as it indicates that the gain is highest at about 0.3 cps and drops sharply above this frequency.

The results in Figs. 16 through 24 for the 2/1/66 Central Kazakh event are rather interesting since the epicenter for this event is located about 900 miles to the southwest of a presumed nuclear test site. The raw data traces do not show clearly the PcP and pP phases, as can be seen from the top trace in these figures. However, the FS traces do show these phases quite clearly. The arrival of pP at about seven seconds after the initial P phase establishes the focal depth of the event, according to seismological travel time tables, as approximately 33 km. This in turn establishes the event as being definitely an earthquake. This conclusion was also reached by the U.S. Coast and Geodetic Survey on the basis of recordings from a worldwide network of stations.

The experimental results show that the frequency-domain filter tends to produce a much smaller precursor, in general, than the time-domain filter. This can be seen by comparing the results on the 11/11/65 Rat Island event. In addition, the signal-to-noise ratio gain of the frequency-domain filter tends to remain about the same outside the fitting interval as it is inside the fitting interval for periods of up to fourteen minutes after the fitting interval. This is not true of the time-domain filter, which for NFP = 21 and 3 minute fitting intervals tends to drop about 4 db just outside the fitting interval. However, the gain of the time-domain filter is still about 2 db better than that of the frequency domain filter outside the fitting interval, for NFP = 21 and 3 minute fitting interval. This loss is quite acceptable when it is recognized that the 7094 computer running time to synthesize the filter is only about 4 minutes for the frequency-domain method as opposed to 30 minutes for the time-domain technique.

```
$JOB          FREQ DOMAIN SYNTHESIS       RJKOLKER
$IBSYS
$ATTACH             B8
$AS                 SYSUT8
$ATTACH             A7
$AS                 SYSUT7
$EXECUTE            IBJOB
$IBJOB
$IBFTC AA
      COMMON /EDIT/ NYEAR,NDAY,NHR,NMIN,NSEC,MSC,NCH,ICH(255)
    1 CALL SPECTR
      CALL FILTER
      CALL DECOMP
       GO TO 1
      END
$IBMAP UN21     6
       ENTRY    .UN21.
.UN21. PZE      UNIT21
UNIT21 FILE     FILE2,UT7,INOUT,BIN,BLK=256
       END
$IBMAP UN20     6
       ENTRY    .UN20.
.UN20. PZE      UNIT20
UNIT20 FILE     FILE1,UT8,BIN,INPUT,BLK=256
       END
$IBMAP JUMP
       ENTRY    .SKTBL
.SKTBL PZE      *+1,,LENGTH
       PZE      .UN20.,,20
       PZE      .UN21.,,21
LENGTH EQU      *-.SKTBL-1
       END
$ORIGIN         ALPHA
$IBFTC SPCT
      SUBROUTINE SPECTR
      DIMENSION TBUF(255),DBUF(132,25)
      COMPLEX S(17,25),CBUF(25,25),F(17,325),EXPI(33,17),C1,C2
      COMMON NS,NFP,NIM,AVE(25),KD(25),JCH(25)
      DIMENSION IICH(25),ISIES(25),XC(25),YC(25)
      COMMON/EDIT/ NYEAR,NDAY,NHR,NMIN,NSEC,MSC,NCH,ICH(255)
      EQUIVALENCE (CBUF(1,1),DBUF(1,1))
      INTEGER WARMUP,BLKSZ,BUFSZ,FULLSZ,POCTR
C
6002  FORMAT(4H NO.,5X,11HSIESMOMETER,5X,11HCHANNEL NO.,5X,11HCH. ON LIS
     1T,5X,11HXCOORDINATE,5X,11HYCOORDINATE  //(I4,8X,A6,11X,I5,11X,
     2I5,5X,F11.3,5X,F11.3 ))
      WRITE(6,600)
  600 FORMAT(45H1FREQUENCY DOMAIN SYNTHESIS OF FILTER WEIGHTS   )
C
      READ(5,500)NSKP,WARMUP,BLKSZ,NSAMB,NIM,NS,NFP,POCTR
  500 FORMAT(8I5)
      NPRC=(NSAMB*NIM*BLKSZ-1)/20
      WRITE(6,601)NSKP,WARMUP,BLKSZ,NSAMB,NIM,NS,NFP,NPRC,POCTR
  601 FORMAT(26H0NUMBER OF RECORDS SKIPPED   I5//
     *       44H NUMBER OF RECORDS USED GETTING AVERAGES      I5//
     *       44H NUMBER OF SAMPLE POINTS PER SAMPLE BLOCK     I5//
     *       44H NUMBER OF SAMPLE BLOCKS                     I5//
     *       44H NUMBER OF RECORDS PER SAMPLE POINT           I5//
```

```
     *        44H NUMBER OF SIESMOMETERS                        I5//
     *        46H NUMBER OF FREQUENCIES(AND FILTER PTS/CHANNEL)  I5//
     *        44H NUMBER OF SECONDS IN FITTING INTERVAL           I5 //
     *        39H FLAG FOR CONTROL OF MATRIX PRINTOUTS           I5   )
C
C
      READ(5,5001)(IICH(I),ISIES(I),XC(I),YC(I),I=1,NS)
 5001 FORMAT( I5,5X,A6,4X,2F10.3   )
      REWIND 20
      REWIND 21
      READ(20) NYEAR,NDAY,NHR,NMIN,NSEC,MSC,NCH,(ICH(I),I=1,NCH)
      DO    1   I=1,NS
    1 JCH(I)=0
      IFLAG=0
C
      DO  2  I=1,NS
      DO  2  J=1,NCH
      IF(IICH(I).EQ.ICH(J)) JCH(I)=J
    2 CONTINUE
C
      DO  3  I=1,NS
      IF(JCH(I).EQ.0) IFLAG=1
      IF( JCH(I).EQ. 0) WRITE(6,6300) IICH(I)

    3 CONTINUE
 6300 FORMAT( 8H CHANNEL I4,23HNOT ON TAPE--EXIT TAKEN)
      IF(IFLAG.EQ.1)  STOP
C
      WRITE(6,6330) NYEAR,NDAY,NHR,NMIN,NSEC,MSC
 6330 FORMAT(/// 49H DATING INFORMATION FOR THE EVENT BEING PROCESSED
     *//5H YEARI4,3HDAYI4,4HHOURI4,3HMINI4,3HSECI4,3HMSCI4)
C
      WRITE(6,6331) NCH,(ICH(I),I=1,NCH)
 6331 FORMAT( ////30H INFORMATION ABOUT EDIT-3 TAPE   ///
     *30H NUMBER OF CHANNELS ON TAPE IS  I5,12HAND THEY ARE  /(20I5))
C
C
      WRITE(6,6002)(I,ISIES(I),IICH(I),JCH(I),XC(I),YC(I),I=1,NS)
C
C
      NFPP=(NFP-1)/2 + 1
C
      PI2=2.0*3.1415962
      YY=PI2/FLOAT(NFP-1)
C
C
      DO  4  L=1,NFPP
      XL=L-1
      DO  4  M=1,BLKSZ
      FAC=XL*YY*FLOAT(M)
    4 EXPI(M,L)  = COS(FAC)*(1.0,0.0)+ SIN(FAC)*(0.0,1.0)
C
C
      READ(5,505) AZ,HVEL
  505 FORMAT(2F10.3)
      WRITE(6,605)  AZ,HVEL
  605 FORMAT( 7H AZMUTH  F10.3,26H HORIZONTAL PHASE VELOCITY  F10.3   )
      CALL DELAY( AZ,HVEL,NS,XC,YC,KD)
      MIN=100
```

```
      DO  5  I=1,NS
      IF( KD(I).LT.MIN) MIN=KD(I)
    5 CONTINUE
      DO  6  I=1,NS
    6 KD(I)=KD(I)-MIN
      WRITE(6,606)(KD(I),I=1,NS)
  606 FORMAT(22H BIASED CHANNEL DELAYS //(20I5))
C
C
      BUFSZ=NIM*BLKSZ
      FULLSZ=2*BUFSZ
      IF(FULLSZ.GT. 132)  IFLAG=1
      IF(IFLAG.EQ. 1) WRITE(6,607) FULLSZ
  607 FORMAT(22H0DOUBLE BUFFER SIZE OF  I5,23HEXCEEDS AVAILABLE SPACE  )
      IF(IFLAG.EQ.1)  STOP
C
C
      CALL AHEAD(20,0,NSKP)
       DO  7  I=1,NS
    7 AVE(I)=0
      DO  8  K=1,WARMUP
      READ(20)(TBUF(I),I=1,NCH)
      DO  8  KS=1,NS
      N=JCH(KS)
    8 AVE(KS)=AVE(KS)+TBUF(N)
      XW=WARMUP
      DO  9  I=1,NS
    9 AVE(I)=AVE(I)/XW
      WRITE(6,702)(AVE(I),I=1,NS)
C
C
      DO  10  K=1,BUFSZ
      KK=BUFSZ+K
      READ(20)(TBUF(I),I=1,NCH)
      DO  10  KS=1,NS
      N=JCH(KS)
   10 DBUF(KK,KS)=TBUF(N)-AVE(KS)
C
C
C
      DO  11  I=1,325
      DO  11  L=1,NFPP
   11 F(L,I)=(0.0,0.0)
C
C
C
      DO  24  MAJ=1,NSAMB
      DO  20  K=1,BUFSZ
      KK=BUFSZ+K
      READ(20)(TBUF(I),I=1,NCH)
      DO  20  KS=1,NS
      DBUF(K,KS)=DBUF(KK,KS)
      N=JCH(KS)
   20 DBUF(KK,KS)=TBUF(N)-AVE(KS)
      DO  22  LFR=1,NFPP
      DO  22  KS=1,NS
      C1=(0.0,0.0)
      MT=0
      DO  21  M=1,BUFSZ,NIM
```

```
      MT=MT+1
      MM=M+KD(KS)
   21 C1=C1+DBUF(MM,KS)*EXPI(MT,LFR)
   22 S(LFR,KS)=C1
      DO  23      JS=1,NS
      LSUM=JS*(JS-1)/2
      DO  23  KS=1,JS
      LSM=LSUM+KS
      DO  23  LFR=1,NFPP
   23 F(LFR,LSM)=S(LFR,JS)*CONJG(S(LFR,KS))+F(LFR,LSM)
   24 CONTINUE
      XDIV=FLOAT(NSAMB)*FLOAT(BLKSZ)
      DO  25  I=1,325
      DO  25  L=1,NFPP
   25  F(L,I)=F(L,I)/XDIV
      DO 30  LFR=1,NFPP
      DO  26  JS=1,NS
      LSUM=JS*(JS-1)/2
      DO  26  KS=1,JS
      LSM=LSUM+KS
   26 CBUF(JS,KS)=F(LFR,LSM)
       DO  27  JS=1,NS
      JJ=JS+1
      DO  27  KS=JJ,NS
   27 CBUF(JS,KS)=CONJG(CBUF(KS,JS))
      WRITE(21)((CBUF(J,K),J=1,NS),K=1,NS)
      IF ( POCTR.NE. 0) WRITE(6,700)  LFR
      IF(POCTR.EQ.0)  GO TO 30
      DO 28 K=1,NS
   28  WRITE(6,701) (CBUF(J,K),J=1,NS)
   30 CONTINUE
  700 FORMAT(15H PSD MATRIX FOR I5,15H-1 TH FREQUENCY  //)
  701 FORMAT(//((F12.4,F12.4),7X,(F12.4,F12.4),7X,(F12.4,F12.4),7X,(F11
     *.4,F12.4)))
      END FILE 21
      REWIND20
      WRITE(21) (AVE(I),I=1,NS)
      END FILE 21
  702 FORMAT(47H AVERAGE DC LEVELS PREVIOUS TO FITTING INTERVAL  //(12
     *F10.4))
      REWIND  21
      RETURN
      END
$IBFTC DLAY
      SUBROUTINE DELAY(AZ,HVEL,NS,XC,YC,KD)
      DIMENSION  XC(NS),YC(NS),KD(NS),RS(100),TH(100)
      DO 3 I=1,NS
      IF(SQRT(XC(I)**2+YC(I)**2).EQ.0.0) GO TO 2
      RS(I)=SQRT(XC(I)**2 +YC(I)**2)
      TH(I)=ATAN2( YC(I),XC(I))
      GO TO 3
    2 TH(I)=0.0
      RS(I)=0.0
    3 CONTINUE
      CONST=180.0/3.14159
      DO 66 I=1,NS
   66 TH(I)=TH(I)*CONST
      T=0.05
      C=3.1415927/180.0
```

```
      CC=-1.0/(HVEL*T)
      DO  50  I=1,NS
      DEL  =RS(I)*CC*COS(C*(TH(I)-AZ))
      IF( DEL.LE.0.0)GOTO25
      GO TO 30
   25 KD(I)=DEL-0.5
      GO TO 50
   30 KD(I)=DEL+0.5
   50 CONTINUE
      RETURN
      END
$ORIGIN        ALPHA
$IBFTC SOSO    FULIST
      SUBROUTINE  FILTER
      COMPLEX CBUF(25,25),ABUF(25),QBUF(25,25),C1,C2,C3
      DIMENSION     W(25,33),OPF(25),RHO(26,26),RR(26,26)
      COMMON NS,NFP,NIM,AVE(25),KD(25),JCH(25)
 6800 FORMAT( 21H VARIANCE OF INPUT IS F13.4,5X,
     121HVARIANCE OF OUTPUT IS F13.4/
     225H NOISE REDUCTION FOR WS. F13.4,5X,
     *24HNOISE REDUCTION FOR WDS   F13.4   //
     *20H VARIANCE OF WDS          F13.4  )
 6900 FORMAT(4H FORI5 ,25HTH MULTIPLE OF FREQUENCY  / 19H INPUT VARIANC
     1E IS F13.4,18HOUTPUT VARIANCE IS F13.4,25HNOISE REDUCTION IN DB I
     2S F13.4  )
 6802 FORMAT(23H OPTIMAL ONE PT FILTER   //(9F13.4))
 6801 FORMAT(13H FOR CHANNEL I5,18HFILTER WEIGHTS ARE//
     1 (9F14.8))
      SUMO=0
      NFPP=(NFP-1)/2 + 1
      XNS=NS
      NFPM=NFP-1
      XNF=NFPM
      XDEN=(XNF+1.0)/(XNF*XNF)
      REWIND 21
      TRIG= 2.0*3.141592/XNF
      SUMI=0
      XDIV=XNS*XNF
      DO  30  LFREQ=1,NFPP
      LF=LFREQ-1
      READ(21)((CBUF(I,J),I=1,NS),J=1,NS)
      XLF=LF
      C1=(0.0,0.0)
      DO  10  J=1,NS
   10 C1=C1+CBUF(J,J)
      VI=REAL(C1)/XDIV
C   NOW THE COMPLEX INVERSION
      CALL COMINV (25,NS,CBUF,QBUF)
      C1=(0.0,0.0)
      DO  14  J=1,NS
      DO  14  I=1,NS
   14 C1=C1+QBUF(I,J)
      VO=REAL(C1)
      VO=XDEN/VO
      DO  16  K=1,NS
      C2=(0.0,0.0)
      DO  15  J=1,NS
   15 C2=C2+QBUF(K,J)
   16 ABUF(K)=C2/C1
```

```
      IF(LFREQ.EQ.1) GO TO 18
      IF(LFREQ.EQ.NFPP)GO TO 21
      DO   17    J=1,NS
      FAC1=REAL(ABUF(J))
      FAC2=AIMAG(ABUF(J))
      DO   17    K=1,NFP
      XXK=K-1
      YYK=-FLOAT((NFP-1)/2)+XXK
      TAC=FAC1*COS(YYK*XLF*TRIG)+FAC2*SIN(YYK*XLF*TRIG)
   17 W(J,K)=W(J,K)+2.0*TAC
      GO TO 23
   18 DO  19   J=1,NS
      DO  19   K=1,NS
   19 RHO(K,J)=REAL(CBUF(K,J))
      DO   20  J=1,NS
      FAC1= REAL(ABUF(J))
      DO   20  K=1,NFP
   20 W(J,K)=FAC1
      GO TO 23
   21 DO  22   J=1,NS
      DO  22   K=1,NFP
      KK=K-1
      KZ=KK-(NFP-1)/2
      KZ=IABS(KZ)
      FAC1=1.0
      DO 2022 II=1,NFP
      IF(II-1-KZ)   2022,22,2022
 2022 FAC1=-FAC1
   22 W(J,K)=W(J,K)+FAC1*REAL(ABUF(J))
      DO 222 J=1,NS
      DO 222 K=1,NS
  222 RHO(K,J)=RHO(K,J)+ REAL (CBUF(K,J))
   23 IF(LFREQ.EQ.1) GO TO 25
      IF(LFREQ.EQ.NFPP) GO TO 25
      DO  24   J=1,NS
      DO  24   K=1,NS
   24 RHO(J,K)=RHO(J,K)+ 2.0*REAL(CBUF(J,K))
   25 IF (LFREQ.EQ.1) GO TO 26
      IF(LFREQ.EQ. NFPP) GO TO 26
      SUMO=SUMO+2.0*VO
      SUMI=SUMI+2.0*VI
      GO TO 28
   26 SUMO=SUMO+VO
      SUMI=SUMI+VI
   28 DBR= 10.0*ALOG10(VI/VO)
      WRITE(6,6900) LF,VI,VO,DBR
   30 CONTINUE
      CALL AHEAD(21,2,0)
      DO   31    K=1,NFP
      DO   31    J=1,NS
   31 W(J,K)=W(J,K)/FLOAT(NFP-1)
      WRITE(21)((W(J,K),J=1,NS),K=1,NFP)
      DO   32    J=1,NS
      DO   32    K=1,NS
   32 RHO(K,J)=RHO(K,J)/FLOAT(NFP)
      NS1=NS+1
      DO   33   I=1,NS
      RHO(NS1,I)=1.0
   33 RHO(I,NS1)=1.0
```

```
      RHO(NS1,NS1)=0
      CALL MATINV(26,NS1,RHO,RR)
      DO  34   I=1,NS
   34 OPF(I)=RR(I,NS1)
      WRITE(21)(OPF(I),I=1,NS)
      ENDFILE 21
      REWIND 21
      DBR=10.0*ALOG10(SUMI/SUMO)
      DBS=10.0*ALOG10(-SUMI/RR(NS1,NS1))
      VAR=-RR(NS1,NS1)
      WRITE(6,6800) SUMI,SUMO,DBR,DBS,VAR
      WRITE(6,6902)
 6902 FORMAT(36H ZERO SHIFT CROSS-CORRELATION MATRIX      )
      DO 333 I=1,NS
  333 WRITE(6,6901) (RHO(I,J),J=1,NS)
 6901 FORMAT(//(9F13.4))
      DO  35   J=1,NS
   35 WRITE(6,6801)  J,(W(J,K),K=1,NFP)
      WRITE(6,6802) (OPF(I),I=1,NS)
      DO 356 K=1,NFP
      SUM=0
      DO 355  J=1,NS
  355 SUM=SUM+W(J,K)
  356 WRITE(6,6922) K,SUM
 6922 FORMAT(29H SUM OF WEIGHTS AT FILTER PT.  I3,7X, F20.9)
      RETURN
      END
$IBFTC CINV
      SUBROUTINE COMINV(NDIM,NORD,C,CI)
      COMPLEX   C(NDIM,NDIM),CI(NDIM,NDIM)
      DIMENSION  B(50,50),BI(50,50)
      DO 10 J=1,NORD
      JJ=J+NORD
      DO   10 I=1,NORD
      II=I+NORD
      B(I,J)=REAL(C(I,J))
      B(I,JJ)=AIMAG(C(I,J))
      B(II,JJ)=REAL(C(I,J))
      B(II,J)=-AIMAG(C(I,J))
   10 CONTINUE
      NDORD=2*NORD
      CALL MATINV(50,NDORD,B,BI)
      DO  20 J=1,NORD
      JJ=J+NORD
      DO 20 I=1,NORD
   20 CI(I,J)=BI(I,J)*(1.0,0.0)+BI(I,JJ)*(0.0,1.0)
      RETURN
      END
$IBFTC MAIN
C     MATRIX INVERSION BY BORDERING
      SUBROUTINE MATINV(NU,NORD,A,AI)
      DIMENSION   A(NU,NU),AI(NU,NU)
      DIMENSION ICH(50),KCH(50),COL(50),ROW(50)
      DO 1  K=1,NORD
      KCH(K)=K
    1 ICH(K)=K
      DO 2  K=1,NORD
      IF (A(K,1))  3,2,3
    2 CONTINUE
```

```
      WRITE(6,1000)
 1000 FORMAT(1H ,22H EXIT1,MATRIX SINGULAR)
      CALL EXIT
    3 ICH(1)=K
      DO 4  J=1,NORD
      TEMP=A(1,J)
      A(1,J)=A(K,J)
    4 A(K,J)=TEMP
      AI(1,1)= 1.0/A(1,1)
      DO 15 N=2,NORD
      DO 111 NK=N,NORD
      KCH(N)=NK
      DO 90 J=1,NORD
      TEMP=A(J,NK)
      A(J,NK)=A(J,N)
   90 A(J,N)=TEMP
      DO 11 KK=N,NORD
      ICH(N)=KK
      DO 5  J=1,NORD
      TEMP=A(N,J)
      A(N,J)=A(KK,J)
    5 A(KK,J)=TEMP
      L=N-1
      DO 7  I=1,L
      SUM1=0.0
      SUM2=0.0
      DO 6  J=1,L
      SUM1=SUM1+AI(I,J)*A(J,N)
    6 SUM2=SUM2+A(N,J)*AI(J,I)
      COL(I)=SUM1
    7 ROW(I)=SUM2
      SUM1=0.0
      SUM2=0.0
      DO 8  J=1,L
      SUM1= SUM1+ A(N,J)*COL(J)
    8 SUM2= SUM2+ A(J,N)*ROW(J)
      AVE = (SUM1 + SUM2)/2.0
      ADIV= A(N,N) -AVE
      IF(ADIV) 12 ,9, 12
    9 ICH(N)=N
      DO 10 J=1,NORD
      TEMP= A(N,J)
      A(N,J)=A(KK,J)
   10 A(KK,J)=TEMP
   11 CONTINUE
      KCH(N)=N
      DO  91 J=1,NORD
      TEMP=A(J,NK)
      A(J,NK)=A(J,N)
   91 A(J,N)=TEMP
  111 CONTINUE
      WRITE(6,1001)N
 1001 FORMAT(1H ,21HEXIT2,MATRIX SINGULAR            ,I2    )
      CALL EXIT
   12 AI(N,N)=1.0/ADIV
      DO 13 J=1,L
      AI(J,N)=-COL(J)/ADIV
   13 AI(N,J)=-ROW(J)/ADIV
      DO 14 I=1,L
```

```
      DO 14 J=1,L
   14 AI(I,J)=AI(I,J)+COL(I)*ROW(J)/ADIV
   15 CONTINUE
      DO 116JJ=1,NORD
      KR=NORD+1-JJ
      KL=KCH(KR)
      DO 116 J=1,NORD
      TEMP=A(J,KR)
      A(J,KR)=A(J,KL)
      A(J,KL)=TEMP
      TEMP=AI(KR,J)
      AI(KR,J)=AI(KL,J)
  116 AI(KL,J)=TEMP
      DO 16 J=1,NORD
      KR=NORD+1-J
      KN=ICH(KR)
      DO 16 LL=1,NORD
      TEMP=A(KR,LL)
      TEMPP=AI(LL,KR)
      A(KR,LL) =A(KN,LL)
      AI(LL,KR)=AI(LL,KN)
      A(KN,LL)=TEMP
   16 AI(LL,KN)=TEMPP
      RETURN
      END
$ORIGIN         ALPHA
$IBFTC PROC     LIST
      SUBROUTINE DECOMP
      COMMON NS,NFP,NIM,AVE(25),KD(25),JCH(25)
      COMMON /EDIT/ NYEAR,NDAY,NHR,NMIN,NSEC,MSC,NCH,ICH(255)
      COMMON/BETH/ NJMP,NSQSM ,W(25,33),OPF(25)
      READ(5,5000) NJMP,NSQSM,NSKP,NPLOT
 5000 FORMAT(4I5)
      REWIND 20
      REWIND 21
      READ(20) NY,ND,NH,NMN,NSC,MSC,NCH,(ICH(I),I=1,NCH)
       CALL AHEAD(21,2,0)
      READ(21)((W(J,K),J=1,NS),K=1,NFP)
      READ(21)(OPF(K),K=1,NS)
      IF(NJMP.EQ.0) GO TO 10
      CALL PLOTT
   10 IF(NSKP.EQ.0) GO TO 20
   20 RETURN
      END
$ORIGIN         BETA
$JOB
$IBSYS
$EXECUTE        IBJOB
$IBJOB
$IBFTC PLTT     FULIST
      SUBROUTINE PLOTT
      COMMON NS,NFP,NIM,AVE(25),KD(25),JCH(25)
      COMMON /EDIT/ NYEAR,NDAY,NHR,NMIN,NSEC,MSEC,NCH,ICH(255)
      COMMON /BETH/ NJMP,NSQSM,W(25,33),OPF(25)
      DIMENSION  TBUF(255),DBUF(25,200),IYLINE(4),XSEC(6),IXSEC(6)
      DIMENSION   MB(4),MT(4)
      DIMENSION   ARRAY(1000,4)
      DIMENSION   LD(25)
      INTEGER  CHOP,DELTA
```

```
      NDEL=(NFP-1)*NIM/2
      DO 970 I=1,NS
      LD(I)=KD(I)
  970 KD(I)=KD(I)+NDEL
       WRITE(6,8700)  NJMP,NSQSM
 8700 FORMAT(26H NUMBER OF RECORDS SKIPPED    I5,
     *  29H NUMBER OF RECORDS PROCCESSED    I5  )
      READ(5,570)  TRCMX,NSITE
  570 FORMAT(  F10.3,   I5)
      TRCMN=-TRCMX
      CALL STOIDV(15HFOUR TRACE PLOT   ,3)
      CALL BRITEV
      CALL CAMRAV(9)
      CALL FRAMEV(0)
      CALL PRINTV(-13,13HDATE OF EVENT    ,400,670)
      CALL PRINTV (-4,4HYEAR ,450,630)
      CALL PRINTV(-3, 3HDAY ,450,600)
      CALL PRINTV(-21, 21HSTARTING TIME OF PLOT ,400,560)
      CALL PRINTV ( -4 ,4HHOUR ,450,530)
      CALL PRINTV(-6 ,6HMINUTE ,450,500)
      CALL PRINTV(-6,6HSECOND ,450,470)
      CALL PRINTV(-11,11HSITE NUMBER ,400,420)
      CALL PRINTV(-2,2HFS  ,950,148  )
      CALL PRINTV(-3,3HWDS ,950,648)
      CALL PRINTV(-2,2HDS ,950,398)
      CALL PRINTV(-4,4HDEEP  ,950,898)
      CALL PRINTV(-4,4HWELL  ,950,882)
      DELTA=50*NSQSM
      CHOP=MSC+1000*NSEC+60000*NMIN+3600000*NHR
      CHOP=DELTA+CHOP
      IHR=CHOP/3600000
      CHOP=CHOP-3600000*(CHOP/3600000)
      IMIN=CHOP/60000
      CHOP=CHOP-60000*(CHOP/60000)
      ISEC=CHOP/1000
      X=NYEAR
      CALL LABLV(X,530,630,4,1,4)
      X=NDAY
      CALL LABLV(X,530,600,4,1,4)
      X=IHR
      CALL LABLV(X,530,530,4,1,4)
      X=IMIN
      CALL LABLV(X,530,500,4,1,4)
      X=ISEC
      CALL LABLV(X,530,470,4,1,4)
      X=NSITE
      CALL LABLV(X,530,420,4,1,4)
      CALL FRAMEV(0)
         IYLINE(1)=148
      IYLINE(2)=648
      IYLINE(3)=398
      IYLINE(4)=898
      XSEC(1)=0
      XSEC(2)=10.0
      XSEC(3)=20.0
      XSEC(4)=30.0
      XSEC(5)=40.0
      XSEC(6)=50.0
      IXSEC(1)=12
```

```
      IXSEC(2)=212
      IXSEC(3)=412
      IXSEC(4)=612
      IXSEC(5)=812
      IXSEC(6)=978
      DO  30   I=1,4
      MT(I)=1023-(IYLINE(I)+125)
   30 MB(I)=IYLINE(I)-125
      CFS=0
      CWDS=0
      CAP=0
      XNS=NS
      KBL=0
      WRITE(6,6300)
      WRITE(6,6400)
      NN=NJMP+1
      CALL AHEAD (20,0,NN)
      DO  35  I=101,200
      READ(20)(TBUF(K),K=1,NCH)
      DO  35  KS=1,NS
      N=JCH(KS)
   35 DBUF(KS,I)=TBUF(N)-AVE(KS)
      NBLOK=NSQSM/100
   50 IF(NBLOK.LE.0)  GO TO 71
      IF(NBLOK-10)  51,52,52
   51 IBMAX=NBLOK
      GO TO 53
   52 IBMAX=10
   53 NBLOK=NBLOK-10
      DO  58  IB=1,IBMAX
      KBL=KBL+1
      DO  54  I=1,100
      II=I+100
      READ(20)(TBUF(K),K=1,NCH)
      DO   54  KS=1,NS
      DBUF(KS,I)=DBUF(KS,II)
      N=JCH(KS)
   54 DBUF(KS,II)=TBUF(N)-AVE(KS)
      IADX=(IB-1)*100
      BFS=0
      BWDS=0
      BAP=0
      DO  57  IT=1,100
      SUMAP=0
      SUMC=0
      SUMW=0
      SUMD=0
      IADX=IADX+1
      DO  56  KS=1,NS
      IDFS=IT+LD(KS)-NIM
      IDWS=IT+KD(KS)
      SUMF=0
      DO  55  KF=1,NFP
      IDFS=IDFS+NIM
   55 SUMF=SUMF+DBUF(KS,IDFS)*W(KS,KF)
      SUMC=SUMC+SUMF
      SUMW=SUMW+DBUF(KS,IDWS)*OPF(KS)
      SUMD=SUMD+DBUF(KS,IDWS)
   56 SUMAP=SUMAP+(DBUF(KS,IDWS)**2)
```

```
      BFS=BFS+SUMC*SUMC
      BWDS=BWDS+SUMW*SUMW
      BAP=BAP+(SUMAP/XNS)
      ARRAY(IADX,1)=SUMC
      ARRAY(IADX,2)=SUMW
      ARRAY(IADX,3)=SUMD/XNS
      IZ=IT+KD(1)
   57 ARRAY(IADX,4)=DBUF(1,IZ)
      BFS=BFS/100.0
      BWDS=BWDS/100.0
      BAP=BAP/100.0
      CFS=CFS+BFS
      CAP=CAP+BAP
      CWDS=CWDS+BWDS
   58 WRITE(6,6200)KBL,BFS,CFS,BWDS,CWDS,BAP,CAP
      IAD=IADX-1
      IX1=12
      IX2=1012
      DO  60  I=1,4
   60 CALL LINEV(IX1,IYLINE(I),IX2,IYLINE(I))
      DO 61 I=1,6
   61 CALL LABLV(XSEC(I),IXSEC(I),12,4,1,4)
      DO  65  I=1,6
   65  XSEC(I)=XSEC(I)+50.0
      DO  62  I=1,4
      IXS=12
      CALL YSCALV(TRCMN,TRCMX,MB(I),MT(I))
      DO  62  KP=1,IAD
      IXL=IXS+1
      Z1=ARRAY(KP,I)
      Z2=ARRAY(KP+1,I)
      IY1=IYV(Z1)
      IY2=IYV(Z2)
      CALL LINEV(IXS,IY1,IXL,IY2)
   62 IXS=IXL
      CALL FRAMEV(0)
      GO TO 50
   71 CALL PLTND
      REWIND 20
      RETURN
 6400 FORMAT(7H BLOCK ,12X,2HFS,14X,7HFS CUM.,14X,3HWDS,13X,8HWDS CUM.,
     $15X,2HAP,14X,7HAP CUM.  //)
 6200 FORMAT(I6,6(7X,F12.4))
 6300 FORMAT(14H SQASM OUTPUT      //////)
      END
```

693

REFERENCES

1. Capon, J. and R. J. Greenfield, Asymptotically Optimum Multidimensional Filtering for Sampled-Data Processing of Seismic Arrays, Technical Note 1965-57, M. I. T., Lincoln Laboratory, (17 December 1965).

2. Grenander, U. and M. Rosenblatt, Statistical Analysis of Stationary Time Series, (John Wiley and Sons, Inc., New York; 1957).

3. Blackman, R. B. and J. W. Tukey, The Measurement of Power Spectra From the Point of View of Communications Engineering (Dover, New York; 1959).

4. Blackman, R. B., Data Smoothing and Prediction (Addison-Wesley, Massachusetts; 1965).

5. Goodman, N. R., On the Joint Estimation of the Spectrum, Cospectrum and Quadrature Spectrum of a Two-Dimensional Stationary Gaussian Process, Scientific Paper No. 10, Engineering Statistics Laboratory, New York University.

6. Rosenblatt, M., Statistical Analysis of Stochastic Processes with Stationary Residuals, Probability and Statistics, the Harald Cramér Volume, edited by U. Grenander (John Wiley and Sons, Inc., New York; 1959).

7. Jones, R. H., A Reappraisal of the Periodogram in Spectral Analysis, Technometrics, 7, (November, 1965).

8. Goodman, N. R., Statistical Analysis Based on a Certain Multivariate Complex Gaussian Distribution, Annals Math. Stat., 34 (1963).

9. Cooley, J. W. and J. W. Tukey, An Algorithm for the Machine Calculation of Complex Fourier Series, Math. of Comp., 19 (1965).

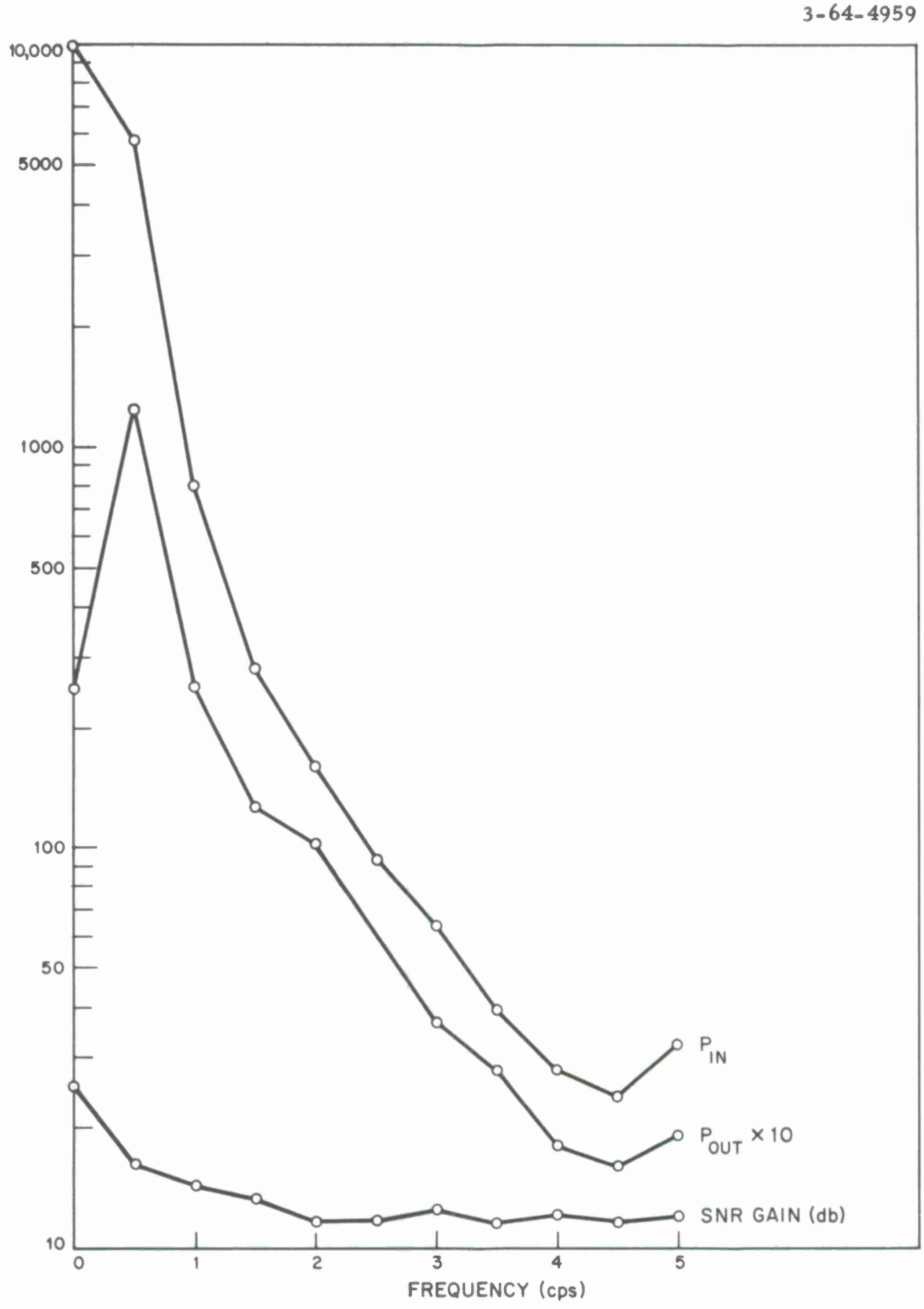

Figure 1. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray B1, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4960

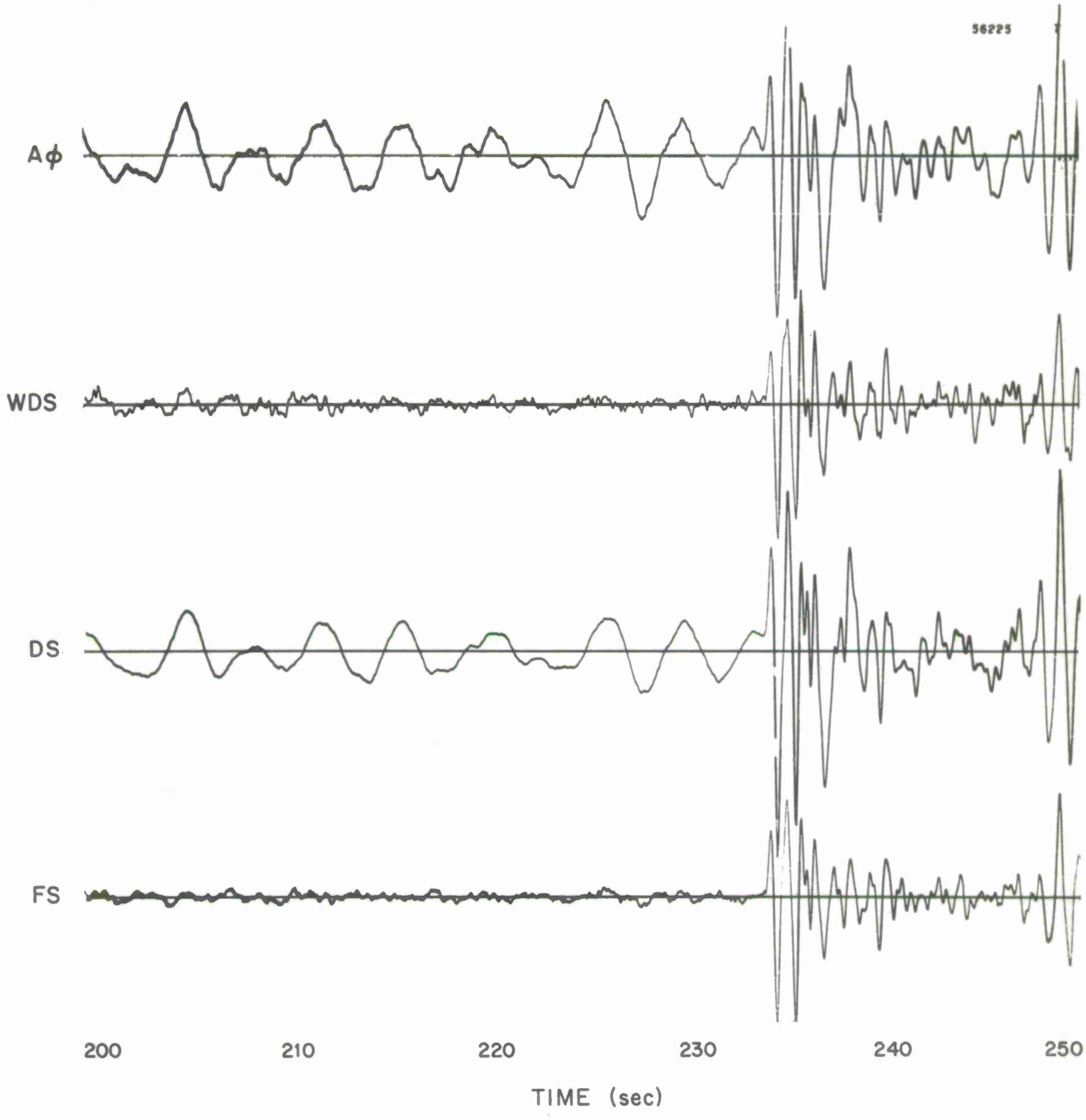

Figure 2. Processed traces obtained by frequency-domain synthesis procedure, 11/11/65 Rat Island event, subarray B1, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4961

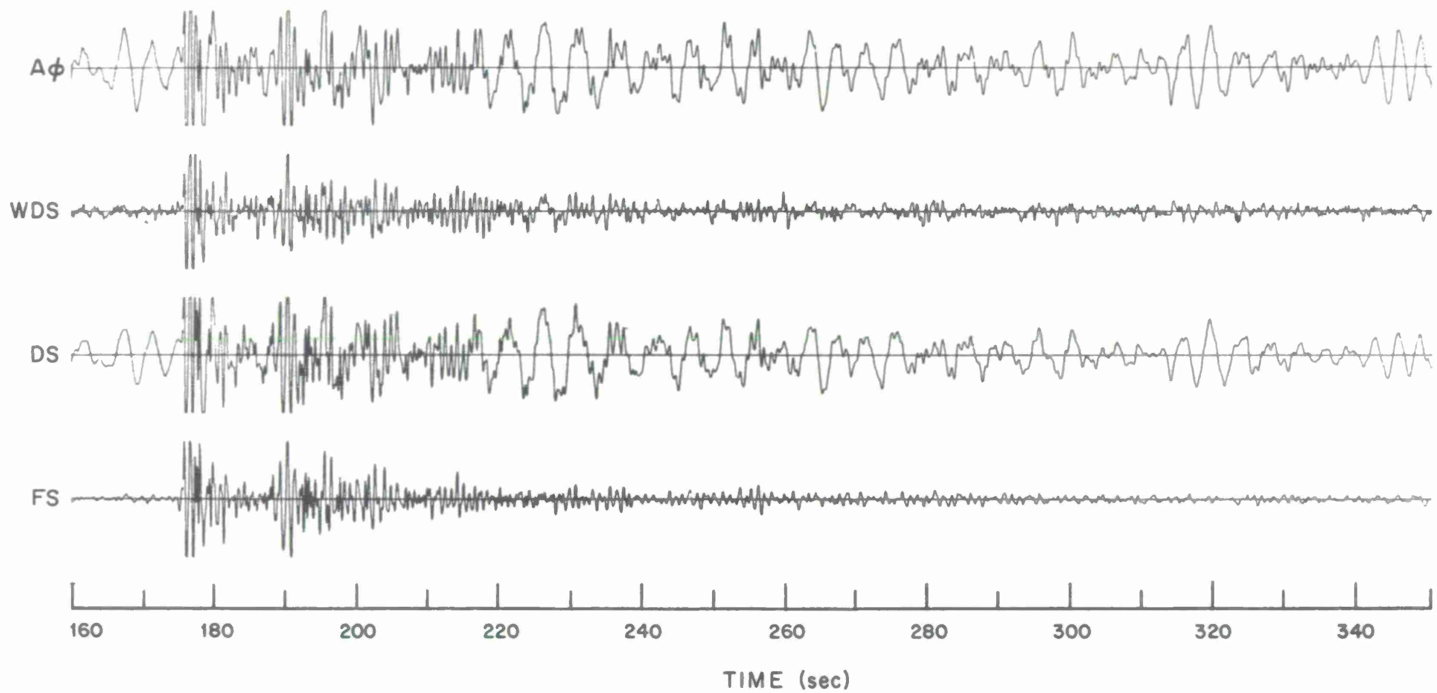

Figure 3. Processed traces obtained by time-domain synthesis procedure, 11/11/65 Rat Island event, subarray B1, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4962

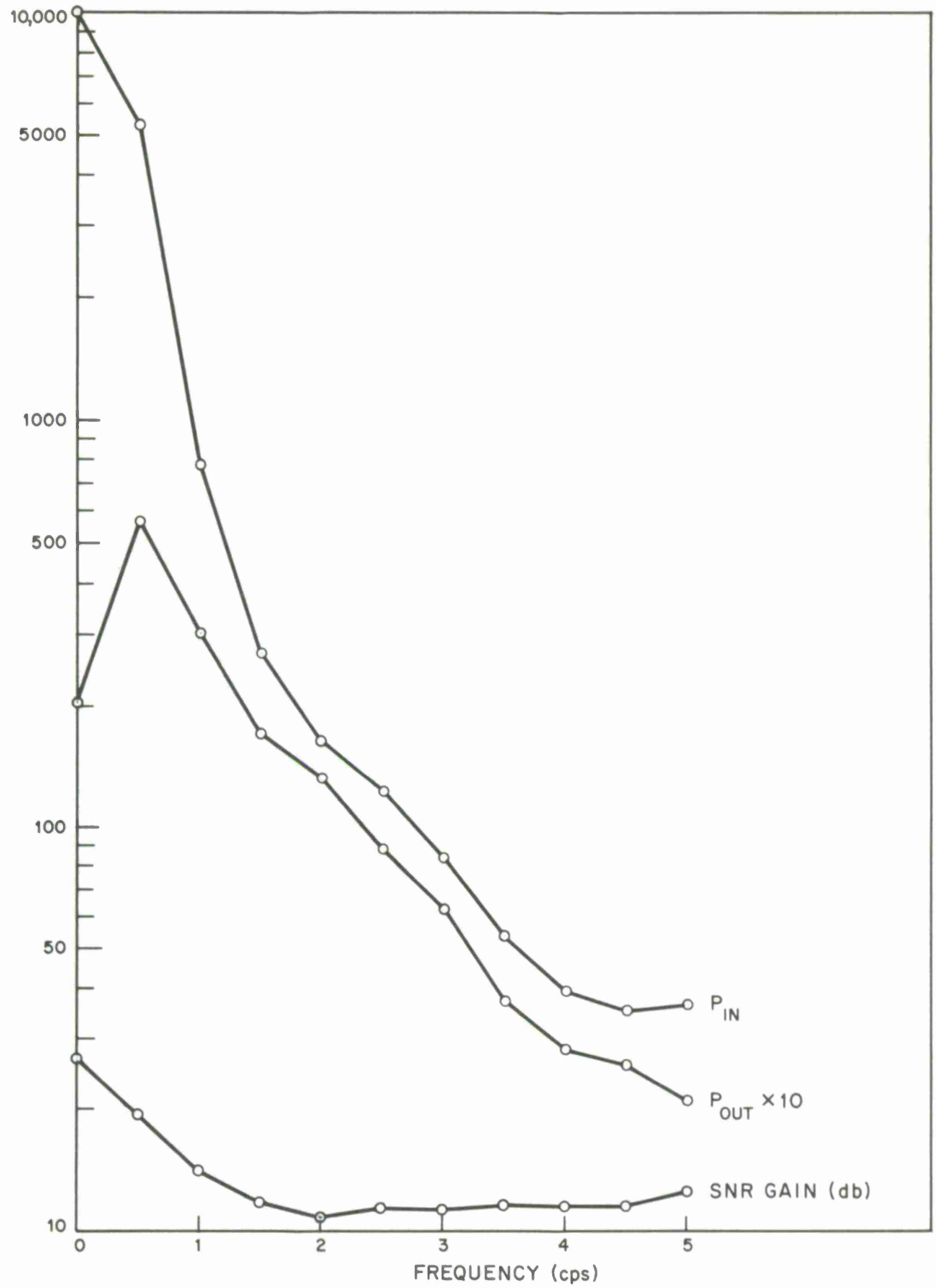

Figure 4. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray AØ, NIM = 2, NFP = 21, 3 minute fitting interval.

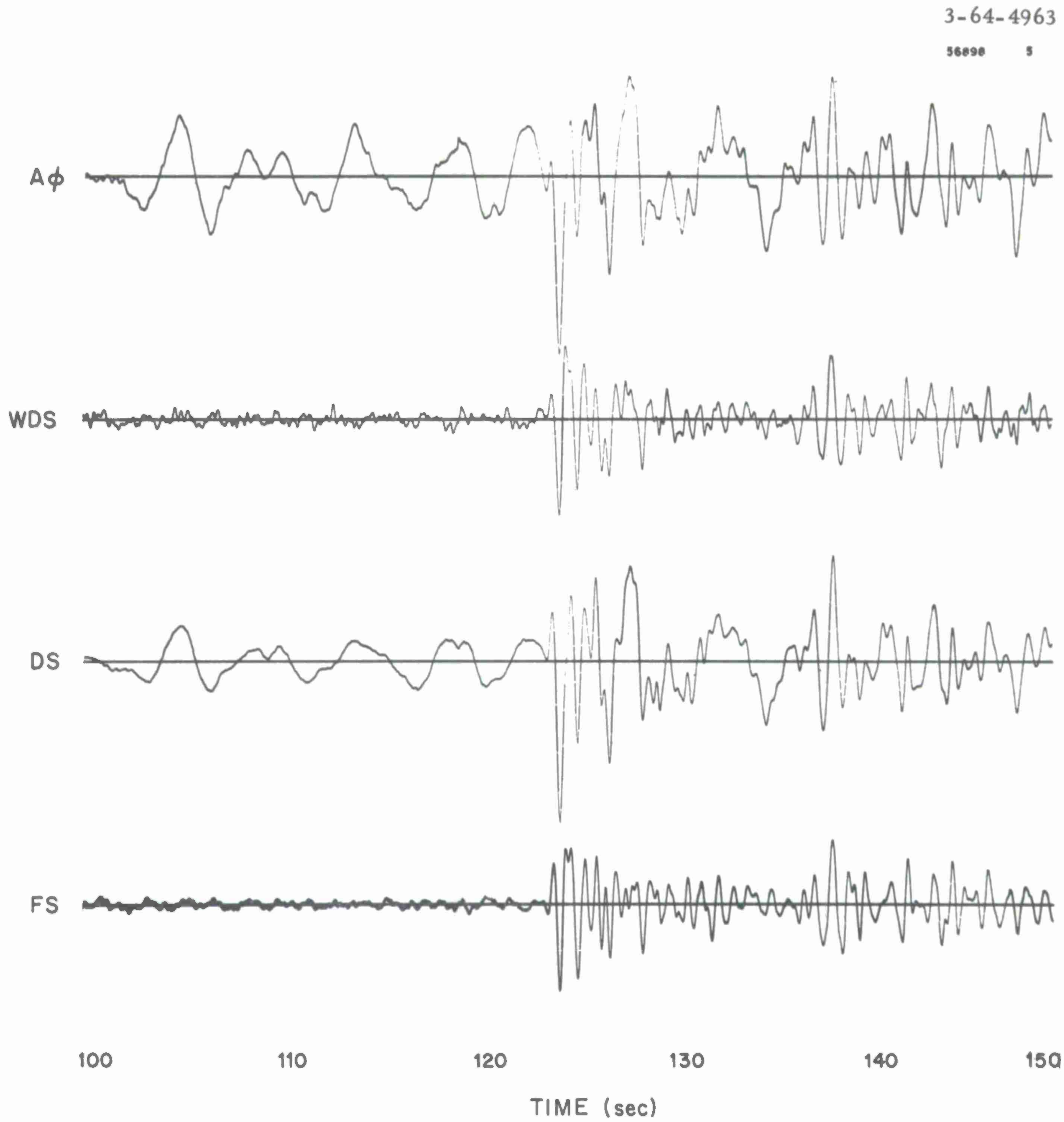

Figure 5. Processed traces obtained by frequency-domain synthesis procedure, 11/11/65 Rat Island event, subarray AØ, NIM = 2, NFP = 21, 3 minute fitting interval.

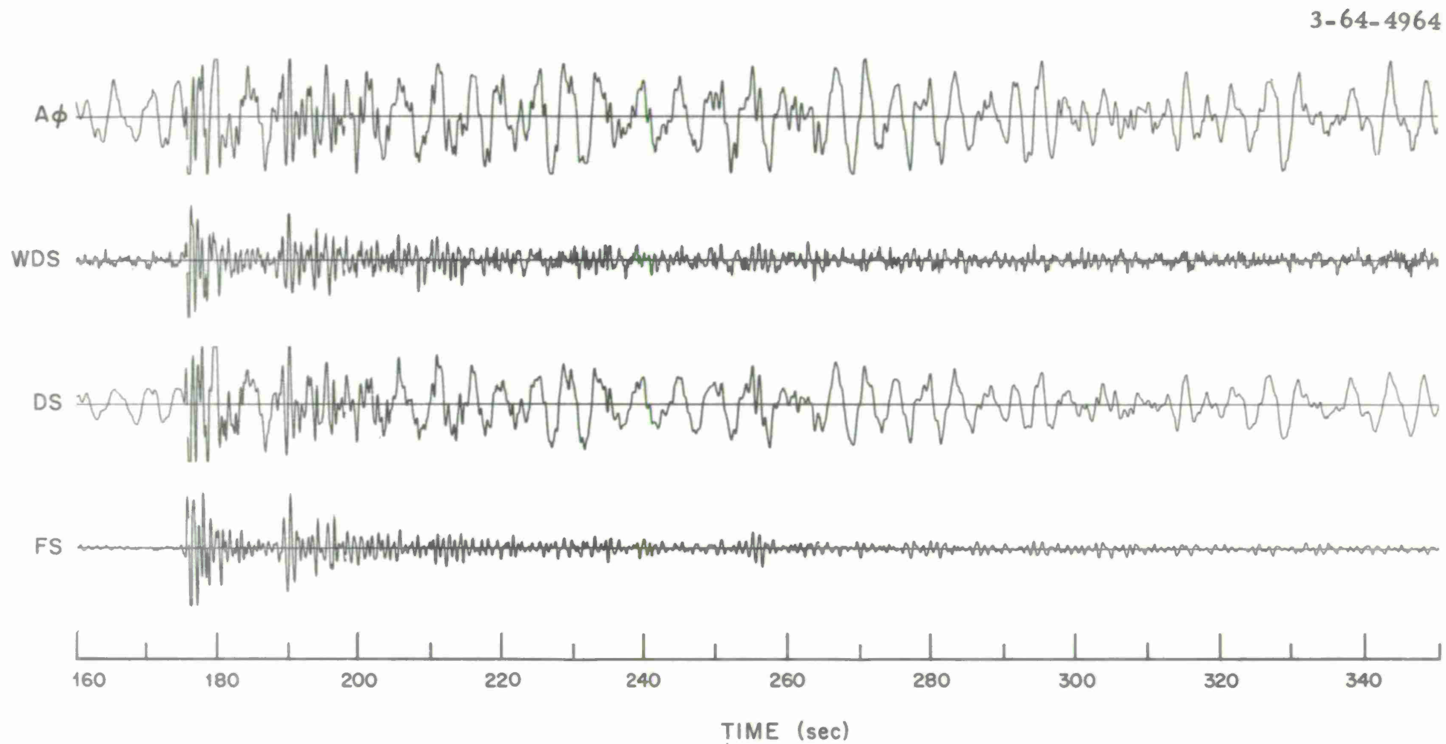

Figure 6. Processed traces obtained by time-domain synthesis procedure, 11/11/65 Rat Island event, subarray AØ, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4965

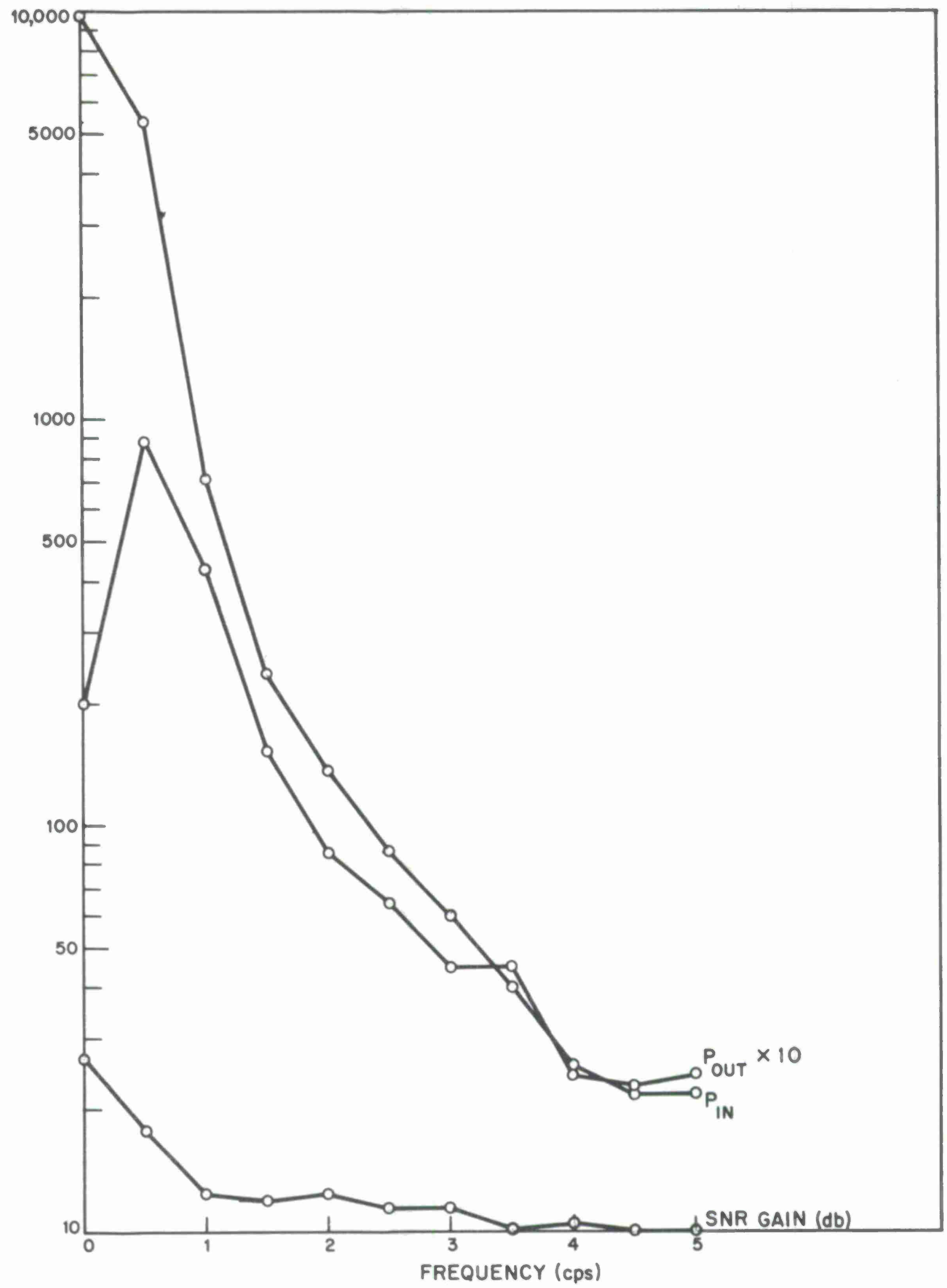

Figure 7. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray B3, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4966

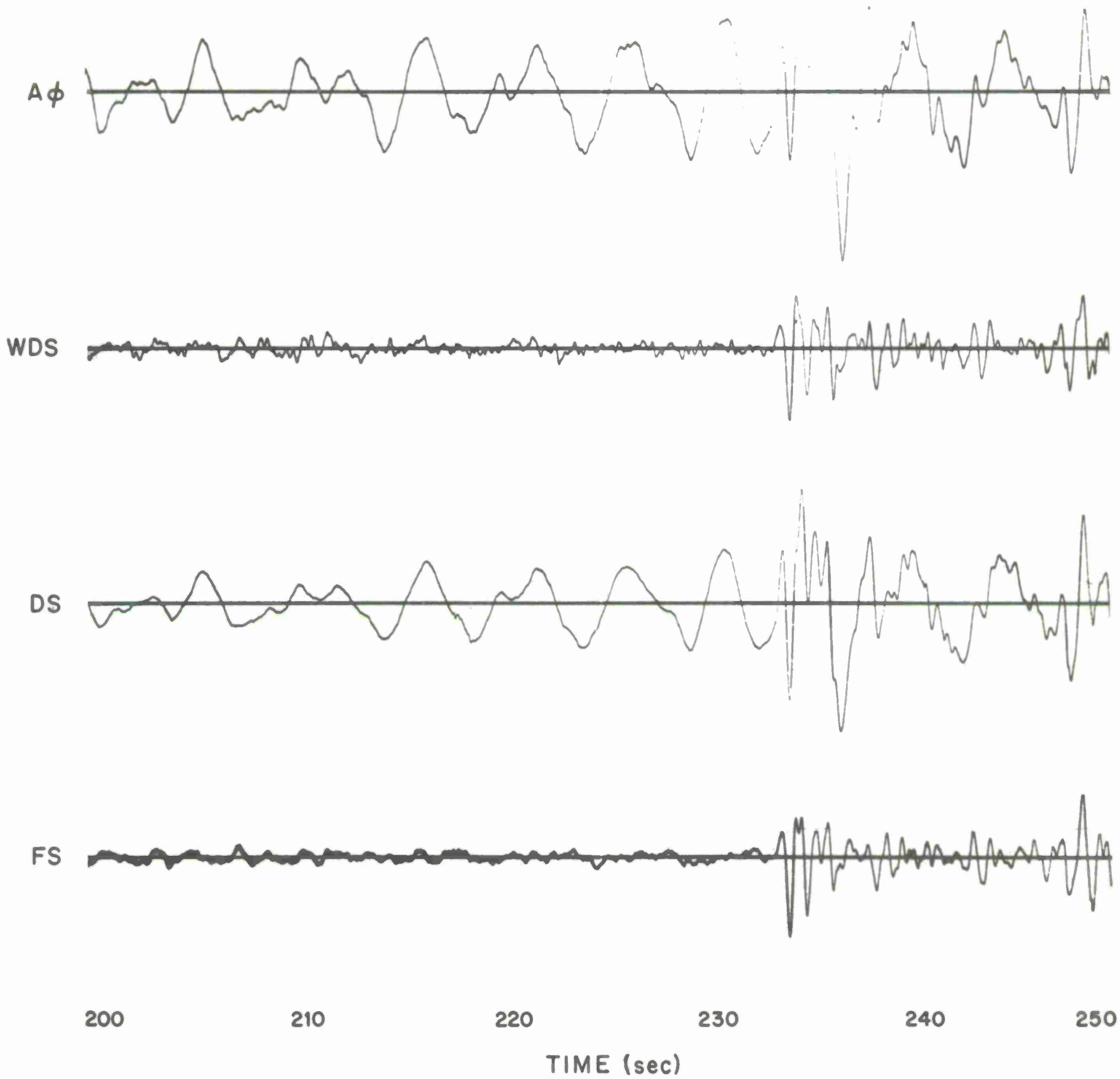

Figure 8. Processed traces obtained by frequency-domain synthesis procedure, 11/11/65 Rat Island event, subarray B3, NIM = 2, NFP = 21, 3 minute fitting interval.

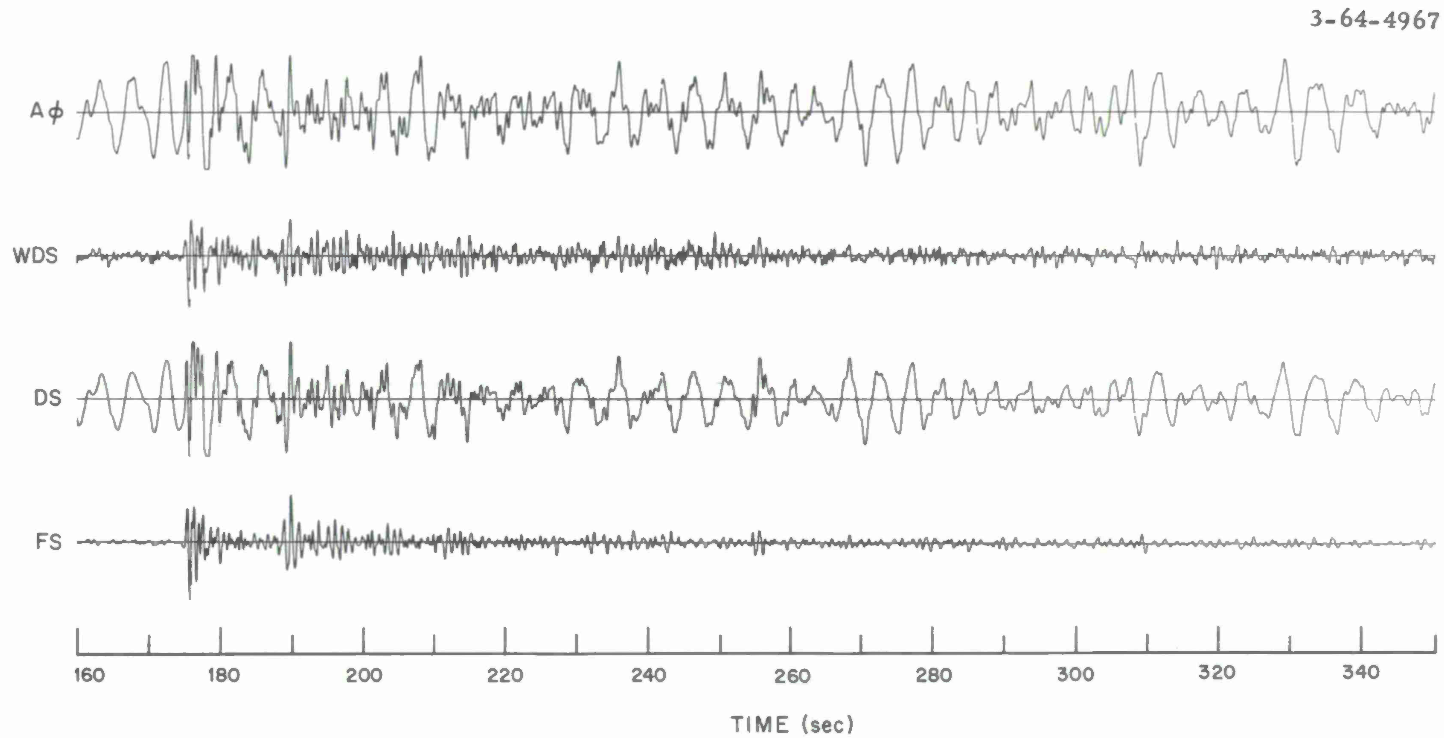

Figure 9. Processed traces obtained by time-domain synthesis procedure, 11/11/65 Rat Island event, subarray B3, NIM = 2, NFP = 21, 3 minute fitting interval.

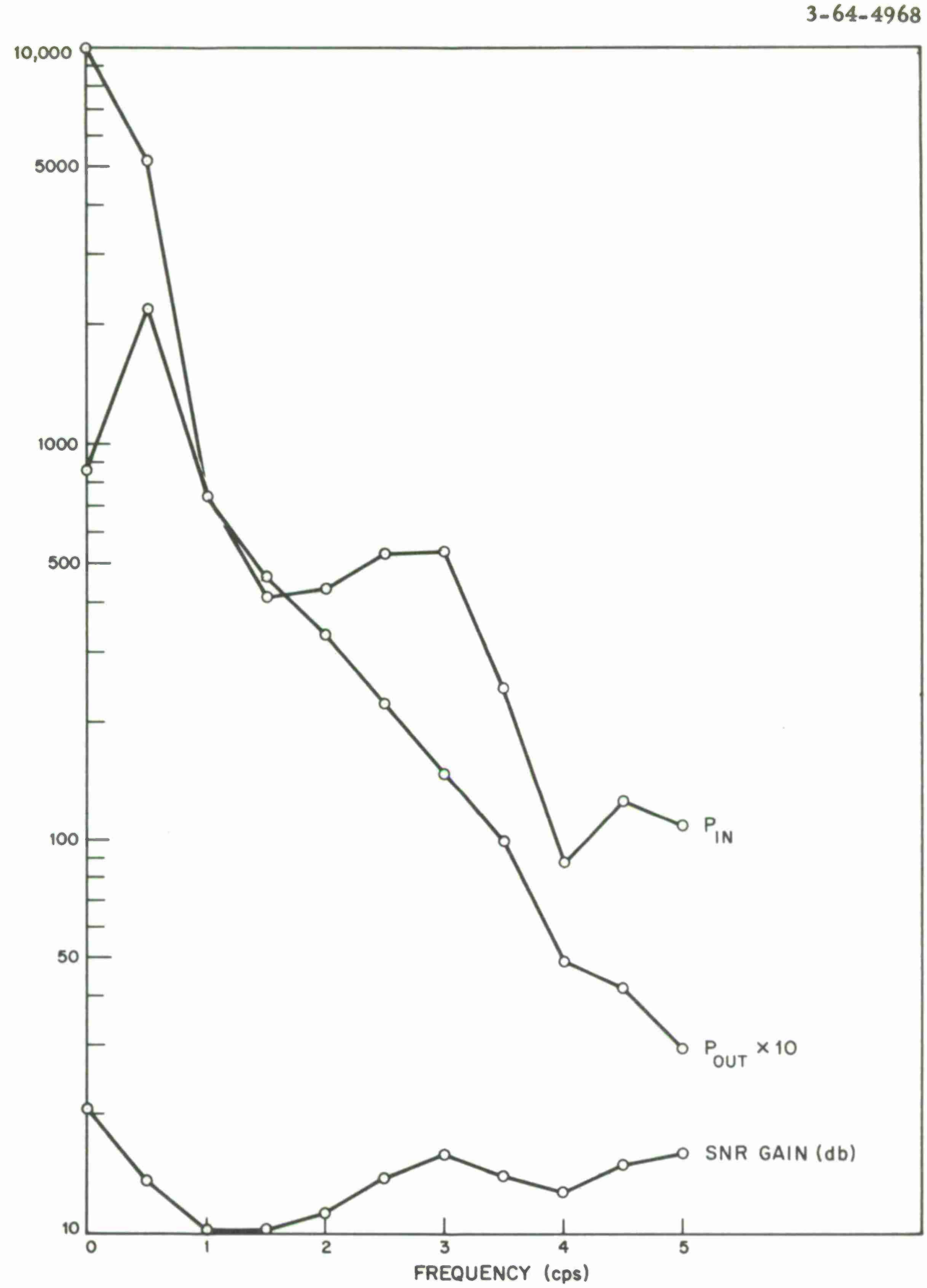

Figure 10. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray B4, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4969

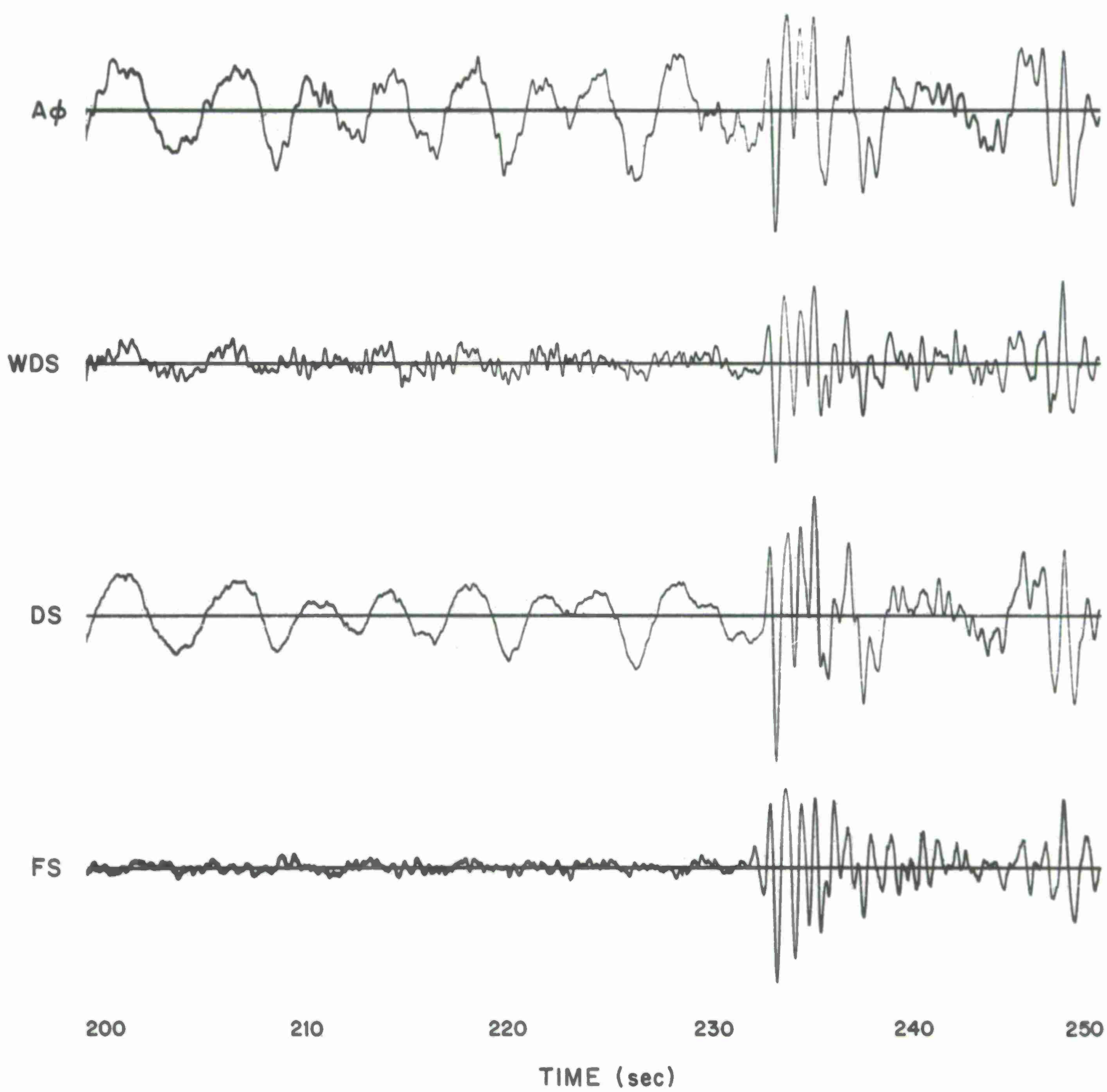

Figure 11. Processed traces obtained by frequency-domain synthesis procedure, 11/11/65 Rat Island event, subarray B4, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4970

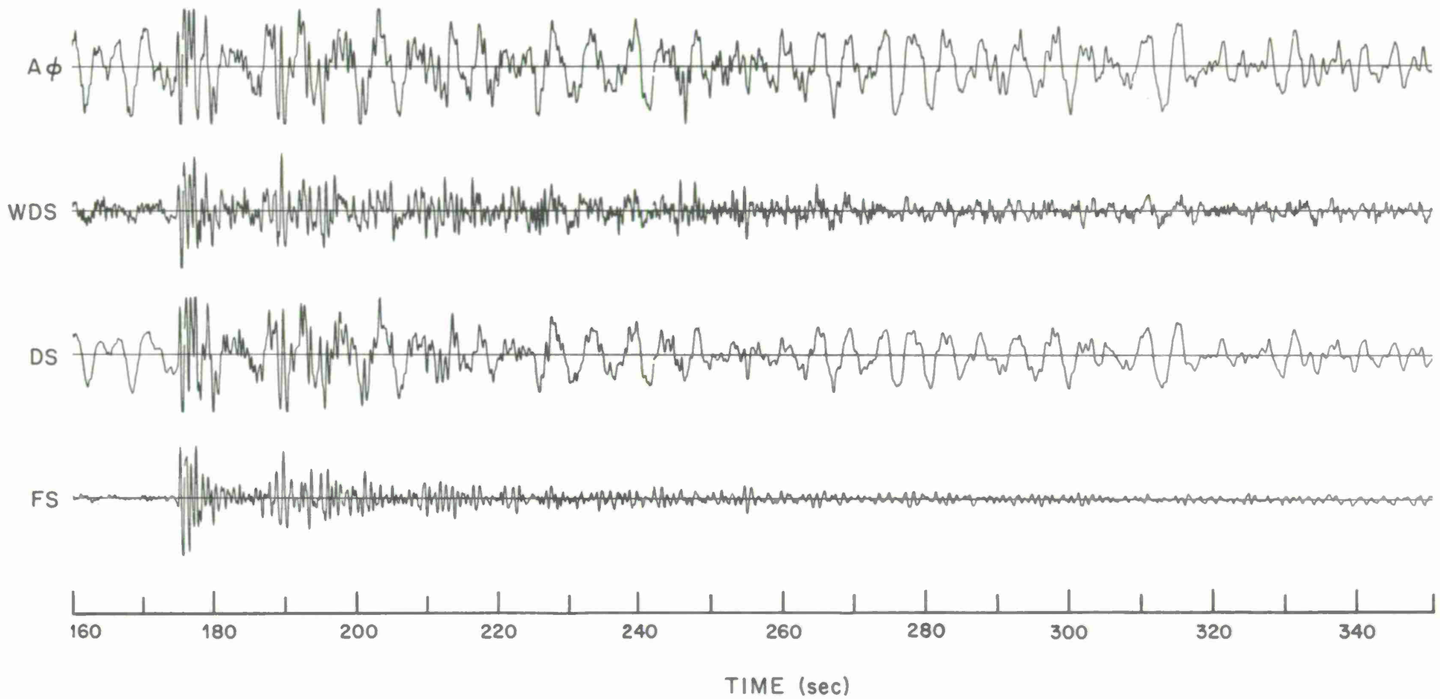

Figure 12. Processed traces obtained by time-domain synthesis procedure, 11/11/65 Rat Island event, subarray B4, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4971

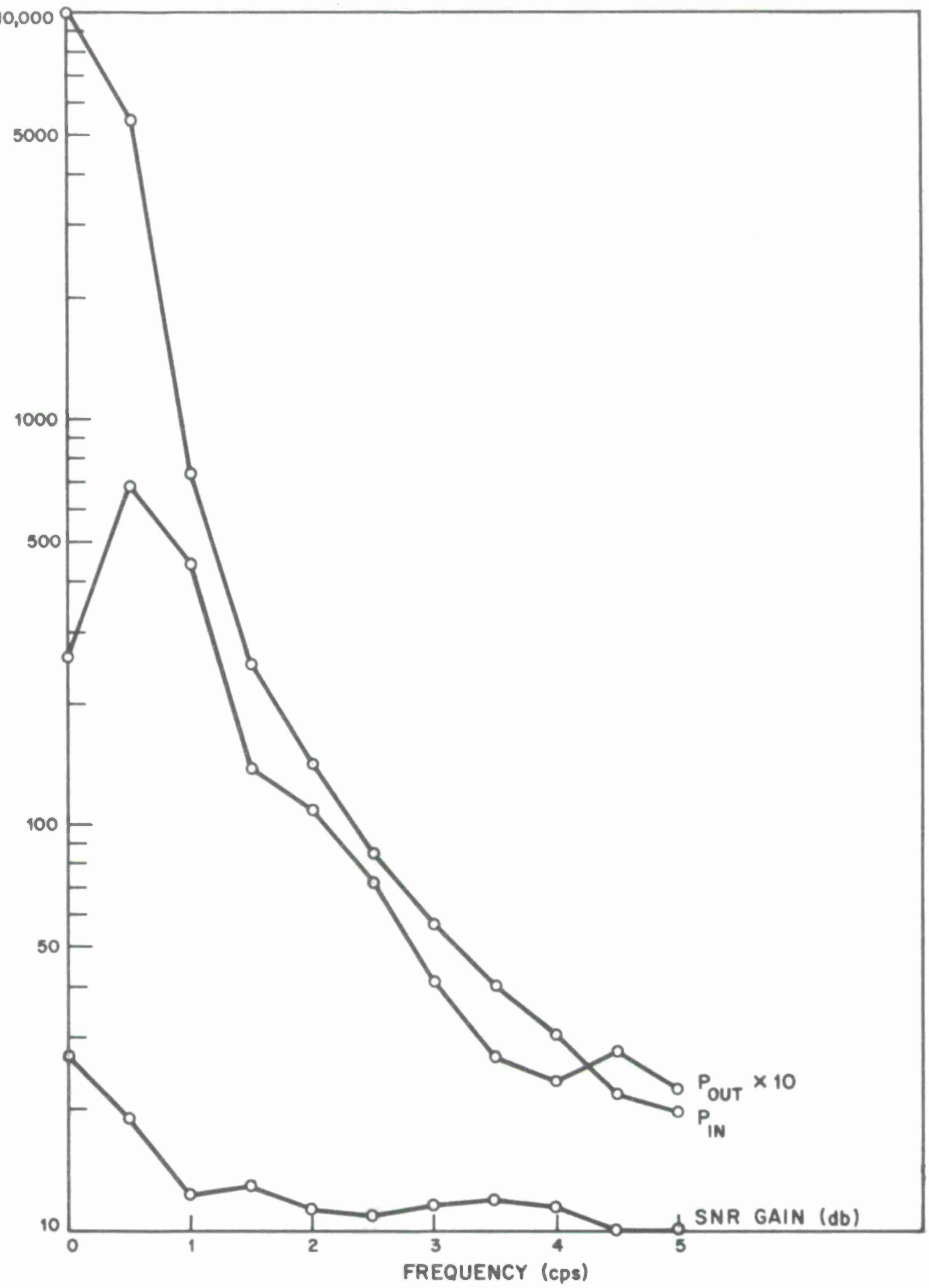

Figure 13. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray B2, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4972

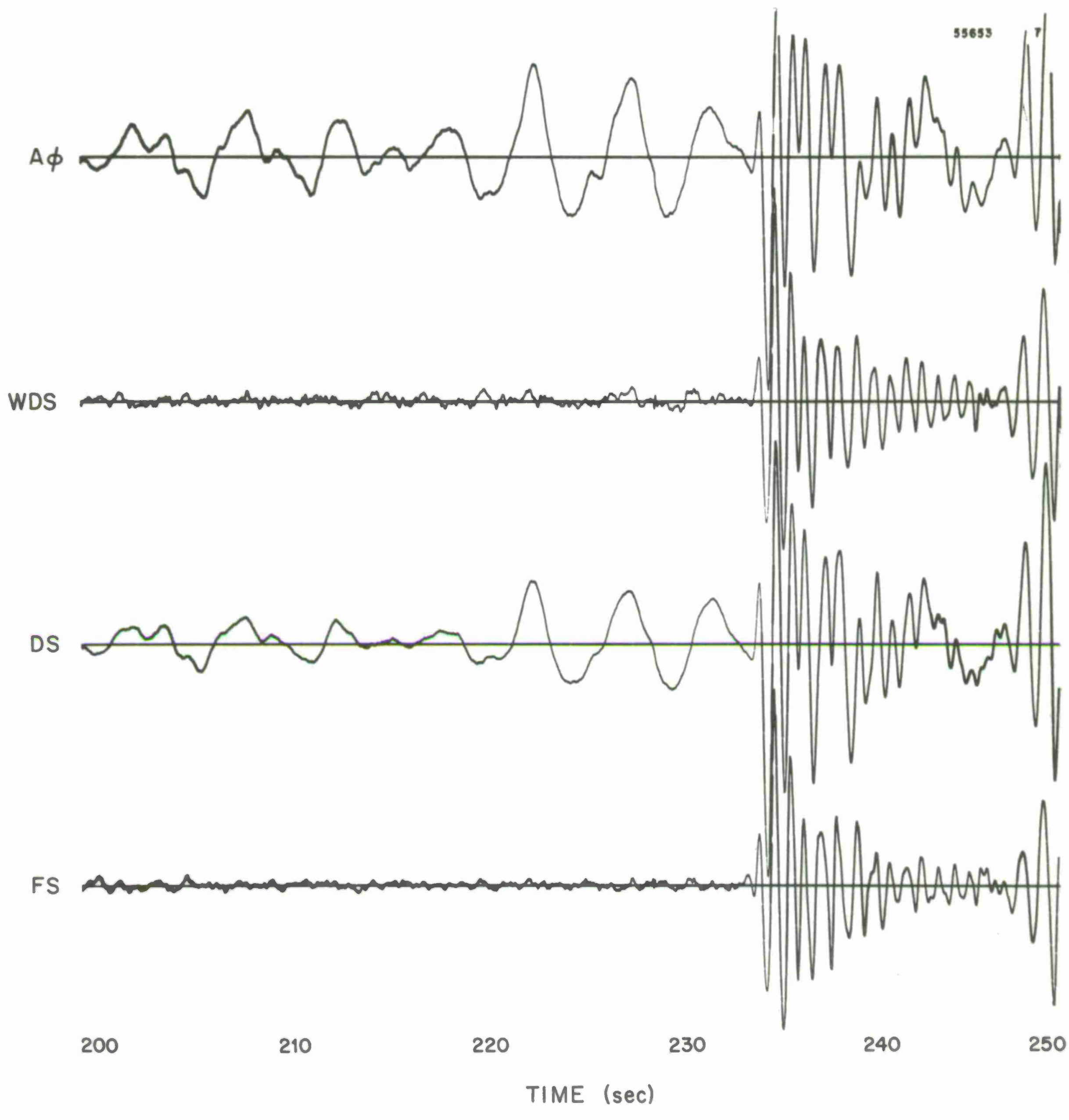

Figure 14. Processed traces obtained by frequency-domain synthesis procedure, 11/11/65 Rat Island event, subarray B2, NIM = 2, NFP = 21, 3 minute fitting interval.

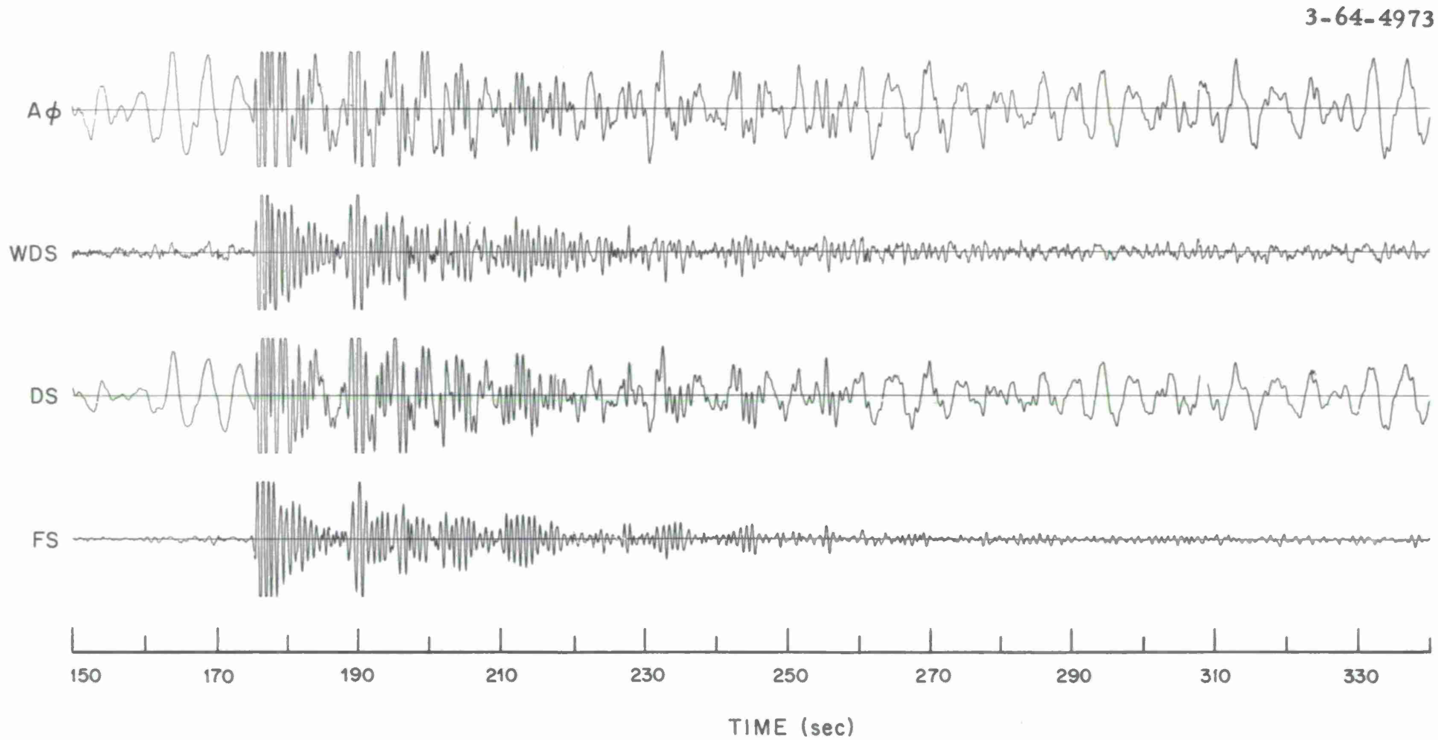

Figure 15. Processed traces obtained by time-domain synthesis procedure, 11/11/65 Rat Island event, subarray B2, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4974

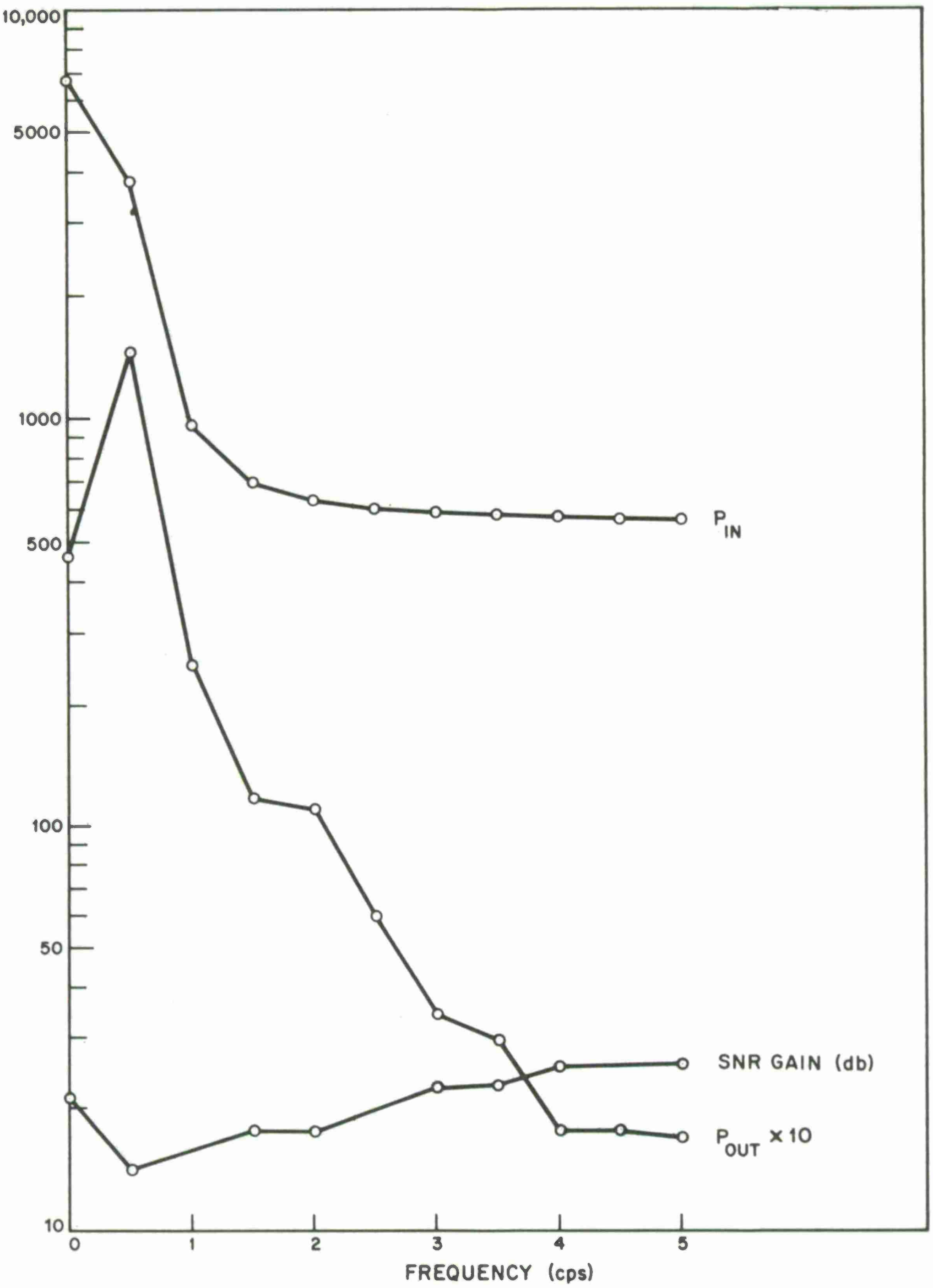

Figure 16. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 2/1/66 Central Kazakh event, subarray B1, NIM = 2, NFP = 21, 3 minute fitting interval.

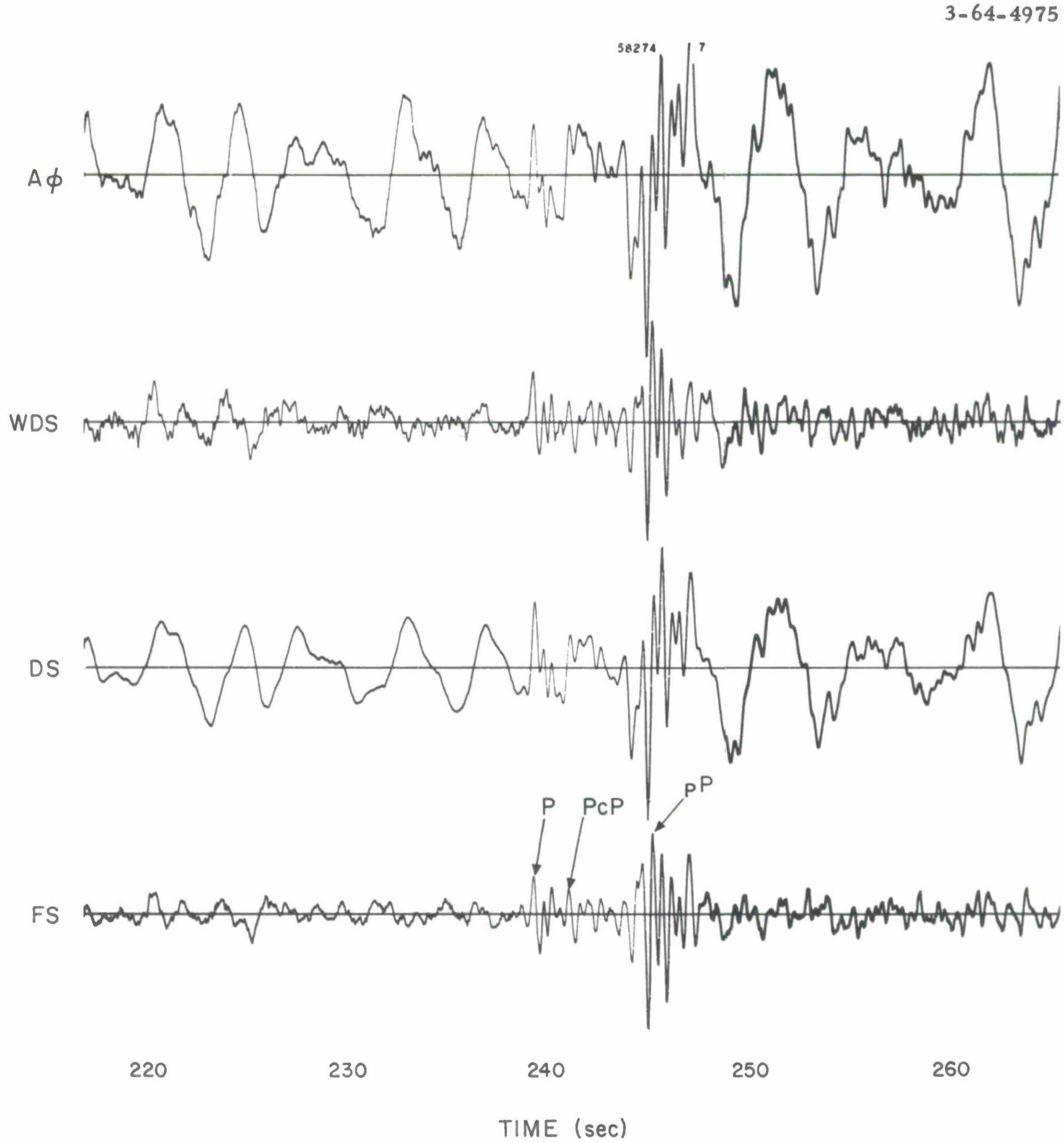

Figure 17. Processed traces obtained by frequency-domain synthesis procedure, 2/1/66 Central Kazakh event, subarray B1, NIM = 2, NFP = 21, 3 minute fitting interval.

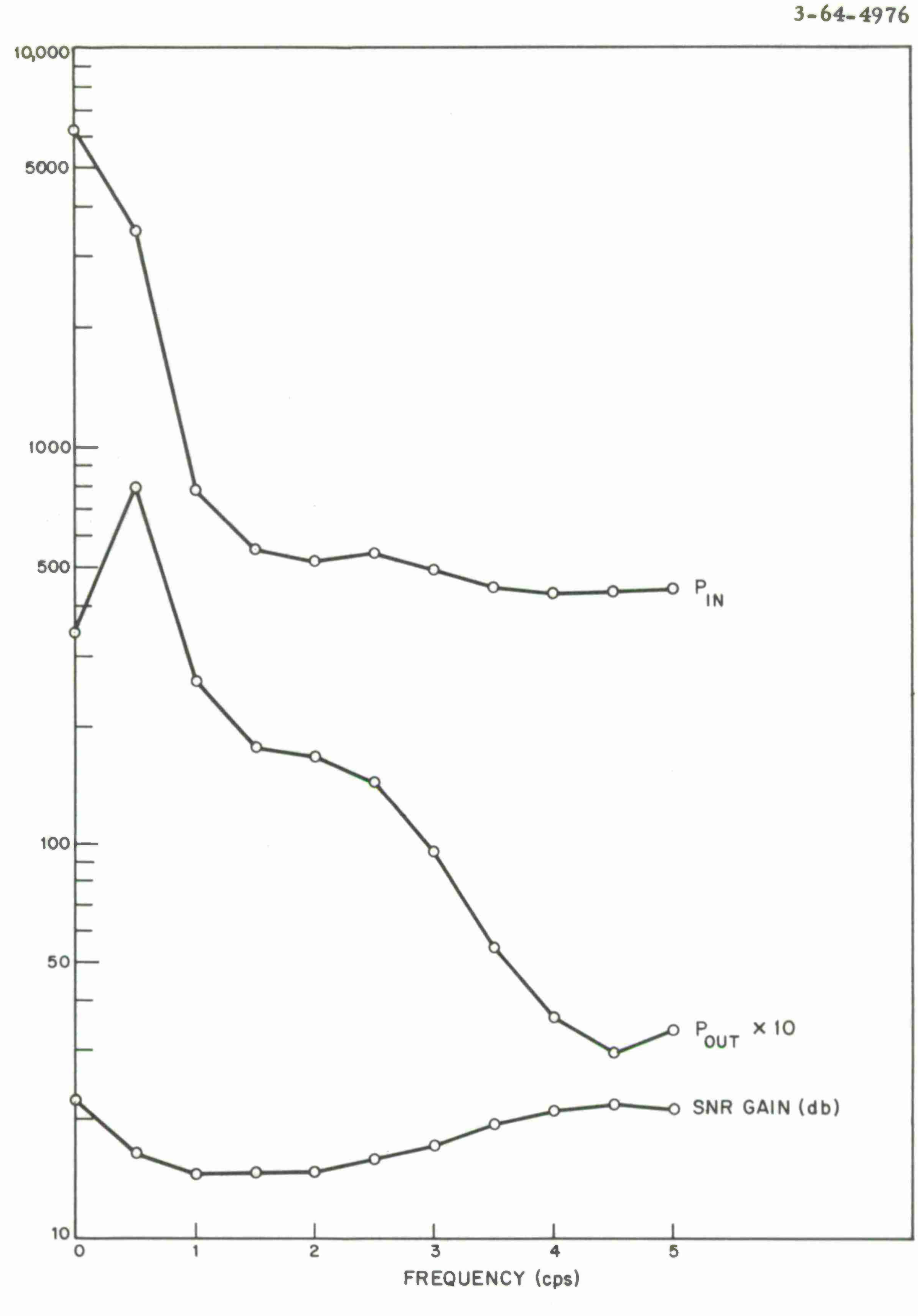

Figure 18. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 2/1/66 Central Kazakh event, subarray AØ, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4977

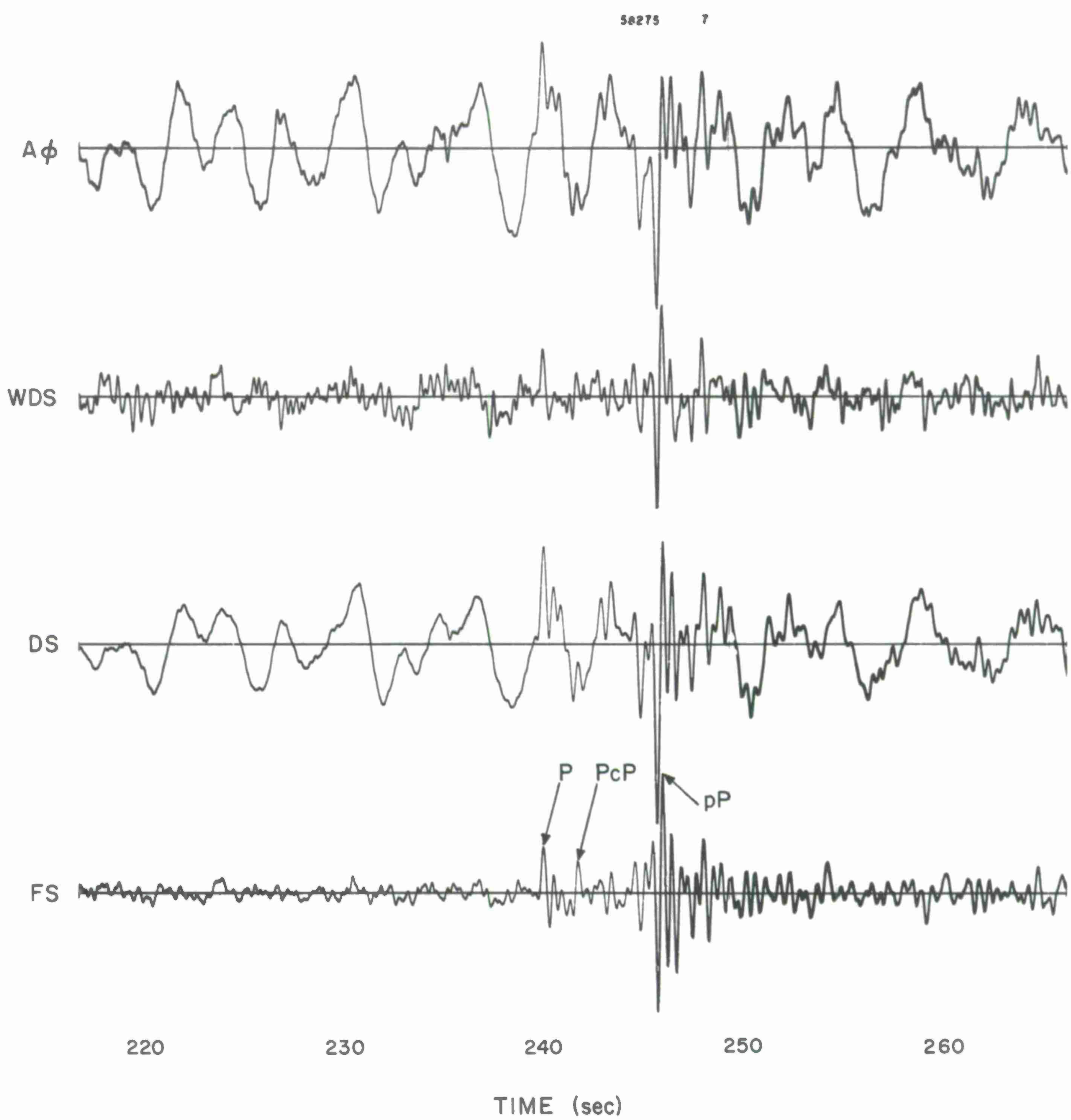

Figure 19. Processed traces obtained by frequency-domain synthesis procedure, 2/1/66 Central Kazakh event, subarray AØ, NIM = 2, NFP = 21, 3 minute fitting interval.

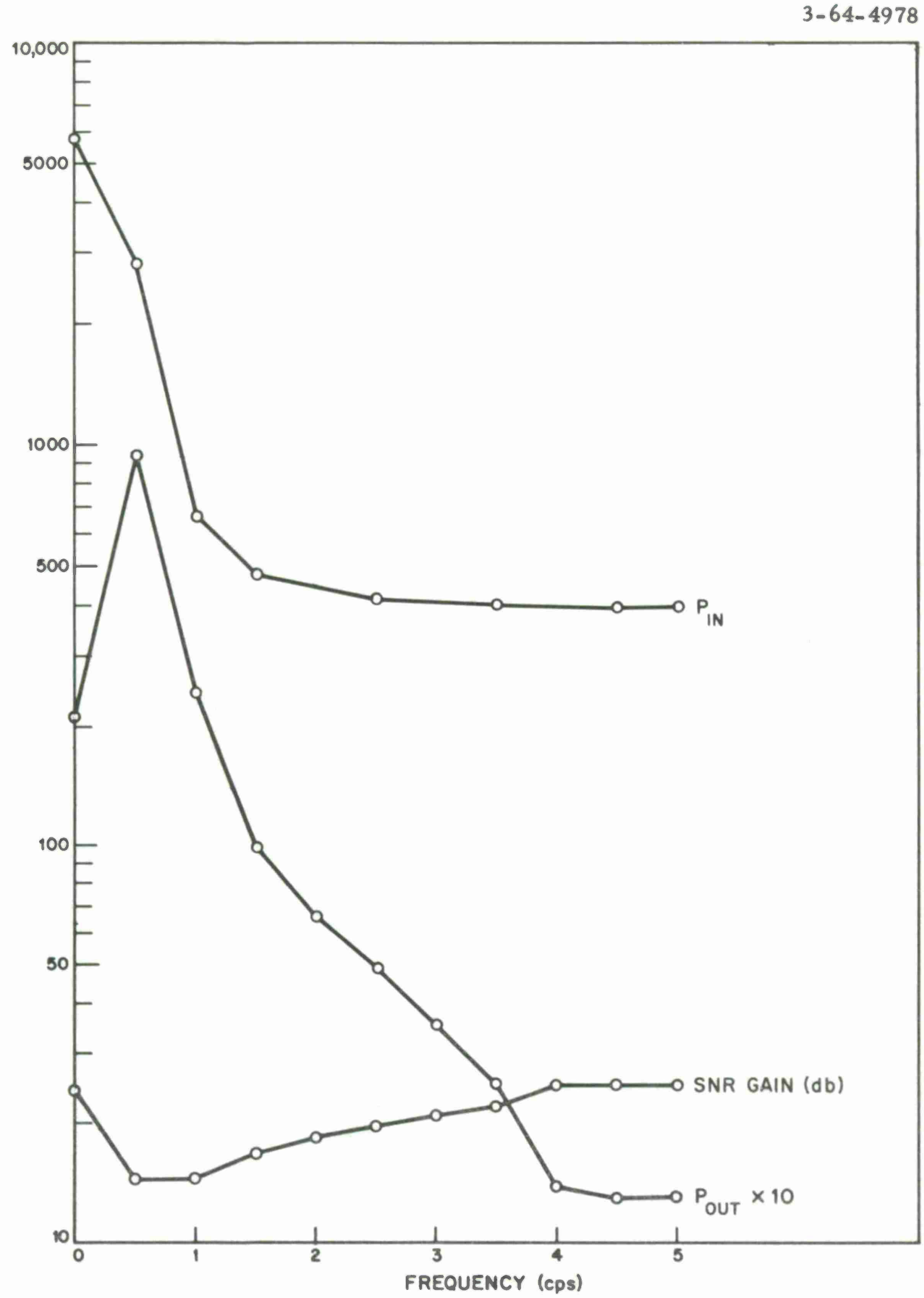

Figure 20. Input power, output power, and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 2/1/66 Central Kazakh event, subarray B3, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4979

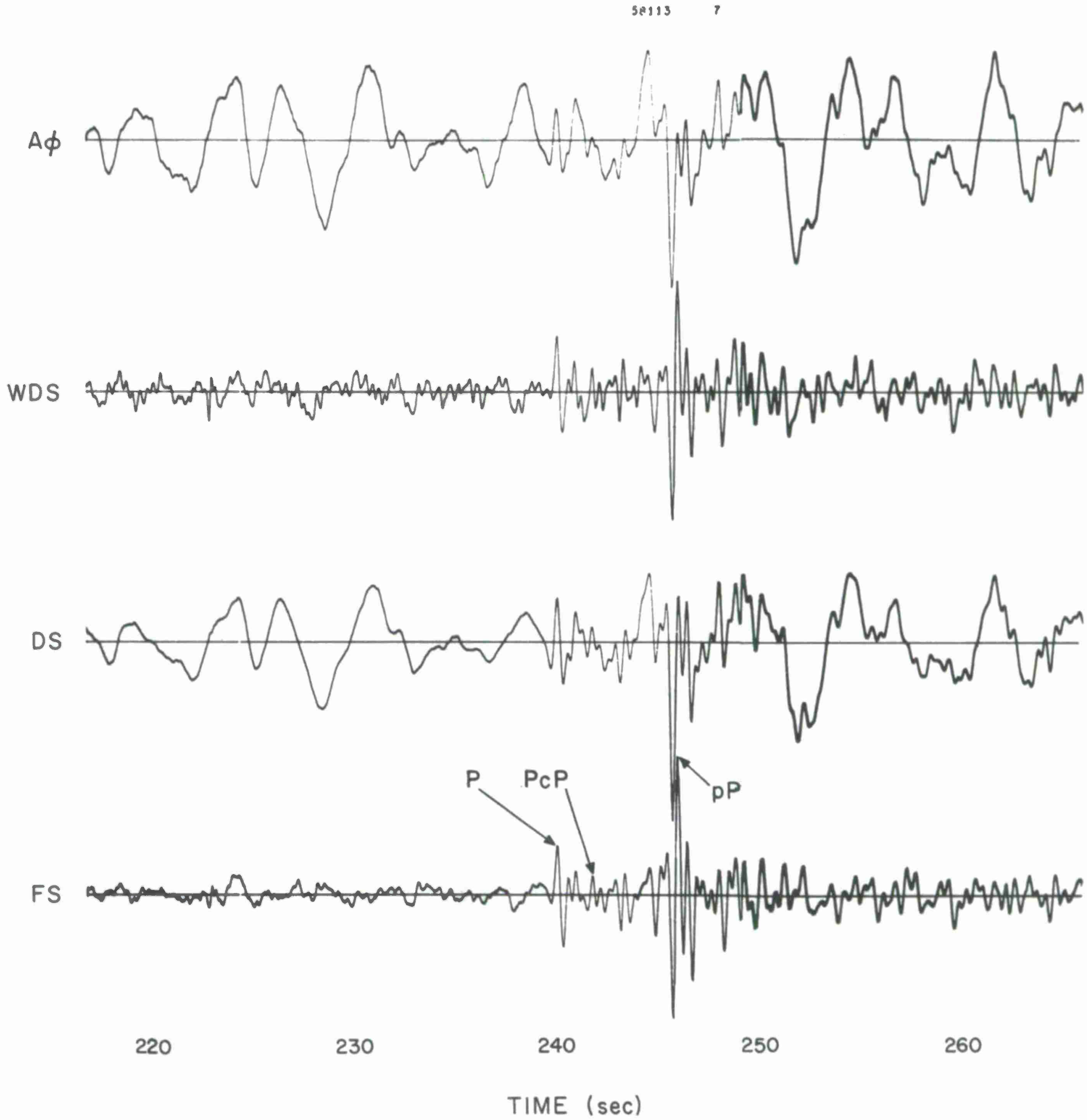

Figure 21. Processed traces obtained by frequency-domain synthesis procedure, 2/1/66 Central Kazakh event, subarray B3, NIM = 2, NFP = 21, 3 minute fitting interval.

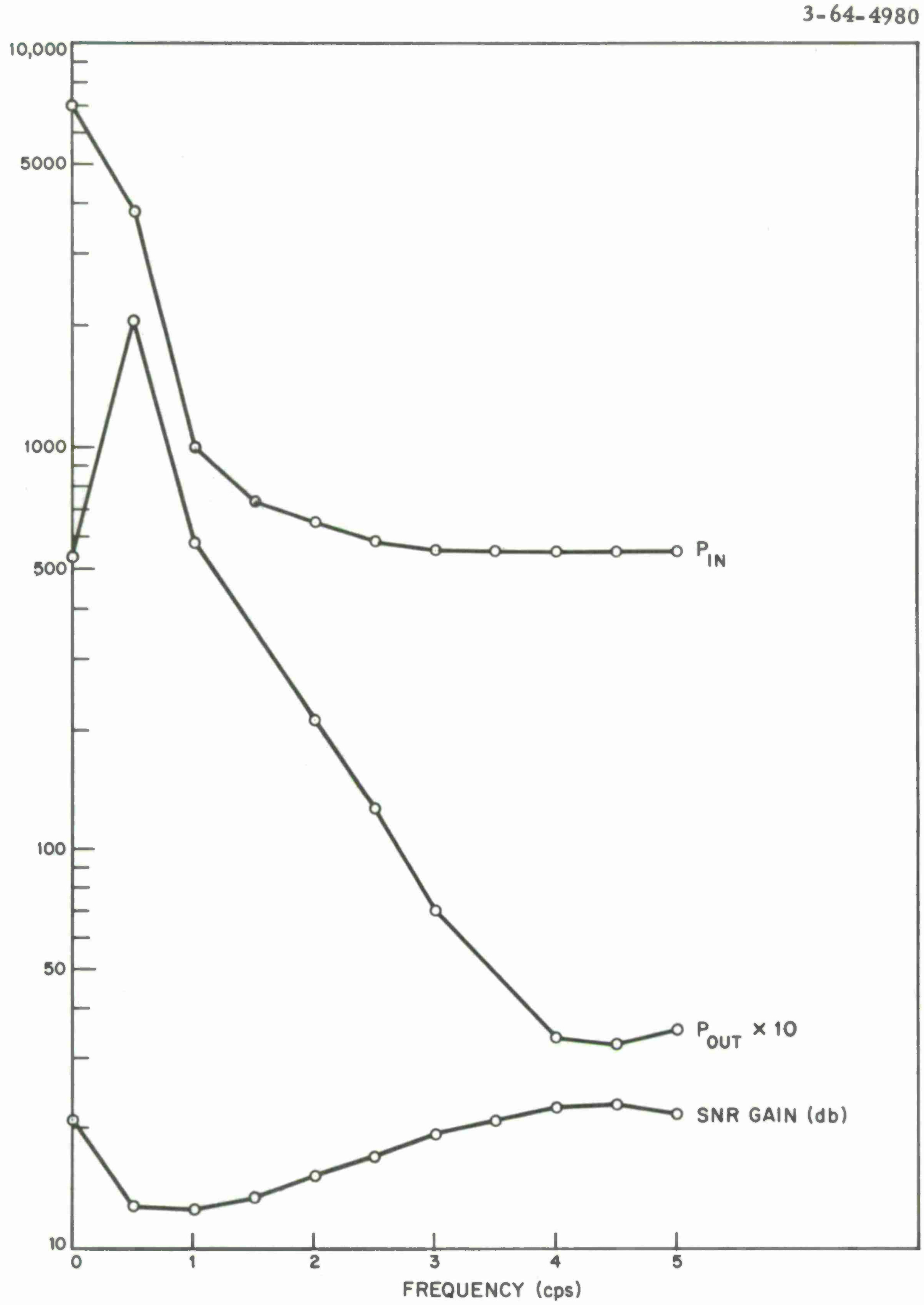

Figure 22. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 2/1/66 Central Kazakh event, subarray B4, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4981

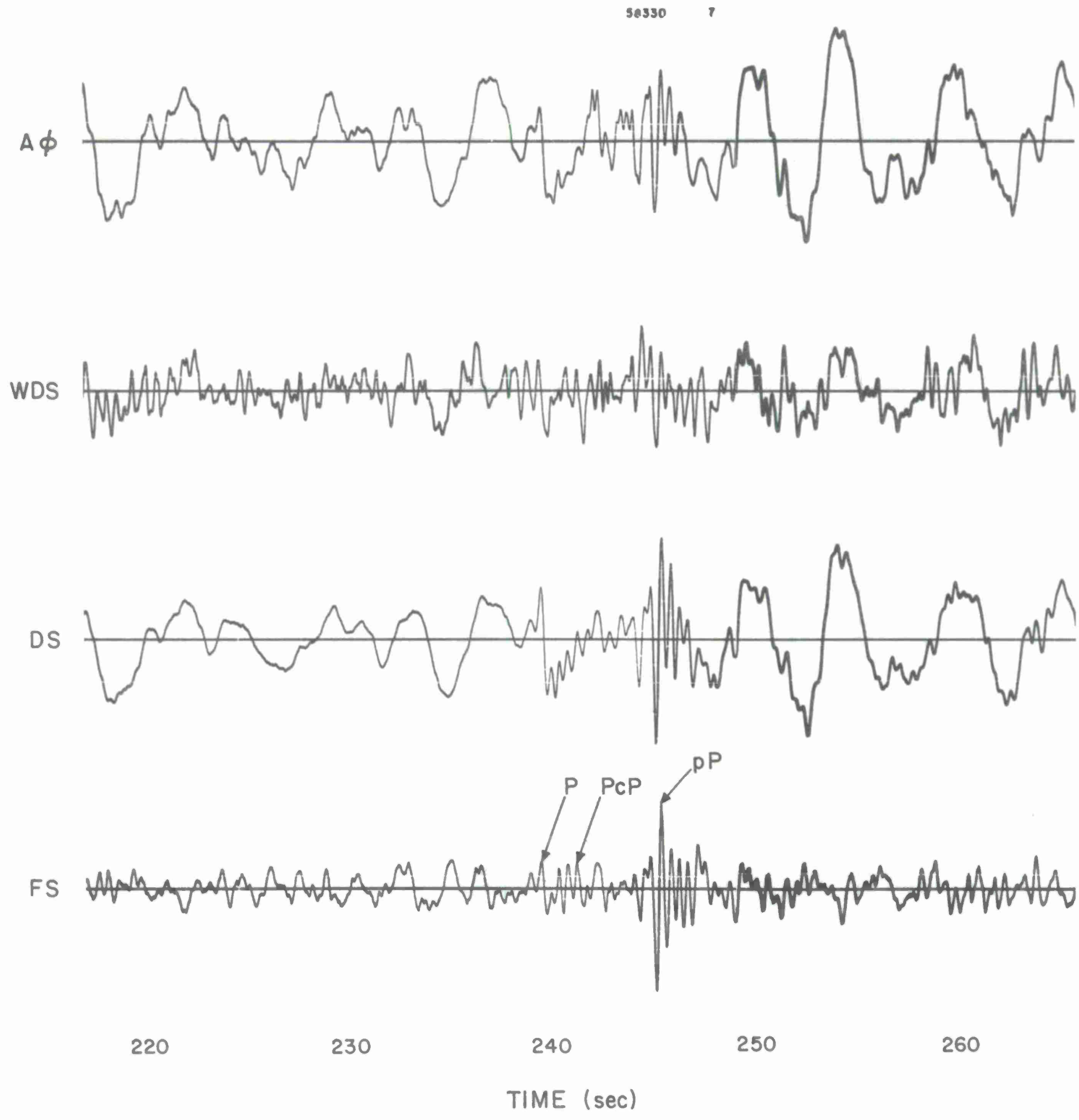

Figure 23. Processed traces obtained by frequency-domain synthesis procedure, 2/1/66 Central Kazakh event, subarray B4, NIM = 2, NFP = 21, 3 minute fitting interval.

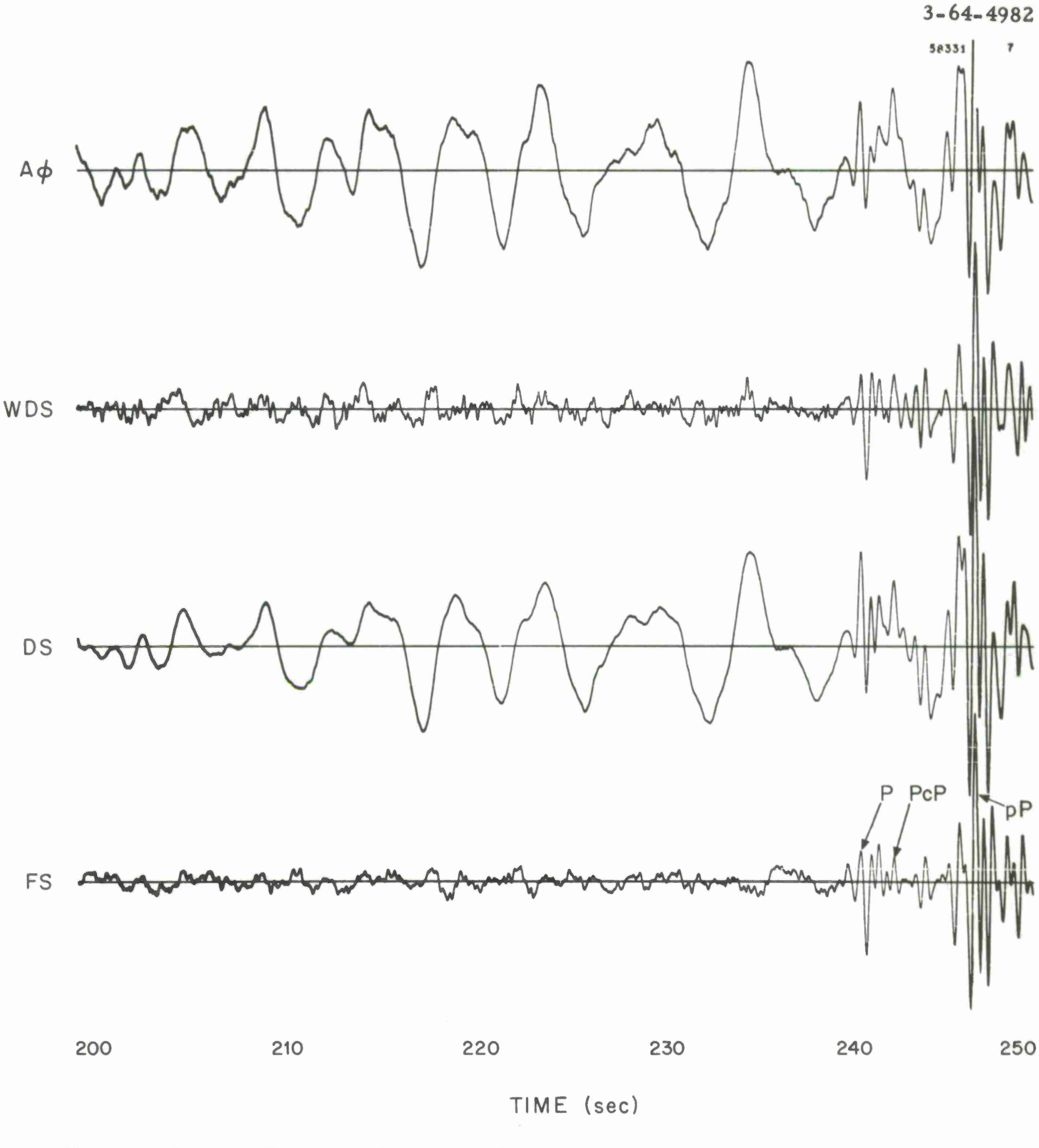

Figure 24. Processed traces obtained by frequency-domain synthesis procedure, 2/1/66 Central Kazakh event, subarray B2, NIM = 2, NFP = 21, 3 minute fitting interval.

3-64-4983

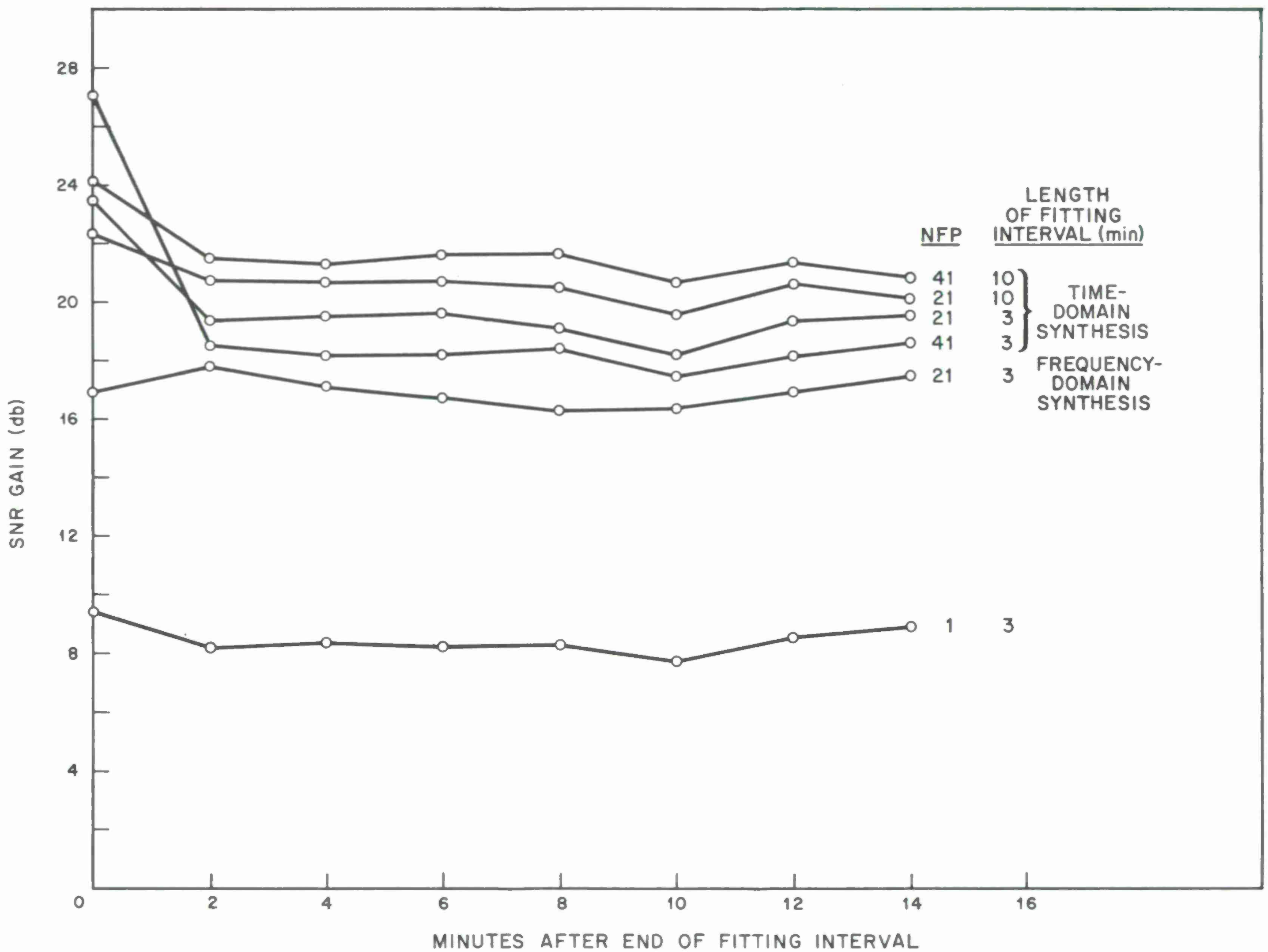

Figure 25. Signal-to-noise ratio gain of maximum-likelihood filter outside the fitting interval for various NFP and fitting interval lengths, noise from 2/4/66, subarray AØ, Azimuth = 0^{o}, Horizontal Velocity = 15 km/sec, SNR gain measured over 2 minute intervals.

3-64-4984

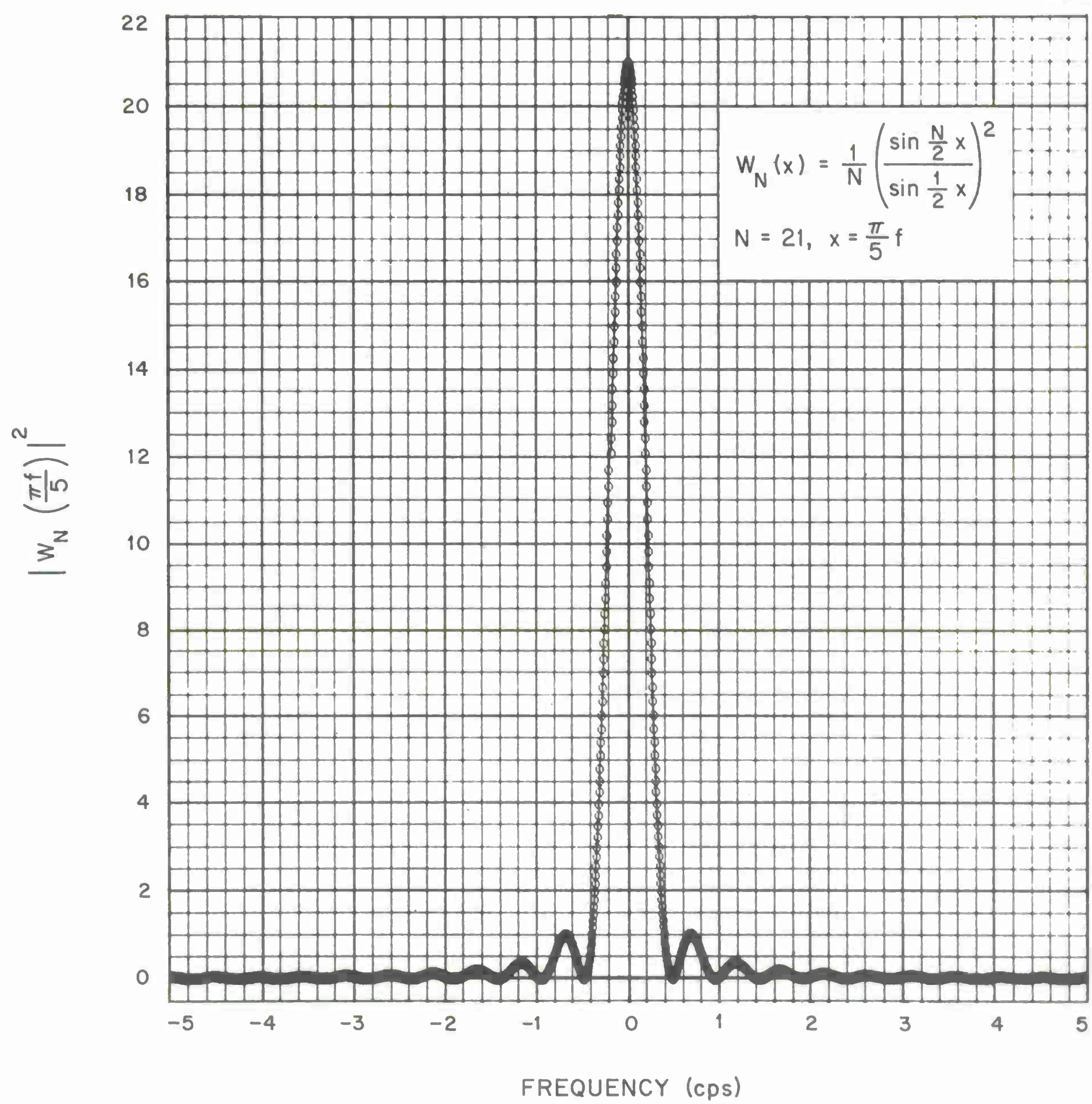

Figure 26. Plot of frequency window $[W_N(\pi f/5)]^2$, N = 21.

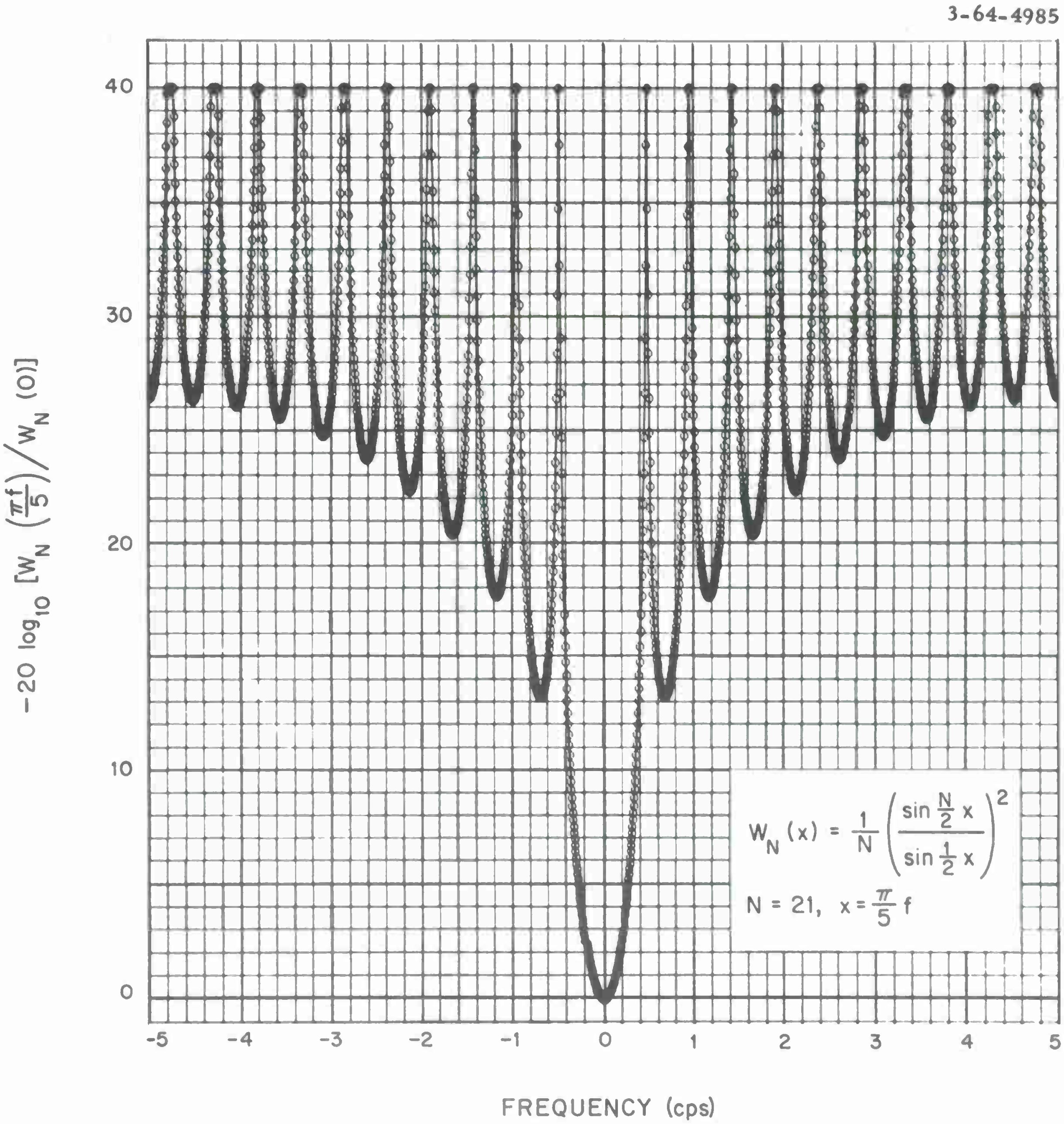

Figure 27. Plot of frequency window $-10 \log_{10} [W_N(\pi f/5)/W_N(0)]^2$, N = 21.

3-64-5023

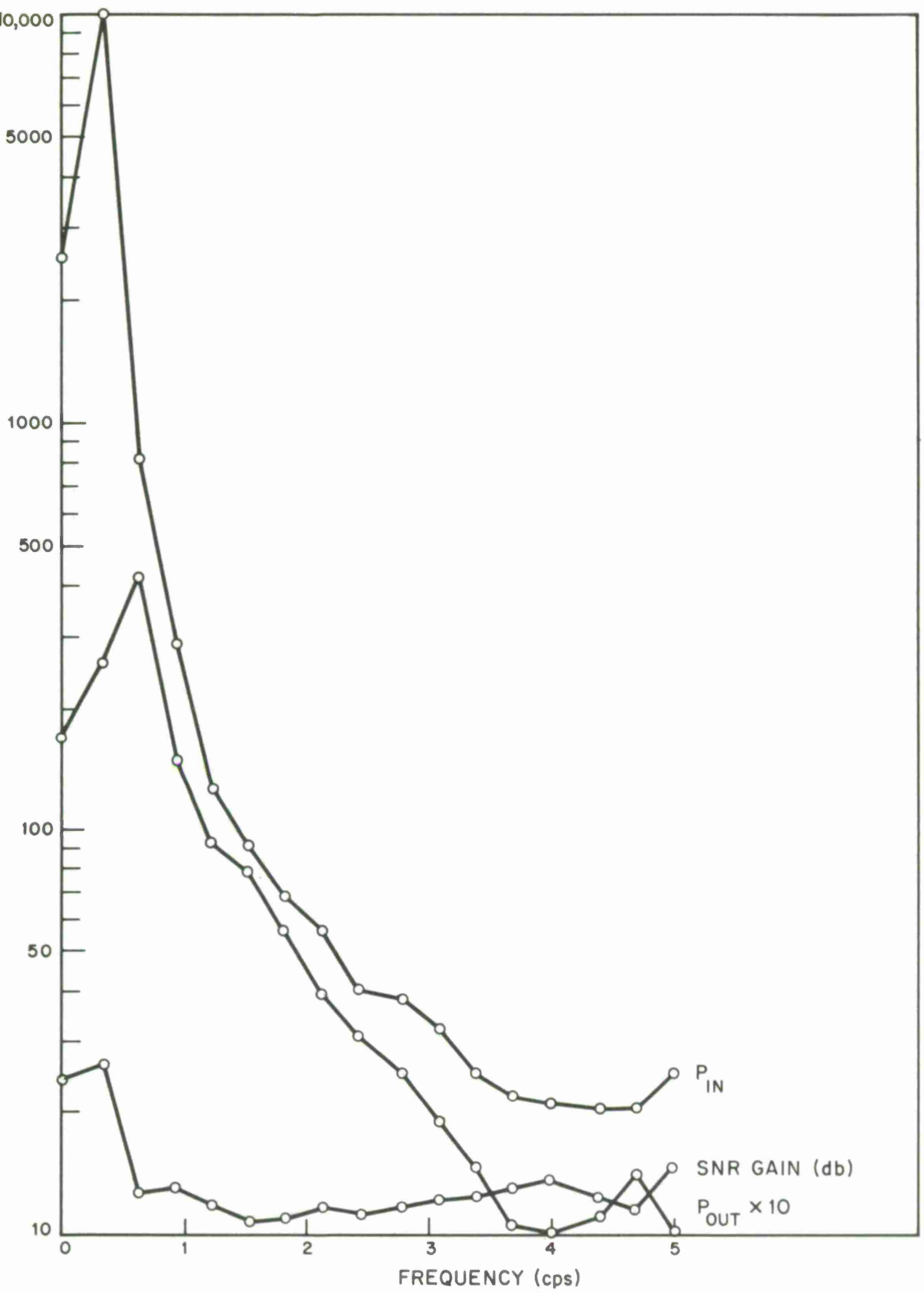

Figure 28. Input power, output power and signal-to-noise ratio gain vs. frequency for frequency-domain maximum-likelihood filter, 11/11/65 Rat Island event, subarray B1, NIM = 2, NFP = 33, 3 minute fitting interval.

DOCUMENT CONTROL DATA – R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Lincoln Laboratory, M. I. T.	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP None

3. REPORT TITLE

A Frequency-Domain Synthesis Procedure for Multidimensional Maximum-Likelihood Processing of Seismic Arrays

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Technical Note

5. AUTHOR(S) *(Last name, first name, initial)*

Capon, Jack   Greenfield, Roy J.   Kolker, Robert J.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
6 May 1966	88	9

8a. CONTRACT OR GRANT NO. AF 19 (628)-5167 b. PROJECT NO. ARPA Order 512 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Technical Note 1966-29 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* ESD-TR-66-203

10. AVAILABILITY/LIMITATION NOTICES

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
None	Advanced Research Projects Agency, Department of Defense

13. ABSTRACT

A frequency-domain synthesis method for multidimensional maximum-likelihood filtering of sampled data obtained from seismic arrays is presented. This procedure is shown to possess several advantages relative to the time-domain synthesis technique. The primary advantage is that the frequency-domain method requires approximately ten times less computer time to synthesize the filter than does the time-domain technique.

The details of a direct segment method for the spectral matrix estimation required in the frequency-domain approach are presented. In addition, the bias, variance, mean square error, limiting distribution, and other properties of the spectral estimates are discussed. The details of a Fortran IV computer program implementation of the frequency-domain method are given. The experimental results obtained by processing two events recorded at the Large Aperture Seismic Array are presented as well as a comparison of the performance of the frequency-domain method relative to the time-domain synthesis technique. It is found that the processed noise power reduction for the frequency-domain method is typically about two out of a total of 20 db worse than that of the time-domain technique.

14. KEY WORDS

multidimensional filtering	seismic arrays
maximum-likelihood processing	Fortran IV
frequency-domain synthesis	LASA

Printed by
United States Air Force
L. G. Hanscom Field
Bedford, Massachusetts