

AD-A097 336

CLARKSON COLL OF TECHNOLOGY POTSDAM N Y DEPT OF ELEC--ETC F/6 9/3
SUPPRESSION OF NARROWBAND INTERFERENCE IN PSEUDO-NOISE SPREAD S--ETC(U)
FEB 81 J 6 PROAKIS, J W KETCHUM F30602-78-C-0102

UNCLASSIFIED

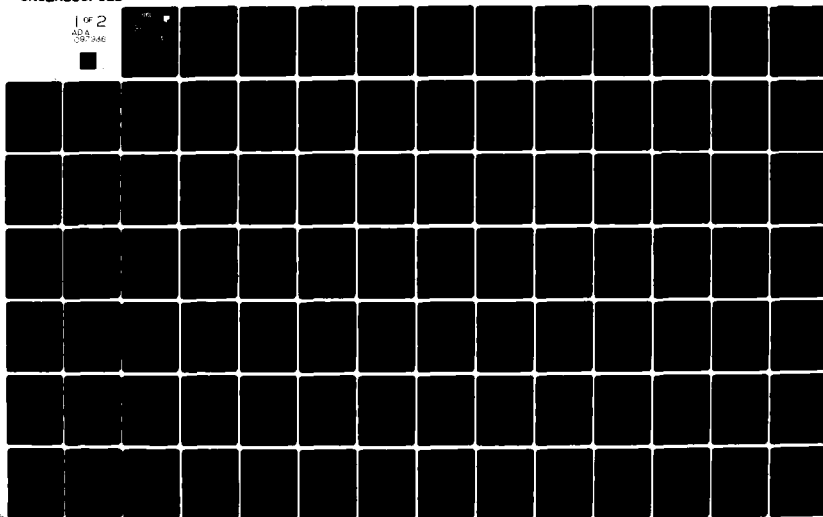
RADC-TR-81-6

NL

1 of 2

ADA

08-146



LEVEL *12*

RADC-TR-81-6
Final Technical Report
February 1981



AD A 097336

SUPPRESSION OF NARROWBAND INTERFERENCE IN PSEUDO-NOISE SPREAD SPECTRUM SYSTEMS

Northeastern University

John G. Proakis
John W. Ketchum

DTIC
ELECTRIC
APR 6 1981
S **C**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

DTIC FILE COPY

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

81 4 06 100

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-81-6 has been reviewed and is approved for publication.

APPROVED:



JOHN T. GAMBLE
Project Engineer

APPROVED:



FRED I. DIAMOND, Technical Director
Communications and Control Division

FOR THE COMMANDER:



JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (DCCL), Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19) REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADCR-TR-81-6	2. GOVT ACCESSION NO. AD-A097336	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SUPPRESSION OF NARROWBAND INTERFERENCE IN PSEUDO-NOISE SPREAD SPECTRUM SYSTEMS,		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report. 1 June 79-- 30 September 80
7. AUTHOR(s) John G./Proakis John W./Ketchum		6. PERFORMING ORG. REPORT NUMBER N/A
9. PERFORMING ORGANIZATION NAME AND ADDRESS Northeastern University Boston MA 02115.		8. CONTRACT OR GRANT NUMBER(s) F30602-78-C-0102
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (DCCL) Griffiss AFB NY 13441		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 230518P4
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same		12. REPORT DATE Feb 1981
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		13. NUMBER OF PAGES 115
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Same		15. SECURITY CLASS. (of this report) UNCLASSIFIED
18. SUPPLEMENTARY NOTES RADCR Project Engineer: Dr. John T. Gamble (DCCL)		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Algorithms Adaptive Algorithms Interference Cancellation Spread Spectrum		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report describes results of an investigation of narrowband interference suppression in a pseudo-noise (PN) spread spectrum digital communication system in which the signal is also distorted by channel multipath and fading. Techniques for determining the coefficients of a linear, interference suppression filter are described, which are based on linear prediction and conventional spectral analysis methods. Numerical results are presented on the effectiveness of the linear filter in suppressing the		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

406 744

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

interference. Error rate results, obtained via Monte Carlo simulation are presented for a receiver that employs interference suppression and equalization for a fading multipath channel.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

PREFACE

This final technical report was prepared by John G. Proakis (Principal Investigator) and John W. Ketchum of the Department of Electrical Engineering, Northeastern University, Boston, Massachusetts. It describes work performed under the RADC Post Doctoral Program, Contract No. F30602-78-C-0102., during the period June 1979 to September 1980. The technical monitor at RADC was Dr. John Gamble.

The authors wish to express their appreciation to Dr. John Gamble for the technical support and useful suggestions that he provided during the course of this work.

Approved for	
NTIS Grant	
DISTRIBUTION	
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

TABLE OF CONTENTS

ABSTRACT	i
I. INTRODUCTION	1
1.1 Mathematical Model of PN Spread Spectrum Communication System	2
II. ALGORITHMS FOR ESTIMATION AND SUPPRESSION OF NARROW BAND INTERFERENCE	7
2.1 Interference Suppression Based on Nonparametric Spectral Estimates	7
2.2 Interference Suppression Based on Linear Prediction	9
2.3 Algorithms for Computing the Prediction Coefficients	15
2.4 Lattice Structure for Linear Prediction Filters	22
2.5 Generalization of Linear Pre- diction Algorithms	28
III. PERFORMANCE RESULTS ON INTERFERENCE SUPPRESSION	39
3.1 SNR Improvement Factor Resulting from Interference Suppression	39
3.2 Characteristics of the Inter- ference Suppression Filter	44
3.3 Performance of Interference Suppression Filter Based on Nonparametric Spectral Estimates	66
3.4 Performance of Equalized PN Spread Spectrum System	77
REFERENCES	103

LIST OF ILLUSTRATIONS

Figure 2.1	Lattice Filter -----	25
Figure 2.2	An M-Step Linear Predictor -----	29
Figure 2.3	A Lattice Stage for the Generalized Levinson M-Step Linear Predictor -----	34
Figure 2.4	A Lattice Stage for the Generalized Burg M-Step Linear Predictor -----	36
Figure 3.1	Improvement Factor for Single-Band Interference -----	45
Figure 3.2	Improvement Factor for Single-Band Interference - Simulation Data -----	47
Figure 3.3	Improvement Factor for Sinusoidal Interference -----	48
Figure 3.4	Improvement Factor as a Function of Filter Order for Multi-Band Interference -----	50
Figure 3.5	Improvement Factor for Eight-Tap and Sixteen-Tap Filters -----	51
Figure 3.6	Frequency Response for Eight-Tap Filter -----	52
Figure 3.7	Frequency Response for Sixteen-Tap Filter -----	53
Figure 3.8	Frequency Response for 29-Tap Filter -----	54
Figure 3.9	Improvement Factor as a Function of Filter Order for 8-Band and 16- Band Interference -----	56
Figure 3.10	Improvement Factor for 40-Tap and 80-Tap Filters -----	57
Figure 3.11	Frequency Response for 8-Tap Filter -----	58
Figure 3.12	Frequency Response for 16-Tap Filter -----	59
Figure 3.13	Frequency Response for 32-Tap Filter -----	60

Figure 3.14	Frequency Response for 48-Tap Filter -----	61
Figure 3.15	Position of Zeros for 2, 4, 6, 8, 10, 12-Order Predictors -----	63
Figure 3.16	Improvement Factor for Predictor in Cascade with its Matched Filter -----	65
Figure 3.17	Improvement Factor Obtained with Linear Prediction Algorithms -----	67
Figure 3.18	Frequency Response of Filter Designed from Least Squares Algorithm -----	68
Figure 3.19	Frequency Response of Filter Designed from Burg Algorithm -----	69
Figure 3.20	Frequency Response of Filter Designed from Levinson Algorithm -----	70
Figure 3.21	Estimate of Power Spectral Density -----	72
Figure 3.22	Frequency Response of 15-Tap Filter -----	73
Figure 3.23	Estimate of Power Spectral Density -----	74
Figure 3.24	Frequency Response of 15-Tap Filter -----	75
Figure 3.25	Improvement Factor for 15-Tap Filter -----	76
Figure 3.26	Frequency Response of 4-Tap Filter for a Multipath Spread of Two Chips -----	79
Figure 3.27	Frequency Response of 4-Tap Filter for a Multipath Spread of Two Chips -----	80
Figure 3.28	Frequency Response of 4-Tap Filter for a Multipath Spread of Two Chips -----	81
Figure 3.29	Frequency Response of 4-Tap Filter for a Multipath Spread of Two Chips -----	82
Figure 3.30	Frequency Response of 4-Tap Filter for a Multipath Spread of Two Chips -----	83
Figure 3.31	Frequency Response of 4-Tap Filter for a Multipath Spread of Four Chips -----	84
Figure 3.32	Frequency Response of 4-Tap Filter for a Multipath Spread of Four Chips -----	85

Figure 3.33	Frequency Response of 4-Tap Filter for a Multipath Spread of Four Chips -----	86
Figure 3.34	Frequency Response of 4-Tap Filter for a Multipath Spread of Four Chips -----	87
Figure 3.35	Frequency Response of 4-Tap Filter for a Multipath Spread of Four Chips -----	88
Figure 3.36	Frequency Response of 4-Tap Filter for a Multipath Spread of Five Chips -----	89
Figure 3.37	Frequency Response of 4-Tap Filter for a Multipath Spread of Five Chips -----	90
Figure 3.38	Frequency Response of 4-Tap Filter for a Multipath Spread of Five Chips -----	91
Figure 3.39	Frequency Response of 4-Tap Filter for a Multipath Spread of Five Chips -----	92
Figure 3.40	Frequency Response of 4-Tap Filter for a Multipath Spread of Five Chips -----	93
Figure 3.41	Variance of Noise Terms at Output of PN Correlator -----	95
Figure 3.42	Probability of Error for Equalized PN Spread Spectrum System with -20 dB Signal-to-Noise Interference Ratio -----	98
Figure 3.43	Probability of Error for Equalized PN Spread Spectrum System with -10 dB Signal-to-Noise Ratio -----	99
Figure 3.44	Comparison of Performance for Equalized PN Spread Spectrum System with -10 dB and -20 dB Signal-to- Interference Ratios -----	100
Figure 3.45	Probability of Error for Equalized PN Spread Spectrum System with -20 dB Signal-to-Interference Ratio and Chip Decisions Into the PN Correlator -----	101
Figure 3.46	Probability of Error for Equalized PN Spread Spectrum System with -10 dB Signal-to-Interference Ratio and Chip Decisions Into the PN Correlator -----	102

EVALUATION

This research has quantitatively established the validity of adaptive excision of additive narrowband interference from direct sequence spread spectrum communication signals. More importantly, it has shown that this strategy is suitable for dispersive channels such as the skywave High Frequency (HF) radio communication channel. Consequently, this work is directly applicable to the accomplishment of RADC TPO 1B in the area of sub-UHF communications, especially HF radio communications. We anticipate that the research described in this report will be directly applied in our forthcoming exploratory development work in HF signal processing and in our advanced development new initiative in HF communications.


JOHN T. GAMBLE
Project Engineer

I. INTRODUCTION

Spread spectrum, direct sequence or pseudo-noise (PN) modulation is employed in digital communication systems to reduce the effects of interference due to other users and intentional jamming. When the interference is narrow band the cross-correlation of the received signal with the replica of the PN code sequence reduces the level of the interference by spreading it across the frequency band occupied by the PN signal. Thus, the interference is rendered equivalent to a lower level noise with a relatively flat spectrum. Simultaneously, the cross-correlation operation collapses the desired signal to the bandwidth occupied by the information signal prior to spreading.

The interference immunity of a PN spread spectrum communication system corrupted by narrow band interference can be further improved by filtering the signal prior to cross-correlation, where the objective is to reduce the level of the interference at the expense of introducing some distortion on the desired signal. This filtering can be accomplished by exploiting the wideband spectral characteristics of the desired PN signal and the narrow band characteristic of the interference. Since the spectrum of the PN signal is relatively flat across the signal frequency band, the presence of a strong narrow band interference is easily recognized. Then, the interference can be suppressed by means of an appropriately designed linear filter.

Our approach to the interference suppression problem has been greatly influenced by the previous work of Hsu and Giordano [1]. They considered the problem of narrow band interference estimation and suppression by means of two linear prediction algorithms, the Burg algorithm [2,3], and

the Levinson algorithm [2,4]. The channel through which the PN spread spectrum signal is transmitted was assumed to be nondispersive. Results were presented on the effectiveness of the linear prediction filter in suppressing the interference. Performance was measured in terms of signal-to-noise ratio at the output of the PN correlator.

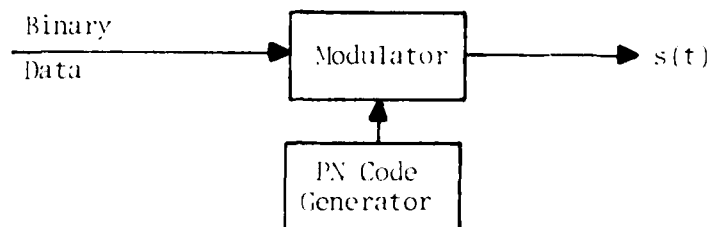
Our research work extends the results obtained by Hsu and Giordano on filter requirements and characteristics in single and multiple frequency band interference. In addition to a nondispersive channel, we consider the transmission of the PN spread spectrum signal over a channel characterized by fading and multipath (time dispersion). This serves as a model for radio channels such as HF. The existence of time dispersion in the received signal necessitates some means for dealing with this type of distortion at the receiver. We have considered the use of an adaptive decision-feedback equalizer preceding the PN correlator for mitigating the effects of time dispersion due to multipath and the linear, interference suppression filter.

Section II of this report presents the algorithms for estimating and suppressing narrow band interference in a wideband PN spread spectrum signal. Performance results are presented in Section III. Before concluding this section, we present a brief description of the mathematical model of the PN spread spectrum binary communication system which is used in the analysis and in Monte Carlo simulation. A baseband system is used throughout this report.

1.1 Mathematical Model of PN Spread Spectrum Communication System

Transmitted Signal

The transmitted signal is generated as shown in the block diagram.



The number of PN chips per information bit is L . Thus, the signal for the k^{th} information bit can be expressed as

$$b_k(t) = \sum_{j=1}^L p_{kj} q(t - j\tau_c) \quad (1.1)$$

where $\{p_{kj}\}$ represent the output sequence from the PN code generator for the k^{th} information bit and $q(t)$ is a rectangular pulse of duration τ_c and unit energy. The total transmitted signal may be expressed in the form

$$s(t) = \sum_k I_k b_k(t - kT_b) \quad (1.2)$$

where $\{I_k\}$ represents the binary information sequence and $T_b = L\tau_c$ is the bit interval (reciprocal of the bit rate).

Channel

The fading multipath channel is modeled as having discrete multipath components with relative delays equal to multiples of the chip duration τ_c . The impulse response may be expressed as

$$h(t; \tau) = \sum_{j=1}^K \alpha_j(t) \delta(\tau - j\tau_c) \quad (1.3)$$

where the $\{\alpha_j(t)\}$ are complex-valued, statistically independent, narrow band Gaussian random processes. Since the multipath components cannot be distinguished with a resolution better than τ_c , this is a reasonable model for the multipath. The fading is inherent in the time variations of the narrow band processes $\{\alpha_j(t)\}$.

The signal is corrupted by additive white Gaussian noise $n(t)$ and by narrow band interference denoted by $i(t)$. The narrow band interference is modeled as consisting of either a number of CW tones, i.e.,

$$i(t) = \sum_{m=1}^Q A_m \cos(2\pi f_m t + \phi_m) \quad , \quad (1.4)$$

or a filtered narrow band Gaussian noise process. In some cases it may be appropriate to model the interference as a narrow band random process that arrives at the receiver through another statistically independent fading multipath channel. Although we have not considered this model of the interference explicitly in our analysis, the algorithms presented in Section II for estimating and suppressing the interference still apply.

Received Signal

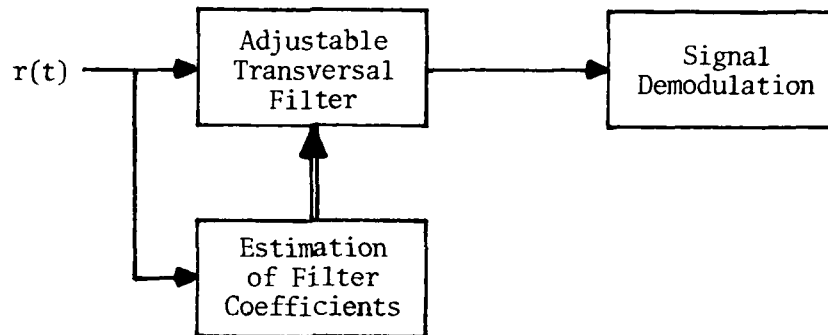
From the description given above, the received signal has the form

$$r(t) = \sum_{j=1}^K \alpha_j(t) s(t - j\tau_c) + i(t) + n(t) \quad (1.5)$$

The receiver attempts to suppress the interference $i(t)$ and then to recover the information sequence by further processing, which involves equalization and cross-correlation with a replica of the PN sequence.

Estimation and Suppression of the Interference

The estimation and suppression of the narrow band interference is accomplished prior to signal demodulation, as illustrated in general terms in the block diagram shown below.



The estimation of the filter coefficients may be accomplished by means of a linear prediction algorithm or by means of a spectral analysis algorithm based on the Fast Fourier Transform (FFT) algorithm. In any case, the objective is to design an adaptive transversal filter that highly attenuates the received signal in those frequency bands which contain strong interference.

Signal Demodulator

The signal at the output of the interference suppression filter is processed by an adaptive equalizer operating on a chip-by-chip basis, followed by a PN correlator which employs a replica of the transmitted PN sequence. Perfect synchronization of the PN sequence is assumed.

The type of equalizer employed in our performance evaluation is a decision-feedback equalizer. It serves to reduce the detrimental effects of time dispersion due to the interference suppression filter and the

channel multipath.

II. ALGORITHMS FOR ESTIMATION AND SUPPRESSION OF NARROW BAND INTERFERENCE

In this section we present a number of algorithms for estimating and suppressing a narrow band interference embedded in a wideband PN spread spectrum signal. The algorithms may be classified into two general categories. The algorithms in the first category employ the Fast Fourier Transform (FFT) algorithm for performing a spectral analysis from which an appropriate transversal filter is specified. These algorithms may be termed nonparametric, since no prior knowledge of the characteristics of the interference is assumed in forming the estimate. The algorithms in the second category are based on linear prediction and may be termed parametric. That is, the interference is modeled as having been generated by passing white noise through an all-pole filter [2].

2.1 Interference Suppression Based on Nonparametric Spectral Estimates

The basis for this method is that the power density spectrum of the PN sequence is relatively flat while the spectrum of the narrow band interference is highly peaked. The first step in this method is to estimate the power spectral density of the received signal. The spectral estimate can be obtained by any one of the well-known spectral analysis techniques described in [5]. For illustrative purposes, we have selected the Welch method and we have made use of the computer program listed in [6] to generate the numerical results presented in Section III.

Once the power spectral density of the received signal is estimated, the interference suppression filter can be designed. A transversal filter is an appropriate filter structure for this application, since we desire to use a filter that contains zeros in the frequency range occupied by the

interference. A relatively simple method for designing the transversal filter in the discrete-time (sampled-data) domain is to select its discrete Fourier transform (DFT) to be the reciprocal of the square root of the power spectral density at equally spaced frequencies. To elaborate, suppose that the transversal filter has K taps. The problem is to specify the K tap coefficients $\{h(n)\}$ or, equivalently, the DFT $H(k)$, defined as

$$H(k) = \sum_{n=0}^{K-1} h(n) e^{-j \frac{2\pi}{K} nk}, \quad k = 0, 1, \dots, K-1 \quad (2.1)$$

The DFT $H(k)$, $k = 0, 1, \dots, K-1$ is selected as

$$H(k) = \frac{1}{\sqrt{P(\frac{k}{K})}} e^{-j \frac{2\pi}{K} (\frac{K-1}{2}) k} \quad (2.2)$$

where $P(f)$, $0 \leq f \leq R_s$, denotes the estimate of the power spectral density and R_s denotes the sampling rate, which is normalized to unity.

It is desirable to have a transversal filter that has linear phase. This can be achieved if the impulse response $h(n)$ is real and satisfies the symmetry condition

$$h(n) = h(K - 1 - n) \quad (2.3)$$

The symmetry condition in Eq. (2.3) is satisfied if $H(k) = H^*(K - k)$. But $H(k)$ as defined in Eq. (2.2) does satisfy this condition since $P(f) = P(R_s - f)$. Hence, $h(n)$ is symmetric.

In effect, the filter characteristic obtained from Eq. (2.2) attempts to approximate an inverse filter to the power spectral density. That is,

the interference suppression filter attempts to whiten the spectrum of the incoming signal. Thus, the filter will have a large attenuation in the frequency range occupied by the interference and a relatively small attenuation elsewhere.

There is an alternative filter design procedure that may lead to a relatively smaller transversal filter. The method simply involves the selection of the position of the zeros so as to obtain an appropriate set of notches in the frequency response characteristic of the filter. We have not investigated any ad hoc methods for selecting the zeros of the transfer function, since the linear prediction approach described in the following section is a systematic method for attaining the same goal.

2.2 Interference Suppression Based on Linear Prediction

In contrast to the nonparametric spectral analysis method described in the previous section, the method presented in this section for estimating the narrow band interference is based on modeling the interference as white noise passed through an all-pole filter. That is, instead of using the received signal to estimate the spectrum directly, the signal is used to estimate the pole positions. This estimation is accomplished by means of linear prediction. An estimate of the power spectral density is easily obtained from the all-pole model. However, this step can be omitted. That is, the power density spectrum need not be computed explicitly for the purpose of designing the suppression filter. The interference suppression filter is simply a transversal (all-zero) filter having zero positions that coincide with the estimated pole positions. Thus, the spectrum of the signal at the output of the transversal filter is rendered white.

In order to develop the mathematical formulation for the all-pole model, we assume that the channel through which the signal is transmitted is nondispersive (no multipath). Let $s(t)$ denote the equivalent lowpass transmitted PN spread spectrum signal and let $r(t)$ denote the equivalent lowpass received signal. The latter is expressed as

$$r(t) = s(t) + i(t) + n(t) \quad (2.4)$$

where $i(t)$ denotes the narrow band interference and $n(t)$ is assumed to be a sample function of a white Gaussian noise process. For convenience, we assume that $r(t)$ is sampled at the chip rate of the PN sequence. Thus, Eq. (2.4) can be expressed as

$$r(k) = s(k) + i(k) + n(k), \quad k = 1, 2, \dots \quad (2.5)$$

We assume that $s(k)$, $i(k)$ and $n(k)$ are mutually uncorrelated.

An estimate of the interference $i(t)$ is formed from $r(k)$. Assume for the moment that the statistics of $i(t)$ are known and are stationary. Then, we can predict $i(k)$ from $r(k-1)$, $r(k-2)$, ..., $r(k-m)$. That is,

$$\hat{i}(k) = \sum_{\ell=1}^m a_{\ell} r(k-\ell) \quad (2.6)$$

where $\{a_{\ell}\}$ are the coefficients of the linear predictor.⁺ It should be emphasized that Eq. (2.6) predicts the interference but not the signal $s(k)$, because $s(k)$ is uncorrelated with $r(k-\ell)$ for $\ell = 1, 2, \dots, m$.

For convenience, our treatment of linear prediction is based on real-valued signals. The extension to complex-valued signals is straightforward.

as a consequence of the sampling being done at the chip rate.

The coefficients in Eq. (2.6) are determined by minimizing the mean square error between $r(k)$ and $\hat{i}(k)$, which is defined as

$$\begin{aligned} E(m) &= E[r(k) - \hat{i}(k)]^2 \\ &= E[r(k) - \sum_{\ell=1}^m a_{\ell} r(k - \ell)]^2 \end{aligned} \quad (2.7)$$

Minimization of E with respect to the predictor coefficients $\{a_{\ell}\}$ can be easily accomplished by invoking the orthogonality principle in mean square estimation [7]. This leads to the set of linear equations

$$\sum_{\ell=1}^m a_{\ell} \rho(k - \ell) = \rho(k) \quad , \quad k = 1, 2, \dots, m \quad (2.8)$$

where

$$\rho(k) = E[r(m) r(k + m)] \quad (2.9)$$

is the autocorrelation function of the received signal $r(k)$. The equations in (2.8) are usually called the Yule-Walker equations [2]. They can be solved efficiently by means of the Levinson algorithm [4,2].

The Levinson algorithm is an order-recursive method for solving Eq. (2.8). That is, it solves for the coefficients of an m -order predictor recursively from the coefficients of an $(m - 1)$ -order predictor. Starting with a first-order predictor and introducing another subscript in the prediction coefficients to indicate the order, we have

$$\rho(0) a_{11} = \rho(1)$$

and, hence,

$$a_{11} = \frac{\rho(1)}{\rho(0)} \quad (2.10)$$

For a second-order predictor, the two equations obtained from Eq. (2.8) are

$$\rho(0) a_{21} + \rho(1) a_{22} = \rho(1)$$

$$\rho(1) a_{21} + \rho(0) a_{22} = \rho(2) \quad (2.11)$$

The first equation in (2.11) can be used to solve for a_{21} . By substituting a_{11} for $\rho(1)/\rho(0)$ we obtain

$$a_{21} = a_{11} - a_{22} a_{11} \quad (2.12)$$

Thus, a_{21} is related to a_{11} . Next, a_{22} can be solved from the second equation in (2.11). By using Eq. (2.12) to eliminate a_{21} from the second equation, we obtain a_{22} in the form

$$a_{22} = \frac{\rho(2) - \rho(1) a_{11}}{\rho(0) - \rho(1) a_{11}} \quad (2.13)$$

Therefore, Eq. (2.13) is used to solve for a_{22} and, then, a_{21} is obtained from Eq. (2.12). This is the Levinson recursion for order two. In general, it can be shown [2,4], that the Levinson recursion for the

coefficients of the m-order predictor are

$$a_{mk} = a_{m-1 k} - a_{mm} a_{m-1 m-k} \quad , \quad k = 1, 2, \dots, m-1 \quad (2.14)$$

where

$$a_{mm} = \frac{\rho(m) - \underline{\beta}_{m-1}^r \underline{a}_{m-1}}{\rho(1) - \underline{\beta}_{m-1}^r \underline{a}_{m-1}^r} \quad (2.15)$$

The vectors $\underline{\beta}_n$, $\underline{a}_n$ and $\underline{a}_n^r$ in Eq. (2.15) are defined as

$$\underline{\beta}_n = \begin{bmatrix} \rho(n) \\ \rho(n-1) \\ \vdots \\ \rho(1) \end{bmatrix} \quad , \quad \underline{a}_n^r = \begin{bmatrix} a_{nn} \\ a_{nn-1} \\ \vdots \\ a_{n1} \end{bmatrix} \quad , \quad \underline{a}_n = \begin{bmatrix} a_{n1} \\ a_{n2} \\ \vdots \\ a_{nn} \end{bmatrix} \quad (2.16)$$

We note from Eqs. (2.16) that the vector $\underline{a}_n^r$ is simply the vector $\underline{a}_n$ in reverse order. Furthermore, if we express Eq. (2.8) in the matrix form

$$\underline{R}_m \underline{a}_m = \underline{b}_m \quad (2.17)$$

where $\underline{R}_m$ is the $(m \times m)$ autocorrelation matrix, the vector $\underline{\beta}_n$ in Eqs. (2.16) is simply the vector $\underline{b}_n$ in reverse order. These relations will be used later in our discussion of the lattice realization of the prediction filter.

The minimum mean square error is a measure of the effectiveness of

the prediction filter. The expression for $E_{\min}(m)$ is easily shown to be

$$E_{\min}(m) = \rho(0) - \sum_{k=1}^m a_{mk} \rho(k) \quad (2.18)$$

A recursion relation can also be obtained for $E_{\min}(m)$. Using Eq. (2.18) we have

$$E_{\min}(m) = \rho(0) - \sum_{k=1}^{m-1} a_{mk} \rho(k) - a_{mm} \rho(m) \quad (2.19)$$

If we substitute for a_{mk} in Eq. (2.19) the Levinson recursion given in Eq. (2.14), and rearrange the terms, we obtain

$$\begin{aligned} E_{\min}(m) &= [\rho(0) - \sum_{k=1}^{m-1} a_{m-1k} \rho(k)] - a_{mm} [\rho(m) - \sum_{k=1}^m a_{m-1, m-k} \rho(k)] \\ &= E_{\min}(m-1) - a_{mm} [\rho(m) - \beta_{m-1}' a_{m-1}] \end{aligned} \quad (2.20)$$

From Eq. (2.15), we note that

$$a_{mm} = \frac{\rho(m) - \beta_{m-1}' a_{m-1}}{E_{\min}(m-1)} \quad (2.21)$$

Hence, Eq. (2.20) becomes

$$E_{\min}(m) = (1 - a_{mm}^2) E_{\min}(m-1) \quad (2.22)$$

The result in Eq. (2.22) implies that $|a_{mm}| < 1$.

Once the prediction coefficients are determined, the estimate $i(k)$ of the interference, given by Eq. (2.6), is subtracted from $r(k)$ and the difference signal is processed further in order to extract the digital information. Thus, the equivalent transversal filter for suppressing the interference is described by the transfer function

$$A_m(z) = 1 - \sum_{k=1}^m a_{mk} z^{-k} \quad (2.25)$$

where z^{-1} denotes a unit of delay. The corresponding all-pole model for the interference signal is $1/A_m(z)$.

The solution of Eq. (2.8) for the coefficients $\{a_{mk}\}$ of the prediction filter requires knowledge of the autocorrelation function $c(k)$. In practice, the autocorrelation function of $i(k)$ and, hence, $r(k)$ is unknown and it may also be slowly varying in time. Consequently, one must consider methods for obtaining the predictor coefficients directly from the received signal $\{r(k)\}$. This may be accomplished in a number of ways. In this investigation, three different methods were considered. In all cases, we obtained the predictor coefficients by using a block of N samples of $\{r(k)\}$. The three methods are described in the following section.

2.5 Algorithms for Computing the Prediction Coefficients

Since the autocorrelation function $c(k)$ is not known *a priori*, we shall describe three algorithms for computing the prediction coefficients from the received signal $r(k)$. We assume throughout that a block of N samples of $r(k)$ are available.

Direct Application of the Levinson Algorithm

The first method is simply based on the direct estimation of $\rho(k)$ from the block of N samples. The estimate of $\rho(k)$ is

$$\hat{\rho}(k) = \sum_{n=1}^{N-k} r(n) r(n+k) \quad , \quad k = 0, 1, \dots, m \quad (2.24)$$

The estimate $\hat{\rho}(k)$ may then be substituted in Eq. (2.8) in place of $\rho(k)$ and the Levinson algorithm can be used to solve the equations efficiently. Thus, the recursive relations given in the previous section apply with $\rho(k)$ replaced by $\hat{\rho}(k)$.

Burg Algorithm

The second method considered for obtaining the prediction coefficients is the Burg algorithm [2,3]. Basically, the Burg algorithm may be viewed as an order-recursive least squares algorithm in which the Levinson recursion is used in each iteration. To be specific, we begin with the determination of the coefficient in a first-order predictor, based on the method of least squares.

The performance index used by Burg is the mean (time-average) square error, which, for a first-order predictor, is defined as

$$\begin{aligned} E_B(1) &= \sum_{i=2}^N \{ [r(i) - a_{11} r(i-1)]^2 + [r(i-1) - a_{11} r(i)]^2 \} \\ &= \sum_{i=2}^N [f_1^2(i) + b_1^2(i)] \end{aligned} \quad (2.25)$$

The first term in the sum represents the error in a first-order predictor operating on the received signal in the forward direction and, hence, it

is called the forward error $f_1(i)$. The second term represents the error in a first-order predictor operating on the data in the reverse direction and, hence, it is called the backward error $b_1(i)$. Minimization of $E_B(1)$ with respect to a_{11} yields the predictor coefficient a_{11} in the form

$$a_{11} = \frac{2 \sum_{i=2}^N r(i) r(i-1)}{\sum_{i=2}^N [r^2(i) + r^2(i-1)]} \quad (2.26)$$

For the second-order predictor we have the performance index

$$E_B(2) = \sum_{i=3}^N [f_2^2(i) + b_2^2(i)] \quad (2.27)$$

where $f_2(i)$ and $b_2(i)$ denote the forward and backward errors, respectively, which are defined as

$$\begin{aligned} f_2(i) &= r(i) - \sum_{k=1}^2 a_{2k} r(i-k) \\ b_2(i) &= r(i-2) - \sum_{k=1}^2 a_{2k} r(i-2+k) \end{aligned} \quad (2.28)$$

We use the Levinson recursion given in Eq. (2.12) to express $f_2(i)$ and $b_2(i)$ in terms of a_{11} and a_{22} , where a_{11} is the coefficient of the first-order predictor. For $f_2(i)$ we obtain the relation

$$\begin{aligned}
f_2(i) &= r(i) - a_{11} r(i-1) - a_{22}[r(i-2) - a_{11} r(i-1)] \\
&= f_1(i) - a_{22} b_1(i-1)
\end{aligned} \tag{2.29}$$

and for $b_2(i)$ we obtain the relation

$$b_2(i) = b_1(i-1) - a_{22} f_1(i) \tag{2.30}$$

Minimization of Eq. (2.27) with respect to a_{22} yields the result

$$a_{22} = \frac{2 \sum_{i=3}^N f_1(i) b_1(i-1)}{\sum_{i=3}^N [f_1^2(i) + b_1^2(i-1)]} \tag{2.31}$$

Let us now consider an m^{th} order predictor. The performance index is defined as

$$E_B(m) = \sum_{i=m+1}^N [f_m^2(i) + b_m^2(i)] \tag{2.32}$$

where the forward error $f_m(i)$ and the backward error $b_m(i)$ are defined as

$$\begin{aligned}
f_m(i) &= r(i) - \sum_{k=1}^m a_{mk} r(i-k) \\
b_m(i) &= r(i-m) - \sum_{k=1}^m a_{mk} r(i-m+k)
\end{aligned} \tag{2.33}$$

The general Levinson recursion, given in Eq. (2.14), is used in Eqs. (2.33) to express $f_m(i)$ and $b_m(i)$ in terms of the predictor coefficients at the $(m - 1)$ iteration. The result of this substitution is

$$\begin{aligned} f_m(i) &= f_{m-1}(i) - a_{mm} b_{m-1}(i - 1) \\ b_m(i) &= b_{m-1}(i - 1) - a_{mm} f_{m-1}(i) \end{aligned} \quad (2.34)$$

Upon substituting Eqs. (2.34) into Eq. (2.32) and minimizing $E_B(m)$ with respect to a_{mm} we obtain

$$a_{mm} = \frac{2 \sum_{i=m+1}^N f_{m-1}(i) b_{m-1}(i - 1)}{\sum_{i=m+1}^N [f_{m-1}^2(i) + b_{m-1}^2(i - 1)]} \quad (2.35)$$

A recursion relation can also be obtained for the minimum mean square error. By substituting Eqs. (2.34) into Eq. (2.32), expanding the squared terms and using Eq. (2.35), we obtain the recursive relation

$$E_{B \min}(m) = (1 - a_{mm}^2) E_{B \min}(m - 1) \quad (2.36)$$

The recursive relations in Eqs. (2.34), (2.35) and (2.36) constitute the Burg algorithm for computing the predictor coefficients and the minimum mean square error. The estimate of the power spectral density obtained by means of the Burg algorithm using a predictor of order m is

$$P(f) = \frac{E_B \min^{(m)}}{\left| 1 - \sum_{k=1}^m a_{mk} e^{-j2\pi kf} \right|^2} \quad (2.37)$$

Least Squares Algorithm

As we indicated in the discussion above, the Burg algorithm is basically a least squares algorithm with the added constraint that the predictor coefficients satisfy the Levinson recursion. As a result of this constraint, an increase in the order of the predictor requires only a single parameter optimization at each stage. In contrast to this approach, we shall now describe an unconstrained least squares algorithm. That is, the algorithm computes the optimum predictor coefficients, in the sense of least squares, at each stage of the iteration.

As in the Burg algorithm, we minimize the mean square error performance given in Eq. (2.32). The forward and backward prediction errors are defined in Eqs. (2.33). The global minimization of Eq. (2.32) with respect to the set of predictor coefficients $\{a_{mk}\}$ yields the set of linear equations

$$\sum_{k=1}^m a_{mk} \phi(\ell, k) = \phi(\ell, 0) \quad , \quad \ell = 1, 2, \dots, m \quad (2.38)$$

where

$$\phi(\ell, k) = \sum_{i=m+1}^N [r(i-k) r(i-\ell) + r(i-m+k) r(i-m+\ell)] \quad (2.39)$$

The linear equations in (2.38) can be expressed in matrix form as

$$\underline{\phi}_m \underline{a}_m = \underline{\phi}_m \quad , \quad (2.40)$$

where the matrix $\underline{\phi}_m$ is an $(m \times m)$ autocorrelation matrix with elements $\{\phi(k, \ell)\}$ and $\underline{\phi}_m$ is an m -dimensional vector with elements $\phi(\ell, 0)$, $\ell = 1, 2, \dots, m$. The matrix $\underline{\phi}_m$ is symmetric. However, in contrast to the autocorrelation matrix R_m in Eq. (2.17), which is Toeplitz, the matrix $\underline{\phi}_m$ is not Toeplitz. Consequently, the Levinson algorithm cannot be used to solve Eq. (2.40) recursively. In spite of the fact that $\underline{\phi}_m$ is not Toeplitz, it is still possible to derive a recursive algorithm for the predictor coefficients based on the least squares performance index. Morf et al [8-11] have developed such a recursive least squares algorithm that not only allows one to recursively increase the order of the predictor, but also allows one to update the predictor coefficients recursively in time for a predictor of any given order. A detailed development of this recursive least squares algorithm has been given by Pack and Satorius [12] and, for the sake of brevity, will not be repeated here. An order-recursive version of this algorithm has also been described recently by Marple [13]. The point that we wish to make is that the prediction coefficients based on the least squares criterion can be solved efficiently by means of an algorithm that is both recursive in order and in time. The order-recursive part of the algorithm fits the lattice formulation described briefly in the following section.

To conclude this section, we observe that the minimum mean square error in the least squares solution can be expressed as

$$\begin{aligned}
E_{LS \min}^{(m)} &= \sum_{i=m+1}^N [f_m(i) r(i) + b_m(i) r(i-m)] \\
&= \phi(0,0) - \sum_{k=1}^m a_{mk} \phi(k,0)
\end{aligned} \tag{2.41}$$

As a final comment we mention the well-known property of the unconstrained least squares solution, namely, that the resulting prediction coefficients do not necessarily yield a minimum phase filter $A_m(z)$. In our case and in some other practical applications of linear prediction this is not a problem.

2.4 Lattice Structure for Linear Prediction Filters

In this section we demonstrate that the transversal filter with transfer function

$$A_m(z) = 1 - \sum_{k=1}^m a_{mk} z^{-k} \tag{2.42}$$

where the $\{a_{mk}\}$ are the prediction coefficients, can also be realized as a lattice filter. As we shall observe below, the lattice filter structure has a number of properties that may prove desirable in a practical implementation of the interference suppression filter.

The starting point for this development is the Levinson recursive algorithm for the predictor coefficients $\{a_{mk}\}$ given in Eq. (2.14). If we substitute this relation into Eq. (2.42) and rearrange terms, we obtain

$$\begin{aligned}
\Lambda_m(z) &= [1 - \sum_{k=1}^{m-1} a_{m-1-k} z^{-k}] - a_{mm} z^{-m} [1 - \sum_{k=1}^{m-1} a_{m-1-k} z^{-k}] \\
&= \Lambda_{m-1}(z) - a_{mm} z^{-m} [1 - \sum_{k=1}^{m-1} a_{m-1-k} z^{-k}] \\
&= \Lambda_{m-1}(z) - a_{mm} z^{-m} \Lambda_{m-1}(z^{-1})
\end{aligned} \tag{2.43}$$

The filter defined by the transfer function

$$G_m(z) = z^{-m} \Lambda_m(z^{-1}) \tag{2.44}$$

is the transversal filter resulting from backward prediction. Consequently, we may express Eq. (2.43) in the form

$$\Lambda_m(z) = \Lambda_{m-1}(z) - a_{mm} z^{-1} G_{m-1}(z) \tag{2.45}$$

Thus, the transfer function $\Lambda_m(z)$ for the transversal filter arising from forward prediction is related to the transversal filter resulting from backward prediction. A similar relationship holds for $G_m(z)$. This can be obtained by substituting for $\Lambda_m(z^{-1})$ from Eq. (2.45) into Eq. (2.44). This substitution yields

$$\begin{aligned}
G_m(z) &= z^{-m} [\Lambda_{m-1}(z^{-1}) - a_{mm} z G_{m-1}(z^{-1})] \\
&= z^{-1} [z^{-(m-1)} \Lambda_{m-1}(z^{-1})] - a_{mm} z^{-(m-1)} G_{m-1}(z^{-1}) \\
&= z^{-1} G_{m-1}(z) - a_{mm} \Lambda_{m-1}(z)
\end{aligned} \tag{2.46}$$

Thus, Eq. (2.46) gives the recursive relation for $G_m(z)$.

Let $f_m(i)$ and $b_m(i)$ denote the output sequences of the filters $A_m(z)$ and $G_m(z)$, respectively, for an input sequence $x(i)$. Since, $F_m(z) = A_m(z) X(z)$ and $B_m(z) = G_m(z) X(z)$, it follows from Eqs. (2.43) and (2.45) that $F_m(z)$ and $B_m(z)$ satisfy the recursive relations

$$\begin{aligned} F_m(z) &= F_{m-1}(z) - a_{mm} z^{-1} B_{m-1}(z) \\ B_m(z) &= z^{-1} B_{m-1}(z) - a_{mm} F_{m-1}(z) \end{aligned} \quad (2.47)$$

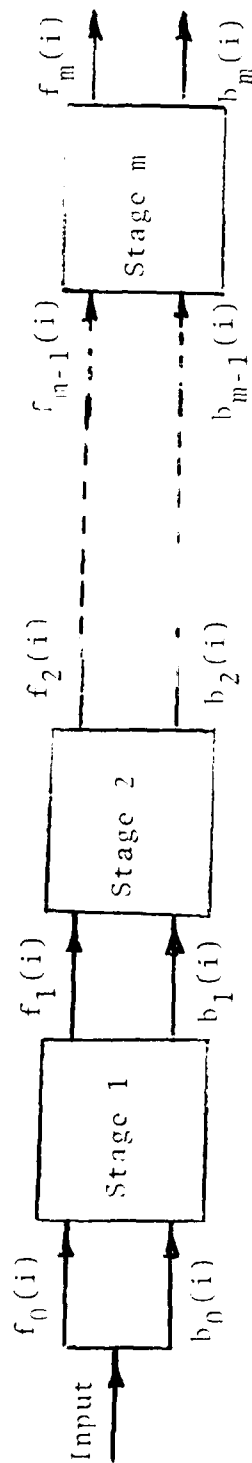
where initially we have $F_0(z) = B_0(z) = X(z)$. Alternatively, in the time domain the relations in Eqs. (2.47) are

$$\begin{aligned} f_m(i) &= f_{m-1}(i) - a_{mm} b_{m-1}(i-1) \\ b_m(i) &= b_{m-1}(i-1) - a_{mm} f_{m-1}(i) \end{aligned} \quad (2.48)$$

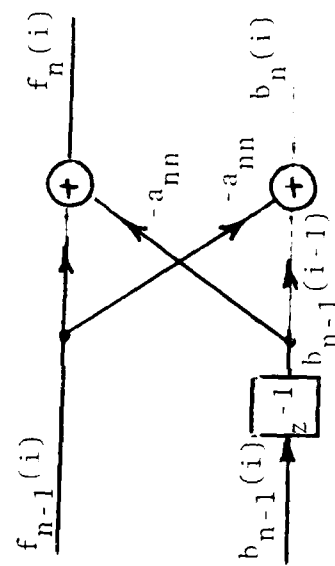
with initial conditions $f_0(i) = b_0(i) = x(i)$.

The recursive relations in Eqs. (2.48) describe a lattice filter as shown in Figure 2.1. In our case the input to the lattice is the received signal sequence $r(i)$ and the desired output is $f_m(i)$. Each stage of the lattice has two inputs and two outputs. The two inputs and outputs are the forward and backward prediction errors defined previously. These errors, usually called the residuals, satisfy a number of interesting properties [14], which we state here without proof.

First of all, there is the orthogonality property between the



(a) An m-stage lattice filter



(b) A typical stage of a lattice filter

Figure 2.1 Lattice Filter

residuals and the input:

$$E[f_m(i) r(i - k)] = 0, \quad 1 \leq k \leq m$$

$$E[b_m(i) r(i - k)] = 0, \quad 1 \leq k \leq m-1 \quad (2.49)$$

These relations are simply a restatement of the orthogonality principle in linear mean square estimation [7]. Secondly, the backward residuals are self-orthogonal, in the sense that

$$E[b_m(i) b_n(i)] = E_{\min}^{(m)} \delta_{mn} \quad (2.50)$$

where $E_{\min}^{(m)}$ is the minimum mean square error for an m -stage lattice or predictor, and δ_{mn} is the Kronecker delta. This orthogonality relation means that successive stages of the lattice are decoupled statistically. In terms of the forward residuals, the expression for the minimum mean square error is

$$E_{\min}^{(m)} = E[f_m^2(i)] \quad (2.51)$$

Two additional expressions for $E_{\min}^{(m)}$ are

$$E_{\min}^{(m)} = E[f_m(i) r(i)] \quad (2.52)$$

$$E_{\min}^{(m)} = E[b_m(i) r(i - m)] \quad (2.53)$$

Finally, the prediction coefficients $\{a_{kk}\}$ computed at each stage of the lattice are related to the cross-correlation between the forward and backward residuals. Specifically, we have

$$a_{kk} = \frac{E[f_{k-1}(i) b_{k-1}(i-1)]}{E_{\min}(k-1)} \quad (2.54)$$

In using the lattice structure instead of the transversal structure for the interference suppression filter, one must compute the lattice gains $\{a_{kk}\}$ as specified by Eq. (2.15) or Eq. (2.21), in conjunction with the Levinson recursion (2.14). In this computation, the estimate (time-average) of the autocorrelation function $\hat{\rho}(k)$ is used in place of the statistical average $\rho(k)$.

The Burg algorithm as described by the relations in Eqs. (2.34), (2.35) and (2.36) is basically a lattice implementation of the linear predictor. The lattice stages are specified by the recursive relations in Eqs. (2.34) for $f_m(i)$ and $b_m(i)$, with the lattice gains $\{a_{mm}\}$ given by Eq. (2.35). It is interesting to note that Eq. (2.35), which was obtained by performing a least squares optimization, is the time-average equivalent of the statistical average given by Eq. (2.54). In fact, the forward and backward residuals have identical statistical mean square values. Consequently, the denominator in Eq. (2.35) is equivalent to twice the statistical mean square value. Thus, the factors of two in the numerator and denominator cancel.

From a computational viewpoint, the Burg algorithm has the advantage that the lattice gains are computed directly from the forward and backward residuals, as indicated by the relation in Eq. (2.35), whereas in

the direct application of the Levinson algorithm one must first compute the estimate of the autocorrelation function from the data. As a consequence, the Burg algorithm is computationally more efficient.

The least squares algorithm described in the previous section can also be formulated in terms of a lattice filter structure. Such a formulation was developed by Morf et al [8-11]. For a clear tutorial presentation of the lattice, recursive least squares formulation the interested reader is referred to the report by Pack and Satorius [12].

2.5 Generalization of Linear Prediction Algorithms

Up to this point, our treatment of linear prediction algorithms has been limited to one-step prediction. In the PN spread spectrum problem under consideration, one-step linear prediction is appropriate if the received signal is sampled at a rate of one sample per PN chip. In such a case the predictor forms an estimate of the narrow band interference and is insensitive to the presence of the desired signal. As a consequence, the desired signal is not suppressed.

If, for some reason, the received signal is sampled at a higher rate, say M samples per chip, then successive samples of the desired signal are highly correlated. In this case, the one-step linear predictor will attempt to predict and suppress not only the narrow band interference but also the desired signal. Since this situation is undesirable, the remedy is to employ an M -step linear predictor, which can be implemented as illustrated in Figure 2.2.

A similar problem arises in attempting to suppress a narrow band interference in the presence of resolvable signal multipath. If the

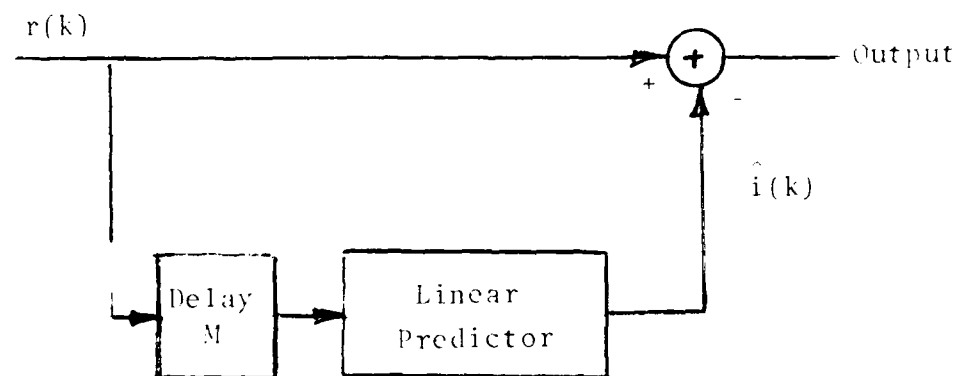


Figure 2.2 An M-Step Linear Predictor

multipath components in the PN spread spectrum signal span a time interval of M samples, a one-step linear predictor of order M or greater will attempt to estimate and suppress the multipath components in the received signal. Since the multipath signal components represent a form of signal diversity, their suppression by the linear predictor may be undesirable. To avoid this situation an M -step linear predictor can be employed. In the case of multipath, however, the major problem with this approach is that if M is large the interference components $i(k)$ and $i(k - M - n)$, $n = 0, 1, \dots, m-1$ are highly decorrelated. As a consequence, the estimate of the interference is poor and so is the performance of the suppression filter. When the interference is much stronger than the desired signal, the multipath components of the signal are completely masked by the interference. In this case a one-step predictor is appropriate, since the predictor responds to the interference and is relatively unaffected by the much weaker multipath components of the signal. This situation is demonstrated in Section III by some numerical results.

In spite of this one shortcoming of the M -step predictor, it may still be appropriate to use it in some situations, as for example, in the case of multiple samples per chip. For this reason we briefly outline the appropriate algorithms for performing M -step prediction and interference suppression.

First, we adopt a statistical approach. The M -step linear predictor of order m is

$$i(t) = \sum_{k=1}^m c_k r(t - k - M + 1) \quad (2.55)$$

and the corresponding mean square error is

$$E(m) = E\{|r(t) - \hat{i}(t)|^2\} \quad (2.56)$$

Minimization of $E(m)$ with respect to the predictor coefficients $\{c_k\}$ yields the set of linear equations

$$\sum_{k=1}^m c_k r(\ell - k) = r(\ell + M - 1), \quad \ell = 1, 2, \dots, m \quad (2.57)$$

The difference between this set of linear equations and the set given in Eq. (2.8) for the one-step predictor are the terms in the right-hand side of the equations. The solution of Eq. (2.57) requires a generalization of the Levinson algorithm which incorporates the recursion in Eq. (2.14) and includes a second recursive relation. It is straightforward to show that the second recursive relation is

$$c_{mk} = c_{m-1, k} - c_{mn} a_{m-1, m-k}^r, \quad k = 1, 2, \dots, m-1 \quad (2.58)$$

where the coefficients c_{mn} is given by the expression

$$c_{mn} = \frac{\rho(M-1+m) - \beta_{m-1}^T c_{m-1}}{\rho(0) - \beta_{m-1}^T a_{m-1}^r} \quad (2.59)$$

The vectors a_n^r and β_n in Eq. (2.59) were previously defined in Eqs. (2.16), and the vector c_n is the n -dimensional vector of the coefficients $\{c_{nk}\}$, $k = 1, 2, \dots, n$. The coefficients $\{a_{mn}\}$ are given by Eq. (2.15). Finally,

the initial conditions are

$$a_{11} = \frac{\rho(1)}{\rho(0)}, \quad c_{11} = \frac{\rho(M)}{\rho(0)} \quad (2.60)$$

The above equations (2.58)-(2.60) along with Eq. (2.14) constitute the generalized version of the Levinson algorithm which is appropriate for solving the linear equations in Eq. (2.57).

In a practical implementation of the generalized Levinson algorithm for the M-step predictor, an estimate of the autocorrelation function $\hat{\rho}(k)$ is used in place of $\rho(k)$. Furthermore, the interference suppression filter can be implemented either as a transversal filter or in the form of a lattice. In the lattice structure, the transfer function

$$C_m(z) = 1 - \sum_{k=1}^m c_{mk} z^{-k} \quad (2.61)$$

can be expressed as

$$C_m(z) = C_{m-1}(z) - c_{mm} z^{-1} G_{m-1}(z) \quad (2.62)$$

where Eq. (2.62) follows from substituting Eq. (2.58) into Eq. (2.61).

$G_m(z)$ was defined previously in Eq. (2.44). For an input signal $X(z)$, let $V_m(z) = C_m(z) X(z)$. Then, the output sequence $v_m(i)$ can be expressed as

$$v_m(i) = v_{m-1}(i) - c_{mm} b_{m-1}(i-1) \quad (2.63)$$

This expression gives the necessary addition to the basic lattice in order to generate the output $y_m(i)$ of the M -step predictor. Figure 2.5 illustrates a typical stage of the lattice implementation for the M -step predictor.

The Burg algorithm can also be modified to render it appropriate for M -step prediction. In this case we define the forward and backward errors as

$$\begin{aligned} f'_m(i) &= r(i) - \sum_{k=1}^m c_{mk} r(i - k - M + 1) \\ b'_m(i) &= r(i - m - M + 1) - \sum_{k=1}^m c_{mk} r(i - m - M + 1 + k) \end{aligned} \quad (2.64)$$

and the performance index is the mean square error

$$E_{GB}(m) = \sum_{k=m+M}^N [f'^2_m(i) + b'^2_m(i)] \quad (2.65)$$

$E_{GB}(m)$ is minimized with respect to the single parameter c_{mm} subject to the constraint that $\{c_{mk}\}$, for $1 \leq k \leq m-1$, satisfy the generalized Levinson recursion given by Eqs. (2.58) and (2.14). If we substitute Eq. (2.58) into Eq. (2.64) we obtain the recursive algorithm for the forward and backward residuals $f'_m(i)$ and $b'_m(i)$ in the form

$$\begin{aligned} f'_m(i) &= f'_{m-1}(i) - c_{mm} b'_{m-1}(i - M) \\ b'_m(i) &= b'_{m-1}(i - 1) - c_{mm} f'_{m-1}(i - M + 1) \end{aligned} \quad (2.66)$$

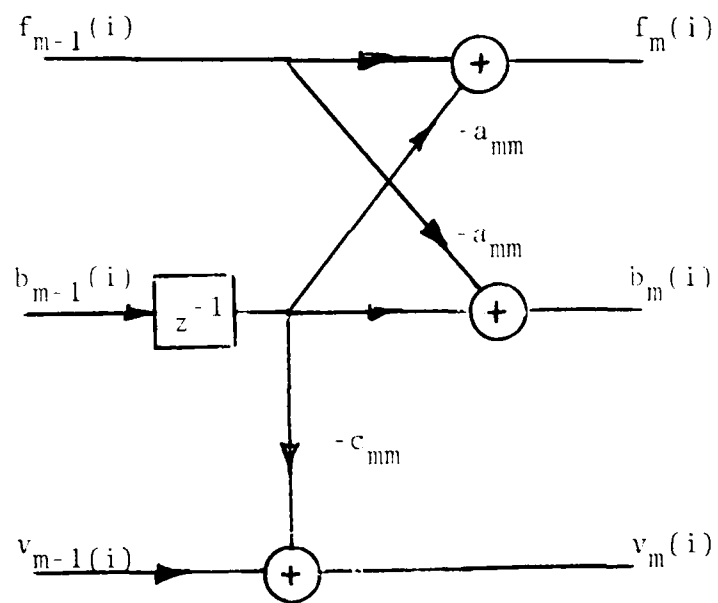


Figure 2.3 A Lattice Stage for the Generalized Levinson M-Step Linear Predictor

Substituting Eqs. (2.66) into Eq. (2.65) and minimizing the resulting $E_{GB}(m)$ with respect to c_{mn} yields the equation for c_{mn} in the form

$$c_{mn} = \frac{\sum_{i=m+M}^N [f'_{m-1}(i) b_{m-1}(i-M) + b'_{m-1}(i-1) f_{m-1}(i-M+1)]}{\sum_{i=m+M}^N [f_{m-1}^2(i-M+1) + b_{m-1}^2(i-M)]} \quad (2.67)$$

Thus, Eqs. (2.66), (2.67) along with (2.34) and (2.35) constitute a generalized version of the Burg algorithm appropriate for M-step prediction. Figure 2.4 illustrates a single stage in a lattice implementation of this algorithm.

If we drop the constraint that the predictor coefficients $\{c_{mk}\}$ satisfy the generalized Levinson algorithm, and simply minimize the mean square error in Eq. (2.65) with respect to the entire set of coefficients, we obtain the solution for the least squares M-step predictor. The appropriate set of equations for the coefficients are

$$\sum_{k=1}^m c_{mk} \psi(i, k) = \psi(i, 0) \quad , \quad i = 1, 2, \dots, m \quad (2.68)$$

where, by definition,

$$\begin{aligned} \psi(i, k) = & \sum_{i=m+M}^N [r(i-k-M+1) r(i-i-M+1) \\ & + r(i-m+k) r(i-m+i)] \quad , \quad i, k = 1, 2, \dots, m \end{aligned} \quad (2.69)$$

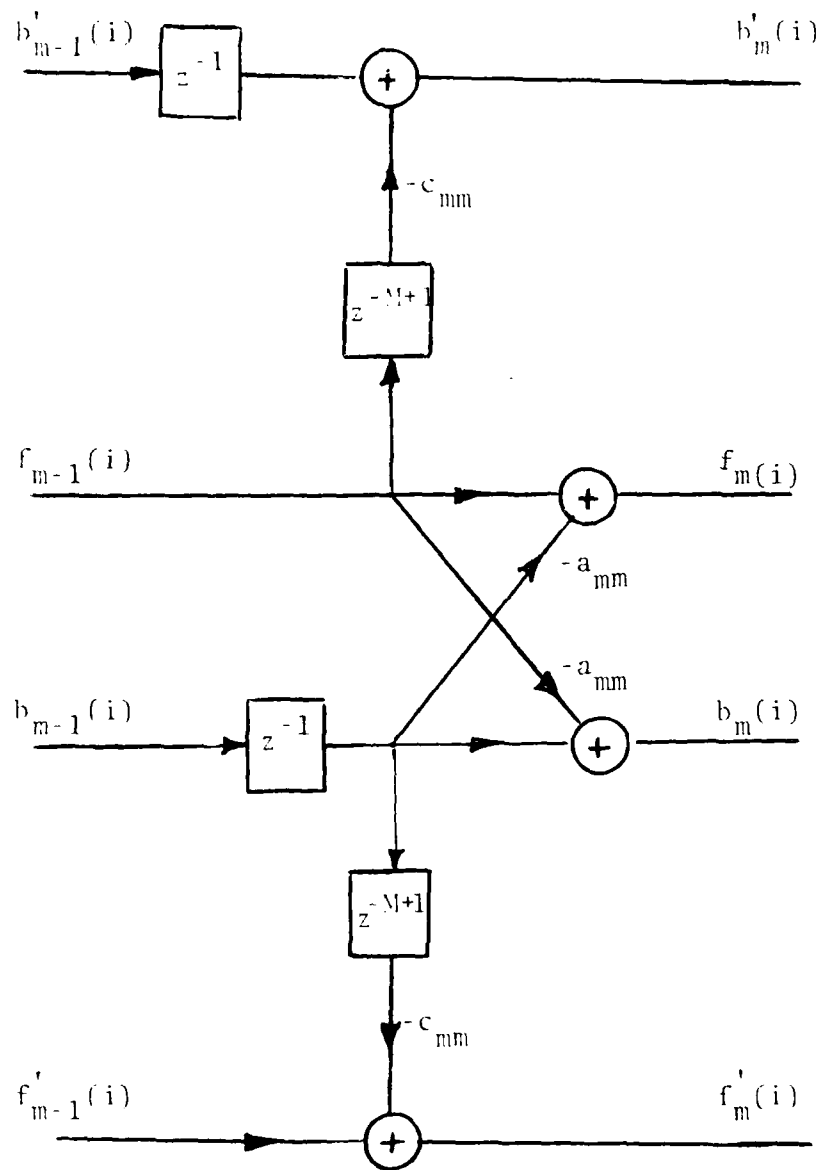


Figure 2.4 A Lattice Stage for the Generalized Burg M-Step Predictor

The minimum mean square error is

$$E_{\text{GLS}}^{(m)} = \psi(0,0) - \sum_{k=1}^m c_{mk} \psi(k,0) \quad (2.70)$$

The solution to the above equation can be formulated in terms of recursive relations, which are simply generalizations of the work of Morf et al [8-11] and Pack and Satorius [12]. The recursive relations represent a lattice formulation of the least squares, M-step prediction problem.

When the statistical characteristics of the received signal sequence $r(i)$ are changing with time (nonstationary signal) great care must be taken in selecting the block size of the data record from which the prediction coefficients are determined. In particular, N must be small relative to the number of samples over which a significant change occurs in the statistics of the signal. This is a situation where the parametric methods based on linear prediction show their superiority over the conventional nonparametric spectral estimation methods. That is, for the same spectral resolution, the parametric methods require a much smaller set of samples in comparison with conventional spectral estimation. Consequently, the parametric method will accommodate more rapidly changing signal statistics.

A convenient method for dealing with the slowly time-variant properties of the received signal is to employ a weighting function in the performance index, which places more emphasis on errors from current data and less on errors due to past data. A simple weighting function for accomplishing this type of weighting is an exponential.

In the context of one-step prediction, Morf et al [8-11] and Pack and Satorius [12] have presented a recursive least squares algorithm for an exponentially weighted mean square error. Also in the context of one-step prediction, Nuttall [15] has generalized the Burg algorithm by including a weighted mean square error performance index.

This concludes our discussion of algorithms for narrow band interference estimation and suppression. In the next section we present a number of numerical results that illustrate the effectiveness of interference suppression.

III. PERFORMANCE RESULTS ON INTERFERENCE SUPPRESSION

This section deals with the performance of the interference suppression filter. First, we determine the improvement provided by interference suppression as measured in terms of the signal-to-noise ratio (SNR) at the output of the PN correlator. The output SNR is by far the most convenient performance index for obtaining numerical results. This performance index is used to assess the improvement in performance obtained by an interference suppression filter. In addition, we describe a number of other characteristics of the interference suppression filter, including its frequency response and the location of its zeros. Finally, we present some Monte Carlo simulation results on the performance of the receiver as measured in terms of the probability of error. A two-path, Rayleigh fading channel is used in the simulation. The receiver consists of an interference suppression filter followed by a decision-feedback equalizer and a PN correlator.

3.1 SNR Improvement Factor Resulting from Interference Suppression

In order to demonstrate the effectiveness of the interference suppression algorithms, we shall compare the performance of the receiver with and without the suppression filter. Since the channel characteristic is not an issue in this type of comparison, we assume that the channel is ideal, i.e., nondispersive. Consequently, the received signal, sampled at the chip rate, can be represented as

$$r(k) = p(k) + i(k) + n(k) \quad , \quad k = 1, 2, \dots \quad (3.1)$$

where the binary sequence $\{p(k)\}$ represents the PN chips, $\{i(k)\}$ represents the sequence of samples of the narrow band interference and $\{n(k)\}$ represents the sequence of wideband noise samples.

Let the impulse response of the interference suppression filter be denoted by $\{h(k)\}$, $k = 0, 1, \dots, K$. In the case of the one-step predictor of order m , we have $h(0) = 1$, $h(k) = -a_{mk}$, and $K = m$. For an M -step predictor ($M > 1$), we have $h(0) = 1$, $h(k) = 0$ for $1 < k < M-1$, $h(k + M - 1) = -c_{mk}$ for $k = 1, 2, \dots, m$ and $K = m + M - 1$. The input to the filter is $r(k)$ and its output is

$$\begin{aligned} y(k) &= \sum_{\ell=0}^K h(\ell) r(k - \ell) \quad , \quad k = 1, 2, \dots \\ &= \sum_{\ell=0}^K h(\ell) [p(k - \ell) + i(k - \ell) + n(k - \ell)] \end{aligned} \quad (3.2)$$

It is apparent from Eq. (3.2) that the time-dispersive characteristic of the interference suppression filter results in inter-chip interference which can be mitigated by use of some form of equalization [16]. In the following computation, however, no equalizer is employed. Instead, the output of the interference suppression filter is fed directly to the PN correlator. Thus, the output of the PN correlator, which is the decision variable for recovering the binary information, is expressed as

$$U = \sum_{k=1}^L y(k) p(k) \quad (3.3)$$

where L represents the number of chips per information bit or the

processing gain.

By substituting Eq. (3.2) into Eq. (3.3), the decision variable can be expressed in the form

$$\begin{aligned}
 U &= \sum_{k=1}^L p(k) \left\{ \sum_{\ell=0}^K h(\ell) [p(k - \ell) + i(k - \ell) + n(k - \ell)] \right\} \\
 &= \sum_{k=1}^L p^2(k) + \sum_{\ell=1}^K h(\ell) \sum_{k=1}^L p(k) p(k - \ell) + \sum_{k=1}^L \sum_{\ell=0}^K h(\ell) p(k) \\
 &\quad \cdot [i(k - \ell) + n(k - \ell)] \\
 &= L + \sum_{\ell=1}^K h(\ell) \sum_{k=1}^L p(k) p(k - \ell) + \sum_{k=1}^L \sum_{\ell=0}^K h(\ell) p(k) i(k - \ell) \\
 &\quad + \sum_{k=1}^L \sum_{\ell=0}^K h(\ell) p(k) n(k - \ell) \tag{3.4}
 \end{aligned}$$

The first term in the right-hand side of Eq. (3.4) represents the desired signal component; the second term represents the self-noise caused by the dispersive characteristic of the filter; the third term represents the residual narrow band interference at the output of the PN correlator and the last term represents the additive wideband noise.

For the comparison that we wish to make, the SNR at the output of the PN correlator is a mathematically tractable performance index. To determine the expression for the SNR we must compute the mean and variance

of U . We assume that the binary PN sequence is white, the interference $i(k)$ has zero mean and autocorrelation function $\rho_i(k)$, and the additive noise $n(k)$ is white with variance σ_n^2 . Then, the mean of U is

$$E(U) = L \quad (5.5)$$

and the variance is

$$\begin{aligned} \text{var}(U) = & L \sum_{\ell=1}^K h^2(\ell) + L \sum_{\ell=0}^K \sum_{m=0}^K h(\ell) h(m) \rho_i(\ell - m) \\ & + L \frac{\sigma_n^2}{n} \sum_{\ell=0}^K h^2(\ell) \end{aligned} \quad (5.6)$$

The first term on the right-hand side of the expression for the variance represents the mean square value of the self-noise due to the time dispersion introduced by the interference suppression filter. The second term is the mean square value of the residual narrow band interference. The last term is the mean square value of the wideband noise.

The SNR at the output of the correlator is defined as the ratio of the square of the mean to the variance. Thus,

$$\text{SNR}_o = \frac{L^2}{L \sum_{\ell=1}^K h^2(\ell) + L \sum_{\ell=0}^K \sum_{m=0}^K h(\ell) h(m) \rho_i(\ell - m) + L \frac{\sigma_n^2}{n} \sum_{\ell=0}^K h^2(\ell)} \quad (5.7)$$

If there is no suppression filter, $h(\cdot) = 1$ for $\ell = 0$ and zero otherwise.

Therefore, the corresponding output SNR is

$$\text{SNR}_{\text{no}} = \frac{L}{\rho_i(0) + \sigma_n^2} \quad (3.8)$$

where $\rho_i(0)$ represents the total power of the narrow band interference.

The ratio of the SNR in Eq. (3.7) to the SNR in Eq. (3.8) represents the improvement in performance due to the use of the interference suppression filter. This ratio, denoted by η , is

$$\eta = \frac{\rho_i(0) + \sigma_n^2}{\sum_{\ell=1}^K h^2(\ell) + \sum_{\ell=0}^K \sum_{m=0}^K h(\ell) h(m) \rho_i(\ell - m) + \sigma_n^2 \sum_{\ell=0}^K h^2(\ell)} \quad (3.9)$$

We observe that η is independent of the processing gain L .

In plotting the improvement factor, it is convenient to use a logarithmic scale. Thus, we define

$$\eta_{\text{dB}} = 10 \log \eta \quad (3.10)$$

This factor will be plotted against the normalized SNR at the output of the PN correlator when there is no suppression filter. In other words, the abscissa is

$$\frac{\text{SNR}_{\text{no}}}{L} = \frac{1}{\rho_i(0) + \sigma_n^2} \quad (3.11)$$

As a consequence, the graphs of η_{dB} vs. SNR_{no}/L are universal plots in

the sense that they apply to any PN spread spectrum system with arbitrary processing gain.

3.2 Characteristics of the Interference Suppression Filter

In this section we shall discuss the characteristics of the interference suppression filter and we shall illustrate its performance as measured in terms of the improvement factor η_{dB} .

First we consider two models for the narrow band interference. One type consists of a sum of equally spaced sinusoids covering 20% of the signal band.

In particular, $i(k)$ is expressed as

$$i(k) = \sum_{m=1}^Q A_m \cos(2\pi f_m k + \phi_m) \quad (3.12)$$

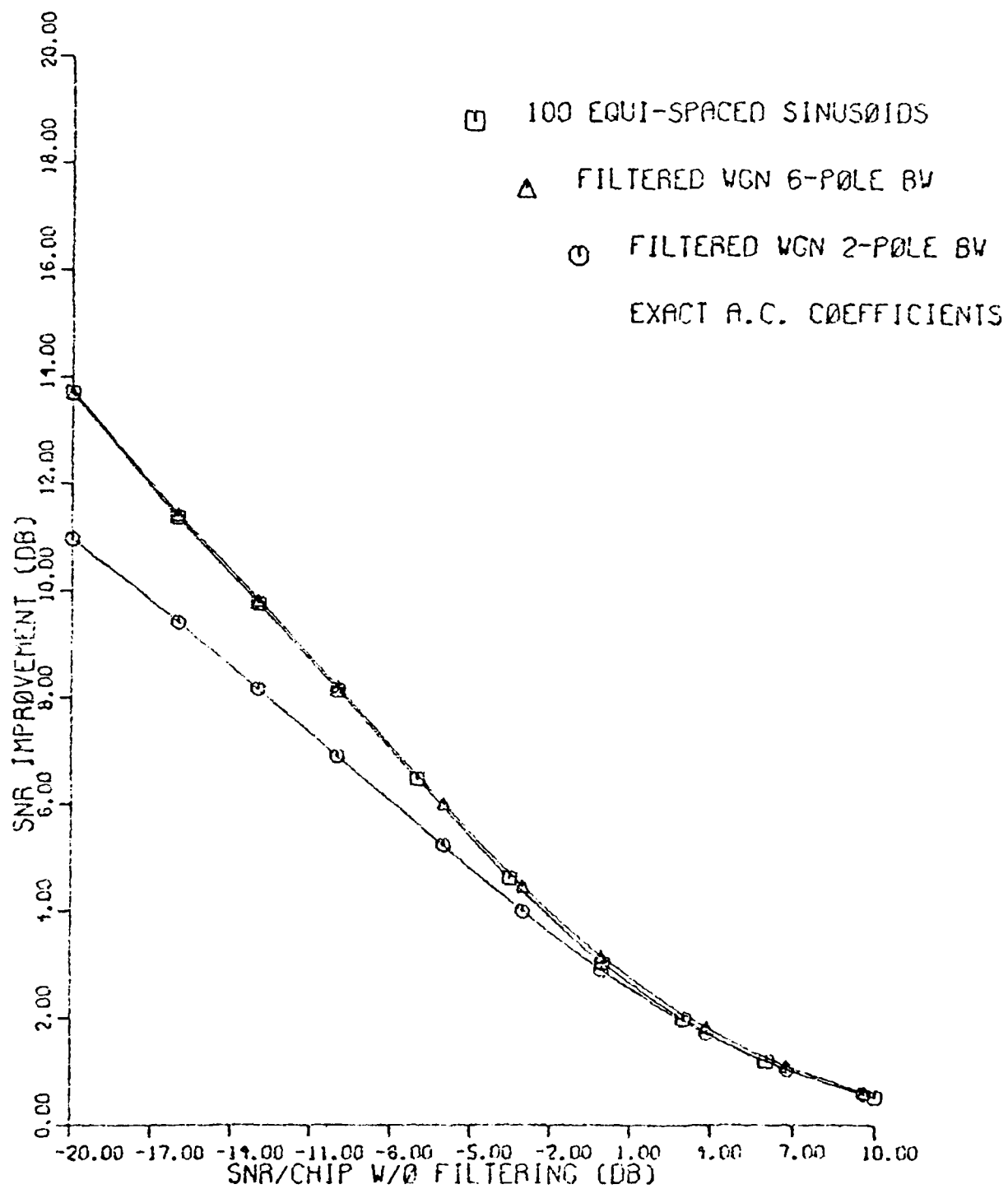
where the amplitudes $\{A_m\}$ were selected to be identical and the phases are uniformly distributed on the interval $(0, 2\pi)$. The autocorrelation function of $i(k)$ is

$$\rho_1(k) = \frac{1}{2} \sum_{m=1}^Q A_m^2 \cos 2\pi f_m k \quad (3.13)$$

The number of tones used in $i(k)$ was either $Q = 100$ or $Q = 10$. A second type of narrow band interference employed was filtered white noise. The filter characteristic was that of a Butterworth filter having a 3dB bandwidth covering 20% of the signal band.

Figure 3.1 illustrates the improvement factor for the $Q = 100$ sinusoidal interference and the filtered white noise interference for a six-pole Butterworth and a two-pole Butterworth filter. The interference

Figure 3.1 Improvement Factor for Single-Band Interference



suppression filter consists of four taps with exact values for the predictor coefficients determined from solving the linear equations given by (2.17). We observe that the improvement factors for the sinusoidal interference and the interference from the six-pole Butterworth are practically identical. On the other hand, the interference suppression filter is not quite as effective in suppressing the interference from the two-pole Butterworth primarily because of the difficulty in estimating and suppressing the rather significant amount of out-of-band power. Figure 3.2 illustrates the improvement factor for the same conditions as those in Figure 3.1 except that the predictor coefficients were determined from simulation data using the Levinson algorithm. The results of the simulation agreed very well with the analytical results shown in Figure 3.1 except at low values of interference where the improvement factor approaches zero dB, theoretically, but the simulation data indicates a small loss in performance. We have observed this phenomenon in other simulation data, which suggests that for small interference it is best to arbitrarily set the predictor coefficients to zero. Figure 3.3 illustrates the improvement factor for sinusoidal interference with $Q = 10$ and $Q = 100$ tones. The suppression filter consisted of either four taps or fifteen taps with the predictor coefficients estimated from simulated data. There appears to be little difference in performance between $Q = 10$ and $Q = 100$. Furthermore, there is very little gain in performance when the number of taps is increased from four to fifteen. The major conclusion that we have reached from the above results is that the model for the interference is not critical. Consequently, in most of our numerical results we used sinusoidal interference with $Q = 100$

Figure 3.2 Improvement Factor for Single-Band Interference --
Simulation Data

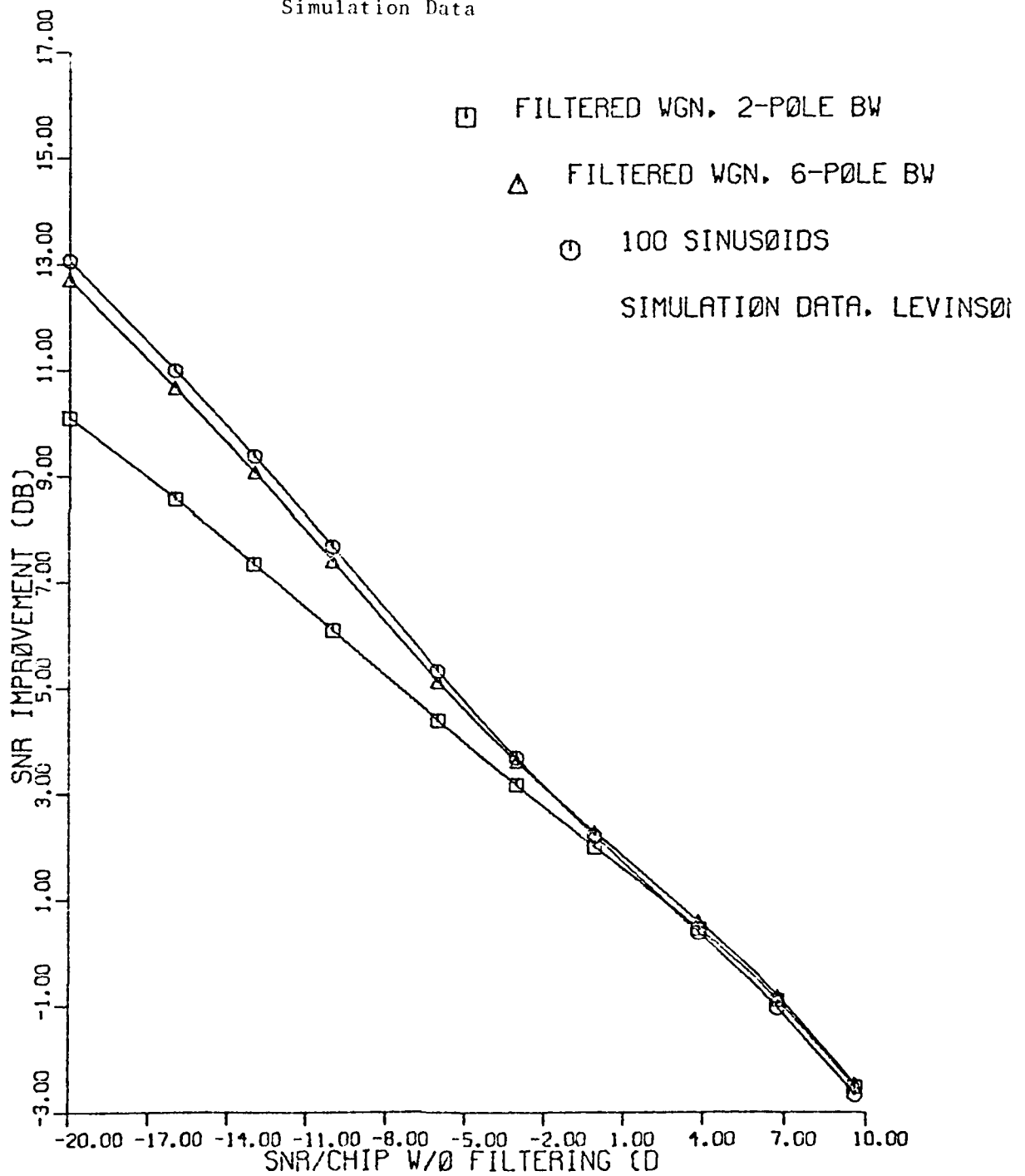
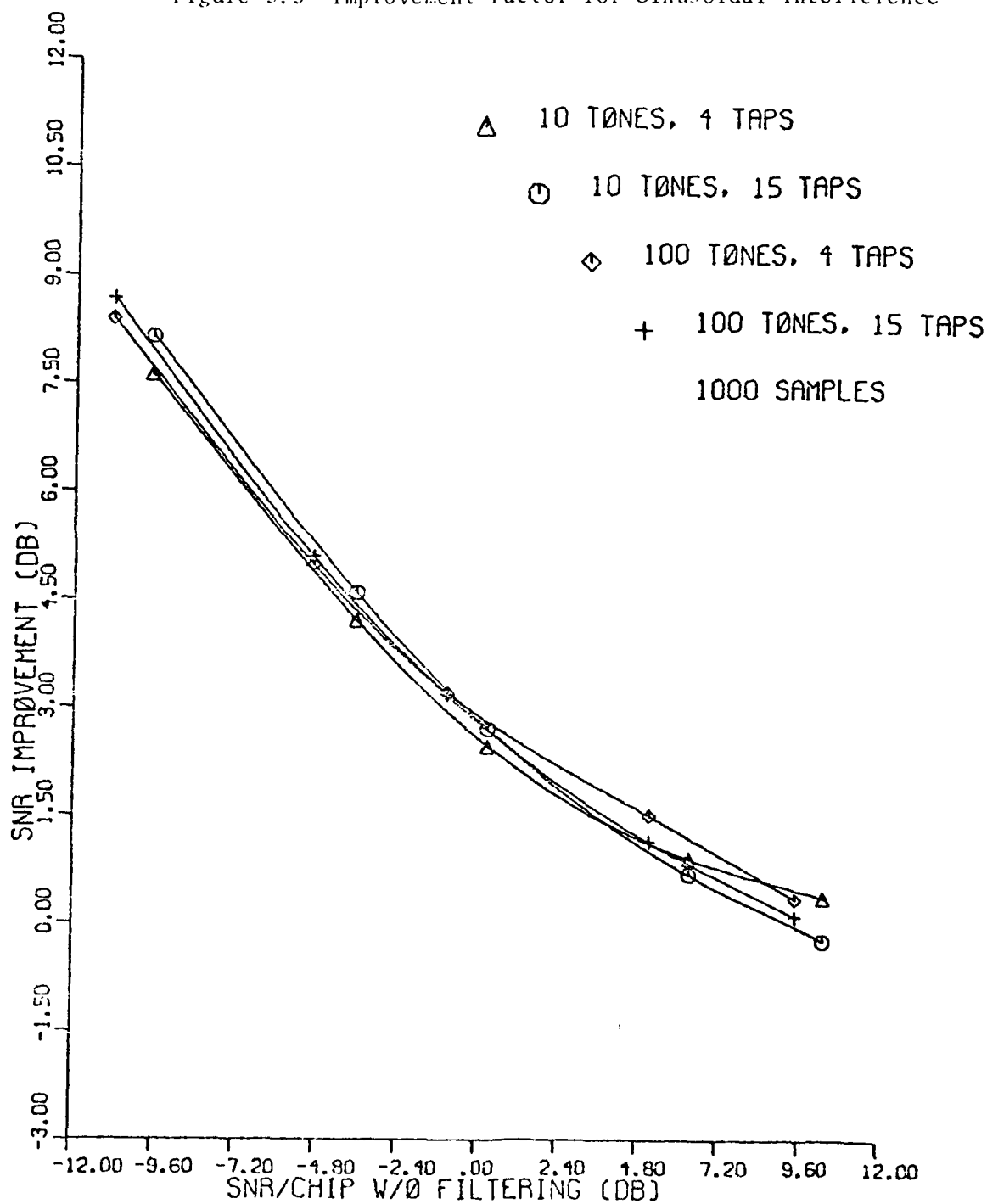


Figure 3.3 Improvement Factor for Sinusoidal Interference



tones. To a relatively small interference suppression filter such interference is indistinguishable from filtered white noise.

Secondly, we have investigated the length of the interference suppression filter required to achieve good performance. In this computation we maintained the 20% bandwidth occupancy for the interference, but we distributed it equally in several non-overlapping frequency bands. Figure 3.4 illustrates the improvement factor as a function of the number of filter taps when the SNR per chip without filtering is -20dB. From the graphs we observe that a filter having about eight taps performs well when the interference is split into two frequency bands; whereas a filter having sixteen to eighteen taps is required to achieve good performance when the interference is split into four frequency bands. Figure 3.5 illustrates the performance as a function of the SNR per chip for eight-tap and sixteen-tap filters when the interference occupies two bands and four bands, respectively. The graphs show that the sixteen-tap filter with the four interference bands closely approaches the performance gain of the eight-tap filter that suppresses the interference in two bands.

The frequency response characteristics of the eight-tap and sixteen-tap filters for an SNR per chip of -20dB are shown in Figures 3.6 and 3.7, respectively. It appears that the sixteen-tap filter introduces some distortion in the frequency range between notches. On the other hand, if the number of taps is increased beyond sixteen, the frequency response is improved. For example, Figure 3.8 illustrates the frequency response characteristic of a 29-tap filter when the SNR per chip without filtering is -20dB.

The computations reported above were repeated for eight-band and

Figure 3.4 Improvement Factor as a Function of Filter Order for Multi-Band Interference

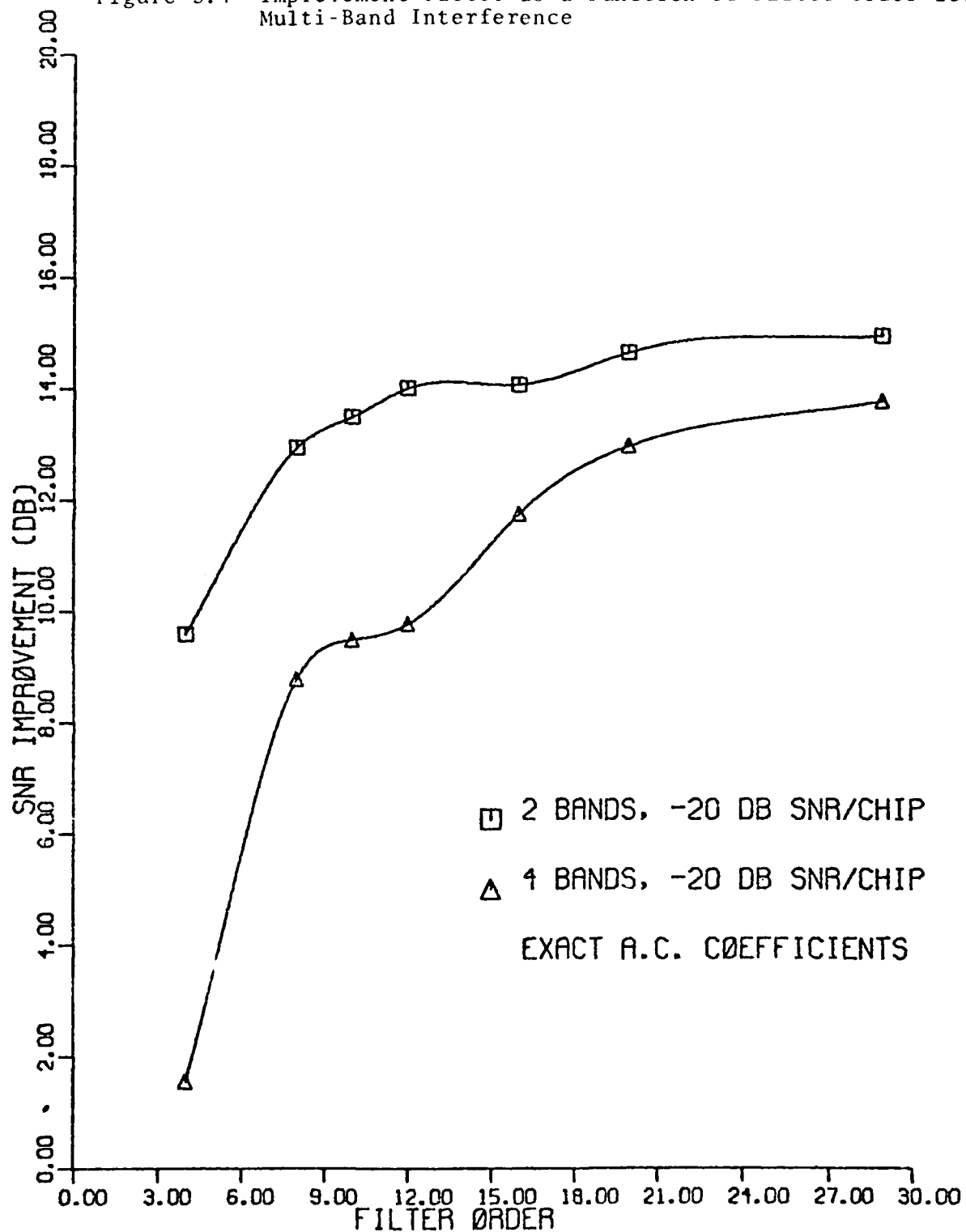
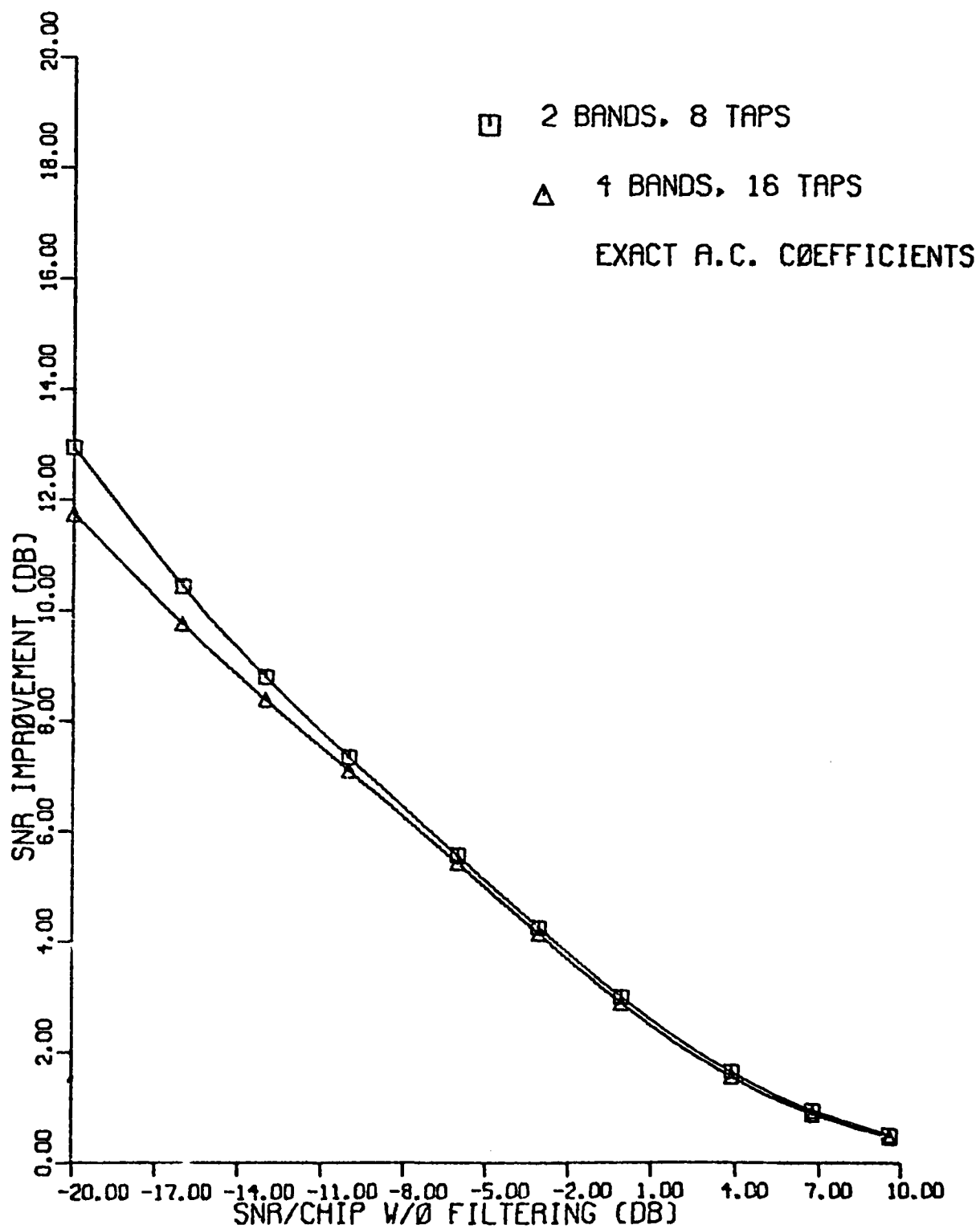


Figure 3.5 Improvement Factor for Eight-Tap and Sixteen-Tap Filters



YVSTAT, NV=.01

FILTER ORDER=8.

SNR/CHIP WITHOUT FILTERING = -20dB

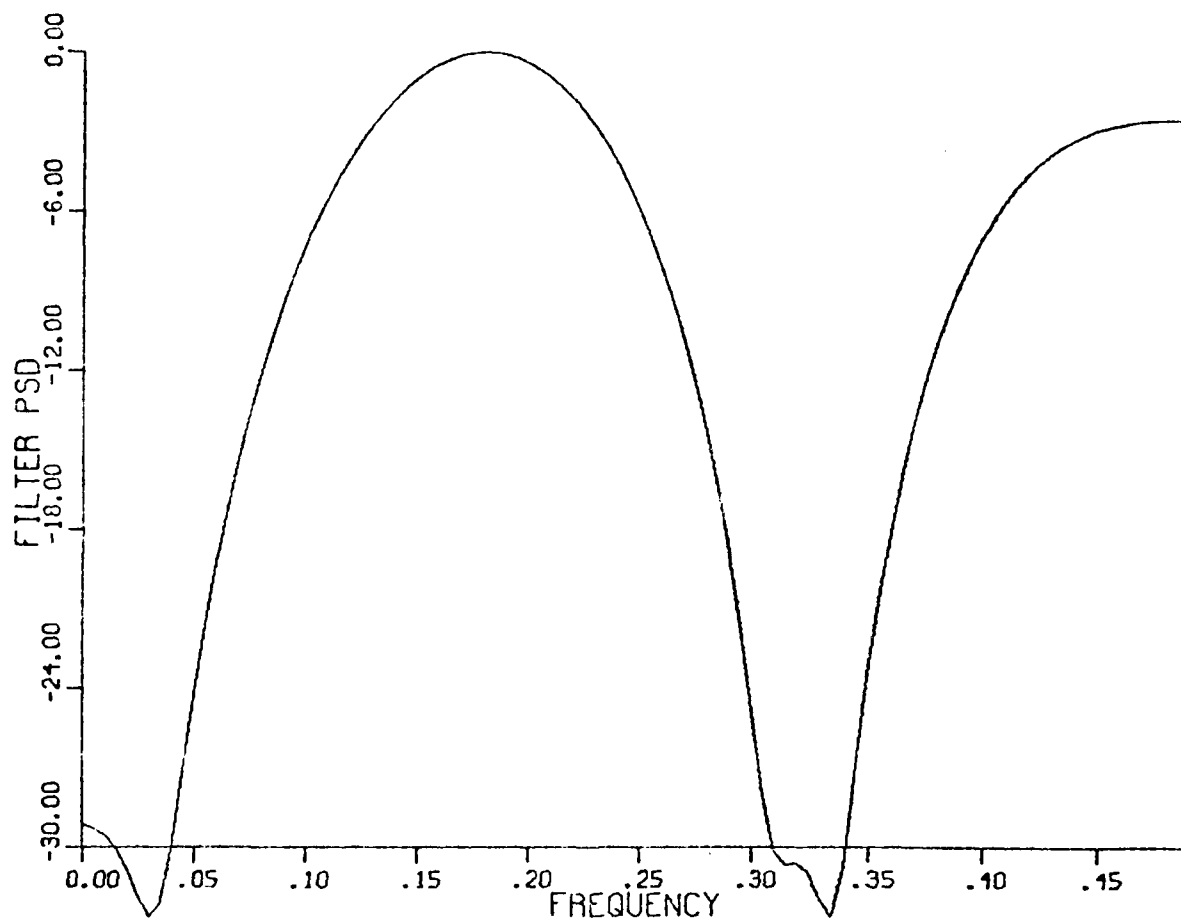


Figure 3.6 Frequency Response for Eight-Tap Filter

YVSTAT, NV=.01

FILTER ORDER=16.

SNR/CHIP WITHOUT FILTERING = - 20dB

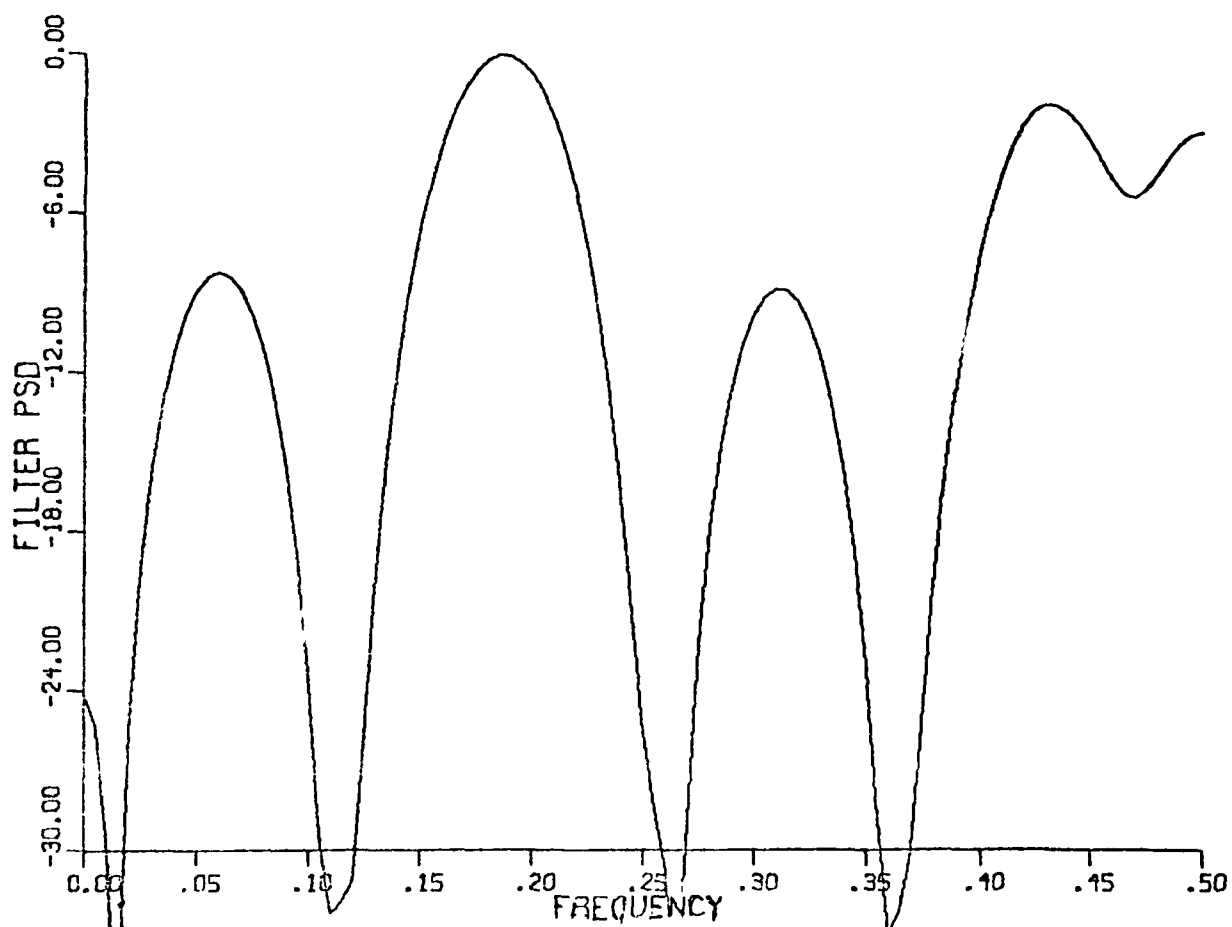


Figure 3.7 Frequency Response for Sixteen-Tap Filter

YVSTAT, NV=.01

FILTER ORDER=29.

SNR/CHIP WITHOUT FILTERING = -20dB

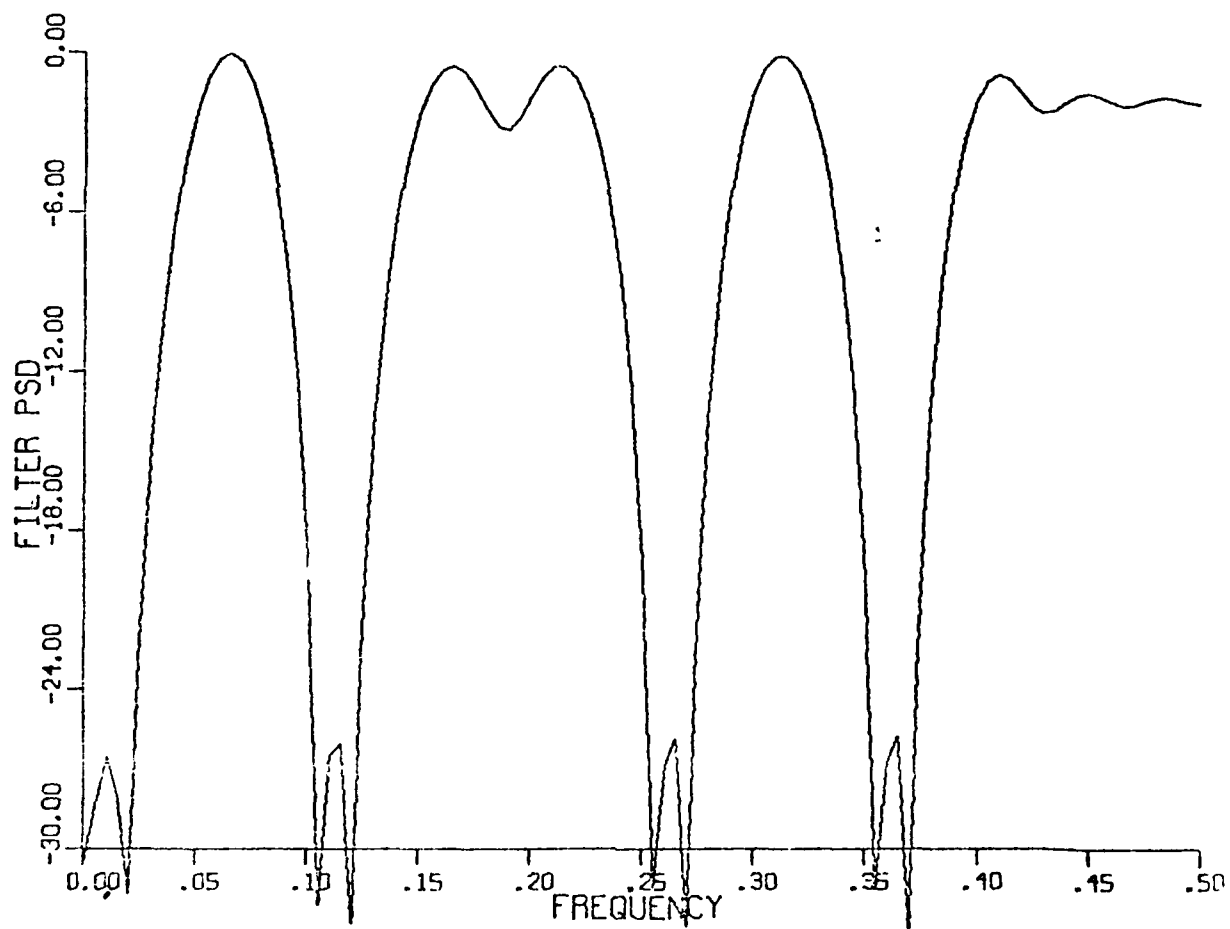


Figure 3.8 Frequency Response for 29-Tap Filter

sixteen-band interference. Figure 3.9 shows the improvement factor as a function of the number of filter taps when the SNR per chip is -20dB. There appear to be two threshold regions in these graphs. When the interference occupies eight bands, the first threshold occurs at 16 taps, where basically one complex-conjugate zero pair is assigned to each interference band. The second threshold occurs at 32 taps, in which case there are two complex-conjugate zero pairs assigned to each interference band. When the interference occupies sixteen bands, the first threshold occurs at 32 taps and the second threshold occurs at about 64 taps. The improvement factor as a function of the SNR per chip without filtering is shown in Figure 3.10 for a 40-tap predictor operating with eight interference bands and an 80-tap predictor operating with sixteen interference bands. The performance gain is very similar to that obtained when the interference is spread over fewer bands.

The conclusion that we have reached from observation of the above results is that the filter will suppress the multi-band interference provided that it has enough degrees of freedom, i.e., it is sufficiently long, to assign at least one complex-conjugate pair of zeros to each band. This behavior is substantiated further by the following frequency response characteristics of the suppression filter when there are eight bands of interference. Figures 5.11 through 5.14 illustrate the frequency response of filters corresponding to predictors of order 8, 16, 32 and 48, respectively. We observe that a filter with eight degrees of freedom is basically an all-pass filter. It does not have enough degrees of freedom to place notches at the eight frequency bands. On the other hand, with 16 degrees of freedom it does place the notches at the desired frequencies.

Figure 3.9 Improvement Factor as a Function of Filter Order for
8-Band and 16-Band Interference

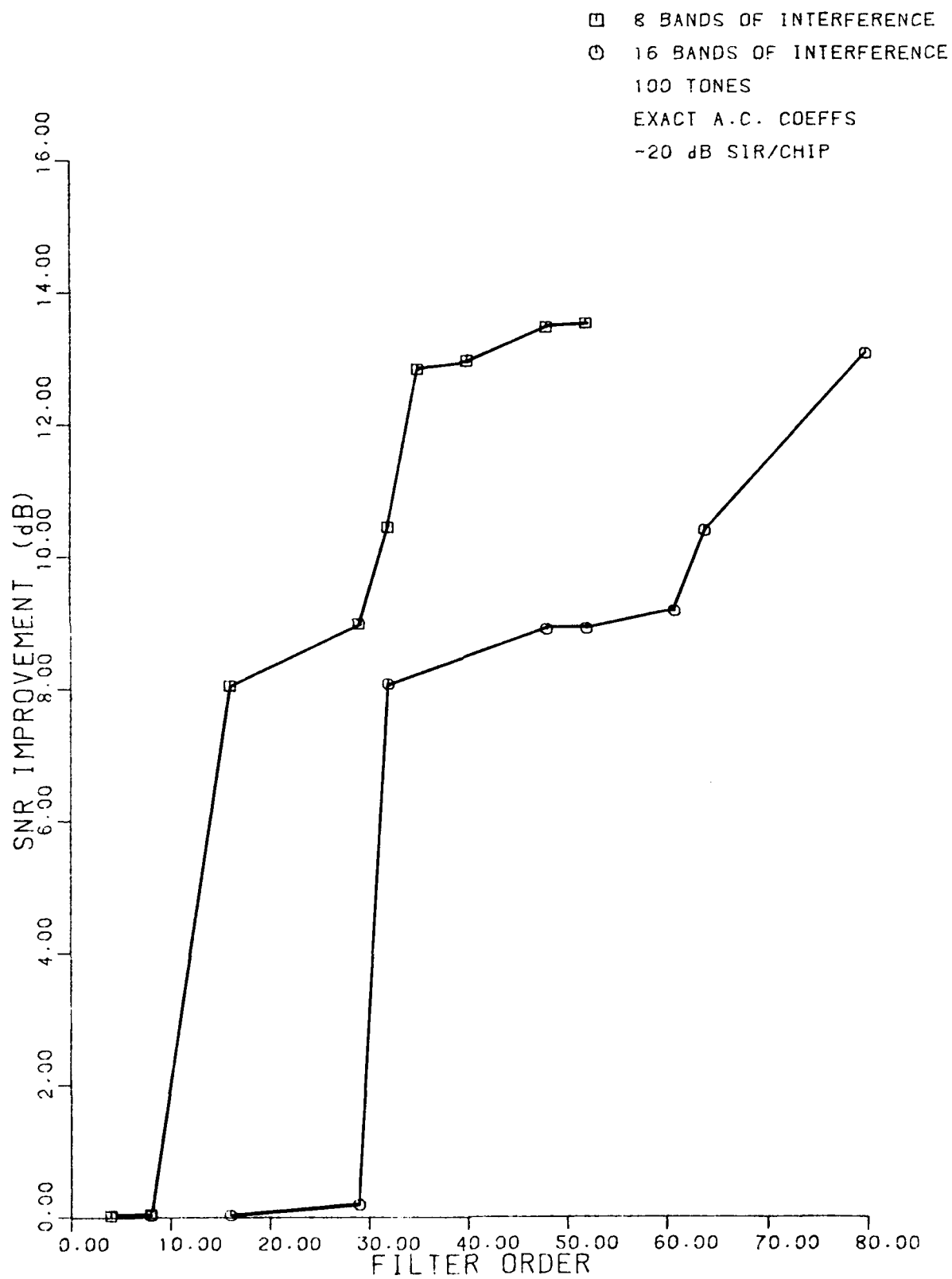
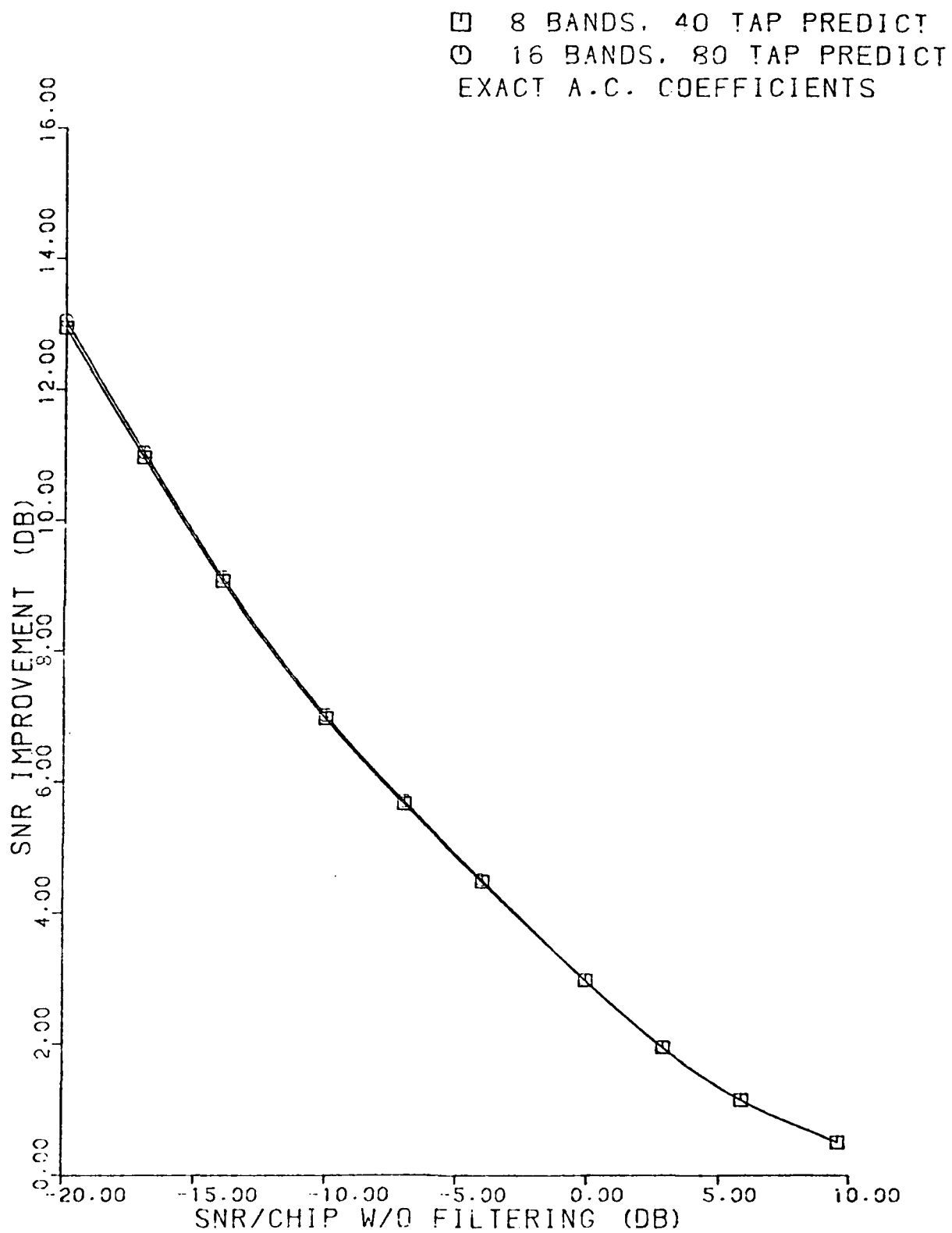


Figure 3.10 Improvement Factor for 40-Tap and 80 Tap Predictors



YVSTAT, NV = . 01 # BANDS=8.000

INTERFERENCE VARIANCE= 100.0000 SNR/CHIP WITHOUT FILTERING = -20dB

FILTER ORDER= 8.

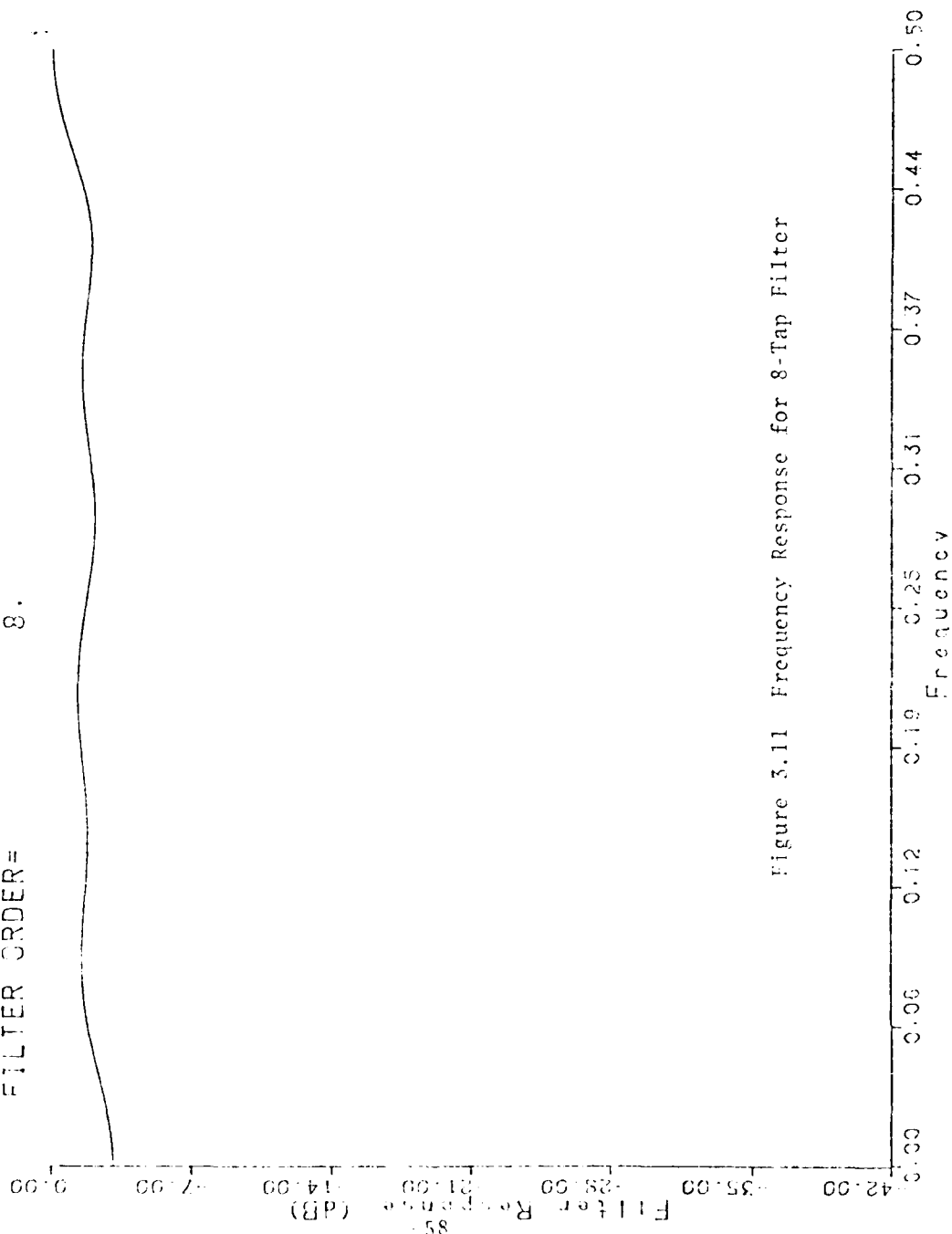


Figure 3.11 Frequency Response for 8-Tap Filter

YVSTAT, NV=. 01 # BANDS=8.000

INTERFERENCE VARIANCE= 100.0000

FILTER ORDER= 16.

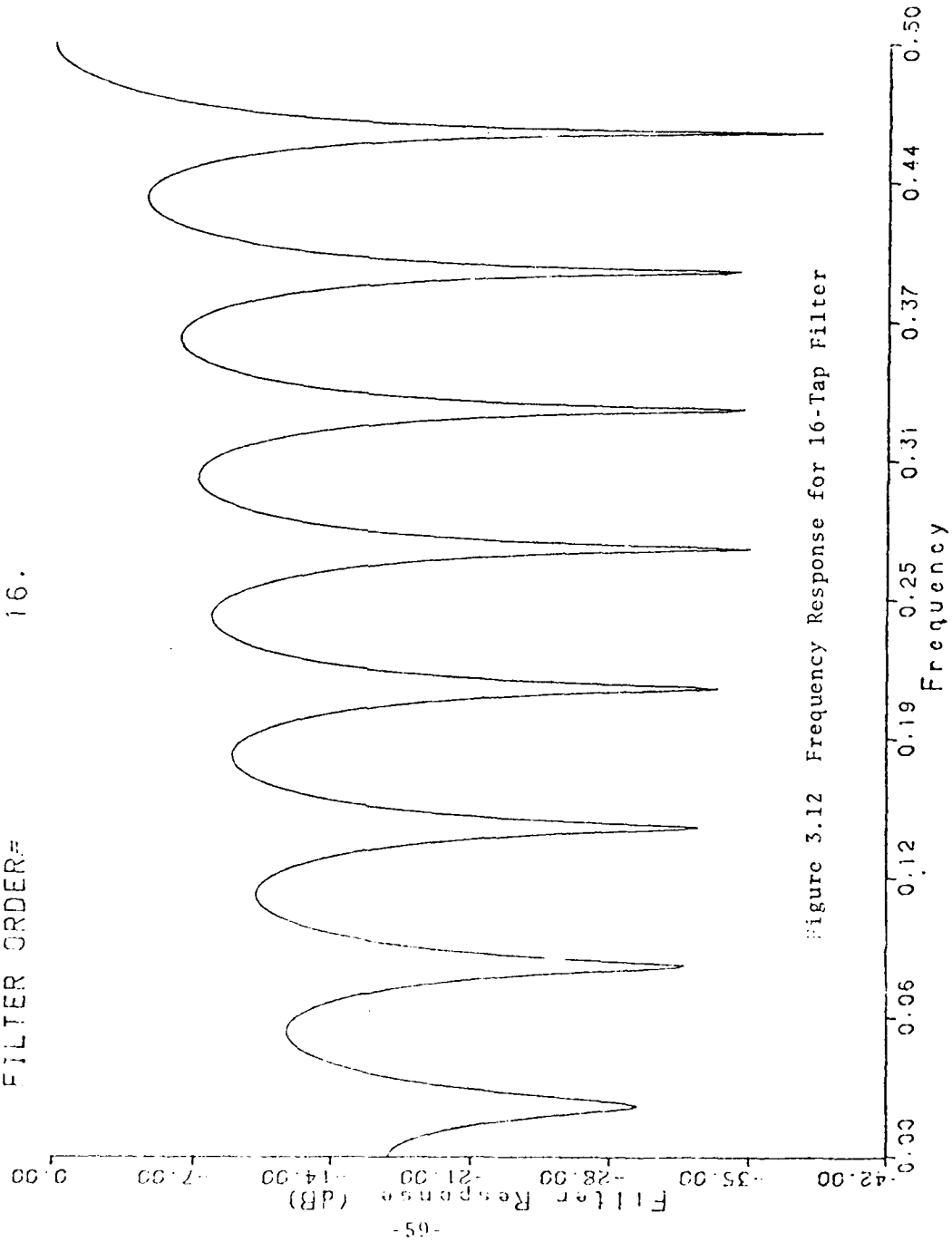
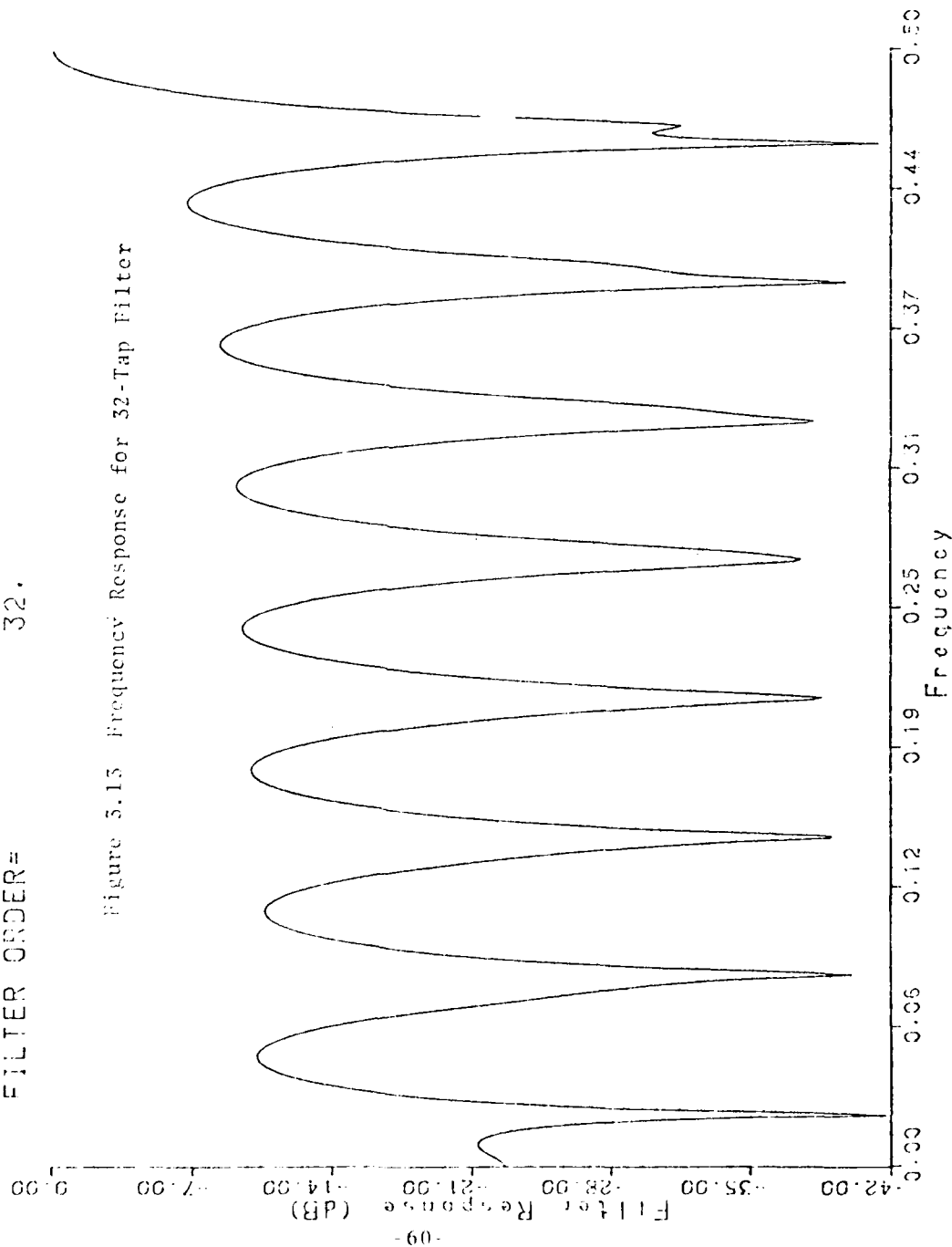


Figure 3.12 Frequency Response for 16-Tap Filter

YVSTAT, NV=. 01 # BANDS=8.000

INTERFERENCE VARIANCE= 100.0000

FILTER ORDER= 32.



YVSTAT, NV=. 01 # BANDS=8.000

INTERFERENCE VARIANCE= 100.0000

FILTER ORDER= 48.

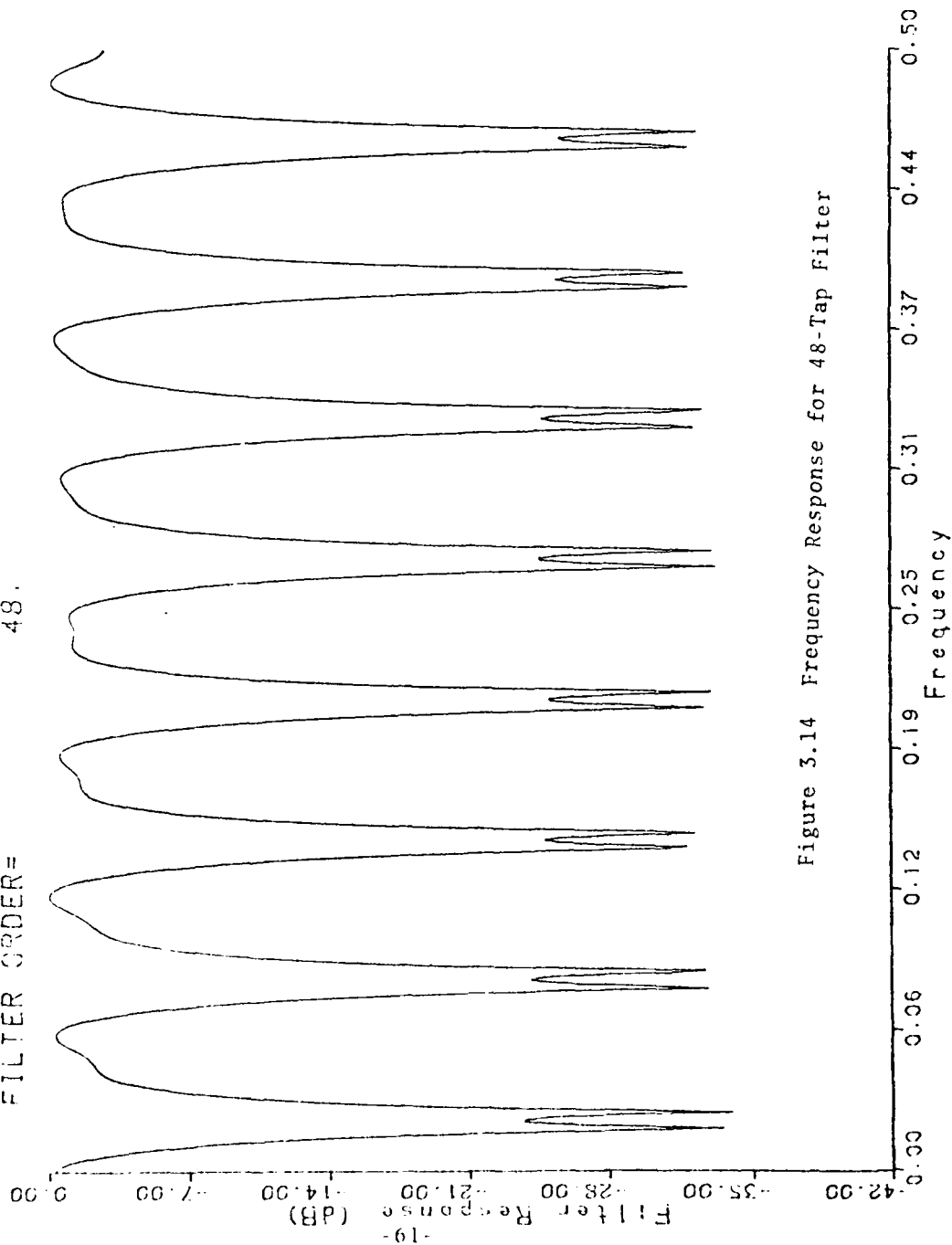


Figure 3.14 Frequency Response for 48-Tap Filter

For higher-order predictors, the frequency response is improved as shown in Figures 3.13 and 3.14.

Another view of the characteristics of the suppression filter is provided by the position of its zeros in the z -plane. For example, Figure 3.15 illustrates one-half of the unit circle with the positions of the zeros for the suppression filter corresponding to 2, 4, 6, 8, 10, 12-order predictors. In this case, the interference is concentrated in two bands. As shown in the plot, a second-order predictor places its complex-conjugate pair of zeros far from the unit circle and roughly midway (in angle) between the two interference bands. Thus, its performance is poor. However, a fourth or higher-order predictor does have zeros within the interference regions.

The above results indicate that a prediction filter having a number of coefficients that is fewer than twice the number of interference bands is useless, in the sense that it does nothing, i.e., it is an all-pass filter. Apparently, this is a limitation of the mean square error criterion used to design the prediction filter. If one knows that the number of degrees of freedom is fewer than twice the number of interference bands, an ad hoc scheme such as arbitrarily assigning a complex-conjugate pair of zeros to each band, up to the maximum number of bands that can be suppressed with the given number of degrees of freedom, appears to be better.

Since we are dealing with a digital communication problem, we have also investigated the characteristics of the interference suppression filter as a noise-whitening filter in a matched filter realization. That is, we view the combined narrow band interference plus wideband noise as

O INTERFERENCE VARIANCE= 100.00
 2 TAP PREDICTOR
 A INTERFERENCE VARIANCE= 100.00
 4 TAP PREDICTOR
 + INTERFERENCE VARIANCE= 100.00
 X INTERFERENCE VARIANCE= 100.00
 6 TAP PREDICTOR
 o INTERFERENCE VARIANCE= 100.00
 8 TAP PREDICTOR
 x INTERFERENCE VARIANCE= 100.00
 10 TAP PREDICTOR
 A INTERFERENCE VARIANCE= 100.00
 12 TAP PREDICTOR
 TWO BANDS, 100 TONES

SNR/CHIP WITHOUT FILTERING = -20dB

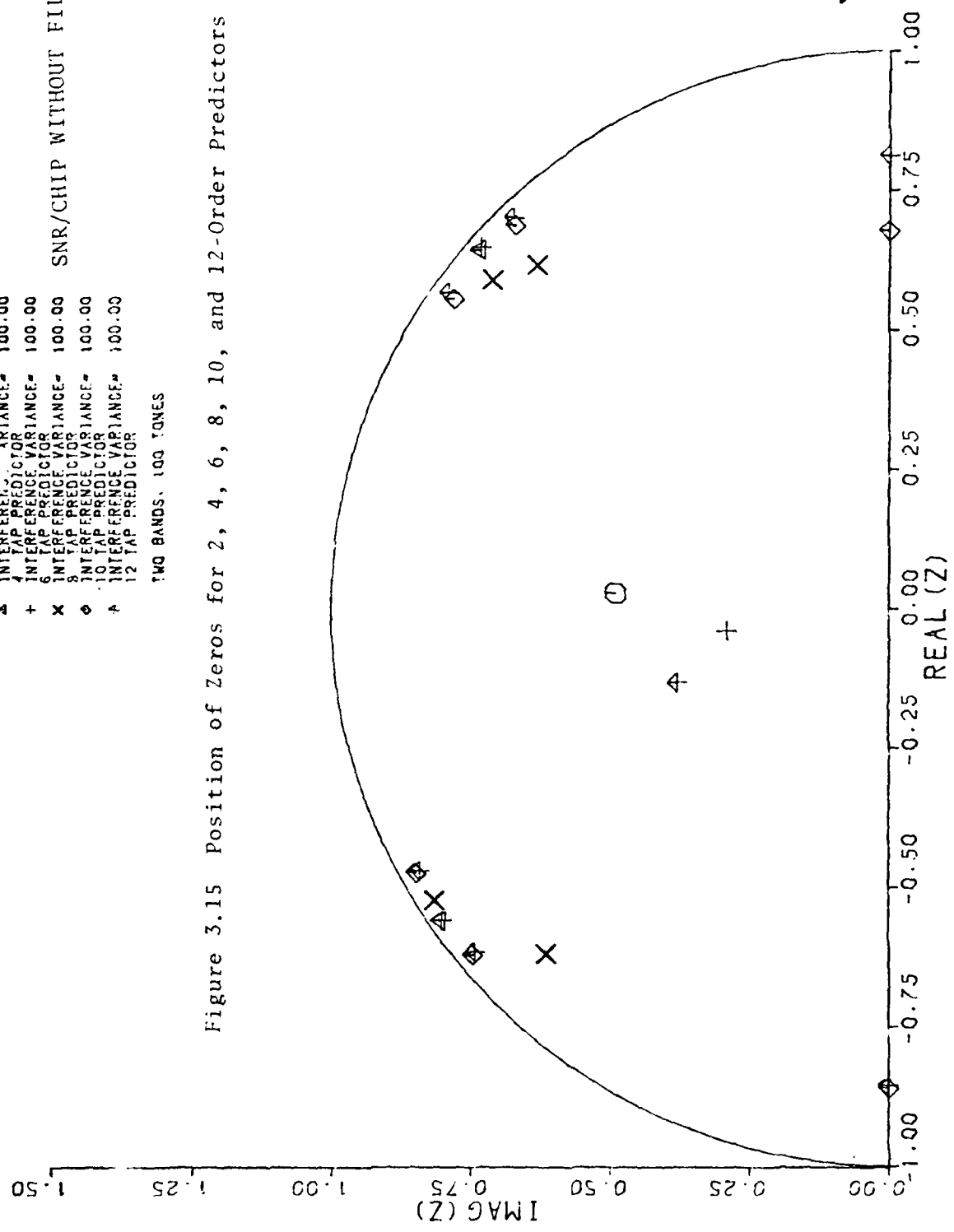
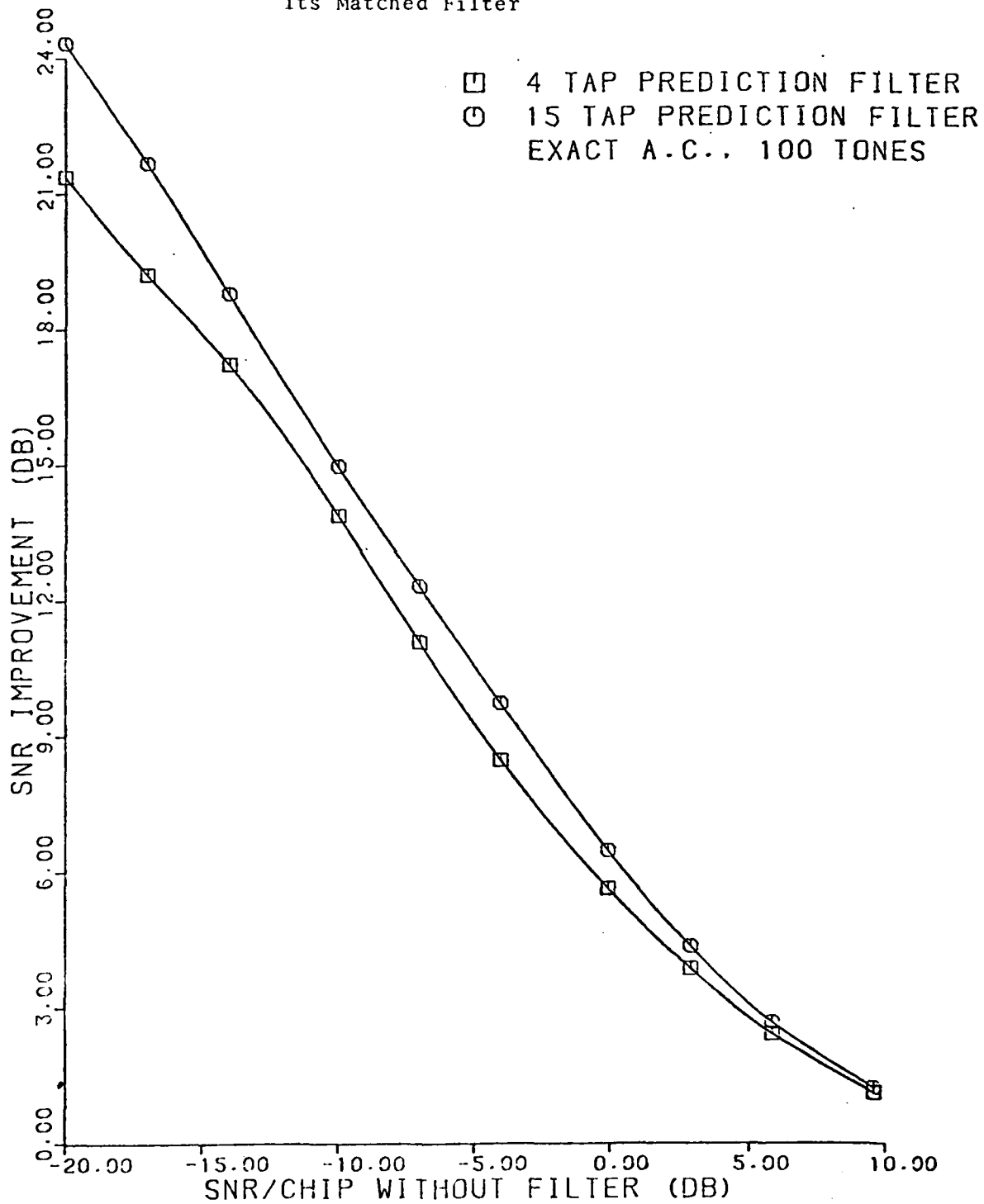


Figure 3.15 Position of Zeros for 2, 4, 6, 8, 10, and 12-Order Predictors

an equivalent colored noise process. Now, in detection of a signal $S(f)$ in colored noise with power spectral density $P(f)$, the output SNR at the receiver is maximized when the receiver consists of a noise-whitening filter, say $H(f)$, followed by a filter matched to $H(f) S(f)$. Thus, a matched filter with frequency response characteristic $H^*(f) S^*(f)$ will maximize the output SNR. If $H(f)$ represents the interference suppression filter with impulse response $h(t)$, then $H^*(f)$ represents a filter with impulse response $h(-t)$. Thus, the cascade of these two filters is a filter having an even impulse response. Since we have determined the coefficients of $H(f)$ by means of linear prediction, the coefficients of $H^*(f)$ are simply the time reverse of those obtained for $H(f)$. Therefore, the cascade of $H(f)$ and $H^*(f)$ results in a linear phase filter. Use of such a filter prior to the PN correlator improves performance. This is illustrated in Figure 3.16 for a four-tap and a fifteen-tap predictor. We observe that at -20dB per chip SNR the four-tap predictor in cascade with its matched filter provides about 21dB of improvement. The fifteen-tap predictor with its matched filter provides about 24dB of improvement. In comparison, the four-tap predictor without its matched filter provides about 13dB of improvement. Therefore, the inclusion of the matched filter has resulted in about 8dB gain at a -20dB SNR per chip. Such a large gain is highly significant and suggests that the use of the matched filter is very desirable.

Finally, we turn our attention to the performance characteristics of the filters designed from the data by means of the prediction algorithms described in Section II. For this discussion it suffices to consider only single band interference. The main point that we wish to make with regard

Figure 3.16 Improvement Factor for Predictor in Cascade with its Matched Filter

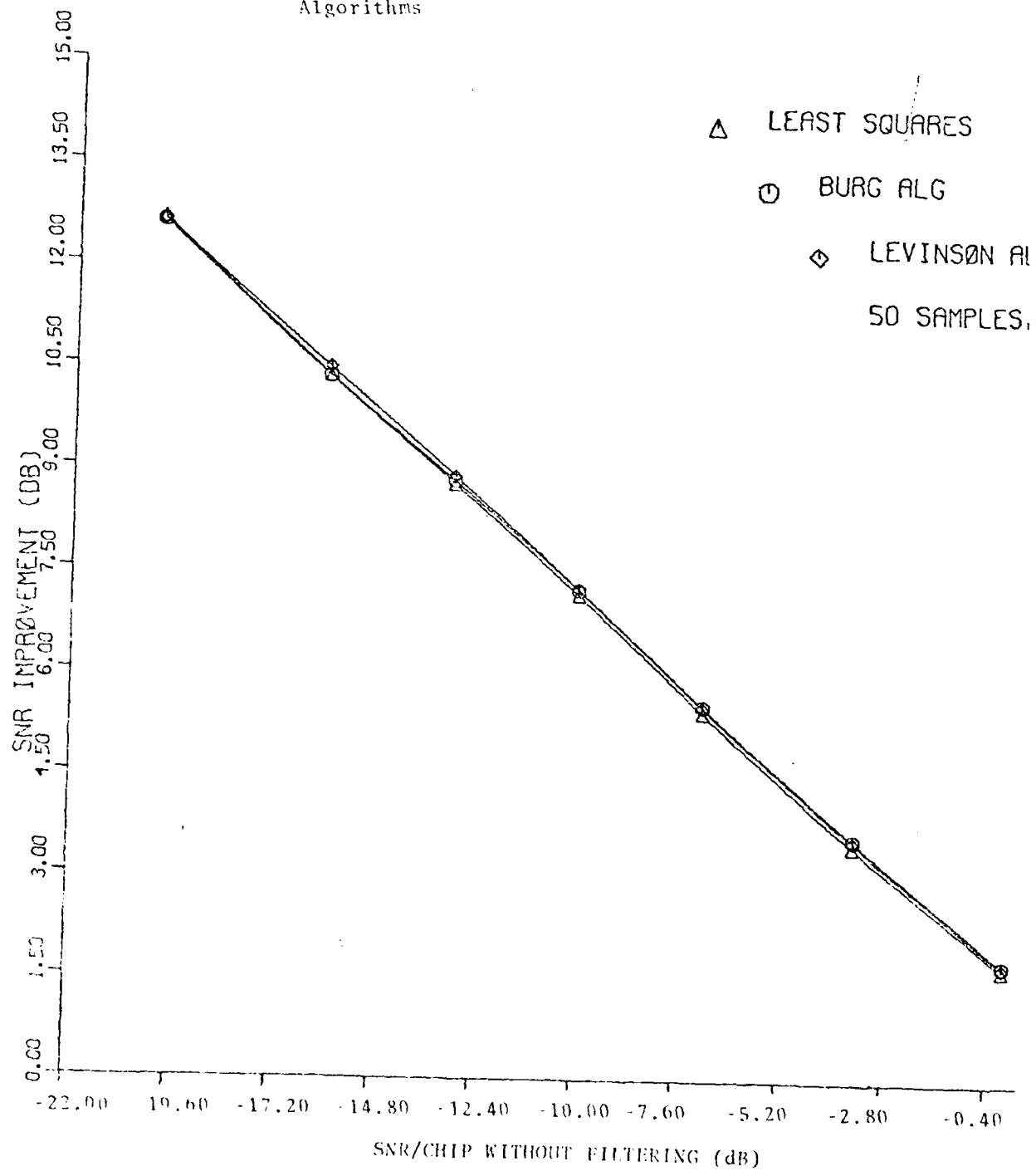


to the performance of the algorithms is best illustrated with the results shown in Figure 3.17. We used a block of fifty data samples to compute the coefficients of a fourth-order predictor by means of the least squares algorithm, the Burg algorithm and the Levinson algorithm. For the latter, the data was used to generate the estimate of the autocorrelation function. The predictor coefficients obtained from the data were used in the computation of the improvement factor given by Eq. (3.9). The graphs indicate that all three algorithms perform equally well. In other words, the difference in performance among the three algorithms is insignificant. This behavior is further substantiated by observing the corresponding frequency response characteristics of the suppression filter. For example, Figures 3.18, 3.19 and 3.20 illustrate the frequency response characteristics of the suppression filter designed from fifty samples of data on the basis of the three algorithms. Here, we also observe very minor differences in the frequency response characteristics. On the other hand, when the order of the predictor is a large fraction of the data record length N , the Burg algorithm and the least squares algorithm are expected to yield better performance relative to the Levinson algorithm.

5.5 Performance of Interference Suppression Filter Based on Nonparametric Spectral Estimates

In Section 2.1 we described a method for designing an interference suppression filter based on conventional, nonparametric methods for spectral estimation. As an illustration of the effectiveness of this approach, we computed estimates of the power spectral density from simulated received data and used the resultant estimates to specify a filter

Figure 3.17 Improvement Factor Obtained with Linear Prediction Algorithms



LSQRS, LEAST SQUARES

RECORD LENGTH=50.

FILTER ORDER=1.

NO OF TONES=100

SNR/CHIP WITHOUT FILTERING = -20dB

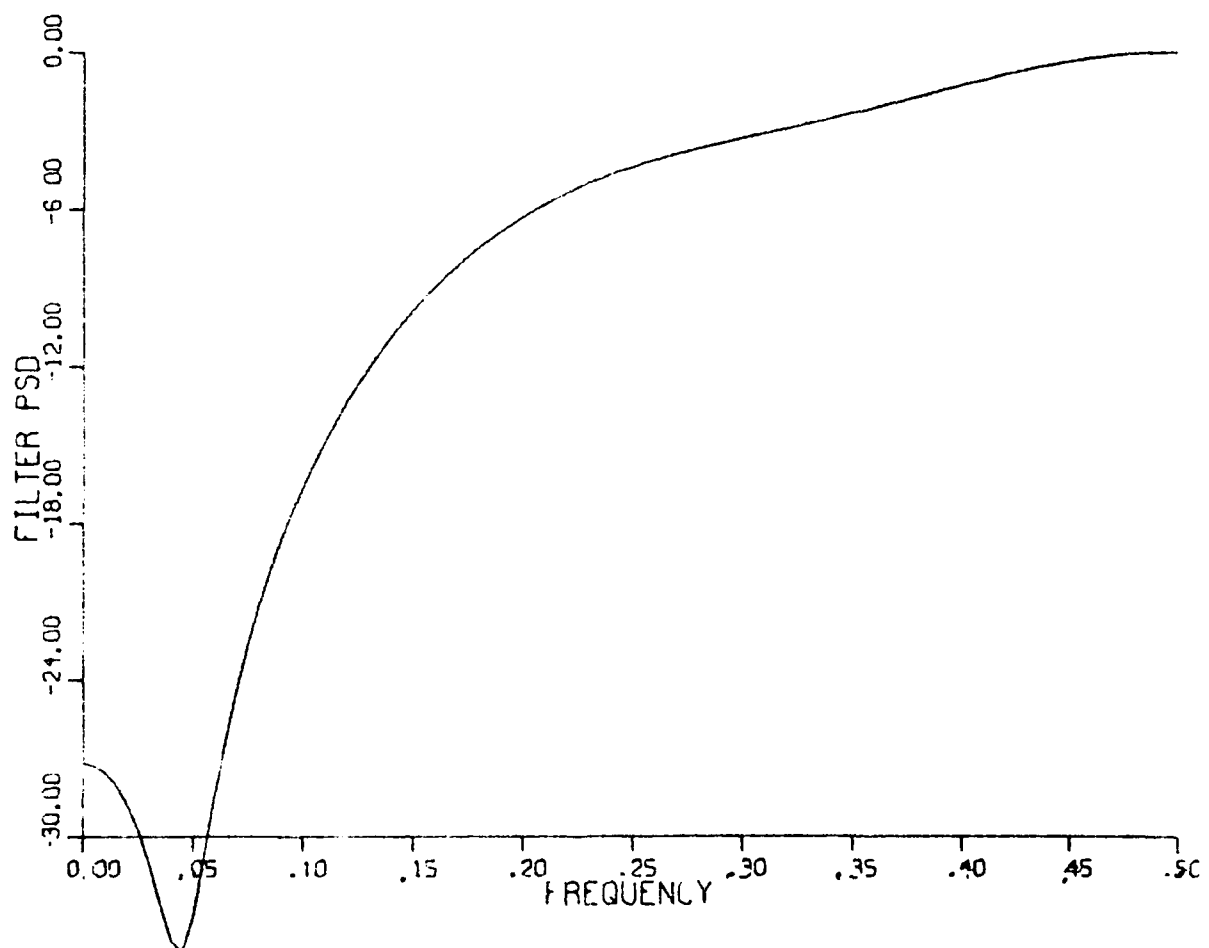


Figure 3.18 Frequency Response of Filter Designed from Least Squares Algorithm

MFMPR, BURG ALG

RECORD LENGTH=50.

FILTER ORDER=4.

NØ OF TONES=100

SNR/CHIP WITHOUT FILTERING = -20dB

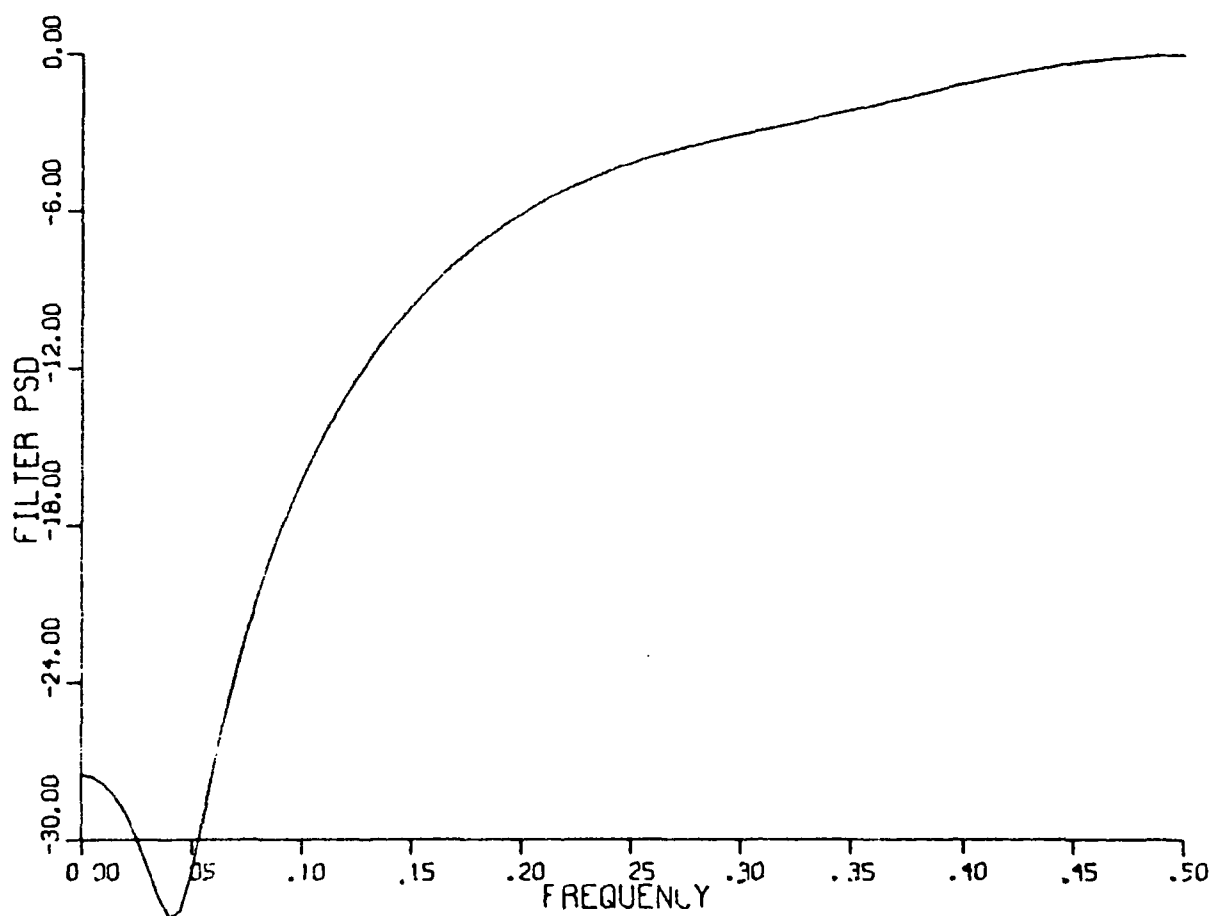


Figure 3.19 Frequency Response of Filter Designed from Burg Algorithm

YWPR, LEVINSON ALG

RECORD LENGTH=50.

FILTER ORDER=4.

NO OF TONES=100.

SNR/CHIP WITHOUT FILTERING = - 20dB

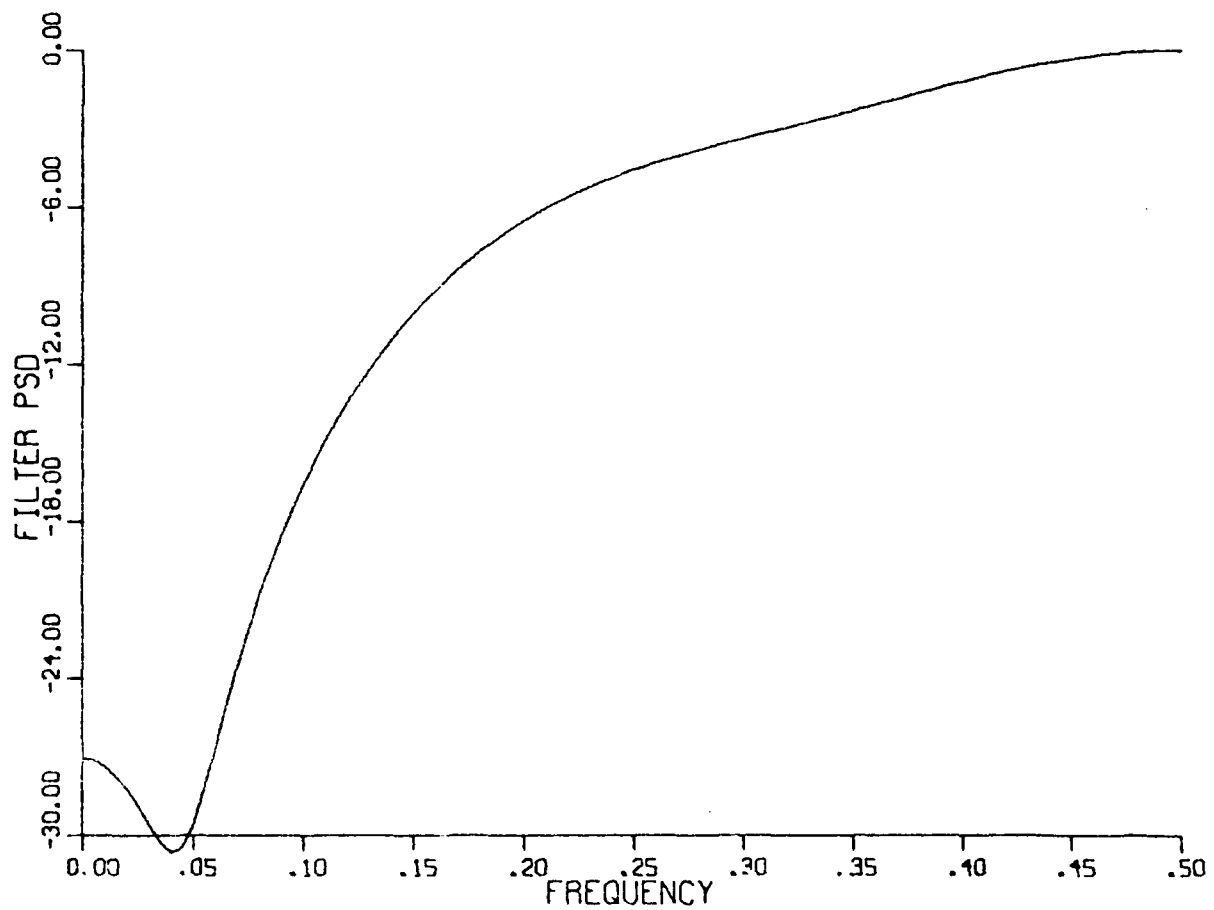


Figure 3.20 Frequency Response of Filter Designed from Levinson Algorithm

characteristic in accordance with Eq. (2.2). The Welch method [5] was used to generate the estimates of the power spectral density. For this computation, the FFT size selected was 64 points. The resultant interference suppression filter consists of fifteen taps. The number of data points used to generate the spectral estimate is 992.

Figure 3.21 illustrates the estimate of the power spectral density for an SNR per chip of -20dB. The corresponding frequency response of the fifteen-tap interference suppression filter is shown in Figure 3.22. The interference occupied 20% of the signal band as shown in the graph of the estimate, and the filter contains a notch in the desired frequency band. For comparison, the spectral estimate shown in Figure 3.23 is for an SNR of -10dB per chip without filtering. The notch in the filter is now more shallow as shown in Figure 3.24. This behavior is similar to that obtained by means of linear prediction.

As a final computation, the coefficients $h(n)$ of the interference suppression filter were substituted into Eq. (3.9) and the improvement factor was evaluated. Figure 3.25 illustrates the improvement factor as a function of the SNR per chip without filtering. When compared with our previous results for single band interference using linear prediction, we find that the performance improvement has similar characteristics. Therefore, it appears that the nonparametric method for spectral estimation coupled with the filter design formula in Eq. (2.2) provides a viable means for suppressing narrow band interference in a wideband signal. The one disadvantage of this method is the relatively large sample size required to generate the spectral estimate.

INTERFERENCE VARIANCE= 100.00

SNR CHIP WITHIN FILTERING = 20dB

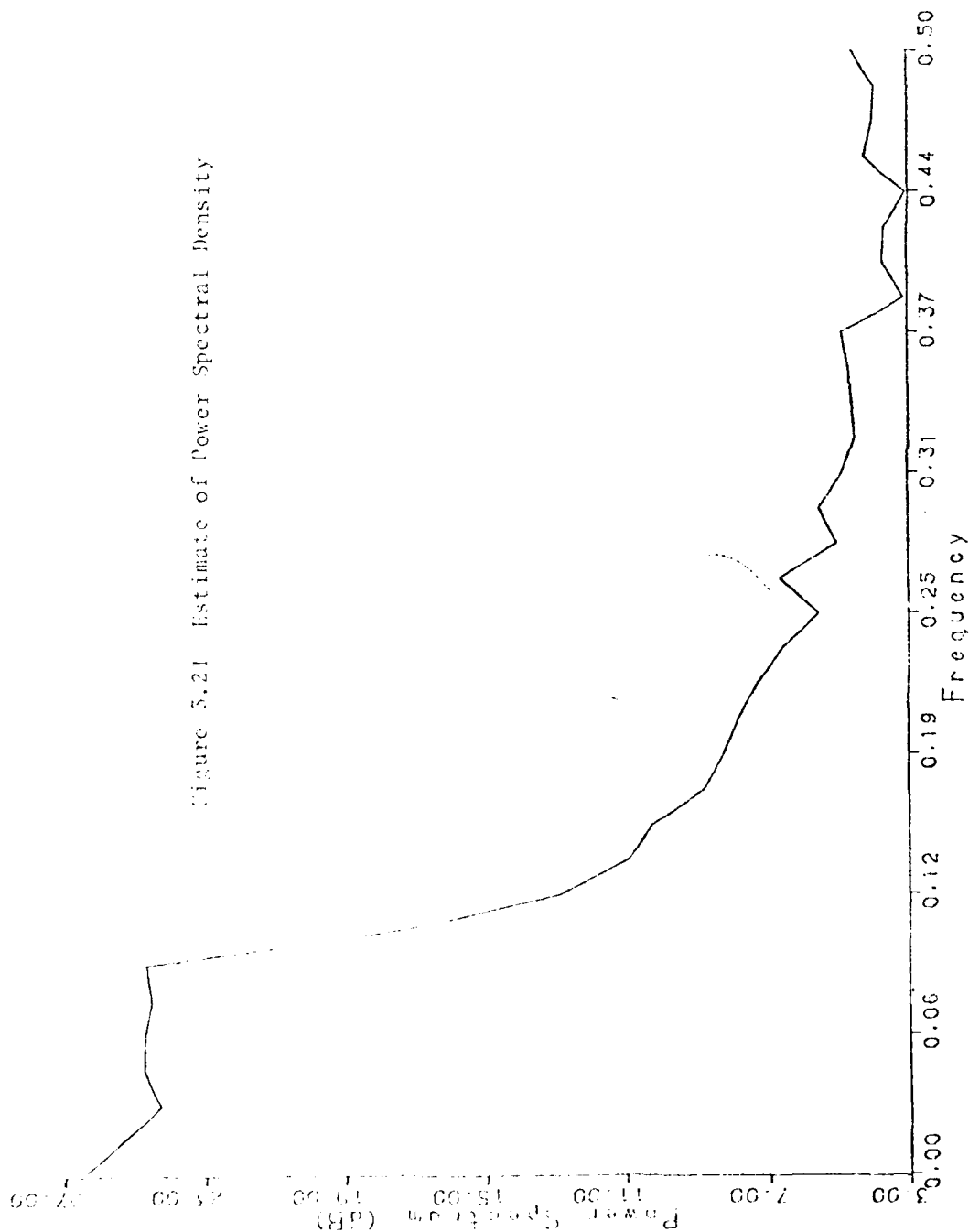


Figure 5.21 Estimate of Power Spectral Density

PER VARIANCE= 100.000

15.

LENGTH=

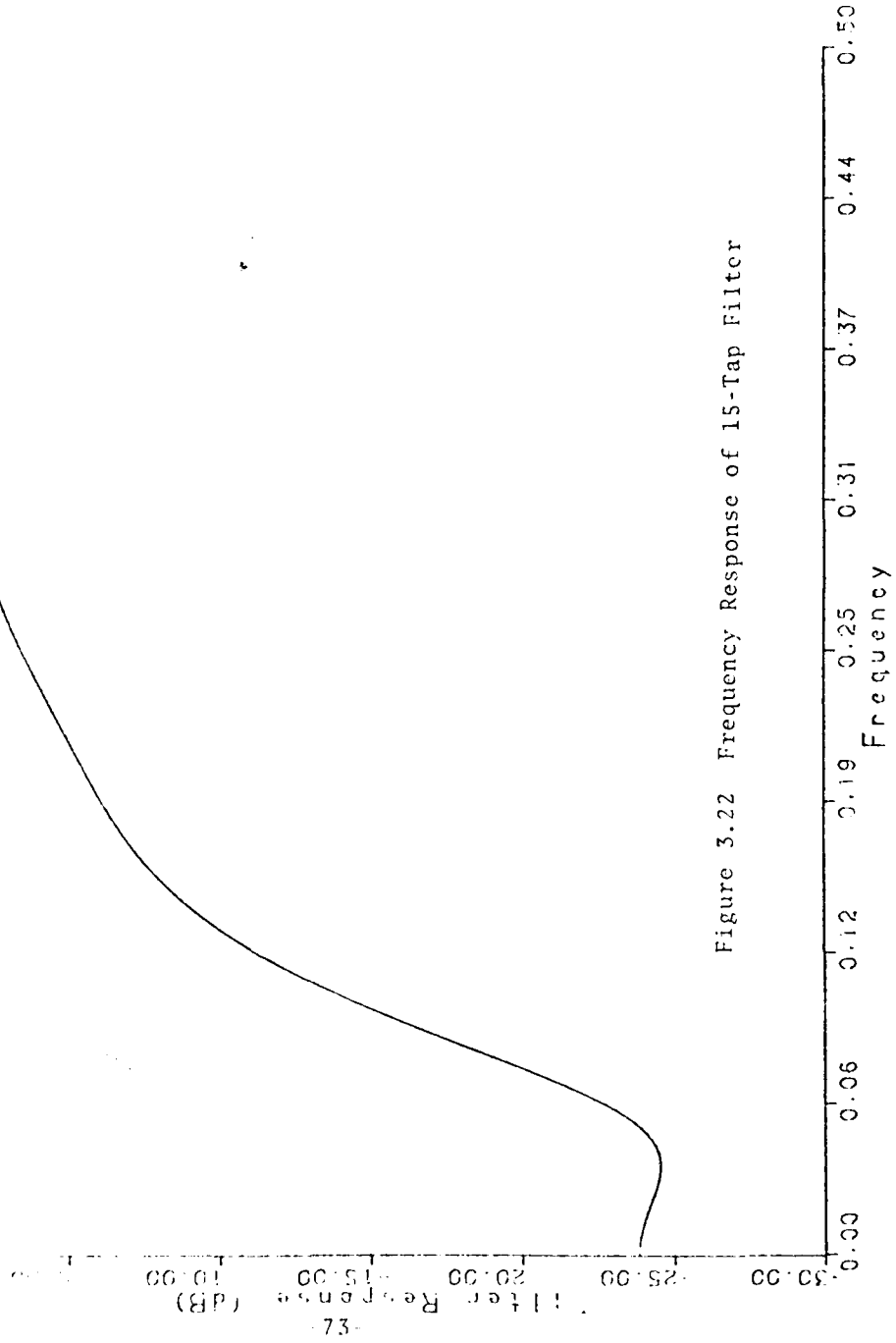
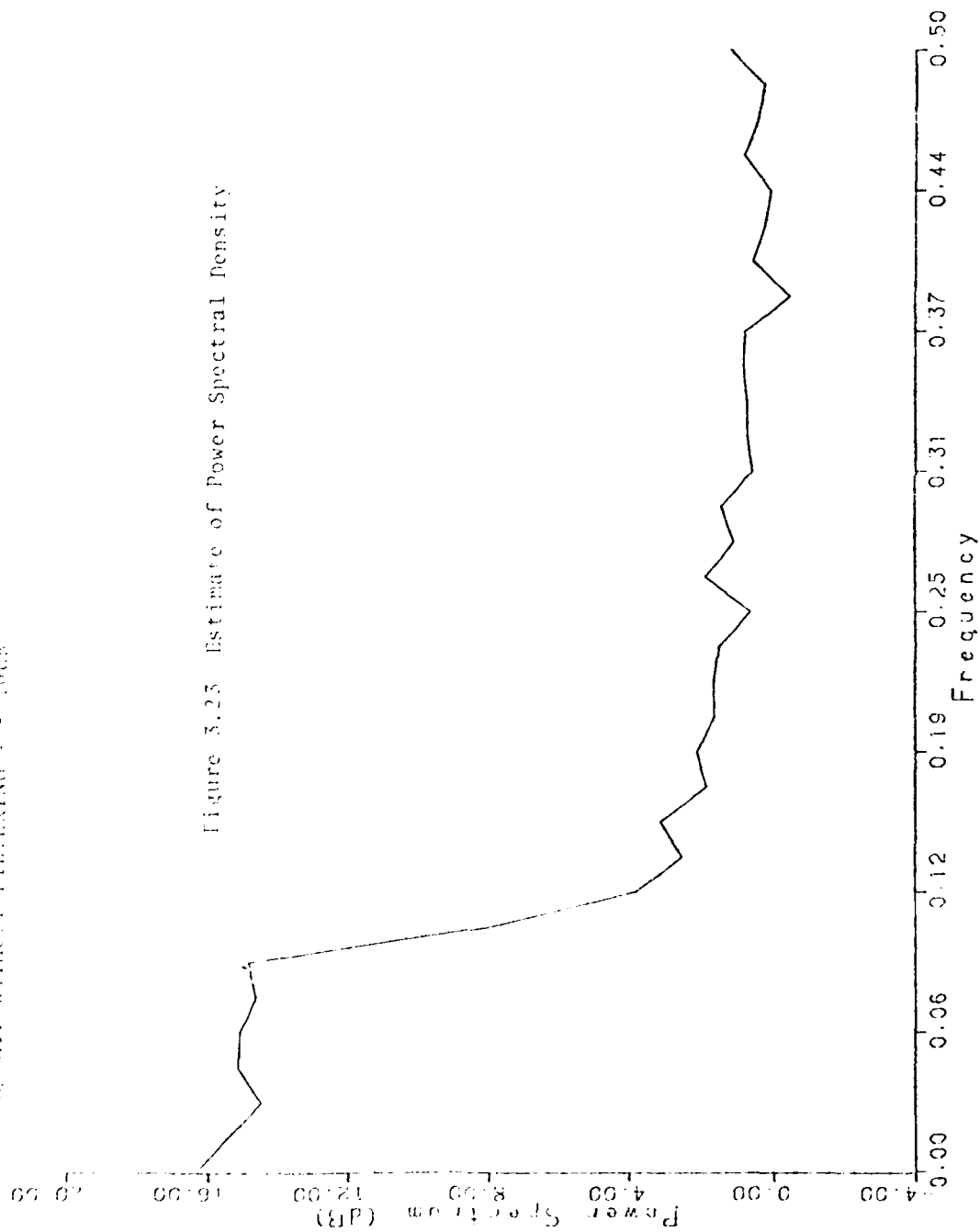


Figure 3.22 Frequency Response of 15-Tap Filter

INTERFERENCE VARIANCE= 10.00

SNR/CHIP WITHOUT FILTERING = -10dB



INTERFERENCE VARIANCE= 10.000

FILTER LENGTH= 15.

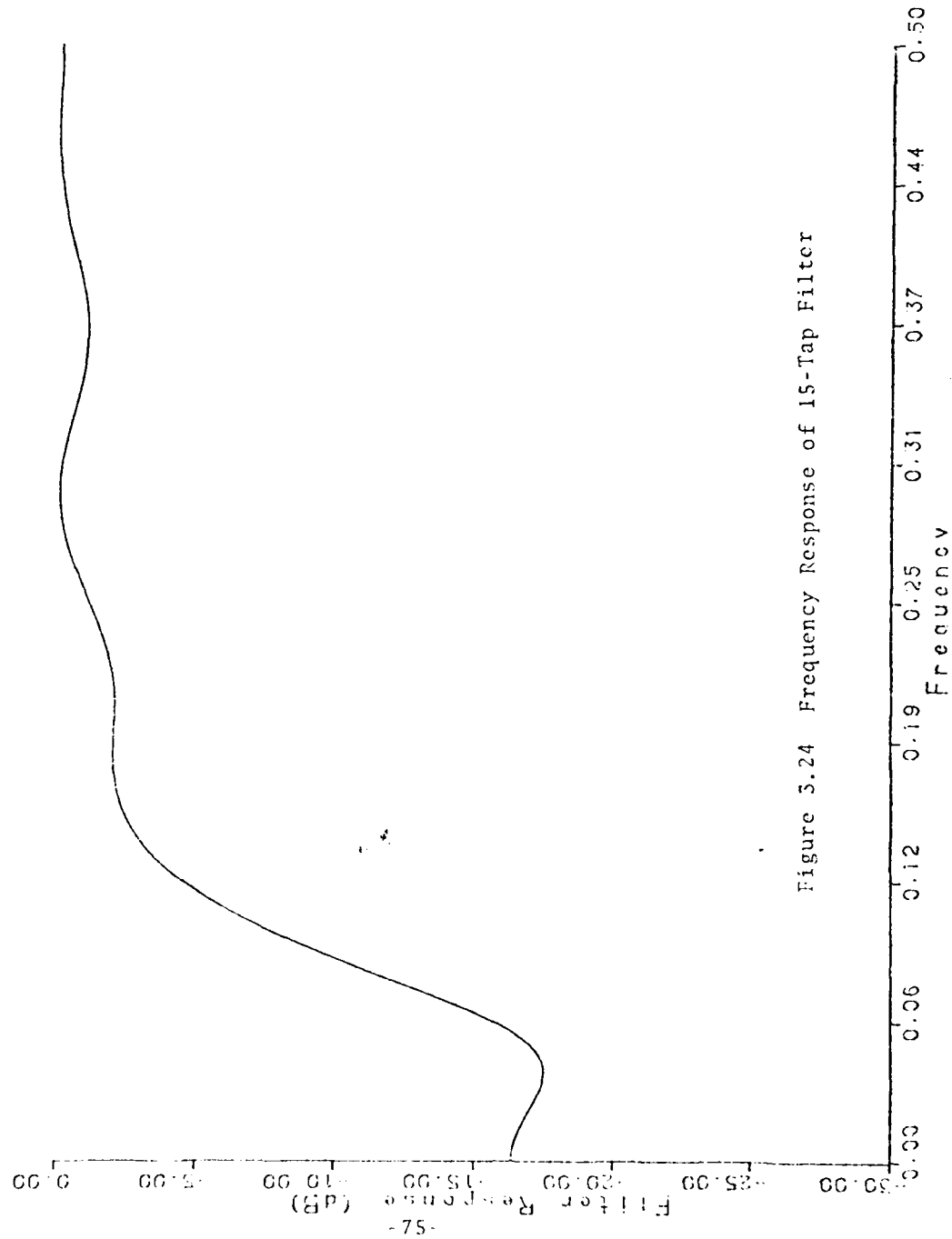
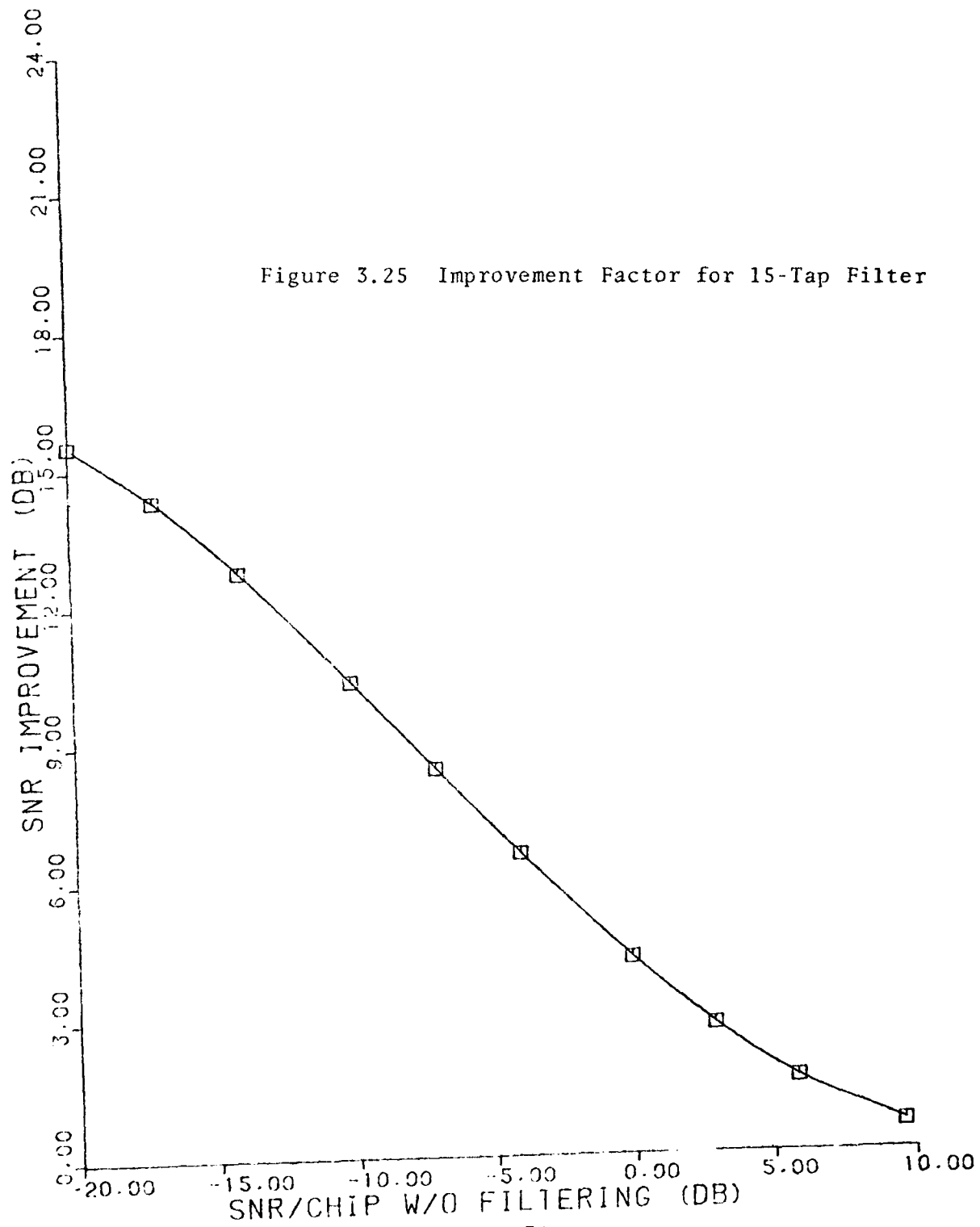


Figure 3.24 Frequency Response of 15-Tap Filter

□ 15 TAP LIN PHASE FILTER
BASED ON WELCH METHOD



5.4 Performance of Equalized PN Spread Spectrum System

This section of the report is devoted to the performance of the receiver in the presence of fading multipath signal components and narrow band interference. As indicated previously in Section 2.5, a major concern with the use of interference suppression in the presence of multipath is the sensitivity of the predictor to the channel multipath structure. Ideally, one would like to have the prediction filter respond only to the interference components. However, this is possible only if (1) the delay M , as illustrated in Figure 2.2, exceeds the duration of the channel multipath spread, i.e., use an M -step predictor, or (2) the delay M plus the time span of the prediction filter is less than or equal to the smallest time interval between successive multipath components of the received signal. The problem with condition (2) is that it places an unreasonably severe constraint on the length of the prediction filter. In practice, the multipath characteristics of the channel are not controllable to the extent necessary to satisfy condition (2). Consequently, this condition cannot be achieved realistically. On the other hand, imposition of condition (1) results in relatively poor estimates of the narrow band interference when the multipath spread is large. This is due to the fact that the interference $i(t)$ and its delayed version $i(t - M)$ are highly decorrelated when M is large. Consequently, the estimate of the interference at the output of the prediction filter is poor. This phenomenon was observed in simulation results on a two-path channel characteristic. The conclusion is that imposition of condition (1) leads to such poor performance to render it impractical.

Since the M -step predictor yields poor performance for large M due

to the decorrelation in the interference signal, we investigated the use of a one-step predictor in the presence of multipath. The following frequency response characteristics of the interference suppression filter resulted from one-step prediction of a signal consisting of narrow band interference plus two multipath signal components of equal strength. Single-band interference was employed in this computation. The exact autocorrelation function was used in the computation of the coefficients of a fourth-order predictor according to Eq. (2.8).

Figures 5.26 through 5.30 illustrate the frequency response of the suppression filter for varying amounts of narrow band interference and a multipath spread of two chips. We observe that a strong narrow band interference completely masks the signal multipath components and, hence, the filter, or one-step predictor, responds to the interference by placing a notch at the desired frequency band. On the other hand, when the interference is weak, the predictor responds primarily to the multipath components and attempts to implement an inverse channel filter. The transition point between a strong interference and a weak interference appears to be in the range where the signal and interference are comparable in power.

A similar set of curves was obtained for a multipath spread of four chips as illustrated by the frequency responses in Figures 5.31 through 5.35. Here, again, the suppression filter adapts to the inverse channel filter when the interference is small, but is unaffected by the multipath when the interference is large. On the other hand, if we increase the multipath spread beyond four chips, the four-tap predictor will not see the multipath component. This condition is illustrated by the frequency response curve in Figures 5.36 through 5.40 for a multipath spread

TWO PATH CHANNEL, DELAY= 2.000

INTERFERENCE VARIANCE= 200.0000

EXACT A.C. COEFFS, FILT LEN=4.

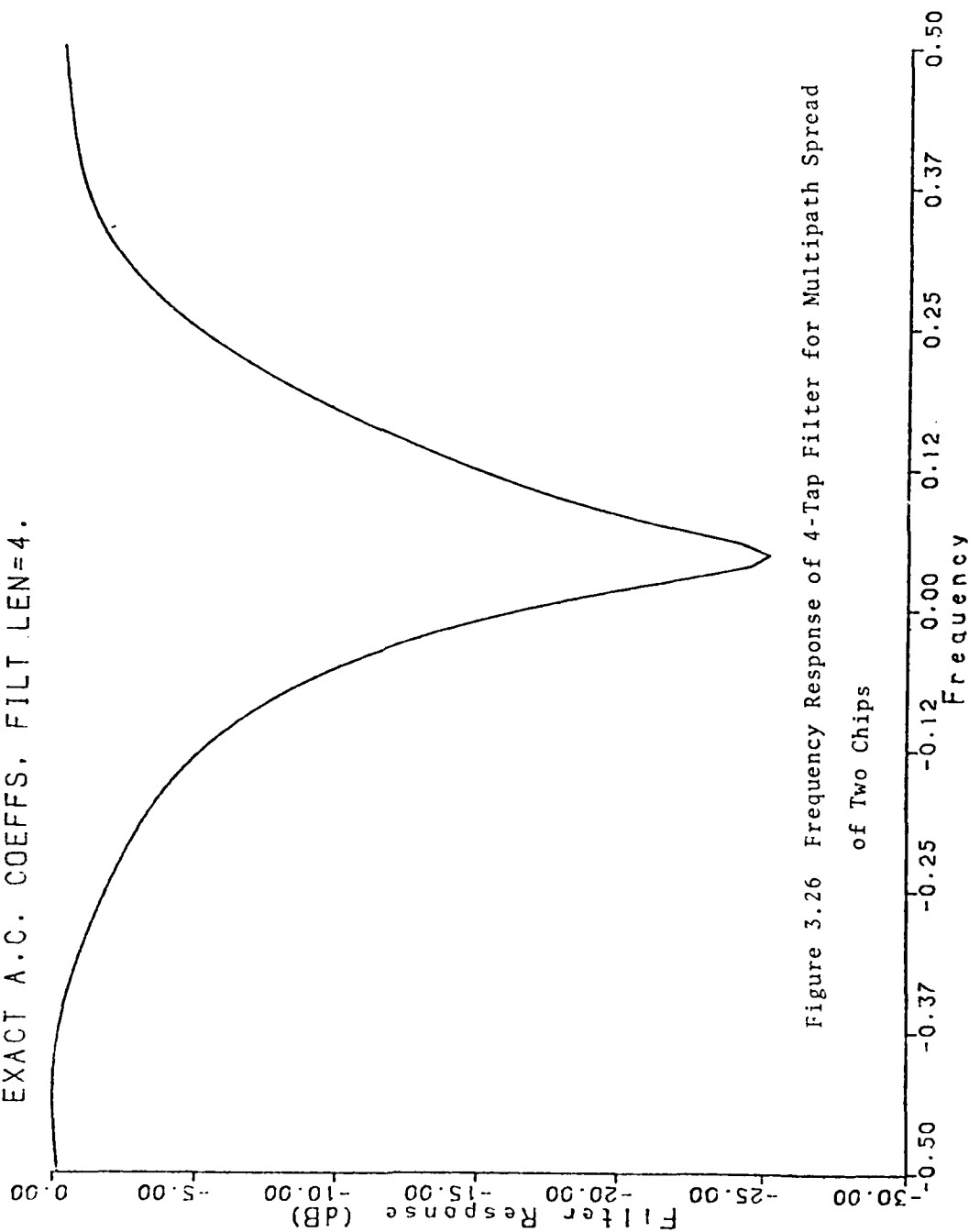


Figure 3.26 Frequency Response of 4-Tap Filter for Multipath Spread of Two Chips

TWO PATH CHANNEL, DELAY= 2.000

INTERFERENCE VARIANCE= 20.0000

EXACT A.C. COEFFS, FILT LEN=4.

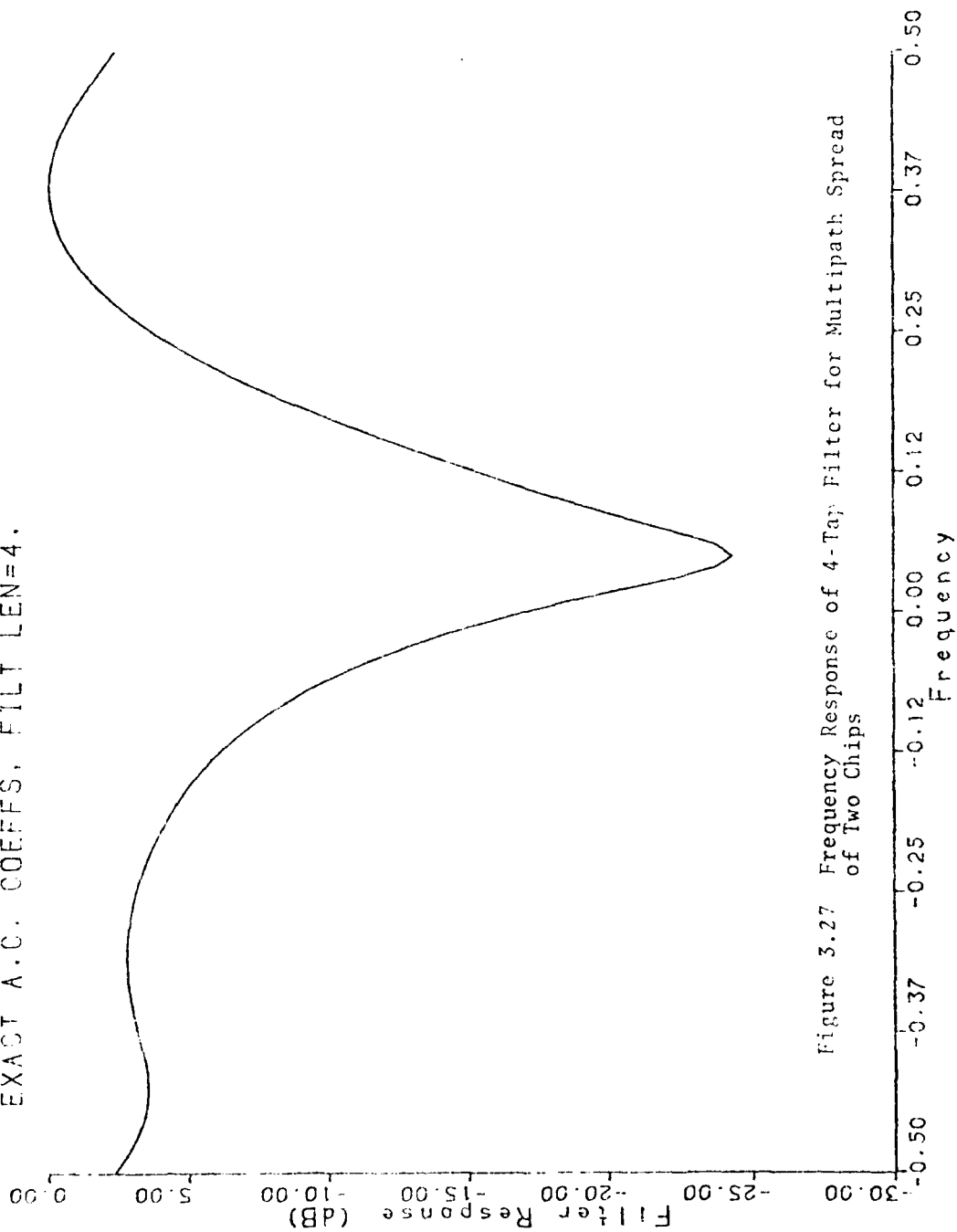


Figure 3.27 Frequency Response of 4-Tap Filter for Multipath Spread of Two Chips

TWO PATH CHANNEL, DELAY= 2.000
 INTERFERENCE VARIANCE= 2.0000
 EXACT A.C. COEFFS, FILT LEN=4.

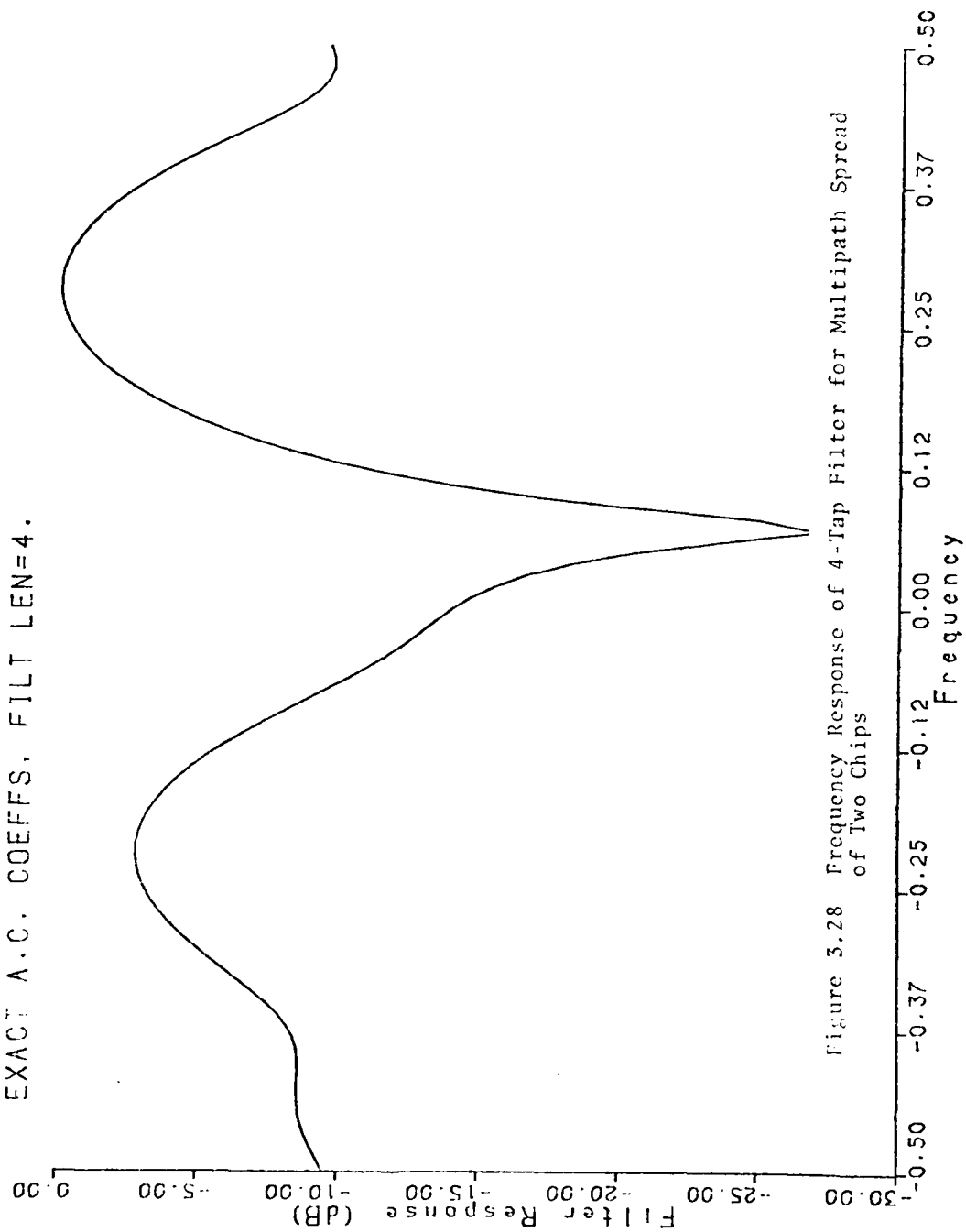


Figure 3.28 Frequency Response of 4-Tap Filter for Multipath Spread of Two Chips

TWO PATH CHANNEL, DELAY= 2.000
 INTERFERENCE VARIANCE= 0.2000
 EXACT A.C. COEFFS, FILT LEN=4.

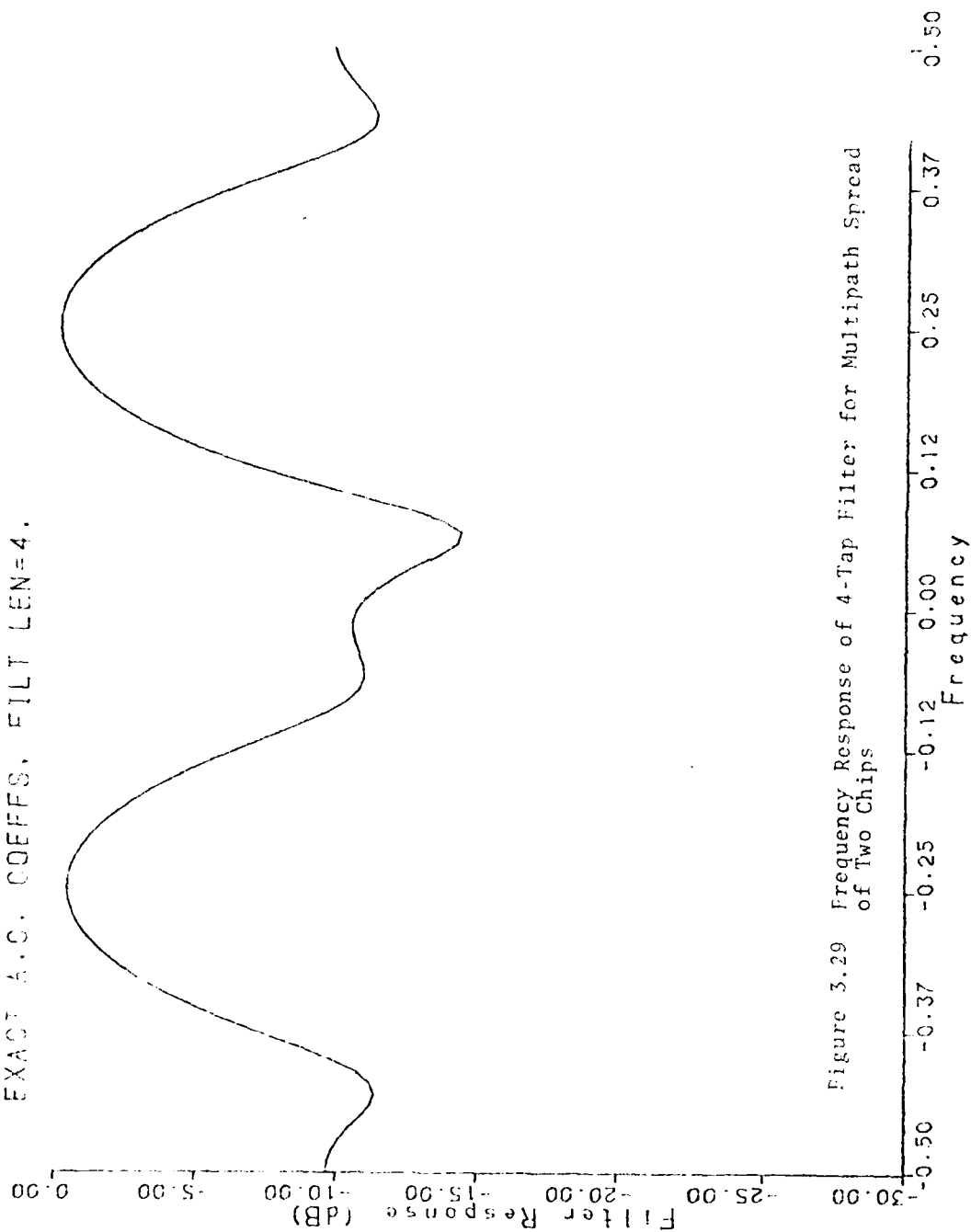


Figure 3.29 Frequency Response of 4-Tap Filter for Multipath Spread of Two Chips

TWO PATH CHANNEL, DELAY= 2.000
 INTERFERENCE VARIANCE= 0.0200
 EXACT A.C. COEFFS, FILT LEN=4.

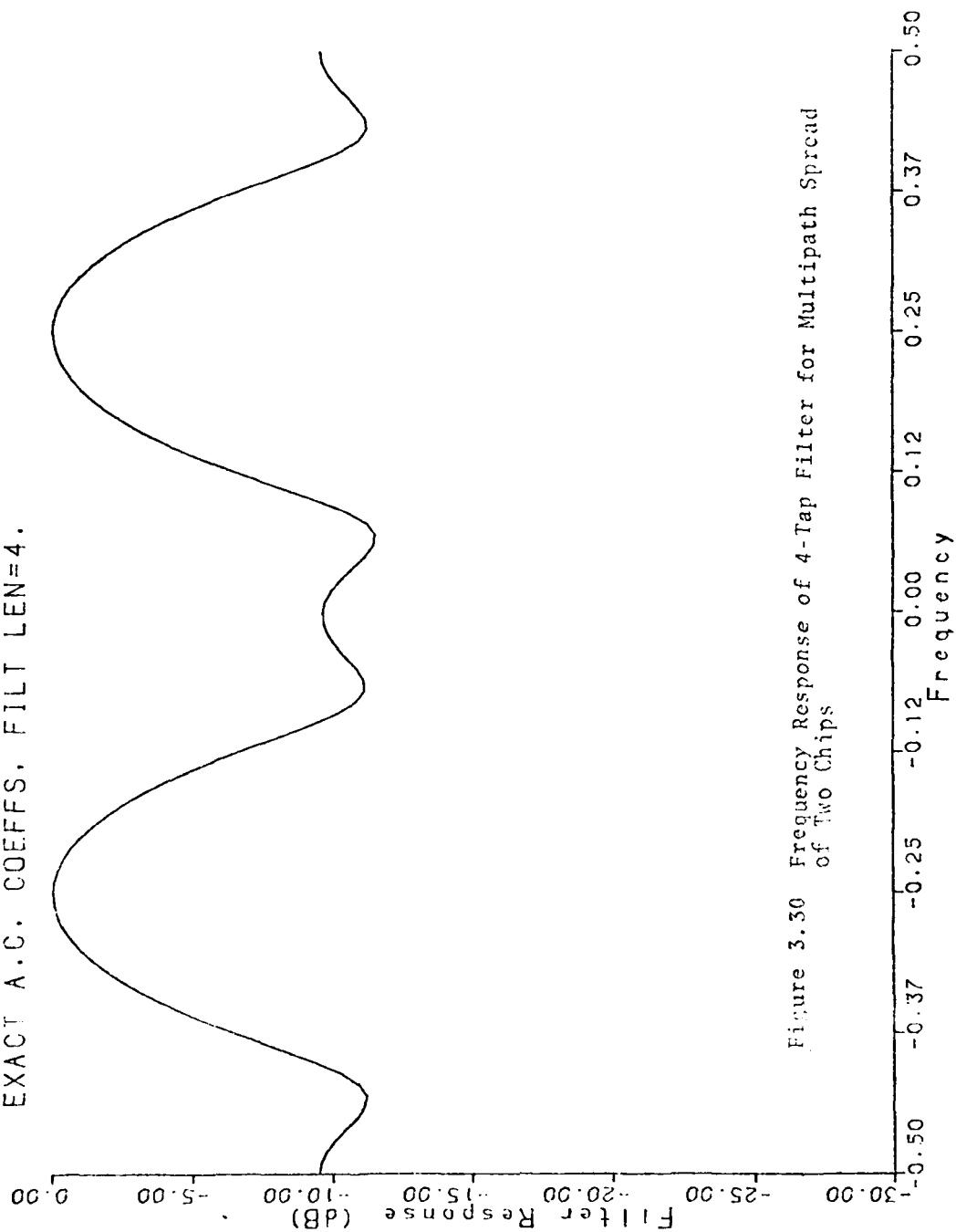


Figure 3.30 Frequency Response of 4-Tap Filter for Multipath Spread of Two Chips

TWO PATH CHANNEL, DELAY= 4.000

INTERFERENCE VARIANCE= 200.0000

EXACT A.C. COEFFS, FILT LEN=4.

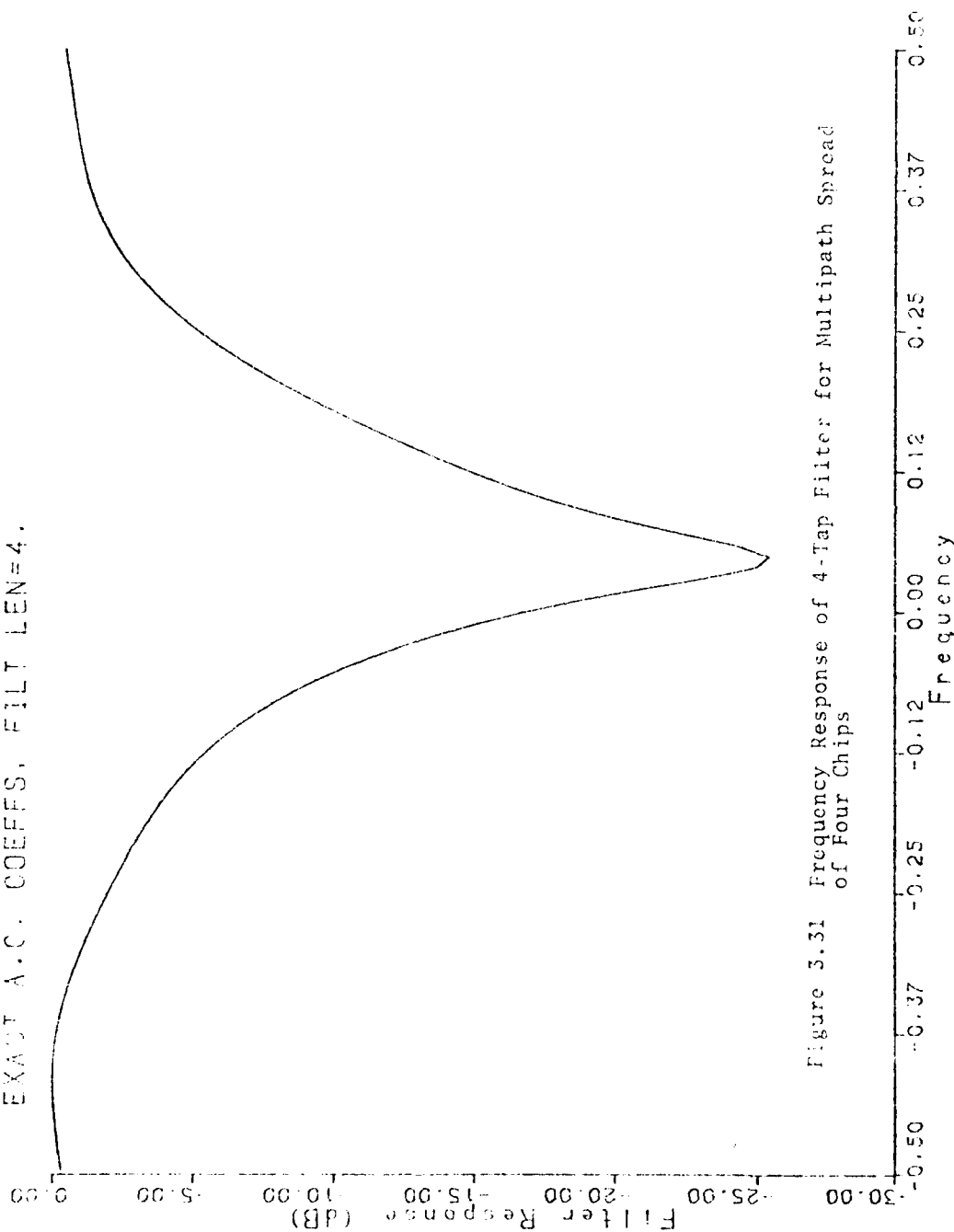


Figure 3.31 Frequency Response of 4-Tap Filter for Multipath Spread of Four Chips

TWO PATH CHANNEL, DELAY= 4.000

INTERFERENCE VARIANCE= 20.0000

EXACT A.C. COEFFS. FILT.LEN=4.

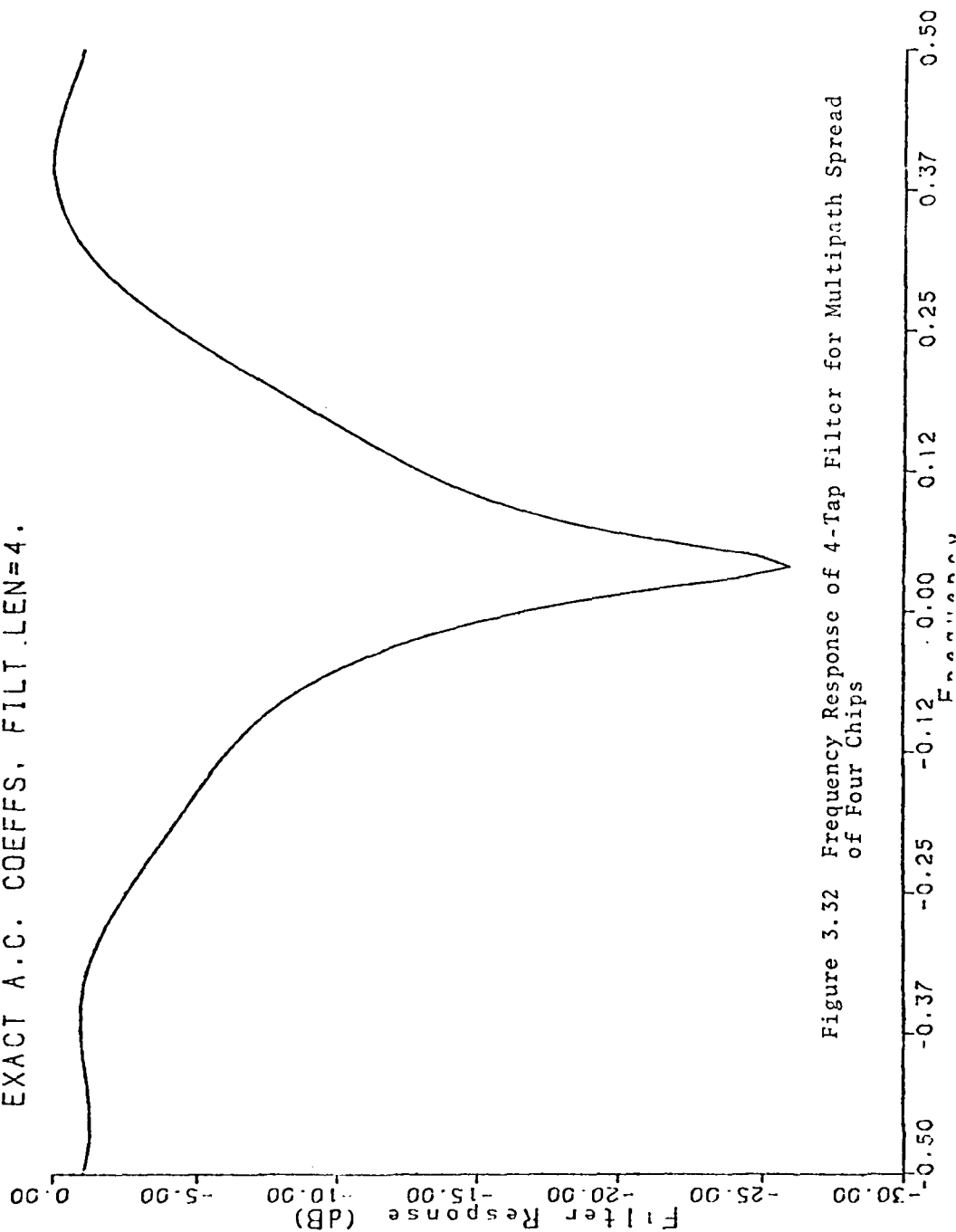


Figure 3.32 Frequency Response of 4-Tap Filter for Multipath Spread of Four Chips

TWO PATH CHANNEL, DELAY= 4.000
 INTERFERENCE VARIANCE= 2.0000
 EXACT A.C. COEFFS, FILT LEN=4.

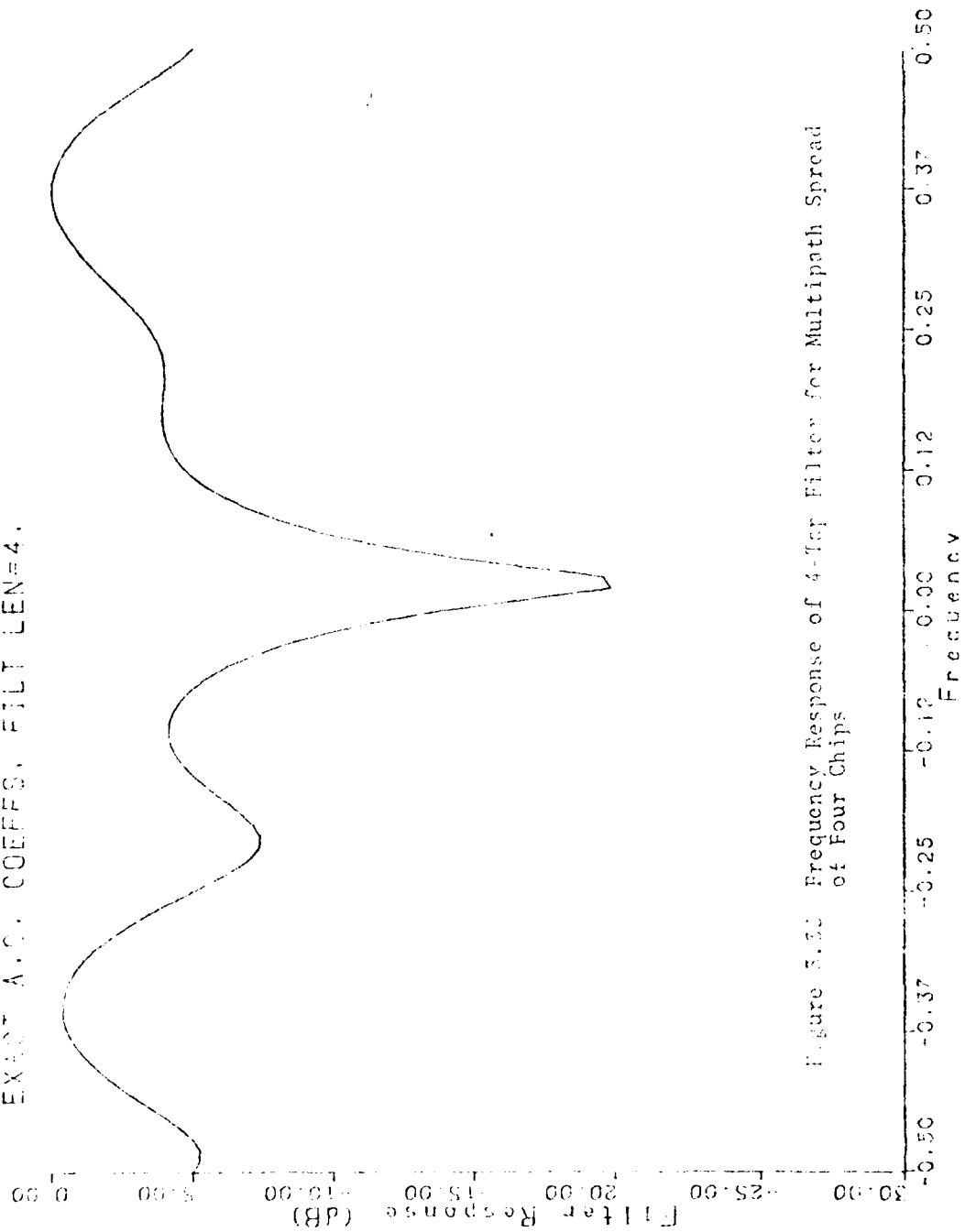


Figure 3.31 Frequency Response of 4-Tap Filter for Multipath Spread of Four Chips

AD-A097 336

CLARKSON COLL OF TECHNOLOGY POTSDAM N Y DEPT OF ELEC--ETC F/6 9/3
SUPPRESSION OF NARROWBAND INTERFERENCE IN PSEUDO-NOISE SPREAD S--ETC(U)
FEB 81 J G PROAKIS, J W KETCHUM F30602-78-C-0102

UNCLASSIFIED

RADC-TR-81-6

ML

2 of 2
AD-A
179 350



END
DATE
FILMED
5-81
DTIC

TWO PATH CHANNEL, DELAY= 4.000
 INTERFERENCE VARIANCE= 0.2000
 EXACT A.C. COEFFS, FILT LEN=4.

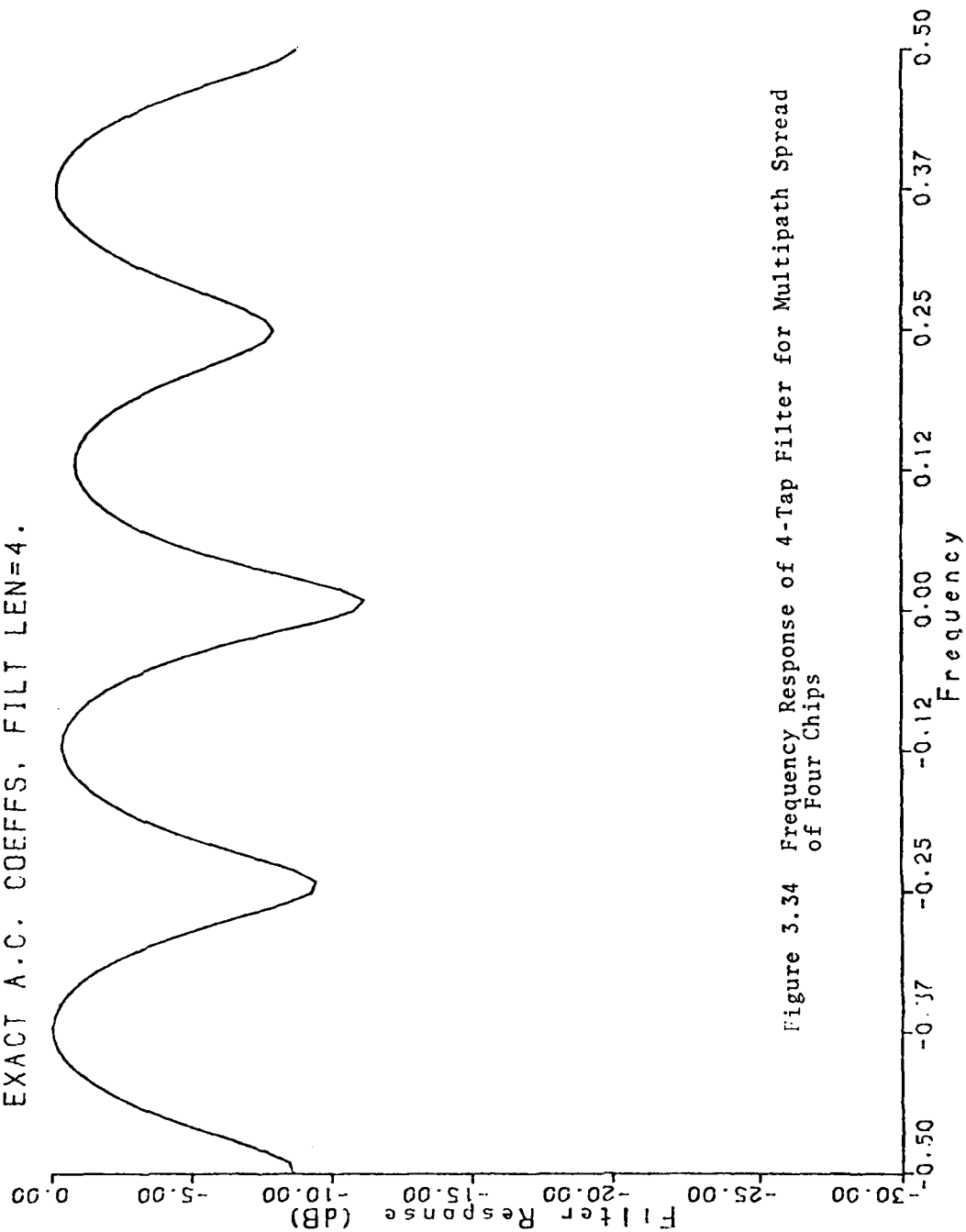


Figure 3.34 Frequency Response of 4-Tap Filter for Multipath Spread of Four Chips

TWO PATH CHANNEL, DELAY= 4.000

INTERFERENCE VARIANCE= 0.0200

EXACT A.C. COEFFS. FILT LEN=4.

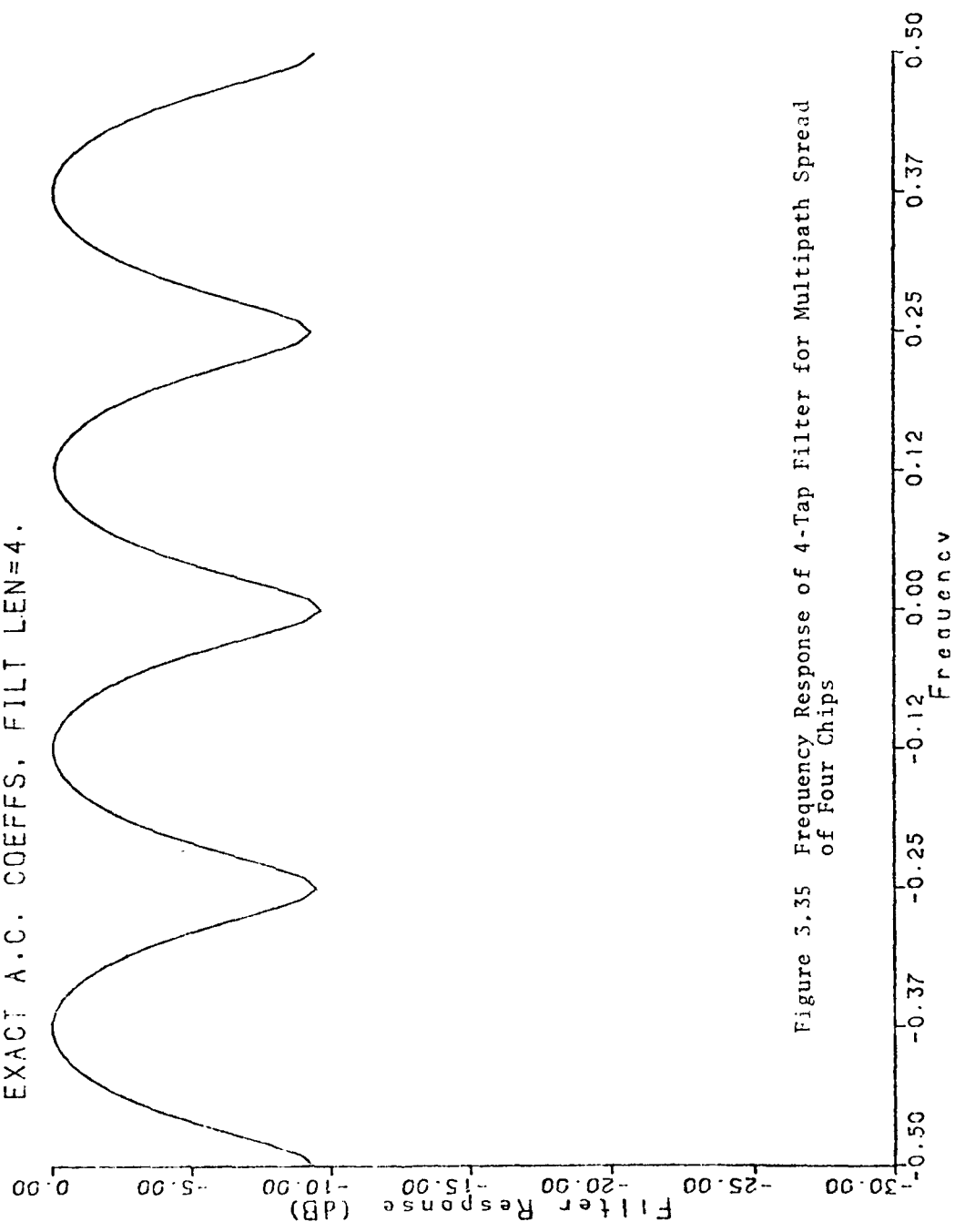


Figure 3.35 Frequency Response of 4-Tap Filter for Multipath Spread of Four Chips

TWO PATH CHANNEL, DELAY= 5.000

INTERFERENCE VARIANCE= 200.0000

EXACT A.C. COEFFS, FILT LEN=4.

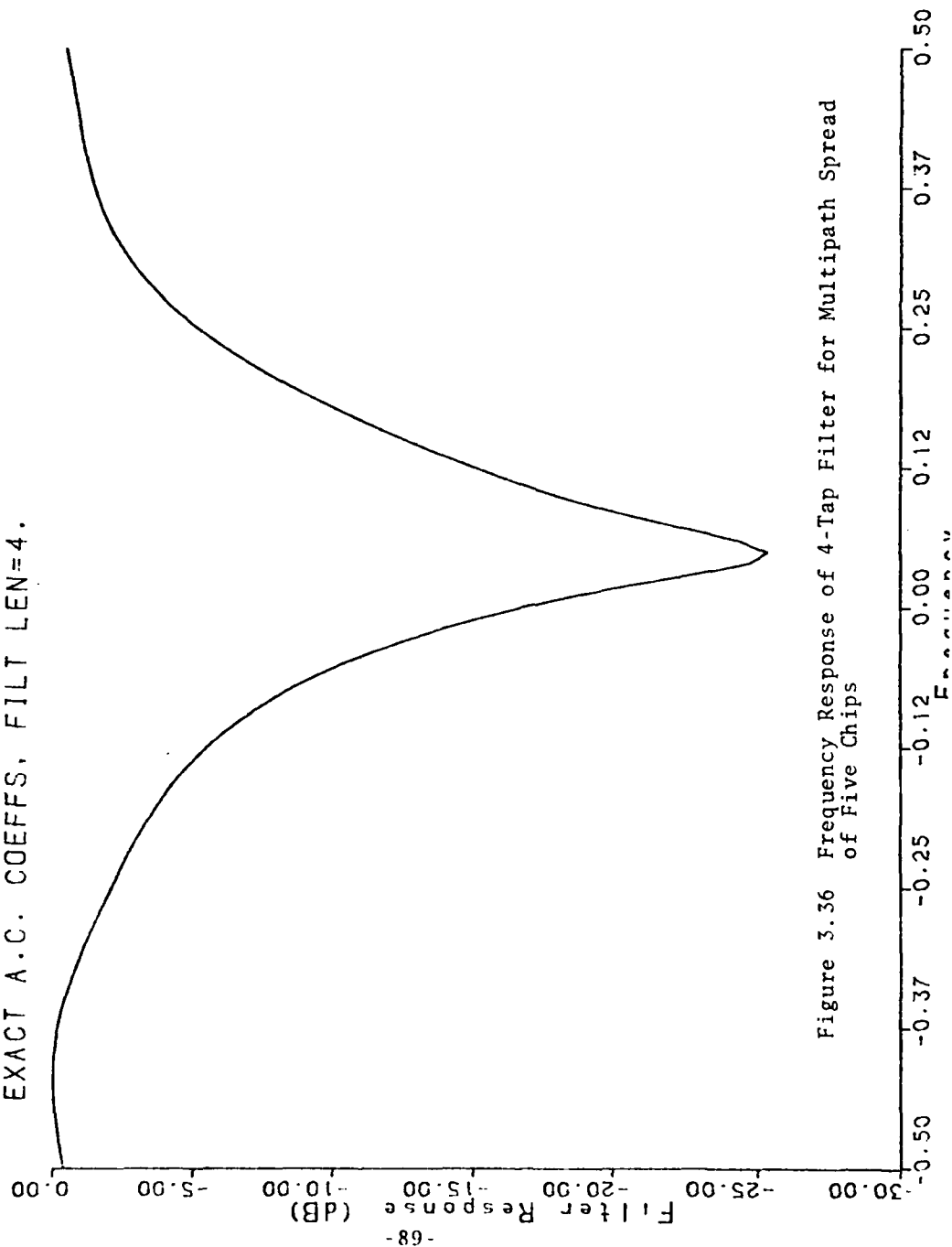


Figure 3.36 Frequency Response of 4-Tap Filter for Multipath Spread of Five Chips

TWO PATH CHANNEL, DELAY= 5.000

INTERFERENCE VARIANCE= 20.0000

EXACT A.C. COEFFS, FILT LEN=4.

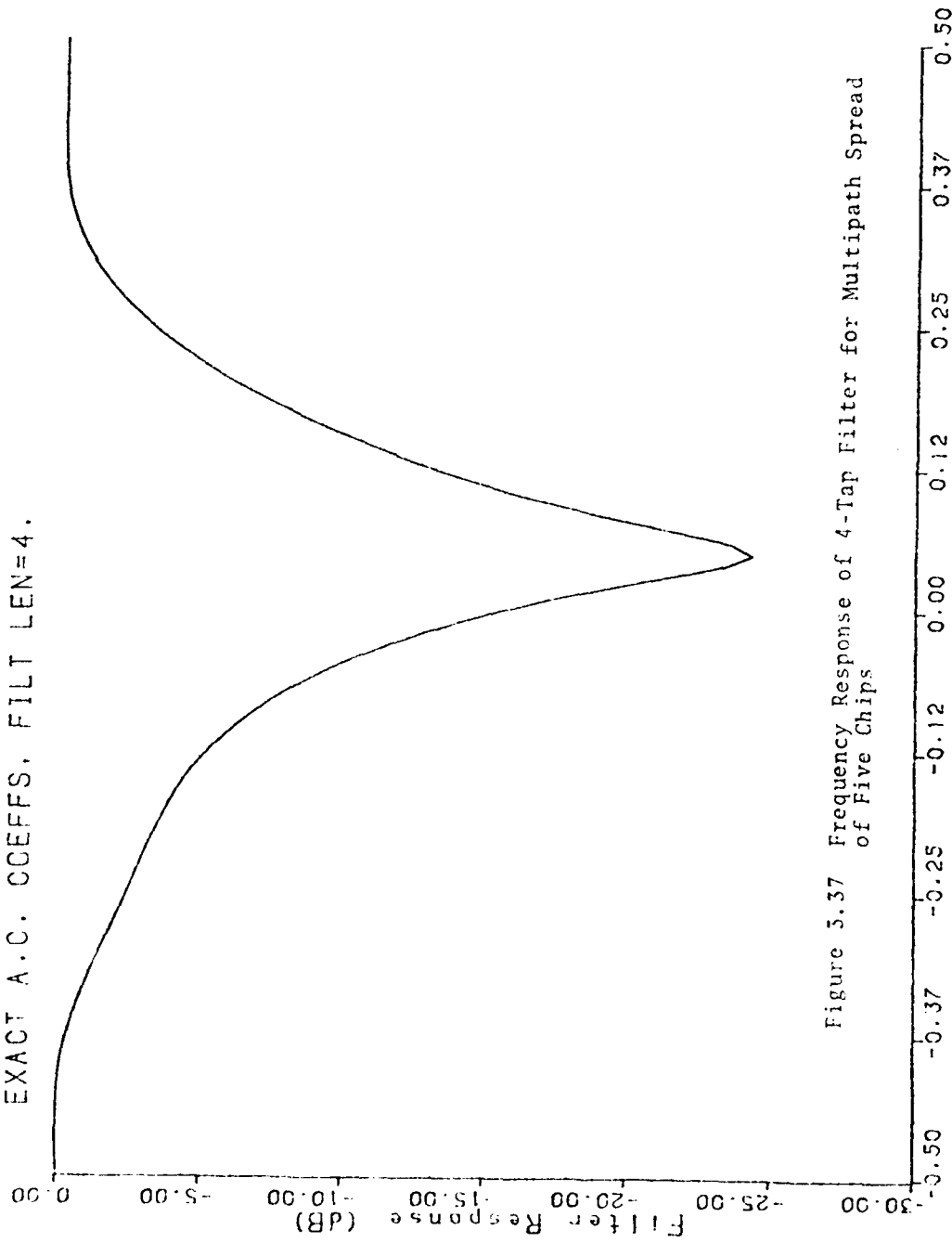


Figure 3.37 Frequency Response of 4-Tap Filter for Multipath Spread of Five Chips

TWO PATH CHANNEL, DELAY= 5.000
INTERFERENCE VARIANCE= 2.0000
EXACT A.C. COEFFS, FILT LEN=4.

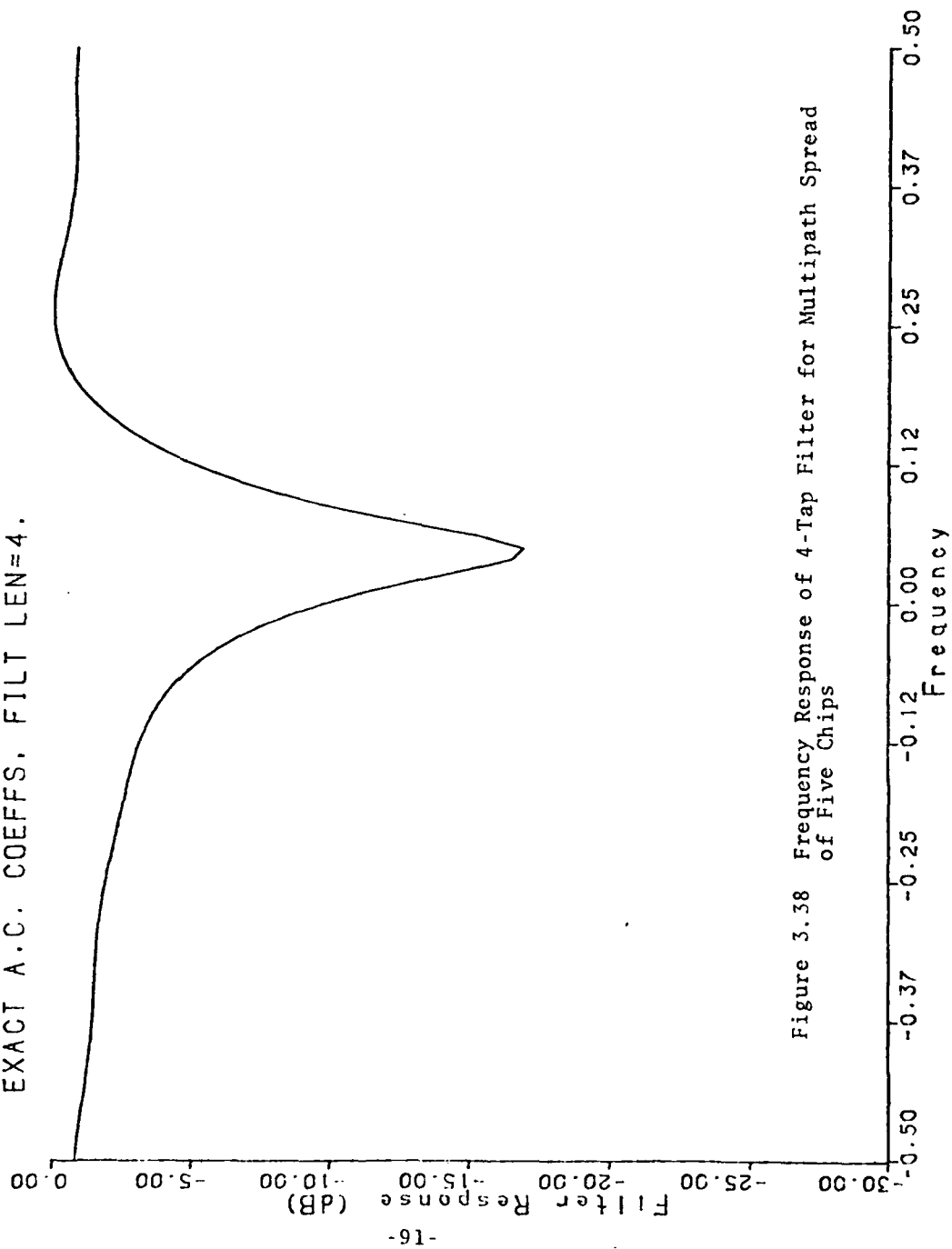


Figure 3.38 Frequency Response of 4-Tap Filter for Multipath Spread of Five Chips

TWO PATH CHANNEL, DELAY= 5.000

INTERFERENCE VARIANCE= 0.2000

EXACT A.C. COEFFS, FILT LEN=4

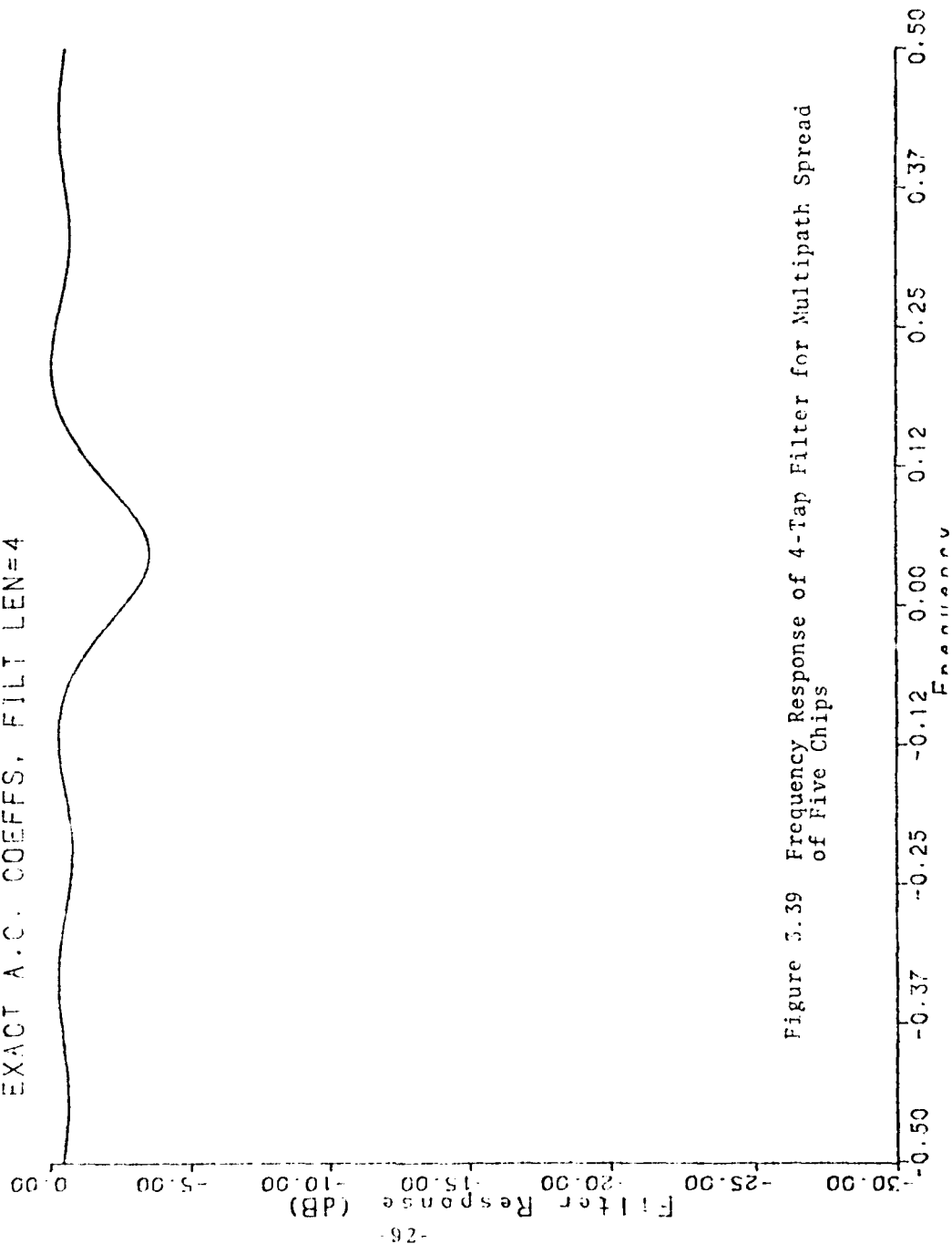


Figure 3.39 Frequency Response of 4-Tap Filter for Multipath Spread of Five Chips

TWO PATH CHANNEL, DELAY= 5.000

INTERFERENCE VARIANCE= 0.0200

EXACT A.C. COEFFS, FILT LEN=4.

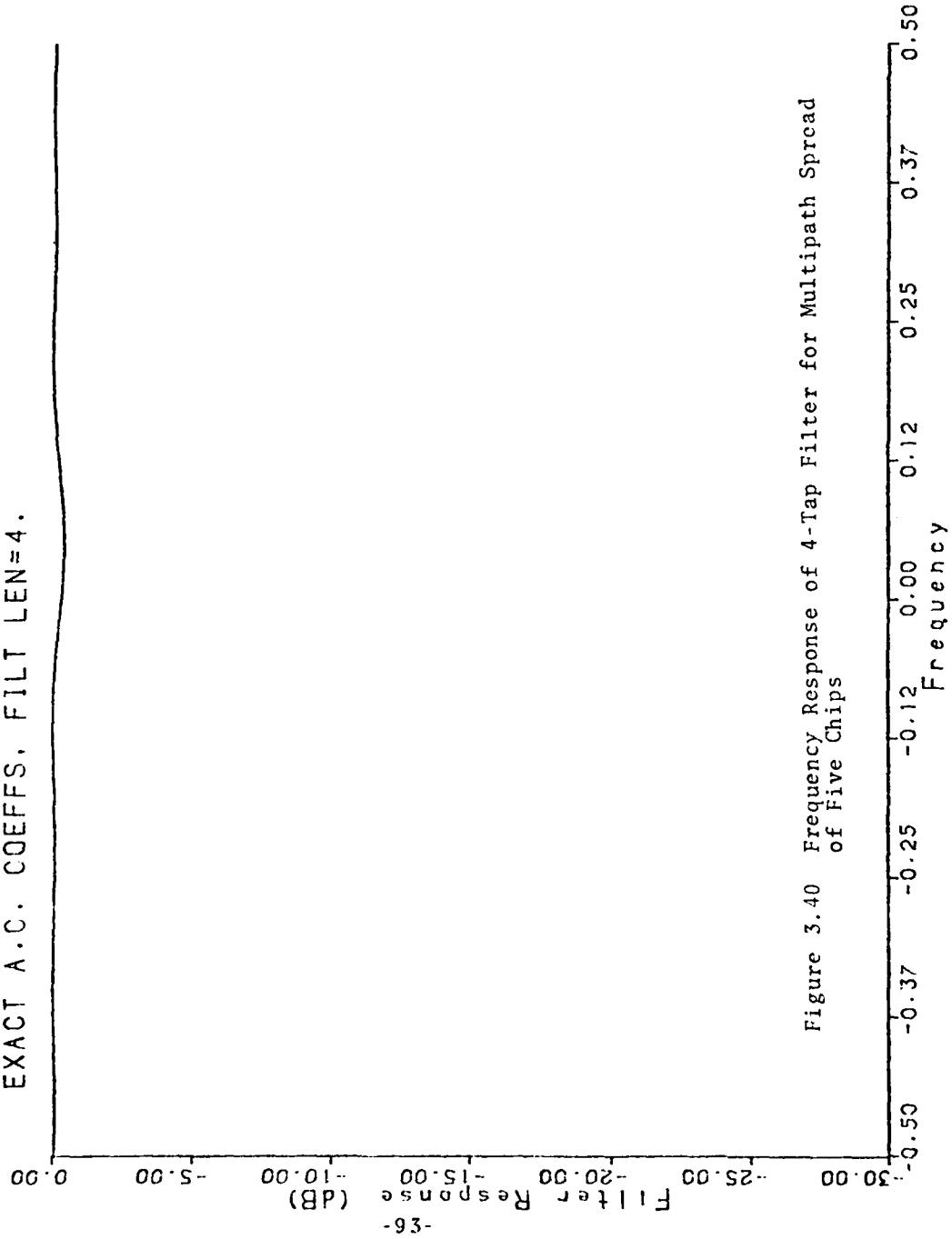
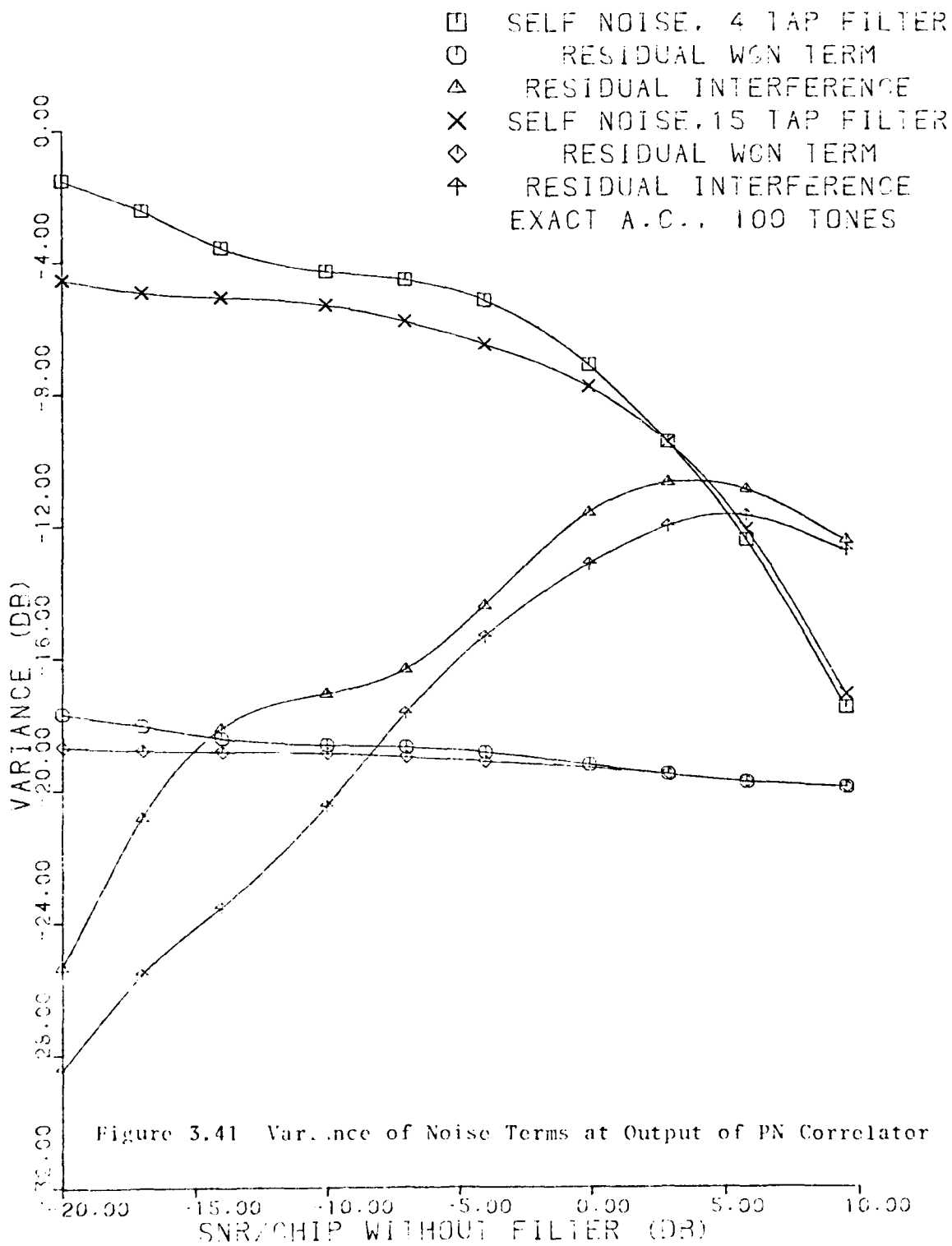


Figure 3.40 Frequency Response of 4-Tap Filter for Multipath Spread of Five Chips

corresponding to five chips. In this case, when the interference is small, the suppression filter approaches an all pass filter.

The results given above demonstrate that the one-step predictor can be used effectively in a multipath environment. On the other hand, the M-step predictor, when M is large, is not very effective except, perhaps, for a pure CW interference.

An important issue in our investigation of narrow band interference suppression in a PN spread spectrum system is the amount of intersymbol interference introduced by the suppression filter and whether or not there is a need for equalization. In order to demonstrate the magnitude of the distortion caused by the interference suppression filter, we have plotted in Figure 3.41 the variance of each of the three terms in the denominator of the expression for the SNR given by Eq. (3.7). In this computation, the channel is ideal (no multipath) and the interference is confined to a single band. The suppression filter consists of the filter $A_{\text{IF}}(z)$ in cascade with its matched filter. The term "self-noise" refers to the time dispersive distortion caused by the suppression filter. It is evident that this is the dominant term in the denominator of the expression for the SNR in Eq. (3.7). Hence, it is the term that limits the performance of the system when the narrow band interference is strong. For example, at an SNR per chip of -20dB, the distortion due to the filter is more than 15dB above the white noise and more than 20dB above the residual interference. When the interference is weak, the filter approaches an all-pass characteristic and, hence, the distortion due to time dispersion decreases. Therefore, the need for equalization of the time-dispersive effects of the suppression filter is established by these



numerical results. Of course, if there are resolvable multipath signal components in the received signal the need for equalization is already established. In such a case, the equalizer will not only compensate for the channel distortion but also for the signal distortion caused by the suppression filter.

In order to demonstrate the effectiveness of an equalizer in compensating for the distortion of the filter and the multipath components in a fading channel, we have obtained, via Monte Carlo simulation on a digital computer, the probability of error for a receiver consisting of an interference suppression filter followed by a decision-feedback equalizer (DFE) and a PN correlator. A channel consisting of two equal-strength (on the average) paths was simulated with a multipath spread corresponding to four chips. In order to maximize the efficiency of the simulation, we used a "snapshot" technique to simulate the effect of Rayleigh fading. That is, the complex valued, Gaussian-distributed tap weights of the channel were pseudo-randomly selected and kept fixed for 2000 PN chips. The equalizer was first trained for 1000 chips and then error rate data was collected on the other 1000 chips. Then, the channel was changed by pseudo-randomly selecting another set of tap weights and the procedure was repeated. Thus, we obtained a probability of error averaged over the fading. A DFE was employed instead of a linear equalizer primarily because of the superiority of the DFE in compensating for a channel and/or suppression filter that contains spectral nulls. In the simulation, the interference was confined to a single band. Hence a four-tap predictor was used. The DFE consisted of eight feed forward taps, and seven feed back taps.

The results of the Monte Carlo simulations are shown in Figures 3.42 through 3.46. The first two figures illustrate the error probability for a signal-to-interference ratio of -20dB and -10dB per chip prior to filtering, respectively. In this case, the input to the PN correlator consists of the chip estimates from the DFE prior to binary quantization. Figure 3.44 illustrates the effectiveness of the interference suppression filter. That is, the performance of the system with a signal-to-interference ratio of -20dB is almost as good as that for -10dB. A check of the slope of these error rate curves indicates that they correspond to the performance of a dual diversity system. In other words, the feed-forward part of the equalizer acts as an equivalent (coherent) diversity combiner. Figures 3.45 and 3.46 illustrate the performance of the receiver when the input to the PN correlator are the hard decisions from the DFE. Comparison of these graphs with those in Figures 3.42 and 3.43 leads us to conclude that chip estimates into the PN correlator result in better performance at low error rates.

The error rate results given above indicate that inter-chip interference caused by channel multipath and the suppression filter can be compensated by the DFE. It is important to emphasize, however, that, for the DFE to be effective, it must be sufficiently long to span the time-dispersion of the filter and the channel multipath. That is, an increase in the length of the interference suppression filter must be accompanied by an increase in the length of the DFE.

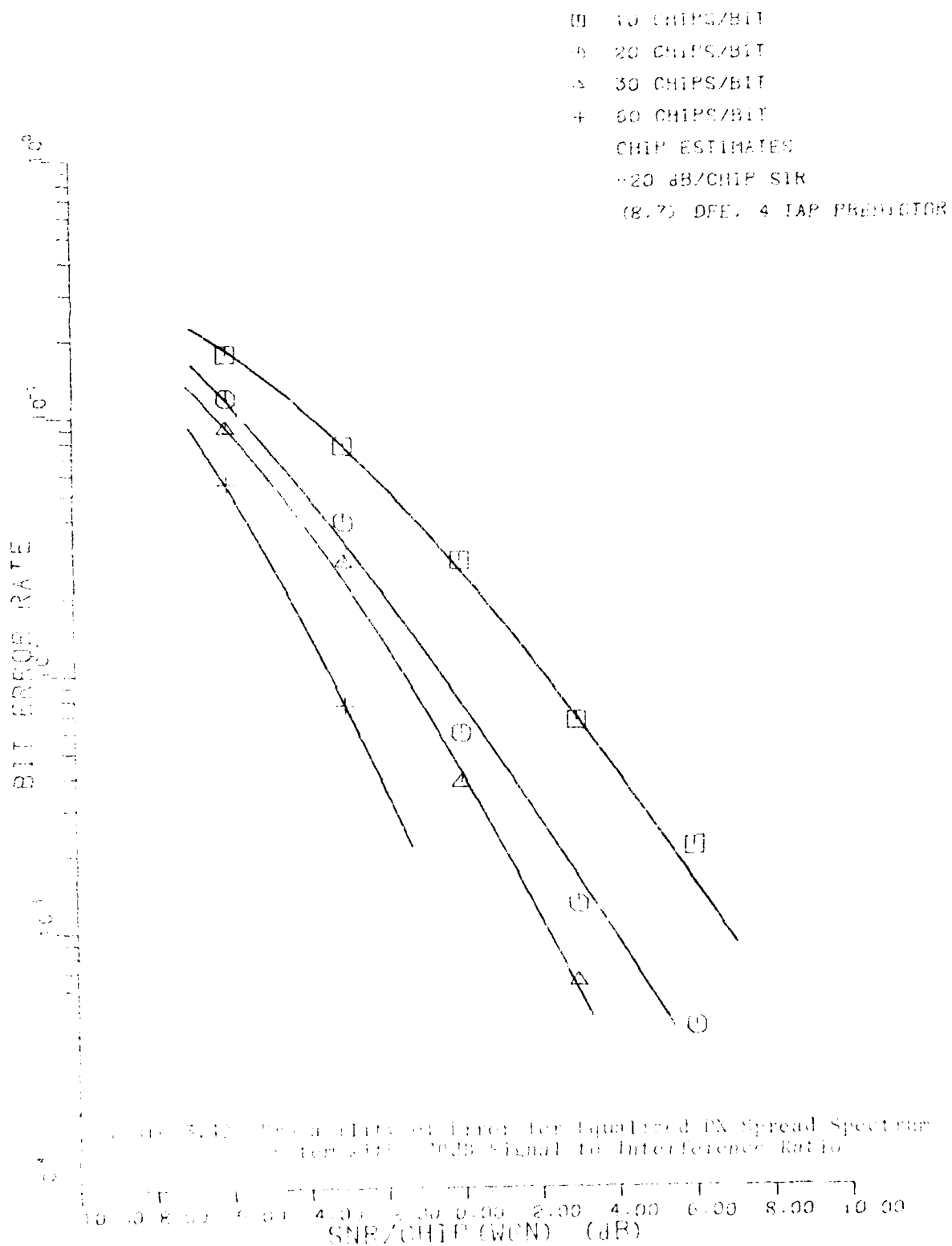
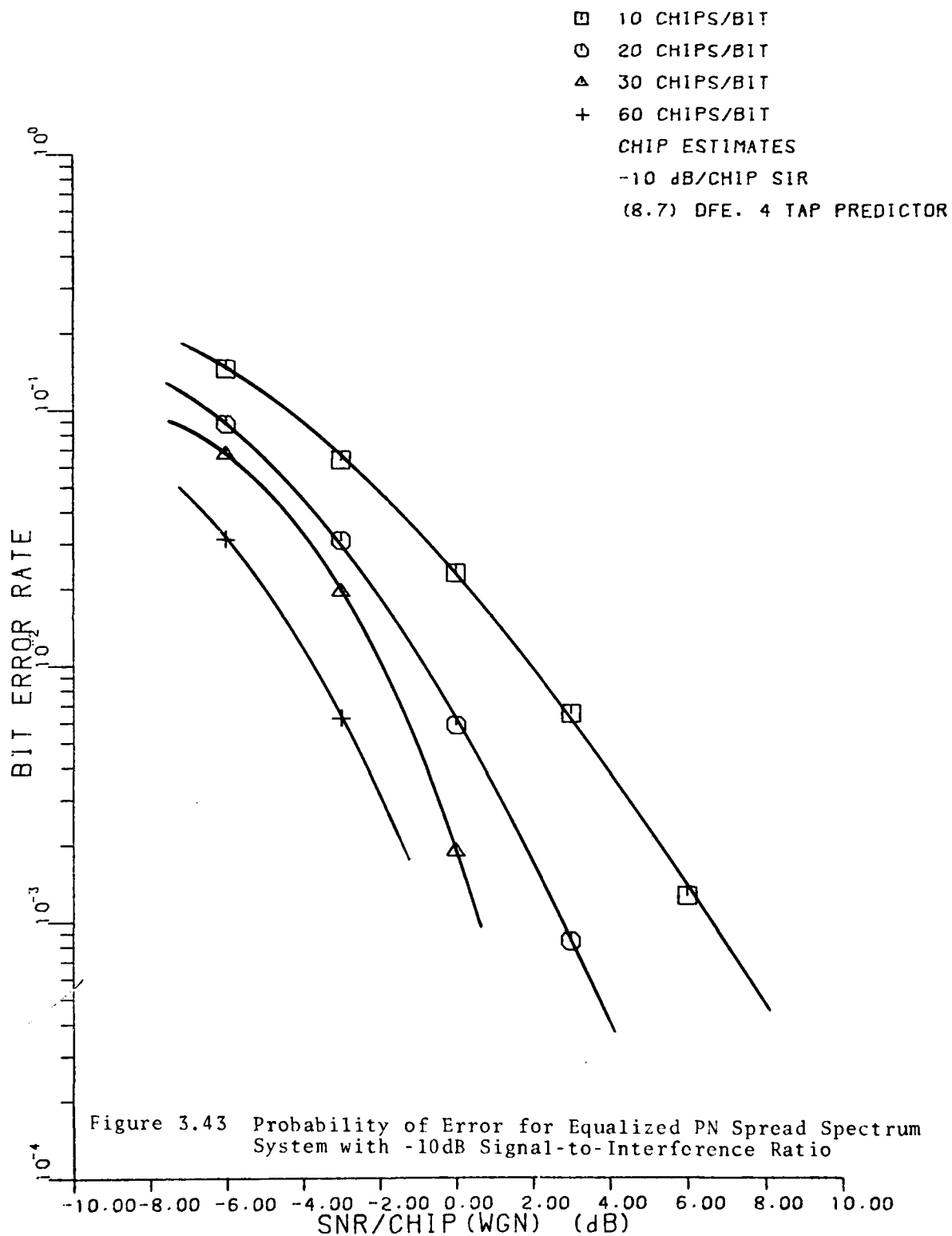


Figure 3.12: Probability of Error for Equalized PS Spread Spectrum System with -20dB Signal to Interference Ratio



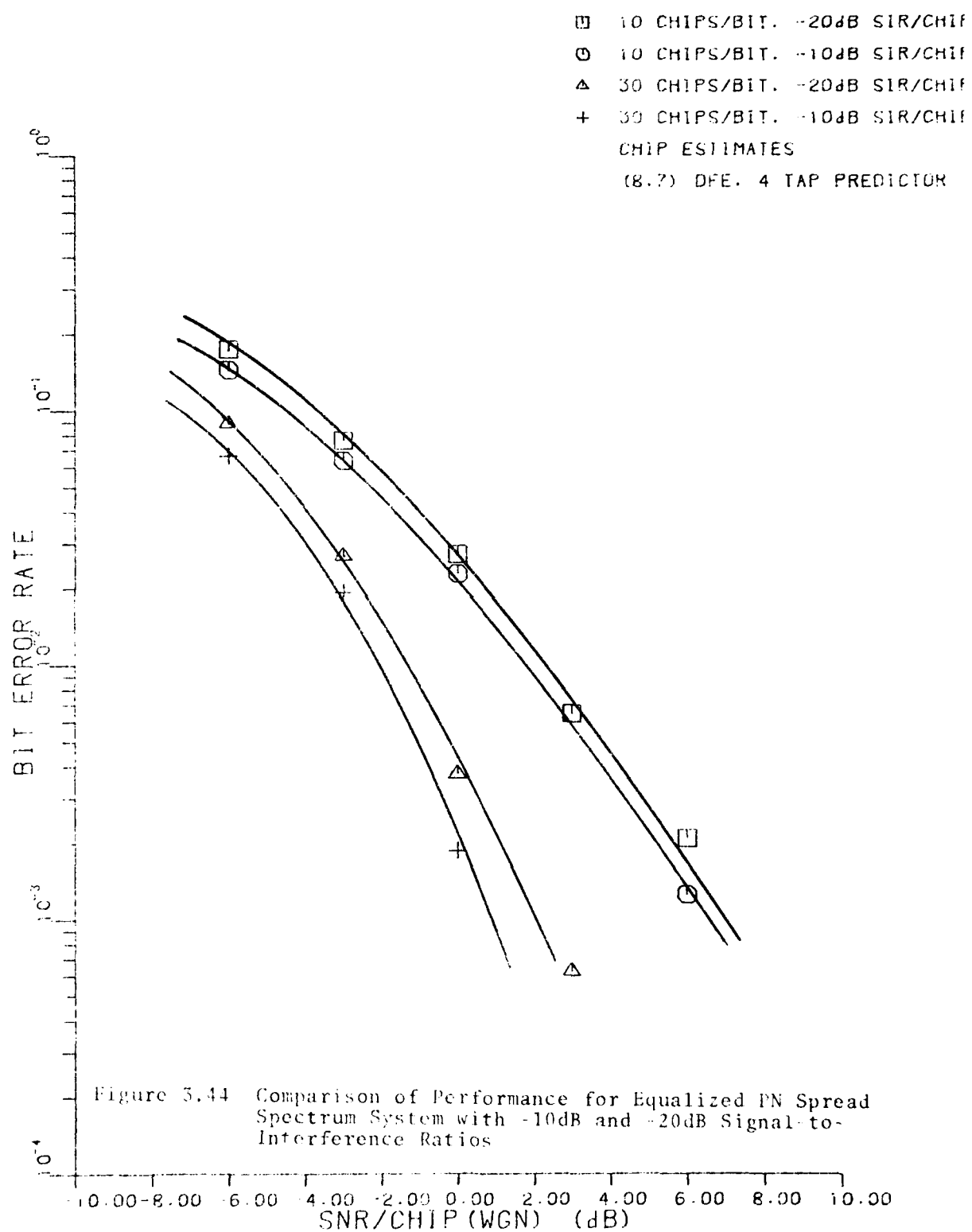
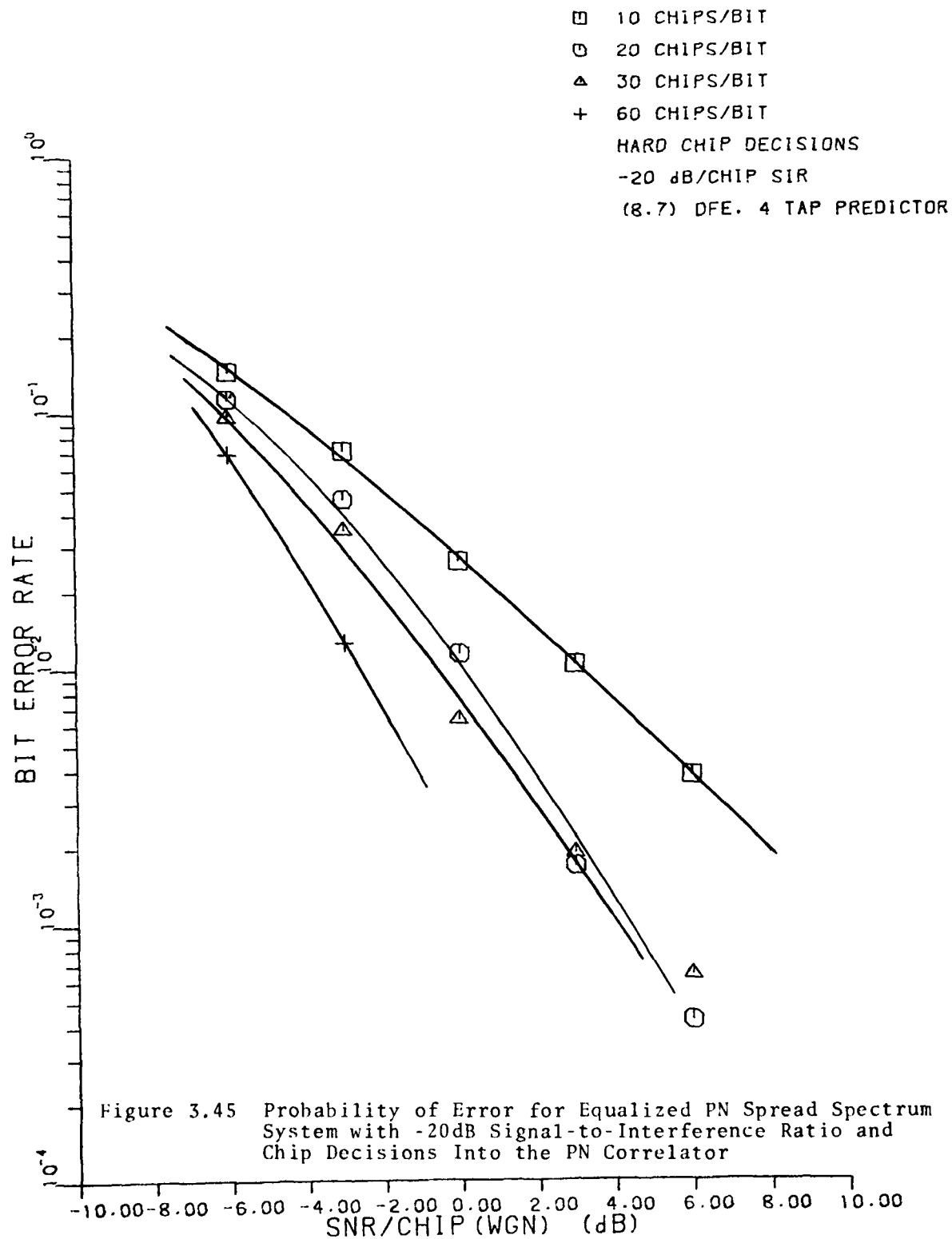
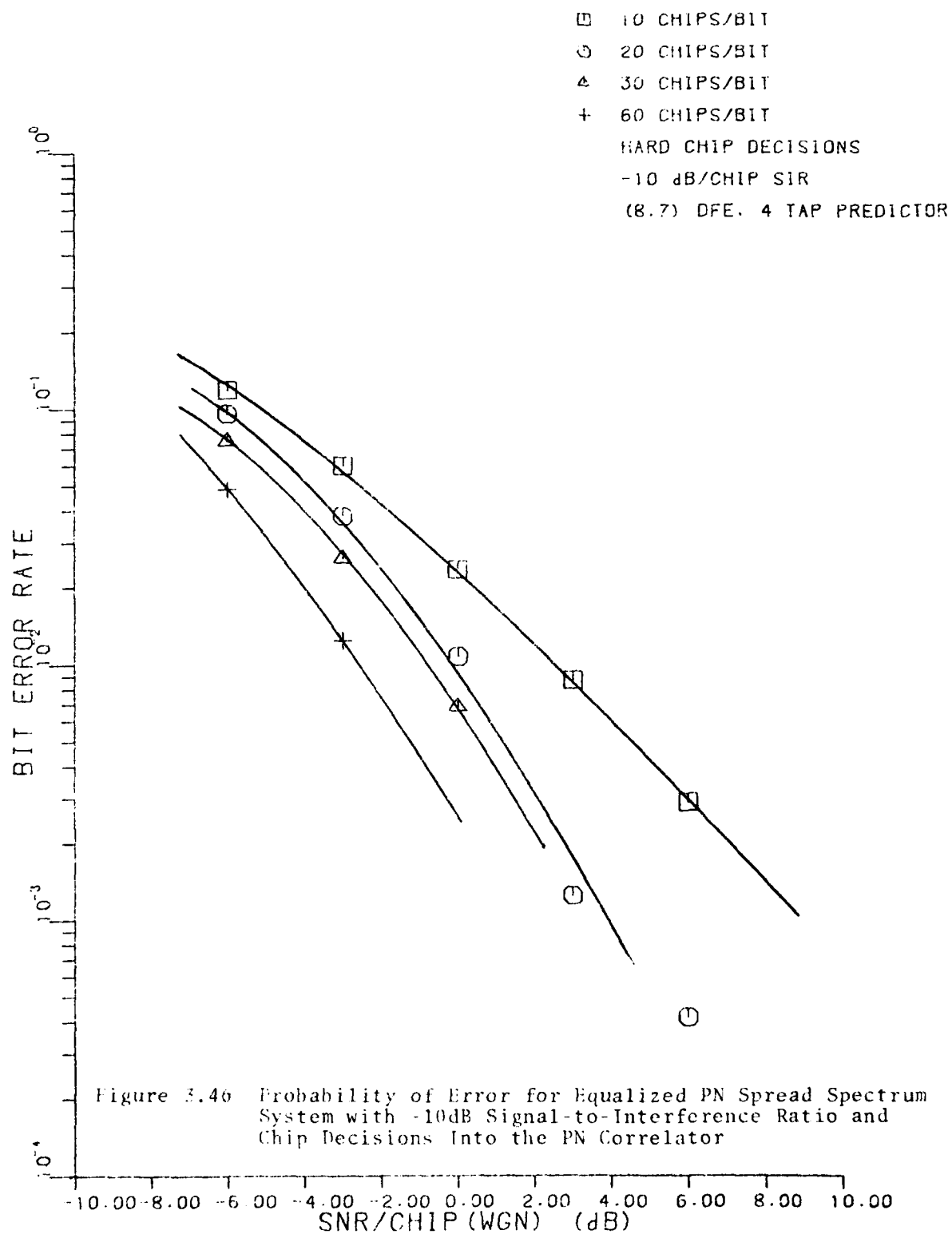


Figure 3.44 Comparison of Performance for Equalized PN Spread Spectrum System with -10dB and -20dB Signal-to-Interference Ratios





REFERENCES

- [1] Hsu, F. M. and Giordano, A. A., "Digital Whitening Techniques for Improving Spread Spectrum Communications Performance in the Presence of Narrowband Jamming and Interference," IEEE Trans. Comm., Vol. COM-26, No. 2, pp. 209-216, February 1978.
- [2] Ulrych, T. J. and Bishop, T. N., "Maximum Entropy Spectral Analysis and Autoregressive Decomposition," Rev. Geophysics and Space Phys., Vol. 13, pp. 183-200, February 1975.
- [3] Burg, J. P., "Maximum Entropy Spectral Analysis," Proc. 37th Meeting of the Society of Exploration Geophysicists, 1967; also reprinted in Modern Spectrum Analysis (D. G. Childers, Ed.), pp. 54-41, IEEE Press, 1978.
- [4] Levinson, H., "The Wiener RMS Error Criterion in Filter Design and Prediction," J. Math. Phys., Vol. 25, pp. 261-278, 1947.
- [5] Oppenheim, A. V. and Schaffer, R. W., Digital Signal Processing, Prentice-Hall, 1975.
- [6] Programs for Digital Signal Processing, IEEE Press, 1979.
- [7] Papoulis, A., Probability, Random Variables and Stochastic Processes, McGraw-Hill, 1965.
- [8] Morf, M., Vieira, A. and Lee, D. T., "Ladder Forms for Identification and Speech Processing," Proc. 1977 IEEE Conf. Decision and Control, New Orleans, LA, December 1977, pp. 1074-1078.
- [9] Morf, M. and Lee, D. T., "Recursive Least Squares Ladder Forms for Fast Parameter Tracking," Proc. 1978 IEEE Conf. Decision and Control, San Diego, CA, January 10-12, 1978, pp. 1362-1367.

- [10] Morf, M., Lee, D. F., Nicholls, J. R., and Vieira, A., "A Classification of Algorithms for ARMA Models and Ladder Realizations," Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Hartford, CT, May 12-17, pp. 13-17.
- [11] Morf, M., Dickinson, B., Kallath, T., and Vieira, A., "Efficient Solution of Covariance Equations for Linear Prediction," IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-25, No. 5, pp. 429-455, October 1977.
- [12] Pack, J. D. and Satorius, E. H., "Least Squares, Adaptive Lattice Algorithms," Technical Report 123, Naval Ocean Systems Center, San Diego, CA, April 1979.
- [13] Marple, L., "A New Autoregressive Spectrum Analysis Algorithm," IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-28, No. 4, pp. 141-154, August 1980.
- [14] Makhoul, J., "A Class of All-Zero Lattice Digital Filters: Properties and Applications," IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-26, No. 4, pp. 304-314, August 1978.
- [15] Nuttall, A. H., "Multivariate Linear Predictive Spectral Analysis Employing Weighted Forward and Backward Averaging: A Generalization of Burg's Algorithm," NUSC Technical Report 5501, Naval Underwater Systems Center, Newport, RI, October 15, 1976.
- [16] Proakis, J. G., "Advances in Equalization for Intersymbol Interference," in Advances in Communication Systems, Vol. 4 (A. J. Viterbi, Ed.), Academic Press, 1975.

*MISSION
of
Rome Air Development Center*

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C³I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

DATE
FILMED
—8