To the Graduate Council:

I am submitting herewith a dissertation written by Charles Lynn Carnal entitled "Power Spectrum Estimation with Logarithmically Uniform Resolution in Frequency." I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Robert W. Rochelle

Robert W. Rochelle, Major Professor

We have read this dissertation
and recommend its acceptance:

Edward J. Harris

Roy D. Joseph

J. M. Lee

Accepted for the Council:

Carl Minkel

The Graduate School

POWER SPECTRUM ESTIMATION WITH LOGARITHMICALLY UNIFORM RESOLUTION IN FREQUENCY

A Dissertation

Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Charles Lynn Carnal

August 1984

DEDICATION

This dissertation is dedicated to the author's grandmother, Nova Bell Brasfield.

ACKNOWLEDGMENTS

The author thanks Dr. Robert W. Rochelle, Major Professor, for originally suggesting the topic of this dissertation research and for many fruitful discussions. Recognition and appreciation are due to Dr. Joseph M. Googe, Professor and former Electrical Engineering Department Head, for providing an instructorship which has been a most challenging and rewarding experience and has provided the financial means whereby the author could pursue the doctorate. The author is also greatly indebted to Dr. Roy D. Joseph, Committee Member, who provided initial encouragement to the author to pursue a doctorate. Thanks are also expressed to Dr. Edward G. Harris, Committee Member, for providing a stimulating experience in graduate courses which the author attended. Finally, the author wishes to express deep appreciation to his wife, Toshiko, and children, Charles and Lana, whose love and devotion have helped the author more than any words can express.

ABSTRACT

Conventional harmonic analysis involves the representation of the average power of a random process in frequency bands which are equally spaced and equal in frequency resolution or bandwidth. In practical applications such as auditory acoustics and data reduction, a canonical representation of the power spectrum sometimes consists of estimates of the average power in frequency bands which are equally spaced and of equal bandwidth on a logarithmic frequency scale.

This dissertation develops two general techniques for logarithmic spectrum estimation. The first approach uses a bank of lowpass filters whose output average powers are differenced to obtain the required estimates. The second applies a family of frequency dependent windows to the periodogram of the data to obtain the spectrum estimates.

The computational requirements and statistical properties of the proposed approaches are derived and compared to conventional methods. Other logarithmically uniform frequency resolution approaches which have appeared in the literature are also presented and compared with the two new techniques introduced. To illustrate the general principles involved the new estimators are implemented and applied to a set of sinusoidal test data and a set of stochastic test data.

TABLE OF CONTENTS

SECTION		PAGE
1.	INTRODUCTION	1
	1.1 Overview	1
	1.2 Applications	3
	1.3 Mathematical Preliminaries	5
2.	FILTER BANK APPROACH	15
	2.1 General Theory	15
	2.2 Lowpass Filter Bank	25
	2.3 Other Filter Methods	41
3.	DISCRETE FOURIER TRANSFORM APPROACH	53
	3.1 Frequency Domain Windowing	53
	3.2 Logarithmic Windowing	60
	3.3 Other Fourier Transform Methods	67
4.	IMPLEMENTATION	72
	4.1 Logarithmic Parameters	72
	4.2 Experimental Data Generation	75
	4.3 Lowpass Filter Bank Implementation	81
	4.4 Frequency Domain Window Implementation	87
	4.5 Implementation Results	88
5.	CONCLUSIONS	93
LIST OF REFERENCES		96
APPENDIX		102
VITA		130

LIST OF TABLES

TABLE		PAGE
2.1	List of Normalized Variance for r = 2, s = 4, and M = 20 .	35
3.1	Approximate Upper Bound Computational Load Comparison . . .	66
4.1	LURF Bands for r = 2, s = 3, M = 11 and ω_1 = 50/1024 . . .	74
4.2	Transition Band Sample Magnitudes	84
4.3	LURF PSD Results for Sinusoidal Data	89
4.4	LURF PSD Results for Stochastic Data	90

LIST OF FIGURES

FIGURE		PAGE
2.1	Ideal Bandpass Filter Magnitude Function	16
2.2	Average Power Density Bandpass Filter Processing	17
2.3	Magnitude Functions for Ideal Bandpass Filter Bank	21
2.4	Bandpass Filter Bank Processing Scheme for Power Spectrum Estimation	22
2.5	Frequency Scale Demarcation Comparison	23
2.6	Magnitude Functions for Ideal Lowpass Filter Bank	26
2.7	Lowpass Filter Bank Processing Scheme for Power Spectrum Estimation	28
2.8	Plot of Log Normalized Variance	34
2.9	Computational Advantage of Lowpass over Bandpass	40
2.10	Nested Variance Power Spectrum Window Function	43
2.11	Enochson's Recursive Filter Bank Processing Scheme	48
2.12	Enochson's Recursive Filter Bank Magnitude Functions . . .	49
2.13	Harris' Recursive Octave Preprocessor	51
3.1	Quadratic Windowing Power Spectrum Estimation	57
3.2	Linear Windowing Power Spectrum Estimation	58
3.3	Daniell Frequency Domain Window	62
3.4	Frequency Domain Windowing LURF PSD Estimation	65
4.1	Ideal LURF PSD for Sinusoidal Test Data	76
4.2	Pole-Zero Locations of Stochastic Test Data	78
4.3	Ideal PSD for Stochastic Test Data	80
4.4	Window Designed Typical Equivalent Bandpass Magnitude Function	82
4.5	Frequency Sample Designed Typical Equivalent Bandpass Magnitude Function	85

1. INTRODUCTION

1.1 Overview

Frequency domain signal and system analysis techniques have become important tools for scientists, engineers and mathematicians in modern times. The foundation of frequency domain methods was laid in 1807 by Jean Baptiste Joseph de Fourier (Robinson, 1982). Fourier discovered that almost any arbitrary, discontinuous function of time can be represented over a finite interval of time by an infinite summation of cosine and sine functions whose frequencies are integral multiples of a fundamental frequency. The squared coefficients of the exponential terms in the Fourier series express a signal's average power as a function of frequency because of the orthogonal property of the exponential basis functions of the series (Peebles, 1976, 20-22). Thus the Fourier series provides an estimate of a signal's power spectrum at frequencies which are harmonically related to a fundamental frequency.

Most of the currently known techniques for power spectrum estimation exploit the harmonic relationship between analysis frequencies and obtain estimates representing the average power in frequency bands which are equally spaced and equal in frequency resolution or bandwidth. In some practical applications, a canonical representation of the power spectrum consists of estimates of the average power in frequency bands which are equally spaced and of equal bandwidth on a logarithmic scale.

This dissertation addresses the theory and implementation of power spectrum estimation schemes which provide point estimates with logarithmically uniform resolution in frequency (LURF). Two fundamentally different approaches are proposed as new methods to obtain LURF power spectrum estimates. The first approach is based on a set of lowpass filters. The second approach utilizes the discrete Fourier transform (DFT).

The first proposed approach for obtaining the LURF power spectrum conceptually consists of passing the time series data simultaneously through a set of bandpass filters having center frequencies and bandwidths which are set according to the LURF criteria. The average power density is computed by averaging the instantaneous output power of each filter and dividing each average power point estimate by its associated bandwidth. This provides LURF point estimates of the power spectrum density (PSD). The unique and original feature of the proposed scheme is that the bandpass filter bank effect is produced using only lowpass filters. Specific properties associated with the proposed LURF lowpass scheme are derived. LURF filter bank schemes which have appeared in the literature are presented and compared with the proposed scheme.

The second proposed approach for obtaining the LURF PSD is presented after the filter bank approach. It consists of computing the discrete Fourier series of the raw time series data, computing the square magnitude of each of the Fourier coefficients, and applying a frequency domain "window." The windowing operation is

actually a convolution which smears the unwindowed, conventional harmonic PSD estimate to produce the LURF PSD. A computational advantage can be realized by judiciously choosing the type of window employed. Other DFT-based approaches which have been previously discovered are presented and compared with the new proposed approach.

Both of the approaches are implemented and tested using two sets of experimental data having very different characteristics. The results of the specific implementations are compared to the theoretical LURF PSD for each set of experimental data.

1.2 Applications

The first major application of LURF is in physiological acoustics. This is concerned with hearing and sound phenomena in the auditory frequency range. The theory of physiological acoustics provides the scientific basis of musical theory. The aesthetic relationship of musical tones depends upon the fact that the established musical notes are equally separated on a logarithmic frequency scale (Helmholtz, 1877), (Jeans, 1937). A digital spectrum analyzer has been proposed for the express purpose of analyzing music with logarithmically uniform frequency resolution (Teaney, 1980). Studies have shown that the human ear processes sound with logarithmically uniform frequency spacing and resolution (Schafer, 1950), (Moller, 1968), (Pfeiffer, 1975). Research on auditory spectrum analysis models has focused on the LURF aspect of the problem (Gambardella, 1970),

(Altes, 1978), (Kates, 1983). In the area of speech signal processing, a logarithmically uniform spectrum analysis and synthesis can be used to exploit the human ear's decreasing frequency selectivity with increasing frequency (Rabiner, 1978). Because of the frequency response properties of the human ear, some research in industrial acoustical noise analysis has focused on estimation of the power spectrum with logarithmically uniform frequency resolution (Negron, 1966), (Maling, 1967), (Otnes, 1968), (Enochson, 1968).

The second major application area is that of data reduction. Standard harmonic analysis transforms N time domain data points into N frequency domain points. Even though the frequency domain representation may convey information of physical significance more directly than the time domain representation, in many cases it is unnecessary to use a total of N frequency domain points to do so. Thus a canonical representation of the information consists, in some instances, of estimates of average signal power in frequency bands which are equally spaced and of equal bandwidth on a logarithmic frequency scale. The practical need for data reduction is very severe in space telemetry applications. To optimize the yield of scientific information from a space misson, it is necessary to reduce a large number of measurements to a few quantities of important physical meaning. This must be done on-board the spacecraft to make the most efficient use of the telemetry resources in sending data to Earth. Data reduction requirements for telemetry and severe computational constraints have been the

driving force behind research which has sought to find LURF algorithms with very low computational requirements. Methods which have been proposed have been almost totally concerned with reducing the computational burden for LURF algorithms at the expense of of flexibility (Southeastern Scientific Co., 1978), (Spitzer, 1981), (Durand, 1983).

Two miscellaneous areas of LURF power spectrum estimation application are space physics data analysis (Owens, 1977), (Coleman, 1966) and biomedical membrane noise analysis (DeFelice, 1976).

This investigation does not address a particular application of LURF spectrum estimation but focuses on the general statistical and computational estimation problem and proposes two new techniques. The proposed schemes are presented in general terms to allow application to a broad class of practical problems with as much flexibility as possible.

1.3 Mathematical Preliminaries

The fundamental concepts of the theory and practice of spectrum analysis are found in both engineering and statistics literature. In order to provide consistent definitions and concepts, the following section contains the development of the basic mathematical relationships which are relevant to this study.

The signal processing investigation of this dissertation is concerned with the processing of signals which arise from a random process. A signal-producing random phenomenon can be interpreted

as an experiment for which there exists a set called the sample space consisting of all possible outcomes of the experiment. Each possible outcome is a time function $x_i(t)$, where i denotes the particular outcome. The set of all possible outcome time functions $x_i(t)$, $i = 1, 2, 3, \ldots$ comprise what is called a random or stochastic process, denoted $X(t)$.

For any fixed t, $X(t)$ is a random variable and is characterized by a probability distribution function

$$F_X(x;t) = P\{X(t) \leq x\} \tag{1.3.1}$$

where $P\{\cdot\}$ is the probability of the event $\{\cdot\}$ and x is a real number. The probability density function is defined by

$$f_X(x;t) = \frac{\partial F_X(x;t)}{\partial x} . \tag{1.3.2}$$

The expected value of the process $X(t)$ is defined as

$$E\{X(t)\} = \int_{-\infty}^{+\infty} x f_X(x;t)dx = \overline{X}(t) \tag{1.3.3}$$

and is called the mean of the process. The autocorrelation of a process is defined as

$$R(t_1,t_2) = E\{X(t_1)X(t_2)\}$$

$$= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x_1 x_2 f_X(x_1,x_2;t_1,t_2)dx_1dx_2 \tag{1.3.4}$$

where $f_X(x_1,x_2;t_1,t_2)$ is the second order density function of $X(t)$. This density function is defined through

$$f_X(x_1,x_2;t_1,t_2) = \frac{\partial^2 F_X(x_1,x_2;t_1,t_2)}{\partial x_1 \partial x_2}. \qquad (1.3.5)$$

where

$$F_X(x_1,x_2;t_1,t_2) = P\{(X(t_1) \leq x_1) \cap (X(t_2) \leq x_2)\}. \qquad (1.3.6)$$

The autocovariance of a process is defined as

$$\begin{aligned} C(t_1,t_2) &= E\{(X(t_1)-\overline{X}(t_1))(X(t_2)-\overline{X}(t_2))\} \\ &= R(t_1,t_2) - \overline{X}(t_1)\overline{X}(t_2). \end{aligned} \qquad (1.3.7)$$

If all the statistical properties of the process are unchanged by a shift in the time origin, then the process is said to be strictly stationary. This means that for any given τ, the two processes $X(t)$ and $X(t+\tau)$ have the same statistical properties. If $f_X(x;t) = f_X(x;t+\tau)$ for any τ, then it is also true that $f_X(x;t) = f_X(x)$; the first order density is independent of t. Consequently, it must be true that the mean of the process is a constant independent of t, viz. $\overline{X}(t) = \overline{X}$. If the second order density is also unchanged by a shift of the time origin $f_X(x_1,x_2;t_1,t_2) = f_X(x_1,x_2;t_1+t_0,t_2+t_0)$, then $f_X(x_1,x_2;t_1,t_2) = f_X(x_1,x_2;\tau)$ where $\tau = t_1-t_2$. In this case it can be concluded that the autocorrelation $R(t_1,t_2)$ depends only on $\tau = t_1-t_2$, and

$R(\tau) = E\{X(t+\tau)X(t)\} = R(-\tau)$ which indicates that the autocorrelation of a second order stationary process is an even function of τ.
A process is said to be wide-sense stationary if

$$E\{X(t)\} = \overline{X} = \text{a constant} \tag{1.3.8}$$

and

$$E\{X(t+\tau)X(t)\} = R(\tau). \tag{1.3.9}$$

In the sequel, the term stationary will be used to indicate wide-sense stationarity.

A Gaussian process is characterized by the first order density function

$$f_X(x;t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left\{\frac{(x-\overline{X})^2}{2\sigma^2}\right\}} \tag{1.3.10}$$

and normal density functions of all higher orders. Consequently, a Gaussian process thus defined is strictly stationary. The variance σ^2 is given by

$$\sigma^2 = C(\tau=0) = R(\tau=0) - \overline{X}^2. \tag{1.3.11}$$

The density functions of higher order than the first order density function only involves information embodied by

$$C(\tau) = R(\tau) - \overline{X}^2 \tag{1.3.12}$$

and thus the Gaussian process is completely described statistically

by the mean $\overline{X}$ and autocorrelation $R(\tau)$. The Gaussian process is analytically convenient, and as a consequence of the central limit theorem, many physical signal phenomena are adequately described as Gaussian (Helstrom, 1984).

To estimate the statistical mean $\overline{X}$ and the statistical autocorrelation $R(\tau)$ requires many sample functions from the process unless the process is ergodic. A process is said to be ergodic if statistical averages of the entire process are equal to time averages involving a typical sample function from the process. The time average mean $\overline{x}$ of a particular sample function $x(t)$ from the process $X(t)$ is given by

$$\overline{x} = \lim_{T\to\infty} \frac{1}{2T}\int_{-T}^{+T} x(t)dt. \qquad (1.3.13)$$

The time autocorrelation function $r(\tau)$ of $x(t)$ is defined as

$$r(\tau) = \lim_{T\to\infty} \frac{1}{2T}\int_{-T}^{+T} x(t+\tau)x(t)dt. \qquad (1.3.14)$$

If the statistical mean $\overline{X}$ and autocorrelation $R(\tau)$ of the process $X(t)$ are equal to the time average mean $\overline{x}$ and time autocorrelation $r(\tau)$ of a typical sample function $x(t)$ from the process, the process is ergodic. Note that in the case of the Gaussian process, knowledge of a single sample function is sufficient to completely describe the statistical properties of the process. The ergodic property is very important, because in most instances of practical concern, only a single sample function from the process can be observed.

The autocorrelation function has a frequency domain representation which exhibits the same information in a format which can provide greater insight to the underlying physical phenomena. This frequency domain representation is derived below.

Take a typical sample function $x(t)$ from the process $X(t)$ and truncate it according to

$$x_T(t) = \begin{cases} x(t) & -T \leq t \leq T \\ 0 & \text{elsewhere.} \end{cases} \tag{1.3.15}$$

The Fourier transform of $x_T(t)$ is defined by

$$X_T(\omega) = \int_{-\infty}^{+\infty} x_T(t) e^{-j\omega t} dt. \tag{1.3.16}$$

Define the instantaneous power of $x(t)$ to be $x^2(t)$. Parseval's formula is then an expression of the conservation of energy between the time domain representation $x_T(t)$ and the frequency domain representation $X_T(\omega)$, viz.

$$\int_{-\infty}^{+\infty} x_T^2(t) dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |X_T(\omega)|^2 d\omega. \tag{1.3.17}$$

The average power is obtained by dividing both sides by $2T$

$$\frac{1}{2T} \int_{-\infty}^{+\infty} x_T^2(t) dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \frac{|X_T(\omega)|^2}{2T} d\omega. \tag{1.3.18}$$

Taking the expected value and then taking the limit as $T \to \infty$ yields the average power P if the process is ergodic

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-\infty}^{+\infty} E\{x_T^2(t)\}dt$$

$$= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \lim_{T\to\infty} E\{\frac{|X_T(\omega)|^2}{2T}\}d\omega. \qquad (1.3.19)$$

The integrand in the frequency domain integration can be interpreted as a density function of the average power in the process, the power spectrum density $S(\omega)$, i.e.

$$S(\omega) = \lim_{T\to\infty} E\{\frac{|X_T(\omega)|^2}{2T}\}. \qquad (1.3.20)$$

This definition of the PSD is illuminating from two points of view. First it relates a frequency domain representation of the power in a single sample function to the probabilistic frequency characteristics of the entire process. Secondly it suggests how a practical estimate of the PSD can be obtained from a single sample function. It can be shown from equation (1.3.20) that

$$S(\omega) = \int_{-\infty}^{+\infty} R(\tau)e^{-\omega\tau}d\tau \qquad (1.3.21)$$

which is called the Wiener-Khinchin theorem (Wiener, 1949). Hence the PSD $S(\omega)$ and autocorrelation $R(\tau)$ are Fourier transform pairs.

Even before the proliferation of computers, many signal producing phenomena which have been studied involved discrete-time signals. The preponderance of digital computers in recent years has driven the theory and practice of signal analysis, including spectrum estimation, toward the concept of sampling a continuous-time

signal $x(t)$ at regular intervals Δt to obtain a discrete-time signal $x[n]$ as follows

$$x[n] = x(n\Delta t). \tag{1.3.22}$$

The relevant relations in spectrum analysis are as follows. The Fourier transform $X(e^{j\omega})$ of a discrete time signal $x[n]$, $n = 0, 1, 2, \ldots$ becomes

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}. \tag{1.3.23}$$

The inverse Fourier transform is given by

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} X(e^{j\omega})e^{j\omega n}\, d\omega. \tag{1.3.24}$$

The Fourier transform can be interpreted also as the value of the two-sided z-transform on the unit circle in the z-plane (Rabiner, 1975, 30)

$$X(e^{j\omega}) = X(z)\Big|_{z=e^{j\omega}} = \sum_{n=-\infty}^{\infty} x[n]z^{-n}\Big|_{z=e^{j\omega}}. \tag{1.3.25}$$

The discrete Fourier transform is defined by "sampling" the Fourier transform at equal intervals of frequency

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N} \tag{1.3.26}$$

for $k = 0, 1, 2, \ldots, N-1$. The inverse discrete Fourier transform (IDFT) is defined consistently with equation (1.3.26) to be

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{+j2\pi nk/N} \qquad (1.3.27)$$

for n = 0, 1, 2, ... , N-1. Such a representation of a discrete sequence which is defined only over $0 \leq n \leq N-1$ is periodic with period N for both x[n] and X[k] outside the interval. This fact must not be overlooked when considering the PSD for discrete-time processes.

The statistical autocorrelation sequence of a discrete-time process Y[n] is defined by

$$R[m] = E\{Y[n+m]Y[n]\} \qquad (1.3.28)$$

where the lag m takes on values from $-\infty$ to $+\infty$. Consequently, the Fourier transform of the statistical autocorrelation yields the PSD of the discrete process according to the Wiener-Khinchin theorem

$$S(e^{j\omega}) = \sum_{m=-\infty}^{\infty} R[m] e^{-j\omega m}. \qquad (1.3.29)$$

The autocorrelation is given by

$$R[m] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} S(e^{j\omega}) e^{j\omega m} d\omega. \qquad (1.3.30)$$

The conditions of practical application are that a single, discrete sample sequence x[n], n = 0, 1, 2, ... , N-1 is known, and the PSD must be estimated from this set of time series data.

There are two basic approaches to formulate an estimate $\hat{S}(e^{j\omega})$ of the true PSD $S(e^{j\omega})$. The most recent (Cadzow, 1982) and popular approach consists of assuming a parametric pole-zero model form of the

PSD called an autoregressive, moving average (ARMA) model

$$S(e^{j\omega}) = |K|^2 \frac{\prod_{k=1}^{q} (1-z_k e^{-j\omega})^*(1-z_k e^{-j\omega})}{\prod_{k=1}^{p} (1-p_k e^{-j\omega})^*(1-p_k e^{-j\omega})} \qquad (1.3.31)$$

and estimating the parameters K, q, p, p_k, z_k for the indicated values of k. The oldest approach consists of estimating the value of the PSD at discrete values of ω, i.e. $S(e^{j\omega_k})$ where $\omega_k = 2\pi k/N$ for k = 0, 1, 2, ... , N-1. Thus the oldest approach is conceptually a nonparametric approach. It can be shown (Cadzow, 1982) however that the oldest approach is a special case of the ARMA parametric model where only zeros are assumed to comprise the model.

In this study only the nonparametric approach is used. The nonparametric formulation has traditionally been called harmonic analysis because of the relative values of the analysis frequencies. This designation has passed down from the historical development of the Fourier series (Lee, 1960).

2. FILTER BANK APPROACH

2.1 General Theory

The general theory of the filter bank approach is presented first for bandpass filter banks. The following development serves to lay a foundation for the new proposed lowpass filter bank which is presented in the following subsection. The bandpass filter bank development also provides a precise definition of a LURF power spectrum.

If a signal $x(t)$ is fed into an ideal, unity-gain bandpass filter with transfer function $H_{BPI}(\omega)$, the signal components which have frequencies within the filter's passband emerge from the output of the filter. The passband of the ideal filter $H_{BPI}(\omega)$ is simply the band of frequencies in between the bandedge frequencies ω_0 and ω_1 as shown in Figure 2.1.

Assuming that the output signal $x_{BP}(t)$ from a bandpass filter is a one-ohm resistor, the instantaneous output power is simply the square of the output signal. In the sequel, the definition of signal power is thus construed. The actual signal units are of little consequence to the problem being investigated.

By integrating the instantaneous output power over some interval of time $0 \leq t \leq T$, the total energy passing through the filter over $0 \leq t \leq T$ within the frequency band $\omega_0 \leq \omega \leq \omega_1$ is obtained. Dividing the total energy over $0 \leq t \leq T$ by the factor T gives the average power. Dividing the average power by the filter bandwidth $\Delta\omega_1 = \omega_1 - \omega_0$, gives the average power density. These operations are depicted in Figure 2.2.

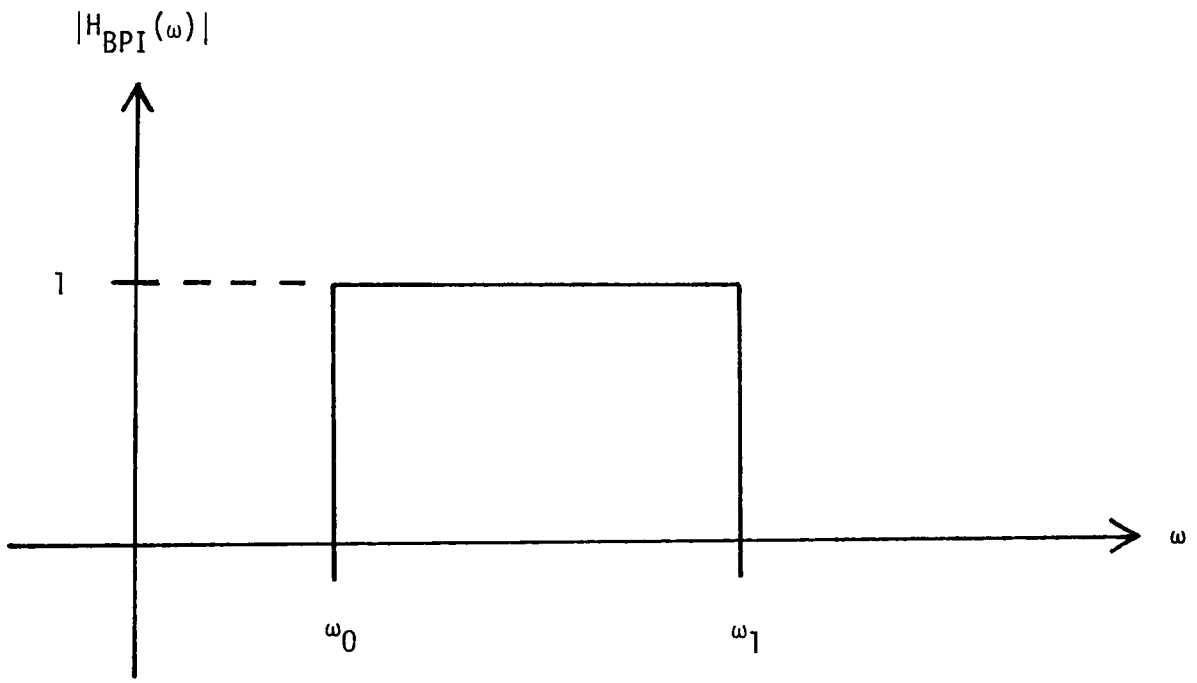


Figure 2.1 Ideal Bandpass Filter Magnitude Function

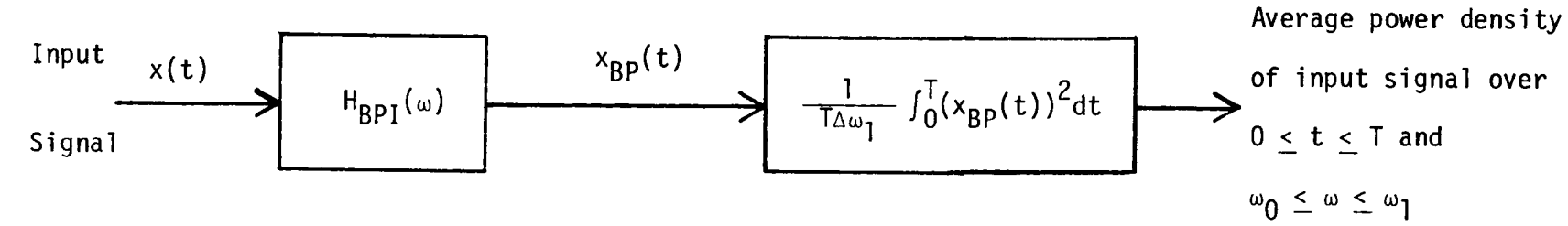


Figure 2.2 Average Power Density Bandpass Filter Processing

For these operations to represent an unbiased estimate of the average power density of the input signal over the frequency band $\omega_0 \leq \omega \leq \omega_1$ and time span $0 \leq t \leq T$, it is necessary that the filter be in steady state operation at $t = 0$. The convolution implied by

$$X_{BP}(\omega) = H_{BP}(\omega)X(\omega) \tag{2.1.1}$$

is

$$x_{BP}(t) = \int_{-\infty}^{t} x(\tau)h_{BP}(t-\tau)d\tau. \tag{2.1.2}$$

where $h_{BP}(t)$ is the impulse response of a causal bandpass filter. In practice the filter will not have been running since $t = -\infty$, and the actual filtering operation is represented by

$$x_{BP}(t) = \int_{-T_s}^{t} x(\tau)h_{BP}(t-\tau)d\tau \tag{2.1.3}$$

where T_s is the settling time of the filter. The settling time is inversely proportional to the filter bandwidth and the sharpness of the filter bandedge cutoff (Papoulis, 1977). This filter "warm-up" time T_s will prevent the transient response of the filter from biasing the estimate of the input signal's average power density over $0 \leq t \leq T$, assuming that the input signal's average power density does not change over the interval $-T_s \leq t \leq T$. The settling time T_s is the amount of time required for the filter's impulse response to decay approximately to zero. Since the absence of input signal prior to $t = -T_s$ is transparent to the power averaging operation, the filter can be started

at $t = -T_s$ if the averaging operation is started at $t = 0$. If the filter is started later than $t = -T_s$ and the averaging operation is started at $t = 0$, the power estimate is biased downward in magnitude because of the stationary assumption. In this case, a portion of the input signal is implicitly zero when it should not be.

Assuming steady state operation and an ergodic, stationary random process, another feature of the bandpass filter average power density estimation scheme is the time and frequency trade-off for the estimation error. The confidence which can be placed in average power estimates obtained is an inverse relation to the variance of the estimation error. It has been shown (Rice, 1943) that the variance of the normalized estimation error of the average power is approximately $(TB)^{-1}$ where B is the filter bandwidth in Hertz, and T is the time in seconds which both the filter and the power averaging operation has been running. The larger the bandwidth, the less integration time and filter operation time required to obtain a given level of confidence in the average power estimate. Since the filter bandwidth is roughly inversely proportional to the settling time of the filter T_s, the variance of the normalized estimation error is proportional to (T_s/T). By using a suitable definition of bandwidth, the variance is approximately given by (T_s/T) (Helstrom, 1984) indicating that the filter and power averaging operation should run an amount of time T in excess of the filter's settling time T_s to obtain a level of variance which is smaller in magnitude than the magnitude of the power being estimated. This does not include the warm-up time of the filter when the averaging operation is not being done.

To cover the entire band of frequencies which can be present in a signal to be analyzed, a bank of bandpass filters whose bandedges coincide can be employed. The magnitude functions for M such filters are individually shown in Figure 2.3a and compositely shown in Figure 2.3b. The bandwidth of the i^{th} bandpass filter is

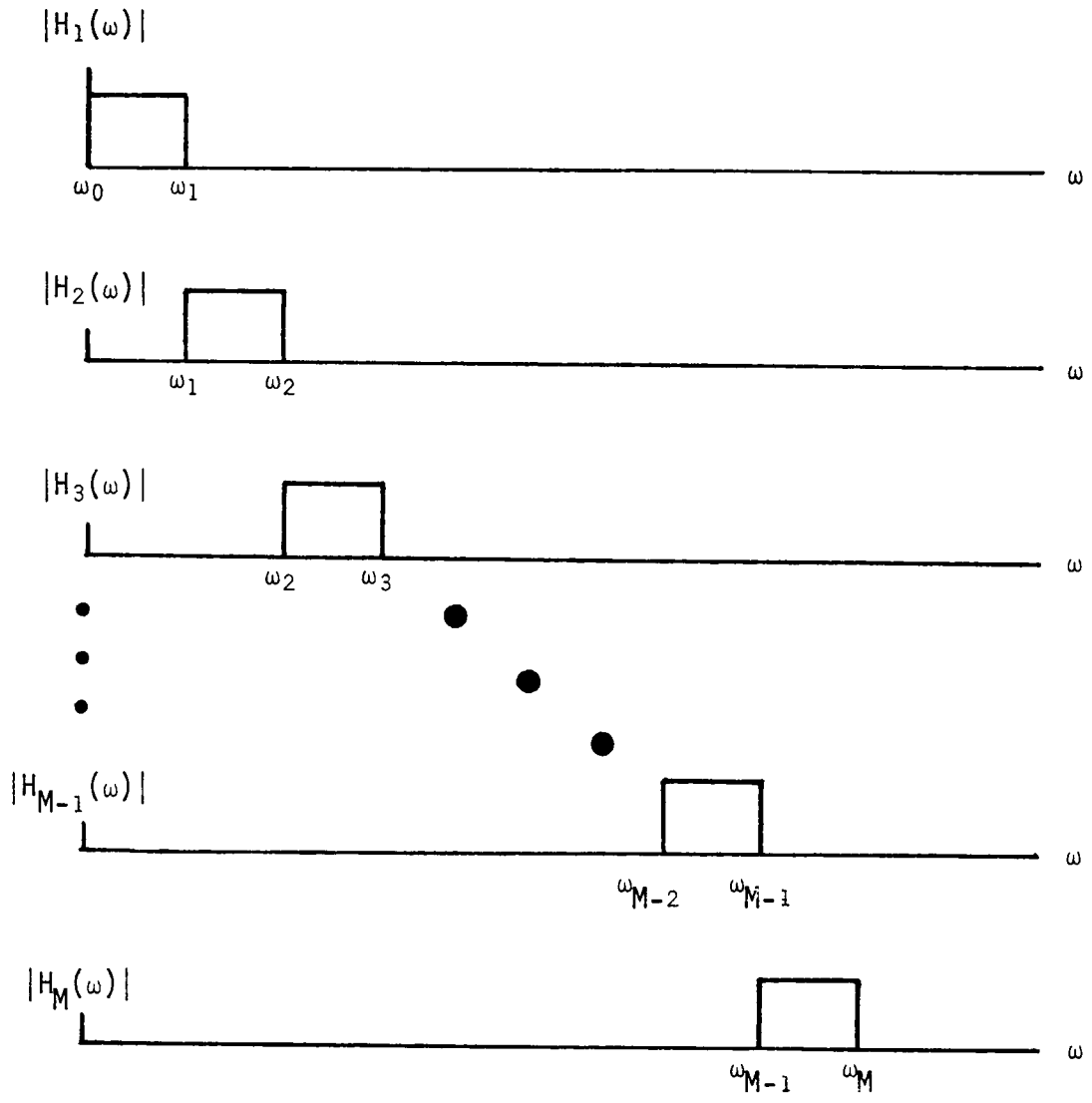
$$\Delta\omega_i = \omega_i - \omega_{i-1}. \qquad (2.1.4)$$

The resolution in frequency of the average power density over the spectrum of the input signal is obtained by selecting the bandwidth of each filter $\Delta\omega_i$, i = 1, 2, ... , M in the bank. The signal from which the average power densities are to be estimated is simultaneously fed into each of the filters in the bank. The average power density of the output of each channel is computed. These average power densities constitute point estimates on the true power spectrum density function characterizing the random process which produced the signal. These operations are depicted in Figure 2.4.

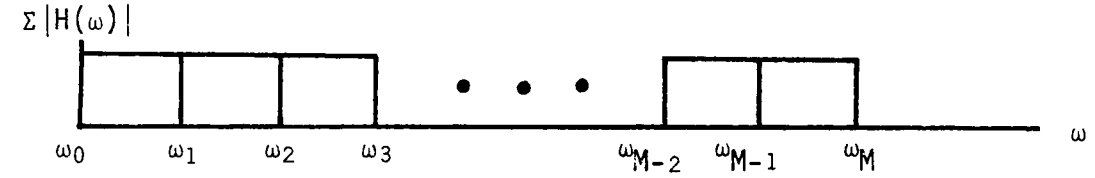
The LURF PSD is obtained by selecting adjacent bandedge frequencies according to

$$\omega_m/\omega_{m-1} = r^{1/s} \qquad (2.1.5)$$

The parameter r is the base of the logarithm, usually the integer 2 or 10. The parameter s determines the number of estimates between $\omega_1 r^{m-1}$ and $\omega_1 r^m$, i.e. the frequency resolution. For example, if r = 10 and s = 4 there are four analysis frequency estimate points per decade of frequency. Figure 2.5a depicts the frequency band demarcations for



(a) Individual filters



(b) Composite filter bank

Figure 2.3 Magnitude Functions for Ideal Bandpass Filter Bank

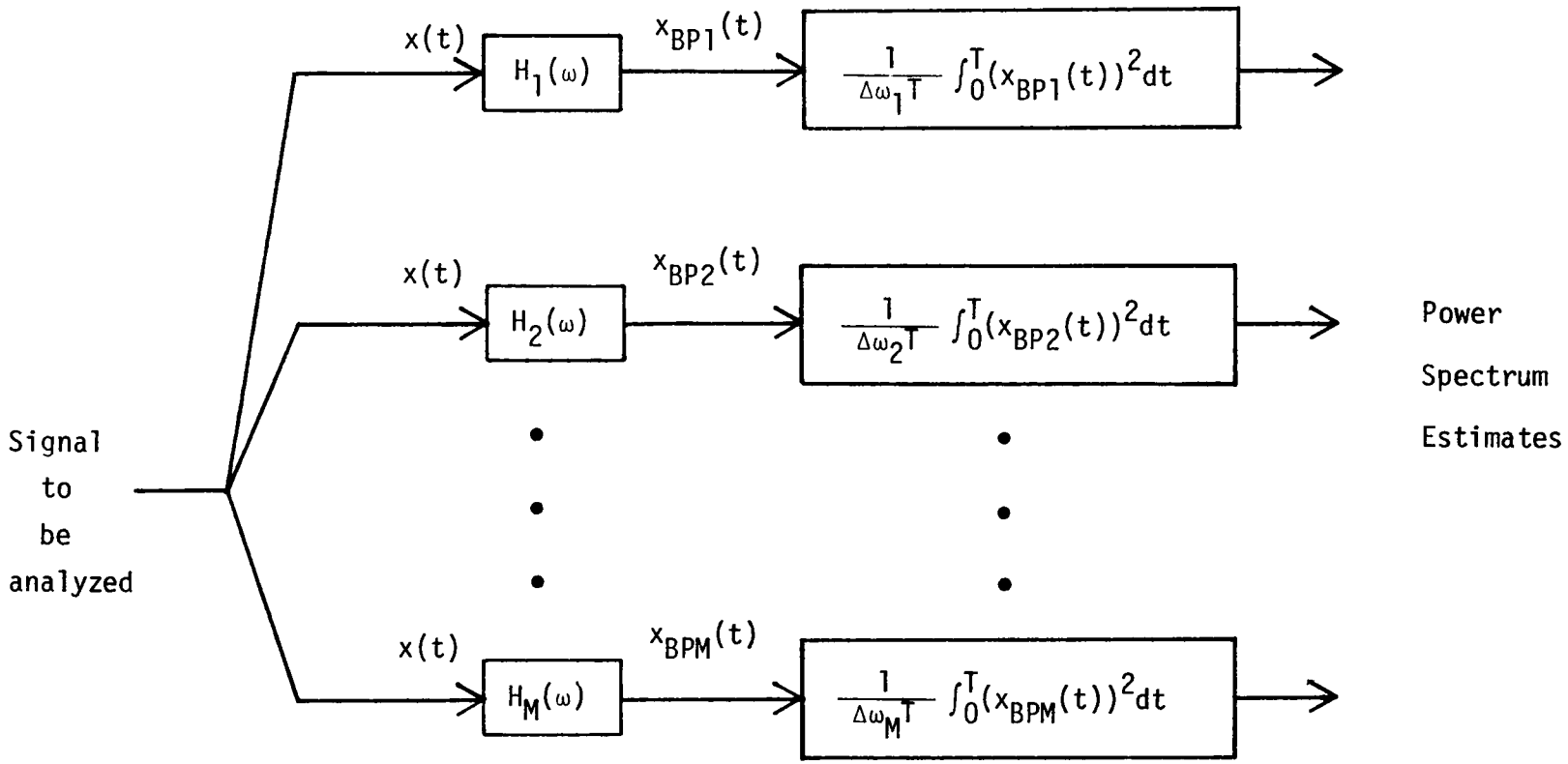
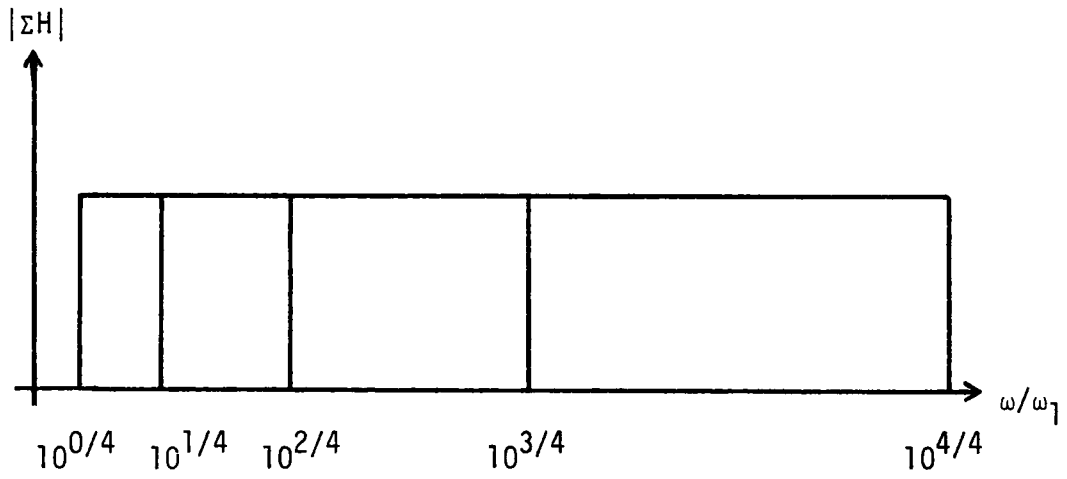
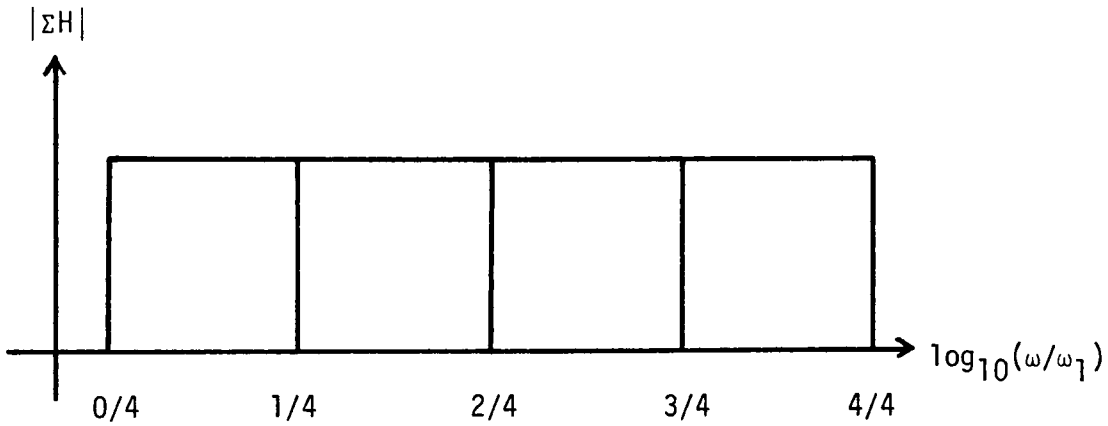


Figure 2.4 Bandpass Filter Bank Processing Scheme for Power Spectrum Estimation



(a) Linear frequency scale



(b) Logarithmic frequency scale

Figure 2.5 Frequency Scale Demarcation Comparison

such a set of ideal bandpass filters on a linear frequency scale over one decade, and Figure 2.5b depicts how these frequency band demarcations appear uniformly spaced on a logarithmic frequency scale.

The passbands of these ideal filters are contiguous. This insures that all of the power in the signal is represented. If there are gaps between the bandedges of adjacent filters, the power exhibited at the frequencies in the gaps would not be detected. If the passbands of the filter overlap, the power in the signal at the overlap band of frequencies would be represented twice. Since the bandedges of the ideal filters coincide, the bandwidths of the filters increase logarithmically uniform according to equation (2.1.5) and equation (2.1.4) as a function of frequency. The center frequency ω_m' of each filter is given in terms of its bandedge frequencies ω_m and ω_{m-1}, and the adjacent band's center frequency ω_{m-1}' by

$$\omega_m' = \sqrt{(\omega_m)(\omega_{m-1})} = \omega_m\sqrt{r^{-1/s}} = \omega_{m-1}\sqrt{r^{1/s}} = \omega_{m-1}' r^{1/s} \qquad (2.1.6)$$

A relation describing the normalized bandwidth is

$$\frac{\Delta\omega_m}{\omega_m'} = 1/Q = \sqrt{r^{+1/s}} - \sqrt{r^{-1/s}} \qquad (2.1.7)$$

which indicates that the quality factor Q for each filter in the bank is the same. Thus the LURF filter bank is also known as a constant-Q filter bank. Equation (2.1.7) also indicates that the bandwidth $\Delta\omega_m$

is proportional to the center frequency ω_m', giving rise to the LURF filter bank being called a proportional bandwidth filter bank.

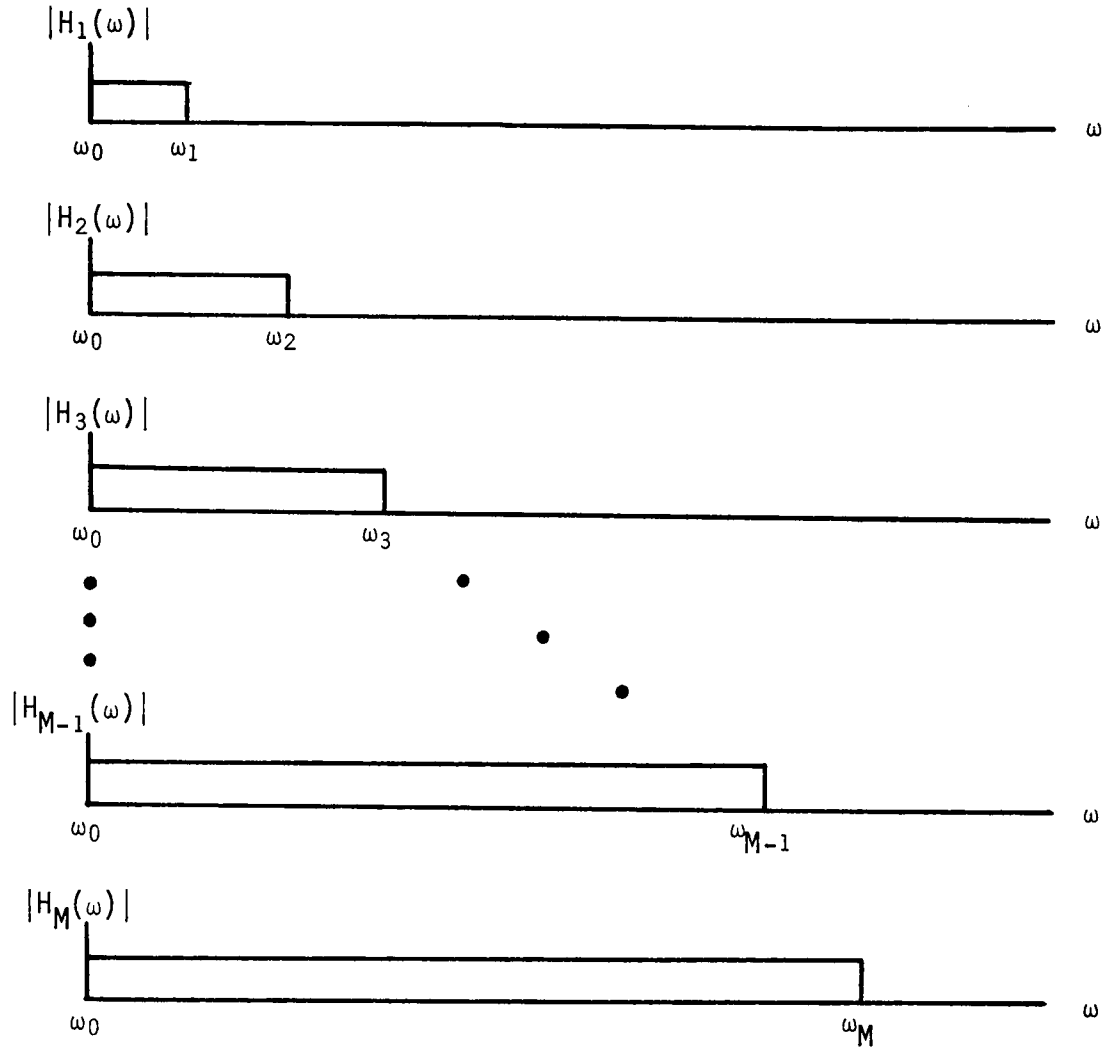
For nonideal filters, there have been many different bandwidth criteria which have been proposed and investigated (Peebles, 1976). In this study, no specific bandwidth criteria is adopted. The approaches proposed are presented in a general framework so that any definition of bandwidth can be employed.

The earliest proposed approaches to LURF spectrum estimation employed bandpass filter banks (Negron, 1966), (Maling, 1967), (Enochson, 1968), (Otnes, 1968), (DeFelice, 1976), (Strandberg, 1977). The use of bandpass, finite impulse response (FIR) digital filters has been proposed (Schafer, 1975), (Rabiner, 1978). Infinite impulse response (IIR), digital bandpass, constant-Q filter banks have been proposed which use the standard Butterworth magnitude approximation (Weaver, 1968), (Childers, 1975). It is obvious that other standard filter function approximation schemes are applicable.

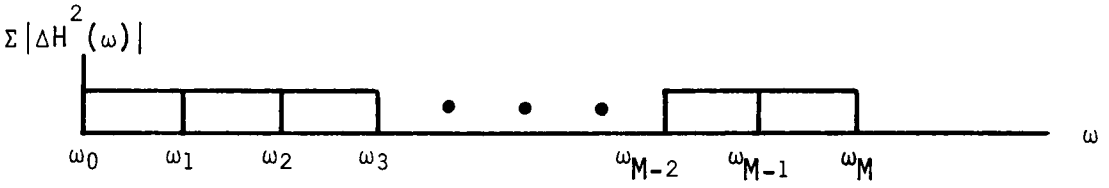
2.2 Lowpass Filter Bank

In this subsection a general approach to obtain the LURF PSD estimate using lowpass filters to achieve the effect of bandpass filters is presented.

Consider a set of ideal lowpass filters whose cutoff frequencies coincide with the bandedges in the bandpass filter bank of Figure 2.3. The magnitude functions for such a filter bank are individually shown in Figure 2.6a. A composite filter bank is formed which provides the same effect for power averaging purposes as the bandpass filter bank



(a) Individual filters



(b) Composite filter bank

Figure 2.6 Magnitude Functions for Ideal Lowpass Filter Bank

by differencing the adjacent magnitude functions

$$\Sigma|\Delta H^2(\omega)| = \sum_{i=1}^{M} (|H_i(\omega)|^2 - |H_{i-1}(\omega)|^2) \qquad (2.2.1)$$

as shown in Figure 2.6b. This assumes of course that the magnitude functions are equal in magnitude for all the filters in the bank. By making the cutoff frequencies coincide with the bandedges of the LURF bandpass filter bank bandedge definition given by equation (2.1.5), the LURF lowpass filter bank is realized. The processing scheme requires one more step than the bandpass approach. The outputs of the integrators must be differenced to obtain the bandpass effect. Thus point estimates of the PSD at LURF analysis frequencies are given by

$$\hat{S}(\omega_i') = \frac{1}{\Delta\omega_i T} \int_0^T [(x_{LPi}(t))^2 - (x_{LPi-1}(t))^2]dt \qquad (2.2.2)$$

where the analysis frequencies are given by equation (2.1.6) for $i = 1, 2, \ldots, M$. The lowpass signals $x_{LPi}(t)$ are the outputs of the lowpass filters having impulse responses $h_{LPi}(t)$. The filtering operation is described by

$$x_{LPi}(t) = \int_{-\infty}^{t} x(\tau) h_{LPi}(t-\tau)d\tau. \qquad (2.2.3)$$

The zeroth lowpass filter is not used, i.e. $h_{LP0}(t) = 0$ for all t. This processing scheme is shown in Figure 2.7 and differs from the bandpass filter bank scheme shown in Figure 2.4 because all the filters are lowpass and the average power density estimates are subtracted.

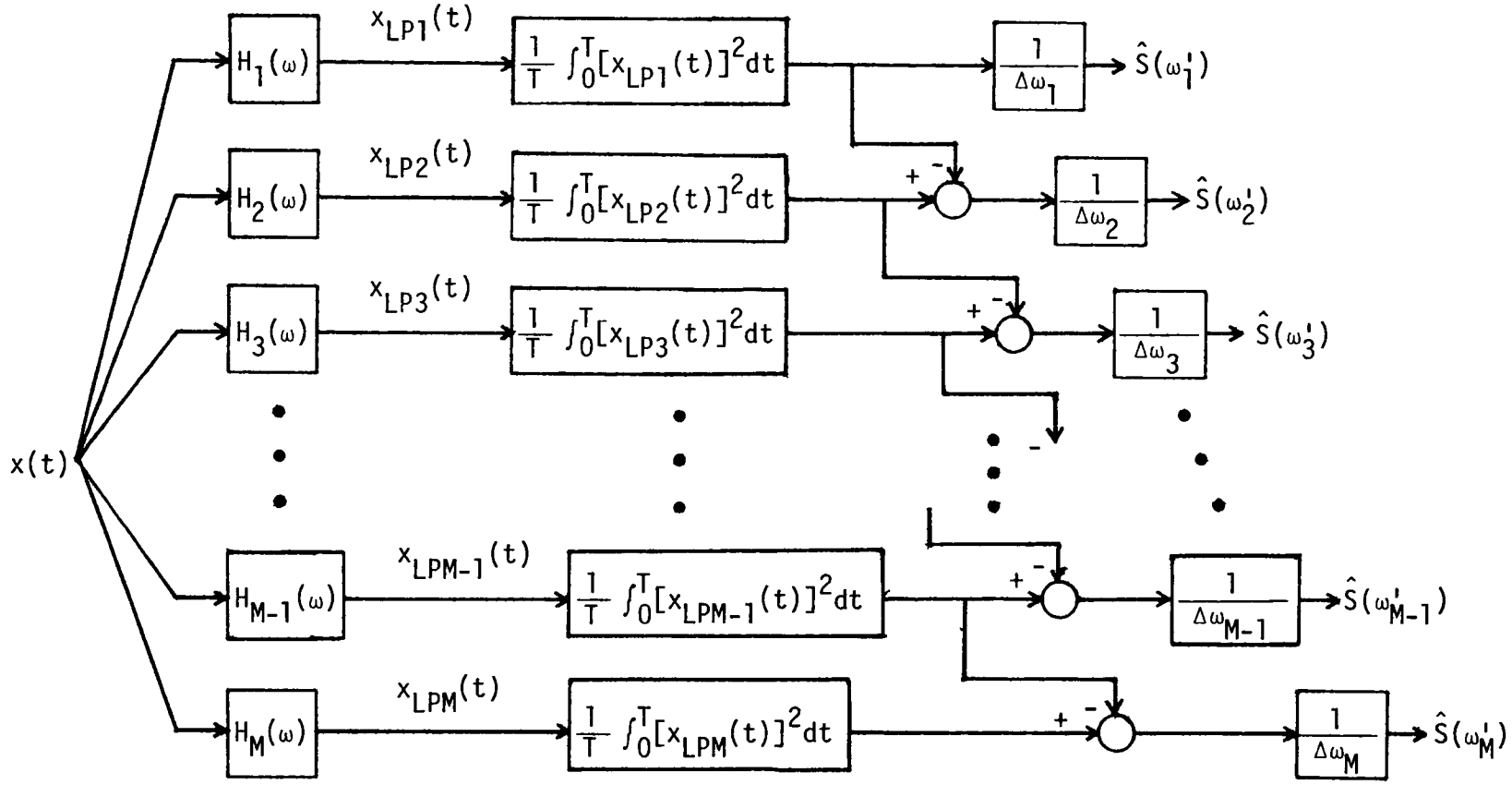


Figure 2.7 Lowpass Filter Bank Processing Scheme for Power Spectrum Estimation

To evaluate the confidence in the LURF PSD estimates obtained with lowpass filters, bounds on the estimation error variance and bias are developed below.

Let the true average power in the frequency band $|\omega| \leq \omega_i$ be P_i. The power in this lowpass frequency band for $0 \leq t \leq T$ is estimated by

$$\hat{P}_i = \frac{1}{T}\int_0^T\left(\int_{-\infty}^{+\infty}X(t-\tau)h_i(\tau)d\tau\right)^2dt. \qquad (2.2.4)$$

where $X(t)$ is the wide-sense stationary, ergodic, random process which generates the sample function $x(t)$. Taking the expected value of $\hat{P}_i$ yields

$$\begin{aligned}\overline{\hat{P}}_i &= E\{\hat{P}_i\} \\ &= \frac{1}{T}\int_0^T\left(\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}E\{X(t-\tau_1)X(t-\tau_2)\}h_i(\tau_1)h_i(\tau_2)d\tau_1d\tau_2\right)dt\end{aligned} \qquad (2.2.5)$$

Simplifying, equation (2.2.5) can be written as a convolution of the autocorrelation with the impulse response in the forward direction and then in the reverse direction

$$\overline{\hat{P}}_i = R(\tau) * h_i(-\tau) * h_i(\tau)\Big|_{\tau=0} \qquad (2.2.6)$$

where "*" denotes convolution, and τ is set equal to zero after both convolutions have been evaluated. The frequency domain interpretation of equation (2.2.6) is made by taking the Fourier transform of the right side of equation (2.2.6) before letting $\tau = 0$

$$S_{LPi}(\omega) = S(\omega)|H_i(\omega)|^2 \tag{2.2.7}$$

where $S(\omega)$ is the true PSD of the process $X(t)$, and $S_{LPi}(\omega)$ is the PSD of the lowpass filtered process $X_{LPi}(t)$. If the filter magnitude function is defined according to

$$|H_i(\omega)|^2 = \begin{cases} 1 & |\omega| \leq \omega_i \\ 0 & \text{elsewhere} \end{cases} \tag{2.2.8}$$

it is apparent that the filtering operation selects the power in the lowpass frequency band. The average power in the process $X_{LPi}(t)$ is given by the true power in the process $X(t)$ over the band of frequencies $|\omega| \leq \omega_i$ according to

$$P_i = \frac{1}{2\pi}\int_{-\omega_i}^{+\omega_i} S(\omega)d\omega = \frac{1}{2\pi}\int_{-\infty}^{+\infty} S_{LPi}(\omega)d\omega. \tag{2.2.9}$$

This result indicates that the lowpass filter power estimate $\hat{P}_i$ is an unbiased estimate of the true power P_i under ideal conditions.

Consider two such estimates $\hat{P}_1$ and $\hat{P}_2$ corresponding to average power estimates from two consecutive lowpass filters. $\hat{P}_1$ and $\hat{P}_2$ are random variables and are characterized by means P_1 and P_2 which are the true values. The variances represent the expected value of the square of the estimation error since the estimators are unbiased

$$\sigma_1^2 = E\{(\hat{P}_1 - P_1)^2\}, \tag{2.2.10}$$

$$\sigma_2^2 = E\{(\hat{P}_2 - P_2)^2\}. \tag{2.2.11}$$

There is also some correlation coefficient ρ_{12} which is defined by

$$\rho_{12} = \frac{C_{12}}{\sigma_1\sigma_2} = \frac{R_{12} - P_1P_2}{\sigma_1\sigma_2} \qquad (2.2.12)$$

where the crosscovariance C_{12} and the crosscorrelation R_{12} are defined by

$$C_{12} = E\{(\hat{P}_1 - P_1)(\hat{P}_2 - P_2)\} \qquad (2.2.13)$$

and

$$R_{12} = E\{\hat{P}_1\hat{P}_2\}, \qquad (2.2.14)$$

respectively.

An estimate of the average power in the band $\omega_1 \leq \omega \leq \omega_2$ is given by

$$\hat{P}_{21} = \hat{P}_2 - \hat{P}_1 \qquad (2.2.15)$$

where $\hat{P}_2$ and $\hat{P}_1$ are both found using the same sample function x(t) from the process X(t). The expected value of the estimate is given by

$$E\{\hat{P}_{21}\} = E\{\hat{P}_2 - \hat{P}_1\} = P_2 - P_1 = P_{21} \qquad (2.2.16)$$

thus establishing the unbiased property of this estimation scheme. The variance σ_{21}^2, which is the expected value of the square effective bandpass estimation error, is found after a few algebraic manipulations to be

$$\sigma_{21}^2 = E\{(\hat{P}_{21} - P_{21})^2\} = \sigma_2^2 - 2\rho_{21}\sigma_1\sigma_2 + \sigma_1^2. \qquad (2.2.17)$$

It is obvious that the correlation coefficient is always positive, i.e. $0 < \rho_{21} < 1$, because an increase in P_1 will always cause an increase in P_2. Hence bounds on the variance are given by

$$(\sigma_1 - \sigma_2)^2 < \sigma_{21}^2 < \sigma_1^2 + \sigma_2^2. \qquad (2.2.17)$$

Defining the time duration T_i (seconds) of the impulse response of a filter with bandwidth B_i (Hertz) to be

$$T_i = 1/B_i, \qquad (2.2.18)$$

the bounds on the variance become

$$\frac{(T_1 - T_2)^2}{T^2} < \sigma_{21}^2 < \frac{T_1^2 + T_2^2}{T^2}. \qquad (2.2.19)$$

For LURF spectrum estimation, adjacent lowpass bandedges are related by

$$\frac{f_i}{f_{i-1}} = r^{1/s} = \frac{T_{i-1}}{T_i} \qquad (2.2.20)$$

where $2\pi f_i = \omega_i$ and $2\pi f_{i-1} = \omega_{i-1}$ from equation (2.1.5). The i^{th} filter's settling time T_i can also be expressed in terms of the settling time of the first filter, viz.

$$T_i = T_1 r^{-i/s}. \qquad (2.2.21)$$

Using Rice's result (Rice, 1943) for the estimation error, the bounds on the variance for the i^{th} average power estimate can now be expressed by

$$\left(\frac{T_1}{T}\right)^2\left(r^{-\frac{i-1}{s}} - r^{-\frac{i}{s}}\right)^2 < \sigma_{i,i-1}{}^2 \qquad (2.2.22)$$

and

$$\sigma_{i,i-1}{}^2 < \left(\frac{T_1}{T}\right)^2\left[\left(r^{-\frac{i-1}{s}}\right)^2 + \left(r^{-\frac{i}{s}}\right)^2\right] . \qquad (2.2.23)$$

Factoring out the term $(r^{-i/s})^2$, yields

$$\left(\frac{T_1}{T}\right)^2\left(r^{-\frac{i}{s}}\right)^2\left(r^{+\frac{1}{s}} - 1\right)^2 < \sigma_{i,i-1}{}^2 \qquad (2.2.24)$$

and

$$\sigma_{i,i-1}{}^2 < \left(\frac{T_1}{T}\right)^2\left(r^{-\frac{i}{s}}\right)^2\left[\left(r^{+\frac{1}{s}}\right)^2 + 1\right] . \qquad (2.2.25)$$

In Figure 2.8, the normalized log of the upper and lower bounds of the variance are plotted. As can be seen upon inspection of equation (2.2.24) and (2.2.25), the upper and lower bounds plots are straight lines when plotted on a logarithmic abscissa. Also, the distance between the lower and upper bound is constant even though the overall variance is decreasing linearly on the logarithmic scale. Table 2.1 lists values of the normalized variance for the upper and lower bounds for r = 2, s = 4, and i = 1, 2, ... , M.

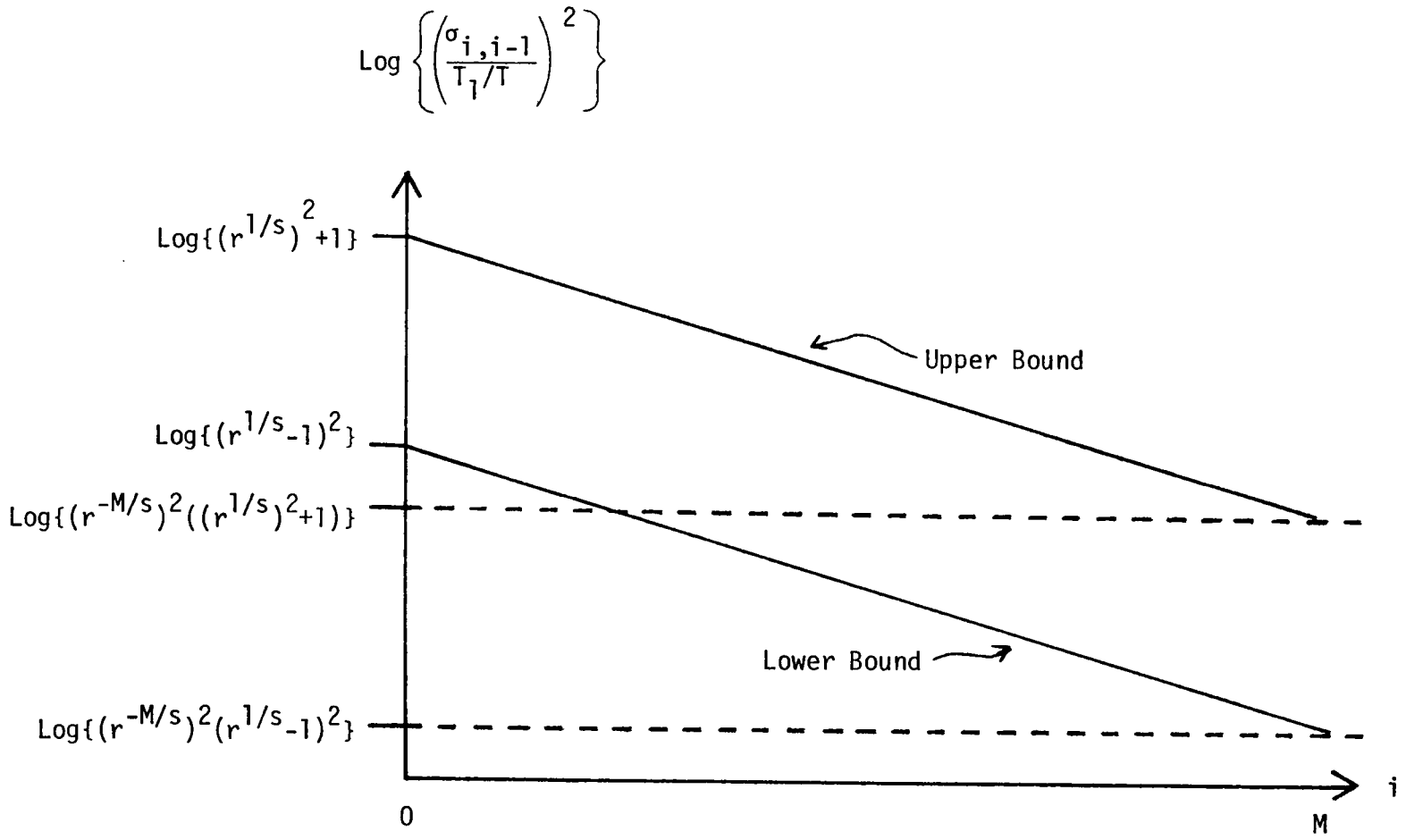


Figure 2.8 Plot of Log Normalized Variance

Table 2.1 List of Normalized Variance for r = 2, s = 4 and M = 20

Discrete Analysis Frequency	Normalized Variance $\left(\frac{\sigma_{i,i-1}}{T_1/T}\right)^2$	
	Lower Bound	Upper Bound
1	0.025314	1.707107
2	0.017900	1.207107
3	0.012657	0.853553
4	0.008950	0.603553
5	0.006328	0.426777
6	0.004475	0.301777
7	0.003164	0.213388
8	0.002237	0.150888
9	0.001582	0.106694
10	0.001119	0.075444
11	0.000791	0.053347
12	0.000559	0.037722
13	0.000396	0.026674
14	0.000280	0.018861
15	0.000198	0.013337
16	0.000140	0.009431
17	0.000099	0.006668
18	0.000070	0.004715
19	0.000049	0.003334
20	0.000035	0.002358

To compare the computation burden of the lowpass filter bank implementation of a LURF PSD estimator with the bandpass filter bank implementation, the amount of computation required to implement a LURF FIR digital filter will be compared. A FIR digital filtering operation is expressed by a discrete convolution

$$y[n] = \sum_{i=0}^{p} h[i]x[n-i] \qquad (2.2.26)$$

where x[n] is the filter input, y[n] is the filter output, and h[n] is a finite sequence of constants comprising the digital filter. The parameter p is the discrete settling time of the filter and is inversely proportional to the bandwidth of the filter. The settling time also depends on factors such as the transition band roll-off rate. As an arbitrary definition of settling time, let

$$p = N/N_c \qquad (2.2.27)$$

where N_c is the one-sided bandwidth of the digital filter

$$\Delta\omega = 2\pi N_c/N = 2\pi(N_2 - N_1)/N \qquad (2.2.28)$$

with bandedges described by N_2 and N_1 and folding frequency $\omega = \pi$. It should be remembered that the actual value of p in an implementation would also influence other factors such as stopband attenuation, roll-off rate, etc. To implement an FIR digital filter bank requires the help of a digital computer to design it. Available computer-aided FIR filter design algorithms usually do not permit the specification

of bandedge frequencies consistent with conventional bandwidth definitions. The algorithm inputs usually include the desired cutoff frequency, stopband attenuation and the number of filter terms p. The specified cutoff frequency is not used to provide bandwidth according to a given criteria. The resulting 3db. bandwidth cutoff frequency, for example, cannot be specified (Rabiner, 1975).

However, it is easy to show that if all of the characteristics except for bandwidth remain constant, the settling time for a lowpass and bandpass filter are both the same inverse proportion of bandwidth. Any prototype lowpass filter can be converted to a bandpass filter by modulating the lowpass impulse response by a sinusoid of the appropriate frequency

$$h_{BP}[n] = \cos[\omega_0' n] h_{LP}[n] \qquad (2.2.29)$$

where $\omega_0' = 2\pi N_0'/N$ is the center frequency of the desired bandpass. Taking the Fourier transform of equation (2.2.29), it is seen that the bandwidth of the resulting bandpass is twice that of the prototype lowpass with the same settling time.

For the digital filter bank, the LURF criteria can be expressed by

$$\omega_m/\omega_{m-1} = N_m/N_{m-1} = r^{1/s} \qquad (2.2.30)$$

and the bandwidth is given by

$$\Delta N_m = (N_m - N_{m-1})/N = N_m(1-r^{-1/s})/N. \qquad (2.2.31)$$

In terms of the lowest and narrowest analysis frequency bandwidth,

$$\Delta N_1 = N_1/N, \tag{2.2.32}$$

the bandwidth expression for m = 2, 3, ... , M becomes

$$N_m = r^{(m-1)/s}(1-r^{-1/s})N_1/N. \tag{2.2.33}$$

Thus for a bandpass filter bank, the settling time for the m^{th} filter in the bank is given by

$$P_{BPm} = \frac{N}{N_1(1-r^{-1/s})r^{(m-1)/s}} . \tag{2.2.34}$$

For M such filters in the bandpass bank, the total number of arithmetic operations is proportional to

$$P_{BP,total} = \frac{N}{N_1} + \frac{N}{N_1(1-r^{-1/s})}\left[r^{-1/s} + r^{-2/s} + \ldots + r^{-(M-1)/s}\right]. \tag{2.2.35}$$

Using the closed form expression for a geometric series and some algebraic manipulation, equation (2.2.35) can be expressed in closed-form as

$$P_{BP,total} = \frac{N}{N_1(1-r^{-1/s})}\left[\left(\frac{1-r^{-M/s}}{1-r^{-1/s}}\right) - r^{-1/s}\right] . \tag{2.2.36}$$

In the lowpass filter bank implementation, there are also M filters. The settling time or number of multiplications and additions required for each filter m = 1, 2, ... , M is proportional to

$$p_{LPm} = \frac{N}{N_1 r^{(m-1)/s}} \quad . \tag{2.2.37}$$

For M such filters in the lowpass bank, the total number of arithmetic operations is proportional to

$$p_{LP,total} = \frac{N}{N_1} \left[1 + r^{-1/s} + r^{-2/s} + \ldots + r^{-(M-1)/s} \right]$$

$$= \frac{N}{N_1} \left(\frac{1-r^{-M/s}}{1-r^{-1/s}} \right). \tag{2.2.38}$$

The computational advantage of the lowpass bank versus the bandpass bank is given by

$$\frac{p_{BP,total}}{p_{LP,total}} = \left(\frac{1}{1-r^{-1/s}} \right) - \left(\frac{r^{-1/s}}{1-r^{-M/s}} \right) \tag{2.2.39}$$

Depending on the LURF parameters r and s, the computational advantage can be dramatic. Figure 2.9 shows the computational advantage of the lowpass LURF over the bandpass LURF for a FIR digital filter bank for r = 10. The finer the resolution, the more dramatic the computational advantage becomes. For coarse resolution, the bandwidths of the bandpass filters are very wide at the higher analysis frequencies and are closer to the bandwidths of the lowpass filters. Thus the computational advantage is less for coarse resolution.

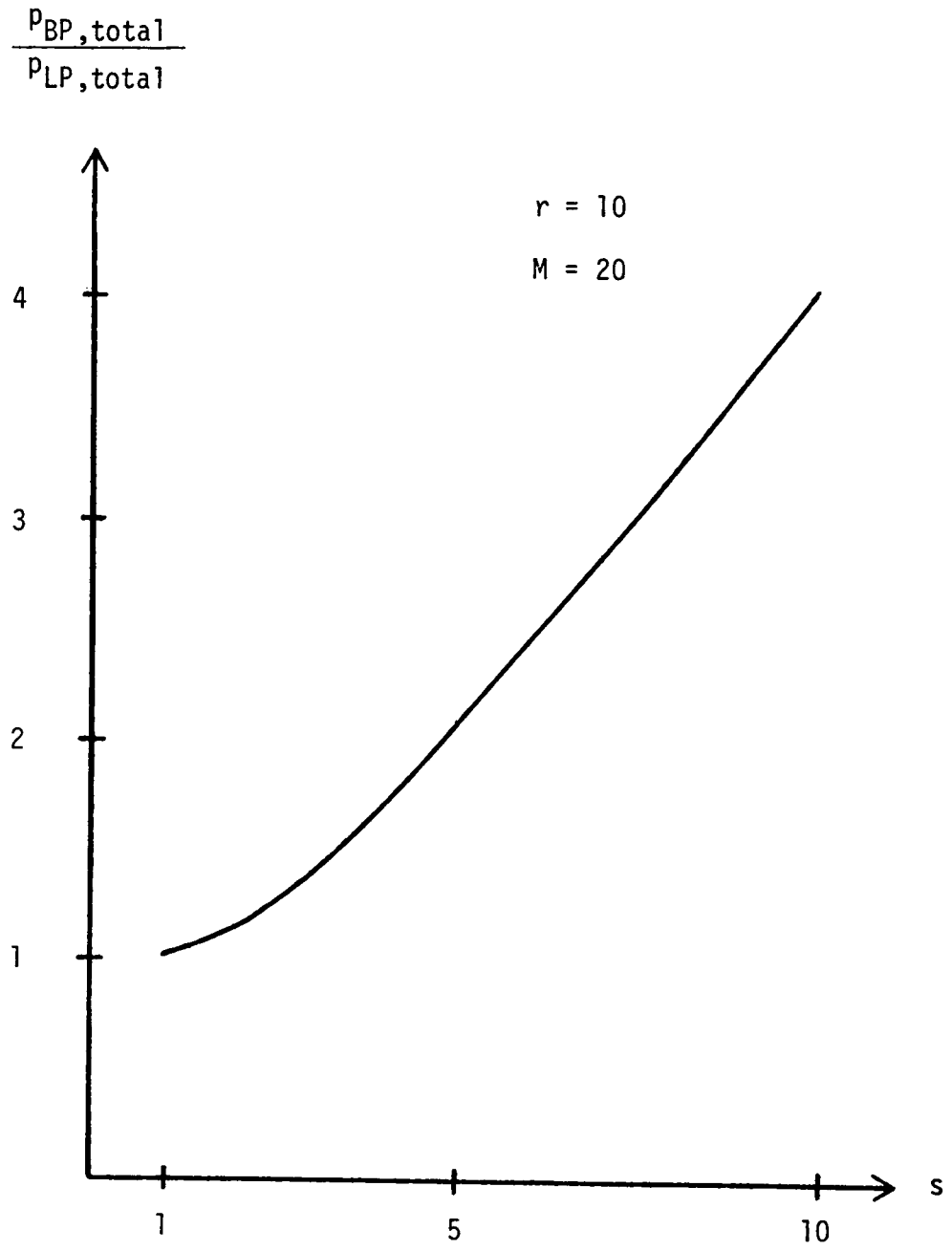


Figure 2.9 Computational Advantage of Lowpass over Bandpass

2.3 Other Filter Methods

There are some other schemes that have been proposed which share some of the characteristics of the general lowpass LURF approach just presented.

The nested variance power spectrum (NVPS) (Owens, 1977) approach to LURF PSD estimation provides $\log_2 N$ estimates from N time series data points. Later, the similarity between the NVPS approach and the LURF lowpass filter bank approach will be examined. The nested variance approach was originally called pilot estimation and was suggested as a technique to synthesize a prewhitening filter to improve subsequent harmonic spectrum estimation of time series data (Blackman, 1958). The pilot estimation technique was inadvertently modified by (Owens, 1977), who developed a derivation of the NVPS based on statistical concepts.

An estimate of the mean of a zero-mean, ergodic, Gaussian random process is given by

$$\bar{x}_T = \frac{1}{T}\int_0^T x(t)dt \qquad (2.3.1)$$

where x(t) is a sample function from the process. The variance σ_T^2 of this mean estimate is given by

$$\sigma_T^2 = E\{(\bar{x}_T)^2\}. \qquad (2.3.2)$$

Letting x(t) be represented by its Fourier transform $X(\omega)$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(\omega) e^{j\omega t} d\omega \qquad (2.3.3)$$

it has been shown (Owens, 1977) that the frequency domain representation of the variance is given by

$$\sigma_T^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S(\omega) |H_{LP}(\omega;T)|^2 d\omega \qquad (2.3.4)$$

where $S(\omega)$ is the PSD of the process and

$$|H_{LP}(\omega;T)|^2 = \frac{\sin^2(\omega T/2)}{(\omega T/2)^2} \qquad (2.3.5)$$

is a filter power transfer function or spectral window.

Now if the data are averaged over another time interval, say 2T, the difference

$$\sigma_T^2 - \sigma_{2T}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S(\omega) |H_{BP}(\omega;T,2T)|^2 d\omega \qquad (2.3.6)$$

where

$$|H_{BP}(\omega;T,2T)|^2 = |H_{LP}(\omega;T)|^2 - |H_{LP}(\omega;2T)|^2$$

$$= \frac{\sin^4(\omega T/2)}{(\omega T/2)^2}, \qquad (2.3.7)$$

gives an estimate of the power in the process according to the equivalent bandpass filter function. The filter function of equation (2.3.7) provides a rough bandpass filter shape as shown in Figure 2.10. If one interprets the bandedge frequencies to be $f_1 = 1/4T$ and

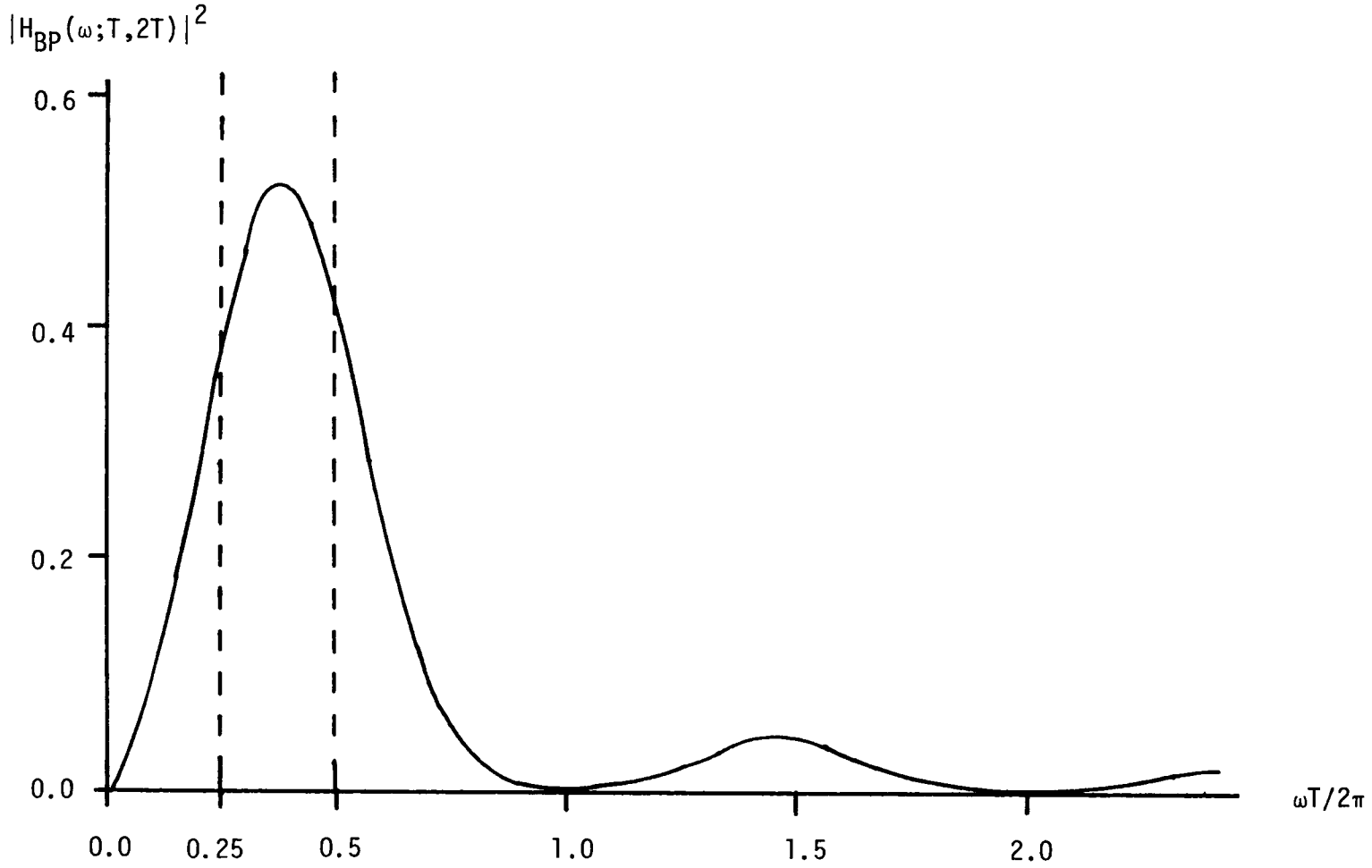


Figure 2.10 Nested Variance Power Spectrum Window Function

$f_2 = 1/2T$, the power in this band is given by

$$P_{T,2T} = 4T(\sigma_T^2 - \sigma_{2T}^2). \qquad (2.3.8)$$

By calculating the variance for data taken over 4T, subtracting from the variance for data taken over 2T and letting T become 2T in equation (2.3.8); the power over the band from f = 1/8T to f = 1/4T is obtained. This process is repeated until all of the available data have been used. Thus the implementation of this technique requires the estimation of the variance σ_{nT}^2 and the associated weighted subtractions.

The severe limitations of the NVPS approach to spectrum estimation have been pointed out in the literature (Rickard, 1979). The filter function, as shown in Figure 2.10, is so broad that approximately 43 percent of the power in a particular analysis frequency comes from the two adjacent analysis frequencies. This overly broad spectral window causes gross inaccuracies in the estimates when the actual spectrum is steeper than a power law relation of second order, viz.

$$S(\omega) \propto \omega^{-2}. \qquad (2.3.9)$$

The actual implementation of the NVPS is very sensitive to the zero-mean assumption and linear trends because of the integration operation on the raw data. In addition to the limitations mentioned above, the sampled-data implementation introduces a severe aliasing problem (University of Tennessee, 1982). The equivalent sampled-data filter function has been shown to be

$$|H_{BP}(\omega;m,2m)|^2 = \frac{\sin^4(\omega m\Delta t/2)}{\sin^2(\omega\Delta t/2)} \qquad (2.3.10)$$

where Δt is the sample rate, $m\Delta t$ is T, the interval of available data.

Although the NVPS approach has severe limitations, it can be used in applications where the computational constraints are very severe. The practical implementation can be realized using only fixed point additions, subtractions and squaring operations. Since the implementation requires virtually no multiplications, the computational speed is very fast. The implementation details are presented next and compared with the lowpass LURF PSD estimation scheme.

The relationship between the NVPS and the lowpass LURF PSD estimator is seen by examining the implementation of the NVPS in terms of filtering and averaging operations instead of statistical considerations. The computation of the variance σ_T^2 is accomplished using an integration operation of finite time duration

$$\sigma_T^2 = \sigma^2(t)\Big|_{t=T} = \left(\frac{1}{T}\int_0^t x(\tau)d\tau\right)^2\Big|_{t=T}. \qquad (2.3.11)$$

The conventional operation of a filter produces a whole waveform at the output; the NVPS algorithm only uses the terminal value of the filter output at t = T. If the available input waveform has a total length of LT where L is an integer, the NVPS algorithm calculates σ_T^2 by averaging L finite integrations, each of duration T. Therefore, the variance calculations are made as follows

$$\sigma_T^2 = \frac{1}{L} \sum_{n=1}^{L} \sigma_T^2[n] \qquad (2.3.12)$$

where

$$\sigma_T^2[n] = \left(\frac{1}{T}\int_{(n-1)T}^{nT} x(\tau)d\tau\right)^2 . \qquad (2.3.13)$$

The sampled data implementation merely uses a summation operation instead of the integration operation. The integration operations indicated in equation (2.3.13) correspond to lowpass filters with the following impulse response

$$h_{nT}(t) = u(t-(n-1)T) - u(t-nT) \qquad (2.3.14)$$

where $u(t)$ is a unit step function. The transfer function is given by

$$H_{nT}(\omega) = Te^{-j\omega(n-1/2)T} \frac{\sin \omega T/2}{\omega T/2} . \qquad (2.3.15)$$

There are some important differences between the NVPS approach and the lowpass LURF filter bank. First, the NVPS algorithm only uses the terminal value of each filter operation. This corresponds to a data decimation operation in the sampled-data implementation and accounts for the aliasing problem which has been previously pointed out (Durand, 1983). Second, the highest frequency estimates are obtained first, followed by progressively lower frequency estimates. This order of computation appears to be opposite of the lowpass LURF algorithm. The calculations proceed this way because the lower

analysis frequency estimates make use of the previous integration calculations in the filters as follows

$$\sigma_{2T}^2[n] = \left(\frac{1}{T}\int_{(n-1)T}^{nT} x(\tau)d\tau + \frac{1}{T}\int_{nT}^{(n+1)T} x(\tau)d\tau\right)^2 . \qquad (2.3.16)$$

Thus the calculations do not proceed in the same order as the lowpass LURF but do implement the same type of computation. This "nesting" accounts for a significant reduction in the computations required in the NVPS approach.

Although the lowpass filter bank LURF PSD estimation approach requires much more computation than the NVPS approach, it can be employed to provide much greater quality estimates than the NVPS because the filter function can be designed to overcome the limitations of the NVPS. Except for applications where computational efficiency is the overwhelming consideration, the lowpass LURF approach provides important advantages over the NVPS approach due to its much greater flexibility.

Another filtering scheme that employs lowpass filters similar to the lowpass LURF approach has been proposed (Enochson, 1968). The proposed approach consists of implementing a highpass filter, two bandpass filters and a lowpass filter with output decimation in a recursive manner. A block diagram of these operations appears in Figure 2.11. If the data were originally sampled at intervals of every T_0 seconds, where the filter functions are chosen as shown in Figure 2.12, after decimation the new sample rate is $T_1 = 2T_0$, and the data are passed through the filters again. For this second pass,

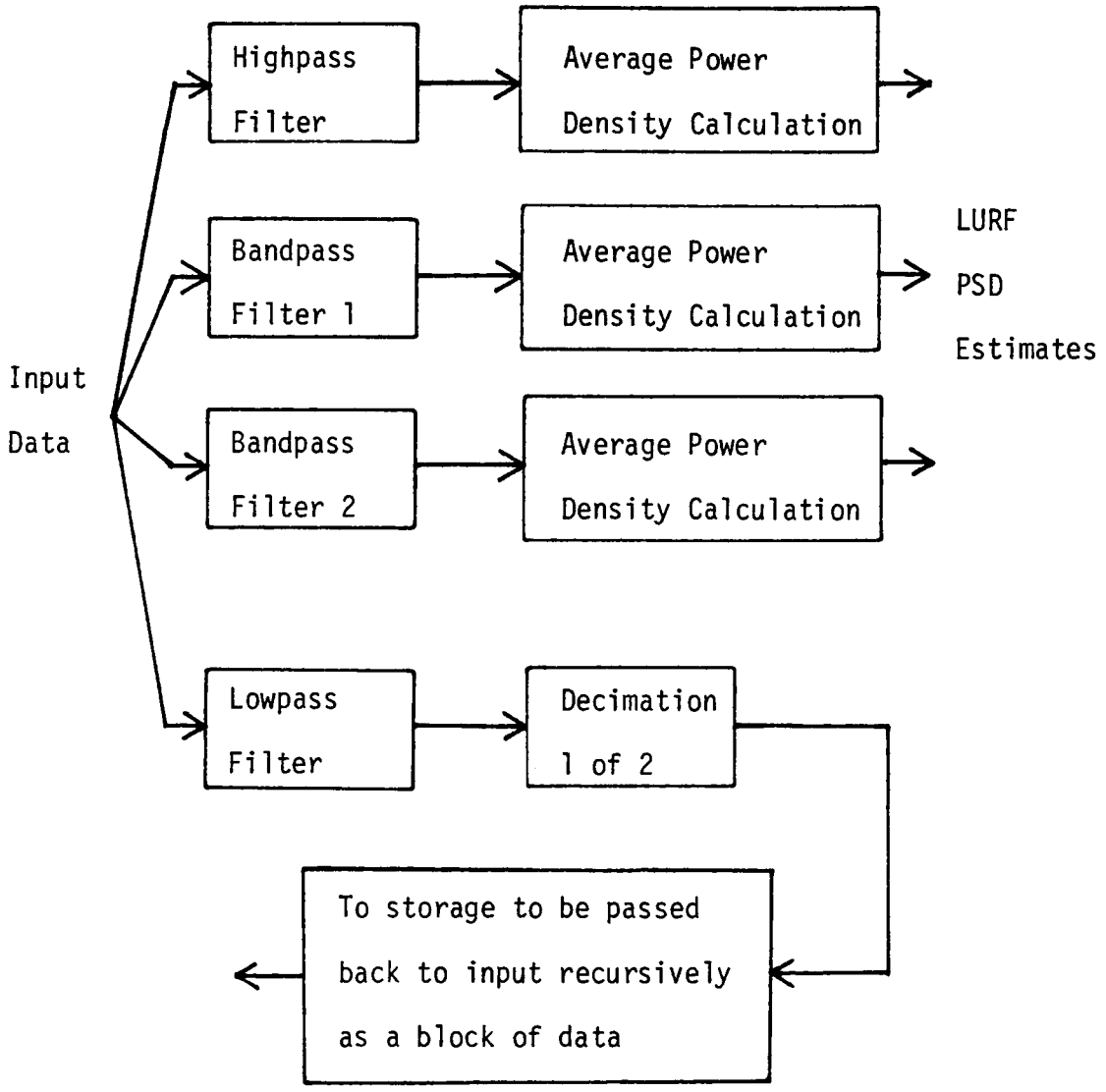


Figure 2.11 Enochson's Recursive Filter Bank Processing Scheme

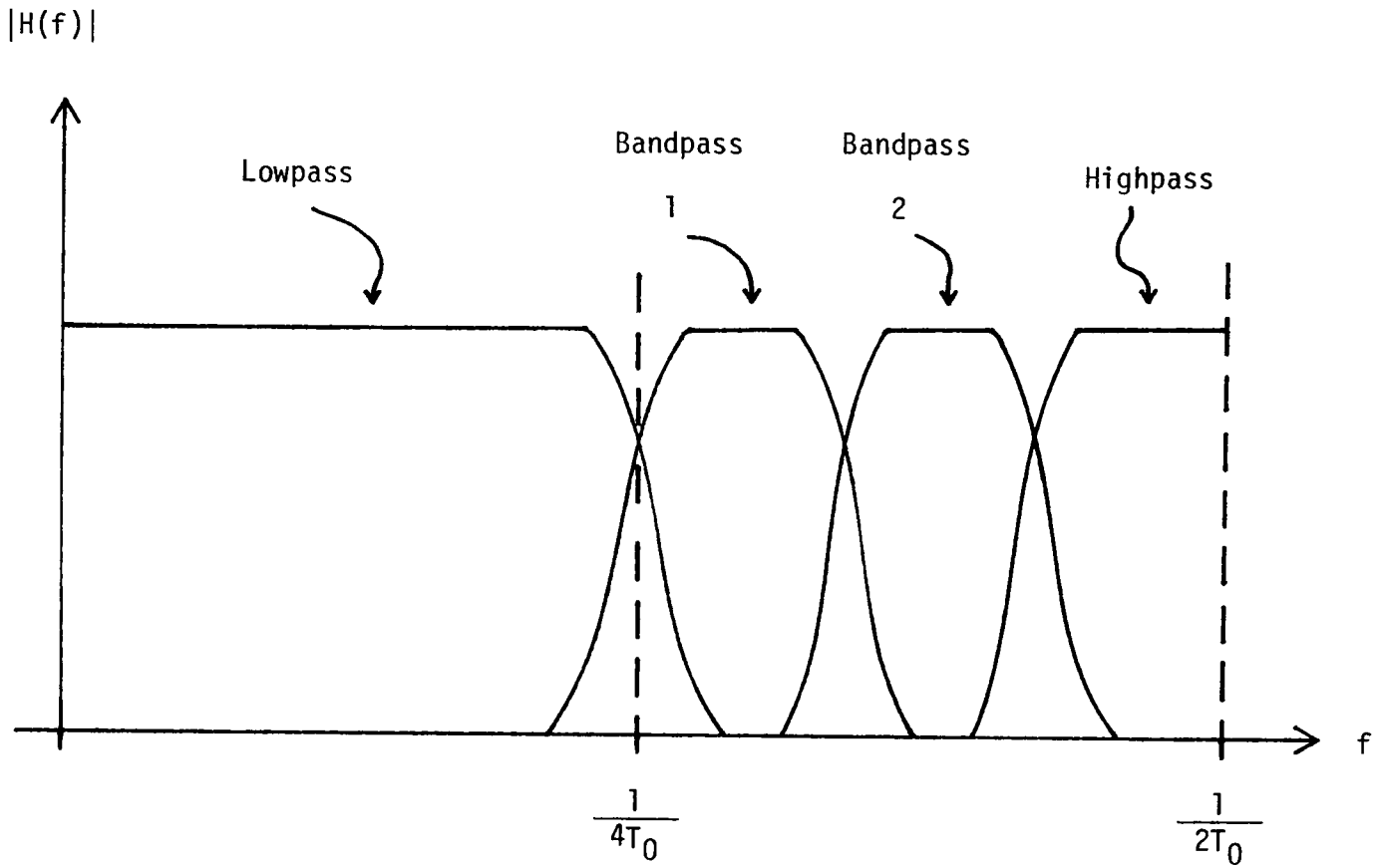


Figure 2.12 Enochson's Recursive Filter Bank Magnitude Functions

the filter functions have the same shape as shown in Figure 2.12 except that the frequency bands are now one octave lower. This approach is not strictly a lowpass scheme, but suggests another way the lowpass LURF filter bank could be implemented.

Instead of modulating a lowpass prototype filter's impulse response as indicated by equation (2.2.29) to obtain a bandpass filter, one can repeatedly demodulate the signal itself. This basic notion has been implemented (Harris, 1976). The sequence of operations is shown in Figure 2.13. Only the preprocessing operations have been shown to illustrate the complexity of this algorithm. Subsequent processing of the output data from the upper path consists of computing the DFT and averaging adjacent spectrum estimates with a frequency domain window. The window consists of seven values for each analysis frequency and is constructed by varying the Kaiser-Bessel window's stopband attenuation parameter. The windowing operation takes the output of the octave preprocessor and convolves it with the family of windows to obtain the LURF power spectrum estimates over one octave. The recursive operation of this scheme provides coverage for lower octaves at each recursion. This is similar to the approach of frequency domain windowing which is developed later.

The quality with which logarithmically uniform spectrum estimates are produced depends on how well the lowpass filters prevent aliasing and the formulation of the frequency domain window. Since the filters are applied recursively, any aliasing will worsen every time the filter reprocesses the data. The quality of the digital filters' antialiasing capabilities depends on many factors including the settling time.

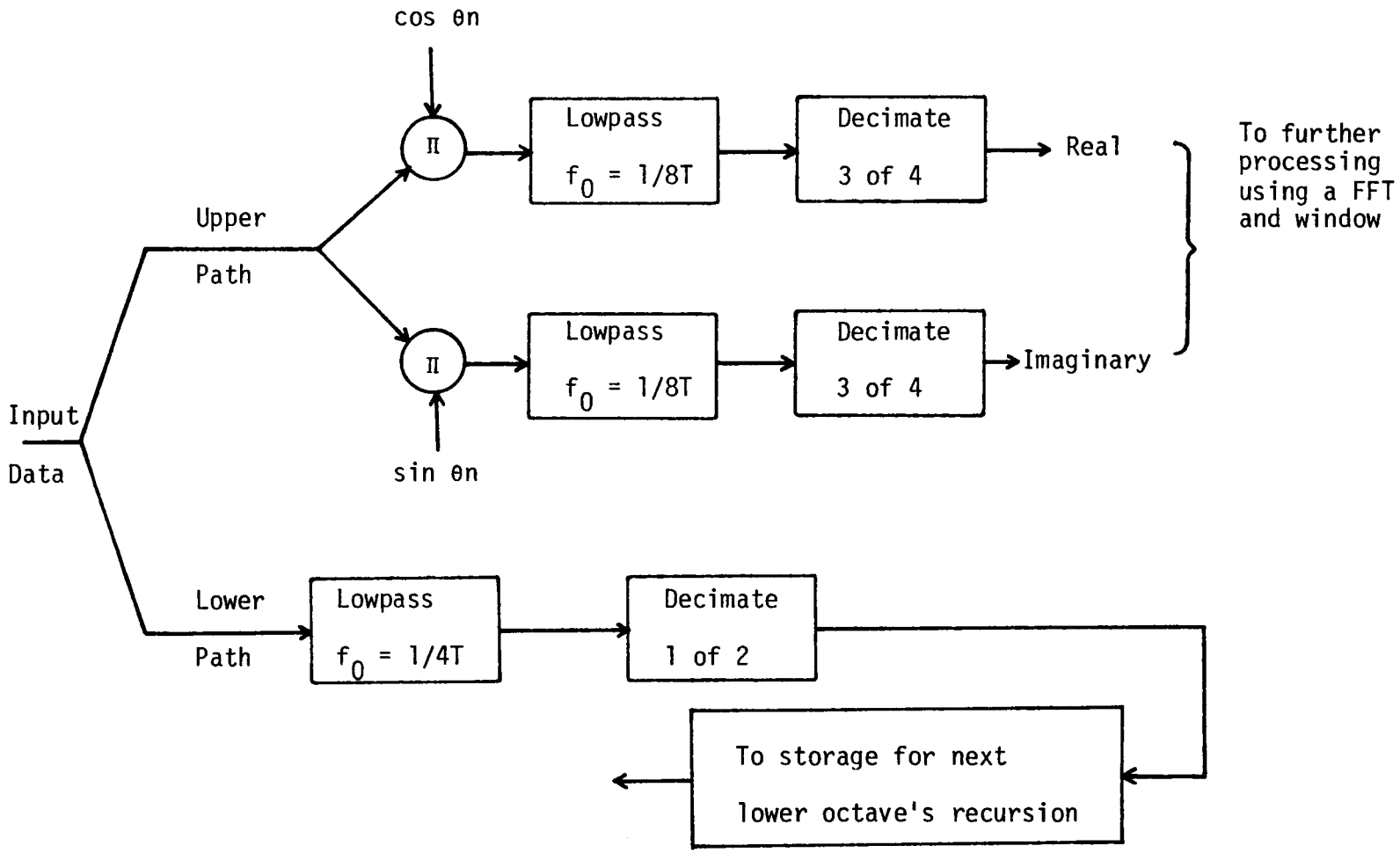


Figure 2.13 Harris' Recursive Octave Preprocessor

As proposed in the literature (Harris, 1976), no specific implementation or experimental results were presented to illustrate the performance of this scheme.

3. DISCRETE FOURIER TRANSFORM APPROACH

3.1 Frequency Domain Windowing

The LURF power spectrum can be obtained using the discrete Fourier transform. This approach consists of computing discrete points of the periodogram of the raw time series data with a fast Fourier transform algorithm and then smoothing the periodogram by convolving it with a frequency dependent window. For harmonically spaced frequency estimates, the DFT can be computed using a FFT algorithm with great computational savings as compared to the naive, brute force evaluation of the DFT. To obtain computational efficiency, FFT algorithms exploit the fact that the point estimates of the power spectrum are obtained for analysis frequencies which are integral multiples of a fundamental frequency. The DFT equation also possesses the property of equal frequency resolution for each point estimate. This corresponds to a constant bandwidth analysis and not a constant-Q analysis (using a filter bank analogy). This second difficulty arises because a time domain window is multiplied with the sample autocorrelation function. When a time domain windowed, sample autocorrelation function is Fourier transformed in accordance with the Wiener-Khinchin theorem to obtain the PSD, the frequency domain representation of the window is frequency invariant and thus independent of each analysis frequency.

To obtain the LURF power spectrum it is necessary to vary the bandwidth of the frequency domain window linearly with the analysis frequency. This can be accomplished by performing the windowing operations in the frequency domain. This approach constitutes the

second LURF PSD estimation scheme presented. First, a brief discussion of DFT-based spectrum estimation is presented to lay the foundation for the frequency domain windowing approach.

The DFT of an N-point discrete-time sequence was defined by equation (1.3.26) to be

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi nk/N}, \qquad (3.1.1)$$

for k = 0, 1, ... , N-1. The frequency argument k corresponds to N discrete, uniformly spaced radian frequencies $\omega_k = 2\pi k/N\Delta t$, for k = 0, 1, ... , N-1. This computation appears to require on the order of N^2 complex multiplications and additions. Using an FFT algorithm, the N quantities X[k]; k = 0, 1, ... , N-1 can be obtained with a computational burden which is on the order of $N(n_1+n_2+\ldots+n_p)$ complex additions and multiplications where $N = n_1n_2n_3\cdots n_p$, and the n_i for i = 1, 2, ... , p are the p prime factors of N (Cooley, 1965). Thus the use of the FFT entails a substantial computational savings when N is a highly composite number such as $N = 2^L$, where L is an integer.

To obtain an estimate of the PSD with harmonic spacing and constant resolution on a linear scale, the periodogram

$$I[k] = \frac{1}{N}|X[k]|^2, \qquad (3.1.2)$$

can be employed (Schuster, 1898), (Box, 1976). It can be shown that the sample autocorrelation

$$R[m] = \frac{1}{N} \sum_{n=0}^{N-1-m} x[n]x[n+m] \qquad (3.1.3)$$

for m = 0, 1, ... , N-1 is related to the periodogram by the DFT

$$I[k] = \sum_{m=0}^{N-1} R[m]e^{-j2\pi mk/N} \qquad (3.1.4)$$

if x[n] is a N/2-point sequence to which N/2 zero values have been appended (Brigham, 1974).

Without further processing, the periodogram is a poor estimator of the PSD. The variance of the periodogram approaches the square of the true PSD as N grows without bound (Priestley, 1981). Since the variance does not asymptotically converge to zero, the periodogram is not a consistent estimator of the PSD (Jenkins, 1968). A graphic portrayal of this can be found in (Oppenheim, 1975, 546).

For harmonic analysis the variance of the PSD point estimates can be reduced using a procedure which can also utilize the computational advantages of a FFT algorithm. The procedure begins by computing the sample autocorrelation using equation (3.1.3), or by computing the periodogram via equations (3.1.1) and (3.1.2) and taking the inverse DFT of the periodogram

$$R[m] = \frac{1}{N} \sum_{k=0}^{N-1} I[k]e^{+j2\pi mk/N} . \qquad (3.1.5)$$

Next, the sample autocorrelation R[m] is multiplied by a "lag window" function w[m] which tapers the values of R[m] to zero in M lag values.

Thus the windowed autocorrelation function becomes

$$R_w[m] = R[m]w[m] \tag{3.1.6}$$

where w[m] is maximum at m = 0 and tapers off to zero in M lag values, i.e. m = M. Due to the periodic nature of the DFT equation, the window is symmetrical about the point m = N/2,

$$w[m] = w[N-m]. \tag{3.1.7}$$

Finally, the PSD estimates are computed by taking the DFT of $R_w[m]$

$$\hat{S}[k] = \sum_{m=0}^{N-1} R_w[m]e^{-j2\pi km/N} \quad . \tag{3.1.8}$$

This sequence of operations is depicted in Figure 3.1.

As shown in Figure 3.1 these operations require two DFT calculations, one IDFT calculation, one magnitude squaring operation and one time function window multiplication operation. This sequence of operations is required because the window is applied to the sample autocorrelation function. This is called quadratic windowing (Slone, 1969). Another type of windowing is sometimes performed which eliminates one DFT and the IDFT operation. The window is multiplied directly to the raw time series data and then the periodogram is computed. This is called linear windowing. The sequence of operations for linear windowing to obtain the PSD estimate is shown in Figure 3.2.

The purpose of windowing is two-fold from a statistical point of view. First, the application of a window should improve (decrease) the variance of the PSD estimates. This is accomplished using quadratic

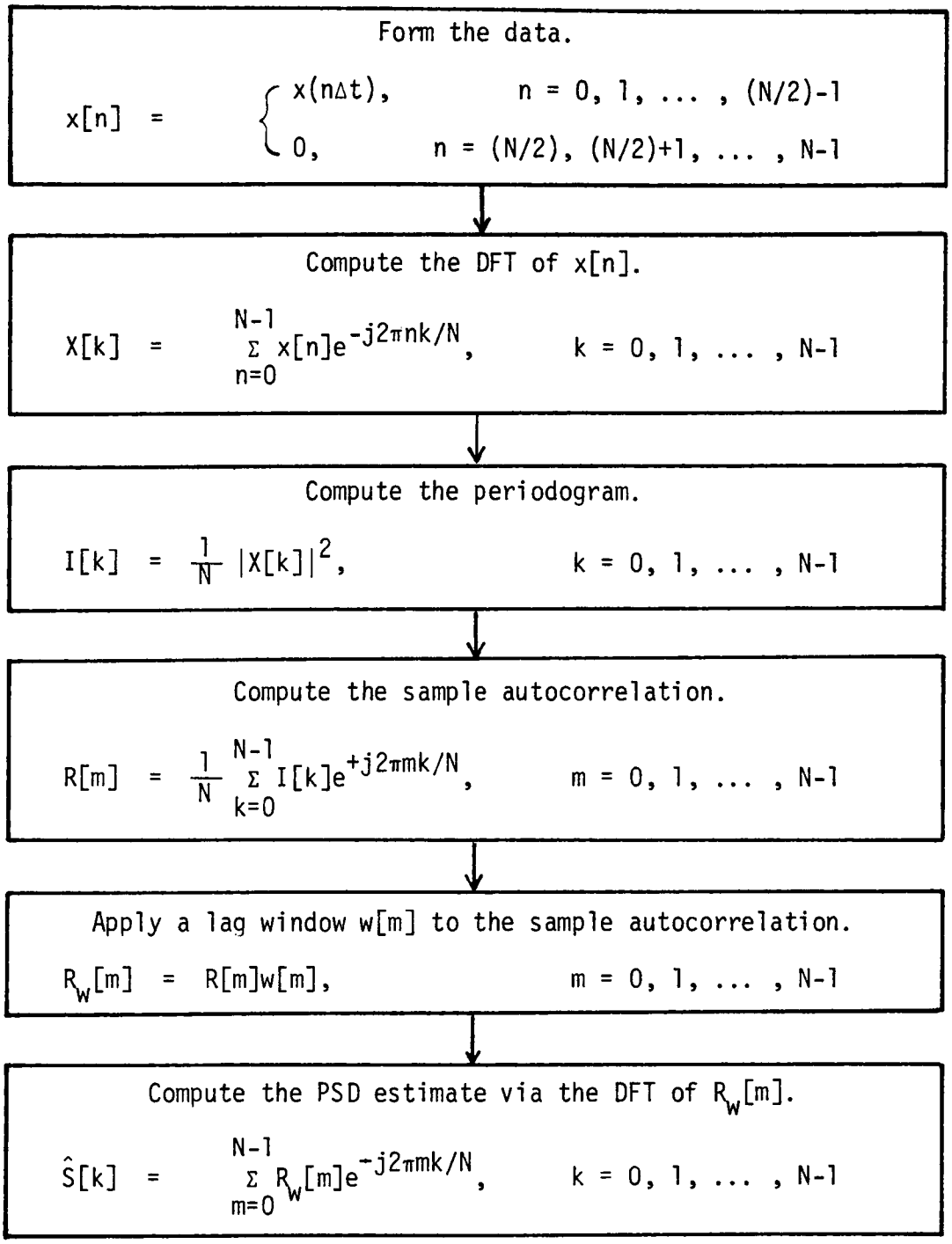


Figure 3.1 Quadratic Windowing Power Spectrum Estimation

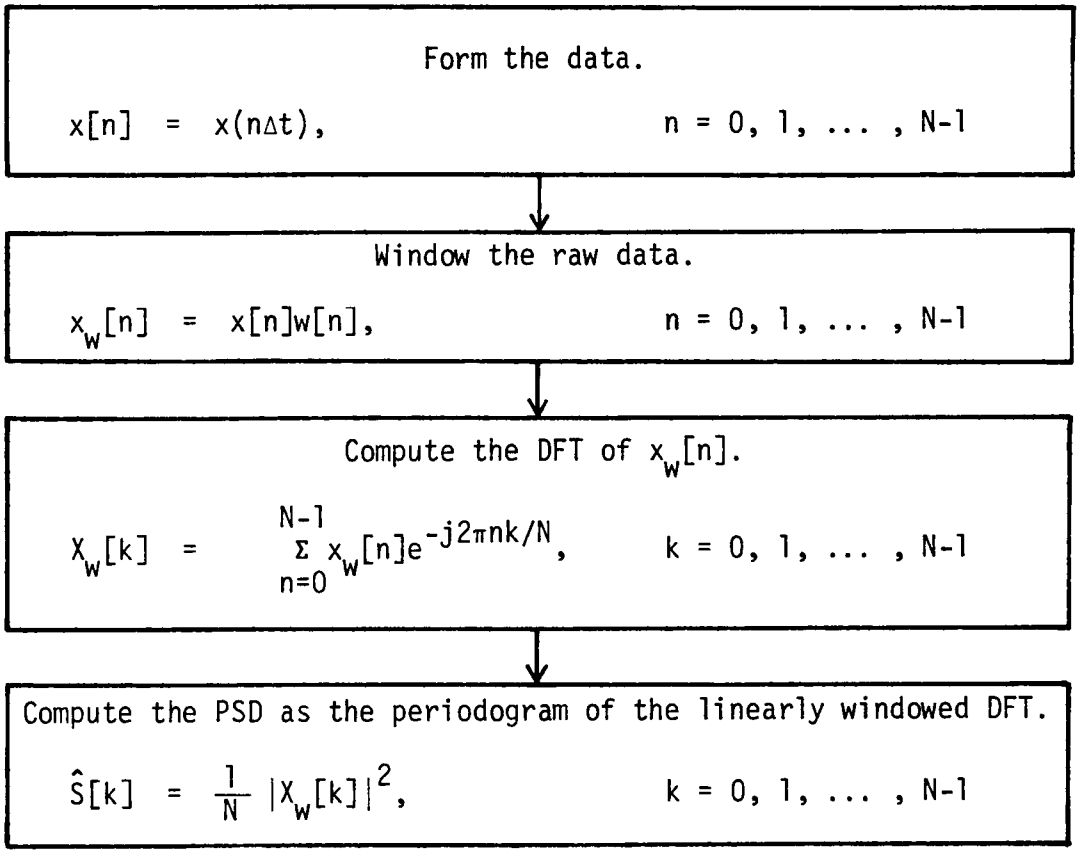


Figure 3.2 Linear Windowing Power Spectrum Estimation

windowing, but linear windowing does not reduce the variance (Yuen, 1979, 86), (Beauchamp, 1979, 157), (Slone, 1967). The second purpose of windowing is to reduce spectral leakage which results from the sudden truncation of the time series data to zero. The mechanics of this phenomenon are very clearly presented in (Brigham, 1974, 140) and will not be presented here. It is important however to note that both quadratic windowing and linear windowing reduce spectral leakage (Beauchamp, 1979, 158). Due to the desirable variance reduction feature of quadratic windowing, this study does not address linear windowing. Henceforth the term "windowing" shall only refer to quadratic windowing unless otherwise stated.

It should be noted that even when no window is explicitly applied, a rectangular shaped window of length N is implicitly being used. To simultaneously decrease the variance and leakage it is necessary to use a lag window which is nonzero over an interval shorter than N and which smoothly decreases to zero over its nonzero values. Many such windows have been proposed and a good discussion of several types of windows can be found in (Harris, 1978).

The resolution of PSD estimates is affected by the application of a lag window. In general, the variance and resolution are related by

$$(\text{variance})\cdot(\text{resolution}) = \text{constant} \qquad (3.1.9)$$

(Priestley, 1981, 525). The reason for this can be seen by considering the windowing operation from a frequency domain point of view. The DFT of the product of two time domain waveforms is equal to the frequency domain convolution of the DFT of each time domain waveform

$$DFT\{R[m]w[m]\} = \frac{1}{N} \sum_{i=0}^{N-1} I[i]W[k-i] \qquad (3.1.10)$$

(Stanley, 1984, 260), where the frequency domain window W[k] is the DFT of w[m]. This convolution operation smears the resolution of the PSD estimates according to the bandwidth of W[k]. It is well known that the bandwidth of a frequency domain function such as W[k] is roughly inversely proportional to the time duration of its time domain representation w[m] (Kay, 1981). This feature will be used advantageously in formulating the frequency domain windowing approach in LURF PSD estimation.

3.2 Logarithmic Windowing

Convolving the periodogram with a frequency domain window averages adjacent analysis frequency points which simultaneously reduces frequency resolution and estimation error variance. Thus for harmonic analysis, the PSD estimates are given by equation (3.1.10)

$$\hat{S}[k] = \frac{1}{N} \sum_{i=0}^{N-1} I[i]W[k-i] \qquad (3.2.1)$$

for k = 0, 1, ... , N-1.

To compute the LURF power spectrum using frequency domain windowing of the periodogram, it is necessary to generalize the window used in equation (3.2.1) and evaluate the convolution only at the LURF analysis frequencies indicated by equation (2.1.6)

$$\hat{S}(\omega_k') = \sum_{i=0}^{N-1} I[i]W[i,\omega_k']. \qquad (3.2.2)$$

The ω_k' in equation (3.2.2) are the center frequencies of each of the M LURF bands for k = 1, 2, ... , M according to

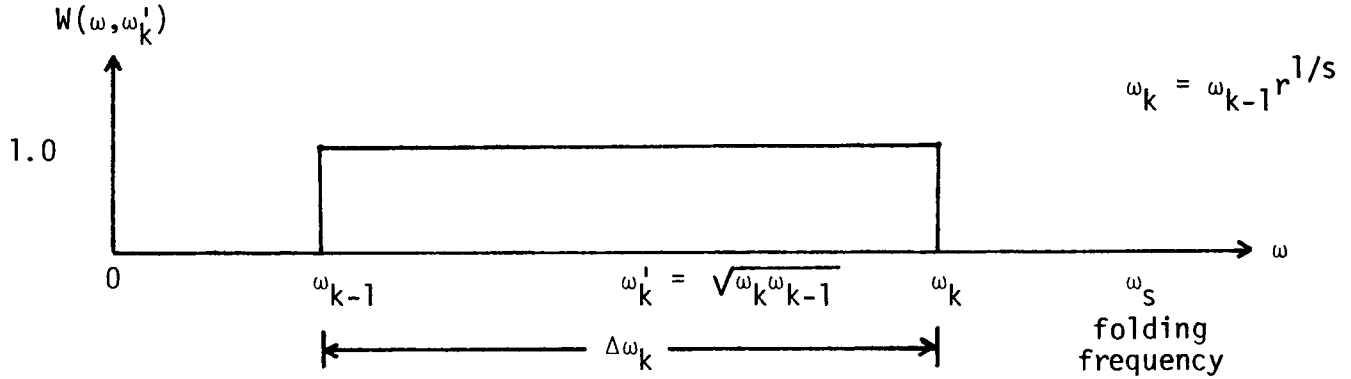
$$\omega_k' = \omega_{k-1}' r^{1/s}. \qquad (3.2.3)$$

The dependence of $W[i,\omega_k']$ on ω_k' is that the bandwidth of $W[i,\omega_k']$ is proportional to ω_k' according to

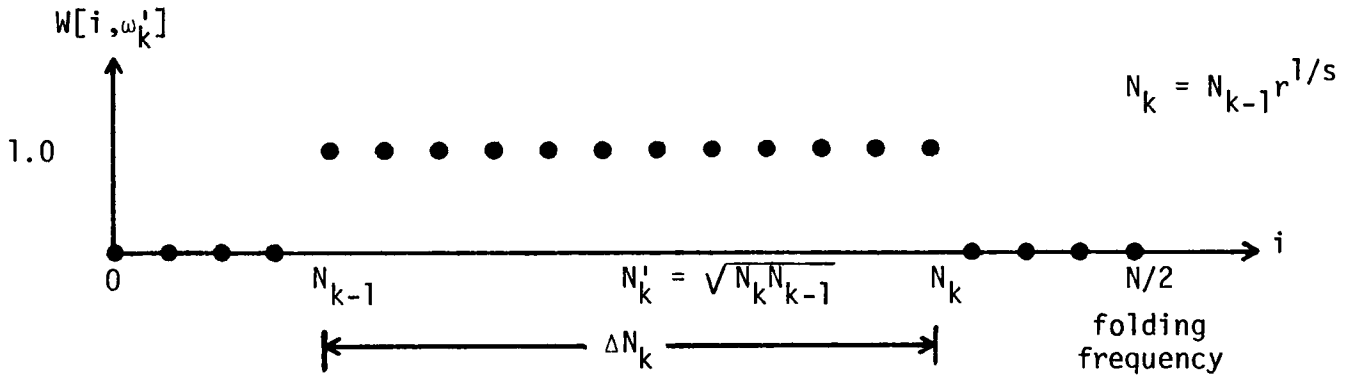
$$\Delta\omega_k = \omega_k' \left(\sqrt{r^{1/s}} - \sqrt{r^{-1/s}} \right) . \qquad (3.2.4)$$

One very simple frequency domain window is the Daniell spectral window (Priestly, 1981, 441) as shown in Figure 3.3. Figure 3.3a shows the continuous Daniell spectral window, and Figure 3.3b shows the discrete Daniell spectral window which is obtained by sampling the continuous window in frequency. The application of this window requires only N/2 real additions beyond the computational requirements of the periodogram.

In general, by using LURF frequency domain windows which are non-zero only within the approximate bandwidth implied by equation (3.2.4), the number of terms required in the convolution sum of equation (3.2.2) is reduced from N for each analysis frequency to only the number of terms encompassing the bandwidth. Thus the LURF PSD convolution sum can be expressed by



(a) Continuous



(b) Discrete

Figure 3.3 Daniell Frequency Domain Window

$$\hat{S}[k] = \sum_{i=N_{k-1}}^{N_k} I[i]W[i,\omega_k'] \qquad (3.2.5)$$

for k = 0, 1, ... , M where $\hat{S}[k]$ are the LURF PSD estimates at the frequencies given by equation (3.2.3)

$$\hat{S}[k] = \hat{S}(\omega_k'). \qquad (3.2.6)$$

N_k and N_{k-1} are related by

$$N_k = N_{k-1} r^{1/s}. \qquad (3.2.7)$$

Equation (3.2.7) will yield values of N_k which in general are not integers. To retain definition of the LURF bandwidth, it is desirable to taper a few of the window samples at the bandedges. Thus, the N_k for k = 0, 1, ... , M do not necessarily satisfy equation (3.2.7) exactly and the window samples around the bandedge taper smoothly from 1.0 to 0.0. If the number of taper window weights is small compared to the number of window weights equal to unity, the computational load of the modified Daniell window is only slightly higher than the regular Daniell window due to its slightly enlarged nonzero width and nonunity terms.

For the case where $N = 2^L$ (and L is an integer), each DFT or IDFT requires approximately $8N\log_2 N$ real multiplications and additions. For a real random process, the raw time series data are real and some computational savings in the FFT algorithm can be realized by exploiting this (Cooley, 1970a, 1970b). Also, by "pruning" the FFT algorithm,

more computational savings can be realized (Markel, 1971) due to the presence of zero data points which have been appended to the input raw time series data. Without attempting to obtain these savings or those which accrue from trival multiplications and symmetry in the radix-2 algorithm, the $8N\log_2 N$ real multiplications and additions are approximate upper bounds for the number of required arithmetic operations in the radix-2 FFT algorithm (Cochran, 1967). Since the input data sequence is real, the transform X[k] will be symmetrical about the point k = N/2 or X[N-k] = X[k]. Therefore, the periodogram must only be computed for ~N/2 values. For the LURF estimates, the LURF frequency domain window only must cover the periodogram from i = 0 to i = N/2. Hence, conventional harmonic spectrum estimation as shown in Figure 3.1 requires approximately $(24N\log_2 N)+5N/2$ real multiplications and $(24N\log_2 N)+N/2$ real additions. The sequence of operations for LURF PSD frequency domain window estimates is shown in Figure 3.4. Since the LURF estimation scheme eliminates two transform operations and has a different windowing computation, its computational burden amounts to approximately $(8N\log_2 N)+2N$ real multiplications and $(8N\log_2 N)+N$ real additions. Table 3.1 lists for a few values of N these approximate upper computational bounds for the conventional harmonic PSD estimation scheme and the LURF PSD frequency domain windowed estimation scheme. By listing the upper computational bounds, a better comparison can be made when considering mixed radix FFT algorithms which cannot easily accommodate the possible computational savings mentioned earlier.

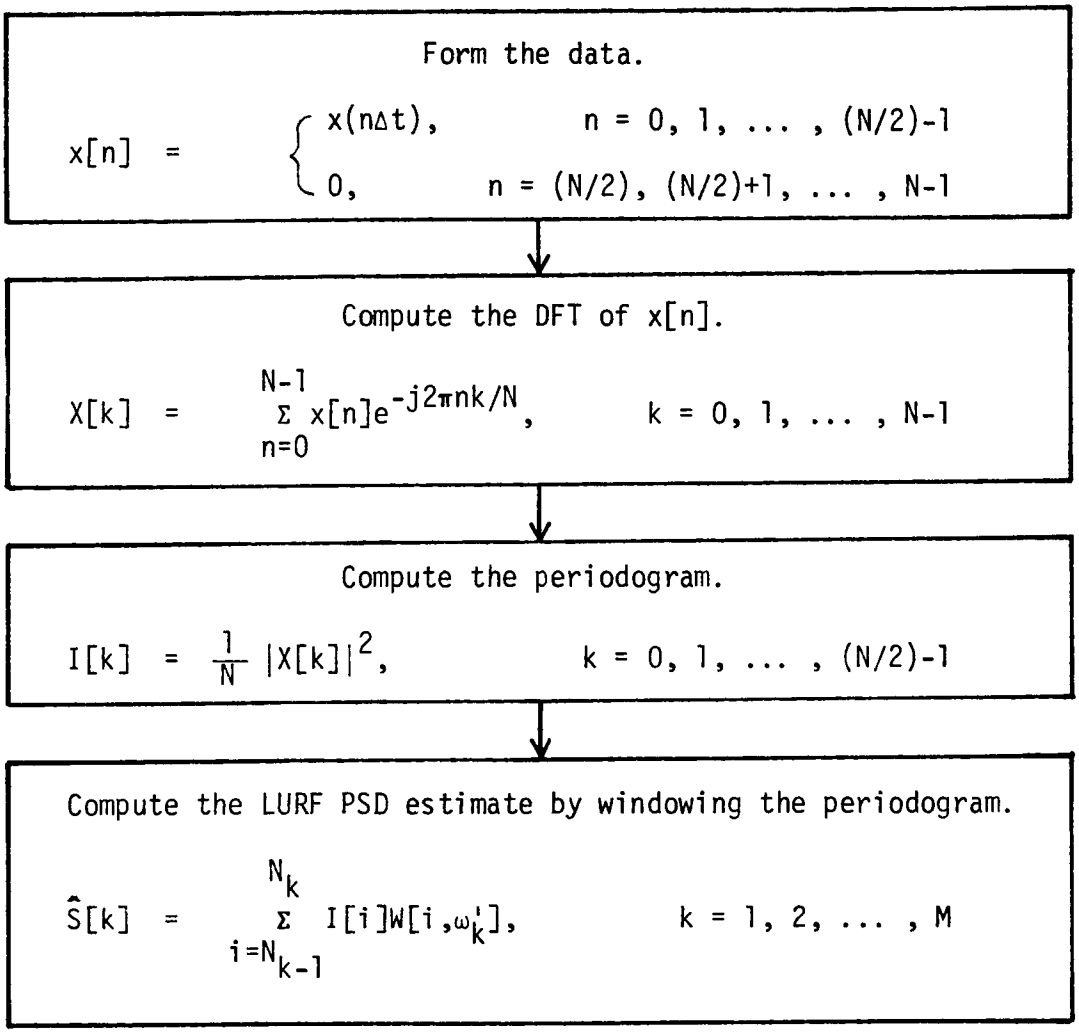


Figure 3.4 Frequency Domain Windowing LURF PSD Estimation

Table 3.1 Approximate Upper Bound Computational Load Comparison

Number of Data Points	Conventional Harmonic Spectrum with Time Domain Window		LURF Spectrum with General Frequency Domain Window	
	Real Multiplies	Real Additions	Real Multiplies	Real Additions
256	111,872	110,848	37,888	37,376
512	248,320	246,272	83,968	82,944
1024	545,792	541,696	184,320	182,272
2048	1,189,888	1,181,696	401,408	397,312
4096	2,576,384	2,560,000	868,104	860,160
8192	5,523,600	5,490,952	1,860,244	1,843,920

A statistical evaluation of the properties of the Daniell frequency domain window has appeared in the literature (Parzen, 1961). General evaluation of windows has recently been very well expounded (Priestley, 1981) and will not be repeated here. For the discrete case, the expected value of the square of the estimation error has been shown to be given approximately by

$$\text{var}\{\hat{S}[k]\} \equiv E\{(\hat{S}[k]-S[k])^2\} \simeq \frac{1}{\tilde{N}} S^2[k] \qquad (3.2.8)$$

where $\tilde{N}$ is the number of nonzero window weights of the Daniell frequency domain window, and $S[k]$ is the true PSD of the random process.

For the LURF case, the variance becomes

$$\text{var}\{\hat{S}[k]\} \simeq \frac{1}{N_k - N_{k-1} + 1} S^2[k]. \qquad (3.2.9)$$

Thus the estimation error variance for the frequency domain window approach to LURF PSD estimation is much like the LURF filter bank approach result and is consistent with the result for the periodogram's variance, viz. when $N_k = N_{k-1}$. Thus Figure 2.8 which shows the bounds of the log normalized variance apply in a qualitative fashion to the frequency domain windowing approach also.

3.3 Other Fourier Transform Methods

Various schemes have been proposed in the literature which utilize the DFT to obtain a LURF power spectrum. The hybrid method of (Harris, 1976) which uses modulation, lowpass filtering and frequency domain

windowing has already been discussed. Other techniques which have been proposed have mostly concentrated on computational efficiency at the expense of quality and flexibility.

There have been two DFT methods which provide $\log_2 N$ spectrum estimates from N time series data points. Both approaches implement the DFT in such a way that all the complex multiplications of the DFT become trivial. This involves computation of the DFT sum for frequencies where the exponential factor $e^{-j2\pi nk/N}$ becomes +1, -1, +j and -j in

$$Y[k] = \sum_{n=0}^{N-1} w[n]x[n]e^{-j2\pi nk/N}. \qquad (3.3.1)$$

Equation (3.3.1) indicates that linear windowing is being used. The trivial evaluation of equation (3.3.1) occurs for the frequency estimate at k = N/4 since

$$Y[N/4] = C_c - jC_e \qquad (3.3.2)$$

where

$$C_c = \sum_{n=0}^{(N/2)-1} (-1)^n w[2n]x[2n] \qquad (3.3.3)$$

and

$$C_e = \sum_{n=0}^{(N/2)-1} (-1)^n w[2n+1]x[2n+1]. \qquad (3.3.4)$$

The first approach (Southeastern Scientific Co., 1978) obtains subsequent estimates at Y[N/8], Y[N/16], ... by averaging adjacent samples of x[n] according to

$$x_1[n] = (x[n] + x[n-1])/2, \qquad (3.3.5)$$

deleting every other sample and recursively applying equations (3.3.2), (3.3.3) and (3.3.4). By judiciously choosing the window, these calculations can be performed in fixed-point arithmetic. Unfortunately this approach has a severe aliasing problem (Durand, 1983). The averaging operation in equation (3.3.5) corresponds to a filter which is inadequate to satisfy the Nyquist criterion for the decimation operation which is subsequently performed.

The second approach (Durand, 1983) was formulated to eliminate the aliasing problem of the first approach. Instead of antialias filtering and decimation, lower frequency estimates are obtained by extending the length of the time series with more data. For each analysis frequency, a different linear window is applied and a single analysis frequency is computed. Here, the aliasing problem is solved, but the computational burden is greatly increased by the multiple time domain window multiplications.

This second trivial multiplication DFT scheme is similar to the present author's LURF frequency domain window approach. Important differences are that Durand's approach peforms the windowing operation in the time domain and does so on the raw time series data instead of the sample autocorrelation function. The frequency domain windowing

operation of the present author's proposed scheme could be performed in the time domain for each analysis frequency. This however would be computationally more costly, require large memory storage for the different window functions and would be a more complex processing scheme since the FFT could not be efficiently utilized to compute the DFT for each analysis frequency.

There is one approach (Kates, 1979) which employs a more generalized form of the DFT, the chirp z-transform (CZT), to obtain LURF spectrum estimates. This method consists conceptually of modulating the signal to be analyzed by a complex sinusoid at the analysis frequency ω_0, lowpass filtering to obtain the desired bandwidth, and finding the average power of the filter output. By choosing a special class of lowpass filters, i.e.

$$H(\omega,\omega_0) = \frac{1}{\omega_0} \frac{1}{(\alpha+j\omega/\omega_0)^p} \qquad (3.3.6)$$

the CZT can be employed. The 3db bandwidth of the filter is given by

$$BW = \alpha\omega_0(2^{1/p} - 1)^{1/2} \qquad (3.3.7)$$

where α is a free parameter, p is the number of filter poles, and ω_0 is the analysis frequency. Since the bandwidth is proportional to the analysis frequency, this approach is seen to correspond to a constant-Q filter bank. It has been discovered (Kates, 1983) that this approach has a severe dynamic computation range limitation. Practical limits are N = 256 and p = 2 or 3 when using double precision (about 17

decimal digits) floating point arithmetic. This, of course, renders this approach virtually useless.

Another approach which has been compared to LURF spectrum estimation is called the spectrum prewarping method (Oppenheim, 1971, 1972), (Bell, 1978). The spectrum prewarping method does not give true LURF PSD estimates but provides an orthogonal transformation of the data through a set of basis functions. This is accomplished by passing the data through a sequence of allpass filters. Without affecting the magnitude of the frequency content of the signal, the frequency scale is nonlinearly warped for the raw data's DFT according to

$$\omega = \theta(\Omega) = \arctan \left\{ \frac{(1-a^2)\sin \Omega}{(1+a^2)\cos \Omega - 2a} \right\}. \qquad (3.3.8)$$

The parameter "a" provides some control over the amount and nature of warping. The error with which the spectrum prewarping approach approximates a true logarithmically uniform warping has been described (Braccini, 1974).

Actually, the spectrum prewarping approach was not developed for the purpose of estimating the PSD of a random process. It was developed to render divers' speech intelligible (Oppenheim, 1972) and is a transformation of the discrete signal from N points to N points. LURF PSD estimation, by definition, transforms N points into M points. Hence, LURF PSD estimation does not allow recovery of the original signal since $M < N$.

4. IMPLEMENTATION

4.1 Logarithmic Parameters

Two different sets of time series data were generated to test the implementation of both of the proposed estimation schemes. In both cases, the PSD is found independently of the implemented estimation schemes for comparison purposes. Each set of data consists of 2048 points, 1024 of which are used for analysis. For the LURF estimation, the LURF parameters were chosen to be r = 2 and s = 3. This corresponds to a third-octave band analysis which is often used in industrial acoustical noise analysis (Maling, 1967). The lowest frequency band was selected to have its cutoff frequency at $\omega_1 = 2\pi 50/1024$. By selecting the number of analysis frequencies to be M = 11, the uppermost cutoff frequency becomes $\omega_M \simeq 2\pi 504/1024$. This is close to the folding frequency of $\omega = 2\pi 512/1024$ but does not exceed it. Thus most of the available frequency spectrum is covered.

The FORTRAN program FILSPC, which is listed in the Appendix, was used as an aid in selecting these parameters. There are basically five parameters associated particularly with the LURF estimation formulation. The parameters are as follows:

1. ω_1, the initial frequency band demarcation;
2. ω_M, the terminal frequency band demarcation;
3. M, the number of LURF PSD analysis frequencies;
4. r, the logarithm base;
5. s, the resolution with respect to the logarithmic base.

By choosing any four of the five parameters, the fifth parameter can be calculated as follows:

$$\omega_1 = \omega_M r^{-\frac{M-1}{s}}, \tag{4.1.1}$$

$$\omega_M = \omega_1 r^{+\frac{M-1}{s}}, \tag{4.1.2}$$

$$M = 1 - s\log_r(\omega_1/\omega_M), \tag{4.1.3}$$

$$r = e^{s\frac{\ln(\omega_1/\omega_M)}{1-M}}, \tag{4.1.4}$$

$$s = \frac{1-M}{\log_r(\omega_1/\omega_M)}. \tag{4.1.5}$$

Intermediate frequency band demarcations are calculated according to

$$\omega_k = \omega_1 r^{\frac{k-1}{s}}. \tag{4.1.6}$$

Table 4.1 lists the frequency band demarcations and two-sided bandwidths for the 11 analysis frequencies used in this investigation. Instead of radian frequency, the frequencies are given in units of normalized frequency, $0 \leq \omega \leq 0.5$, where the sampling frequency is taken to be 1. Since the actual physical frequency depends upon the sample rate and the estimation algorithms are invariant with respect to the sample rate, the actual frequency units in a given practical situation can always be incorporated after the estimation computations are completed by scaling.

Table 4.1 LURF Bands for r = 2, s = 3, M = 11 and $\omega_1 = 50/1024$

k	Band Demarcations ω_k	Two-sided Bandwidths $2\Delta\omega_k$
1	0.0488	0.0977
2	0.0615	0.0254
3	0.0775	0.0320
4	0.0977	0.0403
5	0.1230	0.0506
6	0.1550	0.0640
7	0.1953	0.0806
8	0.2461	0.1016
9	0.3100	0.1278
10	0.3906	0.1612
11	0.4922	0.2032

4.2 Experimental Data Generation

The first set of data consisted of 2048 points generated from

$$x_1[n] = 9\cos(2\pi n55/1024) + 15\sin(2\pi n175/1024) + 21\cos(2\pi n300/1024). \quad (4.2.1)$$

The ideal LURF PSD is easily deduced from equation (4.2.1) above and the LURF parameters of Table 4.1. The average power of each sinusoidal term is given by half of its peak magnitude squared. To get the power spectrum density, the average power of each sinusoidal term is divided by the two-sided bandwidth of the frequency band in which each term lies. Thus the actual PSD of this data set using the aforementioned LURF parameters is represented by the histogram shown in Figure 4.1.

The second set of data consisted of 2048 points which were generated by passing Gaussian white noise through a linear, digital filter. It is well known that the output signal's PSD of a linear filter is equal to the product of the filter's input signal PSD and the filter's magnitude squared transfer function, i.e.

$$S_{output}(\omega) = S_{input}(\omega)|H(\omega)|^2. \quad (4.2.2)$$

If the input is white noise, then

$$S_{input}(\omega) = \text{a constant} \quad (4.2.3)$$

and the resulting output PSD is proportional to the magnitude squared transfer function.

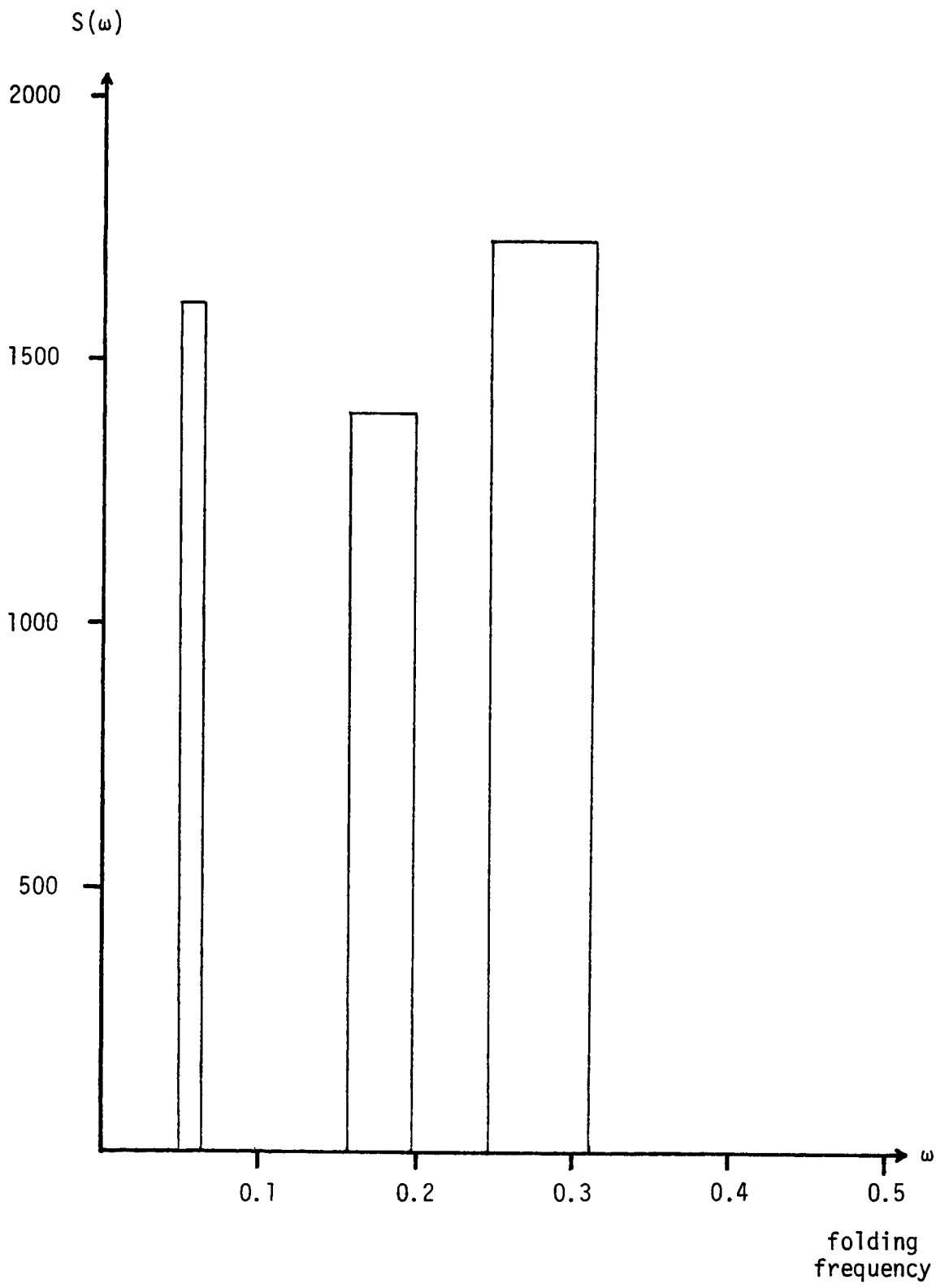


Figure 4.1 Ideal LURF PSD for Sinusoidal Test Data

In this case the white noise was generated using the linear, uniform number generator subroutine RAN on a Digital Equipment Corp. model DEC-10 computer. This routine produces independent random numbers uniformly distributed on [0,1). Using the polar method (Knuth, 1981) the numbers were converted to a set of numbers with a Gaussian distribution having a mean of zero and unit variance.

Next, these random numbers were applied as the input signal to a linear digital filter. The digital filter was designed by specifying its poles and zeros in the z-plane. By placing the poles and zeros close to the unit circle in the z-plane, peaks and valleys in the PSD can be selected in a qualitative manner.

The zeros were placed at $z = 0.95$ and $z = \cos 50^{o} \pm j\sin 50^{o}$. The poles were placed at $z = 0.9(\cos 100^{o} \pm j\sin 100^{o})$ and at $z = 0.9(\cos 20^{o} \pm j\sin 20^{o})$. These pole-zero z-plane locations are shown in Figure 4.2. From this information, the z-transfer function is given by

$$H(z) = \frac{b_1 z^3 + b_2 z^2 + b_3 z + b_4}{z^4 + a_1 z^3 + a_2 z^2 + a_3 z + a_4} \qquad (4.2.4)$$

where $b_1 = 1.0$, $b_2 = -2.23557522$, $b_3 = +2.221296459$, $b_4 = -0.95$, $a_1 = -1.378879998$, $a_2 = +1.09131005$, $a_3 = -1.116892798$ and $a_4 = +0.6561$. The filter was implemented with a direct form I structure (Ziemer, 1983) using double precision arithmetic to avoid any implementation effects on the generated time series data. Hence the following difference equation was used to generate the data for LURF estimation

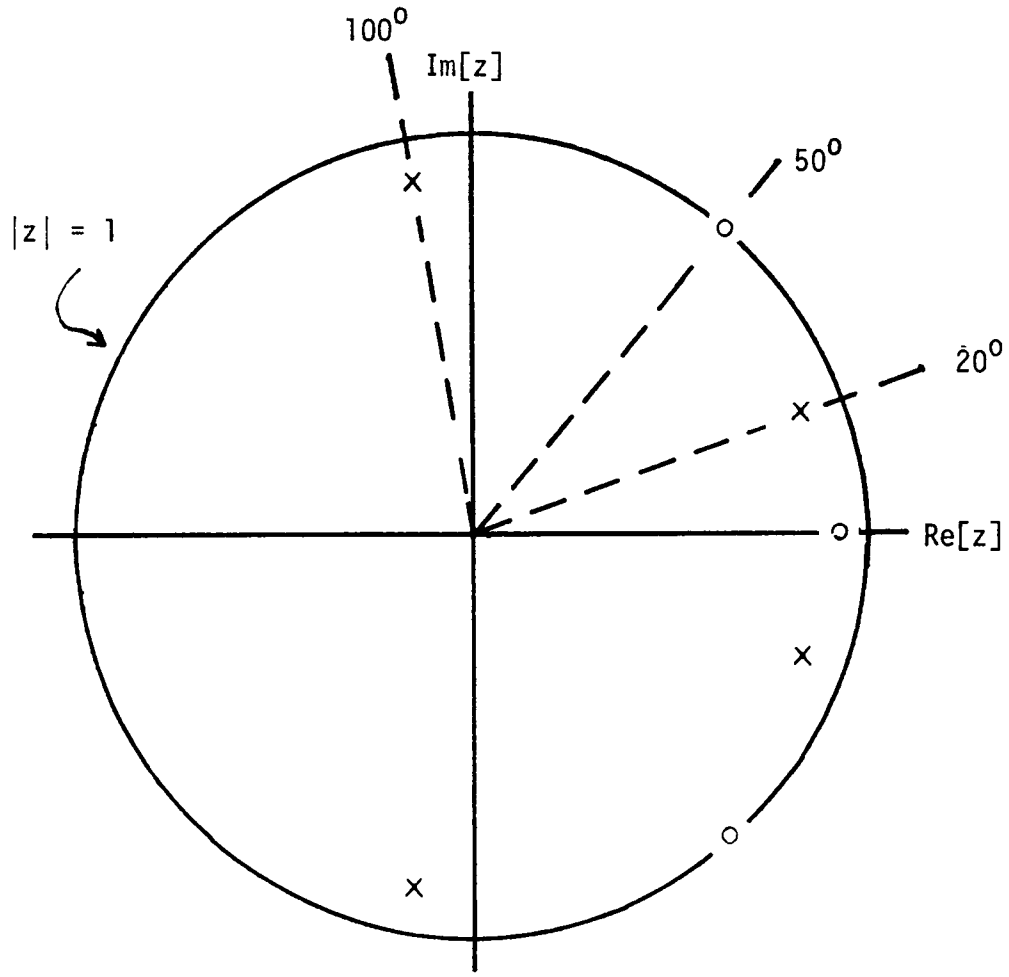


Figure 4.2 Pole-Zero Locations of Stochastic Test Data

$$x_2[k] = -a_1x_2[k-1] - a_2x_2[k-2] - a_3x_2[k-3] - a_4x_2[k-4] + b_1x_0[k-1] + b_2x_0[k-2] + b_3x_0[k-3] + b_4x_0[k-4] \quad (4.2.5)$$

where $x_2[k]$ is the filter output, and $x_0[k]$ is the filter input at the discrete sampling instant k.

In order to insure that the data represented the steady-state response of the data generation filter and to provide initial conditions for the LURF lowpass filter bank estimation processing, a total of 2048 points were generated even though the lowpass LURF scheme ran for 1024 time samples. Since there were no poles placed near z = +1, the extra initial 1024 points were sufficient to eliminate any bias in the estimates from either the data generation filter warm-up or the lowpass LURF warm-up.

The ideal PSD for this data set can be found by considering the magnitude squared Fourier transform of the z-transform function

$$|H(e^{j\omega})|^2 = |H(z)|^2\Big|_{z=e^{j\omega}} \quad (4.2.6)$$

The PSD curve is shown as a function of ω in Figure 4.3. Assuming that the LURF estimation schemes used behave like ideal bandpass filters in smearing the resolution at each analysis frequency, the LURF PSD values are ideally given by

$$S[k] = \frac{1}{2\pi(\omega_k - \omega_{k-1})} \int_{\omega_{k-1}}^{\omega_k} S(\omega)d\omega. \quad (4.2.7)$$

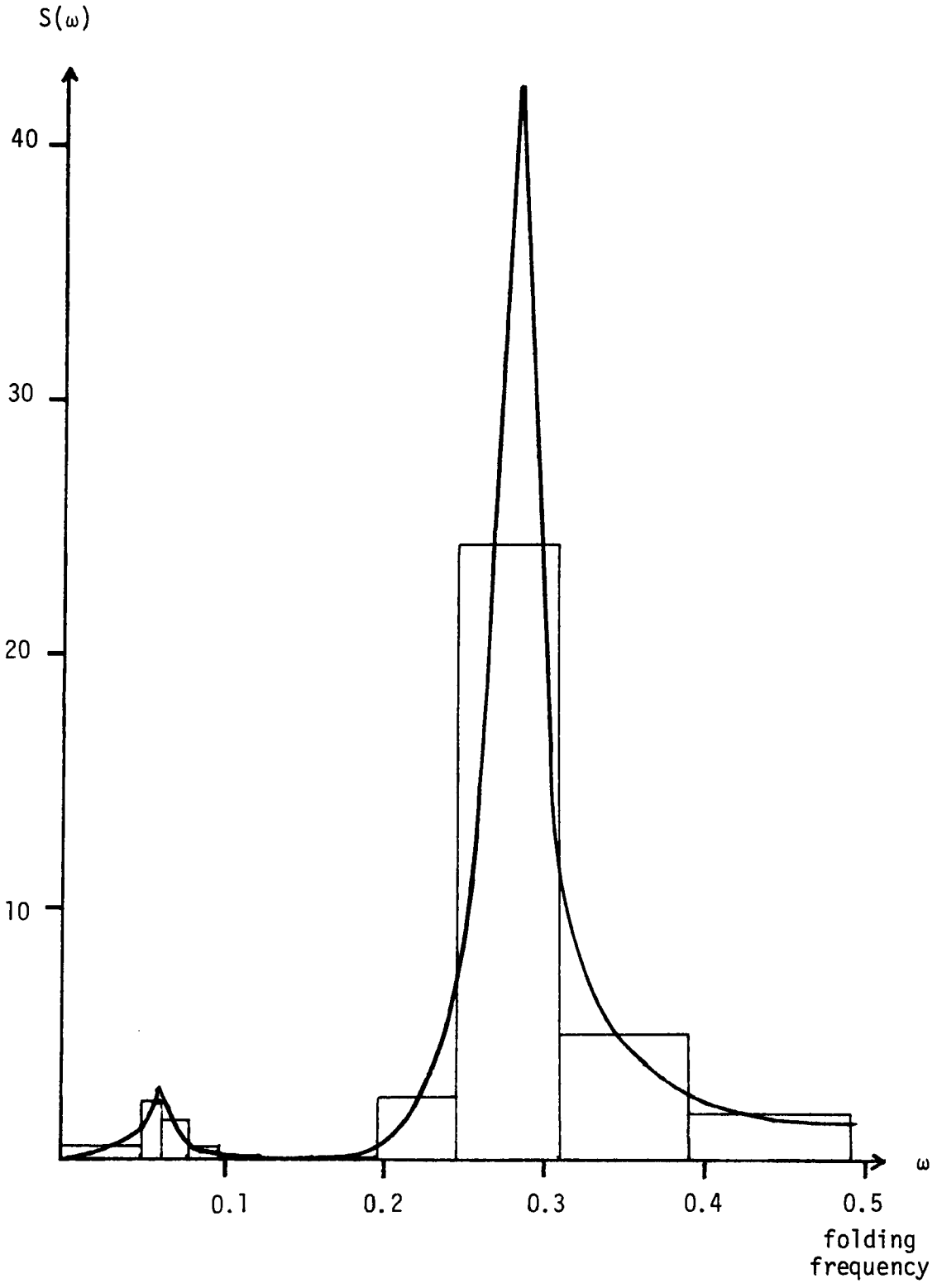


Figure 4.3 Ideal PSD for Stochastic Test Data

Using a numerical approximation to this integration and using normalized frequency units yield the ideal LURF histogram which is superimposed on the smooth PSD in Figure 4.3.

4.3 Lowpass Filter Bank Implementation

To implement a LURF lowpass filter bank, two different approachs were used, viz. the window method (Oppenheim, 1975, 239) and the frequency sampling method (Rabiner, 1975, 105).

The filter bank designed by the windowing method employed the Kaiser-Bessel (Kaiser, 1963) family of windows. These windows approximate prolate spheroidal wavefunctions which are time-limited functions having minimum energy outside some frequency interval (Slepian, 1961), (Landau, 1961).

The FORTRAN program FILDES was used to generate the FIR filter coefficients for the filter bank designed by the window method. A list of the program statements of FILDES is found in the Appendix.

The frequency band demarcations listed in Table 4.1 were used as the cutoff frequencies. The stopband attenuation is a design parameter for window designed FIR filters using Kaiser-Bessel windows (Kaiser, 1974) and was arbitrarily specified to be -50 db. The number of filter coefficients was set at 199. The resulting attenuations at the bandedges were approximately 0.5 for each lowpass filter in the bank. Since each of the eleven filters implemented were very similar in lowpass and equivalent bandpass characteristics, just one typical equivalent bandpass magnitude function is depicted in Figure 4.4 for

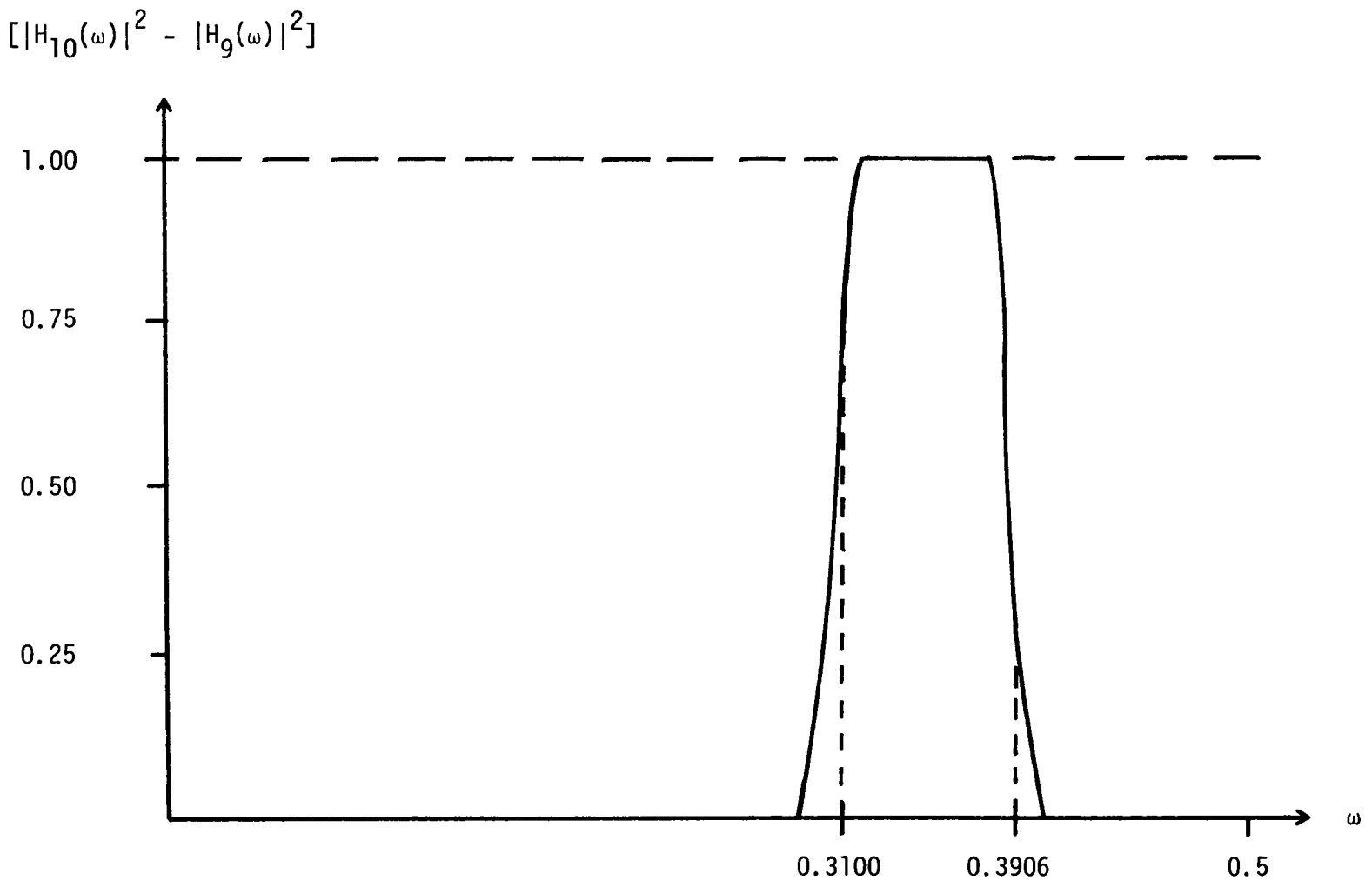


Figure 4.4 Window Designed Typical Equivalent Bandpass Magnitude Function

the window designed implementation. There are some general features of the resulting equivalent bandpass magnitude functions which are peculiar to the practical lowpass LURF filter bank implementation. After the details of the frequency sampling implementation are presented, these general lowpass LURF characteristics will be discussed.

The second filter bank was designed using the frequency sampling method with the help of the FORTRAN program FRESAM which is listed in the Appendix. As is usually done in the frequency sampling design method, the passband filter magnitude samples were set to 1.0 and the stopband filter magnitude samples were set to 0.0. The number of filter coefficients was selected to be 256. The transition band samples were selected as follows.

The program FRESAM reads in the number of desired transition band samples and calculates the slope of a straight line which connects the passband to the stopband. The LURF frequency band demarcations given in Table 4.1 are placed at the center of the transition bands. For example, if one of the frequency samples coincides with a LURF cutoff frequency, then the magnitude of that transition band frequency sample is 0.5. The actual transition band samples which were used are listed in Table 4.2. By implicitly making the bandedge frequency magnitudes be approximately 0.5, the frequency sampling design is comparable to the window designed implementation. The equivalent bandpass magnitude filter function of a typical case is shown in Figure 4.5.

Table 4.2 Transition Band Sample Magnitudes

k	Normalized Cutoff Frequency	Transition Band Frequencies	Transition Band Frequency Sample Values
1	0.0488	0.0469 0.0508	0.750000 0.250000
2	0.0615	0.0586 0.0625	0.874507 0.374507
3	0.0775	0.0742 0.0781	0.921257 0.421257
4	0.0977	0.0977	0.500000
5	0.1230	0.1211 0.1250	0.749013 0.249013
6	0.1550	0.1523 0.1563	0.842513 0.342513
7	0.1953	0.1953	0.500000
8	0.2461	0.2422 0.2461	0.998026 0.498026
9	0.3100	0.3086 0.3125	0.685026 0.185026
10	0.3906	0.3906	0.500000
11	0.4922	0.4883 0.4922	0.996053 0.496053

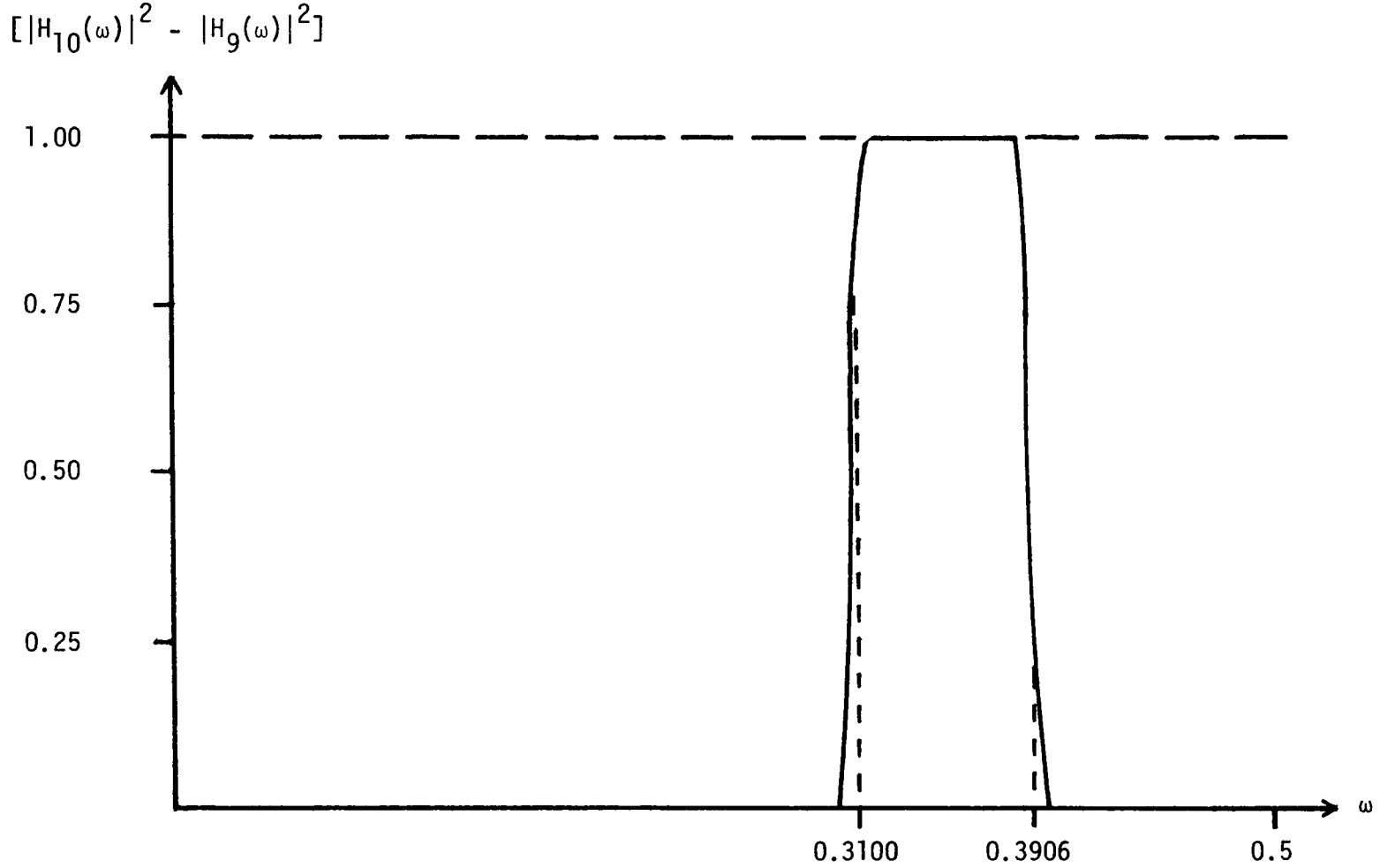


Figure 4.5 Frequency Sample Designed Typical Equivalent Bandpass Magnitude Function

For both the window designed and frequency sample designed lowpass LURF filter bank, the magnitude of each lowpass filter is approximately 0.5 at the cutoff frequency. This means that the power magnitude is 0.25 at the bandedges of each lowpass filter. At the upper bandedge of the equivalent bandpass function (cf. equation (2.2.1)) where the effective bandpass magnitude is given by ~ 0.25 - 0.0 = 0.25, the magnitude is down more than at the lower bandedge ~ 1.0 - 0.25 = 0.75. Thus the equivalent bandpass magnitude functions for the lowpass bank are not symmetrical at the bandedges. If the bandedge magnitudes could be set at a magnitude of $1/\sqrt{2}$, the power magnitude at the lowpass bandedges would be 0.5, and at the bandedge frequencies of the equivalent bandpass there would be symmetry. However, this symmetry would only exist at the bandedge frequency and not over the whole transition band due to the nonlinear way two lowpass filters are used to produce a bandpass effect.

Indeed, even in the stopbands, there is no symmetry if both filters which are combined to make an equivalent bandpass have similar stopband attenuation. Again, equation (2.2.1) provides the reason for this. If the amount of ripple ε of any single lowpass filter in the bank is equal in the stopband and passband, the worst case lower stopband magnitude is given by

$$(1 + \varepsilon)^2 - (1 - \varepsilon)^2 = 4\varepsilon \qquad (4.3.1)$$

The worst case upper stopband ripple is given by

$$(\varepsilon)^2 - (0)^2 = \varepsilon^2. \qquad (4.3.2)$$

Since the ripple is usually designed to be as small as possible, i.e. $\varepsilon << 1.0$, it follows that the upper stopband worst case magnitude will be less than the lower stopband worst case magnitude because

$$\varepsilon^2 < 4\varepsilon. \qquad (4.3.3)$$

In both FIR implementations, the general level of upper stopband attenuation was greater than the general level of lower stopband attenuation.

The LURF lowpass filter bank operation depicted in Figure 2.7 was implemented by the FORTRAN program LURFLP which appears in the appendix. The program reads in twice the number of time series data points which the filter calculates over. As stated earlier, the extra initial points read in provide the initial conditions for the filter which are needed to avoid biasing the estimates. For the data which were generated for this study, 2048 points are read in and the filter operated for 1024 time samples. Results for the filter method and frequency domain window methods will be presented later.

4.4 Frequency Domain Window Implementation

The DFT approach proposed in this investigation requires the computation of the periodogram of the time series data and then the application of a family of frequency domain windows to provide the LURF PSD estimates. Thus the LURF implementation requires the selection of a family of frequency domain windows.

The window family chosen to implement this approach is similar to the set of frequency magnitude functions which were used to design the frequency sampling FIR filters. The windows which were used were modified Daniell windows. The modification was a linear taper which was applied to both ends of each window in the family. The transition window weights from the unity magnitude window weights to the zero magnitude window weights were calculated for both ends of each window using the same scheme which was used to generate the transition band frequency samples in the FIR frequency sampling lowpass design. The FORTRAN program DFTFDW was used to implement the DFT frequency domain windowing scheme to obtain the LURF PSD estimates. A program statement listing of DFTFDW is found in the Appendix. Since only 2 transition samples were specified at each bandedge and a 2048 point DFT was computed, the characteristics of the modified Daniell used was almost identical to that of a regular Daniell window family.

4.5 Implementation Results

The LURF PSD estimates for the data which consists of pure sinusoids are listed in Table 4.3 for all the implementations used. The estimates for the data generated by passing white noise through a linear filter are listed in Table 4.4.

The reason for the occurance of negative estimates for some of the points in the lowpass filter bank approaches is that the differencing operation can create a negative magnitude response in the stopbands of the equivalent bandpass representation. Thus the negative estimates

Table 4.3 LURF PSD Results for Sinusoidal Data

k	Ideal LURF PSD	Window Design Lowpass FIR Estimates	Frequency Sample Design Lowpass FIR Estimates	Frequency Domain DFT Window Estimates
1	0.00	2.06	0.29	3.41
2	1595.74	1588.45	1593.78	1601.30
3	0.00	-1.31	-1.63	7.60
4	0.00	0.59	2.52	2.57
5	0.00	-0.19	-0.63	2.35
6	0.00	-0.20	-0.03	5.94
7	1396.13	1397.35	1392.06	1388.87
8	0.00	-0.04	3.49	0.65
9	1723.73	1727.88	1723.46	1722.40
10	0.00	-3.95	-0.04	5.06
11	0.00	-0.37	0.01	1.04

Table 4.4 LURF PSD Results for Stochastic Data

k	Ideal LURF PSD	Window Design Lowpass FIR Estimates	Frequency Sample Design Lowpass FIR Estimates	Frequency Domain DFT Window Estimates
1	0.484	0.464	0.427	0.495
2	2.25	2.26	2.29	2.56
3	1.51	1.68	1.73	1.37
4	0.501	0.539	0.489	0.554
5	0.108	0.120	0.107	0.108
6	0.0083	0.0067	0.0050	0.0110
7	0.172	0.184	0.206	0.201
8	2.16	1.94	2.04	2.47
9	24.1	26.4	26.6	24.5
10	4.86	5.70	5.25	5.23
11	1.76	1.79	1.80	1.68

must be due to power outside the passbands. Indeed, as in any practical realization of a filter, there is always finite attenuation in the stopband. In terms of DFT approaches, this represents the spectrum leakage from one analysis frequency band into another.

The presence of positive nonzero estimates at analysis frequencies which are zero in the ideal case are also due to finite stopband attenuation in the LURF lowpass filter bank results and leakage in the LURF DFT results. These nonideal and undesirable leakage characteristics are not peculiar to the LURF estimation procedures but are features which are associated with general spectrum estimation techniques in practical implementations.

It is a well known result that a function cannot be both time-limited and band-limited (Papoulis, 1977, 184). Since the filters employed were of FIR type, it is immediately obvious that the stopband attenuation was finite. The spectral leakage which occurs in the LURF DFT frequency domain window method is a phenomenon which has been investigated by many authors (Priestley, 1981), (Harris, 1978), (Blackman, 1958) for general harmonic analysis and will not be repeated here.

These particular implementations of the LURF lowpass filter bank approach and the LURF DFT frequency domain windowing approach were arbitrarily selected for demonstration purposes. The implementations presented do not represent all the possible realizations of the two major LURF PSD estimation procedures set forth in this study. There are many different ways to design an FIR filter. Indeed there are no

impediments to employing IIR filters using the LURF lowpass filter bank approach. There are a multitude of window functions which have been studied and could be employed with relative ease using the LURF DFT frequency domain window approach. Indeed, the implementation results presented in Table 4.3 and 4.4 establish the feasibility of the two major LURF estimation procedures proposed by this study, but they do not imply a restriction as to implementation methods.

5. CONCLUSIONS

Two general methods for LURF PSD estimation have been presented. Their ideal statistical properties and their computational requirements have been derived and compared with conventional approaches. Other LURF approaches which have appeared in the literature have been presented, discussed and compared to the proposed approaches. Two sets of experimental test data having different characteristics have been analyzed using specific implementations of the proposed approaches. The results demonstrate the feasibility of the proposed approaches for general LURF PSD estimation.

The proposed FIR digital lowpass filter bank method has been shown to have a computational advantage over the constant-Q bandpass filter bank. This advantage depends on the LURF parameters r and s and increases as the resolution, described by r and s, increases. The upper and lower bounds on the estimator variance have been derived and indicate that the expected value of the mean square estimation error decreases logarithmically uniform as a function of the analysis frequency. The NVPS approach is compared to the proposed method and derived using a lowpass filter bank interpretation along with the statistical derivation. Although the NVPS method entails a very low computational burden, it has very bad spectral leakage. The proposed general lowpass approach can be implemented to provide almost any arbitrary stopband attenuation. The stopband attenuation obtained depends on the composite attenuation resulting from the differenced lowpass filter bank implementation. Unfortunately, the computational

efficiency of the NVPS approach is lost because the filter impulse response coefficients become nonunity in general and make the convolution multiplications nontrivial in the proposed lowpass approach. Other approaches which use filters, decimation and modulation are presented. These approaches generally require a complex recursive processing scheme for implementation. Although the proposed lowpass scheme requires sequential subtraction to obtain the LURF PSD estimates, only one subtraction per estimate is required. This means that for practical purposes, the proposed lowpass approach can operate in real-time if necessary. Since each small group of estimates in the other recursive schemes require the complete reprocessing of a whole block of data, the real-time implementation of these schemes is not as easy as it is in the proposed lowpass approach.

The proposed frequency domain windowing approach has been shown to have approximately a 3 to 1 computational advantage over conventional harmonic spectrum estimation. The approximate estimator variance for frequency domain windows is well known and has appeared in the literature for harmonic analysis. These results, when applied to the proposed approach, indicate that the expected value of the mean square estimation error decreases logarithmically as a function of the analysis frequency as it did in the case of the filter bank method. The spectral leakage for specific windows is also well known and can be applied in a straightforward manner to the proposed approach. Most approaches for DFT-based LURF PSD estimation have concentrated on computational efficiency as an overwhelming consideration. The

method introduced here incorporates more flexibility and offers the potential for higher quality estimates at increased computational cost.

Both new techniques were implemented and used to estimate a LURF PSD for two sets of artificially generated experimental data. The two sets of data were very different in character, the first being sinusoidal and the second being stochastic in nature. The results of the specific implementations of the methods demonstrate the feasibility of the proposed methods for general LURF PSD estimation.

LIST OF REFERENCES

LIST OF REFERENCES

Altes, R. A., "The Fourier-Mellin transform and mammalian hearing," J. Acoust. Soc. Am., vol. 63, pp. 174-183, Jan. 1978.

Beauchamp, K. G., and C. K. Yuen, Digital Methods for Signal Analysis, London: George Allen and Unwin, 1979.

Bell, D. M., and J. N. Gowdy, "Power spectral estimation via nonlinear frequency warping," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-26, pp. 436-441, Oct. 1978.

Blackman, R. B., and J. W. Tukey, The Measurement of Power Spectra from the Point of View of Communications Engineering, New York: Dover, 1958.

Box, G. E. P., and G. M. Jenkins, Time Series Analysis, Forecasting and Control, San Francisco: Holden-Day, 1976.

Braccini, C., and A. V. Oppenheim, "Unequal bandwidth spectral analysis using digital frequency warping," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-22, pp. 236-244, Aug. 1974.

Brigham, E. O., The Fast Fourier Transform, Englewood Cliffs, N. J.: Prentice-Hall, 1974.

Cadzow, J. A., "Spectral estimation: an overdetermined rational model equation approach," Proc. IEEE, vol. 70, pp. 907-938, Sept. 1982.

Childers, D. G., and A. E. Durling, Digital Filtering and Signal Processing, New York: West, 1975.

Cochran, W. T., et. al., "What is the fast Fourier transform ?," IEEE Trans. Audio, Electroacoust., vol. AU-15, pp. 45-55, Feb. 1967.

Coleman, P. J., "Variations in the interplanetary magnetic field: Mariner 2," J. Geophy. Res., vol. 71, pp. 5509-5531, Dec. 1966.

Cooley, J. W., and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comp., vol. 19, pp. 297-301, Apr. 1965.

Cooley, J. W., P. A. Lewis, and P. D. Welch, "The fast Fourier transform algorithm: programming considerations in the calculation of sine, cosine, and Laplace transforms," J. Sound Vibration, vol. 12, pp. 315-337, July 1970a.

Cooley, J. W., P. A. Lewis, and P. D. Welch, "The application of the fast Fourier transform algorithm to the estimation of spectra and cross-spectra," J. Sound Vibration, vol. 12, pp. 339-352, July 1970b.

DeFelice, L. J., and B. A. Sokol, "A new analysis for membrane noise," Biophysical J., vol. 16, pp. 827-838, July 1976.

Durand, Y., Notes on Spaceflight Spectrum Analysis Implementation with Reference to the ISPM Algorithm, to appear as a NASA report, 1983.

Enochson, L. D., and R. K. Otnes, Programing and Analysis for Digital Time Series Data, Washington, D. C.: United States Navy, 1968.

Gambardella, G., "Properties of short-time spectral analysis performed by the peripheral auditory system," First Int. Congr. Cybernetics, London: Gordon and Breach, 1970.

Harris, F. J., "High-resolution spectral analysis with arbitrary spectral centers and arbitrary spectral resolutions," Comp. Elect. Engr., vol. 3, pp. 171-191, June 1976.

Harris, F. J., "On the use of windows for harmonic analysis with the discrete Fourier transform," Proc. IEEE, vol. 66, pp. 51-83, Jan. 1978.

Helmholtz, H. L. F., On the Sensations of Tone as a Physiological Basis for the Theory of Music, trans. from the 1877 German ed. by H. Margenau, New York: Dover, 1954.

Helstrom, C. W., Probability and Stochastic Processes for Engineers, New York: Macmillan, 1984.

Jeans, J. H., Science and Music, New York: Macmillan, 1937.

Jenkins, G. M., and D. G. Watts, Spectral Analysis and its Applications, San Francisco: Holden-Day, 1968.

Kaiser, J. F., "Design methods for sampled data filters," Proc. First Allerton Conf. Circuit System Theory, pp. 221-236, Nov. 1963.

Kaiser, J. F., "Nonrecursive digital filter design using the I_0-sinh window function," Proc. 1974 IEEE Int. Symp. Circuits, Systems, pp. 20-23, 1974.

Kates, J. M., "Constant-Q analysis using the chirp z-transform," IEEE Int. Conf. Acoust., Speech, Signal Processing, pp. 314-317, Apr. 1979.

Kates, J. M., "An auditory spectral analysis model using the chirp z-transform," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, pp. 148-156, Feb. 1983.

Kay, S. M., and S. L. Marple, "Spectrum analysis - a modern perspective," Proc. IEEE, vol. 69, pp. 1380-1419, Nov. 1981.

Knuth, D. E., The Art of Computer Programing Vol. 2: Seminumerical Algorithms, Reading, Mass.: Addison-Wesley, 1981.

Landau, H., and H. Pollack, "Prolate-spheroidal wave functions, Fourier analysis and uncertainty - II," Bell Sys. Tech. J., vol. 40, pp. 65-84, Jan. 1961.

Lee, Y. W., Statistical Theory of Communication, New York: Wiley, 1960.

Maling, G. C., W. T. Morrey, and W. W. Lang, "Digital determination of third-octave and full-octave spectra of acoustical noise," IEEE Trans. Audio, Electroacoust., vol. AU-15, pp. 98-104, June 1967.

Markel, J. D., "FFT pruning," IEEE Trans. Audio, Electroacoust., vol. AU-19, pp. 305-311, Dec. 1971.

Moller, A. R., "Frequency selectivity of single auditory-nerve fibers," J. Acoust. Soc. Am., vol. 43, pp. 1120-1128, July, 1968.

Negron, C. D., "Digital one-third octave spectral analysis," J. Assoc. Comp. Mach., vol. 13, pp. 605-614, Oct. 1966.

Oppenheim, A. V., D. H. Johnson, and K. Steiglitz, "Computation of spectra with unequal resolution using the fast Fourier transform," Proc. IEEE, vol. 59, pp. 299-301, Feb. 1971.

Oppenheim, A. V., and D. H. Johnson, "Discrete representation of signals," Proc. IEEE, vol. 60, pp. 681-691, June 1972.

Oppenheim, A. V., and R. W. Schafer, Digital Signal Processing, Englewood Cliffs, N. J.: Prentce-Hall, 1975.

Otnes, R. K., and L. D. Enochson, "An algorithm for digital one-third octave analysis," Sound and Vibration, vol. 2, pp. 17-20, Apr. 1968.

Owens, A. J., "The nested variance power spectrum," J. Geophy. Res., vol. 82, pp. 3315-3318, Aug. 1977.

Papoulis, A., Signal Analysis, New York: McGraw-Hill, 1977.

Parzen, E., "Mathematical considerations in the estimation of spectra," Technometrics, vol. 3, pp. 167-190, May 1961.

Peebles, P. Z., Communication System Principles, Reading, Mass.: Addison-Wesley, 1976.

Pfeiffer, R. R., et. al., "Cochlear nerve filter responses: distribution along the cochlear partition," J. Acoust. Soc. Am., Vol. 58, pp. 867-869, June 1975.

Priestley, M. B., Spectral Analysis and Time Series, New York: Academic Press, 1981.

Rabiner, L. R., and B. Gold, Theory and Application of Digital Signal Processing, Englewood Cliffs, N. J.: Prentice-Hall, 1975.

Rabiner, L. R., and R. W. Schafer, Digital Processing of Speech Signals, Englewood Cliffs, N. J.: Prentice-Hall, 1978.

Rice, S. O., "Filtered thermal noise-fluctuation of energy as a function of interval length," J. Acoust. Soc. Am., vol. 14, pp. 216-227, Apr. 1943.

Rickard, J. J., "The nested variance power spectrum: a second look," J. Geophy. Res., vol. 84, pp 4457-4458, Aug. 1979.

Robinson, E. A., "A historical perspective of spectrum estimation," Proc. IEEE, vol. 70, pp. 885-906, Sept. 1982.

Schafer, R. W., L. R. Rabiner, and O. Herrmann, "FIR digital filter banks for speech analysis," Bell Sys. Tech. J., vol. 54, pp. 531-544, Mar. 1975.

Schafer, T. H., et. al., "The frequency selectivity of the ear as determined by masking experiments," J. Acoust. Soc. Am., vol. 52, pp. 490-496, Apr. 1950.

Schuster, A., "On the investigation of hidden periodicities with application to a supposed 26-day period of meteorological phenomena," Terr. Mag. Atmos. Elect., vol. 3, pp. 13-41, Mar. 1898.

Slepian, D., and H. Pollack, "Prolate-spheroidal wave functions, Fourier analysis and uncertainty - I," Bell Sys. Tech. J., vol. 40, pp. 43-63, Jan. 1961.

Slone, E. A., "Comparison of linearly and quadratically modified spectral estimates of Gaussian signals," IEEE Trans. Audio, Electroacoust., vol. AU-17, pp. 133-137, June 1969.

Southeastern Scientific Co., Inc., Software Implementation of Data-Processing Techniques for the Solar Polar Mission, P. O. Box 8721, University Station, Knoxville, Tenn. 37916, 1978.

Spitzer, J. W., A Microprocessor Based Control for a Spacecraft Experiment: A Structured Approach, M. S. Thesis, University of Tenn., Aug. 1981.

Stanley, W. D., G. R. Dougherty, and R. Dougherty, Digital Signal Processing, 2nd Ed., Reston, Vir.: Reston, 1984.

Stranberg, M. W. P., "Use of octave filters in the measurement of random noise spectral density," Biophysical J., vol. 18, pp. 132-134, Apr. 1977.

Teaney, D. T., V. L. Moruzzi, and F. C. Mintzer, "The tempered Fourier transform," J. Acoust. Soc. Am., vol. 67, pp. 2063-2067, June 1980.

University of Tennessee, Nested Variance Signal Processing, Dept. of Elect. Engr., Knoxville, Tenn., 37996-2100, 1982.

Weaver, C. S., et. al., "Digital filtering and applications to electrocardiogram processing," IEEE Trans. Audio, Electroacoust., vol. AU-16, pp. 350-391, Sept. 1968.

Wiener, N., The Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications, New York: Wiley, 1949.

Yuen, C. K., and D. Fraser, Digital Spectral Analysis, London: Pittman, 1979.

Ziemer, R.E., W. H. Tranter, and D. R. Fannin, Signals and Systems, New York: Macmillan, 1983.

APPENDIX

```
C=====================================================================
C MAIN PROGRAM:  FILSPC COMPUTES THE LURF BANDEDGE FREQUENCIES GIVEN
C                4 OF THE LURF PARAMETERS   THE 5TH PARAMETER IS
C                CALCULATED FROM THE FOUR WHICH ARE ENTERED.
C                FILTER PARAMETERS ARE ALSO READ IN AND WRITTEN OUT
C                TO A FILE WITH THE LURF BANDEDGE FREQUENCIES
C
C AUTHOR:        CHARLES L. CARNAL
C                UNIVERSITY OF TENNESSEE AT KNOXVILLE
C
C DATE           MAY 1, 1984
C
C INPUT:         N  THE NUMBER OF DATA POINTS
C                NVAR  A NUMERIC DESIGNATION OF THE LURF PARAMETER TO BE
C                      CALCULATED
C                (4 OF THE FOLLOWING 5 VARIABLES)
C                   NF1, THE INITIAL FREQUENCY
C                   NFM, THE TERMINAL FREQUENCY
C                      (NOTE: THE UNITS OF FREQUENCY ARE THAT THE FOLDING
C                             FREQUENCY IS EQUAL TO N/2)
C                   M  THE NUMBER OF ANALYSIS FREQUENCIES
C                   S, THE LOGARITHMIC RESOLUTION
C                   R, THE LOGARITHM BASE
C                NFF1, THE NUMBER OF FILTER TERMS FOR THE FILTER BANK
C                ATT, THE ATTENUATION FOR THE KAISER BESSEL WINDOW BANK
C
C OUTPUT:        M, THE NUMBER OF ANALYSIS FREQUENCIES
C                (THE FOLLOWING ARE OUTPUT FOR I = 1, ... , M)
C                   NFF1, THE NUMBER OF FILTER TERMS
C                   FC, THE LURF BANDEDGE CUTOFF FREQUENCY IN UNITS
C                       OF FREQUENCY WHERE 0.5 IS THE FOLDING FREQUENCY
C                   ATT, THE ATTENUATION IN DB. FOR THE KAISER BESSEL
C                        WINDOW BANK
C=====================================================================
```

```
      PROGRAM FILSPC
      INTEGER OLUN
C
C INPUT THE NUMBER OF DATA POINTS N, WHERE N/2 IS THE FOLDING FREQUENCY
C
      WRITE (5,8990)
      READ (5,9990) N
      FN = FLOAT(N)
C
C SET UP CONNECTION BETWEEN THE OUTPUT LOGICAL UNIT NUMBER AND THE DISK
C FILE FOR46.DAT  USING 46 AS THE LUN
C
      OLUN = 46
C
C PROMPT FOR THE LURF PARAMETER WHICH MUST BE CALCULATED
C
      WRITE (5,8980)
      READ (5,9990) NVAR
C
C READ IN THE OTHER 4 LURF PARAMETERS
C
      IF (NVAR .EQ. 1) GO TO 120
      WRITE (5,8970)
      READ (5,9990) NF1
      F1 = FLOAT(NF1)
120   CONTINUE
      IF (NVAR .EQ. 2) GO TO 130
      WRITE (5,8960)
      READ (5,9990) NFM
      FM = FLOAT(NFM)
130   CONTINUE
      IF (NVAR .EQ. 3) GO TO 140
      WRITE (5,8950)
      READ (5,9990) M
140   CONTINUE
```

```
      IF (NVAR .EQ. 4) GO TO 150
      WRITE (5,8940)
      READ (5,9980) S
150   CONTINUE
      IF (NVAR .EQ. 5) GO TO 160
      WRITE (5,8930)
      READ (5,9980) R
160   CONTINUE
C
C CALCULATE THE OMITTED LURF PARAMETER
C
      GO TO (210,220,230,240,250) NVAR
210   CONTINUE
      F1 = FM*R**( (FLOAT(M) 1)/S)
      GO TO 260
220   CONTINUE
      FM = F1*R**((FLOAT(M) 1)/S)
      GO TO 260
230   CONTINUE
      M = INT(1.0 S*ALOG(F1/FM)/ALOG(R))
C
C ADJUST THE TERMINAL FREQUENCY SO THAT IT IS CONSISTENT WITH M
C WHICH HAS BEEN TRUNCATED TO BE AN INTEGER
C
      FM = F1*R**((FLOAT(M)-1)/S)
      GO TO 260
240   CONTINUE
      S = FLOAT(1-M)*ALOG(R)/ALOG(F1/FM)
      GO TO 260
250   CONTINUE
      R = EXP(S*ALOG(F1/FM)/FLOAT(1-M))
260   CONTINUE
C
C READ IN THE NUMBER OF DESIRED FILTER COEFFICIENTS
C
```

```
      WRITE (5,8920)
      READ (5,9990) NFF1
C
C READ IN THE DESIRED K-B ATTENUATION
C
      WRITE (5,8910)
      READ (5,9980) ATT
C
C FORCE NFF1 TO BE AN ODD NUMBER OF FILTER COEFFICIENTS
C
      IF (MOD(NFF1,2) .EQ. 0) NFF1 = NFF1 + 1
C
C WRITE OUT THE NUMBER OF ANALYSIS FREQUENCIES
C
      WRITE (OLUN,9990) M
C
C CALCULATE THE BANDEDGE FREQUENCIES FOR EACH OF THE M LURF
C ANALYSIS BANDS AND WRITE OUT WITH THE FILTER DESIGN PARAMETERS
C
      DO 300 IM = 1, M
         FIM = FLOAT(IM)
         FFIM = F1*R**((FIM-1.0)/S)
         FC = FFIM/FN
         WRITE (OLUN,9990) NFFI1
         WRITE (OLUN,9980) FC
         WRITE (OLUN,9980) ATT
300   CONTINUE
      STOP
```

```
9990  FORMAT (I4)
9980  FORMAT (F20.10)
8990  FORMAT (' N = ? ',$)
8980  FORMAT (' UNKNOWN PARAMETER ?'/'      1=F1,2=FM,3=M,4=S,5=R, ?',$)
8970  FORMAT (' F1 = ? ',$)
8960  FORMAT (' FM = ? ',$)
8950  FORMAT ('  M = ? ',$)
8940  FORMAT ('  S = ? ',$)
8930  FORMAT ('  R = ? ',$)
8920  FORMAT (' NFF1 = ? ',$)
8910  FORMAT ('  ATT = ? ',$)
      END
```

```
C=======================================================================
C MAIN PROGRAM: FILDES GENERATES THE FIR DIGITAL FILTER COEFFICIENTS FOR
C               LURF LOWPASS FILTER BANK WHICH IS DESIGNED USING A
C               KAISER-BESSEL WINDOW
C
C AUTHOR:       CHARLES L. CARNAL
C               UNIVERSITY OF TENNESSEE AT KNOXVILLE
C
C DATE:         MAY 1, 1984
C
C INPUT:        M, THE NUMBER OF FILTERS IN THE BANK
C               (THE FOLLOWING PARAMETERS ARE READ IN FOR I = 1,...,M)
C                  NF, THE NUMBER OF FILTER COEFFICIENTS
C                  FC, THE CUTOFF LOWPASS BANDEDGE FREQUENCY
C                      (NOTE: THE UNITS ARE NORMALIZED SO THAT THE
C                             FOLDING FREQUENCY IS 0.5)
C                  ATT, THE KAISER-BESSEL ATTENUATION IN DECIBELS
C
C OUTPUT:       M, THE NUMBER OF FILTERS IN THE BANK
C               (THE FOLLOWING ARE OUTPUT FOR I = 1,...,M)
C                  NF, THE NUMBER OF FILTER COEFFICIENTS
C                  BW2, THE TWO-SIDED BANDWIDTH OF THE FILTER
C                  (FOR EACH I, THE FOLLOWING ARE OUTPUT
C                   FOR J = 1, ..., NF)
C                     H( ), THE FILTER COEFFICIENTS
C=======================================================================
      PROGRAM FILDES
      PARAMETER ND = 1024
      REAL W(ND), H(ND)
      INTEGER ILUN, OLUN
C
C SET UP CONNECTION BETWEEN THE INPUT AND OUTPUT LOGICAL UNIT NUMBERS
C AND DISK FILES FOR46.DAT, FOR47.DAT
C
```

```
      ILUN = 46
      OLUN = 47
C
C COMPUTE THE VALUE OF PI
C
      PI = 4.0 * ATAN(1.0)
      TWPI = 2.0 * PI
C
C READ IN AND WRITE BACK OUT THE NUMBER OF FILTERS IN THE BANK
C
      READ (ILUN,9990) M
      WRITE (OLUN,9990) M
C
C CALCULATE THE COEFFICIENTS FOR EACH OF THE M FILTERS IN THE BANK
C AND WRITE THEM OUT
C
      DO 800 M1=1,M
         READ (ILUN,9990) NF
C
C CALCULATE THE HALFLENGTH OF THE FILTER, ONLY HALF OF THE FILTER
C COEFFICIENTS WILL BE CALCULATED SINCE THE FILTER IS SYMMETRICAL
C
         N = (NF + 1) / 2
         READ (ILUN,9980) FC
C
C CALCULATE THE TWO-SIDED BANDWIDTH
C
         BW2 = 2.0 * FC
C
C COMPUTE THE SIN X/X IDEAL BRICK WALL IMPULSE RESPONSE FOR N TERMS
C
         H(1) = BW2
         DO 100 I = 2, N
            FI = FLOAT(I - 1)
            H(I) = SIN(TWPI * FI * FC) / (PI * FI)
100      CONTINUE
```

```
C
C COMPUTE THE BESSEL FUNCTION ARGUMENT BETA FROM THE ATTENUATION SPEC.
C
          READ (ILUN,9980) ATT
          IF (50.0 - ATT) 150,150,180
150       CONTINUE
          BETA = 0.1102*(ATT-8.7)
          GO TO 190
180       CONTINUE
          BETA = 0.58417 * (ATT - 20.96)**0.4 + 0.07886 * (ATT - 20.96)
190       CONTINUE
C
C COMPUTE AND APPLY THE KAISER-BESSEL TIME DOMAIN WINDOW TO THE IDEAL
C SIN X/X IMPULSE RESPONSE
C
          CALL KAIBES (NF, W, ND, N, BETA)
          DO 200 I = 1, N
             H(I) = H(I) * W(I)
200       CONTINUE
C
C WRITE THE FILTER DATA OUT TO THE DISK
C
          WRITE (OLUN,9990) NF
          WRITE (OLUN,9980) BW2
          DO 300 I = 1, N
             WRITE (OLUN,9980) H(N+1-I)
300       CONTINUE
          DO 400 I = 2, N
             WRITE (OLUN,9980) H(I)
400       CONTINUE
800    CONTINUE
       STOP
9990   FORMAT (I4)
9980   FORMAT (F20.10)
       END
```

```
C=======================================================================
C MAIN PROGRAM: FRESAM GENERATES THE FIR DIGITAL FILTER COEFFICIENTS
C               FOR THE LURF LOWPASS FILTER BANK.  THE FREQUENCY
C               SAMPLING METHOD IS USED
C
C AUTHOR:       CHARLES L. CARNAL
C               UNIVERSITY OF TENNESSEE AT KNOXVILLE
C
C DATE:         MAY 1, 1984
C
C INPUT:        N, THE NUMBER OF FILTER TERMS
C               NTRANS, THE NUMBER OF TRANSITION SAMPLES
C               M, THE NUMBER OF FILTERS IN THE BANK
C                (THE FOLLOWING ARE READ IN FOR I = 1,...,M)
C                  NF, THE NUMBER OF FILTER TERMS GENERATED BY FILSPC.
C                  ATT, THE KAISER-BESSEL ATTENUATION GENERATED BY FILSPC
C                    (NOTE: BOTH NF AND ATT ARE NOT USED IN FRESAM)
C                  FC, THE CUTOFF LURF FREQUENCIES
C
C OUTPUT:       M, THE NUMBER OF FILTERS IN THE BANK
C               (THE FOLLOWING ARE OUTPUT FOR I = 1,...,M)
C                  N, THE NUMBER OF FILTER COEFFICIENTS
C                  BW2, THE TWO-SIDED BANDWIDTH
C                  (FOR EACH I, THE FOLLOWING ARE OUTPUT
C                   FOR J = 1,...,N)
C                     XR, THE FILTER COEFFICIENTS
C
C=======================================================================
      PROGRAM FRESAM
C
C SET UP CONNECTION BETWEEN THE INPUT AND OUTPUT LOGICAL UNIT NUMBERS
C AND DISK FILES FOR46.DAT, FOR47.DAT
C
      PARAMETER ILUN=46
      PARAMETER OLUN=47
```

```
      PARAMETER NXC=1023, NW=511
      COMPLEX X(0:NXC,0:1), W(0:NW)
      TWPI = 8.0*ATAN(1.0)
C
C READ IN THE NUMBER OF FILTER COEFFICIENTS AND THE NUMBER OF TRANSITION
C SAMPLES.  CALCULATE THE SLOPE OF A STRAIGHT LINE FROM THE PASSBAND
C TO THE STOPBAND USING THE NUMBER OF TRANSITION SAMPLES AS THE
C WIDTH OF THE TRANSITION BAND
C
      WRITE (5,8990)
      READ (5,9990) N
      WRITE (5,8980)
      READ (5,9990) NTRANS
      DLFT = FLOAT(NTRANS)/FLOAT(N)
      XFTSLP = -1.0/DLFT
      NM1 = N-1
      NV2 = N/2
      FNV2 = FLOAT(NV2)
      FN = FLOAT(N)
      READ (ILUN,9990) M
C
C WRITE OUT THE NUMBER OF FILTERS AND START THE LOOP INCREMENTING
C BY EACH FILTER
C
      WRITE (OLUN,9990) M
      DO 700 IM = 1, M
C
C NF AND ATT ARE READ AND DISCARDED SO THAT THE FILE RECORD POINTER
C CAN GET THE LURF BANDWIDTHS
C
         READ (ILUN,9990) NF
         WRITE (OLUN,9990) N
         READ (ILUN,9980) FC
         BW2 = 2.0*FC
         WRITE (OLUN,9980) BW2
         READ (ILUN,9980) ATT
```

```
C
C COMPUTE THE LOWER AND UPPER ENDS OF THE TRANSITION BAND
C
          XFTO = 0.5 - XFTSLP*FC
          FNTL = FNV2*(FC - (0.5*DLFT))/0.5
          FNTU = FNV2*(FC + (0.5*DLFT))/0.5
C
C SET UP THE MAGNITUDES
          DO 510 I = 0, NV2-1
             FI = FLOAT(I)
C
C IF IN THE PASSBAND SET MAGNITUDE EQUAL TO 1.0
C
             IF (FI .GT. FNTL) GO TO 400
             X(I,0) = CMPLX(1.0,0.0)
             GO TO 500
400          CONTINUE
C
C IF IN THE STOPBAND SET MAGNITUDE EQUAL TO 0.0
C
             IF (FI .LT. FNTU) GO TO 450
             X(I,0) = 0.0
             GO TO 500
450          CONTINUE
C
C IF IN THE PASSBAND SET MAGNITUDE ACCORDING TO A STRAIGHT LINE
C CONNECTING THE PASSBAND TO THE STOPBAND
C
             FFI = FI/FN
             XR = XFTSLP*FFI + XFTO
             X(I,0) = CMPLX(XR,0.0)
             IF ((XR .LE. 1.0) .AND. (XR .GE. 0.0)) GO TO 500
             STOP
500          CONTINUE
510       CONTINUE
```

```
C COMPUTE THE PHASE TO BE LINEAR
C
          X(NV2,0) = CMPLX(0.0,0.0)
          DO 600 I = NV2+1, NM1
             X(I,0) = X(N-I,0)
600       CONTINUE
          PHARG = (-TWPI/FN)*FLOAT(NM1)/2.0
          DO 625 I = 0, NV2-1
             FI = FLOAT(I)
             X(I,0) = X(I,0)*CEXP(CMPLX(0.0,FI*PHARG))
625       CONTINUE
          PHARG = -PHARG
          DO 650 I = NV2+1, N-1
             FNMI = FLOAT(N-I)
             X(I,0) = X(I,0)*CEXP(CMPLX(0.0,FNMI*PHARG))
650       CONTINUE
C
C TAKE THE IDFT OF THE FREQUENCY SAMPLE ASSIGNMENTS
C
          L = INT((ALOG(FN)/ALOG(2.0))+0.5)
          CALL WCOEF2 (W,NW,N)
          CALL FFT2 (X,NXC,W,NW,N,L,+1)
C
C WRITE OUT THE IMPULSE RESPONSE
C
          DO 675 I = 0, N-1
             XR = REAL(X(I,0))
             WRITE (OLUN,9980) XR
675       CONTINUE
700    CONTINUE
       STOP
9990   FORMAT (I4)
9980   FORMAT (F20.10)
8990   FORMAT ('   ENTER  N: ',$)
8980   FORMAT ('   ENTER THE NUMBER OF TRANSITION SAMPLES: ',$)
       END
```

```
C=======================================================================
C MAIN PROGRAM: LURFLP IMPLEMENTS THE LURF LOWPASS FILTER BANK ON A SET
C               OF DATA SUPPLIED TO THE PROGRAM
C
C AUTHOR:       CHARLES L. CARNAL
C               UNIVERSITY OF TENNESSEE AT KNOXVILLE
C
C DATE:         MAY 1, 1984
C
C INPUT:        IDALUN, THE LOGICAL UNIT NUMBER WHERE THE DATA IS FOUND
C               N, THE OPERATION TIME OF THE FILTERS
C               TSAMP, THE TIME INTERVAL AT WHICH THE DATA WAS SAMPLED
C               (THE FOLLOWING ARE READ FOR I = 1,...,2N)
C                  X( ), THE RAW TIME SERIES DATA
C               M, THE NUMBER OF FILTERS IN THE BANK
C               (THE FOLLOWING ARE READ FOR I = 1,...,M)
C                  NF, THE NUMBER OF FILTER TERMS
C                  BW2, THE TWO-SIDED BANDWIDTH IN FREQUENCY UNITS
C                       WHERE THE FOLDING FREQUENCY IS 0.5
C                  (THE FOLLOWING ARE READ IN FOR J = 1,...,NF)
C                     H( ), THE FILTER COEFFICIENTS
C
C OUTPUT:       (THE FOLLOWING ARE OUTPUT FOR L = 1,...,M)
C                  L, THE ANALYSIS FREQUENCY NUMBER
C                  SLURF, THE LURF PSD ESTIMATES
C
C=======================================================================
      PROGRAM LURFLP
      PARAMETER NXL = -1024, NXH = 1023
      PARAMETER NH = 512
C
C SET UP CONNECTION BETWEEN THE INPUT AND OUTPUT LOGICAL UNIT NUMBERS
C AND DISK FILES FOR47.DAT, FOR57.DAT
C
      PARAMETER ILUN = 47
      PARAMETER OLUN = 57
```

```
      REAL X(NXL:NXH), H(0:NH)
C
C READ IN THE LUN WHICH GIVES THE FILE IN WHICH THE RAW TIME SERIES
C DATA IS FOUND, FOR"IDALUN".DAT
C
      WRITE (5,8990)
      READ (5,9990) IDALUN
C
C READ IN THE FILTER OPERATION TIME N, THE SAMPLE INTERVAL TSAMP
C AND THE 2N RAW TIME SERIES DATA POINTS
C
      READ (IDALUN,9990) N
      READ (IDALUN,9980) TSAMP
      DO 200 L1 = -N, (N-1)
         READ (IDALUN,9980) X(L1)
200   CONTINUE
C
C READ IN THE NUMBER OF FILTERS IN THE BANK, INITIALIZE THE PARAMETERS
C WHICH ARE USED IN THE DIFFERENCING OPERATION, AND START THE MAJOR
C LOOP FOR EACH FILTER IN THE BANK
C
      READ (ILUN,9990) M
      SLAST = 0.0
      BWLAST = 0.0
      DO 600 L = 1, M
C
C READ IN THE NUMBER OF FILTER TERMS, THE TWO-SIDED BANDWIDTH, AND
C THE FILTER COEFFICIENTS OF THE LTH FILTER
C
         READ (ILUN,9990) NF
         READ (ILUN,9980) BW2
         DO 300 L1 = 0, (NF-1)
            READ (ILUN,9980) H(L1)
300      CONTINUE
         S = 0.0
```

```
C
C START THE TIME OPERATION OF THE LURF PROCESSING
C
         DO 500 I = 0, (N-1)
            V = 0.0
C
C PERFORM THE CONVOLUTION OPERATION OF THE FILTER
C
            DO 400 J = 0, (NF-1)
               V = V + H(J)*X(I-J)
400         CONTINUE
C
C COMPUTE THE INSTANTANEOUS POWER P AND ACCUMULATE FOR AVERAGE POWER
C
            P = V*V
            S = S + P
500      CONTINUE
C
C COMPUTE THE LURF PSD ESTIMATE BY SUBTRACTING THE ENERGY, DIVIDING BY
C THE TIME INTERVAL AND BANDWIDTH TO GET THE AVERAGE POWER DENSITY
C
         SLURF = (S - SLAST)/(N*(BW2 - BWLAST))
         SLAST = S
         BWLAST = BW2
C
C WRITE OUT THE LURF PSD ESTIMATES
C
         WRITE (OLUN,9970) L, SLURF
600   CONTINUE
      STOP
9970  FORMAT (I5,5X,F20.10)
9980  FORMAT (F20.10)
9990  FORMAT (I4)
8990  FORMAT ('  ENTER THE LUN OF THE TIME SERIES DATA ? ',$)
      END
```

```
C=====================================================================
C MAIN PROGRAM: DFTFDW COMPUTES THE LURF PSD USING THE FREQUENCY DOMAIN
C               WINDOW METHOD.  A MODIFIED DANIELL WINDOW IS USED.
C               THE MODIFICATION CONSISTS OF LINEARLY TAPERING THE ENDS
C               OF THE WINDOW
C
C AUTHOR:       CHARLES L. CARNAL
C               UNIVERSITY OF TENNESSEE AT KNOXVILLE
C
C DATE:         MAY 1, 1984
C
C INPUT:        IDALUN, THE LUN OF THE INPUT RAW TIME SERIES DATA
C               NV2, THE LENGTH OF THE INPUT DATA TO BE USED
C               TSAMP, THE SAMPLING TIME INTERVAL OF THE INPUT DATA
C               (THE FOLLOWING ARE READ FOR I = 1,...,2N)
C                  XR, THE INPUT RAW TIME SERIES DATA
C               NTRANS, THE NUMBER OF TRANSITION WINDOW WEIGHTS
C               M, THE NUMBER OF LURF ANALYSIS FREQUENCIES
C               (THE FOLLOWING ARE READ FOR I = 1,...,M)
C                  NF, THE NUMBER OF FILTER TERMS (THROWN AWAY)
C                  FC, THE LURF BANDEDGES
C                  ATT, THE FILTER ATTENUATION (THROWN AWAY)
C
C OUTPUT:       (THE FOLLOWING ARE OUTPUT FOR IM = 1,...,M)
C                  IM, THE ANALYSIS FREQUENCY NUMBER
C                  XLURF, THE LURF PSD ESTIMATES
C=====================================================================
      PROGRAM DFTFDW
C
C SET UP THE CONNECTION BETWEEN THE INPUT AND OUTPUT LOGICAL UNIT NUMBERS
C AND DISK FILES FOR46.DAT, FOR57.DAT
C
      PARAMETER ILUN=46
      PARAMETER OLUN=57
      PARAMETER NXC = 2047, NXP = 1023, NW = 1023
```

```
      COMPLEX X(0:NXC,0:1), W(0:NW)
      REAL XP(0:NXP), WF(0:100)
C
C COMPUTE THE VALUE OF 2*PI
C
      TWPI = 8.0*ATAN(1.0)
C
C READ IN THE LUN WHERE THE RAW TIME SERIES DATA WILL BE FOUND
C READ IN THE TIME SERIES DATA FROM THE FILE FOR"IDALUN".DAT
C
      WRITE (5,8990)
      READ (5,9990) IDALUN
C
C SET NV2 AS HALF THE NUMBER OF DATA POINTS AVAILABLE
C
      READ (IDALUN,9990) NV2
      N = 2*NV2
      FN = FLOAT(N)
      FNV2 = FLOAT(NV2)
C
C THROW AWAY THE FIRST N/2 DATA POINTS AND PUT THE VALUE OF
C ZERO IN THEIR PLACE. THIS WILL MAKE THIS IMPLEMENTATION MORE
C COMPARABLE WITH THE LURF LOWPASS IMPLEMENTATION
C
      DO 100 I = 0, NV2-1
         READ (IDALUN,9980) XR
         X(I,0) = CMPLX(0.0,0.0)
100   CONTINUE
C
C READ IN THE SECOND N/2 RAW TIME SERIES DATA POINTS FOR ANALYSIS
C
      DO 200 I = NV2, N-1
         READ (IDALUN,9980) XR
         X(I,0) = CMPLX(XR,0.0)
200   CONTINUE
```

```
C COMPUTE THE PERIODOGRAM OF THE RAW TIME SERIES DATA
C
      L = INT((ALOG(FN)/ALOG(2.0))+0.5)
      CALL WCOEF2 (W,NW,N)
      CALL FFT2 (X,NXC,W,NW,N,L,-1)
      CALL PDGRM(X,NXC,XP,NXP,N)
C
C READ IN THE NUMBER OF TRANSITION BANDEDGE SAMPLES FOR THE DANIELL
C WINDOW MODIFICATION
C COMPUTE THE SLOPE OF A STRAIGHT LINE ACROSS THE BANDEDGE
C INITIALIZE THE WINDOWING PARAMETERS
C
      WRITE (5,8980)
      READ (5,9990) NTRANS
      DLFT = FLOAT(NTRANS)/FN
      XFTSLP = -1.0/DLFT
      XLURF = 0.5*XP(0)
      FNTLO = 0.0
      FNTUO = 0.0
      NTLO = 0
      NTUO = 0
      WINSUM = 1.0
      FCIM1 = 0.0
C
C READ IN THE NUMBER OF LURF ANALYSIS FREQUENCIES
C AND START THE MAJOR LOOP FOR EACH ANALYSIS FEQUENCY
C
      READ (ILUN,9990) M
      DO 800 IM = 1, M
C
C READ IN THE NUMBER OF FILTER TERMS NF, THE LURF LOWPASS BANDEDGE FC,
C AND THE KAISER-BESSEL ATTENUATION ATT.  THROW AWAY NF AND ATT
C
         READ (1LUN,9990) NF
         READ (ILUN,9980) FC
         READ (1LUN,9980) ATT
```

```
C
C FIND THE LOWER AND UPPER EDGES OF THE LINEAR TAPER TO BE
C APPLIED AT THE ENDS OF THE DANIELL WINDOW
C
          XFTO = 0.5 - XFTSLP*FC
          FNTL1 = (FC - (0.5*DLFT))
          FNTU1 = (FC + (0.5*DLFT))
          NTL1 = INT(FNTL1*FN + 0.5) + 1
          NTU1 = INT(FNTU1*FN + 0.5) - 1
C
C ON THE FIRST PASS THERE IS NO LOWER EDGE
C
          IF (NTUO .EQ. 0) GO TO 500
C
C APPLY THE LINEAR TAPER TO THE LOWER WINDOW EDGE
C
          DO 400 I = NTLO, NTUO
             WINSUM = WINSUM*WF(I-NTLO)
             XLURF = XLURF + XP(I)*WF(I-NTLO)
400       CONTINUE
500       CONTINUE
C
C APPLY THE UNITY WINDOW WEIGHTS
C
          DO 600 I = NTUO+1, NTL1-1
             WINSUM = WINSUM + 1.0
             XLURF = XLURF + XP(I)
600       CONTINUE
C
C COMPUTE THE LINEAR TAPER WINDOW WEIGHTS FOR THE CURRENT UPPER
C EDGE OF THE WINDOW, SAVE FOR THE FOLLOWING LOWER WINDOW EDGE, AND
C APPLY
C
```

```
            DO 700 I = NTL1, NTU1
               FI = FLOAT(I)
               FFI = FI/FN
               WFACT = XFTSLP*FFI + XFTO
               WF(NTU1-I) = WFACT
               WINSUM = WINSUM + WFACT
               XLURF = XLURF + XP(I)*WFACT
700         CONTINUE
C
C COMPUTE THE LURF PSD ESTIMATE.  MUST MULTIPLY BY TWO SINCE THE
C SYMMETRIC OTHER HALF OF THE PERIODOGRAM WAS NOT COMPUTED. THE
C DIVISION BY THE SUM OF THE WINDOW WEIGHTS MAINTAINS THE SAME UNITS
C OF POWER DENSITY
C
            XLURF = 2.0*XLURF/WINSUM
            WRITE (OLUN,8970) IM,FNTLO,FCIM1,FNTUO,FNTL1,FC,FNTU1,XLURF
C
C POST--COMPUTATION INITIALIZATION FOR THE NEXT ITERATION
C
            FCIM1 = FC
            XLURF = 0.0
            WINSUM = 0.0
            NTLO = NTL1
            NTUO = NTU1
            FNTLO = FNTL1
            FNTUO = FNTU1
800      CONTINUE
         STOP
9990     FORMAT (I4)
9980     FORMAT (F20.10)
8990     FORMAT ('   ENTER  LUN  OF INPUT DATA:  ',$)
8980     FORMAT ('   ENTER THE NUMBER OF TRANSITION SAMPLES: ',$)
8970     FORMAT (T2,I4,6(5X,F7.5),10X,F15.4)
         END
```

```
C-----------------------------------------------------------------------
C SUBROUTINE: KAISBES COMPUTES THE KAISER-BESSEL TIME DOMAIN WINDOW
C
C INPUT:      NF, TOTAL LENGTH OF SYMMETRICAL WINDOW
C             ND, DIMENSION OF W
C             BETA, WINDOW PARAMETER
C             N, THE NUMBER OF WINDOW WEIGHTS TO CALCULATE
C
C OUTPUT:     W( ), THE WINDOW WEIGHTS
C-----------------------------------------------------------------------
      SUBROUTINE KAIBES(NF, W, ND, N, BETA)
      REAL W(ND)
      REAL IO
      BES = IO(BETA)
      FNFM1 = FLOAT(NF - 1)
      XIND = FNFM1 * FNFM1
      DO 100 I = 1, N
         FIM1 = FLOAT(I - 1)
         W(I) = IO(BETA * SQRT(1.0 - (4.0 * FIM1 * FIM1 / XIND))) / BES
100   CONTINUE
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C FUNCTION: IO CALCULATES THE OTH ORDER BESSEL FUNCTION USING AN ALGORITHM
C           ORIGINALLY PROPOSED BY KAISER, 1974
C-----------------------------------------------------------------------
      REAL FUNCTION IO(X)
      XV2 = X / 2.0
      T = 1.0E-08
      E = 1.0
      DE = 1.0
      DO 100 I = 1, 25
         FI = FLOAT(I)
         DE = DE * XV2 / FI
         SDE = DE * DE
         E = E + SDE
         IF (E * T - SDE) 100, 100, 200
100   CONTINUE
200   IO = E
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C SUBROUTINE: FFT2 COMPUTES THE RADIX-2 DFT OR IDFT
C
C INPUT:      X, THE INPUT WAVEFORM IN X(*,0)
C             NX, DIMENSION OF X
C             W, TWIDDLE FACTORS
C             NW, DIMENSION OF W
C             N, THE LENGTH OF THE TRANSFORM
C             L, MUST SATISFY N=2**L, WHERE L IS AN INTEGER
C             INV, IF INV EQUALS +1 THE IDFT IS PERFORMED
C                  OTHERWISE THE DFT IS PERFORMED
C
C OUTPUT:     X, THE OUTPUT TRANSFORM IN X(*,0)
C-----------------------------------------------------------------------
      SUBROUTINE FFT2(X,NX,W,NW,N,L,INV)
      COMPLEX X(0:NX,0:1), W(0:NW), XTMP0, XTMP1, WTWIDL
      NV2=N/2
C
C SET UP TOGGLE VARIABLES TO SWAP THE INTERMEDIATE RESULTS BACK AND
C FORTH BETWEEN X(*,0) AND X(*,1) DURING THE FFT COMPUTATION
C
      ITOG=0
      JTOG=1
      J0=NV2
      J1=1
C
C START LOOP FOR ALGORITHM COLUMN ITERATION
C
      DO 300 I0=1,L
         J2=0
         J3=J0
C
C SET UP LOOP FOR INDEX COMPUTATION
C
         DO 200 I1=1,J1
```

```
C
C SET UP TWIDDLE FACTOR FOR THE BUTTERFLY OPERATIONS
C FOR EITHER DFT OR IDFT
C
            WTWIDL = W(J2)
            IF (INV .EQ. 1) WTWIDL = CONJG(W(J2))
C
C SET UP INNER LOOP FOR INDEX COMPUTATION OVER BUTTERFLIES
C
            DO 100 I2=J2,J3-1
C
C DO BUTTERFLY COMPUTATIONS
C
               XTMP0=X(I2+J2,ITOG)
               XTMP1=WTWIDL*X(I2+J3,ITOG)
               X(I2,JTOG)=XTMP0+XTMP1
               X(I2+NV2,JTOG)=XTMP0-XTMP1
100         CONTINUE
            J2=J3
            J3=J3+J0
200      CONTINUE
         J0=J0/2
         J1=2*J1
C
C TOGGLE THE TOGGLE PARAMETERS FOR A SWAP ON THE NEXT ITERATION
C
         ITMP=ITOG
         ITOG=JTOG
         JTOG=ITMP
300   CONTINUE
```

```
C
C MAKE SURE THAT THE RESULTS ARE LEFT IN X(*,0) AND NOT X(*,1)
C
      IF (ITOG .EQ. 0) GO TO 500
      DO 400 IO=0,N-1
         X(IO,0)=X(IO,1)
400   CONTINUE
500   CONTINUE
C
C IF INVERSE DFT, DIVIDE RESULTS BY N
C
      IF (INV .NE. +1) GO TO 600
      FN = FLOAT(N)
      DO 550 IO = 0, N-1
         X(IO,0) = X(IO,0)/FN
550   CONTINUE
600   CONTINUE
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C SUBROUTINE: WCOEF2 COMPUTES THE TWIDDLE FACTORS FOR A RADIX-2 FFT
C
C INPUT:      NW, THE DIMENSION OF W
C             N, THE LENGTH OF THE TRANSFORM
C
C OUTPUT:     W( ), THE TWIDDLE FACTORS
C-----------------------------------------------------------------------
      SUBROUTINE WCOEF2 (W,NW,N)
      COMPLEX W(0:NW)
C
C ASSIGN THE VALUE OF 2*PI
C
      DATA TWPI/6.28318530717958/
      NV2=N/2
      NV4=N/4
      FN=FLOAT(N)
      EARG=-TWPI/FN
      W(0)=CMPLX(1.0,0.0)
      W(NV4)=CMPLX(0.0,-1.0)
C
C USING SYMMETRY, ONLY APPROX. N/8 VALUES HAVE TO BE COMPUTED
C THE FFT ALGORITHM ONLY USES N/2 VALUES, SO ONLY N/2 ARE EVEN ASSIGNED
C
      DO 100 I=1,(N/8)
         FI=FLOAT(I)
         W(I)=CEXP(CMPLX(0.0,FI*EARG))
         WR=REAL(W(I))
         WI=AIMAG(W(I))
         W(NV4-I)=CMPLX(-WI,-WR)
         W(NV4+I)=CMPLX(WI,-WR)
         W(NV2-I)=CMPLX(-WR,WI)
100   CONTINUE
      RETURN
      END
```

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

C SUBROUTINE: PDGRM COMPUTES THE PERIODOGRAM OF ITS INPUT
C
C INPUT:      X, INPUT TRANSFORM DATA
C             NXC, DIMENSION OF X
C             NXP, DIMENSION XP
C             N, THE WHOLE LENGTH OF THE INPUT TRANSFORM
C
C OUTPUT:     XP, THE PERIODOGRAM WITH ONLY N/2 POINTS
C-----------------------------------------------------------------------
      SUBROUTINE PDGRM(X,NXC,XP,NXP,N)
      COMPLEX X(0:NXC,0:1)
      REAL XP(0:NXP)
      FN = FLOAT(N)
C
C ONLY COMPUTE N/2 VALUES SINCE THE PERIODOGRAM IS SYMMETRICAL SINCE
C THE ORIGINAL TIME SERIES DATA WAS REAL
C
      DO 100 I = 0, N/2 - 1
         XP(I) = ((CABS(X(I,0)))**2)/FN
100   CONTINUE
      RETURN
      END
```

VITA

Charles Lynn Carnal was born in Humboldt, Tennessee on December 13, 1954. He attended Alamo High School and was graduated in June 1972. In 1973 he enlisted in the United States Marine Corps where he attended Communications-Electronics School. Upon leaving active duty in 1975 he began undergraduate study at the University of Tennessee at Martin and was graduated with a Bachelor of Science degree in Engineering Technology in 1978. He entered the University of Tennessee Space Institute in September 1978 where he received the Master of Science degree with a major in Electrical Engineering in 1980. In 1981 he became a full-time instructor at the University of Tennessee at Knoxville to pursue a doctorate in Electrical Engineering.