DIGITAL FILTER DESIGN BY MINIMIZING THE
WEIGHTED SQUARE ERROR IN THE FREQUENCY DOMAIN

By

James L. McNett
A Thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science in Geophysical Engineering.

Signed: James L. McNett

Golden, Colorado
Date: January, 1967

Approved: Frank A. Hadsell
Frank A. Hadsell
Thesis Advisor

John C. Hollister
Head of Department

Golden, Colorado
Date: Jan 19th, 1967
ABSTRACT

In the hopes of shifting error from one portion of the spectrum to another and hence, achieving acceptable filter characteristics in the important part of the spectrum with a discrete filter of fewer points (shorter time duration), N-point filters were designed by minimizing the weighted square difference between the specified and actual frequency characteristics. The filters must satisfy an Nth order set of linear simultaneous equations whose coefficient matrix is made up of the inverse transform of the error weighting function and whose constant vector is made up of the inverse transform of the product of the weighting function with the specified response. For uniformly weighted error (normal least squares) the coefficient matrix reduces to an identity matrix and it is only necessary to sample the specified filters impulse response.

Truncation and sampling effects for some uniformly weighted error filters are effectively explained and
illustrated using Fourier-series kernels and plots of square error vs N. The error plots provide a quick and quantitative method for choosing the smallest N which gives the desired error level in a filter.

Discrete filters with error weighting functions of blocks, exponentials, and various combinations of blocks and exponentials were computed using a Gauss-elimination technique. Weighting the square error nonuniformly produces smaller errors in the heavily weighted portions of the spectrum, larger errors in the lightly weighted portions of the spectrum, and greater total square error as compared to the case of uniform weighting. The major shortcoming of the error weighting technique is that as weight contrast increases, the error in the lightly weighted portions of the spectrum increases disproportionately to the decrease in error in the heavily weighted portions. Obtaining a significant improvement in one portion of the spectrum can only be accomplished at the expense of an error several orders of magnitude larger in the remainder of the spectrum. If a portion of the spectrum can be totally ignored the filters may be shortened. In this study, the lowpass filters for N = 13-31 and the bandpass filters for N = 15-41 could be shortened up to six points while maintaining the same total square error in the spectral band of importance.
However, for the filters examined, better overall characteristics could be obtained for a filter with a fixed number of points by doubling the sample interval in the unweighted filter than by using any weighting configuration.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>xii</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>BACKGROUND</td>
<td>4</td>
</tr>
<tr>
<td>Discrete Analysis and Sampling</td>
<td>4</td>
</tr>
<tr>
<td>Discrete Fourier Transforms</td>
<td>7</td>
</tr>
<tr>
<td>Discrete Convolution</td>
<td>9</td>
</tr>
<tr>
<td>Sampling and Truncation Effects</td>
<td>11</td>
</tr>
<tr>
<td>FILTERING CONCEPTS AND THESIS OBJECTIVES</td>
<td>16</td>
</tr>
<tr>
<td>Filtering Concepts</td>
<td>16</td>
</tr>
<tr>
<td>Discrete or Digital Filters</td>
<td>18</td>
</tr>
<tr>
<td>Time-Frequency Filter Relationships</td>
<td>21</td>
</tr>
<tr>
<td>Discrete Filter Computation</td>
<td>23</td>
</tr>
<tr>
<td>Thesis Objectives</td>
<td>25</td>
</tr>
<tr>
<td>ERROR CRITERIA AND OPTIMIZATION</td>
<td>26</td>
</tr>
<tr>
<td>Error Criteria</td>
<td>26</td>
</tr>
<tr>
<td>Weighted Square Error</td>
<td>30</td>
</tr>
<tr>
<td>Minimization of the Weighted Square Error</td>
<td>32</td>
</tr>
<tr>
<td>Normal Equations</td>
<td>37</td>
</tr>
</tbody>
</table>
ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Finite sequence of impulses</td>
<td>6</td>
</tr>
<tr>
<td>2.</td>
<td>Illustration of sampling process</td>
<td>8</td>
</tr>
<tr>
<td>3.</td>
<td>Superposition property of convolution</td>
<td>10</td>
</tr>
<tr>
<td>4.</td>
<td>The Fourier transform of the sampling function</td>
<td>13</td>
</tr>
<tr>
<td>5.</td>
<td>Comparison of true spectrum and sampled-data spectrum</td>
<td>14</td>
</tr>
<tr>
<td>6.</td>
<td>Sampled-data spectra for two sample intervals. No truncation effects</td>
<td>15</td>
</tr>
<tr>
<td>7.</td>
<td>Seven-point digital filter</td>
<td>19</td>
</tr>
<tr>
<td>8.</td>
<td>Filter breakdown in the frequency and time domain</td>
<td>22</td>
</tr>
<tr>
<td>9.</td>
<td>Error importance $EI_p$ vs error magnitude</td>
<td>29</td>
</tr>
<tr>
<td>10.</td>
<td>Generalized flow chart for weighted error filter design</td>
<td>44</td>
</tr>
<tr>
<td>11.</td>
<td>Impulse responses of four filters with uniformly weighted error</td>
<td>51</td>
</tr>
<tr>
<td>12.</td>
<td>Normalized Fourier transforms of even unit sampling sequences</td>
<td>52</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>13.</td>
<td>Amplitude responses of a progressively truncated lowpass discrete filter sampled at $l\Delta t$</td>
<td>54</td>
</tr>
<tr>
<td>14.</td>
<td>Amplitude responses of a progressively truncated lowpass discrete filter sampled at $l\Delta t$</td>
<td>55</td>
</tr>
<tr>
<td>15.</td>
<td>Amplitude responses of a progressively truncated bandpass discrete filter sampled at $l\Delta t$</td>
<td>56</td>
</tr>
<tr>
<td>16.</td>
<td>Amplitude responses of a progressively truncated bandpass discrete filter sampled at $l\Delta t$</td>
<td>57</td>
</tr>
<tr>
<td>17.</td>
<td>Amplitude responses of some truncated broadband discrete filters sampled at $l\Delta t$</td>
<td>60</td>
</tr>
<tr>
<td>18.</td>
<td>Fourier-series kernel width vs $N$ for two sampling rates</td>
<td>62</td>
</tr>
<tr>
<td>19.</td>
<td>Amplitude responses of a lowpass discrete filter for $N = 7$ and sample rates of $l\Delta t$, $2\Delta t$, and $3\Delta t$</td>
<td>63</td>
</tr>
<tr>
<td>20.</td>
<td>Amplitude responses of a bandpass discrete filter for $N = 15$ and sample rates of $l\Delta t$, $2\Delta t$, and $3\Delta t$</td>
<td>64</td>
</tr>
<tr>
<td>21.</td>
<td>Amplitude responses of a lowpass discrete filter holding time span fixed and varying $N$ and sample rate</td>
<td>66</td>
</tr>
<tr>
<td>22.</td>
<td>Amplitude responses of a bandpass discrete filter holding time span fixed and varying $N$ and sample rate</td>
<td>67</td>
</tr>
<tr>
<td>23.</td>
<td>Percent square error vs $N$ for a discrete lowpass filter sampled at $l\Delta t$ and $2\Delta t$</td>
<td>70</td>
</tr>
<tr>
<td>24.</td>
<td>Percent square error vs $N$ for a discrete bandpass filter sampled at $l\Delta t$ and $2\Delta t$</td>
<td>71</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>25. Percent square error vs N for an assortment of discrete filters sampled at $1\Delta t$</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>26. Percent square error, percent error, and percent of impulse response sampled all vs N for a lowpass filter and bandpass filter</td>
<td>75</td>
<td></td>
</tr>
<tr>
<td>27. Error weighting functions</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>28. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td>29. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>85</td>
<td></td>
</tr>
<tr>
<td>30. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>86</td>
<td></td>
</tr>
<tr>
<td>31. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>87</td>
<td></td>
</tr>
<tr>
<td>32. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td>33. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>89</td>
<td></td>
</tr>
<tr>
<td>34. Amplitude responses of a weighted error filter for various weight contrasts</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>35. Comparison of percent square error curves for weighted and unweighted error lowpass filters</td>
<td>92</td>
<td></td>
</tr>
<tr>
<td>36. Comparison of percent square error curves for weighted and unweighted error bandpass filters</td>
<td>93</td>
<td></td>
</tr>
<tr>
<td>37. Weighted error sampling sequences at $1\Delta t$ for a lowpass filter</td>
<td>102</td>
<td></td>
</tr>
<tr>
<td>38. Fourier transforms of weighted error sampling sequences</td>
<td>105</td>
<td></td>
</tr>
</tbody>
</table>
Figures

39. Amplitude responses of a weighted error filter for various weight positions ........ 107
40. Amplitude responses of a weighted error filter for various weight positions ........ 108
41. Amplitude responses of a weighted error filter for various weight positions ........ 109
42. Amplitude responses of a weighted error filter for various weight positions ........ 110
43. Amplitude responses of a weighted error filter for various weight positions ........ 111
44. Amplitude responses of a discrete filter with exponentially weighted error .......... 119
45. Amplitude responses of a discrete filter with exponentially weighted error .......... 120
46. Amplitude responses of a discrete filter with exponentially weighted error .......... 121

Tables

Percent square error vs $N$ for six weighting contrasts: Lowpass filter ............... 95
Percent square error vs $N$ for six weighting contrasts: Bandpass filter ............. 96
Percent square error vs $N$ for five weighting positions: Lowpass filter ............ 113
Percent square error vs $N$ for three weighting positions: Bandpass filter .......... 115
ACKNOWLEDGMENTS

The author gratefully acknowledges the assistance of Professor Frank A. Hadsell, thesis advisor, for patiently reading the manuscript and making many helpful suggestions.

Special thanks are due Russell L. Gray for stimulating the author's interest in digital filtering and for his help and guidance in doing the research and preparing the manuscript.

The author is particularly indebted to the Pan American Petroleum Corporation, the Phillips Petroleum Company, and the Cities Service Oil Company whose financial aid made this thesis possible.
The latest technological advance in geophysics has probably been the application of digital computers to geophysical problems. Computers are being used not only for routine data reduction, but of even greater importance, they are being used to gain insight into the basic problems of geophysics and to develop new techniques for solving the problems. One of the main tools is the digital filter. Practically all filtering in the past was done by electronic circuits and more recently, by magnetic delay lines. Digital filters, with their speed, flexibility, and ability to represent physically nonrealizable functions, are rapidly superseding these older techniques.

There are many types of discrete time domain filters: for example, bandpass filters, inverse filters, notch or bandstop filters, multichannel filters, time-varying
filters, and compensation filters. Most of these filters can be specified in the frequency domain in terms of their amplitude and phase. An inverse Fourier transform produces a time function which can be sampled to give a discrete filter. The other major technique for obtaining discrete filters, with either the time or frequency domain as the starting points, involves characterizing some measure of error between a desired response or output of a filter and that response or output actually obtained. The discrete filter is then determined that minimizes this error. The error measure most frequently used, chiefly because of its mathematical simplicity, is the mean square error or total square error. Most so-called optimum filters are determined by this technique.

The purpose of this thesis is to study the design of digital filters using a weighted square error technique. The square error method is used for mathematical ease, and the weighting is added to provide some control over the nature and occurrence of the error. It is assumed that the desired filter will be specified in the frequency domain. The objective is to determine a discrete time-domain filter with specified length and sample interval
and with amplitude and phase characteristics which approximate the specified characteristics in the sense that the total weighted square error is a minimum.

The first chapter of the thesis is a review of discrete analysis and a discussion of factors causing errors in discrete filters. The second chapter is a review of the general filtering problem and of some of the most frequently used methods for designing filters. Error criteria and the characterization and minimization of the weighted square error are described in the third chapter. The computational results are analyzed and discussed in the fourth chapter. Because one of the major design factors is the length or number of points in the filter, much of the analysis of errors and error weighting has been done in relation to filter length.

It is believed that the simple cases considered are adequate to study error behavior, to check the theory of weighted error design, and to determine the effects of error weighting. Illustration by means of simple cases -- namely, several basic, phase-distortionless filters with a 41-point maximum length -- is economically desirable, for it makes the problem feasible on the School's CDC-8090 computer.
BACKGROUND

Fourier analysis and convolution techniques for continuous functions form the basis for most discussions on filter theory and design. Since these subjects are well treated in several texts (Lee, 1960; Papoulis, 1962; Bracewell, 1965), no review will be given here. However, discrete analysis techniques as applied to Fourier transforms and convolutions will be reviewed, since these concepts, although treated in the texts, are less well-known yet essential to the topics discussed in this thesis.

Discrete Analysis and Sampling

To be processed on a digital computer, continuous data must be handled in discrete pieces. The process of choosing values of the data at certain points is referred to as sampling. Since only finite numbers of discrete points can be considered, generally the data must be truncated. The effects of sampling and truncation are
two of the principal factors governing error in the application of Fourier techniques to data analysis and in the design and implementation of discrete filters.

A basic concept in discrete analysis is that of the impulse symbol. The impulse symbol is not a true function, and justification of its properties mathematically is difficult. For persons interested, advanced treatments are given in Lighthill (1958), Gelfand and Shilov (1964), Bremermann (1965), and Zemanian (1965); and more elementary discussions are given in Aseltine (1958), Papoulis (1962), Gupta (1964), and Bracewell (1965).

A common notation and definition of the impulse symbol follows.

\[ \delta(t) = 0 \quad t \neq 0 \]
\[ \int_{-\infty}^{\infty} \delta(t) \, dt = 1 \quad (1) \]

The most common meaning and usage of the impulse symbol is associated with the integral

\[ \int_{-\infty}^{\infty} \delta(t) \, h(t) \, dt = h(0) \quad (2) \]

Integration of an ordinary function with the impulse sifts out the value of the ordinary function at \( t = 0 \).
In the more general case
\[ \int_{-\infty}^{\infty} \delta(t-t_n)h(t)\,dt = h(t_n). \quad (3) \]

In equations 1, 2, and 3, t is used as the independent variable since in many applications this is the role played by time. The association of t with time is not essential; however, t will be referred to as time throughout the rest of the thesis.

An important value of impulses lies in their use for representing sampled functions. Consider a finite sequence of unit impulses represented by the expression
\[ s(t) = \sum_{n=-K}^{L} \delta(t-t_n). \quad (4) \]

Figure 1. Finite sequence of impulses.

Equation 4 and figure 1 illustrate the general case of an unequal sampling interval and asymmetric sampling. Throughout the rest of the thesis a constant sampling
interval and symmetric sampling are used. Equation 4 can then be written

\[ s(t) = \sum_{n=-M}^{M} \delta(t-nAt), \]  

(5)

where \( \Delta t \) is the sampling interval.

The sequence \( s(t) \) can now be used as a sampling function. Multiplication of a continuous function by \( s(t) \) results in a sequence of impulses weighted by the amplitudes or values of the continuous function for \( t = nAt \). In equation form, the sampled function would be

\[ f_S(t) = \sum_{n=-M}^{M} f(nAt) \delta(t-nAt). \]  

(6)

See figure 2 for a graphical illustration of the sampling process.

**Discrete Fourier Transforms**

The Fourier transform of a discrete function can be written

\[ F_S(\omega) = \int_{-\infty}^{\infty} f_S(t)e^{-j\omega t} \, dt \]

\[ = \int_{-\infty}^{\infty} \sum_{n=-M}^{M} f(nAt) \delta(t-nAt)e^{-j\omega t} \, dt \]

\[ = \sum_{n=-M}^{M} f(nAt) \int_{-\infty}^{\infty} \delta(t-nAt)e^{-j\omega t} \, dt \]  

(7)
By the sifting property of the impulse symbol,
\[ F_S(\omega) = \sum_{n=-M}^{M} f(n\Delta t)e^{-j\omega n\Delta t} . \]  
(8)

The transform of a sampled function, unlike the transform of a continuous function, is a periodic function.

\[ F_S(\omega+k \frac{2\pi}{\Delta t}) = \sum_{n=-M}^{M} f(n\Delta t)e^{-i(\omega+k \frac{2\pi}{\Delta t})n\Delta t} \]

\[ = \sum_{n=-M}^{M} f(n\Delta t)e^{-i\omega n\Delta t} e^{-i\frac{2\pi}{\Delta t}k} , \]
but for integer \( k, e^{-i k 2\pi} = 1; \) hence

\[
F_S(\omega + k \frac{2\pi}{\Delta t}) = F_S(\omega).
\] (9)

Equation 9 is the definition of periodicity with period \( \frac{2\pi}{\Delta t} \). The periodic property of the spectra of equispaced \( \Delta t \) sampled functions will be brought out in more detail in the section "Sampling and Truncation Effects."

**Discrete Convolution**

Convolution of a sampled function with a continuous function yields a continuous function.

\[
h(t) = \int_{-\infty}^{\infty} f_S(\tau) g(t-\tau) d\tau
\]

\[
= \int_{-\infty}^{\infty} \sum_{n=-M}^{M} f(n\Delta\tau) \delta(\tau-n\Delta\tau) g(t-\tau) d\tau
\]

\[
= \sum_{n=-M}^{M} f(n\Delta\tau) \int_{-\infty}^{\infty} \delta(\tau-n\Delta\tau) g(t-\tau) d\tau
\]

By the sifting property then

\[
h(t) = \sum_{n=-M}^{M} f(n\Delta\tau) g(t-n\Delta\tau).
\] (10)

This result may be used to illustrate the superposition property of the convolution integral. As illustrated in
Figure 3, equation 10 can be viewed as the superposition of \((2M+1)\) shifted and weighted \(g(t)\)'s. The weights are the values of \(f(t)\) at the points \(t = n\Delta t\).

![Diagram of superposition property of convolution](image)

Figure 3. Superposition property of convolution.

The convolution of two discrete functions follows in a manner similar to the previous case. Instead of the superposition of a number of continuous functions, it would be the superposition of a number of weighted discrete functions. The multiplication of equation 10 by another sampling function would yield the discrete convolution. For either type of convolution, as \(n \to \infty\)
and $\Delta t \to 0$ the single or double summation would converge to the convolution integral. For a detailed discussion on convolutions see Bracewell (1965, chap. 3).

**Sampling and Truncation Effects**

The periodic nature of spectra computed from sampled data can be shown in several ways. The following method is particularly useful, since it illustrates not only sampling effects but truncation effects in a single expression.

In equation 5 and figure 2 it was shown how a continuous function could be sampled by multiplication with a sampling function. Since multiplication in the time domain is equivalent to convolution in the frequency domain, the transform of the sampled function will equal the convolution of the transform of the continuous function with the transform of the sampling function.

The transform of the sampling function, equation 5, is

$$S(\omega) = \int_{-\infty}^{\infty} \sum_{n=-M}^{M} \delta(t-n\Delta t)e^{-j\omega t}dt$$

$$= \sum_{n=-M}^{M} \int_{-\infty}^{\infty} \delta(t-n\Delta t)e^{-j\omega t}dt$$

$$= \sum_{n=-M}^{M} e^{-j\omega n\Delta t}$$

(11)
The summation in equation 11 can be shown (Papoulis, 1962, p. 44) to be

\[ S(\omega) = \frac{\sin\left(M + \frac{1}{2}\right)\omega \Delta t}{\sin \frac{\omega \Delta t}{2}} \quad (12) \]

If the sampling function were not symmetrical, equation 12 would have real and imaginary parts and would be more complicated, but for use in this thesis the symmetrical case is adequate.

Equation 12 is periodic, with a period of \( \frac{2\pi}{\Delta t} \).

\[ S(\omega + k \frac{2\pi}{\Delta t}) = \frac{\sin \left[(M + \frac{1}{2})\omega \Delta t + (M + \frac{1}{2})k \cdot 2\pi\right]}{\sin \left[\frac{\omega \Delta t}{2} + k\pi\right]} \quad (13) \]

If \( k \) is incremented by 1, both numerator and denominator simply change sign; therefore the right member of equation 13 equals \( S(\omega) \). Using equation 12 and applying the convolution rule, the transform of the sampled function is

\[ F_S(\omega) = F(\omega) * \frac{\sin\left(M + \frac{1}{2}\right)\omega \Delta t}{\sin \frac{\omega \Delta t}{2}} \quad (14) \]

where \( F(\omega) \) is the exact, nonperiodic transform of the continuous function \( f(t) \) and where \( * \) denotes the convolution operation.
The shape and periodic nature of $S(ω)$ are shown in figure 4. The amplitude, the width of the large peaks, and the oscillation rate of $S(ω)$ are controlled by $(2M + 1)$, the total number of samples. Thus, the smaller features in figure 4 represent truncation effects. The sampling interval, $Δt$, controls the spacing or period of the large peaks. The convolution of the periodic function $S(ω)$ with the nonperiodic function $F(ω)$ results in a periodic function which is identical to equation 8.

![Figure 4. The Fourier transform of the sampling function.](image)

$ω = \frac{2\pi}{(2M+1)Δt}$

Sampling and truncation have caused two problems in obtaining a good estimate of the transform of a function. Sampling causes the spectrum to repeat, and truncation causes smearing and oscillations in the spectrum. An illustration of these effects appears in figure 5.
It can be shown (Papoulis, 1962, p. 45) that as $M$ becomes very large, $S(u \Delta t)$ approaches an infinite sequence of impulses spaced $\frac{2\pi}{\Delta t}$. Increasing $M$ is equivalent to removing the truncation effects and leaving only the sampling effects.

Convolution of some $F(u \Delta t)$ with the sequence of impulses spaced at $\frac{2\pi}{\Delta t}$ simply causes the spectrum $F(u \Delta t)$ to repeat at intervals of $\frac{2\pi}{\Delta t}$. The destructive effects of sampling occur when $\Delta t$ becomes so large that the repeating spectra begin to overlap. See figure 6 for a simple example.

This phenomenon is referred to as aliasing. The frequency $\frac{\pi}{\Delta t}$ is called the Nyquist frequency or folding frequency.
As $\Delta t$ becomes smaller, the Nyquist frequency increases and the aliasing effects are lessened. If $F(\omega)$ is band limited, that is

$$F(\omega) = 0 \text{ for } |\omega| > \omega_c$$

then aliasing may be eliminated by making $\Delta t \leq \pi/\omega_c$. Aliasing simply reflects the inability of a coarse sampling to resolve high frequencies, and can be compared to the stroboscopic effect observed in motion picture films.
FILTERING CONCEPTS AND THESIS OBJECTIVES

The purpose of this chapter is to outline the general filtering problem and to state the objectives of the thesis. Topics considered in this chapter are: filtering concepts, digital filters, time-frequency relationships for filters, computation of digital filters, and thesis objectives.

Filtering Concepts

Although a comprehensive review of filter theory and design is beyond the scope of this thesis, a comprehensive set of references pertaining to filter theory, design, and computation for geophysical applications has been included. Those interested in specific filters or filter applications should consult these references.

Generally, data are contaminated by so-called "noise". The term "noise", as used here, means the undesirable portion of the data. Noise may or may not be coherent with respect to desirable or information bearing components of the data, usually called signals. A typical reflection
seismogram is an example of data with both coherent and non-coherent noise superimposed on the signals which are reflections from the subsurface strata.

Because of noise, some type of signal enhancement is a requisite for the proper interpretation of geophysical data. One scheme which has been found effective for signal enhancement is linear filtering to suppress noise. Linear filtering is the process of convolving data with an appropriate operator called a filter. The appropriate filter depends on the specific problem at hand.

When a filtering application is defined by preliminary analysis of the data, filters are usually designed which can be routinely applied to all the data of a similar nature. In other words, a certain suite of filters are often designed for a particular class of problems. For example, a suite of lowpass, highpass, bandpass, and notch filters with various characteristics are conventionally used for the rejection of noise. A filter that best rejects noise for certain data can be chosen from the suite, usually by trying until one is found that gives acceptable results. On the other hand, more specialized filters must often be designed for specific applications and cannot be used for a class of problems.

The first step in filter design is to estimate the
signal and noise content of the data. Using this information, an appropriate filter can usually be designed accordingly. Filters may be both designed and applied in either the time domain or frequency domain. The insight and flexibility gained by designing a filter in the frequency domain makes the frequency-domain approach the most desirable for many problems. For application to the data though, the ease and speed in which convolution may be performed makes time-domain filtering the most practical and economical. Thus once the frequency domain characteristics are specified, the major objective is to determine the time function or impulse response whose frequency characteristics approximate or match those specified. The problem of obtaining the discrete time function from specified frequency characteristics is the major topic of this thesis.

Discrete or Digital Filters

The use of digital computers for processing data is quite common at the present time and, no doubt, will increase. In order to be processed on a digital computer, both filters and data must be handled in discrete pieces. In other words, the continuous functions representing both the filter and the data must be sampled as discussed under "Background." The effects of sampling are very important in the design and application of a discrete filter.
As opposed to an analog representation of a filter, the digital filter is simply a sequence of numbers ordered in time. It can be represented mathematically as a sequence of weighted impulses.

\[
f_S(t) = \sum_{n=-J}^{L} f_n \delta(t-n\Delta t)
\]

(15)

The \( f_i \) are referred to as the filter weights. The numbers \( J, L, \) and \( \Delta t \) define the range of the filter over time.

Once the sampling interval \( \Delta t \) is defined, the filter weights can be identified by integer indices. The term \( N \)-point filter refers to a digital filter with a total of \( N \) filter weights. Figure 7 shows a 7-point digital filter. The time duration of an \( N \)-point filter is \( (N-1)\Delta t \).

![Figure 7. Seven-point digital filter.](image)
The amplitude and phase response of a digital filter is given exactly by the Fourier transform of equation 15, namely,

\[ F_s(w) = \sum_{n=-N}^{L} f_n e^{-jwn\Delta t} \quad (16) \]

The properties of equation 16 are discussed in chapter 1 of this thesis.

For the application of a digital filter to data, the sampling interval of the filter must be the same, or some integral multiple of the data's sampling interval. No comment on the mechanics of performing discrete convolutions will be given here, but for those readers interested in the details, Domenico (1965), Jones and others (1955), Robinson and Treitel (1964), Smith (1958), and Treitel and Robinson (1964) review and explain the techniques of filtering with discrete operators.

The number of multiplications involved when performing a discrete convolution on a computer is proportional to the number of points in the filter. When large amounts of data are filtered, filter length becomes an important economic factor. For this reason, it is desirable to design digital filters so that the number of points necessary to realize desirable frequency characteristics is minimum.
Time-Frequency Filter Relationships

Some general relationships between functions in the time domain and their Fourier transforms are described in the following paragraphs. These rules and relationships are quite useful because they can be used to predict the gross characteristics of a filter's time-domain form.

As a general rule, the time duration of a filter varies inversely with the width of the amplitude spectrum. Hence, a very narrow amplitude spectrum implies a filter with a long time duration. If the phase response is zero or is a linear function of frequency with intercept at the origin, the filter will be symmetric about some time reference. This becomes important when computing filters since for linear phase with zero intercept only half the filter need be computed.

The location of the passband in the frequency domain determines the basic oscillation rate of the time-domain filter. A passband centered at high frequencies has a time domain representation which oscillates at a high frequency. Basically a time-domain filter will consist of an envelope which is the inverse transform of the response centered at zero frequency and a modulation factor which is governed by the center frequency of the filter. See figure 8.

The shape of the amplitude response of a filter
influences the time-domain shape a great deal. For example, responses with abrupt changes in slope are more difficult to define with a time-domain function of finite length than those with smooth changes in shape. This is analogous to a function of time with abrupt changes in slope which require more high-frequency components to synthesize the abrupt changes. This effect is referred to as Gibb's phenomenon.

![Diagram](image)

**Figure 8.** Filter breakdown in the frequency and time domain.
The previous discussion covered only the very general relationships between time and frequency functions. Both Papoulis (1962) and Bracewell (1965) give well-illustrated and detailed discussions of these relationships.

**Discrete Filter Computation**

Computing time-domain digital filters from characteristics called for in the design specifications can be done in several ways. Filter specifications given in the time domain usually lead to a computational technique based on the solution of a set of simultaneous equation. Filters computed in such a manner are usually called optimum or least squares filters. The computation of optimum discrete filters will not be discussed here; the reader interested in this type of filter should consult the references (Bode and others, 1950; Claerbout and Robinson, 1964; Ford and Hearne, 1965; Foster and others, 1962; Rice, 1962; Robinson, 1957, 1963). If filter specifications are given in terms of amplitude and phase characteristics, two routes for obtaining the discrete time-domain filter may be followed.

The most direct method to obtain a discrete filter from specified frequency characteristics is to perform an inverse Fourier transform. The impulse response thus obtained can be sampled to get the discrete filter. The
inverse transformation should be done analytically if possible; otherwise a numerical transform must be used. Practical considerations make it necessary to truncate the impulse response and so the resulting discrete filter does not have the specified frequency response. A Fourier transformation of the truncated filter is necessary to determine how well the actual characteristics of the truncated filter approximate design specifications. Various truncation lengths are tried until an acceptable filter is obtained. This technique for obtaining discrete filters is probably the most commonly used.

Another technique to determine the discrete filter from frequency-domain specifications is to characterize some measure of the error between the specified filter characteristics and the actual characteristics of the finite-length (truncated) discrete filter. A discrete filter is then determined which minimizes this error. It will be shown in chapter 3 that the discrete filters obtained for a square error minimization process are sometimes the same as those obtained by sampling the continuous impulse response. One of the problems with this technique is selection of a suitable error measure. The major part of the thesis deals with the determination of discrete filters by considering the error between specified and true characteristics.
Thesis Objectives

The purpose of this thesis is to investigate a weighted square error measure for discrete filter design in the frequency domain. The reasons for choosing the weighted square error measure are discussed in the next chapter.

Specifically, the objectives of the thesis are:

1. To determine an N-point time-domain filter which minimizes the weighted square error between specified amplitude and phase characteristics and the actual characteristics of the discrete filter.

2. To examine the error as a function of N when the square error is uniformly weighted; i.e., when all portions of the frequency spectrum below the Nyquist frequency are given equal importance.

3. To examine the effects of various weighting configurations in discrete filter design.
Before the main problem of minimizing the weighted square error is attacked, a review of various error criteria will be given.

Error Criteria

The first step in developing an error measure is to characterize the error. This is quite straightforward for discrete filters. As in the last section, a discrete filter can be denoted by

\[ f_N(t) = \sum_{n=-J}^{L} f_n \delta(t-nA_t). \]  

(17)

In equation 17, \( \Delta t \) is the sampling interval and \( f_n \) is the \( n \)th filter weight. The total number of weights is \( N \) or \( (J+L+1) \).

The amplitude and phase response of the discrete filter are given by
Let the specified amplitude and phase response be given by
\[ F(\omega) = A(\omega)e^{-j\phi(\omega)} \]  
(19)
where \( A(\omega) \) is the amplitude response and \( \phi(\omega) \) is the phase response.

The error between the specified response and that obtained with the filter weights can be written
\[ \varepsilon(\omega) = F(\omega) - F_s(\omega) \]  
(20)
or
\[ \varepsilon(\omega) = F(\omega) - \sum_{n=-J}^{L} f_n e^{-j\omega_n t}. \]  
(21)

The error \( \varepsilon(\omega) \) can be minimized with respect to the filter weights in some sense. What measure of error to minimize is the problem.

Error measures which are independent of time or frequency are referred to as norms. The name "distance function" is used sometimes since some form of the "distance" between the true and approximated response is considered. Rice (1964) gives a detailed discussion on error norms and their properties. A commonly used set of norms are
\[ \rho_{pc} \triangleq \int_\Omega |\varepsilon(\omega)|^p \, d\omega \quad \text{and} \quad (22) \]

\[ \rho_{pd} \triangleq \sum_{i=1}^Q |\varepsilon(\omega_i)|^p \quad (23) \]

Equations 22 and 23 define the norms of the continuous and discrete cases, respectively. The limits of integration or summation depend on the problem being considered.

The value of \( p \) determines the relative importance of errors of different magnitudes. Let the factor \( EI_p \) be defined as the error importance.

\[ EI_p \triangleq \frac{|\varepsilon(\omega)|^p}{|\varepsilon(\omega)|} \]

\[ \triangleq |\varepsilon(\omega)|^{p-1} \]

For \( p=1 \), all errors have equal importance. As \( p \) increases, the larger errors have more and more importance. Figure 9 illustrates the effects of several \( p \)'s on error importance. As \( p \) increases, the minimization of the norm is completed with less and less regard to small errors. In the limit as \( p \) approaches infinity, the largest error is the only error of importance in the distance function or norm. Hence, the maximum error is minimized. This is the Chebyshev error norm, or minimax principle.
Depending on the problem being considered, one of these norms may be appropriate. For example, the Chebyshev norm is used in approximation and interpolation of functions because an upper bound on the error can sometimes be established. Unfortunately, although a particular p may be desirable, mathematical difficulties in finding the filter weights or parameters of the problem which minimize the norm often present a formidable obstacle.

![Diagram of error importance $E^{*}_p$ vs error magnitude for different values of p]

Figure 9. Error importance $E^{*}_p$ vs error magnitude.

From all standpoints, the $p = 2$ or least squares norm is the easiest to use. The mathematical characterization of the least squares criterion is straightforward, and the minimization process gives a unique and explicit result. The parameters minimizing the norm must satisfy
a set of linear equations. Other norms can be characterized and minimized, but the parameters appear nonlinearly in the equations. Practically all optimum processes in the past were designed with the least squares norm since it is the only technique feasible without high-speed computing facilities. Now that computers are readily available, some of the other norms are being used.

**Weighted Square Error**

Error norms can be generalized by the addition of a weighting function \( W(\omega) \), i.e.

\[
\bar{\rho}_{pc} \triangleq \int_{\Omega} W(\omega) |\epsilon(\omega)|^p \, d\omega, \tag{24}
\]

or for the discrete case,

\[
\bar{\rho}_{pd} \triangleq \sum_{i=1}^{Q} W(\omega_i) |\epsilon(\omega_i)|^p \tag{25}
\]

In most cases, the addition of the weighting function has little effect on the minimization process and the computation of optimum approximations. For this reason, the addition of a weighting function to a practical error norm gives a more flexible technique.

Rice (1964) discusses the properties of weighted error norms. He states that given an optimum approximation
for a certain \( p \), a weighting function can be chosen such that the same optimum approximation may be obtained using another value of \( p \). Obtaining approximations for various \( p \) using the weighting technique is difficult to apply in practice because the connection between the approximation obtained and the possible norm used is obscure. At any rate, weighting functions provide some control over the error behavior and presumably have some useful applications.

The weighted square error norm is used in this thesis because of its mathematical simplicity and computational advantages. The weighting function is added to provide flexibility and to exercise control over the nature and occurrence of the error. No real attempt is made to achieve best filters for another \( p \) using this technique; the main objective is to see whether any improvement in filter design can be obtained by application of various weighting functions.

The weighted square error norm for discrete filters can be written as

\[
\overline{\rho}_2 = \int_{\Omega} W(\omega) |\epsilon(\omega)|^2 \, dw, \tag{26}
\]
where $\varepsilon(\omega)$ is given by

$$
\varepsilon(\omega) = F(\omega) - \sum_{n=-J}^{L} f_n e^{-j\omega n\Delta t}.
$$

(21)

The weighting function $W(\omega)$ is assumed to be real and even. The problem now is to choose limits of integration for equation 26 and proceed with the minimization of $\bar{\rho}_2$.

**Minimization of the Weighted Square Error**

The minimization process which follows is completely general. Any filter specified in the frequency domain can be determined by use of this technique. In addition to the amplitude and phase response, it is necessary to specify the sampling interval and length of the filter.

The Nyquist frequency, $\pi/\Delta t$, was chosen for the limits of integration in equation 26. Consideration of errors outside this range seems meaningless since the discrete filter's response simply repeats. For convenience in the rest of the thesis, the Nyquist frequency will be denoted by $\mathcal{N}$.

Equation 26 can be written as

$$
\bar{\rho}_2 = 2 \int_{0}^{\mathcal{N}} W(\omega) \varepsilon(\omega) \varepsilon^*(\omega) d\omega.
$$

(27)
Substituting equation 20 into 27 gives

\[ \bar{\rho}_2 = 2 \int_0^\infty W(\omega) \left[ F(\omega) - F_S(\omega) \right] \left[ F^*(\omega) - F_{S}^*(\omega) \right] d\omega. \] (28)

Substituting equation 18 into 28, one gets

\[ \bar{\rho}_2 = 2 \int_0^\infty W(\omega) \left[ F(\omega) - \sum_{n=-J}^{L} f_n e^{-j\omega n\Delta t} \right]. \]

\[ \left[ F^*(\omega) - \sum_{n=-J}^{L} f_n e^{j\omega n\Delta t} \right] d\omega \] (29)

The value of \( \bar{\rho}_2 \) has no maximum since one can choose the filter weights \( f_n \) to make the approximation as poor as desired. Since \( \bar{\rho}_2 \) is a differentiable function of the \( f_n \), the minimum must occur where all of the partial derivatives

\[ \frac{\partial \bar{\rho}_2}{\partial f_i} \quad i = -J, -J+1, \ldots, 0, \ldots, L-1, L \]

are zero. Differentiating \( \bar{\rho}_2 \) with respect to \( f_i \) gives
\[
\frac{\partial^2 \psi}{\partial f_i^2} = 2 \int_0^\infty W(\omega) \left[ F(\omega) - \sum_{n=-J}^{L} f_n e^{-j\omega n \Delta t} \right] \left[ -e^{j\omega t} \right] d\omega \\
+ 2 \int_0^\infty W(\omega) \left[ F^*(\omega) - \sum_{n=-J}^{L} f_n e^{j\omega n \Delta t} \right] \left[ -e^{-j\omega t} \right] d\omega,
\]
for \( i = -J, -J+1, \ldots, 0, \ldots, L-1, L \). (30)

By expanding and simplifying equation 30, it can be seen that
\[
\int_0^\infty W(\omega) F(\omega) e^{j\omega t} d\omega + \int_0^\infty W(\omega) F^*(\omega) e^{-j\omega t} d\omega = \\
\int_0^\infty W(\omega) \left( \sum_{n=-J}^{L} f_n e^{-j\omega n \Delta t} \right) e^{j\omega t} d\omega + \\
\int_0^\infty W(\omega) \left( \sum_{n=-J}^{L} f_n e^{j\omega n \Delta t} \right) e^{-j\omega t} d\omega
\]
for \( i = -J, -J+1, \ldots, 0, \ldots, L-1, L \). (31)

The first term of the left member of equation 31 is the complex conjugate of the second term. This holds also for the first and second terms of the right member. Since the sum of a complex number and its conjugate equals twice the real part, equation 31 can be written as
\[
2 \text{Re} \int_0^\infty W(\omega) F(\omega) e^{j\omega t} d\omega =
\]
Rearranging and simplifying equation 32,

\[ 2 \text{Re} \int_{0}^{\infty} W(\omega) \sum_{n=-J}^{L} f_n e^{-jwn\Delta t} e^{j\omega_{\Delta t}} d\omega = \sum_{n=-J}^{L} f_n \text{Re} \int_{0}^{\infty} W(\omega) e^{j\omega(i-n)\Delta t} d\omega \]

for \( i = -J, -J+1, \ldots, 0, \ldots, L-1, L \). \hspace{1cm} (33)

Both integrals in equation 33 are in the form of inverse Fourier transforms for real time functions. For example, if \( G(\omega) \) is the complex Fourier transform of some real \( g(t) \) then

\[ g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(\omega) e^{j\omega t} d\omega \]

\[ = \text{Re} \frac{1}{\pi} \int_{0}^{\infty} G(\omega) e^{j\omega t} d\omega \] \hspace{1cm} (34)

(Papoulis, 1962, p. 11). From equation 34 then,

\[ h(t)|_{t=i\Delta t} = \text{Re} \int_{0}^{\infty} W(\omega) F(\omega) e^{j\omega_{\Delta t}} d\omega. \] \hspace{1cm} (35)

Since the limits of integration in equation 33 are not infinite, the truncation theorem applies and
\[ h_a(t) \bigg|_{t=i\Delta t} = \text{Re} \int_0^{\eta} W(\omega) F(\omega) e^{j\omega i\Delta t} \, d\omega \]

\[ = h(t) \ast \frac{\sin \eta t}{t} \bigg|_{t=i\Delta t} \quad (36) \]

for the left member. In the same manner, the coefficients of \( f_n \) in the right member of equation 33 can be expressed as

\[ w_a(t) \bigg|_{t=(i-n)\Delta t} = \text{Re} \int_0^{\eta} W(\omega)e^{j\omega (i-n)\Delta t} \, d\omega \]

\[ = w(t) \ast \frac{\sin \eta t}{t} \bigg|_{t=(i-n)\Delta t} \quad (37) \]

where

\[ w(t) \bigg|_{t=(i-n)\Delta t} = \text{Re} \int_0^{\infty} W(\omega)e^{j\omega (i-n)\Delta t} \, d\omega \quad (38) \]

In practically all cases, \( F(\omega) \) and \( W(\omega) \) will be bandlimited with respect to \( \eta \), i.e.

\[ F(\omega) = 0, \ W(\omega) = 0 \text{ for } |\omega| > \eta. \]

Assuming bandlimited functions, then

\[ h_a(t) = h(t) \text{ and } w_a(t) = w(t). \quad (39) \]
Equation 33 can now be written

\[ h_a(t) \bigg|_{t=i\Delta t} = \sum_{n=-J}^{L} f_n w_a(t) \bigg|_{t=(i-n)\Delta t} \] (40)

for \( i = -J, -J+1, \ldots, 0, \ldots, L-1, L. \)

For the case of bandlimited functions, equations 33 would be

\[ h(t) \bigg|_{t=i\Delta t} = \sum_{n=-J}^{L} f_n w(t) \bigg|_{t=(i-n)\Delta t} \] (41)

for \( i = -J, -J+1, \ldots, 0, \ldots, L-1, L. \)

If the value of a continuous function at \( t=i\Delta t \) is denoted by the subscript \( i \), equation 41 becomes

\[ h_i = \sum_{n=-J}^{L} f_n w_{i-n} \quad i=-J, \ldots, 0, \ldots, L \] (42)

Equation 40 and 42 defines a set of \((J+L+1)\) linear simultaneous equations.

**Normal Equations**

In matrix form, equation 42 would appear as

\[
\begin{bmatrix}
  w_0 & w_{-1} & w_{-2} & \cdots & w_{-J-L} \\
  w_1 & w_0 & w_{-1} & \cdots & \\
  w_2 & w_1 & w_0 & \cdots & \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  w_{J+L} & \cdots & w_1 & w_0 \\
\end{bmatrix}
\begin{bmatrix}
  f_{-J} \\
  f_{-J+1} \\
  \vdots \\
  f_0 \\
  f_{L-1} \\
\end{bmatrix}
= 
\begin{bmatrix}
  h_{-J} \\
  h_{-J+1} \\
  \vdots \\
  h_0 \\
  h_{L-1} \\
\end{bmatrix}
\]
If the coefficient matrix is nonsingular, then the solution of the set of \((J+L+1)\) equations above gives the time domain filter weights \(f_i\) whose transform approximates that specified in a weighted least squares sense. Since \(W(w)\) is real and even, its inverse transform is real and even. Hence, \(w_i = w_{-i}\) and the coefficient matrix is symmetric. Also, if \(W(w)\) is positive, the maximum value of \(w(t)\) occurs at \(t = 0\) so that the largest terms fall on the main diagonal. These factors make the solution of equations 40 and 42 somewhat simpler.

If the specified filter has a zero phase, the time domain representation is symmetric and \(f_{-i} = f_i\). It is necessary then to solve only a \(\frac{N+1}{2}\) order system where \(N = J+L+1\), as opposed to a \(N\)th order system, for an \(N\)-point filter. In matrix form the \(\frac{N+1}{2}\) order system would appear as

\[
\begin{bmatrix}
  (w_0 + w_{-J-L}) & (w_{-1} + w_{-J-L+1}) & \cdots & (w_{-J-L+1})
  \\
  (w_1 + w_{-J-L+1}) & & & \\
  \cdot & & & \\
  \cdot & & & \\
  (w_{J+L} + w_{-J-L}) & \cdots & \cdots & \cdots & (w_0)
\end{bmatrix}
\begin{bmatrix}
  f_{-J} \\
  f_{-J+1} \\
  \vdots \\
  f_{-1} \\
  f_0
\end{bmatrix} =
\begin{bmatrix}
  h_{-J} \\
  h_{-J+1} \\
  \vdots \\
  h_{-1} \\
  h_0
\end{bmatrix}
\]
A special case of much interest occurs when the weighting function \( W(u) \) is unity. Letting \( W(u) \) be unity corresponds to a regular least squares criterion. When this occurs,

\[
w(t) = \frac{\sin \int t}{t}.
\]  

Substituting \( \pi/\Delta t \) for \( \int \) and considering values of \( t = (i-n)\Delta t \), one gets

\[
w_{i-n} = \frac{\sin(\pi/\Delta t)(i-n)\Delta t}{(i-n)\Delta t} = \frac{\sin \pi(i-n)}{(i-n)\Delta t}
\]

Since \( \sin(i-n)\pi \) is zero, \( w_{i-n} \) is zero for all \( i \neq n \).

For \( i=n \), l'Hospital's rule can be applied:

\[
\lim_{i \to n} \frac{\sin \pi(i-n)}{(i-n)\Delta t} = \lim_{i \to n} \frac{\pi \cos \pi(i-n)}{\Delta t}
\]

\[
\lim_{i \to n} \frac{\pi \cos \pi(i-n)}{\Delta t} = \frac{\pi}{\Delta t}
\]
So, for \( i = n \), \( w_{i-n} = \pi/\Delta t \) and for \( i \neq n \), \( w_{i-n} \) is zero, and equation 42 would become

\[
\hat{h}_i = f_i \pi/\Delta t \text{ for } i = -J, -J+1, \ldots, 0, \ldots, L \quad (46)
\]

and the filter weights would be

\[
f_i = \frac{\Delta t}{\pi} h_i \text{ for } i = -J, \ldots, 0, \ldots, L. \quad (47)
\]

Since \( W(\omega) \) was unity for all frequencies up to \( \pi/\Delta t \), \( h_i \) simply represents the value of \( h(t) \), the inverse transform of the specified response, at \( i\Delta t \). The above arguments hold for equation 40 in a similar manner.

The preceding derivations have shown that the optimum \( N \)-point filter in the weighted square error sense is determined by solving an \( N \)th order set of simultaneous equations. The coefficient matrix is made up of the inverse transform of the weighting function, and the constant vector is made up of the inverse transform of the product of the weighting function and the specified response. Both of these inverse transforms are integrated up to the Nyquist frequency.

For the special case of unit error weighting, the discrete filter with minimum square error is obtained by sampling the continuous time function or impulse response of the specified filter with a sequence of unit impulses.
For filter specifications with energy at frequencies exceeding the Nyquist frequency, the continuous function to be sampled must represent the specified response only up to the Nyquist frequency; otherwise the square error will not be a minimum. The theory provides for the prevention of aliasing errors by truncating any response past the Nyquist frequency.
The purpose of this chapter is to discuss the application of error weighting theory to the design of some simple filters. Upon initiation of this part of the study it was thought that a few simple cases would suffice to determine the value of error weighting, but with the computation of several cases it appeared that a more extensive investigation would be necessary to determine whether or not error weighting is of value. The discussion that follows is concerned chiefly with the general effects caused by error weighting.

The chapter is divided into three sections: Computing Weighted Error Filters, Uniform Error Weighting, and Nonuniform Error Weighting. The first section includes a description of the steps necessary to compute a weighted error filter and a discussion on the accuracy and form of the computer programs. The second section is a fairly detailed discussion of unweighted or uniformly weighted
error filters. The square error of some simple unweighted discrete filters is studied as a function of filter length N. The third section on Nonuniform Error Weighting illustrates the general effects of various weighting functions.

Computing Weighted Error Filters

The following material is a brief description of the techniques used to compute weighted error filters. Details on the structure and accuracy of the CDC-8090 Fortran computer programs are given in Appendix I and Appendix II.

Flow Chart: Two inverse Fourier transforms and the solution to a set of simultaneous equations are required to compute a weighted error filter. For analysis of the computed filters, a Fourier transform is necessary. Computer programs for the School's CDC-8090 digital computer were developed to perform the above tasks. Programs to compute various functions of the error between the frequency response of the specified filter and the response of the computed filter were also written.

A flow chart of the operations involved in computing and analyzing a weighted error filter is shown in figure 10. The symbols in the flow chart correspond to those used in chapter 3. The desired response and the error weights are generated in the first two steps and multiplied to
Figure 10. Generalized flow chart for weighted error filter design.
produce the weighted response in the third step. The inverse Fourier transforms of the weighting function and weighted response are computed in step 4. Using the results of step 4, the simultaneous equations are set up and solved with a Gauss-elimination technique to produce the filter weights $f_i$. The computation of the actual frequency characteristics of the filter weights and the comparison with the specified characteristics are the final two steps.

For the case of uniform error weighting, most of the steps described above are unnecessary. Since the filters used in this thesis can be transformed analytically, their impulse responses can be determined and sampled to get the unweighted discrete filter. The unweighted filter computation is indicated in step la on the flow chart. The discrete filter is transformed and its response compared to that specified, as in the general case.

Several types of error were computed and listed for each discrete filter's amplitude response. The total error and total square error were computed by summing the absolute values of the errors and the squares of the errors, respectively. Mean deviation, standard deviation, and variance were also computed. The percent error was computed by referring total error to the area of the specified response, and percent square error was computed by referring
total square error to the area of the square of the specified response. All these errors were computed cumulatively over four ranges on a dimensionless frequency scale.

The system of programs is not completely general, because of limitations on computer storage. Only zero-phase filters of a 41-point maximum length can be computed. The solutions to sets of equations of order 21 and greater become questionable because of accumulated round-off errors, and two-dimensional systems of order 21 and greater take too much core storage in the computer. Since a zero-phase filter's time-domain form is even, the imaginary part of the Fourier transform is zero and a cosine transform suffices for both steps 4 and 6 in the flow chart. The numerical transform is used in step 4 because most weighted responses and weighting functions are too difficult to transform analytically.

Program Accuracy: Since the main subject of this study is error in discrete filters, some comments on the numerical accuracy of the various programs should be made. Most of the numerical results used in the thesis, particularly the filter responses, discrete filters and weighting functions, have a maximum which is about 1.0. For this reason, the number of places to the right of the decimal point provides a convenient measure of accuracy. In the
following discussion, the term "decimal places" is used in this sense to denote computational accuracy. Controlled test data have shown the cosine transform programs to be good to 4 decimal places. Aliasing errors are always a possibility, but close observance of all transforms done in the study indicated no serious cases of aliasing. The major concern regarding accuracy was the solution of the simultaneous equations.

Test systems of simultaneous equations similar to those used in determining weighted error filters were generated and solved with a Gauss-elimination technique. Accuracy to 5 decimal places and better was obtained in systems up to order 16 (a 31-point filter). Systems of order greater than 16 tested showed accuracy to 6 decimal places in some cases and to only 2 decimal places in other cases depending on the configuration of the matrices and solution vectors. The errors are significant only in the smaller terms of the solution and are not considered a serious drawback. Production of various weighted error filters gave reasonable results which seems to indicate that even the higher order systems (up to 21) are reliable. Any systems greater than order 21 would be questionable.

The overall accuracy in the complete system from filter specification to computation of filter response should be
good to 3 decimal places in the worst case. Most cases of interest fall in the range of the shorter filters, which are more accurate. It is possible to plot the data, at best, only to 3 decimal places.

**Dimensionless Frequency:** As shown on page 8, the transform of a discrete filter; i.e., a sequence of weighted impulses in time, is a continuous function of frequency in which \( f \) is always multiplied by the sample interval, \( \Delta t \). The transform can thus be thought of as a function of the dimensionless variable \( f\Delta t \). Where it is necessary to sample at discrete frequencies, the transform can be thought of as a function of the discrete parameter \( \Delta f\Delta t \).

As an example of the use of dimensionless frequency, consider a sequence of impulses whose transform is plotted on a scale in increments of \( \Delta f\Delta t = 0.002 \). If \( \Delta t \) is specified as 0.001 second, then \( \Delta f \) is 2 cycles per second. If \( \Delta t \) is 0.002 second, then \( \Delta f \) is 1 cycle per second. Conversely, \( \Delta f \) can be specified, in which case the value of \( \Delta t \) is fixed. Thinking in terms of the dimensionless parameter gives added flexibility to a single discrete filter-transform pair. The Nyquist frequency on the dimensionless scale always occurs at \( f\Delta t = 0.5 \). Since the Nyquist frequency is
\[ \omega_c = \frac{\pi}{\Delta t}, \]

it can be written

\[ 2\pi f_c = \frac{\pi}{\Delta t} \]

or

\[ f_c = \frac{1}{2\Delta t}. \]

Then

\[ f_c \Delta t = 0.5. \]

The discussion in Appendix I on cosine transforms illustrates why the dimensionless frequency concept is convenient for numerical transforms.

**Uniform Error Weighting**

The logical starting point for a study of weighted error filter design is the case of uniform error weighting. As derived in the section on the minimization of the weighted square error, the best discrete filter with uniform error weighting is obtained by sampling the inverse transform of the specified filter response. It is important to remember that the transform is truncated at the Nyquist frequency; i.e., only if the specified response does not exceed the Nyquist frequency, does the inverse transform represent the specified filter exactly. If a response is specified which exceeds the Nyquist frequency, then the time function to be sampled represents the truncated response, not the exact
response. This is the discrete filter whose square error is minimum over the frequency scale up to the Nyquist frequency.

Several simple lowpass and bandpass discrete filters for uniform weighting for \( N = 3 \) to \( N = 51 \) were generated at sampling intervals of \( \Delta t \), \( 2\Delta t \), and \( 3\Delta t \). The transforms for all \( N \) were computed and the errors determined. Filter responses specified were bandlimited and easily transformed analytically. The filters used are described in Appendix I. It would have been desirable to observe the error behavior for a larger variety of filters, but the amount of time available for computing and data analysis was limited. From the author's experience, the error behavior tends to be similar for all filters of the types being considered.

**General Error Characteristics:** The continuous impulse responses of four of the filters studied in this section are shown in figure 11. All filters discussed in this section are sampled at \( l\Delta t \). For later reference a set of Fourier-series kernels for various \( N \) are illustrated in figure 12. As derived in the background, the Fourier transform of a sampled and truncated impulse response is given by the convolution of the proper kernel with the specified or exact transform. The kernels represent the sampling and truncation effects and help provide insight into the error behavior.
Figure 11. Impulse responses of four filters with uniformly weighted error.
Figure 12. Normalized Fourier transforms of even unit sampling sequences for $N$ equal to 7, 15, 17, 21, 33, and 47 at a sampling interval of $1\Delta t$. 

Fourier-Series Kernel

$S(\omega) = \frac{\sin((M+1/2)\omega\Delta t)}{\sin(M\omega\Delta t/2)}$

$M = (N-1)/2$
It is important to remember that in the case where the filter response exceeds the Nyquist frequency, the kernel must be convolved with the truncated response to assure minimum square error over the range up to the Nyquist frequency. Although it is possible to compute discrete transforms by a convolution in the frequency domain such as mentioned above, the direct transform has been used throughout this thesis since it is faster.

Figures 13 through 16 are truncation studies showing the amplitude responses of a lowpass filter and a bandpass filter for various N and for a sampling interval of \( \Delta t \). The number of points, N, in the filters increases from top to bottom in the illustrations. The discrete filter corresponding to each amplitude curve is shown at the upper right. The top curves in figures 13 and 15 illustrate what happens when the impulse response is severely truncated by making N small. The amplitude spectrum of the short filter does not reach the specified spectrum in the passband and displays a broadening and overall distortion. The reject region contains peaks and nulls whose oscillation rate depends on N. For small N, these peaks are of sufficient amplitude to pass unwanted frequencies. Exactly what amplitude of the small peaks can be tolerated depends on the filtering problem.

As N increases, the passband amplitude comes up to the
Figure 13. Amplitude responses of a progressively truncated lowpass discrete filter sampled at $1\Delta t$. Discrete filter at the upper right and exact response indicated by thin lines in each plot.
Figure 14. Amplitude responses of a progressively truncated lowpass discrete filter sampled at $1\Delta t$. Discrete filter at the upper right and exact response indicated by thin lines in each plot.
Figure 15. Amplitude responses of a progressively truncated bandpass discrete filter sampled at $1\Delta t$. Discrete filter at the upper right and exact response indicated by thin lines in each plot.
Figure 16. Amplitude responses of a progressively truncated bandpass discrete filter sampled at $1\Delta t$. Discrete filter at the upper right and exact response indicated by thin lines in each plot.
specified level and then overshoots. The passband approaches a peak rather than a plateau as specified. The broadening of the passband is lessened, but still is significant. As the frequency of the oscillations in the reject region increases, their amplitude usually decreases. Figures 13 and 14 for $N = 13, 17$ and figures 15 and 16 for $N = 17, 21, 33$ illustrate these typical effects. In all cases, the largest errors occur at the corners of the specified response. Normally one would round these corners when designing filters, but for illustrative purposes angular breaks are more advantageous.

The large overshoot in the passband reaches a maximum and then decreases as $N$ is further increased. When the amplitude at the center of the peak approaches 1.0 with increasing $N$, smaller overshoots begin to appear on each side. The broadening of the spectrum practically disappears except near the corners, and the bumps in the reject region become small and oscillate rapidly. As $N$ increases, the oscillations in the amplitude spectrum become more rapid, but display amplitudes which are usually insignificant. When $N$ becomes sufficiently large, the oscillations start appearing in the flat passband of the filters. The origin of the oscillations is discussed later in the paragraphs on Fourier-series kernels. The filters in figures 13 through 16
are too narrow to show the oscillations in the pass region well, so some broadband filters, which appear in figure 17, are used to illustrate the effects. Increasing N further results in a slow convergence to the exact specification.

Naturally, the various effects described above occur at different N for different filter specifications, as would be anticipated from the associated impulse responses. Normally, the range of effects occurs at a smaller N for broadband filters than for narrowband filters. The behavior described in the preceding paragraphs is quite typical for all filters of the type examined here.

The effects observed in figures 13-17 result mainly from truncation. A comparison of the filter responses with the kernel functions (figure 12) for corresponding N and sample interval helps to explain the observed effects. The kernels for small N have a broad center and large, low-frequency oscillations. The reader should bear in mind that the kernel functions are periodic; however, since the kernels have been plotted on the same scale as the amplitude responses to make comparisons easier, this periodicity is not evident. The broad peak for small N is responsible for the broadening effect on the amplitude response. The oscillations after the main peak cause the similar oscillations in the spectrum of the discrete filter. A comparison of
Figure 17. Amplitude responses of some truncated broadband discrete filters sampled at $1\Delta t$. Discrete filter at upper right and specified response denoted by thin lines in each plot.
kernels and discrete filter responses shows that the oscillations have approximately the same periods. As \( N \) increases, the peaks of the kernels narrow, lessening smearing effects, and the rates of the oscillations increase while their amplitudes decrease. As \( N \) approaches infinity, the kernels approach a series of impulses whose convolution with the exact transform causes only repetition. Figure 18 shows plots of the first zero crossing of the kernel function vs \( N \) for two sampling intervals. The first zero crossing indicates half the width of the large peak of the kernels. As the curve shows, when a certain \( N \) is reached, say \( N = 41 \), the rate of change in the peaks width with \( N \) is very slow.

**Effect of Sampling Interval:** To study the effects of sample interval, the impulse responses of the filters (figure 11) were sampled at 2 and 3 times the basic interval \( \Delta t \). The filters were transformed with zeros in place of the missing samples. Replacing the samples by zeros causes the frequency responses to alias at 0.25 and 0.167 \( f_{\Delta t} \) for the case of sampling at 2\( \Delta t \) and 3\( \Delta t \), respectively.

Figures 19 and 20 show the effects of the sample interval for lowpass and bandpass filters when \( N \) is held constant. Thus these figures show the combined effects of sampling and truncation errors. The obvious improvement
First zero crossing of Fourier-series kernel vs \( N \) for \( 1\Delta t \) and \( 2\Delta t \)

\[
S(w) = \sin(M+\frac{1}{2})\frac{\sin(M\Delta t/2)}{\sin(N\Delta t/2)}
\]

\( M = (N-1)/2 \)

Figure 18. Fourier-series kernel width vs \( N \) for two sampling rates.
Figure 19. Amplitude responses of a lowpass discrete filter for \( N = 7 \) and sample rates of \( 1\Delta t \), \( 2\Delta t \), and \( 3\Delta t \). Discrete filter at upper right and exact response denoted by thin lines in each plot.
Figure 20. Amplitude responses of a bandpass discrete filter for $N = 15$ and sample rates of $1\Delta t$, $2\Delta t$, and $3\Delta t$. Discrete filter at upper right and exact response denoted by thin lines in each plot.
in the passband and reject region is caused by the increase in sample interval. Holding N constant and increasing sample interval results in a filter with greater time duration and less truncation. The improvement occurs at the expense of a large decrease in the length of the reject region since the Nyquist frequency decreases.

For contrast to the case of constant N and changing sample interval, figures 21 and 22 show lowpass and bandpass filter responses when the product of N and sample interval is a constant. In other words, the portion of the impulse response being sampled is held constant while both the sampling interval and N change. Over ranges of 0.0-0.25 \( f_\Delta t \) for filters sampled at \( 1_\Delta t \) and \( 2_\Delta t \) and 0.0-0.167 \( f_\Delta t \) for filters sampled at \( 1_\Delta t \), \( 2_\Delta t \) and \( 3_\Delta t \), the amplitude responses are practically identical. The percent square error of the curves varied no more than 0.02 for the comparable ranges. The only significant difference is the change in Nyquist frequency which produces the repetition of the responses at closer intervals.

The N-dependence of the amplitude characteristics for filters with sample intervals other than \( 1_\Delta t \) is the same as that described in the previous section on general error characteristics. As would be anticipated from the behavior of the Fourier-series kernel (equation 12) as a function of
Figure 21. Amplitude responses of a lowpass discrete filter holding time span fixed and varying N and sample rate. Discrete filter at upper right and exact response denoted by thin lines in each plot.
Figure 22. Amplitude responses of a bandpass discrete filter holding time span fixed and varying N and sample rate. Discrete filter at upper right and exact response denoted by thin lines in each plot.
Δt, the N-dependent truncation effects occur at different N as the sample interval changes. As the sampling interval is increased and N held constant, the large peaks of the kernel become narrower and move closer together, and the oscillation rate of the kernel increases. For example, if the sampling interval is doubled, the width and spacing of the large peaks are halved and the oscillation rate doubled. Except for the change in Nyquist frequency, convolution of the kernel at 2Δt with the specified response would produce characteristics similar to those at double the N for a sample interval of 1Δt. This point will be illustrated in the following section on error curves. Figure 18 gives an indication of the effect of doubling sample interval on the kernel width vs N.

**Error Curves and Time Duration:** To make a more quantitative analysis of the error, percent square error and percent error (defined on page 45) are plotted against N. These graphs give a quantitative and compact presentation of filter quality. Of equal importance, they provide insight into the behavior of discrete filters as the number of points change. Most of the curves to be discussed are percent square error vs N since a square error measure is used in the design; however several percent error curves are shown for contrast.
Figures 23 and 24 show the error curves for the lowpass and bandpass filters discussed in the last section. The top four curves in each figure are cumulative errors over the ranges 0-0.125, 0-0.25, 0-0.375 and 0-0.5 on the dimensionless scale. The bottom two curves represent cumulative errors over ranges of 0.0-0.125 and 0.0-0.25 Δt for filters sampled at 2Δt.

All the curves drop steeply for small N and then level out at some point. After this point there may be several steps downward to a point where the errors are of the order of 0.10 percent square error. The errors then converge slowly towards zero as N increases. The curves for the range 0-0.125 Δt show that most of the error occurs in the passband regions of the filters. This could be expected, since the largest errors occur at the breaks in the specified response.

The general characteristics can be related to the behavior of the Fourier-series kernels as a function of N as illustrated in figure 18. The width of the main peak, which causes the largest errors, decreases rapidly for small N and then converges very slowly towards zero for large N. The characteristics of the error curves are related to the impulse responses of the filters later on in this section.
Figure 23. Percent square error vs \( N \) for a discrete lowpass filter sampled at \( 1\Delta t \) and \( 2\Delta t \). Specified response: 0.0 0.0 0.05 0.10 \( \Delta t \). Errors summed over frequency ranges indicated at upper right of each curve.
Figure 24. Percent square error vs N for a discrete bandpass filter sampled at 1\(\Delta t\) and 2\(\Delta t\). Specified response: 0.02 0.05 0.09 0.12 \(f\Delta t\). Errors summed over frequency ranges indicated at upper right of each curve.
Error curves for filters sampled at coarser intervals behave the same as those discussed above, except for very slight variations. The error curves for the coarser intervals are plotted to make the curves directly comparable over the same ranges. The points represent filters of equal time duration but different N and sampling interval. All four curves in figures 23 and 24 for a sampling interval of $2\Delta t$ match comparable curves for a sampling interval of $1\Delta t$.

For all the filters studied, in no case was the total square error found to increase with N. Some of the error curves in figures 23 and 24 show reversals, but these are for portions of the total range. None of the curves for the range out to the Nyquist frequency show a reversal.

The value of these curves lies in the fact that once the desired error level is specified, the filter with the minimum number of points can be chosen which falls at or below the error level. The upper curves in figure 24 have two places, $N = 27-31$ and $N = 33-37$, where no significant change in error occurs with N. Obviously, one would want to choose the smallest N at these flat places. The lowpass filter's error drops rather rapidly to a low level before leveling off. When the errors reach a low level, such as $N = 25$ in figure 23 and $N = 47$ in figure 24 for the filters at $1\Delta t$, the decrease in error with N is very slow and small.
Figure 25 shows error curves for filters of several different shapes and sizes. These plots have some interesting characteristics and show the variations in error curves as the filter shape and filter specifications change. In some cases there are steep drops in the error with $N$. By increasing $N$ slightly, a large reduction in error can be obtained. To choose $N$ by looking at the frequency characteristics for each $N$ would not be as simple or quantitative as using error plots, although the first method gives an indication of error as a function of frequency.

Some percent error plots are shown in the center of figure 26. The general characteristics parallel those of the square error, but there are several reversals in the curves. Different scales necessary for plotting tend to emphasize these reversals. A close look at the actual errors shows a wide distribution of small errors which add up to make a noticeable change in the percent error. When these small errors are squared, their effect is diminished, since they are much less than 1, and their squares make little change in the percent square error. The reversals in the curves do not reflect a significant change. The differences between the percent error and percent square error are an effect of the error measure used. The least squares error development puts more value on the larger
Figure 25. Percent square error vs \( N \) for an assortment of discrete filters sampled at \( 1\Delta t \). Specified responses indicated at the right of each curve. All errors summed from 0.0-0.50 \( f\Delta t \).
Figure 26. Percent square error, percent error, and percent of impulse response sampled all vs N for lowpass (0.0 0.0 0.05 0.10 fAt) and bandpass (0.02 0.05 0.09 0.12 fAt) discrete filters sampled at 1AT.
errors as compared to the smaller errors.

The characteristics of the error curves can be directly related to the shape of the filter's impulse response. If the error curve for the lowpass filter sampled at $l_{At}$ is compared to its impulse response (figure 11), it can be seen that the large break at $N = 11$ is associated with the sampling of the center lobe of the impulse response. The drop to the next level at $N = 23$ can be related to the sampling of the two smaller side lobes. The major portion of the impulse response is sampled by the time $N = 23$ and the portions beyond contribute very little, as the error curve indicates. The case for the bandpass filter is somewhat similar. The first large break in the error curve at $N = 19$ occurs when the center and two side lobes are sampled. The level portions of the error plot can be associated with the zero crossings where the amplitude of the impulse response is small. The second large break in the curve occurs at $N = 47$ and correlates with the sampling of the last lobes with significant amplitude.

The filter with specifications 0.02, 0.04, 0.08, 0.15 $f_{At}$ has a large flat area in its curve error (figure 25). Looking at its impulse response (figure 11), one can see the cause of the leveling. The impulse response has very small amplitudes near the zero crossing from $N = 29$ to $N = 41$. 
The filter with specifications $0.06, 0.09, 0.13, 0.16 \, \Delta t$ has several steps in its error curve (figure 25). It is the same shape filter as the one specified $0.02, 0.05, 0.09, 0.12 \, \Delta t$, except for a shift to a higher frequency. The envelope of the impulse responses (figure 11) for the two filters is the same, but the higher frequency passband of the one filter causes a more rapid modulation of the envelope. Both error curves converge to the same low values at $N = 47$. In practically all cases, the error curves reflect the variations in shape and behavior of the impulse responses.

Another way to relate impulse response behavior to the error curves is to look at the percent absolute area included in the impulse response for each $N$. Crude approximations to the percent area taken for a certain $N$ were computed by summing the absolute values of the amplitudes of the bandpass and lowpass filters sampled at $\Delta t$ and dividing by the sums for $N = 101$ and $N = 51$, respectively. These curves are plotted at the bottom of figure 26. Their trends match those of the error curves quite well, except for the small $N$.

**Other Sampling Functions:** As a direct result from the minimization of the weighted square error, it was found that the best uniformly weighted discrete filter is obtained by
sampling the filter's impulse response with a sequence of unit impulses, bearing in mind the restriction on specifications exceeding the Nyquist frequency. The sequence of unit impulses, also known as a comb, and its application to sampling are described in chapter 1 and the preceding sections. Since sampling with a comb gives the minimum square error up to the Nyquist frequency, use of any other sampling function will result in a total square error greater than that for comb sampling. However, a different distribution of error may be achieved with another sampling function which may be useful even though total square error is increased. For example, some of the undesirable frequency characteristics of filters, such as overshoot, broadening, and oscillations in the reject region, can be attenuated but only by sacrificing something else.

From Fourier transform theory we know that the transform of a sampled function is equal to the convolution of the transform of the sampling function with the true transform. If various sampling functions are chosen, different effects can be achieved when the transform of the sampling function is convolved with the true transform. Sometimes it is difficult to visualize the effect a certain sampling transform will have when convolved with the true transform. The uncertainty makes it necessary sometimes to determine
useful sampling functions by trial and error.

Some typical choices for sampling functions are sequences of impulses with envelopes of squares, triangles, Gaussian functions, sinc (sin x/x) functions, and sinc squared functions. The comb and its transform, the Fourier-series kernel, have already been discussed. A triangular sequence of impulses transforms into a sinc squared function in the frequency domain. The sinc squared frequency function is twice as broad as the Fourier-series kernel but is positive and dies out more rapidly. For a specific N, the sampled filter's spectrum would have greater broadening effects but smaller oscillations in the reject region. A Gaussian sampling function transforms into itself; hence, its effect is to broaden the spectrum and eliminate or attenuate the oscillatory effects. The sinc and sinc squared functions transform into a square and triangle, respectively. The square and triangle act as moving averages under convolution and attenuate overshoot at the expense of broadening the spectrum. Of course, all the spectra of these various sampling functions repeat on the interval $\frac{2\pi}{\Delta t}$, just as the Fourier-series kernel does. The effects of the sampling functions can be varied by varying the parameters describing the functions.
No sampling functions other than the comb, discussed under "Uniform Error Weighting" will be considered in detail in this thesis. The use of error weighting, however, can be interpreted in terms of sampling functions, although not in as direct or simple a manner as that described above, and in the following section, "Nonuniform Error Weighting", some of the results will be examined in terms of sampling functions.

Nonuniform Error Weighting

The general effects of error weighting are examined in this section. The first part is a brief description of the error weighting functions used. The next three parts review the quantitative and qualitative effects of several error weighting functions used in the design of some simple lowpass and bandpass filters. The final part is a general discussion on error weighting and possible further experiments.

Weighting Functions: Several real, even, and positive weighting functions were developed for use in the study of weighting effects. It was difficult to choose a set of weighting functions which might yield the most significant effects because of the infinite number available; however, the ones chosen offer some flexibility and serve to illustrate the general effects of error weighting. The computer
programs for the weighting functions are discussed briefly in Appendix I.

The weighting functions used are illustrated in figure 27. The most flexible type consists of a series of blocks whose amplitude and width can be specified. Any number or combination of blocks can be specified. The other weighting functions are exponentials which decay with \( w \) or \( w^2 \). The decay constants and point of maximum amplitude can be specified. The blocks and exponentials can be combined to obtain weighting functions like those at the bottom of figure 27.

Provisions for using forms of the error itself for weighting functions were considered, but the computer was too small to include the necessary programming.

**Weighting Magnitude Effects:** To study the effects that the relative magnitudes of the error weights cause, a set of weighted error filters were generated using various relative magnitudes. The filters were designed at a sample interval of \( 1\Delta t \) with a maximum \( N \) of 41. The block weighting functions were used with weights of 1.0 for the range 0 to 0.25 on the dimensionless scale. Weights of 0.5, 0.1, 0.01, 0.001, 0.0001, and 0 were used on the portion of the scale from 0.25 to 0.50 \( f\Delta t \). The passbands of both filters studied fell within the zone weighted with 1.0.
Figure 27. Error weighting functions used.
Figures 28 through 34 illustrate the effects of various relative magnitudes of weighting for lowpass and bandpass filters. The weighting curves are indicated by the dashed lines. For comparison, the filters with a uniform weight of 1.0 appear at the beginning of each series of filters. The improvement in filter characteristics in the region with weight 1.0 is apparent as the contrast in weights increases. More obvious than the improvements are the increases in error in the lightly weighted regions. By the time the weight is decreased to 0.001 in the region 0.25-0.50 $f\Delta t$, the error constitutes the major portion of the energy in the spectrum. Each time more benefit in the weighted region is obtained a disproportionately higher price is paid in the lightly weighted region.

The improvement in the region 0-0.25 $f\Delta t$ for the various weights is of the same nature as that obtained by increasing $N$ in an unweighted filter. The error weighting effects are greatest for the smaller $N$ since the error in the unweighted filters is largest for small $N$. The improvement for zero weight is the best one can expect to accomplish using error weighting. Of course more improvement might be obtained by weighting a larger region of the spectrum but no reject region would be left. The error in the zero weighted region is the largest of all errors for the various weights.
Figure 28. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $l\Delta t$. 
Figure 29. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $\Delta t$. 
Figure 30. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $1\Delta t$. 
Figure 31. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $1\Delta t$. 
Figure 32. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $1\Delta t$. 
Figure 33. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval $l_\Delta t$. 
Figure 34. Amplitude responses of a weighted error filter for various weight contrasts. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at upper right in each plot. Sample interval 1Δt.
In theory, with a zero weight the error is left completely unrestricted and, as is obvious in figures 29, 31, 33, and 34, behaves accordingly.

The discrete filters obtained using weighted error appear quite unusual when compared to the unweighted filter time-domain weights. As the inserts of the filter weights show in figures 28 to 34, the discrete filters become more radical as the contrast in weighting increases. The lines in the small plots of the filter weights indicate only relative sizes; as the contrast in the error weights increases, some of the filter weights may reach amplitudes 100 times greater than the amplitudes for the unweighted filters. The largest changes occur at or near the extremities of the filters. As the contrast in weighting is increased, the filter weights at the ends become larger with respect to those near the center and oscillate rapidly, and eventually the weighted error filter bears no resemblance to the unweighted filter. Some of these unusual looking filters will be examined later in terms of sampling functions.

Plots of percent square error vs N are shown in figures 35 and 36 for filters weighted with a variety of weight magnitudes. Error curves are included for both the lowpass and bandpass filters weighted with unity from 0.0-0.25 $f\Delta t$ and with 0.1, 0.01, 0.0001, and 0 from
Figure 35. Comparison of percent square error curves for weighted (crosses) and unweighted (circles) error lowpass filters over several frequency ranges. Filter specifications 0.0 0.0 0.05 0.10 fAt and sample interval 1At.
Figure 36. Comparison of percent square error curves for weighted (crosses) and unweighted (circles) error bandpass filters over several frequency ranges. Filter specifications 0.02 0.05 0.09 0.12 f_\Delta t and sample interval 1\Delta t.
0.25-0.50 fΔt. Separate plots are made for the ranges 0.0-
0.25 fΔt and 0.0-0.50 fΔt. The percent square errors from
the corresponding unweighted filters are included for
comparison in the plots. Since the errors in the region
0.25-0.50 for weights of 0.0001 and 0 are so large, these
plots have been left out. Also, errors exceeding 5 square
percent in the other plots have, as before, been left out.

To complement the percent square error plots, a more
complete tabulation of the percent square errors is given
on pages 95 and 96. The errors for a variety of weights
are tabulated over the range 0.0-0.25 fΔt in the top tables
and 0.0-0.50 fΔt in the bottom tables, with N ranging from
5 to 35 for the lowpass filter and 11 to 41 for the bandpass
filter. The percent square errors for the corresponding
unweighted filters are in the second columns. Percent
square errors are included in the last column for unweighted
filters at the same N but for twice the sample interval as
for the other filters.

Examination of the error tables and curves will show
in detail the variations of square error with N and the
error weights. In the region from 0.0-0.25 fΔt, weighted
with unity, there is a decrease in the percent square error
as the weight in the area 0.25-0.50 fΔt is decreased.
Because of the smaller weights, the error in the section
Percent Square Error vs N for Six Weighting Contrasts:
Lowpass Filter

Specifications: 0.0 0.0 0.05 0.10 fΔt, sample rate: 1Δt

Weights: All filters weighted with 1.0 from 0.0 to 0.25 fΔt
Weights for region from 0.25 to 0.50 fΔt indicated below.
Last column is uniformly weighted filter sampled at 2Δt

<table>
<thead>
<tr>
<th>N</th>
<th>Weights From 0.25 to 0.50 fΔt</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
<td>0.5</td>
</tr>
<tr>
<td>----</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>5</td>
<td>23.81</td>
<td>21.77</td>
</tr>
<tr>
<td>7</td>
<td>10.84</td>
<td>9.96</td>
</tr>
<tr>
<td>9</td>
<td>4.27</td>
<td>3.87</td>
</tr>
<tr>
<td>11</td>
<td>2.53</td>
<td>2.48</td>
</tr>
<tr>
<td>13</td>
<td>2.41</td>
<td>2.41</td>
</tr>
<tr>
<td>15</td>
<td>2.31</td>
<td>2.26</td>
</tr>
<tr>
<td>17</td>
<td>1.76</td>
<td>1.68</td>
</tr>
<tr>
<td>19</td>
<td>1.04</td>
<td>0.96</td>
</tr>
<tr>
<td>21</td>
<td>.46</td>
<td>.42</td>
</tr>
<tr>
<td>23</td>
<td>.20</td>
<td>.19</td>
</tr>
<tr>
<td>27</td>
<td>.14</td>
<td>.14</td>
</tr>
<tr>
<td>31</td>
<td>.09</td>
<td>.09</td>
</tr>
<tr>
<td>35</td>
<td>.05</td>
<td>.05</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.25 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>3.03</td>
</tr>
<tr>
<td>7</td>
<td>2.49</td>
</tr>
<tr>
<td>9</td>
<td>1.53</td>
</tr>
<tr>
<td>11</td>
<td>1.00</td>
</tr>
<tr>
<td>13</td>
<td>0.78</td>
</tr>
<tr>
<td>15</td>
<td>0.40</td>
</tr>
<tr>
<td>17</td>
<td>0.05</td>
</tr>
<tr>
<td>19</td>
<td>0.05</td>
</tr>
<tr>
<td>21</td>
<td>0.05</td>
</tr>
<tr>
<td>23</td>
<td>0.05</td>
</tr>
<tr>
<td>27</td>
<td>0.03</td>
</tr>
<tr>
<td>31</td>
<td>0.01</td>
</tr>
<tr>
<td>35</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.50 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>100.61</td>
</tr>
<tr>
<td>7</td>
<td>101.00</td>
</tr>
<tr>
<td>9</td>
<td>100.50</td>
</tr>
<tr>
<td>11</td>
<td>99.88</td>
</tr>
<tr>
<td>13</td>
<td>99.83</td>
</tr>
<tr>
<td>15</td>
<td>99.88</td>
</tr>
<tr>
<td>17</td>
<td>100.00</td>
</tr>
<tr>
<td>19</td>
<td>100.00</td>
</tr>
<tr>
<td>21</td>
<td>100.00</td>
</tr>
<tr>
<td>23</td>
<td>100.00</td>
</tr>
<tr>
<td>27</td>
<td>100.10</td>
</tr>
<tr>
<td>31</td>
<td>90000</td>
</tr>
<tr>
<td>35</td>
<td>30000</td>
</tr>
</tbody>
</table>

* > 50,000
Percent Square Error vs N for Six Weighting Contrasts:
Bandpass Filter

Specifications: 0.02 0.05 0.09 0.12 fΔt, sample rate: 1fΔt

Weights: All filters weighted with 1.0 from 0.0 to 0.25 fΔt
Weights for region 0.25 to 0.50 fΔt indicated below
Last column is uniformly weighted filter sampled at 2fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>Weights From 0.25 to 0.50 fΔt</th>
<th>2fΔt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.25 fΔt

<table>
<thead>
<tr>
<th></th>
<th>11</th>
<th>13</th>
<th>15</th>
<th>17</th>
<th>19</th>
<th>21</th>
<th>23</th>
<th>25</th>
<th>27</th>
<th>29</th>
<th>31</th>
<th>33</th>
<th>35</th>
<th>37</th>
<th>39</th>
<th>41</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>34.88</td>
<td>33.39</td>
<td>29.21</td>
<td>26.52</td>
<td>21.11</td>
<td>15.65</td>
<td>6.01</td>
<td>1.74</td>
<td>21.52</td>
<td>20.05</td>
<td>17.60</td>
<td>11.99</td>
<td>9.96</td>
<td>8.05</td>
<td>3.14</td>
<td>1.30</td>
</tr>
<tr>
<td></td>
<td>10.27</td>
<td>9.25</td>
<td>6.76</td>
<td>5.34</td>
<td>3.32</td>
<td>2.41</td>
<td>1.50</td>
<td>1.28</td>
<td>3.70</td>
<td>3.28</td>
<td>2.73</td>
<td>1.90</td>
<td>1.71</td>
<td>1.61</td>
<td>1.51</td>
<td>1.13</td>
</tr>
<tr>
<td></td>
<td>1.80</td>
<td>1.74</td>
<td>1.63</td>
<td>1.60</td>
<td>1.60</td>
<td>1.59</td>
<td>1.43</td>
<td>1.13</td>
<td>1.67</td>
<td>1.67</td>
<td>1.65</td>
<td>1.60</td>
<td>1.60</td>
<td>1.43</td>
<td>1.23</td>
<td>.58</td>
</tr>
<tr>
<td></td>
<td>1.61</td>
<td>1.58</td>
<td>1.50</td>
<td>1.43</td>
<td>1.32</td>
<td>1.29</td>
<td>1.23</td>
<td>.11</td>
<td>1.38</td>
<td>1.36</td>
<td>1.32</td>
<td>1.26</td>
<td>1.25</td>
<td>1.25</td>
<td>1.21</td>
<td>.08</td>
</tr>
<tr>
<td></td>
<td>1.27</td>
<td>1.27</td>
<td>1.26</td>
<td>1.25</td>
<td>1.25</td>
<td>1.24</td>
<td>1.12</td>
<td>.07</td>
<td>1.27</td>
<td>1.27</td>
<td>1.26</td>
<td>1.23</td>
<td>1.20</td>
<td>1.16</td>
<td>1.10</td>
<td>.07</td>
</tr>
<tr>
<td></td>
<td>1.15</td>
<td>1.15</td>
<td>1.13</td>
<td>1.12</td>
<td>1.12</td>
<td>1.12</td>
<td>1.02</td>
<td>.07</td>
<td>1.13</td>
<td>1.12</td>
<td>1.09</td>
<td>1.03</td>
<td>.95</td>
<td>.84</td>
<td>.43</td>
<td>.06</td>
</tr>
<tr>
<td></td>
<td>.70</td>
<td>.66</td>
<td>.57</td>
<td>.44</td>
<td>.34</td>
<td>.24</td>
<td>.09</td>
<td>.06</td>
<td>.76</td>
<td>.77</td>
<td>.14</td>
<td>5.26</td>
<td>50.96</td>
<td>328</td>
<td>--</td>
<td>100.27</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.50 fΔt

<table>
<thead>
<tr>
<th></th>
<th>11</th>
<th>13</th>
<th>15</th>
<th>17</th>
<th>19</th>
<th>21</th>
<th>23</th>
<th>25</th>
<th>27</th>
<th>29</th>
<th>31</th>
<th>33</th>
<th>35</th>
<th>37</th>
<th>39</th>
<th>41</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>36.55</td>
<td>37.19</td>
<td>52.84</td>
<td>138</td>
<td>2000</td>
<td>10000</td>
<td>--*</td>
<td>100.27</td>
<td>23.98</td>
<td>24.56</td>
<td>34.33</td>
<td>250</td>
<td>785</td>
<td>10000</td>
<td>--</td>
<td>100.21</td>
</tr>
<tr>
<td></td>
<td>11.88</td>
<td>12.35</td>
<td>21.75</td>
<td>68.44</td>
<td>929</td>
<td>3000</td>
<td>--</td>
<td>100.20</td>
<td>4.54</td>
<td>4.72</td>
<td>6.86</td>
<td>37.03</td>
<td>88.00</td>
<td>587</td>
<td>--</td>
<td>100.23</td>
</tr>
<tr>
<td></td>
<td>1.97</td>
<td>2.00</td>
<td>2.43</td>
<td>3.23</td>
<td>3.97</td>
<td>45.34</td>
<td>--</td>
<td>100.23</td>
<td>1.68</td>
<td>1.68</td>
<td>1.76</td>
<td>3.49</td>
<td>23.30</td>
<td>512</td>
<td>--</td>
<td>100.17</td>
</tr>
<tr>
<td></td>
<td>1.64</td>
<td>1.66</td>
<td>1.93</td>
<td>4.86</td>
<td>41.71</td>
<td>118</td>
<td>50000</td>
<td>100.15</td>
<td>1.43</td>
<td>1.44</td>
<td>1.58</td>
<td>3.22</td>
<td>5.66</td>
<td>22.77</td>
<td>--</td>
<td>100.15</td>
</tr>
<tr>
<td></td>
<td>1.28</td>
<td>1.28</td>
<td>1.30</td>
<td>1.44</td>
<td>3.33</td>
<td>69.43</td>
<td>185</td>
<td>--</td>
<td>100.15</td>
<td>1.27</td>
<td>1.27</td>
<td>1.33</td>
<td>2.25</td>
<td>16.37</td>
<td>185</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>1.17</td>
<td>1.17</td>
<td>1.23</td>
<td>1.72</td>
<td>2.18</td>
<td>9.18</td>
<td>--</td>
<td>100.15</td>
<td>1.13</td>
<td>1.14</td>
<td>1.26</td>
<td>3.18</td>
<td>42.46</td>
<td>363</td>
<td>--</td>
<td>100.15</td>
</tr>
<tr>
<td></td>
<td>.76</td>
<td>.77</td>
<td>1.14</td>
<td>5.26</td>
<td>50.96</td>
<td>328</td>
<td>--</td>
<td>100.15</td>
<td>.76</td>
<td>.77</td>
<td>1.14</td>
<td>5.26</td>
<td>50.96</td>
<td>328</td>
<td>--</td>
<td>100.15</td>
</tr>
</tbody>
</table>

* > 50,000
0.25-0.50 \, f_{\Delta t} \text{ increases. As the error tables indicate, the distortion in the region } 0.25-0.50 \, f_{\Delta t} \text{ increases disproportionately with respect to the decrease in error in the region weighted with unity. As a rough average, with each decrease in weight by a factor of 10, the percent square error in the area weighted with unity decreases between several percent and a fraction of a percent, depending on N, whereas the error in the remaining portion of the spectrum increases by an order of magnitude. The only weights which did not produce such a large loss in the lightly weighted region were those greater than or nearly equal to 0.1. In these cases a small improvement in one region was obtained for a small overall loss.}

When the decreases in square error for the frequencies from 0.0-0.25 \, f_{\Delta t} \text{ are large, the increases in square error in the range } 0.25-0.50 \, f_{\Delta t} \text{ are greater. If very little decrease in square error is obtained in the heavily weighted region, then the loss in the other portion of the spectrum is less. Typical cases of the latter can be observed in the neighborhoods of } N = 13 \text{ and } 23 \text{ for the low-pass filter and } N = 21, 27, \text{ and } 33 \text{ for the bandpass filter for practically all weights except zero. For some } N, \text{ both the improvements and losses caused by error weighting are attenuated. Reasons for this behavior will be discussed later.}
As is obvious from both the error plots and the error tables, the largest improvements and losses caused by error weighting occur for small \( N \). The weighting effects then decrease as \( N \) is increased for all weights in the region 0.25-0.50 \( f_\Delta t \) except zero. In the case of the lowpass filter, practically no weighting effects are observed by the time \( N = 35 \). The weighting effects decrease in the same manner for the bandpass filter up to \( N = 41 \), and though no observations were possible past \( N = 41 \), the effects would undoubtedly continue to die out. If \( N \) were sufficiently large, it is reasonable to assume that weighting effects would die out even when the error is left unrestricted in the region 0.25-0.50 \( f_\Delta t \) as with the zero weighting. The above error weighting behavior can be expected since as \( N \) increases the square error for the unweighted filters decreases towards zero and the weighting applied to the error loses its significance.

An important point brought out by the error curves in figures 35 and 36 is that the weighting effects are accentuated for \( N \) occurring on the sloping portions of the unweighted error plots. The accentuation of weighting effects for some \( N \) and the attenuation of weighting effects for the other \( N \) mentioned earlier can both be explained by observing the nature of the improvements occurring in
weighted error filters. As mentioned earlier when discussing the spectra of weighted error filters, the improvement in the more heavily weighted regions is similar to that achieved when \(N\) is increased in an unweighted filter. Examination of the error curves for the weighted filters in figures 35 and 36 over the range 0.0-0.25 \(f\Delta t\) shows that their shape and character are similar to the unweighted error curves. For examples, if the unweighted error curves were projected back in \(N\) by approximately 4 and 6 points for weights of 0.0001 and 0, respectively, the curves would nearly match. Each point on the weighted error plots appears to be a projection of the error several points ahead, depending on the magnitude of the contrasts in weights.

The projection of the error ahead by several filter points as a result of weighting accounts for the accentuation of weighting effects on the sloping portions of the error curves. Since the decrease of error with \(N\) is rapid on the steep parts of the unweighted error curves, projecting the error ahead results in a large change. For a similar reason, projecting the error ahead at the beginning of a flat section of the error curves will result in little change. The projection of the error is the reason for the failure of the weighting to cause a significant improvement for certain \(N\). The \(N\) for which weighting effects are attenuated will be
found near or before the flat or less steeply sloping sections of the unweighted error curves. This behavior perhaps does not adequately explain why the error losses in the more lightly weighted portions of the spectrum are attenuated also for these N; but it seems that if nothing is gained in the heavily weighted region, a smaller loss is incurred in the other portions of the spectrum.

The error tables on pages 95 and 96 include the errors for unweighted filters with the same N but with the sample interval increased to $2\Delta t$. Doubling sample interval for a specific N means that the time duration of the filter is doubled and the truncation effects are reduced. Listing the errors for these filters provides a comparison which shows the decreases in error obtained by error weighting in one case and doubling sample interval in the other case. Of course, the examples considered here form a special case, since it is possible to double the sampling interval of both filters from $\Delta t$ to $2\Delta t$ without an overlap of aliases. If the filters fell in the upper half of the spectrum (0.25-0.50 $f_{\Delta t}$) this would be impossible. The decrease in error for frequencies of 0.0-0.25 $f_{\Delta t}$ for both the lowpass and bandpass filters obtained by doubling the sample interval was greater than that obtained even with zero weights in the range 0.25-0.50 $f_{\Delta t}$. Some of the weighted filters for small
N come close to matching the decrease in error of the filters at $2\Delta t$, but in the lightly weighted portion from 0.25 to 0.50 $f\Delta t$ the error is much greater. These examples indicate that in a certain frequency range, a better filter can be obtained by doubling sampling interval than by weighting error.

As a possible means of gaining insight into error weighting, the effects of error weighting were interpreted in terms of sampling functions. By dividing the impulse weights of the discrete filter obtained with the frequency-domain weighting process by the values of the continuous impulse response of the unweighted filter a sampling function is obtained. The discrete filter which corresponds to frequency-domain weighting may be obtained from the continuous impulse response by simply multiplying it by the proper weighted error sampling sequence. The weighted sampling sequences in figure 37 were computed in this manner for several N for the lowpass filter with a weight of 0.0001 from 0.25-0.50 $f\Delta t$. The magnitudes of these weighted sampling sequences vary widely and hence the diagrams of figure 37 indicate relative sizes only.

The most unusual aspect of the weighted sampling sequences is the large weights at the extremities. The sequences for N = 5 and 7 do not show this behavior but do oscillate rapidly. The inability of the lowpass filter for
Figure 37. Weighted error sampling sequences at $1\Delta t$ for the lowpass filter (0.0 0.0 0.05 0.10 $f\Delta t$) weighted with 0.0001 from 0.25 to 0.50 $f\Delta t$. 
N = 11 to respond well to weighting is reflected in the sampling function for N = 11 which approximates a sequence of unit impulses; i.e., the usual unweighted sampling sequence. As N increases, the sampling sequences start approaching sequences of unit impulses, and the weighting effects begin to attenuate. This effect occurs for N = 33, where the sequence near the center approximates unit impulses and the effects on the ends appear to be beginning to die out. Although the weights at the ends of the sequences are quite large, it should be remembered that the impulse response being sampled dies out so that as N increases, the large weights are operating on small numbers.

A possible reason for the large amplitudes at the ends of the sampling sequences is the nature of the transform of a discrete function. The transform of an even sequence of weighted impulses is a summation of cosines of varying frequencies and amplitudes. As the discrete filter weights become farther removed from the origin, they represent the addition of higher frequency cosines to the transform and, as such, are responsible for the resolution of the detail in a filter's response. As the error curves and filter responses showed in the section on unweighted filters, the gross characteristics of a discrete filter are generally defined for small N and the details brought out by the
larger \( N \). It seems logical, then, since the points about the origin define the gross characteristics of the filter, that the effects and improvements arising from weighted error design come chiefly from manipulation of the points at the extremities of the filters.

The Fourier transforms of the sampling sequences described above are shown in figure 38. It should be emphasized that these transforms are periodic on the interval 1.0 on the dimensionless scale. The transforms bear little resemblance to each other or to the Fourier-series kernel for the different \( N \), with the exception of \( N = 11 \). It is difficult to visualize the effects these transforms would have on the specified filter response under convolution. The transform of the sampling sequence for \( N = 17 \) was convolved with the specified filter response and found to give a result identical to the discrete transform for the weighted error discrete filter for \( N = 17 \). No significant interpretations of these sampling sequences or their transforms were made; however, it is interesting to note that in designing sampling functions, it is unnecessary to be restricted to functions whose transforms peak about zero frequency and die out as do the Fourier-series kernels.
Figure 38. Fourier transforms of sampling sequences derived from a lowpass filter (0.0 0.0 0.05 0.10 fΔt) with error weights of 1.0 from 0.0-0.25 fΔt and 0.0001 from 0.25-0.50 fΔt. Sample interval of 1Δt.
**Weighting Position Effects:** To study the effects of variations in the length of the weighted regions of the filter spectra, several lowpass and bandpass filters were generated with different size portions of the spectra being weighted. The first portion of the spectrum, containing the passbands, was weighted 1.0 out to 0.15, 0.25, 0.35, and finally 0.40 fΔt for the lowpass filter and out to 0.15, 0.25 and 0.36 fΔt for the bandpass filter. The remaining portions of the spectra out to 0.5 fΔt were weighted with 0.0001 so as to provide a good contrast in effects.

Figures 39 through 43 illustrate the effects of varying the length of the weighted region for the lowpass and bandpass filters. More improvement in the area weighted with 1.0 is gained as the region weighted with 0.0001 is increased in length. The improvement, as in the case of error weighting magnitude variations, is similar to that achieved by increasing N in an unweighted filter. The errors in the regions weighted with 0.0001 are large and distributed quite evenly in most cases.

The discrete filters themselves behave like those obtained by varying the magnitude of weighting, with the exception of those filters where the region weighted 0.0001 is withdrawn past 0.35 fΔt so that more of the low-frequency response is improved. Both the lowpass and bandpass filters
Figure 39. Amplitude responses of a weighted error filter for various weight positions. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at sample interval $1 \Delta t$ at upper right in each plot.
Figure 40. Amplitude responses of a weighted error filter for various weight positions. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at sample interval $\Delta t$ at upper right in each plot.
Figure 41. Amplitude responses of a weighted error filter for various weight positions. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at sample interval $1\Delta t$ at upper right in each plot.
Figure 42. Amplitude responses of a weighted error filter for various weight positions. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at sample interval $1\Delta t$ at upper right in each plot.
Figure 43. Amplitude responses of a weighted error filter for various weight positions. Weighting curve - dashed lines; specified response - thin lines; and discrete filter at sample interval $1\Delta t$ at upper right in each plot.
which are weighted 0.0001 past 0.35 $f \Delta t$ behave similarly for most $N$. Every other point in the filters reverses sign, and the weights converge towards zero instead of growing large at the ends. For the lowpass filter at the bottom of figure 40, the convergence of the discrete filter weights is very similar to the convergence in the corresponding unweighted filter.

A more complete and detailed tabulation of the percent square error for the various weighting positions is given on pages 113 through 116. It was necessary to include all four ranges of the error to illustrate the effects of different lengths being weighted. The effects observed in the response curves, above, hold for most positions and $N$. The error in the region weighted with 1.0 decreases as the lightly weighted region is extended into the lower frequencies. As the portion weighted with 0.0001 decreases in length towards 0.50 $f \Delta t$, the weighting effects disappear. There are several $N$ for both the lowpass and bandpass filters where the weighting effects are attenuated. These $N$ (11, 13, 21, 23 for the lowpass and 19, 21, 27, 29 and 33 for the bandpass) are the same or next to the $N$ for which this behavior occurred in the filters with various weighting magnitudes.

For the lowpass filter and for some $N$ of the bandpass filter, the total error is largest for those cases where the smaller weight begins in the neighborhood of 0.35 $f \Delta t$;
Percent Square Error vs N for Five Weighting Positions: Lowpass Filter

Specifications: 0.0 0.0 0.05 0.10 fΔt, sample rate: 1Δt

Weights: Error weight is 1.0 from 0.0 fΔt to indicated frequency and 0.0001 from that frequency to 0.50 fΔt.
Last column is uniformly weighted filter sampled at 2Δt

<table>
<thead>
<tr>
<th>N</th>
<th>Error Weight of 1.0 to This Frequency</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.50</td>
<td>0.45</td>
</tr>
<tr>
<td>----</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>5</td>
<td>18.93</td>
<td>17.96</td>
</tr>
<tr>
<td>7</td>
<td>8.55</td>
<td>7.86</td>
</tr>
<tr>
<td>9</td>
<td>3.35</td>
<td>3.09</td>
</tr>
<tr>
<td>11</td>
<td>2.19</td>
<td>2.18</td>
</tr>
<tr>
<td>13</td>
<td>2.30</td>
<td>2.31</td>
</tr>
<tr>
<td>15</td>
<td>2.14</td>
<td>2.06</td>
</tr>
<tr>
<td>17</td>
<td>1.48</td>
<td>1.36</td>
</tr>
<tr>
<td>19</td>
<td>0.50</td>
<td>0.76</td>
</tr>
<tr>
<td>21</td>
<td>0.31</td>
<td>0.26</td>
</tr>
<tr>
<td>23</td>
<td>0.14</td>
<td>0.13</td>
</tr>
<tr>
<td>27</td>
<td>0.13</td>
<td>0.13</td>
</tr>
<tr>
<td>31</td>
<td>0.08</td>
<td>0.07</td>
</tr>
<tr>
<td>35</td>
<td>0.05</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.125 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>23.81</td>
</tr>
<tr>
<td>7</td>
<td>10.84</td>
</tr>
<tr>
<td>9</td>
<td>4.27</td>
</tr>
<tr>
<td>11</td>
<td>2.53</td>
</tr>
<tr>
<td>13</td>
<td>2.41</td>
</tr>
<tr>
<td>15</td>
<td>2.31</td>
</tr>
<tr>
<td>17</td>
<td>1.76</td>
</tr>
<tr>
<td>19</td>
<td>1.04</td>
</tr>
<tr>
<td>21</td>
<td>0.46</td>
</tr>
<tr>
<td>23</td>
<td>0.20</td>
</tr>
<tr>
<td>27</td>
<td>0.14</td>
</tr>
<tr>
<td>31</td>
<td>0.09</td>
</tr>
<tr>
<td>35</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.25 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>23.81</td>
</tr>
<tr>
<td>7</td>
<td>10.84</td>
</tr>
<tr>
<td>9</td>
<td>4.27</td>
</tr>
<tr>
<td>11</td>
<td>2.53</td>
</tr>
<tr>
<td>13</td>
<td>2.41</td>
</tr>
<tr>
<td>15</td>
<td>2.31</td>
</tr>
<tr>
<td>17</td>
<td>1.76</td>
</tr>
<tr>
<td>19</td>
<td>1.04</td>
</tr>
<tr>
<td>21</td>
<td>0.46</td>
</tr>
<tr>
<td>23</td>
<td>0.20</td>
</tr>
<tr>
<td>27</td>
<td>0.14</td>
</tr>
<tr>
<td>31</td>
<td>0.09</td>
</tr>
<tr>
<td>35</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Table continues on following page
<table>
<thead>
<tr>
<th>N</th>
<th>Error Weight of 1.0 to This Frequency</th>
<th>fΔt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.50</td>
<td>0.45</td>
</tr>
<tr>
<td>5</td>
<td>26.48</td>
<td>24.87</td>
</tr>
<tr>
<td>7</td>
<td>11.74</td>
<td>10.92</td>
</tr>
<tr>
<td>9</td>
<td>4.71</td>
<td>4.38</td>
</tr>
<tr>
<td>11</td>
<td>2.62</td>
<td>2.54</td>
</tr>
<tr>
<td>13</td>
<td>2.42</td>
<td>2.42</td>
</tr>
<tr>
<td>15</td>
<td>2.34</td>
<td>2.30</td>
</tr>
<tr>
<td>17</td>
<td>1.83</td>
<td>1.73</td>
</tr>
<tr>
<td>19</td>
<td>1.10</td>
<td>.99</td>
</tr>
<tr>
<td>21</td>
<td>.52</td>
<td>.44</td>
</tr>
<tr>
<td>23</td>
<td>.22</td>
<td>.19</td>
</tr>
<tr>
<td>27</td>
<td>.14</td>
<td>.14</td>
</tr>
<tr>
<td>31</td>
<td>.09</td>
<td>.09</td>
</tr>
<tr>
<td>35</td>
<td>.05</td>
<td>.05</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.375 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>Error Weight of 1.0 to This Frequency</th>
<th>fΔt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.50</td>
<td>0.45</td>
</tr>
<tr>
<td>5</td>
<td>27.52</td>
<td>28.90</td>
</tr>
<tr>
<td>7</td>
<td>12.41</td>
<td>14.03</td>
</tr>
<tr>
<td>9</td>
<td>5.06</td>
<td>6.31</td>
</tr>
<tr>
<td>11</td>
<td>2.69</td>
<td>3.18</td>
</tr>
<tr>
<td>13</td>
<td>2.42</td>
<td>2.42</td>
</tr>
<tr>
<td>15</td>
<td>2.36</td>
<td>2.92</td>
</tr>
<tr>
<td>17</td>
<td>1.88</td>
<td>4.16</td>
</tr>
<tr>
<td>19</td>
<td>1.16</td>
<td>5.35</td>
</tr>
<tr>
<td>21</td>
<td>.55</td>
<td>5.24</td>
</tr>
<tr>
<td>23</td>
<td>.23</td>
<td>3.28</td>
</tr>
<tr>
<td>27</td>
<td>.14</td>
<td>.39</td>
</tr>
<tr>
<td>31</td>
<td>.10</td>
<td>7.12</td>
</tr>
<tr>
<td>35</td>
<td>.05</td>
<td>3.93</td>
</tr>
</tbody>
</table>

* > 50,000
Percent Square Error vs N for Three Weighting Positions:

Bandpass Filter

Specifications: 0.02 0.05 0.09 0.12 fΔt, sample rate: 1Δt

Weights: Error weight is 1.0 from 0.0 fΔt to indicated frequency and 0.0001 from that frequency to 0.50 fΔt.

Last column is uniformly weighted filter sampled at 2Δt

<table>
<thead>
<tr>
<th>N</th>
<th>Error Weight of 1.0 to This Frequency</th>
<th>0.50</th>
<th>0.36</th>
<th>0.25</th>
<th>0.15</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Errors Summed From 0.0 to 0.125 fΔt</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>28.79</td>
<td>20.66</td>
<td>11.62</td>
<td>2.41</td>
<td>1.55</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>16.78</td>
<td>10.27</td>
<td>5.27</td>
<td>1.01</td>
<td>1.17</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>7.19</td>
<td>3.88</td>
<td>1.51</td>
<td>1.06</td>
<td>1.20</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>2.36</td>
<td>1.48</td>
<td>1.33</td>
<td>1.05</td>
<td>1.06</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1.38</td>
<td>1.44</td>
<td>1.43</td>
<td>1.02</td>
<td>1.03</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1.51</td>
<td>1.45</td>
<td>1.27</td>
<td>1.03</td>
<td>.39</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1.42</td>
<td>1.27</td>
<td>1.16</td>
<td>1.02</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1.22</td>
<td>1.17</td>
<td>1.18</td>
<td>.98</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1.18</td>
<td>1.20</td>
<td>1.16</td>
<td>.95</td>
<td>.06</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1.20</td>
<td>1.16</td>
<td>1.07</td>
<td>.97</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>1.07</td>
<td>1.06</td>
<td>1.07</td>
<td>.70</td>
<td>.06</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>1.06</td>
<td>.95</td>
<td>.69</td>
<td>.21</td>
<td>.06</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>.55</td>
<td>.33</td>
<td>.15</td>
<td>.07</td>
<td>.05</td>
<td></td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.25 fΔt

<table>
<thead>
<tr>
<th>N</th>
<th>Errors Summed From 0.0 to 0.125 fΔt</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>34.88</td>
<td>25.92</td>
<td>15.65</td>
<td>5000</td>
<td>1.74</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>21.52</td>
<td>13.88</td>
<td>8.05</td>
<td>1000</td>
<td>1.30</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>10.27</td>
<td>5.89</td>
<td>2.41</td>
<td>235</td>
<td>1.28</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>3.70</td>
<td>2.20</td>
<td>1.61</td>
<td>248</td>
<td>1.13</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1.80</td>
<td>1.62</td>
<td>1.59</td>
<td>94.65</td>
<td>1.13</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1.67</td>
<td>1.63</td>
<td>1.43</td>
<td>37.88</td>
<td>.58</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1.61</td>
<td>1.45</td>
<td>1.29</td>
<td>50.46</td>
<td>.11</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1.38</td>
<td>1.28</td>
<td>1.25</td>
<td>54.16</td>
<td>.08</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1.27</td>
<td>1.26</td>
<td>1.24</td>
<td>92.45</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1.27</td>
<td>1.24</td>
<td>1.16</td>
<td>59.03</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>1.15</td>
<td>1.13</td>
<td>1.12</td>
<td>612</td>
<td>.07</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>1.13</td>
<td>1.05</td>
<td>.84</td>
<td>207</td>
<td>.06</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>.70</td>
<td>.48</td>
<td>.24</td>
<td>.65</td>
<td>.06</td>
<td></td>
</tr>
</tbody>
</table>

Table continues on following page
<table>
<thead>
<tr>
<th>N</th>
<th>Error Weight of 1.0 to This Frequency</th>
<th>2Δt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.50</td>
<td>0.36</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.375 fΔt

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>35.73</td>
<td>32.34</td>
<td>9000</td>
<td>10000</td>
</tr>
<tr>
<td>13</td>
<td>23.12</td>
<td>15.86</td>
<td>2000</td>
<td>2000</td>
</tr>
<tr>
<td>15</td>
<td>11.21</td>
<td>7.44</td>
<td>1000</td>
<td>286</td>
</tr>
<tr>
<td>17</td>
<td>4.19</td>
<td>3.89</td>
<td>300</td>
<td>315</td>
</tr>
<tr>
<td>19</td>
<td>1.90</td>
<td>1.81</td>
<td>33.86</td>
<td>138.26</td>
</tr>
<tr>
<td>21</td>
<td>1.68</td>
<td>1.77</td>
<td>290</td>
<td>45.89</td>
</tr>
<tr>
<td>23</td>
<td>1.63</td>
<td>1.50</td>
<td>51.39</td>
<td>76.22</td>
</tr>
<tr>
<td>25</td>
<td>1.41</td>
<td>1.40</td>
<td>15.82</td>
<td>100.18</td>
</tr>
<tr>
<td>27</td>
<td>1.28</td>
<td>1.27</td>
<td>42.40</td>
<td>153.45</td>
</tr>
<tr>
<td>29</td>
<td>1.27</td>
<td>1.42</td>
<td>107</td>
<td>71.93</td>
</tr>
<tr>
<td>33</td>
<td>1.16</td>
<td>1.18</td>
<td>7.78</td>
<td>881</td>
</tr>
<tr>
<td>37</td>
<td>1.13</td>
<td>1.80</td>
<td>183</td>
<td>368</td>
</tr>
<tr>
<td>41</td>
<td>.73</td>
<td>.74</td>
<td>183</td>
<td>102</td>
</tr>
</tbody>
</table>

Errors Summed From 0.0 to 0.50 fΔt

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>36.55</td>
<td>6000</td>
<td>10000</td>
<td>20000</td>
</tr>
<tr>
<td>13</td>
<td>23.98</td>
<td>10000</td>
<td>10000</td>
<td>3000</td>
</tr>
<tr>
<td>15</td>
<td>11.88</td>
<td>4000</td>
<td>3000</td>
<td>307</td>
</tr>
<tr>
<td>17</td>
<td>4.54</td>
<td>467</td>
<td>587</td>
<td>348</td>
</tr>
<tr>
<td>19</td>
<td>1.97</td>
<td>75.79</td>
<td>45.34</td>
<td>162.74</td>
</tr>
<tr>
<td>21</td>
<td>1.68</td>
<td>56.65</td>
<td>512</td>
<td>46.67</td>
</tr>
<tr>
<td>23</td>
<td>1.64</td>
<td>256.52</td>
<td>118.8</td>
<td>89.05</td>
</tr>
<tr>
<td>25</td>
<td>1.43</td>
<td>53.45</td>
<td>22.77</td>
<td>130.57</td>
</tr>
<tr>
<td>27</td>
<td>1.28</td>
<td>3.19</td>
<td>69.43</td>
<td>189.12</td>
</tr>
<tr>
<td>29</td>
<td>1.27</td>
<td>37.99</td>
<td>185</td>
<td>73.50</td>
</tr>
<tr>
<td>33</td>
<td>1.17</td>
<td>18.24</td>
<td>9.18</td>
<td>1000</td>
</tr>
<tr>
<td>37</td>
<td>1.13</td>
<td>92.69</td>
<td>363</td>
<td>493</td>
</tr>
<tr>
<td>41</td>
<td>.76</td>
<td>205</td>
<td>328</td>
<td>131</td>
</tr>
</tbody>
</table>
however, the weights starting at 0.40 $f_{\Delta t}$ and 0.45 $f_{\Delta t}$, have diminishing effects on the total error. The square errors for unweighted filters at a sampling interval of $2\Delta t$ are given in the last column of the error tables. For a weight of 0.0001 beginning at 0.15 $f_{\Delta t}$, the errors in the first range are smaller for the weighted filters than for the unweighted filters at $2\Delta t$. This holds for small $N$ only and as $N$ increases, the unweighted filters are much better. Of course, for the weighted filters the large errors occur for all of the response past 0.15 $f_{\Delta t}$.

Several experiments were tried where the reject region of the filters was weighted more heavily than the passband. The error was diminished in the reject region as expected, but the error in the passband increased in an unexpected manner. For small $N$ the weighting simply caused the amplitude of the filter to decrease for all frequencies and consequently, the passband was attenuated. As the weighting contrast was increased, the filter amplitudes approached zero. For the larger $N$, the amplitudes did not decrease, but instead, the overshoot in the responses became greater.

**Weighting Shape Effects:** Some weighted error filters were designed using the exponential weighting functions and combinations of the exponentials and blocks. Although the
block weighting functions are adequate to define general weighting effects, the other weighting functions are useful to illustrate that weighting effects have the same general characteristics for various shapes of weighting curves.

The amplitude responses for some of the exponentially weighted error filters are shown in figures 44 through 46. The weighting curves are indicated by the dashed lines. No detailed discussion of the weighting effects will follow, since the general error behavior is the same as that for the block weighting functions. The weighting effects are attenuated as N increases and as the contrasts in weights decrease. Improvement in the more heavily weighted portions of the response increases with the weighting contrast. Also, the attenuation of the weighting effects for certain N occurs as in the cases discussed in the preceding sections on magnitude and position effects.

The error behavior in the responses reflects the behavior of the weighting curves closely. The smooth decrease in the weighting curves is reflected by the smoothly increasing error amplitudes. The weighting curves which decrease as \( \exp(-\alpha^2_{w} u^2) \) decrease more rapidly over a shorter length than do the curves decaying at \( \exp(-\alpha_w) \) and the growth of error follows accordingly. The dependence of error behavior on weighting function shape appears much as
Figure 44. Amplitude responses of a discrete filter with exponentially weighted error. Weighting curve - dashed line; specified response - thin lines; and discrete filter sampled at $1\Delta t$ in the upper right of each plot.
Figure 45. Amplitude responses of a discrete filter with exponentially weighted error. Weighting curve - dashed line; specified response - thin lines; and discrete filter sampled at \(1\Delta t\) in the upper right of each plot.
Figure 46. Amplitude responses of a discrete filter with exponentially weighted error. Weighting curve - dashed line; specified response - thin lines; and discrete filter sampled at 1Δt in the upper right of each plot.
one would expect after observing the behavior for block weighting functions.

Further Comments on Error Weighting: Comments on some aspects of error weighting not considered in the main body of the thesis should now be made. These subjects were not included either because no experiments were done or because the experiments were too brief and inconclusive.

One point not covered in the preceding discussions is the effect of error weighting on filters at sampling intervals other than $1\Delta t$. Computer storage limitations did not permit designing filters at sample intervals less than $1\Delta t$, but several filters were designed at a sampling interval of $2\Delta t$. The frequency scale out to the Nyquist frequency is halved for a sampling interval at $2\Delta t$ and, consequently, there is less room to manipulate the error. Of course, the reverse would occur if the sample interval was cut in half. For the filters computed at $2\Delta t$ with various weighting functions, the weighting effects behaved in exactly the same manner as those for filters at $1\Delta t$. The only major differences are that the effects occur at smaller $N$ and are compressed into a smaller portion of the spectrum. It is logical to assume that the weighting effects observed when going from $\Delta t$ to $\Delta t/2$ would be similar to those observed when going from $2\Delta t$ to $1\Delta t$. 
Several configurations of block weighting functions were tried in an attempt to alter and improve the shape of the passband characteristics. Chiefly, this involved weighting various narrow portions of the passband region with small weights in order to influence the overshoot and rounding of the corners. The results were somewhat unpredictable and inconclusive. To find a combination of weights that would affect one part of the passband in the desired manner, one must proceed by trial and error, and this takes a great deal of time.

One interesting application of error weighting involves the design of filters in error norms other than least squares. In one case, that of the Chebyshev norm, this involves computing the unweighted filter and using some form of the error, and errors from successive filters, as weighting functions in the successive filter generations. It takes some difficult programming, and an investigation was not attempted in this thesis. One simple attempt was made, however, but it failed because of either a faulty program or a poor test case.
The summary in the first section of this chapter is a brief restatement of the objectives of the thesis and a discussion of the most important observations made pertaining to the determination and study of weighted error filters. The second part of the chapter is a list of conclusions derived from the study.

Summary

To reiterate, the major objectives of the thesis are: given the amplitude and phase response for a filter, determine the optimum, N-point time-domain filter using a weighted square error measure; examine the error behavior of discrete filters with uniformly weighted (unweighted) error; and examine the error behavior of discrete filters with non-uniformly weighted error. Since the case for uniform error weighting reduces to one of the common methods for obtaining discrete filters, that part of the work is not new but is
necessary for use as a reference when studying error weighting effects.

The optimum N-point filter in the weighted square error sense is determined by solving an Nth-order set of simultaneous equations. The coefficient matrix is made up of the inverse transform of the weighting function and the constant vector is made up of the inverse transform of the product of the weighting function with the specified amplitude and phase response. All the inverse transforms are integrated up to the Nyquist frequency only so that responses and weighting curves extending beyond are truncated. For the case of uniform error weighting, solution of the equations is unnecessary: one only samples the impulse response which corresponds to the frequency domain characteristics defined out to the Nyquist frequency.

The responses of the unweighted discrete filters all range through a similar set of characteristics as N is increased. The truncation effects — overshoot and broadening of the passband and the appearance of amplitudes in the reject region — decrease as N is increased. An effective way to illustrate and explain truncation and sampling effects is provided by the Fourier-series kernels. Plots of total percent square error versus N for the various filters all decrease as N is increased. The major portion of the total
square error occurs in the passband regions of the filters. The behavior of the square error plots correlates well with the behavior of the impulse responses of the filters. The error plots provide a quick and quantitative method for choosing the smallest N which will give the error level desired in the filter. Discrete filters at sampling intervals different from $1\Delta t$ behave similarly to the filters at $1\Delta t$ in all respects except for the change in the Nyquist frequency. The square error for a particular N is approximately halved by doubling the sampling interval.

Results of the study on error weighting show that improvement of the filter characteristics in the more heavily weighted region of the spectrum increases as the contrast in weights increases. The error in the more lightly weighted portions of the spectrum increases disproportionately compared to the improvements in the more heavily weighted portions as the contrast in the weights increases. Only contrasts in weights of the order of 1.0 to 0.1 or smaller showed an overall loss in error comparable to the gain in the heavily weighted portion of the spectrum.

When the contrast in weights is held constant and the length of the weighted portions of the spectrum is varied, the improvement in the more heavily weighted region increases with the length of the lightly weighted region. The total
error does not necessarily increase as the length of the lightly weighted region is increased but is distributed over the wider area. The error weighting effects all decrease and the total square error approaches that of the corresponding unweighted filters either as the contrast in weights decreases towards zero or as the length of the lightly weighted regions decreases to zero.

Error weighting effects reflect the shape of the weighting curves closely. Sharp breaks in the error weighting curves produce sharp discontinuities in the error magnitudes, whereas smoothly changing weighting curves result in gradual changes in the error magnitudes.

The weighted error filter weights in most cases bear little resemblance to the corresponding unweighted filter weights. The major changes occur at or near the extremities of the discrete filters where the weights becomes very large with respect to those near the center. A possible reason for this behavior is that the weights near the ends of the filter are responsible for defining the details of the specified response. Since error weighting results in improvement of these smaller details of the filter responses, it is more likely the end filter weights will be affected.

The nature of the improvement in the filter characteristics caused by weighting one portion of the spectrum more
than another is of similar character in all cases and is
directly comparable to the improvement recorded when \( N \) is
increased in an unweighted filter. The percent square error
vs \( N \) curves for the weighted error filters are similar in
shape to the corresponding curves for the unweighted filters
except that the entire curve has a shift whose magnitude
depends on the size of the weighting contrast. For the
examples considered in this thesis, a maximum shift of six
points could be obtained; hence, if a portion of the spectrum
can be totally ignored, a savings of up to six points seems
feasible using error weighting.

For each filter examined, there were several \( N \) for
which the error weighting seemed to have a subdued effect
and also other \( N \) for which the weighting effects were
accentuated. These effects can be explained by the pro­
jection in \( N \) caused by weighting. On the steeply sloping
portions of the unweighted error curves, a projection ahead
results in a large change, whereas on or before one of the
flatter portions of the curves a projection ahead causes
little change.

The error weighting effects are attenuated and the
weighted error filters approach the unweighted filters as
\( N \) is increased. If \( N \) is increased sufficiently, the errors
in the unweighted filter responses decrease towards zero
and, consequently, weighting the error has little or no effect.

Interpretation of the weighting effects for several N in terms of sampling functions for the lowpass filter showed highly variable results. The sampling sequences derived from the weighted error filters oscillated rapidly for small N and showed large amplitudes at the ends for larger N. The transforms of the weighted error sampling sequences were of various shapes with some of the largest peaks at a frequency of 0.5. This suggests that it is unnecessary to require that a sampling function's transform peak at zero frequency, as the Fourier-series kernels do, to obtain useful results.

For the set of filters examined, the square error could be decreased more in the first part of the spectrum with less overall loss in square error in the latter part of the spectrum by doubling the sample interval, hence truncating less, and using an unweighted filter than by applying error weighting for the same N. Of course, aliasing may make doubling the sampling interval impractical for some filters.
Conclusions

Using discrete filters with uniformly weighted error (normal least squares) as a basis for comparison, the following conclusions can be made regarding the design of discrete filters in the frequency domain with a nonuniformly weighted square error.

1. The total square error of the nonuniformly weighted error filters examined was found to be larger than that of corresponding uniformly weighted error filters; although if \( N \) is made large enough, the errors from both types of filter become equal.

2. Weighting the square error nonuniformly results in a filter whose characteristics have smaller errors in the heavily weighted portions of the spectrum and larger errors in the lightly weighted portions of the spectrum.

3. The major disadvantage of the error weighting technique is that as the contrast in the error weights increases, the errors in the lightly weighted portions of the spectrum increase disproportionately to the error reductions in the heavily weighted portions of the spectrum.
4. If a portion of the spectrum can be totally ignored, the cases considered in this thesis show that the same filter characteristics over the heavily weighted portion of the spectrum can be achieved with a non-uniformly weighted error filter as much as six points shorter than a filter with uniformly weighted error.

5. For the type of filters examined (cut-off frequency less than half the Nyquist frequency) doubling the sample interval of the filter with uniformly weighted error gave better overall characteristics than any weighting configuration used for the same $N$ and a sample interval of $1\Delta t$. 
SELECTED REFERENCES


The following discussion outlines the algorithms and programs used for computing discrete filters, weighted error discrete filters, and their spectra on the CDC-8090 computer. Fortran programs are listed for the basic unweighted filters and the cosine transform, and a fairly detailed derivation of the algorithms used is presented. Programs for the other operations are not listed either because they are very simple to write or because they are tailored for the CDC-8090 exclusively and consist mainly of operations to maximize the use of the available core storage; however, some of these techniques used are mentioned briefly. It was necessary to use three successive programs with storage of intermediate results on magnetic tape to compute a weighted error filter. On a large computer much of the programming needed for handling these operations would be unnecessary.
Filter Description

The basic building unit of the filter is a zero-phase lowpass filter with a flat passband and a linear skirt. Using various combinations of this basic component, high-pass, lowpass, bandpass, and notch filters can be generated. The frequency domain specifications are described by the four parameters $\omega_1$, $\omega_2$, $\omega_3$, and $\omega_4$. Figure I-1 illustrates how the various filters are specified. For example, a bandpass filter is designed by specifying two lowpass filters in the proper sizes and subtracting them. Since these filters will be used as discrete operators, the limits on the specifications imposed by the Nyquist frequency must be observed.

An inverse Fourier transform gives the impulse response of the basic lowpass filter. By combining the impulse responses of the basic filters, the impulse response of the desired filter can be obtained.

**Inverse Transform:** The lowpass filter in the frequency domain can be represented as:

$$A(\omega) = 1 \quad |\omega| < \omega_1$$

$$A(\omega) = \frac{\omega_2 - \omega}{\omega_2 - \omega_1} \quad \omega_1 < |\omega| < \omega_2 \quad \text{I-1}$$

$$A(\omega) = 0 \quad |\omega| > \omega_2$$
Since $A(\omega)$ is zero phase the cosine transform

$$h(t) = \int_{0}^{\omega_2} A(\omega) \cos \omega t d\omega$$

can be used. To correspond with the theory in the
chapter on minimization of the weighted square error, the $1/\pi$ term normally occurring in the transform is left out. Letting $a = 1/(\omega_2 - \omega_1)$ and breaking equation I-2 into parts we see that:

\[ h_1(t) = \int_0^{\omega_1} l \cdot \cos\omega t d\omega \]

\[ = \frac{\sin\omega_1 t}{t} \quad \text{and} \quad I-3 \]

\[ h_2(t) = a \int_{\omega_1}^{\omega_2} (\omega_2 - \omega) \cos\omega t d\omega \]

\[ = a \left[ \frac{1}{t^2} (\cos\omega_1 t - \cos\omega_2 t) + \frac{\sin\omega_1 t}{t} (\omega_1 - \omega_2) \right] \quad I-4 \]

The impulse response of the lowpass filter is:

\[ h(t) = h_1(t) + h_2(t) \]

\[ = \frac{1}{\omega_2 - \omega_1} \left[ \frac{1}{t^2} (\cos\omega_1 t - \cos\omega_2 t) \right] \quad I-5 \]
The value for \( t = 0 \) can be determined by applying l'Hospital's rule twice:

\[
\lim_{t \to 0} h(t) = \lim_{t \to 0} \frac{1}{\omega_2 - \omega_1} \left[ \frac{-\omega_1 \sin \omega_1 t + \omega_2 \sin \omega_2 t}{2t} \right] \\
= \lim_{t \to 0} \frac{1}{\omega_2 - \omega_1} \left[ \frac{-\omega_1 \cos \omega_1 t + \omega_2 \cos \omega_2 t}{2} \right] \\
= \frac{\omega_2 + \omega_1}{2}
\]

Various combinations of equation I-5 can be used to compute the impulse response of the desired filter.

**Computing Algorithm:** The computation of the filter weights \( f \), as described in the chapter on minimization, involves sampling \( h(t) \) and multiplying by \( \Delta t / \pi \). Sampling \( h(t) \) is done by computing \( h(k \Delta t) \) for various \( k \). Substituting \( k \Delta t \) into \( h(t) \) and changing from angular to linear frequency we get:

\[
h(k \Delta t) = \frac{\Delta t}{\pi} \frac{1}{2 \pi f_2 - 2 \pi f_1} \left[ \frac{\cos 2 \pi f_1 k \Delta t - \cos 2 \pi f_2 k \Delta t}{k^2 \Delta t^2} \right]
\]

\[
h(0) = \frac{\Delta t}{\pi} \cdot \frac{2 \pi f_2 + 2 \pi f_1}{2}
\]
Rearranging terms,

\[ h(k\Delta t) = \frac{1}{\pi(2\pi f_2 \Delta t - 2\pi f_1 \Delta t)} \left( \frac{\cos 2\pi f_1 \Delta t - \cos 2\pi f_2 \Delta t}{k^2} \right) \]

\[ h(o) = \frac{2\pi f_2 \Delta t + 2\pi f_1 \Delta t}{2\pi} \]

The combined terms \( f_2 \Delta t \) and \( f_1 \Delta t \) are the dimensionless frequency specifications of the filter. Equations I-7 and I-8 are the algorithms used in the following Fortran program which generates the discrete filter for \( t \geq 0 \).

Care must be taken when using the filter program on page I-7 that infinite slopes are not specified \((\omega_1 = \omega_2 \text{ or } \omega_3 = \omega_4)\) and that the Nyquist frequency is not exceeded. The program generates only the center point and positive half of a lowpass or bandpass filter which is symmetric about \( t = 0 \). To obtain a notch filter a very narrow bandpass filter must be generated and then subtracted from a wide lowpass filter.
SUBROUTINE FILTR1(HT, NT, W1, W2, W3, W4)

C
C LOWPASS-BANDPASS FILTER SUBROUTINE FILTR1
C PRODUCES ONE LOBE OF A SYMMETRICAL FILTER
C NT= THE LENGTH OF THE ONE LOBE PLUS THE CENTER POINT
C HT(I)= THE DISCRETE FILTER VALUES IN THE TIME DOMAIN
C THE FILTERS RESPONSE IS DEFINED BY THE FOUR DIMENSIONLESS
C W1= THE RESPONSE IS ZERO UP TO THIS FREQUENCY
C W2= THE RESPONSE IS LINEAR FROM W1 TO THIS FREQUENCY
C W3= THE RESPONSE IS UNITY FROM W2 TO THIS FREQUENCY
C W4= THE RESPONSE IS LINEAR FROM W3 TO THIS FREQUENCY WHERE
C THE RESPONSE IS ZERO AGAIN
C FOR A LOWPASS FILTER W1 AND W2 ARE SET EQUAL TO ZERO
C W1 AND W2 MAY BE VERY CLOSE TOGETHER BUT NEVER EQUAL
C W3 AND W4 MAY BE VERY CLOSE TOGETHER BUT NEVER EQUAL
C W4 MAY NOT EXCEED THE NYQUIST FREQUENCY AT 0.5
C
DIMENSION HT(41)
.TPI=6.2831853
.PI=3.1415926
DO 2 J=1, NT
2 HT(J)=0.
   W1=W1*TPI
   W2=W2*TPI
   W3=W3*TPI
   W4=W4*TPI
   DEL1=PI*(W2-W1)
   DEL2=PI*(W4-W3)
   IF(DELI)4,3,4
3 DELI=1.0
4 HT(1)= (W4+W3-W2-W1)/TPI
   DO 1 I=2, NT
      XI=I-1
      XI2=XI*XI
      CO1=(COSF(W2*XI)-COSF(W1*XI))/(XI2*DEL1)
      CO2=(COSF(W4*XI)-COSF(W3*XI))/(XI2*DEL2)
1 HT(I)= CO1-CO2
   W1=W1/TPI
   W2=W2/TPI
   W3=W3/TPI
   W4=W4/TPI
RETURN
END
Discrete Fourier Transforms

The actual computer program discussed here is used only for one-half of an even time function -- i.e., real frequency functions -- but the algorithms and techniques are developed for the general transform. A time or frequency function is given which has been sampled at some \( \Delta t \) or \( \Delta \omega \). The objective is to perform a numerical Fourier transform on this data.

A transform for discrete time-domain functions can be expressed as

\[
F_S(\omega) = \sum_{k=-L}^{M} f_k e^{-j\omega k \Delta t}, \tag{I-9}
\]

where \( f_k \) represents the sample amplitude and \( \Delta t \) the spacing between samples. Conversely, the time-domain function corresponding to a discrete frequency-domain function is

\[
f_S(t) = \sum_{k=-R}^{N} F_k e^{jtk \Delta \omega}. \tag{I-10}
\]

The inverse transform will not be discussed further since the computing techniques are identical to the regular transform.

Breaking the exponential in equation I-9 down into sines and cosines we get:
The respective parts in equation I-11 are referred to as a cosine transform and a sine transform.

The discrete transform is a continuous function of $\omega$ and is periodic at $2\pi/\Delta t$. Since it is impossible to handle a continuous function on a digital computer, $F_S(\omega)$ must be computed for discrete values of $\omega$.

$$F_S(\omega) = \sum_{k=-L}^{M} f_k \cos w_k \Delta t - j \sum_{k=-L}^{M} f_k \sin w_k \Delta t \quad I-11$$

The respective parts of $F_S(\omega)$ can be thought of as a function of the discrete variable $n \Delta f \Delta t$. This is the dimensionless frequency concept used in computing discrete transforms. The transform is performed with a fixed $\Delta f \Delta t$ and holds for any combination of $\Delta f$ and $\Delta t$ whose product is that constant. Fixing either $\Delta f$ or $\Delta t$ determines the magnitude of the other. The Nyquist frequency occurs at $n \Delta f \Delta t = 0.5$ on the dimensionless scale since:

$$\omega_c = \pi/\Delta t \text{ or } 2\pi n \Delta f = \pi/\Delta t \text{ and } n \Delta f \Delta t = 0.5.$$  

The maximum $n$ necessary to define the transform is then $N = 0.5/\Delta f \Delta t$. 

$$F_S(n \Delta \omega) = \sum_{k=-L}^{M} f_k \cos 2\pi n k \Delta f \Delta t - j \sum_{k=-L}^{M} f_k \sin 2\pi n k \Delta f \Delta t \quad I-12$$

$$n = 0, 1, \ldots, N$$
The expression I-12 is the algorithm used to perform a discrete Fourier transform. The major operations involve two computational loops. The \( n \) is specified in an outer loop and the summation over \( k \) for each \( n \) is done in an inside loop. To eliminate the problem of dealing with negative times in the arguments of the trigonometric functions, time functions which are not even or odd can be shifted to fall in positive time at the expense of introducing a linear phase component or can be separated into odd and even parts whose portions in negative time are unnecessary in the transform.

To speed up computation time, a recursion relation may be used to compute the many trigonometric functions instead of letting the computer evaluate every function as the argument is incremented. By using multiple angle formulas and/or sum difference relations, the computing time can be diminished a great deal. Another technique, which is even faster than those mentioned above, is described in Ralston and Wilf (1960, p. 258-262). The technique will not be described here but does incorporate another type of trigonometric identity.

The technique from Ralston and Wilf is used in the cosine transform program on page I-12 for the inner
computing loop. The trigonometric functions in the outer loop are determined using a three-term multiple-angle formula rather than incrementing their arguments and letting the computer evaluate them each time. Since the functions dealt with in the thesis are even, the transform I-12 reduces to:

\[
F_s(n\Delta\omega) = \sum_{k=-L}^{M} f_k \cos(2\pi nk \Delta t) n = 0, 1, \ldots, N
\]

\[
= f_0 + 2 \sum_{k=1}^{M} f_k \cos(2\pi nk \Delta t) \quad \text{I-13}
\]

Equation I-13 is the algorithm used in the program on page I-12.

**Error Weighting**

The error weighting programs are quite straightforward. The exponential weighting function in the frequency domain is generated by evaluating the exponential for the dimensionless arguments with the chosen decay factor. The time-domain equivalent is obtained by a discrete cosine transform. The block weighting curves in the frequency domain are simple to generate. The time-domain equivalent is computed exactly by evaluating the cosine-modulated sinc functions which represent the various shifted blocks. The weighted response
SUBROUTINE CTRAN1(X, NX, A, NA, DELFT)

SUBROUTINE CTRAN1 COMPUTES THE COSINE TRANSFORM OF AN EVEN FUNCTION
A = THE INPUT ARRAY FROM THE ORIGIN TO THE END OF THE FUNCTION
NA = THE NUMBER OF ELEMENTS IN ARRAY A
X = THE TRANSFORM VALUES
NX = THE NUMBER OF ELEMENTS IN ARRAY X
DELFT = THE DIMENSIONLESS FREQUENCY INCREMENT -- DELTA F*DELTA T

DIMENSION A(21), X(126)

ARG = 6.2831853 * DELFT
CSAVE = COSF(ARG)
C = 2. * CSAVE
COS = 1.
A1 = A(1)/2.
NSET + NA + 1

OUTER LOOP

DO 3 N=1,NX
NDEX = NSET
U1 = 0
U2 = 0
SCALE = 2. * COS

INNER LOOP

DO 1 K=2,NA
NDEX = NDEX - 1
UO = A(NDEX) + (SCALE * U1) - U2
U2 = U1
1 U1 = UO
X(N) = (A1 + COS * U1 - U2)*2.
SAVE = COS
COS = COS * C - CSAVE
CSAVE = SAVE
3 CONTINUE
RETURN
END
is computed by a multiplication of the weighting curve and the specified response and is transformed with the discrete cosine transform program to get the time-domain equivalent.

**Simultaneous Equations**

The components for the simultaneous equations are obtained from the inverse transforms of the error weighting function and the product of the weighting function and the specified response. To set up the coefficient matrix as described on page 37 the array from the transform of the weighting function is rearranged in the proper manner and stored in a two dimensional array. Since this thesis deals only with even filters, the set of equations has pairs of solution components which are identical. These solutions and their coefficients are grouped so that the matrix size is halved before solving. See the coefficient matrix on page 38.

A program using a Gauss-elimination backward substitution technique was used to solve the equations. Since this technique is generally known, no further comments are given.
Error Computations

The computation of the various forms of error in the discrete filter responses is quite simple. For the unweighted filters, the discrete weights are computed using the filter program and then transformed progressively to obtain the responses. For the weighted error filters, the solutions to the simultaneous equations are transformed to obtain the responses. The differences between the computed responses and the specified responses are used to compute total error, total square error, percent error, percent square error, mean deviations, variance, and standard deviation by the usual methods.
A CDC-8090 digital computer was used for all the computations in the thesis. The computations were done in floating point arithmetic which carries 8 significant figures and an exponent. Round off and conversion effects make the last of the eight figures uncertain, so that only seven figures are truly significant.

**Function Evaluation**

Many of the necessary computations involved the evaluation of various functions. The unweighted filter impulse responses, the specified amplitude responses, the Fourier-series kernels, and the weighting functions were computed in this manner. Evaluation of these functions for a specific argument usually involved a trigonometric look-up and several simple arithmetic operations. Checks on the computer evaluation of these functions showed that
most results were accurate to six or seven significant figures and, in the worst case, were accurate to five significant figures.

**Cosine Transform**

To check the accuracy and validity of the cosine transform program, a sequence of unit impulses was transformed and compared to a generated Fourier-series kernel of the proper N. Both sets of results were normalized to unity and were found to match out to five places beyond the decimal point. From this point on, the term "decimal places" will be used to describe accuracy. The specified filter responses, the weighting functions, and of course the responses of the N-point discrete filters and N-point weighted error discrete filters all peak at or near 1.0. The discrete filter weights, both unweighted and weighted error, have maximum values in the neighborhood of 0.1-0.2 and minimum values down to zero. For this reason, the number of decimal places to the right of the decimal point is a convenient way to express accuracy. Any change in a number of the order of 0.002 or in the third decimal place does not show in the amplitude response curves plotted and does not contribute significantly to the
square error after squaring. A more practical check on the cosine transform involved the comparison of the numerical inverse transform of a sampled frequency response with the generated impulse response. Because of the sampling effects the results matched to only four decimal places. The inherent accuracy of the cosine transform is good to five places, but the quality of the actual results obtained from a transform was downgraded because of sampling effects.

Simultaneous Equations

The most questionable point in the calculation of a weighted error filter involves the higher order simultaneous equations. The techniques for obtaining bounds on the error of large-order systems of equations are very complex and often pessimistic. To obtain an estimate of the accuracy of the solutions used in this thesis, test systems similar to those used to obtain weighted error filters were solved and the solutions compared to the true solutions. A Gauss-elimination backward-substitution technique was used to solve the simultaneous equations. This is a direct solution technique and if enough figures are carried in the computation, an exact solution is obtained. However, on the CDC-8090 computer only seven figures can be carried, and the
progressive effects of truncation can cause severe errors in the solutions of large-order systems.

A small system of order 5 with a coefficient matrix and a solution of integers was used to check the logic of the program. For this small system, an exact solution was obtained. For testing larger systems, various portions of an array of 21 filter weights from an unweighted band-pass filter were used as test solutions whose values are similar to those in real systems. The values of these filter weights ranged from a maximum of 0.14 down to a minimum of 0.0009 with both positive and negative elements. Various portions of the array were also used to construct the coefficient matrix as indicated on pages 37 and 38. The sequence of the values in the array were reversed and the positions mixed to obtain a variety of test solutions and coefficient matrices. After constructing the coefficient matrix and the solution, a matrix multiplication was done to obtain the constant vector.

Numerical solutions of systems of order nine matched the true solutions in the first six decimal places. Relative to the maximum number of 0.14 in the solution the errors in these systems would be insignificant. Several systems of order 15 were solved next and found to be accurate to 4
decimal places in the worst case and accurate to 6 places in the majority of the cases. If these filter values were transformed, errors of this size would make no change in the filter response.

Several systems of order 21 were solved next and found to vary in accuracy from 2 to 6 decimal places. Most of the solutions were accurate to 4 places or better. The worst case is illustrated in figure II-1, where the numerical solution and the true solution are plotted. Relative to the largest member of the solution the largest error is 5.2 percent in figure II-1. If the errors are considered relative to the smaller members of the solution, some of them are very large, but since the smaller members of the solution are so small, the errors are of less significance in a transform. It seems that, in the majority of cases, the 21st order systems should give reliable results and even in the worst cases, should not produce entirely useless results. Additional evidence that none of these bad cases were encountered is given in the two following paragraphs.

As discussed in the section on weighted error, weighting effects decrease as N increases. Several 41-point weighted error filters with small contrasts in
Figure II-1. Plot of true solution and worst case numerical solution for a test system of order 21. The x's below the abscissa indicate errors in sign.
weights were computed and found to match the corresponding unweighted filters closely. The percent square error in the responses were identical. It was necessary to solve a set of equations of order 21 to obtain these filters.

The weighted error filters all showed similar and regular behavior in their response characteristics. Any significant errors occurring in the solutions to the equations would have effects distributed over the entire spectrum and would occur randomly. None of the results obtained indicated this behavior.

**Overall Accuracy**

The results of the testing of the various programs used and the results obtained in the production of weighted error filters indicate that the work done should be accurate enough to define the general effects caused by weighting error. Three decimal places of accuracy is adequate to define the curves in the illustrations and the percent square error changes. More accurate calculations would not affect the curves.