INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
$L_p$-approximation by the iteratively reweighted least squares method and the design of digital FIR filters in one dimension

Barreto, José Antonio, M.S.
Rice University, 1993
RICE UNIVERSITY

$L_p$-Approximation by the Iteratively Reweighted Least Squares Method and the Design of Digital FIR Filters in One Dimension

by

José Antonio Barreto

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree Master of Science

Approved, Thesis Committee:

Dr. Charles S. Burrus, Chairman
Professor of the Department of Electrical and Computer Engineering

Dr. Ian Walker
Professor of the Department of Electrical and Computer Engineering

Dr. Richard Tapia
Professor of the Department of Mathematical Science

Houston, Texas
August, 1992
$L_p$-Approximation by the Iteratively Reweighted Least Squares Method and the Design of Digital FIR Filters in One Dimension

José Antonio Barreto

Abstract

In this thesis a new and simple to program approach is proposed in order to obtain an $L_p$ approximation, $2 < p < \infty$, based on the Iteratively Reweighted Least Squares (IRLS) method, for designing a linear phase digital finite impulse response (FIR) filter. This technique, interesting in its own right, can also be used as an intermediate design method between the least squared error and the minimum Chebyshev error criteria.

Various IRLS algorithms are evaluated through comparison of the number of iterations required for convergence. It is shown that Kahng's (or Fletcher's et al) method with a modified acceleration technique developed in this work performs better, for most practical cases, than the other algorithms examined. A filter design method which allows different norms in different bands is proposed and implemented. An important extension of this method also considers the case of different $p$'s (or different norms) in the stopband.
Acknowledgments

I would like to express my sincere thanks to my advisor Dr. Charles S. Burrus for his unconditional and steady support in the fulfillment of this thesis.

I would also like to thank Dr. Ian Walker for consenting to be in my thesis committee and for taking his time to carefully review this work.

Special thanks go to Dr. Richard Tapia for also being part of my thesis committee and for providing, many times, that particular extra-academic atmosphere in which the, very often, overwhelming nature of graduate studies was made easier.

Additionally, I would like to thank Ivan Selesnick, Jan Odegard and Ramesh Gopinath for their comments, which undoubtedly helped me shape this research.

I would also like to thank FUNDAYACUCHO, Venezuela, for providing the funds which gave me the opportunity to complete the Master’s program.

Finally, I would like to thank and dedicate this thesis to my wife, Mayita, for her never fading encouragement, for her constant support, for having enough patience to explain me that intricate mathematical language, which, at times, I was not able to decipher, and for all those other things that my writing fails to convey.
Contents

Abstract ii
Acknowledgments iii
List of Illustrations vi
List of Tables viii

1 INTRODUCTION 1
1.1 The Iteratively Reweighted Least Squares (IRLS) Filter Design Problem 1
1.2 History of the $p^{th}$-Power error approximation and the IRLS Algorithm in the FIR Filter Design 2
1.3 Contribution of this Research 3

2 The Least $p$-Power Error Design of 1-D Linear-Phase Digital FIR Filters 5
2.1 Frequency Response Formulation 5
2.2 The Least $p$-Power Error One-Dimensional FIR Filter Design and the Weighted Least Squares Formulation 8

3 The $L_p$ Norm Minimization Problem and the Iteratively Reweighted Least Squares (IRLS) Algorithm 10
3.1 Notation, Specifications and Convergence Properties of the Algorithms 11
3.1.1 Standard IRLS Method 11
3.1.2 Modified IRLS Method (Burrus' Method) 12
3.1.3 Karlovitz' Method 13
3.1.4 Extended Lawson's Algorithm 15
3.1.5 Kahng's or Fletcher's et al Algorithm 19
3.2 A Comparison of the Accelerated Kahng's Method and the Extended Lawson Algorithm 21
4 Acceleration of the Convergence in Kahng's Method and the One-Dimensional Digital FIR Filter Design 24
  4.1 A Prediction Technique as an Acceleration Scheme for Kahng's Method .................. 24
  4.2 A Perturbed Problem ...................................... 27
  4.3 The Accelerated Kahng's Method Modified .................... 30

5 $L_p$-Approximation of One-Dimensional Digital FIR Filters using Different $p$'s in Different Bands 35
  5.1 Different Norms in Different Bands with Zero Error Weighting in the Transition Band. 35
  5.2 The $L_p$ Approximation Problem with Different $p$'s in the Stopband .................... 40
  5.3 Numerical Results and Design Examples. ........................................ 42
    5.3.1 The general $p^{th}$-power error digital FIR filter design ..................... 42
    5.3.2 $L_{pp,ps}$ Filter Design .................................. 42
    5.3.3 $L_{pp,ps,1}$ FIR Filter Design .................................. 54

6 CONCLUSIONS ........................................... 63

A MATLAB PROGRAMS .................................. 65
  A.1 Kahng's Algorithm with the Modified Acceleration Technique 65
  A.2 Karlovit'z Algorithm with no Line Search .................. 67
  A.3 Modified IRLS Method (Burrus' Method) .................... 68
  A.4 The Extended Lawson Algorithm .............................. 69
  A.5 Kahng's Method (Different $p$'s in Different Bands) ...... 70
  A.6 Kahng's Method (Different $p$'s in the Stopband) ........ 72
  A.7 Kahng's method using the Prediction Technique ............ 74

Bibliography ........................................... 76
Illustrations

3.1 Modified Extended Lawson Algorithm ........................................ 17
3.2 Kahng's Algorithm and the Modified Extended Lawson Algorithm . 22
3.3 LP error vs iteration plot ...................................................... 23
3.4 L2 error vs iteration plot ...................................................... 23

4.1 Comparison of the Prediction Scheme vs. the Acceleration Technique for length-21 filter and $p = 86$ ........................................ 28
4.2 Iterations vs Transition Width for the Accelerated Kahng's Method . 31
4.3 Iterations vs Transition Width for the Extended Lawson algorithm . 31
4.4 Comparison of the Performance, i.e. Iterations vs Transition Width, for the Perturbed Problem ................................................. 32
4.5 Effect of $\mu$ on the number of iterations for different transition width (TW) specifications ..................................................... 34
4.6 Effect of $\mu$ on the number of iterations for different transition width (TW) specifications ..................................................... 34

5.1 Ideal Low-pass Filter with no Error Contribution from the Transition Band, where $f_s - f_p = TW$ ............................................. 36
5.2 The $L_{pp,ps}$ approximation for $pp = 2$ and $ps = 3 - 13$ ............... 39
5.3 Effect of TW on the stopband deviation .................................... 43
5.4 Shifting Property of the Unpredictable Behavior for Different Starting $f_p$'s ................................................................. 44
5.5 Effect of $p$ on the number of ripples ...................................... 45
5.6 Comparison of the amplitude response for an $L_p$-approx. ($p=4$) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations ............................................. 46
5.7 Comparison of the amplitude response for an $L_p$-approx. ($p=10$) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations .................................................. 47
5.8 Comparison of the amplitude response for an $L_p$-approx. ($p=40$) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations .................................................. 48
5.9 $L_2$ and Mixed $p$ ($pp=2$, $ps=6$) Frequency Response .................. 49
5.10 $L_2$ and Mixed $p$ ($pp=2$, $ps=7$) Frequency Response .............. 50
5.11 $L_2$ and Mixed $p$ ($pp=2$, $ps=10$) Frequency Response ........... 51
5.12 $L_2$ and Mixed $p$ ($pp=2$, $ps=13$) Frequency Response .......... 52
5.13 $L_{pp,ps}$-Approximation for a length-31 FIR low-pass filter with $pp = 20$
    and $ps = 2$ ...................................................................... 55
5.14 $L_{pp,ps}$-Approximation for a length-31 FIR low-pass filter with $pp = 2$
    and $ps = 20$ ..................................................................... 56
5.15 $L_{pp,ps}$-Approximation for a length-21 FIR low-pass filter with
    $pp = 40$ and $ps = 2$ .......................................................... 57
5.16 $L_{pp,ps}$-Approximation for a length-21 FIR low-pass filter with $pp = 2$
    and $ps = 40$ ..................................................................... 58
5.17 $L_{pp,ps}$-Approximation for a length-31 FIR low-pass filter with $pp = 2$
    and $ps = 20$ ..................................................................... 59
5.18 $L_{pp,ps}$-Approximation for a length-21 FIR low-pass filter with
    $pp = 40$ and $ps = 2$ .......................................................... 60
5.19 Narrow-band filter amplitude response ($Ks1 = 15$) ................. 61
5.20 Narrow-band filter amplitude response ($Ks1 = 22$) ................. 61
5.21 Wide-band filter amplitude response ($Ks1 = 12$) .................... 62
Tables

4.1 Prediction Formulas using Extrapolation Technique ............. 27
Chapter 1

INTRODUCTION

The least squared ($L_2$ norm) error and the minimum Chebyshev ($L_\infty$ norm) error or minimax error have been the most widely used criteria in the design of FIR digital filters. The former criterion produces large errors near the discontinuities in the desired frequency response, causing the well-known Gibb’s phenomenon, but at the same time its solution is easy to compute. On the other hand the $L_\infty$ criterion can be used to reduce the Gibb’s phenomenon, but the filters designed under this criterion have larger stopband energies than least-squared filters, which is in many applications undesirable.

A compromise between these two error criteria that allows for no Gibb’s phenomenon and lower stopband energies, as well as different approximation requirements for different regions on the frequency spectrum, are possible if we pose the filter design problem as a general $L_p$ norm minimization problem.

1.1 The Iteratively Reweighted Least Squares (IRLS) Filter Design Problem

Although several approaches are possible for finding the solution of the $L_p$ norm approximation problem in this work only the iteratively reweighted least squares method will be considered.

The $L_p$ norm approximation problem consists of minimizing

$$
\|\epsilon\|_p = \|A(f, h) - A_d(f)\|_p
$$

(1.1)

where $h$ is the vector of parameters sought, $A(f, h)$ is a linear combination of continuous and independent functions $\phi_1(f), \phi_2(f), \cdots, \phi_{L-1}(f)$ as it is shown below

$$
A(f, h) = \sum_{n=0}^{L-1} h_n\phi_n(f)
$$

(1.2)

and $A_d(f)$ is the desired frequency response.
By definition of the $L_p$ norm, equation (1.1) can be written as
\[ \|e\|_p = \left( \int_0^{FS} |A(f, h) - A_d(f)|^p df \right)^{1/p} \]  
(1.3)
for $2 \leq p < \infty$ and where $[0, FS]$ corresponds to the frequency interval on which $A(f, h)$ and $A_d(f)$ are defined.

However the value of $h$ that minimizes $\|e\|_p$ also minimizes $\|e\|_p^p = \varepsilon^p$ [11], and the problem is better posed as the minimization of
\[ \varepsilon^p = \int_0^{FS} |A(f, h) - A_d(f)|^p df. \]  
(1.4)

The $L_p$ approximation problem can then be expressed as a weighted least squares problem by minimizing
\[ \varepsilon^p = \int_0^{FS} w |A(f, h) - A_d(f)|^2 df \]  
(1.5)
where
\[ w(f, h) = |A(f, h) - A_d(f)|^{p-2}. \]  
(1.6)

This problem can not be solved exactly because the solution is needed to find the weights. An iterative solution is then required, which can be attained by initially setting $w = 1$, and finding the solution of a standard least squares problem, which can be used as the initial weighting function value for the next iteration. Thus at each iteration the weights are up-dated from the previous error and the problem resolved when convergence is reached. This process is known as the iteratively reweighted least squares method (IRLS).

1.2 History of the $p^{th}$-Power error approximation and the IRLS Algorithm in the FIR Filter Design

The use of the $p^{th}$-power error criterion to evaluate the performance of electric circuits and systems was first introduced, to our knowledge, by Temes and Zai in [44], who generalized a least-squares/Taylor technique introduced in [43] to a least $p^{th}$/Taylor method, $p > 2$.


In 1973, Maria and Fahmy [25] extended Fischl’s results to the non-equidistant case using exponentials to obtain a solution to the $l_p$ approximation problem, with $p$ a positive even integer.

More recently, Távora in [42] suggested a new iterative approach to the method of successive projections based on an acceleration technique derived from the method of parallel tangents.

On the other hand, the IRLS algorithm has been widely used in the FIR filter design to obtain minimax filters in one and two dimensions, specially in the form of Lawson's algorithm.


Two years later Suk and Algazi [2] developed a technique for the iterative design of FIR filters using weighted least squares.

Algazi, Suk, and Rim [3] in 1986 relate their previous work in the iterative design of FIR filters with Lawson's algorithm to exploit the relationship between the weighted least square (WLS) and the Chebyshev approximations.

In 1992, Lim, Lee, Chen and Yang [22] presented a new iterative algorithm that derives the least squares frequency response weighting function which produces a quasi-equiripple design and that is faster than Lawson's algorithm.

The iteratively reweighted least squares algorithm has also been used in other areas of signal processing such as frequency estimation [36] and the design of FIR filters in the discrete coefficient space [23].

1.3 Contribution of this Research

The purpose of this thesis is to study, analyze and implement several algorithms that can be used to design one-dimensional FIR digital filters which are optimal in the least $p$-power error sense. Some of these algorithms have never been used before in any FIR digital filter design as is the case of Kahng's (or Fletcher et al [11]) algorithm [15] and Karlovitz's algorithm [16]. The former algorithm when properly accelerated presents very promising characteristics in terms of speed of convergence and number of iterations when compared with the extended Lawson's algorithm [34] (or Rice-Usow-Lawson Algorithm).

In chapter two the general frequency response formulation for the one-dimensional discrete FIR digital filter is presented, as well as the formulation that exploits the
symmetry of the linear phase case requiring separate expressions for odd and even length filters. The least \( p \)-power error design of FIR filters in its own right is developed next and its characteristics are also discussed.

In chapter three the different methods are presented in detail and also their similarities and differences are discussed in the context of the \( L_p \) norm approximation problem. Besides, some of the convergence properties germane to the discrete FIR filter design problem are examined.

Kahng's method and the effect of different acceleration schemes in the rate of convergence are examined and discussed in chapter four.

In chapter 5 an important extension of the \( L_p \) approximation problem is introduced by allowing different \( p \)'s in different bands of the frequency spectrum, which permits the design of filters that have an \( L_2 \) error measure in the passband and nearly Chebyshev \( (L_\infty) \) error measure in the stopband, or vice-versa. This approach can be further extended to allow different \( p \)'s in the same band of interest such as the stopband. This method can produce discrete FIR digital filters with mixed norm in the stop band similar to those presented by Adams in [1], but with a greater degree of flexibility due to the inherent nature of the method used in this work which allows the segmentation of the frequency spectrum into several sub-bands of variable width.

Finally, in chapter 5 various numerical results and design examples are presented.
Chapter 2

The Least $p$-Power Error Design of 1-D Linear-Phase Digital FIR Filters

The digital FIR filters considered in this work are characterized by the following parameters:

- $f_p$, the passband edge frequency,
- $f_s$, the stopband edge frequency,
- $L$, the length of the impulse response $h$,
- $h = \{h_0, h_1, \cdots, h_{L-1}\}$, the filter coefficients or adjustable parameters of the filter.
- $A(f, h)$, the frequency response vector of the filter as a function of frequency $(f)$ for a given set of filter coefficients $h$,
- $A_d(f)$, the desired frequency response vector as a function of frequency.

2.1 Frequency Response Formulation

The frequency response can be expressed by the discrete-time Fourier transform (DTFT) of the impulse response $h$ as

$$H(f) = \sum_{n=0}^{L-1} h_n e^{-j2\pi fn}$$  \hspace{1cm} (2.1)

where $H(f)$ is a periodic function of $f$ with period $1 \text{ Hz}$.

The complex frequency response $H(f)$ can be written in terms of a real valued amplitude $A(f)$ and a phase function $\theta(f)$ [31] as

$$H(f) = A(f)e^{j\theta(f)}$$  \hspace{1cm} (2.2)
where $A(f)$, the amplitude response function, may be positive or negative, and $\theta(f)$ is a continuous function.

In general, for a causal linear-phase filter, the phase function is given by

$$\theta(f) = -M2\pi f$$

(2.3)

where $M$ is the smallest constant of linearity given by

$$M = \frac{L - 1}{2}$$

(2.4)

which also requires a special even symmetry in $h$ of the form

$$h_n = h_{L-1-n}.$$  

(2.5)

Using (2.1), (2.2) and (2.4) the amplitude response can then be expressed as

$$A(f, h) = \sum_{n=0}^{L-1} h_n \cos(2\pi f(M - n)).$$

(2.6)

If the symmetry of $h_n$ in (2.5) is used, a simplification of the above expression can be found for $L$ odd as

$$A(f, h) = \sum_{n=0}^{M-1} 2h_n \cos(2\pi f(M - n)) + h_M$$

(2.7)

and for $L$ even as

$$A(f, h) = \sum_{n=0}^{L/2-1} 2h_n \cos(2\pi f(M - n)).$$

(2.8)

These formulas can be even further simplified by defining a new set of coefficients, $a_n$, for $L$ odd such that

$$a_n = \begin{cases} 
2h_n & \text{for } 0 \leq n \leq M - 1 \\
h_M & \text{for } n = M \\
0 & \text{Otherwise}
\end{cases}$$

(2.9)

so that (2.7) becomes

$$A(f, a) = \sum_{n=0}^{M} a_n \cos(2\pi f(M - n))$$

(2.10)

and for $L$ even such that

$$a_n = \begin{cases} 
2h_n & \text{for } 0 \leq n \leq L/2 - 1 \\
0 & \text{Otherwise}
\end{cases}$$

(2.11)
so that (2.8) becomes

\[ A(f, a) = \sum_{n=0}^{L/2-1} a_n \cos(2\pi f(M - n)) \]  \hspace{1cm} (2.12)

A similar formulation can be found for the more general case where the phase response \( \theta(f) \) is given by

\[ \theta(f) = -M2\pi f + \frac{1}{4} \text{Hz}. \]  \hspace{1cm} (2.13)

In this case the impulse response is odd-symmetric and the amplitude response formulas have sine functions rather than cosine functions [31].

To calculate the amplitude response numerically for \( L \) odd, it is necessary to evaluate (2.10) on a dense grid of frequencies, which is denoted here by a set of \( LF \) samples, linearly distributed from \( f = 0 \text{ Hz} \) to \( f = 0.5 \text{ Hz} \). Whence, equation (2.10) can now be written as

\[ A(f_k, a) = \sum_{n=0}^{L/2-1} a_n \cos(2\pi f_k(M - n)), \quad \text{for } k = 0, 1, \ldots, LF - 1 \]  \hspace{1cm} (2.14)

which leads to a set of linear equations that can be expressed in matrix form as

\[ A = Ca \]  \hspace{1cm} (2.15)

where \( A \) is an \( LF \) by 1 vector of the samples of the amplitude response of equation (2.10) \( C \) is the \( LF \) by \( M \) real matrix of cosines from equation (2.10) and \( a \) is the \( M \) by 1 vector of filter coefficients related to the impulse response by (2.9). A similar set of equations can be written for \( L \) even by using (2.12) and (2.11).

This formulation becomes a filter design method [31] by giving the samples of a desired amplitude response as \( A_d(f_k) \) and solving (2.15) for the filter coefficients \( a_n \). If the number of independent frequency samples is equal to the number of independent filter coefficients and \( C \) is not singular, this is the frequency sampling filter design method and the frequency response of the designed filter will interpolate the specified samples. But, if, instead, a larger number of samples, and by larger it is assumed at least five times the number of filter coefficients \( L \), the solution of (2.15) may be found approximately by minimizing a given error norm, where the error function may be defined as

\[ \varepsilon(f_k, a) = A(f_k, a) - A_d(f_k) = Ca - A_d(f_k). \]  \hspace{1cm} (2.16)

It is the purpose of this work to explore and study the case when the error norm minimized is the \( L_p \) norm, for \( 2 < p < \infty \), as an intermediate design method between
the two most common used error norms in the filter design problem, the $L_2$ and $L_\infty$ error norms.

### 2.2 The Least $p$-Power Error One-Dimensional FIR Filter Design and the Weighted Least Squares Formulation.

The $L_p$ approximation problem for the FIR filter is to find $a$ such that the error function

$$\epsilon = A(f, a) - A_d(f)$$

is minimized in the $L_p$ norm sense. Using the $L_p$ norm definition, the problem can be posed as minimizing

$$\| \epsilon \|_p = \left( \int_0^{FS} |A(f, a) - A_d(f)|^p \right)^{1/p}.$$  

(2.18)

In order to use a formulation better suited for numerical work the integration can be approximated by a summation [17] giving

$$\| \epsilon \|_p = \left( \sum_{k=0}^{LF-1} |A(f_k, a) - A_d(f_k)|^p \right)^{1/p}.$$  

(2.19)

but minimizing (2.19) is equivalent to minimizing

$$\| \epsilon \|_p^p = \sum_{k=0}^{LF-1} |A(f_k, a) - A_d(f_k)|^p.$$  

(2.20)

It is well known that there are not analytical methods which can be used to solve this approximation problem, therefore, it is necessary to use an iterative method such as the iteratively reweighted least squares method (IRLS) which, by appropriately choosing the weighting function, allows us to find a solution to the $L_p$ approximation problem as the limit to the successive approximations of a weighted least squares problem.

By setting the weights to be

$$w_k = |A(f_k) - A_d(f_k)|^{p-2},$$

(2.21)

the $p^{th}$-power error norm can be defined as a weighted least squares as follows

$$\| \epsilon \|_p^p = \sum_{k=0}^{LF-1} w_k (A(f_k, a) - A_d(f_k))^2.$$  

(2.22)
or in matrix notation using (2.16) as

\[ \| \varepsilon \|_p^2 = (Ca - Ad(f_k))^T(W^{1/2})^T W^{1/2} [Ca - Ad(f_k)]. \]  \hspace{1cm} (2.23)

where \( W = \text{diag}(w_0, w_1, \ldots, w_{L_{\text{F-1}}}) \). This can be minimized with respect to \( a \) by solving the weighted normal equations

\[ C^T(W^{1/2})^T W^{1/2} Ca = C^T(W^{1/2})^T W^{1/2} Ad(f_k). \] \hspace{1cm} (2.24)

Solving the above system is a direct method of designing a least \( p \) power error FIR filter using IRLS. Unfortunately, this method applied directly does not always lead to convergence and has numerical problems for many practical cases \([45]\).

In the next chapter several algorithms are presented and discussed which overcome the problems of using the IRLS problem directly and therefore permitting one to find a solution to the \( p \) power error design problem of FIR filters.
Chapter 3

The \( L_p \) Norm Minimization Problem and the Iteratively Reweighted Least Squares (IRLS) Algorithm

It is well known that the \( L_p \) norm is smooth and strictly convex for \( 2 \leq p < \infty \) [45, 21], hence the solution for the minimization of the \( L_p \) norm occurs at a point where the partial derivatives of

\[
\| \epsilon \|_p = \| A(f, a) - A_d(f) \|_p
\]

with respect to \( a \) become zero.

However, the value of \( a \) which minimizes \( \| \epsilon \|_p \) also minimizes \( \epsilon^p = \| \epsilon \|_p^p \) [11], and the problem is better posed as the minimization of

\[
\epsilon^p = \| A(f, a) - A_d(f) \|_p^p.
\]

It is known from chapter two (2.20) that equation (3.2) can be written as

\[
\epsilon^p = \sum_{k=0}^{LF-1} |A(f_k, a) - A_d(f_k)|^p
\]

or

\[
\epsilon^p = \sum_{k=0}^{LF-1} |C_k a - A_d(f_k)|^p
\]

where \( C \) corresponds to the \( k^{th} \) row of the \( LF \times M \) matrix of shifted cosines as given by equation (1.2) and \( a = (a_0, a_1, \ldots, a_{M-1}) \in \mathbb{R}^M \).

Let the residual vector be defined as

\[
\xi = C a - A_d
\]

then, differentiating (3.4) results in

\[
\frac{\partial \epsilon^p}{\partial a_n} = \sum_{k=0}^{LF-1} p|\epsilon_k|^{p-1} \text{sgn}(\epsilon_k) C_{kn}
\]
where $\epsilon_k$ corresponds to the $k^{th}$ element of $\epsilon$ and $\text{sgn}[:]$ is the sign function given by

$$
\text{sgn}[:] = \begin{cases} 
-1 & Ca < A_d \\
0 & Ca = A_d \\
1 & Ca > A_d 
\end{cases}
$$

or equivalently

$$
\text{sgn}[(\epsilon)_k] = \frac{\epsilon_k}{|\epsilon_k|}.
$$

This results in a system of equations, called the normal equations, which can be written in matrix notation as

$$
\xi^T WC = 0
$$

or

$$
(Ca - A_d)^T WC = 0
$$

where the common factor $p$ in (3.5) has been divided out and $W$ is a diagonal weighting matrix with $(k,k)$ element $|\epsilon_k|^{p-2}$.

It is the purpose of this chapter to present a number of algorithms that have been proposed to find the vector $a$ that satisfies (3.9) for $2 < p < \infty$.

### 3.1 Notation, Specifications and Convergence Properties of the Algorithms

#### 3.1.1 Standard IRLS Method

The standard IRLS method is essentially a fixed point iteration where the algorithm is started by making the weighting matrix $W$ equal to the identity matrix $I$, and then solving for $a$, at the $m^{th}$ iteration,

$$
(W^m)^{1/2}Ca = (W^m)^{1/2}A_d
$$

in the least square sense. Until convergence is achieved, the new weighting matrix is found by

$$
W^{m+1} = \text{diag} \left[|\epsilon|^p \right]^{p-2}
$$

and (3.10) is solved again. Hence the standard IRLS recursion follows for the $m^{th}$ iteration.
1. set \( W_0 = I \), \((I = \text{Identity Matrix}) \)

   for \( m = 0, 1, \cdots \)

2. Solve \((W^{1/2})^m Ca^m = (W^{1/2})^m A_d\) in the least square sense.

3. \( \xi^m = Ca^m - A_d \),

4. \( W^{m+1} = \text{diag}(\|\xi^m\|^{p-2}) \),

5. continue until convergence is achieved.

Unfortunately this algorithm does not converge for most practical cases, Merle and Späth [27] has reported that this algorithm converges for \( p > 2 \) only when \( 2 < p < 2 + \alpha \), with \( \alpha \approx 0.5 \).

### 3.1.2 Modified IRLS Method (Burrus’ Method)

This new approach proposed by Burrus [5] seeks to improve the convergence of the standard IRLS by partially updating the weights at each iteration. The partial update procedure is achieved by setting the weighting matrix to be \( q \) times the weights at the \( m^\text{th} \) iteration plus \((1 - q)\) times the weights from the \( m^\text{th} \) \(-1\) iteration, where \( q \in (0, 1) \).

As for the standard IRLS method, the modified IRLS algorithm is started by making \( W^0 = I \), where \( I \) is the identity matrix, then solve the system given by equation (3.10) in the least square sense. If convergence is achieved, stop; if not, find the new weighting matrix by first computing \( W^{m+1} \) as in (3.12) and proceed to partially update the weighting matrix in the following way

\[
W^{m+1} = qW^{m+1} + (1 - q)W^m.
\]

(3.13)

The Modified IRLS recursion can be written as

1. set \( W^0 = I \),

   for \( m = 0, 1, \cdots \)

2. Solve \((W^{1/2})^m Ca^m = (W^{1/2})^m A_d\) in the least square sense.

3. \( \xi^m = Ca^m - A_d \),
\[ W^{m+1} = \text{diag}([|E^n|^p]^{-2}), \]

\[ W^{m+1} = qW^{m+1} + (1 - q)W^m, \]

6. continue until convergence is achieved.

Although this approach improves the convergence of the standard IRLS algorithm, it depends heavily on the selection of the value of \( q \) to attain convergence. It has been found experimentally that the larger the value of \( p \), the smaller the value of \( q \) must be for convergence. Convergence properties for this method have not yet been established.

3.1.3 Karlovitz’ Method

Karlovitz [10] developed an iterative method based on the solution of a sequence of weighted least square problems for finding best \( L_p \) approximations, \( p \) an even integer, and Chalmers, Egger and Taylor [6] extended these results for \( 1 < p < \infty \). Theorem 1 in Karlovitz’ paper [16] is reproduced here without proof to clarify the relationship between the \( L_p \) approximation problem and the weighted least squares problem. Karlovitz’ theorem states that

**Theorem 3.1** Let \( p = 2m, 2 \leq m, \infty \), and let \( f \) and \( M \) satisfy

\[ A_d(f) \neq A(f), \quad \text{a.e. for all } A \in M \]  \hspace{1cm} (3.14)

where \( M \) is a closed convex subset and the unique point in \( M \) which lies nearest to \( f \) in the \( L_p \) norm, \( p \) finite, is denoted by \( A^* \).

Then for any initial approximation \( A^0(f) \in M \), the sequence

\[ A^0, B^0, A^1, B^1, \ldots, A^n, B^n, \ldots \]  \hspace{1cm} (3.15)

where \( B^n \in M \) is uniquely defined by

\[ \|A_d(f) - B^n(f)\|_{L^{2,w}} \leq \|A_d(f) - A(f)\|_{L^{2,w}} \text{ for all } A(f) \in M \]  \hspace{1cm} (3.16)

with \( w = (A_d - A^n)^{p-2} \), satisfies the following:

1. For each \( n \), either \( A^n = A^* \) or

\[ \|A_d(f) - A^{n+1}\|_{L^p} < \|A_d(f) - A^n\|_{L^p}; \]
where $A^{n+1}$ is uniquely defined by
\[
A^{n+1} = A^n + \lambda^n (B^n - A^n)
\]
(3.17)

and $\lambda^n$ minimizes $\phi(\lambda) = \|A_d - A^n - \lambda(B^n - A^n)\|_{L^p}$ for all real $\lambda$.

2. $B^n, A^n \to A^*$, uniformly, as $n \to \infty$;

3. for each $n$, either $A^n = A^*$ or
\[
\|A_d - B^n\|_{L^2,w} \leq \|A_d - A^*\|_{L^2,w} < \|A_d - A^n\|_{L^2,w},
\]

where $w = (A_d - A^n)^{p-2}$.

It can be observed from part (a) of the above theorem that Karlovitz’ algorithm requires a line search procedure to find the value of $\lambda$ at each iteration. The parameter $\lambda$ can be compared with the $q$ parameter used in the modified IRLS method but with the difference than $\lambda$ is changed at each iteration (variable step size) while $q$ is kept fixed throughout the iteration process.

For all practical purposes and in order to avoid the computational cost of performing a line search at each iteration, Karlovitz’ method was implemented, in this work, without the line search procedure, and the value of $\lambda$ ($0 < \lambda < 1$) was chosen in a similar manner than $q$ for the modified IRLS. Whence the Karlovitz’ recursion is

1. set $W^0 = I$

2. $A^0 = Ca^0$

for $m = 0, 1, \cdots$

3. $\varepsilon^m = A^m - A_d$

4. $W^m = \text{diag}((\varepsilon^m)^{p-2})$

5. Solve $(W^{1/2})^m Ca^m = (W^{1/2})^m A_d$ in the least square sense.

6. $A^{m+1} = A^m + \lambda(Ca^m - A^m)$

7. continue until convergence is achieved.
Instead of up-dating the frequency solution vector as done in step (6) of the above algorithm, one could choose to up-date the solution vector \( \mathbf{a} \) based on the fact that the discrete Fourier transform (dft) is a linear transformation. In this work the former way of up-dating (i.e. in the frequency domain) has been chosen over the latter in order to allow for the implementation of Karlovitz’ algorithm for the case of different \( p \)'s in different bands (see chapter 5). However, the numerical performance of the algorithm when \( \mathbf{a} \) is up-dated rather than \( \mathbf{A} \) proved to make the convergence slightly faster in two or three iterations, but it does not allow for the use of different norms in different bands of the frequency spectrum.

In theory this algorithm will converge for any initial approximation \( A_0 \in M \) but in practice it has been observed that for very large \( p \)'s the weighting function becomes very close to zero causing an ill-conditioned system of equations, which have an infinite number of solutions for \( \mathbf{a}^n \), but after a finite number of iterations it will find a solution which belongs to the subset \( M \) and eventually the algorithm will converge.

The numerical data obtained in this work shows that, for all the cases tested, choosing the parameter \( \lambda \) very small (\( \lambda < 0.005 \)) and having as initial approximation an \( L_2 \) solution, Karlovitz’ algorithm converges, although very slowly. To always choose \( \lambda \) small is not practical, in particular for small \( p \)'s (\( 4 < p < 20 \)) where a larger value of \( \lambda \) would accelerate considerably the convergence. Therefore an efficient, but very slow, implementation of Karlovitz’ method requires

\[
\min \phi(\lambda) = \|A_d - A^{m+1}\|_{L^p}
\]  

(3.18)

where \( A^{m+1} = A_n + \lambda^m (B^n - A^n) \), which is the line search procedure proposed by Karlovitz in his paper [16].

### 3.1.4 Extended Lawson’s Algorithm

Rice and Usow [34] extended the results for best \( L_\infty \) approximations obtained by Lawson [19, 34] to compute \( L_p \) approximations, \( 2 < p < \infty \), as the limit of a special sequence of best weighted \( L_p \) approximations with \( p \) fixed. This computation is possible based on the following two theorems, which are presented below without proof.

**Theorem 3.2** (Motzkin and Walsh ([28])) Let \( \{\phi_i(f)\} \) be a Chebyshev set and define

\[
A(f, a) = \sum_{n=0}^{M-1} a_n \phi_n(f),
\]

(3.19)
where $\mathbf{a}$ denotes the parameter vector $(a_0, a_1, \cdots, a_{M-1})$.

Then, given $H(f)$ continuous on $[0,1]$ and $1 < q < p \leq \infty$, we have three pairs of identical sets:

1. \{ $\mathbf{a}|A(f, \mathbf{a})$ is a best weighted $L_p$ approximation to $H(f)$ on $[0,1]$ \}
   \{ $\mathbf{a}|A(f, \mathbf{a})$ strongly interpolates $H(f)$ on $[0,1]$ \}.

2. \{ $\mathbf{a}|A(f, \mathbf{a})$ is a best weighted $L_1$ approximation to $H(f)$ on $[0,1]$ \}
   \{ $\mathbf{a}|A(f, \mathbf{a})$ weakly interpolates $H(f)$ on $[0,1]$ \}.

3. \{ $\mathbf{a}|A(f, \mathbf{a})$ is a best weighted $L_p$ approximation to $H(f)$ on $[0,1]$ \}
   \{ $\mathbf{a}|A(f, \mathbf{a})$ is a best $L_q$ approximation to $H(f)$ on $[0,1]$ \}.

**Theorem 3.3** The conclusions of the previous theorem are valid if the interval $[0,1]$ is replaced by a finite point set $\mathbf{F} \subset [0,1]$ upon which the approximation is made.

From the third set of theorem 2 it can be seen that a best $L_p$ approximation for $2 < p < \infty$ can be found by computing a certain weighted least-squares approximation. Rice-Usow-Lawson's algorithm, or simply the extended Lawson's algorithm, consists of generating the required weighting function in the following way

$$w_k^{m+1} = \frac{((w_k^m)^\alpha |e_k^m|\beta)}{\sum_{k=0}^{Lp-1}((w_k^m)^\alpha |e_k^m|\beta)},$$

where $\alpha$ and $\beta$ are positive and satisfy the following relationship

$$\alpha(p-2) + \beta = p - 2.$$  \hfill (3.21)

From Rice and Usow's experience the best choice for faster convergence is given in the case $\alpha = \beta$, which then implies

$$w_k^{m+1} = \frac{(w_k^m |e_k^m|^{p-2/p-1})}{\sum_{k=0}^{Lp-1}(w_k^m |e_k^m|^{p-2/p-1}).}$$  \hfill (3.22)

The convergence of the above scheme can be further improved by introducing a convergence control parameter $g$. Whence, equation (3.20) can be expressed as

$$w_k^{m+1} = \frac{(w_k^m |e_k^m|^{p-2/g(p-1)})}{\sum_{k=0}^{Lp-1}(w_k^m |e_k^m|^{p-2/g(p-1))}.}$$  \hfill (3.23)

Adjusting the value of the parameter $g$ the rate of convergence of the algorithm can indeed be improved. See Fig. 3.1.

Thus, the corresponding extended Lawson's algorithm for $L_p$ approximation can be stated
Figure 3.1  Modified Extended Lawson Algorithm
1. $W^0 = I$
   for $m = 0, 1, \cdots$

2. $\varepsilon^m = Ca^m - A_d$

3. $W^{m+1} = \text{diag} \left[ \frac{\{w^m|\varepsilon^m\}(p-2)/(p-1)}{\sum \{w^m|\varepsilon^m\}(p-2)/(p-1)} \right]$

4. Solve $(W^{1/2})^{m+1}Ca^m = (W^{1/2})^{m+1}A_d$ in the least square sense.

5. continue until convergence is achieved.

A serious drawback, although very infrequent, to the extended Lawson’s algorithm is that the error vector $(\varepsilon(f_k))$ “accidentally” becomes zero at a point $f_0$ of $F$ in the early stages of the algorithm setting $w^m(f_k) = 0$ permanently, therefore preventing the $L_p$ approximation attained, from being the best $L_p$ approximation.

According to Rice and Usow [11], there are two possible ways to remedy this situation. The first possibility is to restart the algorithm with a specific choice for $w^l(f_k)$, for $k = 0, 1, \cdots, LF - 1$, as established in the following theorem.

**Theorem 3.4** If $P_0$ is a proper subset of $F$, then the algorithm may be restarted with

$$w^l(f_k) = (1 - \lambda)w^0(f_k) + \lambda u(f_k), \quad 0 \leq \lambda < 1, \quad (3.24)$$

where $u(f_k) = 0$ for $f_k \neq z$ and $u(z) = 1$, where $z \in F - P_0$ and $L(A_0, z) - H(z) \neq 0$.

The second possibility is to modify the definition of the weighting function given by (3.22) and set

$$w^{m+1}(f_k) = \begin{cases} w^m(f_k)((\varepsilon(f_k))(p-2)/(p-1), & \varepsilon^m(f_k) \neq 0, \\ w^m(f_k), & \varepsilon^m(f_k) = 0 \end{cases}, \quad (3.25)$$

for $k = 0, 1, \cdots, LF - 1$ and where the normalizing factors are omitted for simplicity.

The latter modification led to divergence for the majority of the cases tested in this work. On the other hand, it was never the case that the error curve became accidentally zero. This makes the restarting alternative the most adequate choice to overcome this problem, if it arises.
The extended Lawson's algorithm implemented as described above proved to perform better than the three previous algorithms presented in this section. Performance here is measured in terms of the number of iterations required for the algorithm to converge and the robustness of the method to the variations in the value of \( p \). While the other three methods present very poor numerical characteristics for \( p \geq 20 \) the extended Lawson's algorithm showed less sensitivity to variations in \( p \) and it was possible to obtained solutions for very high \( p \)'s, \( p > 100 \). However, the convergence becomes very slow as \( p \) increases.

### 3.1.5 Kahng's or Fletcher's et al Algorithm

A very efficient algorithm for determining \( L_p \) approximations was developed almost simultaneously and independently of each other by Kahng [15] and by Fletcher, Grant and Hebden [11]. The algorithm is an adaptation of Newton-Rapson's method and is applicable both on a finite interval and on a finite point set with \( 2 < p < \infty \). Contrary to the extended Lawson's algorithm, Kahng's or Fletcher's algorithm does not require \( \{ \phi(f) \} \), as defined in (1.2), to be a Chebyshev set.

The derivation of the method follows the iterative scheme shown at the beginning of this chapter which leads to the standard IRLS method that almost always diverges, as has been previously seen. In order to assess convergence the following scheme is proposed and later proved in [11, 15]

\[
a^{m+1} = a^m + \alpha(a^{m+1} - a^m)
\]  

(3.26)

or in the frequency domain

\[
A^{m+1} = A^m + \alpha(Ca^m - A^m)
\]  

(3.27)

where

\[
\alpha = \frac{1}{p-1} .
\]  

(3.28)

The algorithm for best \( L_p \) approximation is then as follows

1. set \( W^0 = I \)
2. \( A^0 = Ca^0 \)

for \( m = 0, 1, \cdots \)
3. $\xi^m = A^m - A_d$

4. $W^m = \text{diag}[(\xi^m)^{p-2}]$

5. Solve $(W^{1/2})^m C a^m = (W^{1/2})^m A_d$ in the least square sense.

6. $A^{m+1} = A^m + \frac{1}{p-1} (C a^m - A^m)$

7. continue until convergence is achieved.

Although derived differently this method and Karlovitz' method are equivalent and they only differ from each other in the line search procedure which is absent in Kahng’s or Fletcher’s method. It should be observed that if in Karlovitz’ method $\lambda = \frac{1}{p-1}$ the two algorithms are exactly the same.

Convergence of Kahng’s or Fletcher’s et al method is slow when the starting point $A^0$ is not sufficiently close to the best approximation $A^*$. Otherwise the convergence is quadratic which is consistent with the method being a modification of Newton-Rapson’s method.

Kahng claims in [15] that this algorithm, in contrast with the extended Lawson’s method, needs not be restarted given that the method will not “accidentally” set the weighting function to zero at any point.

In order to accelerate convergence, Kahng and Fletcher et al propose the following acceleration scheme

$$p_0 = 2, \quad p_m = \min(p, 2p_{m-1}) \quad (3.29)$$

which is guaranteed to converge, since $p_m$ becomes the actual $p$ after several iterations, besides the time required to carry out one iteration for each one of the intermediate steps $a(p)$, i.e. $a(2)$, $a(4)$, $a(8)$, $a(16)$, is approximately equal [11] to the time required for one higher $p$, say $a(32)$.

In this work it is shown that Kahng’s or Fletcher’s et al method with a modified acceleration technique shows faster convergence, for most practical cases, than all the previous methods discussed in this section. The results are particularly promising for $2 < p < 100$, since for $p > 100$ some numerical problems arise, in particular for long filters, say $L > 21$, as a consequence of taking the $(p-2)^{th}$ power when $p > 15$, which makes this method less robust than the Extended Lawson algorithm.
3.2 A Comparison of the Accelerated Kahng’s Method and the Extended Lawson Algorithm

In order to obtain a comparison between the Extended Lawson algorithm and the accelerated Kahng’s method the following stopping criterion, suggested in [27],

$$\sum_{k=1}^{n} |a_k^{(m+1)} - a_k^{(m)}| < \zeta \sum_{k=1}^{n} |a_k^{(m+1)}|$$  \hspace{1cm} (3.30)

is used, where $\zeta$ can be heuristically chosen to meet a given tolerance. Also, the algorithms were tested in terms of their performance to variations in the filter’s transition widths.

Under these conditions, it was observed that the number of iterations required for convergence in the Extended Lawson algorithm, when $\zeta = 5 \times 10^{-4}$, is comparable, for most cases, to the number of iterations needed for the accelerated Kahng’s method to converge when $\zeta = 5 \times 10^{-7}$. Furthermore, when the same accuracy was required, say $\zeta = 5 \times 10^{-7}$, for both algorithms, Kahng’s method with the acceleration scheme clearly outperformed the Extended Lawson algorithm, except for few particular cases. See Fig. 3.2.

When convergence was assumed to take place once the $p_{th}$-power error norm curve leveled out, the Extended Lawson algorithm showed faster initial convergence than the accelerated Kahng’s method. See Fig. 3.3. On the other hand, when convergence was assumed once the least squares error norm curve leveled out, Kahng’s method with the acceleration technique converged faster than the Extended Lawson algorithm. See Fig. 3.4. In other words, the accelerated Kahng’s method, although having slower initial convergence, approaches the best $L_p$ approximation faster than the Extended Lawson algorithm.

The abrupt increases in the number of iterations showed by the accelerated Kahng’s method for some particular cases is an inherent fact of the acceleration technique proposed in [11, 15] when applied to the filter design problem as it will be seen in chapter 4.
Figure 3.2  Kahng's Algorithm and the Modified Extended Lawson Algorithm
Figure 3.3  LP error vs iteration plot

Figure 3.4  L2 error vs iteration plot
Chapter 4

Acceleration of the Convergence in Kahng’s Method and the One-Dimensional Digital FIR Filter Design

Kahng’s method with the acceleration scheme as proposed in [11, 15], reproduced below for convenience

\[ p_0 = 2, \quad p_m = \min(p, 2p_{m-1}), \quad (4.1) \]

was found by Kahng and Fletcher et al in [11, 15] to be effective and satisfactory, for all test cases, in accelerating the convergence of Kahng’s method when the starting point was not sufficiently close to the best approximation. However, in the least \( p^\text{th} \)-power error design of one-dimensional digital FIR filters Kahng’s method was found to perform satisfactorily, in particular, in the case of \( p \) small, \( 2 < p \leq 15 \). For larger \( p \)’s, it was observed that the number of iterations needed for convergence, for some particular transition width specifications, increased abruptly which rendered the acceleration technique useless for these particular cases. This phenomenon, hereon denoted as the unpredictable behavior of the accelerated Kahng’s method, was found to depend heavily on the filter’s transition width, as shown in figure (3.2).

Various alternatives were studied in this work in order to overcome the difficulties posed by Kahng’s (or Fletcher’s \textit{et al}) original acceleration scheme, which are discussed below.

4.1 A Prediction Technique as an Acceleration Scheme for Kahng’s Method

The use of a prediction technique in order to accelerate the convergence of Kahng’s method was first proposed by Fletcher \textit{et al} in [11] and later developed by the same authors in [12]. In this latter work Fletcher \textit{et al} established that a minimax approximation can generally be found as the limit of a sequence of best \( L_p \) approximations.
Based on this idea an extrapolation technique is developed in [12] which enables minimax approximations to be predicted from best $L_p$ approximations and it is also noted that this extrapolation scheme could be used to predict intermediate values of the corresponding solution vector when an iterative process, such as the IRLS method, is used to calculate successive vectors of the best $L_p$ approximation.

Before the main results stated in [12] are presented below without proof, let us recall from chapter two that

$$\epsilon = A(f, a) - A_d(f) = Ca - A_d(f),$$

is the error function. Let now the vector of parameters of the best $L_p$ approximation be denoted by $a_p$, where the subscript $p$ in $a$ emphasizes the dependence of the solution on the type of norm used in the minimization problem.

It is then shown in [12] that an extrapolation technique can be derived in such a way that an estimate of the vector of parameters of the best $L_\infty$ approximation, denoted $\hat{a}_\infty$, is related to a sequence of vectors of parameters for best $L_p$ approximations in the following way

$$\hat{a}_\infty = (\eta^2 a_p + 2\eta a_{\eta p} + a_p)/(\eta - 1)^2$$

where $\eta \in \mathbb{R}^+$ is used to control both the number of values and the largest value of $p$ used in the extrapolation.

The above result can be further generalized [12] by setting up tables with entries

$$b^{(i,0)} = a_{\eta^i p},$$

$$b^{(i,j)} = (\eta b^{(i,j-1)} - b^{(i-1,j-1)}/(\eta - 1), \quad j \leq i, \quad j = 1, 2$$

$$b^{(i,j)} = (\eta^2 b^{(i,j-1)} - b^{(i-1,j-1)}/(\eta^2 - 1), \quad j \leq i, \quad j = 3, 4, 5,$$

such that $b^{(i,2)}$, $i = 2, 3, \cdots$ in (4.5) forms a first order succession of $a_\infty$ estimates and $b^{(i,5)}$, $i = 5, 6, \cdots$ in (4.6) forms a second order succession of estimates of $a_\infty$.

This extrapolation technique, as mentioned before, can then be useful in predicting a value for $a_{\eta^i p}$, namely

$$[(\eta^2 + 2\eta) a_{\eta^2 p} - (2\eta + 1) a_{\eta p} + a_p]/\eta^2$$

whose derivation is illustrated below to clarify the procedure employed.

Using (4.4), (4.5) and (4.6) it can be established that, if $i = 3$

$$b^{(3,0)} = a_{\eta^3 p}$$

$$b^{(3,2)} = (\eta b^{(3,1)} - b^{(2,1)}/(\eta - 1)$$

$$b^{(3,1)} = (\eta b^{(3,0)} - b^{(2,0)}/(\eta - 1)$$
where
\[ b^{(3,0)} = a_{\eta^3p} \]
\[ b^{(2,0)} = a_{\eta^2p} \]
and
\[ b^{(2,1)} = (\eta b^{(2,0)} - b^{(1,0)})/(\eta - 1) \]
where
\[ b^{(1,0)} = a_{\eta p} \]
then
\[ b^{(3,1)} = (\eta a_{\eta^3p} - a_{\eta^2p})/(\eta - 1) \]
\[ b^{(2,1)} = (\eta a_{\eta^2p} - a_{\eta p})/(\eta - 1) \]
whence
\[ b^{(3,2)} = \left[ \frac{\eta (\eta a_{\eta^3p} - a_{\eta^2p}) - (\eta a_{\eta^2p} - a_{\eta p})}{\eta - 1} \right]/(\eta - 1) \]
\[ = (\eta^2 a_{\eta^3p} - 2a_{\eta^2p} + a_{\eta p})/(\eta - 1)^2 \]
but
\[ b^{(3,2)} = \hat{a}_\infty \]
therefore using (4.3) we finally get
\[ \frac{\eta^2 a_{\eta^3p} - 2a_{\eta^2p} + a_{\eta p}}{(\eta - 1)^2} = \frac{\eta^2 a_{\eta^2p} - 2\eta a_{\eta p} + a_p}{(\eta - 1)^2} . \]

Using this extrapolation technique in a similar way, various other expressions can be derived which allow the prediction of several \( a_{\eta'p} \), some of which are contained in table 4.1.

It should be noted that before the prediction technique can be applied, the value of \( \eta \) and the initial value of \( p \) must be chosen. Regarding the effects of roundoff error and truncation errors on the application of (4.3) Fletcher et al in [12] suggest that \( \eta = 2 \); and since the least squares solution, corresponding to \( p = 2 \), is directly available from the normal equations \( p^0 \), the initial \( p \), should be 2 as well.
Table 4.1 Prediction Formulas using Extrapolation Technique

<table>
<thead>
<tr>
<th>$a_{np}$</th>
<th>General Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_{np}$</td>
<td>$a_{np}(\eta^2+2\eta) - a_{np}(2\eta+1) + a_p / \eta^2$</td>
</tr>
<tr>
<td>$a_{np}$</td>
<td>$a_{np}(\eta^3+2\eta^2+3\eta) - a_{np}(2\eta^2+4\eta+2) + a_p(2+\eta) / \eta^3$</td>
</tr>
<tr>
<td>$a_{np}$</td>
<td>$a_{np}(\eta^4+2\eta^3+3\eta^2+4\eta) - a_{np}(2\eta^3+4\eta^2+6\eta+3) + a_p(\eta^2+2\eta+3) / \eta^4$</td>
</tr>
<tr>
<td>$a_{np}$</td>
<td>$a_{np}(\eta^5+2\eta^4+3\eta^3+4\eta^2+5\eta) - a_{np}(2\eta^4+4\eta^3+6\eta^2+8\eta+4) + a_p(\eta^3+2\eta^2+3\eta+4) / \eta^5$</td>
</tr>
</tbody>
</table>

A direct application of the prediction technique to Kahng’s method for the $p^{th}$-power error design of one-dimensional FIR filters is not possible, unless the values of $a_{(4)}$ and $a_{(8)}$ be known. Therefore, it is necessary to compute these estimates using the original acceleration technique proposed by Kahng and Fletcher, before the prediction technique can be used to its full extent.

Although we found the prediction technique effective in terms of accelerating Kahng’s method when the starting point is not close to the best $L_p$ approximation, it did not help to solve the unpredictable behavior observed when the original acceleration scheme proposed by Kahng and Fletcher in [11, 15] was used. When the $a_p$ estimate, for $p \geq 64$, was computed using the prediction technique in a recursive way, i.e. by calculating $a_{(32)}$ from $a_{(16)}, a_{(8)}$ and $a_{(4)}$, then $a_{(64)}$ was found from $a_{(32)}$, $a_{(16)}$ and $a_{(8)}$ and so on until the actual $a_p$ was reached, instead of directly using the expressions listed in table 4.1, the overall numerical performance of this prediction technique was improved. See Fig. 4.1.

In order to increase the rate of convergence and to overcome some of the numerical problems posed by Kahng’s method a different approach to that of the acceleration technique proposed in [15, 11] was explored which is presented in the next section.

4.2 A Perturbed Problem

In [7] Hakan Ekblom claims that the results presented by Kahng and Fletcher et al in [11, 15], which are derived over a continuum, do not always carry over to approximations on a discrete point set, due to the appearance of zero residuals when Newton’s method is applied to the $L_p$ minimization problem.

In [7] a technique is proposed which essentially consists in perturbing, “pushing away”, the given points from the approximating function to avoid the appearance of zero residuals in the early stages of the algorithm.
Figure 4.1 Comparison of the Prediction Scheme vs. the Acceleration Technique for length-21 filter and $p = 86$
Let us recall that our problem is to minimize
\[ \|\varepsilon\|^p_p = \sum_{k=0}^{LF-1} |A(f_k, a) - A_d(f_k)|^p = \sum_{k=0}^{LF-1} (\varepsilon_k^2)^{p/2}. \] (4.8)

Then a perturbation of the original problem can be accomplished by rewriting (4.8) as
\[ \|\varepsilon^{(\delta)}\|^p_p = \sum_{k=0}^{LF-1} (\varepsilon_k^2 + \delta)^{p/2}, \] (4.9)
where \( \delta \in \mathbb{R} \). The vector of parameters of the best \( L_p \) approximation of the new problem is denoted \( a^{(\delta)} \) and has the following property when regarded as a function of \( \delta \) [7]
\[ \lim_{\delta \to 0} a^{(\delta)} = a^{(0)}, \] (4.10)
and since \( a^{(0)} \) is unique, then \( a = a^{(0)} \).

Although equation (4.9) could be written as a weighted least squares problem, the minimization problem could not be solved using the IRLS technique due to the appearance of a quadratic term in \( a^{(\delta)} \).

Using a similar approach to the one proposed by Ekblom in [7] the problem can be solved in a slightly different way by minimizing, instead
\[ \|\varepsilon^{(\delta)}\|^p_p = \sum_{k=0}^{LF-1} |\varepsilon_k + \delta|^p, \] (4.11)
where (4.11) can be expressed as a weighted least squared problem
\[ \|\varepsilon^{(\delta)}\|^p_p = \sum_{k=0}^{LF-1} (w_k^{(\delta)})^2 (\varepsilon_k + \delta)^2 = [Ca - A_d]^{\top} WT W[Ca - A_d] \] (4.12)
where \( w_k^{(\delta)} = |(\varepsilon_k + \delta)^{p-2}|^{1/2} \) and \( W = \text{diag}(w_1^{(\delta)}, \ldots, w_{LF-1}^{(\delta)}) \).

Equation (4.12) can then be minimized with respect to \( a^{(\delta)} \) by means of the normal equations
\[ C^{\top} WT W C a^{(\delta)} = C^{\top} WT W (A_d - \delta). \] (4.13)
The actual solution \( a \) to our original problem can be found by making \( \delta \) fairly big in the first iteration and then successively reducing the value of \( \delta \) at each iteration until \( \delta = 0 \).

Choosing appropriately the initial value of \( \delta \) is troublesome and there is no guarantee that none of the residuals will eventually become zero, contrary to the case when
(4.9) is minimized. However, it was observed for all test cases that the unpredictable behavior in the accelerated Kahng's method was not caused by the appearance of zero residuals as stated by Ekblom, but rather to a marked difference in the size of the different components of the error vector, which caused in turn that the \( p^{th} \)-power error norm sharply increased from one iteration to the next instead of smoothly decreasing. This also brought about a significant increase in the number of iterations needed for the algorithm to converge.

By closely examining the values of the error vector components it was possible to first observe and later show that the unpredictable behavior was caused by a disparity between the number of ripples in the initial \( L_2 \) approximation of the FIR filter and the actual number of ripples for the best \( L_p \) approximation. This phenomenon is intimately related to the filter's transition width specifications as seen in figures (4.2), (4.3) and (4.4).

It is shown in the next section that the incidence of the transition width specifications in the design of one-dimensional digital FIR filters can be controlled by a simple modification of the original acceleration scheme presented in [11, 15].

### 4.3 The Accelerated Kahng’s Method Modified

It was observed that the unequal ripples phenomenon discussed in the preceding section occurred, in all test cases, during the first three or four iterations of the algorithm. This fact could be associated with the possibility that what causes the accelerated Kahng's method to diverge in some particular cases may be caused by the way the acceleration scheme, as proposed by Kahng in [15] and by Fletcher et al in [11], approaches the best \( L_p \) approximation. Therefore, by modifying the rate of increase of \( p \) in the original acceleration technique, the following scheme was obtained

\[
p_0 = 2, \quad p_m = \min(p, \mu p_{m-1}),
\]

where \( \mu \in \mathbb{R}^+ \) is the convergence control parameter. It is by means of this parameter that the effect of the filter's transition width in the rate of convergence of Kahng's algorithm is controlled. See figures (4.5) and (4.6).

Although, there is not only one \( \mu \) which best suits all filter specifications, it was found that when

\[
p_0 = 2, \quad p_m = \min(p, 1.6 p_{m-1}),
\]

(4.15)
Figure 4.2 Iterations vs Transition Width for the Accelerated Kahng's Method

Figure 4.3 Iterations vs Transition Width for the Extended Lawson algorithm
Figure 4.4 Comparison of the Performance, i.e. Iterations vs Transition Width, for the Perturbed Problem
was used, the rate of convergence for the accelerated Kahng’s method was the fastest for most practical applications. For values of $\mu > 2$ the performance of the algorithm was always found to be worse than for $\mu \leq 2$.

An equivalent modification was done to the prediction technique by adjusting the value of the parameter $\eta$, and it was found that the best results were obtained when $\eta \approx 1.6$, which is consistent with the results found for the accelerated Kahng’s method modified. However, it was not possible, using this modification, to improve the numerical stability of this acceleration scheme, making the accelerated Kahng’s method the best option to attain rapid convergence for Kahng’s method.
**Figure 4.5** Effect of $\mu$ on the number of iterations for different transition width (TW) specifications

**Figure 4.6** Effect of $\mu$ on the number of iterations for different transition width (TW) specifications
Chapter 5

$L_p$-Approximation of One-Dimensional Digital FIR Filters using Different $p$'s in Different Bands

Although the least-squares and the Chebyshev approximations are the most commonly used in the design of digital FIR filters they each have their disadvantages. As an alternative to this problem it has been proposed in this work that the $L_p$ approximation, as introduced and discussed in previous chapters, be used. Let us now extend our results to the case where different error measures are defined over different bands, say an $L_\infty$ approximation over the passband and an $L_2$ approximation over the stopband, or vice-versa. This mixed $L_p$ approximation problem will be denoted in this work as an $L_{pp,ps}$ approximation, where $pp$ refers to the $p$ used in the passband and $ps$ corresponds to the $p$ used in the stopband.

The frequency domain approximation analysis will be developed in the next section for an ideal low-pass filter.

5.1 Different Norms in Different Bands with Zero Error Weighting in the Transition Band.

In the most general case an ideal low-pass filter is given by a frequency response which is unity for frequencies between zero (0Hz.) and $f_o$ and is zero for frequencies between $f_o$ and 0.5Hz. For the $L_p$ (or for the $L_{pp,ps}$) approximation analysis in this work a transition region [31] is used between the passband and the stopband, hence the amplitude frequency response of our ideal low-pass filter is as shown in Fig. 5.1.

The $L_p$ approximation problem for the one dimensional FIR filter, as stated in chapter two, consists of minimizing

$$\|\epsilon\|_p^p = \int_0^{0.5} |A(f,a) - A_d(f)|^p \, df$$

(5.1)
Figure 5.1  Ideal Low-pass Filter with no Error Contribution from the Transition Band, where $f_s - f_p = TW$
If the error measure does not include any contribution from the transition band, a
different formulation for (5.1) can be used, namely

$$\|\varepsilon\|^p_p = \int_\Gamma |A(f, a) - A_d(f)|^p \, df$$  \hspace{1cm} (5.2)

where $\Gamma = (0 \leq f \leq f_p) \cup (f_s \leq f < 0.5)$.

Defining

$$w_p(f, a) = |A(f, a) - A_d(f)|^{p-2}$$  \hspace{1cm} (5.3)

equation (5.2) can be written as a weighted least squares

$$\|\varepsilon\|^p_p = \int_\Gamma w_p(f, a)^2 (A(f, a) - A_d(f))^2 \, df.$$  \hspace{1cm} (5.4)

Now, using a formulation better suited for numerical work [17] and expanding equation (5.4)

$$\|\varepsilon\|^p_p = \sum_{0}^{NP} w_{p}(f, a)^2 (A^p(f, a) - A^p_d(f))^{pp} + \sum_{NS}^{LF-1} w_{s}(f, a)^2 (A^s(f, a) - A^s_d(f))^{ps}$$  \hspace{1cm} (5.5)

where the superscripts and subscripts $p$ and $s$ denote the corresponding dependence of
the respective vectors on the frequency band of interest, and the limits $NP$ and $NS$
in the summations specify the integer number of samples pertaining to the passband
or to the stopband respectively.

Minimizing (5.5) when $pp \neq ps$ using IRLS requires a method that allows the
possibility of specifying two separate vectors, one for each band, at each step of the
algorithm. In this sense the extended Lawson algorithm seems most suitable for this
case since the algorithm does not involve an up-date, which depends on $p$, of the
vector of parameters $a$ at each iteration. However, it is also possible to use Kahng’s
method if the up-date of the vector of parameters $a$ is always done in the frequency
domain as indicated by equation (3.26), rather than in the time domain, allowing
step 6 of Kahng’s algorithm to be written as follows

$$A^{m+1} = \left[ A^m_{pp} + \frac{1}{pp - 1} (Ca^k - A^m_{pp}), A^m_{pp}, + \frac{1}{ps - 1} (Ca^k - A^m_{ps}) \right]$$  \hspace{1cm} (5.6)

In order to assess the convergence properties of the IRLS algorithm in the $L_{pp,ps}$
filter design problem when $pp \neq ps$, Kahng’s method and the extended Lawson algo-

rithm are compared and analyzed. The number of iterations required for convergence
of each method is found using (3.29) by varying the value of $ps$ and keeping $pp = 2.$
Then, the amplitude response of the best $L_{pp,ps}$ approximation is compared to the amplitude response obtained from the starting point which is given by an $L_2$ approximation.

It was observed that both algorithms converged fairly reliably for most test cases. However, the rate of convergence obtained for both methods was much slower than that achieved for the general $L_p$ approximation with only one $p$ over the entire frequency band. It was also observed that the accelerated Kahng’s method showed faster convergence than the extended Lawson algorithm, for all test cases. See Fig. 5.2.

The rate of convergence of Kahng’s algorithm with the acceleration technique can be further improved if instead of having two different vectors in step 6 of the accelerated Kahng’s method, as suggested in (5.6), only one vector is used for updating the frequency response

$$A^{m+1} = A^m + \frac{1}{p_a - 1}(Ca^k - A^m)$$

(5.7)

and $p_a$ is given by

$$p_a = \rho \times \max(pp, ps)$$

(5.8)

where $\rho \in \mathbb{R}$ and is used to control the influence of the higher $p$ in the update process, which in turn makes $\rho$ a convergence control parameter.

Furthermore, the relation between the number of ripples of the initial $L_2$ approximation and the number of ripples of the best $L_{pp,ps}$ approximation was investigated. It was found that for $p > 9$ the number of ripples in the amplitude response of the $L_2$ approximation and in the amplitude response of the best $L_{pp,ps}$ approximation always differed. On the other hand, it was also observed that for the same values of $p$ the number of ripples of the best $L_{pp,ps}$ approximation and of the Chebyshev approximation coincided.

The discrepancy in the number of ripples between the initial approximation and the best $L_{pp,ps}$ approximation may explain, as it has already been seen in chapter 4, the slow convergence of the mixed $L_p$ approximation problem, since it is precisely the difference in the number of ripples that causes the acceleration scheme, as expressed in equation (3.28), to sharply increase the number of iterations required for convergence.

Because the slow rate of convergence of Kahng’s algorithm in the $L_{pp,ps}$ approximation problem appears to be a consequence of the number of ripples in the initial approximation, it seems appropriate to look for a method of design which does not
Figure 5.2 The $L_{pp,ps}$ approximation for $pp = 2$ and $ps = 3 - 13$
necessarily yield the same accuracy, but which is computationally more efficient. The method to be described next is based on finding a new starting point, in the form of $L_{2,ps}$.

This technique consists of

- First an $L_2$ solution is computed and its residual vector is saved.

- Next the corresponding $L_p$ solution is computed using Kahng's method until convergence is obtained, and then the residual vector is also saved.

- Now the respective portions of the passband and the stopband residual vectors are combined in one vector which is then used as a starting point.

- Finally 10 more iterations of Kahng's algorithm are run before the iterative procedure is stopped. It is assumed then that the answer obtained at that point is the best $L_{2,ps}$ approximation.

- the matlab program that implements this method is *ppkdiffp7.m* and is presented in the appendix.

Though this technique does not guarantee a best $L_{2,ps}$ approximation, since the algorithm is stopped before convergence is achieved, it was found, for most test cases, that the vector of parameters a obtained in this way agreed in 3 or 4 digits to the actual best $L_{2,ps}$ solution. The primary advantage of this method is that it allows one to reduce by half the number of iterations required to find a fairly acceptable solution.

### 5.2 The $L_p$ Approximation Problem with Different $p$'s in the Stopband

The technique applied to the $L_{pp,ps}$ approximation problem can be extended in order to attack the mixed $L_p$ norm problem as posed by Adams in [1]. As in equation (5.2) the problem of minimizing $\|\varepsilon\|^p_p$ when different $p$'s are also considered in the stopband, denoted here as $L_{pp,p_s,s_1}$, can be equivalently formulated as minimizing

$$\|\varepsilon\|^p_p = \int_{\hat{\Gamma}} |A(f,a) - A_d(f)|^p df$$  \hspace{1cm} (5.9)$$

where $\hat{\Gamma} = (0 \leq f \leq f_p) \cup (f_s \leq f < f_{s1}) \cup (f_{s1} \leq f < 0.5)$, and $f_{s1}$ defines a second stopband frequency edge, with $f_{s1} > f_s$. 
Using equation (5.4), expanding it according to $\tilde{\Gamma}$ and replacing the integration by summations, it is found that (5.9) becomes

$$\|s\|_P^p = \sum_{p=0}^{NP} w_{fp}(f, a)^2 (A^p(f, a) - A_d^p(f))^p + \sum_{NS} w_{f_1}(f, a)^2 (A^s(f, a) - A_d^s(f))^s + \sum_{NS_1}^{NS_1} w_{f_{s_1}}(f, a)^2 (A^{s_1}(f, a) - A_d^{s_1}(f))^s$$

(5.10)

where the superscripts and subscripts $p$, $s$ and $s_1$ are used to emphasize dependence on the respective frequency band.

The approach just described is not a peak-gain constraint problem as Adams [1], since the method requires that the second stopband frequency edge $f_{s_1}$ be a known parameter and not a variable found by the optimization procedure as in Adams' problem. This fact however causes our method to be extremely flexible since it allows for narrower and wider specifications of the two stopbands to be minimized under different norms. Furthermore, this more general formulation allows the frequency band to be further segmented into many small stopbands as well as passbands which in turn can be minimized using different error measures in all of these frequency band portions.

In order to assess a smooth transition at $f_{s_1}$ it is necessary to weight more the portion of the stopband which is minimized with the smaller error measure which, in this work, usually corresponds to the second stopband since normally $ps \geq 40$ and $p_{s_1} = 2$. For this particular case equation (5.10) can then be written as

$$\|e\|_P^p = \sum_{p=0}^{NP} w_{fp}(f, a)^2 (A^p(f, a) - A_d^p(f))^p + \sum_{NS} w_{f_1}(f, a)^2 (A^s(f, a) - A_d^s(f))^s + \sum_{NS_1}^{NS_1} w_{f_{s_1}}(f, a)^2 (A^{s_1}(f, a) - A_d^{s_1}(f))^2$$

(5.11)

where $K_{s_1} > 1$ allows more weight in the second stopband.

A serious drawback in the above technique is the slow rate of convergence which can not be significantly improved, as in the $L_{pp,ps}$ case, using the different starting point approach. The main disadvantage of this technique for this particular case lies on the fact that normally three, instead of two, different residual vectors need to be computed before the starting point can be found, causing the number of iterations required by the iterative process to increase significantly.
5.3 Numerical Results and Design Examples.

5.3.1 The general $p^{th}$-power error digital FIR filter design

For the case of the $p^{th}$-power error filter design the effect of the transition width, denoted $TW$, on the stopband deviation was computed for four different filter lengths. The value of $p$ was kept fixed at a value of 20. The value of $f_p$ was chosen to be 0.2 Hz. and $f_s$ was varied from 0.201 Hz. – 0.45 Hz.. See Fig. 5.3.

Also the shifting property shown by the unpredictable behavior for different starting $f_s$’s is presented in figure 5.4.

The effect of $p$ on the number of ripples and on the number of iterations is shown in figure 5.5.

Figures 5.6 through 5.8 show the amplitude responses of three length-21 FIR low-pass filters with the following specifications:

- **Design Example 1.**
  - Passband edge: 0.2 Hz.
  - Stopband edge: 0.23 Hz.
  - Value of $p$: 4

- **Design Example 2.**
  - Passband edge: 0.2 Hz.
  - Stopband edge: 0.23 Hz.
  - Value of $p$: 10

- **Design Example 3.**
  - Passband edge: 0.2 Hz.
  - Stopband edge: 0.23 Hz.
  - Value of $p$: 40

5.3.2 $L_{pp,ps}$ Filter Design

For this case figures 5.9 through 5.12 present a sequence of frequency amplitude response approximations versus the corresponding $L_2$ amplitude response for different $p$’s ($p > 2$) in the stopband while $p$ in the passband was kept fixed to the value of 2. The sequence of approximations show how the number of ripples changes with $p$.

Figures 5.13 and 5.18 show the amplitude responses for several mixed $L_p$ FIR low-pass filters with the following specifications:
Figure 5.3  Effect of TW on the stopband deviation
Figure 5.4 Shifting Property of the Unpredictable Behavior for Different Starting $f_p$'s
Figure 5.5  Effect of $p$ on the number of ripples
**Figure 5.6** Comparison of the amplitude response for an $L_p$-approx. ($p=4$) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations.
Figure 5.7 Comparison of the amplitude response for an $L_p$-approx. ($p=10$) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations.
Figure 5.8 Comparison of the amplitude response for an $L_p$-approx. (p=40) with the corresponding amplitude responses of an $L_2$ and of an $L_\infty$ approximations.
Different p's Comparison, pp=2, ps=6, Length=21, g1=.9, fs=.2, fp=.25 Hz.

Figure 5.9  $L_2$ and Mixed $p$ (pp=2, ps=6) Frequency Response
Figure 5.10 \( L_2 \) and Mixed \( p \) (pp=2, ps=7) Frequency Response
Figure 5.11  $L_2$ and Mixed $p$ (pp=2, ps=10) Frequency Response
Figure 5.12  $L_2$ and Mixed $p$ ($pp=2$, $ps=13$) Frequency Response
• **Design Example 1.**
  Passband edge: 0.2 Hz.
  Stopband edge: 0.25 Hz.
  Value of $p$ in the passband ($pp$): 20
  Value of $p$ in the stopband ($ps$): 2
  Filter length ($L$): 21

• **Design Example 2.**
  Passband edge: 0.2 Hz.
  Stopband edge: 0.25 Hz.
  Value of $p$ in the passband ($pp$): 2
  Value of $p$ in the stopband ($ps$): 20
  Filter length ($L$): 31

• **Design Example 3.**
  Passband edge: 0.2 Hz.
  Stopband edge: 0.25 Hz.
  Value of $p$ in the passband ($pp$): 40
  Value of $p$ in the stopband ($ps$): 2
  Filter length ($L$): 21

• **Design Example 4.**
  Passband edge: 0.2 Hz.
  Stopband edge: 0.25 Hz.
  Value of $p$ in the passband ($pp$): 2
  Value of $p$ in the stopband ($ps$): 40
  Filter length ($L$): 21

• **Design Example 5.**
  Passband edge: 0.22 Hz.
  Stopband edge: 0.25 Hz.
  Value of $p$ in the passband ($pp$): 2
  Value of $p$ in the stopband ($ps$): 20
  Filter length ($L$): 31

• **Design Example 6.**
  Passband edge: 0.32 Hz.
Stopband edge: 0.35 Hz.
Value of p in the passband (pp): 40
Value of p in the stopband (ps): 2
Filter length (L): 21

5.3.3 $L_{pp,ps}$ FIR Filter Design

The influence of the weighting parameter $K_{s1}$ on the smoothness of the transition at $f_{s1}$ is shown by comparison in figures 5.19 and 5.20.

For completeness the amplitude response approximations of the narrow and wideband filters introduced in [1] are presented in figures 5.19 and 5.21 whose specifications are:

- **Design Example 1.**
  - Passband edge ($f_p$): 0.021 Hz.
  - First stopband edge ($f_2$): 0.03125 Hz.
  - Second stopband edge ($f_{s1}$): 0.005 Hz.
  - Filter length (L): 129
  - Value of p in the passband (pp): 30
  - Value of p in the first stopband (ps): 70
  - Value of p in the second stopband (ps1): 2

- **Design Example 2.**
  - Passband edge ($f_p$): 0.06 Hz.
  - First stopband edge ($f_2$): 0.08 Hz.
  - Second stopband edge ($f_{s1}$): 0.11 Hz.
  - Filter length (L): 95
  - Value of p in the passband (pp): 30
  - Value of p in the first stopband (ps): 60
  - Value of p in the second stopband (ps1): 2
Figure 5.13 $L_{pp,ps}$-Approximation for a length-31 FIR low-pass filter with $pp = 20$ and $ps = 2$
Mixed p Frequency Response, length-31, pp = 2 & ps = 20

\[
\begin{align*}
fp &= 0.2 \text{ Hz.} \\
fs &= 0.25 \text{ Hz.}
\end{align*}
\]

**Figure 5.14** \( L_{pp,ps} \)-Approximation for a length-31 FIR low-pass filter with \( pp = 2 \) and \( ps = 20 \)
Figure 5.15 \( L_{pp,ps} \)-Approximation for a length-21 FIR low-pass filter with \( pp = 40 \) and \( ps = 2 \)

Mixed p Frequency Response, length-21, \( pp = 40 \) & \( ps = 2 \)

\( fp = 0.2 \text{ Hz} \quad fs = 0.25 \text{ Hz} \)
Figure 5.16 $L_{pp,ps}$-Approximation for a length-21 FIR low-pass filter with $pp = 2$ and $ps = 40$
Mixed p Frequency Response, length-31, pp = 2 & ps = 20

\[ fp = 0.22 \text{ Hz.} \quad fs = 0.25 \text{ Hz.} \]

**Figure 5.17** \( L_{pp,ps} \)-Approximation for a length-31 FIR low-pass filter with \( pp = 2 \) and \( ps = 20 \)
Mixed p Frequency Response, length-21, pp = 40 & ps = 2

$fp = 0.32 \text{ Hz. } fs = 0.35 \text{ Hz.}$

**Figure 5.18** $L_{pp,ps}$-Approximation for a length-21 FIR low-pass filter with $pp = 40$ and $ps = 2$
Figure 5.19  Narrow-band filter amplitude response ($Ks1 = 15$)

Figure 5.20  Narrow-band filter amplitude response ($Ks1 = 22$)
Figure 5.21  Wide-band filter amplitude response ($K_{s1} = 12$)
Chapter 6

CONCLUSIONS

In this thesis an efficient and simple to program filter design method using a general \( L_p \)-approximation and based on the weighted least squares technique is proposed. This design method allows one to find an approximation that is a trade-off between the two most commonly used error measures, namely the \( L_2 \) norm and the \( L_\infty \) norm.

The extended Lawson algorithm [34] and Kahng's [15] (or Fletcher's et al. [11]) method were applied in this research, to our knowledge, for the first time to the \( p^{th} \)-power error design of one-dimensional digital FIR filters.

The convergence properties of the extended Lawson algorithm and of Kahng's method with the acceleration scheme proposed in [15] were investigated and compared. It was found that Kahng's method showed faster convergence rates than the extended Lawson algorithm for most practical applications. However, for long filters ( \( L > 21 \) ) and for \( p > 100 \), the extended Lawson algorithm showed better numerical performance.

Kahng's method with the modified acceleration scheme proposed in chapter 4 (see equation (4.14)) of this work and reproduced here for convenience

\[
p_0 = 2, \quad p_m = \min(p, \mu p_{m-1}),
\]

where \( \mu \in \mathbb{R}^+ \), showed even faster convergence. Besides, it was found that the rate of increase in the value of \( p \) as given by the above equation was critical to the rate of convergence for some particular transition width specifications due to the discrepancy in the number of ripples between the \( L_2 \) starting point and the best \( L_p \)-approximation.

A new technique was developed here to design FIR filters with different error measures in different bands of the frequency spectrum. This design method allowed for an \( L_2 \) norm in the passband and an \( L_p \) norm in the stopband or
vice-versa. An extension of this technique which allowed different $p$'s in the stopband proved to be an effective way and a more flexible approach to attack the mixed norm problem posed by Adams in [1]. However, for both applications this technique showed slow convergence, thus further work must be done towards producing a faster design technique.

Future research should investigate the extension of the results presented in this work for the one-dimensional case to two-dimensions. Furthermore, special consideration should be given to the possibility of finding an $L_\infty$ or minimax approximation as the limit of a sequence of best $L_p$ approximations, in particular for the two-dimensional case or for complex Chebyshev approximations, since Kahng's method, unlike the Remes algorithm [35, 31], is not restricted to approximations where the set $\{\phi_i(f)\}$ as given in equation (1.3) be a Chebyshev set.
Appendix A

MATLAB PROGRAMS

A.1 Kahng’s Algorithm with the Modified Acceleration Technique

This program designs a best $L_p$ approximation FIR filter using Kahng’s method with the modified acceleration technique proposed in section 4.3.

```matlab
function ppkang6(L,LF,fs fp,p,g)
    if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end;
    M = (L-1)/2; E2 = []; EC = []; EP = []; ap = []; pk = 2;
    Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
    dp = fp/Np; ds = (.5 - fs)/Ns;
    Ad = [ones(Np,1); zeros(Ns,1)]; % Desired Frequency Response
    f=[((0:Np-1)*dp + dp/2),((0:Ns-1)*ds + fs + ds/2)]'; % F. Samples
    C = cos(2*pi*(f*[0:R-1])); WC = C; % Cosine Matrix
    a = C\Ad; ap = [ap a]; gn = C*a;
    for k = 1:500
        pk = min(p,g*pk)
        e = gn - Ad;
        w = (abs(e)).^((pk-2)/(2)); w = w/sum(w); % Weighting Function
        for m = 1:R, WC(:,m) = w*C(:,m); end;
        a = WC\(w.*Ad); ap = [ap a]; % Solve System of Equations.
        gn = ((pk-2)*gn + C*a)/(pk-1); % Freq. Response Up-Date
        E2 = [E2 sqrt((e'*e))]; % Squared Error
        EC = [EC max(abs(e))]; % Chebyshev Error
        EP = [EP ((sum(abs(e.*pk)))^((1/pk)); pth-Power Error
        plot(EP);
    end

    % STOPPING CRITERION
```
ac = ap(:,k+1) - ap(:,k);
chang = norm(e, inf)
if norm(ac, 1) < 0.0000005 * norm(ap(k+1), 1),
    break
end
end
disp('NO. of ITERATIONS ='), disp(k)
pplot(a, E2, EC, EP, M);
A.2 Karlovit’z Algorithm with no Line Search

This program implements Karlovitz’ approach to design a one-dimensional FIR filter without having to compute the value of \( \lambda(q) \) at each iteration.

```matlab
function ppkarl(L,LF,fs,fp,p,q)
    s = (p-2)/2; M=(L-1)/2;
    E2 = []; EC = []; EP = []; EPM = [];
    Np = round(LF*fp/(2*(fp+.5-fs))); Ns = LF/2-Np;
    d1 = fp/(2*Np); d = fp/(2*Np);
    Ad = [ones(Np,1);zeros(2*Ns,1);ones(Np-1,1)]; % Desired F. R.
    f=[(0:Np-1)*(fp-d)/(Np-1),((0:2*Ns-1)*(1-2*fs-2*d)/(2*Ns-1)+fs+d)]
    f = [f,((0:Np-2)*(fp-3*d1)/(Np-2)+(1-fp+d1))]; % F. Samples
    C = cos(2*pi*f*[0:M])); WC = C; % Cosine Matrix
    a = C\Ad; % Initial Approximation
    e = C*a-Ad; gn = C*a;
    Lam = ones(length(e),1)*(q);
    for k = 1:16
        e = gn-Ad;
        ep = (e.^(2*(1+s)));
        E2 = [E2 e'*e];
        EC = [EC max(abs(e))];
        EP = [EP sum(ep)];
        ew = (e.^(p-2)/2);
        for m = 1:M+1
            WC(:,m) = ew.*C(:,m);
        end;
        a = WC\(ew.*Ad);
        gn = gn + Lam.*(C*a - gn);
    end
    pplot(a,E2,EC,EP,M);
```
A.3 Modified IRLS Method (Burrus' Method)

This program computes the best $L_p$ approximation using a Modified Iteratively Reweighted Least Squares algorithm as described in section 3.1.2.

```matlab
function a = ppbur(L,LF,fs,fp,p,q)
if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end; erc = [];
M = (L-1)/2; E2 = []; EC = []; EP = []; ap = [];
Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
dp = fp/Np; ds = (.5 - fs)/Ns;
Ad = [ones(Np,1); zeros(Ns,1)]; % Desired Frequency Response
f = [(0:Np-1)*dp + dp/2,((0:Ns-1)*ds + fs + ds/2)]; % F. Samples
C = cos(2*pi*(f*[0:R-1])); WC = C; % Cosine Matrix
w = 10*ones(LF,1)/LF;
a = 100*(C\Ad); ap = [ap a];
for k = 1:200
    for m = 1:R, WC(:,m) = w*C(:,m); end;
a = WC\(w.*Ad); ap = [ap a];
e = C*a - Ad;
wt = abs(e.^((p-2)/2)); wt = wt/sum(wt);
w = q*wt + (1-q)*w; %Weighting Function Up-Date
E2 = [E2 sqrt((e'*e)/LF)]; % Squared Error.
EC = [EC max(abs(e))]; % Chebyshev Error.
EP = [EP ((sum(abs(e.^p))/LF)^(1/p)); % pth-Power Error.
    plot(EP);
    %
    % STOPPING CRITERION
    ac = ap(:,k+1) - ap(:,k);
    if norm(ac,1) < 0.00005*norm(ap(k+1),1),
        break
    end
end
disp('NUMBER OF ITERATIONS:') disp(k)
pplot(a,E2,EC,EP,M);
```
A.4 The Extended Lawson Algorithm

This program implements the Extended Lawson Algorithm to compute a best 
$L_p$ approximation to a desired frequency response $A_d$

```matlab
function pprulaw(L,LF,fs,fp,p,g)
    if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end;
    M = (L-1)/2; E2 = []; EC = []; EP = [];
    Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
    dp = fp/Np; ds = (.5 - fs)/Ns;
    Ad = [ones(Np,1); zeros(Ns,1)]; % Desired Frequency Response
    f=[((0:Np-1)*dp + dp/2),((0:Ns-1)*ds + fs + ds/2)]; % F. Samples
    C = cos(2*pi*(f*[0:R-1])); WC = C; % Cosine Matrix
    w = ones(LF,1);
    for k = 1:500
        for m = 1:R, WC(:,m) = w.*C(:,m); end
        a = WC\(w.*Ad); ap = [ap a]; % Solving System of Equations
        %
        % STOPPING CRITERION
        ac = ap(:,k+1) - ap(:,k);
        if norm(ac,1) < 0.0005*norm(ap(k+1),1),
            break
        end
        e = C*a - Ad;
        w = ((w.*(2*g)).*abs(e)).^((p-2)/(2*(g*(p-2)+1)));
        w = w/sum(w); % Weighting Function
        E2 = [E2 sqrt((e'*e)/LF)]; % Squared Error.
        EC = [EC max(abs(e))]; % Chebyshev Error.
        EP = [EP ((sum(abs(e.^p)))/LF)^(1/p)]; % pth-Power Error.
        plot(EP);
    end
    disp('NO OF ITERATIONS:'), disp(k)
pplot(a,E2,EC,EP,M);
```
A.5 Kahng's Method (Different p's in Different Bands)

This program uses Kahng's algorithm with the acceleration technique described at the end of section 5.1.

function adk=ppkdifpp7(L,LF,fs,fp,pp,ps,Ks,g1)
if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end;
M = (L-1)/2; E2 = []; EC = []; EP = [];
pp = 2; pss = 2;
ppk = 2; % initial p in the passband for acceleration scheme
psk = 2; % initial p in the stopband for acceleration scheme
pn = g1*max(pp,ps); % constant value of p used in the FR update
Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
dp = fp/Np; ds = (.5 - fs)/Ns;
Ad = [ones(Np,1); zeros(Ns,1)];
f = [((0:Np-1)*dp + dp/2),((0:Ns-1)*ds + fs + ds/2)]';
C = cos(2*pi*(f*[0:R-1])); WC = C;
a = C\Ad;
gn = C*a; gp = gn(1:Np); gs = gn(Np+1:LF); % Pass and Stopband FR
e1 = gn - Ad;
for k = 1:2
    ppp = min(pp,2*ppp); % Increasing p in the passband
    e = gn - Ad;
    w = (abs(e)).^-((ppp-2)/(2)); w = w/sum(w);
    for m = 1:R, WC(:,m) = w.*C(:,m); end;
    a = WC\(w.*Ad);
    CA = C*a;
    gn = ((ppp-2)*gn + CA)/(ppp-1); % Total FR update
    E2 = [E2 sqrt((e'*e)/LF)]; EC = [EC max(abs(e))];
    EP = [EP (((sum(abs(e.^ppp)))/(1/ppp))];
plot(EP);
end

e2 = CA - Ad; gn = C*a;
gn = C*a;
e = [e2(1:Np);e1(Np+1:LF)]; gp = gn(1:Np);
for k = 1:40
pss = min(ps, 2*pss)  \% Increasing p in the passband
e = gn - Ad;
w = (abs(e)).^((pss-2)/(2)); w = w/sum(w);
for m = 1:R, WC(:,m) = w.*C(:,m); end;
a = WC\(w.*Ad);
CA = C*a;
 gn = ((pss-2)*gn + CA)/(pss-1);  \% Total FR update
E2 = [E2 sqrt((e'*e)/LF)]; EC = [EC max(abs(e))];
EP = [EP (((sum(abs(e.'*pss)))/(1/pss))]
plot(EP);
end

e3 = CA - Ad; gn = C*a;
e = [e2(1:Np); e3(Np+1:LF)]; gs = gn(Np+1:LF);
for k = 1:200
  \% ppk = min(pp, 2*ppk)  \% Increasing p in the passband
  \% psk = min(ps, 2*psk)  \% Increasing p in the stopband
  ep = e(1:Np); es = e(Np+1:LF);
w = (abs(ep)).^((pp-2)/(2)); wp = wp/sum(wp);
ws = (abs(es)).^((ps-2)/(2)); ws = ws*(Ks/sum(ws));
w = [wp(ws];
for m = 1:R, WC(:,m) = w.*C(:,m); end;
a = WC\(w.*Ad);
CA = C*a;
 gn = ((pn-2)*gn + CA)/(pn-1);  \% Total FR update
 \% gp = ((pp-2)*gp + CA(1:Np))/(pp-1);
 \% gs = ((ps-2)*gs + CA(Np+1:LF))/(ps-1);
 \% gn = [gp;gs];
e = gn - Ad;
E2 = [E2 sqrt((e'*e)/LF)]; EC = [EC max(abs(e))];
EP = [EP (((sum(abs(ep.'*pp))/Np)^(1/pp)+
           Ks*((sum(abs(es.'*ps))/Ns)^(1/ps))]
plot(EP);
end
pplot(a,E2,EC,EP,M)
A.6  Kahng’s Method (Different p’s in the Stopband)

This program implements Kahng’s method using different p’s in different frequency bands. It also allows the use of two different p’s (ps and ps1) for the stopband in order to achieve similar filters to the ones presented by Adams in his paper. The frequency band edges for each p are denoted by fp (passband), fs (first stopband) and fs1 (second stopband). The parameter Ks1 allows to weigh differently the influence of the second stopband in order to get a smooth transition from the equi-ripple approximation to the least-square approximation. Typical values used for Ks1 are in the 10 - 20 range. Ks allows to weigh differently the first stopband from the passband. The parameter pn is used in the update of the frequency response and is given by the following expression

\[ pm = g1 \times \max(pp, ps) \]

where g1 controls the influence of the greater p. The value of g1 must be chosen between 0.6 and 1.

NOTE: FR stands for Frequency Response.

```matlab
function adk=ppkmultp(L,LF,fs,fs1,fp,pp,ps,ps1,Ks,Ks1,g1)
if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end;
M = (L-1)/2; E2 = []; EC = []; EP = [];
ppk = 2;    \% initial p in the passband for acceleration scheme
psk = 2;    \% initial p in the stopband for acceleration scheme
psk1 = 2;

 pn = g1*max(pp,ps); \% constant value of p used in the FR update
 Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
 Ns1 = round((fs1-fs)*Ns/fs); Ns2 = Ns-Ns1;
 dp = fp/Np; ds = (.5 - fs)/Ns; ds1 = (.5 - fs1)/Ns2;
 Ad = [ones(Np,1); zeros(Ns,1)];
 f1 = [((0:Np-1)*dp + dp/2)];
 f2 = [((0:Ns1-1)*ds + fs + ds/2)];
 f3 = [((0:Ns2-1)*ds1 + fs1 + ds1/2)];
 f = [f1 f2 f3];
 C = cos(2*pi*(f*[0:R-1])); WC = C;
 a = C\Ad;
```
gn = C*a; gp = gn(1:Np); gs = gn(Np+1:LF); % Pass and Stopband FR
for k = 1:400
    ppk = min(pp,1.61*ppk) % Increasing p in the passband
    psk = min(ps,1.61*psk) % Increasing p in the stopband
    psk1 = min(psk,1.61*psk1)
    e = gn - Ad;
    ep = e(1:Np); es = e(Np+1:LF-Ns2); es1 = e(LF-Ns2+1:LF);
    wp = (abs(ep))^((ppk-2)/(2)); wp = wp/sum(wp);
    ws = (abs(es))^((psk-2)/(2)); ws = ws*(Ks/sum(ws));
    ws1 = (abs(es1))^((psk1-2)/(2)); ws1 = ws1*(Ks1/sum(ws1));
    w = [wp;ws;ws1];
    for m = 1:R, WC(:,m) = w.*C(:,m); end;
    a = WC\(w.*Ad);
    CA = C*a;
    gn = ((pn-2)*gn + CA)/(pn-1); % Total FR update
E2 = [E2 sqrt((e'*e)/LF)];
EC = [EC max(abs(e))];
EP = [EP (((sum(abs(ep.'^ppk))/Np)^(1/ppk)+
    Ks*((sum(abs(es.'^psk))/Ns1)^(1/psk))+
    Ks1*((sum(abs(es.'^psk1))/Ns2)^(1/psk1)))
    plot(EP);
end
pplot(a,E2,EC,EP,M)
A.7  Kahng’s method using the Prediction Technique

This program implements Fletcher et al’s (or Kahng’s) Algorithm using a prediction (extrapolation) technique to accelerate the convergence, instead of the approach of updating p at each iteration until the actual p is reached. This program uses the acceleration technique of updating p in order to find the initial values (a2, a4 and a8) needed to start the prediction process. The parameter c can be used to adjust the speed of convergence by varying the multiplicative factors of the prediction formulas as well as the power of the consecutive \( a_p \) estimates. The initial p is always chosen to be 2 in order to use the least-square direct solution. The value of the parameter c should range between 1.6 and 2.2

```matlab
function afl=ppfletxp(L,LF,fs,fp,p,c)
if rem(L,2)==0, R = L/2; else, R = (L+1)/2; end;
M = (L-1)/2; E2 = [[]; EC = [[]; EP = [[]; ap = [[];
pk = 2;
Np = round(LF*fp/(fp+.5-fs)); Ns = LF - Np;
dp = fp/Np; ds = (.5 - fs)/Ns;
Ad = [ones(Np,1); zeros(Ns,1)]; % Desired Frequency Response
f=((0:Np-1)*dp + dp/2),((0:Ns-1)*ds + fs + ds/2)]';%F. Samples
C = cos(2*pi*(f*[0:R-1])); WC = C; % Cosine Matrix
a = C\Ad; gn = C*a;
for k = 1:3
    ap = [ap a];
    pk = min(p,c*pk) % Acceleration Technique
    e = gn - Ad;
    w = (abs(e)).^((pk-2)/(2)); w = w/sum(w);%
    for m = 1:R, WC(:,m) = w.*C(:,m); end;
    a = WC\(w.*Ad); % Solve System of Equations.
    gn = ((pk-2)*gn + C*a)/(pk-1); % Freq. Response Up-Date.
    E2 = [E2 sqrt((e'*e))]; % Squared Error.
    EC = [EC max(abs(e))]; % Chebyshev Error.
    plot(EP);
end
%```
% Prediction Formula for 64 in terms of a32,a16 and a8
%
% \( a = (2c*ap(:,5) - ap(:,4) + (c^2)*ap(:,3) - 2c*ap(:,2) + ap(:,1))/(c^2) \)
%
% Extrapolation for a128
%
% \( a = (57*ap(:,3) - 54*ap(:,2) + 13*ap(:,1))/16 \)
%
% Prediction Formula for a64
%
% \( a = (((c^4 + 2c^3 + 3c^2 + 4c)*ap(:,3)) - ((2c^3 + 4c^2 + 6c + 3)*ap(:,2)) + ((c^2 + 2c + 3)*ap(:,1))/(c^4) \)
%
% Prediction Formula for a16
%
% \( a = (((c^2 + 2c)*ap(:,3)) - ((2c + 1)*ap(:,2)) + ((1)*ap(:,1)))/(c^2) \)
%
k = p;
gn = ((pk-2)*gn + C*a)/(pk-1);  % Frequency Response Up-Date.
for k = 4:40
    e = gn - Ad;
    w = (abs(e)).^((pk-2)/(2)); w = w/sum(w);  % Weighting Function
    for m = 1:R, WC(:,m) = w.*C(:,m); end;
    a = WC\(w.*Ad);  % Solve System of Equations
    gn = ((pk-2)*gn + C*a)/(pk-1);
    E2 = [E2 sqrt((e'*e))];
    EC = [EC max(abs(e))];
    EP = [EP ((sum(abs(e.*pk))))^(-1/pk)];
    plot(EP);
end
pplot(a,E2,EC,EP,M);
Bibliography


