Two optimization techniques for designing multiplierless Fir filters

Torrance Matthew Lawton
University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation
https://digitalscholarship.unlv.edu/rtds/3372

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
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.

UMI
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
TWO OPTIMIZATION TECHNIQUES FOR DESIGNING
MULTIPLIERLESS FIR FILTERS

by

Torrance M. Lawton

Bachelor of Science
California Institute of Technology
1989

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Electrical Engineering

Department of Electrical and Computer Engineering
University of Nevada, Las Vegas
December 1997

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Thesis Approval
The Graduate College
University of Nevada, Las Vegas

November 19, 1997

The Thesis prepared by

TORRANCE M. LAWTON

Entitled

TWO OPTIMIZATION TECHNIQUES FOR
DESIGNING MULTIPLIERLESS FIR FILTERS

is approved in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

Peter Stubblefield
Examination Committee Chair

Dean of the Graduate College

W. L. Morgan
Examination Committee Member

J. Branc
Examination Committee Member

Graduate College Faculty Representative
ABSTRACT

Two Optimization Techniques for Designing
Multiplierless FIR Filters

by
Torrance M. Lawton

Dr. Peter Stubberud, Examination Committee Chair
Professor of Electrical Engineering
University of Nevada, Las Vegas

This thesis introduces two optimal design techniques for discrete time finite impulse response (FIR) multiplierless filters with constant group delay. The first technique minimizes a weighted least squared optimality criteria. The second technique minimizes a weighted least squared optimality criteria subject to user specified frequency constraints. Examples are included of transport processor applications and finite wordlength coefficient applications.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>iii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vi</td>
</tr>
<tr>
<td>Chapter 1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Chapter 2 Digital Signal Processing and Optimization</td>
<td>3</td>
</tr>
<tr>
<td>2.1 Filter Frequency Response</td>
<td>4</td>
</tr>
<tr>
<td>2.2 Digital Filter Design for Filters with Constant Group delay</td>
<td>6</td>
</tr>
<tr>
<td>2.3 Optimization</td>
<td>10</td>
</tr>
<tr>
<td>2.4 Lagrange Multipliers</td>
<td>11</td>
</tr>
<tr>
<td>2.5 Integer Programming</td>
<td>12</td>
</tr>
<tr>
<td>Chapter 3 Optimal Design Techniques</td>
<td>18</td>
</tr>
<tr>
<td>3.1 The Optimality Criteria</td>
<td>19</td>
</tr>
<tr>
<td>3.2 The Weighted Least Squared Optimal Criteria</td>
<td>20</td>
</tr>
<tr>
<td>3.3 The Amplitude Constrained Optimal Criteria</td>
<td>24</td>
</tr>
<tr>
<td>3.4 Solving for the Optimal Real Valued Solution vector</td>
<td>28</td>
</tr>
<tr>
<td>3.5 Reduction of the Feasible Solution Space</td>
<td>30</td>
</tr>
<tr>
<td>3.6 The Integer First Approximation</td>
<td>32</td>
</tr>
<tr>
<td>3.7 The Branch and Bound Method</td>
<td>36</td>
</tr>
<tr>
<td>3.8 Filter Order and Other Factors</td>
<td>45</td>
</tr>
<tr>
<td>Chapter 4 Applications</td>
<td>47</td>
</tr>
<tr>
<td>4.1 Multiplierless Filter Design Application</td>
<td>47</td>
</tr>
<tr>
<td>4.2 Finite Wordlength Application</td>
<td>67</td>
</tr>
<tr>
<td>Chapter 5 Conclusion</td>
<td>82</td>
</tr>
<tr>
<td>Bibliography</td>
<td>84</td>
</tr>
<tr>
<td>VITA</td>
<td>86</td>
</tr>
<tr>
<td>FIGURE</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>FIGURE 2.1</td>
<td>The Branching Structure of an Integer Program</td>
</tr>
<tr>
<td>FIGURE 2.2</td>
<td>Integer Branch and Bounding Sequence for Program 2.16</td>
</tr>
<tr>
<td>FIGURE 3.1</td>
<td>Flowchart Outline for Determining the Integer First Approximation</td>
</tr>
<tr>
<td>FIGURE 3.2</td>
<td>The Graphical Representation of the Branch Structure for Example 3.18</td>
</tr>
<tr>
<td>FIGURE 4.1</td>
<td>Magnitude Response for the Filter Described in Table 4.2</td>
</tr>
<tr>
<td>FIGURE 4.2</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.2</td>
</tr>
<tr>
<td>FIGURE 4.3</td>
<td>Branch Structure of the 32 Branch for Example 4.1</td>
</tr>
<tr>
<td>FIGURE 4.4</td>
<td>Magnitude Response for the Filter Described in Table 4.3</td>
</tr>
<tr>
<td>FIGURE 4.5</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.3</td>
</tr>
<tr>
<td>FIGURE 4.6</td>
<td>Magnitude Response for the Filter Described in Table 4.4</td>
</tr>
<tr>
<td>FIGURE 4.7</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.4</td>
</tr>
<tr>
<td>FIGURE 4.8</td>
<td>Passband Magnitude Response for the Filters Described in Table 4.2, Table 4.3, and Table 4.4</td>
</tr>
<tr>
<td>FIGURE 4.9</td>
<td>Magnitude Response for the Filter Described in Table 4.5</td>
</tr>
<tr>
<td>FIGURE 4.10</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.5</td>
</tr>
<tr>
<td>FIGURE 4.11</td>
<td>Magnitude Response for the Filter Described in Table 4.7</td>
</tr>
<tr>
<td>FIGURE 4.12</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.7</td>
</tr>
<tr>
<td>FIGURE 4.13</td>
<td>Magnitude Response for the Filter Described in Table 4.8</td>
</tr>
<tr>
<td>FIGURE 4.14</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.8</td>
</tr>
<tr>
<td>FIGURE 4.15</td>
<td>Magnitude Response for the Filter Described in Table 4.9</td>
</tr>
<tr>
<td>FIGURE 4.16</td>
<td>Passband Magnitude Response for the Filter Described in Table 4.9</td>
</tr>
<tr>
<td>FIGURE 4.17</td>
<td>Passband Magnitude Response for the Filters Described in Table 4.5, Table 4.7, Table 4.8, and Table 4.9</td>
</tr>
</tbody>
</table>
LIST OF TABLES

TABLE 4.1 Optimal Real Impulse Response Coefficients for the Multiplierless Application ........................................... 50
TABLE 4.2 Rounded Integer Solutions and their Costs ............................................................... 53
TABLE 4.3 Optimal Integer Impulse Response Coefficients for the Multiplierless Application ........................................... 59
TABLE 4.4 Amplitude Constrained Integer Impulse Response Coefficients for the Multiplierless Application ............................ 63
TABLE 4.5 Optimal Real Impulse Response Coefficients for the Finite Wordlength Application ........................................... 68
TABLE 4.6 Optimal Integer Impulse Response Coefficients for the Finite Wordlength Application ........................................... 73
TABLE 4.7 Optimal Fixed Length Impulse Response Coefficients for the Finite Wordlength Application ........................................... 73
TABLE 4.8 Amplitude Constrained Fixed Length Impulse Response Coefficients for the Finite Wordlength Application with One Amplitude Constraint ....................................................... 74
TABLE 4.9 Amplitude Constrained Fixed Length Impulse Response Coefficients for the Finite Wordlength Application with Two Amplitude Constraints ........................................... 77
CHAPTER 1

INTRODUCTION

Digital filters are used in a variety of applications including sonar, radar, communications, and biomedical engineering. They are used to improve the reception of cellular phones, to look for anomalies in EKGs, and to locate those pesky Libyan bombers headed for the heartland. While theoretical digital signal processing dates back to mathematical studies of the 17th and 18th centuries, modern implementation of digital filters was not achieved until the development of digital electronic computers in the 1950s. Advances in integrated circuit (IC) technology and design have allowed the creation of smaller, faster, and cheaper digital filters. Conventional digital filters often use three components, the delay, the add, and the multiply. Of these three components, the multiply is the slowest and requires the most area on an IC. If a filter could be designed using only delays and adds, the resultant filter could be faster and smaller than a conventional digital filter. This concept has been realized in the field of multiplierless filter design.

Transport processors are a type of multiplierless filter that allows only adds and delays in its implementation.
Integer impulse responses can be realized by convolving filter subsections in various architectural schemes [1]. Various filter design methods can be applied to the transport processor implementation [1] - [6].

Another type of multiplierless filter uses powers-of-two coefficients to implement a multiplierless digital filter [7] - [12]. These types of filters shift the binary point in a binary signal to effectively multiply it by a power of two. Depending upon the architectural implementation of the filter, such a filter's impulse response can have values which are only powers of two, or values which are the sum of multiple powers of two.

In this thesis, two optimal filter design techniques are introduced that determine the filter coefficients for an N point discrete time finite impulse response (FIR) digital filter with constant group delay. The first method minimizes a weighted least squared optimality criteria. The second method minimizes a weighted least squared optimality criteria subject to user specified frequency constraints. This second method can result in filters with improved stopband and passband amplitude responses in comparison to the filter produced by the first method. These techniques can be applied to both transport processors applications and filters with a finite wordlength implementation.
CHAPTER 2

DIGITAL SIGNAL PROCESSING AND OPTIMIZATION

Filter design can be accomplished by minimizing an optimality criteria. An optimality criteria can be chosen to minimize a measure between a desired frequency response and the designed filter's frequency response. Some common optimality measures include minimizing the weighted energy between the designed response and the desired response (a weighted least squared measure), minimizing the absolute difference between the designed frequency response and the desired response (a min-max measure which results in an equiripple filter), and the criteria that produces a filter with maximally flat responses (a maximally flat measure).

A weighted least squared optimality criteria produces a quadratic measure which can be expressed as a function of the filter's impulse response. Determining the optimal impulse response for a specific weighted least squared design criteria can be accomplished by various methods. One method for solving the quadratic measure can be achieved by setting the Jacobian of the optimality criteria with respect to the impulse response coefficients equal to zero and solving the
resulting system of equations. This is the approach used in this thesis.

A multiplierless filter design problem with a weighted least squared optimality criteria requires that the impulse response, \( h_n \), be a member of the set of integers, \( I \). These types of integer constrained optimization criteria can be solved by integer programming techniques. This thesis uses integer branch and bounding techniques to solve the problem.

2.1 Filter Frequency Response

The discrete time Fourier transform (DTFT) of a sequence, \( x_n \), can be expressed by

\[
X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_n e^{-j\omega n}, \quad (2.1)
\]

where \( X(e^{j\omega}) \) is the DTFT, \( x_n \) is a sequence of numbers, and \( \omega \) is the frequency in radians/sample. If a filter is causal and has a finite impulse response (FIR), \( h_n \), of length, \( N \), the filter's frequency response can be written as

\[
H(e^{j\omega}) = \sum_{n=0}^{N-1} h_n e^{-j\omega n}. \quad (2.2)
\]

For an FIR filter with a real impulse response to have
constant group delay, the necessary and sufficient condition is

\[ h_n = h_{N-1-n} \quad \text{for } n = 0, 1, \ldots, N-1, \]  

(2.3)

or

\[ h_n = -h_{N-1-n} \quad \text{for } n = 0, 1, \ldots, N-1. \]  

(2.4)

In this thesis, the condition in (2.3) will be referred to as even symmetry, and the condition in (2.4) will be referred to as odd symmetry [13].

Depending upon whether \( N \) is even or odd and whether the impulse response has even or odd symmetry, four filter types can be produced [13]. The frequency responses of these filter types are

Case 1. Even Symmetry (\( h_n = h_{N-1-n} \)) and \( N \) even

\[ H(e^{j\omega}) = e^{-j\frac{N-1}{2}\omega} \sum_{n=0}^{N-1} 2h_n \cos\left(\frac{N-1}{2} - n\right) \omega. \]  

(2.5)

Case 2. Even Symmetry (\( h_n = h_{N-1-n} \)) and \( N \) odd
\[ H(e^{j\omega}) = e^{-j\frac{N-1}{2}\omega} \left[ h_{\frac{N-1}{2}} + \sum_{n=0}^{\frac{N-1}{2}} 2 h_n \cos \left( \left( \frac{N-1}{2} - n \right) \omega \right) \right]. \]  \hspace{2cm} (2.6)

Case 3. Odd Symmetry \((h_n = -h_{-n})\) and \(N\) even

\[ H(e^{j\omega}) = e^{-j\left(\frac{N-1}{2}\omega - \frac{\pi}{2}\right)} \sum_{n=0}^{N-1} 2 h_n \sin \left( \left( \frac{N-1}{2} - n \right) \omega \right). \]  \hspace{2cm} (2.7)

Case 4. Odd Symmetry \((h_n = -h_{-n})\) and \(N\) odd

\[ H(e^{j\omega}) = e^{-j\left(\frac{N-1}{2}\omega - \frac{\pi}{2}\right)} \sum_{n=0}^{\frac{N-1}{2}-1} 2 h_n \sin \left( \left( \frac{N-1}{2} - n \right) \omega \right). \]  \hspace{2cm} (2.8)

For filters described by Case 3 or Case 4, \(H(e^{-j\omega}) = 0\). Thus these filter cases are unsuitable for lowpass filter applications. For filters described by Case 1 or Case 4, \(H(e^{j\omega}) = 0\), and as a result, these cases are unsuitable for highpass filter applications.

2.2 Digital Filter Design for Filters with Constant Group Delay

FIR digital filters with constant group delay are commonly designed by optimizing a measure between a desired
magnitude frequency response and the designed filter's magnitude frequency response. The design process will specify magnitude responses for the passbands and the stopbands. Transition bands between the passband and stopband may also be specified. To constrain the designed filter response within a certain distance of the desired filter response, passband and stopband magnitude response limits can be imposed. The order of the filter, phase response restrictions, and more can be specified for the filter design. To design a filter to particular specifications, an optimality criteria that can satisfy the filter specifications must be determined.

Some common FIR filter design methods minimize the energy between the designed filter's amplitude response and the desired filter's amplitude response over specified frequency ranges [13] - [15]. Other methods minimize the absolute difference between the designed filter's amplitude response and the desired filter's amplitude response [13] - [15], and other FIR filter design methods produce filters with maximally flat amplitude responses [13] - [15].

In this thesis, a weighted least squared optimality criteria is applied to the constant group delay FIR filters described in Equations (2.5) - (2.8). The optimization criteria minimizes the integrated squared error between the filter's desired amplitude response and the filter's designed amplitude response over a particular range of frequencies. In
particular, this method minimizes the optimality criteria, $E$, where

$$E = \int_{-\pi}^{\pi} W(\omega)(H(\omega) - H_d(\omega))^2 d\omega, \quad (2.9)$$

$W(\omega)$ is a weighting function, $H(\omega)$ is the designed amplitude response, $H_d(\omega)$ is the desired amplitude response, and $\omega$ is the frequency in radians/sample [2]. If the impulse response values are restricted to be real numbers, $H(e^{j\omega}) = H'(e^{j\omega})$ for $-\pi < \omega < \pi$, where $H'(e^{j\omega})$ is the complex conjugate of $H(e^{j\omega})$. This implies that $H(\omega) = H(-\omega)$ and thus the error measure can be expressed as

$$E = \int_{0}^{\pi} W(\omega)(H(\omega) - H_d(\omega))^2 d\omega. \quad (2.10)$$

The weighting function, $W(\omega)$, is used to place emphasis on certain regions of the amplitude response over other regions of the amplitude response in order to obtain a particular amplitude response. Weighting functions can be determined by various means. A trial and error method can be used by breaking the frequency range into zones and assigning different weights for each zone. For example, if an unweighted designed frequency response does not meet a desired
frequency response specification in a section of the passband frequency, a weighting function, \( W(\omega) \), could be created composed of three zones, a zone less than the passband area, a zone around the passband area, and a zone greater than the passband area. Corresponding weights of 1, 10, and 1 could be assigned to the three zones. If the designed filter still does not satisfy the desired filter's specifications, \( W(\omega) \)'s passband zone coefficient could be increased, or \( W(\omega) \) could be redesigned with more zones and different coefficient values.

The choice of the number of frequency zones, the frequency zone breakpoints, and the frequency zone weights are all variable. The drawback to this method is that a weighting function may not exist which meets all of the desired filter's specifications. Examples of sophisticated weighting function design methods are described in [2] and [16].

Because some weighting functions can be difficult to integrate, the integral in (2.10) can be difficult to obtain. An alternative to integrating Equation (2.10) can be adequately approximated by

\[
E = \sum_{k=0}^{L-1} W(\omega_k) ( H(\omega_k) - H_d(\omega_k) )^2 , \tag{2.11}
\]

where \( 0 \leq \omega \leq \pi \) are discrete frequencies. Equation (2.11) is a reasonable approximation to Equation (2.10) provided that \( L \) is sufficiently large.
2.3 Optimization

Optimization problems minimize or maximize a cost function, which is called an objective or an objective measure, subject to certain constraints on the variables of the cost function. For example, the standard form for an optimization problem with constraint equations is

\[
\text{maximize:} \quad z = f(\mathbf{x}) \\
\text{subject:} \quad g_1(\mathbf{x}) = 0 \quad (2.12) \\
\quad g_2(\mathbf{x}) \leq 0 \\
\quad \vdots \\
\quad g_m(\mathbf{x}) \geq 0 \\
\text{with:} \quad \mathbf{x} = [x_1, x_2, \ldots, x_n]^T,
\]

where \( z \) is called the objective measure, \( \mathbf{x} \) is called the objective variable vector, \( f(\mathbf{x}) \) is called the objective function, \( g_i(\mathbf{x}) \) are the objective constraint functions and \( m < n \) (which implies fewer constraints than variables). The system of equations in (2.12) is termed a mathematical program. The objective function and the constraint functions can be linear, nonlinear, or a combination of both linear and nonlinear functions. The constraints can be equality constraints, inequality constraints, or both.
2.4 Lagrange Multipliers

Mathematical programs that have linear equality constraint functions can be solved using the method of Lagrange multipliers. The Lagrange multiplier method creates an augmented cost function which is the sum of the objective function and the constraint functions scaled by variables known as Lagrange multipliers.

For example, consider the mathematical program described by (2.12), where the objective function and constraint functions are linear equality constraints. The augmented cost function or Lagrangian function for this program is

\[
L(x_1, x_2, \ldots, x_n, \lambda_1, \lambda_2, \ldots, \lambda_m) = f(x) - \sum_{i=1}^{m} \lambda_i g_i(x), \quad (2.13)
\]

where \( \lambda_i \) are variables called Lagrange multipliers. The solution to the mathematical program can be found by solving the \( n + m \) equations,

\[
\frac{\partial L}{\partial x_j} = 0 \quad (j = 1, 2, \ldots, n) \quad (2.14)
\]

\[
\frac{\partial L}{\partial \lambda_i} = 0 \quad (i = 1, 2, \ldots, m).
\]

If a solution to the mathematical program exists, it is contained among the solutions to the system of equations described in (2.14), provided \( f(x) \) and \( g_i(x) \) all have
continuous first partial derivatives and the \( m \times n \) Jacobian matrix,

\[
J = \left[ \frac{\partial g_i}{\partial x_j} \right], \quad (2.15)
\]

has rank \( m \) at \( \mathbf{x} = \mathbf{x}^* \), where \( \mathbf{x}^* \) is the optimal solution.

2.5 Integer Programming

When a mathematical program requires an integer valued solution, the mathematical program is initially solved without integer constraints. If the resulting solution is not completely composed of integers, then two new programs are created with additional constraints placed upon one of the noninteger solution elements. For example, suppose \( x_i \) in the first solution is not an integer, and \( i_1 < x_i < i_2 \) where \( i_1 \) and \( i_2 \) are two consecutive integers. One of the new programs has the additional constraint \( x_i \leq i_1 \) and the other new program has the additional constraint \( x_i \geq i_2 \). This branching process shrinks the feasible solution region in a manner that eliminates the current noninteger solution while allowing all possible integer solutions to the original problem. Both of these two new programs must be solved. If the solutions to either of these two new programs are noninteger, then two additional programs are created for each program which returned a noninteger solution. The newly created programs
have additional constraints placed upon one of the noninteger solution elements.

The original program can require two new programs each of which can require two new programs and so on. This branching effect creates the tree-like structure shown in Figure 2.1. Each of these branchings reduces the feasible solution region away from noninteger solutions. To ensure that the optimal integer solution is obtained, each of the branches must be solved until each branch terminates. A branch is said to be terminated if an integer solution is reached, or if the added constraints create a program which has no solution, or the solution of the branch returns a cost objective which is less than the cost objective of an integer solution which has previously been obtained. Once all of the branches are terminated, all of the integer solutions' cost objectives are

![Figure 2.1](image-url)  
*Figure 2.1 The branching structure of an integer program.*
compared with one another, and the optimal integer solution is retained.

A method to expedite this search is to pursue only the branch which returns the more optimal cost objective at each branching until an integer solution is reached. This integer solution is called the integer first approximation to the program, and its cost objective is called the initial integer cost. All of the remaining branches must still be solved until they are terminated, but the possession of an initial integer solution allows any branch which has a cost objective which is worse than the initial integer cost to be terminated. This process reduces the number of branches which must be searched by eliminating nonoptimal branches earlier in the search process. If a new integer solution that has a more optimal cost objective is found, then this new integer solution and the integer cost are used as the current optimal integer solution. After all of the branches have been terminated, the optimal integer solution is the current integer first approximation, and the optimal integer cost objective is the initial integer cost.

For example, consider the integer optimization program

\[
\begin{align*}
\text{maximize: } & \quad 11x_1 + 16x_2 \\
\text{subject to: } & \quad 6x_1 + 8x_2 \leq 39 \quad (2.16) \\
\text{where: } & \quad x_1, x_2 \geq 0
\end{align*}
\]
The branching process for this program is illustrated in Figure 2.2 where the circled numbers indicate the branch number, the numbers in parentheses are the optimal solution pairs to that branch, the number below the solution pairs is the objective cost for that branch, and the constraints for each branch are indicated along the interconnecting branch lines.

The optimal real solution to this program is \( x_1 = 0, \ x_2 = 39/8 \ (0, 39/8) \), with an optimal objective value of 78. Because the solution is not completely integer valued, two new programs are created. The first new program is identical to Program 2.16 with the additional constraint \( x_2 \leq 4 \). The optimal solution pair for this new program is \( (7/6, 4) \) and its objective value is 76.83. The second program is identical to Program 2.16 with the additional constraint \( x_2 \geq 5 \). However if \( x_2 > 5, \ 8x_2 > 39 \), which violates the constraint in Program 2.16. Therefore this branch is terminated. Because the solution on Branch 2 is noninteger, two new programs are created by adding additional constraints to the program at Branch 2. Branch 4 produces the first integer solution to Program 2.16. This integer solution is the first integer approximation to Program 2.16, and has a cost of 75.

To search for a better integer solution, Branch 5 is solved resulting in the noninteger solution \( (2, 27/8) \) and a cost value of 76. Because the cost value of Branch 5 is greater than the cost value of the integer approximation at
Program 2.16

maximize $11x_1 + 16x_2$
subject to $6x_1 + 8x_2 \leq 39$
$x_1, x_2 \geq 0$

Figure 2.2 Integer Branch and Bounding Sequence for Program 2.16.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Branch 4, two new programs are created stemming from branch 5. The additional constraint on Branch 7 creates a program without a feasible solution, and thus this branch is terminated. The program of Branch 6 results in a noninteger solution, and thus two more programs are created. Branch 8 yields an integer solution resulting in the termination of this branch. Because the cost objective of Branch 8 is smaller than the initial integer cost, no integer solution update is required. Branch 9 yields a noninteger solution with an objective cost equal to the initial integer cost. Because the objective cost of Branch 9 will decrease if more constraints are added to its program, any branches stemming from Branch 9 will have a smaller objective cost than Branch 9. Because the objective cost of Branch 9 equals the initial integer cost, Branch 9 and all branches stemming from Branch 9 can be terminated. At this point, all branches have been terminated, and the optimal integer solution to Program 2.16 is the current initial integer approximation (1,4) which results in an optimal cost objective of 75.
CHAPTER 3

OPTIMAL DESIGN TECHNIQUES

This thesis introduces a method of determining the optimal impulse response for an FIR filter with constant group delay. Because transport processors require integer coefficients for their implementation, the impulse response is constrained such that \( h(n) \) is an integer. For finite wordlength applications, the integer coefficients can be scaled down to the appropriate resolution. This new design method approximates a desired frequency response by minimizing a weighted least squared error criteria. While other papers have approached this problem [1] - [2], none have produced a method which absolutely produces the optimal solution. Stubberud [1] uses a coefficient rounding scheme which produces optimal results, but the paper does not derive nor explain how this method is optimal. Lim and Parker [2] use a modified branch and bound integer programming method which "does not guarantee an optimal solution" but does produce a solution "significantly superior to simple coefficient rounding...". The reason Lim and Parker's method does not guarantee an optimal solution is that the method considers only a small subset of the total feasible solution region and
potentially ignores the region where the optimal solution exists.

The first design technique developed in this thesis generates a weighted least squared optimality criteria that when minimized produces an optimal real valued impulse response. This real solution is incorporated into a coefficient rounding algorithm that produces an optimum integer solution. This integer solution becomes the first integer approximation in a branch and bound integer program. The resulting solution from the integer program is the optimal integer impulse response for the weighted least squared error criteria.

The second design technique adds amplitude response constraints to the weighted least squared error criteria. These constraints achieve equiripple like properties in the amplitude response of the designed filter. These frequency constraints are incorporated in the optimality criteria by the addition of Lagrangian multipliers to the optimality criteria that is created using the first technique. The solution process parallels that of the first technique.

3.1 The Optimality Criteria

In this thesis, the designed filter is assumed to be a discrete time FIR filter with a constant group delay and a real impulse response. Because this filter will be used in a multiplierless application, the impulse response is
constrained to be integer, and it is assumed that the impulse response coefficients are bounded by some integer \( b \). For example, if the filter design specifies a signed eight bit filter, then the impulse response bound, \( b \), is 128, and \(-127 \leq h(n) \leq 128\). The desired filter is assumed to be of some constant value in the passband and zero in the stopband.

3.2 The Weighted Least Squared Optimality Criteria

The weighted least squared optimality criteria for a constant group delay FIR filter was derived in Chapter 2 as

\[
E = \sum_{k=0}^{L-1} W(\omega_k) (H(\omega_k) - H_d(\omega_k))^2. \tag{3.1}
\]

The desired amplitude response is assumed to be a constant \( A \) in the passband and zero in the stopband. The frequency responses for constant group delay FIR filters are described by Equations 2.5 - 2.8. In general, these equations can be written as

\[
H(\omega) = h^T t(\omega), \tag{3.2}
\]

where \( H(\omega) \) is the amplitude response, \( h \) is a vector containing the filter coefficients, \( t(\omega) \) is a vector of trigonometric functions and the superscript \( T \) denotes the matrix transpose.
operation. In particular, for Case 1 (Even Symmetry, $h_n = h_{N-1-n}$, N even)

$$h = [h_0, h_1, \ldots, h_{\frac{N-1}{2}}]^T,$$

$$t_i(\omega) = 2\cos\left(\left(\frac{N-1}{2}-i\right)\omega\right), \quad i = 0, 1, \ldots, \frac{N-1}{2} ;$$

for Case 2 (Even Symmetry, $h_n = -h_{N-1-n}$, N odd)

$$h = [h_0, h_1, \ldots, h_{\frac{N-1}{2}}]^T,$$

$$t_i(\omega) = 2\cos\left(\left(\frac{N-1}{2}-i\right)\omega\right), \quad i = 0, 1, \ldots, \frac{N-1}{2} - 1,$$

$$t_{\frac{N-1}{2}}(\omega) = 1 ;$$

for Case 3 (Odd Symmetry, $h_n = -h_{N-1-n}$, N even)

$$h = [h_0, h_1, \ldots, h_{\frac{N-1}{2}}]^T,$$

$$t_i(\omega) = 2\sin\left(\left(\frac{N-1}{2}-i\right)\omega\right), \quad i = 0, 1, \ldots, \frac{N-1}{2} ;$$
and for Case 4 (Odd Symmetry, \( h_n = -h_{N-1-n} \), \( N \) odd)

\[
\mathbf{h} = [h_0 \ h_1 \ldots \ h_{\frac{N-1}{2}-1}]^T,
\]

\[
t_i(\omega) = 2 \sin \left( \left( \frac{N-1}{2} - i \right) \omega \right), \quad i = 0, 1, \ldots, \frac{N-1}{2}-1.
\]

Substituting Equation 3.2 into Equation 3.1,

\[
E(h, A) = \sum_{\{k: \omega_k \in \omega_{pb}\}} W(\omega_k)(A - \mathbf{h}^T \mathbf{t}(\omega_k))^2 + \\
\sum_{\{k: \omega_k \in \omega_{sb}\}} W(\omega_k)(\mathbf{h}^T \mathbf{t}(\omega_k))^2,
\]

(3.3)

where \( \omega_{pb} \) is the set of passband frequencies, \( \omega_{sb} \) is the set of stopband frequencies, and \( A \) is the passband amplitude of the desired filter. Defining

\[
h_{\omega} = \begin{bmatrix} A \\ \mathbf{h} \end{bmatrix},
\]

and

\[
\mathbf{T}(\omega_k) = \mathbf{t}(\omega_k) \mathbf{t}(\omega_k)^T,
\]
Equation 3.3 can be expressed as

\[ E(h_M) = \sum_{\{k: \omega_k \in \omega_{pb}\}} W(\omega_k) h_M^r T_p(\omega_k) h_M + \sum_{\{k: \omega_k \in \omega_{sb}\}} W(\omega_k) h_M^r T_s(\omega_k) h_M \quad (3.4) \]

\[ = h_M^r \left[ \sum_{\{k: \omega_k \in \omega_{pb}\}} W(\omega_k) T_p(\omega_k) + \sum_{\{k: \omega_k \in \omega_{sb}\}} W(\omega_k) T_s(\omega_k) \right] h_M, \]

where \( T_p(\omega_k) \) is the passband trigonometric matrix defined by

\[ T_p(\omega_k) = \begin{bmatrix} 1 & -t^r(\omega_k) \\ -t(\omega_k) & T(\omega_k) \end{bmatrix}, \]

and \( T_s(\omega_k) \) is the stopband trigonometric matrix defined by

\[ T_s(\omega_k) = \begin{bmatrix} 0 & 0 \\ 0 & T(\omega_k) \end{bmatrix}. \]

Defining

\[ T_A = \sum_{\{k: \omega_k \in \omega_{pb}\}} W(\omega_k) T_p(\omega_k) + \sum_{\{k: \omega_k \in \omega_{sb}\}} W(\omega_k) T_s(\omega_k), \]

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
the optimality criteria, or objective measure, in Equation 3.4 can be expressed as

\[ E(h_n) = h_n^T A_n h_n. \]  

(3.5)

Thus, the weighted least squared error design program for a constant group delay FIR filter with a real impulse response can be written as

\[
\text{minimize } \quad E(h_n) = h_n^T A_n h_n \\
\text{subject to } \quad -d \leq h(n) \leq b \\
\quad \quad \quad \quad \quad h(n) \in I, \quad n = 0, 1, \ldots, N,
\]

where \( b \) is the coefficient bound, and \( d = b - 1 \) if a fixed bit coefficient representation such as two's complement is used. Otherwise, \( d = b \).

3.3 The Amplitude Constrained Optimality Criteria

The weighted least squared error (WLSE) optimality criteria minimizes the energy difference between the designed filter's amplitude response and the desired filter's amplitude response. The min-max optimality criteria minimizes the absolute difference between the designed filter's amplitude response and the desired filter's amplitude response creating

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
an equiripple filter. The amplitude constrained optimality criteria used in this thesis combines the least squared error criteria with discrete amplitude constraints to create an optimality criteria which fills the middle ground between the WLSE design and the min-max design. At one extreme, if no amplitude constraints are imposed, the amplitude constrained optimality criteria becomes a WLSE optimality criteria, and at the other extreme, an equiripple filter can be designed with the appropriate number and placement of amplitude constraints in the amplitude constrained optimality criteria.

To constrain a filter's amplitude response to a particular value at a particular frequency, amplitude response constraints can be added to the weighted least squared error criteria. The amplitude response constraints are incorporated into the optimality criteria by using Lagrange multipliers for the specified amplitude response constraints. For example, if the amplitude response at the frequency $\omega_\ell$ is constrained to a certain value, the amplitude response equation is

$$H(\omega_\ell) = \alpha A,$$  \hspace{1cm} (3.7)

where $\omega_\ell$ is the discrete frequency of the constraint in radians/ sample, $H(\omega_\ell)$ is the frequency amplitude response at $\omega_\ell$ radians/ sample, $\alpha$ is a positive scalar, and $A$ is the desired passband magnitude. The amplitude constrained
optimality criteria for this example can be written as

\[ E(h, \lambda) = E(h) - \lambda (H(\omega_L) - \alpha) , \quad (3.8) \]

where \( \lambda \) is a Lagrangian multiplier. In general, multiple amplitude response constraints can be added to the optimality criteria. The optimality criteria for a design program with \( M \) amplitude constraints can be written as

\[ E(h, \lambda) = E(h) - \sum_{i=1}^{M} \lambda_i (H(\omega_{L,i}) - \alpha_i A) , \quad (3.9) \]

where \( \lambda_i \) are the Lagrangian multipliers, \( \omega_{L,i} \) are the discrete frequencies of the desired constraints in radians/sample, and \( \alpha_i \) are positive constraint scalars.

The amplitude response constraint equation used in optimality criteria must be expressed as a fraction of \( A \). If the amplitude response constraint is expressed in dB, this value can be converted from dB to a fractional passband amplitude value by using the equation

\[ \alpha = 10^{\text{ATTN}/20} , \quad (3.10) \]

where ATTN is the amplitude response constraint expressed in dB. For example, if a stopband attenuation constraint of -40
dB is specified, $\alpha = 0.01$. For an allowable passband deviation of 0.1 dB, $\alpha = 0.989$ to constrain the signal 0.1 dB below the passband magnitude and $\alpha = 1.012$ to constrain the signal 0.1 dB above the passband magnitude.

The design program for a constant group delay FIR filter with an amplitude constrained optimality criteria can be expressed as

$$\begin{align*}
\text{minimize } & \quad E(h^*, \lambda) = E(h^*) - \sum_{i=1}^{M} \lambda_i (H(\omega_{Li}) - \alpha_i A) \\
\text{subject to } & \quad -(b-1) \leq h(n) \leq b \\
& \quad h(n) \in I; \quad n = 0, 1, \ldots, N, \\
& \quad \sum_{i=1}^{M} (H(\omega_{Li}) - \alpha_i A) = 0.
\end{align*}$$

(3.11)

This design program is capable of designing FIR filters ranging from a WLSE filter to a min-max filter.

When using the amplitude response optimality criteria in 3.10 for designing a multiplierless filter, the designer must be aware of the design limitations. Due to the finite length of the filter coefficients, multiple amplitude response constraints may not all be met. If this problem is encountered in the design process, the coefficient bound, $b$, can be increased, or the order of the filter can be increased,
or both can be increased in an attempt to meet all necessary amplitude response constraints.

3.4 Solving for the Optimal Real Valued Solution

To solve the weighted least squared design program in Equation 3.6, the program is first solved by setting the desired filter's passband magnitude, $A$, equal to 1 and removing the integer constraints on the impulse response. This program solution, which can be determined by solving the system of equations

$$\frac{\partial E(h_x)}{\partial h} = 0 \ , \quad (3.12)$$

results in a real valued impulse response. If the optimal solution of 3.12 with $A = 1$ is $h_1^*$, then the solution to 3.12 with $A$ unconstrained yields a family of optimal solutions described by

$$h_A^* = A \ h_1^* \ , \quad (3.13)$$

where the superscript * denotes an optimal solution vector. The optimal solution vector, $h_A^*$, produces the same relative filter amplitude response as the amplitude response of $h_1^*$; however, the resultant objective measure for $h_A^*$ increases proportional to the square $A$ in comparison to the objective.
measure of $h_i'$. To normalize the objective measures from 3.6, 
the objective measure is divided by $A^2$ such that

$$E_{R}(h_{M}) = \frac{h_{M}^T T_{A} h_{M}}{A^2}. \quad (3.14)$$

This error criteria returns the same value for any $A$ and $h_{A}'$.

The solution process for the amplitude constrained optimality program in Equation 3.11 closely parallels the solution process for the weighted least squared design program. The program is first solved by setting the desired filter's passband magnitude, $A$, equal to 1 and removing the integer constraints on the impulse response. The program solution to Equation 3.11 can be found by solving the system of equations

$$\frac{\partial E(h_{M})}{\partial h} = 0 ,$$

$$\frac{\partial E(h_{M})}{\partial \lambda} = 0 , \quad (3.15)$$

where $\lambda$ is the $M$ vector of Lagrange multipliers. The resultant optimal solution vector, $h_{i}'$, can be used to create
a family of optimal real valued impulse responses of the form

\[ h_A^* = A h_1^*. \] (3.16)

The relative error criteria for the amplitude constrained design program can be expressed as

\[ E_R(h, \lambda) = \frac{E(h) - \sum_{i=1}^{i=N} \lambda_i (H(\omega_L) - a_i A)}{A^2}. \] (3.17)

3.5 Reduction of the Feasible Solution Space

After the optimal solution vector, \( h_k^* \), is obtained, the design program is solved with the integer constraints included. The optimal integer solution can be found using the branch and bound integer programming method. The possession of an integer first approximation to the design program can greatly reduce the number of branches which must be searched in the branch and bound process. Another way to reduce the number of calculations required to find the optimal integer solution vector is to reduce the feasible solution space for the design program. Because the search for the integer first approximation can be simplified with a reduction in the feasible solution space, the reduction of the feasible solution space will be addressed first.
The feasible solution space for a multiplierless constant group delay FIR filter design program depends upon the number of unique filter coefficients $M$ and the coefficient bound $b$. The total number of solutions to the design program is $(2b)^M$. For a simple eight point filter which has four unique filter coefficients and a coefficient bound of eight, $65536$ feasible integer solutions exist. The feasible solution space extends from the solution $[8,8,8,8]$ to the solution $[-7,-7,-7,-7]$. Many of the integer solutions in this feasible space are scalar multiples of another solution ($[6,4,-2,0]$ and $[3,2,-1,0]$), one solution is trivial ($[0,0,0,0]$), and other solutions can be excluded for other reasons.

The first great reduction in the feasible solution space can be accomplished by reducing multiple solutions. Call the element in $h_j$ with the largest magnitude $h_{\text{max}}$. $h_{\text{max}}$ can take on any integer value from $b - 1$ to $b$ in the feasible solution space, and each of these $2b$ values can be combined with the $(2b)^{M-1}$ possible combinations of the remaining $M-1$ elements. Any integer solution $h$, where $h_{\text{max}} < 0$ has a mirror image integer solution $-h$ which will produce the same filter response as $h$. Therefore, the feasible solution space can be reduced to exclude any solution where $h_{\text{max}}$ is negative. For the example above, the feasible solution space would now contain $36864$ integer solutions ranging from the solution $[8,8,8,8]$ to $[0,-7,-7,-7]$ (assuming $h_{\text{max}}$ is the first element in this example).
Because $h^*_A$ is the optimal solution to the design program, any deviation from the $h^*_A$ solution vector results in an increase in the objective measure. So the optimal integer solution will lie somewhere near (based on the Euclidean measure) the optimal solution vector. This property allows the feasible solution space to exclude solution vectors where $h_{a_1} = 0$ because any integer solution in this region of the optimal solution vector will not have an objective measure lower than the trivial optimal integer zero solution. The fact that the optimal integer solution exists near the optimal solution vector allows the exclusion of any integer solution where $h_{\text{max}}$ is less than the first integer greater than $b/2$. Continuing the above example, the feasible solution space is reduced to 4096 possible integer solutions ranging from $[8,8,8,8]$ to $[4,-7,-7,-7]$.

Further reductions in the feasible solution space can be made, but the technique used for selecting an integer first approximation coupled with the branch and bounding process effectively ignores these regions.

3.6 The Integer First Approximation

A good first integer approximation can be obtained by comparing the integer vectors nearest the optimal solution vector $h^*_A$ for many values of $A$, and retaining the integer vector which returns the smallest objective measure. The first step toward finding the integer first approximation
requires the determination of the element in $h_1$ with the largest magnitude, $h_{\text{max}}$. $h_1$ is then scaled such that $h_{\text{max}}$ is equal to the coefficient bound, and the resultant vector is named $h_1^\prime$. The elements of $h_1$ are then rounded to their nearest neighboring integer, and the resulting integer solution vector is named $h_1^0$. This process is repeated with $h_1$ being scaled such that $h_{\text{max}}$ is equal to $b-1$, and the elements of $h_{b-1}$ are rounded to their nearest neighboring integer forming $h_{1(b-1)}$. This process is continued, decrementing the value $h_{\text{max}}$ by one each iteration, until $h_{\text{max}}$ is equal to $c$, the first integer greater than or equal to $b/2$. After the $(b-c+1)$ integer solution vectors have been found, the optimal passband magnitude response, $A_n$, must be found for each integer solution vector by solving the equation

$$\frac{\partial E_b(h_n)}{\partial A_n} = 0; \quad n = c, c+1, \ldots, b. \quad (3.18)$$

The $(b-c+1)$ objective measures can be calculated by substituting

$$h_n = h_{1n} = \begin{bmatrix} A_n \\ h_{1n} \end{bmatrix}, \quad (3.19)$$

into Equation 3.14 for each integer solution vector. The integer vector solution which produces the smallest objective
measure is called the first integer approximation $h_{\text{IFA}}$. This solution process can be implemented in a loop where the optimal integer solution vector is updated in the first integer approximation vector $h_{\text{IFA}}$ and its objective measure is updated in the current cost variable $c_{\text{IFA}}$. This loop is illustrated in the flowchart in Figure 3.1.

For example, consider the weighted least squared design program

$$E(h_{\mathbf{w}}) = h_{\mathbf{w}}^T T_A h_{\mathbf{w}} ,$$

where

$$T_A = \begin{bmatrix}
121.00 & -43.94 & -121.78 & -193.43 & -236.29 \\
-43.94 & 792.21 & -156.44 & 35.76 & 206.45 \\
-121.78 & -156.44 & 865.57 & 17.23 & 160.31 \\
-193.43 & 35.76 & 17.23 & 990.11 & -28.91 \\
-236.29 & 206.45 & 160.31 & -28.91 & 800.90 \\
\end{bmatrix} ,$$

$$h_{\mathbf{w}} = [ A \quad h_0 \quad h_1 \quad h_2 \quad h_3 ]^T ,$$

and the integer coefficient bound is $b = 8$. Solving Equation 3.12,

$$\frac{\partial E(h_{\mathbf{w}})}{\partial h} = 0 ,$$
Input the filter design program.

\[ b = \text{Integer coefficient bound} \]
\[ c = \text{int}(b + 1)/2 \]
\[ \text{count} = 1\%

Solve the design program for the optimal real vector solution, \( h_i' \).

Scale \( h_i' \) such that \( h_{\text{max}} = b \).

Round each element of \( h_i' \) to the nearest integer. The resultant integer vector solution is \( h_i^b \).

Calculate \( h_i^b \)'s objective measure, \( c_{i}^b \).

\[ \text{count} = 1? \]
\[ N \]

\[ C_{i}^b < C_{i}^{\text{FA}}? \]
\[ N \]

\[ C_{i}^{\text{FA}} = C_{i}^b \]
\[ h_{i}^{\text{FA}} = h_{i}^b \]

\[ b = b - 1 \]
\[ \text{count} = \text{count} + 1 \]

\[ b < c? \]
\[ Y \]

DONE

Figure 3.1 Flowchart outlining the process for obtaining the integral first approximation, \( h_{i}^{\text{IFA}} \), for a design program.
produces $h' = [ -0.0133 0.0806 0.2029 0.2896 ]$, where $h_{\text{max}} = 0.2896$. Scaling the optimal solution vector such $h_{\text{max}} = 8$ produces $h_8' = [ -0.3674 2.2265 5.6050 8 ]$ and rounding each element to their nearest integer produces $h_{18}' = [ 0 2 6 8 ]$. Using Equation 3.18, $A=\text{equals} 28.2693$ which implies that $h_{\text{IFA}} = h_{m8} = [ 28.2693 0 2 6 8 ]$ and that the objective measure $c_{\text{IFA}}$ equals 4.4614. Next, $h_1'$ is scaled such $h_{\text{max}} = 7$ which implies that $h_7' = [ -0.3215 1.9482 4.9044 7 ]$. Rounding produces $h_{17}' = [ 0 2 5 7 ]$, $h_{m7} = [ 24.5288 0 2 5 7 ]$ and the associated cost objective is 4.2093. Since this cost objective is less than $c_{\text{IFA}}$, $c_{\text{IFA}} = 4.2093$ and $h_{\text{IFA}} = [ 24.5288 0 2 5 7 ]$. Continuing the process, $h_1'$ is scaled such that $h_{\text{max}} = 6$ producing $h_6' = [ -0.2756 1.6699 4.2037 6 ]$ which results in $h_{16}' = [ 0 2 4 6 ]$, $h_{m6} = [ 20.8918 0 2 4 6 ]$ and an associated cost objective of 4.4460. Since this cost objective is greater than $c_{\text{IFA}}$, no updating is required. Finally $h_1'$ is scaled such that $h_{\text{max}} = 5$ producing $h_5' = [ -0.2296 1.3916 3.5031 5 ]$ which results in $h_{15}' = [ 0 1 4 5 ]$, $h_{m5} = [ 17.9659 0 1 4 5 ]$ and an associated cost objective of 5.3952. This cost objective is greater than $c_{\text{IFA}}$, and so, the first integer approximation is $h_{\text{IFA}} = [ 24.5288 0 2 5 7 ]$ with an associated cost objective $c_{\text{IFA}} = 4.2093$.

3.7 The Branch and Bound Method

After obtaining the integer first approximation, the
The branch and bound method can be applied to the design program to determine the optimal integer impulse response. For a filter with $M$ unique coefficients and an integer coefficient bound $b$, $(2b)^M$ feasible integer solution vectors exist. The feasible solution space can be reduced as described in Section 3.6. The feasible solution space can be represented as a branching structure that has $M$ levels of branches where $2b$ branches stem from each branch in the level above. Each level constrains one of the filter coefficients to an integer value. The first level constrains the element in the optimal solution vector, $h_1'$, with the greatest magnitude to an integer. The second level adds the constraint that the element in $h_1'$ with the second greatest magnitude is an integer, and so on with the last level adding the constraint that the element in $h_1'$ with the smallest magnitude is an integer. At the last level, every element of $h_1'$ has been constrained to an integer producing an integer solution vector.

Each branch in the feasible solution space must be terminated to ensure that the optimal integer solution vector has been obtained. A branch can be terminated when the objective measure of the branch exceeds the current integer solution vector's objective measure. A branch can also be terminated when an integer solution vector is produced.

The branch and bound process begins by initializing a current optimal integer solution vector, $h_{\text{ICO}}$, and its objective measure $c_{\text{ICO}}$. Initially, $h_{\text{ICO}} = h_{\text{IPA}}$, and $c_{\text{ICO}} = c_{\text{IPA}}$. 

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Whenever an integer solution vector which is obtained in the branch and bound design process produces an objective measure less than $c_{icc}$, $h_{ico}$ and $c_{icc}$ are updated with the new integer solution vector and its objective measure respectively. On each level, each feasible branch is solved for the optimal solution vector, and the most optimal branch is followed to the next level down. On this level the process of pursuing the most optimal feasible branch is continued until the branch is terminated. At this point, all of the feasible branches which were bypassed must be searched until they are terminated. This depth first approach produces integer solution vectors earlier in the search which can be used to terminate branches that might be searched if a breadth first approach is used.

The branch and bound search starts on the first level where $h_{max}$ is constrained to an integer. On this level there are $b - c + 1$ feasible optimal solution vectors, where $c$ is the lower bound of the feasible region for $h_{max}$ as described above. The optimal solution vectors are $h_{b}^{*}, h_{b-1}^{*}, \ldots, h_{c}^{*}$. Because all of these optimal solution vectors produce the same objective measure, any branch from this level can be pursued. For the sake of order, the branch defined by $h_{max} = b$ is pursued first. Later, the branch defined by $h_{max} = b - 1$ is searched next, and so on until the branch defined by $h_{max} = c$ is searched last.
On the second level, the element of second largest magnitude in $\mathbf{h}_1'$ is constrained to the integer nearest the corresponding element of the optimal solution vector from the branch above. The optimal solution vector for this level is obtained. If the resultant objective measure is less than $c_{iCO}$ then the element of second largest magnitude is constrained to the next larger integer. This process continues until either the resultant objective measure is greater than $c_{iCO}$, or the coefficient bound $b$ is exceeded. Similarly, this process extends in the negative direction until the resultant objective measure is greater than $c_{iCO}$, or the coefficient bound exceeds $-b + 1$. The objective measures are compared for all of the feasible solution vectors on this level, and the most optimal branch is pursued to the next level. This process continues pursuing the most optimal feasible vector solution from each level down to the next level until the branch is terminated.

Once the branch has been terminated, the bypassed branches must be pursued until they too are terminated. Once all of the branches stemming from $h_{\text{max}} = b$ are terminated, the branch stemming from $h_{\text{max}} = b - 1$ can be pursued. This process continues until the branch structure stemming from $h_{\text{max}} = c$ is searched. Once all of the feasible solution region has been searched, the optimal integer vector solution for the design program is the current optimal integer vector solution, $\mathbf{h}_1' = \mathbf{h}_{iCO}$. 

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
A branch naming nomenclature is used to simplify the description of the current branch level in the design process. The current branch is defined by the integer values that led to the current branch starting from the first level. For example, if the current branch is on the third level, and stemmed from the impulse response coefficient on the first level being set to 15, and the impulse response coefficient on the second level being set to 6, then the current branch is termed branch 15/6 or the 15/6 branch.

Continuing the example from the previous section, the optimal solution vector is $h_i^* = [-0.0133, 0.0806, 0.2029, 0.2896]$, the current optimal integer vector solution is equal to the integer first approximation, $h_{ico} = h_{ifa} = [24.5288, 0, 2, 5, 7]$ with $c_{ico} = c_{ifa} = 4.2093$, $b = 8$, and $c = 5$. On the first level, four feasible branches exist. The top most branch, branch 8, produces the optimal solution vector $[27.6243, 0.3674, 2.2265, 5.6050, 8]$, branch 7 produces $[24.1713, -0.03215, 1.9482, 4.9044, 7]$, branch 6 produces $[20.7182, -0.2756, 1.6699, 4.2037, 6]$, and branch 5 produces $[17.2652, -0.2296, 1.3916, 3.5031, 5]$. Each of these branches has an objective measure of 4.0812, which is less than $c_{ico}$, and are therefore feasible branches. The search is continued on branch 8 down to the next level. The integer constraint is imposed on the fourth element of the optimal solution vector for branch 8. Constraining the fourth element of branch 8 to the nearest integer and solving produces branch 8/6's optimal solution.
vector \[ [28.26, -0.375, 2.279, 6, 8] \] with an objective measure of 4.2077. Since this measure is less than \( c_{\text{zii}} \), the next larger branch is considered for feasibility. Branch 8/7 is solved producing \( [29.89, -0.396, 2.41, 7, 8] \) with an objective measure of 5.4792. Since this measure is larger than \( c_{\text{zii}} \), the 8/7 branch and the 8/8 branch can be terminated. Searching the lower branches on the second level of branch 8 produces \( [26.64, -0.353, 2.148, 5, 8] \) on the 8/5 branch with a cost of 4.4118. This cost is greater than \( c_{\text{zii}} \) and results in termination of the 8/5 branch and all lower branches (branch 8/4, branch 8/3, ...).

The 8/6 branch is the only feasible branch stemming from branch 8 that leads to the third level. The constraint for this level is on the third element of the optimal vector solution for the 8/6 branch. Constraining this element to the nearest larger integer produces the 8/6/3 branch's optimal solution vector \( [28.71, -0.381, 3, 6, 8] \) with a cost of 4.6343. Because this cost is greater than \( c_{\text{zii}} \), this branch and all higher branches (branch 8/6/4, branch 8/6/5, ...) are terminated. The 8/6/2 branch produces \( [28.09, -0.372, 2, 6, 8] \) with a cost of 4.3051 resulting in the termination of this branch and all lower branches. At this point, all branches on the third level have been terminated. Because no feasible branches on the second level exist, the search of branch 8 has been completed. The search is continued on the 7 branch. Constraining the fourth element of branch 7's optimal solution
vector to the nearest larger integer produces \[ [24.32\ -0.323 \\
1.961\ 5\ 7]\] with a cost of 4.0921. Less than \(c_{\infty}\), the cost of the 7/5 branch necessitates a search of the higher branches. Branch 7/6 produces \[ [25.95\ -0.344\ 2.092\ 6\ 7]\] with a cost of 5.2255 which terminates branch 7/6 and all higher branches. Branch 7/4 produces \[ [22.70\ -0.301\ 1.830\ 4\ 7]\] with a cost of 5.0974 which terminates this branch and all lower branches. The feasible 7/5 branch leads to branch 7/5/2 which produces \[ [24.35\ -0.323\ 2\ 5\ 7]\] with a cost of 4.0931. Branch 7/5/3 produces \[ [24.96\ -0.331\ 3\ 5\ 7]\] with a cost of 5.3900 which terminates this branch and all higher branches. Branch 7/5/1 produces \[ [23.74\ -0.315\ 1\ 5\ 7]\] with a cost of 5.3683 which terminates this branch and all lower branches. Again only one viable branch remains which leads to branch 7/5/2/0. The optimal solution vector for this branch is \[ [24.32\ 0\ 2\ 5\ 7]\] with a cost of 4.2192 which is equal to \(h_{55}\). All of the higher branches are terminated because the branches above cannot be better than this branch. Branch 7/5/2/-1 produces \[ [24.42\ -1\ 2\ 5\ 7]\] with a cost of 4.7421, a nonoptimal integer vector solution. The search on branch 7 is now complete. The first search on branch 6 is the 6/5 branch which produces the optimal vector solution \[ [22.01\ -0.292\ 1.774\ 5\ 6]\] with an objective cost of 4.9217. This cost is greater than \(c_{\infty}\) which results in the termination of this branch and all higher branches. Branch 6/4 produces \[ [20.39\ -0.270\ 1.644\ 4\ 6]\] with a cost of 4.1458, necessitating the search of branch 6/3.
Branch 6/3 produces the optimal vector solution \([18.76 - 0.249 1.513 3 6]\) with a cost of 6.7164 which terminates this branch and all lower branches. Branch 6/4 leads to branch 6/4/2 which produces \([20.60 -0.273 2 4 6]\) with a cost of 4.3984. This cost leads to the termination of this branch and all higher branches. Branch 6/4/1 produces the optimal vector solution \([19.99 -0.265 1 4 6]\) with a cost of 4.8902 which leads to the termination of this branch and all lower branches. This completes the search of branch 6. The first branch to be searched off of Branch 5 is the 5/4 branch which produces the optimal vector solution \([18.07 -0.240 1.457 4 5]\) with an objective cost of 4.5673. This cost is greater than \(c_{ii}:\) and results in the termination of this branch and all higher branches. The 5/3 branch produces the optimal solution vector \([16.45 -0.218 1.326 3 5]\) with a cost of 4.6806. This cost results in the termination of this branch and all lower branches, and branch 5 has no remaining feasible branches. At this point, all of the branches for this design program have been searched resulting in the optimal integer vector solution \(\mathbf{h}_f = [24.32 0 2 5 7].\) A graphical representation of this example's solution process is shown in Figure 3.2.
Figure 3.2  The graphical representation of the branch structure search for the optimal integer solution vector to filter design Example 3.18, $b = 8$, $N = 4$. $X$ denotes a branch termination.
3.8 Filter Order and Other Factors

Many filter design problems specify passband and stopband criteria for the filter amplitude response. For example, a lowpass filter design might specify

\[-P \leq 20 \log_{10}|H(\omega)| \leq P \quad 0 < \omega < \omega_p,\]

\[20 \log_{10}|H(\omega)| \leq S \quad \omega_s < \omega < \pi,\]

where \(\omega_p\) is the passband cutoff frequency in radians/s, \(\omega_s\) is the stopband cutoff frequency in radians/s, \(P\) is the passband threshold in dB, and \(S\) is the stopband attenuation in dB. These filter design specifications are incorporated into the filter design problem through the length of the filter \(N\), the weighting function, \(W(\omega)\), and through the coefficient bound \(b\).

No equation exists that specifies an \(N\) for a given set of stopband/passband restrictions for the multiplierless filter design problem, so \(N\) must be determined empirically. An \(N\) is selected, the program solved, and the amplitude response of the resultant filter is checked to see if the filter specifications have been met. If the specifications have not been met, \(N\) is increased, and the design program is solved again until the filter specifications are satisfied. If the
specifications are exceeded, a smaller $N$ can be tried. The smallest possible filter is desired in order to reduce the processing time in finding an optimal solution to the design program as well as requiring less space to implement on a chip.

Because obtaining the optimal integer solution to the filter design program can be a very lengthy problem, an alternative to solving the optimal integer solution is to use the integral first approximation or even the optimal real solution to see if the filter specifications are being met. The filter design could be overspecified in anticipation of losing some fidelity in the amplitude response of the optimal integer solution.

$b$ is an important variable in the filter design program. If many of the impulse response coefficients in the optimal integer vector solution are zero, or if increasing $N$ does not improve the filter response performance, $b$ may need to be increased. If a more compact filter is desired, acceptable reductions in filter response characteristics may be achieved with smaller values for $b$.

The weighting function, $W(\omega)$, can be found empirically or through a weighting function design algorithm such as the one described in [2] or [16]. The judicious application of a weighting function to a filter design problem can greatly reduce the order of the filter required to meet specified filter requirements.
CHAPTER 4

APPLICATIONS

The design techniques described in Chapter 3 can be applied to multiplierless FIR filter design problems as well as finite wordlength FIR filter design problems. Two applications, a multiplierless integer coefficient problem and a finite wordlength problem, are presented. Both of these applications are solved using both the weighted least squared error criteria and the amplitude constrained error criteria.

4.1 Multiplierless Application

Consider a linear phase FIR discrete time multiplierless filter that approximates the frequency response,

\[ H(e^{j\omega}) = e^{-j\omega \frac{(N-1)}{2}} H(\omega), \]

where

\[ H(\omega) = \begin{cases} A & 0 \leq \omega \leq \omega_p = 0.3 \pi \\ 0 & 0.4 \pi = \omega_s \leq \omega \leq \pi \end{cases} \]
A is a constant greater than 1, $\omega_p$ is the passband cutoff frequency in rad/ sample, $\omega_s$ is the stopband cutoff frequency in rad/ sample, and $N$ is the length of the filter's impulse response. In this example, the filter amplitude response, $H_d(\omega)$, is designed to approximate $H(\omega)$ in a least squared optimality sense and satisfies the constraints

$$-1.5 \, \text{dB} \leq 20 \log_{10} |H_d(\omega)| \leq 1.5 \, \text{dB} \quad 0 \leq \omega \leq \omega_p$$

$$20 \log_{10} |H_d(\omega)| \leq 20 \, \text{dB} \quad \omega_s \leq \omega \leq \pi$$

$$|h_d(n)| \leq 32 \quad n = 0, 1, \ldots, N .$$

To satisfy the amplitude constraints of this filter design program, $N$ is found to be at least 22. Sampling at 500 equally spaced points in frequency, the design program can be expressed in the form of Equation 3.6

$$\text{minimize} \quad E(h) = h^T \mathbf{T} \mathbf{h}$$

subject to

$$-31 \leq h_d(n) \leq 32 \quad 4.1$$

$$h_d(n) \in I; \ n = 0, 1, \ldots, N$$

where
Solving the design program without the integer constraints produces the optimal solution vector, $h^*$, listed in Table 4.1.
Table 4.1 Optimal Impulse Response $h^*$

<table>
<thead>
<tr>
<th>$h^*(n)$</th>
<th>Value</th>
<th>$h^*(n)$</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h^*(0)$</td>
<td>-0.013379</td>
<td>$h^*(6)$</td>
<td>-0.061782</td>
</tr>
<tr>
<td>$h^*(1)$</td>
<td>-0.016106</td>
<td>$h^*(7)$</td>
<td>-0.054413</td>
</tr>
<tr>
<td>$h^*(2)$</td>
<td>0.002833</td>
<td>$h^*(8)$</td>
<td>0.048156</td>
</tr>
<tr>
<td>$h^*(3)$</td>
<td>0.028764</td>
<td>$h^*(9)$</td>
<td>0.209119</td>
</tr>
<tr>
<td>$h^*(4)$</td>
<td>0.028720</td>
<td>$h^*(10)$</td>
<td>0.331239</td>
</tr>
<tr>
<td>$h^*(5)$</td>
<td>-0.012405</td>
<td>$h^*(11)$</td>
<td></td>
</tr>
<tr>
<td>$h^*(16)$</td>
<td>-0.012405</td>
<td>$h^*(15)$</td>
<td>-0.061782</td>
</tr>
<tr>
<td>$h^*(17)$</td>
<td>-0.012405</td>
<td>$h^*(14)$</td>
<td>-0.054413</td>
</tr>
<tr>
<td>$h^*(18)$</td>
<td>0.028764</td>
<td>$h^*(13)$</td>
<td>0.048156</td>
</tr>
<tr>
<td>$h^*(19)$</td>
<td>0.002833</td>
<td>$h^*(12)$</td>
<td>0.209119</td>
</tr>
<tr>
<td>$h^*(20)$</td>
<td>-0.016106</td>
<td>$h^*(11)$</td>
<td>0.331239</td>
</tr>
</tbody>
</table>

Figure 4.1 shows the magnitude response for $h'$ given in Table 4.1, and Figure 4.2 shows the magnitude response of the passband in detail. For this $h'$, the optimal passband magnitude is 1, and the associated objective measure is approximately 0.1708.

After calculating the real solution, the integer first approximation is calculated. The reduction of the feasible solution space sets $b = 32$, and $c = 17$ (the first integer greater than $b/2$). Scaling and rounding $h'$ such that $h_{\text{max}} = 32$ produces $h_{\text{FA}} = [96.607, 1, -1, -2, 0, 3, 3, -1, -6, 5, 5, 20, 32]$ with an objective measure of 0.2454. Because this is the first integer solution vector, $h_{\text{FA}} = h_{\text{FA}}$, and $c_{\text{FA}} = 0.2454$.

The next step scales and rounds $h'$ such that $h_{\text{max}} = 31$ producing $h_{\text{FA}} = [94.488, -1, -2, 0, 3, 3, -1, -6, -5, 5, 20, 31]$ with an objective measure of 0.2725. Since this measure is greater than $c_{\text{FA}}$, the current integer solution and the current integer cost are not updated. This process of scaling and rounding $h'$ continues until $h'$ is scaled such that $h_{\text{max}} = 17$. The scaled...
Figure 4.1
Magnitude response \(20 \log_{10}(|H(\omega)/A|)\) for the designed filter with impulse response coefficients \(h\) listed in Table 4.1. The maximum stopband attenuation, -20 dB, occurs at 0.3 \(\pi\) radians/ sample.
Figure 4.2
Passband magnitude response \((20 \log_{10}\frac{|H(\omega)|}{A})\) for the designed filter with impulse response coefficients, \(h'\), listed in Table 4.1. The peak to peak passband ripple is 1 dB.
and rounded integer solution vectors and their objective measures are shown in Table 4.2. From Table 4.1 it can be seen that $h_{ipa}$ is first updated at $h_{max} = 27$ and again at $h_{max} = 25$ due to improving objective measures. At the end of solution process, $h_{ipa} = [76.115 -1 -1 0 2 2 -1 -5 -4 4 16 25]$ with $c_{ipa} = 0.2311$.

Table 4.2

Rounded Integer Solutions and their Costs

<table>
<thead>
<tr>
<th>Cost</th>
<th>$h_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2454</td>
<td>[96.401 -1 -2 0 3 3 -1 -6 -5 5 20 32]</td>
</tr>
<tr>
<td>0.2725</td>
<td>[94.488 -1 -2 0 3 3 -1 -6 -5 5 20 31]</td>
</tr>
<tr>
<td>0.2786</td>
<td>[90.667 -1 -1 0 3 3 -1 -6 -5 4 19 30]</td>
</tr>
<tr>
<td>0.2917</td>
<td>[86.931 -1 -1 0 3 3 -1 -5 -5 4 18 29]</td>
</tr>
<tr>
<td>0.2773</td>
<td>[84.819 -1 -1 0 2 2 -1 -5 -5 4 18 28]</td>
</tr>
<tr>
<td>0.2427</td>
<td>[81.378 -1 -1 0 2 2 -1 -5 -4 4 17 27]</td>
</tr>
<tr>
<td>0.2468</td>
<td>[78.053 -1 -1 0 2 2 -1 -5 -4 4 16 26]</td>
</tr>
<tr>
<td>0.2311</td>
<td>[76.115 -1 -1 0 2 2 -1 -5 -4 4 16 25]</td>
</tr>
<tr>
<td>0.2521</td>
<td>[71.784 -1 -1 0 2 2 -1 -4 -4 3 15 24]</td>
</tr>
<tr>
<td>0.2722</td>
<td>[69.864 -1 -1 0 2 2 -1 -4 -4 3 15 23]</td>
</tr>
<tr>
<td>0.2322</td>
<td>[66.518 -1 -1 0 2 2 -1 -4 -4 3 14 22]</td>
</tr>
<tr>
<td>0.2710</td>
<td>[63.112 -1 -1 0 2 2 -1 -4 -3 3 13 21]</td>
</tr>
<tr>
<td>0.2764</td>
<td>[61.185 -1 -1 0 2 2 -1 -4 -3 3 13 20]</td>
</tr>
<tr>
<td>0.3262</td>
<td>[58.875 -1 -1 0 2 2 -1 -4 -3 3 12 19]</td>
</tr>
<tr>
<td>0.4870</td>
<td>[54.189 -1 -1 0 2 2 -1 -3 -3 3 11 18]</td>
</tr>
<tr>
<td>0.5838</td>
<td>[51.516 -1 -1 0 1 1 -1 -3 -3 2 11 17]</td>
</tr>
</tbody>
</table>
After the integer first approximation is determined, the branch and bound algorithm can be employed to calculate the optimal integer solution vector for the filter design program. The branch and bound process must be carried out for sixteen major branches. Because this process involves many calculations and is of a repetitive nature, only part of the search on the 32 branches will be shown.

Each branch will be identified by the integer values leading to that branch. First, $h_{\text{ICO}} = h_{\text{IFA}}$ and $c_{\text{ICO}} = c_{\text{IFA}}$. The solution vector for the first level of the 32 branch is found by constraining $h_{\text{max}} = 32$ and solving Program 4.1 without the remaining integer constraints. The optimal solution vector produced is $h'_{32} = [-1.29 -1.56 0.27 2.78 2.77 -1.20 -5.97 -5.26 4.65 20.20 32]$ with an objective measure of 0.1708. Since this cost is less than $c_{\text{ICO}}$ and the solution is not composed entirely of integers, the branch must be explored further.

On the second level of the 32 branch, $h'_{32}(9)$ is the element which is fixed. Rounding up from the optimal vector creates the 32/21 branch which produces the optimal solution vector $[98.144 -1.41 -1.50 0.46 2.91 2.69 -1.43 -6.11 -5.16 -4.94 21 32]$ with a cost of 0.2082. Since this cost is less than $c_{\text{ICO}} = 0.2311$, the next branch up must be considered. Branch 32/22 produces $[100.181 -1.56 -1.42 0.07 3.08 2.56 -1.75 -6.32 -5.03 5.34 22 32]$ with a cost of 0.3530 which exceeds $c_{\text{ifa}}$, and branch 32/22 and all of the higher branches
on this level (32/23, 32/24, ... ) can be terminated. Rounding down from the optimal vector creates branch 32/20 which produces the optimal solution vector [96.180 -1.25 -1.58 0.20 2.73 2.81 -1.11 -5.91 -5.29 4.55 20 32] with a cost of 0.1733. The next lower branch, branch 32/19, must be considered. It produces the optimal solution vector [94.295 -1.10 -1.66 -0.06 2.56 2.93 -0.79 -5.71 -5.43 4.15 19 32] with a cost of 0.2628. Branch 32/19 and all lower branches (32/18, 32/17, ...) can be terminated because their objective measures exceed $c_{\infty}$. 

Pursuing the active 32/21 branch, $h'(6)$ becomes the new branching point on level three. Branching up leads to the 32/21/-5 branch which produces [97.432 -1.26 -1.53 0.27 2.75 2.76 -1.20 -5 -5.28 4.67 21 32] with a cost of 0.3272. Branch 32/21/-5 and all higher branches can be terminated. Branching down leads to the 32/21/-6 branch which produces [98.062 -1.39 -1.50 0.44 2.89 2.69 -1.41 -6 -5.17 4.91 21 32] with a cost of 0.2097. Searching the next lower branch, the 32/21/-7 branch produces [98.812 -1.53 -1.48 0.62 3.03 2.63 -1.61 -7 -5.06 21 32] with a cost of 0.2759. Branch 32/21/-7 and all lower branches can be terminated.

$h'(7)$ is the branching point element on level four. Rounding down produces branch 32/21/-6/-5 which produces the optimal solution vector [98.118 -1.41 -1.48 0.49 2.91 2.66 -1.46 -6 -5 4.97 21 32] with a cost of 0.2118. Branch 32/21/-6/-4 is searched next producing [98.500 -1.51 -1.37 0.75 3.00
2.46 -1.76 -6 -4 5.29 21 32 ] with a cost of 0.3156 which results in the termination of this branch and all higher branches. Branch 32/21/-6/-6 is searched producing [ 97.838 -1.31 -1.62 0.23 2.81 2.86 -1.16 -6 -6 4.65 21 32 ] with a cost of 0.2654 which terminates this branch and all lower branches.

The only viable branch is the 32/21/-6/-5 branch, and element h'(8) of this branch is the branching point for level five. Branch 32/21/-6/-5/5 produces [98.142 -1.41 -1.47 0.50 2.91 2.65 -1.47 -6 -5 5 21 32 ] with a cost of 0.2171 requiring the exploration of branch 32/21/-6/-5/6 which produces [ 99.017 -1.55 -1.37 0.76 3.05 2.49 -1.79 -6 -5 6 21 32 ] with a cost of 0.2778. Branching down leads to branch 32/21/-6/-5/4 which produces [ 97.355 -1.27 -1.58 0.23 2.77 2.82 -1.15 -6 -5 4 21 32 ] with a cost of 0.2792.

Branching from the fifth element of the sole viable branch on level six, the 32/21/-6/-5/5 branch, leads to branch 32/21/-6/-5/5/3 which produces [ 98.106 -1.41 -1.51 0.45 2.91 3 -1.42 -6 -5 5 21 32 ] with a cost of 0.2229, and branch 32/21/-6/-5/5/2 which produces [ 98.250 -1.42 -1.40 0.58 2.92 2 -1.56 -6 -5 5 21 32 ] with a cost of 0.2504. The higher branch requires further exploration leading to branch 32/21/-6/-5/5/4 which produces [ 98.081 -1.41 -1.63 0.33 2.90 4 -1.29 -6 -5 5 21 32 ] with a cost of 0.3783.

Pursuing the only viable branch to level seven and setting the branch point at the fourth element leads to branch 32/21/-6/-5/5/3/3, which produces the optimal solution vector

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
[ 98.133 -1.42 -1.52 0.46 3 3 -1.43 -6 -5 5 21 32 ] with a cost of 0.2236. Branch 32/21/-6/-5/5/3/2 produces [ 97.904 -1.31 -1 48 0.37 2 3 -1.32 -6 -5 5 21 32 ] with a cost of 0.3007. Exploring the branch above 32/21/-6/-5/5/3/3 leads to branch 32/21/-6/-5/5/3/4 which produces [ 98.482 -1 54 -1.55 0.55 4 3 -1.55 -6 -5 5 21 32 ] with a cost of 0.3318.

Starting a new branch at the second element of the 32/21/-6/-5/5/3/3 branch leads to branch 32/21/-6/-5/5/3/3/-1 which produces [ 98.222 -1.41 -1 0.50 3 3 -1.48 -6 -5 5 21 32 ] with a cost of 0.2481 and branch 32/21/-6/-5/5/3/3/-2 which produces [ 98.078 -1.44 -2 0.43 3 3 -1.40 -6 -5 5 21 32 ] with a cost of 0.2457. Both of the costs associated with these last two branches are greater than the current integer cost $C_{\text{int}}$ which effectively terminates the search of the 32/21 branch. The graphical representation of the 32/21 branch search is shown in Figure 4.3.

To complete the search of the 32 branch, the 32/20 branch must also be searched until all levels and all branches are terminated. The search of the 32/20 branch parallels the process used in the search of the 32/21 branch and is not shown here. All of the other branches on this level were terminated.

Once the 32 branch is completed, branch 31 must be searched, and then branch 30, and so on down to branch 17. If at any time an integer solution is produced with an associated cost less than $C_{\text{int}}$, then $C_{\text{int}}$ is set to the new cost and $h_{\text{rco}}$.
The branch diagram of the search for the optimal integer vector solution on the 32 branch. X denotes a terminated branch, ? denotes an unexplored branch.

Figure 4.3 Branch diagram of the search for the optimal integer vector solution on the 32 branch. X denotes a terminated branch, ? denotes an unexplored branch.
is set to the new integer solution. After the completion of
the search of branch 17, \( \mathbf{h}_{\text{co}} \) will contain the optimal integer
solution vector to the filter design program's optimality
criteria.

The optimal integer vector solution, \( \mathbf{h}_r \), for this example
is shown in Table 4.3. \( A \) is approximately 76.115, and the
resultant filter cost is approximately 0.2311. The
corresponding magnitude response and passband magnitude
response for the filter described by the coefficients in Table
4.3 are shown in Figures 4.4 and 4.5. This filter is the
optimal integer filter for the specified design criteria
applied to a weighted least squared error measure. The filter

\[ \begin{align*}
\mathbf{h}_r(0) &= \mathbf{h}_r(21) = -1 & \mathbf{h}_r(6) &= \mathbf{h}_r(15) = -5 \\
\mathbf{h}_r(1) &= \mathbf{h}_r(20) = -1 & \mathbf{h}_r(7) &= \mathbf{h}_r(14) = -4 \\
\mathbf{h}_r(2) &= \mathbf{h}_r(19) = 0 & \mathbf{h}_r(8) &= \mathbf{h}_r(13) = 4 \\
\mathbf{h}_r(3) &= \mathbf{h}_r(18) = 2 & \mathbf{h}_r(9) &= \mathbf{h}_r(12) = 16 \\
\mathbf{h}_r(4) &= \mathbf{h}_r(17) = 2 & \mathbf{h}_r(10) &= \mathbf{h}_r(11) = 25 \\
\mathbf{h}_r(5) &= \mathbf{h}_r(16) = -1
\end{align*} \]
Figure 4.4
Magnitude response ($20 \log_{10}(|H(\omega)/A|)$) for the designed filter with impulse response coefficients $h_n$ shown in Table 4.3. The maximum stopband attenuation, $-21.0$ dB, occurs at $\omega_s = 0.4 \pi$ radians/sample.
Figure 4.5
Passband magnitude response (20log₁₀(H(ω)/A)) for the designed filter with impulse response coefficients, hₜ, shown in Table 4.3. The passband peak to peak ripple is 1.27 dB.
magnitude response passes through the stopband frequency cutoff, $\omega_s$, at a value of approximately -21.0 dB.

To improve the stopband suppression characteristics of this filter, the amplitude response optimization criteria described in Chapter 3 can be employed. Creating the constraint

$$H(0.4\pi) = -40 \text{ dB},$$

requires subtraction of the term

$$\lambda \left( -0.01 A + \sum_{n=0}^{10} h_n \cos((21/2 - n)\omega) \right)$$

from the optimality criteria $E(h_n)$. The choice of the frequency constraint used in the optimality criteria is highly variable, and as a result, the integer first approximation for this design program returns an appropriate integer solution to the constrained problem. Solving the resultant optimization program created by the difference of optimality criteria in Program 4.1 and the Lagrangian expressed in 4.2 returns the optimal integer impulse response, $h_n$, listed in Table 4.4.
The corresponding magnitude response and passband magnitude response are shown in Figures 4.6 and 4.7. The optimal passband magnitude is approximately 89.792 and the associated cost is 0.3231.

The resultant magnitude response shows almost equal stopband gain with the peak magnitude response occurring at \( \omega = 0.4325 \pi \) radians/sample with a value of \(-28.6\) dB. This one iteration of the amplitude constraint technique has achieved an additional 7.5 dB (37.5%) of suppression in the stopband while the passband gain has deteriorated by only an additional 0.25 dB. The passband amplitude responses are shown in Figure 4.8 for the preceding three filter designs.

| \( h_{II}(0) = h_{II}(21) = -2 \) | \( h_{II}(6) = h_{II}(25) = -6 \) |
| \( h_{II}(1) = h_{II}(20) = -1 \) | \( h_{II}(7) = h_{II}(14) = -4 \) |
| \( h_{II}(2) = h_{II}(19) = 1 \) | \( h_{II}(8) = h_{II}(13) = 5 \) |
| \( h_{II}(3) = h_{II}(18) = 3 \) | \( h_{II}(9) = h_{II}(12) = 19 \) |
| \( h_{II}(4) = h_{II}(17) = 2 \) | \( h_{II}(10) = h_{II}(11) = 29 \) |
| \( h_{II}(5) = h_{II}(16) = -2 \) |
Figure 4.6
Magnitude response (\(20 \log_{10}(H(\omega)/A)\)) for the designed filter with impulse response coefficients, \(h_\omega\), shown in Table 4.3. The maximum stopband attenuation, \(-28.6\) dB, occurs at \(\omega = 0.4325\pi\) radians/sample.
Figure 4.7
Passband magnitude response ($20 \log_{10}(|H(\omega)/A|)$) for the designed filter with impulse response coefficients, $h_\cdot$, shown in Table 4.4. The passband peak to peak ripple is 1.52 dB.
Passband magnitude responses (20\log_{10}(H(\omega)/A)) of the three designed filters. \(h\)' is plotted with 'o's and has a passband peak to peak ripple of 1 dB, \(h_r\) is plotted with 'x's and has a passband peak to peak ripple of 1.27 dB, \(h_r\) is plotted with '+'s and has a passband peak to peak ripple of 1.52 dB.
4.2 Finite Wordlength Application

For finite wordlength applications, the design problem is treated as a multiplierless filter design problem, and the integer solution is scaled by the appropriate factor to create the finite wordlength solution. Consider a linear phase FIR discrete time filter that has seven bit signed finite wordlength and that approximates the frequency response,

\[ H(e^{-j\omega}) = e^{-j\omega \cdot \frac{H-1}{2}} H(\omega), \]

where

\[ H(\omega) = \begin{cases} A & 0 \leq \omega \leq \omega_p = 0.24 \pi \\ 0 & 0.37 \pi = \omega_s \leq \omega \leq \pi \end{cases} \]

A is a constant greater than 1, \( \omega_p \) is the passband cutoff frequency in rad/sample, \( \omega_s \) is the stopband cutoff frequency in rad/sample, and \( N \) is the length of the filter's impulse response. In this example the filter, \( H_d(\omega) \), is designed to approximate \( H(\omega) \) in a least squared error measure and satisfy the criteria

\[-0.5 \text{ dB} \leq 20 \log |H_d(\omega)| \leq 0.5 \text{ dB} \quad 0 \leq \omega \leq \omega_p \]

\[ 20 \log |H_d(\omega)| \leq 28 \text{ dB} \quad \omega_s \leq \omega \leq \pi \]

\[ |h_d(n)| \leq 64 \]
The passband and stopband criteria require that the designed filter have an order of \( N = 28 \). The weighted least squared error filter design technique is applied resulting in the real optimal vector solution, \( h' \), listed in Table 4.5.

\[
\begin{align*}
 h'(0) &= h'(27) = 0.001595 & h'(7) &= h'(20) = -0.002286 \\
 h'(1) &= h'(26) = -0.003985 & h'(8) &= h'(19) = -0.040109 \\
 h'(2) &= h'(25) = -0.009716 & h'(9) &= h'(18) = -0.056696 \\
 h'(3) &= h'(24) = -0.007613 & h'(10) &= h'(17) = -0.017306 \\
 h'(4) &= h'(23) = 0.005664 & h'(11) &= h'(16) = 0.083175 \\
 h'(5) &= h'(22) = 0.021416 & h'(12) &= h'(15) = 0.207176 \\
 h'(6) &= h'(21) = 0.022275 & h'(13) &= h'(14) = 0.292678
\end{align*}
\]

The magnitude response of \( h' \) is plotted in Figure 4.9. Figure 4.10 shows the magnitude response of the passband. The optimal passband magnitude, \( A \), is 1 and the associated cost is approximately 0.0228.

The branch and bounding technique gives the optimal integer impulse response, \( h_r \), listed in Table 4.6. The finite wordlength solution is found by dividing \( h_r \) by 64 and is shown in Table 4.7. The corresponding magnitude response and passband magnitude response are shown in Figures 4.11 and 4.12. \( A \) is approximately 3.234, and the resultant filter cost is approximately 0.0365. This filter is the optimal
Figure 4.9
Magnitude response (20\log_{10}(H(\omega)/A)) for the designed filter with impulse response coefficients, h', shown in Table 4.5.
The maximum stopband attenuation, -28.1 dB, occurs at \omega_s = 0.37 \pi radians/sample.
Figure 4.10
Passband magnitude response ($20 \log_2 (|H(\omega)/A|)$) for the designed filter with impulse response coefficients, $h'$, shown in Table 4.5. The passband peak to peak ripple is 0.51 dB.
Figure 4.11
Magnitude response ($20\log_{10}(H(\omega)/A)$) for the designed filter with impulse response coefficients, $h_r$, shown in Table 4.7. The maximum stopband attenuation, $-25.5$ dB, occurs at $\omega_s = 0.37 \pi$ radians/sample.
Figure 4.12
Passband magnitude response \(20\log_{10}(H(\omega)/A)\) for the designed filter with impulse response coefficients, \(h_r\), shown in Table 4.7. The passband peak to peak ripple is 0.43 dB.
filter for the specified design criteria applied to a weighted least squared optimality criteria.

<table>
<thead>
<tr>
<th>h_I(0) = h_I(27) = 0</th>
<th>h_I(7) = h_I(20) = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>h_I(1) = h_I(26) = -1</td>
<td>h_I(8) = h_I(19) = -8</td>
</tr>
<tr>
<td>h_I(2) = h_I(25) = -2</td>
<td>h_I(9) = h_I(18) = -12</td>
</tr>
<tr>
<td>h_I(3) = h_I(24) = -2</td>
<td>h_I(10) = h_I(17) = -4</td>
</tr>
<tr>
<td>h_I(4) = h_I(23) = 1</td>
<td>h_I(11) = h_I(16) = 17</td>
</tr>
<tr>
<td>h_I(5) = h_I(22) = 4</td>
<td>h_I(12) = h_I(15) = 43</td>
</tr>
<tr>
<td>h_I(6) = h_I(21) = 5</td>
<td>h_I(13) = h_I(14) = 61</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>h_F(0) = h_F(27) = 0</th>
<th>h_F(7) = h_F(20) = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>h_F(1) = h_F(26) = -0.015625</td>
<td>h_F(8) = h_F(19) = -0.125</td>
</tr>
<tr>
<td>h_F(2) = h_F(25) = -0.03125</td>
<td>h_F(9) = h_F(18) = -0.1875</td>
</tr>
<tr>
<td>h_F(3) = h_F(24) = -0.03125</td>
<td>h_F(10) = h_F(17) = -0.0625</td>
</tr>
<tr>
<td>h_F(4) = h_F(23) = 0.015625</td>
<td>h_F(11) = h_F(16) = 0.265625</td>
</tr>
<tr>
<td>h_F(5) = h_F(22) = 0.0625</td>
<td>h_F(12) = h_F(15) = 0.671875</td>
</tr>
<tr>
<td>h_F(6) = h_F(21) = 0.078125</td>
<td>h_F(13) = h_F(14) = 0.953125</td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
In Figure 4.11, the first sidelobe has a magnitude of approximately -35 dB, but the filter magnitude response passes through the stopband frequency cutoff, \( \omega_s \), at a value of approximately -25.5 dB. To improve the stopband suppression characteristics of this filter, the constraint

\[
H(0.37\pi) = -40 \text{ dB},
\]

can be added to the filter design program. Solving the resultant program produces the optimal fixed length impulse response, \( h_{p1} \), listed in Table 4.8. Figures 4.13 and 4.14 show the corresponding frequency magnitude response and passband frequency magnitude response. The optimal passband magnitude is approximately 3.125 and the associated cost is 0.0445.

**Table 4.8 Impulse Response \( h_{p1} \)**

<table>
<thead>
<tr>
<th>( h_{p1} ) index</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-0.015625</td>
</tr>
<tr>
<td>2</td>
<td>-0.03125</td>
</tr>
<tr>
<td>3</td>
<td>-0.015625</td>
</tr>
<tr>
<td>4</td>
<td>0.03125</td>
</tr>
<tr>
<td>5</td>
<td>-0.015625</td>
</tr>
<tr>
<td>6</td>
<td>0.0625</td>
</tr>
<tr>
<td>7</td>
<td>-0.03125</td>
</tr>
<tr>
<td>8</td>
<td>-0.015625</td>
</tr>
<tr>
<td>9</td>
<td>-0.03125</td>
</tr>
<tr>
<td>10</td>
<td>-0.03125</td>
</tr>
<tr>
<td>11</td>
<td>0.03125</td>
</tr>
<tr>
<td>12</td>
<td>0.078125</td>
</tr>
<tr>
<td>13</td>
<td>0.0625</td>
</tr>
<tr>
<td>14</td>
<td>0.03125</td>
</tr>
<tr>
<td>15</td>
<td>0.078125</td>
</tr>
<tr>
<td>16</td>
<td>0.03125</td>
</tr>
<tr>
<td>17</td>
<td>0.0625</td>
</tr>
<tr>
<td>18</td>
<td>0.078125</td>
</tr>
<tr>
<td>19</td>
<td>0.03125</td>
</tr>
<tr>
<td>20</td>
<td>-0.03125</td>
</tr>
<tr>
<td>21</td>
<td>0.0625</td>
</tr>
<tr>
<td>22</td>
<td>0.078125</td>
</tr>
<tr>
<td>23</td>
<td>0.03125</td>
</tr>
<tr>
<td>24</td>
<td>-0.015625</td>
</tr>
<tr>
<td>25</td>
<td>-0.03125</td>
</tr>
<tr>
<td>26</td>
<td>-0.015625</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
</tr>
</tbody>
</table>
Figure 4.13
Magnitude response ($20\log_{10}(|H(\omega)/A|)$) for the designed filter with impulse response coefficients, $h_{\text{pl}}$, shown in Table 4.8. The maximum stopband attenuation is approximately $-31.3$ dB.
Figure 4.14
Passband magnitude response \(20\log_{10}(|H(\omega)/A|)\) for the designed filter with impulse response coefficients, \(h_p\), shown in table 4.8. The passband peak to peak ripple is approximately 0.78 dB.
While the magnitude response has dropped to approximately $-50 \text{ dB}$ at $\omega_s$, the first sidelobe has risen to a magnitude of approximately $-31.3 \text{ dB}$. If a new program is created with the constraints

$$H(0.37\pi) = -40 \text{ dB}$$
$$H(0.4\pi) = -40 \text{ dB},$$

the optimal fixed length impulse response, $h_{r2}$, is produced and is listed in Table 4.9. Figures 4.15 and 4.16 show the corresponding magnitude response and passband magnitude response. $A$ is approximately 3.344 with an associated cost of approximately 0.0512.

### Table 4.9 Impulse Response $h_{r2}$

<table>
<thead>
<tr>
<th>$h_{r2}(0)$</th>
<th>$h_{r2}(27)$</th>
<th>$h_{r2}(7)$</th>
<th>$h_{r2}(20)$</th>
<th>$-0.015625$</th>
<th>$-0.015625$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_{r2}(1)$</td>
<td>$h_{r2}(26)$</td>
<td>$h_{r2}(8)$</td>
<td>$h_{r2}(19)$</td>
<td>$-0.03125$</td>
<td>$-0.15625$</td>
</tr>
<tr>
<td>$h_{r2}(2)$</td>
<td>$h_{r2}(25)$</td>
<td>$h_{r2}(9)$</td>
<td>$h_{r2}(18)$</td>
<td>$-0.03125$</td>
<td>$-0.171875$</td>
</tr>
<tr>
<td>$h_{r2}(3)$</td>
<td>$h_{r2}(24)$</td>
<td>$0$</td>
<td>$h_{r2}(10)$</td>
<td>$h_{r2}(17)$</td>
<td>$-0.015625$</td>
</tr>
<tr>
<td>$h_{r2}(4)$</td>
<td>$h_{r2}(23)$</td>
<td>$0.046875$</td>
<td>$h_{r2}(11)$</td>
<td>$h_{r2}(16)$</td>
<td>$0.3125$</td>
</tr>
<tr>
<td>$h_{r2}(5)$</td>
<td>$h_{r2}(22)$</td>
<td>$0.078125$</td>
<td>$h_{r2}(12)$</td>
<td>$h_{r2}(15)$</td>
<td>$0.6875$</td>
</tr>
<tr>
<td>$h_{r2}(6)$</td>
<td>$h_{r2}(21)$</td>
<td>$0.046875$</td>
<td>$h_{r2}(13)$</td>
<td>$h_{r2}(14)$</td>
<td>$0.9375$</td>
</tr>
</tbody>
</table>
Figure 4.15
Magnitude response \((20 \log_2 (H(\omega)/A))\) for the designed filter with impulse response coefficients, \(h_{r2}\), shown in Table 4.9. The maximum stopband attenuation is approximately \(-40.0\) dB.
Figure 4.16
Passband magnitude response \(20\log_{10}(|H(\omega)/A|)\) for the designed filter with impulse response coefficients, \(h_{f_2}\), shown in table 4.9. The passband peak to peak ripple is approximately 1.05 dB.
The marked improvement in the stopbands of the filters can be seen at each iteration of the amplitude constraint design technique. The magnitude response of Figure 4.15 shows a major sidelobes with a magnitude of approximately -43 dB and at \( \omega_s \), the magnitude is approximately -40 dB. Comparing the magnitude response of \( h_1 \) and \( h_2 \), it can be seen that an additional 15 dB of stopband suppression has been achieved with minimal (\(< 0.5 \text{ dB}\)) deterioration in the passband (Figure 4.17). This 15 dB gain was realized using the technique described in Chapter 3.5 in an attempt to equalize the stopband energy.
Figure 4.17
Passband magnitude responses ($20\log_{10}(H(\omega)/A)$) of the four designed filters. $h'$ is plotted with '*'s and has a passband peak to peak ripple of 0.51 dB, $h_r$ is plotted with '+'s and has a passband peak to peak ripple of 0.43 dB, $h_r^+$ is plotted with 'x' s and has a passband peak to peak ripple of 0.78 dB, $h_r^-$ is plotted with 'o' s and has a passband peak to peak ripple of 1.05 dB.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
CHAPTER 5

CONCLUSION

This thesis introduced two optimal design techniques for multiplierless discrete time FIR filters with constant group delay. The first technique solved a weighted least squared error criteria. The second technique solved an amplitude constrained weighted least squared error criteria. The amplitude constrained optimality design technique has the capability to design a wide range of filters. A filter that is produced using a weighted least squared error criteria can be achieved by applying no amplitude constraints in the amplitude constrained optimality criteria. By imposing multiple amplitude constraints, the amplitude constrained design technique can produce an equiripple filter. Both of the design techniques return an optimal integer valued impulse response which can be used in transport processor applications, or the optimal integer solution vector can be scaled for fixed wordlength coefficient applications. Examples are provided showing the design technique used for a multiplierless filter application and for a finite precision coefficient filter application. Both examples produce the optimal filter solutions to the design problems using a
weighted least squared error criteria. Subsequent optimal solutions are produced using the amplitude constraint optimality criteria.
BIBLIOGRAPHY


VITA

Graduate College
University of Nevada, Las Vegas

Torrance M. Lawton

Home Address:
7073 Grasswood Dr.
Las Vegas, NV. 89117

Degrees:
Bachelor of Science, Electrical Engineering, 1989
California Institute of Technology

Thesis Title:
Two Optimization Techniques for Designing
Multiplierless FIR Filters

Thesis Examination Committee:
Chairperson, Dr. Peter Stubberud, PhD.
Committee Member, Dr. William Brogan, PhD.
Committee Member, Dr. Lori Bruce, PhD.
Graduate Faculty representative, Dr. Georg Mauer, PhD.