QMF FILTER BANK DESIGN USING NONLINEAR OPTIMIZATION

BY

TING YU

B.E., Tsinghua University, 1991
M.E., Tsinghua University, 1994

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1997

Urbana, Illinois

Abstract

This research is about QMF filter bank design using nonlinear optimization methods. Filter bank has applications in areas like video and audio coding, data communication, etc. Its improvement will have important impacts in those areas.

Filter bank design is a multi-objective optimization problem. One important issue in this research is that the design objectives are not unique and are conflicting, leading to designs with different tradeoffs. Here, we define reconstruction error as the objective function, and set stopband energy, passband energy, stopband ripple, passband ripple, and transition bandwidth as constraints. Therefore, filter bank design can be formulated as a single-objective multiple-constraint optimization problem. As both the objective function and constraints are nonlinear, methods for solving nonlinear constrained optimization problems are applied.

Because reconstruction error, stopband energy, and passband energy are expressed as integrations, in order to keep precision and to save computation time, closed-form formulae are derived for both functions and their derivatives.

The difficulty in solving a nonlinear optimization problem is to find the global solution. The Novel method, a new nonlinear optimization package, is applied here. A heuristic function is implemented in Novel, which can lead the search trajectory to explore the whole problem space. Two methods for handling inequality constraints, the Maxq and slack-variable methods, are compared in this research.

Experimental results are presented in the end. Issues like the impacts of constraint formulation, the weight on the objective function, etc. are discussed. We compare the performance of our designs with the best known solutions, and show the improvements of our designs.

Acknowledgments

I would like to thank my advisor, Professor Benjamin Wah, for choosing this interesting research topic. Professor Wah spent a lot of time discussing the research progress with me. Without his advice, it would have been impossible for me to finish this thesis.

I thank my colleagues, Yi Shang and Tao Wang, for their wonderful work on the Novel optimization package and the Maxq method. Their work makes it possible for me to further apply them to the filter bank design problem. I also thank all the group members, who have given me lots of suggestions and comments on this research.

I want to thank my husband, Ye Zhang, for his encouragement and support. This thesis is dedicated to him.

I would like to thank the National Science Foundation. This research is under the support of grant MIP96-32316.

Table of Contents

Chapter

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivations

Filter bank design has become an important research topic, because its improvement can have significant impact in many engineering fields. For example, filter banks have been applied in image and video coding, speech and audio coding, communication, data transmission, and digital audio broadcasting. Filter bank design can be formulated as a nonlinear optimization problem, and optimization techniques developed for filter bank design problems can be generalized to other problems as well.

1.1.1 Filter Bank Applications

Basically, filter banks are used to differentiate the information in different frequency bands. It is necessary in situations in which different processing procedures are applied according to a signal's frequency. One example is the quantization tables used in JPEG. JPEG is an image compression standard. Images encoded in JPEG format have much less storage than the original ones, so bandwidth, transmission time, and storage could be saved. In JPEG encoding, a still image is first partitioned into blocks, and in each block there are 8*8 Luminance components and 16*8 Chrominance components which are subsampled by two in horizontal frequency. Each block is then applied with an 8*8 DCT transform to remove the correlation among pixel values. After the DCT transform, the image is converted from spatial domain to frequency domain, and

each DCT coefficient is quantized by the nearest quantization value and coded using Huffman coding or Arithmetic coding.

Table 1.1: JPEG Luminance and Chrominance quantization tables. Here the horizontal and vertical directions denote horizontal and vertical frequencies, respectively

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

(a) Luminance

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

(b)Chrominance

Studies show that Human Visual System (HVS) response is highly dependent on spatial frequency. We know human eyes are more sensitive to low frequencies than to high frequencies. Better compression can be achieved by exploiting this feature. We need to decompose the image into different frequency space, and apply appropriate processing to the image structure that eyes can see and the structure that eyes are insensitive to. Hence, the quantization values used in JPEG, which are shown in Table 1.1, are visually weighted and frequency-dependent [37]. It is clear from Table 1.1 that quantization values of high frequencies are much larger than those of low frequencies, as human eyes are insensitive to high frequencies.

Another simple example is given below to show the application of filter banks. Suppose an input signal $x(n)$ has frequency of 10K Hz. According to the Nyquist Theorem, the minimum sampling rate should be 20,000 samples per second. If each sample is encoded by 8 bits, then the minimal bit rate will be 160K bits per second. Suppose the energy distribution of the given signal is like Fig. 1.1.

Clearly there is energy in high frequency. It is small compared to that of the low frequency band, but not small enough to be ignored completely. In this case the direct use of *decimator* will result in unacceptable information loss. Using filter banks can avoid information loss, but achieve compression at the same time.

A 2-channel filter bank is shown in Fig. 1.2. There are four filters involved: $H_0(z)$, $H_1(z)$, $G_0(z)$, and $G_1(z)$. $H_0(z)$ and $G_0(z)$ in the upper channel are low-pass (LP) filters, and $H_1(z)$

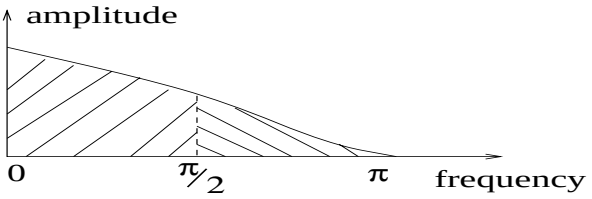

Figure 1.1: Frequency response of a signal whose major information is located in the low frequency band

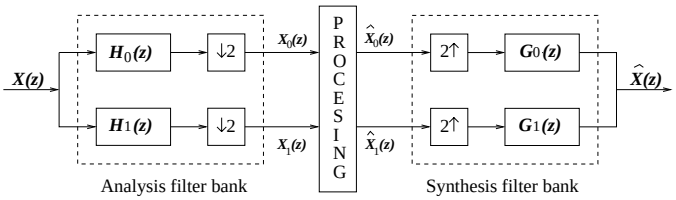

Figure 1.2: Structure of a 2-channel filter bank

and $G_1(z)$ in the lower channel are high-pass (HP) filters. Two decimators and interpolators are used for subsampling and oversampling. The overall filter bank can be partitioned into two components: analysis filter bank and synthesis filter bank. As illustrated in the figure, the analysis filter bank is composed of $H_0(z)$ and $H_1(z)$ plus two decimators, while the synthesis filter bank consists of two interpolators and filters $G_0(z)$ and $G_1(z)$. When an incoming signal passes through the analysis filter bank, it is decomposed into low and high subchannels, and applied with the desired processing. The reconstructed signal $\hat{x}(z)$ is obtained after it passes through the synthesis filter bank.

The low frequency signal $x_0(n)$ passes through the upper channel, while the high frequency signal $x_1(n)$ goes through the lower channel. They can be defined as:

$$\text{upper channel:} \quad x_0(n) = \mathcal{F}^{-1}\{\tfrac{1}{2}(X(\omega)H_0(\omega) + \tfrac{1}{2}X(\Leftrightarrow\omega)H_0(\Leftrightarrow\omega)\}$$

$$\text{lower channel:} \quad x_1(n) = \mathcal{F}^{-1}\{\tfrac{1}{2}(X(\omega)H_1(\omega) + \tfrac{1}{2}X(\Leftrightarrow\omega)H_1(\Leftrightarrow\omega)\}$$

where $X(\omega)$ is the Fourier transform of the input signal, and $\mathcal{F}^{-1}$ stands for inverse Fourier transform. From our assumption of the input signal information distribution, we know $x_1(n)$ has less energy than $x_0(n)$, so we can achieve compression by allocating less bits to the high frequency signal $x_1(n)$. For instance, we use 4 bits to encode $x_1(n)$. In the upper channel (low frequency subband), we still use 8 bits to encode $x_0(n)$. Note that both upper channel and

lower channel frequencies are half of the input signal frequency, i.e., the sampling rate of each channel becomes 10K samples per second. The resulting bit rate becomes (8 bits/sample + 4 bits/sample) * 10K samples/second = 120K bits/second. So the compression ratio is $\frac{160K}{120K} = \frac{4}{3}$.

1.2 Introduction to Filter Bank Design

1.2.1 Functionality and structure

The functionality of a filter bank, in short, is to decompose the frequency band of an input signal into multiple subbands, so that different processing procedures can be applied to the subband signals. As shown in Fig. 1.2, a 2-channel analysis filter bank splits the frequency space into two parts, with the low frequency band ranging from 0 to $\frac{\pi}{2}$ and the high frequency band from $\frac{\pi}{2}$ to π.

Band splitting can be done recursively, which may result in unequal segmentation of frequency space. Fig 1.3 shows an analysis filter bank and its resulting frequency segmentation.

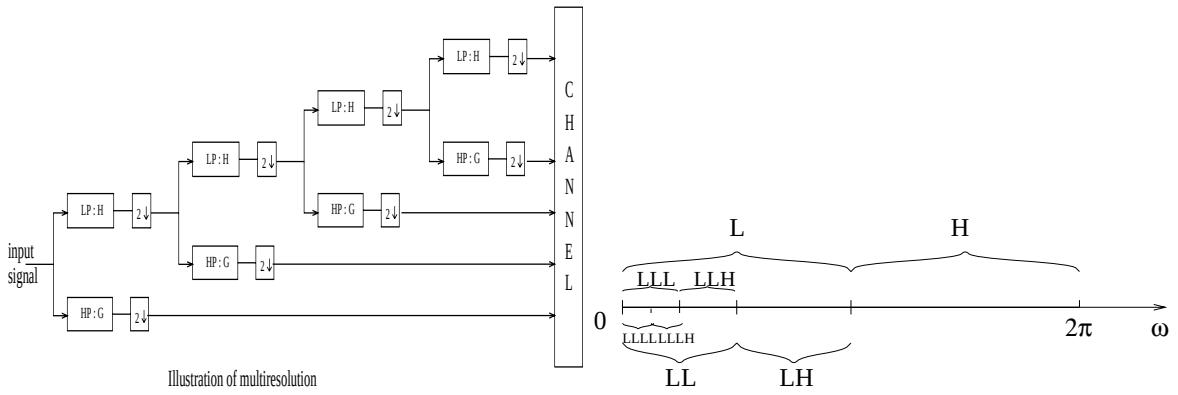

Figure 1.3: Recursive band splitting and its resulting unequal segmentation of frequency space. Here, L denotes the low frequency subband; H, the high frequency subband; LL, the lower segment of the low frequency band; LH, the higher part of the low frequency band; and so on

The above frequency segmentation results in much higher resolution for the low frequency band than that for the high frequency band. Keep in mind that human eyes are more sensitive to low frequencies.

Generally a multiple channel filter bank may have nonuniform width for each channel. The 2-channel filter bank shown in Fig. 1.2 is a special case.

Time domain analysis of 2-channel filter banks. Let's analyze the time domain data sequences of a trivial filter bank. Here, $h_0(n) = \{1, 1\}$, $h_1(n) = \{1, \Leftrightarrow 1\}$, $g_0(n) = \{1, 1\}$, and $g_1(n) = \{\Leftrightarrow 1, 1\}$. Table 1.2 shows its data sequences.

Table 1.2: Data sequences in time domain

Input signal	$x(n)$	$\cdots, x4, x3, x2, x1, x0$
Upper Channel	$x_0(n)$	$\cdots, x4 + x3, x3 + x2, x2 + x1, x1 + x0$
	$x_0'(n)$	$\cdots, x3 + x2, x1 + x0$
	$\hat{x}_0'(n)$	$\cdots, 0, x3 + x2, 0, x1 + x0$
	$\hat{x}_0(n)$	$\cdots, x3 + x2, x3 + x2, x1 + x0, x1 + x0$
Lower Channel	$x_1(n)$	$\cdots, \Leftrightarrow x4 + x3, \Leftrightarrow x3 + x2, \Leftrightarrow x2 + x1, \Leftrightarrow x1 + x0$
	$x_1'(n)$	$\cdots, \Leftrightarrow x3 + x2, \Leftrightarrow x1 + x0$
	$\hat{x}_1'(n)$	$\cdots, 0, \Leftrightarrow x3 + x2, 0, \Leftrightarrow x1 + x0$
	$\hat{x}_0(n)$	$\cdots, \Leftrightarrow x3 + x2, \Leftrightarrow x3 + x2, \Leftrightarrow x1 + x0, \Leftrightarrow x1 + x0$
Reconstructed Signal	$\hat{x}(n)$	$\cdots, x4, x3, x2, x1, x0$

Here, x followed by a number m represents the signal value at time m. Table 1.2 indicates that the reconstructed signal $\hat{x}(n)$ is exactly the same as the input signal. This phenomenon is called *Perfect Reconstruction* (PR), which is a desirable property for filter-bank design. If the composing filters are FIR filters, only in this trivial case when all the filters are 2-tap filters can PR be achieved. Here, $h_0(n)$ and $g_0(n)$ actually take the average, while $h_1(n)$ and $g_1(n)$ take the difference of inputs. We will explain this concept later.

Classification. The classification of the filter banks depends on the partition criteria. Table 1.3 shows some instances:

Table 1.3: Filter bank classification criteria

Criterion	**Category**
composing filters:	FIR or IIR
number of subchannels:	2-channel or multichannel
filter bank structure:	QMF, Non-QMF, Cosine-modulated, lattice ...

Each criterion given above classifies filter banks from one aspect. The combination of those criteria will locate a filter bank's category more precisely. For instance, our research focuses on 2-channel QMF FIR filter-bank design. That means our target design is a 2-channel filter

bank, so the input signal is decomposed into low frequency and high frequency channels only. The high-pass filter is symmetric to the low-pass filter with respect to $\frac{\pi}{2}$, i.e., the high-pass filter is the QMF filter of the low-pass filter; and the composing filters are FIR filters. We will briefly introduce some categories of filter banks in Section 1.2.3.

1.2.2 Performance metrics

As shown in Table 1.4, the performance metrics of filter banks belong to two categories: the overall filter-bank response and the response of each individual filter.

Table 1.4: Possible design objectives of filter banks

Category	Performance metrics	Expression for the 2-channel filter banks
Overall	reconstruction error (E_r)	$H_0(\Leftrightarrow z)G_0(z) + H_1(\Leftrightarrow z)G_1(z)$
	alias distortion (E_a)	$\int_{\omega=0}^{\pi}(\|T(e^{j\omega})\|^2 \Leftrightarrow 1)^2 d\omega$
	phase distortion (D_p)	$\int_{\omega=0}^{\pi}(P(e^{j\omega}) \Leftrightarrow P_d(e^{j\omega}))^2 d\omega$
Single filter	stopband energy (E_s)	$\int_{\omega=\omega_s}^{\pi} \|H(e^{j\omega})\|^2 d\omega$
	passband energy (E_p)	$\int_{\omega=0}^{\omega_p}(\|H(e^{j\omega})\| \Leftrightarrow 1)^2 d\omega$
	stopband ripple (δ_s)	Max $\|H(e^{j\omega})\|$ $\omega \in [\omega_s, \pi]$
	passband ripple (δ_p)	Max $\|\|H(e^{j\omega})\| \Leftrightarrow 1\|$ $\omega \in [0, \omega_p]$
	transition bandwidth ($\Delta\omega$)	$\omega_s \Leftrightarrow \omega_p$

Here, $P_d(e^{j\omega})$ is the desired phase response, $P(e^{j\omega})$, the actual phase response, and $|T(e^{j\omega})|$, the overall amplitude response. ω_s and ω_p are stopband and passband edges.

$T(e^{j\omega})$ is defined as:

$$\begin{aligned} T(z) &= \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)] \\ T(e^{j\omega}) &= |T(e^{j\omega})|e^{jP(e^{j\omega})} \end{aligned}$$

The reconstructed signal can be written as:

$$\hat{X}(z) = T(z)\, X(z) + \frac{1}{2}[H_0(\Leftrightarrow z)G_0(z) + H_1(\Leftrightarrow z)G_1(z)]\, X(\Leftrightarrow z).$$

The expressions of the overall performance measures in Table 1.4 are illustrated using the 2-channel filter bank.

Fig. 1.4 illustrates ripples and energies in a single low-pass filter [55, 36]. *It is important to*

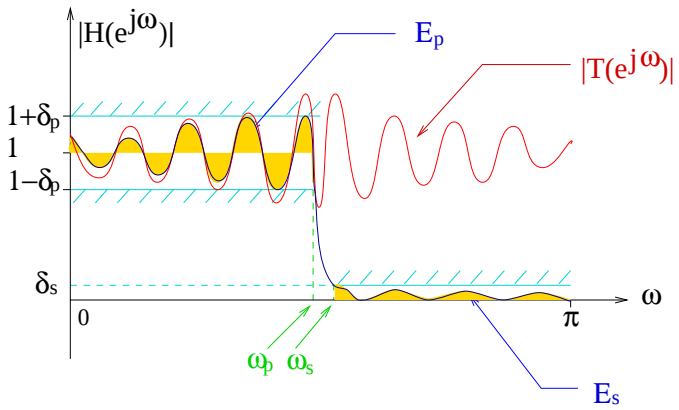

Figure 1.4: Illustration of design objectives of a single filter ($[0, \omega_p]$ is the pass band; $[\omega_s, \pi]$, the stop band; $[\omega_p, \omega_s]$, the transition band)

note that the objectives specified in Table 1.4 are not unique.

1.2.3 Previous research

QMF FIR filter banks By setting high-pass (HP) filter $h_1(n)$ in Figure 1.2 to be the quadrature mirror filter (QMF) of the low-pass (LP) filter $h_0(n)$, the filter bank design problem is simplified to be a single prototype ($h_0(n)$) filter design.

Designing $H_0(z)$ has been done in both the frequency [19, 57, 5, 7, 22, 8, 67, 46, 48, 30] and time domains [27, 47]. In the frequency domain, the most often considered objectives are reconstruction error and stopband ripple. As stopband ripple can't be expressed by a mathematical formula, stopband attenuation (stopband energy) is used instead (denoted as E_s in Eq. (1.1)). The objectives then become

$$\begin{aligned} \text{Minimize } E_r &= \int_{\omega=0}^{\pi} (|H_0(e^{j\omega})|^2 + |H_0(e^{j(\omega-\pi)})|^2 \Leftrightarrow 1)^2 \, d\omega \quad \text{and} \\ E_s &= \int_{\omega=\omega_s}^{\pi} |H_0(e^{j\omega})|^2 d\omega \end{aligned} \tag{1.1}$$

where ω_s is the stopband edge. One approach to solve Eq. (1.1) is to use unconstrained optimization techniques to minimize the weighted sum of E_r and E_s [19, 10, 52, 5, 31]. Another is to optimize the nonlinear constrained formulation using Lagrange multiplier methods [17, 15]. Nayebi [27] gave a time domain formulation with constraints in the frequency domain.

The drawback of FIR filter banks is that it is impossible to obtain perfect reconstruction in a non-trivial case [34]. It becomes clearer when we use a polyphase representation. In polyphase form, filter $H(z)$ is expressed as $A_0(z^2) + z^{-1}A_1(z^2)$, where $A_0(z)$ and $A_1(z)$ are polyphase filters. If $H_1(z)$ is the quadrature mirror image of low-pass filter $H_0(z)$, then $H_1(z) = A_0(z) \Leftrightarrow z^{-1}A_1(z^2)$. In order to eliminate reconstruction error, $A_0(z)$ and $A_1(z)$ need to be designed as all-pass filters. We know that in the FIR case, only the impulse function has uniform response in the whole frequency space. That is why the example shown in Table 1.2 can have the PR property. Another disadvantage of using QMF FIR filters is the long time delay. If the composing filter is N-tap, then the delay of the filter bank becomes $N \Leftrightarrow 1$.

Non-QMF filter banks Methods for designing non-QMF FIR filter banks include unconstrained optimization methods, similar to those for designing QMF FIR filter banks except that the former has more dimensions [34], and constrained optimization methods [31, 1], including Lagrange multiplier [1, 45] and time domain methods [29].

IIR filter banks Since IIR filters can be decomposed into two all-pass subfilters, amplitude distortion can be zero in an IIR filter bank [55]. Usually IIR filter banks have shorter time delay and require fewer computations than FIR filter banks.

However, IIR filters are generally more difficult to design than FIR filters because of their strong nonlinearality. There exist many methods for designing IIR filter banks [9, 41, 16, 25, 56, 53, 51, 33, 21, 4, 3, 68], including polyphase all-pass filter banks [41, 16, 25, 56], unitary and non-unitary filter banks with all-pass subfilters [33, 32, 51], cosine modulated filter banks [21, 33], and direct optimization methods [36, 39, 55]. The main problem of using IIR filters in filter bank design is phase distortion. This has been addressed in the design of approximate linear phase IIR filters [18, 44, 12].

Multiband and multirate filter banks Multiband and multirate filter banks use different sampling rates in different channels [6]. They have been actively studied, due to their flexibility and tolerance to errors [11, 54].

To restrict the search space, many properties, such as linear phase, paraunitary, and cosine modulation, are imposed. To further simplify the problem, most studies only consider aliasing errors in adjacent channels by assuming that stopband attenuation in non-adjacent subbands is

small. Existing design methods are based on either the frequency domain or the time domain. In the frequency domain, both unconstrained formulations [64, 23, 21, 38, 35] and constrained formulations [2] have been proposed. Other strategies include spectral factorization [48, 62] and hybrid methods [30].

1.2.4 2-channel QMF FIR LP filter bank design

In this research, 2-channel QMF FIR LP filter bank design is studied as it is most widely used. The reconstructed signal obtained at the end is defined as follows [11]:

$$\hat{X}(z) = \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)]\, X(z) + \frac{1}{2}[H_0(\Leftrightarrow z)G_0(z) + H_1(\Leftrightarrow z)G_1(z)]\, X(\Leftrightarrow z) \tag{1.2}$$

where $X(z)$ is the original signal, and $H_i(z)$ and $G_i(z)$ (i=1, 2) are the analysis and synthesis filters, respectively. To perfectly reconstruct the original signal based on $\hat{X}$, we have to eliminate aliasing distortion (caused by decimation) [11, 59, 60, 62], amplitude distortion, and phase distortion [54]. By setting $G_0(z) = H_1(\Leftrightarrow z)$, $G_1(z) = \Leftrightarrow H_0(\Leftrightarrow z)$ and $H_1(z) = H_0(\Leftrightarrow z)$, aliasing distortion can be eliminated, so only one prototype filter $H_0(z)$ exists in the system. Let $\mathsf{h}(n)$ represent the filter parameters. If $\mathsf{h}_0(n)$ is symmetric, then $\mathsf{h}_1(n) = (\Leftrightarrow 1)^n \mathsf{h}_0(n)$ is antisymmetric, meaning that the system has linear phase, so there is no phase distortion for the overall filter bank. With zero aliasing and phase distortions, the primary goal becomes reducing reconstruction error. Hence, the design problem becomes an optimization problem that searches for $\mathsf{h}_0(n)$ which minimizes amplitude distortion (represented as E_r in Eq. (1.1)).

1.2.5 Problem statement

Algorithms for designing filter banks can be classified into two categories: optimization-based and non-optimization-based. Optimization-based methods formulate the design problem as a multi-objective nonlinear optimization problem [52], whose form may depend on the application and the composing filter type. Methods described in Section 2.1 are then applied to convert the problem into a single-objective optimization problem [19, 17, 15]. Finally, the problem is solved by existing optimization methods that include gradient-descent, Lagrange-multiplier, quasi-

Newton, mixed integer-linear programming, simulated-annealing, and genetics-based methods [19, 15].

Filter-bank design problems have also been solved by non-optimization algorithms, like spectral factorization [22] and heuristic methods (as in IIR-filter design [36, 39]). These methods generally do not continue to find better designs once a suboptimal design has been found [55]. Note that in the design problem,

- The objectives are not unique and may be conflicting, leading to designs with different tradeoffs; and
- Some design objectives and their derivatives are not in closed forms.

For the design of a 2-channel QMF FIR filter bank, as aliasing and phase distortions are eliminated, the outstanding objectives of a design are to:

1. minimize reconstruction error (E_r),
2. minimize stopband energy (E_s),
3. minimize passband energy (E_p),
4. minimize stopband ripple (δ_s),
5. minimize passband ripple (δ_p), and
6. minimize transition bandwidth ($\Delta\omega$).

To solve the problem, we keep the reconstruction error as our objective, and convert all the others into constraints by using the best known solution as our reference. Therefore, the problem formulation is:

$$\begin{aligned} \text{Minimize} \quad & E_r \\ \text{subject to} \quad & E_s \leq c_{E_s} \\ & E_p \leq c_{E_p} \\ & \delta_s \leq c_{\delta_s} \\ & \delta_p \leq c_{\delta_p} \\ & \Delta\omega \leq c_{\Delta\omega} \end{aligned} \tag{1.3}$$

This is a single-objective, constrained nonlinear optimization problem.

1.3 Summary and Significance of This Research

In this chapter, we have given a brief introduction to the filter bank design problem, its functionality, structure, classification and applications. It is obvious that the improvements in filter bank design will benefit video and audio coding, as well as many other areas.

Both multi-objective unconstrained formulations and single-objective constrained formulations have been proposed. Our approach uses the reconstruction error as the objective, and sets stopband energy, passband energy, stopband ripple, passband ripple and transition bandwidth as constraints by using the best known design as our reference. The design problem, therefore, becomes a general nonlinear constrained optimization problem. This research is important as techniques developed for designing QMF filter banks can be applied to the the design of other filter banks.

Chapter 2

Problem Formulations

Generally speaking, the design of QMF FIR filter banks is a multiple-objective optimization problem. As mentioned before, reconstruction error is used to evaluate the overall performance of a filter bank, and the stopband energy, passband energy, stopband ripple, passband ripple, transition bandwidth, etc. assess the performance of the composing filters.

In this chapter, we discuss the formulations proposed for the QMF filter bank design problem. Also, we present our problem formulation, and propose the derived closed-form formulae for some objectives and their derivatives.[1]

2.1 General Multi-Objective Optimization Problems

The general *nonlinear continuous constrained optimization problem* has the following form:

$$\begin{array}{lll} \text{Minimize} & \mathsf{F}(X) & \\ \text{Subject to} & \mathsf{H}(X) = 0 & X = (x_1, \ldots, x_n) \in R^n \\ & \mathsf{G}(X) \leq 0 & \end{array} \tag{2.1}$$

where X is an n-dimensional vector, and $\mathsf{F}(X)$, $\mathsf{H}(X)$, and $\mathsf{G}(X)$ are three sets of real-valued nonlinear continuous functions that stand for the objective functions, equality constraints, and inequality constraints, respectively.

[1] We refer to the QMF FIR filter bank in the following sections without explicit specification, and use *filter bank* and *QMF filter bank* interchangeably.

There is no existing method that can optimize all objectives simultaneously for a general multi-objective problem [50]. We are often interested in a set of feasible solutions for which there is no other feasible solution that does at least as well on every objective and better on at least one objective [26, 49]. Such feasible solutions, called *non-dominated solutions*, form a trade-off space.

Previous approaches dealing with multi-objective problems are divided into three classes depending on the way that preferences are set: (a) *Methods that set prior preferences*, which form a weighted sum of the objectives [40, 20] and use reference points [50]; (b) *Methods that interactively change preferences*, which utilize local gradient search [42, 26] and iteratively reduce the non-dominated set [58]; and (c) *Methods that use posterior preferences*, which find all non-dominated solutions first before selecting the best one [49, 14].

2.2 Multi-Objective Unconstrained Formulations

A very popular multi-objective unconstrained formulation for QMF filter bank design is to minimize E_r and E_s. It utilizes the symmetry of LP and HP filters in a QMF filter bank.

$$\begin{aligned} &\text{Min} \quad E_r \quad \text{and} \quad E_s \\ &\text{where} \quad E_r = \int_{\omega=0}^{\pi} (|H_0(e^{j\omega})|^2 + |H_0(e^{j(\omega-\pi)})|^2 \Leftrightarrow 1)^2 \; d\omega \\ &\text{and} \quad E_s = \int_{\omega=\omega_s}^{\pi} |H_0(e^{j\omega})|^2 d\omega \end{aligned} \tag{2.2}$$

where ω_s is the stopband edge.[2] The HP filter is symmetric to the LP filter with respect to $\frac{\pi}{2}$. Reconstruction error computes the sum of squares of deviation of LP and HP from ideal filters. Hence, by suppressing the reconstruction error and stopband energy of the LP filter, hopefully, we can control the passband ripple and passband energy, and let them satisfy the passband specification.

This formulation simplifies the original design problem by optimizing a subset of the measures defined in Table 1.4. Unfortunately, optimal solutions to the simplified optimization problem are not necessarily optimal solutions to the original problem. Oftentimes, performance

[2]Note that in QMF filter banks, E_r is non-zero. A multi-rate filter bank that enforces perfect reconstruction ($E_r = 0$) can be formulated as a constrained optimization problem with a goal of minimizing E_s [17, 15].

of the properties not included in the formulation is compromised. For example, when E_r and E_s are the objectives to be minimized, stopband ripple, passband energy, and passband ripple are indirectly bounded. However, the solution that gives the smallest E_r and E_s will probably have a large transition band, which is an undesirable property.

As mentioned in Section 2.1, there is no method that can optimize all the objectives simultaneously. One approach is to optimize a weighted sum of all the objectives and to solve the new problem using unconstrained nonlinear optimization methods [19, 10, 52, 5, 31]. The weight on each objective must be chosen properly in order to find a good solution, and solving such a problem usually results in designs that trade one performance measure with respect to another.

2.3 Single-Objective Constrained Formulations

A multi-objective optimization problem can be transformed into a single-objective constrained problem. One difficulty of the transformation is how to choose an appropriate objective measure and the constrained measures, and another is to define the constraint values. The conversion depends on the application and the composing filters [52]. Note that the constraints can be defined with respect to a reference design.

Constraint-based methods have been applied to design QMF filter banks in both the frequency [19, 5, 7, 22, 46, 48] and time domains [27, 47]. As reconstruction error is the only metric to evaluate the performance of the overall filter bank, it is usually considered as the objective. In the frequency domain, the stopband ripple is often used as a constraint. As stopband ripple can't be formulated in closed form, stopband energy is used instead. Nayebi [27] proposed a time domain formulation with constraints in the frequency domain, and designed filter banks using an iterative time domain design algorithm (mixed time and frequency domain).

In this research, we formulate the design of a QMF filter bank as a constrained nonlinear optimization problem, using the reconstruction error as the objective and other measures (stopband ripple, stopband energy, passband ripple, passband energy, and transition bandwidth) as constraints (see Table 2.1). We use all the performance measures listed, because they are conflicting. Further, reconstruction error plus stopband ripple or stopband energy alone cannot represent the performance of a filter bank completely.

Table 2.1: Single-objective constrained formulation of QMF filter banks. Here, c_{E_s}, c_{E_p}, c_{δ_s}, c_{δ_p}, $c_{\Delta\omega}$ are constraint values given by the reference design. We choose Johnston's design, the best known solution, as our reference

objective	$f(x) =$	E_r
Constraints	$g_1 =$	$E_s \leq c_{E_s}$
	$g_2 =$	$E_p \leq c_{E_p}$
	$g_3 =$	$\delta_s \leq c_{\delta_s}$
	$g_4 =$	$\delta_p \leq c_{\delta_p}$
	$g_5 =$	$\Delta\omega \leq c_{\Delta\omega}$

The goal here is to find solutions whose constrained performance measures are better than or at least equal to those of the reference design, while its objective measure is better than that of the reference. Since the objective and constraints are nonlinear, the problem terrain is multi-modal, namely, with many local minima.

2.3.1 Performance evaluation

In both multi-objective unconstrained and single-objective constrained formulations, we need to evaluate the performance of a design. For example, in a single-objective constrained formulation, our goal is to find a design with better objective value, and all constraints are within the region defined by the reference design. The performance metrics listed in Table 1.4 are restated here:

$$\begin{aligned}
E_r &= \int_0^{\pi} (h^2(x,\omega) + h^2(x,\pi \Leftrightarrow \omega) \Leftrightarrow 1)^2 d\omega \\
E_s &= \int_{\omega_s}^{\pi} h^2(x,\omega) d\omega \\
E_p &= \int_0^{\omega_p} (h(x,\omega) \Leftrightarrow 1)^2 d\omega \\
\delta_s &= \mathrm{Max}|h(x,\omega)| \qquad \omega \in (\omega_s, \pi) \\
\delta_p &= \mathrm{Max}|h(x,\omega) \Leftrightarrow 1| \qquad \omega \in (0, \omega_p) \\
\Delta\omega &= \omega_s \Leftrightarrow \omega_p
\end{aligned} \tag{2.3}$$

where $x(n)$ is the N-tap prototype filter parameters, and x is symmetric, $1 \leq n \leq N$. The Fourier transform of $x(n)$ is:

$$\begin{aligned} \mathcal{F}(x(n)) &= \sum_{n=0}^{N-1} x(n)\ e^{-jn\omega} \\ &= e^{-j\frac{N-1}{2}w} \sum_{n=1}^{N/2} x(n)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \end{aligned} \tag{2.4}$$

From Eq. (2.4), phase response is linear with amplitude:

$$h(x,\omega) = \sum_{n=1}^{N/2} x(n)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \tag{2.5}$$

2.3.1.1 Derivation of closed-form formulae for E_r, E_s, and E_p

As shown in Eq. (2.3), E_r, E_s, and E_p are expressed as integrations; their function values can be estimated using numerical integration. In order to reduce computational complexity and retain accuracy, we derive closed-form formulae for E_r, E_s, and E_p here.

Objective function evaluation Reconstruction error E_r can be expressed as:

$$\begin{aligned} E_r = \frac{1}{2}\pi \sum_{n=1}^{n=N/2} \sum_{i=1}^{i=N/2} x(n) * x(i) * \Bigg[& x(n) * x(i) * 2 \\ & + \sum_{lb_1 \leq m \leq ub_1, m-n=2k, m\neq n} x(m) * x(n \Leftrightarrow m + i) \\ & + \sum_{lb_2 \leq m \leq ub_2, m-n=2k, m\neq n} x(m) * x(\Leftrightarrow n + m + i) \\ & + \sum_{lb_3 \leq m \leq ub_3, m+n=2k+1} x(m) * x(N+1 \Leftrightarrow n \Leftrightarrow m + i) \\ & + \sum_{lb_4 \leq m \leq ub_4, m+n=2k+1} x(m) * x(\Leftrightarrow N \Leftrightarrow 1 + n + m + i) \\ & + \sum_{lb_5 \leq m \leq ub_5, m-n=2k} x(m) * x(N+1 \Leftrightarrow n + m \Leftrightarrow i) \\ & + \sum_{lb_6 \leq m \leq ub_6, m-n=2k} x(m) * x(N+1+n \Leftrightarrow m \Leftrightarrow i) \\ & + \sum_{lb_7 \leq m \leq ub_7, m+n=2k+1} x(m) * x(n + m \Leftrightarrow i) \Bigg] \end{aligned}$$

$$\Leftrightarrow 2\pi \sum_{n=1}^{n=N/2} x^2(n) \tag{2.6}$$

where k is an integer, and $1 \leq k \leq N/2$. The expensive integration operation is replaced by three nested summations. The lower and upper bounds for the seven innermost summations are shown in Table 2.2.

Table 2.2: Lower and upper bounds for E_r in Eq. (2.6)

Index	Lower Bound	Upper Bound
1	Max(1, $\Leftrightarrow N/2 + n + i$)	Min(N/2, $\Leftrightarrow 1 + n + i$)
2	Max(1, $1 + n \Leftrightarrow i$)	Min(N/2, $N/2 + n \Leftrightarrow i$)
3	Max(1, $N/2 + 1 \Leftrightarrow n + i$)	Min(N/2, $N \Leftrightarrow n + i$)
4	Max(1, $N + 2 \Leftrightarrow n \Leftrightarrow i$)	Min(N/2, $3N/2 + 1 \Leftrightarrow n \Leftrightarrow i$)
5	Max(1, $\Leftrightarrow N + n + i$)	Min(N/2, $\Leftrightarrow N/2 \Leftrightarrow 1 + n + i$)
6	Max(1, $N/2 + 1 + n \Leftrightarrow i$)	Min(N/2, $N + n \Leftrightarrow i$)
7	Max(1, $1 \Leftrightarrow n + i$)	Min(N/2, $N/2 \Leftrightarrow n + i$)

In order to derive Eq. (2.6), we introduce an intermediate function $f(x, \omega)$,

$$\begin{aligned}
f(x,\omega) &= h^2(x,\omega) + h^2(x,(\pi \Leftrightarrow \omega)) \\
&= \sum_{n=1}^{N/2}\sum_{m=1}^{N/2} x(n)x(m)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega\ \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)\omega \\
&\quad + \sum_{n=1}^{N/2}\sum_{m=1}^{N/2} x(n)x(m)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)(\pi \Leftrightarrow \omega)\ \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)(\pi \Leftrightarrow \omega) \\
&= \frac{1}{2}\sum_{n=1}^{N/2}\sum_{m=1}^{N/2} x(n)x(m)\Big[\mathrm{Cos}(n \Leftrightarrow m)\omega + \mathrm{Cos}(N+1 \Leftrightarrow n \Leftrightarrow m)\omega \\
&\quad + \mathrm{Cos}(n \Leftrightarrow m)(\pi \Leftrightarrow \omega) + \mathrm{Cos}(N+1 \Leftrightarrow n \Leftrightarrow m)(\pi \Leftrightarrow \omega)\Big]
\end{aligned} \tag{2.7}$$

E_r can then be expressed in terms of f:

$$\begin{aligned}
E_r &= \int_0^{\pi} (f \Leftrightarrow 1)^2\ d\omega \\
&= \int_0^{\pi} (f^2 \Leftrightarrow 2f + 1)\ d\omega
\end{aligned}$$

$$= \int_0^{\pi} f^2 \; d\omega \Leftrightarrow 2 \int_0^{\pi} f \; d\omega + \pi \tag{2.8}$$

Substituting f by Eq. (2.7), E_r becomes:

$$\begin{aligned} E_r \;=\; & \int_0^{\pi} \sum_{n=1}^{n=N/2} \sum_{m=1}^{m=N/2} x(n)x(m) \Big[\mathrm{Cos}(n \Leftrightarrow m)\omega + \mathrm{Cos}(N+1 \Leftrightarrow n \Leftrightarrow m)\omega \\ & + \mathrm{Cos}(n \Leftrightarrow m)(\pi \Leftrightarrow \omega) + \mathrm{Cos}(N+1 \Leftrightarrow n \Leftrightarrow m)(\pi \Leftrightarrow \omega) \Leftrightarrow 1 \Big] \\ & \sum_{p=1}^{p=N/2} \sum_{i=1}^{i=N/2} \Big[\mathrm{Cos}(p \Leftrightarrow i)\omega + \mathrm{Cos}(N+1 \Leftrightarrow p \Leftrightarrow i)\omega \\ & + \mathrm{Cos}(p \Leftrightarrow i)(\pi \Leftrightarrow \omega) + \mathrm{Cos}(N+1 \Leftrightarrow p \Leftrightarrow i)(\pi \Leftrightarrow \omega) \Big] \; d\omega \end{aligned} \tag{2.9}$$

Notice that the integration interval for reconstruction error is from 0 to π. We know that if the integrand is the product of two cosine functions with integer variables, then the resulting integration value could only be π, $\frac{\pi}{2}$, or 0, depending on the integer variables:

$$\int_0^{\pi} \mathrm{Cos}(a\omega)\ \mathrm{Cos}(b\omega)\ d\omega = \begin{cases} \pi & \text{if a = b = 0} \\ \frac{\pi}{2} & \text{if } a = \pm b \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.10}$$

For example, given n, i, and $1 \leq n, i, m, p \leq N/2$, then

$$\int_0^{\pi} \mathrm{Cos}(n \Leftrightarrow m)\omega\ \mathrm{Cos}(p \Leftrightarrow i)\omega\ d\omega = \begin{cases} \pi & \text{if } m = n,\ p = i \\ \frac{\pi}{2} & \text{if } m \Leftrightarrow n = \pm(p \Leftrightarrow i) \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.11}$$

The lower and upper bounds are derived by enforcing every integer variable m, n, p, and i to be in the range $[1, N/2]$. For example, $p = \pm(n \Leftrightarrow m) + i$, and $1 \leq p \leq N/2$, means that:

$$\begin{aligned} 1 \leq n \Leftrightarrow m + i \leq N/2 \Rightarrow \quad & n + i \Leftrightarrow N/2 \;\leq m \leq\; n + i \Leftrightarrow 1 \\ 1 \leq \Leftrightarrow n + m + i \leq N/2 \Rightarrow \quad & n \Leftrightarrow i + 1 \;\leq m \leq\; n \Leftrightarrow i + N/2 \\ & 1 \;\leq m \leq\; N/2 \end{aligned} \tag{2.12}$$

The lower bounds and upper bounds for m are, therefore,

- $lb_1 = \mathrm{Max}(1, \Leftrightarrow N/1 + n + i), ub_1 = \mathrm{Min}(N/2, \Leftrightarrow 1 + n + i) \quad \text{if } p = +(n \Leftrightarrow m) + i$
- $lb_2 = \mathrm{Max}(1, 1 + n \Leftrightarrow i), ub_2 = \mathrm{Min}(N/2, N/2 + n \Leftrightarrow i) \quad \text{if } p = \Leftrightarrow(n \Leftrightarrow m) + i$

Constraint evaluation

Stopband energy. Replacing the amplitude response in Eq. (2.3) by Eq. (2.5), we have the stopband energy:

$$\begin{aligned}
E_s &= \int_{\omega_s}^{\pi} \sum_{n=1}^{N/2} x(n)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \sum_{m=1}^{N/2} x(m)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)\omega \; d\omega \\
&= \sum_{n=1}^{N/2}\sum_{m=1}^{N/2} x(n)x(m) \int_{\omega_s}^{\pi} \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \; \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)\omega \; d\omega \\
&= \frac{1}{2}\sum_{n=1}^{N/2} x(n)\; x(n)\left[(\pi \Leftrightarrow \omega_s) \Leftrightarrow \frac{\mathrm{Sin}(N+1 \Leftrightarrow 2n)\omega_s}{N+1 \Leftrightarrow 2n}\right] \\
&\quad \Leftrightarrow \frac{1}{2}\sum_{n=1}^{N/2}\sum_{m=1, m\neq n}^{N/2} x(n)\; x(m)\left[\frac{\mathrm{Sin}(n \Leftrightarrow m)\omega_s}{n \Leftrightarrow m} + \frac{\mathrm{Sin}(N+1 \Leftrightarrow n \Leftrightarrow m)\omega_s}{N+1 \Leftrightarrow n \Leftrightarrow m}\right] \qquad (2.13)
\end{aligned}$$

Passband energy. Similarly, the passband energy can be expressed as:

$$\begin{aligned}
E_p &= \int_0^{\omega_p} \left(\sum_{n=1}^{N/2} x(n)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \Leftrightarrow 1\right) \left(\sum_{m=1}^{N/2} x(m)\mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)\omega \Leftrightarrow 1\right) d\omega \\
&= \sum_{n=1}^{N/2}\sum_{m=1}^{N/2} x(n)x(m) \int_0^{\omega_p} \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \; \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow m\right)\omega \; d\omega \\
&\quad \Leftrightarrow 2\sum_{n=1}^{N/2} x(n) \int_0^{\omega_p} \mathrm{Cos}\left(\frac{N+1}{2} \Leftrightarrow n\right)\omega \; d\omega + \int_0^{\omega_p} d\omega \\
&= \frac{1}{2}\sum_{n=1}^{N/2} x(n)\; x(n)\left[\omega_p + \frac{\mathrm{Sin}(N+1 \Leftrightarrow 2n)\omega_p}{N+1 \Leftrightarrow 2n}\right] \\
&\quad + \frac{1}{2}\sum_{n=1}^{N/2}\sum_{m\neq n, m=1}^{N/2} x(n)\; x(m)\left[\frac{\mathrm{Sin}(n \Leftrightarrow m)\omega_p}{n \Leftrightarrow m} + \frac{\mathrm{Sin}(N+1 \Leftrightarrow n \Leftrightarrow m)\omega_p}{N+1 \Leftrightarrow n \Leftrightarrow m}\right]
\end{aligned}$$

$$\Leftrightarrow 2\sum_{n=1}^{N/2} x(n)\frac{\mathrm{Sin}(\frac{N+1}{2}\Leftrightarrow n)\omega_p}{\frac{N+1}{2}\Leftrightarrow n}+\omega_p \tag{2.14}$$

2.3.1.2 Procedures for computing δs, δp, $\Delta\omega$

There are no closed-form formulae for stopband ripple (δs), passband ripple (δp), and transition bandwidth ($\Delta\omega$). We use Newton's method to estimate their values.

Stopband ripple. It is computed with three steps:

1. Take samples of the amplitude response, and find coarse local maxima in the range $[\frac{\pi}{2},\pi]$.
2. Use Newton's method to refine local maxima by solving the equation $\frac{dh(x,\omega)}{d\omega}=0$. Start from the sampled local maxima.
3. Set the ripple as the largest deviation, i.e., $\delta_s=\mathrm{Max}_{i=1,N_{lms}}|h(x,\omega_i)|$. Assume that there are N_{lms} local maxima in the stopband.

The iterative updating of ω_s is computed by:

$$\omega_{new} \;=\; \omega_{old}\Leftrightarrow\frac{\frac{dh(x,\omega)}{d\omega}}{\frac{dh^2(x,\omega)}{d\omega}}\Bigg|_{\omega_{old}} \tag{2.15}$$

The first-order and second-order derivatives of $h(x,\omega)$ used in the above equation are in the following forms:

$$\begin{aligned}\frac{dh(x,\omega)}{d\omega} &= \Leftrightarrow\sum_{n=0}^{N/2-1} x(n)\ \mathrm{Sin}\left(\left(\frac{N\Leftrightarrow 1}{2}\Leftrightarrow n\right)\omega\right)\left(\frac{N\Leftrightarrow 1}{2}\Leftrightarrow n\right)\\ \frac{d^2h(x,\omega)}{d\omega} &= \Leftrightarrow\sum_{n=0}^{N/2-1} x(n)\ \mathrm{Cos}\left(\left(\frac{N\Leftrightarrow 1}{2}\Leftrightarrow n\right)\omega\right)\left(\frac{N\Leftrightarrow 1}{2}\Leftrightarrow n\right)^2\end{aligned} \tag{2.16}$$

Passband ripple. Similar to stopband ripple, passband ripple is the maximum deviation from ideal passband response. It can be estimated by finding local maxima in the range $[0,\frac{\pi}{2}]$. $\delta_p=\mathrm{Max}_{i=1,N_{lmp}}(|h(x,\omega_i)\Leftrightarrow 1.0|)$. Suppose there are N_{lmp} local maxima in passband.

Transition bandwidth. $\Delta\omega$ is the width between the passband edge and the stopband edge. Newton's method is used to find stopband edge ω_s and passband edge ω_p by solving

equations:

$$\begin{aligned} h(x, \omega_s) &= \delta_s \\ h(x, \omega_p) &= 1.0 \Leftrightarrow \delta_p \\ \Delta\omega &= \omega_s \Leftrightarrow \omega_p \end{aligned} \tag{2.17}$$

Although we don't have closed-form formulae for the stopband ripple, passband ripple, and transition bandwidth, we can evaluate those values with high precision by using Newton's method, which converges superlinearly.

2.3.2 Verification of derived formulae for E_r, E_s, E_p

In order to verify the correctness of closed-form formulae for E_r, E_s, and E_p (given in Eq. (2.6), (2.13), and (2.14)), we use an integration package to evaluate these functions, and compare the results in the following table:

Table 2.3: Verification of the derived closed-form formulae

Performance Measure	Derived Closed Form		Integration Package	
	result	exec. time (sec)	result	exec. time (sec)
E_r	4.8618413068488E-5	0.00872	4.8618413066912E-5	0.0330
E_s	2.5640741950173E-4	0.00165	2.5640741950161E-4	0.00260
E_p	4.3583697912908E-6	0.00162	4.3583697908885E-6	0.00258

Suppose ω_s and ω_p have been estimated rather accurately.[3] From the table, we can see that the function values computed from the closed-form formulae are very close to those from the integration package. The difference is in the order of 10^{-10}, so we can conclude that our derivation of reconstruction error, stopband energy, and passband energy is correct.

[3]The evaluated filter here is of 24-tap (D type). See reference [19] or Table 4.7 for filter parameters.

2.3.3 Other closed-form formulae

Reference [13] gave closed-form expressions for stopband energy and passband energy in quadratic form:

$$E_s = x^T P_s x \tag{2.18}$$

$$E_p = x^T P_p x \tag{2.19}$$

where

$$P_s = \int_{\omega_s}^{\pi} W(\omega) s(\omega) s^T(\omega) d\omega \tag{2.20}$$

$$P_p = \int_0^{\omega_p} \left[s(\omega) \Leftrightarrow \frac{H_d(\omega)}{H_d(\omega_0)} s(\omega_0) \right] . \left[s(\omega) \Leftrightarrow \frac{H_d(\omega)}{H_d(\omega_0)} s(\omega_0) \right]^T d\omega \tag{2.21}$$

$$s^T(\omega) = \begin{cases} \{1 \ \mathrm{Cos}(\omega) \ \mathrm{Cos}(2\omega) \cdots \ \mathrm{Cos}[0.5(N \Leftrightarrow 1)\omega]\} & \text{N odd} \\ \{\mathrm{Cos}(0.5\omega) \ \mathrm{Cos}(1.5\omega) \ \mathrm{Cos}(2.5\omega) \cdots \ \mathrm{Cos}[(0.5N \Leftrightarrow 0.5)\omega]\} & \text{N even} \end{cases} \tag{2.22}$$

$$H_d(\omega) = \begin{cases} 1 & 0 \leq \omega \leq \omega_p \\ 0 & \omega_s \leq \omega \leq \pi \end{cases} \tag{2.23}$$

where $W(\omega)$ is the weight function. It generally takes different weights for stopband, passband, and transition band. $H_d(\omega)$ is the desired frequency response. Stopband and passband energies given in reference [13] are with respect to constant weight, i.e., $W(\omega) = 1$ for all ω. In that case, the symmetric matrices P_s and P_p are:

i) When N is even,

$$(P_s)_{r,c} = \begin{cases} \frac{\pi}{2} \Leftrightarrow \frac{\omega_s}{2} \Leftrightarrow \frac{\mathrm{Sin}[(2r-1)\omega_s]}{2(2r-1)} & r = c \\ \Leftrightarrow \frac{\mathrm{Sin}[(r+c-1)\omega_s]}{2(r+c-1)} \Leftrightarrow \frac{\mathrm{Sin}[(r-c)\omega_s]}{2(r-c)} & r \neq c \end{cases} \tag{2.24}$$

$$(P_p)_{r,c} = \begin{cases} \frac{3}{2}\omega_p + \frac{\mathrm{Sin}[(2r-1)\omega_p]}{2(2r-1)} \Leftrightarrow 2\frac{\mathrm{Sin}[(r-0.5)\omega_p]}{(r-0.5)} & r = c \\ \omega_p + \frac{\mathrm{Sin}[(r+c-1)\omega_p]}{2(r+c-1)} + \frac{\mathrm{Sin}[(r-c)\omega_p]}{2(r-c)} \Leftrightarrow \frac{\mathrm{Sin}[(r-0.5)\omega_p]}{(r-0.5)} \Leftrightarrow \frac{\mathrm{Sin}[(c-0.5)\omega_p]}{(c-0.5)} & r \neq c \end{cases} \tag{2.25}$$

ii) When N is odd

$$(P_s)_{r,c} = \begin{cases} \pi \Leftrightarrow \omega_s & r = c = 1 \\ \frac{\pi}{2} \Leftrightarrow \frac{\omega_s}{2} \Leftrightarrow \frac{\mathrm{Sin}[2(r-1)\omega_s]}{4(r-1)} & r = c \neq 1 \\ \Leftrightarrow \frac{\mathrm{Sin}[(r+c-2)\omega_s]}{2(r+c-2)} \Leftrightarrow \frac{\mathrm{Sin}[(r-c)\omega_s]}{2(r-c)} & r \neq c \end{cases} \tag{2.26}$$

$$(P_p)_{r,c} = \begin{cases} \frac{3\omega_p}{2} + \frac{\mathrm{Sin}[2(r-1)\omega_p]}{4(r-1)} \Leftrightarrow 2\frac{\mathrm{Sin}[(r-1)\omega_p]}{(r-1)} & r = 1 \text{ or } c = 1 \text{ or } r = c \neq 1 \\ \omega_p + \frac{\mathrm{Sin}[(r+c-2)\omega_p]}{2(r+c-2)} + \frac{\mathrm{Sin}[(r-c)\omega_p]}{2(r-c)} \Leftrightarrow \frac{\mathrm{Sin}[(r-1)\omega_p]}{(r-1)} \Leftrightarrow \frac{\mathrm{Sin}[(c-1)\omega_p]}{(c-1)} & r \neq c \end{cases} \tag{2.27}$$

By carefully examining the resulting quadratic function when N is even [4], we find that we have the same formula for stopband energy, but a different passband energy expression. The cause for that difference is that reference [13] uses a scaling factor γ when computing the deviation of the actual amplitude response in passband ($e_p(\omega)$):

$$e_p(\omega) = H_a(\omega) \Leftrightarrow \gamma H_d(\omega) \tag{2.28}$$

$$\gamma = \frac{H_a(\omega_0)}{H_d(\omega_0)} \tag{2.29}$$

where ω_0 is a reference frequency. The expression of passband energy in Eq. (2.19) takes the sum of squares of deviation from a scaled ideal filter, $\gamma H_d(\omega)$, instead of $H_d(\omega)$ itself. If γ is modified to be:

$$\gamma = \frac{x^T s(\omega_0)}{H_a(\omega_0)} \tag{2.30}$$

then P_p will become:

$$P_p = \int_0^{\omega_p} \left[s(\omega) \Leftrightarrow \frac{H_d(\omega)}{H_a(\omega_0)} s(\omega_0) \right] . \left[s(\omega) \Leftrightarrow \frac{H_d(\omega)}{H_a(\omega_0)} s(\omega_0) \right]^T d\omega \tag{2.31}$$

[4] In our research, we only consider filter banks with even order. The results can be directly applied to those with odd order.

Here, ω_0 is set to 0. As $H_a(\omega_0)$ usually does not equal to 1, $H_a(\omega_0) \neq H_d(\omega_0)$. The resulting matrix for passband energy becomes:

$$(P_p)_{r,c} = \begin{cases} \omega_p + \frac{\omega_p}{2H_a^2(\omega_0)} + \frac{\mathrm{Sin}[(2r-1)\omega_0]}{2(2r-1)} \Leftrightarrow 2\frac{\mathrm{Sin}[(r-0.5)\omega_0]}{(r-0.5)H_a(\omega_0)} & \mathrm{r} = \mathrm{c} \\ \frac{\omega_p}{H_a^2(\omega_0)} + \frac{\mathrm{Sin}[(r+c-1)\omega_0]}{2(r+c-1)} + \frac{\mathrm{Sin}[(r-c)\omega_0]}{2(r-c)} \Leftrightarrow \frac{\mathrm{Sin}[(r-0.5)\omega_0]}{(r-0.5)H_a(\omega_0)} \Leftrightarrow \frac{\mathrm{Sin}[(c-0.5)\omega_0]}{(c-0.5)H_a(\omega_0)} & r \neq c \end{cases} \quad (2.32)$$

We evaluate E_s and E_p of some filters given in [19] with the above formulae and the formulae we derived, and list the results in Table 2.4 and Table 2.5.

Table 2.4: Stopband energy from two closed-form formulae

Filter Type	Derived Closed Form	Reference [13]
24D	2.5640741950173E-4	2.5640741950160E-4
32D	3.8600626793561E-5	3.8600626793499E-5
48D	6.8922566225094E-7	6.8922566291406E-7
64D	1.2118908737646E-8	1.2118908697082E-8

Table 2.5: Passband energy from two closed-form formulae

Filter Type	Derived Closed Form	Reference [13]	Revised
24D	4.3583697912908E-6	4.9655639900012E-6	4.3583697912121E-6
32D	5.8291825966172E-7	6.6904730518861E-7	5.8291826005458E-7
48D	3.5448644020164E-8	3.5266978332550E-8	3.5448642804942E-8
64D	9.2525735961857E-9	1.3417417828001E-8	9.2525662096412E-9

It is shown in the table, for example, that E_s of the 24D filter bank computed by reference [13] is the same as our result, but E_p is larger. After we revise the matrix of passband energy $(P_p)_{r,c}$ to Eq. (2.32), E_p becomes $4.3583697912121E \Leftrightarrow 6$, which matches our result.

2.3.4 Gradient evaluation

As we describe later in chapter 3, the Novel method [65, 43] is used to solve the nonlinear constrained optimization problem in our research. Novel is descent-based in the local search stage; gradients of the objective function and constraints need to be evaluated for each local descent. Here, we derive closed-form formulae for the derivatives of reconstruction error, stop-

band energy, and passband energy. The difference equation method is used to estimate the gradients of the other three constraints.

Gradients of objective function

$$\begin{aligned}
\frac{dE_r}{dx(i)} &= 2\pi \sum_{n=1}^{n=N/2} x(n) * \Bigg[x(n) * x(i) * 2 \\
&+ \sum_{lb_1 \le m \le ub_1, m-n=2k, m \neq n} x(m) * x(n \Leftrightarrow m + i) \\
&+ \sum_{lb_2 \le m \le ub_2, m-n=2k, m \neq n} x(m) * x(\Leftrightarrow n + m + i) \\
&+ \sum_{lb_3 \le m \le ub_3, m+n=2k+1} x(m) * x(N + 1 \Leftrightarrow n \Leftrightarrow m + i) \\
&+ \sum_{lb_4 \le m \le ub_4, m+n=2k+1} x(m) * x(\Leftrightarrow N \Leftrightarrow 1 + n + m + i) \\
&+ \sum_{lb_5 \le m \le ub_5, m-n=2k} x(m) * x(N + 1 \Leftrightarrow n + m \Leftrightarrow i) \\
&+ \sum_{lb_6 \le m \le ub_6, m-n=2k} x(m) * x(N + 1 + n \Leftrightarrow m \Leftrightarrow i) \\
&+ \sum_{lb_7 \le m \le ub_7, m+n=2k+1} x(m) * x(n + m \Leftrightarrow i) \Bigg] \\
&\Leftrightarrow 4\pi x(i)
\end{aligned} \tag{2.33}$$

Gradients of constraints

$$\begin{aligned}
\frac{dE_s}{dx(i)} &= x(i) \left[(\pi \Leftrightarrow \omega_s) \Leftrightarrow \frac{\mathrm{Sin}(N + 1 \Leftrightarrow 2i)\omega_s}{N + 1 \Leftrightarrow 2i} \right] \\
&\Leftrightarrow \sum_{n=1, n \neq i}^{N/2} x(n) \left[\frac{\mathrm{Sin}(n \Leftrightarrow i)\omega_s}{n \Leftrightarrow i} + \frac{\mathrm{Sin}(N + 1 \Leftrightarrow n \Leftrightarrow i)\omega_s}{N + 1 \Leftrightarrow n \Leftrightarrow i} \right] \\
&\Leftrightarrow \frac{d\omega_s}{dx(i)} h^2(x, \omega) \Big|_{\omega = \omega_s}
\end{aligned} \tag{2.34}$$

$$\begin{aligned}
\frac{dE_p}{dx(i)} &= x(i) \left[\omega_p + \frac{\mathrm{Sin}(N + 1 \Leftrightarrow 2i)\omega_s}{N + 1 \Leftrightarrow 2i} \right] \\
&+ \sum_{n=1, n \neq i}^{N/2} x(n) \left[\frac{\mathrm{Sin}(n \Leftrightarrow i)\omega_p}{n \Leftrightarrow i} + \frac{\mathrm{Sin}(N + 1 \Leftrightarrow n \Leftrightarrow i)\omega_p}{N + 1 \Leftrightarrow n \Leftrightarrow i} \right] \\
&\Leftrightarrow 2 \frac{\mathrm{Sin}(\frac{N+1}{2} \Leftrightarrow i)\omega_p}{\frac{N+1}{2} \Leftrightarrow i}
\end{aligned}$$

$$+ \frac{d\omega_p(x)}{dx(i)} \left(h(x,\omega)\Big|_{\omega=\omega_p} \Leftrightarrow 1.0 \right)^2 \tag{2.35}$$

Note that $\frac{dE_s}{dx(i)}$ and $\frac{dE_p}{dx(i)}$ are partly in closed form, since $\frac{d\omega_s}{dx(i)}$ and $\frac{d\omega_p}{dx(i)}$ have to be estimated using the difference equation method. For example,

$$\frac{d\omega_p}{dx(i)} = \frac{\omega_p(x(i)+dx) \Leftrightarrow \omega_p(x(i) \Leftrightarrow dx)}{2dx} \tag{2.36}$$

$\frac{d\delta_s}{dx(i)}$, $\frac{d\delta_p}{dx(i)}$, $\frac{d\Delta\omega}{dx(i)}$ are estimated using the difference equation method, too.

2.4 Summary

Two formulations for the filter bank design problem, multi-objective unconstrained and single-objective constrained forms, are discussed in this chapter. As there exist no methods to solve general multi-objective optimization problems, the second form is considered in our research. An important issue involved is performance evaluation. Closed-form formulae for reconstruction error, stopband energy, passband energy, and their derivatives are presented, as well as procedures to evaluate stopband ripple, passband ripple, and transition bandwidth. By comparing our results to the results given by a numerical integration package, we conclude that our derivation is correct.

Chapter 3

Optimization Methods

Finding global optimal solutions for nonlinear continuous constrained problems is one of the most challenging tasks in optimization, because the whole space has to be explored before identifying the optimum. Filter bank design, as described in the previous chapters, can be formulated as a single objective constrained nonlinear optimization problem:

$$\begin{aligned} \text{Minimize} \quad & E_r \\ \text{subject to} \quad & E_p \leq C_{E_p} \qquad E_s \leq C_{E_s} \\ & \delta_p \leq C_{\delta_p} \qquad \delta_s \leq C_{\delta_s} \\ & \Delta\omega \leq C_{\Delta\omega} \end{aligned} \tag{3.1}$$

3.1 Definitions and Conditions of Optimal Solutions

Optimization problems can be divided into two classes: unconstrained and constrained. There exist many well-developed methods for the former. In general, constrained problems are first converted into unconstrained problems, then solved by using unconstrained optimization methods. We start from an unconstrained optimization problem:

$$\begin{aligned} \text{Minimize} \quad & f(\boldsymbol{x}) \\ & \boldsymbol{x} \in \Omega \end{aligned} \tag{3.2}$$

where Ω is the feasible region.

Formal definitions of global and local solutions [24] are as follows:

Global minimum point A point $x^* \in \Omega$ is said to be the global minimum point of $f(x)$ over Ω if $f(x) \geq f(x^*)$ for all $x \in \Omega$.

Local minimum point A point $x^* \in \Omega$ is said to be a local minimum point of $f(x)$ over Ω if there is $\varepsilon > 0$ such that $f(x) \geq f(x^*)$ for all $x \in \Omega$ within a distance of ε of x^* (that is, $x \in \Omega$ and $|x \Leftrightarrow x^*| < \varepsilon$).

Correspondingly, there are two kinds of searches based on the search objective: global search and local search. **The first order necessary condition** for local minimum solutions of an unconstrained problem is as follows:

Let Ω be a subset of E^n, and let $f \in C^1$ be a function on Ω. If x^* is a local minimum point of f over Ω and if x^* is an interior point of Ω, then

$$\nabla f(x^*) = 0 \tag{3.3}$$

Note that the above condition holds only when x^* is inside the feasible region. If the local minimum point of f over Ω is on the boundary, then the derivative of f at x^* could be any value. To find out local optimal solutions, one needs to solve Eq. (3.3) and evaluate the function values at boundary points.

There exist many methods for solving unconstrained problems by utilizing the information of descents, such as Newton's method, steepest descent method, conjugate gradient method, and quasi-Newton method. One way to solve Eq. (3.3) is by transforming it into an ordinary differential equation (ODE):

$$\frac{dx}{dt} = \Leftrightarrow \nabla f(x) \tag{3.4}$$

and use an ODE solver to get the solution.

3.2 Previous Research on Global Optimization

The first-order necessary condition given in Eq. (3.3) is a property of local optimal solutions. Since a solution depends on the starting point, it may or may not be a global solution. We take the filter bank problem of Eq. (3.1) as an example to show three global optimization approaches.

The first approach uses a local search to determine a local minimum, and to focus on bringing the search out of the local minimum once it gets there. This approach can be classified into two categories: deterministic and probabilistic. Deterministic methods, such as covering methods and generalized descent methods, do not work well when the search space is large. On the other hand, probabilistic methods are weak in either their local or global search. For instance, gradient information is not used well in simulated annealing and evolutionary algorithms. In contrast, gradient descent algorithms with multi-starts and random probing are not efficient for global search.

In the second approach, penalty methods, all constraints are absorbed into the objective function and weighted by penalty coefficients (Eq. (3.5)). This converts a constrained problem into an unconstrained problem, allowing many well-developed optimization methods to be applicable:

$$\begin{aligned} \text{Minimize} \quad & f = E_r + w_1(E_p \Leftrightarrow C_{E_p}) + w_2(E_s \Leftrightarrow C_{E_s}) \\ & + w_3(\delta_p \Leftrightarrow C_{\delta_p}) + w_4(\delta_s \Leftrightarrow C_{\delta_s}) + w_5(\Delta\omega \Leftrightarrow C_{\Delta\omega}) \end{aligned} \tag{3.5}$$

The difficulty of using penalty methods lies in finding appropriate penalty coefficients, $w_i, i = 1, \cdots, 5$, in the above equation. Selected ahead of time, these coefficients are used to penalize a constraint when it is violated. For multiple-constraint problems, the convergence process highly relies on the relative penalty values among the constraints.

In the third approach, Lagrangian methods, the problem is solved by minimizing a Lagrangian function defined as the sum of the objective function and the weighted sum of all the constraints:

$$\begin{aligned} \text{Minimize} \quad & L = E_r + \lambda_1(E_p \Leftrightarrow C_{E_p}) + \lambda_2(E_s \Leftrightarrow C_{E_s}) \\ & + \lambda_3(\delta_p \Leftrightarrow C_{\delta_p}) + \lambda_4(\delta_s \Leftrightarrow C_{\delta_s}) + \lambda_5(\Delta\omega \Leftrightarrow C_{\Delta\omega}) \end{aligned} \tag{3.6}$$

This is similar to a penalty-based method, but is more powerful because the Lagrange multipliers, $\lambda_i, i = 1, \cdots, 5$, are dynamically adjusted in order to push the search towards a feasible region.

In a Lagrangian formulation, a local minimum in a feasible region is called a *saddle point* at which

$$\begin{aligned} \nabla_{x_i} L &= 0, \quad i = 1, \cdots, n \\ \nabla_{\lambda_j} L &= 0, \quad j = 1, \cdots, 5 \end{aligned} \tag{3.7}$$

By using this property, optimal solutions of the original problem (Eq. (3.1)) can be found by *local search methods* that perform gradient descents in both the variable space and Lagrange-variable space.

Since a saddle point is only a local minimum in a feasible region, *global search methods* are needed to bring the search out of a local optimal solution. Strategies like random restarts are not effective because the search space is usually too large to be covered completely. The success of global search methods relies on how well the problem space is traversed.

In this research, we use Novel (*Nonlinear Optimization Via External Lead*), a nonlinear optimization approach proposed by Wah, Chang, and Shang to solve global optimization problems (see references [63], [65] and [43].) In Novel, a search trajectory is formed by both external lead and gradient descent, where the external lead is responsible for pulling the search out of a local terrain, and the gradient descent is used to locate the exact local optimal points.

3.3 Novel Global Optimization Method

Novel is a global optimization method developed to solve nonlinear unconstrained and constrained optimization problems [65, 43]. The implementation of Novel includes two stages: *a global exploration stage* and *a local search stage*.

Since local searches are expensive, in order to avoid unpromising local search, Novel explores the variable space in the first stage and looks for promising regions in the local search stage. This is important because it identifies good starting points before applying expensive local searches. Novel is more efficient than multi-start algorithms that randomly probe a problem space and choose random starting points.

A heuristic gradient-independent trace function is introduced to pull the trajectory out of a local minimum. The trace function is as follows:

$$T_i(t) = \rho \ \sin\left[2\pi\left(\frac{t}{2}^{0.95+0.45\frac{i-1}{n}} + 2\pi\frac{i \Leftrightarrow 1}{n}\right)\right] \quad i = 1, \cdots, n \tag{3.8}$$

where ρ is a coefficient specifying the search range, i represents the ith dimension, and n is the order of the variable space. This function is designed to be aperiodic, differentiable, and bounded, in order for it to explore the whole search space continuously. Also, the trace function is chosen to traverse the search space from coarse to fine.

In the *global search stage*, the dynamics of the trajectory is controlled by two forces: trace function and local gradient. The latter leads the trajectory towards a local minimum, whereas the force exerted by the trace function leads the trajectory to traverse the whole space. The force due to the trace is particularly important because it provides a continuous means of going from one local region to another, without being trapped inside a local region. The dynamic changes of variable x and Lagrange variable μ are described as:

$$\begin{aligned} \frac{dx}{dt} &= \Leftrightarrow \mu_g \nabla_x L(x(t), \mu(t)) \Leftrightarrow \mu_t (x(t) \Leftrightarrow T_x(t)) \\ \frac{d\mu}{dt} &= \nabla_\mu L(x(t), \mu(t)) \end{aligned} \tag{3.9}$$

where μ_g and μ_t are weights of the local gradient and the trace function, respectively. For a constrained problem, only the original variables are affected by the external trace function, but not the Lagrange multipliers. This is true because the original variables are only bounded by the given feasible region, but the Lagrange multipliers are related to the current objective value and the constraint values. If a constraint is violated, then that particular multiplier will increase in order to add more weight of that constraint in the Lagrangian function. If an external trace function is added upon the Lagrange multipliers, it may contradict with the direction pointed to by the Lagrange multipliers, hence, mislead the search.

In the *local search stage*, Novel starts from the promising starting points identified in the global search stage and applies local searches to find saddle points in the Lagrangian function space. These local searches include gradient descents in the original variable space and gradient ascents in the Lagrange multiplier space. The solutions thus found correspond to the solutions

whose objective function value is a local minimum and all constraints are satisfied. After finite search time, the best solution from local searches is selected as the global solution.

The local search stage in our experiments is implemented by LSODE, an ordinary differential equation solver developed at the Lawrence Livermore National Lab. As mentioned before, we want to find saddle points with descents in the original variable space and ascents in the Lagrange multiplier space:

$$\begin{aligned} \frac{dx}{dt} &= \Leftrightarrow \nabla_x L(x(t), \mu(t)) \\ \frac{d\mu}{dt} &= \nabla_\mu L(x(t), \mu(t)) \end{aligned} \tag{3.10}$$

An error tolerance acts as the input parameter to LSODE: whenever the gradient change is within the error tolerance, the ODE solver will stop execution and return current variable values as the solution.

3.4 Constrained Optimization Problems

Constrained optimization problems can be classified into two categories by the form of their constraints: equality constrained and inequality constrained.

3.4.1 Equality constrained problems

An equality constrained problem with one objective function and m equality constraints can be expressed as follows:

$$\begin{aligned} &\text{Minimize} && f(x) \\ &\text{such that} && h_k(x) = 0, \quad 1 \leq k \leq m \end{aligned} \tag{3.11}$$

The Lagrange multiplier method is a well-known method to solve equality constrained problems. It converts the original constrained problem into an unconstrained problem as follows:

$$\text{Minimize } L(x, \lambda) = f(x) + \sum_{k=1}^{m} \lambda_k \, h_k(x) \tag{3.12}$$

where $\lambda_k, k = 1, \cdots, m$ are Lagrange multipliers.

In our experiments, we use the Augmented Lagrangian method, which introduces a quadratic term in order to be more stable than the original Lagrangian method:

$$\text{Minimize } L(x, \lambda) \quad = \quad f(x) + \sum_{k=1}^{m} \lambda_k \, h_k(x) + \frac{1}{2} c \sum_{k=1}^{m} h_k(x)^2 \tag{3.13}$$

The last term in Eq. (3.13) puts more weight on the constraints. In our experiments, we set c to be a constant of 2.

The Lagrangian function $L(x, \lambda)$ is a function of both the original variables (original variable space) and the Lagrange multipliers (Lagrange-variable space). Applying the first-order necessary condition for local optimum shown in Eq. (3.3), the derivative of the Lagrangian function with respect to both the original and the Lagrange multipliers should be zero at a local optimum:

$$\nabla_x L(x, \lambda) = 0 \text{ and } \nabla_\lambda L(x, \lambda) \quad = \quad 0 \tag{3.14}$$

Substituting L by the right hand side of Eq. (3.13), we get:

$$\nabla_x f(x) + \sum_{k=1}^{m} \lambda_k \nabla_x h_k(x) + \sum_{k=1}^{m} 2 h_k(x) \nabla_x h_k(x) \quad = \quad 0 \tag{3.15}$$

$$h_k(x) \quad = \quad 0, \quad 1 \leq k \leq m \tag{3.16}$$

3.4.2 Inequality constrained problems

Inequality constraints are generally converted to equality constraints before applying the methods introduced in the last section. Without loss of generality, we consider problems with inequality constraints only in the following discussion. We assume the total number of constraints involved is p.

$$\begin{array}{ll} \text{Minimize} & f(x) \\ \text{such that} & g_j(x) <= 0, \quad 1 \leq j \leq p \end{array} \tag{3.17}$$

Maxq Method The Maxq method [66] converts an inequality constraint as follows.

$$g_j(x) \leq 0 \quad \Leftrightarrow \quad Q_j(x) = Max(0, g_j(x))^{q_j} = 0 \tag{3.18}$$

where $q_j > 1$ $(j = 1, 2, \ldots, p)$ are control parameters. Whenever x is in the feasible region defined by constraint $g_j(x)$, $Q_j(x)$ is equal to zero. Therefore, $Q_j(x)$ carries zero weight in the Lagrangian function when the constraint is satisfied.

The augmented Lagrangian function becomes

$$\text{Minimize } L(x, \mu) = f(x) + \sum_{j=1}^{p} \mu_j Max(0, g_j(x))^{q_j} + \sum_{j=1}^{p} Max(0, g_j(x))^{2q_j} \tag{3.19}$$

Assuming that the q_js are constant, the corresponding dynamic system is

$$\begin{aligned} \frac{dL}{dx_i} &= \frac{df(x)}{dx_i} + \sum_{j=1}^{p} \mu_j q_j \frac{dg_j(x)}{dx_i} Max(0, g_j(x))^{q_j - 1} \\ &\quad + \sum_{j=1}^{p} 2q_j \frac{dg_j(x)}{dx_i} Max(0, g_j(x))^{2q_j - 1} \\ \frac{dL}{d\mu_j} &= Max(0, g_j(x))^{q_j} \end{aligned} \tag{3.20}$$

As shown in Eq. (3.20), the gradient of the original variables has a term with power of $q_j \Leftrightarrow 1$. If the solution happens to be on the boundary of a constraint $g_j(x)$, then $Max(0, g_j(x)) = 0$. In order to avoid being divided by zero $(Max(0, g_j(x))^{q_j - 1})$, q_j must be larger than 1.

Therefore, the dynamic ODE equations to be solved using the Maxq method are:

$$\begin{aligned} \frac{dx_i}{dt} &= \Leftrightarrow \left(\frac{df(x(t))}{dx_i} + \sum_{j=1}^{p} \mu_j q_j \frac{dg_j(x(t))}{dx_i} Max(0, g_j(x(t)))^{q_j - 1} \right. \\ &\quad \left. + \sum_{j=1}^{p} 2q_j \frac{dg_j(x(t))}{dx_i} Max(0, g_j(x(t)))^{2q_j - 1} \right) \\ \frac{d\mu_j}{dt} &= Max(0, g_j(x(t)))^{q_j} \end{aligned} \tag{3.21}$$

The limitations of using the Maxq method are as follows.

1. The Maxq method approaches the boundary asymptotically from the outside of a feasible region. This means that when the solution is exactly on the boundary, Maxq takes a very long time to converge.

One way to address this problem is to use a conditional conversion: convert one or more inequality constraints into equality constraints when the violation of the constraint(s) is small enough. In this way, the constraint(s) can be exactly satisfied.

2. Heuristic selection of q_j values. As illustrated in Eq. (3.18), the original inequality constraint $g_j(x)$ is converted to equation $Q_j(x) = 0$. Only when the introduced control parameter q_j equals one are the constraints before and after the conversion equivalent. But as mentioned before, q_j must be greater than zero in order to avoid overflow. These two requirements for the value of q_j are contradictory. Hence, we choose q_j to be a function of the relative constraint value $Q_j(x)$ instead of a constant. In our experiment, it is implemented as a piecewise function,

$$q_j(x) = \begin{cases} a & \text{if } Q_j(x) > 1.0 \\ \frac{b}{1.0+e^{-cx}} & \text{if } Q_j(x) <= 1.0 \end{cases} \tag{3.22}$$

Here, $q_j(x)$ is a constant of value a when $Q_j(x)$ is greater than 1.0, and an exponential decay from b with decay ratio of c when $Q_j(x)$ is less than 1.0. Clearly a, b, and c are heuristically chosen, and their values will affect the convergence time.

3. In Eq. (3.20), the derivation of the Lagrangian gradients assumes that q_j is a constant. But in our experiments q_j is a function of $Q_j(x)$, so that q_j is, indirectly, a function of x. When computing the derivative of the Lagrangian function L with respect to x, we should also consider $\frac{dq_j(x)}{dx_i}$ besides $\frac{df(x)}{dx_i}$ and $\frac{dg_i(x)}{dx_i}$. Negligence of the effect of $q_j(x)$ could mislead the search to points far from the optimal solution. (This has been corrected; see reference [66] for updated information.)

Slack-Variable Method The inequality constraints are transformed to equality constraints by adding slack variables. For example, each constraint $g_j(x) \leq 0$ becomes

$$g_j(x) + z_j^2 = 0 \tag{3.23}$$

where z_j is a slack variable. Similarly, the augmented Lagrangian function is as follows:

$$L = f(x) + \sum_{j=1}^{p} \mu_j (g_j(x) + z_j^2) + \frac{1}{2} \sum_{j=1}^{p} (g_j(x) + z_j^2)^2 \tag{3.24}$$

After simplification, the gradients of the Lagrangian function with respect to the original variables and Lagrange multipliers are

$$\begin{aligned} \frac{dL}{dx_i} &= \frac{df(x)}{dx_i} + \sum_{j=1}^{p} \frac{dg_j(x)}{dx_i} Max(0, \mu_j + g_j(x)) \\ \frac{dL}{d\mu_j} &= Max(0, \mu_j + g_j) \Leftrightarrow \mu_j \end{aligned} \tag{3.25}$$

Observations show that using this method, when a solution is on the boundary, the trajectory can get in and out of the feasible region before reaching the solution. These oscillations may fasten the convergence process.

There is one disadvantage of using the slack-variable method. As long as Eq. (3.23) is not satisfied, the expression $Max(0, \mu_j + g_j)$ is not zero. Even when the trajectory is inside the feasible region, i.e., $g_j(x) \leq 0$, the effect of the constraint(s) in the augmented Lagrangian function still exists (see Eq. (3.25)). While in the Maxq method (see Eq. 3.20), whenever $g_j(x) \leq 0$, there is only the objective function left in the Lagrangian function. In that circumstance, the slack-variable method is less efficient, because the trajectory does not go directly to decrease the function value.

3.5 Summary

In this chapter, we present the definition of global and local optimal solutions, and compare the pros and cons of previous global optimization methods. A brief introduction is given for Novel, the nonlinear global optimization method we used for QMF filter bank design. Two ways for handling inequality constraints are described in the end.

Chapter 4

Experimental Results

This research focuses on the 2-channel QMF filter bank design due to its simplicity and popularity. Also, the approach for designing 2-channel filter banks can be easily extended to the design of multi-channel filter banks.

In this chapter, we present our experimental results of the 2-channel filter bank designs. We propose issues regarding constraint formulations, inequality constraint transformation, global search and local search, etc. In the end, we compare our designs for some filter types with Johnston's designs [19] with respect to the performance metrics described in Chapters 1 and 2.

4.1 Constraint Formulations: Normalization versus No-Normalization

We discuss the impact of constraint formulation in this section. There are various ways of formulating the constraints; the constraint formulation is heuristic.

As mentioned before, we are designing filter banks with respect to a reference design [19]. The reconstruction error, stopband energy, passband energy, stopband ripple, passband ripple, and transition bandwidth of the reference design are represented by $E_{r,J}$, $E_{s,J}$, $E_{p,J}$, $\delta_{s,J}$, $\delta_{p,J}$, and $\Delta\omega_J$, respectively. Those performance metrics are related to each other; for example, smaller transition bandwidth usually results in worse (larger) stopband energy and stopband ripple.

Table 4.1: Tradeoffs of performance metrics

Performance	Johnston's 24C	Johnston's 24D
$E_{r,J}$	3.6962874878109E-6	4.8618413068488E-5
$E_{s,J}$	2.2036661635930E-5	2.5640741950173E-4
$E_{p,J}$	2.7781787381720E-7	4.3583697912908E-6
$\delta_{s,J}$	9.9225925339396E-3	3.2632578620193E-2
$\delta_{p,J}$	1.1051894198224E-3	4.1243806149276E-3
$\Delta\omega_J$	0.73483684656043	0.53989994313178

As illustrated in Table 4.1, Johnston's 24D filter bank trades reconstruction error, stopband energy, passband energy, stopband ripple, and passband ripple for a sharper transition band. Note that both 24C and 24D designs have the filter order of 24. Consideration of a subset of the performance metrics could be misleading. In order to evaluate the design performance completely, we use all the performance metrics listed in Table 1.4 in our experiments.

Two outstanding constraint formulations are shown in Table 4.2.

Table 4.2: Formulation with normalization versus no-normalization

objective/constraint	without normalization	with normalization
obj. $f(x) =$	E_r	$\frac{E_r}{E_{r,J}}$
s.t. $g_1 =$	$E_s \Leftrightarrow E_{s,J} \leq 0$	$\Leftrightarrow 1 + \frac{E_s}{E_{s,J}} \leq 0$
$g_2 =$	$E_p \Leftrightarrow E_{p,J} \leq 0$	$\Leftrightarrow 1 + \frac{E_p}{E_{p,J}} \leq 0$
$g_3 =$	$\delta_s \Leftrightarrow \delta_{s,J} \leq 0$	$\Leftrightarrow 1 + \frac{\delta_s}{\delta_{s,J}} \leq 0$
$g_4 =$	$\delta_p \Leftrightarrow \delta_{p,J} \leq 0$	$\Leftrightarrow 1 + \frac{\delta_p}{\delta_{p,J}} \leq 0$
$g_5 =$	$\Delta\omega \Leftrightarrow \Delta\omega_J \leq 0$	$\Leftrightarrow 1 + \frac{\Delta\omega}{\Delta\omega_J} \leq 0$

Observations show that some performance metrics may be neglected for the formulation without normalization. For instance, the absolute values of E_s and E_p are small; they could be in the order of 10^{-6}. Hence, even when there exists unacceptable violation for E_s and E_p, the corresponding penalty could be too small to be taken into account. Hence, we apply normalization to put equal weights on all constraints, regardless of their raw values. Starting from Johnston's 24D design, the solutions obtained using these two formulations are shown in Table 4.3.

Table 4.3: 24D QMF filter bank designs using normalized and no-normalized constraint formulation

		without normalization	with normalization	Johnston's
desc time (sec)		118063	369501	
Performance	E_r	0.78871156	0.63227342	4.86184122E-5
	E_s	1.00009630	1.00010723	2.56380626E-4
	E_p	0.83706263	0.73892847	4.35836979E-6
	δ_s	1.00000000	1.00000100	3.26325786E-2
	δ_p	1.00000000	1.00000110	4.12438063E-3
	$\Delta\omega$	1.00000000	1.00709532	5.39899943E-1

We append Johnston's design to the last column of Table 4.3 in order to show the reference performance values. The performance values listed in columns 2 and 3 are normalized with those of Johoston's. Here, "desc time" represents the CPU time consumed in the local search stage to locate the first local optimal solution. The Maxq method [66] is used to deal with the inequality constraints. It is clear that:

1. The normalized formulation takes a longer time for the local search to converge than the unnormalized formulation.

2. The normalized formulation results in a better objective value, but with more degradation in its constraints. For example, its transition bandwidth is violated by $7.095E \Leftrightarrow 3$, while the unnormalized formulation has a violation of $9.324E \Leftrightarrow 10$, which is almost zero.

3. Compared to the unnormalized formulation, the normalized formulation puts more weight on the objective function. This explains why the local search time of the normalized formulation is worse. When the search trajectory approaches the optimal solution, little disturbance of variable values could result in long-time oscillation of the objective value. So reducing the weight of the objective function may help to decrease the execution time. We discuss the impact of that weight later in Section 4.3.

4.2 Comparison of the Maxq and the Slack-Variable Methods

As we have mentioned before, in converting an inequality constraint to an equality constraint, a power of q is introduced to each constraint (see Eq. (3.18)). Maxq converges slowly if the

solution is exactly located on the boundary. We show later, from the best results we have obtained, that four out of the five constraints are on the boundary. Thus, it is hard for the Maxq method to converge quickly. For example, for the 24D type filter bank design based on a formulation with normalized constraints, the Maxq method takes 369,501 seconds to finish one local descent, while the slack-variable method takes only 82,327 seconds.

To compensate for this, inequality constraints are converted to equality constraints when the search trajectory is close enough to the boundary, for instance, less than 10^{-4}. Despite this conversion, the Maxq method is still very slow.

Also mentioned before, the contribution of the control parameter q in the Maxq method is not considered in the derivative of the Lagrangian function (see Eq. (3.20)). This could cause severe deviation from the actual derivative values in some occasions. The following 24D design shows this case:

Table 4.4: A 24D design and its Lagrangian gradients when the impact of the control parameter q is ignored

Variable	Parameter Values	$\frac{dL}{dx_i}(\frac{dL}{d\mu_j})$ using Maxq	$\frac{dL}{dx_i}(\frac{dL}{d\mu_j})$ using DE
x_1	1.0566434094766E-2	9.3446286014398E-4	-2.3830869222419
x_2	-2.0052998909459E-2	-1.9784592759411E-3	8.1974821375397E-2
x_3	-3.6460887164769E-3	3.6584968861604E-3	2.2138317373743
x_4	3.7748750063020E-2	-4.5094618504180E-3	1.6900496419003
x_5	-6.5583563806509E-3	5.0454718705115E-3	-9.4422394625315E-1
x_6	-6.4669862838735E-2	-6.9389020185753E-3	2.1250636222936
x_7	2.8743589268758E-2	5.6573585683566E-3	-3.1952787915568E-1
x_8	1.0806165431451E-1	-1.6229377601817E-4	1.2515739361563
x_9	-7.9753523822012E-2	-1.8044542924720E-3	6.9624582488270E-1
x_{10}	-2.0140947383266E-1	4.2827417026419E-3	-1.1360258844739E-1
x_{11}	2.6072812412241E-1	2.5503871726593E-3	-2.5656511082328E-1
x_{12}	9.3081124304593E-1	1.2275850230949E-3	-2.1908978242724E-1
μ_1	5.2346877744204E-1	1.1915063056069E-3	1.1915052278155E-3
μ_2	0.0	0.0	0.0
μ_3	7.1975778876265E-1	1.8874619749303E-3	1.8874624085896E-3
μ_4	0.0	0.0	0.0
μ_5	6.1266631862280	1.3188664619685E-2	1.3188664049757E-2

Here, $x_i, i = 1, \cdots, 12$, denote the filter parameters, and $\mu_j, j = 1, \cdots, 5$, represent the Lagrange multiplier values for the five constraints. Because the filter is symmetric, the number

of independent variables is half of the filter order of 24. Column 3 shows the gradient values of $\frac{dL}{dx_i}$ or $\frac{dL}{d\mu_i}$ obtained by applying Eq. (3.20). Column 4 shows the corresponding gradient values obtained using a difference equation (DE), i.e.,

$$\nabla_x f(x) = \frac{f(x + \delta x) \Leftrightarrow f(x \Leftrightarrow \delta x)}{2\delta x} \tag{4.1}$$

We let each variable perturb to both positive and negative directions by a small distance δx, and estimate the gradient value for that dimension by the resulting difference of function values. The selection of the perturbing distance is tricky: if it is too large, the estimation will be inaccurate; if it is too small, the value might be illuminated by the roundoff errors of the machine. By trial and error, we find out that a perturbing distance of 10^{-8} gives reliable gradient estimation.

A comparison of columns 3 and 4 shows that the discrepancy could be as large as in the order of 4. For this design, the gradients obtained from the Maxq method are small. This gives an illusive impression to the local search that a local optimal solution is approached, when it is actually far from the equilibrium point.

Due to these problems, we use the slack-variable method in the following experiments. Note that the designs we showed in Table 4.3 are designed using the Maxq method and have been verified; the impact of q is negligible for those two cases. These issues are further addressed in reference [66].

4.3 Weight on the Objective Function

As described in Section 4.1, the formulation with normalized constraints is better than the one without normalization. The purpose of normalizing constraints is to put an equal weight for each constraint, in order to eliminate the impact of its raw value. But when we normalize the objective function by the objective value of the reference design, as it is usually very small, the normalized function value could be very large. Hence, little disturbance of the variables might take the local search a very long time to get to a stable point.

The experimental results in Table 4.3 imply that reducing the weight on the objective function could decrease the local descent time. Thus, we put a scale factor α to the normalized

objective function:

$$\text{Obj. } f(x) \;=\; E_r/E_{r,J} * \alpha \tag{4.2}$$

To verify this hypothesis, we tried different scale factors for some filter bank designs. We show in Table 4.5 the 32D filter bank designs obtained with α values of 0.01 and 0.001, respectively.

Table 4.5: The local search time and the objective values of two sets of 32D filter bank designs. The weight of the objective function, α, for the two sets is 0.01 and 0.001, respectively

Local descent index	Local search time (sec)		Objective value (10^{-6})	
	$\alpha = 0.01$	$\alpha = 0.001$	$\alpha = 0.01$	$\alpha = 0.001$
1	7168.7	3368.5	5.169991	5.169993
2	18376.2	3816.4	5.169991	5.169990
3	4748.3	4106.2	5.169991	5.169991
4	16567.9	3766.0	5.169991	5.169996
5	31148.9	3452.7	5.169991	5.169992
6	16118.7	12182.9	5.169991	5.169992
7	6229.0	3303.3	5.169991	5.169990
7	31533.7	595.5	5.169992	5.169991
8	33064.7	11047.4	5.169991	5.169991
9	3914.8	4556.6	5.169991	5.169997
average	18763.4	5557.3	5.169991	5.169993

The properties of the first 9 local optimal solutions of each set are listed here. Table 4.5 shows that the formulation with smaller weight on the objective function spent less time searching for the local optimal solution, and the weight of the objective function has no impact on the quality of the design, as the objective and constraint values of both designs can be considered the same. Due to the size of the table, we don't show the constraint values here.

4.4 Global Search Results

In our experiments, we have applied Novel [65, 43], a global optimization method, to the filter bank design problem. There are two phases in Novel: global search stage and local search stage. A heuristic function, called the trace function (see Eq. (3.8)), is used in the global search stage. It is designed to pull the search trajectory out of a local region, so that we are not stuck with

a local solution. In a fairly long time, we should be able to find a good solution, which is or is close to the global solution.

When solving the 24D type filter bank design problem, we executed the program for 1340 logical time units. 344 local optimal solutions were found during that time. By looking at the design obtained from each local search, we find two outstanding solutions, shown in the table below.

Table 4.6: Two 24D filter bank designs obtained using Novel

Performance	Solution1	Solution2	Johnston's
E_r	0.78869230	0.75283526	4.86184131E-5
E_s	1.00000010	1.00000000	2.56407420E-4
E_p	0.83704002	0.76830929	4.35836979E-6
δ_s	1.00000000	1.00000000	3.26325786E-2
δ_p	1.00000000	1.00000000	4.12438061E-3
$\Delta\omega$	1.00000000	1.00000000	5.39899943E-1

It is clear from Table 4.6 that Solution 1 and Solution 2 all satisfy the constraints, and Solution 2 has 3.5% improvement of the objective function value. This shows that Novel can find multiple local optima. Note that all the performance values are normalized with respect to those of the corresponding Johnston's design. Also, in the following sections, we always present the normalized performance without explicit specification.

In Figure 4.1, we demonstrate the objective value for each local solution obtained within 500 logical time units. Here, the horizontal axis represents logical time, and the vertical axis stands for the objective value, i.e., the reconstruction error, of each local optimum. The figure shows that the local solutions oscillate between the two solutions given in Table 4.6. Note that all the solutions satisfy all the five constraints.

4.5 Local Search Time

We did experiments for three types of QMF filter banks. The prototype FIR filter length is 24, 32, and 48, respectively. As we have mentioned before, in order to eliminate the phase distortion of the filter bank design, the prototype filter has to be implemented as either symmetric or antisymmetric, so the optimization problem size is reduced by two. As we formulate five

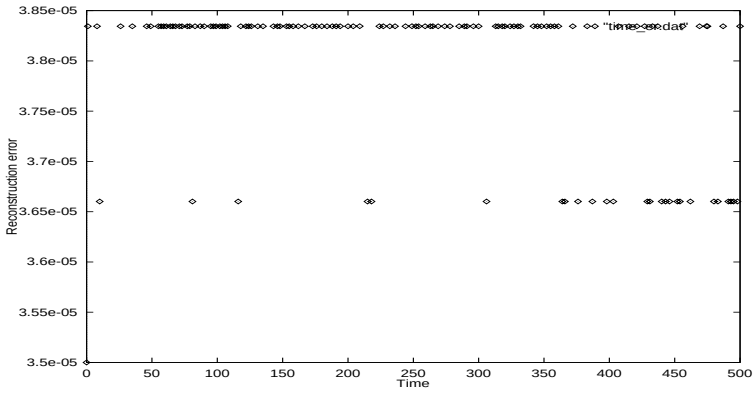

Figure 4.1: Reconstruction errors of 24D filter bank designs obtained within 500 logical time units

performance metrics as constraints, after adding the Lagrangian multipliers to the problem space, the problem dimension becomes 17, 21, and 29, respectively.

The problem size is not very large. But in our experiments of 24D filter bank designs, it generally took about 1200 seconds for each local descent. In Figure 4.2, the horizontal axis

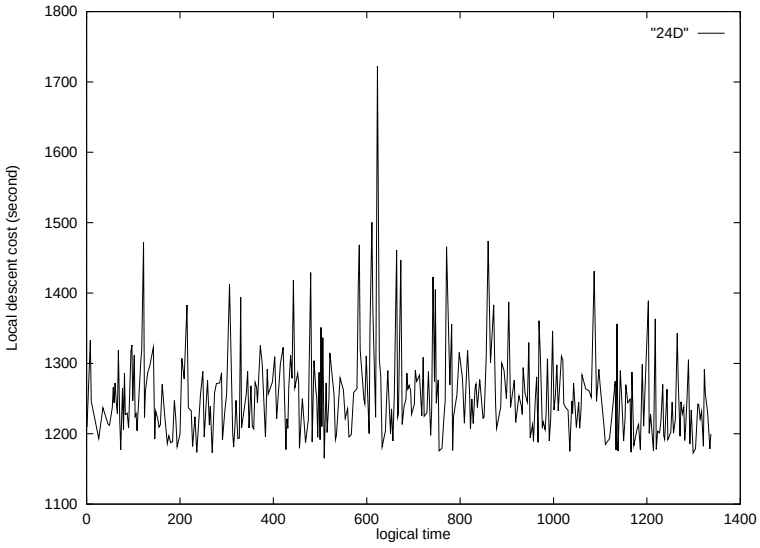

Figure 4.2: Local descent cost for the 24D filter bank designs

shows the logical time when Novel finds a promising starting point and LSODE is called for

each local descent. The vertical axis represents the CPU time (in seconds) consumed by each local descent.

In Figure 4.3, we show the calling frequency of the local search. Here, the horizontal axis

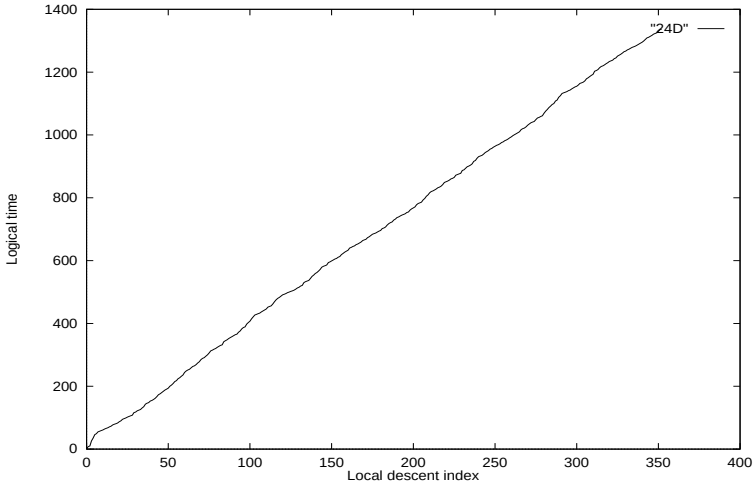

Figure 4.3: Local descent frequency for the 24D filter bank designs

denotes the index of local descents, and the vertical axis represents the logical time when the local search is started. For constrained problems, a Lyapunov function is defined to represent the sum of squares of Lagrangian gradient components. For instance, the Lyapunov function for Eq. (3.20) is:

$$L(x) = \sum_i (\frac{dL}{dx_i})^2 + \sum_j (\frac{dL}{d\mu_j})^2 \tag{4.3}$$

The smaller the Lyapunov function value is, the closer the current solution is to the optimal solution. In the global stage, the Lyapunov function is evaluated at each time unit, and the current function value is compared to that of the last time. If the Lyapunov function value becomes smaller, then a local search is started from the current solution.

Figure 4.3 shows that the curve is almost linear, meaning after approximately equal time periods, about every 4 logical time units, the trace function detects a new promising starting point and calls LSODE. This demonstrates that the trace function is able to push the global search to explore the whole search space; the search trajectory is not stuck inside a local terrain.

4.6 Result Analysis

In our experiments, we obtained filter bank designs that have improvement over Johnston's. In this section, we present our designs, and compare their performance with Johnston's designs.

4.6.1 Parameter comparison

The prototype filter parameters of our designs and Johnston's are given in this subsection. As each filter is symmetric, we only show the independent parameter values. For example, the beginning 12 parameters of the 24D filter banks are shown in Table 4.7.

Table 4.7: Filter parameters of 24D filter bank designs

Index	Our design	Johnston's
x_1	9.78368E-3	9.39685E-3
x_2	-2.09084E-2	-2.12280E-2
x_3	-3.20685E-3	-3.14284E-3
x_4	3.93088E-2	3.95276E-2
x_5	-7.72637E-3	-8.02756E-3
x_6	-6.59873E-2	-6.59168E-2
x_7	3.04950E-2	3.09479E-2
x_8	1.08343E-1	1.08060E-1
x_9	-8.12989E-2	-8.17844E-2
x_{10}	-2.00126E-1	-1.99688E-1
x_{11}	2.59799E-1	2.60224E-1
x_{12}	9.31250E-1	9.30858E-1

Column 2 represents the filter parameters of our design. We list Johnston's design in column 3 for comparison. Obviously, our design is very close to Johnston's. We demonstrate the filter parameters for 32D and 48D designs in Tables 4.8 and 4.9, respectively.

4.6.2 Comparison of performance metrics

We compare the performance of our designs and Johnston's in this subsection. We list the values of reconstruction error, stopband energy, passband energy, stopband ripple, passband ripple, and transition bandwidth of each design normalized with respect to the corresponding Johnston's design in Tables 4.10, 4.11, and 4.12.

Table 4.8: Filter parameters of 32D filter bank designs

Index	Our design	Johnston's
x_1	4.62704E-3	4.49028E-3
x_2	-7.73505E-3	-7.94230E-3
x_3	-3.90762E-3	-3.93934E-3
x_4	1.62637E-2	1.63639E-2
x_5	1.66690E-3	1.68537E-3
x_6	-2.84369E-2	-2.84580E-2
x_7	4.06670E-3	4.13894E-3
x_8	4.54622E-2	4.55883E-2
x_9	-1.58241E-2	-1.59235E-2
x_{10}	-6.99564E-2	-6.99288E-2
x_{11}	3.87978E-2	3.89444E-2
x_{12}	1.09667E-1	1.09624E-1
x_{13}	-8.89155E-2	-8.90485E-2
x_{14}	-1.98712E-1	-1.98677E-1
x_{15}	2.65868E-1	2.65945E-1
x_{16}	9.27359E-1	9.27348E-1

Table 4.10 shows that all the five constraints (stopband energy, passband energy, stopband ripple, passband ripple and transition bandwidth) are satisfied, meaning that our design has smaller or the same values compared to those of Johnston's. Further, the objective value of our design is 24.7% better than Johnston's. A more careful look at the performance values shows that except for the passband energy, which is 23.2% smaller, the other four constraints are exactly the same as Johnston's. From the point of view of constrained optimization, our solution is on the boundary defined by those four constraints.

Tables 4.11 and 4.12 show the performance of 32D and 48D designs.

Similarly, both our 32D and 48D filter bank designs have improvement in the objective function, with improvement ratios of the reconstruction error of 13.0% and 5.3%, respectively. Also, both designs are inside the boundary defined by the passband energy, and on the boundaries defined by the other four constraints.

4.6.3 Comparison of frequency response

To illustrate the performance improvement of our design, the frequency response of each type of filter bank design is given in Figures 4.4, 4.5, and 4.6, respectively.

Table 4.9: Filter parameters of 48D filter bank designs

Index	Our design	Johnston's
x_1	8.72569E-4	8.16587E-4
x_2	-1.25106E-3	-1.22166E-3
x_3	-1.29455E-3	-1.32943E-3
x_4	2.98186E-3	3.00331E-3
x_5	1.76584E-3	1.79958E-3
x_6	-5.87310E-3	-5.92227E-3
x_7	-1.89620E-3	-1.91184E-3
x_8	1.02392E-2	1.02979E-2
x_9	1.27599E-3	1.27295E-3
x_{10}	-1.64312E-2	-1.64949E-2
x_{11}	6.52032E-4	6.65854E-4
x_{12}	2.48763E-2	2.49314E-2
x_{13}	-4.67379E-3	-4.71493E-3
x_{14}	-3.62066E-2	-3.62438E-2
x_{15}	1.19960E-2	1.20453E-2
x_{16}	5.16032E-2	5.16263E-2
x_{17}	-2.47898E-2	-2.48451E-2
x_{18}	-7.38005E-2	-7.38127E-2
x_{19}	4.79935E-2	4.80401E-2
x_{20}	1.10759E-1	1.10758E-1
x_{21}	-9.74078E-2	-9.74623E-2
x_{22}	-1.96885E-1	-1.96876E-1
x_{23}	2.72746E-1	2.72796E-1
x_{24}	9.22793E-1	9.22790E-1

Table 4.10: Performance comparison of 24D filter bank designs

Performance	Our design	Johnston's
E_r	0.75283526	4.86184131E-5
E_s	1.00000000	2.56407420E-4
E_p	0.76830929	4.35836979E-6
δ_s	1.00000000	3.26325786E-2
δ_p	1.00000000	4.12438061E-3
$\Delta\omega$	1.00000000	5.39899943E-1

Table 4.11: Performance comparison of 32D filter bank designs

Performance	Our design	Johnston's
E_r	0.87011070	5.94176292E-6
E_s	0.99999998	3.86006268E-5
E_p	0.80201908	5.82918260E-7
δ_s	1.00000000	1.47359387E-2
δ_p	1.00000000	1.83488395E-3
$\Delta\omega$	1.00000000	5.03368844E-1

The frequency response of the 32D and 48D types is similar, i.e., we have smoother amplitude response around $\omega = 0$, except that the performance improvement ratios are different. As we mentioned in the last section, the reconstruction error improves 24.7%, 13.0%, and 5.3% for 24D, 32D and 48D types, respectively.

For each type of filter bank design, as the frequency response is symmetric with respect to the vertical axis, we only draw the function values in the frequency range from 0 to π. The top left diagram shows the amplitude response. The top right figure is also amplitude response

Table 4.12: Performance comparison of 48D filter bank designs

Performance	Our design	Johnston's
E_r	0.94733624	5.93631985E-7
E_s	0.99999995	6.89225662E-7
E_p	0.75598972	3.54486440E-8
δ_s	1.00000000	2.48187108E-3
δ_p	1.00000000	4.80833895E-4
$\Delta\omega$	1.00000000	4.72106925E-1

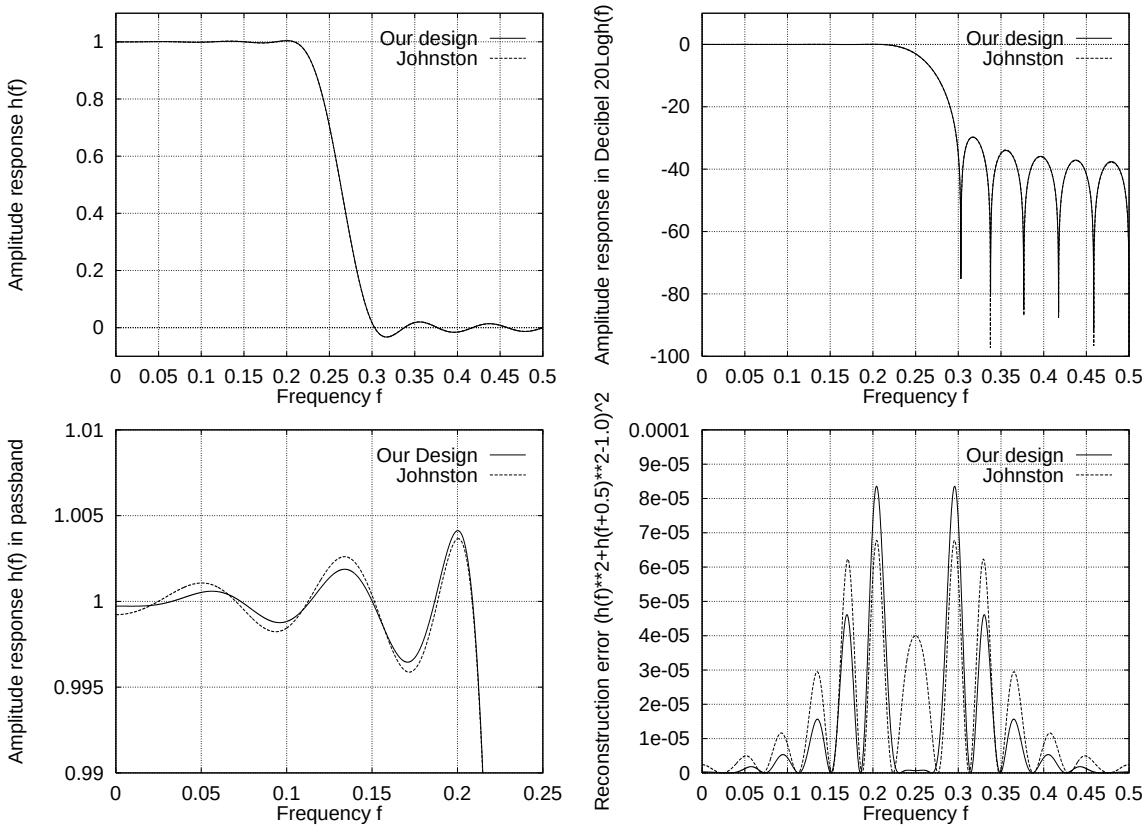

Figure 4.4: Frequency response of 24D filter bank designs. The solid line stands for our design, and the dashed line represents Johnston's

with the Y-axis in terms of decibel. We see the details of the stopband, and compare the discrepancy of stopband ripples and stopband energies. The bottom left diagram is a zoom-in of the amplitude response within the passband. The bottom right figure illustrates the integrand of the reconstruction error. We can estimate the reconstruction error by computing the area under the curve.

Note that In the top left and top right figures, the dashed line curves are almost invisible. Because the stopband energy, stopband ripple, passband ripple, and transition bandwidth of our designs are almost the same as Johnston's, the solid line curves overlap with the dashed line curves in those diagrams. In the passband, our design is much smoother around DC than that of Johnston's. This feature results in smaller passband energy. Similarly, although we have higher pulses of side-lobes, a smooth DC response causes the improvement in reconstruction

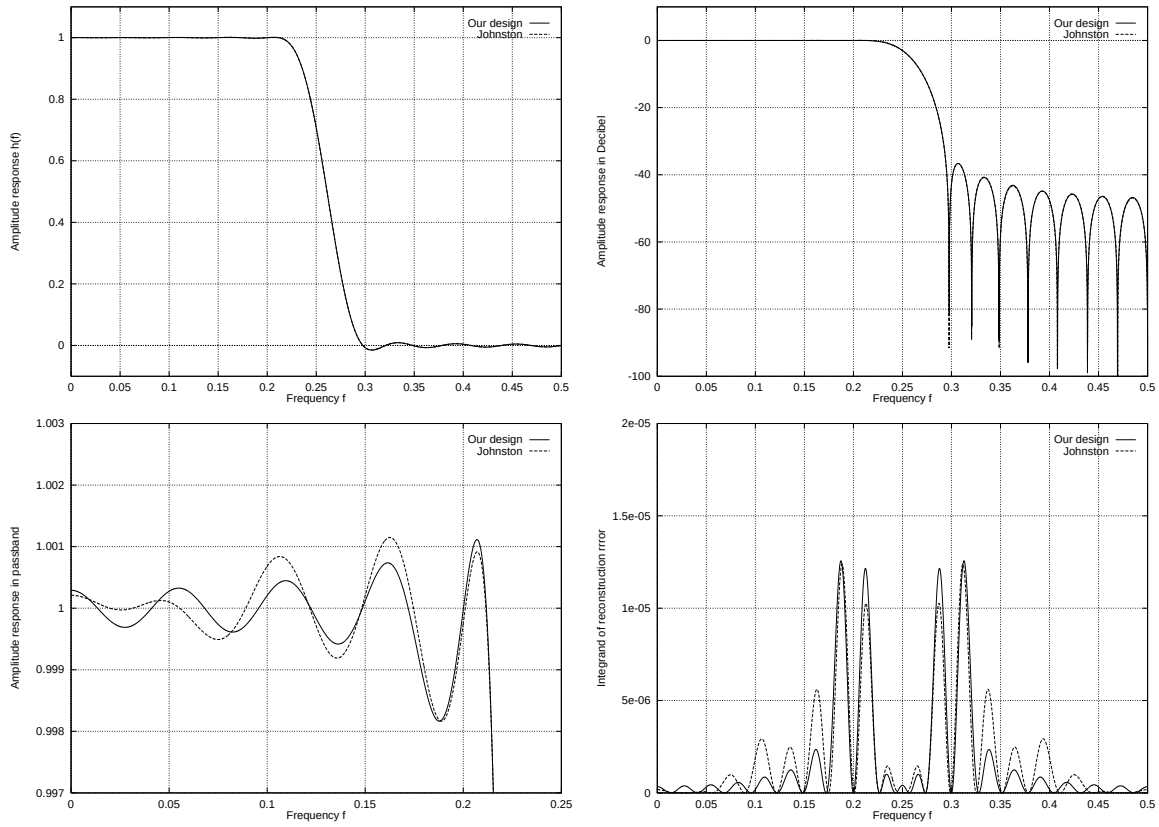

Figure 4.5: Frequency response of 32D filter bank designs. The solid line stands for our design, and the dashed line represents Johnston's

error. This is consistent with our conclusion from comparison of the performance values of our designs and Johnston's.

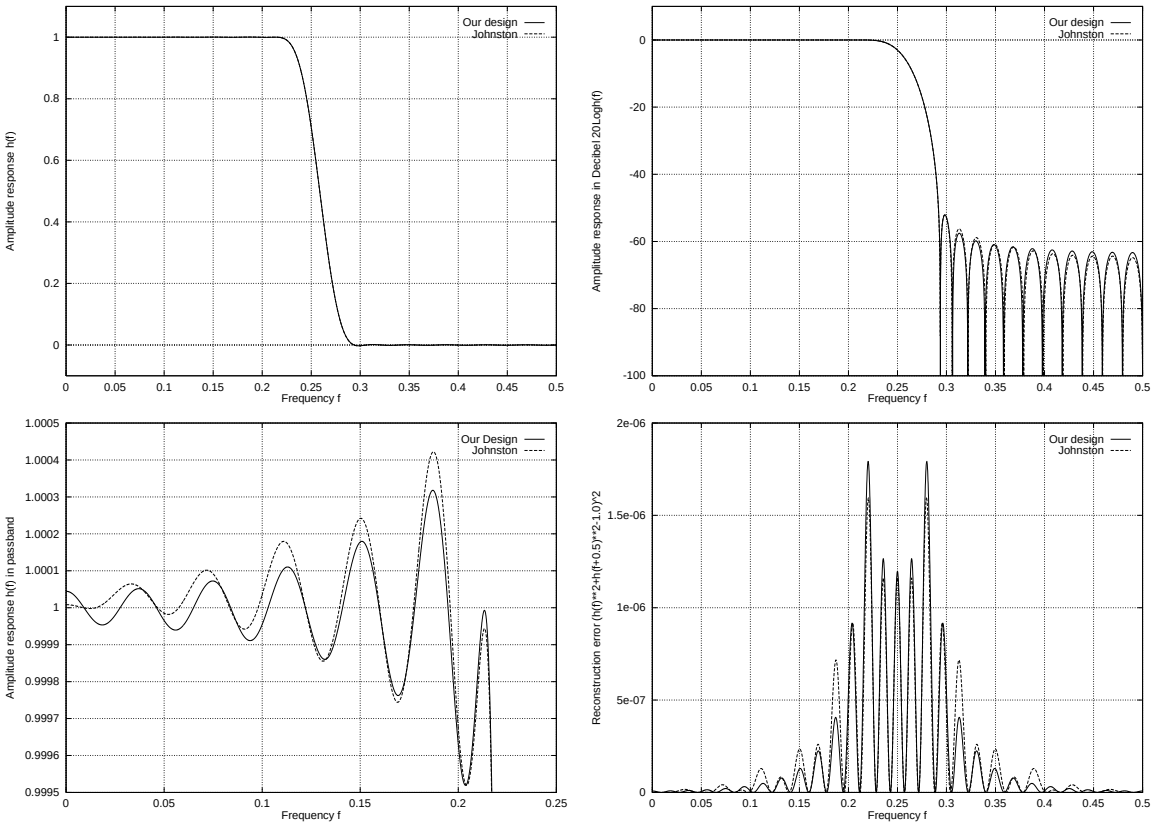

Figure 4.6: Frequency response of 48D filter bank designs. The solid line stands for our design, and the dashed line represents Johnston's

4.6.4 More results

In this subsection, we present our experimental results for some other filter bank types, including 24B, 24C, 32C, 32E, 48C, and 48E. The number 24, 32 or 48 stands for the filter bank order, i.e., the length of the individual filter. B, C, and E represent filter bank type. The objective is to show that our design scheme can be extended to general 2-channel filter bank design problems.

Table 4.13: Filter parameters of 24B and 24C filter bank designs

24B	Our design	Johnston's	24C	Our design	Johnston's
x_1	7.36327E-4	7.66619E-4	x_1	4.79542E-3	4.65853E-3
x_2	-2.64416E-3	-2.78582E-3	x_2	-1.02996E-2	-1.03660E-2
x_3	-2.71512E-3	-2.74772E-3	x_3	-4.44081E-3	-4.54629E-3
x_4	1.26626E-2	1.29718E-2	x_4	2.69059E-2	2.70802E-2
x_5	2.96826E-3	2.89292E-3	x_5	-1.24195E-3	-1.30093E-3
x_6	-3.76430E-2	-3.80399E-2	x_6	-5.50326E-2	-5.51039E-2
x_7	7.53762E-3	7.78304E-3	x_7	1.99390E-2	2.00924E-2
x_8	8.81358E-2	8.84795E-2	x_8	1.01799E-1	1.01763E-1
x_9	-5.08095E-2	-5.12307E-2	x_9	-6.91493E-2	-6.92829E-2
x_{10}	-1.96479E-1	-1.96596E-1	x_{10}	-1.99809E-1	-1.99758E-1
x_{11}	2.31635E-1	2.32071E-1	x_{11}	2.49223E-1	2.49290E-1
x_{12}	9.46439E-1	9.46258E-1	x_{12}	9.37302E-1	9.37296E-1

Table 4.14: Performance comparison of 24B and 24C filter bank designs

24B	Our design	Johnston's	24C	Our design	Johnston's
E_r	0.96367907	7.45865200E-7	E_r	0.90958870	3.69628749E-6
E_s	1.00000047	4.45402917E-8	E_s	1.00000000	2.20366616E-5
E_p	0.80117231	4.81675255E-8	E_p	0.76955987	2.77817874E-7
δ_s	1.00000000	6.31364062E-4	δ_s	1.00000000	9.92259253E-3
δ_p	1.00000000	3.66308150E-4	δ_p	1.00000000	1.10518942E-3
$\Delta\omega$	1.00000000	1.13597728	$\Delta\omega$	1.00000000	7.34836847E-1

It is clear from Table 4.14 that we have improvement to Johnston's 24B and 24C filter bank designs. Recall the performance of our 24D filter bank design (see Table 4.10), we see that the improvement ratios of reconstruction error (E_r) for those three types are 4.4%, 9.4%, and 24.7%, respectively. Note that 24B, 24C, and 24D all have filter length of 24, but the transition

Table 4.15: Filter parameters of 32C and 32E filter bank designs

32C	Our design	Johnston's	32E	Our design	Johnston's
x_1	1.44238E-3	1.38212E-3	x_1	1.17941E-2	1.02465E-2
x_2	-2.83810E-3	-2.80759E-3	x_2	-2.12952E-2	-2.25513E-2
x_3	-2.46698E-3	-2.53661E-3	x_3	-2.84853E-3	-1.92492E-3
x_4	8.40314E-3	8.46839E-3	x_4	3.05701E-2	3.13616E-2
x_5	2.79209E-3	2.82849E-3	x_5	-4.13956E-3	-5.22408E-3
x_6	-1.88189E-2	-1.89166E-2	x_6	-4.18629E-2	-4.20765E-2
x_7	-2.82369E-4	-2.60772E-4	x_7	1.35687E-2	1.47598E-2
x_8	3.58804E-2	3.59629E-2	x_8	5.64816E-2	5.62452E-2
x_9	-8.31393E-3	-8.37497E-3	x_9	-2.78330E-2	-2.91380E-2
x_{10}	-6.24264E-2	-6.24772E-2	x_{10}	-7.73005E-2	-7.66123E-2
x_{11}	2.90631E-2	2.91369E-2	x_{11}	5.18448E-2	5.32486E-2
x_{12}	1.05894E-1	1.05895E-1	x_{12}	1.12481E-1	1.11414E-1
x_{13}	-7.86285E-2	-7.86976E-2	x_{13}	-1.01391E-1	-1.02765E-1
x_{14}	-1.99614E-1	-1.99605E-1	x_{14}	-1.96656E-1	-1.95368E-1
x_{15}	2.57049E-1	2.57116E-1	x_{15}	2.76253E-1	2.77528E-1
x_{16}	9.32820E-1	9.32811E-1	x_{16}	9.20543E-1	9.19291E-1

bandwidth reduces in the order of B, C, and D. So we have more improvement for the filter bank type with sharper transition bandwidth.

Table 4.16: Performance comparison of 32C and 32E filter bank designs

32C	Our design	Johnston's	32E	Our design	Johnston's
E_r	0.95862635	1.00356001E-6	E_r	0.52989534	3.33253741E-4
E_s	1.00000001	1.01774616E-6	E_s	1.00099059	6.45958608E-4
E_p	0.74727092	5.30133462E-8	E_p	0.62451611	2.63578553E-5
δ_s	1.00000000	2.48695746E-3	δ_s	1.00057496	5.76649847E-2
δ_p	1.00000000	5.12305122E-4	δ_p	1.00096035	1.17760856E-2
$\Delta\omega$	1.00000000	6.99339105E-1	$\Delta\omega$	1.00998856	3.15415454E-1

The performance of our 32C and 32E filter bank designs is similar to that of our 24s' filter bank designs. Table 4.16 and Table 4.11 also show that the improvement ratio of reconstruction error increases when the design's transition bandwidth becomes sharper.

We didn't run the programs for 48C and 48E designs to the end, because the local search didn't converge after about two and a half days (wallclock time). We made snapshots of the local search periodically, and dumped variable values and gradient component values at each

Table 4.17: Filter parameters of 48C and 48E filter bank designs

48C	Our design	Johnston's	48E	Our design	Johnston's
x_1	2.86830E-5	-2.92381E-5	x_1	5.29942E-3	5.08086E-3
x_2	-2.55089E-4	-1.84110E-4	x_2	-7.06778E-3	-7.53341E-3
x_3	-1.28848E-4	-1.12315E-4	x_3	-3.46926E-3	-3.38197E-3
x_4	7.02286E-4	6.02345E-4	x_4	1.10923E-2	1.12896E-2
x_5	4.55943E-4	4.84504E-4	x_5	2.00039E-3	1.98249E-3
x_6	-1.81079E-3	-1.70588E-3	x_6	-1.56132E-2	-1.57055E-2
x_7	-9.56961E-4	-1.04111E-3	x_7	1.42816E-5	5.09002E-5
x_8	4.12282E-3	4.06803E-3	x_8	2.09277E-2	2.09065E-2
x_9	1.42946E-3	1.56668E-3	x_9	-2.93377E-3	-3.00988E-3
x_{10}	-8.38414E-3	-8.43077E-3	x_{10}	-2.72357E-2	-2.71866E-2
x_{11}	-1.35890E-3	-1.51230E-3	x_{11}	7.13180E-3	7.24849E-3
x_{12}	1.55139E-2	1.56806E-2	x_{12}	3.49374E-2	3.48744E-2
x_{13}	-2.17028E-4	-1.22340E-4	x_{13}	-1.31890E-2	-1.33923E-2
x_{14}	-2.66097E-2	-2.68832E-2	x_{14}	-4.46417E-2	-4.45714E-2
x_{15}	4.90861E-3	4.92536E-3	x_{15}	2.20595E-2	2.23205E-2
x_{16}	4.31529E-2	4.34722E-2	x_{16}	5.77087E-2	5.76276E-2
x_{17}	-1.54433E-2	-1.56033E-2	x_{17}	-3.57696E-2	-3.60788E-2
x_{18}	-6.78974E-2	-6.81804E-2	x_{18}	-7.69801E-2	-7.68845E-2
x_{19}	3.73401E-2	3.76190E-2	x_{19}	5.90689E-2	5.93504E-2
x_{20}	1.10850E-1	1.08652E-1	x_{20}	1.10732E-1	1.10615E-1
x_{21}	-8.68431E-2	-8.71928E-2	x_{21}	-1.07721E-1	-1.07981E-1
x_{22}	-1.19879E-1	-1.98769E-1	x_{22}	-1.93613E-1	-1.93456E-1
x_{23}	2.26384E-1	2.64158E-1	x_{23}	2.81459E-1	2.81647E-1
x_{24}	9.92865E-1	9.28483E-1	x_{24}	9.16498E-1	9.16359E-1

Table 4.18: Performance comparison of 48C and 48E filter bank designs

48C	Our design	Johnston's	48E	Our design	Johnston's
E_r	0.71369417	9.96137964E-8	E_r	0.85169242	3.33648191E-5
E_s	0.94954359	1.04709451E-8	E_s	1.00001332	1.08376791E-4
E_p	0.68521390	7.29422767E-9	E_p	0.83918304	3.11932643E-6
δ_s	0.75665926	4.11935425E-4	δ_s	1.00000470	2.95068419E-2
δ_p	0.92556413	1.81060739E-4	δ_p	1.00000167	4.57616541E-3
$\Delta\omega$	1.02479425	7.18020445E-1	$\Delta\omega$	1.00001673	2.71373933E-1

time period. We choose the filter parameter values from the last snapshot of each run, and list them in the Table 4.17.

Table 4.18 shows that for 48C filter bank design, we have improvement for the upper five performance measures, but our transition bandwidth is worse than Johnston's. Not surprisingly, our design is a tradeoff. The transition bandwidth constraint is not satisfied because the local search has not converged yet. For 48E filter bank design, the result we got has smaller or the same performance values for all the six measures. We expect better solution if given more CPU time.

All the above experiments were run on Sun UltraSparc1 workstations. We list the execution time for each filter bank design in Table 4.19. The asterisks in columns 48C and 48E mean

Table 4.19: Experiments execution time

CPU time	24			32			48		
second	24B	24C	24D	32C	32D	32E	48C*	48D	48E*
global search	4278.5	192.4	13.1	478.7	147.6	123.5	40581.2	2375.0	799.1
local search	119891.0	11546.9	733.3	826.7	4205.2	2059.5	184162.0	2754.8	173279.0

that the CPU time shown is the time of getting the results listed in Table 4.17, but the local search didn't converge by that time.

Note that the programs were not run on dedicated machines, so the execution time consumed by each run depends on the CPU load at the running time. Table 4.19 is just for reference.

4.7 Summary

In this chapter, we present our experimental results for the 2-channel QMF FIR filter bank designs. We discuss the impacts of the constraint formulation, the weight on the objective function, and the inequality constraint handling methods. Our results show that the Novel method can detect promising local search starting points, and the trace function can lead the trajectory to traverse the whole problem space. In the end, we compare the performance of our designs with Johnston's for some filter types. It is clear that we have improvement in the objective function, which is the reconstruction error, and have either the same or better performance for the other five measures. The frequency response verifies our performance analysis.

Chapter 5

Conclusion

In this research, we have applied nonlinear optimization methods to the QMF FIR filter bank design problem. We formulate the filter bank design as a single-objective multiple-constraint optimization problem.

To evaluate the performance of a filter bank design, the performance measures for both the overall filter bank and the composing filters need to be considered. Reconstruction error, alias distortion, and phase distortion are metrics used for the filter bank, and stopband energy, passband energy, stopband ripple, passband ripple, and transition bandwidth are used for a single filter. As our research focuses on the QMF FIR filter, by choosing the prototype filter to be symmetric, alias and phase distortion are eliminated. Thus we define the reconstruction error as our design objective, and set all the other measure as constraints.

Because reconstruction error, stopband energy, and passband energy are all expressed as integration, numerical integration causes much computation overhead. We derived closed-form formulae for both those functions and their derivatives in order to achieve high precision and to reduce computation cost. Comparison to an integration package shows that the derivation is correct.

We applied NOVEL, a new nonlinear optimization package, to the 2-channel QMF filter bank design. The experimental results show that our designs have improvement over the best known solution.

The results of the QMF filter bank design can be generalized to other filter bank design problems. Non-QMF filter bank design, IIR filter bank design, and multi-channel filter bank design are topics for future research.

Bibliography

[1] E. Abdel-Raheem, El Guibaly, and A. Antoniou. Design of low-delay FIR QMF banks using the Lagrange-multiplier approach. *Electronics Letters*, 30(12):924–1430, June 1994.

[2] O. Alkin and H. Caglar, "Design of efficient m-band coders with linear-phase and perfect-reconstruction properties," *IEEE Trans. on Signal Processing*, vol. 43, pp. 1579–1590, July 1995.

[3] S. Basu, C.-H. Chiang, and H. M. Choi. Causal IIR perfect reconstruction subband coding. In *Proc. of 1993 Int'l Symposium on Circuit and Systems*, pp. 367–370, 1993.

[4] S. Basu, C.-H. Chiang, and H. M. Choi. Wavelets and perfect reconstruction subband coding with causal stable IIR filters. *IEEE Trans. on Circuits and Systems II*, 42(1):24–38, January 1995.

[5] C.-K. Chen and J.-H. Lee. Design of quadrature mirror filters with linear phase in the frequency domain. *IEEE Trans. on Circuits and Systems II*, 39(9):593–605, September 1992.

[6] T. Chen and P. P. Vaidyanathan, "Recent developments in multidimensional multirate systems," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 3, pp. 116–137, Apr. 1993.

[7] C. D. Creusere and S. K. Mitra. A simple method for designing high-quality prototype filters for m-band pseudo QMF banks. *IEEE Trans. on Signal Processing*, 43(4):1005–1007, April 1995.

[8] Z. Doganata, P. P. Vaidyanathan, and T. Q. Nguyen. General synthesis procedures for FIR lossless transfer matrices, for perfect-reconstruction multirate filter bank applications. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 36(10):1561–1574, October 1988.

[9] M. M. Ekanayake and K. Premaratne. Tow-channel IIR QMF banks with approximately linear phase analysis and synthesis filters. *IEEE Trans. on Signal Processing*, 43(10):2313–2322, October 1995.

[10] M. H. Er and C. K. Siew. Design of FIR filters using quadrature programming approach. *IEEE Trans. on Circuits and Systems II*, 42(3):217–220, March 1995.

[11] N. J. Fliege, *Multirate Digital Signal Processing.* John Wiley and Sons, 1994.

[12] G. Gu and B. A. Shenoi. A novel approach to the synthesis of recursive digital filters with linear phase. *IEEE Trans. on Circuits and Systems*, 38(6):602–612, June 1991.

[13] M. T. Hanna, "Design of linear phase fir filters with a maximally flat passband," *IEEE Trans. on Circuits and Systems II*, vol. 43, pp. 142–147, Feb. 1996.

[14] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *IEEE Symposium on Circuits and Systems*, pp. 2264–2267, 1991.

[15] B. R. Horng and A. N. Willon, Jr. Lagrange multiplier approaches to the design of two-channel perfect-reconstruction linear-phase FIR filter banks. *IEEE Trans. on Signal Processing*, 40(2):364–374, February 1992.

[16] J. H. Husoy and T. A. Ramstad. Application of an efficient parallel IIR filter bank to image subband coding. *Signal Processing*, 20:279–292, August 1990.

[17] V. K. Jain and R. E. Crochiere. Quadrature mirror filter design in the time domain. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 32(2):353–361, April 1984.

[18] B. Jaworski and T. Saramaki. Linear phase IIR filters composed of two parallel allpass sections. In *Proc. of 1994 Int'l Symposium on Circuit and Systems*, pp. 537–540, 1994.

[19] J. D. Johnston. A filter family designed for use in quadrature mirror filter banks. In *Proc. of Int'l Conf. on ASSP*, pp. 291–294, 1980.

[20] P. P. Khargonekar and M. A. Rotea, "Multiple objective optimal control of linear systems: The quadratic norm case," *IEEE Trans. on Automatic Control*, vol. 36, pp. 14–24, Jan. 1991.

[21] R. D. Koilpillai and P. P. Vaidyanathan. Cosine modulated FIR filter banks satisfying perfect reconstruction. *IEEE Trans. on Signal Processing*, 40(4):770–783, April 1992.

[22] R. D. Koilpillai and P. P. Vaidyanathan. A spectral factorization aparoach to pesudo-QMF design. *IEEE Trans. on Signal Processing*, 41(1):82–92, January 1993.

[23] Y.-P. Lin and P. P. Vaidyanathan, "Linear phase cosine modulated maximally decimated filter banks with perfect reconstruction," *IEEE Trans. on Signal Processing*, vol. 42, pp. 2525–2539, Nov. 1995.

[24] D. G. Luenberger, *Linear and Nonlinear Programming.* Addison-Wesley Publishing Company, 1984.

[25] S. K. Mitra, C. D. Creusere, and H. Babic. A novel implementation of perfect reconstruction QMF banks using IIR filters for infinite length signals. In *Proc. of 1992 Int'l Symposium on Circuit and Systems*, pp. 2312–2315, 1992.

[26] M. Mollaghasemi and G. W. Evans, "Multicriteria design of manufacturing systems through simulation optimization," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 24, pp. 1407–1411, Sept. 1994.

[27] K. Nayebi, T. P. B. III, and M. J. T. Smith, "Time-domain filter bank analysis: A new design theory," *IEEE Transactions on Signal Processing*, vol. 40, pp. 1412–1429, June 1992.

[28] K. Nayebi, T. P. Barnwell III, and M. J. T. Smith, "Nonuniform filter banks: A reconstruction and design theory," *IEEE Trans. on Signal Processing*, vol. 41, pp. 1114–1127, Mar. 1993.

[29] K. Nayebi, T. P. Barnwell III, and M. J. T. Smith. Low delay FIR filter banks: Design and evaluation. *IEEE Trans. on Signal Processing*, 42(1):24–31, January 1994.

[30] T. Q. Nguyen, "Near-perfect reconstruction pseudo-QMF banks," *IEEE Trans. on Signal Processing*, vol. 42, pp. 65–76, Jan. 1994.

[31] T. Q. Nguyen. Digital filter bank design quadratic-constrained formulation. *IEEE Trans. on Signal Processing*, 43(9):2103–2108, September 1995.

[32] T. Q. Nguyen, T. I. Laakso, and R. D. Koilpillai. Eigenfilter approach for the design of allpass filters approximating a given phase response. *IEEE Trans. on Signal Processing*, 42(9):2257–2263, September 1994.

[33] T. Q. Nguyen, T. I. Laakso, and T. E. Tuncer. On perfect-reconstruction allpass-based cosine-modulated IIR filter banks. In *Proc. of 1994 Int'l Symposium on Circuit and Systems*, pp. 33–36, 1994.

[34] T. Q. Nguyen and P. P. Vaidyanathan. Two-channel perfect-reconstruction FIR QMF structures which yield linear-phase analysis and synthesis filters. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(5):676–690, May 1989.

[35] T. Q. Nguyen and P. P. Vaidyanathan, "Structures for m-channel perfect-reconstruction FIR QMF banks which yield linear-phase analysis filters," *IEEE Trans. on Acoutics, Speech, and Signal Processing*, vol. 38, pp. 433–446, Mar. 1990.

[36] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing.* Prentice Hall, 1989.

[37] W. Pennebaker and J. Mitchell, *JPEG still image data compression standard.* Van Nostrand Reinhold, 1993.

[38] J. Princen, "The design of nonuniform modulated filterbanks," *IEEE Trans. on Signal Processing*, vol. 43, pp. 2550–2560, Nov. 1995.

[39] J. G. Proakis. *Digital Signal Processing: Principles, Algorithms, and Applications.* Maxwell Macmillan, 1992.

[40] J. Radecki, J. Konrad, and E. Dubois, "Design of multidimensional finite-wordlength FIR and IIR filters by simulated annealing," *IEEE Trans. on Circuits and Systems II*, vol. 42, pp. 424–431, June 1995.

[41] T. A. Ramstad. IIR filterbank for subband coding of images. In *Proc. of 1988 Int'l Symposium on Circuit and Systems*, pages 827–830, 1988.

[42] J. L. Ringuest and T. R. Gulledge, "An interactive multi-objective gradient search," *Operations Research Letters*, vol. 12, pp. 53–58, July 1992.

[43] Y. Shang and B. W. Wah, "Global optimization for neural network training," *IEEE Computer*, vol. 29, pp. 45–54, Mar. 1996.

[44] B. A. Shenoi and P. Misra. Design of two-dimensional IIR digital filters with linear phase. *IEEE Trans. on Circuits and Systems II*, 42(2):124–129, February 1995.

[45] J.-J. Shyu. Design of two-channel perfect-reconstruction linear-phase filter banks for subband image coding by the Lagrange multiplier approach. *IEEE Trans. on Circuits and Systems for Video Technology*, 5(1):48–51, February 1995.

[46] M. J. T. Smith and T. P. Barnwell III. Exact reconstruction techniques for tree-structured subband coders. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 34(3):434–441, June 1986.

[47] Iraj Sodagar, Kambiz Naybei, and Thomas P. Barnwell III. Time-varying filter banks and wavelets. *IEEE Trans. on Signal Processing*, 42(11):2983–2996, November 1994.

[48] A. K. Soman, P. P. Vaidyanathan, and T. Q. Nguyen. Linear phase paraunitary filter banks: Theory, factorizations and designs. *IEEE Trans. on Signal Processing*, 41(12):3480–3496, December 1993.

[49] A. A. Song, A. Mathur, and K. R. Pattipati, "Design of process parameters using robust design techniques and multiple criteria optimization," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 25, pp. 1437–1446, Nov. 1995.

[50] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation and Application.* Krieger Publishing Company, 1989.

[51] T. E. Tuncer and T. Q. Nguyen. M-channel approximately linear phase IIR filters using allpass subfilters. In *Proc. of 1993 Int'l Symposium on Circuit and Systems*, pages 1315–1319, 1993.

[52] T. E. Tuncer and T. Q. Nguyen. General analysis of two-band QMF banks. *IEEE Trans. on Signal Processing*, 43(2):544–548, February 1995.

[53] T. E. Tuncer and T. Q. Nguyen. Interpolated IIR mth-band filter design with allpass subfilters. *IEEE Trans. on Signal Processing*, 43(8):1986–1990, August 1995.

[54] P. P. Vaidyanathan. Multirate digital filters, filter banks, polyphase networks, and applications: A tutorial. *Proc. of the IEEE*, 78(1):56–93, January 1990.

[55] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Printice-Hall Inc., 1993.

[56] P. P. Vaidyanathan and T. Chen. Structure for time-reversed inversion in filter banks. In *Proc. of 1995 Int'l Symposium on Circuit and Systems*, pages 585–588, 1995.

[57] P. P. Vaidyanathan and P. Q. Hoang. Lattice structures for optimal design and robust implementation of two-channel perfect-reconstruction QMF banks. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 36(1):81–94, January 1988.

[58] D. Vanderpooten and P. Vincke, "Description and analysis of some representative interactive multicriteria procedures," in *Models and Methods in Multiple Criteria Decision Making* (C. D. B. G. Colson and E. Y. Rodin, eds.), pp. 1221–1238, Pergamon Press, 1989.

[59] M. Vetterli. Filter banks allowing perfect reconstruction. *Signal Processing*, 10(3):219–244, April 1986.

[60] M. Vetterli. A theory of multirate filter banks. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 35(3):356–372, March 1987.

[61] M. Vetterli and D. L. Gall, "Perfect reconstruction FIR filter banks: Some properties and factorizations," *IEEE Trans. on Acoutics, Speech, and Signal Processing*, vol. 37, pp. 1057–1071, July 1989.

[62] M. Vetterli and D. Le Gall. Perfect reconstruction filter banks: Some properties and factorizations. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(7):1057–1071, July 1989.

[63] B. W. Wah, *The Ubiquitous Search*. Toulouse, France: Keynote Speech, Int'l Conf. on Tools with Artificial Intelligence, Nov. 16, 1996.

[64] S. Wada, "Design of nonuniform division multirate FIR filter banks," *IEEE Trans. on Circuits and Systems II Analog and Digital Signal Processing*, vol. 42, pp. 115–121, Feb. 1995.

[65] B. W. Wah and Y.-J. Chang, *Trace-Based Methods for Solving Nonlinear Global Optimization Problems.* J. of Global Optimization, (accepted to appear) 1996.

[66] T. Wang and B. W. Wah, "Handling inequality constraints in continuous nonlinear global optimization," in *Proc. 2nd World Conference on Integrated Design and Process Technology*, vol. 1, (Austin, TX), pp. 267–274, Dec. 1996.

[67] R. H. Yang and Y. C. Lim. Novel efficient approach for the design for equiripple quadrature mirror filters. *IEE Proc. on Vision, Image Signal and Processing*, 141(2):95–100, April 1994.

[68] X. Zhang and H. Iwakura. Design of QMF banks using allpass filters. *Electronics Letters*, 31(3):172–174, February 1995.

Vita

Ting Yu received her B.E. and M.E. degrees from the Department of Automation, Tsinghua University in 1991 and 1994, respectively. She is expecting a M.S. degree in Computer Science from the University of Illinois at Urbana-Champaign in May 1997.

Her interests include networking, distributed systems, optimization, and parallel computing.