Analysis of Algorithms for Filter Bank Design Optimization

By

Ahmed ElGarewi
B.S., University of Tripoli, Tripoli, Libya, 2013

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

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

Analysis of Algorithms for Filter Bank Design Optimization

By

Ahmed ElGarewi
B.S., University of Tripoli, Tripoli, Libya, 2013

Supervisory Committee

Dr. Pan Agathoklis, Supervisor

(Department of Electrical and Computer Engineering)

Dr. Wu-Sheng Lu, Departmental Member

(Department of Electrical and Computer Engineering)

Supervisory Committee

Dr. Pan Agathoklis, Supervisor

(Department of Electrical and Computer Engineering)

Dr. Wu-Sheng Lu, Departmental Member

(Department of Electrical and Computer Engineering)

Abstract

This thesis deals with design algorithms for filter banks based on optimization. The design specifications consist of the perfect reconstruction and frequency response specifications for finite impulse response (FIR) analysis and synthesis filters. The perfect reconstruction conditions are formulated as a set of linear equations with respect to the analysis filters' coefficients and the synthesis filters' coefficients. Five design algorithms are presented. The first three are based on an unconstrained optimization of performance indices, which include the perfect reconstruction error and the error in the frequency specifications. The last two algorithms are formulated as constrained optimization problems with the perfect reconstruction error as the performance index and the frequency specifications as constraints. The performance of the five algorithms is evaluated and compared using six examples; these examples include uniform filter bank, compatible non-uniform filter bank and incompatible non-uniform filter bank designs. The evaluation criteria are based on distortion and aliasing errors, the magnitude response characteristics of analysis and synthesis filters, the computation time required for the optimization, and the convergence of the performance index with respect to the number of iterations. The results show that the five algorithms can achieve almost perfect reconstruction and can meet the frequency response specifications at an acceptable level. In the case of incompatible non-uniform filter banks, the algorithms have challenges to achieve almost perfect reconstruction.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AC	Aliasing Component matrix
BFB	Biorthogonal Filter Banks
CC	Cross-talk Component matrix
CLS	Constrained Least Squares
CMFB	Cosine-Modulated Filter Bank
DSP	Digital Signal Processing
FIR	Finite Impulse Response
IIR	Infinite Impulse Response
LCM	Least Common Multiple
MR	Magnitude Response
MRAF	Magnitude Response for Analysis Filters
MRSF	Magnitude Responses of the Synthesis Filters
NPR	Nearly Perfect Reconstruction
NUFB	Non-Uniform Filter Bank
OFB	Orthogonal Filter Bank
PR	Perfect Reconstruction
PRFB	Perfect Reconstruction Filter Bank
QMF	Quadrature Mirror Filter
QP	Quadratic Programming
SIP	Semi-Infinite Programming
SQP	Sequential Quadratic Programming
UFB	Uniform Filter Bank

List of Notations

Unless stated otherwise, this list defines all notations as they appear in this work.

K	Number of sub-bands in the filter bank.
k	Denotes the kth sub-band number.
$H_k(z)$	FIR Analysis filter in z-Domain.
$F_k(z)$	FIR Synthesis filter in z-Domain.
$\uparrow$	Upsampling.
$\downarrow$	Downsampling.
$x(n)$	Input signal.
$y_U[n]$	Upsampled signal.
$y_D[n]$	Downsampled signal.
S	Sampling rate set $\{n_1, n_2\ n_3, \ldots, and\ n_K\}$
c	Constant.
$\hat{x}(n)$	Output signal.
$\widehat{\mathrm{X}}(\mathrm{z})$	Output signal in z-domain.
$\mathrm{T}_0(\mathrm{z})$	Overall distortion error.
$\mathrm{T}_\ell(\mathrm{z})$	Overall aliasing error.
N	Number of filter coefficients.
M	Least Common Multiple.
H	$\mathrm{M}(2\mathrm{N}-1)\ \mathrm{x}\ \mathrm{KN}$ matrix contains analysis filter coefficients *(Italic).*
Γ	Toeplitz operator.

$\mathrm{h_k}$	Vector contains analysis filter's coefficients as $[\mathrm{h_{k,0}}, \mathrm{h_{k,1}}, \mathrm{h_{k,2}}, \dots, \mathrm{h_{k,N-1}}]^{\mathrm{T}}$.
$\otimes$	Element wise product.
T	Matrix Transpose.
f	KN x 1 vector contains synthesis filters' coefficients as $[\mathrm{f_1^T}, \mathrm{f_2^T}, \mathrm{f_3^T}, \dots, \mathrm{f_K^T}]^T$.
$\mathrm{f_k}$	Vector contains synthesis filter's coefficients as $[\mathrm{f_{k,0}}, \mathrm{f_{k,1}}, \mathrm{f_{k,2}}, \dots, \mathrm{f_{k,N-1}}]^{\mathrm{T}}$.
$\mathrm{A(z)}$	$\mathrm{M(2N-1)}$ x KN matrix contains analysis filters' polynomials.
$\mathrm{B(z)}$	KN x 1 vector contains synthesis analysis filters' polynomials.
Δ	N^{th} -1.
$\mathrm{c(z)}$	KN x 1 vector all its elements are zeros except element $\mathrm{z}^{-\Delta}$.
$\tilde{\mathrm{Z}}$	M x MN matrix contains z-domain polynomials.
$\bar{\bar{\mathrm{H}}}$	MN x K matrix contains analysis filters' coefficients.
$\bar{\bar{\mathrm{F}}}$	K x MN diagonal matrix contains synthesis filters' coefficients.
U	MN x MN matrix all zeros except an element in the first column $(\Delta + 1)^{th}$=1.
F	$\mathrm{M(2N-1)}$ x KN matrix contains synthesis filters' coefficients *(Italic).*
$\lvert H_k(e^{j\omega})\rvert$	Analysis filter's magnitude response.
$\mathcal{R}(\omega)$	N x N matrix for filter magnitude response.
$\lvert F_k(e^{j\omega})\rvert$	Synthesis filter's magnitude response.
$\min_f$	Minimization with respect to synthesis filters.
E_{pr1}	PR error with respect to synthesis filters.
$\min_h$	Minimization with respect to analysis filters.
E_{pr2}	PR error with respect to analysis filters.
b	KN x 1 vector contains all zeros except for the N^{th} element =1 *(Italic).*
E_{m1}	Analysis filters' magnitude response error.

E_{m2}	Synthesis filters' magnitude response error.
ρ	Frequency resolution for $\omega_i \in [0\ to\ \pi]$.
J	Performance index.
w_{pr}	Weight for perfect reconstruction (PR) conditions error.
w_m	Weight for the magnitude response specifications error.
w_{pr1}	Weight for the PR error (PR with respect to synthesis filters' coefficients).
w_{m1}	Weight for the MR error (MR with respect to synthesis filters' coefficients).
w_{pr2}	Weight for the PR error (PR with respect to Analysis filters' coefficients).
w_{m2}	Weight for the MR error (MR with respect to synthesis filters' coefficients).
J_f	Performance index with respect to the synthesis filters.
J_h	Performance index with respect to the analysis filters.
$\bar{J}$	Modified Performance index.
$\bar{E}_{pr1}$	Modified PR error with respect to synthesis filters.
Ω_k^{pb}	Passband area of the k^{th} sub-band filter.
Ω_k^{sb}	Stopband area of the k^{th} sub-band filter.
$\Sigma^{(k)}$	Squared magnitude response for synthesis filter at ω_i.
$\hat{\Sigma}^{(k)}$	Summation for $\mathcal{R}(\omega)$ matrices for $\omega_i \in \Omega_k^{sp}$.
$\hat{\Psi}$	KN x KN contains $\hat{\Sigma}^{(k)}$ matrices as the diagonal.
Ψ_f	The squared stopband magnitude responses of synthesis filters at [0 to π].
Q	Symmetric matrix.
Ψ_h	The squared stopband magnitude responses of analysis filters at [0 to π].
$\overline{\overline{\psi_f}}(\mathrm{f_k}, \omega_i)$	The squared passband magnitude response of a k^{th} synthesis filter.
$\overline{\overline{\psi_h}}(\mathrm{h_k}, \omega_i)$	The squared passband magnitude response of a k^{th} synthesis filter.
δs	Stopband constraints tolerance.
δp	Passband constraints tolerance.

f_c	Cut-off frequency.
ρ_{sp}	Number of frequency grid points at the stopband area.
ρ_{pb}	Number of frequency grid points at the passband area.
A_s	Stopband attenuation.
A_p	Passband ripple.

Acknowledgments

I would first like to express my sincere gratitude to my supervisor Dr. Pan Agathoklis. The door to Dr. Agathoklis's office was always open to provide his patient guidance, enthusiastic encouragement, and useful critiques. He consistently encouraged me to achieve the very best of me, not only that, but also Dr. Agathoklis steered me in the right direction whenever he thought I needed it for my educational and professional career.

I would also like to thank all the members of the examining committee for their valuable feedback to make this work better. Special thank you to Dr. Wu-Sheng Lu. I enjoyed taking three courses with Dr. Lu as part of this degree.

I want to express my very great appreciation to the Ministry of Higher Education in Libya for providing me with the necessary funding, and for Libyan-North American Scholarship program's staff for the continued assistance.

My special thanks are extended to my supervisors and colleagues in Waha Oil Company in Libya for their continues support and encouragement as I was working toward completing this degree.

Finally, I must express my very profound gratitude to my parents, all my family members, and friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Special thank you.

Dedication

To my beloved country, Libya*.*

To my family, friends, and colleagues who supported and encouraged me during this journey.

and

To all my professors, teachers, and instructors who taught me so much more.

CHAPTER 1

Introduction

1.1 Introduction

Digital signal processing (DSP) has become one of the most important means of modern technology in many fields of science, i.e. telecommunications, medicine, radars, sonar, music and the oil and gas industry. The development of DSP technology combines mathematical principles, physical theories and technical methods, making them into complex systems that can be used to deliver specific functions [1].

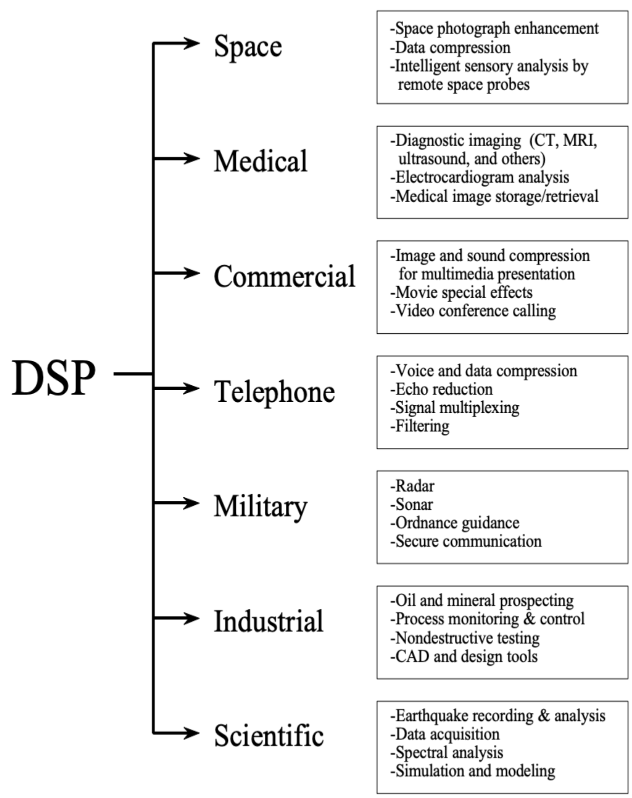

Figure 1-1 DSP applications as listed by Smith in [1].

There are numerous applications in the field of modern technology and science where DSP plays a significant role. Many of these applications include one of the fundamental elements in DSP, which is the multirate system. A multirate system is a system that deals with different sampling rates. This helps to reduce the cost of practical implementations and also enhance the performance of the implementations[2, 3]. The multirate systems have different structures, and one of the commonly used structures is the multirate filter banks. The main idea of using multirate is the system's ability to split a signal into two or more signals or to compose two or more different signals into a single signal. The process of splitting a signal and subsampling it with certain factors can be seen in many applications such as digital communications, speech and image compression, beamforming, etc. [4].

1.2 Filter Banks in Digital Signal Processing

In DSP, filter banks are used to divide a signal into frequency sub-bands with analysis filters, and then reconstruct this signal with a set of synthesis filters. Such a process may occur in sub-band coding, noise cancellation, signal transmission, or any signal operation in the frequency domain [5-8]. Figure 1-2 shows the types of a filter bank, which also is referred to as analysis-synthesis system.

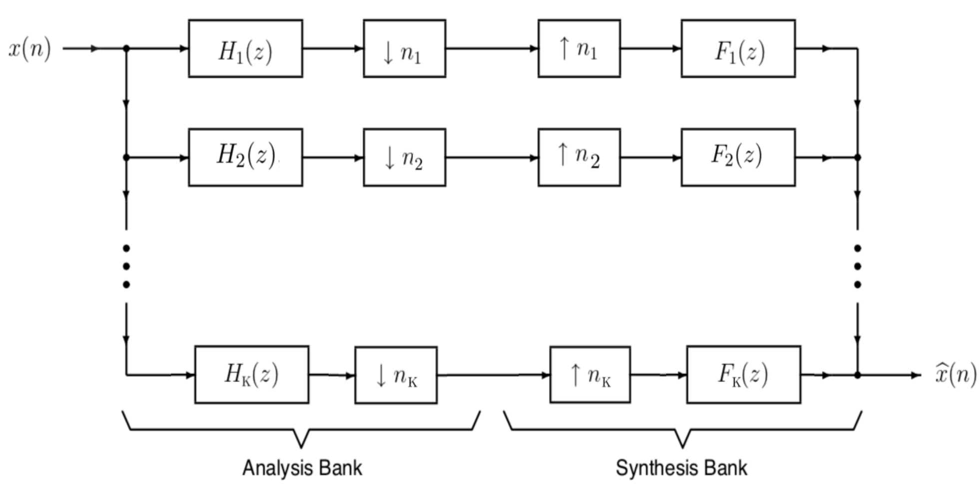

Figure 1-2 General structure of filter bank [9].

Multirate filter bank divides (Analysis) an original input signal in the frequency domain into multiple signals; and it reconstructs (Synthesize) these signals in the frequency domain into one signal in order to be the same as the original input [7, 8, 10]. This process may cause the signal to suffer from errors, e.g. aliasing distortion and magnitude distortion. Therefore, the following features in the reconstructed signal are desired: first, cancellation of aliasing error, therefor the signal in the output appears as a delayed version of the original signal and, second, not to have any distortion. These two conditions are called Perfect Reconstruction (PR) conditions [11]. In many DSP applications' requirements, designing a filter bank has to deal with PR. Designing filter banks has been a subject of research for several years, and various types of approaches and methods have been proposed. A recent survey by N. Subbulaakshmi and R. Manimegalai shows some of the different approaches and methods of filter banks (Figure 1-3) [12].

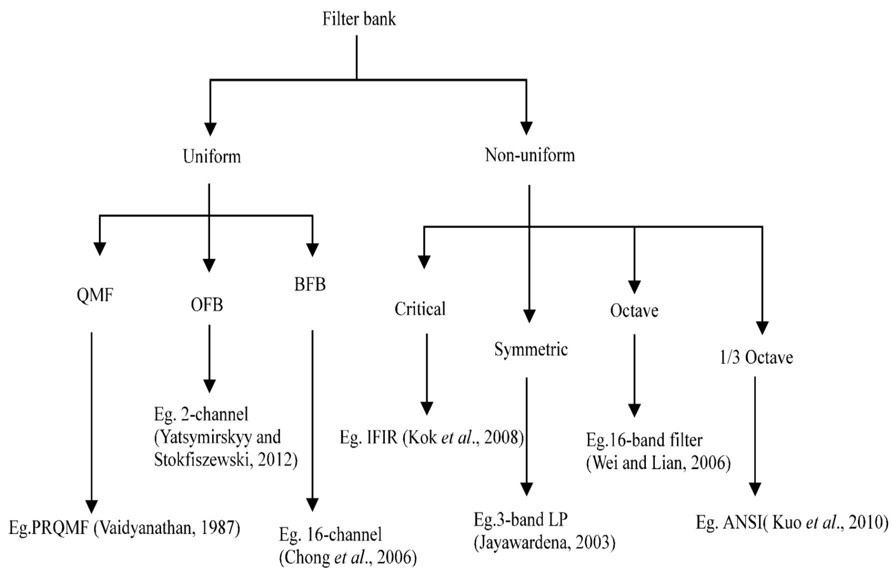

Figure 1-3 Some of filter bank types [12].

On the one hand, a Uniform Filter Bank (UFB) consists of a group of filters that partition the frequency spectrum into equal widths. As an example, the two-channel UFB consists of low-pass and high-pass filters, and down-samplers at the analysis end, up-samplers, and low-pass and high-pass filters at the synthesis end. The M-channel UFB has the same structure with extra band-pass filters, down-samplers, and up-samplers for the middle channels. [13]. On the other hand, a Non-Uniform Filter Bank (NUFB) consists of a group of filters that partition the frequency spectrum into unequal widths, and the samplers in the channels are not the same.

Many papers in the literature discuss different designs for the filter banks listed in Figure 1-3. The following section is a short review of several selected papers that are relevant to the work presented in this thesis.

One of the first designs for the maximally decimated Finite Impulse Response (FIR) filter bank is adapted by Hoang and Vaidyanathan in [14]. The design presented is based on the Quadrature Mirror Filter (QMF). The interest in that design was to achieve the PR property and meet frequency responses characteristics as well. Nayebi and Barnwel also suggest another two approaches: first, design for incompatible NUFB. The authors stated that aliasing cancellation could be achieved by introducing enough freedom to the design filters, and filtering performance can be improved by appropriate selection for passbands. Second, design for a UFB that carried out with a set of procedures in the time-domain analysis. The proposed algorithm is based on an iterative process in which the reconstruction error is reduced to the desired value [15-17]. In another design by Li et al., a method for non-uniform integer-decimated filter banks is proposed [18]; the Nearly-Perfect-Reconstruction (NPR) filter bank is based on uniform Cosine Modulated Filter Banks (CMFBs). The results show that the overall distortion and aliasing are small and comparable with the stopband attenuation of the prototype filter. Later, the general linear dual-rate system is studied by Chen et al. in [19]; the used scheme is applied to the synthesis part of the NUFB to improve the performance of this type. If the synthesis subsystems are replaced by appropriate dual-rate structures, incompatibility and structural dependency disappear; and PR is always possible.

Further, as introduced by Ho et al. in [20-22], a semi-infinite programming (SIP) optimization algorithm was utilized. The non-uniform FIR transmultiplexer is formulated as a quadratic function subject to improve the magnitude response in every filter by minimizing the stopband ripples. These algorithms have to maintain PR conditions such as the amplitude and aliasing distortions. As suggested by Xie et al. in [23], another idea is also based on direct determination of the realizable NUFB, by choosing the analysis filters to be the exact corresponding sub-bands' locations of the system input. Also using an optimization algorithm, Zhong et al. in [24] illustrate a structure with block decimation transformation to rationalize the decimation criteria CMFB. The reason behind this principle is used to achieve a linear-phase and cancel aliasing distortion in NUFB. The aliasing distortion cancellation is based on the assumption that the stopband attenuation is high enough. Likewise, a method is studied to accomplish a highly desired linear-phase for the oversampled NUFB; this method is based on an efficient modulation technique. This helps to reduce the complexity of such a design. It was also commented that, by employing the Parks-McClellan algorithm for filter design, the expected magnitude response of each filter in the design could be ensured [25].

According to Kumar et al. in [26], adjacent filters of uniform CMFB can be merged to form a NUFB, and the frequency specifications are modified, therefore the error in the PR is minimized. As they also argue in [27], the complex objective functions in a linear optimization procedure can be omitted to design an NPR NUFB. The frequency specifications are optimized to be equiripple when the filter coefficients are specified.

Moazzen and Agathoklis present two optimization techniques for UFB and NUFB using unconstrained objective functions [28, 29]. These techniques are unlike the one used by Naybi in[15-17]. On the one hand, Moazzen and Agathoklis implement their design in the z-domain and minimize one PR objective function that includes all analysis and synthesis filters as one set of linear equations. They also include both passband and stopband magnitude specifications in their first suggested algorithm. Later, this algorithm is improved as a fast optimization technique by modifying the cost function presented in [28]. This cost function is formulated as an unconstrained quadratic objective function by not including the passband response characteristics [29]. On the other hand, Naybi implements his design in the time domain and minimizes the number of sets of linear equations to satisfy the PR conditions in the filter bank design.

1.3 Outlines and Objective of this Thesis

Chapter 1 introduces different types of DSP applications, and highlights the general structure of filter banks Then it illustrates some of the common types of these filter banks. The first chapter also reviews relevant literature and the background of filter bank design. The chapter outlines the main sections and describes the objective of this thesis.

In **Chapter 2**, part of the theoretical concepts of DSP are reviewed, such as the essential terms in multirate systems: decimation, interpolation and the Nobel identities. The chapter also defines the types of sampling operations and shows the general structure of filter banks. After the PR conditions are presented, and three fundamental lemmas from [28, 29] are reviewed, these lemmas are considered as the core of this thesis. They are also used in the following chapter to introduce five optimization algorithms for UFBs and compatible and incompatible NUFBs. The chapter concludes with a summary.

Optimization techniques for filter bank designs are introduced in **Chapter 3**. These techniques include five unconstrained and constrained optimization algorithms. In algorithms 1 and 2 from [28, 29], and the proposed algorithms 3, unconstrained optimization approaches are used. However, in the proposed algorithms 4 and 5, constrained ones are adapted. Furthermore, Algorithm 3 is a modified version of algorithm 1, algorithm 4 uses SIP optimization, and algorithm 5 uses constrained least square optimization. Chapter 3 also highlights the main steps in each algorithm and ends with a brief conclusion for the work presented.

In **Chapter 4** numerous designs are presented using the algorithms outlined in Chapter 3. These designs are evaluated based on the PR conditions, the magnitude response of the analysis and synthesis filters, and the computation time taken by each algorithm. These results are illustrated in figures and tables and, after that, extensively examined. For each group of examples, results are also summarized in a tabular comparison and followed by a discussion. Later, a conclusion for the chapter is included.

Chapter 5 summarizes the work completed in this thesis and discusses the findings. After that, it draws conclusions and provides recommendations for future work. The proposed future work suggests further implementations may be used to improve the algorithms and the designed filter banks.

CHAPTER 2

Fundamentals of Multirate DSP

2.1 Introduction

In this chapter, some of the fundamentals for multirate DSP systems are introduced. These include upsampling, downsampling, decimation and interpolation. Also, Noble identities, critical sampling, under-sampling, and over-sampling are defined. Following that, the chapter outlines filter bank structures and defines the PR conditions. Three lemmas for filter bank PR, and magnitude response characteristics are reviewed from [28, 29], and then used for the algorithms in Chapter 3.

2.2 Fundamental Building Blocks

Multirate DSP systems consist of fundamental building blocks such as digital filters, upsamplers, and downsampler. Thus, downsampling and upsampling operations are essential parts [30].

2.2.1 Downsampling and Upsampling

Processing elements which perform these operations are called downsampler and upsamplers, respectively.

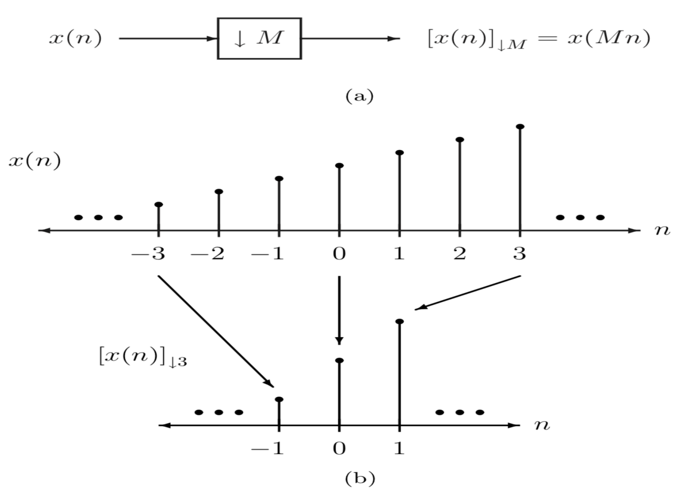

Figure 2-1 M-fold downsampler input/output: a) Block diagram, b) Example for M= 3 [9].

On the one hand, processing elements which decrease the sampling rate of the input signal $x[n]$ by a factor of M are called downsamplers. Downsampling mathematically is defined in equation (2-1), where $M \geq 2$. The downsampling process retains only every Mth sample of the input signal $x[n]$. Figure 2-1 shows downsampling when M=3.

$$y_D[n] = [x(n)]_{\downarrow M} = x[Mn] \qquad (2\text{-}1)$$

On the other hand, processing elements which increase the sampling rate of the input signal $x[n]$ by a factor of M are called upsamplers. Upsampling mathematically is defined in equation (2-2), where $M \geq 2$. The upsampling process inserts $M - 1$ zeros between the original samples of the input signal $x[n]$. Figure 2-2 shows the upsampling when M=2.

$$y_U[n] = [x(n)]_{\uparrow M} = \begin{cases} x\frac{n}{M}, & n = multiple\ of\ M \\ 0, & otherwise \end{cases} \qquad (2\text{-}2)$$

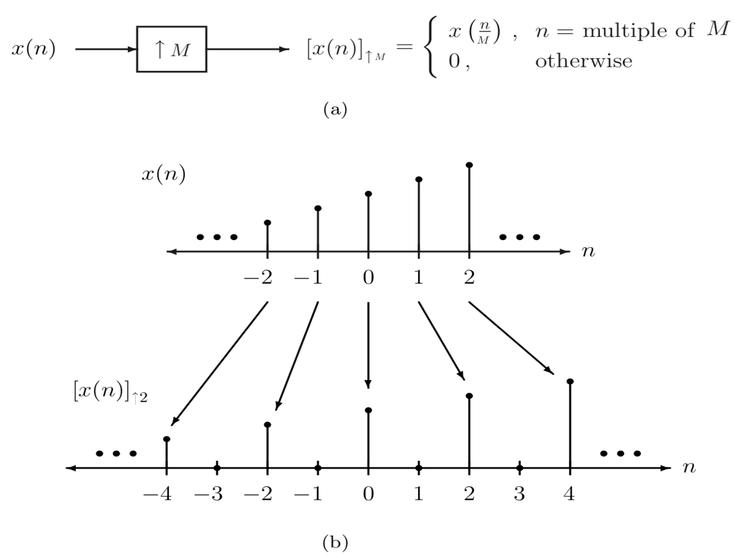

Figure 2-2 M-fold upsampler input/output: (a) Block diagram, (b) Example for M= 2 [9].

2.2.2 Decimation and Interpolation

Decimation is the process of filtering and downsampling a signal when decreasing its effective sampling rate, and this is done by adding a filter H(z) that is employed before the downsampler. If a signal is only downsampled by just throwing away the intermediate samples, this produces aliasing. Building a decimation structure as the one illustrated in Figure 2-3 prevents aliasing.

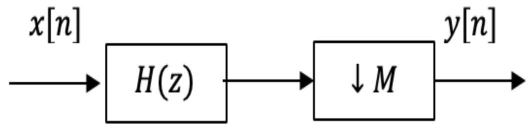

Figure 2-3 Decimator consisting of filter H(z) and a downsampler.

Interpolation is the process of upsampling and filtering a signal. Interpolation is done by adding a filter $F(z)$ that is employed after the upsampler [31]. Building an interpolation structure as the one illustrated in Figure 2-4 prevents the extra spectral copies that might result from upsampling.

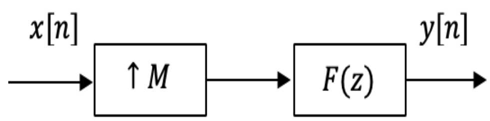

Figure 2-4 Interpolator consisting of an upsampler and filter F(z).

2.3 Noble Identities

There are two noble identities in digital signal processing. These identities describe the property of reversing the order of the filters and the samplers in the multirate system. In a typical multirate system such as the ones presented in Figure 2-6, the input signal is being first filtered and then downsampled. This approach is computationally inefficient because it calculates samples that will be eventually discarded [30] It would be much better to perform downsampling in the first place and then apply filtering on the reduced number of samples Figure 2-5-a and Figure 2-5-b illustrate equivalent block diagrams for the first and the second Noble identities of decimation and interpolation, respectively.

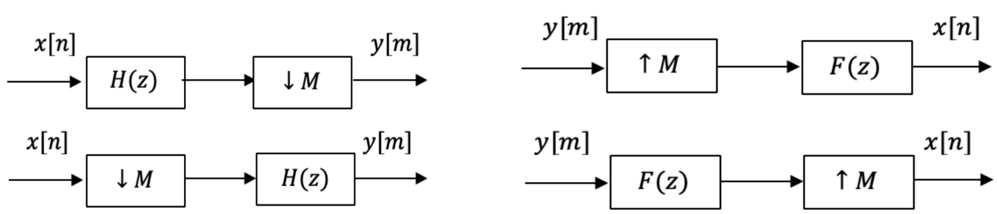

Figure 2-5-a First noble identity.   Figure 2-5-b – Second noble identity.

Figure 2-5 Interchanging between filtering and sampling.

2.3.1 Over-sampling, Under-sampling, and Critical sampling

According to the overall sampling rates $\{n_1, n_2\, n_3, \ldots, and\, n_K\}$ of the sub-band signals in the filter bank presented in Figure 2-6 , special names are given depending on the nature of the employed samplers values n_k at the inputs. It is important to mention that the overall rate of the sub-band signals is simply $\left(\sum_{k=1}^{K} \frac{1}{n_k}\right)$ times the input rate [9] , therefor:

filter banks are said to be critically sampling systems if

$$\left(\sum_{k=1}^{K} \frac{1}{n_k}\right) = 1. \qquad (2\text{-}3)$$

It is also common to refer to such types of filter banks as maximally decimated filter banks. In such a system, the output $\hat{x}(n)$ may not have a loss of information.

Filter banks are said to be under-sampling systems if

$$\left(\sum_{k=1}^{K} \frac{1}{n_k}\right) < 1. \qquad (2\text{-}4)$$

These filter banks incur a loss of information, and it is common to refer to them as over-decimated filter banks.

Filter banks are said to be over-sampling systems if

$$\left(\sum_{k=1}^{K} \frac{1}{n_k}\right) > 1. \qquad (2\text{-}5)$$

In this case, the filter bank systems have redundancy, and it is common to refer to them as under-decimated filter banks [9, 32].

2.4 Filter Banks

In the filter banks, a K-channel (or K-band) analysis filter bank is a structure that transforms an input signal to a set of K sub-band signals. A corresponding K-channel synthesis filter bank transforms these K sub-band signals to a full band signal; in other words, the inverse operation. The common filter banks structures are the tree structure and the parallel structure [33]. The parallel structure is more general and favourable because of its simplicity. Figure 2-6 shows the analysis and synthesis filters for such filter bank.

In the case of the structure shown in Figure 2-6, a uniform filter bank has K sub-bands with equal bandwidths, and each sub-band must be decimated by a K factor [34]. On the other hand, a non-uniform filter bank has K sub-bands and these sub-bands have unequal bandwidths. In the non-uniform filter-bank, decimation factors n_k are not necessarily equal [30, 35].

In a non-uniform filter bank design, there are two types of sampling rates sets. These types are compatible set or incompatible set. The definition of a compatible sampling rate set is introduced in [14, 36]. From the definition of the compatibility presented in [14]. Let the sampling rates in Figure 2-6 defined as a set $S = \{n_1, n_2\, n_3, \ldots, and\; n_K\}$ and arranged as following $n_1 \leq n_2 \leq n_3 \leq, \ldots, \leq n_K$. The sampling rate set S is compatible if it satisfies equation (2-3), and pairing up the alias components in the aliasing matrix is possible [14, 37, 38].

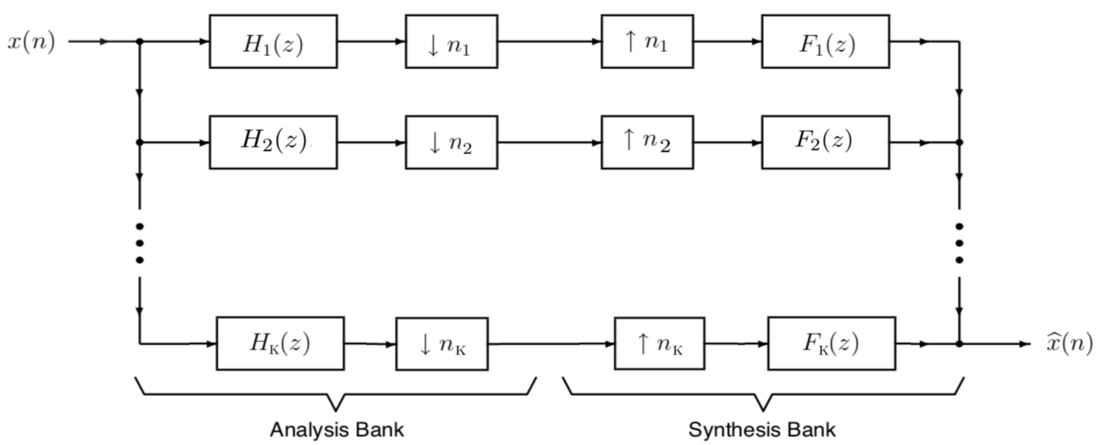

Figure 2-6 Filter bank structure with K sub-bands [9].

2.5 Perfect Reconstruction Filter Banks

Perfect reconstruction conditions are desirable properties for many DSP applications. These conditions are discussed explicitly in [35, 36, 39]. Generally speaking, for a filter bank illustrated as in Figure 2-6, it is said to have a perfect reconstruction if the output $\hat{x}(n)$ of the synthesis bank is a delayed and possibly scaled version of the input $x(n)$, i.e.

$$\hat{x}(n) = cx(n - n_0) \qquad (2\text{-}6)$$

For some $c \neq 0$ and some integer n_0. Such a filter bank is referred to as a PR filter bank, and this means that $x(n)$ can be perfectly reconstructed using a proper synthesis filter bank without any aliasing, amplitude, and phase distortions.

Consider a filter bank with integer sampling rates $\{n_1, n_2\, n_3, \ldots, and\ n_K\}$ shown in Figure 2-6, where K is the number of total sub-bands, assume also that this filter structure design satisfies the critically sampling condition presented in equation (2-3) [35].

The output of this filter bank in the z-domain can be expressed by:

$$\hat{X}(z) = \sum_{k=1}^{K} \frac{1}{n_k} F_k(z) \sum_{\ell=0}^{n_k-1} X(ze^{-j\frac{2\pi\ell}{n_k}})\ H(ze^{-j\frac{2\pi\ell}{n_k}}) \qquad (2\text{-}7)$$

$$\hat{X}(z) = \sum_{k=1}^{K} \frac{1}{n_k} F_k(z)\, X(z)\, H_k(z) + \sum_{k=1}^{K} \frac{1}{n_k} F_k(z)\, X(ze^{-j\frac{2\pi\ell}{n_k}})\ H(ze^{-j\frac{2\pi\ell}{n_k}}) \qquad (2\text{-}8)$$

$$\hat{X}(z) = X(z)T_0 + \sum_{\ell=1}^{n_k-1} X(ze^{-j\frac{2\pi\ell}{n_k}})\, T_\ell\,(z) \qquad (2\text{-}9)$$

where

$$T_0(z) = \sum_{k=1}^{K} \frac{1}{n_k} F_k(z)\, H_k(z) \qquad (2\text{-}10)$$

$$T_\ell(z) = \sum_{k=1}^{K} \frac{1}{n_k} F_k(z) H_k\left(ze^{-j\frac{2\pi\ell}{n_k}}\right) \qquad (2\text{-}11)$$

$T_0(z)$ is called overall distortion transfer function, and $T_\ell(z)$ $\ell \neq 0$ is called overall aliasing transfer function corresponding to $X\,(ze^{-j\frac{2\pi\ell}{n_k}})$.
Therefore, PR is achieved when $T_0(z)$ is a pure delay, and $T_\ell(z)$ is zero, and this leads to the following two conditions for PR:

Distortion Cancellation Condition:

$$\sum_{k=1}^{K} \beta_{k,0}\, F_k(z)\, H_k(z) = z^{-\Delta} \qquad (2\text{-}12)$$

Aliasing Cancellation Condition:

$$\sum_{k=1}^{K} \beta_{k,\ell}\, F_k(z) H_k\left(ze^{-j\frac{2\pi\ell}{n_k}}\right) = 0, \quad \text{for } \ell = 1,2,3,\ldots,M-1 \qquad (2\text{-}13)$$

where

M is the Least Common Multiple (LCM) between $\{n_1, n_2\ n_3, \dots, and\ n_K\}$

$\alpha_{k,\ell}$ for $k = 1,2,3,\dots,K$, and $\ell = 1,2,3,\dots,M-1$ can be expressed as follows:

$$\alpha_{k,\ell} = \begin{cases} 1, & \text{if} \quad e^{j\frac{2\pi\ell}{M}} \in \left\{ e^{j\frac{2\pi}{n_k}(1)}, e^{j\frac{2\pi}{n_k}(2)}, e^{j\frac{2\pi}{n_k}(3)}, \dots, e^{j\frac{2\pi}{n_k}(n_k-1)} \right\} \\ 0, & \text{Otherwise} \end{cases} \tag{2-14}$$

$$\beta_{k,\ell} = \begin{cases} \frac{1}{n_{k,}}, & \ell = 0 \\ \frac{\alpha_{k,\ell}}{n_{k,}}, & \ell \neq 0 \end{cases} \tag{2-15}$$

The use of $\alpha_{k,\ell}$ indicates that for $\alpha_{k,\ell} = 0$ the analysis aliasing terms $H(ze^{-j\frac{2\pi\ell}{n_k}})$ are not included in the equation (2-13).

The next section reviews two lemmas for the PR conditions. The perfect reconstruction condition equations represented in [28] are formulated in the z-domain as one set of linear equations for all sub-bands.

Lemma 1 [28]**:** Giving FIR analysis and synthesis filters of the same length N, the perfect reconstruction conditions in equations (2-12) and (2-13) can be expressed as one set of linear equations as follows:

$$Hf = b \tag{2-16}$$

where

$$H_{M(2N-1)xKN} = \begin{bmatrix} \beta_{1,0}\Gamma(h_1) & \beta_{2,0}\Gamma(h_2) & \cdots & \beta_{K,0}\Gamma(h_K) \\ \beta_{1,1}\Gamma(h_1 \otimes \Lambda_1) & \beta_{2,1}\Gamma(h_2 \otimes \Lambda_1) & \cdots & \beta_{K,1}\Gamma(h_K \otimes \Lambda_1) \\ \beta_{1,2}\Gamma(h_1 \otimes \Lambda_2) & \beta_{2,2}\Gamma(h_2 \otimes \Lambda_2) & \cdots & \beta_{K,2}\Gamma(h_K \otimes \Lambda_2) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{1,M-1}\Gamma(h_1 \otimes \Lambda_{M-1}) & \beta_{2,M-1}\Gamma(h_2 \otimes \Lambda_{M-1}) & \cdots & \beta_{K,M-1}\Gamma(h_K \otimes \Lambda_{M-1}) \end{bmatrix} \tag{2-17}$$

the Γ operator is defined as follows:

$$\Gamma([a_1, a_2, a_3, \ldots\ a_N]^T) = \begin{bmatrix} a_1 & 0 & 0 & \ldots & 0 \\ a_2 & a_1 & 0 & \ldots & 0 \\ a_3 & a_2 & a_1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ a_N & a_{N-1} & a_{N-2} & \ldots & a_1 \\ 0 & a_N & a_{N-1} & \cdots & a_2 \\ 0 & 0 & a_N & \ldots & a_3 \\ 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & a_N \end{bmatrix} \qquad (2\text{-}18)$$

and h_k for (k= 1,2,3, …, K) is a vector containing analysis filters' coefficients of the k[th] sub-band, i.e. $h_k = [h_{k,0}, h_{k,1}, h_{k,2}, \ldots, h_{k,N-1}]^T$, T is the transpose operator, β is defined in equation (2-15), M is the LCM of the sampling rates $\{n_1, n_2\ n_3, \ldots, and\ n_K\}$.

The symbol ⊗ represents element-wise product, and $\wedge_i$ is defined as follows:

$$\wedge_i = \left[e^{j\frac{2\pi}{n_k}i(0)}, e^{j\frac{2\pi}{n_k}i(1)}, e^{j\frac{2\pi}{n_k}i(2)}, \ldots, e^{j\frac{2\pi}{n_k}i(N-1)}\ \right]^T \text{for } i = 1,2,3, \ldots, M-1 \qquad (2\text{-}19)$$

and

$$f = \left[f_1^T, f_2^T, f_3^T, \ldots, f_K^T\right]^T. \qquad (2\text{-}20)$$

f is the $KN \times 1$ vector that contains all the synthesis filters' coefficient, where each f_k for (k= 1,2,3, …, K) is a vector contains synthesis filters' coefficients of a k[th] sub-band $f_k^T = \left[f_{k,0}, f_{k,1}, f_{k,2}, \ldots, f_{k,N-1}\right]$.

b is a vector whose elements are all zeros except the $(\Delta + 1)^{th}$ element, which equals 1, and it follows the right side from equation (2-12).

Proof of Lemma 1 [28]: Given the k^{th} FIR analysis and synthesis filters in he z-domain with equations (2-21) and (2-22), and by assuming N is an even number which represents the number of the filters' coefficients.

$$H_k(z) = \sum_{n=0}^{N-1} h(n) z^{-n} \tag{2-21}$$

and

$$F_k(z) = \sum_{n=0}^{N-1} f(n) z^{-n} \tag{2-22}$$

The perfect reconstruction equations in (2-12) and (2-13) are analyzed in the z-domain, and a proof in the z-domain for equation (2-23) is obtained in the next section.

$$\mathrm{A(z)B(z) = c(z)} \tag{2-23}$$

To form A(z), $\alpha_{k,\ell}$ introduced in equation (2-14) implies that the aliasing term from $e^{j\frac{2\pi\ell}{M}}$ is present at the k^{th} sub-band and zero otherwise, and the term $\beta_{k,\ell}$ in equation (2-15) represents the sampling rate for each bank. i.e. The Aliasing Component (AC) matrix in the z-domain [37, 40-42], and it can be depicted as follows:

$$A(z)_{M(2N-1)xKN} = \begin{bmatrix} \beta_{1,0}H_1(z) & \beta_{2,0}H_2(z) & \cdots & \beta_{K,0}H_K(z) \\ \beta_{1,1}H_1(ze^{-j\frac{2\pi}{M}}) & \beta_{2,1}H_2(ze^{-j\frac{2\pi}{M}}) & \dots & \beta_{K,1}H_K(ze^{-j\frac{2\pi}{M}}) \\ \beta_{1,2}H_1(ze^{-j\frac{2\pi}{M}(2)}) & \beta_{2,2}H_2(ze^{-j\frac{2\pi}{M}(2)}) & \dots & \beta_{K,2}H_K(ze^{-j\frac{2\pi}{M}(2)}) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{1,M-1}H_1(ze^{-j\frac{2\pi}{M}(M-1)}) & \beta_{2,M-1}H_2(ze^{-j\frac{2\pi}{M}(M-1)}) & \cdots & \beta_{K,M-1}H_K(ze^{-j\frac{2\pi}{M}(M-1)}) \end{bmatrix} \tag{2-24}$$

$$B(z) = \begin{bmatrix} F_1(z) \\ F_2(z) \\ F_3(z) \\ \vdots \\ F_K(z) \end{bmatrix}_{KN\,x1}, \text{ and } c(z) = \begin{bmatrix} z^{-\Delta} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{KN\,x1} \qquad (2\text{-}25), (2\text{-}26)$$

Assume the $\tilde{Z}$ operator is

$$\tilde{Z}_{M\,x\,MN}\begin{bmatrix} p & 0^T & 0^T & \cdots & 0^T \\ 0^T & p & 0^T & \dots & 0^T \\ 0^T & 0^T & p & \dots & 0^T \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ 0^T & 0^T & 0^T & \cdots & p \end{bmatrix} \tag{2-27}$$

where the $\tilde{Z}$ operator is described by the following:

$p_{1\,x\,N} = \left[1,\ z^{-1}, z^{-2}, \dots, z^{-(N-1)}\right]$, $0_{Nx1} = [0,0,0,\dots,0]^T$, and $e_{M\,x\,1}\,[1,1,1,\dots,1]^T$.

Therefore, equation (2-23) can be rewritten as follows:

$$\tilde{Z}\left(\overline{\overline{H}}\overline{\overline{F}}\right)\tilde{Z}^T e = \tilde{Z}(U)\tilde{Z}^T e \tag{2-28}$$

where

$$\overline{\overline{H}}_{MN\,x\,K} = \begin{bmatrix} \beta_{1,0}h_1 & \beta_{2,0}h_2 & \cdots & \beta_{K,0}h_k \\ \beta_{1,1}\,h_1 \otimes \Lambda_1 & \beta_{2,1}h_2 \otimes \Lambda_1 & \dots & \beta_{K,1}h_k \otimes \Lambda_1 \\ \beta_{1,2}h_1 \otimes \Lambda_2 & \beta_{2,2}h_2 \otimes \Lambda_2 & \dots & \beta_{K,2}h_K \otimes \Lambda_2 \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{1,M-1}h_1 \otimes \Lambda_{M-1} & \beta_{2,M-1}h_2 \otimes \Lambda_{M-1} & \cdots & \beta_{K,M-1}h_K \otimes \Lambda_{M-1} \end{bmatrix} \tag{2-29}$$

and

$$\overline{\overline{F}}_{K\,x\,MN} = \begin{bmatrix} f_1^T & 0^T & 0^T & \cdots & 0^T \\ 0^T & f_2^T & 0^T & \dots & 0^T \\ 0^T & 0^T & f_3^T & \dots & 0^T \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ 0^T & 0^T & 0^T & \cdots & f_K^T \end{bmatrix} \tag{2-30}$$

U is the MN x MN matrix whose elements are all zeros except the one located on the first column $(\Delta + 1)^{th}$. This element follows the right side of equation (2-12).

Therefore, from equation (2-28)

$$\tilde{Z}\left(\bar{\bar{\mathrm{H}}}\bar{\bar{\mathrm{F}}} - \mathrm{U}\right)\tilde{Z}e = 0_{Mx1} \tag{2-31}$$

and equation (2-31) can be expressed as follows:

$$\tilde{Z}\mathrm{G}z^T = 0_{Mx1} \tag{2-32}$$

where

$$z_{1\,x\,MN} = e^T\,\tilde{Z} = [p, p, p, \dots, p] \tag{2-33}$$

$$\mathrm{G}_{MN\,x\,MN} = \bar{\bar{\mathrm{H}}}\bar{\bar{\mathrm{F}}} - \mathrm{U} \tag{2-34}$$

Due to the structure of z and $\tilde{Z}$, equation (2-32) can be simplified further. In order to do that, matrix G needs to be split into $1\,x\,N$ vectors as follows:

$$\mathrm{G} = \begin{bmatrix} g_1^{(1)} & g_2^{(1)} & g_3^{(1)} & \cdots & g_M^{(1)} \\ g_1^{(2)} & g_2^{(2)} & g_3^{(2)} & \cdots & g_M^{(2)} \\ g_1^{(3)} & g_2^{(3)} & g_3^{(3)} & \cdots & g_M^{(3)} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ g_M^{(MN)} & g_M^{(MN)} & g_M^{(MN)} & \cdots & g_M^{(MN)} \end{bmatrix} \tag{2-35}$$

$g_j^{(i)}$corresponds to the elements of matrix G located on the $i^{th}\,row$ ($i = 1,2,\dots,MN$) and $(j-1)N + 1^{th}\;to\;jN^{\,th}$ columns (j =1,2,...,M).

Equation (2-34) can be replaced by:

$$\tilde{Z}\widehat{\mathrm{G}}p^T = 0_{Mx1} \tag{2-36}$$

where

$$\widehat{\mathrm{G}} = \begin{bmatrix} \hat{g}^{(1)} \\ \hat{g}^{(2)} \\ \hat{g}^{(3)} \\ \vdots \\ \hat{g}^{(MN)} \end{bmatrix} \tag{2-37}$$

By using $\hat{g}^{(i)} = g_1^{(i)} + g_2^{(i)} + g_3^{(i)} + \cdots + g_M^{(i)}$ for $i = 1,2,...,MN$, the matrix $\widehat{\mathrm{G}}$ can be split into N x N sub-matrixes as follows:

$$\widehat{\mathrm{G}}_{MN\,x\,N} = \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \vdots \\ \varphi_M \end{bmatrix} \text{,where } \varphi_{i(N\,x\,N)=} \begin{bmatrix} \hat{g}^{((i-1)N+1)} \\ \hat{g}^{((i-1)N+2)} \\ \hat{g}^{((i-1)N+3)} \\ \vdots \\ \hat{g}^{(iN)} \end{bmatrix} \text{ for } i = 1,2,...,MN, \tag{2-38}$$

it can be shown that equation (2-39) means that the summation of all elements in every single off-diagonal of φ_i ($i = 1,2,...,M$) is equal to zero.

$$\varphi_i = \begin{bmatrix} \varphi_{1,1} & \varphi_{1,2} & \varphi_{1,3} & \cdots & \varphi_{1,n} \\ \varphi_{2,1} & \varphi_{2,2} & \varphi_{2,3} & \cdots & \varphi_{2,n} \\ \varphi_{3,1} & \varphi_{3,2} & \varphi_{3,3} & \cdots & \varphi_{3,n} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ \varphi_{m,1} & \varphi_{m,2} & \varphi_{m,3} & \cdots & \varphi_{m,n} \end{bmatrix} \tag{2-39}$$

where

$m = n = N$ the number of coefficients.

In equation (2-39), the off-diagonal of each sub-matrix φ_i generates the summation of all coefficients $\varphi_{m,n}$ when they have the same order z. If this element $\varphi_{m,n}$ is not a zero, it is cancelled by a conjugate in the off-diagonal at the summation. Equation (2-39) can be formulated as a set of linear equations, as shown in equation (2-16).

End of proof.

This approach in Lemma 1 is similar to the approach introduced in [16]. However, the one in [16] is based on the time-domain analysis for the filter bank design, and it is also formulated as a number of sets conditions for the exact reconstruction, instead of having one set of linear equations.

In Lemma 2, the perfect reconstruction conditions are formulated using a similar approach. The difference in Lemma 2 is that the equations are formulated in terms of filter coefficients. Also, the approach is based on generating a matrix from the synthesis synthesis filters coefficients as it will be shown in equation (2-41). This matrix is similar to a matrix called Cross-talk Component (CC) matrix used in transmultiplexers, and it is introduced in detail in the literature [37, 40-42].

Lemma 2 [29]: Giving the same conditions in Lemma 1, the perfect reconstruction error with respect to synthesis filters coefficients can also be formulated as one set of linear equations, and this can be expressed as follows:

$$Fh = b \qquad (2\text{-}40)$$

where

$$F_{M(2N-1)xKN} \begin{bmatrix} \beta_{1,0}\Gamma(f_1) & \beta_{2,0}\Gamma(f_2) & \cdots & \beta_{K,0}\Gamma(f_K) \\ \beta_{1,1}\Gamma(f_1 \otimes \Lambda_1) & \beta_{2,1}\Gamma(f_2 \otimes \Lambda_1) & \cdots & \beta_{K,1}\Gamma(f_K \otimes \Lambda_1) \\ \beta_{1,2}\Gamma(f_1 \otimes \Lambda_2) & \beta_{2,2}\Gamma(f_2 \otimes \Lambda_2) & \cdots & \beta_{K,2}\Gamma(f_K \otimes \Lambda_2) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{1,M-1}\Gamma(f_1 \otimes \Lambda_{M-1}) & \beta_{2,M-1}\Gamma(f_2 \otimes \Lambda_{M-1}) & \cdots & \beta_{K,M-1}\Gamma(f_K \otimes \Lambda_{M-1}) \end{bmatrix} \qquad (2\text{-}41)$$

The above matrix F in equation (2-41) is structured the same as the matrix H in equation (2-17). It includes the synthesis filters' coefficients instead of analysis filters' coefficients, and b is defined as a vector whose elements are all zeros except the $(\Delta + 1)^{th}$ element which is 1, [28, 29, 43].

f_k for (k= 1,2,3, …, K) is vector which contains synthesis filters' coefficients of the k[th] sub-band, i.e. $f_k = [f_{k,0}, f_{k,1}, f_{k,2}, \dots, f_{k,N-1}]^T$, where T is the transpose operator. The $\otimes$ represents the element-wise product, and $\wedge_i$ is defined as follows:

$$\wedge_i = \left[e^{-j\frac{2\pi}{n_k}i(0)}, e^{-j\frac{2\pi}{n_k}i(1)}, e^{-j\frac{2\pi}{n_k}i(2)}, \dots, e^{-j\frac{2\pi}{n_k}i(N-1)} \right]^T \; for\; i = 1,2,3, \dots, M-1 \qquad (2\text{-}42)$$

and

$$h = \left[\mathrm{h}_1^\mathrm{T}, \mathrm{h}_2^\mathrm{T}, \mathrm{h}_3^\mathrm{T}, \dots, \mathrm{h}_\mathrm{K}^\mathrm{T}\right]^T. \qquad (2\text{-}43)$$

h is the KN x 1 vector that contains all the analysis filters' coefficient, where each h_k for (k= 1,2,3, …, K) is a vector contains analysis filters' coefficients of a k[th] sub-band $\mathrm{h}_\mathrm{k}^\mathrm{T} = \left[\mathrm{h}_{\mathrm{k},0}, \mathrm{h}_{\mathrm{k},1}, \mathrm{h}_{\mathrm{k},2}, \dots, \mathrm{h}_{\mathrm{k},\mathrm{N}-1}\right]$, and b is a vector whose elements are all zeros except the $(\Delta + 1)^{\mathrm{th}}$ which equals 1, and it follows the right side of equation (2-12).

In addition, to achieve the perfect reconstruction in the analysis-synthesis system, it is also necessary to have a design that satisfies some prescribed requirement for the magnitude response of the analysis and synthesis filters. For our design , only the passband and stopband energies and considered and formulated as in [43]. It is assumed that the linear phase response of the total system is guaranteed as the perfect reconstruction is attained. For the magnitude response, the next lemma is presented.

Lemma 3 [28]: Giving a k[th] analysis FIR filter with $\mathrm{h}_\mathrm{k}^\mathrm{T} = \left[\mathrm{h}_{\mathrm{k},0}, \mathrm{h}_{\mathrm{k},1}, \mathrm{h}_{\mathrm{k},2}, \dots, \mathrm{h}_{\mathrm{k},\mathrm{N}-1}\right]$, where N is the number of coefficients . The magnitude response of this filter at the frequency ω can be expressed as the square root of a quadratic function:

$$\left|H_k\left(e^{j\omega}\right)\right| = \left|\sum_{i=0}^{N-1} h_{k,i} e^{-j(i\omega)}\right| = \sqrt{\mathrm{h}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega)\mathrm{h}_\mathrm{k}} \qquad (2\text{-}44)$$

where

$$\mathcal{R}(\omega) = \begin{bmatrix} 1 & cos\,\omega & cos\ 2\omega & \cdots & cos(N-1)\omega \\ cos\,\omega & 1 & cos\,\omega & \cdots & cos(N-2)\omega \\ cos\ 2\omega & cos\,\omega & 1 & \cdots & cos(N-3)\omega \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ cos(N-1)\omega & cos(N-2)\omega & cos(N-3)\omega & \cdots & 1 \end{bmatrix}_{N\,x\,N} \qquad (2\text{-}45)$$

Proof of Lemma 3 [28]: Using Euler equation, the magnitude response of the kth analysis filter can be written as:

$$\left|H_k\left(e^{j\omega}\right)\right| = \sqrt{\left(\sum_{i=0}^{N-1} \mathrm{h}_{\mathrm{k,i}}\, cos(i\omega)\right)^2 + \left(\sum_{i=0}^{N-1} \mathrm{h}_{\mathrm{k,i}}\, sin(i\omega)\right)^2} \qquad (2\text{-}46)$$

The equation in (2-46) can be formulated as:

$$\left|H_k\left(e^{j\omega}\right)\right| = \sqrt{\mathrm{h}_\mathrm{k}^\mathrm{T}\, \xi_{cos}(\omega)\xi_{cos}^T(\omega)\mathrm{h}_k + \mathrm{h}_\mathrm{k}^\mathrm{T}\, \xi_{sin}(\omega)\xi_{sin}^T(\omega)\mathrm{h}_k} \qquad (2\text{-}47)$$

where

$$\xi_{cos} = [\,1\ \ cos(\omega)\ \ cos(2\omega) \ldots cos\big((N-1)\omega\big)\,]^T$$

$$\xi_{sin} = [\,1\ \ sin(\omega)\ \ sin(2\omega)\ \ldots\ sin((N-1)\omega)]^T$$

Considering $\mathcal{R}_{COS}(\omega) = \xi_{cos}(\omega)\, \xi_{cos}^T(\omega)$ and $\mathcal{R}_{sin}(\omega) = \xi_{sin}(\omega)\, \xi_{sin}^T(\omega)$.

therefore,

$$\left|h_k\left(e^{j\omega}\right)\right| = \sqrt{\mathrm{h}_\mathrm{k}^\mathrm{T}\, \mathcal{R}_{COS}(\omega)\mathrm{h}_\mathrm{k} + \mathrm{h}_\mathrm{k}^\mathrm{T}\, \mathcal{R}_{sin}(\omega)\mathrm{h}_\mathrm{k}} \qquad (2\text{-}48)$$

Finally, by introducing $\Re(\omega) = \mathcal{R}_{COS}(\omega) + \mathcal{R}_{sin}(\omega)$, and using trigonometric identity $cos(a+b) = cos(a)\, cos(b) - sin(a)\, sin(b)$ equation (2-45) can be obtained,

End of proof.

Consider the sub-band kth synthesis filter coefficients $\mathrm{f}_\mathrm{k}^\mathrm{T} = [\mathrm{f}_{\mathrm{k},0}, \mathrm{f}_{\mathrm{k},1}, \mathrm{f}_{\mathrm{k},2}, \dots, \mathrm{f}_{\mathrm{k},\mathrm{N}-1}]$. The magnitude response for this synthesis filter is represented by:

$$|F_k(e^{j\omega})| = \left|\sum_{i=0}^{N-1} f_{k,i} e^{-j(i\omega)}\right| = \sqrt{\mathrm{f}_\mathrm{k}^\mathrm{T} \mathcal{R}(\omega) \mathrm{f}_\mathrm{k}} \tag{2-49}$$

The proof can be done similarly for the magnitude response a kth synthesis FIR filter.

2.6 Conclusion

In this chapter, some main concepts of multirate DSP are reviewed, namely decimation and interpolation, types of sampling rates and structure of filter banks. The PR in the filter bank is defined and demonstrated by equations, and the distortion and aliasing errors are highlighted. Later in the chapter, lemmas 1, 2, and 3 are reviewed. Lemmas 1 and 2 are formulated as one set of linear equations. First, with respect to analysis filters and, second, with respect to synthesis filters. Further, lemma 3 shows the magnitude responses formulation of analysis and synthesis filters. These lemmas are used in Chapter 3 to formulate five optimization algorithms.

CHAPTER 3

Filter Bank Design Using Optimization Techniques.

3.1 Introduction

In this chapter, several optimization design schemes are introduced for designing UFBs and NUFBs. The five optimization algorithms are mainly based on lemmas 1, 2, and 3 presented in Chapter 2. Using these various optimization techniques, the objective is to find filter bank analysis and synthesis filters' coefficients that satisfy the PR or NPR, and also meet the magnitude response design specifications. The first three methods presented in this section use unconstrained optimization to minimize the performance index, while the last two methods use constrained optimization. In Chapter 3, the optimization problem for each algorithm is defined, as well as any constraints involved.

3.2 Filter banks design by optimization

There are different optimization techniques for filter bank design. In this chapter, the focus is on an optimization approach, where the optimization algorithm is applied to an objective function that involves both synthesis and analysis filters coefficients [44]. This filter bank optimization method either employs an unconstrained optimization algorithm where the objective function includes the PR error and the magnitude response error with respect to design specifications, or it employs a constrained optimization approach where the objective function includes the PR error subject to equality and/or inequality constraints related to the magnitude response design specifications. [45, 46].

In Chapter 2, the PR conditions are formulated as a one set of linear equations, as shown in Lemma 1. This implies that to satisfy these conditions, the distortion and aliasing errors in this lemma can be minimized as follows:

$$\min_{f} \quad E_{pr1} = \|Hf - b\|^2 \qquad (3\text{-}1)$$

where

- E_{pr} is the perfect reconstruction error with respect to synthesis filters coefficients.
- $\| \quad \|^2$ is l_2-norm.
- H is a matrix which includes the analysis filters and formed with equation (2-17).
- f is a KN vector which includes all synthesis filters' coefficients .
- b is a KN zeros vector except for the Nth element =1 which implies the delay.

Similarly, from lemma 2 in Chapter 2, the reconstruction error with respect to analysis filters coefficients can be minimized as follows:

$$\min_{h} \quad E_{pr2} = \|Fh - b\|^2 \tag{3-2}$$

where

- E_{pr} is the perfect reconstruction error with respect to analysis filters coefficients.
- F is a matrix which includes the synthesis filters and formed with equation (2-41).
- h is a KN vector which includes all analysis filters' coefficients.
- b is a KN zeros vector except for the Nth element =1 which implies the delay.

In addition, the error function for the squared magnitude response specifications for the analysis filter bank can be obtained using lemma 3 from Chapter 2 by:

$$E_{m1} = \sum_{k=1}^{K} \sum_{i=1}^{\rho} \left(\mathrm{h}_{\mathrm{k}}^{\mathrm{T}} \mathcal{R}(\omega_i) \mathrm{h}_{\mathrm{k}} - \gamma_k(\omega_i) \right)^2 \tag{3-3}$$

In a similar way, the error for the squared magnitude response in synthesis filter bank can be obtained by:

$$E_{m2} = \sum_{k=1}^{K} \sum_{i=1}^{\rho} \left(\mathrm{f}_{\mathrm{k}}^{\mathrm{T}} \mathcal{R}(\omega_i) \mathrm{f}_{\mathrm{k}} - \gamma_k(\omega_i) \right)^2 \tag{3-4}$$

where

- E_{m1} = The sum of the squared magnitude response errors of the analysis filters.
- E_{m2} = The sum of the squared magnitude response errors of the synthesis filters.
- $\mathcal{R}(\omega_i)$ the magnitude response that defined in equation (2-45).
- ρ is the frequency resolution for $\omega_i \in [0 \; to \; \pi]$ for each k^th^ filter.
- $$\gamma_k(\omega_i) = \begin{cases} 1 & if \quad \omega_i, within\ the\ passband\ area\ of\ \ H_k(e^{j\omega}), or\ \ F_k(e^{j\omega}) \\ 0 & if \quad \omega_i, within\ the\ stopband\ area\ of\ \ H_k(e^{j\omega}), or\ \ F_k(e^{j\omega}) \end{cases} \tag{3-5}$$

Equation (3-5) denotes the square of the desired magnitude response for the analysis and synthesis filters.

The equations defined in (3-1), (3-2), (3-3) and (3-4) will be used in the optimization techniques introduced in the next section in this chapter.

3.3 Perfect reconstruction error and magnitude response errors formulation.

This section highlights five different practical optimization schemes. The purpose of these optimization schemes is to design filter banks subject to a given allowable aliasing and distortion errors, and also taking into consideration the magnitude response design specifications of the analysis and synthesis filters. These optimization algorithms are used to find the optimum FIR analysis and synthesis filters' coefficients for a filter bank.[47].

3.3.1 Algorithm 1: Filter bank design as unconstrained nonlinear optimization problem

In the first algorithm, the motivation is to design a filter bank that achieves almost perfect reconstruction and satisfies the required specifications for the Magnitude Response of the Analysis Filters (MRAF). The phase response of each individual filter is not considered in the design; thus, it may not have a linear phase. However, it is assumed that the PR conditions guarantee the linear phase of the whole filter bank design.[28].

Consider the design of filter bank as in Figure 2-6 with sampling rates $\{n_1, n_2\ n_3, \ldots, and\ n_K\}$ that satisfy the critical sampling condition in equation (2-10). Each kth sub-band analysis filter is defined as $h_k = [\mathrm{h_{k,0}}, \mathrm{h_{k,1}}, \mathrm{h_{k,2}}, \ldots, \mathrm{h_{k,N-1}}]^{\mathrm{T}}$ and each kth sub-band synthesis filter is defined as $f_k^T = [f_{k,0}, f_{k,1}, f_{k,2}, \ldots, f_{k,N-1}]$. It is Assumed that each synthesis filter has the same frequency specifications as the corresponding analysis filter.

Designing such a filter bank can be carried out using the two error functions introduced in equations (3-1) and (3-3). As a result, the total performance index is the summation of these two error functions [28]:

$$J = w_{pr}E_{pr1} + w_m E_{m1} \tag{3-6}$$

The performance index J consists of two terms, one term depends on the perfect reconstruction E_{pr}, and the second term depends on the Magnitude Response of the Analysis Filters (MRAF), which is E_{m1}.

By defining

$$E_{pr1} = \|Hf - b\|^2$$

$$E_{m1} = \sum_{k=1}^{K}\sum_{i=1}^{\rho}\left(\mathrm{h}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega_i)\mathrm{h}_\mathrm{k} - \gamma_k(\omega_{\ i})\right)^2$$

equation (3-6) can be expanded as:

$$J = w_{pr}\|Hf - b\|^2 + w_m \sum_{k=1}^{K}\sum_{i=1}^{\rho}\left(\mathrm{h}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega_i)\mathrm{h}_\mathrm{k} - \gamma_k(\omega_{\ i})\right)^2 \quad (3\text{-}7)$$

- J is the performance index for the sum of PR and MRAF.
- w_{pr} is the weight cost function for perfect reconstruction error.
- w_m is the cost function weight for the magnate response specification error.
- $\mathcal{R}(\omega)$ and $\gamma_k(\omega_{\ i})$ are defined in equations (2-45), and (3-5), respectively.
- ρ is the frequency resolution of $\omega_i \in [0\ to\ \pi]$, and used for the kth sub-band filter.

The PR conditions for the FIR analysis and synthesis filters are expressed as a one set of linear equations, and the MRAF is formulated as a quadratic function. In this algorithm, the optimization variables are the analysis filters' coefficients. However, the synthesis filters' coefficients are also optimized as the least square solution of equation (3-1) [28]. Figure 3-1 shows the main steps for algorithm 1, which are used iteratively to minimize the performance index J.

Algorithm 1 [48] is implemented using MATLAB function *fminunc;* this function solves an unconstrained nonlinear optimization problem using BFGS Quasi-Newton method [49]. The performance of this algorithm is evaluated in Chapter 4.

Design specifications:

- N is the number of coefficients of the analysis filters.
- The set of sampling rates, all sampling rates are integers.
- Ratio is the frequency ratio occupied by each individual analysis filter over the normalized frequency [0 to π].
- ρ is the number of frequency grid points with a uniform distribution within stopband and passband areas over the normalized frequency [0 to π], and it is used in Eq. (3-3).
- Set weights for perfect reconstruction and magnitude response characteristics.
- Maximum number of iterations, or performance index error tolerance.

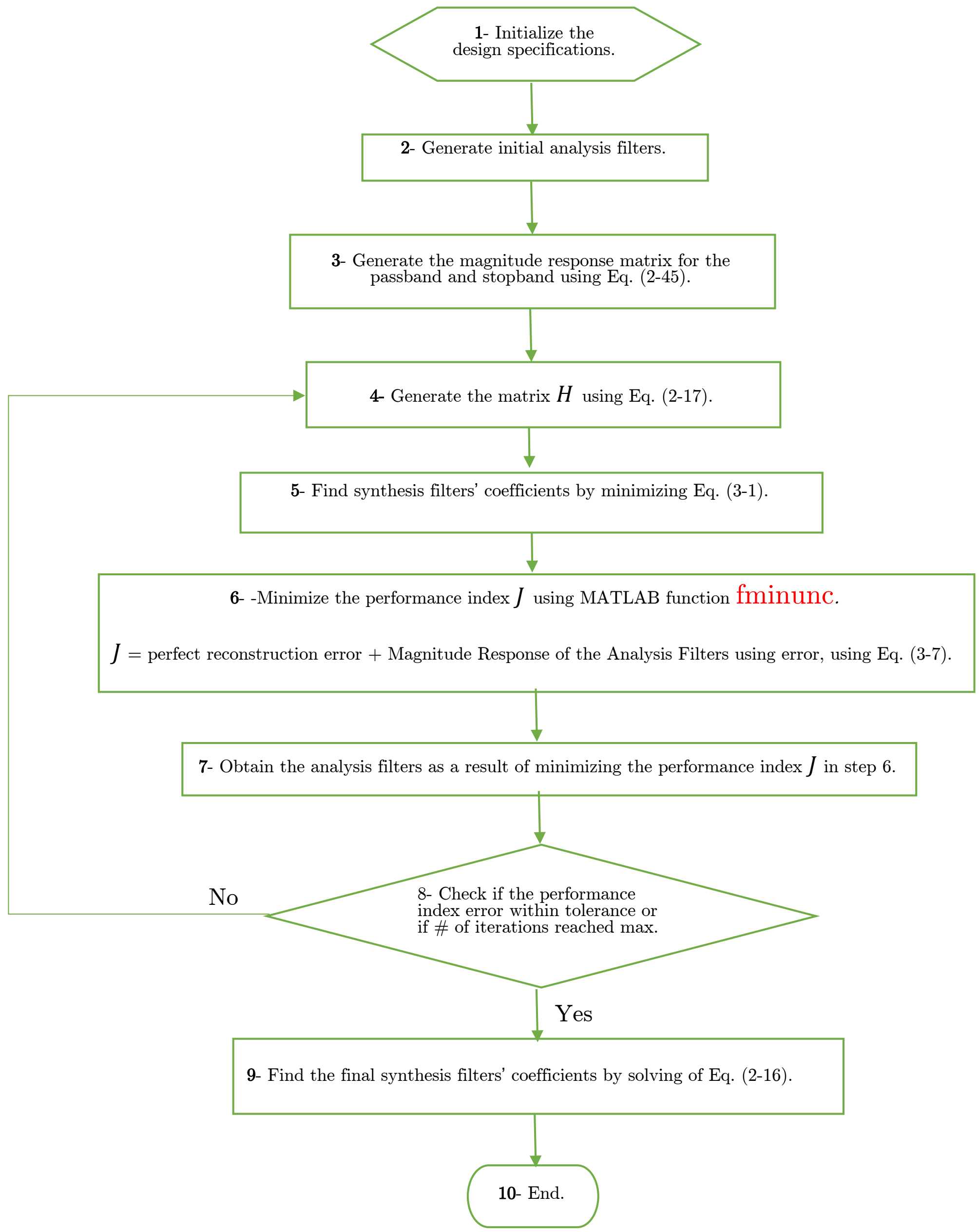

Figure 3-1 Main steps of algorithms 1, and 3.

3.3.2 Algorithm 2: Filter bank design using quadratic optimization problem

The motivation in the second algorithm is to design a filter bank that achieves NPR and meets the required magnitude response specifications. However, in this algorithm only the stopband magnitude responses in all sub-bands' filters are considered, and the passband magnitude responses are not included in the design procedures. Based on this, the performance index introduced earlier in equation (3-7) can be formulated as unconstrained convex quadratic objective functions. These quadratic objective functions are two performance indices, the first is with respect to the synthesis filters' coefficients while the analysis filters are fixed, and the second with respect to the synthesis filters' coefficients while the analysis filters are fixed [29].

$$J_f = w_{pr1}\|Hf - b\|^2 + w_{m1} \sum_{k=1}^{K} \sum_{i=1}^{\rho} \left(\mathrm{f}_\mathrm{k}^\mathrm{T} \mathcal{R}(\omega_i)\mathrm{f}_\mathrm{k} - \gamma_k(\omega_i)\right)^2 \quad (3\text{-}8)$$

and

$$J_h = w_{pr2}\|Fh - b\|^2 + w_{m2} \sum_{k=1}^{K} \sum_{i=1}^{\rho} \left(\mathrm{h}_\mathrm{k}^\mathrm{T} \mathcal{R}(\omega_i)\mathrm{h}_\mathrm{k} - \gamma_k(\omega_i)\right)^2 \quad (3\text{-}9)$$

where

- J_f is the performance index with respect to the synthesis filters.
- J_h is the performance index with respect to the analysis filters.
- H is a $M(2N-1)xKN$ matrix which includes the analysis filters formed with equation (2-17).
- F is a $M(2N-1)xKN$ matrix which includes the synthesis filters formed with equation (2-41).
- f is a KN column vector which includes all synthesis filter coefficients .
- h is a KN column vector which includes all analysis filter coefficients.
- b is a KN zeros column vector, except for the Nth element =1.
- $\gamma_k(\omega_i)$ is defined in equation (3-5).

From Equation (3-5) and the performance index introduced in equation (3-8). The square of the desired passband magnitude response is $\gamma_k(\omega_i) \approx 1$ in the k^{th} filter when $\omega_i \in \Omega_k^{pb}$, and square of desired stopband magnitude response is $\gamma_k(\omega_i) \approx 0$ when $\omega_i \in \Omega_k^{sb}$.

where

Ω_k^{pb} is the passband area of the k^th sub-band filter

Ω_k^{sb} is the stopband area of the k^th sub-band filter

By neglecting the magnitude response of the passband, the term $\gamma_k(\omega_i) \approx 1$ disappears from the performance index in equations (3-8), and the squared stopbands magnitude response characteristics for all filters remains present [29]. This leads to introducing the new two performance indices as follows:

$$J_f = w_{pr1}\|Hf - b\|^2 + w_{m1}\sum_{k=1}^{K}\sum_{i=1}^{\rho} \mathrm{f}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega_i)\mathrm{f}_\mathrm{k} \tag{3-10}$$

and by expanding the PR least square as:

$$\|Hf - b\|^2 = \frac{1}{2} f^T H^T H f - 2 f^T H^T b + b^T b \tag{3-11}$$

the performance index in equation (3-10) becomes:

$$J_f = w_{pr1}(\frac{1}{2} f^T H^T H f - 2 f^T H^T b + b^T b) + w_{m1}\sum_{K=1}^{K}\sum_{i=1}^{\rho} \mathrm{f}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega_i)\mathrm{f}_\mathrm{k} \tag{3-12}$$

Moreover, the two summations in equation (3-12) represent the summation of the stopband magnitude response for the k^th sub-band filter, and it can be expressed using one KN x KN matrix using the following two steps:

Step 1 - Let $\mathrm{f}_\mathrm{k}^\mathrm{T} = [\mathrm{f}_{\mathrm{k},0}, \mathrm{f}_{\mathrm{k},1}, \mathrm{f}_{\mathrm{k},2}, \dots, \mathrm{f}_{\mathrm{k},\mathrm{N}-1}]$ represents the kth sub-band synthesis filter's coefficients, and the square of its stopband magnitude response at ω_i is defined as follows:

$$\Sigma^{(k)} = \begin{bmatrix} \mathrm{f}_{\mathrm{k},0}, \\ \mathrm{f}_{\mathrm{k},1} \\ \mathrm{f}_{\mathrm{k},2} \\ \vdots \\ \mathrm{f}_{\mathrm{k},\mathrm{N}-1} \end{bmatrix}^T \begin{bmatrix} 1 & \cos\omega_i & \cos 2\omega_i & \cdots & \cos(N-1)\omega_i \\ \cos\omega_i & 1 & \cos\omega_i & \cdots & \cos(N-2)\omega_i \\ \cos 2\omega_i & \cos\omega_i & 1 & \cdots & \cos(N-3)\omega_i \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \cos(N-1)\omega_i & \cos(N-2)\omega_i & \cos(N-3)\omega_i & \cdots & 1 \end{bmatrix}_{N\,x\,N} \begin{bmatrix} \mathrm{f}_{\mathrm{k},0}, \\ \mathrm{f}_{\mathrm{k},1} \\ \mathrm{f}_{\mathrm{k},2} \\ \vdots \\ \mathrm{f}_{\mathrm{k},\mathrm{N}-1} \end{bmatrix} \tag{3-13}$$

where $\omega_i \in \Omega_k^{sp}$, Ω_k^{sp} is the stopband area of the k[th] sub-and filter over the normalized frequency [0 to π], and $\mathcal{R}(\omega_i)$ is defined in (2-45).

For all $\omega_i \in \Omega_k^{sp}$, the sum of $\mathcal{R}(\omega_i)$ matrices can be introduced as one matrix $\hat{\Sigma}^{(k)}$

$$\hat{\Sigma}^{(k)} = \sum_{i=1}^{\rho} \begin{bmatrix} 1 & cos\,\omega_i & cos\;2\omega_i & \cdots & cos(N-1)\omega_i \\ cos\,\omega_i & 1 & cos\,\omega_i & \cdots & cos(N-2)\omega_i \\ cos\;2\omega_i & cos\,\omega_i & 1 & \cdots & cos(N-3)\omega_i \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ cos(N-1)\omega_i & cos(N-2)\omega_i & cos(N-3)\omega_i & \cdots & 1 \end{bmatrix} \quad (3\text{-}14)$$

where

$i = 1, 2,3 \ldots\ \rho$ represents the number of frequency grid points at this k[th] sub-band stopband area. ρ also is introduced in equations (3-3) and (3-4) as the frequency resolution.

Therefore, for all $\omega_i \in \Omega_k^{sp}$, it can be shown that the square of the magnitude response for each k[th] sub-band can be obtained as follows:

$$\mathrm{f}_\mathrm{k}^\mathrm{T}\hat{\Sigma}^{(k)}\mathrm{f}_\mathrm{k} \quad (3\text{-}15)$$

Step 2 - The square of the stopband magnitude responses for all sub-band filters can be obtained using the following matrix:

$$\hat{\Psi} = \begin{bmatrix} \hat{\Sigma}^{(1)} & 0^{N\,x\,N} & \cdots & 0^{N\,x\,N} \\ 0^{N\,x\,N} & \hat{\Sigma}^{(2)} & \ldots & 0^{N\,x\,N} \\ \vdots & \vdots & \ddots & \vdots \\ 0^{N\,x\,N} & 0^{N\,x\,N} & \ldots & \hat{\Sigma}^{(K)} \end{bmatrix}_{KN\,x\,KN} \quad (3\text{-}16)$$

where

- $\hat{\Sigma}^{(kth)}$ is the sum of $\mathcal{R}(\omega_i)$ matrices for all $\omega_i \in \Omega_k^{sp}$ in the k[th] sub-band filter.

- $0^{N\,x\,N}$ is a sub-matrix which all its elements are zeros, and it appears when the k[th] sub-band filter's stopband magnitude response is not present.

Thus, the magnitude response of all synthesis filters stopband can be represented as:

$$\Psi_f = \left[f_1^T, f_2^T, f_3^T, \dots, f_K^T\right] \begin{bmatrix} \hat{\Sigma}^{(1)} & 0^{N\,x\,N} & \cdots & 0^{N\,x\,N} & 0^{N\,x\,N} \\ 0^{N\,x\,N} & \hat{\Sigma}^{(2)} & \dots & 0^{N\,x\,N} & 0^{N\,x\,N} \\ 0^{N\,x\,N} & 0^{N\,x\,N} & \hat{\Sigma}^{(3)} & 0^{N\,x\,N} & 0^{N\,x\,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0^{N\,x\,N} & 0^{N\,x\,N} & \cdots & \cdots & \hat{\Sigma}^{(K)} \end{bmatrix}_{KN\,x\,KN} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_k \end{bmatrix} \quad (3\text{-}17)$$

Using $f = \left[f_1^T, f_2^T, f_3^T, \dots, f_K^T\right]^T$, the total squared stopband responses for all synthesis filters can be rewritten as:

$$\Psi_f = f^T \hat{\Psi} f \quad (3\text{-}18)$$

After, formulating the stopband magnitude response of the synthesis filter bank as a quadratic term, equation (3-12) can be rewritten as follows:

$$J_f = w_{pr1}\left(\frac{1}{2} f^T H^T H f - 2 f^T H^T b + b^T b\right) + w_{m1} f^T \hat{\Psi} f \quad (3\text{-}19)$$

Defining:

$$Q_h = 2\left(w_{pr1} H^T H + w_{m1} \hat{\Psi}\right) \quad (3\text{-}20)$$

$$p_h = -2\, w_{pr1} H^T b \quad (3\text{-}21)$$

$$c = w_{pr1} b^T b \quad (3\text{-}22)$$

equation (3-19) can be simplified and expressed as an unconstrained quadratic problem with respect to synthesis filters coefficients

$$\min_f \quad J_f = \frac{1}{2} f^T Q_h f + f^T p_h + c \quad (3\text{-}23)$$

and a global minimizer can be found with:

$$f = -Q_h^{-1}\, p_h \tag{3-24}$$

The lower subscript (h) in equations (3-20) and (3-21) means that the analysis filter coefficients used in these terms are obtained from the previous iteration of minimizing the error with respect to analysis filters coefficients.

Similarly, for equation (3-9), by following the same steps in equations from (3-11) to (3-22), the three terms Q_f , p_f, and Ψ_h can be defined as follows:

$$\Psi_h = h^T \widehat{\Psi} h \tag{3-25}$$

$$Q_f = 2\left(w_{pr2}\, F^T F + w_{m2}\, \widehat{\Psi}\right) \tag{3-26}$$

$$p_f = -2\, w_{pr2}\, F^T\, b \tag{3-27}$$

Using the same constant c defined in equation (3-22), and the 1x KN vector $h = \left[\mathrm{h}_1^{\mathrm{T}}, \mathrm{h}_2^{\mathrm{T}}, \mathrm{h}_3^{\mathrm{T}}, \dots, \mathrm{h}_{\mathrm{K}}^{\mathrm{T}}\right]^T$ which includes all sub-band analysis filters' coefficients. The performance index in equation (3-9) with respect to analysis filters coefficients can be rewritten as an unconstrained quadratic objective function J_h

$$\min_h \; J_h = \frac{1}{2} h^T Q_f h + h^T p_f + c \tag{3-28}$$

and the minimizer can be found with

$$h = -Q_f^{-1}\, p_f \tag{3-29}$$

This concludes that designing a filter bank that meets specific design requirements can be completed using equations (3-23) and (3-28) iteratively. Algorithm 2 [50] minimizes these two performance indices following the main steps introduced in Figure 3-2.The solutions are global minimizers, since Q is symmetric positive semidefinite matrix and the problem is solved as Quadratic Programming (QP) [49].

Design Specifications:

- N is the number of coefficients of the analysis filters.
- The set of sampling rates, all sampling rates are integers
- Ratio is the frequency ratio occupied by each individual analysis filter over the normalized frequency [0 to π].
- ρ_{sp} is the number of frequency grid points with a uniform distribution within each filter's stopband area for $k = 1,2,3,\ldots,K$ used in equation (3-14).
- Set weights for perfect reconstructions and magnitude responses characteristics.
- Maximum number of iterations, or performance index error tolerance.

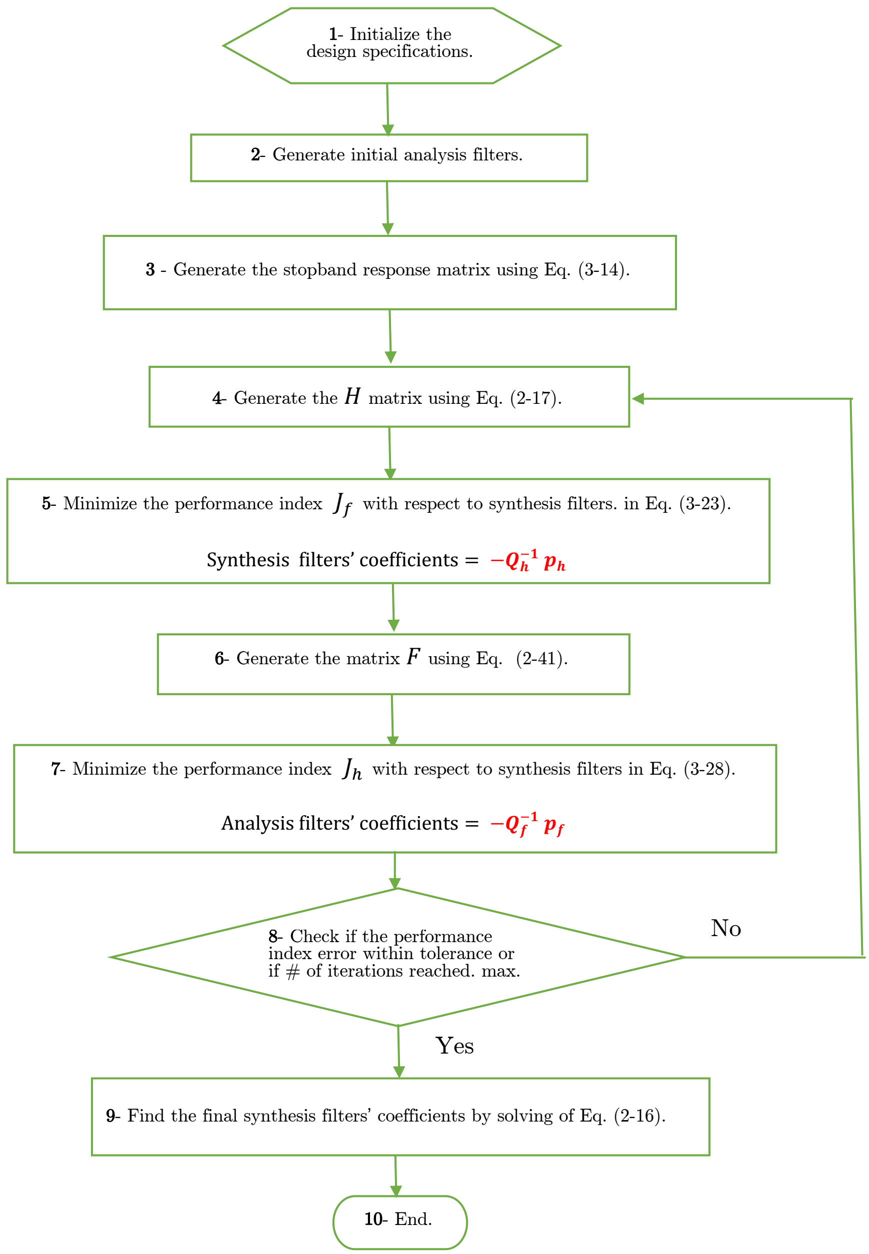

Figure 3-2 Main steps of Algorithms 2.

3.3.3 Algorithm 3: Filter bank design as an unconstrained nonlinear optimization problem (Modified version of Algorithm 1)

The third algorithm is a modified version of Algorithm 1. The motivation behind Algorithm 3 is to improve the stopband response characteristics of the synthesis filters, which are not included in the performance index in Algorithm 1. This is done by replacing the PR error function E_{pr} in the performance index in equation (3-7) with the performance index J_f introduced in equation (3-23). As shown in section 3.3.2, equation (3-23) is an objective function which includes the PR error for the filter bank, and the stopband Magnitude Responses of the Synthesis Filters (MRSF) formed as quadratic function. Therefore, the new modified performance index in Algorithm 3 is defined as follows:

$$\bar{J} = w_{pr}\,\bar{E}_{pr1} + w_m E_{m1} \qquad (3\text{-}30)$$

where

$\bar{E}_{pr1}$ is a modified PR error presented in equation (3-1), the synthesis filters coefficients are obtained as a result of minimizing the performance index in equation (3-23).

$E_{m1} = \sum_{k=1}^{K}\sum_{i=1}^{\rho}\left(\mathrm{h}_\mathrm{k}^\mathrm{T}\mathcal{R}(\omega_i)\mathrm{h}_\mathrm{k} - \gamma_k(\omega_i)\right)^2$, and it is defined in equation (3-3).

w_{pr} and w_m are the optimization weights for the perfect reconstruction error and magnitude response error, respectively.

The main optimization steps in algorithm 3 are the same steps for algorithm 1 shown in Figure 3-1. However, the two differences are replacing equation (3-1) in step 5 with equation (3-23) and replacing equation (3-7) in step 6 with equation (3-30).

Minimizing the error in the stopband magnitude response for the synthesis filters, and MRAF may yield improvement the magnitude responses in the designed sub-band filters, and possible in PR.

Similar to algorithm 1, algorithm 3 is implemented by using the MATLAB function *fminunc*. This function is used to minimize the modified performance index error $\bar{J}$ and find the optimum analysis and synthesis filters' coefficients. The performance of Algorithm 3 will be evaluated and compared to the other algorithms in the next Chapter.

The next two algorithms use constrained optimization to design the filter bank shown in Figure 2-6. The purpose is to design a filter bank that achieves almost perfect reconstruction and meet the magnitude response design specifications.

3.3.4 Algorithm 4: Filter bank design using semi-infinite optimization problem

The motivation in the fourth algorithm is to design a filter bank that meets the same design requirements as in algorithms 1, 2, and 3. In this section, Algorithm 4 minimizes an objective function that includes the perfect reconstruction conditions subject to two types of constraints. These constraints are formulated for the analysis and synthesis as following, the passband magnitude response for each kth sub-band filter is formulated as semi-infinite quadratic constraints [21, 22], and the stopband magnitude responses of the filter banks are formulated as quadratic constraints. The Semi-infinite programming is discussed broadly in [51-54]

In algorithms 4 the stopband magnitude responses for the analysis and synthesis filter banks in equations (3-18) and (3-25) are introduced as quadratic constraints instead of being included in the performance index as in Algorithms 2, and this can be implied from equations (3-8) and (3-9) in algorithms 2.

For the kth sub-band synthesis filter $\mathrm{f}_\mathrm{k}^\mathrm{T}$, and passband area Ω_k^{pb} over the normalized frequency [0 to π]. This synthesis filter has a squared passband magnitude response at $\omega_i \in \Omega_k^{pb}$ as shown in equation (3-31). Using the semi-infinite discretization method mentioned above, the passband magnitude responses of this kth sub-band synthesis filter can be introduced as a function of two variables, ω and the filter coefficients. Further, the kth passband area Ω_k^{pb} is discretized uniformly with the number of frequency grid points. The number of pointes is defined as ρ and named as frequency resolution in equation (3-4). In this algorithm ρ is the same for all passband magnitude responses for analysis and synthesis filters [55].

Therefore, the square of the passband magnitude response for the kth sub-band synthesis filters at each ω_i is defined as follows:

$$\overline{\overline{\psi_f}}(\mathrm{f}_\mathrm{k}, \omega_i) = \mathrm{f}_\mathrm{k}^\mathrm{T} \mathcal{R}(\omega_i) \mathrm{f}_\mathrm{k} \approx 1 \qquad (3\text{-}31)$$

likewise, the square of the passband magnitude response for the k[th] sub-band analysis filters at each ω_i is defined as follows:

$$\overline{\overline{\psi_h}}(\mathrm{h_k}, \omega_i) = \mathrm{h_k^T} \mathcal{R}(\omega_i)\, \mathrm{h_k} \approx 1 \qquad (3\text{-}32)$$

Consider δp and δs implies the maximum passband and stopband constraints tolerance, respectively. As result, using the two performance indices for the perfect reconstruction conditions introduced in equations (3-1) and (3-2), the proposed optimization technique (algorithm 4) can be formulated as a constrained least square SIP.

Algorithm 4 uses two steps. First, it minimizes the cost function of the perfect reconstruction with respect to the synthesis filters coefficients, and subject to the square of synthesis filters magnitude response characteristics constraints as follows:

$$\min_f \quad \|Hf - b\|^2 \qquad (3\text{-}33)$$

$$subject\ to: \quad f^T \widehat{\Psi} f \leq \quad \delta s$$

$$\overline{\overline{\psi_f}}(\mathrm{f_1}, \omega_i) = 1 - \delta p \leq \mathrm{f_1^T} \mathcal{R}(\omega_i)\mathrm{f_1} \leq 1 + \delta p \qquad \omega_i \in \Omega_1^{pb}$$

$$\overline{\overline{\psi_f}}(\mathrm{f_2}, \omega_i) = 1 - \delta p \leq \mathrm{f_2^T} \mathcal{R}(\omega_i)\mathrm{f_2} \leq 1 + \delta p \qquad \omega_i \in \Omega_2^{pb}$$

$$\vdots$$

$$\overline{\overline{\psi_f}}(\mathrm{f_K}, \omega_i) = 1 - \delta p \leq \mathrm{f_K^T} \mathcal{R}(\omega_i)\mathrm{f_K} \leq 1 + \delta p \qquad \omega_i \in \Omega_K^{pb}$$

where

- H a matrix that includes the analysis filters formed with equation (2-17).
- f is a KN column vector which includes all synthesis filter coefficients.
- b is a KN zeros column vector, except for the Nth element =1 which for the delay.
- K is the number of sub-bands for (k= 1,2,3, …, K), and $\mathrm{f_k}$ k[th] synthesis filter.
- $\widehat{\Psi}$ is KN x KN matrix defined in equation (3-16), and $\mathcal{R}(\omega_i)$ is N x N matrix defined in (2-45).
- Ω_{k}^{pb} is the passband magnitude responses for the kth sub-band filter over the normalized frequency [0 to π].
- δs is the stopband constraints tolerance.
- δp is the passband constraints tolerance and it is the same for all sub-bands.

After that, the obtained synthesis filters coefficients are used to construct the matrix in equation in the (2-41), and the performance index in equation (3-2) can be formulated.

The second step in algorithm 4, it minimizes this performance index of perfect reconstruction with respect to the analysis filters coefficients subject to the square of analysis magnitude response characteristics constraints as follows:

$$\min_{h} \quad \|Fh - b\|^2 \tag{3-34}$$

$$\textit{subject to}: \quad h^T \widehat{\Psi} h \leq \quad \delta s$$

$$\overline{\overline{\psi_h}}(\mathrm{h}_1, \omega_i) = \; 1 - \delta p \; \leq \; \mathrm{h}_1^{\mathrm{T}} \mathcal{R}(\omega_i) \mathrm{h}_1 \; \leq 1 + \delta p \qquad \omega_i \in \Omega_1^{pb}$$

$$\overline{\overline{\psi_h}}(\mathrm{h}_2, \omega_i) = \; 1 - \delta p \; \leq \; \mathrm{h}_2^{\mathrm{T}} \mathcal{R}(\omega_i) \mathrm{h}_2 \; \leq 1 + \delta p \qquad \omega_i \in \Omega_2^{pb}$$

$$\vdots$$

$$\overline{\overline{\psi_h}}(\mathrm{h}_\mathrm{K}, \omega_i) = \; 1 - \delta p \; \leq \; \mathrm{h}_\mathrm{K}^{\mathrm{T}} \mathcal{R}(\omega_i) \mathrm{h}_\mathrm{K} \; \leq 1 + \delta p \qquad \omega_i \in \Omega_K^{pb}$$

where

- F a matrix that includes the synthesis filters formed with equation (2-41).
- h is a KN column vector which includes all analysis filter coefficients.
- h_k is the k^{th} sub-bands analysis filter.
- The rest of the variables are in equation (3-33)

The fourth algorithm is implemented using the MATLAB function *fseminf*. This function performs semi-infinite programming method and minimizes constrained problem with Sequential Quadratic Programming (SQP). The main steps for algorithm 4 are introduced in Figure 3-3., In in the flowchart for algorithm 5, step 5 refers to the optimization problem in equation (3-33), and step 7 refers to an optimization problem in equation (3-34). The performance of this algorithm will be evaluated in the next chapter.

Design Specifications:

- N is the number of coefficients of the analysis filters.
- The set of sampling rates, all sampling rates are integers
- Ratio is the frequency ratio occupied by each individual analysis filter over the normalized frequency [0 to π].
- ρ_{sp} is the number of frequency grid points with a uniform distribution within the each filter's stopband area for $k = 1,2,3,\ldots,K$ used in equation (3-14).
- ρ_{pb} is the number of frequency grid points with a uniform distribution within the each filter's passband area for $k = 1,2,3,\ldots,K$ in equations (3-31) and (3-32).
- Maximum number of iterations, or performance index error tolerance.

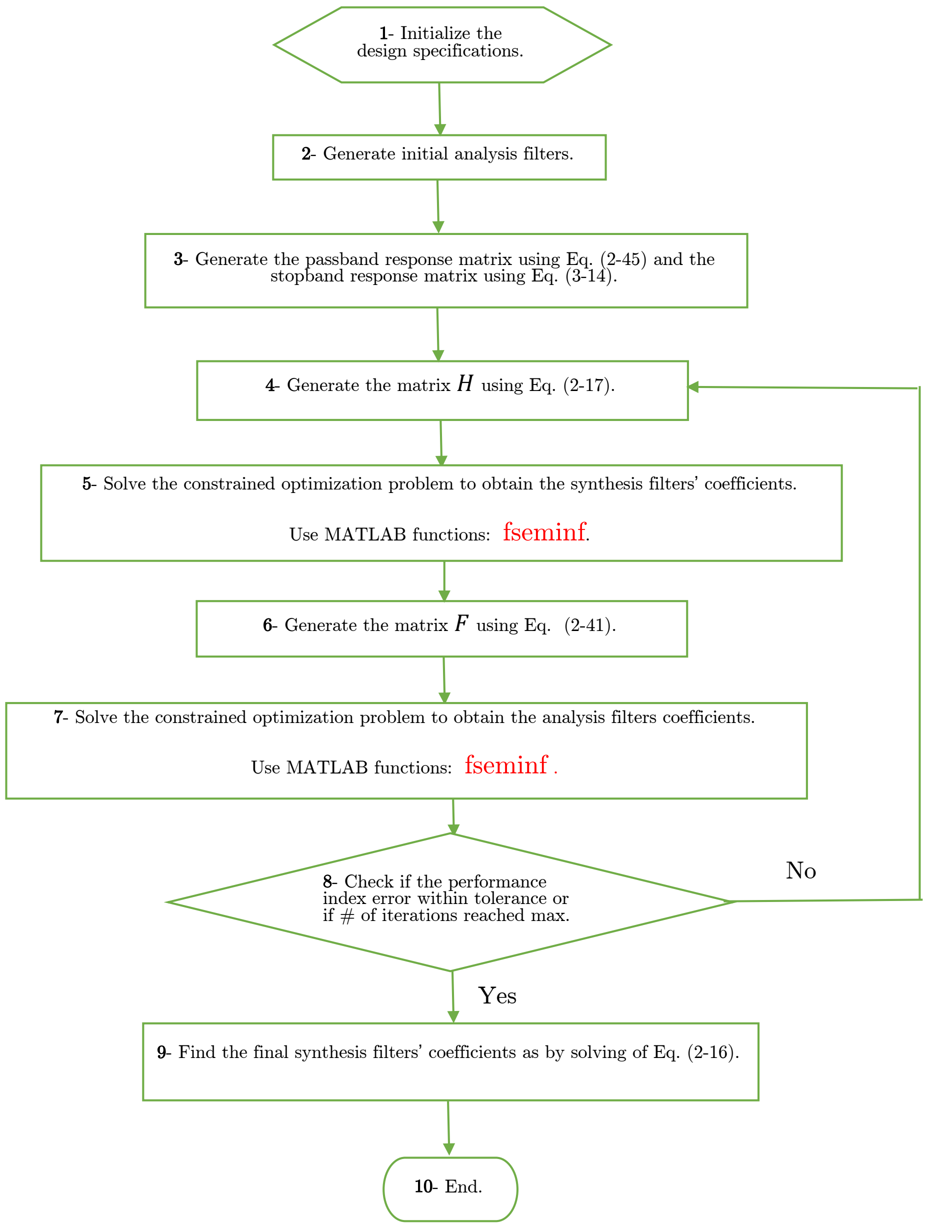

Figure 3-3 Main steps of Algorithms 4.

3.3.5 Filter bank design using constrained least squares optimization problem (Algorithm 5).

In this section, the objective is also to minimize the PR error and meet the magnitude response characteristic in the stopband for the same designs in the previous algorithms. In more detail, algorithm 5 minimizes two performance indices that include only the perfect reconstruction conditions subject to stopband magnitude response of the analysis or the synthesis filters constraints. The performance of algorithm 5 can be compared to the performance of algorithm 2, since both do not include the passband magnitude response characteristics in the optimization design.

The two performance indices introduced in equations (3-1) and (3-2) are minimized in two steps, *first* optimizing the performance index with respect to the synthesis filters' coefficients subject to the constraints of the stopband magnitude responses of these synthesis filters as follows:

$$\min_{f} \quad \|Hf - b\|^2 \tag{3-35}$$

$$subject\ to: \quad f^T \widehat{\Psi} f \leq \delta s$$

After that, the obtained synthesis filters' coefficients from this optimization process are used to construct the performance index and the stopband constraints.

Second, this performance index is minimized with respect to the analysis filters' coefficients subject to the constraints of the stopband magnitude responses of these analysis filters as follows:

$$\min_{h} \quad \|Fh - b\|^2 \tag{3-36}$$

$$subject\ to: \quad h^T \widehat{\Psi} h \leq \delta s$$

The stopband magnitude specifications for synthesis and analysis filters are defined using equations (3-13) and (3-21), respectively. The difference between the equations (3-35) and (3-36) in algorithm 5 and the equations (3-33) and (3-34) in algorithm 4 is the semi-infinite constraints for the passband magnitude response characteristics are not included. Therefore, this method can be solved as constraints least square programing CLS.

Design Specifications:

- N is the number of coefficients of the analysis filters.
- The set of sampling rates, all sampling rates are integers
- Ratio is the frequency ratio occupied by each individual analysis filter over the normalized frequency [0 to π].
- ρ_{sp} is the number of frequency grid points with a uniform distribution within the each filter's stopband area for k = 1,2, 3, …, K used in equation (3-14).
- Maximum number of iterations, or performance index error tolerance.

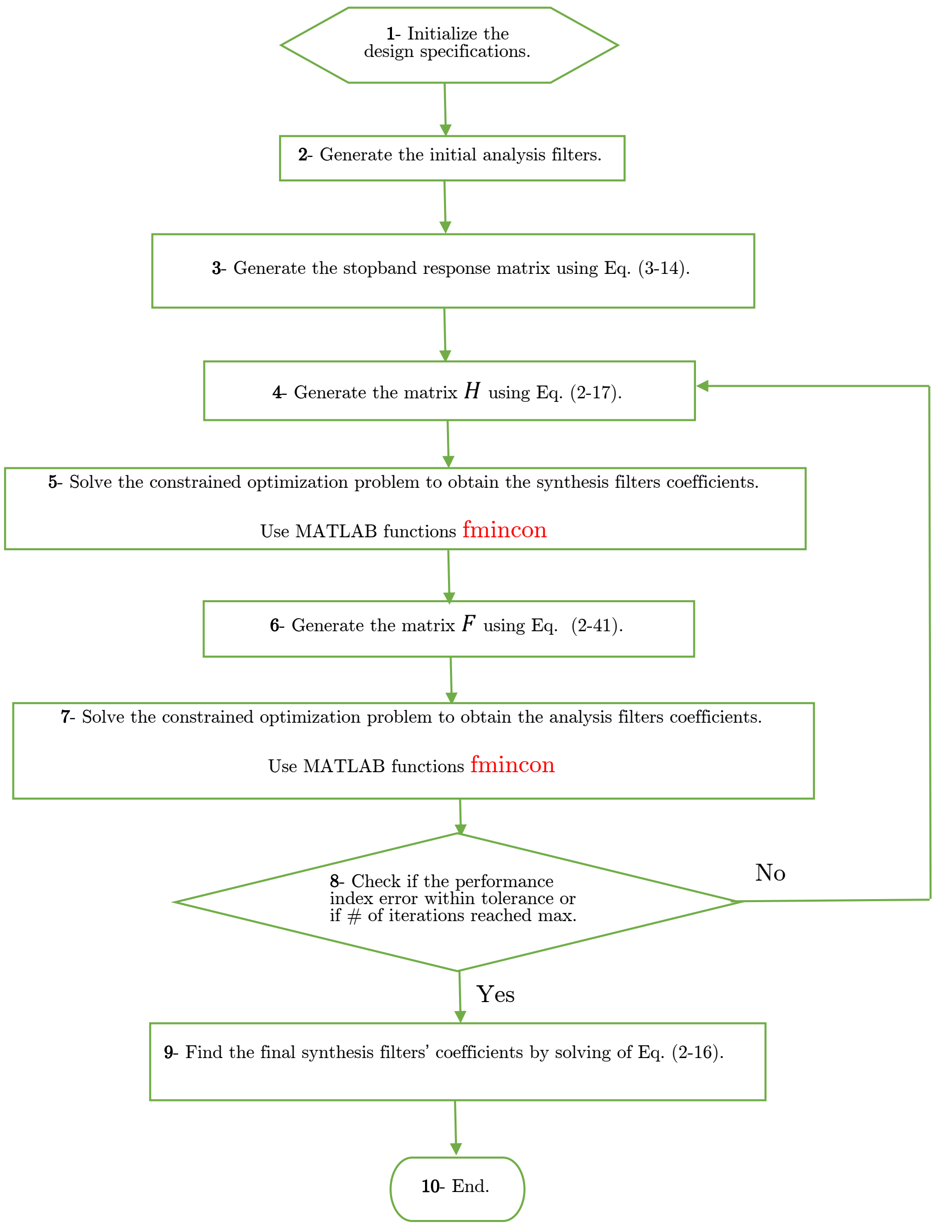

Figure 3-4 Main steps of Algorithms 5.

The fifth algorithm can be implemented using a MATLAB function called *fmincon*. This function is used to optimize an objective function subject to a set of constraints, and it uses SQP technique. The performance of this algorithm will also be evaluated in the next chapter.

In addition, Figure 3-4 shows the main steps in same steps. Algorithm 5 does not include the passband magnitude response in the filter bank design specifications. The other differences between algorithm 4 and algorithm 5 can be seen in step 5 and 7.

Algorithms 1, 2 and 3 have a similarity in the sense that the three algorithms deal with the perfect reconstruction conditions and magnitude response characteristics as part of the performance indices. However the difference is that algorithm 1 includes only the magnitude response of the analysis filters, algorithm 2 includes only the stopband magnitude response of both analysis filters and synthesis filters, and algorithm 3 includes the magnitude response of the analysis filters and only the stopband magnitude response of synthesis filters in the performance index. On the other hand, in algorithm 4, the performance index is the perfect reconstruction conditions minimized subject to the passband and stopband magnitude response characteristics, as constraints. Algorithm 5 is the same the fourth algorithm without the passband response constraints.

Also, the similarity between algorithm 2 and algorithm 5 is that consider only the stopband of the magnitude response characteristics of both analysis filters and synthesis filters. In algorithm 2, it includes stopband response for the filter banks in the performance index, while Algorithm 5 deals with them as constraints.

Figure 3-5 summarizes the five optimization techniques which are introduced in this chapter.

Algorithms' Structure for Filter Bank Design by Optimization

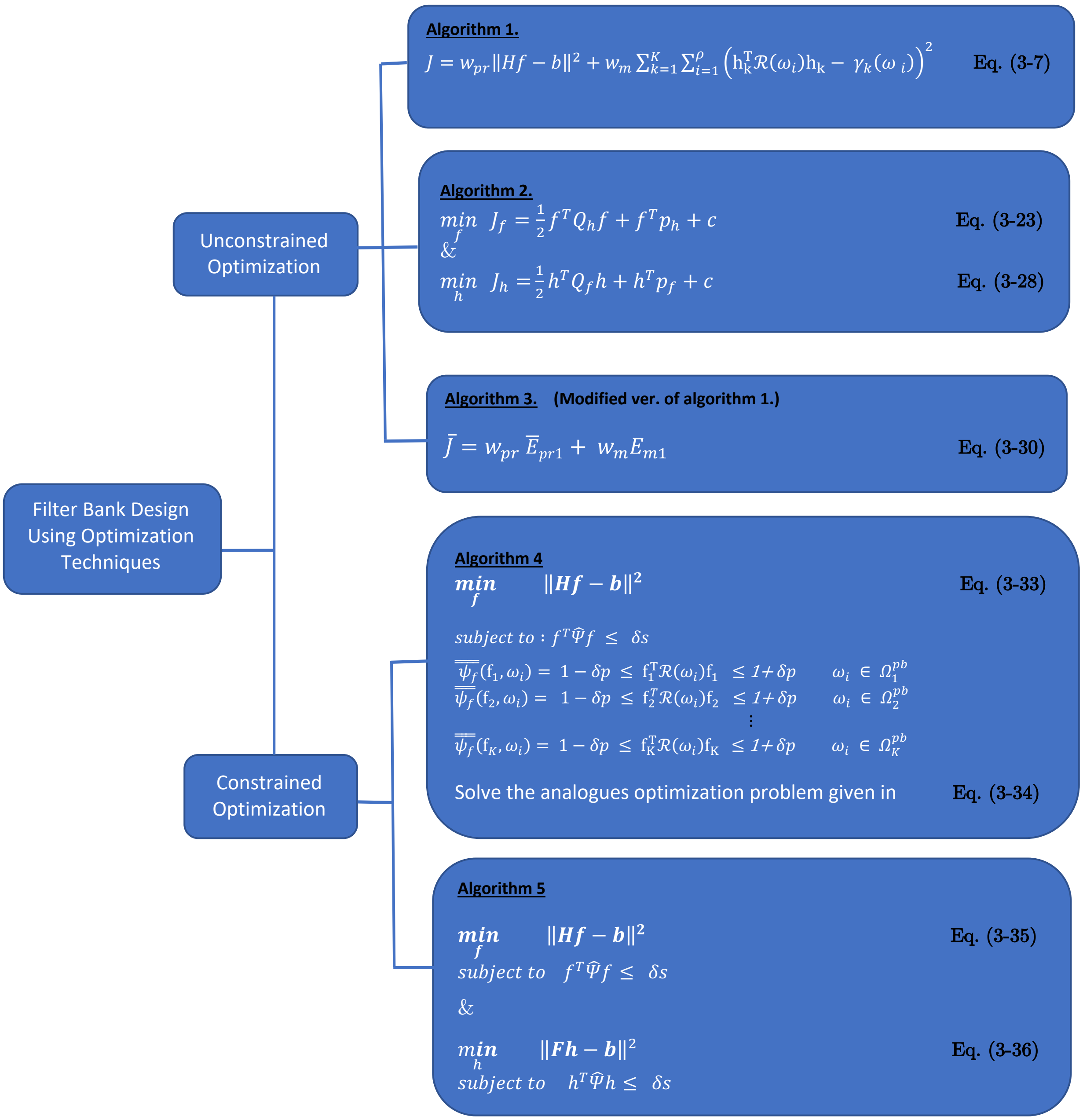

Figure 3-5 Algorithms for filter bank design by optimization.

3.4 Conclusion

In this chapter, five optimization algorithms for filter bank design are introduced, the first two algorithms reviewed from the proposed work introduced in [28, 29]. The approach is based on minimizing unconstrained nonlinear and quadratic optimization problems. Further, algorithm 3 is presented as a modified version of algorithms 1 and 2. By using a nested version that includes magnitude response of the analysis filters and the stopband response synthesis filters in the minimized performance index. As it will be shown in Chapter 4, this method helps to improve the stopband response characteristics in the final analysis and synthesis filters. Later in sections 3.3.4 and 3.3.5, constrained optimization techniques are presented. As shown, algorithm 4 uses semi-infinite constraints for the passband magnitude response specification of the filters, and quadratic constraints for the stopband magnitude response of the analysis and synthesis filter banks. The 5th algorithm is similar to algorithm 2 and 4. As in algorithm 2, It deals with the PR conditions and only the stopband response and, similar to algorithm 4, it minimizes performance index subject to constraints. The performance evaluation for these five algorithms will be shown and studied in Chapter 4, using different examples for uniform and compatible and incompatible NUFBs.

CHAPTER 4

Filter Banks Design and Results.

4.1 Introduction

In this chapter, the performance of the five algorithms presented in Chapter 3 is evaluated by designing various filter banks examples. These examples include UFBs, compatible NUFBs, and incompatible NUFBs. The criteria used for the evaluation are as follows: PR errors, stopband and passband magnitude response characteristics, convergence to the final results of the performance indices, and the computation time required for each optimization method.

4.2 Design Examples and Performance Analysis

The implementations of the five algorithms were completed using MATLAB (R2017a) Version: 9.2.0., and the results obtained using these five algorithms are presented with examples. For each example, the initial design specifications variables are outlined below:

1- N is the number of coefficients of the analysis filters. Analysis and synthesis filters have the same number of coefficients.
2- The set of sampling rates of the filter banks, all sampling rates used in this chapter are integers.
3- Transition factors are the areas of the transition bands in the analysis filters over the normalized frequency [0 to π], and these factors are used to define the passband and stopband cut-off frequencies for these sub-band filters.
4- Ratio is the frequency ratio occupied by each individual analysis filter over the normalized frequency [0 to π].

Consider a 4-band filter bank with a sampling rate set S = [4 4 4 4], ratios defined as $\left[\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right]$, and transition factors defined as [0.08 0.08 0.08 0.08]. The magnitude cut-off frequencies for the *first* filter are f_{c1} and f_{c2}, the area from 0 to f_{c1} is the passband region, the area from f_{c1} and f_{c2} is the transition region, and the area from f_{c2} to π is the stopband region. Thus, f_{c1} and f_{c2} are obtained by the following:

- $f_{c1} = (0.25 - 0.08) \times \pi = 0.5341.$
- $f_{c2} = (0.25 + 0.08) \times \pi = 1.0367.$

Similarly, the magnitude cut-off frequencies for the other sub-band filters for this particular design are defined as:

Table 4-1: Example for magnitude cut-off frequencies design specification.

Filter Type	Passband	Transition band/s	Stopband/s
Low pass	[0 0.5341]	[0.5341 1.0367]	[1.0367 π]
Band pass	[1.0367 1.3195]	[0.5341 1.0367], and [1.3195 1.8221]	[0 0.5341], and [1.8221 π]
Band Pass	[1.8221 2.1049]	[1.3195 1.8221], and [2.1049 2.6075]	[0 1.3195], and [2.6075 π]
High pass	[2.6075 π]	[2.1049 2.6075]	[0 2.1049]

5- In the obtained results, it is assumed that each synthesis filter has the same frequency specifications of the corresponding analysis filter.
6- Frequency resolution is the frequency grid points distributed within the filters' stopband and/or passband areas.
   a. ρ is the number of frequency grid points with a uniform distribution within all stopband and passband areas over the normalized frequency [0 to π]. ρ is used in algorithms 1 and 3.
   b. ρ_{sp} is the number of frequency grid points with a uniform distribution within the filter's stopband area. ρ_{sp} is used in algorithm 2, 4, and 5.
   c. ρ_{pb} is the number of frequency grid points with a uniform distribution within the filter's passband area. ρ_{pb} is used in algorithm 4.
7- Maximum number of iterations or the acceptable error tolerance.
8- Optional weights for the perfect reconstruction and magnitude response of analysis and synthesis filters. For our design in this chapter, all weights are set to 1.

The initial analysis FIR filters are designed using Parks-McClellan optimal equiripple method with a MATLAB function called firpm. The results for each design are FIR analysis and synthesis filters' coefficients. These results are presented with the following figures:

1. Filter banks' magnitude responses which are plotted over the normalized frequency spectrum [0 to π].
2. The distortion and aliasing errors introduced in equations (2-12) and (2-13). These errors are measured in dB and plotted over the normalized frequency spectrum [0 to π].

3. The values of the performance index with respect to the number of iterations for each algorithm. These five performance indices are defined in Chapter 3 with the following equations:
   - See equation (3-7) for algorithm 1.
   - See equation (3-23) and (3-28) for algorithm 2.
   - See equation (3-30) for algorithm 3.
   - See equation (3-33) and (3-34) algorithm 4.
   - See equation (3-35) and (3-36) algorithm 5.

The complete results from the five algorithms for each example are presented. For each example, the design specifications and some numerical results are summarized in a tabular form with the following:

1. Design specifications include Number of filter coefficients N, Transition factors, Ratio, and Frequency resolution/s.
2. Computation time for each design. The time presented in Red means that optimization is completed using Westgrid computer, and the time presented in **Black** means optimization is completed using a workstation with i7-7700 CPU @ 3.60 GHz, 16 GB RAM, and Linux 3.10.0 operating system.
3. Maximum initial and final aliasing and distortion errors. These errors are defined as the largest dB values over the normalized frequency [0 to π].
4. A_s is the minimum stopband attenuation measured in dB.
5. A_p is the passband ripple measured in dB.

Further, after each group of examples, the results are summarized in a table to facilitate comparisons.

4.2.1 Uniform filter bank design examples

In this section, UFB examples for the sampling rates sets [2 2] and [4 4 4 4] are designed using the five algorithms introduced in Chapter 3. The results of these designs are also compared with well-known and efficient algorithms presented in [56, 57] and [58, 59]. These UFB examples are included to show that the five algorithms can be used for such types of designs and to provide a good understanding of the performance of the five algorithms.

For the design of uniform filter banks, there are many existing methods in the literature. As an example, a method is suggested by Nguyen to design an M-channel pseudo-quadrature mirror filter bank [56, 57]. This method is based on finding a prototype filter subject to a number of quadratic constraints. A_s a result, the minimization process yields a small aliasing error. Another recent method is a fast approach that projected in [58]. This method solves a quadratic programming problem subject to linear equality constraints. Using an optimization technique based on QR or SVD decomposition, this algorithm is able to achieve the PR for a uniform filter bank. Unlike the algorithms presented in Chapter 3, the two mentioned algorithms in [56] and [58] use cosine modulated filter banks to design uniform filter banks.

Example 4-1 UFB [2 2] using algorithms 1 to 5

- **Algorithm 1**

Table 4-2: UFB [2 2] using algorithm 1.

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Frequency Resolution ρ	60
Computation Time	40.82 sec
Max. Initial Distortion error	-16.98 dB
Max. Initial Aliasing error	-16.85 dB
Max. Final Distortion error	- 89.57 dB
Max. Final Aliasing error	- 84.73 dB

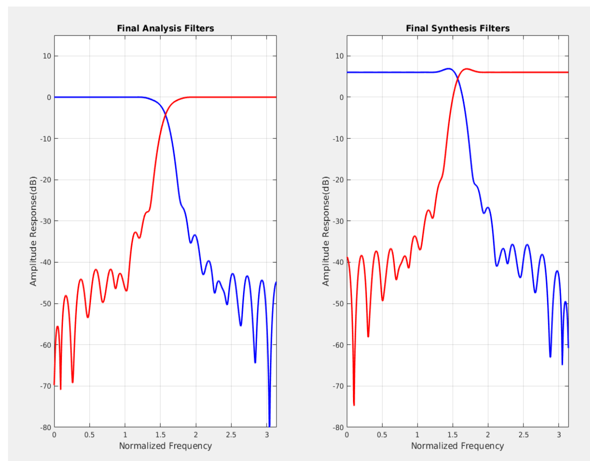

Figure 4-1-a

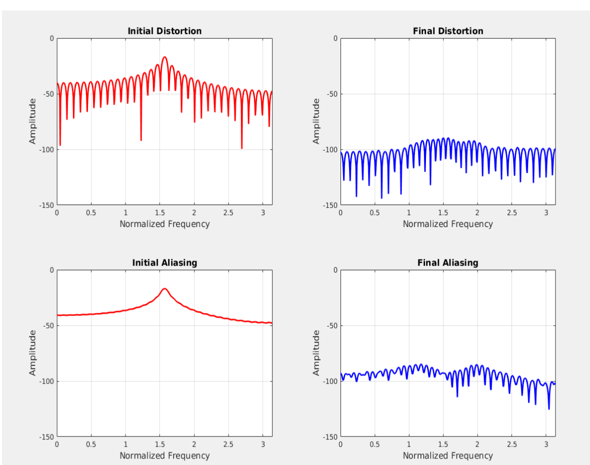

Figure 4-1-b

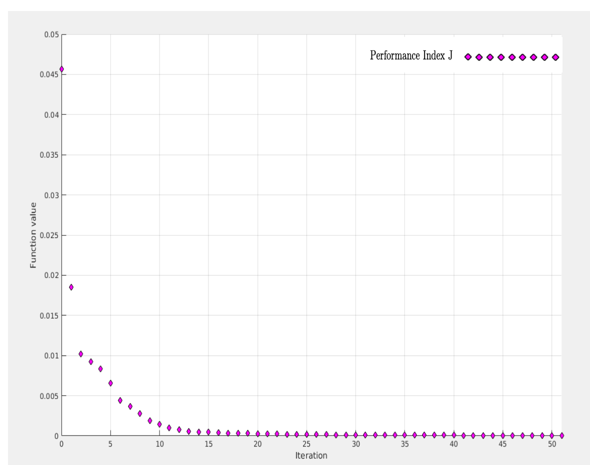

Figure 4-1-c

Figure 4-1 UFB [2 2] using algorithm 1. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors **c** – Performance index value at each iteration.

- **Algorithm 2**

Table 4-3: UFB [2 2] using algorithm 2.

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Frequency Resolution ρ_{sp}	60
Computation Time	1.62 sec
Max. Initial Distortion error	-16.98 dB
Max. Initial Aliasing error	-16.85 dB
Max. Final Distortion error	- 63.48 dB
Max. Final Aliasing error	- 64.53 dB

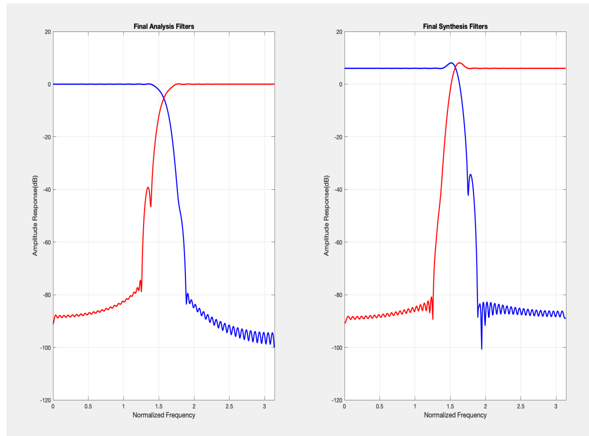

Figure 4-2-a

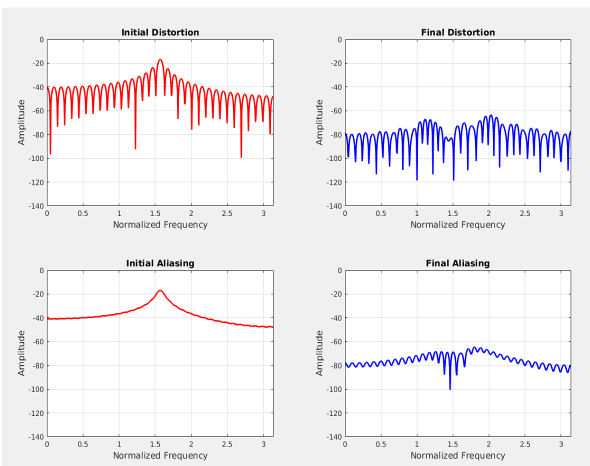

Figure 4-2-b

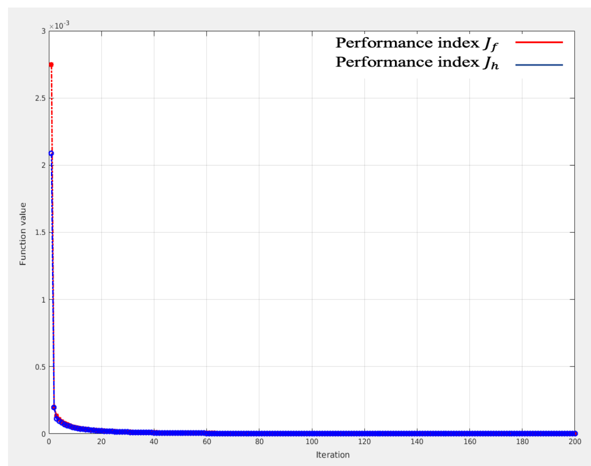

Figure 4-2-c

Figure 4-2 UFB [2 2] using algorithm 2. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

- **Algorithm 3**

Table 4-4: UFB [2 2] using algorithm 3.

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Frequency Resolution ρ	60
Computation Time	2:33 min
Max. Initial Distortion error	-16.98 dB
Max. Initial Aliasing error	-16.85 dB
Max. Final Distortion error	- 52.78 dB
Max. Final Aliasing error	- 47.73 dB

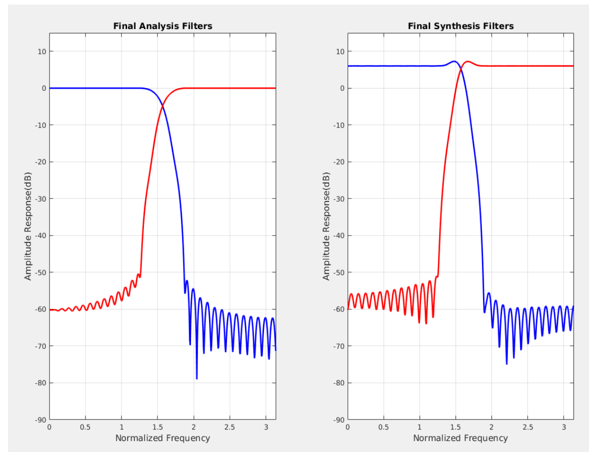

Figure 4-3-a

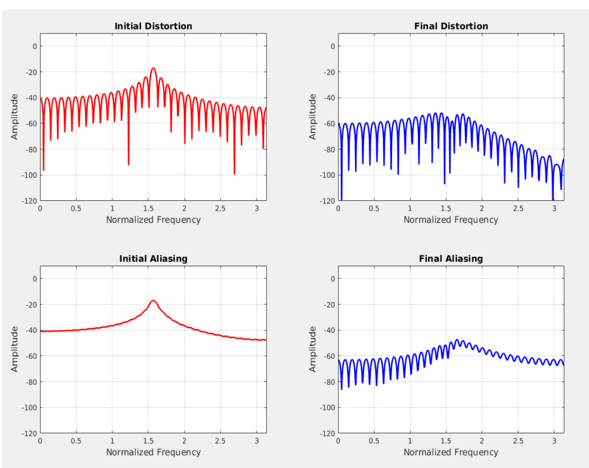

Figure 4-3-b

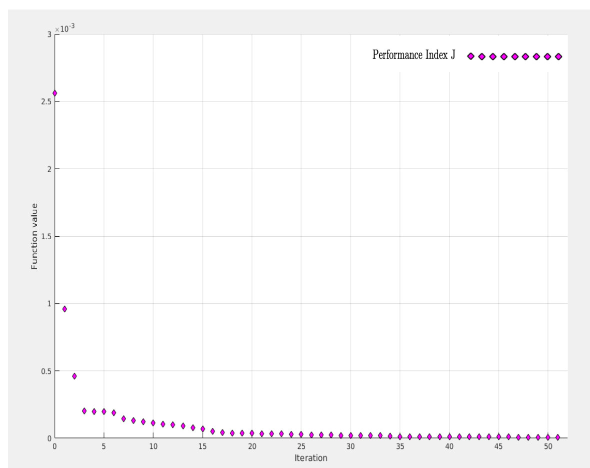

Figure 4-3-c

Figure 4-3 UFB [2 2] using algorithm 3. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

- **Algorithm 4**

Table 4-5: UFB [2 2] using algorithm 4.

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Frequency Resolution ρ_{sp}	30
Frequency Resolution ρ_{pb}	60
Computation Time	2:57 hr
Max. Initial Distortion error	-16.98 dB
Max. Initial Aliasing error	-16.85 dB
Max. Final Distortion error	- 40.38 dB
Max. Final Aliasing error	- 36.15 dB

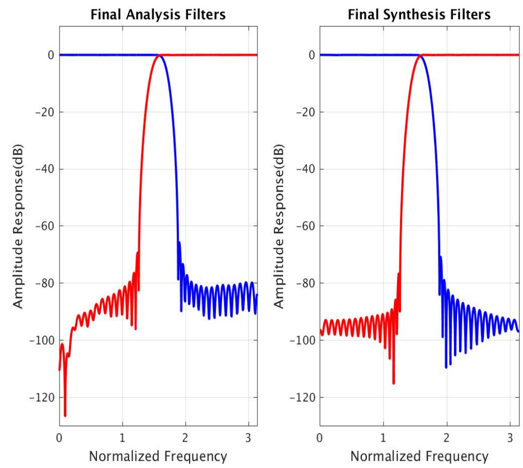

Figure 4-4-a

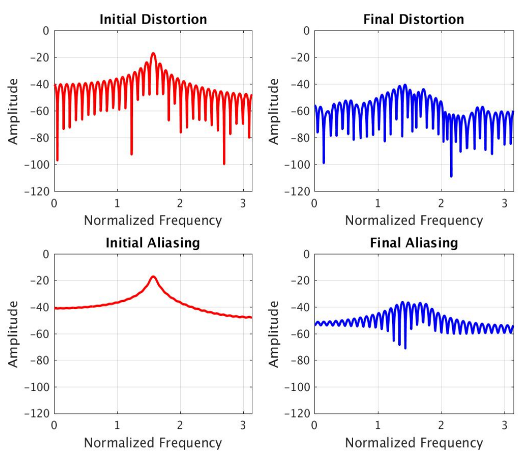

Figure 4-4-b

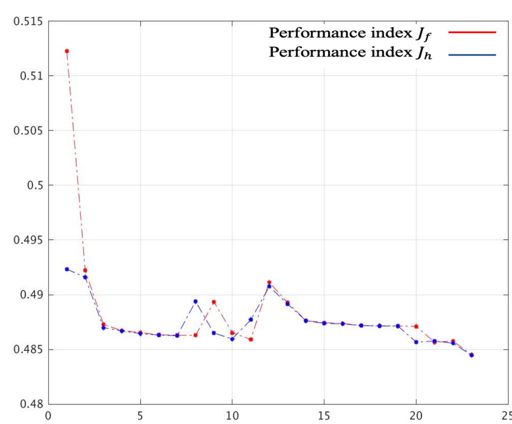

Figure 4-4-c

Figure 4-4 UFB [2 2] using algorithm 4. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 5**

Table 4-6: UFB [2 2] using algorithm 5.

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Frequency Resolution ρ_{sp}	60
Computation Time	45.98 sec
Max. Initial Distortion error	-16.98 dB
Max. Initial Aliasing error	-16.85 dB
Max. Final Distortion error	- 51.93 dB
Max. Final Aliasing error	- 47.42 dB

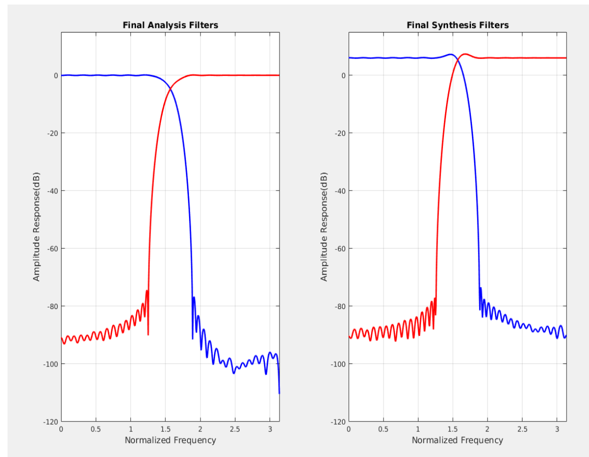

Figure 4-5-a

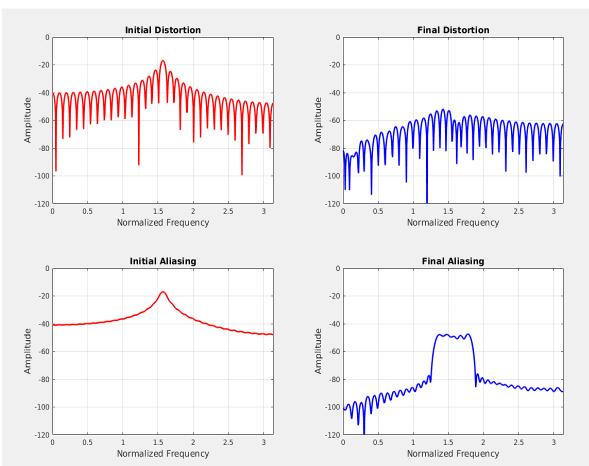

Figure 4-5-b

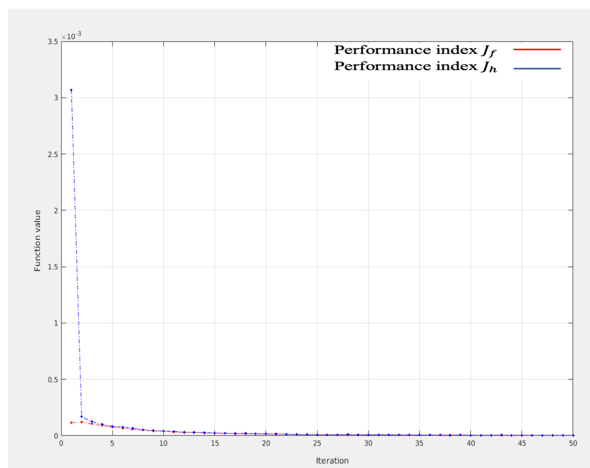

Figure 4-5-c

Figure 4-5 UFB [2 2] using algorithm 5. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

In the next examples, 4-2 and 4-3, the results obtained from the 5 algorithms in example 4-1 are compared with results using the algorithms presented in [56] and [58].

Example 4-2 UFB [2 2] using CMFB [56].

Table 4-7: UFB [2 2] using algorithm in [56].

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Computation Time	1:52 min
A_p	0.056 dB
A_s	-18.94 dB
Max. Final Distortion error	1.02 * e-12 dB
Max. Final Aliasing error	-280.23 dB

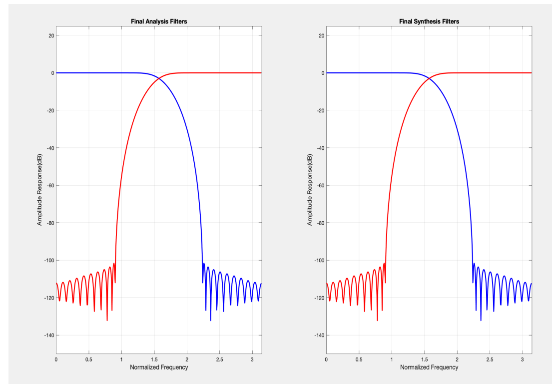

Figure 4-6-a

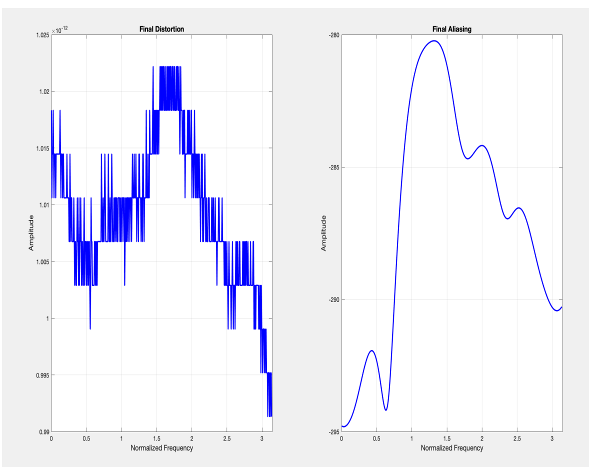

Figure 4-6-b

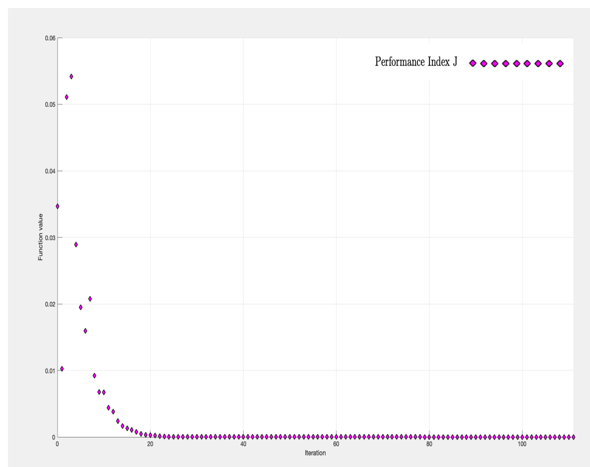

Figure 4-6-c

Figure 4-6 UFB [2 2] using algorithm in [56]. **a** – Final analysis and synthesis filters. **b** – Final distortion and aliasing errors **c** – Performance index value at each iteration.

Example 4-3 UFB [2 2] using CMFB [58].

Table 4-8: UFB [2 2] using algorithm in [58]

Number of Filter Coefficients	64
Transition Factors	[0.1 0.1]
Ratio	[0.5 0.5]
Computation Time	0.16 sec
A_p	6.81 * e-5 dB
A_s	- 45.03 dB
Max. Final Distortion error	- 274.83 dB
Max. Final Aliasing error	- 286.09 dB

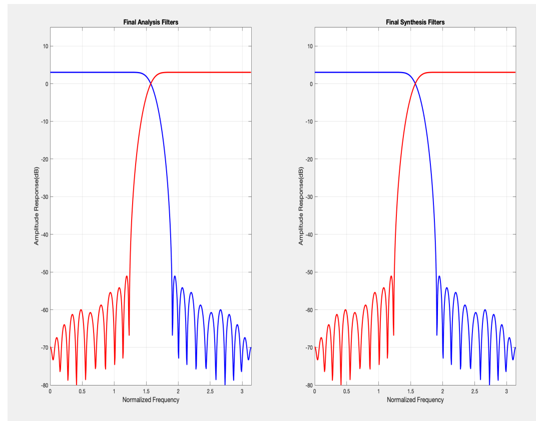

Figure 4-7-a

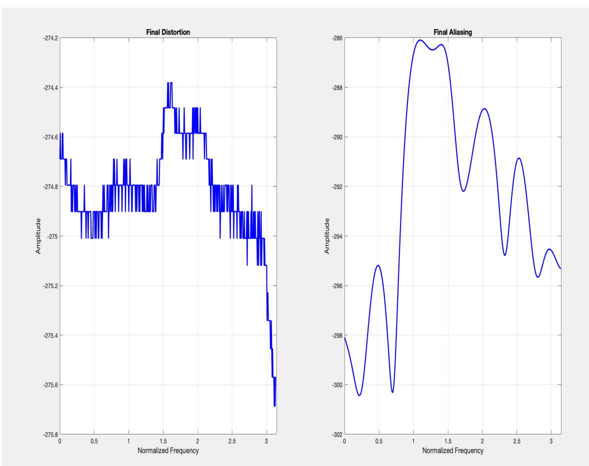

Figure 4-7-b

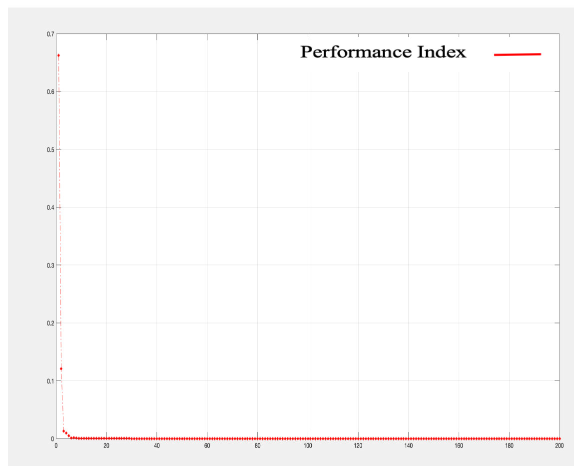

Figure 4-7-c

Figure 4-7 UFB [2 2] using algorithm in [58]. **a** – Final analysis and synthesis filters. **b** – Final distortion and aliasing errors **c** – Performance index value at each iteration.

In the next Example 4-4, a uniform filter bank with 4 sub-bands is designed.

Example 4-4 UFB [4 4 4 4] using algorithms 1 to 5

- **Algorithm 1**

Table 4-9: UFB [4 4 4 4] using algorithm 1.

Number of Filter Coefficients	56
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Frequency Resolution ρ	64
Computation Time	2:36 min
Max. Initial Distortion error	-5.79 dB
Max. Initial Aliasing error	-6.21 dB
Max. Final Distortion error	- 47.24 dB
Max. Final Aliasing error	- 42.70 dB

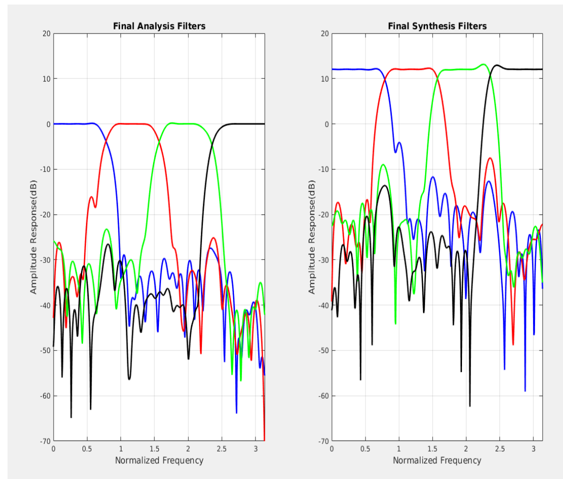

Figure 4-8-a

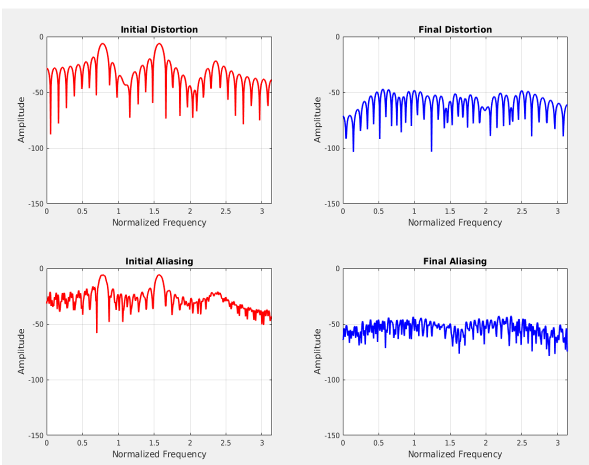

Figure 4-8-b

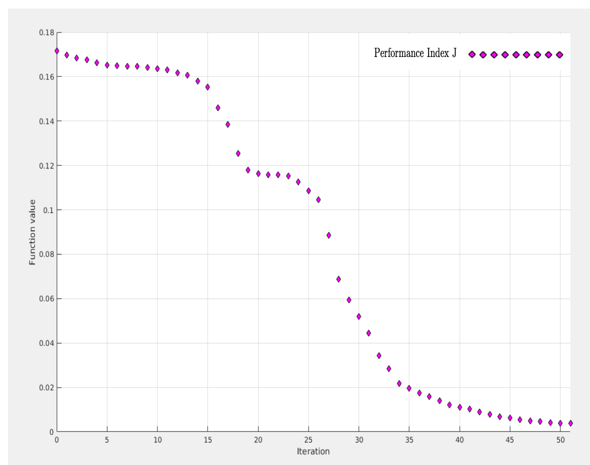

Figure 4-8-c

Figure 4-8 UFB [4 4 4 4] using algorithm 1. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors **c** – Performance index value at each iteration.

- **Algorithm 2**

Table 4-10:UFB [4 4 4 4] using algorithm 2.

Number of Filter Coefficients	56
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Frequency Resolution ρ_{sb}	64
Computation Time	3.97 sec
Max. Initial Distortion error	- 5.79 dB
Max. Initial Aliasing error	- 6.21 dB
Max. Final Distortion error	- 42.77 dB
Max. Final Aliasing error	- 38.50 dB

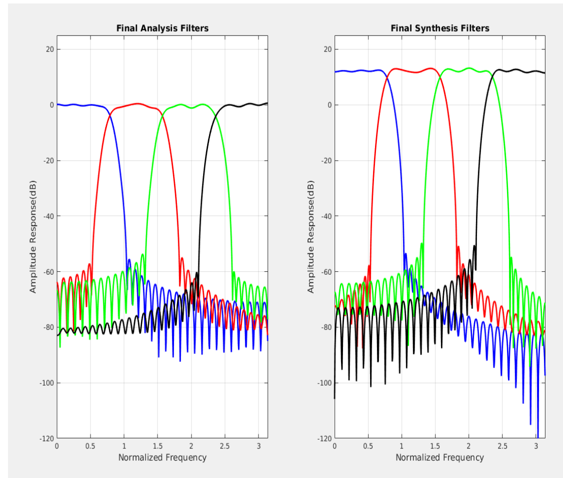

Figure 4-9-a

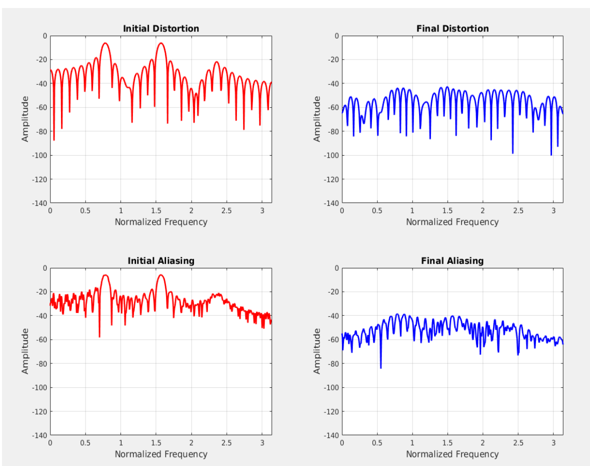

Figure 4-9-b

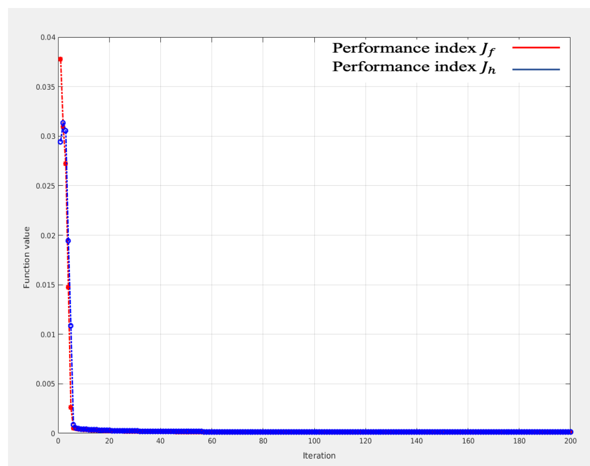

Figure 4-9-c

Figure 4-9 UFB [4 4 4 4] using algorithm 2. **a**– Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

- **Algorithm 3.**

Table 4-11: UFB [4 4 4 4] using algorithm 3.

Number of Filter Coefficients	56
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Frequency Resolution ρ	60
Computation Time	9:18 min
Max. Initial Distortion error	-5.79 dB
Max. Initial Aliasing error	-6.21 dB
Max. Final Distortion error	- 27.80 dB
Max. Final Aliasing error	- 23.71 dB

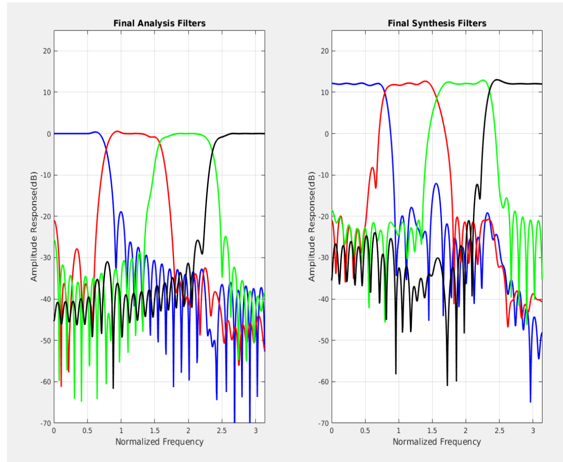

Figure 4-10-a

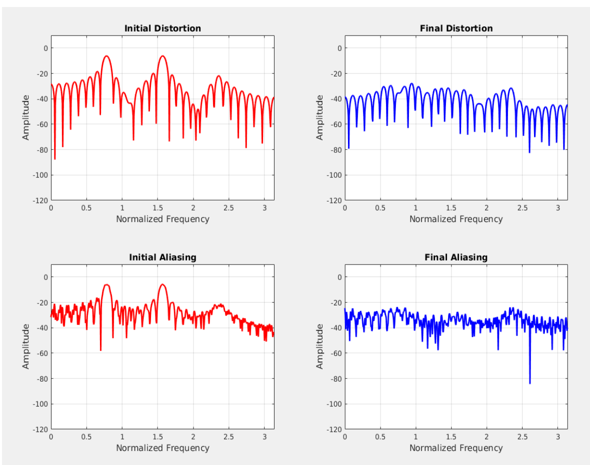

Figure 4-10-b

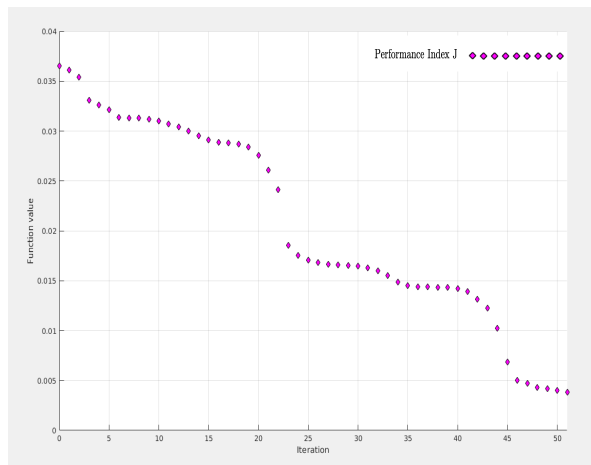

Figure 4-10-c

Figure 4-10 UFB [4 4 4 4] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 4.**

Table 4-12:UFB [4 4 4 4] using algorithm 4.

Number of Filter Coefficients	56
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Frequency Resolution ρ_{sp}	30
Frequency Resolution ρ_{pb}	60
Computation Time	5:11 hr
Max. Initial Distortion error	-5.79 dB
Max. Initial Aliasing error	-6.21 dB
Max. Final Distortion error	- 36.75 dB
Max. Final Aliasing error	- 35.22 dB

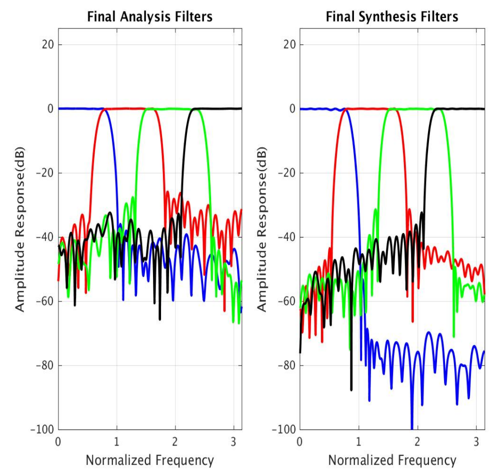

Figure 4-11-a

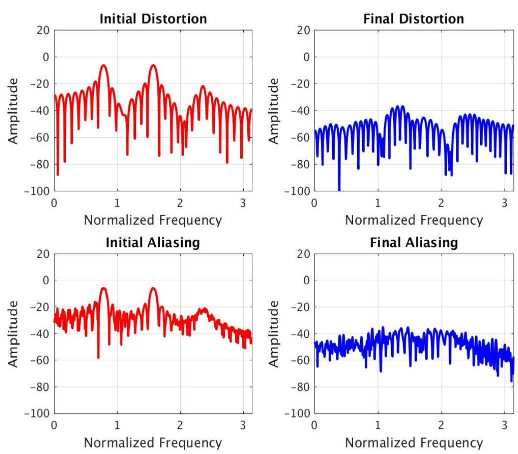

Figure 4-11-b

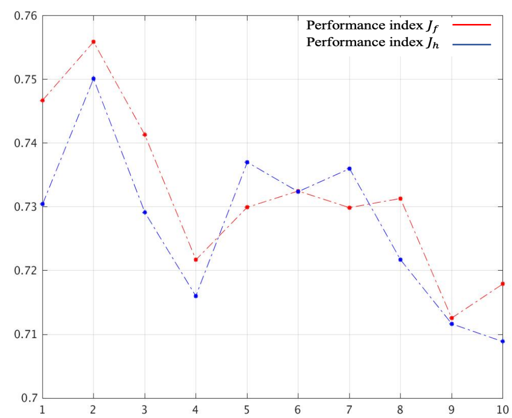

Figure 4-11-c

Figure 4-11 UFB [4 4 4 4] using algorithm 4. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 5.**

Table 4-13:UFB [4 4 4 4] using algorithm 5.

Number of Filter Coefficients	56
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Frequency Resolution ρ_{sp}	30
Computation Time	2:16 min
Max. Initial Distortion error	-5.76 dB
Max. Initial Aliasing error	-6.21 dB
Max. Final Distortion error	- 32.23 dB
Max. Final Aliasing error	- 27.35 dB

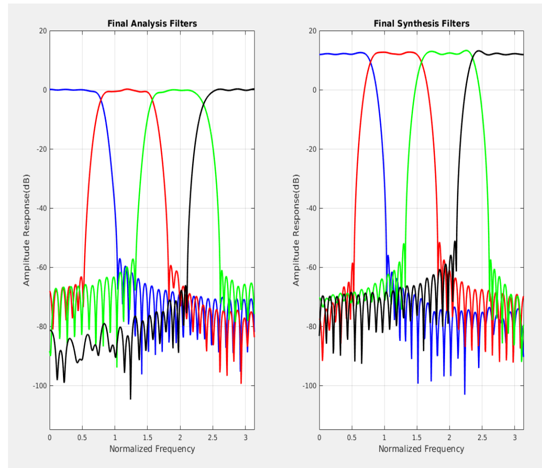

Figure 4-12-a

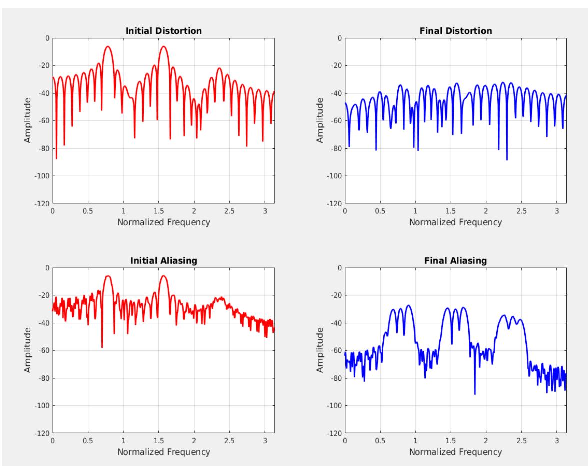

Figure 4-12-b

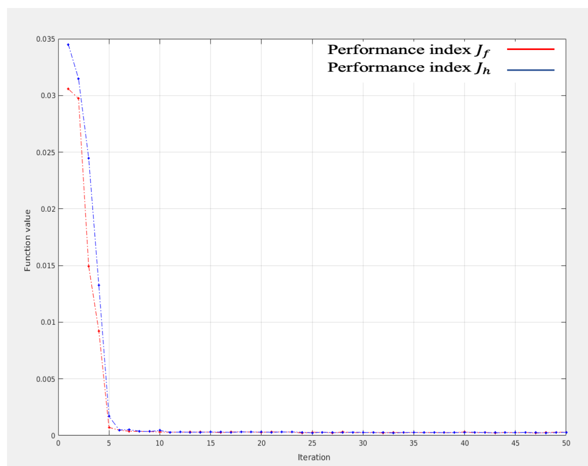

Figure 4-12-c

Figure 4-12 UFB [4 4 4 4] using algorithm 5. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

In the next two examples, 4-5 and 4-6, the results obtained from the 5 algorithms in example 4-4 are compared with results using the algorithms presented in [56] and [58].

Example 4-5 UFB [4 4 4 4] using CMFB [56].

Table 4-14: UFB [4 4 4 4] using algorithm in [56].

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Computation Time	1:37 min
A_p	0.006 dB
A_s	- 28.23 dB
Max. Final Distortion error	- 176.14 dB
Max. Final Aliasing error	- 68.56 dB

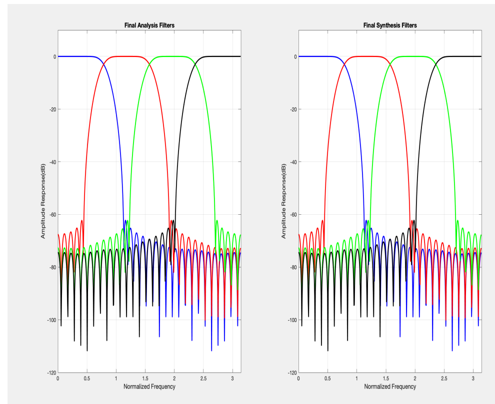

Figure 4-13-a

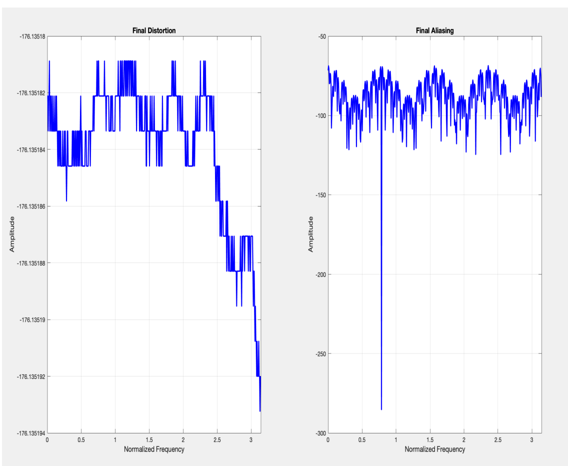

Figure 4-13-b

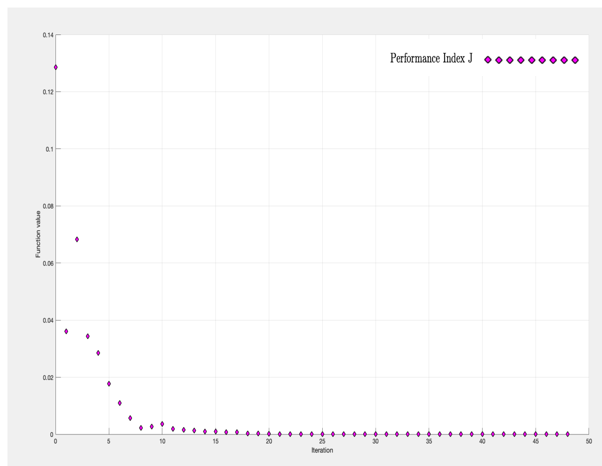

Figure 4-13-c

Figure 4-13 UFB [2 2] using algorithm in [56]. **a** – Final analysis and synthesis filters. **b** – Final and aliasing errors **c** – Performance index value at each iteration.

Example 4-6 UFB [4 4 4 4] using CMFB [58].

Table 4-15: UFB [4 4 4 4] using algorithm in [58]

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.08 0.08]
Ratio	[0.25 0.25 0.25 0.25]
Computation Time	0.16 sec
A_p	8.23 ∗ e-4 dB
A_s	- 31.30 dB
Max. Final Distortion error	- 278.82 dB
Max. Final Aliasing error	- 282.78 dB

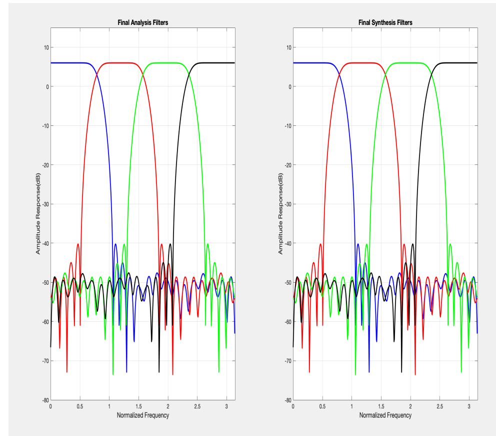

Figure 4-14-a

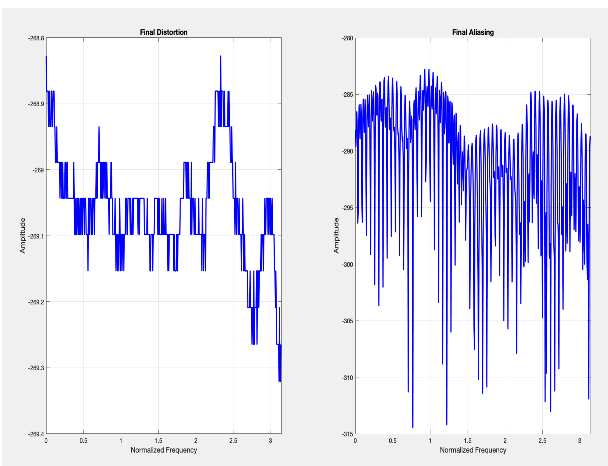

Figure 4-14-b

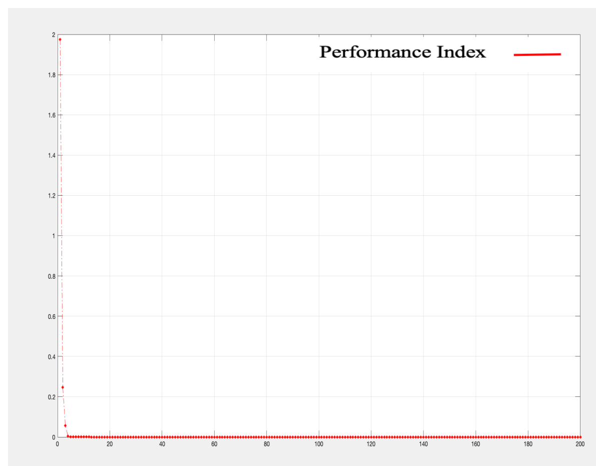

Figure 4-14-c

Figure 4-14 UFB [2 2] using algorithm in [58]. **a –** Final analysis and synthesis filters. **b –** Final distortion and aliasing errors **c –** Performance index value at each iteration.

Table 4-16: Uniform filter bank designs comparison.

Design Specifications	Algorithms	Analysis filters		Synthesis filters		Max. Distortion error (dB)	Max. Aliasing error (dB)	Time
		A_s (dB)	A_p (dB)	A_s (dB)	A_p (dB)			
Sub-bands (K) S= [2 2]	Algorithm 1	30.48	0.046	24.40	0.037	- 89.57	- 84.73	40.82 sec
Filter Coefficients (N) 64	Algorithm 2	73.08	0.037	71.26	0.36	- 63.48	- 64.53	1.62 sec
Transition Factors	Algorithm 3	49.76	0.013	51.29	0.043	- 52.78	- 47.73	2:33 min
[0.1 0.1]	Algorithm 4	65.72	0.019	70.86	0.049	- 40.38	- 36.15	2:57 hr
Ratio [0.5 0.5]	Algorithm 5	74.27	0.21	73.42	0.22	-51.93	-47.42	45.98 sec
Sub-bands (K) S = [4 4 4 4]	Algorithm 1	17.23	0.11	5.23	0.12	- 47.24	-42.70	2:36 min
Filter Coefficients (N)	Algorithm 2	53.62	1.30	49.71	1.29	-42.77	-38.50	3.97 sec
56	Algorithm 3	-20.93	0.16	12.23	0.5	- 27.80	- 23.71	9:18 min
Transition Factors [0.08 0.08 0.08 0.08]]	Algorithm 4	25.90	0.10	31.37	0.40	- 36.75	- 35.22	5:11 hr
Ratio [0.25 0.25 0.25 0.25]	Algorithm 5	57.02	0.72	51.12	0.65	- 32.23	- 27.35	2:16 min

A_s is the minimum stopband attenuation measured in dB.
A_p is the passband ripple measured in dB.

From the observation of the previous UFB examples, it is apparent that the five algorithms can be used for such types of filter banks. In the designs presented, the computation time in these five algorithms is different, and it increases as the number of sub-band filters increases. Algorithm 1 provides the smallest distortion and aliasing errors in the two UFB designs. For example, in the design with the S= [2 2], the distortion error is - 89.57 dB and aliasing error is - 84.73, and for the design with the S = [4 4 4 4], the distortion error is - 47.24 dB and the aliasing error is - 42.70. Algorithms 2, 3, 4 and 5 report distortion and aliasing errors larger than - 65 dB for both examples. Another observation from Figure 4-5-b and Figure 4-12-b is that algorithm 5 shows there are larger aliasing and distortion errors in the transition bands than the rest of the normalized frequency spectrum. Accepting this error in these areas for such designs depends on the application.

From Table 4-16, it is clear that the stopband attenuation in algorithm 1 has the most significant values, while algorithm 5 achieves the smallest values. The stopband attenuation A_s in Algorithm 5 for the uniform set [2 2] is equal to 76 dB, and for the uniform set [4 4 4 4] is equal to 54 dB. By comparison, algorithm 2 also achieves high stopband attenuation, and this is expected. Further, the results also show that the modified version of algorithm 1 (algorithm 3) introduces an improvement in the stopband attenuation in the analysis and synthesis filters.

In addition, it can be noticed that algorithms 1, 2 and 4 offer smaller passband ripples A_p in both examples, compared with algorithms 2 and 5. E.g. in algorithm 4 the largest passband ripple is equal to 0.019 dB in the analysis filters, and 0.049 in the synthesis filters for the UFB set [2 2]. On the other hand, the results from algorithm 2 show that the largest passband ripples for the UFB set [4 4 4 4] are 1.30 dB and 1.29 in the analysis and synthesis filters respectively.

Table 4-16 also shows that algorithm 2 requires less computation time compared with its counterparts, and this can be anticipated. As an example, the computation time for algorithm 1 is 1.62 sec and 45.98 sec for the sampling rate sets [2 2] and [4 4 4 4], respectively. In contrast, the computation times for the same designs using algorithm 5 are 3.97 sec and 2:16 min for the same sets. On the other hand, algorithm 4 requires longer optimization time than the other algorithms, which can be expected. This algorithm deals with a large number of constraints for the stopband and the passband areas in each sub-band filter.

Furthermore, the performance index-Iteration figures show that algorithms 2 and 4 require a smaller number of iterations than algorithms 1 and 3, which can be expected because the former deal with quadratic objective functions, whereas the latter minimize nonlinear objective functions.

Comparing the results with the ones obtained using the two methods presented in [56] and [58], it is clear that these two methods are faster and provide smaller passband ripples when using the same design specifications. From the results shown in Table 4-8, and Table 4-15, it can be noticed that the method in [58] offers significant small aliasing and distortion errors up to -270 dB. In contrast, among the introduced five algorithms for the same UFB sets, the smallest distortion and aliasing errors are - 89.57 and - 84.73 in algorithm 1. However, it can be noted that algorithms 2, 4 and 5 offer smaller stopband attenuation A_s than the method in [58]. In the two-channel UFB design, A_s is 45.03 dB by using the technique in [58], and for the same set, the stopband attenuations are 73.08 dB, 65.72 dB, and 74.27 dB using algorithms 2, 4 and 5, respectively. Table 4-7 and Table 4-14 also present the results from the algorithm offered in [56].

4.2.2 Compatible non-uniform filter bank design examples

After illustrating UFB examples using the five algorithms and comparing these examples with CMFB methods, in this section, two compatible NUFB designs are presented. These two examples are demonstrated using the same manner of the previous examples.

For the first design, a three-channel filter bank with a S = [2 4 4], and for second a four-channel filter bank with a S = [2 4 8 8], are considered. These two designs are maximally decimated filter banks, and the number of filter coefficients N is the same for each FIR analysis and synthesis filters.

As a start, from the previous work for designing compatible NUFBs, it is shown that this topic was a subject of interest in DSP for a few years, and many results for different NUFB sets have been presented. For example, work in [18, 26, 27, 60-64] include results for two designs used in this section, particularly the designs for the sampling rate sets [2 4 4] and [2 4 8 8]. The results obtained with the five algorithms are compared with some of the results from these existing methods at the end of this section.

Example 4-7 Compatible NUFB [2 4 4] using algorithms 1 to 5

- **Algorithm 1.**

Table 4-17: Compatible NUFB [2 4 4] using algorithm 1.

Number of Filter Coefficients	48
Transition Factors	[0.1 0.1 0.1]
Ratio	[0.5 0.25 0.25]
Frequency Resolution ρ	60
Computation Time	1:03 min
Max. Initial Distortion error	-6.23 dB
Max. Initial Aliasing error	-5.92 dB
Max. Final Distortion error	- 56.02 dB
Max. Final Aliasing error	- 53.91 dB

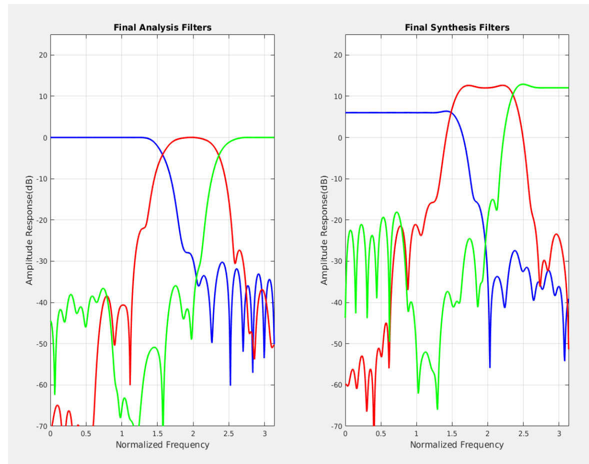

Figure 4-15-a

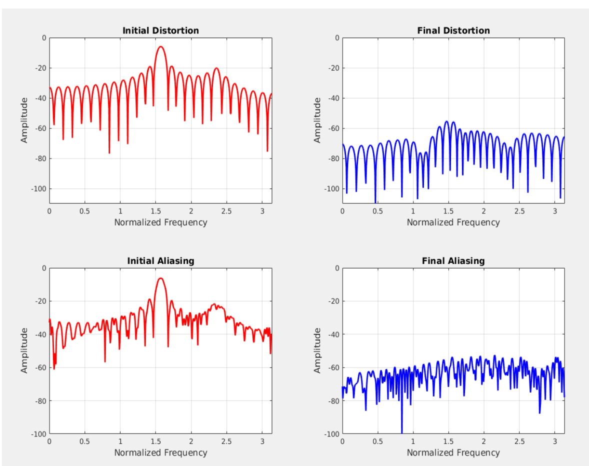

Figure 4-15-b

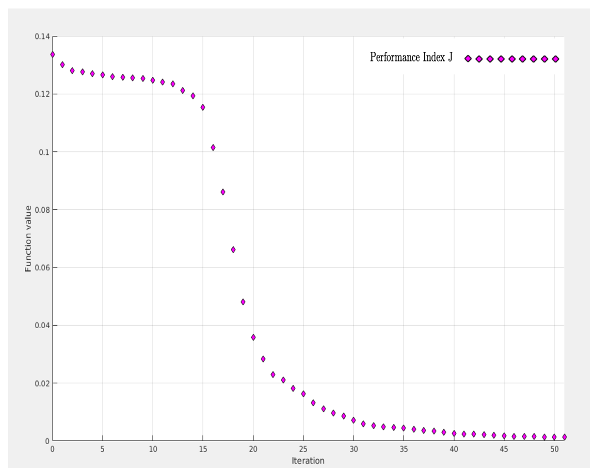

Figure 4-15-c

Figure 4-15 Compatible NUFB [2 4 4] using algorithm 1. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

Algorithm 2.

Table 4-18: Compatible NUFB [2 4 4] using algorithm 2.

Number of Filter Coefficients	48
Transition Factors	[0.1 0.1 0.1]
Ratio	[0.5 0.25 0.25]
Frequency Resolution ρ_{sp}	60
Computation Time	2.04 sec
Max. Initial Distortion error	-6.23 dB
Max. Initial Aliasing error	-5.92 dB
Max. Final Distortion error	- 47.94 dB
Max. Final Aliasing error	- 44.00 dB

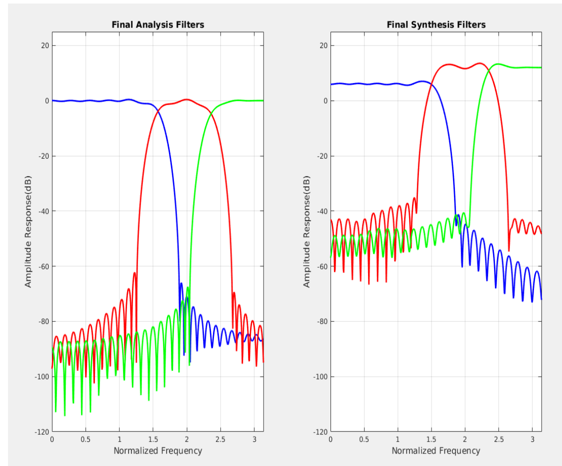

Figure 4-16-a

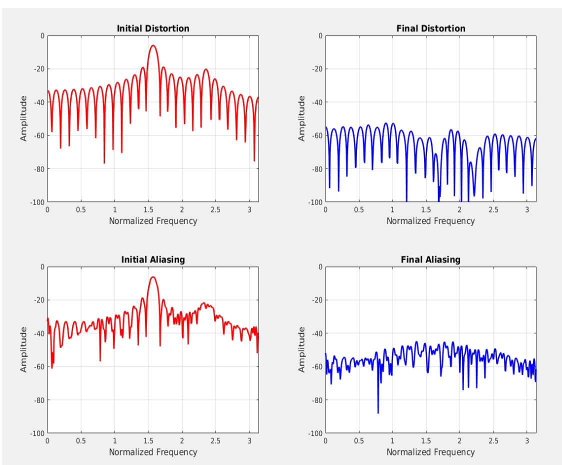

Figure 4-16-b

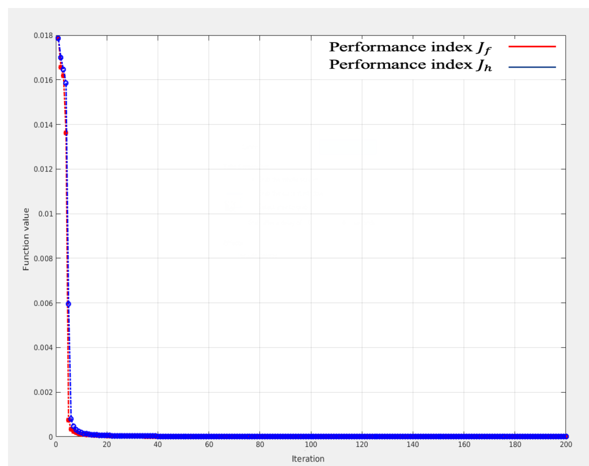

Figure 4-16-c

Figure 4-16 Compatible NUFB [2 4 4] using algorithm 2. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

- **Algorithm 3.**

Table 4-19: Compatible NUFB [2 4 4] using algorithm 3.

Number of Filter Coefficients	48
Transition Factors	[0.1 0.1 0.1]
Ratio	[0.5 0.25 0.25]
Frequency Resolution ρ	60
Computation Time	3:02 min
Max. Initial Distortion error	-6.23 dB
Max. Initial Aliasing error	-5.92 dB
Max. Final Distortion error	- 37.92 dB
Max. Final Aliasing error	- 38.21 dB

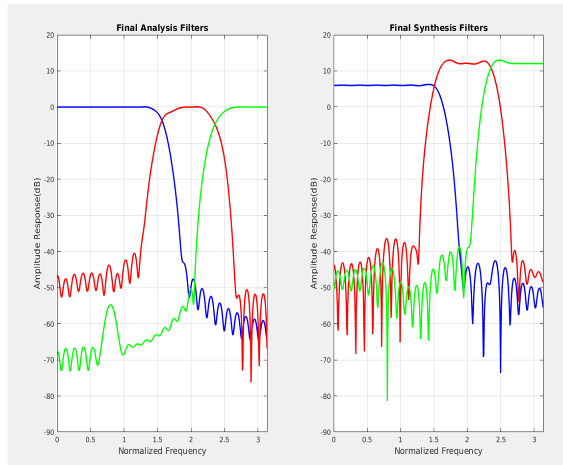

Figure 4-17-a

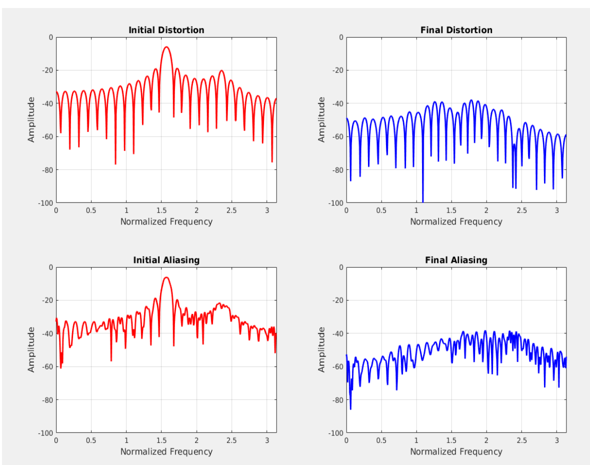

Figure 4-17-b

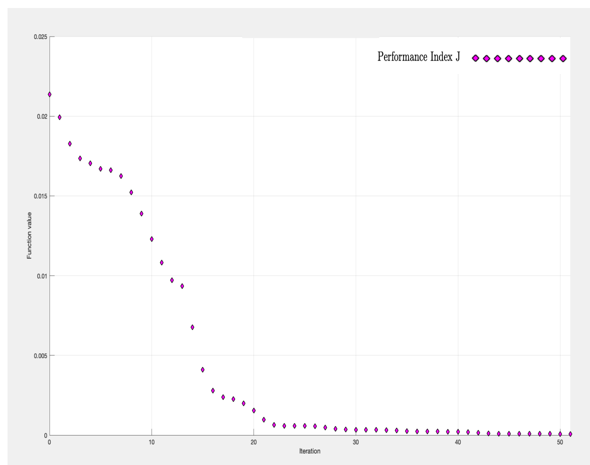

Figure 4-17-c

Figure 4-17: Compatible NUFB [2 4 4] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 4.**

Table 4-20: Compatible NUFB [2 4 4] using algorithm 4.

Number of Filter Coefficients	48
Transition Factors	[0.1 0.1 0.1]
Ratio	[0.5 0.25 0.25]
Frequency Resolution ρ_{sp}	30
Frequency Resolution ρ_{pb}	60
Computation Time	1:15 hr
Max. Initial Distortion error	- 6.23 dB
Max. Initial Aliasing error	-5.92dB
Max. Final Distortion error	- 28.06 dB
Max. Final Aliasing error	- 28.66 dB

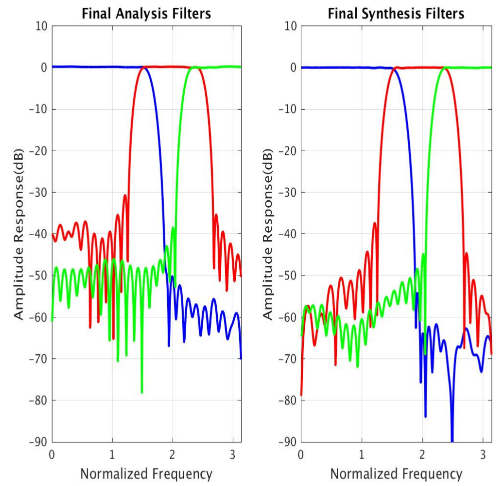

Figure 4-18-a

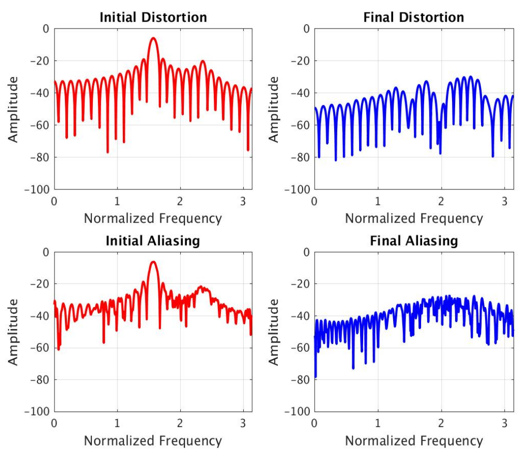

Figure 4-18-b

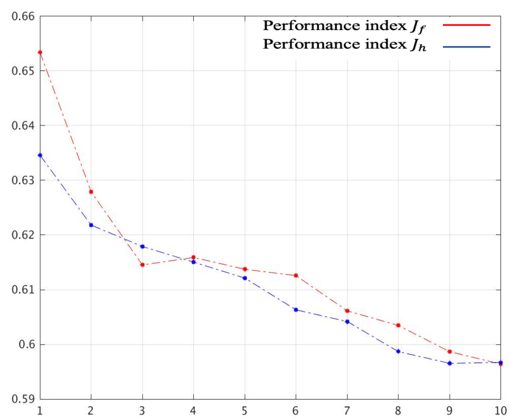

Figure 4-18-c

Figure 4-18: Compatible NUFB [2 4 4] using algorithm 4. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 5.**

Table 4-21: Compatible NUFB [2 4 4] using algorithm 5.

Number of Filter Coefficients	48
Transition Factors	[0.1 0.1 0.1]
Ratio	[0.5 0.25 0.25]
Frequency Resolution ρ_{sp}	60
Computation Time	1:09 min
Max. Initial Distortion error	-6.23 dB
Max. Initial Aliasing error	-5.92 dB
Max. Final Distortion error	- 38.66 dB
Max. Final Aliasing error	- 33:21 dB

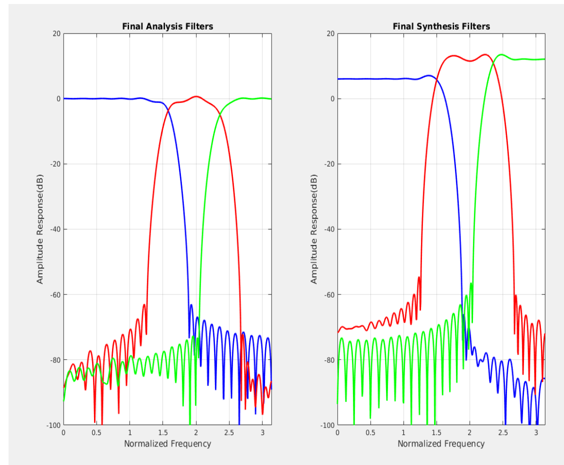

Figure 4-19-a

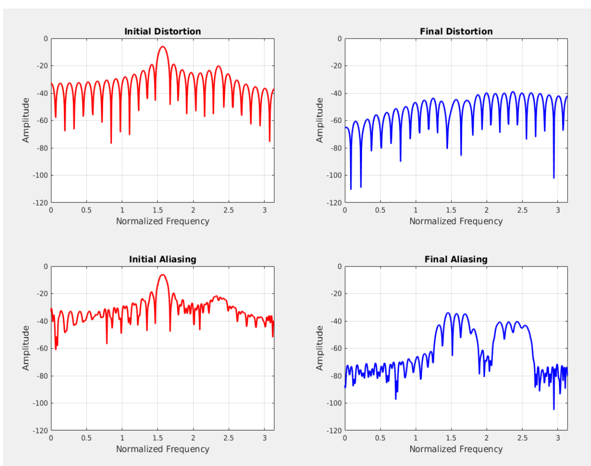

Figure 4-19-b

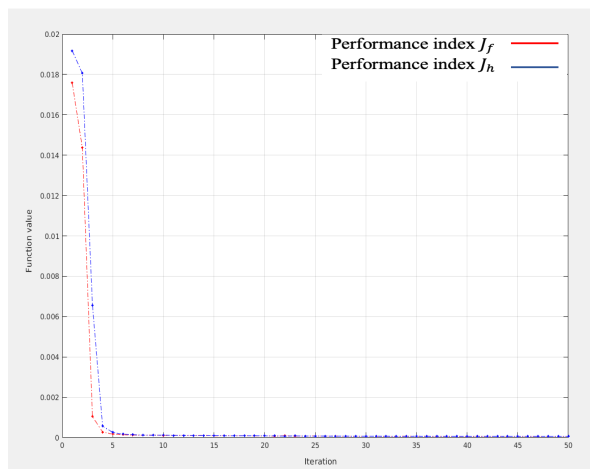

Figure 4-19-c

Figure 4-19: compatible NUFB [2 4 4] using algorithm 5. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

In the next Example 4-4, a compatible non-uniform filter bank with 4 sub-bands is designed.

Example 4-8 Compatible NUFB [2 4 8 8] using algorithms 1 to 5

- **Algorithm 1**

Table 4-22: Compatible NUFB [2 4 8 8] using algorithm 1.

Number of Filter Coefficients	76
Transition Factors	[0.09 0.08 0.062 0.062]
Ratio	[[0.5 0.25 0.125 0.125]
Frequency Resolution ρ	100
Computation Time	22:01 min
Max. Initial Distortion error	-5.91dB
Max. Initial Aliasing error	-6.02 dB
Max. Final Distortion error	- 66.87 dB
Max. Final Aliasing error	- 59.1 dB

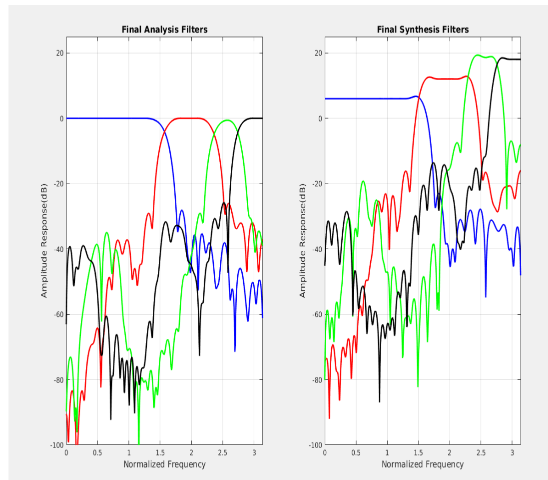

Figure 4-20-a

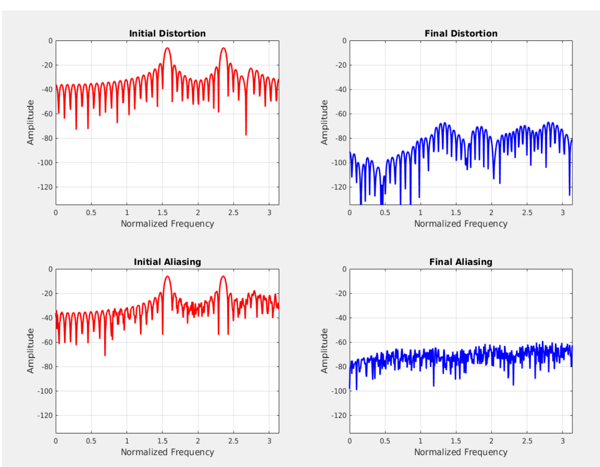

Figure 4-20-b

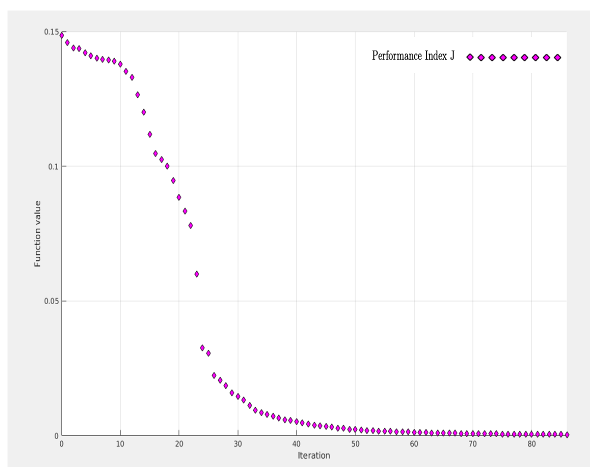

Figure 4-20-c

Figure 4-20: Compatible NUFB [2 4 8 8] using algorithm 1. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

Algorithm 2.

Table 4-23: Compatible NUFB [2 4 8 8] using algorithm 2.

Number of Filter Coefficients	76
Transition Factors	[0.09 0.08 0.062 0.062]
Ratio	[0.5 0.25 0.125 0.125]
Frequency Resolution ρ_{sp}	100
Computation Time	10.05 sec
Max. Initial Distortion error	-5.91 dB
Max. Initial Aliasing error	-6.02 dB
Max. Final Distortion error	- 42.77 dB
Max. Final Aliasing error	-38.50 dB

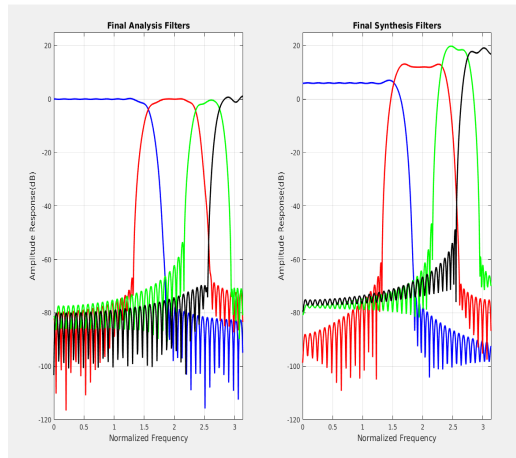

Figure 4-21-a

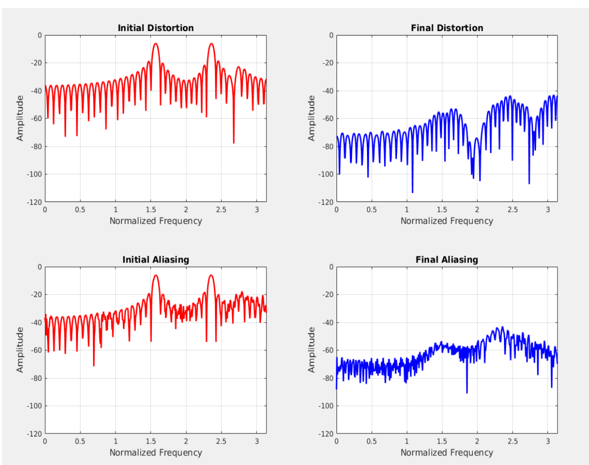

Figure 4-21-b

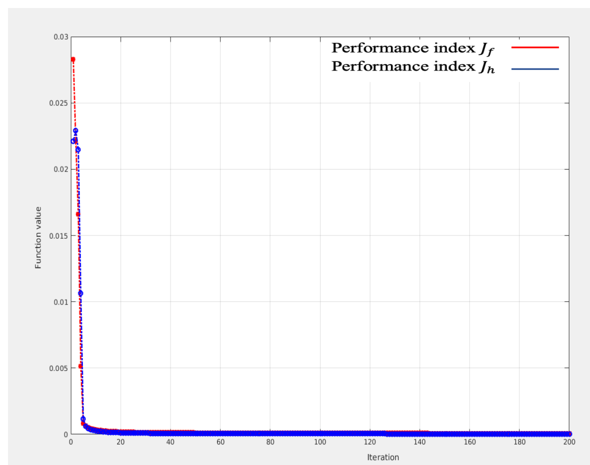

Figure 4-21-c

Figure 4-21 Compatible NUFB [2 4 8 8] using algorithm 2. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 3.**

Table 4-24: Compatible NUFB [2 4 8 8] using algorithm 3.

Number of Filter Coefficients	76
Transition Factors	[0.09 0.08 0.062 0.062]
Ratio	[0.5 0.25 0.125 0.125]
Frequency Resolution ρ	100
Computation Time	42:13 min
Max. Initial Distortion error	- 5.91 dB
Max. Initial Aliasing error	- 6.02 dB
Max. Final Distortion error	- 43.85 dB
Max. Final Aliasing error	- 38:13 dB

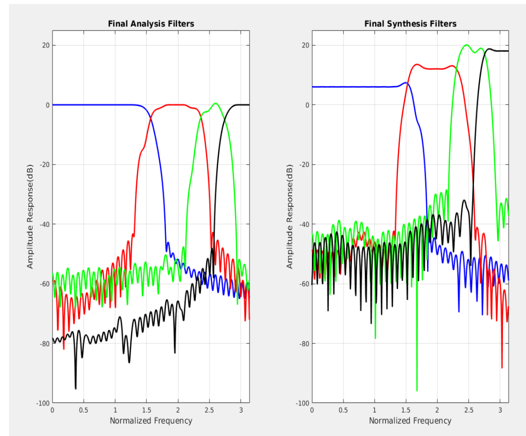

Figure 4-22-a

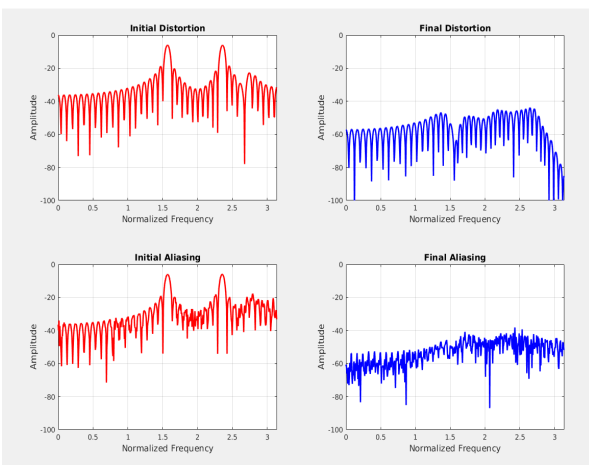

Figure 4-22-b

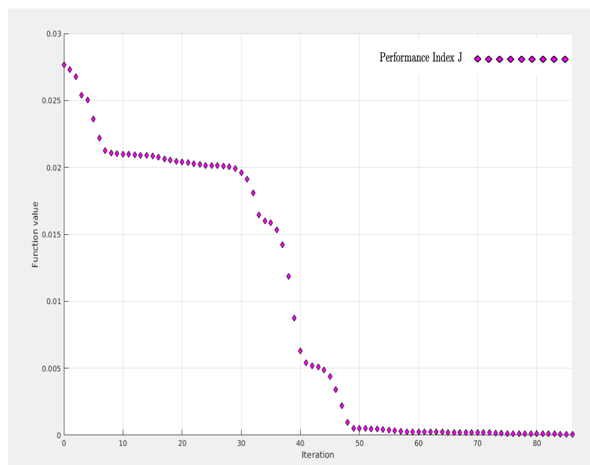

Figure 4-22-c

Figure 4-22 Compatible NUFB [2 4 8 8] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 4.**

The number of filter coefficients in this part is set to 48 instead of 64 because it gives better results. However, the results obtained are still used to evaluate and compare this algorithm with its counterparts.

Table 4-25: Compatible NUFB [2 4 8 8] using algorithm 4.

Number of Filter Coefficients	48
Transition Factors	[0.09 0.08 0.062 0.062]
Ratio	[0.5 0.25 0.125 0.125]
Frequency Resolution ρ_{sp}	50
Frequency Resolution ρ_{pb}	100
Computation Time	2:39 hr
Max. Initial Distortion error	- 5.49 dB
Max. Initial Aliasing error	- 5.80 dB
Max. Final Distortion error	- 10.61 dB
Max. Final Aliasing error	- 4.63 dB

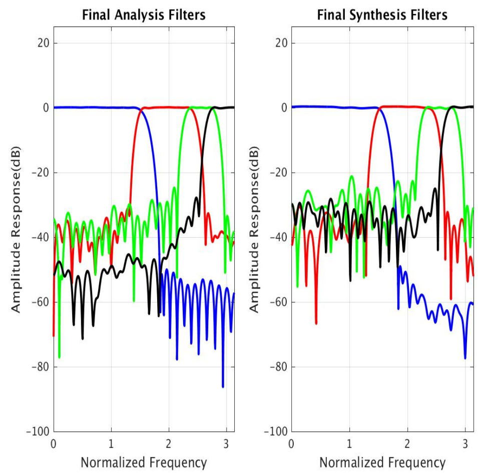

Figure 4-23-a

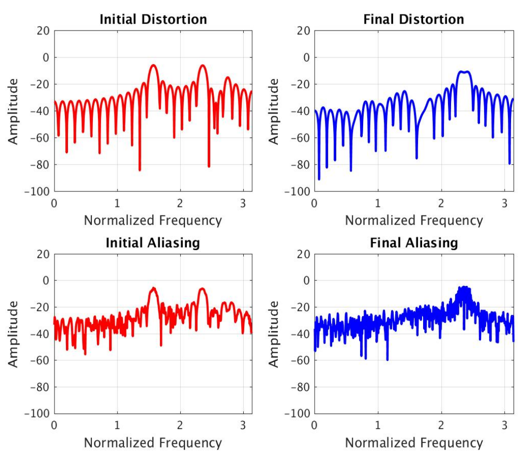

Figure 4-23-b

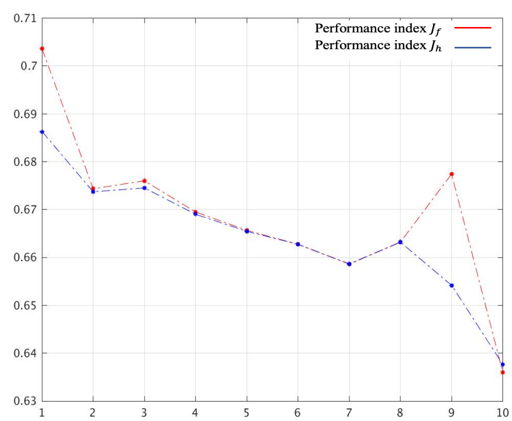

Figure 4-23-c

Figure 4-23 Compatible NUFB [2 4 8 8] using algorithm 4. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

- **Algorithm 5.**

Table 4-26: Compatible NUFB [2 4 8 8] using algorithm 5.

Number of Filter Coefficients	76
Transition Factors	[0.09 0.08 0.062 0.062]
Ratio	[0.5 0.25 0.125 0.125]
Frequency Resolution ρ_{sp}	100
Computation Time	8:10 min
Max. Initial Distortion error	- 5.91 dB
Max. Initial Aliasing error	- 6.02 dB
Max. Final Distortion error	- 30.29 dB
Max. Final Aliasing error	- 30.45 dB

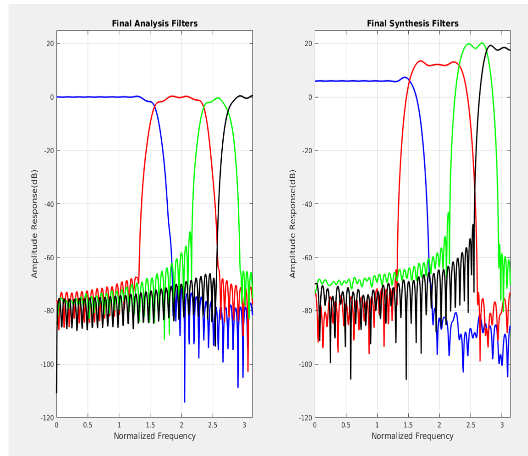

Figure 4-24-a

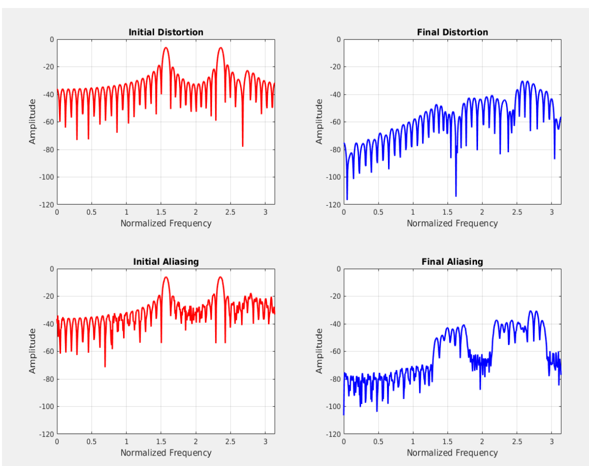

Figure 4-24-b

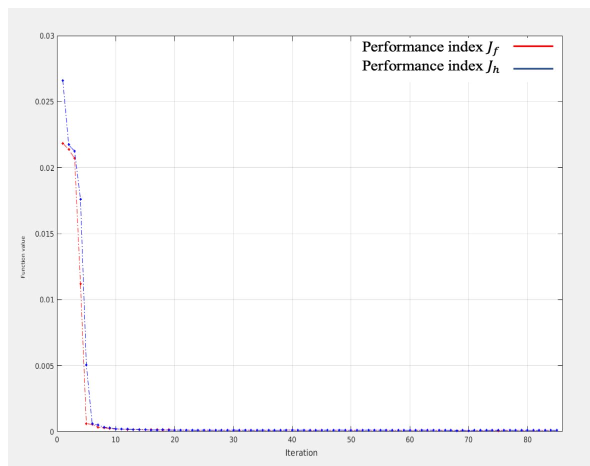

Figure 4-24-c

Figure 4-24 Compatible NUFB [2 4 8 8] using algorithm 5. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

Table 4-27: Compatible NUFB designs comparison

Design Specifications	Algorithms	Analysis filters		Synthesis filters		Max. Distortion error (dB)	Max. Aliasing error (dB)	Time
		A_s (dB)	A_p (dB)	A_s (dB)	A_p (dB)			
Sub-bands (K) S = [2 4 4]	Algorithm 1	24.73	0.075	15.62	0.089	- 56.02	- 53.91	1:03 min
Filter Coefficients (N) 48	Algorithm 2	59.80	0.59	54.73	0.52	- 47.94	- 44.00	2.04 sec
Transition Factors	Algorithm 3	39.51	0.056	36.38	0.17	- 37.92	- 38.21	3:02 min
[0.1 0.1 0.1]	Algorithm 4	45.53	0.093	50.66	0.05	- 28.06	- 28. 66	2:57 hr
Ratio [0.5 0.25 0.25]	Algorithm 5	62.55	0.89	55.16	0.72	- 38.66	- 33:21	1:09 min
Sub-bands (K) S= [2 4 8 8]	Algorithm 1	25.81	0.028	6.93	0.026	- 66.87	- 59:13	22:01 min
Filter Coefficients (N)	Algorithm 2	53.53	2.27	46.52	2.30	- 42.77	- 38.50	10.05 sec
76,	Algorithm 3	31:93	0.071	28.43	0.12	- 43.85	- 38:13	42:13 min
Transition Factors [0.09 0.08 0.062 0.062]	Algorithm 4	22.54	0.19	17.06	0.57	- 10.61	- 4.63	5:11 hr
Ratio [0.5 0.25 0.125 0.125]	Algorithm 5	50.38	1.02	43.09	1.01	- 30.29	- 30.45	8:10 min

A_s is the minimum stopband attenuation measured in dB.
A_p is the passband ripple measured in dB.

From the presented examples of the two compatible NUFB sets, it can be observed that among the five algorithms, algorithm 1 offers the smallest distortion and aliasing errors. For instance, for the S = [2 4 4], the maximum distortion error is - 55.40 dB, and the maximum aliasing error is - 52.60 dB, and for the S = [2 4 8 8], the maximum distortion error is - 66.87 dB and the maximum aliasing error is - 59.24 dB. On the other hand, algorithm 4 leads to the largest distortion and aliasing errors among the five algorithms. Also, it can be seen from Table 4-27 that algorithms, 2, 3 and 5 achieve distortion and aliasing errors between - 30 dB and - 47 dB for the same designs.

For the magnitude response, it is shown that algorithms 2 and 5 still offer the smallest stopband attenuation in the presented examples. For instance, in the analysis filters for NUFB [2 4 4], the stopband attenuation A_s is 59.80 dB using algorithm 2, and it is 62.55 using algorithm 5. Similarly, in the NUFB [2 4 8 8], A_s is 53.53 dB using algorithm 2, and it is 50.38 dB when using algorithm 5. The results show that algorithm 3 improves the stopband attenuation obtained in algorithm 1. In more detail, Table 4-27 shows that the synthesis filters stopband attenuation improved from 15.62 dB in algorithm 1 to 36.38 dB using algorithm 3 for the S = [2 4 4] design, and it is improved from 6.93 dB using algorithm 1 to 28.43 dB using algorithm 3 for the S = [2 4 8 8] design.

For the passband ripple A_p, the results show that algorithms 2 and 5 have the most considerable passband ripples with more than 0.5 dB in the analysis filters. On the other hand, algorithms 1, 3 and 4 offer smaller passband ripples, and this is expected, e.g. in [2 4 4] design, the passband ripples are 0.089 dB, 0.17 dB, and 0.05 dB in these algorithms respectively.

For the computation time, algorithm 2 still requires less computation time compared with the other four algorithms, and algorithm 4 is still the slowest one. Table 4-27 also shows that algorithm 3 takes longer time than algorithm 1, and this can be anticipated. Overall from the five algorithms, a trade-off is required to select between smaller distortion and aliasing errors, better passband and stopband magnitude responses, or less computation time for the design, and this choice depends on what is required for any particular application.

Comparing the results with the ones obtained in the existing methods, Djokovic and Vaidyanathan present a three-channel NUFB design with the S = [2 4 4] in [60]. The results show that the stopband attenuation in the analysis filters is less than 20 dB, whereas the results from the five algorithms show that the largest stopband attenuation for the analysis filters is 24.73 dB in algorithm 1. In another design by Li et al. in [18], a simple approach to design an NPR for the same set is implemented using uniform CMFB.

Their results show that the distortion and aliasing errors are comparable to the stopband attenuation of the prototype filter at about 60 dB. Table 4-27 show better results for the same sampling rate set design specifications. Similar designs for the same compatible NUFB set are implemented with different CMFB approaches in [26, 61]. Another NPR NUFB design that uses constrained equiripple FIR technique is presented in [27]. In this fast design, using 48 filter coefficients for a porotype filter to design a filter bank with S = [2 4 4], the results show that the stopband attenuation in this design is A_s is 0.00050 dB, and the reconstruction error is also minimized. On the other hand, using the same design specification, algorithm 1 leads to distortion and aliasing errors less than − 53.91 dB, and the stopband attenuations in the analysis and synthesis filter banks are 24.73 dB and 15.62 dB, respectively. Algorithms 2 and 5 offer stopband attenuations more than 40 dB in both analysis and synthesis filter banks.

For the compatible NUFB set [2 4 8 8], a design technique using modified window functions is presented by Anurag and Kumar in [62]. The method is an improvement of previous window design for NUFB presented in [63]. For this particular compatible sampling rate set, it is shown that the stopband attenuations are less than 70 dB in the suggested methods. It is also reported that the reconstruction error is minimized. Likewise, in another recent approach, a design of CMFB using Particle Swarm Optimization (PSO) is suggested in [64]. The proposed technique is based on an optimization scheme that minimizes the passband and stopband energy for a prototype filter and, by using this prototype low pass filter, a uniform CMFB is designed first. Then a combination of the adjacent channels of the UFB yields to the NUFBs. For designs with 144 and 160 filter coefficients, it was reported that this method offers a small stopband attenuation and small reconstruction errors. In [27] also, where a filter bank design uses 72 filter coefficients for a prototype filter to design a filter bank with S = [2 4 8 8], the results show that the stopband attenuation in this design is A_s is 0.00016 dB, and the reconstruction error is also minimized. For the same sampling rate set and with 76 filter coefficients, algorithm 1 leads to distortion and aliasing errors less than - 59 dB, and the stopband attenuations in the synthesis and analysis filter banks are 25.8 dB and 6.93 dB, respectively. Algorithms 2 and 5 offer stopband attenuations more than 40 dB in both analysis and synthesis filter banks.

The results obtained from the five algorithms are compared with the results from previous work. The comparison includes magnitude response and PR errors. However, due to the different ways of presenting these results in the other papers, sometimes it is not possible to compare all performance parameters, such as the magnitude responses of synthesis filters and the distortion and aliasing errors over the whole normalized frequency [0 to π].

4.2.3 Incompatible non-uniform filter bank design examples

In this section, two incompatible NUFB designs with sampling rate sets, [2 3 6] and [8 8 4 2 1], are presented. In these two filter bank designs, it is not possible to obtain PR, since an aliasing free condition cannot be achieved in incompatible NUFB systems [65]. However, the goal is to minimize the PR errors as much as possible and present acceptable magnitude responses for the analysis and possibly synthesis filters.

Several researchers have tried to propose different methodologies to design incompatible NUFB systems with almost PR. One of the earlier designs for the incompatible NUFBs is projected by Nayebi and Barnwel in [15], this method is extended from the work proposed in [17]. Another design is offered in [66] for an incompatible non-uniform transmultiplexer. This method includes minimization of the H_∞ norm of the error system to achieve NPR error. Later in [67], it is shown by Ho et al. that the incompatible NUFB can be designed using FIR CMFB subject to a number of constraints. The authors stated that the optimization method in [68] is used for such a filter bank design. Recently, Chandra, Sharma, and Singha offer a CMFB design for ECG signal compression in [69]; their method is carried out by optimizing an interpolated finite impulse response (IFIR) prototype filter. The results include three NUFBs designs and offer small stopband attenuation and small passband ripple for the analysis filters, and a small reconstruction error.

In the next example 4-9, an incompatible NUFB with 3 sub-bands is designed using the five algorithms.

Example 4-9 Incompatible NUFB [2 3 6] using algorithms 1 to 5

- **Algorithm 1.**

Table 4-28: Incompatible NUFB [2 3 6] using algorithm 1.

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.06]
Ratio	$[0.5\ 0.3\overline{3}\ 0.1\overline{6}]$
Frequency Resolution ρ	512
Computation Time	4:8 min
Max. Initial Distortion error	- 6 dB
Max. Initial Aliasing error	- 3.31 dB
Max. Final Distortion error	- 5.83 dB
Max. Final Aliasing error	- 3.15 dB

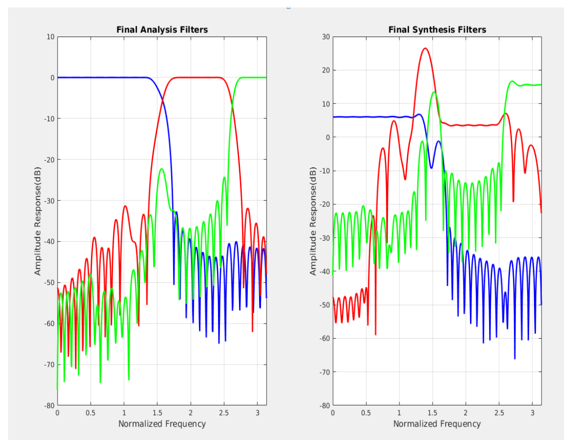

Figure 4-25-a

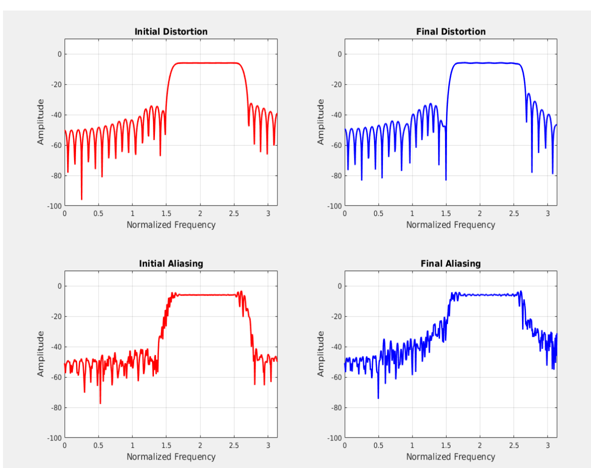

Figure 4-25-b

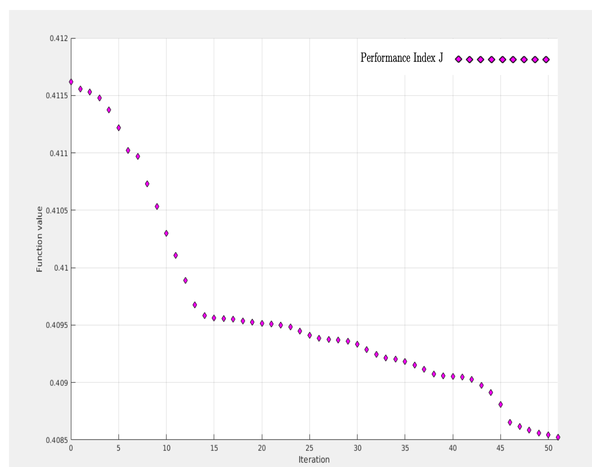

Figure 4-25-c

Figure 4-25 Incompatible NUFB [2 3 6] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 2.**

Table 4-29: Incompatible NUFB [2 3 6] using algorithm 2.

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.06]
Ratio	[0.5 0.3$\overline{3}$ 0.1$\overline{6}$]
Frequency Resolution ρ	512
Computation Time	4.1 sec
Max. Initial Distortion error	- 6 dB
Max. Initial Aliasing error	- 3.31 dB
Max. Final Distortion error	- 0.15 dB
Max. Final Aliasing error	- 5.88 dB

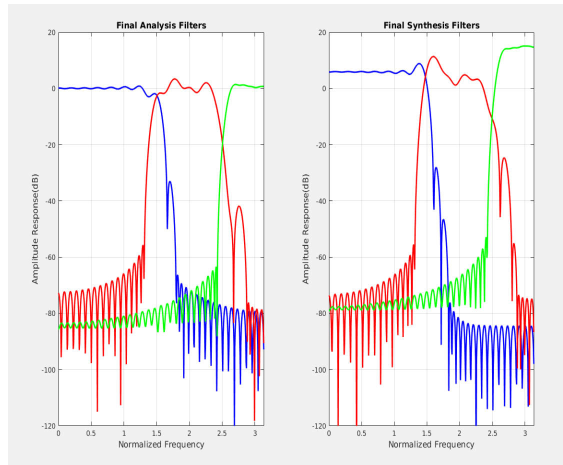

Figure 4-26-a

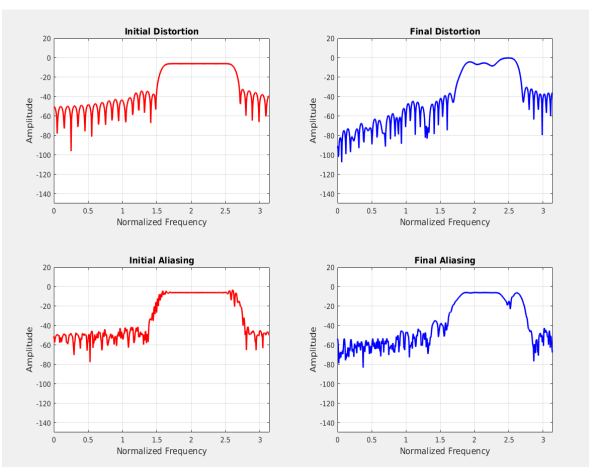

Figure 4-26-b

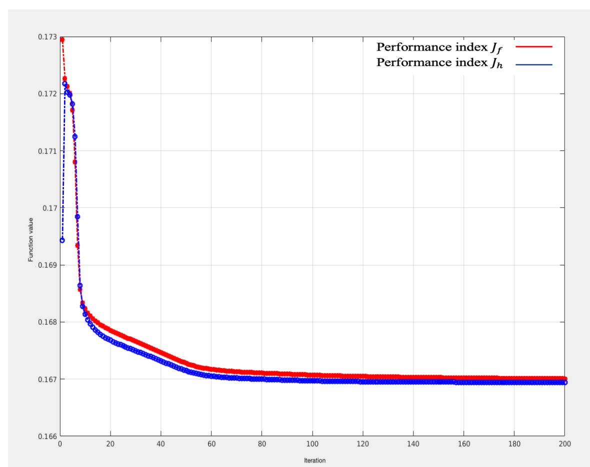

Figure 4-26-c

Figure 4-26 Incompatible NUFB [2 3 6] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 3.**

Table 4-30: Incompatible NUFB [2 3 6] using algorithm 3.

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.06]
Ratio	[0.5 $0.3\overline{3}$ $0.1\overline{6}$]
Frequency Resolution ρ	512
Computation Time	30.36 min
Max. Initial Distortion error	- 6 dB
Max. Initial Aliasing error	- 3.31dB
Max. Final Distortion error	- 5.94 dB
Max. Final Aliasing error	- 3.07 dB

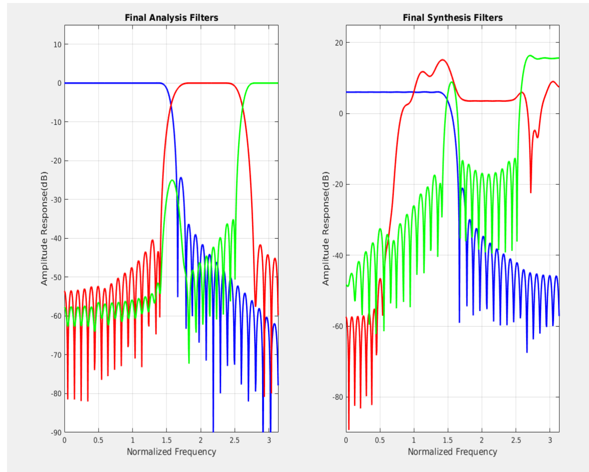

Figure 4-27-a

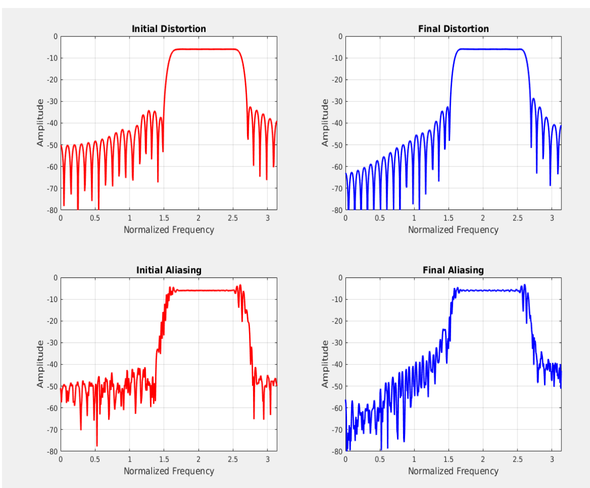

Figure 4-27-b

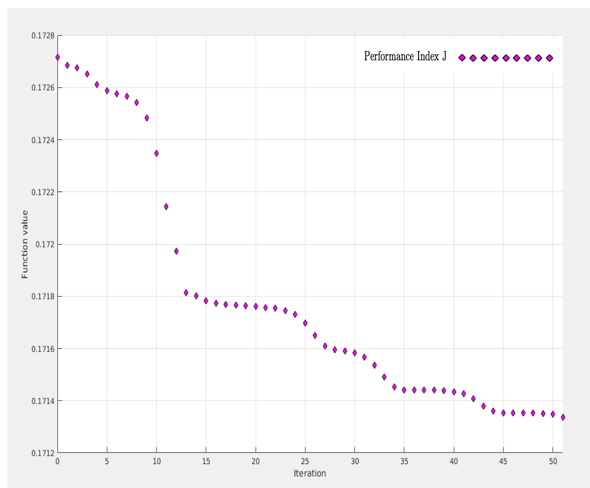

Figure 4-27-c

Figure 4-27 Incompatible NUFB [2 3 6] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 4.**

Table 4-31: Incompatible NUFB [2 3 6] using algorithm 4.

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.06]
Ratio	[0.5 $0.3\overline{3}$ $0.1\overline{6}$]
Frequency Resolution ρ_{sp}	30
Frequency Resolution ρ_{pb}	60
Computation Time	6:36 hr
Max. Initial Distortion error	- 6 dB
Max. Initial Aliasing error	- 3.06 dB
Max. Final Distortion error	- 5.66 dB
Max. Final Aliasing error	- 2.93 dB

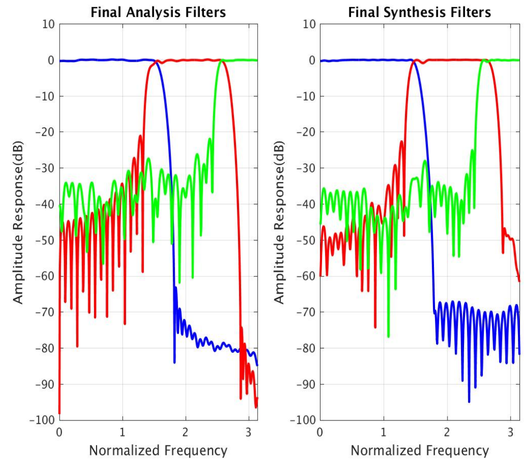

Figure 4-28-a

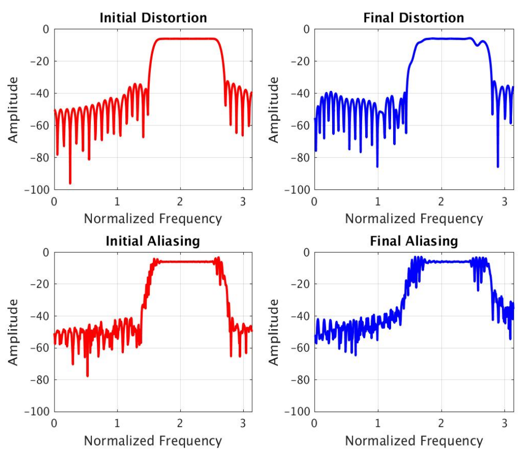

Figure 4-28-b

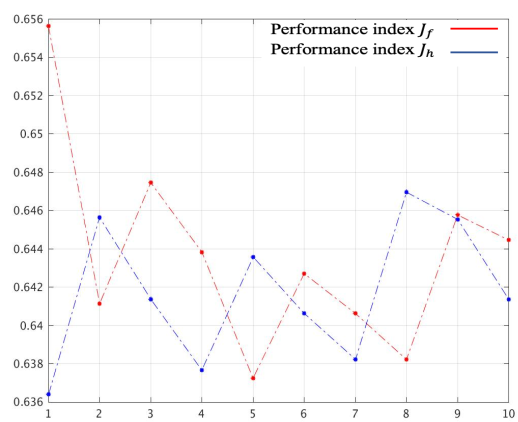

Figure 4-28-c

Figure 4-28 Incompatible NUFB [2 3 6] using algorithm 4. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 5.**

Table 4-32: Incompatible NUFB [2 3 6] using algorithm 5.

Number of Filter Coefficients	64
Transition Factors	[0.08 0.08 0.06]
Ratio	$[0.5\ 0.3\overline{3}\ 0.1\overline{6}]$
Frequency Resolution ρ_{sp}	512
Computation Time	1:54 min
Max. Initial Distortion error	- 6 dB
Max. Initial Aliasing error	- 3.31 dB
Max. Final Distortion error	- 0.49 dB
Max. Final Aliasing error	- 5.85 dB

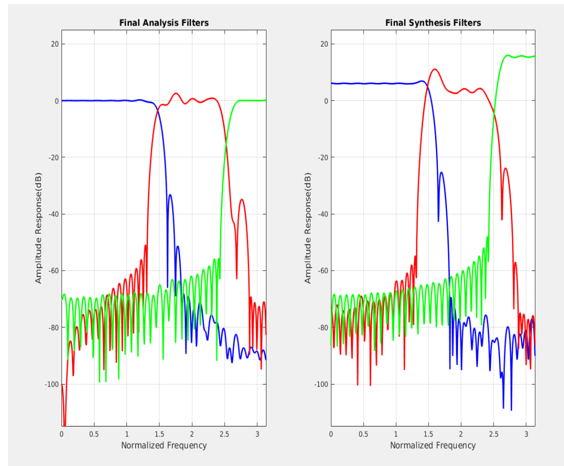

Figure 4-29-a

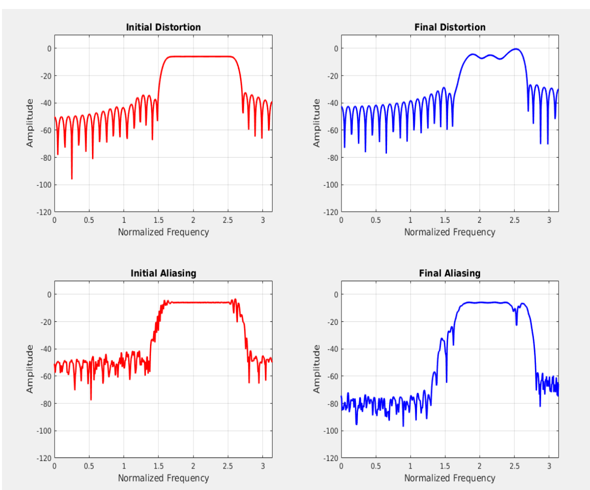

Figure 4-29-b

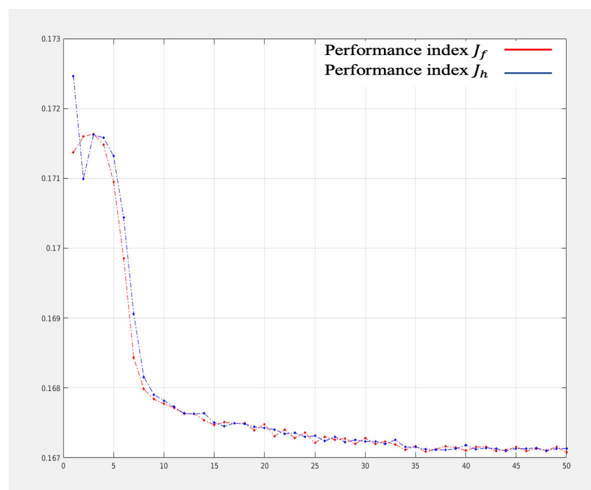

Figure 4-29-c

Figure 4-29 Incompatible NUFB [2 3 6] using algorithm 4. **a** – Final analysis and synthesis filters. **b** – Initial and final distortion and aliasing errors. **c** – Performance index value at each iteration.

Example 4-10 Incompatible NUFB [8 8 4 2 1] using Algorithms 1, 2, 3, and 5.

In example 4-10, four algorithms are used to design the incompatible NUFB with the S = [8 8 4 2 1]. This set is introduced in [28, 43] to design a beamforming system with a nested array. The optimization approach proposed in [28] is able to design a NPR filter bank for this set with a good analysis filters. However, the synthesis filters have poor magnitude response as shown in Figure 4-30-a. In this example, results from algorithms 1, 2, 3, and 5 for this special set are shown. However, the results from algorithms 4 are not presented, because this algorithm is unable to offer acceptable results.

Due to the complexity of this example, it could not be possible to use the same number of filter coefficients in all algorithms. Sine the focus was to show the best performance of each algorithm in this example. The better results are presented with the best number of filter coefficients choice after trying different filter lengths.

- **Algorithm 1.**

Table 4-33: Incompatible NUFB [8 8 4 2 1] using algorithm 1.

Number of Filter Coefficients(N)	64
Transition Factors	[0.015 0.015 0.02 0.025 0.05]
Ratio	[0.0625 0.0625 0.125 0.25 0.5]
Frequency Resolution ρ	256
Computation Time	4:25 min
Max. Initial Distortion error	- 30.48 dB
Max. Initial Aliasing error	- 10.91 dB
Max. Final Distortion error	- 68.06 dB
Max. Final Aliasing error	- 58.97 dB

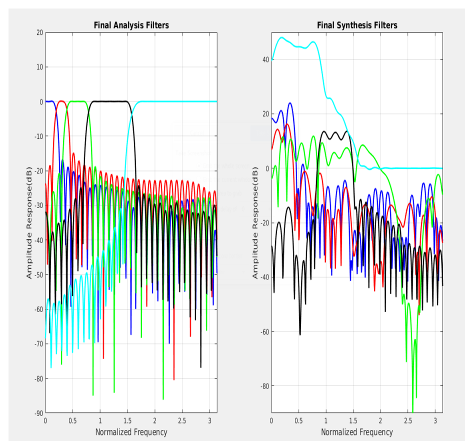

Figure 4-30-a

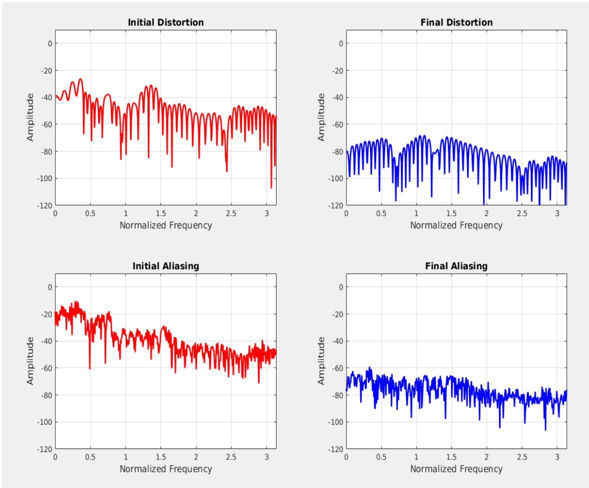

Figure 4-30-b

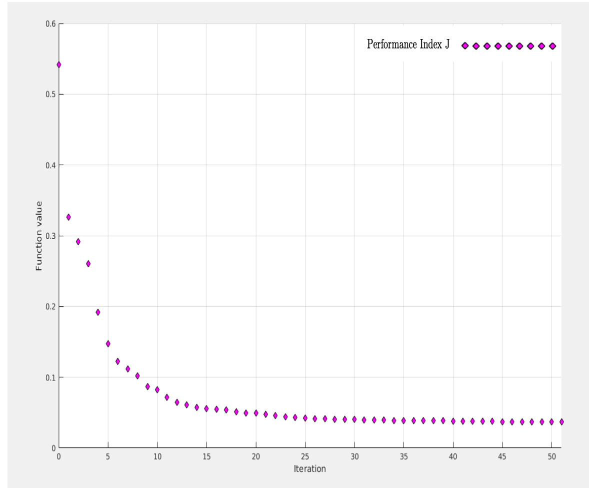

Figure 4-30-c

Figure 4-30 Incompatible NUFB [8 8 4 2 1] using algorithm 1. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 2.**

Table 4-34: Incompatible NUFB [8 8 4 2 1] using algorithm 2.

Number of Filter Coefficients	256
Transition Factors	[0.015 0.015 0.02 0.025 0.05]
Ratio	[0.0625 0.0625 0.125 0.25 0.5]
Frequency Resolution ρ	512
Computation Time	13:1 min
Max. Initial Distortion error	- 47.51 dB
Max. Initial Aliasing error	- 40.67 dB
Max. Final Distortion error	- 67.01 dB
Max. Final Aliasing error	- 62.90 dB

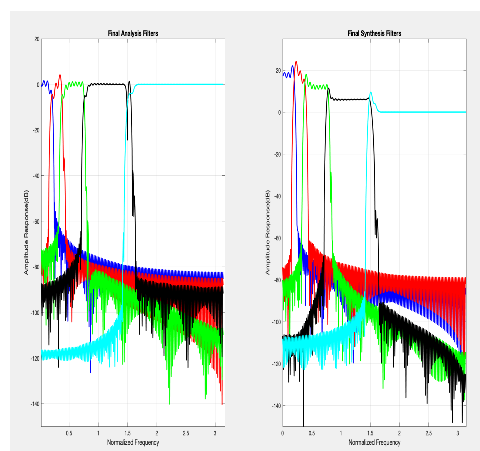

Figure 4-31-a

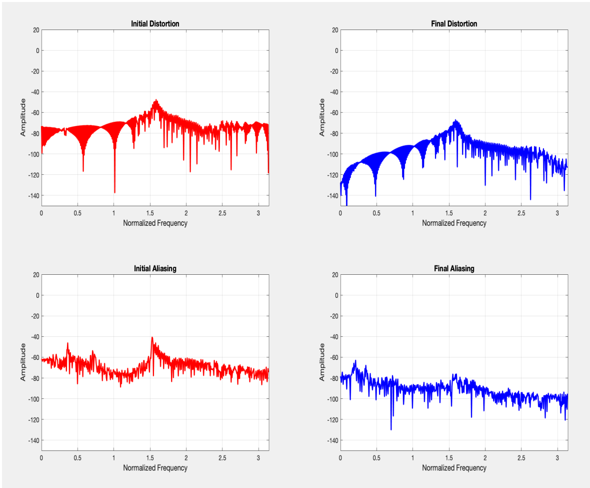

Figure 4-31-b

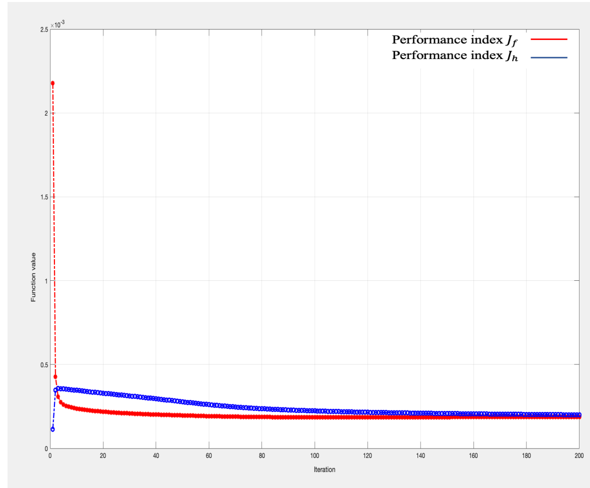

Figure 4-31-c

Figure 4-31 Incompatible NUFB [8 8 4 2 1] using algorithm 2. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 3.**

Table 4-35: Incompatible NUFB [8 8 4 2 1] using algorithm 3.

Number of Filter Coefficients	64
Transition Factors	[0.015 0.015 0.02 0.025 0.05]
Ratio	[0.0625 0.0625 0.125 0.25 0.5]
Frequency Resolution ρ	256
Computation Time	54:40 min
Max. Initial Distortion error	- 21.67 dB
Max. Initial Aliasing error	- 10.27 dB
Max. Final Distortion error	- 28.77 dB
Max. Final Aliasing error	- 10.24 dB

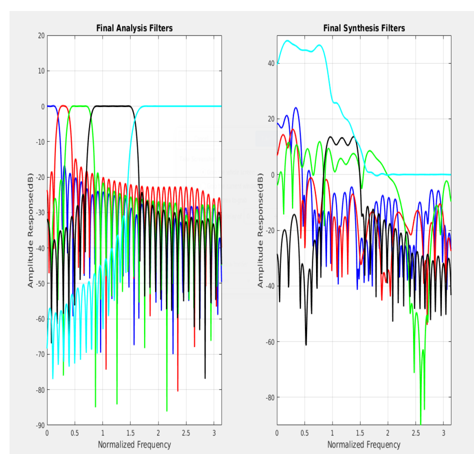

Figure 4-32-a

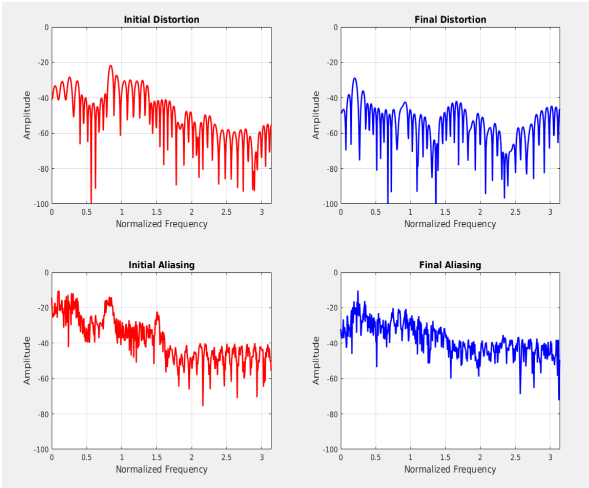

Figure 4-32-b

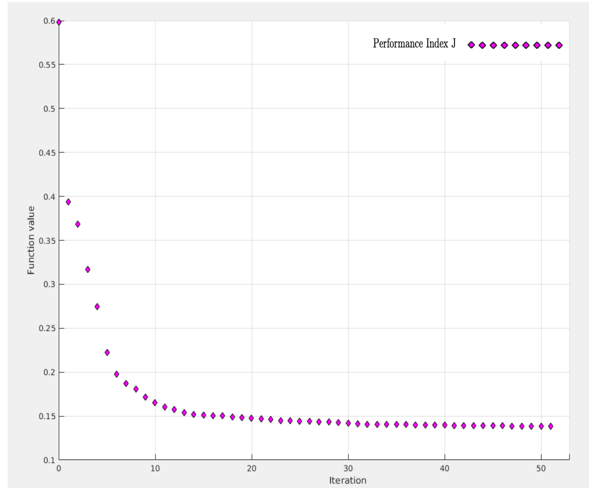

Figure 4-32-c

Figure 4-32 Incompatible NUFB [8 8 4 2 1] using algorithm 3. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

- **Algorithm 5.**

Table 4-36: Incompatible NUFB [8 8 4 2 1] using algorithm 5.

Number of Filter Coefficients	128
Transition Factors	[0.015 0.015 0.02 0.025 0.05]
Ratio	[0.0625 0.0625 0.125 0.25 0.5]
Frequency Resolution ρ	256
Computation Time	1:13 hr
Max. Initial Distortion error	- 37.76 dB
Max. Initial Aliasing error	- 23.24 dB
Max. Final Distortion error	- 4.68 dB
Max. Final Aliasing error	- 13.98 dB

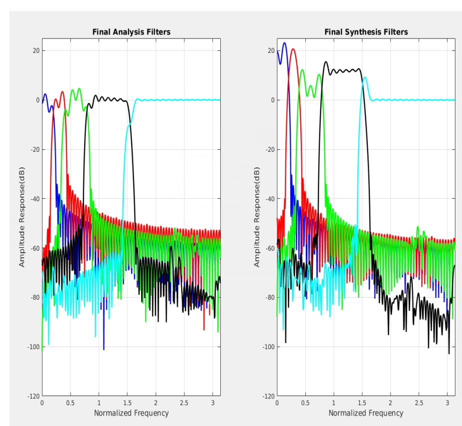

Figure 4-33-a

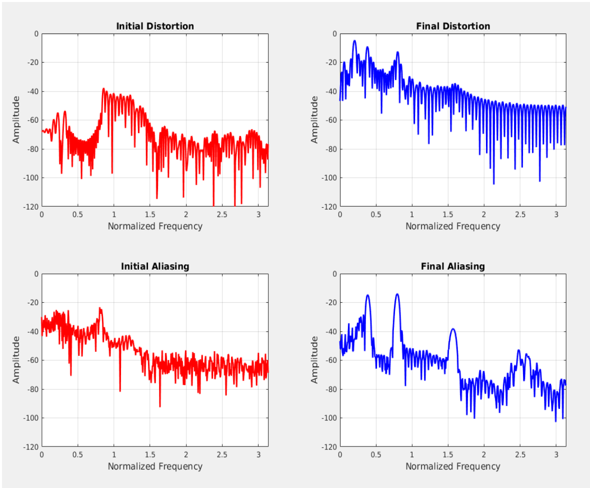

Figure 4-33-b

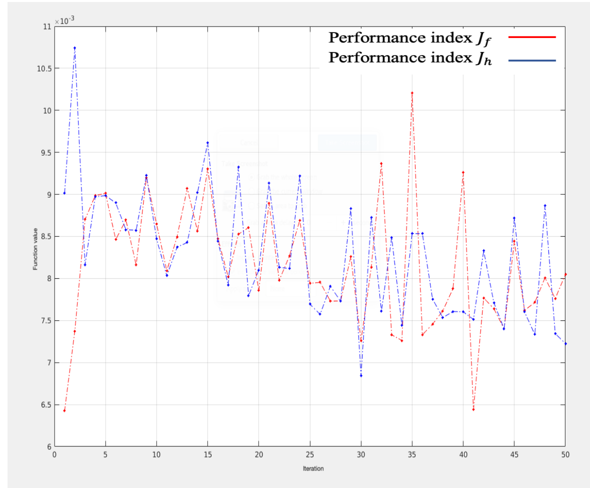

Figure 4-33-c

Figure 4-33 Incompatible NUFB [8 8 4 2 1] using algorithm 5. **a –** Final analysis and synthesis filters. **b –** Initial and final distortion and aliasing errors. **c –** Performance index value at each iteration.

Table 4-37: Incompatible NUFB designs comparison

Design Specifications	N	Algorithms	Analysis filters A_s (dB)	Analysis filters A_p (dB)	Synthesis filters A_s (dB)	Synthesis filters A_p (dB)	Max. Distortion error (dB)	Max. Aliasing error (dB)	Time
Sub-bands (K) S = [2 3 6]		Algorithm 1	22.23	0.011	23.56 +	0.98	- 5.83	- 3.15	4:8 min
Filter Coefficients (N) 64		Algorithm 2	50.25	4.85	49.74	4.53	- 0.15	- 5.88	4.1 sec
Transition Factors		Algorithm 3	24.98	0.005	12.74 +	0.31	- 5.94	- 3.07	30:36 min
$[0.5\ 0.3\bar{3}\ 0.1\bar{6}]$		Algorithm 4	21.03	0.44	22.61	0.37	- 5.66	-2.93	6:36 hr
Ratio [0.5 0.33 0.16]		Algorithm 5	44.81	2.05	47.89	1.74	- 0.49	- 5.85	1:54 min
Sub-bands (K) S = [8 8 4 2 1]	64	Algorithm 1	10.59	0.34	29.23 +	22.35	- 68.06	- 58.97	4:25 min
Filter Coefficients N	256	Algorithm 2	52.01	4.16	53.68	22.73	- 67.01	- 62.90	13:1 min
Transition Factors	64	Algorithm 3	10.83	0.32	48.05+	13.14	- 28.77	- 10:24	54:40 min
[0.015 0.015 0.02 0.025 0.05]	-	Algorithm 4	-	-	-	-	-	-	-
Ratio [0.0625 0.0625 0.125 0.25 0.5]	128	Algorithm 5	29.97	11.24	30.26	10.87	- 4.98	- 13.98	1:13 hr

A_s is the minimum stopband attenuation measured in dB.
A_p is the passband ripple measured in dB.
+ Indicates the stopband attenuation A_s is above the 0 dB.

For the incompatible NUFB example with the S = [2 3 6], the first observation is that, the PR errors are not entirely eliminated over the normalized frequency [0 to π], particularly in the second sub-band. Therefore, evaluating the five algorithms is made by examining and comparing the magnitude response characteristics of the analysis and possibly synthesis filters. The second observation is the use of a greater number of filter coefficients and frequency grid points in this set since this offers more freedom in the design to obtain better results. It is important to mention that an increasing number of filter coefficients and frequency grid points affect the computation time noticeably compared with the time required for UFB and compatible NUFB designs.

In Example 4-9, first, from Figure 4-25-a in algorithm 1, the synthesis filters have poor passband and stopband magnitude response, and this magnitude response is improved slightly by using the modified version (algorithm 3) as shown in Figure 4-27-a.

On the one hand, algorithm 2 provides better synthesis filters, and the passband ripple is equal to 4.53 dB. A large passband ripple also can be noticed in all sub-band filters in algorithm 5, which equals 2.05 dB. This can be expected since the passband response characteristics are not considered in the optimization design in these two algorithms. On the other hand, the stopband attenuation is larger than 44 dB in both analysis and synthesis filters. Moreover, it can be noticed in the filter bank design with the S = [2 3 6], Algorithm 4 takes more than six hours for the computation time. Although this time is considered significantly long, this algorithm gives better magnitude response characteristics than the other four algorithms for both analysis and synthesis filters.

Comparing the results from [67] for the incompatible NUFB design with S = [2 3 6], it can be seen that algorithm 2, 4 and 5 all give better results in the aliasing and distortion errors, and the stopband attenuation in the analysis and synthesis filters. For example, algorithm 4 offers stopband attenuation A_s equals 21.033 dB. However, the method proposed in [67] offers smaller maximum passband ripple A_p than Algorithm 4, which equals 0.2068 dB. The technique proposed in [69] shows results for the same design that reports -48.87 dB for the distortion error and -130.69 dB for the aliasing error. The stopband attenuation for the 3- channel NUFB is about 100 dB as shown in the results.

In example 4-10 for the incompatible NUFB set [8 8 4 2 1], it can be seen that algorithm 1 provides small distortion and aliasing errors, which are less than - 58 dB. However, the synthesis filters have poor magnitude response. Also the stopband attenuation is larger than 0 dB. From Table 4-37, algorithm 3 shows similar results and takes a longer time than algorithm 1. By contrast, increasing the number of filter coefficients to 256 for the same design, algorithm 2 offers better results. The incompatible NUFB still achieves PR errors with - 67.01 for the distortion and - 62.90 dB for the aliasing. However, this design has significant large passband ripples with 4.16 dB and 22.73 dB in the analysis and synthesis filters, respectively. In algorithm 5, the PR errors are far from the efficient, and they are considerably large. Even though the PR errors that are presented in Table 4-37 are the largest PR errors over the normalized frequency [0 to π], the distortion error - 4.98 dB still at the transition band. As shown in Table 4-37, algorithm 5 still has a significant large passband ripple, which is more than 10 dB in both analysis and synthesis filter banks.

4.3 Conclusion

In Chapter 4, an extensive study is made by demonstrating various designs for UFBs, compatible NUFBs, and incompatible NUFBs. The results show that the five algorithms presented in Chapter 3 are able to offer satisfactory designs for such types of filter banks. Results show that algorithm 2 tends to be the fastest algorithm among the five, and algorithm 4 is the slowest one. It can also be seen that algorithm 1 leads to the smallest distortion and aliasing errors in the UFBs and compatible NUFBs. However, in the incompatible non-uniform filter bank examples, this algorithm offers unsatisfactory magnitude responses for the synthesis filters. On the other hand, the magnitude response results show that modifying the performance index in algorithm 1 from [28] improves the stopband attenuation. Also, algorithms 2 and 5 offer larger stopband attenuation compared with the other algorithms and offer better magnitude responses for the synthesis filters in the incompatible NUFBs, particularly, the design with S = [8 8 4 2 1]. Additionally, in this chapter the results show that the CMFB methods from the literature lead to better results compared with the five algorithms, and in many times, they require less computation. However, using the five methods still offer satisfactory results with small number of analysis and synthesis filters' coefficients, particularly in the incompatible filter bank cases. Also, a trade-off is required between having smaller PR errors, and acceptable magnitude response characteristics in the analysis and synthesis filters.

CHAPTER 5

Conclusions and Future Research

5.1 Conclusions

This thesis started with a brief introduction to multirate systems applications and highlighted the importance of these particular systems in our life. The type of filter banks is also illustrated, and some of selected papers that are related to this thesis are reviewed. Following that, the second chapter reviews the main building blocks for multirate systems and shows the differences between sampling rates for the filter bank. The different types of filter banks are defined, and the perfect reconstruction (PR) is introduced. The PR is defined intuitively, and mathematically using a set of equations. Then the discussion about PR conditions is expanded by presenting two errors, distortion and aliasing. Later, three lemmas that contribute to this work enormously are presented. Two of these lemmas consider PR errors, and the third lemma deals with magnitude response design specifications. The five optimization algorithms generated from these lemmas are presented. Different examples of filter bank types such as uniform filter banks (UFBs), compatible non-uniform filter banks (NUFBs) and incompatible NUFBs are designed using these five algorithms. For the presented examples, the conclusions from the evaluated results are:

- **Algorithm 1** gives acceptable results for UFBs and compatible NUFBs. Compared with other algorithms, it provides the smallest aliasing and distortion errors with a considerable number of iterations. Algorithm 1 leads to a satisfactory magnitude response for both analysis and synthesis filters. The drawback of this method that the stopband attenuation is reactively small compared to the other presented algorithms.

- **Algorithm 2** shows that it has a significant fast computation time compared with the other algorithms because it deals with the quadratic objective function. Even though the passband magnitude response characteristics are not included, this algorithm offers an acceptable passband ripple in the filter banks. The examples show also that algorithm 2 gives satisfactory results for incompatible NUFB designs.

- **Algorithm 3** provides better stopband attenuation results than algorithm 1 since this algorithm is a modified version of the first algorithm. Algorithm 1 incudes the PR error, and the magnitude response characteristic error, of the analysis filters, whereas in algorithm 3 includes these two errors in addition to the stopband magnitude response characteristic of the synthesis filters. The performance of algorithm 3 is similar to the optimization scheme in algorithm 1 in terms of optimization computation time and the values of the PR errors.

- **Algorithm 4**, the magnitude response characteristics are not included in the performance index and treated as constraints instead. These constraints are implemented for both analysis and synthesis filters. The results show that using these constraints in this method improves the passband in both analysis and synthesis significantly. The PR conditions and stopband responses characteristics also give satisfactory results in many cases. The method introduced a level of time complexity even though it uses fewer numbers iterations compared with other algorithms in this work.

- **Algorithm 5** is also a constrained optimization approach, and it deals only with the PR, and the stopband magnitude response characteristics for both analysis and synthesis filters as constraints. In terms of time complexity, this algorithm requires less time for computation compared with algorithm 4, and it is comparable to algorithm 2. This method can still provide satisfactory small PR errors. However, these PR errors are considerably higher in the transition areas. Algorithm 5 also gives good stopband magnitude responses; it works with a satisfactory level for the incompatible NUFB designs.

In this work, the five algorithms show that they can achieve NPR and meet the magnitude responses design specifications at an acceptable level. The results from these algorithms are comparable with the results from the literature in many cases and sometimes better, particularly the cosine modulated filter bank (CMFB) method. The trade-off is between the design specifications required when choosing between these algorithms presented in this work. This trade-off deponed on what is the application of the designed filter bank

5.2 Future Research

In this thesis, filter bank design using optimization techniques considers the PR conditions and the stopband and passband magnitude response characteristics. The work might be extended to include other specifications, such as the transition band areas.

It might be interesting to investigate in more depth the performance index introduced in equation (3-7). Instead of not including the passband response characteristic as in algorithm 2, or using the magnitude response characteristic as constraints as in algorithms 4 and 5. Better results might be obtained if this performance index reformulated as a convex problem following the optimization methods presented in [49]

Since the relevant weights for the PR and the frequency specifications are not used in this thesis, it would be useful to use these weights and examine how they contribute to the filter bank designs results.

Finding a criterion for dealing with the number of FIR filters' coefficients for filter bank design might be an area of study since it was noticed that this number of coefficients affects the results. Reducing the number of FIR filters' coefficients is also considered when hardware implementation is involved.

The literature shows several attempts to design an IIR filter bank; it would be an interesting research area to design an IIR filter bank using one of the introduced optimization algorithms in this work.

Bibliography

[1] S. W. Smith, "The scientist and engineer's guide to digital signal processing," 1997.

[2] F. J. Harris, *Multirate signal processing for communication systems*. Prentice Hall PTR, 2004.

[3] B. W. Suter, *Multirate and wavelet signal processing*. Elsevier, 1997.

[4] K. Thyagarajan, *Introduction to Digital Signal Processing Using MATLAB with Application to Digital Communications*. Springer, 2018.

[5] J. Johnston, "A filter family designed for use in quadrature mirror filter banks," in *ICASSP'80. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1980, vol. 5, pp. 291-294: IEEE.

[6] G. Wackersreuther, "A novel approach to the design of filters for filter banks," in *ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1985, vol. 10, pp. 73-76: IEEE.

[7] G. Wackersreuther, "Some new aspects of filters for filter banks," *IEEE transactions on acoustics, speech, and signal processing,* vol. 34, no. 5, pp. 1182-1200, 1986.

[8] P. Vaidyanathan, "Quadrature mirror filter banks, M-band extensions and perfect-reconstruction techniques," *IEEE Assp Magazine,* vol. 4, no. 3, pp. 4-20, 1987.

[9] A. Tkacenko, "Optimization algorithms for realizable signal-adapted filter banks," California Institute of Technology, 2004.

[10] A. Kumar, B. Kuldeep, I. Sharma, G. Singh, and H. Lee, "Advances in multirate filter banks: A research survey," in *Advances in Multirate Systems*: Springer, 2018, pp. 35-57.

[11] Y.-P. Lin and P. Vaidyanathan, "Linear phase cosine modulated maximally decimated filter banks with perfect reconstruction," *IEEE Transactions on Signal Processing,* vol. 43, no. 11, pp. 2525-2539, 1995.

[12] N. Subbulakshmi and R. Manimegalai, "A comprehensive survey on filter bank designs for digital hearing aids," (in English), *Asian Journal of Information Technology* vol. 16, no. 2 pp. 190-199, 2017.

[13] A. Kumar, "A comparative study of performance of Blackman window family for designing cosine-modulated filter bank," in *Proc. Int. Conf. Circuits, Syst. Simul.*, 2011.

[14] P.-Q. Hoang and P. Vaidyanathan, "Non-uniform multirate filter banks: Theory and design," in *IEEE International Symposium on Circuits and Systems*, 1989, pp. 371-374: IEEE.

[15] K. Nayebi and T. Barnwell, "Reconstruction from incompatible nonuniform band filter bank," in *1993 IEEE International Symposium on Circuits and Systems*, 1993, pp. 112-115: IEEE.

[16] K. Nayebi, T. P. Barnwell, and M. J. Smith, "Nonuniform filter banks: A reconstruction and design theory," *IEEE Transactions on Signal Processing,* vol. 41, no. 3, pp. 1114-1127, 1993.

[17] K. Nayebi, T. P. Barnwell, and M. J. Smith, "Time-domain filter bank analysis: A new design theory," *IEEE Transactions on Signal Processing,* vol. 40, no. 6, pp. 1412-1429, 1992.

[18] J. Li, T. Q. Nguyen, and S. Tantaratana, "A simple design method for near-perfect-reconstruction nonuniform filter banks," *IEEE Transactions on Signal Processing,* vol. 45, no. 8, pp. 2105-2109, 1997.

[19] T. Chen, L. Qiu, and E.-W. Bai, "General multirate building structures with application to nonuniform filter banks," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing,* vol. 45, no. 8, pp. 948-958, 1998.

[20] C. Ho, B. Ling, Y. Liu, P. Tam, and K.-L. Teo, "Optimum nonuniform transmultiplexer design," in *International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003*, 2003, vol. 1, pp. 740-743: IEEE.

[21] C.-F. Ho, B. W.-K. Ling, Y.-Q. Liu, P.-S. Tam, and K.-L. Teo, "Optimal design of nonuniform FIR transmultiplexer using semi-infinite programming," *IEEE Transactions on Signal Processing,* vol. 53, no. 7, pp. 2598-2603, 2005.

[22] C.-F. Ho, B.-K. Ling, Y.-Q. Liu, P.-S. Tam, and K.-L. Teo, "Efficient algorithm for solving semi-infinite programming problems and their applications to nonuniform filter bank designs," *IEEE Transactions on Signal Processing,* vol. 54, no. 11, pp. 4223-4232, 2006.

[23] X. Xie, L. Liang, and G. Shi, "Analysis of realizable rational decimated nonuniform filter banks with direct structure," *Progress in Natural Science,* vol. 18, no. 12, pp. 1501-1505, 2008.

[24] W. Zhong, G. Shi, X. Xie, and X. Chen, "Design of linear-phase nonuniform filter banks with partial cosine modulation," *IEEE Transactions on Signal Processing,* vol. 58, no. 6, pp. 3390-3395, 2010.

[25] W. Zhong, L. Fang, L. Ye, Q. Zhang, and S. Shi, "Design of linear-phase oversampled nonuniform filter banks with arbitrary integer sampling factors," in *2014 10th International Conference on Natural Computation (ICNC)*, 2014, pp. 915-920: IEEE.

[26] A. Kumar, G. Singh, and S. Anurag, "An optimized cosine-modulated nonuniform filter bank design for subband coding of ECG signal," *Journal of King Saud University-Engineering Sciences,* vol. 27, no. 2, pp. 158-169, 2015.

[27] A. Kumar, G. K. Singh, and S. Anurag, "Design of nearly perfect reconstructed non-uniform filter bank by constrained equiripple FIR technique," *Applied Soft Computing,* vol. 13, no. 1, pp. 353-360, 2013.

[28] I. Moazzen and P. Agathoklis, "A general approach for filter bank design using optimization," *IET Journal on Signal Processing (Submitted),* 2014.

[29] I. Moazzen and P. Agathoklis, "Design of filterbanks using a fast optimization approach," in *2015 IEEE 6th Latin American Symposium on Circuits & Systems (LASCAS)*, 2015, pp. 1-4: IEEE.

[30] P. P. Vaidyanathan, *Multirate systems and filter banks*. Pearson Education India, 1993.

[31] P. Jacob and B. Anoop, "Design and implementation of polyphase decimation filter," *International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN,* pp. 2250-3501, 2014.

[32] A. Ikhlef and J. Louveaux, "An enhanced MMSE per subchannel equalizer for highly frequency selective channels for FBMC/OQAM systems," in *2009 IEEE 10th Workshop on Signal Processing Advances in Wireless Communications*, 2009, pp. 186-190: IEEE.

[33] N. J. Fliege, *Multirate digital signal processing*. John Wiley New York, 1994.

[34] J. Princen and A. Bradley, "Analysis/synthesis filter bank design based on time domain aliasing cancellation," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 34, no. 5, pp. 1153-1161, 1986.

[35] S. Akkarakaran and P. Vaidyanathan, "New results and open problems on nonuniform filter-banks," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, 1999, vol. 3, pp. 1501-1504: IEEE.

[36] M. J. Absar and S. George, "On the search for compatible numbers in the design of maximally decimated perfect reconstruction non-uniform filter bank," in *2001 IEEE Workshop on Signal Processing Systems. SiPS 2001. Design and Implementation (Cat. No. 01TH8578)*, 2001, pp. 141-148: IEEE.

[37] P. Vaidyanathan, "Theory and design of M-channel maximally decimated quadrature mirror filters with arbitrary M, having the perfect-reconstruction property," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 35, no. 4, pp. 476-492, 1987.

[38] M. Smith and T. Barnwell, "A new filter bank theory for time-frequency representation," *IEEE transactions on acoustics, speech, and signal processing,* vol. 35, no. 3, pp. 314-327, 1987.

[39] Z. Wei, L. Li, and Z. Qin, "Necessary Conditions on Sampling Factors for Nonuniform Filter Banks Satisfying Perfect Reconstruction Property," *Information Technology Journal,* vol. 12, no. 17, p. 3991, 2013.

[40] M. Vetterli, "A theory of multirate filter banks," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol. 35, no. ARTICLE, pp. 356-372, 1987.

[41] P. P. Vaidyanathan and S. K. Mitra, "Polyphase networks, block digital filtering, LPTV systems, and alias-free QMF banks: A unified approach based on pseudocirculants," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 36, no. 3, pp. 381-391, 1988.

[42] M. Vetterli, "Perfect transmultiplexers," in *ICASSP'86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1986, vol. 11, pp. 2567-2570: IEEE.

[43] I. Moazzen, "Array signal processing for beamforming and blind source separation," 2013.

[44] K. Sumanth, S. N. Bhavanam, and B. B. Rao, "Design of Non-Uniform Cosine Modulated Filter Banks Using Windows," *International Journal of Engineering Research and Applications,* vol. 6, no. 4, pp. 32-38, 2016.

[45] T. Q. Nguyen, "Digital filter bank design quadratic-constrained formulation," *IEEE Transactions on Signal Processing,* vol. 43, no. 9, pp. 2103-2108, 1995.

[46] B. D. Riel, "Design and implementation of oversampled modulated filter banks," 2005.

[47] T. Saramäki and J. Yli-Kaakinen, "Design of digital filters and filter banks by optimization: applications," in *2000 10th European Signal Processing Conference*, 2000, pp. 1-2: IEEE.

[48] I. Moazzen and P. Agathoklis. (2016). *Filter Bank Design (MATLAB function Filter Bank Design) (version 1.2.0.0 ed.)*. Available: https://www.mathworks.com/matlabcentral/fileexchange/40128-filter-bank-design

[49] A. Antoniou and W.-S. Lu, *Practical optimization: algorithms and engineering applications*. Springer Science & Business Media, 2007.

[50] I. Moazzen and P. Agathoklis. (2013). *Fast Filter Bank Design (MATLAB function ffbd) (version 1.12.0.0 ed.)*. Available: https://www.mathworks.com/matlabcentral/fileexchange/44023-fast-filter-bank-design-ffbd

[51] NEOS. (N.A.). *Semi-infinite programming*. Available: https://neos-guide.org/content/semi-infinite-programming#references

[52] M. López and G. Still, "Semi-infinite programming," *European Journal of Operational Research,* vol. 180, no. 2, pp. 491-518, 2007.

[53] R. Reemtsen, *Semi-infinite Programming: Discretization Methods: SIP*. Springer, 2009.

[54] T. Tran, M. Dahl, and I. Claesson, "A semiinfinite quadratic programming algorithm with applications to channel equalization," in *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, 2003, vol. 1, pp. 653-656: IEEE.

[55] J. M. d. Haan, S. Nordholm, and I. Claesson, *Design of Nonuniform Filter Banks with Frequency Domain Criteria*. 2004.

[56] T. Q. Nguyen, "Near-perfect-reconstruction pseudo-QMF banks," *IEEE Transactions on signal processing,* vol. 42, no. 1, pp. 65-76, 1994.

[57] I. Moazzen and P. Agathoklis. (2011). *Uniform filter bank (Matlab function PR_Nguyen) (version 1.4.0.0 ed.)*. Available: https://www.mathworks.com/matlabcentral/fileexchange/35053-uniform-filter-bank
[58] G. Doblinger, "A fast design method for perfect-reconstruction uniform cosine-modulated filter banks," *IEEE Transactions on Signal Processing,* vol. 60, no. 12, pp. 6693-6697, 2012.
[59] G. Doblinger. (2012). *A Fast design method for perfect reconstruction uniform cosine-modulated filter bank (MATLAB function prfib)*. Available: https://www.nt.tuwien.ac.at/staff-pages/gerhard-doblinger-filter-bank-design/
[60] I. Djokovic and P. P. Vaidyanathan, "Results on biorthogonal filter banks," *Applied and computational harmonic analysis,* vol. 1, no. 4, pp. 329-343, 1994.
[61] J. Ogale and S. Ashok, "Cosine Modulated Non-Uniform Filter Banks," *J. Signal and Information Processing,* vol. 2, no. 3, pp. 178-183, 2011.
[62] S. Anurag and A. Kumar, "Non-uniform filter bank design using modified window functions," in *2012 2nd National Conference on Computational Intelligence and Signal Processing (CISP)*, 2012, pp. 165-170: IEEE.
[63] R. K. Soni, A. Jain, and R. Saxena, "An Optimized Design of Non-uniform Filterbank using Blackman Window Family," *International Journal of Signal & Image Processing,* vol. 1, no. 1, 2010.
[64] A. Kumar, R. K. Sunkaria, and L. D. Sharma, "Design of Cosine Modulated Non-uniform filter bank using Particle Swarm Optimization," in *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2018, pp. 614-618: IEEE.
[65] W.-K. Ling, *Nonlinear digital filters: analysis and applications*. Academic Press, 2010.
[66] A. S. Mehr, "Alias-component matrices of nonuniform transmultiplexers," *Circuits, Systems & Signal Processing,* vol. 28, no. 1, pp. 85-97, 2009.
[67] B. W.-K. Ling, C. Y.-F. Ho, K.-L. Teo, W.-C. Siu, J. Cao, and Q. Dai, "Optimal design of cosine modulated nonuniform linear phase FIR filter bank via both stretching and shifting frequency response of single prototype filter," *IEEE transactions on signal processing,* vol. 62, no. 10, pp. 2517-2530, 2014.
[68] C. Y.-F. Ho, B. W.-K. Ling, Z.-W. Chi, M. Shikh-Bahaei, Y.-Q. Liu, and K.-L. Teo, "Design of near-allpass strictly stable minimal-phase real-valued rational IIR filters," *IEEE Transactions on Circuits and Systems II: Express Briefs,* vol. 55, no. 8, pp. 781-785, 2008.
[69] S. Chandra, A. Sharma, and G. Singh, "Computationally Efficient Cosine Modulated Filter Bank Design for ECG Signal Compression," *IRBM,* 2019.

Appendix A

MATLAB Functions Calls of the Algorithms

A.1 – Algorithm 1 Function's Call and Description [28]

```
%[h,f]=algorithm_1(nk,ratio,transition_factor,L,ro,iteration_max,function_eva
luation_max,cost_function_weights,initial_filters);

% This function can be used to design all types of filter banks (critically
sampled uniform filter banks, compatible non-uniform filter banks, and
incompatible non-uniform filter banks).
% By using optimization based on the algorithm presented in ref. [1].

% Problem Description: Design K analysis and synthesis FIR filters so that
% the analysis filters satisfy some frequency specifications and the filter
bank
% (almost) meets the perfect reconstruction (PR) conditions. Both goals are
achieved by minimizing
% the performance index in equation (3-7) for Algorithm 1 in ref. [2].

% The optional weights are not used in the provided examples, so they are set
to (1)

% MATLAB Requirements: To run this function, the used MATLAB needs to have
the following toolboxes:
% 1- signal processing toolbox, 2- filter design toolbox
%
% ---------------------------------------------------------------
% Function inputs:
% 1- nk: an M-element vector which includes the set of sampling rates.
% 2- ratio: the frequency ratio occupied by each individual analysis filter.
% e.g. 0.5 means half of the whole normalized frequency spectrum is occupied
by the filter.
% 3- transition_factor: a vector with the same length as the "ratio" vector,
and it is used to define passband & stopband cut-off frequencies. Smaller
transition_factor results in sharper transition band.
% Note: All elements in the "transition_factor" vector must be less than half
of the smallest element in the "ratio" vector.
%
% Usin g the "ratio" vector and the "transition_factor" vector, the analysis
filters' frequency specifications are given as:First filter- lowpass filter
with the following normalized cut-off frequencies:
%   a) passband cut-off frequency: [ratio(1)-transition_factor(1)]*pi
%   b) stopband cut-off frequency: [ratio(1)+transition_factor(1)]*pi
%    Last filter-  highpass filer with the following cut-off frequencies:
%   a) passband cut-off frequency: pi-[ratio(end)-transition_factor(end)]*pi
%   b) stopband cut-off frequency: pi-[ratio(end)+transition_factor(end)]*pi
%    Other filers- bandpass filters with the following cut-off frequencies:
%   a) first stopband cut-off frequency: [sum(ratio(1:i-1))-
transition_factor(i)]*pi
%   b) first passband cut-off frequency: [sum(ratio(1:I
1))+transition_factor(i)]*pi
```

```
%   c) second passband cut-off frequency:[sum(ratio(1:i-1))+ratio(i)-
transition_factor(i)]*pi
%    d) second stopband cut-off frequency:[sum(ratio(1:i-
1))+ratio(i)+transition_factor(i)]*pi
%
% 4- L: filter length which is the same for both FIR analysis and synthesis
filters.
% 5- ro: frequency resolution, i.e. "ro" points with a uniform distribution
from 0 to pi (positive part of the
%    normalized frequencies) are used to compute the frequency response of
the analysis filters(see Eq.(29) in [1]).
% 6- iteration_max: Maximum number of iterations allowed by the optimization
algorithm (see "optimset" in Matlab).
% 7- function_evaluation_max: Maximum number of function evaluations allowed
by the optimization algorithm (see "optimset" in Matlab).
% 8- cost_function_weights: a 2-element vector consisting of optional weights
to compute cost function.
%    The first element is the weight for PR error (w1 in the performance
%    index defined above) and the second element is the weight for frequency
specification error (w2)
% 9- initial_filters: Initial analysis filters for the optimization algorithm
which can be defined by user or automatically
%    generated by the function (set the parameter to "[]")
% -------------------------------------------------------------
% Function outputs:
% 1- h: is a matrix in which each row is corresponding to one of the analysis
filters
%    e.g. if the sampling set (nk) is [2 4 4], the first row of h contains
the coefficients
%    of the analysis filter on the branch with subsampling of 2.
% 2- f: is a matrix in which each row is corresponding to one of the
synthesis filters
%    Note: "distortion" and "aliasing" are calculated for 512 frequencies
distributed uniformly over the normalized frequency [0 to pi]

% This MATLAB code presents the following after completion:
% - Computation time for each design
% - Maximum initial and final aliasing and distortion errors. These errors
are defined as the largest dB values over the normalized frequency [0 to pi]
% - Analysis and Synthesis filters' minimum stopband attenuation among all
filters measured in dB among all filters.
% - Analysis and Synthesis filters' largest passband ripple among all filters
measured in dB.

% ----------------------------- Examples ---------------------------------

% Example 1: Critically Sampled Uniform Filter Bank - Sampling set is {2,2}
% [h,f]=algorithm_1([2 2],[0.5 0.5],[0.1 0.1],64,64,50,20000,[1 1],[]);

% Example 2: Critically Sampled Uniform Filter Bank - Sampling set is
{4,4,4,4}
% [h,f]=algorithm_1([4 4 4 4],[0.25 0.25 0.25 0.25],[0.08 0.08 0.08
0.08],56,64,50,20000,[1 1],[]);


% Example 3: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,4}
```

```
% [h,f]=algorithm_1([2 4 4],[0.5 0.25 0.25],[0.1 0.1 0.1],48,60,50,80000,[1
1],[]);

% Example 4: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,8,8}
% [h,f]=algorithm_1([2 4 8 8],[0.5 0.25 0.125 0.125],[0.09 0.08 0.062
0.062],76,100,100,80000,[1 1],[]);


% Example 5: Incompatible Non-Uniform Filter Bank - Sampling set is {2,3,6}
% [h,f]=algorithm_1([2 3 6],[1/2 1/3 1/6],[0.08 0.08 0.06],64,512,50,80000,[1
1],[]);


% Example 6: Incompatible Non-Uniform Filter Bank - Sampling set is
{8,8,4,2,1}
% [h,f]=algorithm_1([8 8 4 2 1],[0.0625 0.0625 0.125 0.25 0.5],[0.015 0.015
0.02 0.025 0.05],64,256,50,80000,[1 1],[]);

% Notes ----------------------------------------------------------------

% [1] Moazzen I., Agathoklis P., "A General Approach for Filter Bank Design
Using Optimization",
% Technical Report, 2014.
%
% Written By:   Iman Moazzen     2014
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%
% [2] ElGarewi A., Agathoklis P., "Analysis of Algorithms for Filter Bank
Design Optimization",
% MASc. thesis, 2019.

% Modified By:  Ahmed ElGarewi    May 30, 2018
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%***************************************************************************
```

A.2 – Algorithm 2 Function's Call and Description [29]

```
%[h,f]=algorithm_2(N,nk,ratio,transition_factor,wf,wh,ro,iter)

% This function can be used to design all types of filter banks (critically
sampled uniform filter banks, compatible non-uniform filter banks, and
incompatible non-uniform filter banks).
%By using optimization based on the algorithm presented in ref. [1].

% Problem Description: Design K analysis and synthesis FIR filters so that
% the perfect reconstruction (PR) conditions are satisfied (or almost
satisfied) and the energy
% in the analysis and synthesis filters' stopband areas is minimized (strong
stopband attenuation).

% By minimizing the performance indices in equations (3-23) and (3-28) for
Algorithm 2 ref. [2].

% The optional weights are not used in the provided examples, so they are set
to (1)
%
% MATLAB Requirements: To run this function, the used MATLAB needs to have
the following toolboxes:
% 1- signal processing toolbox, 2- filter design toolbox
%
% ---------------------------------------------------------------
% Function inputs:
% 1- N: filter length which is the same for both FIR analysis and synthesis
filters.
% 2- nk: an M-element vector which includes the set of sampling rates.
% 3- ratio: the frequency ratio occupied by each individual
analysis/synthesis filter.
%    e.g. 0.5 means half of the whole normalized frequency spectrum is
occupied by the filter.
% 4- transition_factor: a vector with the same length as the "ratio" vector.
It is used to define
%    passband & stopband cut-off frequencies. Smaller transition_factor
results in sharper transition band.
%    Note: Each element in the "transition_factor" vector must be less than
half of the smallest element
%    of the corresponding "ratio" vector entry.
%
%    Using the "ratio" vector and the "transition_factor" vector, the
analysis/synthesis filters' frequency
%    specifications are given as:
%    First filter- lowpass filter with the following normalized cut-off
frequencies:
%   a) passband cut-off frequency: [ratio(1)-transition_factor(1)]*pi
%   b) stopband cut-off frequency: [ratio(1)+transition_factor(1)]*pi
%    Last filter-  highpass filer with the following cut-off frequencies:
%   a) passband cut-off frequency: pi-[ratio(end)-transition_factor(end)]*pi
%   b) stopband cut-off frequency: pi-[ratio(end)+transition_factor(end)]*pi
%    Other filers- bandpass filters with the following cut-off frequencies:
```

```
%   a) first stopband cut-off frequency: [sum(ratio(1:i-1))-
transition_factor(i)]*pi
%   b) first passband cut-off frequency: [sum (ratio (1:I
1))+transition_factor(i)]*pi
%   c) second passband cut-off frequency: [sum(ratio(1:i-1))+ratio(i)-
transition_factor(i)]*pi
%   d) second stopband cut-off frequency: [sum(ratio(1:i-
1))+ratio(i)+transition_factor(i)]*pi
%
% 5- wf: a 2-element vector consisting of optional weights defined in J1.
%    The first element is the weight for the PR error (wf_1 in J1 defined
above) and the
%    second element is the weight for the energy in stopband area (wf_2)
% 6- wh: a 2-element vector consisting of optional weights defined in J2.
%    The first element is the weight for the PR error (wh_1 in J2 defined
above) and the
%    second element is the weight for the energy in stopband area (wh_2)
%
% 7- ro: frequency resolution, i.e. "ro" points with a uniform distribution
within the filters' stop-band area
%    are considered to compute the energy (see [1]). "ro" must be an even
number.
% 8- iter: Maximum number of iterations allowed.
% -----------------------------------------------------------------
% Function outputs:
% 1- h: is a matrix in which each row is corresponding to one of the analysis
filters
%    e.g. if the sampling set (nk) is [2 4 4], the first row of h contains
the coefficients
%    of the analysis filter on the branch with subsampling of 2.
% 2- f: is a matrix in which each row is corresponding to one of the
synthesis filters
%  Note: "distortion" and "aliasing" are calculated for 512 frequencies
distributed uniformly over the normalized frequency [0 to pi]

% This MATLAB code presents the following after completion:
% - Computation time for each design
% - Maximum initial and final aliasing and distortion errors. These errors
are defined as the largest dB values over the normalized frequency [0 to pi]
% - Analysis and Synthesis filters' minimum stopband attenuation among all
filters measured in dB among all filters.
% - Analysis and Synthesis filters' largest passband ripple among all filters
measured in dB.

% ----------------------------- Examples ----------------------------------
%
% Example 1: Critically Sampled Uniform Filter Bank - Sampling set is {2,2}
% [h,f]=algorithm_2(64,[2 2],[0.5 0.5],[0.1 0.1],[1 1],[1 1],64,50);

% Example 2: Critically Sampled Uniform Filter Bank - Sampling set is
{4,4,4,4}
% [h,f]=algorithm_2(56,[4 4 4 4],[0.25 0.25 0.25 0.25],[0.08 0.08 0.08
0.08],[1 1],[1 1],64,50);
%
% Example 3: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,4}
% [h,f]=algorithm_2(48,[2 4 4],[0.5 0.25 0.25],[0.1 0.1 0.1],[1 1],[1
1],64,50);
```

```
% Example 4: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,8,8}
% [h,f]=algorithm_2(76,[2 4 8 8],[0.5 0.25 0.125 0.125],[0.09 0.08 0.062
0.062]],[1 1],[1 1],64,50);

% Example 5: Incompatible Non-Uniform Filter Bank - Sampling set is {2,3,6}
%[h,f]=algorithm_2(64,[2 3 6],[1/2 1/3 1/6],[0.08 0.08 0.06],[1 1],[1
1],512,50);

% Example 6: Incompatible Non-Uniform Filter Bank - Sampling set is
{8,8,4,2,1}
% [h,f]=algorithm_2(64,[8 8 4 2 1],[0.0625 0.0625 0.125 0.25 0.5],[0.015
0.015 0.02 0.025 0.05],[1 1],[1 1],256,200);

% Notes ----------------------------------------------------------------


% [1] Moazzen I., Agathoklis P., "Design of Filterbanks Using a Fast
Optimization Approach",
% IEEE Sixth Latin American Symposium on Circuits and Systems (LASCAS), 2015.

% Written By:   Iman Moazzen      2014
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%
% [2] ElGarewi A., Agathoklis P., "Analysis of Algorithms for Filter Bank
Design Optimization",
% MASc. thesis, 2019.

% Modified By:  Ahmed ElGarewi    May 30, 2018
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%**************************************************************************
```

A.3 – Algorithm 3 Function's Call and Description

```
%[h,f]=algorithm_3(nk,ratio,transition_factor,L,ro,iteration_max,function_eva
luation_max,cost_function_weights,wf,initial_filters);

% This function can be used to design all types of filter banks (critically
sampled uniform filter banks, compatible non-uniform filter banks, and
incompatible non-uniform filter banks).
%
% Problem Description: Design K analysis and synthesis FIR filters so that
% the analysis filters satisfy some frequency specifications and the filter
bank
% (almost) meets the perfect reconstruction (PR) conditions. Both goals are
achieved by minimizing
% the performance index in equation (3-30) for Algorithm 3 in ref. [2].

% This algorithm is a modified version of algorithm prsneted in ref. [1]. The
modified function is used to improve the
% stopband response characteristics of the synthesis filters, the analysis
filters and possibly the perfect reconstruction (PR).


% The optional weights are not used in the provided examples, so they are set
to (1)

% MATLAB Requirements: To run this function, the used MATLAB needs to have
the following toolboxes:
% 1- signal processing toolbox, 2- filter design toolbox
%
% ----------------------------------------------------------------
% Function inputs:
% 1- nk: an M-element vector which includes the set of sampling rates.
% 2- ratio: the frequency ratio occupied by each individual analysis filter.
%    e.g. 0.5 means half of the whole normalized frequency spectrum is
occupied by the filter.
% 3- transition_factor: a vector with the same length as the "ratio" vector,
and it is used to define
%    passband & stopband cut-off frequencies. Smaller transition_factor
results in sharper transition band.
%    Note: All elements in the "transition_factor" vector must be less than
half of the smallest element in the "ratio" vector.
%
%    Using the "ratio" vector and the "transition_factor" vector,
%    the analysis filters' frequency specifications are given as:
%    First filter- lowpass filter with the following normalized cut-off
frequencies:
%  a) passband cut-off frequency: [ratio(1)-transition_factor(1)]*pi
%  b) stopband cut-off frequency: [ratio(1)+transition_factor(1)]*pi
%    Last filter-  highpass filer with the following cut-off frequencies:
%  a) passband cut-off frequency: pi-[ratio(end)-transition_factor(end)]*pi
%  b) stopband cut-off frequency: pi-[ratio(end)+transition_factor(end)]*pi
%    Other filers- bandpass filters with the following cut-off frequencies:
%   a) first stopband cut-off frequency: [sum(ratio(1:i-1))-
transition_factor(i)]*pi
```

```
%   b) first passband cut-off frequency: [sum(ratio(1:i-
1))+transition_factor(i)]*pi
%   c) second passband cut-off frequency: [sum(ratio(1:i-1))+ratio(i)-
transition_factor(i)]*pi
%   d) second stopband cut-off frequency: [sum(ratio(1:i-
1))+ratio(i)+transition_factor(i)]*pi
%
% 4- L: filter length which is the same for both FIR analysis and synthesis
filters.
% 5- ro: frequency resolution, i.e. "ro" points with a uniform distribution
from 0 to pi (positive part of the
%    normalized frequencies) are used to compute the frequency response of
the analysis filters(see Eq.(29) in [1]).
% 6- iteration_max: Maximum number of iterations allowed by the optimization
algorithm (see "optimset" in Matlab).
% 7- function_evaluation_max: Maximum number of function evaluations allowed
by the optimization algorithm (see "optimset" in Matlab).
% 8- cost_function_weights: a 2-element vector consisting of optional weights
to compute cost function.
%    The first element is the weight for PR error (w1 in the performance
%    index defined above) and the second element is the weight for frequency
specification error (w2)
% 9- wf: a 2-element vector consisting of optional weights for the modified
objective function. These optional weights are used for the PR and MRSF.
% 10- initial_filters: Initial analysis filters for the optimization
algorithm which can be defined by user or automatically
%    generated by the function (set the parameter to "[]")
% -----------------------------------------------------------------
% Function outputs:
% 1- h: is a matrix in which each row is corresponding to one of the analysis
filters
%    e.g. if the sampling set (nk) is [2 4 4], the first row of h contains
the coefficients
%    of the analysis filter on the branch with subsampling of 2.
% 2- f: is a matrix in which each row is corresponding to one of the
synthesis filters
%    Note: "distortion" and "aliasing" are calculated for 512 frequencies
distributed uniformly over the normalized frequency [0 to pi]

% This MATLAB code presents the following after completion :
% - Computation time for each design
% - Maximum initial and final aliasing and distortion errors. These errors
are defined as the largest dB values over the normalized frequency [0 to pi]
% - Analysis and Synthesis filters' minimum stopband attenuation among all
filters measured in dB among all filters.
% - Analysis and Synthesis filters' largest passband ripple among all filters
measured in dB.

% ----------------------------- Examples ----------------------------------

% Example 1: Critically Sampled Uniform Filter Bank - Sampling set is {2,2}
% [h,f]=algorithm_3([2 2],[0.5 0.5],[0.1 0.1],64,64,50,80000,[1 1],[1 1],[]);

% Example 2: Critically Sampled Uniform Filter Bank - Sampling set is
{4,4,4,4}
% [h,f]=algorithm_3([4 4 4 4],[0.25 0.25 0.25 0.25],[0.08 0.08 0.08
0.08],56,64,50,80000,[1 1],[1 1],[]);
```

```
% Example 3: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,4}
% [h,f]=algorithm_3([2 4 4],[0.5 0.25 0.25],[0.1 0.1 0.1],48,60,50,80000,[1
1],[1 1],[]);

% Example 4: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,8,8}
% [h,f]=algorithm_3([2 4 8 8],[0.5 0.25 0.125 0.125],[0.09 0.08 0.062
0.062],76,100,100,80000,[1 1],[1 1],[]);


% Example 5: Incompatible Non-Uniform Filter Bank - Sampling set is {2,3,6}
% [h,f]=algorithm_3([2 3 6],[1/2 1/3 1/6],[0.08 0.08 0.06],64,512,50,80000,[1
1],[1 1],[]);


% Example 6: Incompatible Non-Uniform Filter Bank - Sampling set is {1,2, 4,
8, 8}
% [h,f]=algorithm_3([8 8 4 2 1],[0.0625 0.0625 0.125 0.25 0.5],[0.015 0.015
0.02 0.025 0.05],64,256,50,80000,[1 1],[1 1],[]);

% ----------------------------------------------------------------

% [1] Moazzen I., Agathoklis P., "A General Approach for Filter Bank Design
Using Optimization",
% Technical Report, 2014.
%
% Written By:   Iman Moazzen     2014
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%
% [2] ElGarewi A., Agathoklis P., "Analysis of Algorithms for Filter Bank
Design Optimization",
% MASc. thesis, 2019.

% Modified By:  Ahmed ElGarewi    May 30, 2018
%               Under supervision of Prof. Panajotis Agathoklis
%               Dept. of Electrical Engineering
%               University of Victoria
%*****************************************************************************
```

A.4 – Algorithm 4 Function's Call and Description

```
% [h,f]=algorithm_4(L,nk,ratio,transition_factor,ro_1,ro,iter);

% This function can be used to design all types of filter banks (critically
sampled uniform filter banks, compatible non-uniform filter banks, and
incompatible non-uniform filter banks).
%By using optimization based on the algorithm presented in ref. [1].

% Problem Description: Design K analysis and synthesis FIR filters so that
% the perfect reconstruction (PR) conditions are satisfied (or almost
satisfied). The perfect reconstruction conditions are formulated as a set of
linear equations.
% The analysis and synthesis filters' stopband and passband characteristics
are formulated as constraints for the performance indices in equation (3-33)
and (3-34) Algorithm 4 in ref. [1].

% MATLAB Requirements: To run this function, the used MATLAB needs to have
the following toolboxes:
% 1- signal processing toolbox, 2- filter design toolbox
%
% ---------------------------------------------------------------
% Function inputs:
% 1- N: filter length which is the same for both FIR analysis and synthesis
filters.
% 2- nk: an M-element vector which includes the set of sampling rates.
% 3- ratio: the frequency ratio occupied by each individual
analysis/synthesis filter.
%    e.g. 0.5 means half of the whole normalized frequency spectrum is
occupied by the filter.
% 4- transition_factor: a vector with the same length as the "ratio" vector.
It is used to define
%    passband & stopband cut-off frequencies. Smaller transition_factor
results in sharper transition band.
%    Note: Each element in the "transition_factor" vector must be less than
half of the smallest element
%    of the corresponding "ratio" vector entry.
%
%    Using the "ratio" vector and the "transition_factor" vector, the
analysis/synthesis filters' frequency
%    specifications are given as:
%    First filter- lowpass filter with the following normalized cut-off
frequencies:
%              a) passband cut-off frequency: [ratio(1)-
transition_factor(1)]*pi
%              b) stopband cut-off frequency:
[ratio(1)+transition_factor(1)]*pi
%    Last filter-  highpass filer with the following cut-off frequencies:
%              a) passband cut-off frequency: pi-[ratio(end)-
transition_factor(end)]*pi
%              b) stopband cut-off frequency: pi-
[ratio(end)+transition_factor(end)]*pi
%    Other filers- bandpass filters with the following cut-off frequencies:
%              a) first stopband cut-off frequency: [sum(ratio(1:i-1))-
transition_factor(i)]*pi
```

```
%               b) first passband cut-off frequency: [sum(ratio(1:i-
1))+transition_factor(i)]*pi
%               c) second passband cut-off frequency: [sum(ratio(1:i-
1))+ratio(i)-transition_factor(i)]*pi
%               d) second stopband cut-off frequency: [sum(ratio(1:i-
1))+ratio(i)+transition_factor(i)]*pi

% 5- ro_1: Passband frequency resolution, i.e. "ro" points with a uniform
distribution within the filters' passband-band area
%    are considered to compute the energy (see [1], Ch3.).

% 6- ro: Stopband frequency resolution, i.e. "ro" points with a uniform
distribution within the filters' stop-band area
%    are considered to compute the energy (see [1], Ch3.).

% 7- iter: Maximum number of iterations allowed.
% -----------------------------------------------------------------
% Function outputs:
% 1- h: is a matrix in which each row is corresponding to one of the analysis
filters
%    e.g. if the sampling set (nk) is [2 4 4], the first row of h contains
the coefficients
%    of the analysis filter on the branch with subsampling of 2.
% 2- f: is a matrix in which each row is corresponding to one of the
synthesis filters.

% Note: "distortion" and "aliasing" are calculated for 512 frequencies
distributed uniformly over the normalized frequency [0 to pi]

% This MATLAB code presents the following after completion:
% - Computation time for the design.
% - Maximum initial and final aliasing and distortion errors. These errors
are defined as the largest dB values over the normalized frequency [0 to pi].
% - Analysis and Synthesis filters' minimum stopband attenuation among all
filters measured in dB among all filters.
% - Analysis and Synthesis filters' largest passband ripple among all filters
measured in dB.

% ----------------------------- Examples ---------------------------------
%
% Example 1: Critically Sampled Uniform Filter Bank - Sampling set is {2,2}
% [h,f]=algorithm_4(64,[2 2],[0.5 0.5],[0.1 0.1],60,30,23);

% Example 2: Critically Sampled Uniform Filter Bank - Sampling set is
{4,4,4,4}
% [h,f]=algorithm_4(56,[4 4 4 4],[0.25 0.25 0.25 0.25],[0.08 0.08 0.08
0.08],60,30,10);
%
% Example 3: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,4}
% [h,f]=algorithm_4(48,[2 4 4],[0.5 0.25 0.25],[0.1 0.1 0.1],60,30,10);

% Example 4: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,8,8}
% [h,f]=algorithm_4(48,[2 4 8 8],[0.5 0.25 0.125 0.125],[0.09 0.08 0.062
0.062],100,50,10);

% Example 5: Incompatible Non-Uniform Filter Bank - Sampling set is {2,3,6}
```

```
% [h,f]=algorithm_4(64,[2 3 6],[1/2 1/3 1/6],[0.08 0.08 0.06],60,30,10);

% Notes -----------------------------------------------------------------

% [1] ElGarewi A., Agathoklis P., "Analysis of Algorithms for Filter Bank
Design Optimization",
% MASc. thesis, 2019. Ver 0.0.1

% Developed By: Ahmed ElGarewi    January 17, 2019
%                Under supervision of Prof. Panajotis Agathoklis
%                Dept. of Electrical Engineering
%                University of Victoria
```

A.5 – Algorithm 5 Function's Call and Description

```
% [h,f]=algorithm_4(L,nk,ratio,transition_factor,ro_1,ro,iter);

% This function can be used to design all types of filter banks (critically
sampled uniform filter banks, compatible non-uniform filter banks, and
incompatible non-uniform filter banks).
%By using optimization based on the algorithm presented in ref. [1].

% Problem Description: Design K analysis and synthesis FIR filters so that
% the perfect reconstruction (PR) conditions are satisfied (or almost
satisfied). The perfect reconstruction conditions are formulated as a set of
linear equations.
% The analysis and synthesis filters' stopband characteristics are formulated
as constraints for the performance indices in equations equation (3-35) and
(3-36) Algorithm 5 in ref. [1].

% MATLAB Requirements: To run this function, the used MATLAB needs to have
the following toolboxes:
% 1- signal processing toolbox, 2- filter design toolbox
%
% ---------------------------------------------------------------
% Function inputs:
% 1- N: filter length which is the same for both FIR analysis and synthesis
filters.
% 2- nk: an M-element vector which includes the set of sampling rates.
% 3- ratio: the frequency ratio occupied by each individual
analysis/synthesis filter.
%    e.g. 0.5 means half of the whole normalized frequency spectrum is
occupied by the filter.
% 4- transition_factor: a vector with the same length as the "ratio" vector.
It is used to define
%    passband & stopband cut-off frequencies. Smaller transition_factor
results in sharper transition band.
%    Note: Each element in the "transition_factor" vector must be less than
half of the smallest element
%    of the corresponding "ratio" vector entry.
%
%    Using the "ratio" vector and the "transition_factor" vector, the
analysis/synthesis filters' frequency
%    specifications are given as:
%    First filter- lowpass filter with the following normalized cut-off
frequencies:
%    a) passband cut-off frequency: [ratio(1)-transition_factor(1)]*pi
%    b) stopband cut-off frequency: [ratio(1)+transition_factor(1)]*pi
%    Last filter- highpass filer with the following cut-off frequencies:
%    a) passband cut-off frequency: pi-[ratio(end)-transition_factor(end)]*pi
%    b) stopband cut-off frequency: pi-[ratio(end)+transition_factor(end)]*pi
%    Other filers- bandpass filters with the following cut-off frequencies:
%    a) first stopband cut-off frequency: [sum(ratio(1:i-1))-
transition_factor(i)]*pi
%    b) first passband cut-off frequency: [sum(ratio(1:i-
1))+transition_factor(i)]*pi
%    c) second passband cut-off frequency: [sum(ratio(1:i-1))+ratio(i)-
transition_factor(i)]*pi
%    d) second stopband cut-off frequency: [sum(ratio(1:i-
1))+ratio(i)+transition_factor(i)]*pi
```

```
% 5- ro: Stopband frequency resolution, i.e. "ro" points with a uniform
distribution within the filters' stop-band area
%    are considered to compute the energy (see [1], Ch3.).

% 6- iter: Maximum number of iterations allowed.
% -----------------------------------------------------------------
% Function outputs:
% 1- h: is a matrix in which each row is corresponding to one of the analysis
filters
%    e.g. if the sampling set (nk) is [2 4 4], the first row of h contains
the coefficients
%    of the analysis filter on the branch with subsampling of 2.
% 2- f: is a matrix in which each row is corresponding to one of the
synthesis filters.

% Note: "distortion" and "aliasing" are calculated for 512 frequencies
distributed uniformly over the normalized frequency [0 to pi]

% This MATLAB code presents the following after completion:
% - Computation time for the design.
% - Maximum initial and final aliasing and distortion errors. These errors
are defined as the largest dB values over the normalized frequency [0 to pi].
% - Analysis and Synthesis filters' minimum stopband attenuation among all
filters measured in dB among all filters.
% - Analysis and Synthesis filters' largest passband ripple among all filters
measured in d.

% ----------------------------- Examples ---------------------------------
%
% Example 1: Critically Sampled Uniform Filter Bank - Sampling set is {2,2}
% [h,f]=algorithm_5(64,[2 2],[0.5 0.5],[0.1 0.1],60,50);

% Example 2: Critically Sampled Uniform Filter Bank - Sampling set is
{4,4,4,4}
% [h,f]=algorithm_5(56,[4 4 4 4],[0.25 0.25 0.25 0.25],[0.08 0.08 0.08
0.08],30,50);
%
% Example 3: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,4}
% [h,f]=algorithm_5(48,[2 4 4],[0.5 0.25 0.25],[0.1 0.1 0.1],60,50);

% Example 4: Compatible Non-Uniform Filter Bank - Sampling set is {2,4,8,8}
% [h,f]=algorithm_5(76,[2 4 8 8],[0.5 0.25 0.125 0.125],[0.09 0.08 0.062
0.062],100,85);

% Example 5: Incompatible Non-Uniform Filter Bank - Sampling set is {2,3,6}
% [h,f]=algorithm_5(64,[2 3 6],[1/2 1/3 1/6],[0.08 0.08 0.06],512,50);

% Example 6: Incompatible Non-Uniform Filter Bank - Sampling set is
{8,8,4,2,1}
% [h,f]=algorithm_5(128,[8 8 4 2 1],[0.0625 0.0625 0.125 0.25 0.5],[0.015
0.015 0.02 0.025 0.05],256,50);

% Notes -------------------------------------------------------------
```

```
% [1] ElGarewi A., Agathoklis P., "Analysis of Algorithms for Filter Bank
Design Optimization",
% MASc. thesis, 2019. Ver. 0.1.1

% Developed By: Ahmed ElGarewi    November 20, 2017
%                Under supervision of Prof. Panajotis Agathoklis
%                Dept. of Electrical Engineering
%                University of Victoria
%*****************************************************************************
```