Juuso Alhava
Unified Theory for Biorthogonal Modulated Filter Banks

Juuso Alhava

Unified Theory for Biorthogonal Modulated Filter Banks

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 5th of August 2016, at 12 noon.

Abstract

Modulated filter banks (MFBs) are practical signal decomposition tools for M-channel multi-rate systems. They combine high subfilter selectivity with efficient realization based on polyphase filters and block transforms. Consequently, the $O(M^2)$ burden of computations in a general filter bank (FB) is reduced to $O(M \log_2 M)$ - the latter being a complexity order comparable with the FFT-like transforms.

Often hiding from the plain sight, these versatile digital signal processing tools have important role in various professional and everyday life applications of information and communications technology, including audiovisual communications and media storage (e.g., audio codecs for low-energy music playback in portable devices, as well as communication waveform processing and channelization). The algorithmic efficiency implies low cost, small size, and extended battery life, bringing the devices close to our skins.

The main objective of this thesis is to formulate a generalized and unified approach to the MFBs, which includes, in addition to the deep theoretical background behind these banks, both their design by using appropriate optimization techniques and efficient algorithmic realizations. The FBs discussed in this thesis are discrete-time time-frequency decomposition/reconstruction, or equivalently, analysis-synthesis systems, where the subfilters are generated through modulation from either a single or two prototype filters. The perfect reconstruction (PR) property is a particularly important characteristics of the MFBs and this is the core theme of this thesis. In the presented biorthogonal arbitrary-delay exponentially modulated filter bank (EMFB), the PR property can be maintained also for complex-valued signals.

The EMFB concept is quite flexible, since it may respond to the various requirements given to a subband processing system: low-delay PR prototype design, subfilters having symmetric impulse responses, efficient algorithms, and the definition covers odd and even-stacked cosine-modulated FBs as special cases. Oversampling schemes for the subsignals prove out to be advantageous in subband processing problems requiring phase information about the localized frequency components. In addition, the MFBs have strong connections with the lapped transform (LT) theory, especially with the class of LTs grounded in parametric window functions.

Preface

The writing process of this thesis was long and arduous. The first crude version of the manuscript emerged ten years ago and has undergone many major revisions since. The goal was to formulate a consistent and unifying theory for the EMFB covering its variants and implementation algorithms. At hindsight, this might have been a slightly overambitious attempt causing certain drawbacks. But the journey has been truly interesting.

The seeds of this thesis were sown in the late 1990s at the former Telecommunications Laboratory, Tampere University of Technology (TUT), Finland. Since the early days the concepts and ideas have evolved into a new shape guided by the cumulative givings of the theory itself; thus the separation from the original multicarrier research. I had also the rare opportunity to practice independent research with the support of two grants from Jenny and Antti Wihuri foundation. The solitude enabled me to attune with something that could be called as researcher's intuition or inner guidance and for this freedom I present my gratitude.

This work was finally completed at Department of Electronics and Communications Engineering, TUT during 2012-2016 under the supervision and counseling of Prof. Markku Renfors. I appreciate the effort to chaperon this work to completion. I would also like to thank Prof. Tapio Saramäki for exposure to multirate-common-sense signal processing (or at least mostly common sense). I am also thankful to Dr. Henrique S. Malvar and Prof. Paulo S. R. Diniz who took the ungratifying task of reading the manuscript with great care. Their keen eyes and sharp wit did cause some anxiety and self-suspicions for the author. Nevertheless, the reviewers' constructive comments and suggestions greatly improved the text. As a focused and narrow-minded researcher/programmer, I was humbled by the dedication of the all mentioned scholars to convey and refine knowledge.

Let me also give my thanks to the personnel of the ELT who create this encouraging atmosphere. My special thanks goes to Dr. Juha Yli-Kaakinen, who served his sentence in the same cell, as well as to Dr. Ari Viholainen who regretfully has now a new master rather than our Alma Mater.

Tampere, May, 2016

Juuso Alhava

Contents

List of Notations

Mathematical Symbols

γ	discrete-time ELT window function parameter
α, β	MLBT window function parameters
$X_k[m]$	kth subsignal decompositon coefficients
$x[n]$	discrete-time signal
$h[n]$	discrete-time prototype filter/window function
$X(z)$	z-transform of sequence $x[n]$
$\overline{X}(z)$	z-transform of complex-conjugated sequence $x^*[n]$
ω	angular frequency
W_M	Mth root of unity
$X(e^{j\omega})$	frequency response of sequence $x[n]$
$|X(e^{j\omega})|$	amplitude response of sequence $x[n]$
$G_\ell(z)$	ℓth polyphase filter
$x_\ell[m]$	ℓth polyphase component of $x[n]$
$\boldsymbol{x}_{\mathrm{p}}(z)$	polyphase component vector of $X(z)$
$\boldsymbol{H}_{\mathrm{p}}(z), \boldsymbol{F}_{\mathrm{p}}(z)$	polyphase component matrix (analysis,synthesis)
L	overlap factor
M	decimation factor
N	filter order
D	arbitrary-delay parameter
ρ	roll-off parameter in prototype filter design
τ	analysis-synthesis reconstruction delay
$h_k[n], f_k[n]$	filter bank subfilters (analysis,synthesis)
$f_k^{\mathrm{e}}[n]$	kth synthesis subfilter with modulation identifier
ω_k	the center frequency of kth subfilter
$\delta[n]$	unit impulse response
$\boldsymbol{\delta}_C[\ell\,]$	non-circularly shifted identity matrix combined with unit impulse
$[\mathbf{A}]_{n,k}$	element on nth row and kth column of matrix $\mathbf{A}$
$\tilde{\mathbf{B}}, \mathbf{B}$	shift-biorthogonal LT matrices (forward,inverse)
$\mathbf{P}$	shift-orthogonal LT matrix
$\mathbf{I}_M$	identity matrix, optional subscript indicates dims. $M \times M$
$\mathbf{J}_M$	reversing matrix, optional subscript indicates dims. $M \times M$

x_n	element of a finite sequence
(x_n)	finite sequence
$(x_n)_{n=0}^R$	finite sequence with index limits
$\{0, 1, 2, ...\}$	a set with elements 0,1,2,...
$\cup$	binary union operator
$\in$	set membership
$\mathbb{Z}$	the set of integers
$\mathbb{N}$	the set of natural numbers
$n \bmod M$	modulus of n after division by M
$|y|$	absolute value of y
$\lfloor y \rfloor$	largest integer less than or equal to y
$\lceil y \rceil$	smallest integer greater than or equal to y

Abbreviations

2xOS	oversampled by a factor of two
AFB	analysis filter bank
CELT	complex extended lapped transform
CMFB	cosine-modulated filter bank
CMLBT	complex modulated lapped biorthogonal transform
CS	critically sampled
DCT	discrete cosine transform
DFT	discrete Fourier transform
DFT-FB	discrete Fourier transform filter bank
ELT	extended lapped transform
EMFB	exponentially modulated filter bank
FB	filter bank
FFT	fast Fourier transform
FIR	finite-impulse response
GDFT-FB	generalized discrete Fourier transform filter bank
IIR	infinite-impulse response
KLT	Karhunen-Loève transform
LCB	local cosine bases
LHS	left hand side
LOT	lapped orthogonal transform
LT	lapped transform
MCLT	modulated complex lapped transform
MDFT-FB	modified discrete Fourier transform filter bank
MFB	modulated filter bank
MLBT	modulated lapped biorthogonal transform
MLT	modulated lapped transform
OFDM	orthogonal frequency division multiplexing
OS	oversampled
PR	perfect-reconstruction
RHS	right hand side
SFB	synthesis filter bank
SMFB	sine-modulated filter bank
SRFFT	split-radix fast Fourier transform

Chapter 1

Introduction

1.1 Background and Motivation

Signal analysis can be described as a collection of methods for breaking mathematical objects "with many moving parts" into less complex components and then representing the signal with these elementary building blocks. Among the various decompositions we are mainly interested in a restricted category that consists of reversible operations. These kinds of transforms form analysis-synthesis pairs and can generally be called *invertible transforms* [21, 28, 58, 98]. The easiest way to find such a system is to systematically construct an orthogonal set of functions [22, 35] as the basis of the decomposition. Before actually doing so, it is in order to ask: "What is the nature of the information or hidden phenomena we want to extract from the signal by transforming it?" In practice the available means usually determine the answer a priori. For instance, one can choose a statistical approach [1] or decompose the signal with wavelets (time-scale analysis) [32] and see if anything groundbreaking happens. However, in the sequel, we restrict to methods for detecting the frequency components present in the signal at selected time instants, i.e., on *time-frequency analysis* [17].

The response of engineering community to research problems encountered in time-frequency analysis is focused on transform design [23, 40, 53, 64], implementation [22, 56, 76], and applications [73, 75, 90]. Some of the most prominent triumphs of multichannel signal processing systems can be found in communication systems (multicarrier modulation and adaptive signal enhancement) and in media storage/retrieval (e.g., low bit rate source coding and playback). In both of these cases, efficient algorithms provide reduced implementation costs and increased battery life and, consequently, facilitate portable devices with multimedia capabilities. These portable and wearable devices have even changed our personal interaction, behavior, norms, subcultures, and societies as a whole. There is no surprise that with such broad categories, the splintered bits of information may overwhelm the unwary, and an attempt to find similarities from all this knowledge is appropriate.

Thus one underlying theme in this thesis is to find a compact representation for the multitude of signal decomposition tools known as modulated filter banks (MFBs) [42, 57, 70, 93]. In pursuance of understanding this modern day *Bestiarum vocabulum*, it might be helpful to change ones mindset from continuous time functional analysis into discrete-time signals and systems. Particularly useful is to conceptualize the MFBs as multirate systems and apply the

z-transform domain identities for, say, sampling rate alteration and filtering [98]. Nevertheless, before we may proceed there, it is orderly to nod respectfully towards block transforms - the forerunners of MFBs.

Discrete sinusoidal transforms, such as discrete cosine transforms (DCTs) and the discrete Fourier transform (DFT) [22, 76], can be regarded as special cases of filter banks (FBs) when they are used in block-wise fashion [57]. In this configuration, the subfilters and the input sample sequence are of equal lengths. Due to a parametrically generated transform basis, there exist various options for implementation and the subsamples can be computed efficiently. However, the drawback is limited frequency selectivity [14, 37]. A single strong frequency component that is frequency localized between the center frequencies of two basis functions is "detected" in numerous subchannels. In addition, if the transform size is increased for better selectivity (consequently, also the input sample sequence length is longer), we lose some precision in the time localization of transient signal components.

One necessary property of the common block transforms is invertibility, i.e., we may recover the original signal from the set of transformed subsamples through an inverse transform. This is a crucial property in some signal processing applications, such as in lossless image compression and orthogonal frequency division multiplexing (OFDM). Without invertibility, the transform might be an interesting signal decomposition tool, but might not offer even approximate signal reconstruction.

One attribute that can be attached with block transforms is matrix symmetry (although it might mainly influence the transform algorithm implementation whilst the subband configuration and its relation with the signal autocorrelation property is more important). In the case of the Type IV DCT (DCT-IV) and the DFT, the inverse transform matrices are symmetric (former without conjugation) [76]. The DCT-II has, in turn, a non-symmetric inverse transform matrix [21]. In the previous cases, the forward transform is a (complex conjugate) transpose of the inverse matrix and they satisfy the invertibility condition meaning that the transforms are *orthogonal*. Further on, Type III DCT (DCT-III) forward-inverse transform is equivalent to DCT-II but in reverse order [21, 76]. The fast versions of these block transforms have also an important role in MFB algorithms.

In addition, related with the integer-valued (or finite-wordlength) realizations of the block transforms, the near-orthogonality becomes an issue which implies that the eigenvalues of the transform matrix are close to one. This prevents unwanted fluctuation in the dynamic range of the subsignals, which causes amplification of the noise (due to the quantization or from other sources) in the signal synthesis.

One issue in transform selection and design is the shape of the basis functions. Since the basis functions of the DFT and DCTs are truncated sinusoidals, the transition on the block edge is abrupt [57]. For instance, in lossy image compression this may cause reconstruction artifacts on block boundaries known as the blocking effect. This was one motivation for pioneering FB and transform researchers to develop better alternatives for block transforms.

A less common way to use the DFT and DCTs are sliding transforms [89] where new subsamples are computed for each new input sample. One suggested application can still be found in adaptive filtering textbooks where a forward DCT is applied to decorrelate the input signal to improve the adaptation speed of the least-mean-square algorithm. Such a scenario produces highly oversampled subsignals (when compared with critically sampled subband systems), but some reduction in the computational complexity can be achieved with recursive algorithms. In this thesis it is demonstrated that such sliding algorithms are not restricted to block transforms only.

One might argue what the driving force was, which caused the discrete transforms to enter a new era. At least the early works in continuous time-frequency analysis have been inspirational in this process. The concept itself, windowing sinusoidal waveforms to generate an analysis tool for detecting the local frequency components of the signal, can be traced back to D. Gabor [41]. He wrote already in 1946 the tenets of the time-frequency analysis:

> *But our everyday experiences... insist on a description in terms of both time and frequency. ... Signals are represented in two dimensions, with time and frequency as co-ordinates. Such two-dimensional representation can be called 'information diagrams' as areas in them are proportional to the number of independent data which they can convey.*

On these grounds, the short-time Fourier transform was developed and nowadays it is also known as the Gabor transform. Hindsightly, although Gabor originally studied the problem of the uncertainty of the time-frequency "atoms" (i.e., the detected elements on two-dimensional grid), his works predicted some research topics and applications, such as multicarrier communications systems, which are under active investigation nowadays.

Against this background, the discrete Fourier transform filter bank (DFT-FB), which was proposed considerably later, can be interpreted as being one digital version of the pioneering results by Gabor [30, 93]. In a DFT-FB, a low-pass prototype filter is modulated around new center frequencies to create a set of subfilters [30]. The mainlobes of the subfilters populate the whole frequency range in a uniform manner and the degree of mainlobe overlap and selectivity depend on the chosen or designed prototype. Since the modulation can be implemented efficiently with the fast Fourier transform (FFT) [22], the DFT-FB is well versed for applications like spectrum estimation and sensing. In this particular case, we need only the analysis side bank. The input sample sequence is windowed with the prototype filter, then "folded" and fed into the FFT. However, there is one drawback that limits its use if the transform is required to be invertible using critically sampled (CS) subsignals. This limitation is due to the fact that for non-ideal prototypes whose impulse response is of finite length, the perfect reconstruction (PR) is not feasible [90] as in the analysis-synthesis operation the signal is corrupted by a phenomenon called aliasing.

If we put a single subchannel under scrutiny in a DFT-FB, we observe that each subsignal is a result of bandpass filtering followed by downsampling. In downsampling the frequency axis "folds" and the frequency components merrily mingle in the lower rate signal [93]. If the vanilla DFT-FB is critically sampled, the unwanted aliased components generally can not be canceled in signal synthesis, except in the trivial case where the prototype filter is a rectangular window [90] thus reducing the system into block-wise operating DFT. Likely the first discrete time analysis-synthesis solution was discovered by J. H. Rothweiler [77]. The invention was to alter the subband configuration so that the 0th subfilter mainlobe was no longer symmetrically positioned around zero frequency. Furthermore, the modulation phases were tweaked and the resulting subfilters had non-symmetric impulse responses even though a symmetric prototype was utilized. This modulated subband system with quadrature mirror filters can be considered as the first cosine-modulated filter bank (CMFB).

Few years after this discovery, it was realized that the PR with symmetric subfilters can also be arranged but it requires alternate use of sine- and cosine-modulated sequences when computing the subsamples [72]. This time domain aliasing cancellation FB is then a time varying subband system. From this paper originates our distinction between odd- and even-stacked subfilters, which refers to alternative subchannelization models.

The early work of modulated PR systems was carried out with short parametric window functions as prototypes. The smallest integer overlap in "block sampling" is realized when the subfilter length is twice the number of subchannels (assuming an odd-stacked configuration). Such a playground is quite small for transform design with optimization methods if high frequency selectivity is the goal. After these initial findings, the immediately interesting research problem was to increase the transform overlap with longer subfilters whilst maintaining the invertibility. From numerous proposals for transform design and implementation, two independently written papers had high impact: Malvar's extended lapped transform (ELT) [56] and the paraunitary CMFB by Koilpillai and Vaidyanathan [48]. Both of these are grounded on the realization of the prototype filter with orthogonal rotations which guarantees an inherently PR system. The notable feature of these two paraunitary CMFBs is that the analysis-synthesis reconstruction delay is fixed once the subfilter length (or the number of rotations in a cascade) is selected. Biorthogonal MFBs form a more general classification that includes also the paraunitary designs.

Biorthogonality is generally associated with two sets of mathematical entities that together satisfy an interesting case-dependent property. In this case, the paraunitary and biorthogonal analysis-synthesis systems share the PR property, while in the biorthogonal case there are more degrees of freedom in (jointly) designing the analysis and synthesis banks. (The attached FB labels receive a more formal treatment in Sections 2.6.2 and 2.6.3.) Typically, there are two ways to construct such PR subband systems: the use of either different prototypes for the analysis and synthesis stages [58] or a single non-symmetric prototype to create a low-delay system [42]. In the latter case, the modulating sequences cause the set of subfilters to be biorthogonal. Mapping a biorthogonal MFB into a structurally PR system is possible by implementing the prototype filter with a cascade of lifting steps having polynomial coefficients [47].

CMFBs are subband processing tools for real-valued signals. The extension of the MFB theory into the complex domain just requires switching the modulation sequence to complex exponential. The even-stacked version is called the modified discrete Fourier transform filter bank (MDFT-FB) [46] and odd-stacked the exponentially modulated filter bank (EMFB) [5]. In both cases, the decimation factor is half the number of subbands, but the PR with critical subsampling is arranged by selecting either real or imaginary part of the subsamples for processing. In a way, both of these complex FBs can be regarded as implementations of multicarrier modulation methods proposed by Chang [26] and Saltzberg [78] several decades earlier - only more recently presented in the analysis-synthesis configuration.

1.2 Author's Contribution

1.2.1 ELT-Based Exponentially Modulated Filter Bank

For several decades, the OFDM with its variants was the de facto multicarrier modulation method in radio environment. However, one problem in OFDM multiuser systems dynamically sharing the available bandwidth was fitting into the spectral masks, and thus avoiding interference between users. Secondly, the cyclic prefix (whose length depends on the expected channel delay spread) inserted for equalization purposes reduces the capacity. One alternative considered in the class of FB-based transmultiplexers was the CMFB in the reverse (synthesis-analysis) configuration. Nevertheless, being a real-valued subband system it was intended mainly for digital subscriber lines [37].

Our contribution to the MFB theory originated from a research problem: How to construct an efficient FB-based multicarrier communication system with complex (I/Q) baseband signal model? The "available" component was fast ELT-based M-channel CMFB, having synthesis subfilter definition

$$f_k^c[n] = 2\,h[n]\,\cos\left[\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],\qquad(1.1)$$

where $h[n]$ is the low-pass prototype filter and the subfilter superscript "c" refers to the modulation type[†]. The CMFB alone can be used to generate numerous orthogonal subcarriers in digital subscriber line type of channels, in which case carrier modulation is not needed. However, if the CMFB-based transmultiplexer is applied for radio communications, the modulated signal bandwidth becomes twice the bandwidth that is sufficient for spectrally efficient multicarrier systems. The modulation sequence must thus be replaced with a complex exponential sequence in order to construct a complex-valued transmultiplexer/MFB having single-sided subcarrier spectra. We demonstrated wild imagination and called this FB as the exponentially modulated filter bank.

The idea was crudely sketched in [4]: A companion for the CMFB is the sine-modulated filter bank (SMFB). The synthesis subfilters of the SMFB, $f_k^s[n]$, are generated form the same prototype filter. Further on, the SMFB impulse responses are given by

$$f_k^s[n] = 2\,h[n]\,\sin\left[\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right].\qquad(1.2)$$

The resulting M-channel SMFB is paraunitary and we showed in [5] that the PR holds for this FB as well. In addition, these two real FBs together compose a subband processing system shown in Figure 1.1, where the input signal is complex-valued and it is separated to real and imaginary parts: $x[n] = \mathrm{Re}\{x[n]\} + j\,\mathrm{Im}\{x[n]\}$. By looking at this flowgraph we recognize that if both filter bank components - namely the cosine- and sine-modulated FBs - satisfy the perfect-reconstruction condition then the system as a whole is a PR filter bank. This is because the cross-connection lattice on the synthesis side is the inverse operation for the lattice on the analysis side. In addition, we note that the system is critically sampled as there is no sample rate growth in subbands. Vice versa, if the subband system is applied as a transmultiplexer (synthesis-analysis) configuration, the subcarrier symbol alphabet must be real-valued to avoid intercarrier and intersymbol interference.

How does it fit into the classification scheme for filter banks? Similar to the above discussion on the CMFBs and MDFT-FBs, we can specify the following attractive properties: *The ELT-based EMFB is a critically sampled, paraunitary, complex-valued, and odd-stacked filter bank satisfying the perfect-reconstruction condition. When the input signal is complex-valued, the analysis EMFB generates $2M$ purely real-valued subsignals where M is the decimation factor.* In this case of the complex-valued input signal, the impulse responses of the subfilters are defined as

$$f_k^e[n] = h[n]\,\exp\left[j\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],\qquad(1.3)$$

where $k = 0, 1, \ldots, 2M - 1$ and $n = 0, 1, \ldots, LM - 1$.

[†]In the sequel, the superscripts are used to identify unambiguously among all MFBs of this thesis the MFB being in use. For this purpose, Table 2.2 is recommended for a quick reference for both MFBs under discussion and their corresponding superscripts.
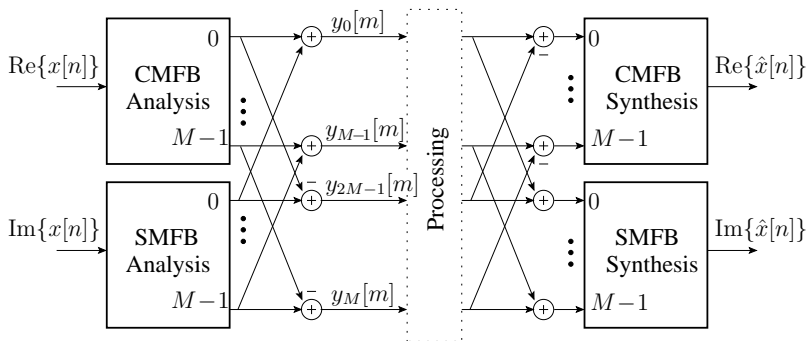
Figure 1.1: ELT-based exponentially modulated filter bank.

1.2.2 The Next Logical Step: Generalize

All that remains is to generalize the EMFB concept and fill the missing gaps. The inclusion of the arbitrary-delay parameter to the EMFB definition leads to biorthogonal subfilters. The perfect-reconstruction condition for the non-symmetric prototype filter is given in [7] and it can be applied also with CMFBs. Biorthogonal EMFB theory is formulated in [8] and implementation algorithms for the presented filter banks are presented in Chapter 7. Computationally efficient algorithms are based on polyphase filters and FFT modulation blocks. As a special case, the paraunitary ELT-based EMFB can be implemented with butterfly lattices [11]. The odd-stacked paraunitary EMFB, in turn, can be frequency shifted into an even-stacked PR filter bank where the phase of the modulating sequence is chosen so that the subfilters have the linear phase with a symmetric prototype [8].

The design of the prototype filters is a non-linear constrained optimization problem with an optional nearly zero DC-leakage requirement. These design issues have been considered in [7] using ℓ_2-norm for the objective function. The ℓ_∞-norm is covered in Chapter 5. Low-delay prototype filters find applications, e.g., in transmultiplexer systems [6] as the non-symmetric prototype reduces a reconstruction lag while maintaining good frequency selectivity. In [3] we have applied oversampling (parallel CMFB and SMFB) for subsignal equalization in the digital subscriber line transmultiplexer setup. The lifting-lattice-based integer-to-integer realization of the odd-stacked EMFB which preserves the PR property has been covered in [10]. Finally, as we can consider the window function generated modulated lapped transforms as a special case of MFBs, we may construct a family of shift-orthogonal lapped transforms. For the simplest parametric window functions we may generate recursive algorithms [9] which implement highly oversampled PR subband systems with recursive filters. The proposed design and implementation algorithms are numerically verified in an LT-toolbox for MATLAB [2].

The author is the primary writer of publications [3-10] under the guidance of Prof. Markku Renfors. The contribution of Dr. Tech. Ari Viholainen made the algorithm experiments possible and led to [11]. However, this is an understatement - the fruitful discussions, tips, and suggestions by my fellow researchers were invaluable for this work.

1.3 Outline of Thesis

The above introductory modulated filter bank discussion served as a prelude to more technical and detailed analysis on discrete-time subband systems. Chapter 2 lists the essential multirate elements, concepts, and notations. These are then linked with the discrete-time lapped transforms in Chapter 3. Chapter 4 starts with a crash course to biorthogonal odd-stacked CMFBs and presents the PR condition for the arbitrary-delay prototype filter using polyphase components. Then we will show that assuming the arbitrary-delay CMFB to be a perfect-reconstruction FB, then the same holds for the odd-stacked biorthogonal EMFB. The EMFB concept is explored further; the variations of the EMFB theme produce a number of filter banks that are characterized by three options:

- Critical sampling *or* 2 times oversampling (2xOS)

- Real-valued *or* complex-valued input signal

- Odd-stacking *or* even-stacking.

Chapter 5 brings the theory closer to praxis: Selected design examples numerically verify the modulation variants and the PR constraints are tweaked to cover arbitrary-length prototypes. The lapped transforms are revisited in Chapter 6 where the EMFB modulation is combined with the LT window functions to generate a family of modulated lapped transforms. Implementation algorithms are discussed in Chapters 6 and 7. First, we contribute the matter with a sliding lapped transform algorithm that can be applied to compute an oversampled non-decimated modulated lapped transform. Then, the block-transform-based algorithms are provided for the biorthogonal MFBs discussed in Chapter 4. It is also shown how the Goertzel-DFT can be applied in per-subband EMFB algorithm. The grande finale (or *lacrimosa dies illa*) of the MFB theme is presented in Chapter 8. The chapter starts with intuitive modifications to the ELT algorithm that converts it to the even-stacked CMFB. The last detail is the verification of the even-stacked ELT algorithm (which might cause an overdose of polyphase notation for the unwary). The biorthogonal versions of the EMFB algorithms in Chapter 7 as well as the even-stacked fast ELT in Chapter 8 are published for the first time in this thesis.

The target is now on sight and the rest is blood, sweat, and clichés.

Chapter 2

Essentials of Multirate Filter Banks

The basic operations in sampling rate alteration are downsampling and upsampling. Usually, these functions are accompanied with a spectrum shaping filter that selects the desired component of the signal. Together they provide methods for decimation and interpolation. Actually, a critically sampled filter bank is constructed from these components. We start by reviewing the properties of these discrete-time functions and then give a spectral interpretation how the downsampling shapes the spectra of bandpass signals. The introductory presentation is not rigorous and its purpose is to highlight some aspects on the sampling of complex-valued signals. In addition, this section presents those well-known identities for down/upsampling and polyphase forms that are useful in the analysis of the multirate systems. Then we turn our attention into filter banks and introduce the key terms that are necessary when we discuss various types of FBs. The focus of this thesis is centered on critically sampled uniform FBs and thus they receive a more accurate mathematical treatment. Perfect-reconstruction conditions for such subband systems are given in both z-transform and polyphase domains. The invaluable sources for the analysis of multirate systems are [57, 93, 98].

2.1 Multirate Operations with Filter Banks

2.1.1 Downsampling and Upsampling

In downsampling by M operation we take every Mth sample of the discrete-time input signal $x[n]$ and discard the rest [30]. Clearly this is not a time invariant operation as there are M possible phases to choose from. The output $y[m]$ is a lower rate signal and this is emphasized with a different time index m. Among the M options, we write, according to the most convenient option, the time domain operation for decimation as

$$y[m] = x[mM] \tag{2.1}$$
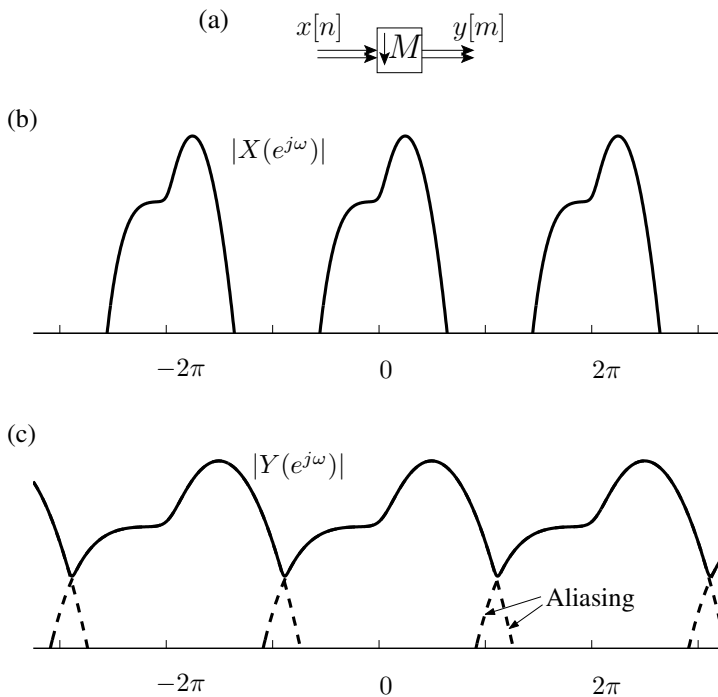
Figure 2.1: (a) Symbol for downsampling by M. (b) Spectrum of the complex-valued input signal before downsampling. (c) Spectrum of the complex-valued output signal after downsampling by 2. The signal is not properly bandlimited and there is aliasing near $\pm\pi$.

and the symbol for downsampler is shown in Figure 2.1[†]. The downsampling operation of equation (2.1) corresponds to a z-transform domain relationship

$$Y(z) = \frac{1}{M} \sum_{i=0}^{M-1} X(z^{1/M} W_M^{-i}) \tag{2.2}$$

where $W_M = \exp\{-j\, 2\pi/M\}$. By substituting $z = e^{j\omega}$ we notice how the spectrum is stretched by the decimation and there are also M frequency domain shifted versions of the spectrum. The spectral overlap of these replicas is called *aliasing*. This means that the input signal can not be recovered from the $y[m]$ because the aliased frequencies will not unfold correctly in the interpolation. Usually, the signal is bandlimited before downsampling to reduce this phenomenon and the filter for this purpose is called the *anti-aliasing* filter. Typically, the decimated signal is a result of low-pass filtering and downsampling. However, in M-channel filter banks the majority of subfilters select a bandpass portion of the spectrum.

Figure 2.1 (b) and (c) exemplifies downsampling by two. In this example, the input signal is complex-valued as can be seen from the non-symmetric 2π-periodic[‡] amplitude spectrum

---

[†]Note: In some flowgraphs, the real- and complex-valued signals are distinguished for clarity with single and double lines, respectively.

[‡]We usually display the spectra of the signals in range $\omega \in [-\pi, \pi]$ or only the positive frequency side $\omega \in [0, \pi]$ when the signals are real-valued.
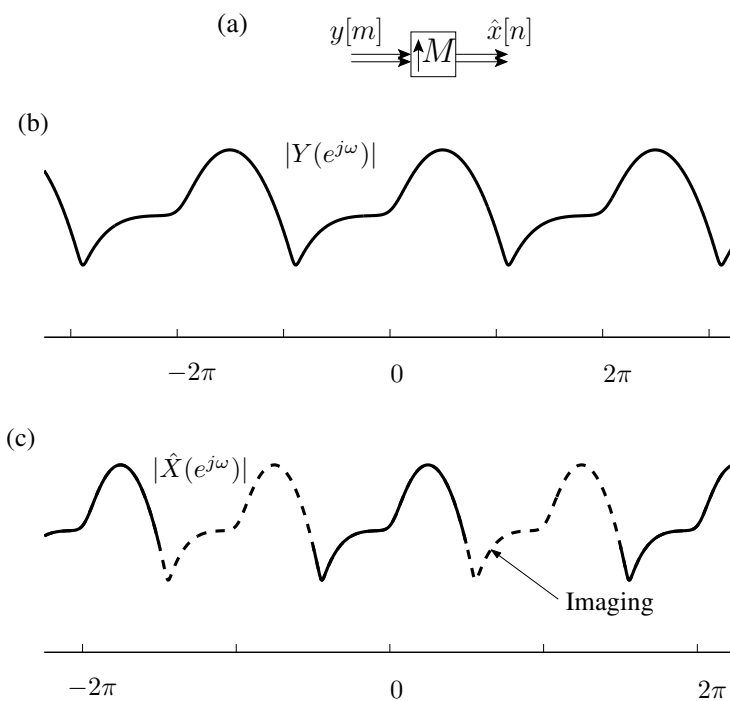
Figure 2.2: (a) Symbol for upsampling by M. (b) Spectrum of the complex-valued input signal that is the output signal shown in Figure 2.1 (c). (c) Spectrum of the complex-valued output signal after upsampling by 2. The imaging part of the spectrum is drawn with dashed lines.

$|X(e^{j\omega})|$. The spectrum of the input signal is not contained within $[-\pi/2, \pi/2]$ and thus the tails of the spectrum fold to wrong frequency locations. The spectrum $|Y(e^{j\omega})|$ of the output signal is stretched and aliasing has reshaped the spectrum near $\pm\pi$.

Upsampling by M operation is required in the interpolation of an input signal $y[m]$ where the sample rate of the output signal $\hat{x}[n]$ is increased by a factor of M. In multirate signal processing, the plain upsampler inserts $M-1$ zeros between the existing input samples. This operation can be formally defined as

$$\hat{x}[n] = \begin{cases} y[n/M] & \text{if } n \bmod M = 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

and the upsampler symbol is shown in Figure 2.2 (a). The corresponding z-transform relation is

$$\hat{X}(z) = Y(z^M). \tag{2.4}$$

Again we shift to the frequency domain via substitution: $\hat{X}(e^{j\omega}) = Y(e^{jM\omega})$. As can be seen, the upsampling operation shrinks the spectrum of $Y(e^{j\omega})$ by a factor of M. Within the frequency range $[-\pi, \pi]$ there is now the contracted original spectrum but in addition multiple replicas of the image spectra. To avoid this *imaging* the output signal can be filtered with a low or bandpass filter that chooses the desired part of the spectrum. This filter is called the *anti-imaging filter*. The output of this filter is the interpolated signal.

11

We continue with the amplitude spectrum example; Figure 2.2 (b) and (c) illustrates up-sampling by two in case where the spectrum of the complex-valued input signal $Y(e^{j\omega})$ in Figure 2.2 (a) is that of the output signal in Figure 2.1 (b). The spectrum of the output $\hat{X}(e^{j\omega})$ is given in Figure 2.2 (c) when the upsampling factor is two. The operation shrinks the spectrum to half and creates the image spectra. By comparing the amplitude spectra $|X(e^{j\omega})|$ and $|\hat{X}(e^{j\omega})|$ in Figures 2.1 (b) and 2.2 (c), respectively, we get an intuition why the input signal can not be recovered because there was aliasing in the downsampling. Even if we use an ideal anti-imaging filter, there will be dissimilarities near $\pm\pi/2$ between the two.

2.2 Decimation of Passband Signals

The decimation of passband signals requires that the spectrum of the signal is bandlimited within certain bounds to avoid the aliasing, i.e., the folding of frequency components on top of each other. However, there are a few differences when compared with the decimation of baseband signals that depend on how we choose the passband edges of the anti-aliasing filters. For simplicity, we assume the use of ideal "brickwall" filters whose center frequencies are selected to match the PR MFBs considered in this thesis, in particular, CMFBs and EMFBs in an odd-stacked configuration. Incidentally speaking, the downsampling of bandlimited complex-valued signals is not so straightforward as it seems from the first sight. We also suggest viewing a downsampling example in PhD thesis by Ari Viholainen [99] which starts on page 12. In this example, the complex-valued signal is properly bandlimited so that the shape of the spectrum is preserved in downsampling by introducing an appropriate frequency shift that brings non-adjacent frequency components next to each other in the lower rate complex-valued signal. In addition, a thorough treatise on general bandpass sampling theory can be found in [91, 96].

It is emphasized that the examples are selected to provide a different glimpse for subband system analysis before the z-transform takes over. For instance, the MFB PR derivations in Chapter 4 are represented using polyphase domain notation, which effectively hides the frequency domain interpretation.

Passband Decimation of Real Signals

In this first example real input signal $x[n]$ is filtered with anti-aliasing filter $H_k(z)$ and the downsampled output signal is denoted with $y[m]$ as shown in Figure 2.3 (a). When we view this example in frequency domain the signal has a conjugate symmetric spectrum with respect to zero frequency and an example amplitude spectrum is shown in Figure 2.3 (b). The bandlimiting is done with an ideal filter and its frequency response is

$$H_k(e^{j\omega}) = \begin{cases} 1 & \frac{k\pi}{M} \leq |\omega| < \frac{(k+1)\pi}{M} \\ 0 & \text{otherwise,} \end{cases} \tag{2.5}$$

where index k can have values $k = 0, 1, \ldots, M - 1$. The passband of the filter is on the positive and negative sides of the frequency axis, $\omega \in [-\pi, \pi]$. Depending on the parity of k the spectrum of the decimated signal is either on the natural frequency ordering or inverted. This is demonstrated in Figure 2.3 (c) and (d). When the filter index is even, we cut a part of the spectrum with $H_k(e^{j\omega})$ and decimate the filtered signal, the frequency components are in the same order as in $|X(e^{j\omega})|$. Inn the odd index case, in turn, the spectrum of the downsampled signal is flipped when compared with the original.

(a)

$x[n] \rightarrow \boxed{H_k(z)} \rightarrow \boxed{\downarrow M} \rightarrow y[m]$

(b)

$|X(e^{j\omega})|$

$-\frac{(k+1)\pi}{M} \quad -\frac{k\pi}{M} \qquad 0 \qquad \frac{k\pi}{M} \quad \frac{(k+1)\pi}{M}$

(c) Subfilter index even

$|Y_k(e^{j\omega})|$

$-\pi \qquad 0 \qquad \pi$

(d) Subfilter index odd

$|Y_{k+1}(e^{j\omega})|$

$-\pi \qquad 0 \qquad \pi$
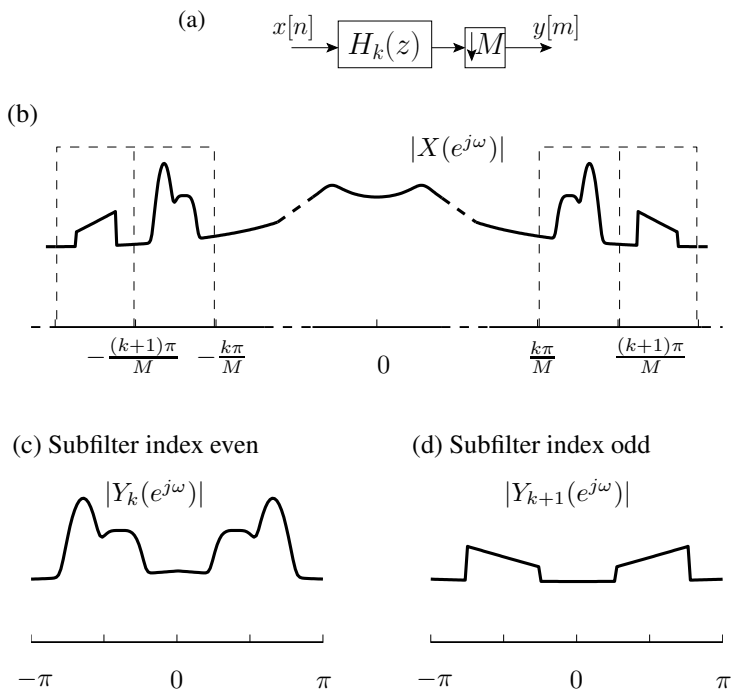
Figure 2.3: (a) Real-valued signal is bandlimited with $H_k(z)$ before downsampling. (b) Spectrum of the original signal and the ideal passbands of two anti-aliasing filters are shown with dashed lines. (c) Spectrum after decimation, subfilter index even. (d) Spectrum after decimation, subfilter index odd. Frequency components are in reverse order.

Passband Decimation of Complex-Valued Signals

The following example is the ideal decimation scheme of the EMFB [8], where $x[n]$ is assumed to be complex-valued signal. In Figure 2.4 (a), a complex anti-aliasing filter bandlimits the signal before downsampling and we take the real part of the lower rate signal. The difference from the previous case is that the amplitude spectrum of the signal is not symmetric and the used anti-aliasing filter is one-sided in Figure 2.4 (b). The subfigures (c) and (d) show the one-sided spectra of the (sub)signals directly after decimation. The parity of the filter index determines how the spectrum "behaves" in the decimation, i.e., the downsampled signal spectrum is single-sided. Nevertheless, the real part operation completes the other side of the spectrum as shown in subfigures (e) and (f), but there is the same frequency reversion as in the real case above.

If we do this kind of signal decomposition as in Figure 2.4, we may ask: Is the signal reconstruction possible from the real parts of the downsampled subsignals $Y_k(e^{j\omega})$ or is there information lost when we discard the imaginary parts of the subsignals? Intuitively, the signal recovery seems likely because the signal is bandlimited before downsampling and interpolation with one-sided anti-imaging filtering can produce the original signal from the decomposed subsignals. We address this question with the EMFB PR proof in Chapter 4.

13
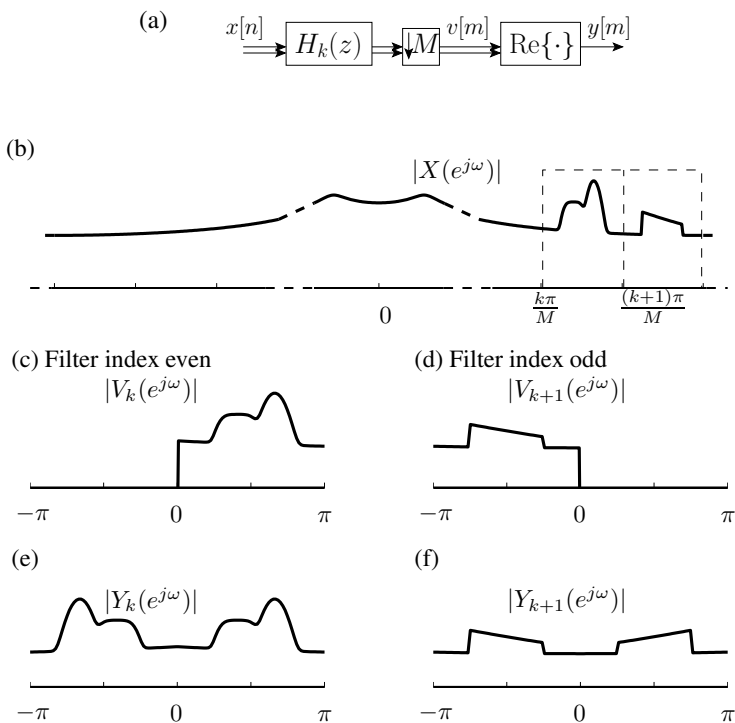
Figure 2.4: (a) Complex-valued signal is bandlimited with $H_k(z)$ before downsampling. (b) Spectrum of the original signal and the ideal one-sided passbands of two anti-aliasing filter are shown with dashed lines. (c) Spectrum after decimation, filter index even. (d) Spectrum after decimation, filter index odd. (e) and (f) Spectra of the signals after taking the real part.

Passband Decimation of Complex-Valued Signals with Real Filters

The third case is a downsampling example where a two-sided (real) filter is used to bandlimit a complex-valued passband signal. The downsampling preserves those parts of input spectrum that are cut with the ideal filters but there is a disadvantage: The subsignal is composed of frequency components from two different parts of the original spectrum. This kind of situation is illustrated in Figure 2.5 (c) and (d) where the high-rate signal has a non-symmetric spectrum with respect to zero frequency. Let us consider a possibility of filtering the complex-valued signal with this kind of real filter bank. This would combine non-adjacent parts of the input signal spectrum into one subsignal. The subband processing algorithm has to split the subsignal spectrum into positive and negative sides in order to process the frequency components independently. This operation can be avoided if we have complex-valued anti-aliasing subfilters with one-sided mainlobes and, alas, we have found one important motivation for developing a variety of analysis methods that suit best for given purposes.
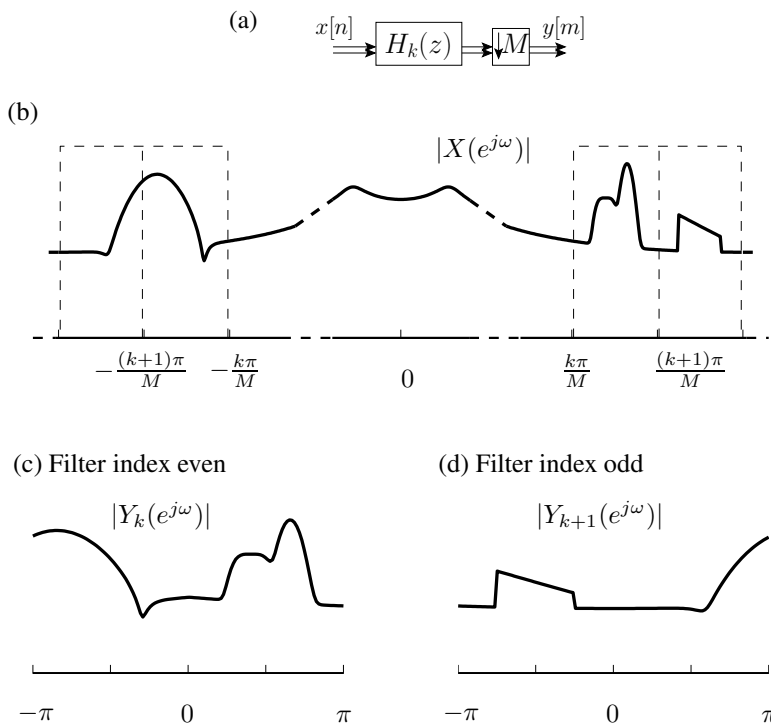
Figure 2.5: (a) Complex-valued signal is bandlimited with real $H_k(z)$ before downsampling. (b) Spectrum of the original signal and the amplitude responses of two ideal real anti-aliasing filters are shown with dashed lines. (c) and (d) Downsampling brings together frequency components from different parts of spectrum.

2.3 Signal Polyphase Components

In the polyphase decomposition of discrete-time signal $x[n]$, it is broken into M phases according to the time index as follows

$$x_\ell[m] = x[\ell + mM], \qquad \ell = 0, 1, \ldots, M-1,$$

where the sequence $x_\ell[m]$ is called the ℓth *polyphase component*. Figure 2.6 shows a causal polyphase transform for the signal that utilizes the downsamplers and upsamplers for this purpose. The forward transform is simply a serial-to-parallel conversion of the input signal in such a way that the polyphase components are expressed in the z-transform domain as

$$X_\ell(z) = \sum_{m=-\infty}^{\infty} x[\ell + mM] \, z^{-m}, \tag{2.6}$$

which excludes the delay caused by input buffering. The inverse transform then reconstructs the signal from its polyphase components by interleaving the subsequences, i.e., parallel-to-serial conversion. The forward-inverse polyphase transform pair of Figure 2.6 yields the fol-
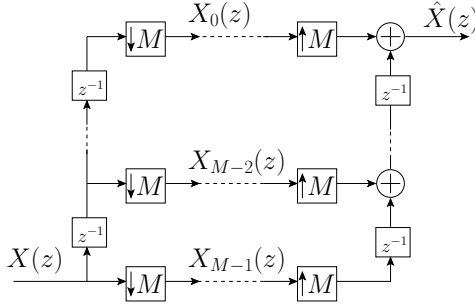
Figure 2.6: Signal decomposition to M polyphase components and its reconstruction with interleaving these components.

lowing output being the input delayed by $M - 1$ samples:

$$\hat{X}(z) = z^{-(M-1)} \sum_{\ell=0}^{M-1} z^{-\ell} X_\ell(z^M) = z^{-(M-1)} \sum_{\ell=0}^{M-1} z^{-\ell} \left(\sum_{m=-\infty}^{\infty} x[\ell + mM] z^{-mM} \right)$$

$$= z^{-(M-1)} \sum_{m=-\infty}^{\infty} \sum_{\ell=0}^{M-1} x[\ell + mM] z^{-(\ell+mM)} = z^{-(M-1)} X(z).$$

(2.7)

In the following, we use the convenient vector notation for the polyphase decomposition. The *polyphase component vector* $\boldsymbol{x}_{\mathrm{p}}(z)$ of $X(z)$ is defined as

$$\boldsymbol{x}_{\mathrm{p}}(z) = \begin{bmatrix} X_0(z) & X_1(z) & \cdots & X_{M-1}(z) \end{bmatrix}^T.$$

We can then write the output signal with vector dot product

$$\hat{X}(z) = z^{-(M-1)} \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-(M-1)} \end{bmatrix} \boldsymbol{x}_{\mathrm{p}}(z^M) = z^{-(M-1)} X(z). \qquad (2.8)$$

This expression for polyphase decomposition/reconstruction is particularly useful in the polyphase domain analysis of filter banks.

2.4 Down/Upsampling Identities and Polyphase Decomposition for FIR filters

Two useful relations in the analysis of the multirate systems are identities for sampling rate alteration and polyphase forms of the filters. The downsampler and upsampler cascaded with filter $H(z)$ have identities as shown in Figure 2.7. The first (second) subfigure indicates that if in the decimation (interpolation) case downsampling (upsampling) by M is moved after (before) the filter transfer function $H(z)$, then the identical overall operation is achieved by replacing $H(z)$ with $H(z^M)$. These relations are referred to as *noble identities* [30, 92].

Writing the filter responses with a polyphase representation is a convenient way to analyze multirate systems. Let us consider a causal FIR filter $h[n]$ of order N cascaded with downsampling by M. The z-transform $H(z)$ for the finite impulse response sequence is defined
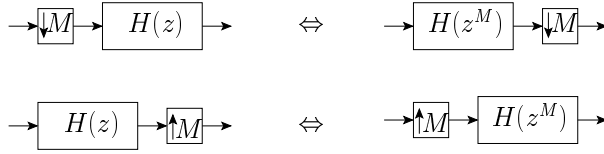
Figure 2.7: Identities for decimation and interpolation.

as

$$H(z) = \sum_{n=0}^{N} h[n] z^{-n}.$$

The polyphase decomposition partitions $H(z)$ into M phases as

$$H(z) = \sum_{\ell=0}^{M-1} z^{-(M-1-\ell)} \tilde{G}_\ell(z^M), \qquad (2.9)$$

where the z-transform of the ℓth polyphase filter is

$$\tilde{G}_\ell(z) = \sum_{m=0}^{L_\ell-1} h[mM + M - 1 - \ell] \, z^{-m}.$$

For arbitrary-length filters $N + 1$ is not divisible by M. We take into account the different length polyphase components by selecting

$$L_\ell = \lfloor (N + 1 + \ell)/M \rfloor.$$

Here, $\lfloor \cdot \rfloor$ is the floor operation. Figure 2.8 (a) shows the polyphase decomposition of filter $H(z)$ combined with the downsampling operation. The first change is to replace $H(z)$ with its polyphase filters. On the right is the final form where the identity from Figure 2.7 is applied and the downsampling operation can be moved before the polyphase filters. Within the scope of this thesis, the polyphase filter representation is useful for studying the PR conditions of filter banks and developing block-transform-based algorithms.

In a similar fashion, the identities for the upsampling operation followed by filter $F(z)$ are given in Figure 2.8 (b). The polyphase decomposition can be written as

$$F(z) = \sum_{\ell=0}^{M-1} z^{-\ell} G_\ell(z^M), \qquad (2.10)$$

where the ℓth polyphase filter is $G_\ell(z) = \sum_{m=0}^{L_\ell-1} f[mM + \ell] \, z^{-m}$ and $L_\ell = \lfloor (N + M - \ell)/M \rfloor$. Figure 2.8 (b) represents this set of polyphase filters. As the upsampling operator is moved next to the output delay line by applying its identity forms, we notice that it corresponds to sample interleaving, a parallel-to-serial conversion. Moreover, positioning the upsampler after the polyphase filters reduces the filtering sample rate by a factor of M.

In the continuation, we drop the summation limits for m in these polyphase equations when they are applied with matrices defining multirate subband systems. The selection of
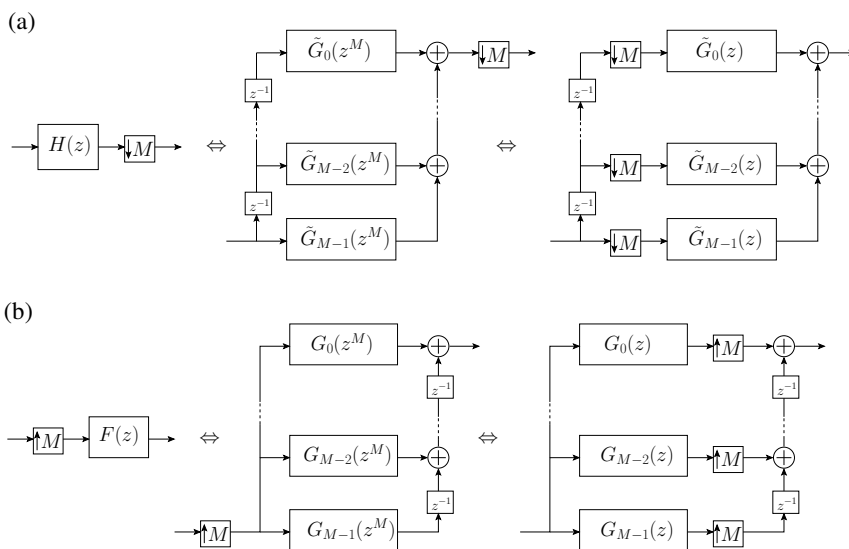
Figure 2.8: (a) Polyphase representation for filter $H(z)$ and decimator for reducing the sampling rate. (b) Polyphase representation for interpolator and filter $F(z)$ for increasing the sample rate.

$N + 1 \bmod M = 0$ is an exception to this, because in this case $N + 1 = LM$ $(L \in \mathbb{N})^{\dagger}$. This results in all polyphase components to have the same length, and the upper sum limit for both $\tilde{G}_\ell(z)$ and $G_\ell(z)$ becomes $L - 1$.

2.5 Filter Banks

Filter banks are signal processing systems that, first, decompose the signal to spectral components, second, manipulate these subsequences and, third, reconstruct the signal(s) in the original domain using the inverse operation. Schematically, this filter bank in the general case consists of the following elements:

1. *Analysis filter bank* (AFB): In the AFB, the input signal is decomposed to subsignals. Alternatively, this can be viewed as a signal expansion with different bases. The motivation behind this signal representation is that our problem can be solved efficiently with subsignals or the signal decomposition itself provides some information (e.g., in multiresolution analysis).

2. *Subband processing section.* Depending on the application, the subsignals are modified with various mathematical operations.

3. *Synthesis filter bank* (SFB): In the SFB, the signal is reconstructed from the subsignals. This is reverse of the AFB signal expansion as the subsignals are converted back into the "natural" domain.

$^{\dagger}L$ belongs to the set of natural numbers.

It is clear that the above description for a filter bank is given at a general level and we need more specific definitions. In fact, there are a number of properties that can be used to classify a filter bank and the most common ones are considered below:

Perfect-Reconstruction vs. Nearly-Perfect-Reconstruction

A perfect-reconstruction filter bank has the property that the signal decomposed with analysis filters can be recovered without error by using the synthesis filters. Nearly-perfect-reconstruction allows small errors in signal synthesis [66, 84]. The amount of error can be included as a controlled parameter in the design of filter banks. Two measures for this purpose are peak-to-peak reconstruction error and aliasing error [48]. A good reference listing of the existing design methods is given in [99].

Two-Channel vs. M-Channel Filter Banks

These terms refer to the number of subbands. The development of the FB theory started from the two-channel filter banks and some of the landmarks were PR designs using spectral factorization [88] and lattice-based filters [94]. Two-channel filter banks are commonly building blocks for hierarchical filter banks and each stage in a cascade splits the spectrum of the previous output in half. By using this idea, one may construct a non-uniform multichannel system. When the low-pass signal from the analysis filter output is repeatedly split in half with the same kind of two-channel filter bank, we actually have an implementation for a discrete-time wavelet series [31, 54].

Generally speaking, wavelets are a product of a different theoretical background that produced the PR filter banks. Wavelet families are mathematical constructs designed to satisfy certain criteria required for the application (e.g., the number of vanishing moments, the symmetry of the scaling/wavelet function, and regularity [90]). In the two-channel FB design, some of these required properties can be formulated into the optimization problem [2, 83] whilst maintaining good frequency selectivity. Thus the design-based two-channel FBs offer greater flexibility and application-tailored methods for time-scale analysis.

The name "M-channel filter bank", in turn, usually associates with uniform multichannel FBs where all subsignals have the same sample rate and this filter bank class is in the essence of this thesis. Section 2.6 gives the mathematical tools for the z-transform and polyphase domain analysis of the critically sampled M-channel filter banks.

Finite vs. Infinite-Impulse Response Filter Banks

The characteristic feature in infinite-impulse response (IIR) filters is the feedback loop; one or more previous output samples are utilized in the computation of the most recent output. Typically, they can be designed to achieve good frequency characteristics with a few coefficients but one must be aware of stability problems. IIR filters can be applied to synthesizing FBs [12, 61, 69], but the majority of the FB related research concentrates on finite-impulse response (FIR) systems. One reason for their popularity is the existing efficient and robust design methods for MFBs with exact perfect-reconstruction property. In this thesis, the focus is on FIR filter banks although recursive filters sneak in when we develop window function-based running-sum algorithms.
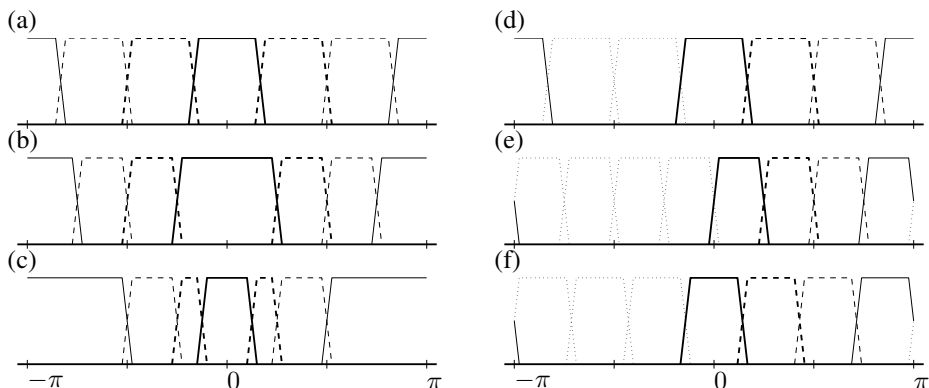
Figure 2.9: Various subband schemes of multirate systems: (a) Even stacking, subfilters with two-sided mainlobe. (b) Odd stacking, subfilters with two-sided mainlobe. (c) Non-uniform stacking. (d) Even stacking, subfilters with one-sided mainlobe. (e) Odd stacking, subfilters with one-sided mainlobe. (f) One low-pass and two high-pass subbands with complex subfilters where the number of subbands is odd. This kind of subchannelization with odd number of subbands is not possible with the complex filter banks considered in this thesis.

Uniform vs. Non-Uniform Filter Banks

A uniform filter bank means that the subfilters' amplitude response mainlobes are equispaced and equally wide. There are a few possibilities how to select the channel stacking, i.e., the location of the center frequencies [42, 68]. Figure 2.9 shows typical characteristics of some uniform filter banks with (a) even stacking and (b) odd stacking. The even/odd stacking FBs have been studied extensively and the subfilter center frequencies can be given systematically: $k\,\pi/M$ and $(2k+1)\pi/(2M)$, respectively[‡].

Non-uniform FBs are composed of subfilters whose passband widths are not equal. One example is shown in Figure 2.9 (c) which is a wavelet type of bandsplitting. By mixing different M channel filter banks in hierarchical configurations we can have quite exotic subband combinations where the frequency selectivity is focused on the desired location but without losing the time resolution of transient frequencies. However, this hierarchical FB or wavelet connection is outside the scope of this thesis and we pass the vast topic with this mention and references [32, 44, 55, 98].

Critically Sampled vs. Oversampled Filter Banks

If the subsignals of a filter bank are critically sampled, it means that the overall sample rate remains fixed in subbands. Practically speaking, when we denote the integer decimation factor of the kth subsignal with M_k and we have

$$\sum_k \frac{1}{M_k} = 1, \qquad (2.11)$$

[‡]Here, parameter M is also the decimation factor. For real-valued signals, the even-stacked FBs have one subband more than odd-stacked FBs with the same M. For these FBs, the passband width of both the low-pass and high-pass subfilters is half that of band-pass filters.

(a)

(b)

(c)
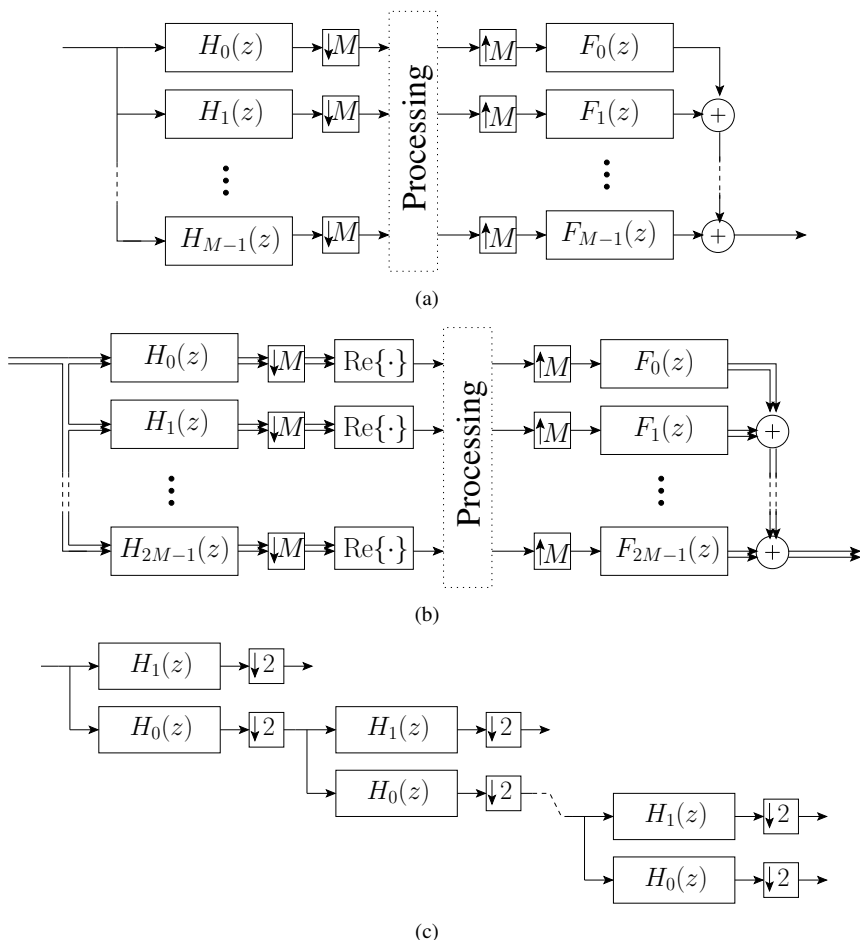
Figure 2.10: Various filter banks: (a) Real uniform FB with critical sampling. (b) Complex uniform FB with critical sampling. (c) Tree-structured real AFB or wavelet-series implementation with two-channel FBs.

then the FB is critically sampled [57]. In a critically sampled FB, the bandwidth of the subfilter is related with the sample rate of the subsignal. For example, in a uniform real M-channel critically sampled PR-FB the decimation factor is M in all subbands and then the ideal passband width is $\Delta w = \pi/M$.

In an oversampled filter bank, the sum on the left-hand side in equation (2.11) is greater than one. The oversampling can be simply achieved by removing the decimation (and interpolation) operations in an analysis-synthesis filter bank and this redundancy is helpful in some applications (e.g. oversampled wavelet-based noise suppression) or when time-invariant transforms are needed [27, 49]. Other ways to accomplish oversampling is to design filter banks where the number of subfilters (R in Figure 2.10 (e)) is higher than the decimation factor [18, 19, 97] and to introduce redundancy by obtaining the full phase information about transient frequency components as in the oversampled EMFB [100].
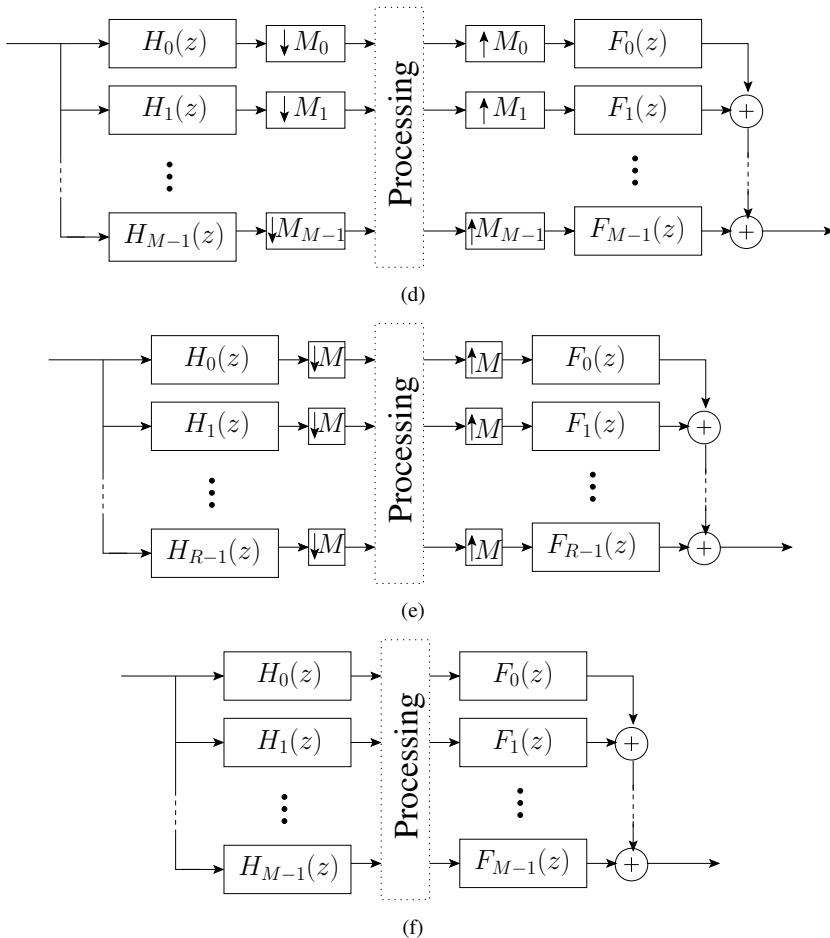
Figure 2.10: Various filter banks (continued): (d) Non-uniform real FB (e.g. tree-structured FB converted with noble identities). (e) Oversampled real FB when $R > M$. (f) M times oversampled filter bank (and the only time-invariant system in this FB gallery).

Real vs. Complex Filter Banks

We can make this distinction between filter banks just by saying that the subfilter impulse responses have complex/real-valued coefficients. However, it might be better to describe whether the filter bank is complex or real by looking at the amplitude responses of the subfilters. Sub-channelization in Figure 2.9 (a) and (d) are both even-stacked but on the right we have a complex subfilters which have the one-sided mainlobe when the amplitude responses are shown in the frequency range $\omega \in [-\pi, \pi]$. Similarly, we can form the complex odd-stacked FB as in Figure 2.9 (e). To emphasize the correspondence between the real and complex cases four matching subfilters on the positive frequency side are shown with the same line style.

The complex filter banks considered in this thesis are restricted to have $2M$ subfilters where M is the decimation factor. This means that with these FBs it is not possible to generate the bandsplitting scheme shown in Figure 2.9 (f) where we have an odd number of complex
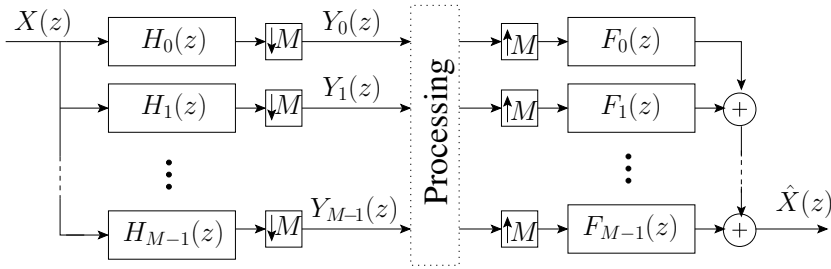
Figure 2.11: Real critically sampled uniform M-channel filter bank.

subfilters.

Paraunitary vs. Biorthogonal Filter Banks

Biorthogonality is a mathematical term and in the context of PR filter banks it refers to a property that the analysis and synthesis filters *together* satisfy the PR condition. There is no such simple dependence between the analysis and synthesis subfilters as the time-reversing property in paraunitary FBs. Thus the biorthogonality allows the design of arbitrary-delay subband systems.

Paraunitary or orthogonal filter banks can be viewed as a special case of biorthogonal FBs and here we have a simple definition: A filter bank is paraunitary if the analysis filters are obtained from the corresponding synthesis filters by time-reversing[§] the impulse responses. This restricts the delay of the paraunitary analysis-synthesis system to be the same as the subfilter order. However, this subclass is very important since the prototype filters for paraunitary MFBs can be mapped into orthogonal rotations for fast algorithms or generate even-stacked subfilters having symmetric impulse responses.

We conclude this superficial prelude with a gallery of various filter banks (cf. Figure 2.10) that can be specified with the presented terms. Note especially the subband system in Figure 2.10 (b) which is a critically sampled filter bank because for each M new complex samples (real and imaginary parts) the analysis bank produces $2M$ real samples. The purpose of the wordplay above is to clear the meaning of certain ambivalent terms so that we can discuss the various filter banks with compact terminology and, hopefully, avoid doublespeak.

2.6 Critically Sampled Uniform M-Channel Filter Banks

A detailed study of the FB theory requires a book of its own and thus we have gathered here only the relevant parts that are necessary for the mathematical representation of the EMFB. As the subtitle promotes, we do a dissection for the real critically sampled uniform M-channel filter bank and review in this section the z-transform and polyphase domain representations to FBs of this kind. Both of these are tools for the analysis of multirate systems.

[§]Accompanied with conjugation if the impulse response is complex-valued.

2.6.1 M-Channel FB and z-Transform Domain Analysis

The M-channel filter bank is again shown in Figure 2.11 to introduce the z-transform domain notations: The real signal $X(z)$ is the input for M analysis filters $H_k(z)$ that bandlimit the signal before downsampling by M. Thus the FB is critically sampled. When the z-transform relationship from equation (2.2) is applied for the kth subsignal $Y_k(z)$, we have

$$Y_k(z) = \frac{1}{M} \sum_{i=0}^{M-1} H_k(z^{1/M} W_M^{-i}) X(z^{1/M} W_M^{-i}), \tag{2.12}$$

where $k = 0, 1, \ldots, M-1$. The purpose is to present the PR condition and we assume that the processing stage does not modify the subsignals. The output signal $\hat{X}(z)$ is reconstructed in the synthesis section. The subsignals are first upsampled and then filtered with the synthesis filters $F_k(z)$ which remove the imaging spectra. Again, we need the identity for the sampling rate alteration; the interpolation relation of equation (2.4) is used for the subsignals and all branches are combined:

$$\begin{aligned}
\hat{X}(z) &= \sum_{k=0}^{M-1} F_k(z) Y_k(z^M) \\
&= \frac{1}{M} \sum_{k=0}^{M-1} F_k(z) \sum_{i=0}^{M-1} H_k(z W_M^{-i}) X(z W_M^{-i}) \\
&= \frac{1}{M} \sum_{i=0}^{M-1} X(z W_M^{-i}) \sum_{k=0}^{M-1} F_k(z) H_k(z W_M^{-i}).
\end{aligned} \tag{2.13}$$

From this double sum we can obtain the conditions for the PR. Namely, in the output signal equation (2.13) there appears the desired $X(z)$ but also its $M-1$ aliased components. We separate them for a better view:

$$\begin{aligned}
\hat{X}(z) &= \frac{1}{M} X(z) \sum_{k=0}^{M-1} F_k(z) H_k(z) \\
&+ \frac{1}{M} \sum_{i=1}^{M-1} X(z W_M^{-i}) \sum_{k=0}^{M-1} F_k(z) H_k(z W_M^{-i}).
\end{aligned} \tag{2.14}$$

When assuming the overall delay of the causal analysis-synthesis system to be τ samples, the necessary requirements for the PR become

$$\frac{1}{M} \sum_{k=0}^{M-1} F_k(z) H_k(z W_M^{-i}) = \delta[i] z^{-\tau}, \qquad i = 0, 1, \ldots, M-1. \tag{2.15}$$

The delay of the system, denoted by τ for the rest of this thesis, is a design parameter. In addition, we use as the reference delay that of a causal paraunitary filter bank. For this FB, τ has a fixed value and it is connected with the length of the subfilters. If the subfilter order is N (i.e., $N+1$ is the length of the impulse response in samples), then the paraunitary FB delay is $\tau = N$. Biorthogonal filter banks are more flexible in the sense that τ can be varied.

In comparison with the reference FB, we define D ($D \in \mathbb{Z}^{\P}$) as a parameter to adjust the signal lag in the FB. Negative values for D cause the FB to advance the signal by $|D|$ samples compared with the reference one, whereas positive values for D correspond a shift towards the maximum-phase FB. In particular, the $D = 0$ case results in the exactly same delay as in the paraunitary reference system *but the FB may still be biorthogonal*. When the delay is less than N, the FB can be called a low-delay, whereas, the other way round, if $\tau > N$, we speak of high-delay filter banks.

The PR condition in (2.15) can also be written as a matrix equation. The so-called *aliasing component matrix* (or modulation-domain matrix) $\boldsymbol{H}_{\mathrm{m}}(z)$ contains the analysis filters

$$
\boldsymbol{H}_{\mathrm{m}}(z) = \frac{1}{\sqrt{M}}
\begin{bmatrix}
H_0(z) & H_0(z\,W_M^{-1}) & \cdots & H_0(z\,W_M^{-M+1}) \\
H_1(z) & H_1(z\,W_M^{-1}) & \cdots & H_1(z\,W_M^{-M+1}) \\
\vdots & \vdots & & \vdots \\
H_{M-1}(z) & H_{M-1}(z\,W_M^{-1}) & \cdots & H_{M-1}(z\,W_M^{-M+1})
\end{bmatrix}. \tag{2.16}
$$

The elements of a row vector $\boldsymbol{f}(z)$ are the synthesis filters,

$$
\boldsymbol{f}(z) = \frac{1}{\sqrt{M}} \begin{bmatrix} F_0(z) & F_1(z) & \cdots & F_{M-1}(z) \end{bmatrix}, \tag{2.17}
$$

and we have included the scaling term $1/\sqrt{M}$ to both $\boldsymbol{H}_{\mathrm{m}}(z)$ and $\boldsymbol{f}(z)$. Then we can write the equivalent form of the PR condition for a critically sampled M-channel filter bank as

$$
\boldsymbol{f}(z)\boldsymbol{H}_{\mathrm{m}}(z) = \begin{bmatrix} z^{-\tau} & \underbrace{0 \cdots 0}_{\#(M-1)} \end{bmatrix}^T. \tag{2.18}
$$

2.6.2 Perfect-Reconstruction in the Polyphase Domain

The critically sampled filter bank in Figure 2.11 has a polyphase representation which is a compact way to analyze filter banks. In order to write the input-output relation of the system we begin by writing the polyphase component vector $\boldsymbol{x}_{\mathrm{p}}(z)$ of signal $X(z)$. The signal polyphase decomposition has been discussed in Section 2.3, but we repeat the equation here for clarity:

$$
\boldsymbol{x}_{\mathrm{p}}(z) = \begin{bmatrix} X_0(z) & X_1(z) & \cdots & X_{M-1}(z) \end{bmatrix}^T,
$$

where $X_\ell(z) = \sum\limits_{m=-\infty}^{\infty} x[\ell + mM]\, z^{-m}$.

Then the analysis and synthesis filters are expressed as matrices and here we apply the polyphase decompositions (2.9) and (2.10). The following polyphase filter equations contain the assumption that all subfilters of the analysis and synthesis bank have the same order N. The polyphase forms of the analysis filters are

$$
H_k(z) = \sum_{\ell=0}^{M-1} z^{-(M-1-\ell)}\, \tilde{G}_{\ell,k}(z^M), \tag{2.19}
$$

where $\tilde{G}_{\ell,k}(z) = \sum_m h_k[M - 1 - \ell + mM]\, z^{-m}$.

$^{\P}\mathbb{Z}$ is the set of integer numbers
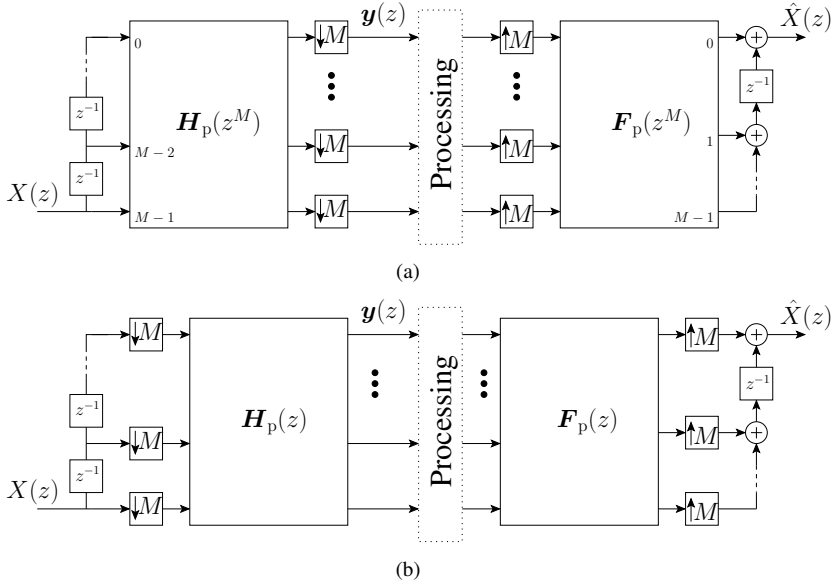
Figure 2.12: (a) Polyphase domain representation of the M-channel filter bank. (b) The same polyphase matrix system transformed with the identities for sampling rate alteration.

Similarly, we have for the synthesis filters

$$F_k(z) = \sum_{\ell=0}^{M-1} z^{-\ell} \, G_{\ell,k}(z^M), \tag{2.20}$$

and $G_{\ell,k}(z) = \sum_m f_k[\ell + mM] \, z^{-m}$. The analysis and synthesis polyphase matrices are defined as follows:

$$\left[\boldsymbol{H}_{\mathrm{p}}(z)\right]_{k,\ell} = \tilde{G}_{\ell,k}(z) \tag{2.21}$$

$$\left[\boldsymbol{F}_{\mathrm{p}}(z)\right]_{\ell,k} = G_{\ell,k}(z), \tag{2.22}$$

where k and $\ell = 0, 1, \ldots, M - 1$. The critically sampled filter bank is shown with the polyphase filters in Figure 2.12 (a). If $\boldsymbol{y}(z)$ is the vector containing the z-transforms of the subsignals,

$$\boldsymbol{y}(z) = [Y_0(z) \ Y_1(z) \ \ldots \ Y_{M-1}(z)]^T,$$

then it can be directly written as

$$\boldsymbol{y}(z) = \boldsymbol{H}_{\mathrm{p}}(z) \, \boldsymbol{x}_{\mathrm{p}}(z). \tag{2.23}$$

The polyphase matrix input-output relation for the critically sampled filter bank is then

$$\hat{X}(z) = \mathbf{c}^T(z) \, \boldsymbol{F}_{\mathrm{p}}(z^M) \boldsymbol{H}_{\mathrm{p}}(z^M) \, \boldsymbol{x}_{\mathrm{p}}(z^M), \tag{2.24}$$

where $\mathbf{c}^T(z) = z^{-(M-1)} \cdot \left[1 \ z^{-1} \ \cdots \ z^{-(M-1)}\right]$. On the first glance, formulating the necessary PR condition for the product of polyphase filter matrices in (2.24) is more difficult when

26

compared with the z-transform domain requirements. Fortunately, our savior has been presented in [90]. When the downsampling and upsampling operations are moved next to delay lines, as in Figure 2.12 (b), we get the revelation that $\boldsymbol{F}_\mathrm{p}(z)\,\boldsymbol{H}_\mathrm{p}(z)$ has to be a pseudo-circulant matrix. The term *circulant matrix* refers to a matrix whose rows are circularly shifted versions of each other and *pseudo* comes from the added delay element.

As previously mentioned, the analysis and synthesis reconstruction delay is $\tau = N + D$, where D is an integer-valued adjustable design parameter. In order to express the PR condition in the polyphase domain, and in the compact matrix form, τ must be split into a multiple of M and the remainder d as $\tau = Ms - 1 + d$, where $s = \lfloor(\tau+1)/M\rfloor$ and $d \in \{0, 1, \ldots, M-1\}$. The reason why τ is broken into this many pieces becomes clear when we look at the pseudo-circulant matrix on the right hand side (RHS) of the PR condition below: Now the biorthogonal critically sampled M-channel causal filter bank is perfectly reconstructing if

$$\boldsymbol{F}_\mathrm{p}(z)\,\boldsymbol{H}_\mathrm{p}(z) = z^{-s+1} \begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix}. \tag{2.25}$$

The notation $\mathbf{I}_n$ stands for an identity matrix with dimensions $n \times n$. The pseudo-circulant matrix will be opened in Section 3.2 to verify that the PR truly follows from this condition.

The subfilter order gives the limits for the advance/lag parameter D; it must be contained within $D \in \{0, \pm 1, ..., \pm(N - M + 1)\}$. For example, one may substitute the minimum delay value $D_{\min} = -N + M - 1$ to the RHS of equation (2.25) and then replace the polyphase filter matrices with the pseudo-circulant matrix (now an identity matrix). This choice reduces the subband system to the direct connections between the input and output buffer which means that $\tau = M - 1$, i.e., the delay of a causal M-phase polyphase decomposition/reconstruction pair shown in Figure 2.6.

2.6.3 Paraunitary Filter Bank

The inherent property of paraunitary (real) filter banks is the exact relation between the impulse responses of the analysis and synthesis filters. These relations mean that the impulse response of the kth synthesis filter is obtained from that of the corresponding analysis filter by the time reversal:

$$f_k[n] = h_k[N - n], \qquad n = 0, 1, \ldots, N. \tag{2.26}$$

Alternatively, the transfer functions of these filters are related through

$$F_k(z) = z^{-N}\,H_k(z^{-1}). \tag{2.27}$$

The reconstruction delay of a paraunitary FB is the same as the subfilter order. When the synthesis filters are replaced with the above time reversed analysis filters in the equation (2.15), we get the following z-transform domain PR requirement for the paraunitary filter bank:

$$\frac{1}{M} \sum_{k=0}^{M-1} H_k(z^{-1})\,H_k(z\,W_M^{-i}) = \delta[i], \qquad i = 0, 1, \ldots, M - 1. \tag{2.28}$$

Here, the common delay term z^{-N} is dropped from both sides of the above equation.

Then we state the same in the polyphase domain. In the definition for the pseudo-circulant matrix in equation (2.25) we insert $\tau = N$. Depending on the remainder $d = N + 1 - Ms$, where $s = \lfloor(N+1)/M\rfloor$, we consider two cases:

1) Time-Reversing Relation when $d \neq 0$

A direct substitution of $f_k[n] = h_k[N - n]$ into the synthesis polyphase matrix equation (2.22) gives

$$\left[\boldsymbol{F}_{\mathrm{p}}(z)\right]_{\ell,k} = \sum_m f_k[\ell + mM]\, z^{-m} = \sum_m h_k[N - (\ell + mM)]\, z^{-m} = \left[\tilde{\boldsymbol{H}}_{\mathrm{p}}(z)\right]_{\ell,k}. \quad (2.29)$$

Due to perfect-reconstruction condition expressed by (2.8) and (2.25), we may write

$$\hat{X}(z) = \boldsymbol{c}^T(z)\, \tilde{\boldsymbol{H}}_{\mathrm{p}}(z^M)\, \boldsymbol{H}_{\mathrm{p}}(z^M)\, \boldsymbol{x}_{\mathrm{p}}(z^M) = z^{-M+1} z^{-Ms+M-d} X(z) = z^{-N} X(z).$$

2) Time-Reversing Relation when $d = 0$

The subfilter length is now an integer multiple of the decimation factor: $N + 1 = LM$. This is an important special case since it yields elegant theoretical results as well as fast algorithms. Now it is convenient to express the elements of the synthesis polyphase matrix as

$$\left[\boldsymbol{F}_{\mathrm{p}}(z)\right]_{\ell,k} = \sum_{m=0}^{L-1} f_k[\ell + mM]\, z^{-m}$$

$$= \sum_{m=0}^{L-1} h_k[LM - 1 - (\ell + mM)]\, z^{-m}$$

$$= \sum_{m'=0}^{L-1} h_k[(L-1)M + M - 1 - \ell - (L - 1 - m')M]\, z^{-(L-1-m')}$$

$$= z^{-(L-1)} \sum_{m'=0}^{L-1} h_k[M - 1 - \ell + m'M]\, z^{m'}$$

$$= z^{-L+1} \left[\boldsymbol{H}_{\mathrm{p}}(z^{-1})\right]_{k,\ell}.$$

Here, summation order reversion provides the desired result. The resulting matrix connection is transposed to flip the indices k and ℓ and then substituted into (2.25). The delay term is further canceled from both sides and the paraunitary polyphase domain analysis matrices produce an identity matrix:

$$\boldsymbol{H}_{\mathrm{p}}^T(z^{-1})\, \boldsymbol{H}_{\mathrm{p}}(z) = \mathbf{I}.$$

This "paraunitarity is contained in biorthogonality"-trend is again demonstrated with the EMFB subfilter definition.

2.7 Survival Guide to Modulated Filter Banks

2.7.1 Mother of All Modulated Filter Banks

We conclude this chapter by reviewing the subfilter modulation concept with the aid of the generalized discrete Fourier transform filter bank (GDFT-FB) as well as positioning the existing paraunitary MFBs (considered in the sequel subsection) in this framework. The GDFT-FB is a flexible definition as it covers CMFBs, EMFBs, and MDFT-FB as special cases. In addition,
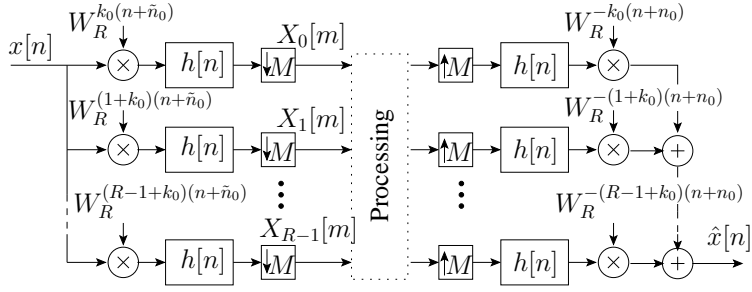
Figure 2.13: Generalized discrete Fourier transform filter bank.

the secondary purpose is to give an appropriate respect to the authors whose discoveries have greatly influenced and inspired the overall process of writing this thesis.

In a sense, continuous time-frequency analysis or, equivalently, short-time Fourier transform is signal element detection assuming that the signal consists of temporary circular motion components. If we are studying discrete-time signals, problems of the same kind are considered in modulated filter bank theory and, in the engineering playground, the emphasis is on design and implementation related issues. In particular, the filter banks studied in this thesis are computationally efficient discrete-frequency, discrete-time, and short-time Fourier analysis tools. The MFB viewpoint is more natural for people with a signal processing background as well as for the author. Thus we start our journey with a versatile filter bank definition known as the generalized discrete Fourier transform filter bank [30]. There are two main reasons in this approach: 1) the GDFT-FB demonstrates the concept of frequency shifted subfilters; and 2) all MFB types under consideration in this thesis can be obtained from the GDFT-FB subfilter definition as a special case.

The GDFT-FB system in the analysis-synthesis configuration is shown in Figure 2.13. The analysis stage divides the input signal $x[n]$ into numerous subsignals $X_k[m]$ where the discrete frequency index ranges as $k = 0, 1, \ldots, R - 1$. The subband processing stage operates with lower rate subsignals as the downsampling operators take every Mth sample and discard the rest. In particular, assuming that the input signal is complex-valued, the subband system is critically sampled when $R = M$. Each analysis branch is composed of a prototype filter $h[n]$ (having low-pass frequency characteristics) and a time varying exponential multiplier $W_R^{(k+k_0)(n+\tilde{n}_0)}$ where the Rth root of unity is applied: $W_R = \exp(-j\frac{2\pi}{R})$. Together they compose the complex-valued analysis subfilters

$$h_k[n] = h[n] \, W_R^{(k+k_0)(N-n+\tilde{n}_0)}, \tag{2.30}$$

where N is the prototype filter order. The synthesis stage reverses the analysis FB operations with subfilters

$$f_k[n] = h[n] \, W_R^{-(k+k_0)(n+n_0)}. \tag{2.31}$$

Parameters k_0 and n_0 adjust the subfilter center frequency ω_k and the phase of the modulation sequence. When the prototype filter symmetry $h[n] = h[N - n]$ holds, the same parameter $n_0 = \tilde{n}_0$ is used in the analysis and synthesis filters.

29
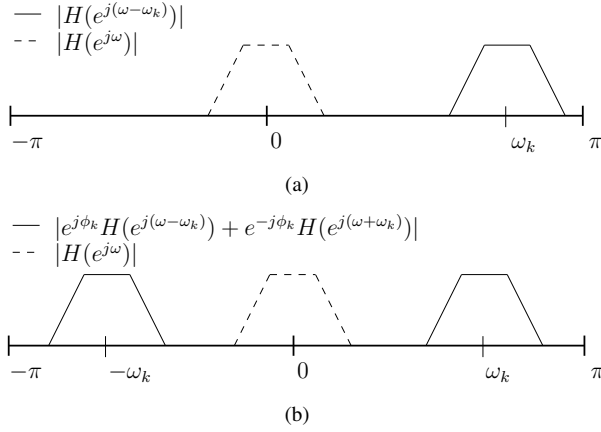
Figure 2.14: Two modulation types contained in GDFT-FB subfilter definition: (a) Exponential modulation is equivalent to frequency shifting by ω_k. (b) Cosine-modulated prototype filter has two mainlobes within $[-\pi, \pi]$.

2.7.2 Frequency Shifting of Low-Pass Prototype Filter

By its complex-valued nature, the GDFT-FB generates R subfilters that divide the frequency range $[-\pi, \pi]$ into uniform subbands. The concept of modulation is the key to understand how the bandsplitting operation actually works. We open the exponential modulation by considering it as shifting in the frequency domain. This can be seen by applying the basic results of Fourier analysis.

The low-pass prototype filter in Figure 2.14 (a) has real-valued coefficients and thus the amplitude response

$$|H(e^{j\omega})| = \Big| \sum_{n=0}^{N} h[n]\, e^{-j\omega n} \Big| \tag{2.32}$$

is symmetric with respect to zero frequency. Similarly, we can find an expression for the synthesis subfilter amplitude response

$$
\begin{aligned}
|F_k(e^{j\omega})| &= \Big| \sum_{n=0}^{N} h[n]\, W_R^{-(k+k_0)(n+n_0)}\, e^{-j\omega n} \Big| \\
&= \Big| e^{j\, 2\pi(k+k_0)n_0/R} \sum_{n=0}^{N} h[n]\, e^{-j(\omega - 2\pi(k+k_0)/R)n} \Big| \\
&= |e^{j\, 2\pi(k+k_0)n_0/R}|\, |H(e^{j(\omega-\omega_k)})| = |H(e^{j(\omega-\omega_k)})|,
\end{aligned}
\tag{2.33}
$$

where $\omega_k = (k+k_0)2\pi/R$. In other words, the exponential modulation of the GDFT-FB shifts the center for the amplitude response of the prototype from $\omega = 0$ to $\omega = \omega_k$.

Cosine modulation can be obtained from the GDFT-FB by taking the real part of the subfilters. When writing the frequency response of cosine-modulated subfilters, it is convenient to

Table 2.1: GDFT-FB synthesis subfilter definitions covering all MFB modulation sequences considered in the sequel as special cases, even though lacking the necessary tweaks to provide the PR in the CS mode.

FB type	GDFT-FB Modulation Sequence	n_0	k_0	# of Subbands (M decimation factor)
DFT-FB	$W_M^{-(k+k_0)(n+n_0)}$	0	0	M
Odd-st. CMFB	$2\,\mathrm{Re}\{W_{2M}^{-(k+k_0)(n+n_0)}\}$	$(M+1)/2$	$1/2$	M
Odd-st. EMFB	$W_{2M}^{-(k+k_0)(n+n_0)}$	$(M+1)/2$	$1/2$	$2M$
Even-st. EMFB$^{\parallel}$	$W_{2M}^{-(k+k_0)(n+n_0)}$	$(M+1)/2$	0	$2M$
MDFT-FB	$W_{2M}^{-(k+k_0)(n+n_0)}$	$-N/2$	0	$2M$

use the complex conjugate $f_k^*[n]$ term as follows

$$
\sum_{n=0}^{N}\left(f_k[n]+f_k^*[n]\right)e^{-j\omega n}=\sum_{n=0}^{N}h[n]\underbrace{\left(W_R^{-(k+k_0)(n+n_0)}+W_R^{(k+k_0)(n+n_0)}\right)}_{2\cos\left(2(k+k_0)(n+n_0)\pi/R\right)}e^{-j\omega n}
$$
$$
=e^{j\,\phi_k}\,H(e^{j(\omega-\omega_k)})+e^{-j\,\phi_k}\,H(e^{j(\omega+\omega_k)})
$$

(2.34)

with short-hand notation $\phi_k = 2\pi(k+k_0)n_0/R$. As shown schematically in Figure 2.14 (b), cosine modulation generates two mainlobes in the frequency range $[-\pi,\pi]$. The center frequencies $\pm\omega_k$ are equally distant from the zero frequency.

2.7.3 GDFT-FB with Even and Odd-Stacked Subchannels

The parameter k_0 in the GDFT-FB subfilter definition influences the offset of the center frequencies ω_k. These two parameter values are linked with the subchannelization schemes relevant to this work. Even-stacked modulation is realized with $k_0 = 0$ and $R = M$; in this mode the center frequency of the 0th subfilter is located at zero frequency. The decimation factor is equal to the number of subbands and the filter bank is critically sampled. The second alternative is odd-stacked modulation with $k_0 = 1/2$ and $R = M$. The center frequencies are then given as $\omega_k = (2k+1)\pi/(2M)$, where we let $k = 0, 1, \ldots, 2M-1$. The subfilter symmetry and anti-symmetry property of GDFT-FB is unanswered unless we fix the parameters with specific values. For instance, the basic DFT-FB subfilters are generated with $k_0 = 0$, $n_0 = 0$, and $R = M$. However, the real/imaginary parts of the subfilters are not symmetric even though the prototype filter is symmetric.

All modulated filter banks to be encountered in the following chapters can be traced back to the GDFT-FB definition, sometimes with a twist in order to satisfy the PR condition in the CS mode. As shown in Table 2.1, the values for k_0 and n_0 determine the modulation

---

$^{\parallel}$Even-stacked EMFB is defined with two sets of subfilters and the listed modulation sequence matches with the first set. Both the even-stacked EMFB and MDFT-FB select subsamples in an alternating manner to provide PR without sample rate growth.

Table 2.2: Summary of modulated filter bank properties.

FB type	Even/ Odd- Stacking	Real/ Complex Input	Subfilter Symbol	Real/ Complex Subsignals	# of Subbands (M decimation factor)
CS CMFB	Odd	Real	$f_k^{\text{c}}[n]$	Real	M
	Even	Real	$f_k^{\text{c}-\text{e}}[n]$	Real	$M+1$
CS SMFB	Odd	Real	$f_k^{\text{s}}[n]$	Real	M
CS EMFB	Odd	Complex	$f_k^{\text{e}}[n]$	Real	$2M$
	Even	Complex	$f_k^{\text{e}-\text{e}}[n]$	Real	$2M$
2xOS EMFB	Odd	Complex	$f_k^{\text{e}}[n]$	Complex	$2M$
	Even	Complex	$f_k^{\text{e}-\text{e}}[n]$	Complex	$2M$
	Odd	Real	$f_k^{\text{e}-\text{r}}[n]$	Complex	M
	Even	Real	$f_k^{\text{e}-\text{re}}[n]$	Complex	$M+1$

type applied with the synthesis subfilters. The particular detail that is a link between various critically sampled perfect-reconstruction MFBs is the parameter $R = 2M$ in the modulation sequence generation. In effect, the center frequency separation of the neighbouring subfilters is halved, together with the passband width, when compared with a basic CS DFT-FB having the same decimation factor. The EMFBs and MDFT-FB have thus $2M$ subchannels while the decimation factor is M. However, they all have different subsample selection strategies (real and/or imaginary parts) to maintain the PR property in CS mode.

The odd-stacked CMFB is the only entry in Table 2.1 that is designed for critically sampled processing of real-valued signals. When generated using a perfect-reconstruction prototype (and regarding subchannels redundant due to the conjugate symmetry), the odd-stacked CMFB connects the GDFT-FB to the "standard" M-channel paraunitary filter bank theory.

In the course of this thesis, various MFBs are encountered due to these configuration options. Table 2.2 brings order to this modulation mayhem and shows the notational conventions used for each parameter combination. The reader can use this table to find a subband system that suits the best for the purpose. Secondly, the superscript MFB identifiers are used frequently and Table 2.2 might serve as a helpful quick reference for the notations.

2.8 Existing Paraunitary Modulated Filter Banks

2.8.1 Cosine-Modulated Filter Banks

A straightforward approach to design M-channel paraunitary FBs would be to minimize the stopband attenuation of each individual subfilter and utilize shift-orthogonality PR conditions as constraints [23]. In practice, the optimization of such system is a difficult task and the number of free parameters is often reduced. In modulated filter banks, the design problem is simplified into a low-pass prototype filter optimization with perfect-reconstruction constraints. Cosine-modulated filter banks have been studied extensively over the past two decades. Re-

search topics have been FB theory [53, 65, 74, 88], FB design techniques [29, 63, 67, 68, 105], and fast algorithms [48, 56, 85]. Possible application areas cover subband coding [36, 75, 104] and transmultiplexing [79, 86, 87], just to name a few. We do not provide in the sequel a thorough literature survey on the subject, but concentrate in gathering the key results for this thesis.

CMFBs belong to a class of filter banks where the subfilters are generated by multiplying a low-pass prototype filter with cosine sequences. In effect, the cosine modulation is a frequency shift in the frequency range $[0, \pi]$ and the mainlobe of the prototype filter is moved around a new center frequency. One branch of research gained progress in designing perfect-reconstruction CMFBs. The first results were obtained $2M$-length subfilters and at that time the authors called them filter banks based on time domain aliasing cancellation [72]. Later on, the PR-study culminated in generalization for longer filters by independent authors; ELTs by Malvar [56] and lattice-based CMFBs by Koilpillai & Vaidyanathan [48]. In both of these papers, the polyphase filter part of the implementation structure was constructed with orthogonal rotations whose coefficients are sines and cosines of design angle variables. The resulting odd-stacked cosine-modulated filter banks are inherently satisfying the PR condition and this allows the use of unconstrained non-linear optimization methods [56, 48, 80, 95].

In our previous work [5, 11], we have preferred the former ELT approach because of the beauty of the flowgraph: the modular implementation structure known as the fast ELT is a canonical cascade and aesthetically pleasing for a right-minded engineer. The ELT-based CMFBs are critically sampled because the decimation factor is equal to the number of subbands. One restriction is that the length of the subfilters is a multiple of $2M$. Parameter L, known as the *overlap factor*, is here an even integer defining the impulse response length in the equation for the kth synthesis subfilter $f_k^c[n]$:

$$f_k^c[n] = 2\, h[n]\, \cos\left[\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],$$

where $n = 0, 1, \ldots, LM - 1$ and $k = 0, 1, \ldots, M - 1$. We denote the prototype filter with $h[n]$. The ELT-based CMFB is a paraunitary filter bank where the kth analysis filter is the time reversed version of the corresponding synthesis filter:

$$h_k^c[n] = f_k^c[LM - 1 - n]. \tag{2.35}$$

This choice of modulating frequencies leads to a uniform FB with odd stacking. Furthermore, it is shown in [57] that the CMFB is a PR system when the symmetric low-pass prototype filter satisfies

$$\sum_{i=0}^{L-2s-1} h[n + iM]\, h[n + iM + 2sM] = \frac{1}{2M}\delta[s], \tag{2.36}$$

where $n = 0, 1, \ldots, M/2 - 1$ and $s = 0, 1, \ldots, L/2 - 1$. Actually, we obtain the PR equation (2.36) as a special case from the PR conditions of biorthogonal CMFBs and we bring the subject forth again in Section 4.2.

Two widely used objectives for minimization are the stopband energy (ℓ_2-norm) and minimax criteria (ℓ_∞-norm) [82]. The common habit is to express the stopband edge ω_s with a roll-off factor ρ:

$$\omega_s = \frac{(1 + \rho)\pi}{2M}.$$

(a)

(b)
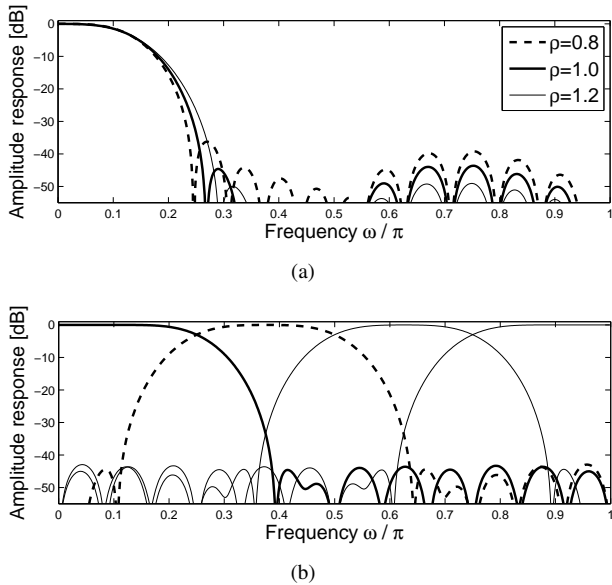
Figure 2.15: ELT-angle-based prototype filter design examples for decimation factor $M = 4$ and prototype filter order $N = 8M - 1$: (a) Three inherently PR prototypes with roll-off factors $\rho = \{0.8, 1.0, 1.2\}$. (b) Paraunitary 4-band filter bank generated from the $\rho = 1.0$ prototype with odd-stacked cosine modulation.

The roll-off controls the spectral overlap of the subfilter passbands. With practical FIR filters there is always a tradeoff between stopband attenuation and the width of the transition band. Typically, the PR conditions take care of the passband shaping to such an extent that the passband criteria are omitted in optimization problems except in situations they are specifically required for some reasons. In addition, the rule of thumb in unrestricted low-pass FIR filter design holds roughly for the PR prototypes: Halving the roll-off parameter *and* maintaining similar sidelobe attenuation levels requires doubling the filter length [81].

In order to bring some concrete examples to the odd-stacked CMFB discussion we show in Figure 2.15 (a) three prototype filters using different roll-off values. They satisfy exactly the PR constraint in (2.36) as the symmetric prototype filter impulse responses are obtained from the ELT butterfly angles. The overlap of the prototype filters is $L = 8$ and they generate a 4-band critically sampled filter bank for real-valued signals. The amplitude responses of $\rho = 1.0$ subfilters are shown in Figure 2.15 (b).

Nevertheless, it is also misleading if one solely considers the prototype filter stopband attenuation as the unique metric. For instance, the (mostly) equiripple sidelobes of ℓ_∞-norm PR designs are not preserved in cosine-modulation (whilst the amplitude responses of the EMFB subfilters are frequency shifted replicates of the prototype filter). The same holds with the prototype filters shaped with ℓ_2-norm. In addition, the application needs may dictate the use of smooth and short parametric window functions [58, 72]. The parameters of the window functions can be selected in such a manner that the generated fast sidelobe decay is preferable over the optimization norms mentioned above. One example of when this has significance is audio coding; aliasing from nearby subbands is less perceptible than aliasing from far-away
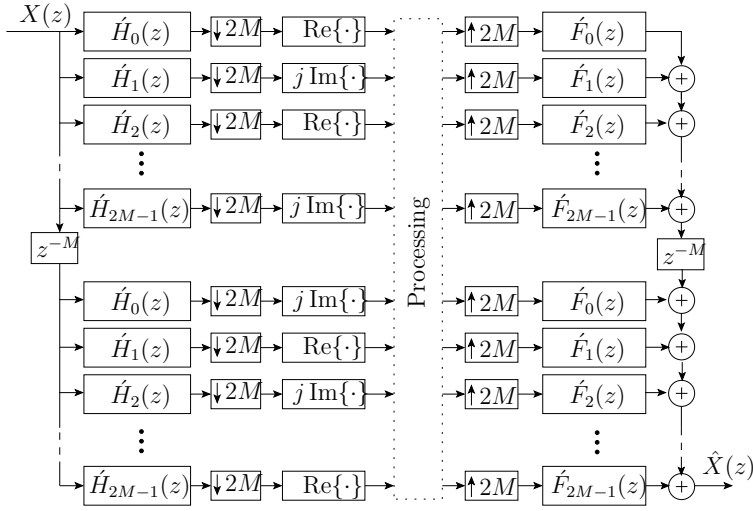
Figure 2.16: Type-I modified DFT-FB.

bands.

2.8.2 Modified Discrete Fourier Transform Filter Bank

When we discuss modulated filter banks, we can not bypass discrete Fourier transform filter banks [13, 30, 38, 103]. The DFT-FB is a complex filter bank and its subfilters are generated by modulating a low-pass prototype filter with exponential sequences W_M^{kn}. The modulation provides even-stacked subchannelization and the conjugation property of the DFT applies with subfilters equidistant from zero frequency. Linear-phase subfilters (assuming a symmetric prototype filter) can be arranged by shifting the modulation phase (parameter n_0 in GDFT-FB definition). Moreover, the key observation is that the implementation of the DFT-FB consists of the FFT-block and the polyphase decomposition of the prototype filter. Thus the computational complexity of the DFT-FB is of the same order with conventional block transforms. Longer prototype filters would provide more selective FBs, but the drawback is that the exact perfect-reconstruction is not feasible except for a trivial case [51, 57]. Nevertheless, this is a convenient way to build filter banks as long as we tolerate the aliasing and reconstruction errors in our application. Impertinently speaking, all further developments of the modulated filter bank theory are simply variations of the DFT-FB theme - though many ingenious steps have been achieved during the overall process.

The problem concerning the PR property of the DFT-FBs has been solved by Karp and Fliege with the MDFT-FB [46]. If the prototype filter is optimized with the same PR criteria derived for the odd-stacked CMFB, then it suits the MDFT-FB as well [42]. The kth subsignal of the MDFT-FB is alternating between purely real and imaginary coefficients; such subsample selection scheme is necessary for the perfect-reconstruction with critical sampling. This time varying representation for the MDFT-FB can be modeled with two parallel subband systems that differ only with the subsignal component selection. The flowgraph in Figure 2.16 shows this kind of fixed MDFT-FB and this representation is comparable with the other MFB types discussed later on.

The synthesis subfilters of the symmetric prototype Type I MDFT-FB can be given as

$$\acute{f}_k[n] = \sqrt{2}\, h[n]\, \exp\left[j\left(n - \frac{N}{2}\right)\frac{k\pi}{M}\right] \tag{2.37}$$

assuming that the prototype filter is scaled as in (2.36). The analysis subfilters are exactly determined by the synthesis subfilters as $\acute{h}_k[n] \equiv \acute{f}_k[n]$. The amplitude responses of the subfilters divide the frequency range $[-\pi, \pi]$ into $2M$ subbands with center frequencies $\omega_k = k\pi/M$, where $k = 0, 1, \ldots, 2M - 1$, i.e., the subchannel splitting scheme is even-stacked. Type II MDFT-FBs utilize a slightly different modulation phase in subfilter generation and it is also an even-stacked filter bank intended for processing complex-valued input signals [42].

Chapter 3

Lapped Transforms

In early subband signal processing applications, orthogonal block transforms were used for signal decorrelation. The Karhunen-Loève transform (KLT) [45, 52] is an optimal choice for this purpose as the autocorrelation properties of the signal define the transform base. When the forward transform is applied on the signal, the resulting subsignal coefficients are uncorrelated and the autocorrelation matrix is thus diagonal. In a way, the widely used discrete cosine transform (DCT) [76] with undulating basis functions "imitates" the KLT computed for certain type of autoregressive signals with frequency components mostly residing on the lower frequencies [43]. The sufficiently close matching with the statistical properties of signals in image compression and signal analysis is one reason why the DCT has been adopted to image processing standards [24]. The other is efficient implementation algorithms; depending on the nature of the application, split-radix, recursive, or integer realizations are available [50, 106]. However, there are applications where the cosine bases leave room for improvement. For instance, in image signal processing we can easily construct such examples where the effect of the transform basis can be visualized. The DCT is an $M \times M$ square transform matrix with basis functions that take the shape of cosine-waves. Even though the basis functions are continuous, the abrupt "cut" at block boundaries easily causes artifacts to the reconstructed signal, as is exemplified in Figure 3.1.

The motivation to reduce this blocking effect was central in the development of lapped transforms (LTs). LTs are characterized by overlapping basis functions whose tails smoothly decay to zero. This forces the reconstructed signal to evolve smoothly from the signal component reconstructed from one block into the signal component reconstructed from the next block, thus creating a smooth signal transition across boundaries, leading to a reduction of blocking artifacts. The perfect-reconstruction of the LT-based subband system is guaranteed with the shift-orthogonality of the transform matrices. Generally speaking, we can identify three types of lapped transforms:

- Transforms (or principal components in filter banks) that are designed to match with the statistical or spectral properties of the signal [1, 23].

- Lapped orthogonal transform (LOT) and its generalized version (GenLOT) that offers symmetric basis functions and DCT-based implementations [33, 60].

(a)   (b)
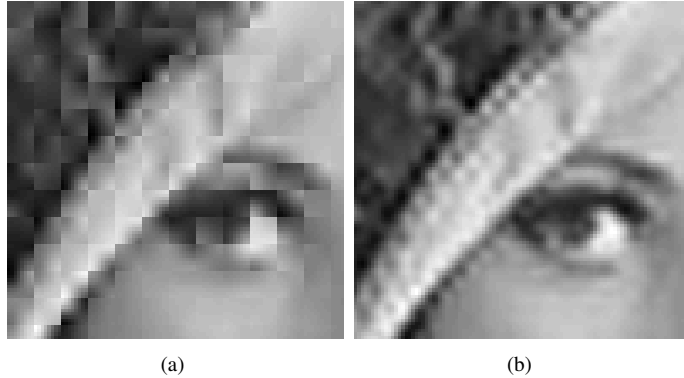
Figure 3.1: Detail of the famous Lena image first decomposed and then reconstructed from a reduced number ($4/64 = 0.0625$) of subband coefficients. (a) Discrete cosine transform. (b) Modulated lapped transform.
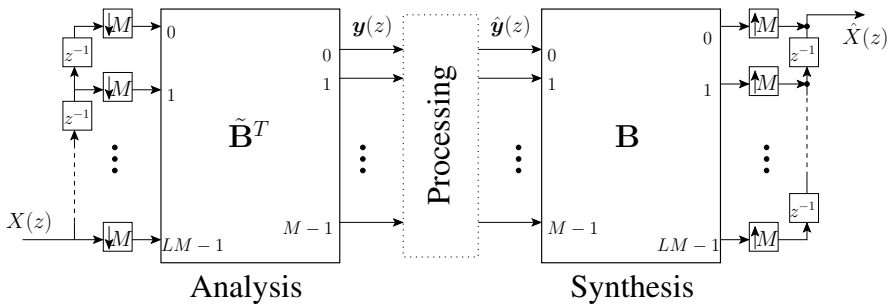
Figure 3.2: Lapped biorthogonal transform.

- Lapped transforms that are generated by modulating window function(s) with cosine sequence(s) [56, 58].

The third category is discussed in Chapter 6 where we combine the EMFB modulation options with LT window functions. In this chapter, we proceed with the LT theory. The shift-orthogonality concept given in [57] is modified to cover arbitrary-delay systems.

3.1 Subband Signal Processing with Lapped Transforms

3.1.1 Lapped Transform Matrices

Let us consider a subband system with lapped transforms in Figure 3.2. It consists of three parts: a forward transform that is applied on discrete-time signal $X(z)$, a subband processing stage, and then an inverse transform which reconstructs the output signal $\hat{X}(z)$. There is a certain resemblance with the polyphase representation of the filter banks and this is not coincidental. We will return to this matter after we take a closer look on the elements in the LT system.
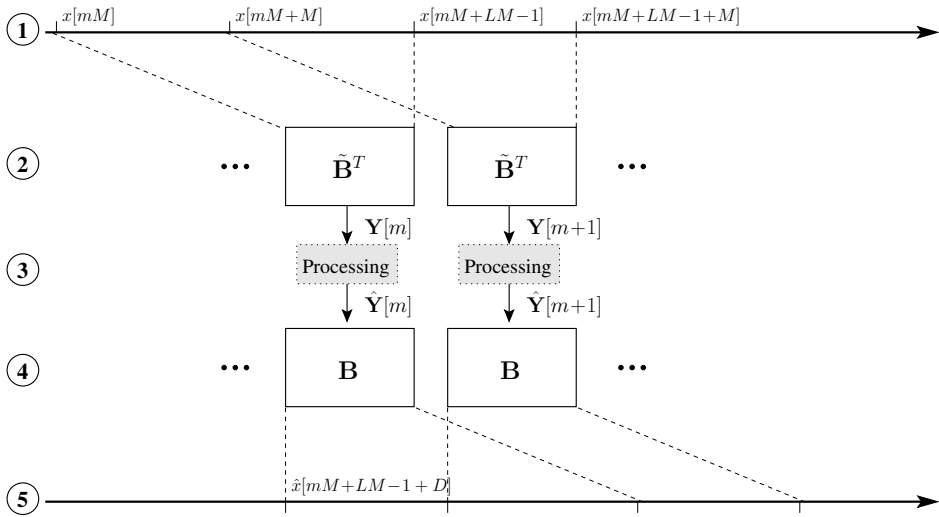
Figure 3.3: Schematic representation of the forward and inverse lapped transform pair applied on a signal $x[n]$.

Two matrices, $\tilde{\mathbf{B}}$ and $\mathbf{B}$ have dimensions $LM \times M$ and together they form the pair of lapped transform matrices. For the time being, the elements of the matrices and the (sub)signals are real-valued. *Forward transform matrix* $\tilde{\mathbf{B}}$ converts LM-length input data vectors in the M-length vectors of transform coefficients. The transform is computed each time when M fresh samples flow into the delay line. The columns of the matrix $\tilde{\mathbf{B}}$ are the *basis functions* of the forward LT. The basis function length is restricted to be an integer multiple of the number of subbands for mathematical tractability. Similar to the ELT, we define L to be an *overlap factor* of the lapped transforms with the following difference: L is now allowed to have odd positive integer values in addition to the corresponding even values. The subband processing part is not discussed in this context, even though the fundamental reason for signal transformations, of course, is based on the solving of the given problem more efficiently using transform-domain signals. The inverse transform matrix $\mathbf{B}$ converts the M subsignals from the processing stage into the output signal vector. The signal reconstruction is an overlap-add procedure because the inverse LT generates LM new samples based on every M-length transform domain coefficient vector. Similarly, with the forward transform, the columns of $\mathbf{B}$ are the basis functions of the inverse lapped transform.

In Figure 3.3 the time domain operation of the forward-inverse transform pair is visualized and it is seen that the term "lapped transform" is aptly chosen.

3.1.2 Polyphase Domain Analysis

The lapped transform system in Figure 3.2 is treated in the polyphase domain with the multirate analysis tools as discussed in Section 2.6.2. For this purpose, it is worth recalling that the causal polyphase decomposition of $X(z)$ is

$$\hat{X}(z) = z^{-(M-1)} \sum_{\ell=0}^{M-1} z^{-\ell} X_\ell(z^M),$$

39

where $X_\ell(z) = \sum_{m=-\infty}^{\infty} x[\ell + mM]\, z^{-m}$. The input signal polyphase vector $\boldsymbol{x}_{\mathrm{p}}(z)$ groups the M different phases $\boldsymbol{x}_{\mathrm{p}}(z) = \begin{bmatrix} X_0(z) & X_1(z) & \ldots & X_{M-1}(z) \end{bmatrix}^T$. Using this z-transform vector notation, the forward-inverse transform pair can be described as shown below. The numbered steps in this description correspond to the time-domain operations in Figure 3.3.

1. The input signal is buffered to LM-length vectors. Sampling window is overlapping because the decimation factor is less than buffer length. ($L \geq 2$ in lapped transforms, whereas $L = 1$ case corresponds to the conventional block transform.) The input vector $\boldsymbol{x}_{\mathrm{LT}}(z)$ can be expressed with M-length polyphase component vectors:

$$\boldsymbol{x}_{\mathrm{LT}}(z) = \left[\left(z^{-(L-1)}\, \boldsymbol{x}_{\mathrm{p}}^T(z) \right) \;\; \cdots \;\; \left(z^{-1}\, \boldsymbol{x}_{\mathrm{p}}^T(z) \right) \;\; \boldsymbol{x}_{\mathrm{p}}^T(z) \right]^T.$$

2. The forward transform $\tilde{\mathbf{B}}$ converts the L-length input vector to transform domain sub-signals $Y_k(z)$, $k = 0, 1, \ldots, M-1$. The sample rate of the subsignals is decreased by a factor of M compared with the rate of the input signal, i.e., critical sampling. When the subsignal vector is denoted with $\boldsymbol{y}(z) = [Y_0(z) \;\; Y_1(z) \;\; \ldots \;\; Y_{M-1}(z)]^T$, we can write

$$\begin{aligned}
\boldsymbol{y}(z) &= \tilde{\mathbf{B}}^T \boldsymbol{x}_{\mathrm{LT}}(z) = \begin{bmatrix} \tilde{\mathbf{B}}_0^T & \tilde{\mathbf{B}}_1^T & \ldots & \tilde{\mathbf{B}}_{L-1}^T \end{bmatrix} \\
&\quad \times \left[\left(z^{-(L-1)}\, \boldsymbol{x}_{\mathrm{p}}^T(z) \right) \;\; \cdots \;\; \left(z^{-1}\, \boldsymbol{x}_{\mathrm{p}}^T(z) \right) \;\; \boldsymbol{x}_{\mathrm{p}}^T(z) \right]^T \\
&= \left(\sum_{m=0}^{L-1} z^{-(L-1-m)} \tilde{\mathbf{B}}_m^T \right) \boldsymbol{x}_{\mathrm{p}}(z),
\end{aligned} \tag{3.1}$$

where $\tilde{\mathbf{B}}_m$ denotes for a partitioned $M \times M$ submatrix

$$\left[\tilde{\mathbf{B}}_m \right]_{i,k} = \left[\tilde{\mathbf{B}} \right]_{i+mM,k}$$

and $m = 0, 1, \ldots, L-1$.

3. The application-dependent subband algorithm processes the transform domain signals:

$$\boldsymbol{y}(z) \xrightarrow{\text{processing}} \hat{\boldsymbol{y}}(z).$$

4. The inverse transform matrix $\mathbf{B}$ is utilized in the first step of signal synthesis:

$$\hat{\boldsymbol{x}}_{\mathrm{LT}}(z) = \mathbf{B}\, \hat{\boldsymbol{y}}(z). \tag{3.2}$$

5. The LM-length intermediate vector output is overlap-added by shifts of M to recover the output signal $\hat{X}(z)$:

$$\begin{aligned}
\hat{X}(z) &= z^{-(M-1)} \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-(LM-1)} \end{bmatrix} \hat{\boldsymbol{x}}_{\mathrm{LT}}(z^M) \\
&= \mathbf{c}^T(z) \left(\sum_{n=0}^{L-1} z^{-nM} \mathbf{B}_n \right) \hat{\boldsymbol{y}}(z^M),
\end{aligned} \tag{3.3}$$

where $\mathbf{c}^T(z) = z^{-(M-1)} \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-(M-1)} \end{bmatrix}$. The $M \times M$ submatrices $\mathbf{B}_n$ of the above inverse transform are given by $\left[\mathbf{B}_n \right]_{i,k} = \left[\mathbf{B} \right]_{i+nM,k}$.

3.2 Shift-Biorthogonality with Arbitrary-Delay

The concept of LT shift-orthogonality has been considered in [57] with the requirement that the forward and inverse transforms are identical. The transform matrix is partitioned into $M \times M$ blocks and these submatrices are utilized in the definition for the shift-orthogonality, i.e., the terms for the invertible LTs. However, we want to generalize the concept to cover the arbitrary-delay case. How do we incorporate arbitrary-delay parameter D from Section 2.6.2 into the shift-orthogonality condition?

The LT-system in Figure 3.2 is causal and the overall delay $\tau = LM - 1 + D$ depends now on the length of the basis functions. The parameter D is limited within $D \in \{0, \pm 1, \ldots, \pm(L-1)M\}$. Negative values for D reduces the signal lag by $|D|$ samples with respect to the shift-orthogonal LT. Positive values have the opposite effect. Similar to the polyphase domain PR conditions for the M-channel filter banks, we write

$$\tau = LM - 1 + Mr + d,$$

where $r = \lfloor D/M \rfloor$ and $d = D - Mr$.

The "non-multiple of M" reconstruction delay has a slightly complicating effect to the shift-orthogonality; the identity matrix is generally split into two shift-instants. For this purpose, we define the matrix function $\boldsymbol{\delta}_C[\ell]$ where C and ℓ are integer parameters:

$$\boldsymbol{\delta}_C[\ell] = \delta[\ell - \lfloor C/M \rfloor] \, \mathbf{S}_{M \lfloor C/M \rfloor - C} + \delta[\ell - \lfloor C/M \rfloor - 1] \, \mathbf{S}_{M - C + M \lfloor C/M \rfloor}. \qquad (3.4)$$

The matrix $\mathbf{S}_c$ is an $M \times M$ shift-block matrix where the index c represents the position of the diagonal. Its formal definition is

$$[\mathbf{S}_c]_{n,m} = \delta[n, m - c] = \begin{cases} 1 & \text{if } n = m - c \\ 0 & \text{otherwise.} \end{cases} \qquad (3.5)$$

The substitution of the delay parameter D into the function $\boldsymbol{\delta}_C[\ell]$ gives then

$$\boldsymbol{\delta}_D[\ell] = \delta[\ell - r] \, \mathbf{S}_{-d} + \delta[\ell - r - 1] \, \mathbf{S}_{M-d}. \qquad (3.6)$$

Since $d \in \{0, 1, \ldots, M - 1\}$, the ones reside on the lower diagonals of matrix $\mathbf{S}_{-d}$. In the other matrix, $\mathbf{S}_{M-d}$, the ones can be found on $(M - d)$th upper diagonal (except when $d = 0$ which corresponds to a zero matrix).

The second definition provided here is a LT matrix shift function $\left(\mathbf{B} \circledast \tilde{\mathbf{B}}\right)[\ell]$ whose arguments are $\mathbf{B}$, $\tilde{\mathbf{B}}$, end ℓ. The matrix partitioning to $M \times M$ subblocks is needed again:

$$\tilde{\mathbf{B}}^T = \begin{bmatrix} \tilde{\mathbf{B}}_0^T & \tilde{\mathbf{B}}_1^T & \cdots & \tilde{\mathbf{B}}_{L-1}^T \end{bmatrix}$$
$$\mathbf{B}^T = \begin{bmatrix} \mathbf{B}_0^T & \mathbf{B}_1^T & \cdots & \mathbf{B}_{L-1}^T \end{bmatrix}.$$

Parameter ℓ is used to control the number of terms in the sum and the side of the shift-function:

$$\left(\mathbf{B} \circledast \tilde{\mathbf{B}}\right)[\ell] = \begin{cases} \displaystyle\sum_{i=0}^{L-1+\ell} \mathbf{B}_i \tilde{\mathbf{B}}_{i-\ell}^T & -L < \ell \leq 0 \\ \displaystyle\sum_{i=0}^{L-1-\ell} \mathbf{B}_{i+\ell} \tilde{\mathbf{B}}_i^T & 0 < \ell < L. \end{cases} \qquad (3.7)$$

The above function manipulates the transform matrices $\tilde{\mathbf{B}}$ and $\mathbf{B}$ by sliding them relative to each other by shifts of M. Those submatrices that are on "the region of overlap" are included in the sum. When $\ell = 0$, there is a maximum number of terms in the sum and, conversely, the summation reduces to contain a single term when $\ell = \pm(L-1)$. From one viewpoint, we may think the matrix shift-function as a convolution type of operation with matrix blocks. The latter viewpoint emerges from the polyphase domain PR proof for the proposed LT system to be done below.

Now we are ready to make our claim:

The pair of transform matrices $\tilde{\mathbf{B}}$ and $\mathbf{B}$ are shift-biorthogonal with arbitrary-delay D, $D \in \{0, \pm 1, \ldots, \pm(L-1)M\}$ when the condition

$$\left(\mathbf{B} \circledast \tilde{\mathbf{B}}\right)[\ell] = \boldsymbol{\delta}_D[\ell] \tag{3.8}$$

is satisfied for $\ell = 0, \pm 1, \ldots, \pm(L-1)$.

Supposedly, it is our responsibility to prove next that the PR follows from the arbitrary-delay shift-biorthogonality.

The PR condition states that when the subsignals are left intact and the lapped transform is invertible, the signal is only delayed in the system

$$\hat{X}(z) = z^{-\tau}\, X(z). \tag{3.9}$$

Here, $\hat{X}(z)$ is given by (3.3) and the purpose is to derive such expression where the shift-biorthogonality condition can be applied. First we substitute the subsignal equation (3.1) into this expression which gives

$$\hat{X}(z) = \mathbf{c}^T(z) \left(\sum_{n=0}^{L-1} z^{-nM}\mathbf{B}_n\right) \left(\sum_{m=0}^{L-1} z^{-(L-1-m)M}\tilde{\mathbf{B}}_m^T\right) \boldsymbol{x}_{\mathrm{p}}(z^M)$$

$$= \mathbf{c}^T(z)\, z^{-(L-1)M} \left(\sum_{n=0}^{L-1}\sum_{m=0}^{L-1} z^{(m-n)M}\, \mathbf{B}_n\, \tilde{\mathbf{B}}_m^T\right) \boldsymbol{x}_{\mathrm{p}}(z^M).$$

We are getting closer to the left hand side (LHS) of the shift-biorthogonality condition but the summation formula needs to be opened. Next the double summation term is considered separately and the matrix terms with the same delay value are gathered

$$z^{-(L-1)M}\sum_{n=0}^{L-1}\sum_{m=0}^{L-1} z^{(m-n)M}\, \mathbf{B}_n\, \tilde{\mathbf{B}}_m^T = \mathbf{B}_0\tilde{\mathbf{B}}_{L-1}^T + z^{-M}\left(\mathbf{B}_0\tilde{\mathbf{B}}_{L-2}^T + \mathbf{B}_1\tilde{\mathbf{B}}_{L-1}^T\right) + \cdots$$

$$+ z^{-(L-1)M}\left(\mathbf{B}_0\tilde{\mathbf{B}}_0^T + \mathbf{B}_1\tilde{\mathbf{B}}_1^T + \cdots + \mathbf{B}_{L-1}\tilde{\mathbf{B}}_{L-1}^T\right) + \cdots$$

$$+ z^{-(2L-3)M}\left(\mathbf{B}_{L-2}\tilde{\mathbf{B}}_0^T + \mathbf{B}_{L-1}\tilde{\mathbf{B}}_1^T\right) + z^{-(2L-2)M}\mathbf{B}_{L-1}\tilde{\mathbf{B}}_0^T.$$

This can be written compactly with the shift-function definition (3.7)

$$z^{-(L-1)M}\sum_{n=0}^{L-1}\sum_{m=0}^{L-1} z^{(m-n)M}\, \mathbf{B}_n\, \tilde{\mathbf{B}}_m^T = \sum_{\ell=-(L-1)}^{L-1} z^{-(L-1+\ell)M}\left(\mathbf{B} \circledast \tilde{\mathbf{B}}\right)[\ell].$$

The assumption states that the LT is invertible with arbitrary-delay D when the shift-function is equal to $\boldsymbol{\delta}_D[\ell]$ as ℓ ranges from $-(L-1)$ to $(L-1)$. The function $\boldsymbol{\delta}_D[\ell]$ is defined by

equation (3.6) so that non-zero matrices are generated with at most two values of ℓ. We apply now this assumption and it yields

$$\hat{X}(z) = \mathbf{c}^T(z) \left(\sum_{\ell=r}^{r+1} z^{-(L-1+\ell)M} \boldsymbol{\delta}_D[\ell] \right) \boldsymbol{x}_\mathrm{p}(z^M)$$

$$= \mathbf{c}^T(z)\, z^{-(L-1)M} \left(z^{-rM} \mathbf{S}_{-d} + z^{-(r+1)M} \mathbf{S}_{M-d} \right) \boldsymbol{x}_\mathrm{p}(z^M)$$

$$= z^{-(L-1+r)M}\, \mathbf{c}^T(z) \begin{bmatrix} \mathbf{0} & z^{-M} \mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix} \boldsymbol{x}_\mathrm{p}(z^M).$$

It can be observed that the shift-biorthogonality definition produced a pseudo-circulant matrix similar to the polyphase domain PR condition for critically sampled filter banks. The proof is completed by writing out the polyphase components of $X(z)$:

$$\hat{X}(z) = z^{-(L-1+r)M}\, z^{-(M-1)} \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-(M-1)} \end{bmatrix}$$

$$\times \Big[\underbrace{z^{-M} X_{M-d}(z^M) \; \cdots \; z^{-M} X_{M-1}(z^M)}_{\#d} \; \underbrace{X_0(z^M) \; \cdots \; X_{M-d-1}(z^M)}_{\#(M-d)} \Big]^T$$

$$= z^{-(LM-1+Mr)} \Bigg(\sum_{m=M-d}^{M-1} z^{-(m-(M-d))}\, z^{-M} X_m(z^M)$$

$$+ \sum_{n=0}^{M-d-1} z^{-(n+d)} X_n(z^M) \Bigg)$$

$$= z^{-(LM-1+Mr)} \Bigg(z^{-d} \sum_{m=M-d}^{M-1} z^{-m} X_m(z^M) + z^{-d} \sum_{n=0}^{M-d-1} z^{-n} X_n(z^M) \Bigg)$$

$$= z^{-(LM-1+Mr+d)} \underbrace{\sum_{\ell=0}^{M-1} z^{-\ell} X_\ell(z^M)}_{X(z),\ \text{cf. equation (2.7)}} = z^{-\tau} X(z).$$

Then we have proved that the arbitrary-delay shift-biorthogonality leads to a PR subband system. Again we stress that the overall signal delay in this forward-inverse LT pair is controlled with two parameters: the overlap factor L that defines the length of the basis functions and the arbitrary-delay design parameter $D = Mr + d$.

3.3   Connection with Filter Banks

The lapped biorthogonal transform with arbitrary-delay can be classified as a filter bank with critical sampling. The exact connection is obtained by equating the FB impulse responses, $h_k[n]$ and $f_k[n]$, with the elements of the transform matrices:

$$h_k[n] = \left[\tilde{\mathbf{B}} \right]_{LM-1-n,k} \tag{3.10}$$

$$f_k[n] = \left[\mathbf{B} \right]_{n,k}, \tag{3.11}$$

where LM is the filter length, $n = 0, 1, \ldots, LM-1$, and the subchannel index $k = 0, 1, \ldots, M-1$.
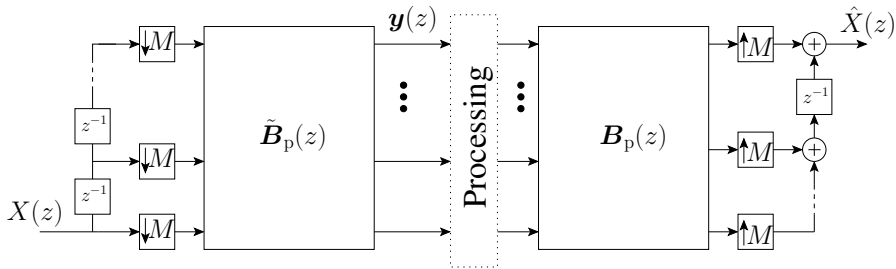
Figure 3.4: Polyphase domain representation of the lapped transform.

Skeptically-minded persons would question the suspiciously simple relation between LTs and M-channel critically sampled filter banks. We try to convince them (and the author as well) by showing that the connection leads to the PR. For this purpose, we recall from Section 2.6.2 equation (2.21) for the analysis polyphase matrix $\boldsymbol{H}_{\mathrm{p}}(z)$ and utilize the fact that $N+1 = LM$. Next the coefficients of the forward transform matrix (cf. equation (3.10)) are substituted into this polyphase equation and manipulation of the indices yields

$$
\begin{aligned}
\left[\boldsymbol{H}_{\mathrm{p}}(z)\right]_{k,\ell} &= \sum_{m=0}^{L-1} h_k[M-1-\ell+mM]\, z^{-m} = \sum_{m=0}^{L-1} z^{-m} \left[\tilde{\mathbf{B}}\right]_{LM-1-(M-1-\ell+mM),k} \\
&= \sum_{m'=0}^{L-1} z^{-(L-1-m')} \left[\tilde{\mathbf{B}}\right]_{\ell+m'M,k} = \sum_{m'=0}^{L-1} z^{-(L-1-m')} \left[\tilde{\mathbf{B}}_{m'}^T\right]_{k,\ell}.
\end{aligned}
$$

The final form in the above equation has been obtained by applying the definition for matrix transpose, i.e., $\left[\mathbf{A}\right]_{\ell,k} = \left[\mathbf{A}^T\right]_{k,\ell}$ together with the matrix partitioning. Similarly, the synthesis polyphase matrix is given by

$$
\begin{aligned}
\left[\boldsymbol{F}_{\mathrm{p}}(z)\right]_{\ell,k} &= \sum_{n=0}^{L-1} f_k[\ell+nM]\, z^{-n} = \sum_{n=0}^{L-1} z^{-n} \left[\mathbf{B}\right]_{\ell+nM,k} \\
&= \sum_{n=0}^{L-1} z^{-n} \left[\mathbf{B}_n\right]_{\ell,k}.
\end{aligned}
$$

Alternatively, we can directly denote the polyphase matrices of the forward and inverse transforms with

$$
\tilde{\boldsymbol{B}}_{\mathrm{p}}(z) = \sum_{m=0}^{L-1} z^{-(L-1-m)}\, \tilde{\mathbf{B}}_m^T \tag{3.12}
$$

$$
\boldsymbol{B}_{\mathrm{p}}(z) = \sum_{n=0}^{L-1} z^{-n}\, \mathbf{B}_n, \tag{3.13}
$$

respectively. The LT system with this polyphase matrix representation is shown in Figure 3.4. Hence, the polyphase matrices $\tilde{\boldsymbol{B}}_{\mathrm{p}}(z)$ and $\boldsymbol{B}_{\mathrm{p}}(z)$ can be substituted to equation (2.24) to get

the following desired output of the forward/inverse transform system

$$\hat{X}(z) = \mathbf{c}^T(z)\,\tilde{\boldsymbol{B}}_{\mathrm{p}}(z^M)\boldsymbol{B}_{\mathrm{p}}(z^M)\,\boldsymbol{x}_{\mathrm{p}}(z^M)$$

$$= \mathbf{c}^T(z)\left(\sum_{n=0}^{L-1} z^{-nM}\mathbf{B}_n\right)\left(\sum_{m=0}^{L-1} z^{-(L-1-m)M}\tilde{\mathbf{B}}_m^T\right)\boldsymbol{x}_{\mathrm{p}}(z^M)$$

$$= \mathbf{c}^T(z)\,z^{-(L-1)M}\left(\sum_{n=0}^{L-1}\sum_{m=0}^{L-1} z^{(m-n)M}\,\mathbf{B}_n\,\tilde{\mathbf{B}}_m^T\right)\boldsymbol{x}_{\mathrm{p}}(z^M)$$

$$= \cdots = z^{-(LM-1+Mr+d)}X(z),$$

on the grounds of Section 3.2. The polyphase filter equations produce thus the same RHS result as the direct analysis of the LT system; the signal is perfectly reconstructed and delayed by $\tau = LM - 1 + Mr + d$ provided that the arbitrary-delay shift-biorthogonality is satisfied.

3.4   Shift-Orthogonal Lapped Transforms

Paraunitary FBs are a special case of biorthogonal filter banks where analysis filters are obtained from the synthesis filters via time-reversing. From the connection between LTs and filter banks we would expect that similar results emerge from the LT theory. For this purpose, a lapped transform matrix $\mathbf{P}$ is used to distinguish the plain shift-orthogonality from the more general arbitrary-delay shift-biorthogonal transforms. It is assumed that the forward and inverse transform matrices are equal: $\tilde{\mathbf{P}} = \mathbf{P}$. The definition requires that the transform matrix to be represented with $M \times M$ submatrices $\mathbf{P}_m$:

$$\left[\mathbf{P}_m\right]_{i,k} = \left[\mathbf{P}\right]_{i+mM,k}$$

where $i, k = 0, 1, \ldots, M - 1$ and $m = 0, 1, \ldots, L - 1$. We can express the necessary and sufficient PR conditions with these partitioned matrices [57]:

The lapped transform matrix $\mathbf{P}$ is *shift-orthogonal* when the condition

$$\sum_{m=0}^{L-1-\ell} \mathbf{P}_m\mathbf{P}_{m+\ell}^T = \delta[\,\ell\,]\mathbf{I} \tag{3.14}$$

is satisfied for $\ell = 0, 1, \ldots, L - 1$.

The proof that the shift-orthogonality leads to the PR can be done with polyphase component matrices:

$$\tilde{\boldsymbol{P}}_{\mathrm{p}}(z) = \sum_{m=0}^{L-1} z^{-(L-1-m)}\,\mathbf{P}_m^T \tag{3.15}$$

$$\boldsymbol{P}_{\mathrm{p}}(z) = \sum_{n=0}^{L-1} z^{-n}\,\mathbf{P}_n. \tag{3.16}$$

When $\tilde{\boldsymbol{P}}_{\mathrm{p}}(z)$ and $\boldsymbol{P}_{\mathrm{p}}(z)$ are substituted to equation (2.24), we may proceed as before until the point where the shift-orthogonality should be utilized. Then we need also the "remaining

half" of the shift-function. It causes us no problem because transposing the LHS and RHS of equation (3.14) gives

$$\left(\sum_{m=0}^{L-1-\ell} \mathbf{P}_m \mathbf{P}_{m+\ell}^T \right)^T = \delta[\ell]\mathbf{I}^T$$

$$\Leftrightarrow \qquad \sum_{m=0}^{L-1-\ell} \mathbf{P}_{m+\ell} \mathbf{P}_m^T = \delta[\ell]\mathbf{I}$$

Then we can equivalently write

$$\left(\mathbf{P} \circledast \mathbf{P} \right)[\ell] = \begin{cases} \sum_{i=0}^{L-1+\ell} \mathbf{P}_i \mathbf{P}_{i-\ell}^T & -L < \ell \leq 0 \\ \sum_{i=0}^{L-1-\ell} \mathbf{P}_{i+\ell} \mathbf{P}_i^T & 0 < \ell < L \end{cases} \tag{3.17}$$

and

$$\left(\mathbf{P} \circledast \mathbf{P} \right)[\ell] = \delta[\ell]\mathbf{I}. \tag{3.18}$$

Thus the polyphase domain equations and the application of the shift-orthogonality assumption produces

$$\hat{X}(z) = \mathbf{c}^T(z) \tilde{\boldsymbol{P}}_{\mathrm{p}}(z^M) \boldsymbol{P}_{\mathrm{p}}(z^M) \, \boldsymbol{x}_{\mathrm{p}}(z^M)$$
$$= \cdots = z^{-(LM-1)} X(z).$$

The forward-inverse reconstruction delay τ of the causal LT system utilizing the shift-orthogonal matrix $\mathbf{P}$ is then $\tau = LM - 1$.

Chapter 4

Biorthogonal Modulated Filter Banks with Perfect Reconstruction

An exponentially modulated filter bank is a critically sampled subband system for processing of complex-valued signals [11, 100]. We can view it as an extension of odd-stacked CMFB to the complex number domain. Conversely, the CMFB is a special case of the EMFB in the real number domain. The filter banks discussed in this chapter belong to a class of critically sampled modulated filter banks generated from a single real-valued prototype filter. Our previous research restricted to a paraunitary EMFB, where the prototype filter part is implemented with orthogonal rotations known as butterfly matrices [57]. The symmetry of the prototype filter impulse response $h[n]$ is the key characteristic of the paraunitary MFBs and the length of the prototype defines the analysis-synthesis reconstruction delay. The main purpose of this chapter is to generalize the initial EMFB concept to allow an arbitrary length for the prototype filter and, consequently, to introduce a parameter to adjust the reconstruction delay. This relaxation leads to biorthogonal filter banks where the analysis and synthesis subfilters are not restricted to have the time-reversal connection as in paraunitary FBs.

A thorough research on critically sampled biorthogonal modulated filter banks has been published by Heller, Karp, and Nguyen [42]. They have presented the polyphase domain analysis of DCT-III/IV, DCT-I/II, and MDFT types of modulated filter banks and it has been shown that all these three cases give similar polyphase domain PR requirements for $h[n]$ in the filter bank design. We begin this chapter with a re-read of the arbitrary-length, arbitrary-delay type DCT-III/IV CMFB. To avoid mixing implementation terminology with the basic MFB theory, it is more convenient to refer these FBs as odd-stacked CMFBs according to [72]. In addition, we slightly modify the phases of the cosine modulating sequences to support the formulation of the EMFB concept.

The perfect-reconstruction property of the odd-stacked CMFB system is addressed in the polyphase domain. One objective is to find a set of constraint functions for the prototype filter so that they stem from the generating cosine sequences. Furthermore, the catch is to show that the same prototype suits for the sine-modulated FB as well and guarantees the PR. On the basis of the introduction, the reader might anticipate that this leads to an arbitrary-delay FB for complex-valued signals, where CMFB/SMFBs are utilized as building blocks (cf. Figure 1.1).

47

We refer to this kind of a FB as an odd-stacked biorthogonal EMFB. Once we are reassured that the EMFB PR follows the properties of its subcomponents, we modify the system into an even-stacked configuration. Since the modifications are elementary reversible operations, the PR is preserved in the even-stacked EMFB. Furthermore, we point out that the even-stacked CMFB can be derived from the former when the analyzed signal is purely real-valued.

Though the focus is on critically sampled subband systems, we side-step occasionally into subsignal oversampling. The EMFB subsignals can be oversampled by a factor of two to provide additional phase information about the frequency components. This feature is interesting for equalization problems where the subfilter orthogonality is lost. Furthermore, the FFT-based implementation for the EMFB to be considered in Section 7.2 produces this subsample "surplus" with little additional effort.

4.1 Biorthogonal Odd-Stacked CMFB with Perfect-Reconstruction

4.1.1 Definitions for Analysis and Synthesis Filters

Now it is time for the M-channel FB theory developed in Section 2.6 to become into fruition. We reduce the level of abstraction and bring the modulation equations into play. The arbitrary-delay, arbitrary-length CMFB is an M-channel critically sampled real-valued filter banks. It is furthermore characterized by the following features:

- The subfilter length is $N + 1$ where N is the subfilter order.

- Analysis and synthesis subfilters are generated from a single low-pass prototype filter $h[n]$ by modulating cosine sequences.

- The analysis-synthesis reconstruction delay is $\tau = N + D$, where D is an adjustable design parameter: $D \in \{0, \pm 1, \ldots, \pm(N - M + 1)\}$. If $D \neq 0$, then the prototype filter is non-symmetric and the resulting CMFB is biorthogonal. If $D = 0$, then the prototype filter is symmetric and the resulting CMFB is paraunitary.

- The prototype filter satisfies the perfect-reconstruction constraints as given in Section 4.1.3.

- Subfilter center frequencies $\omega_k = (2k + 1)\pi/(2M)$ divide the frequency range $[0, \pi]$ into M uniformly spaced subbands, where k is the subchannel index.

Due to the biorthogonality, we have now different modulation sequences compared with, say, the ELT type of modulation. The arbitrary-delay parameter is included in the subfilter equations as a correction term for the modulation phase. The analysis and synthesis subfilters are defined as

$$h_k^c[n] = 2\,h[n] \cos\left[\left(n - \frac{N + D + M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \tag{4.1}$$

$$f_k^c[n] = 2\,h[n] \cos\left[\left(n - \frac{N + D - M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right], \tag{4.2}$$

where $n = 0, 1, \ldots, N$ and $k = 0, 1, \ldots, M - 1$. Unless a parametric window function is used for generating $h[n]$, the prototype filter design leads to a non-linear optimization problem as

we shall see in Chapter 5. In this case, the PR requirement for the subband system forces the use of a set of constraining equations for the prototype filter in its optimization. The remaining degrees of freedom are typically used to maximize the stopband attenuation of the prototype filter. This gives less spectral overlap among the subfilters. Moreover, as we shall see in Chapter 5, depending on the FB under consideration, there are some additional constraints for the DC-leakage, thereby further decreasing the degrees of freedom for maximizing the stopband attenuation of the prototype filter.

In this discourse, it is more useful to put the distracting design issues aside for the time being. Hence, we proceed into the polyphase domain analysis of the odd-stacked biorthogonal CMFB.

4.1.2 Biorthogonal CMFB in the Polyphase Domain

In this analysis, we represent each subfilter of the biorthogonal CMFB as M-phase components. These elements constitute the following $M \times M$ analysis and synthesis polyphase matrices:

$$\left[\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z)\right]_{k,\ell} = \sum_{m} h_k^{\mathrm{c}}[M-1-\ell+mM]\, z^{-m}$$

$$\left[\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\right]_{\ell,k} = \sum_{m} f_k^{\mathrm{c}}[\ell+mM]\, z^{-m}.$$

According to equations (4.1) and (4.2), all subfilters share the prototype filter as a common term. It is thus advantageous to state the PR conditions for these polyphase matrices in such a manner that they are formed by using the decomposition, where the prototype filter part is separated from the modulation block.

The reasoning why the polyphase matrices can be partitioned comes from the following sketch: When the FBs were regarded as lapped transforms in Section 3.3, we had the connection $\left[\mathbf{B}\right]_{n,k} = f_k^{\mathrm{c}}[n]$ between the inverse lapped transform matrix and the synthesis subfilters (assuming that $N+1 = LM$). This can be equivalently written as

$$\mathbf{B} = \mathrm{diag}\{h[0], h[1], \ldots, h[LM-1]\}\, \boldsymbol{\Phi},$$

where

$$\left[\boldsymbol{\Phi}\right]_{n,k} = 2\cos\left[\left(n - \frac{N+D-M}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right].$$

With given parameters, the columns of the above cosine matrix are $4M$ periodic and this can be exploited:

$$\mathbf{B} = \begin{bmatrix} \mathbf{H}_0 & & & \\ & \mathbf{H}_1 & & \\ & & \mathbf{H}_2 & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}^{\mathrm{c}} \\ -\boldsymbol{\Phi}^{\mathrm{c}} \\ \boldsymbol{\Phi}^{\mathrm{c}} \\ \vdots \end{bmatrix},$$

where $\mathbf{H}_i = \mathrm{diag}\{h[2iM], h[2iM+1], \ldots, h[2iM+2M-1]\}$ and $\left[\boldsymbol{\Phi}^{\mathrm{c}}\right]_{n,k} = \left[\boldsymbol{\Phi}\right]_{n,k}$ when we let $n = 0, 1, \ldots, 2M-1$. The polyphase matrix decomposition below extends this concept to cover arbitrary-length subfilters.
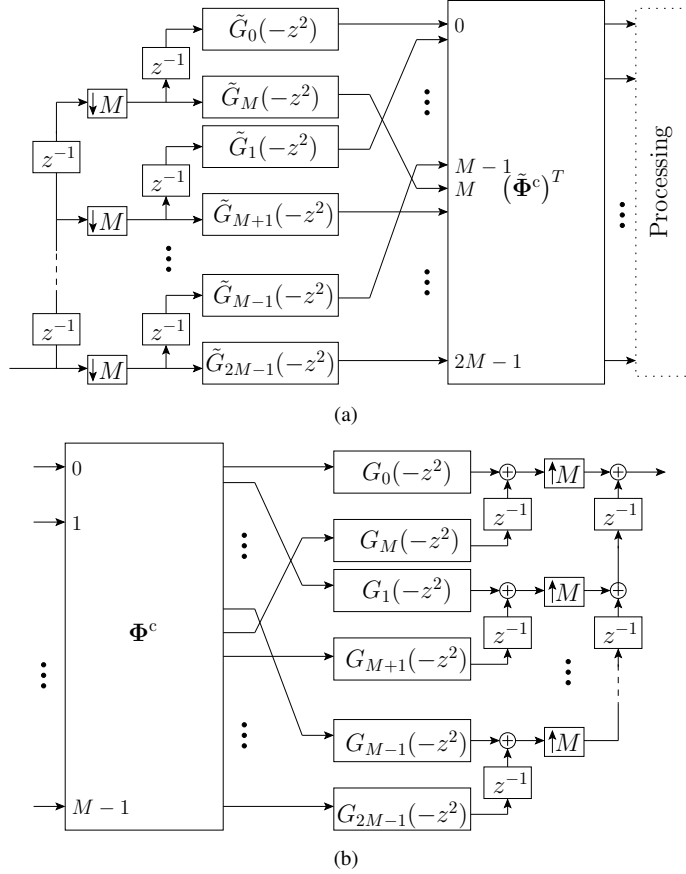
Figure 4.1: Odd-stacked CMFB decomposed into $2M$-length array of polyphase filters corresponding to the prototype and matching modulation blocks: (a) Analysis part. (b) Synthesis part.

To keep the story short and relying on the above intuitive sketch, we bluntly begin by defining the elements of the $2M \times M$ modulation matrices as

$$\left[\tilde{\mathbf{\Phi}}_c\right]_{n,k} = 2 \cos\left[\left(2M - 1 - n - \frac{N + D + M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \tag{4.3}$$

$$\left[\mathbf{\Phi}_c\right]_{n,k} = 2 \cos\left[\left(n - \frac{N + D - M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right], \tag{4.4}$$

where the time index is reversed on the analysis side to the polyphase matrix definition. Then we sort the prototype filter coefficients into $2M$-phase components to match with the cosine modulation

$$\tilde{G}_\ell(z) = \sum_m h[2M - 1 - \ell + 2mM]z^{-m} \tag{4.5}$$

$$G_\ell(z) = \sum_m h[\ell + 2mM], \tag{4.6}$$

where $\ell = 0, 1, \ldots, 2M - 1$. The odd-stacked cosine modulation is $4M$-periodic and, as explained above (with a certain amount of hand waving), the signs of the odd m terms are reversed. We arrive at a compact notation when the sign is embedded in definition as follows:

$$\sum_m (-1)^m h[\ell + 2mM] z^{-2m} = \sum_m h[\ell + 2mM](-z^2)^{-m} = G_\ell(-z^2)$$

and

$$\sum_m (-1)^m h[2M - 1 - \ell + 2mM] z^{-2m} = \tilde{G}_\ell(-z^2).$$

These are the elements of the $2M \times 2M$ diagonal matrices $\left[\tilde{\boldsymbol{G}}(z)\right]_{\ell,\ell} = \tilde{G}_\ell(-z^2)$ and $\left[\boldsymbol{G}(z)\right]_{\ell,\ell} = G_\ell(-z^2)$, which produce the CMFB polyphase components when multiplied with the modulation matrices - almost. The $2M$-phase components must be interleaved into the M-phase polyphase matrix. This leads to the decomposition

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) \, \boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) = \begin{bmatrix} \mathbf{I} & z^{-1}\mathbf{I} \end{bmatrix} \boldsymbol{G}(z) \boldsymbol{\Phi}_{\mathrm{c}} \tilde{\boldsymbol{\Phi}}_{\mathrm{c}}^T \tilde{\boldsymbol{G}}(z) \begin{bmatrix} z^{-1}\mathbf{I} \\ \mathbf{I} \end{bmatrix} \tag{4.7}$$

for the biorthogonal CMFB. A flowgraph corresponding to equation (4.7) is shown in Figure 4.1 where the input and output delay lines/sampling rate alteration matches with the M-phase decomposition discussed in Section 2.6.2. The block-transform-based implementation structure can be derived from the same decomposition, but the algorithm is easier to conceptualize when the input/output buffers are extended to have $2M - 1$ delay elements, as will be seen in Section 7.1.

4.1.3   PR Restrictions for Biorthogonal CMFB Prototype Filter

What are the necessary conditions for the arbitrary-delay biorthogonal CMFB having reconstruction delay $\tau = N + D$ to be a PR system? According to the definition (2.25), it is required that

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) \, \boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) = z^{-s+1} \begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix} \tag{4.8}$$

holds when $s = \lfloor (\tau + 1)/M \rfloor$ and $d = \tau - Ms + 1$. The LHS of the above equation is replaced next with the decomposition given in equation (4.7), where the matrix product is written in terms of the prototype filter polyphase components and modulation matrices. It is apparent that the product of modulation matrices depends solely on the parameters M, N, and D. Then its contribution to the analysis-synthesis decomposition should be replaced with something simpler - a nice and friendly diagonal matrix would do fine.

We make an educated guess that the orthogonality of the roots of unity [98] might be handy; that is

$$\sum_{k=0}^{M-1} W_M^{kn} = \begin{cases} M & n = \lambda M, \lambda \in \mathbb{Z} \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$

As a matter of fact, we are more interested in the sum of cosines $S_\alpha = 2 \sum_{k=0}^{M-1} \cos\left(\frac{2\pi(k+1/2)\alpha}{2M}\right)$

which simplifies based on (4.9) to

$$S_\alpha = \sum_{k=0}^{M-1} \left(W_{2M}^{-(k+1/2)\alpha} + W_{2M}^{(k+1/2)\alpha} \right) = \sum_{k=0}^{2M-1} W_{2M}^{-(k+1/2)\alpha}$$

$$= W_{2M}^{-\alpha/2} \sum_{k=0}^{2M-1} W_{2M}^{-k\alpha} = \begin{cases} (-1)^\lambda 2M & \alpha = \lambda 2M \\ 0 & \alpha \neq \lambda 2M. \end{cases} \tag{4.10}$$

Let us then write open the product $\mathbf{\Phi}^c \left(\tilde{\mathbf{\Phi}}^c \right)^T$ that utilizes the element-wise definitions given in equations (4.3) and (4.4). Now the n,mth element of the product matrix is

$$\left[\mathbf{\Phi}^c \left(\tilde{\mathbf{\Phi}}^c \right)^T \right]_{n,m} = \sum_{k=0}^{M-1} \left[\mathbf{\Phi}^c \right]_{n,k} \left[\tilde{\mathbf{\Phi}}^c \right]_{m,k}$$

$$= 4 \sum_{k=0}^{M-1} \cos \left[\left(n - \frac{N+D-M}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

$$\times \cos \left[\left(2M - 1 - m - \frac{N+D+M}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

$$= \sum_{k=0}^{M-1} \left(W_{2M}^{-(k+1/2)n'} + W_{2M}^{(k+1/2)n'} \right) \times \left(W_{2M}^{-(k+1/2)m'} + W_{2M}^{(k+1/2)m'} \right)$$

$$= \sum_{k=0}^{M-1} W_{2M}^{-(k+1/2)(n'-m')} + W_{2M}^{(k+1/2)(n'-m')} + W_{2M}^{-(k+1/2)(n'+m')} + W_{2M}^{(k+1/2)(n'+m')}$$

$$= S_{n'-m'} + S_{n'+m'}$$

with short-hand notations $n' = n - (N+D-M)/2$ and $m' = 2M-1-m-(N+D+M)/2$. Then, we consider the above two sums separately. According to (4.10),

$$S_{n'-m'} = \begin{cases} (-1)^\lambda \, 2M & \text{if } n + m - M + 1 = \lambda 2M \\ 0 & \text{otherwise.} \end{cases}$$

Due to the fact that matrix product has dimensions $2M \times 2M$, the range for both n and m is $0, 1, \ldots, 2M - 1$ and the condition splits in two parts. If we denote the matrix formed with $S_{n'-m'}$ as $\hat{\mathbf{J}}$, it is actually a constant matrix with elements

$$\left[\hat{\mathbf{J}} \right]_{n,m} = \begin{cases} (-1)^\lambda \, 2M & \text{if } n + m - M + 1 = 0, \quad (\lambda = 0) \\ (-1)^\lambda \, 2M & \text{if } n + m - M + 1 = 2M, \quad (\lambda = 1) \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} 2M & \text{if } n = M - 1 - m \\ -2M & \text{if } n = 3M - 1 - m \\ 0 & \text{otherwise.} \end{cases}$$

A compact notation for $\hat{\mathbf{J}}$ is then given with reverse identity matrices

$$\left[\mathbf{J}_M \right]_{n,m} = \delta[n, M - 1 - m] = \begin{cases} 1 & \text{if } n = M - 1 - m \\ 0 & \text{otherwise,} \end{cases}$$

where $n, m = 0, 1, \ldots, M - 1$. The $S_{n'-m'}$ part of the modulating matrix product is then

$$\hat{\mathbf{J}} = 2M \begin{bmatrix} \mathbf{J}_M & \mathbf{0} \\ \mathbf{0} & -\mathbf{J}_M \end{bmatrix}. \tag{4.11}$$

The second sum $S_{n'+m'} = S_{n-m+2M-1-(N+D)}$ is somewhat trickier as it contains the reconstruction delay parameter. We start the simplification procedure by replacing $\tau + 1$ with $2Mr + d'$ where $r = \lfloor (\tau + 1)/(2M) \rfloor$ and $d' = \tau - 2Mr + 1$. Equation (4.10) then becomes

$$S_{n'-m'} = \begin{cases} (-1)^\lambda \, 2M & \text{if } n - m + 2M - 2rM - d' = \lambda 2M \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} (-1)^\lambda \, 2M & \text{if } n - m - d' = (\lambda + r - 1)2M \\ 0 & \text{otherwise.} \end{cases}$$

The multiple of $2M$ terms in the modulus function contribute for the sign reversion only. The non-zero elements of the matrix $\hat{\mathbf{I}}$ (being the $2M \times 2M$ matrix corresponding to the sum equation $S_{n'+m'}$ when the range of n and m is limited accordingly) are selected together with n, m, and d'. Again two possibilities are considered which land at the $2M \times 2M$ target zone:

$$\left[\hat{\mathbf{I}}\right]_{n,m} = \begin{cases} (-1)^\lambda \, 2M & \text{if } n - m - d' = 0, \quad (\lambda = -r + 1) \\ (-1)^\lambda \, 2M & \text{if } n - m - d' = -2M, \quad (\lambda = -r) \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} (-1)^{r-1} \, 2M & \text{if } n = m + d \\ (-1)^r \, 2M & \text{if } n = m + d' - 2M \\ 0 & \text{otherwise.} \end{cases}$$

The subcases contain two shift matrices that we discussed in Section 3.2. The definition for $\mathbf{S}_c$ is

$$[\mathbf{S}_c]_{n,m} = \delta[n, m - c] = \begin{cases} 1 & \text{if } n = m - c \\ 0 & \text{otherwise,} \end{cases}$$

where c is the shift parameter (distance from the diagonal) and the dimensions of the matrix depend on the context (here, $2M \times 2M$). Then, $S_{n'+m'}$ in the modulation matrix product can be written as

$$\hat{\mathbf{I}} = (-1)^r 2M \big(\mathbf{S}_{2M-d'} - \mathbf{S}_{-d'} \big). \tag{4.12}$$

At this point, it has become clear that the CMFB PR condition given in terms of the impulse response coefficients is a complicated expression. Hence, we still push further!

Let us return to the polyphase domain equation (4.7) for the analysis-synthesis system. The modulation matrix product can be replaced with the parametric matrix sum

$$\boldsymbol{F}^c_p(z) \, \boldsymbol{H}^c_p(z) = \begin{bmatrix} \mathbf{I} & z^{-1}\mathbf{I} \end{bmatrix} G(z) \big(\hat{\mathbf{I}} + \hat{\mathbf{J}} \big) \tilde{G}(z) \begin{bmatrix} z^{-1}\mathbf{I} \\ \mathbf{I} \end{bmatrix}. \tag{4.13}$$

Next, we consider the anti-diagonal terms of this product. If we express the prototype filter part with block matrices, then the corresponding part in (4.13) becomes

$$\begin{bmatrix} \mathbf{I} & z^{-1}\mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{G}_{00}(z) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{G}_{11}(z) \end{bmatrix} \begin{bmatrix} \mathbf{J}_M & \mathbf{0} \\ \mathbf{0} & -\mathbf{J}_M \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{G}}_{00}(z) & \mathbf{0} \\ \mathbf{0} & \tilde{\boldsymbol{G}}_{11}(z) \end{bmatrix} \begin{bmatrix} z^{-1}\mathbf{I} \\ \mathbf{I} \end{bmatrix}$$

$$= z^{-1} \underbrace{\boldsymbol{G}_{00}(z) \mathbf{J}_M \tilde{\boldsymbol{G}}_{00}(z)}_{1^\circ} - z^{-1} \underbrace{\boldsymbol{G}_{11}(z) \mathbf{J}_M \tilde{\boldsymbol{G}}_{11}(z)}_{2^\circ}.$$

Fortunately, the above matrix is a zero matrix. In other words, $\boldsymbol{G}_{00}(z)\mathbf{J}_M\tilde{\boldsymbol{G}}_{00}(z) = \boldsymbol{G}_{11}(z)\mathbf{J}_M\tilde{\boldsymbol{G}}_{11}(z)$. To verify this we check the terms marked with $1°$ and $2°$ in the above equation; matrix $\mathbf{J}_M$ just reverses the order of the polyphase elements and makes the matrices anti-diagonal:

$$1°: \quad \left[\boldsymbol{G}_{00}(z)\mathbf{J}_M\tilde{\boldsymbol{G}}_{00}(z)\right]_{\ell,M-1-\ell} = \left(\sum_m h[\ell + 2mM](-z^2)^{-m}\right) \times$$

$$\left(\sum_m h[2M - 1 - (M - 1 - \ell) + 2mM](-z^2)^{-m}\right) = G_\ell(-z^2)\,G_{\ell+M}(-z^2).$$

$$2°: \quad \left[\boldsymbol{G}_{11}(z)\mathbf{J}_M\tilde{\boldsymbol{G}}_{11}(z)\right]_{\ell,M-1-\ell} = \left(\sum_m h[\ell + M + 2mM](-z^2)^{-m}\right) \times$$

$$\left(\sum_m h[M - 1 - (M - 1 - \ell) + 2mM](-z^2)^{-m}\right) = G_{\ell+M}(-z^2)\,G_\ell(-z^2).$$

Hence, $\hat{\mathbf{J}}$ does not contribute in the PR requirements in case, where the same prototype filter is used in the generation of analysis and synthesis subfilters.

The other part of the PR requirements is together stated with two complementary $2M \times 2M$ shift matrices, i.e., (4.12) has the sparsity pattern of a circularly rotated diagonal matrix. This is not yet a satisfactory form as the delay parameter τ is partitioned into the multiple of $2M$ portion and the remainder d'. Fortunately, we can substitute $\hat{\mathbf{I}}$ partitioned to $M \times M$ matrix blocks into equation (4.13), when using $\tau = Ms-1+d$, $s = \lfloor(\tau+1)/M\rfloor$ and $d = \tau+1-Ms$. This, after skipping the details, leads to two cases:

$$\hat{\mathbf{I}} = \begin{cases} 2M(-1)^{(s-1)/2}\begin{bmatrix} \mathbf{S}_{M-d} & \mathbf{S}_{-d} \\ -\mathbf{S}_{-d} & \mathbf{S}_{M-d} \end{bmatrix} & s \text{ odd} \\[4mm] -2M(-1)^{s/2}\begin{bmatrix} \mathbf{S}_{-d} & -\mathbf{S}_{M-d} \\ \mathbf{S}_{M-d} & \mathbf{S}_{-d} \end{bmatrix} & s \text{ even.} \end{cases}$$

The above block matrix can be substituted into the RHS of the polyphase equation (4.13). Before that, we should note a useful shift-matrix property; when $\mathbf{A}$ and $\mathbf{B}$ are $M \times M$ diagonal matrices, the following holds:

$$\left[\mathbf{A}\,\mathbf{S}_{-d}\mathbf{B}\right]_{m,m-d} = \left[\mathbf{A}\right]_{m,m}\left[\mathbf{B}\right]_{m-d,m-d} \qquad m = d, d+1, \ldots, M-1$$

$$\left[\mathbf{A}\,\mathbf{S}_{M-d}\mathbf{B}\right]_{m,m+M-d} = \left[\mathbf{A}\right]_{m,m}\left[\mathbf{B}\right]_{m+M-d,m+M-d} \qquad m = 0, 1, \ldots, d-1.$$

In other words, the shift-matrix pairs two diagonal elements and shifts the product to a new position. Then we can proceed by writing open the block matrices

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) =$$
$$\begin{cases} 2M(-1)^{(s-1)/2}\big(z^{-1}\boldsymbol{G}_{00}(z)\mathbf{S}_{M-d}\tilde{\boldsymbol{G}}_{00}(z) + z^{-1}\boldsymbol{G}_{11}(z)\mathbf{S}_{M-d}\tilde{\boldsymbol{G}}_{11}(z) \\ \qquad\qquad -z^{-2}\boldsymbol{G}_{11}(z)\mathbf{S}_{-d}\tilde{\boldsymbol{G}}_{00}(z) + \boldsymbol{G}_{00}(z)\mathbf{S}_{-d}\tilde{\boldsymbol{G}}_{11}(z)\big) & s \text{ odd} \\[2mm] -2M(-1)^{s/2}\big(z^{-1}\boldsymbol{G}_{00}(z)\mathbf{S}_{-d}\tilde{\boldsymbol{G}}_{00}(z) + z^{-1}\boldsymbol{G}_{11}(z)\mathbf{S}_{-d}\tilde{\boldsymbol{G}}_{11}(z) \\ \qquad\qquad +z^{-2}\boldsymbol{G}_{11}(z)\mathbf{S}_{M-d}\tilde{\boldsymbol{G}}_{00}(z) - \boldsymbol{G}_{00}(z)\mathbf{S}_{M-d}\tilde{\boldsymbol{G}}_{11}(z)\big) & s \text{ even.} \end{cases}$$

$$(4.14)$$

We may pick the non-zero elements. Furthermore, we can equate them with the PR condition of the M-channel FB in equation (2.25):

$$\left[\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\, \boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) \right]_{\ell,\ell-d} = z^{-s+1} \qquad \ell = d, d+1, \ldots, M-1 \tag{4.15}$$

$$\left[\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\, \boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) \right]_{\ell,\ell+M-d} = z^{-s} \qquad \ell = 0, 1, \ldots, d-1. \tag{4.16}$$

In the simplification, we gather the $2M$-phase components given in equations (4.5) and (4.6) from equation (4.14) which leads to two element-wise cases. Furthermore, we multiply the scaling coefficient by $1/(2M)$ and, when applicable, the delay term z^{-1} on the RHS. This yields the desired requirement in terms of prototype filter coefficients.

The arbitrary-delay, arbitrary-length CMFB is a perfect-reconstruction critically sampled subband system when the prototype filter coefficients in equations (4.5) and (4.6), depending on the parity of s, satisfy the following sets of constraint equations:

a) s odd

$$-z^{-2}G_{\ell+M}(-z^2)\tilde{G}_{\ell-d}(-z^2) + G_\ell(-z^2)\tilde{G}_{\ell+M-d}(-z^2) = \frac{(-1)^{(s-1)/2}}{2M}z^{-s+1}, \\ \ell = d, d+1, \ldots, M-1 \tag{4.17}$$

$$G_\ell(-z^2)\tilde{G}_{\ell+M-d}(-z^2) + G_{\ell+M}(-z^2)\tilde{G}_{\ell+2M-d}(-z^2) = \frac{(-1)^{(s-1)/2}}{2M}z^{-s+1}, \\ \ell = 0, 1, \ldots, d-1 \tag{4.18}$$

b) s even

$$G_\ell(-z^2)\tilde{G}_{\ell-d}(-z^2) + G_{\ell+M}(-z^2)\tilde{G}_{\ell+M-d}(-z^2) = -\frac{(-1)^{s/2}}{2M}z^{-s+2}, \\ \ell = d, d+1, \ldots, M-1 \tag{4.19}$$

$$z^{-2}G_{\ell+M}(-z^2)\tilde{G}_{\ell+M-d}(-z^2) - G_\ell(-z^2)\tilde{G}_{\ell+2M-d}(-z^2) = -\frac{(-1)^{s/2}}{2M}z^{-s}, \\ \ell = 0, 1, \ldots, d-1. \tag{4.20}$$

We assume that these constraints hold in Section 4.4 when we discuss the PR of biorthogonal exponentially modulated filter banks.

In biorthogonal CMFB design, equations (4.17-4.20) are strict non-linear constraints when the PR is desired. The inclusion of these equations as the constraint functions in the optimization routines is somewhat clumsy. This is especially true when deriving the analytic gradient functions for these constraint functions. For this purpose, we process these equations into a more suitable form. This is demonstrated with the important special case, that is, with a $2KM$-length paraunitary CMFB with a symmetric prototype filter.

4.2 Paraunitary CMFB - Special Case of Biorthogonal PR Constraints

Based on the underlying theme, the more general biorthogonal CMFB definition "hides" paraunitary case resulting by selecting $D = 0$. Furthermore, we concentrate on case, where the prototype filter length is $2KM$ so that the symmetry property becomes $h[n] = h[2KM - 1 - n]$.

Since $\tau + 1 = Ms = 2KM$ ($K \in \mathbb{N}$), s is even and $d = 0$. Under these circumstances, we need to consider equation (4.19) only.

The remaining constraint function is written in terms of prototype filter coefficients and we denote the coefficients of the resulting two product polynomials with $c_0[m]$ and $c_1[m]$:

$$
\begin{aligned}
& G_\ell(-z^2)\tilde{G}_\ell(-z^2) + G_{\ell+M}(-z^2)\tilde{G}_{\ell+M}(-z^2) \\
&= \left(\sum_m (-1)^m h[\ell + 2mM]z^{-2m} \right) \left(\sum_m (-1)^m h[2M - 1 - \ell + 2mM]z^{-2m} \right) \\
&\quad + \left(\sum_m (-1)^m h[\ell + M + 2mM]z^{-2m} \right) \left(\sum_m (-1)^m h[M - 1 - \ell + 2mM]z^{-2m} \right) \\
&= \sum_m c_0[m]z^{-2m} + \sum_m c_1[m]z^{-2m} = \sum_m (c_0[m] + c_1[m])z^{-2m}.
\end{aligned}
$$

The coefficients matching index m are obtained with a single sum notation and, additionally, we do some re-arranging:

$$
\begin{aligned}
c_0[m] &= \sum_i (-1)^i h[\ell + 2iM](-1)^{m-i} h[2M - 1 - \ell + 2(m-i)M] \\
&= (-1)^m \sum_i h[\ell + 2iM]h[2M - 1 - \ell + 2mM - 2iM], \\
c_1[m] &= \sum_i (-1)^i h[\ell + M + 2iM](-1)^{m-i} h[M - 1 - \ell + 2(m-i)M] \\
&= (-1)^m \sum_i h[\ell + (2i+1)M]h[2M - 1 - \ell + 2mM - (2i+1)M].
\end{aligned}
$$

In the above case, the number of non-zero terms in $\tilde{G}_\ell(-z^2)$ and $G_\ell(-z^2)$ is K so that the index m of sequences $c_0[m]$ and $c_1[m]$ varies within 0 and $2K - 2$. The variable i, in turn, can be fixed in a parametric manner. Then we proceed by considering the first half of the sequence $c[m] = c_0[m] + c_1[m]$ by letting $m = 0, 1, \ldots, K - 1$ and combine the odd/even multiples of M:

$$
\begin{aligned}
c[m] &= (-1)^m \sum_{i=0}^{m} \big(h[\ell + 2iM]h[2M - 1 - \ell + 2mM - 2iM] \\
&\qquad + h[\ell + (2i+1)M]h[2M - 1 - \ell + 2mM - (2i+1)M] \big) \qquad (4.21) \\
&= (-1)^m \sum_{i=0}^{2m+1} h[\ell + iM]h[2M - 1 - \ell + 2mM - iM],
\end{aligned}
$$

where $m = 0, 1, \ldots, K - 1$. Next, the symmetry property of the prototype filter is utilized as

$$
\begin{aligned}
c[m] &= (-1)^m \sum_{i=0}^{2m+1} h[\ell + iM]h[2KM - 1 - (2M - 1 - \ell + 2mM - iM)] \\
&= (-1)^m \sum_{i=0}^{2m+1} h[\ell + iM]h[\ell + 2(K-1)M - 2mM + iM].
\end{aligned}
$$

Reversing the order of the coefficients provides the following neat formula:

$$c[K-1-m] = (-1)^{K-1-m} \sum_{i=0}^{2K-1-2m} h[\ell+iM]h[\ell+iM+2mM]. \qquad (4.22)$$

This needs to be substituted into the PR condition

$$\sum_m c[m]z^{-2m} = -\frac{(-1)^K}{2M}z^{-2K+2}$$

but the polynomial coefficients in the desired neat formula are time reversed. We reverse them back by re-expressing the LHS of the above equation as

$$z^{-(2K-2)} \sum_{m=0}^{K-1} c[K-1-m]z^{2m} = -\frac{(-1)^K}{2M}z^{-2K+2},$$

which results, after some minor arrangements, in the following attractive form:

$$\sum_{m=0}^{K-1} c[K-1-m]z^{2m} = -\frac{(-1)^K}{2M}.$$

The final step is to substitute the values of $c[K-1-m]$ expressed earlier in equation (4.22) and open the polynomial for a better view

$$\sum_{m=0}^{K-1} \left((-1)^{K-1-m} \sum_{i=0}^{2K-1-2m} h[\ell+iM]h[\ell+iM+2mM] \right) z^{2m}$$

$$= (-1)^{K-1} \sum_{i=0}^{2K-1} h[\ell+iM]h[\ell+iM]$$

$$+ \sum_{m=1}^{K-1} \left((-1)^{K-1-m} \sum_{i=0}^{2K-1-2m} h[\ell+iM]h[\ell+iM+2mM] \right) z^{2m} = -\frac{(-1)^K}{2M}.$$

The polynomial terms on the LHS must be zero except when $m = 0$. Miraculously, also the sign reversion term $(-1)^K$ vanishes. The PR constraints for the $2KM$-length paraunitary prototype filter can then be given as

$$\sum_{i=0}^{2K-1-2m} h[\ell+iM]h[\ell+iM+2mM] = \frac{1}{2M}\delta[m], \qquad (4.23)$$

where $m = 0, 1, \ldots, K-1$ and $\ell = 0, 1, \ldots, M-1$. These are almost the same constraints as those given to the ELT type of cosine modulation in equation (2.36). However, the range on ℓ is twice of the required. The redundancy is revealed when we consider sum

$$\sum_{i=0}^{2K-1-2m} h[M-1-\ell+iM]h[M-1-\ell+iM+2mM] = \frac{1}{2M}\delta[m]$$

and let $\ell = 0, 1, \ldots, M/2 - 1$. We can reverse the summation order of the index i on the LHS of the above equation and then apply the symmetry of the prototype filter:

$$
\sum_{i=0}^{2K-1-2m} h[M - 1 - \ell + (2K - 1 - 2m - i)M]
$$
$$
\times h[M - 1 - \ell + (2K - 1 - 2m - i)M + 2mM]
$$
$$
= \sum_{i=0}^{2K-1-2m} h[2KM - 1 - (M - 1 - \ell + (2K - 1 - 2m - i)M)]
$$
$$
\times h[2KM - 1 - (M - 1 - \ell + (2K - 1 - 2m - i)M + 2mM)]
$$
$$
= \sum_{i=0}^{2K-1-2m} h[\ell + 2mM + iM] \, h[\ell + iM] = \frac{1}{2M} \delta[m].
$$

We are assured and relieved a bit. The number of constraint equations in (4.23) can be reduced by half.

Actually, if we want to be thorough, the next thing is to check the other half of the sequence $c[m]$ and ensure it provides identical constraints as in (4.23) when $m = K, K+1, \ldots, 2K-2$. In author's humble opinion, proving that does not give better insight for the matter and it is proper to move forward.

As for the concluding thoughts, we have revisited again the "paraunitarity is a special case of biorthogonality"-theme and demonstrated how we may process the equations in (4.17-4.20) into design constraints for the optimization routine. One should expect now that tackling with the biorthogonal constraints is more challenging due to arbitrary-length/delay parameters. At least we can say that it requires a certain degree of stubbornness to represent them in an optimization algorithm friendly manner.

4.3   Sine-Modulated Filter Bank

4.3.1   Subfilter Definition and Connection with CMFB

The sine-modulated filter bank can be considered as the companion FB for the CMFB. As the name suggests, the modulation sequences have a different phase offset but otherwise the SMFB is an M-channel odd-stacked filter bank. In certain subband adaptive filtering tasks, the redundancy of the subsignals is a helpful feature and this can be achieved by analyzing the real-valued signal in parallel with cosine- and sine-modulated filter banks. In this section, we need the PR SMFB for analytic purposes; once we verify this property, the odd-stacked EMFB PR proof becomes just an elementary matrix calculation.

The synthesis and analysis subfilter equations for the SMFB are

$$
f_k^{\text{s}}[n] = 2 \, h[n] \, \sin\left[\left(n - \frac{N + D - M}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]
\tag{4.24}
$$

$$
h_k^{\text{s}}[n] = -2 \, h[n] \, \sin\left[\left(n - \frac{N + D + M}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right],
\tag{4.25}
$$

respectively. Then the synthesis subfilter modulation sequence is manipulated so that the

trigonometric angle sum identity can be applied:

$$f_k^{\text{s}}[n] = 2\,h[n]\,\sin\left[\left(n - \frac{N+D-M}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$= 2\,h[n]\sin\left[\left(n - \frac{N+D+M}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M} + \left(k+\frac{1}{2}\right)\pi\right]$$

$$= 2\,h[n]\Bigg(\underbrace{\cos\left((k+1/2)\pi\right)}_{0}\,\sin\left[\left(n - \frac{N+D+M}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$+ \underbrace{\sin\left((k+1/2)\pi\right)}_{(-1)^k}\cos\left[\left(n - \frac{N+D+M}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]\Bigg)$$

$$= (-1)^k h_k^{\text{c}}[n].$$

We see above how the SMFB synthesis subfilters are related with the CMFB analysis subfilters. In a similar fashion, we can verify the following analysis subfilter connection: $h_k^{\text{s}}[n] = (-1)^k f_k^{\text{c}}[n]$.

4.3.2 Perfect-Reconstruction of SMFB in the Polyphase Domain

We exploit the above-observed connections between the CMFB and SMFB subfilters to show that the SMFB is inherently a PR FB. First, based on the connections, we can relate the subfilter polyphase components as

$$\left[\boldsymbol{F}_{\text{p}}^{\text{s}}(z)\right]_{\ell,k}$$

$$= \sum_m f_k^{\text{s}}[\ell + mM]\,z^{-m} = \sum_m (-1)^k h_k^{\text{c}}[\ell + mM]\,z^{-m}$$

$$= (-1)^k\left[\boldsymbol{H}_{\text{p}}^{\text{c}}(z)\right]_{k,M-1-\ell} = (-1)^k\left[\left(\boldsymbol{H}_{\text{p}}^{\text{c}}(z)\right)^T\right]_{M-1-\ell,k}$$

$$\left[\boldsymbol{H}_{\text{p}}^{\text{s}}(z)\right]_{k,\ell} = \sum_m h_k^{\text{s}}[M-1-\ell+mM]\,z^{-m}$$

$$= \sum_m (-1)^k f_k^{\text{c}}[M-1-\ell+mM]\,z^{-m}$$

$$= (-1)^k\left[\boldsymbol{F}_{\text{p}}^{\text{c}}(z)\right]_{M-1-\ell,k} = (-1)^k\left[\left(\boldsymbol{F}_{\text{p}}^{\text{c}}(z)\right)^T\right]_{k,M-1-\ell}.$$

Then, we can substitute the CMFB equations into the product of SMFB polyphase matrices

$$\boldsymbol{F}_{\text{p}}^{\text{s}}(z)\,\boldsymbol{H}_{\text{p}}^{\text{s}}(z) = \mathbf{J}\left(\boldsymbol{H}_{\text{p}}^{\text{c}}(z)\right)^T\mathbf{D}\,\mathbf{D}\left(\boldsymbol{F}_{\text{p}}^{\text{c}}(z)\right)^T\mathbf{J},$$

where $\mathbf{J}$ is the $M \times M$ reverse identity matrix and $\mathbf{D} = \text{diag}\{1, -1, 1, -1, \dots\}$. All matrices have dimensions $M \times M$. We immediately notice that $\mathbf{DD} = \mathbf{I}$ and then (2.25) is applicable

$$\boldsymbol{F}_{\text{p}}^{\text{s}}(z)\,\boldsymbol{H}_{\text{p}}^{\text{s}}(z) = \mathbf{J}\left(\boldsymbol{F}_{\text{p}}^{\text{c}}(z)\,\boldsymbol{H}_{\text{p}}^{\text{c}}(z)\right)^T\mathbf{J}$$

$$= z^{-s+1}\,\mathbf{J}\begin{bmatrix} \mathbf{0} & \mathbf{I}_{M-d} \\ z^{-1}\mathbf{I}_d & \mathbf{0} \end{bmatrix}\mathbf{J} \tag{4.26}$$

$$= z^{-s+1}\begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix}.$$

Hence, we can conclude that the biorthogonal odd-stacked SMFB is a PR FB.
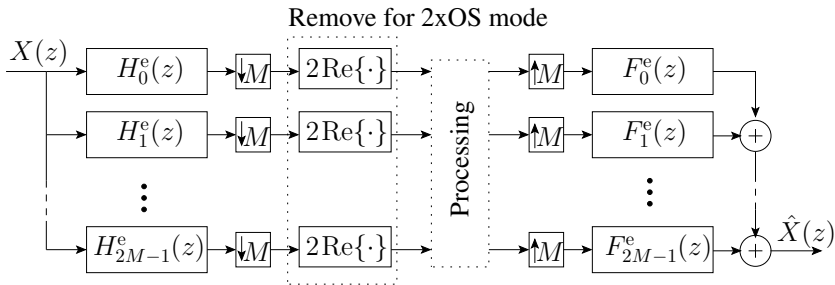
Figure 4.2: Odd-stacked exponentially modulated filter bank with critical and 2x oversampled operation modes.

4.4 Biorthogonal Odd-Stacked Exponentially Modulated Filter Bank

4.4.1 Subband Decomposition and Expansion for Complex-Valued Signals

In general, complex-valued signals have non-symmetric power spectra with respect to zero frequency. A uniform filter bank that provides meaningful signal decomposition must then divide the whole frequency range $[-\pi, \pi]$ into equally wide subbands. Further on, if the PR is required, we may choose an even-stacked MDFT-FB or an odd-stacked EMFB. We consider in this section the latter in detail with polyphase domain analysis. In addition, the modulated complex lapped transform (MCLT) [59] is related with the odd-stacked EMFB, but not exactly the same (the discussion is postponed until Section 4.4.4).

The biorthogonal odd-stacked EMFB is shown in Figure 4.2. It is assumed that the input signal $X(z)$ is complex-valued so that discussion on critically sampled and 2x oversampled operation modes is correct. On the analysis side, subfilters $H_k^{\mathrm{e}}(z)$ decompose the input signal into $2M$ lower rate subsignals where M is the decimation factor. Depending on the nature of the filtering problem, one can switch from critically sampling to oversampling by a factor of two just by leaving out the real-part operation. The annotation "Remove for 2xOS mode" in Figure 4.2 refers to the latter.

If real-valued subsignals are processed, there is no sample rate growth as for every M new complex-valued input samples, the CS EMFB produces $2M$ real-valued subsamples. If the subfilters are fixed in both modes, the CS subsignals are scaled differently (multiply by two after the real-part operator). By doing so, the synthesis subfilters $F_k^{\mathrm{e}}(z)$ recover the same output signal level regardless of CS/2xOS mode.

The above CMFB discussion paved way for the biorthogonal EMFB subfilter definition. The synthesis and analysis subfilter impulse responses are

$$f_k^{\mathrm{e}}[n] = h[n] \exp\left[j\left(n - \frac{N+D-M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \tag{4.27}$$

$$h_k^{\mathrm{e}}[n] = h[n] \exp\left[j\left(n - \frac{N+D+M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right]. \tag{4.28}$$

The design parameters N and D were introduced in Section 4.1.1, but now the discrete frequency index k ranges as $k = 0, 1, \ldots, 2M - 1$. This modulation leads to odd-stacked sub-
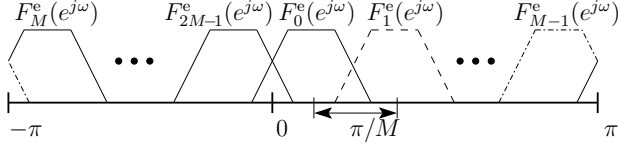
Figure 4.3: Odd-stacked subfilters of the EMFB. The individual subfilters have complex-valued impulse responses and single mainlobes within $[-\pi, \pi]$ in contrast to two mainlobes of the subfilters in the CMFB.

band splitting with complex-valued subfilters and their amplitude response center frequencies are given by $\omega_k = (k + \frac{1}{2})\pi/M$. Figure 4.3 shows the frequency ordering of the subbands; the amplitude responses divide both the positive and "negative" frequency sides into uniformly spaced subchannels. Intuitively, we may view these two sides as the opposite directions of circular motion.

The basic properties of the EMFB subfilters can be verified by re-arranging the terms in the argument of the exponential modulation sequence. We skip the details to keep the focus on a compact MFB theory presentation and list below some basic results.

First, we have the conjugate property of kth and $(2M - 1 - k)$th subfilters

$$f_k^{\mathrm{e}}[n] = (-1)^{N+D+M}\big(f_{2M-1-k}^{\mathrm{e}}[n]\big)^* \tag{4.29}$$

$$h_k^{\mathrm{e}}[n] = (-1)^{N+D+M}\big(h_{2M-1-k}^{\mathrm{e}}[n]\big)^*, \tag{4.30}$$

where $k = 0, 1, \ldots, M-1$. The kth analysis and synthesis subfilters are also simply connected through

$$f_k^{\mathrm{e}}[n] = j(-1)^k h_k^{\mathrm{e}}[n]. \tag{4.31}$$

The critically sampled biorthogonal EMFB with real input signal is equivalent to the biorthogonal CMFB as one can directly deduce from

$$h_k^{\mathrm{c}}[n] = 2\,\mathrm{Re}\big\{h_k^{\mathrm{e}}[n]\big\},$$

where $k = 0, 1, \ldots, M - 1$.

Then, we continue our treatment for the biorthogonal EMFB by moving into the polyphase domain. Basically, Figure 2.12 (b) is a suitable model for this purpose once we take into account the increase of the subsignals from M to $2M$ and the real-part processing in the CS mode. The elements of the EMFB polyphase matrices are defined as

$$\big[\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\big]_{\ell,k} = \sum_m f_k^{\mathrm{e}}[\ell + mM]\,z^{-m}$$

$$\big[\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z)\big]_{k,\ell} = \sum_m h_k^{\mathrm{e}}[M - 1 - \ell + mM]\,z^{-m},$$

where $k = 0, 1, \ldots, 2M - 1$ and $\ell = 0, 1, \ldots, M - 1$. Consequently, the EMFB polyphase matrices are non-square. Since the EMFB subfilters are separable with Euler's formula for cosines and sines, we can write $\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)$ and $\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z)$ in terms of the CMFB and SMFB polyphase matrices:

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z) = \frac{1}{2}\big[\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big) \qquad (-1)^{N+D+M}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\boldsymbol{J}\big]$$

61

$$\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z) = \frac{1}{2} \left[\begin{array}{c} \big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)\big) \\ (-1)^{N+D+M}\,\mathbf{J}\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)\big) \end{array} \right]$$

The negative frequency side submatrices follow from the conjugate symmetries in equations (4.29) and (4.30).

4.4.2 Perfect-Reconstruction of Biorthogonal EMFB

Critically Sampled Subsignals

The real-part operation in the EMFB flowgraph necessitates checking additional details when compared with the basic M-channel case. For this purpose, we denote a complex conjugated sequence, say $y^*[n]$, in the z-transform domain with $\overline{Y}(z) = Y^*(z^*) = \mathcal{Z}\{y^*[n]\}$. Then the z-transform identity for taking the real-part is $\mathcal{Z}\{\mathrm{Re}\{y[n]\}\} = \frac{1}{2}\big(Y(z) + \overline{Y}(z)\big)$. With this notation, we obtain the polyphase domain input-output relation for the CS EMFB:

$$\hat{X}^{\mathrm{cs}}(z) = \boldsymbol{c}^T(z)\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z^M)\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z^M)\,\boldsymbol{x}_{\mathrm{p}}(z^M) + \overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z^M)\,\overline{\boldsymbol{x}}_{\mathrm{p}}(z^M)\big). \qquad (4.32)$$

In the above equation, we see the desired polyphase component vector $\boldsymbol{x}_{\mathrm{p}}(z)$ (cf. Section 2.6.2) as well as the sequence non-grata: $\overline{\boldsymbol{x}}_{\mathrm{p}}(z)$ (i.e., the polyphase components of the conjugated input signal). Thus we have to show the validity of the following two equations:

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z) = z^{-s+1} \begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix} \qquad (4.33)$$

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\,\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z) = \mathbf{0} \qquad (4.34)$$

The direct substitution of the CMFB/SMFB polyphase component matrices into the polyphase matrix product (4.33) yields

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z) = \frac{1}{4}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)\big)$$

$$+ \frac{1}{4}\left((-1)^{N+D+M}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\mathbf{J}\right)\left((-1)^{N+D+M}\mathbf{J}\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) + j'\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\right)$$

$$= \frac{1}{2}\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) + \frac{1}{2}\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)$$

Because the defined biorthogonal cosine- and sine-modulated filter banks are both PR systems - as stated in equations (2.25) and (4.26) - we can deduce that

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z) = z^{-s+1} \begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix}.$$

Similarly, regarding to equation (4.34), we can directly verify that

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e}}(z)\,\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z) = \frac{1}{4}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\big(\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{s}}(z)\big)$$

$$+ \frac{1}{4}\left((-1)^{N+D+M}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\mathbf{J}\right)$$

$$\times \left((-1)^{N+D+M}\mathbf{J}\big(\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{s}}(z)\big)\right) = \mathbf{0}.$$

There is a minor detail that should be mentioned: When a purely real-valued sequence $y_R[n]$ is z-transformed (i.e., $Y_R(z) = \mathcal{Z}\{y_R[n]\}$), then it is apparent that the z-transform of its conjugate $\overline{Y}_R(z) = \mathcal{Z}\{y_R^*[n]\}$ is the same as that of the original one: $Y_R(z) = \overline{Y}_R(z)$. On these grounds, the identities $\boldsymbol{H}_p^c(z) = \overline{\boldsymbol{H}}_p^c(z)$ and $\boldsymbol{H}_p^s(z) = \overline{\boldsymbol{H}}_p^s(z)$ hold for the conjugates of cosine- and sine-modulated FB polyphase matrices. Thus the EMFB polyphase matrix product with conjugated analysis filters reduces to

$$\boldsymbol{F}_p^e(z)\,\overline{\boldsymbol{H}}_p^e(z) = \frac{1}{2}\boldsymbol{F}_p^c(z)\boldsymbol{H}_p^c(z) - \frac{1}{2}\boldsymbol{F}_p^s(z)\boldsymbol{H}_p^s(z) = \boldsymbol{0}$$

and, therefore, the overall bank satisfied the PR condition.

Subsignals Oversampled by Factor Two

The PR proof for the 2x oversampled EMFB is included in the above because we can write

$$
\begin{aligned}
\hat{X}^{os}(z) &= \boldsymbol{c}^T(z)\,\boldsymbol{F}_p^e(z^M)\boldsymbol{H}_p^e(z^M)\boldsymbol{x}_p(z^M) \\
&= z^{-sM+M}\boldsymbol{c}^T(z)\begin{bmatrix} \boldsymbol{0} & z^{-M}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \boldsymbol{0} \end{bmatrix}\boldsymbol{x}_p(z^M) \\
&= z^{-\tau}X(z).
\end{aligned}
\tag{4.35}
$$

The input signal thus is only delayed by the biorthogonal EMFB provided that there is no subband processing.

The summary of the above proofs: If we design a perfect reconstruction biorthogonal CMFB, the same prototype filter generates an odd-stacked biorthogonal EMFB for the subband processing of complex-valued signals. The PR condition is preserved in the EMFB in both the CS and 2xOS operation modes.

4.4.3 Paraunitary Exponentially Modulated Filter Bank

The use of the paraunitary prototype filter, that is, $h[n] = h[N - n]$ and $D = 0$, in the EMFB leads to similar simplified properties as presented for M-channel FBs in Section 2.6.3. First, the kth analysis and synthesis subfilters are connected through the conjugated time-reversing relation

$$f_k^e[n] = \left(h_k^e[N - n]\right)^*.\tag{4.36}$$

Second, paraunitary prototype filters whose length is a multiple of M, that is, $N + 1 = Ms$ are of special interest. In this case, the EMFB synthesis polyphase matrix satisfies the following attractive properties:

$$
\begin{aligned}
\left[\boldsymbol{F}_p^e(z)\right]_{\ell,k} &= \sum_{m=0}^{s-1} \left(h_k^e[Ms - 1 - (\ell + mM)]\right)^* z^{-m} \\
&= \sum_{m'=0}^{s-1} \left(h_k^e[Ms - 1 - (\ell + (s - 1 - m')M)]\right)^* z^{-(s-1-m')} \\
&= z^{-s+1}\sum_{m'=0}^{s-1} \left(h_k^e[M - 1 - \ell + m'M]\right)^* z^{m'} \\
&= z^{-s+1}\left[\overline{\boldsymbol{H}}_p^e(z^{-1})\right]_{k,\ell} = z^{-s+1}\left[\left(\overline{\boldsymbol{H}}_p^e(z^{-1})\right)^T\right]_{\ell,k}
\end{aligned}
$$

Third, substituting the above connection into equations (4.33) and (4.34) yields the following plain PR conditions:

$$\left(\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z^{-1})\right)^{T} \boldsymbol{H}_{\mathrm{p}}^{\mathrm{e}}(z) = \boldsymbol{I} \tag{4.37}$$

$$\left(\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z^{-1})\right)^{T} \overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e}}(z) = \boldsymbol{0}. \tag{4.38}$$

It is worth emphasizing that equations (4.37) and (4.38) hold only when $D = 0$ and $N + 1$ is a multiple of M.

4.4.4 2x Oversampling for Real-Valued Signals

When the 2x oversampled EMFB is used for processing a purely real-valued signal, the above-mentioned conjugation property leads to the consequences that half the subsignals are redundant and the analysis bank in the 2xOS EMFB reduces to an M-band complex FB. In practice, this is equivalent to Malvar's MCLT - the slight difference being biorthogonality of the former and allowing arbitrary-delay prototypes. In [59], it was demonstrated that the oversampled construction considerably reduced the time-domain aliasing present in acoustic echo cancellation and audio enhancement systems when compared with CS subband systems.

As noted by Malvar [59], the signal reconstruction is not unique and, from the algorithmic point of view, we have three possibilities in signal reconstruction depending on application-specific subband processing:

1. Apply the complex conjugate mirroring of the M subsignals to recover the missing half and then use a $2M$-band synthesis EMFB.

2. Take the real part of the subsignals and use a critically sampled synthesis CMFB[†].

3. Use M synthesis subfilters of the EMFB and then take the real part of the output signal, as shown in Figure 4.4.

The first scheme is a practical way to realize the filter bank when we have the full FFT-based EMFB algorithm available. The second one works when the same prototype filter is used for generating both the analysis and synthesis banks, but the PR property fails when these banks are created using their own prototype filters. The two-prototype filter designs will be considered in Section 5.5. The last alternative is the most convenient for the EMFB theory. To make a clear difference from the FB system in Figure 4.2, we denote the subfilters of the 2xOS EMFB used for processing real-valued signals by

$$h_{k}^{\mathrm{e-r}}[n] = h_{k}^{\mathrm{e}}[n] \tag{4.39}$$

$$f_{k}^{\mathrm{e-r}}[n] = f_{k}^{\mathrm{e}}[n], \tag{4.40}$$

where $k = 0, 1, \ldots, M - 1$.

[†]An application example of such a system would be adaptive subband filtering where the reference subsignals are obtained from a CS analysis CMFB. The real and imaginary parts of the kth 2xOS analysis EMFB subsignal are combined into real-valued output subsignal by using a 2-input/single-output per subband adaptive filter.
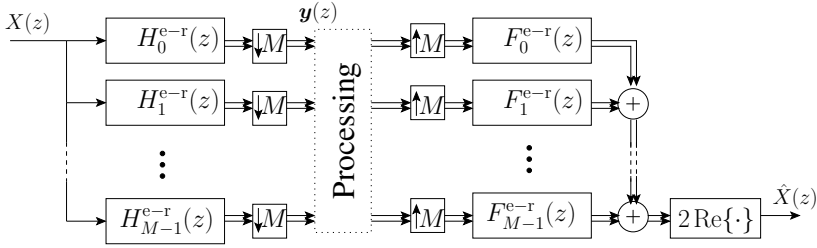
Figure 4.4: EMFB for 2x oversampled subband processing of real-valued input signal $X(z)$. Complex-valued signals are indicated with double lines.

The perfect-reconstruction property for this subband system is now straightforward to check. The polyphase matrices for the FB in Figure 4.4 are

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-r}}(z) = \frac{1}{2}\big(\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z) + j\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\big)$$

$$\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e-r}}(z) = \frac{1}{2}\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) - j\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z)\big).$$

The real-part operations can be done before interpolation if we utilize the polyphase system configuration shown in Figure 2.12 (b). Then, the polyphase domain transfer matrices of the 2xOS EMFB (for real-valued signals) can be shown to satisfy

$$\begin{aligned}
&\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-r}}(z)\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e-r}}(z) + \overline{\boldsymbol{F}}_{\mathrm{p}}^{\mathrm{e-r}}(z)\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e-r}}(z) \\
&= \frac{1}{2}\boldsymbol{F}_{\mathrm{p}}^{\mathrm{c}}(z)\boldsymbol{H}_{\mathrm{p}}^{\mathrm{c}}(z) + \frac{1}{2}\boldsymbol{F}_{\mathrm{p}}^{\mathrm{s}}(z)\boldsymbol{H}_{\mathrm{p}}^{\mathrm{s}}(z) = z^{-s+1}\begin{bmatrix} \mathbf{0} & z^{-1}\mathbf{I}_d \\ \mathbf{I}_{M-d} & \mathbf{0} \end{bmatrix}.
\end{aligned} \tag{4.41}$$

Consequently, this is a PR subband system.

4.5 Even-Stacked Biorthogonal EMFB

4.5.1 Time Varying Even-Stacked EMFB

One can notice from the schematic amplitude responses in Figure 4.3 that the odd-stacked EMFB has, instead of a low-pass filter with the center frequency located at $\omega = 0$, two filters with the center frequencies at $\omega = \pm\pi/(2M)$. Similarly, there are four filters with the center frequencies at $\omega = \pm\pi \mp \pi/(2M)$, instead of two single filters with center frequencies at $\omega = \pm\pi$. If the application requires the complex filter bank to detect a single low-pass/high-pass component, we can frequency shift the EMFB subfilters by $-\pi/(2M)$ so that the center frequency of the 0th subfilter is located at $\omega = 0$. On the synthesis side, this can be handled by modulating the reconstructed signal with an exponential sequence. In addition, we should pick the modulation phase so that the real and imaginary parts of the modified subfilters are (anti-)symmetric with a symmetric prototype.

Let us then consider a modified synthesis EMFB shown in Figure 4.5. There are two new elements in the flowgraph: subsignal sign-reversing sequence $(-1)^{\lfloor m/2 \rfloor}$ and down-modulation multiplier

$$e[n] = \exp\Big\{ -j\Big(n - \frac{N + D - M}{2}\Big)\frac{\pi}{2M}\Big\}.$$
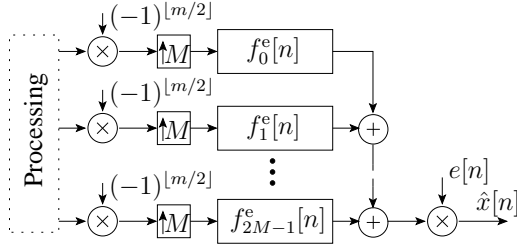
Figure 4.5: Transforming the synthesis EMFB into an even-staked filter bank by shifting the subfilter amplitude responses with an exponential sequence $e[n]$ so that the center frequency of the 0th subfilter is at $\omega = 0$.

Then the low-pass subfilter amplitude response, say $e[n]\, f_0^{\mathrm{e}}[n]$, is centered at zero frequency in the desired manner. Further on, the phase of the down-modulating sequence is chosen so that the imaginary part of this modified subfilter vanishes at subsignal time instant $m = 0$.

The problem with the exponential multiplier is that the frequency shifted EMFB subfilters are now time varying. Due to periodicity of $e[n]$ there are four possible subfilter equations depending on the instant of m. However, the $(-1)^{\lfloor m/2 \rfloor}$ multiplier generates the four-periodic sequence $\ldots, 1, -1, -1, 1, \ldots$ and reduces the number of different subfilter equations by half. Then we can find an (almost) equivalent fixed FB representation for the even-stacked EMFB where two filter banks operate in parallel and the decimation factor is $2M$.

In order to find the matching decomposition, modifications of the same kind turn the fixed odd-stacked analysis EMFB into a time varying even-stacked EMFB. The perfect-reconstruction is guaranteed because the odd-stacked EMFB is a PR system and it is trivial to find the corresponding forward transform including the inverses of the operations $(-1)^{\lfloor m/2 \rfloor}$ and $e[n]$. The analysis-synthesis reconstruction delay of the even-stacked EMFB with the time varying model is $\tau = N + D$. On the grounds of this "verbal proof", we take the even-stacked EMFB PR for granted and proceed directly to the fixed subfilter model. The two subfilters sets for the fixed model are, say, for the analysis side denoted with $h_k^{\mathrm{e-e}}[n]$ and $\check{h}_k^{\mathrm{e-e}}[n]$. The fixed model causes an extra lag of M samples to the reconstruction delay but the approach is advantageous for analytic means.

4.5.2 Subfilters Definition for Even-Stacked EMFB

The biorthogonal even-stacked EMFB for complex-valued signals is shown in Figure 4.6 (a). The CS/2xOS operation modes are inherited from the odd-stacked EMFB. The modifications presented in Figure 4.5 for the synthesis side preserve the PR. The subfilter impulse responses for the first set are

$$f_k^{\mathrm{e-e}}[n] = h[n] \exp\left[j\left(n - \frac{N+D-M}{2} \right) \frac{k\pi}{M} \right] \tag{4.42}$$

$$h_k^{\mathrm{e-e}}[n] = h[n] \exp\left[j\left(n - \frac{N+D+M}{2} \right) \frac{k\pi}{M} \right], \tag{4.43}$$

where $k = 0, 1, \ldots, 2M - 1$, whereas the responses of the kth analysis and synthesis filters in the second set are $\pi/2$-rotated versions of the former ones: $\check{f}_k^{\mathrm{e-e}}[n] = j f_k^{\mathrm{e-e}}[n]$ and $\check{h}_k^{\mathrm{e-e}}[n] = -j\, h_k^{\mathrm{e-e}}[n]$.
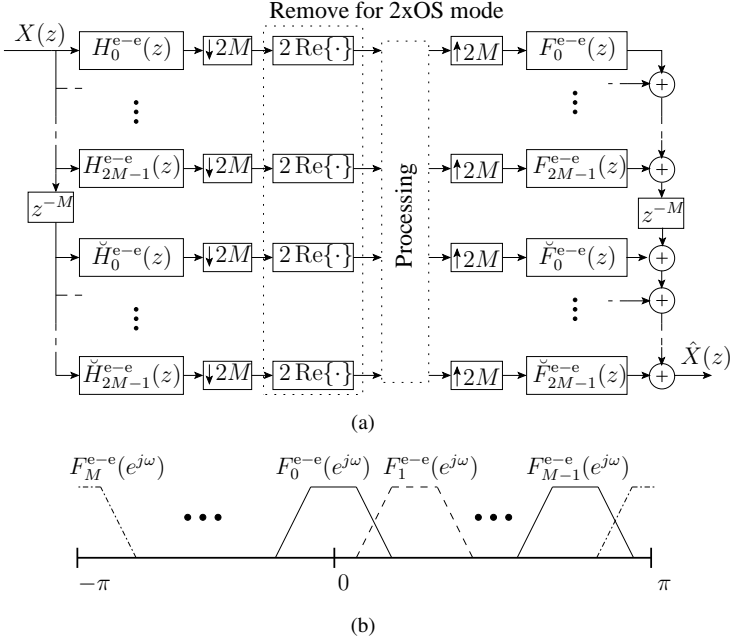
Figure 4.6: (a) Even-Stacked EMFB: Complex-valued data and critical/2x oversampling. (b) Complex subfilters divide the whole frequency range $[-\pi, \pi]$ into $2M$ subbands and the center frequency of the 0th subfilter is located at $\omega = 0$.

The complex subfilters divide the whole frequency range into $2M$ uniformly spaced subbands as schematically shown in Figure 4.6 (b). The center frequency for the kth analysis and synthesis filters in both sets is given with $\omega_k = k\pi/M$.

The kth analysis and synthesis subfilters can be related to their conjugate pairs from the opposite frequency side as

$$f_k^{\mathrm{e-e}}[n] = (-1)^{N+D+M} \left(f_{2M-k}^{\mathrm{e-e}}[n] \right)^* \tag{4.44}$$

$$h_k^{\mathrm{e-e}}[n] = (-1)^{N+D+M} \left(h_{2M-k}^{\mathrm{e-e}}[n] \right)^*, \tag{4.45}$$

where $k = 1, 2, \ldots, M - 1$. The analysis and synthesis subfilters are, in turn, connected by

$$f_k^{\mathrm{e-e}}[n] = (-1)^k h_k^{\mathrm{e-e}}[n].$$

However, transforming the even-stacked EMFB in the polyphase domain is not as straightforward as may be expected from the above odd-stacked EMFB PR conditions. Nevertheless, we can avoid the analysis with two polyphase matrix pairs by hiding the details with a single $2M$-phase matrix pair.

The polyphase domain matrices corresponding to the even-stacked EMFB in Figure 4.6 (a) are

$$\left[\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-e}}(z) \right]_{\ell,k}$$
$$= \begin{cases} \sum_m f_k^{\mathrm{e-e}}[\ell + 2mM - M] \, z^{-m} & k < 2M \\ \sum_m \breve{f}_{k-2M}^{\mathrm{e-e}}[\ell + 2mM] \, z^{-m} & k \geq 2M \end{cases} \tag{4.46}$$

$$\left[\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e-e}}(z)\right]_{k,\ell}$$

$$= \begin{cases} \sum_m h_k^{\mathrm{e-e}}[2M-1-\ell+2mM]\,z^{-m} & k < 2M \\ \sum_m \breve{h}_{k-2M}^{\mathrm{e-e}}[M-1-\ell+2mM]\,z^{-m} & k \geq 2M, \end{cases} \qquad (4.47)$$

where $k = 0, 1, \ldots, 4M-1$ and $\ell = 0, 1, \ldots, 2M-1$. The definition includes the delay by M samples (due to the model induced structural z^{-M} blocks) with the subfilters as in $f_k^{\mathrm{e-e}}[n-M]$ and $\breve{h}_k^{\mathrm{e-e}}[n-M]$ before we applied the $2M$-polyphase decomposition.

4.5.3   Perfect-Reconstruction of Even-Stacked EMFB

The analysis of the fixed even-stacked EMFB model requires minor modifications for the polyphase component vectors. Now the polyphase decomposition-expansion pair is

$$Y_\ell(z) = \sum_{m=-\infty}^{\infty} x[\ell + 2mM]z^{-m}$$

$$\hat{X}(z) = z^{-(2M-1)} \sum_{\ell=0}^{2M-1} z^{-\ell} Y_\ell(z^{2M}) = z^{-(2M-1)} X(z). \qquad (4.48)$$

Then we can plug the polyphase component matrices (4.46) and (4.47) into the vectorized form of (4.48):

$$\hat{X}^{\mathrm{cs}}(z) = \boldsymbol{c}^T(z)\,\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-e}}(z^{2M})\big(\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e-e}}(z^{2M})\,\boldsymbol{y}_{\mathrm{p}}(z^{2M})$$
$$+ \overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e-e}}(z^{2M})\,\overline{\boldsymbol{y}}_{\mathrm{p}}(z^{2M})\big), \qquad (4.49)$$

where $\boldsymbol{c}^T(z) = z^{-(2M-1)}[1 \ z^{-1} \ \cdots \ z^{-(2M-1)}]$ and $\boldsymbol{y}_{\mathrm{p}}(z) = [Y_0(z) \ Y_1(z) \ \cdots \ Y_{2M-1}(z)]^T$. We can deduce the necessary PR conditions for the even-stacked EMFB polyphase matrices as follows. First, the key alternations to the previous polyphase domain treatment is that M is replaced with $2M$ and the structure of Figure 4.6 (a) contains an additional structural delay term z^{-M}. Hence, the analysis-synthesis reconstruction delay is divided to the "multiple of $2M$"-portion and to the left-over d as $\tau + M = 2tM - 1 + d$, where $t = \lfloor (\tau + M + 1)/(2M) \rfloor$ and $d = \tau + M + 1 - 2tM$. Then, the input signal polyphase components in $\boldsymbol{y}_{\mathrm{p}}(z)$ are preserved if

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-e}}(z)\,\boldsymbol{H}_{\mathrm{p}}^{\mathrm{e-e}}(z) = z^{-t+1} \begin{bmatrix} \boldsymbol{0} & z^{-1}\boldsymbol{I}_d \\ \boldsymbol{I}_{2M-d} & \boldsymbol{0} \end{bmatrix} \qquad (4.50)$$

and the unwanted conjugated polyphase sequences are cancelled with

$$\boldsymbol{F}_{\mathrm{p}}^{\mathrm{e-e}}(z)\,\overline{\boldsymbol{H}}_{\mathrm{p}}^{\mathrm{e-e}}(z) = \boldsymbol{0}. \qquad (4.51)$$

Moreover, the resulting overall system has the desired delay because the above $\hat{X}^{cs}(z)$ can be expressed based on equation (4.50) as

$$\hat{X}^{cs}(z) = z^{-2(t-1)M} c^T(z) \begin{bmatrix} \mathbf{0} & z^{-2M}\mathbf{I}_d \\ \mathbf{I}_{2M-d} & \mathbf{0} \end{bmatrix} y_p(z^{2M})$$

$$= z^{-2(t-1)M} z^{-(2M-1)} \left(\sum_{m=0}^{2M-d-1} z^{-(m+d)} Y_m(z^{2M}) \right.$$

$$+ \left. \sum_{m=2M-d}^{2M-1} z^{-2M} z^{-(m-(2M-d))} Y_m(z^{2M}) \right)$$

$$= z^{-(2tM+d-1)} \sum_{\ell=0}^{2M-1} z^{-\ell} Y_\ell(z^{2M}) = z^{-(\tau+M)} X(z).$$

At this point, we quickly return to our previous arguments for a better overview as the eluding motives easily hide in the midst of actions. The even-stacked EMFB PR equations (4.50) and (4.51) must hold on the grounds of the following deduction chain: 1) it is possible to design a prototype filter for the odd-stacked CMFB satisfying the PR condition; 2) the same prototype filter generates a PR odd-stacked EMFB for subband processing of complex-valued signals; 3) the odd-stacked EMFB subfilters are frequency shifted into an even-stacked subband configuration and the modification maintains the PR; and 4) the alternating subfilters of the time varying system in Figure 4.5 are identical with the fixed subfilter model but the latter adds extra M samples to the reconstruction delay.

4.5.4   Paraunitary Even-Stacked EMFB

If the even-stacked EMFB is constructed using a symmetric prototype filter ($D = 0$), then the resulting subfilters are characterized by the following appealing property. They are symmetric complex FIR filters, that is, both the real and imaginary parts of the impulse response of an FIR filter with complex-valued coefficients are symmetric and/or anti-symmetric. In this somewhat refined definition for paraunitarity, the following conjugate symmetry relations can be shown to be valid for the even-stacked EMFB:

$$f_k^{e-e}[n] = (-1)^k \left(f_k^{e-e}[N-n] \right)^*$$
$$h_k^{e-e}[n] = (-1)^k \left(h_k^{e-e}[N-n] \right)^*.$$

Thus the real (imaginary) part of the kth subfilter is either symmetric or anti-symmetric depending on the parity of k. This paraunitary case is very attractive in applications, where the phase linearity of the subfilters plays a pivot role.

4.5.5   2xOS Even-Stacked EMFB for Real-Valued Signals

The second variation of the even-stacked EMFB theme occurs with real-valued input signal data and 2x oversampling. On the basis of the conjugate symmetries (4.44) and (4.45) of the complex even-stacked EMFB subfilters, we are allowed to discard $M - 1$ subsignals as redundant data and denote/express the impulse responses of the analysis and synthesis subfilters
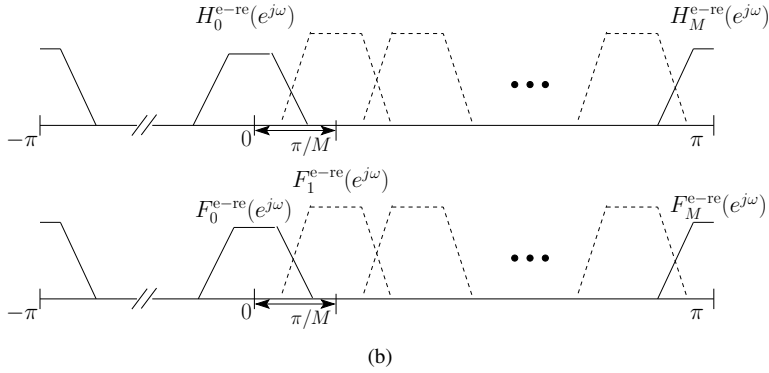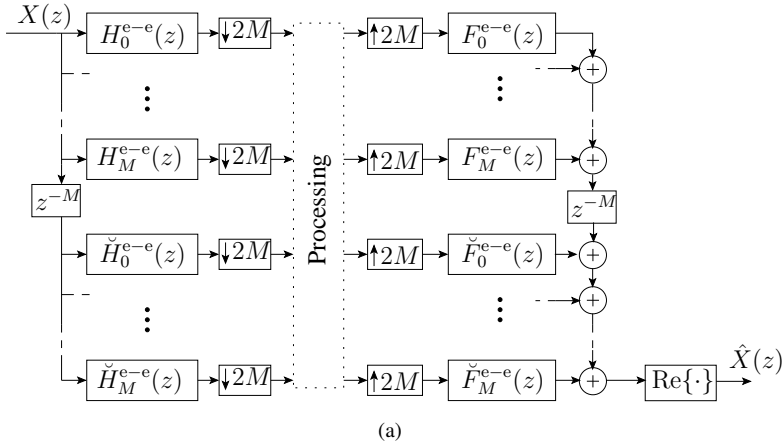
Figure 4.7: (a) Even-Stacked EMFB: Real-valued data and 2x oversampling. (b) The complex filter bank has one-sided spectrum (i.e., the passbands of $M + 1$ subfilters cover only the positive frequency side). Bandpass analysis subfilters are unscaled but the high- and low-pass subfilters are scaled by $1/\sqrt{2}$ to preserve the PR.

as

$$h_k^{\text{e-re}}[n] = c_k\, h_k^{\text{e-e}}[n] \tag{4.52}$$

$$f_k^{\text{e-re}}[n] = c_k\, f_k^{\text{e-e}}[n] \tag{4.53}$$

$$\breve{h}_k^{\text{e-re}}[n] = c_k\, \breve{h}_k^{\text{e-e}}[n] \tag{4.54}$$

$$\breve{f}_k^{\text{e-re}}[n] = c_k\, \breve{f}_k^{\text{e-e}}[n] \tag{4.55}$$

where $k = 0, 1, \ldots, M$ and

$$c_k = \begin{cases} 1/\sqrt{2} & k = 0 \text{ or } k = M \\ 1 & \text{otherwise.} \end{cases}$$

In the resulting $(M + 1)$-channel model, all bandpass subfilters have the same mainlobe level. Actually, the filter bank takes only the positive side of the input signal spectrum. The low/high-pass filters are "on the edge" and they are downscaled by a factor of $\sqrt{2}$ to satisfy

the PR condition. In addition, when the signal is reconstructed with $(M + 1)$-band filter bank, we have to include the real-part operation while generating the final output signal in order to satisfy the PR. This even-stacked EMFB is shown in Figure 4.7 (a). An alert reader would now make an observation that the FB is actually $2(M + 1)/M$ times oversampled. This is not the case because high/low-pass filters are purely real- or purely imaginary-valued. Due to the fact that the input data is real, the corresponding two subsignals are real/imaginary and the overall subsample rate in the FB is exactly two times the input rate.

Each of the remaining bandpass filters has a complex-valued impulse response and one mainlobe within $[-\pi, \pi]$, thereby giving rise to the one-sided FB spectrum as shown in Figure 4.7 (b). The scaling of the 0th and Mth subfilters is somewhat curious. Anyway, these arrangements are required when the output signal is reconstructed using the $(M + 1)$-band system shown in Figure 4.7 (a). Practically speaking, we do not suggest this kind of filter bank implementation that the theory offers. A better idea for processing of real input signals with a 2x oversampled even-stacked subband system is described as

1. Use a $2M$-band even-stacked analysis EMFB to generate the 2xOS subsignals.

2. Due to the coefficient redundancy, only $M + 1$ subsignals are necessary for the subband processing algorithms. If necessary, the alternating real/imaginary sequences from the high/low-pass filters are taken into account.

3. Reconstruct the signal by (conjugate)-mirroring the missing $M - 1$ coefficients and direct the subsignals to the synthesis EMFB algorithm.

4.6 Even-Stacked Cosine-Modulated Filter Bank

If the input signal of the even-stacked CS EMFB is purely real-valued, then the imaginary parts of the subfilters can be left out in filtering operation. Moreover, the symmetry properties in (4.44) and (4.45) make subchannels for $k = M + 1, M + 2, \ldots 2M - 1$ (the channels whose mainlobes occur on the negative frequency half) redundant. What remains is actually a critically sampled even-stacked CMFB. The subfilters of this real-valued FB are contained in the following even-stacked EMFB equations:

$$
\begin{aligned}
f_k^{c-e}[n] &= c_k \operatorname{Re}\{f_k^{e-e}[n]\} \\
h_k^{c-e}[n] &= c_k \operatorname{Re}\{h_k^{e-e}[n]\} \\
\breve{f}_k^{c-e}[n] &= c_k \operatorname{Re}\{\breve{f}_k^{e-e}[n]\} = c_k \operatorname{Re}\{j\, f_k^{e-e}[n]\} \\
\breve{h}_k^{c-e}[n] &= c_k \operatorname{Re}\{\breve{h}_k^{e-e}[n]\} = c_k \operatorname{Re}\{-j\, h_k^{e-e}[n]\}
\end{aligned}
$$

where $k = 0, 1, \ldots, M$. This reduces the subband system into $(M + 1)$-channel FB with two sets of subfilters. High/low-pass subchannels have half the bandwidth of the passband subchannels and they are scaled differently to satisfy the PR condition. Depending on the parity of $\tau - M$, we have the following two types of configurations:

Type I: Even-Stacked CMFB; $\tau - M$ even

In this case, the even-stacked CMFB consists of two "parallel" subband systems where the first set has $M + 1$ subfilters and the second one contains $M - 1$ bandpass subfilters. To ensure that

Figure 4.8: (a) Type I even-stacked CMFB: Real-valued data and critical sampling, $\tau - M$ even. (b) Subchannelization with two sets of filters; the first set contains both the high- and low-pass filters. The second subfilter set consists only of bandpass filters.

this holds, the generated subfilters are written open when $k = 0$ and $k = M$. Immediately, we notice that

$$f_0^{\mathrm{c-e}}[n] = \sqrt{2}\,\mathrm{Re}\big\{h[n]\,\exp(0)\big\} = \sqrt{2}\,h[n]$$

$$\breve{f}_0^{\mathrm{c-e}}[n] = \sqrt{2}\,\mathrm{Re}\big\{j\,h[n]\big\} = 0.$$

The high-pass filter belongs to the first set because the argument of the real-part operation in

$$f_M^{\mathrm{c-e}}[n] = c_M\,\mathrm{Re}\big\{f_M^{\mathrm{e-e}}[n]\big\} = \sqrt{2}\,\mathrm{Re}\Big\{h[n]\,\exp\Big[j\Big(n - \frac{N + D - M}{2}\Big)\pi\Big]\Big\}$$

$$= \sqrt{2}\,\mathrm{Re}\{h[n]\,\underbrace{\exp(j\pi n)}_{1,-1,1,-1,\ldots}\,\exp\big(-j\pi\,\underbrace{(N + D - M)/2}_{\text{integer}}\big)\}$$

is purely real-valued. The second high-pass filter vanishes:

$$\breve{f}_M^{\mathrm{c-e}}[n] = \sqrt{2}\,\mathrm{Re}\big\{j\,f_M^{\mathrm{e-e}}[n]\big\} = 0.$$

Thus the total number of subchannels is $(M+1)+(M-1) = 2M$ which equals the decimation factor of the system and there is no sample rate growth, as is desired.

It is orderly to process the even-stacked CMFB equations into a more familiar form; the equations for the first subfilter set are

$$f_k^{\mathrm{c-e}}[n] = c_k\, h[n] \cos\left[\left(n - \frac{N+D-M}{2}\right)\frac{k\pi}{M}\right] \tag{4.56}$$

$$h_k^{\mathrm{c-e}}[n] = c_k\, h[n] \cos\left[\left(n - \frac{N+D+M}{2}\right)\frac{k\pi}{M}\right], \tag{4.57}$$

where $k = 0, 1, \ldots, M$ and

$$c_k = \begin{cases} \sqrt{2} & k = 0 \text{ and } k = M \\ 2 & \text{otherwise.} \end{cases}$$

The equations for the remaining half - actually little less, i.e., $M - 1$ subfilters - can be written as

$$\breve{f}_k^{\mathrm{c-e}}[n] = -2\, h[n] \sin\left[\left(n - \frac{N+D-M}{2}\right)\frac{k\pi}{M}\right] \tag{4.58}$$

$$\breve{h}_k^{\mathrm{c-e}}[n] = 2\, h[n] \sin\left[\left(n - \frac{N+D+M}{2}\right)\frac{k\pi}{M}\right], \tag{4.59}$$

where $k = 1, 2, \ldots, M - 1$. The flowgraph and subband splitting scheme of the Type-I even-stacked CMFB are presented in Figures 4.8 (a) and (b), respectively. Subfilters $h_k^{\mathrm{c-e}}[n]$ include both high- and low-pass filters and actually the amplitude responses of these two filters are "head over the others". Tyranny of the masses prevails amongst subfilters $\breve{h}_k^{\mathrm{c-e}}[n]$ as all of them are bandpass filters with the same level amplitude responses.

Type II: Even-Stacked CMFB; $\tau - M$ odd

The slight difference between Type I and Type II is caused by the parity of $\tau - M$. When this parity is odd, then the high-pass filter is moved to the second subfilter set. This can be reasoned as above by noticing that $\exp(j\pi(N + D - M)/2)$ is now either j or $-j$. Thus, the subfilter impulse response equations are

$$f_k^{\mathrm{c-e}}[n] = c_k\, h[n] \cos\left[\left(n - \frac{N+D-M}{2}\right)\frac{k\pi}{M}\right] \tag{4.60}$$

$$h_k^{\mathrm{c-e}}[n] = c_k\, h[n] \cos\left[\left(n - \frac{N+D+M}{2}\right)\frac{k\pi}{M}\right] \tag{4.61}$$

where $k = 0, 1, \ldots, M - 1$ and

$$\breve{f}_k^{\mathrm{c-e}}[n] = -c_k\, h[n] \sin\left[\left(n - \frac{N+D-M}{2}\right)\frac{k\pi}{M}\right] \tag{4.62}$$

$$\breve{h}_k^{\mathrm{c-e}}[n] = c_k\, h[n] \sin\left[\left(n - \frac{N+D+M}{2}\right)\frac{k\pi}{M}\right] \tag{4.63}$$

where $k = 1, 2, \ldots, M$. Note the different ranges of the subchannel index k in the above definitions. The flowgraph and subchannelization of the Type-II even-stacked CMFB is shown
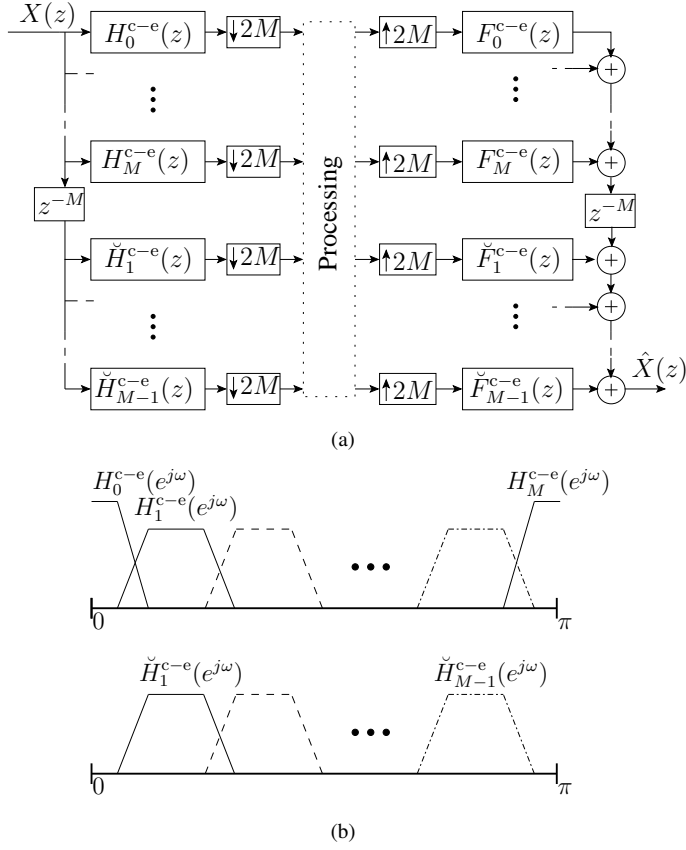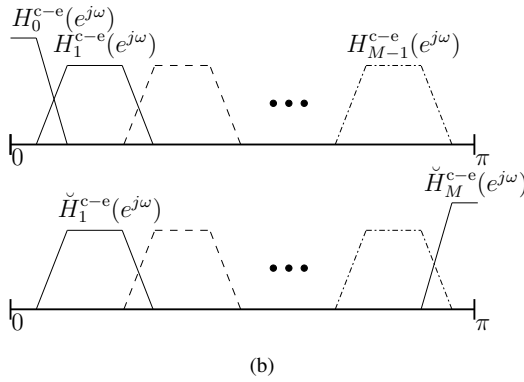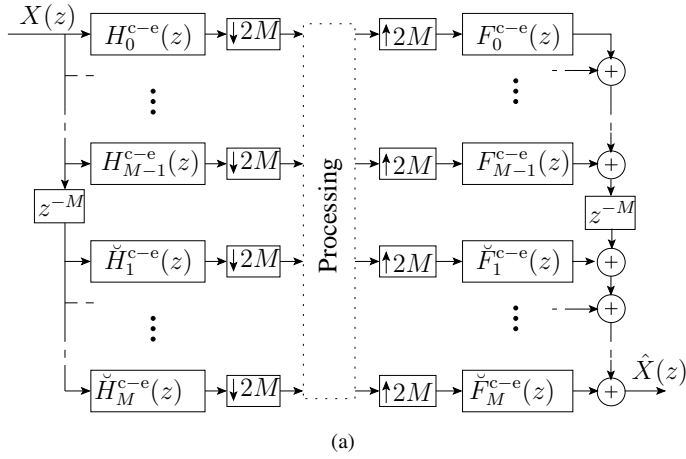
Figure 4.9: (a) Type II even-stacked CMFB: Real-valued data and critical sampling, $\tau - M$ odd. (b) Subchannelization with two FBs each having M-bands; low- and high-pass filters belong to different subfilter sets.

in Figure 4.9 (a) and (b), respectively. There are M members in the sets consisting of the subfilters $h_k^{\mathrm{c-e}}[n]$ and $\breve{h}_k^{\mathrm{c-e}}[n]$, but the high- and low-pass filters belong to different sets. From the Orwellian viewpoint on the amplitude responses, it looks that all mainlobes are equally scaled, but some mainlobes are more equal than the others.

Chapter 5

Prototype Filter Design for Arbitrary-Delay Modulated Filter Banks

The motivation for developing arbitrary-delay subband systems comes from the needs of low-latency applications, such as audio compression for teleconferencing software, or autonomous devices, the response of which is triggered by radar or pattern recognition data. The former compression of speech is a prime example with conflicting requirements. A large number of subbands would be preferable for lossy compression, i.e., discarding numerous subsamples having small magnitude and, consequently, low information content. However, in real-time systems, there is an upper limit in the number of subbands due to the sample buffering and processing delay. This is a favorite playground for the low-delay prototype filter designs.

Before proceeding into the details, it is stressed that some optimization work presented in this chapter is intended only to support and numerically verify the theoretical results of the previous MFB chapter. For instance, some parameter combinations lead to abysmal results in terms of attenuation, but they are practically "floating point PR systems". What hopes to renew our faith in the polyphase-domain PR condition in Section 2.6.2 - although such a general form might be interesting as a theoretical construction only?

The development of efficient and robust optimization methods for modulated FBs has provoked the curiosity of many researchers [39, 48, 56, 64, 68, 80, 82]. The vast optimization topic is out of the scope of this thesis, but the interested reader should check [99] for detailed discussion on various methods for designing modulated filter banks. In this chapter, we formulate optimization problems on a general level for single prototype paraunitary and biorthogonal (arbitrary-delay) designs. Both ℓ_2 and ℓ_∞-norms are covered. The PR constraints are given to the LM-length prototype filter, but it is only a matter of programming trickery to cover arbitrary-length designs as well. Furthermore, we introduce an optional modulation-dependent DC-leakage constraint for applications where the constant level of the signal needs to be captured.

Selected example cases are solved using the Optimization Toolbox [62] provided by the MathWorks, Inc. The quality of the MFB designs is observed from the amplitude response
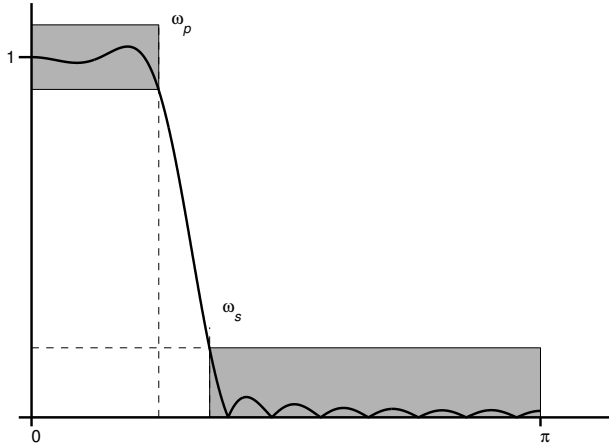
Figure 5.1: Frequency domain specifications for a low-pass filter design; the baseband for the amplitude response is divided into the passband and stopband, as well as the transition band in between them.

curves (stopband attenuation and passband deviation from unity) and by measuring the aliasing/reconstruction errors with the aid of appropriate formulae. These concrete design examples support the analytic and more abstract treatment of the MFBs in Section 4.

In addition, we discuss briefly the possibilities offered by 2x oversampling in EMFB designs. The relaxation of constraints leads to new design opportunities for 2xOS EMFBs; the analysis and synthesis subfilters can be generated from distinct prototypes and the choice of parameters is more flexible than with critically sampled MFBs. These claims are backed with illustrative example cases.

5.1 Design Problem Formulation

5.1.1 Objective Functions for Filter Bank Design

Typically, the minimization problem in the conventional filter design is stated in the frequency domain. For this purpose, the baseband for the amplitude response of the filter is divided into three regions: passband, stopband, and transition ("don't care") band. In this thesis, we need only low-pass filter specification as shown in Figure 5.1. The frequency bands are thus determined by the passband and stopband edges, ω_p and ω_s for later use. Since the prototype filter impulse response is assumed to be real-valued, the passband region is $\Omega_p = [0, \omega_p]$, and the stopband region is $\Omega_s = [\omega_s, \pi]$, whereas the gap between them is the transition band.

Two widely used optimization criteria are based on the use of ℓ_2-norm and ℓ_∞-norm. These two norms can be used to minimize the weighted deviation of the filter amplitude response $|H(e^{j\omega})|$ from the desired response in the least squares or minimax sense, respectively. A more recent candidate for constructing the objective function is peak-constrained least-squares criterion [40]. It combines the ℓ_2-norm with a constraint that limits the stopband ripple. The peak-constrained designs are bypassed with this mention only.

One motivation to gather the various design goals is the MFB-encyclopedic nature of this

thesis. Secondly, it is intended to address optimization specialists and their curiosity. Despite the rudimentary form, it is hoped that some of the posed problem formulations appear as an interesting target for more elaborate design methods. In the sequel, we provide the detailed ℓ_2/ℓ_∞-norm equations in a programming friendly manner. Furthermore, the design problems are formulated separately for prototypes having non-symmetric (arbitrary-delay) and symmetric impulse responses. In the latter case, the reduced number of adjustable variables can be exploited in the objective function, thereby, speeding up the solver iterations.

The frequency response of the causal prototype filter is

$$H(e^{j\omega}) = \sum_{n=0}^{N} h[n]e^{-j\omega n}.$$

In general, all the coefficients $h[n]$ of the arbitrary-delay prototype filter are unknowns and, therefore, the filter has no special property that would lead to a simplified frequency response in the problem formulation. Further on, the twist in biorthogonal MFB designs is that moving farther away from the paraunitary reconstruction delay ($\tau = N$) tends to cause a passband overshoot; the lower the delay τ; the greater the overshoot. In order to compensate this phenomenon, there is a need to control the passband performance. Paraunitary FB designs with symmetric prototype filters are special cases with the following attractive properties. First, due to the coefficient symmetry, the number of unknowns for the odd order is halved. Second, their frequency response can be separated into the zero-phase frequency response and the phase term [80]: $H(e^{j\omega}) = H(\omega)e^{j\phi(\omega)}$, where $H(\omega)$ replaces the frequency response in the objective function and, consequently, significantly simplifies the overall FB synthesis.

The common form of the MFB design problems considered in this thesis is

$$\begin{aligned} \underset{\mathbf{h}}{\text{minimize}} \quad & F(\mathbf{h}) \\ \text{subject to} \quad & \mathbf{c}(\mathbf{h}) = \mathbf{0} \\ & \mathbf{d}(\mathbf{h}) \leq \mathbf{0}, \end{aligned}$$

where $F(\mathbf{h})$ is the ℓ_2 or ℓ_∞-norm objective function and $\mathbf{h}$ is a vector containing the prototype design variables. The objective function includes both passband and stopband regions and uses a weighting parameter to, typically, emphasize the latter for better MFB selectivity. A set of constraints limits the feasible region of $\mathbf{h}$: The prototype filter must satisfy the (ideal) PR constraints $\mathbf{c}(\mathbf{h}) = \mathbf{0}$, which are non-linear by nature. Optionally, the inequality constraints $\mathbf{d}(\mathbf{h}) \leq \mathbf{0}$ can be included to handle the DC-leakage.

In the MFB prototype filter objective function the location of the passband and stopband edges depend on the decimation factor M. The following parametrization with ρ_p and ρ_s[†] sets the passband and stopband edges in the unified manner regardless of M:

$$\Omega_\mathrm{p} = \big[0, \omega_\mathrm{p}\big] = \big[0, (\pi(1 - \rho_\mathrm{p}))/(2M)\big] \tag{5.1}$$

$$\Omega_\mathrm{s} = \big[\omega_\mathrm{s}, \pi\big] = \big[(\pi(1 + \rho_\mathrm{s}))/(2M), \pi\big]. \tag{5.2}$$

ℓ_2-Norm Objective Function for Non-Symmetric $h[n]$

With the above notations the ℓ_2-norm objective function is

$$F(\mathbf{h}) = (1 - W_\mathrm{s}) \int_0^{\omega_\mathrm{p}} |H(e^{j\omega}) - \sqrt{M}e^{-j\omega(N+D)/2}|^2 d\omega + W_\mathrm{s} \int_{\omega_\mathrm{s}}^{\pi} |H(e^{j\omega})|^2 d\omega, \tag{5.3}$$

[†]The parameter ρ_s that controls the stopband edge is also known as the roll-off factor.

where W_s is a weighting parameter that allows us to set the relative weight between the passband and stopband in an arbitrary manner since W_s is limited by $0 < W_s \leq 1$. Regarding the desired function in the passband in the above objective function, it is worth emphasizing two facts. First, $(N + D)/2$ is half the overall filter bank delay. Second, the constant $\sqrt{M}$ originates from the fact that when using synthesis interpolators (subfilters), preserving the signal level requires scaling these subfilters by M and the assumption that the analysis and synthesis subfilters are scaled in the same manner.

ℓ_∞-Norm Objective Function for Non-Symmetric $h[n]$

The objective function to be minimized in the ℓ_∞-norm case is

$$F(\mathbf{h})$$
$$= \max_{\omega \in \Omega_p \cup \Omega_s} \left\{ \omega \in \Omega_p : (1 - W_s) \big| H(e^{j\omega}) - \sqrt{M} e^{-j\omega(N+D)/2} \big| \right\} \cup \left\{ \omega \in \Omega_s : W_s \big| H(e^{j\omega}) \big| \right\}.$$
(5.4)

When using iterative optimization methods for minimizing the above objective function, we sample it along a sufficiently dense enough grid of discrete frequencies ω_i on $\Omega_p \cup \Omega_s$.

5.1.2 Perfect-Reconstruction Constraints for Prototype Filter

In shaping the polyphase domain theory for designing PR biorthogonal MFBs, we assumed that the prototype filter coefficients satisfy equations (4.17-4.18) or (4.19-4.20). However, a product of two polynomials is not an ideal starting point for involving them as a subroutine for an optimization solver. We prefer to use sum formulae akin to (4.23) where all indices are neatly limited with parameters derived from N, M, and D. However, processing equations (4.17-4.20) into solver friendly sum equations is a cumbersome task as we learned from the paraunitary constraint derivation in Section 4.2. A more tractable subset of PR constraints can be obtained if we fix the prototype filter length as LM ($L \in \mathbb{N}$). The loss of generality is later patched with design examples, where the constraint functions for LM-length prototypes are coaxed for handling arbitrary-length designs.

In the biorthogonal EMFB screenplay we now need a *Deus ex Machina* event to get the plot moving. The required constraint functions are listed in Table 5.1 and have been published in [7]. These equations were originally obtained by combining ELT type of modulation with the LT-theory in Section 3.2[‡]. This derivation was excluded from the manuscript of this thesis during the writing process and for good reason. However, the constraint functions are still applicable despite the arbitrary-length generalization. The demonstration is done by optimizing a set of prototypes but we need first tools to measure the MFB imperfections.

Table 5.1 presents the constraints for the biorthogonal case, where the reconstruction delay $\tau = N+D = LM-1+D$ and $D \in \{0, \pm 1, \ldots, \pm(L-1)M\}$. Further on, the adjustable delay parameter is split into $D = Mr + d$ ($r = \lfloor D/M \rfloor$). Depending on the parity of r, we have two sets of constraints denoted with $B^\mathrm{I} - B^\mathrm{IV}$. Beautiful sum equations, which straightforwardly translate into a few nested for-loops.

[‡]It is worth pointing out that the desired passband value remains to be $\sqrt{M}$ in case where the scaling term $1/(2M)$, instead of $1/M$, is user in the RHS of the constraint equations.

Table 5.1: Prototype filter PR conditions for LT-based arbitrary-delay CMFBs.

Constraint Equation	Range of s	Range of n
BIORTHOGONAL with $D \in \{0, \pm 1, \pm 2, \ldots, \pm(L-1)M\}$, $D = Mr + d$ where $r = \lfloor D/M \rfloor$, $c_s = (-1)^{s + \lceil r/2 \rceil}$		$/(2M)$
Even r		
$B^{\mathrm{I}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n + d + iM]\, h[LM - 1 - (n + (i + 2s)M)] = c_s \delta[-2s - r]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil \frac{M-d}{2} \rceil - 1$
$B^{\mathrm{II}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n + iM]\, h[LM - 1 - (n + M - d + (i + 2s + 1)M)] = -c_s \delta[-2s - 2 - r]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil d/2 \rceil - 1$
$B^{\mathrm{III}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n + d + (i + 2s)M]\, h[LM - 1 - (n + iM)] = c_s \delta[2s - r]$	$1, 2, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil \frac{M-d}{2} \rceil - 1$
$B^{\mathrm{IV}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n + (i + 2s + 1)M]\, h[LM - 1 - (n + M - d + iM)] = c_s \delta[2s - r]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil d/2 \rceil - 1$
Odd r		
$B^{\mathrm{I}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n + iM]\, h[LM - 1 - (n - d + M + (i + 2s)M)] = -c_s \delta[-2s - r - 1]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil d/2 \rceil - 1$
$B^{\mathrm{II}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n + d + iM]\, h[LM - 1 - (n + (i + 2s + 1)M)] = -c_s \delta[-2s - r - 1]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil \frac{M-d}{2} \rceil - 1$
$B^{\mathrm{III}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n + (i + 2s)M]\, h[LM - 1 - (n - d + M + iM)] = -c_s \delta[2s - r - 1]$	$1, 2, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil d/2 \rceil - 1$
$B^{\mathrm{IV}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n + d + (i + 2s + 1)M]\, h[LM - 1 - (n + iM)] = c_s \delta[2s + 1 - r]$	$0, 1, \ldots, \lceil \frac{L}{2} \rceil - 1$	$0, 1, \ldots, \lceil \frac{M-d}{2} \rceil - 1$

Table 5.2: Special cases of the prototype filter PR conditions obtained from Table 5.1.

Constraint Equation	Range of s	Range of n
BIORTHOGONAL with L even and $D \in \{\pm 2M, \pm 4M, \ldots, \pm(L-2)M\}$, $D = Mr$ where r even, $c_s = (-1)^{s+\lfloor r/2 \rfloor}/(2M)$		
$\hat{B}^{\mathrm{I}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n+iM]\, h[LM-1-(n+(i+2s)M)] = c_s\delta[-2s-r]$	$0,1,\ldots,L/2-1$	$0,1,\ldots,\lceil\frac{M}{2}\rceil-1$
$\hat{B}^{\mathrm{II}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n+(i+2s)M]\, h[LM-1-(n+iM)] = c_s\delta[2s-r]$	$1,2,\ldots,L/2-1$	$0,1,\ldots,\lceil\frac{M}{2}\rceil-1$
BIORTHOGONAL with L odd and $D \in \{\pm M, \pm 3M, \ldots, \pm(L-2)M\}$, $D = Mr$ where r odd, $c_s = (-1)^{s+\lfloor r/2 \rfloor}/(2M)$		
$\hat{B}^{\mathrm{I}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n+iM]\, h[LM-1-(n+(i+2s+1)M)] = -c_s\delta[-2s-r-1]$	$0,1,\ldots,\frac{L-1}{2}-1$	$0,1,\ldots,\lceil\frac{M}{2}\rceil-1$
$\hat{B}^{\mathrm{II}} : \displaystyle\sum_{i=0}^{L-2-2s} h[n+(i+2s+1)M]\, h[LM-1-(n+iM)] = c_s\delta[2s+1-r]$	$0,1,\ldots,\frac{L-1}{2}-1$	$0,1,\ldots,\lceil\frac{M}{2}\rceil-1$
PARAUNITARY with $D = 0$, $h[n] = h[LM-1-n]$, $c_s = 1/(2M)$		
$P^{\mathrm{I}} : \displaystyle\sum_{i=0}^{L-1-2s} h[n+iM]\, h[n+(i+2s)M] = c_s\delta[s]$	$0,1,\ldots,\lceil L/2\rceil-1$	

80

Certain selections of the biorthogonal design parameters are of particular interest and the other half of the equations sets in Table 5.1 can be omitted. These correspond to cases where the RHS of the polyphase matrix PR condition in equations (2.25) is diagonal and $N + 1 = LM$. In addition, the parity of L sets requirements for D: when L is even (odd), the delay parameter D must be an even (odd) multiple of M. These choices provide the simplified constraints denoted with $\hat{B}^{\mathrm{I}} - \hat{B}^{\mathrm{II}}$ in Table 5.2. Furthermore, in Section 4.2, we have already discussed on the requirements of paraunitarity in terms of prototype filter coefficients. The last entry P^{I} in Table 5.2 covers that case and, additionally, handles odd L.

5.1.3   Optional MFB Design Constraint: DC-Leakage

The zero DC-leakage is the property of those filter banks or block transforms which perfectly capture the DC-component (i.e., the constant level of the signal). This property is exactly achieved by an M-channel PR FB provided that

$$\left| \sum_{n=0}^{N} h_k[n] \right| = \sqrt{M}\delta[k], \tag{5.5}$$

where $k = 0, 1, \ldots, M - 1$. This equation can be utilized in the MFB optimization problem as an optional constraint, where the $k = 0$ case is already handled by the PR constraints. Let us denote by δ_{dc} the tolerance of the DC-leakage when the low-pass subfilter is omitted. The modulation type is first fixed (odd-stacked CMFB/EMFB or even-stacked CMFB) as the choice selects the applied DC-leakage formula. The modulation-dependent DC-leakage constraints obtained by refining equation (5.5) are considered next for various types of MFBs.

Odd-Stacked CMFB and Paraunitary EMFB

For the PR odd-stacked CMFB, we obtain $M - 1$ inequality constraints:

$$-\delta_{\mathrm{dc}} \leq \sum_n h_k^{\mathrm{c}}[n] \leq \delta_{\mathrm{dc}},$$

where $k = 1, 2, \ldots, M - 1$. This holds for both paraunitary and arbitrary-delay CMFBs. Additionally, the time-reversing property of the paraunitary odd-stacked EMFBs causes these condition be sufficient for controlling the zero DC-leakage without the sine-modulated constraint due to the fact that $(-1)^k h_k^{\mathrm{c}}[n] = f_k^{\mathrm{s}}[n] = h_k^{\mathrm{s}}[N - n]$. However, the scaling factor $\sqrt{2}$ should be included in the constraint sums as $(1/\sqrt{2}) \sum_n h_k^{\mathrm{c}}[n]$ so that the tolerance levels become $|\sum_n h_k^{\mathrm{e}}[n]| \leq \delta_{\mathrm{dc}}$, where $k = 1, 2, \ldots, 2M - 2$. It should be noted that when utilizing odd-stacked exponential modulation, the DC-component is inherently captured within the given tolerances by subfilters $h_0^{\mathrm{e}}[n]$ and $h_{2M-1}^{\mathrm{e}}[n]$.

Even-Stacked CMFB

For even-stacked CMFBs, there are the following $2M - 1$ inequality constraints:

$$-\delta_{\mathrm{dc}} \leq \sum_n h_k^{\mathrm{c-e}}[n] \leq \delta_{\mathrm{dc}}, \qquad \begin{cases} k = 1, 2, \ldots, M & \text{Type I} \\ k = 1, 2, \ldots, M - 1 & \text{Type II}, \end{cases}$$

$$-\delta_{\mathrm{dc}} \leq \sum_n \breve{h}_k^{\mathrm{c-e}}[n] \leq \delta_{\mathrm{dc}}, \qquad \begin{cases} k = 1, 2, \ldots, M-1 & \text{Type I} \\ k = 1, 2, \ldots, M & \text{Type II.} \end{cases}$$

In other words, only the low-pass subfilter is excluded from the set of zero DC-leakage constraints. For backtracking purposes, one can find the above alternative subchannel configurations in Figures 4.8 and 4.9.

Odd-Stacked EMFB with Arbitrary-Delay Prototypes

The zero DC-leakage for the odd-stacked EMFB can be forced with real-valued MFB subfilter equations. If we generate the corresponding odd-stacked cosine- and sine-modulated subfilters ($h_k^{\mathrm{c}}[n]$ and $h_k^{\mathrm{s}}[n]$), we get

$$-\delta_{\mathrm{dc}} \leq \frac{1}{\sqrt{2}} \sum_n h_k^{\mathrm{c}}[n] \leq \delta_{\mathrm{dc}}$$

and

$$-\delta_{\mathrm{dc}} \leq \frac{1}{\sqrt{2}} \sum_n h_k^{\mathrm{s}}[n] \leq \delta_{\mathrm{dc}},$$

where $k = 1, 2, \ldots, M-1$. The scaling term is again necessary so that $|\sum_n h_k^{\mathrm{e}}[n]| \leq \delta_{\mathrm{dc}}$ with non-symmetric prototypes as well.

There are also two interesting zero DC-leakage cases when $h[n]$ is a parametrized prototype window function. This topic will be revisited in Section 6.1.1.

5.2 Methods to Evaluate the Quality of MFB Design

5.2.1 Transfer Functions for EMFB-Based Systems

The analytic polyphase domain study of the MFBs assumed that the PR condition holds exactly. Though this is true for MFB implementations that are structurally perfectly reconstructing, the condition is commonly compromised in practical implementations. Sources of distortion are coefficient rounding, intentional nearly-perfect-reconstruction designs, and truncating the fractional part in floating point arithmetic. Relevant to this chapter is the residual error in PR MFB designs. When the optimization algorithm finds a feasible point, it means that the constraints in Table 5.1 are satisfied - or actually the numeric value is bounded with tight tolerance value. Usually, we are content with a moderate level of MFB self-interference that does not yet degrade the filtering performance. In the design examples to be considered in the following, the tolerances to the aliasing and reconstruction errors are roughly 10^{-14} being definitely small enough.

To numerically measure the MFB imperfections, we return to M-channel FB theory to find the required tools. The transfer function becomes useful once it is modified to match the various modulation types. The basic formulae for distortion and aliasing can then be adapted accordingly.

Transfer functions were encountered in Section 2.6.1, where the PR conditions for M-channel filter banks were discussed: $T_i(z)$ describes the input-output relation of the filter bank in the z-transform domain. For M-channel filter banks it is given as

$$T_i(z) = \frac{1}{M} \sum_{k=0}^{M-1} F_k(z) \, H_k(zW_M^{-i}), \qquad (5.6)$$

Table 5.3: Transfer functions for odd-stacked modulated filter banks.

FB type	Transfer Function $T_i(z)$
CMFB	$\dfrac{1}{M} \displaystyle\sum_{k=0}^{M-1} F_k^{\text{c}}(z)\, H_k^{\text{c}}(zW_M^{-i})$
SMFB	$\dfrac{1}{M} \displaystyle\sum_{k=0}^{M-1} F_k^{\text{s}}(z)\, H_k^{\text{s}}(zW_M^{-i})$
CS EMFB	$\dfrac{1}{M} \displaystyle\sum_{k=0}^{2M-1} F_k^{\text{e}}(z)\left(H_k^{\text{e}}(zW_M^{-i}) + \left(H_k^{\text{e}}(z^*W_M^{i}) \right)^* \right)$
2xOS EMFB complex signal	$\dfrac{1}{M} \displaystyle\sum_{k=0}^{2M-1} F_k^{\text{e}}(z)\, H_k^{\text{e}}(zW_M^{-i})$
2xOS EMFB real signal	$\dfrac{1}{M} \displaystyle\sum_{k=0}^{M-1} \left(F_k^{\text{e-r}}(z)\, H_k^{\text{e-r}}(zW_M^{-i}) + \left(F_k^{\text{e-r}}(z^*) \right)^* \left(H_k^{\text{e-r}}(z^*W_M^{i}) \right)^* \right)$
	PR condition: $T_i(z) = \delta[i]\, z^{-\tau}$, where $i = 0, 1, \ldots, M-1$

where the range of i depends on the decimation factor (here $i = 0, 1, \ldots, M-1$). When $i = 0$, equation (5.6) is called the *distortion transfer function*. When $i \neq 0$, we refer to $T_i(z)$ as the *aliasing transfer function*.

The transfer functions for the modulated filter banks considered in this thesis are summarized in Tables 5.3 and 5.4. These FBs are divided into two groups ($T_i(z)$ and $T_i^{\text{even}}(z)$ for odd and even-stacked, respectively) because of different decimation factors. All equations can be derived by applying the z-transform domain analysis discussed in Section 2.6.1. For example, the 2x oversampled even-stacked EMFB consists of two parallel filter banks and delays. The first branch (upper in Figure 4.6) has the delay element z^{-M} after the synthesis filters and hence it corresponds to

$$\frac{1}{2M}\, z^{-M} \sum_{k=0}^{2M-1} F_k^{\text{e-e}}(z)\, H_k^{\text{e-e}}(zW_{2M}^{-i}).$$

It should be noted that the decimation factor is $2M$ in this filter bank model and the $2M$th root of unity is used in the argument of $H_k^{\text{e-e}}(zW_{2M}^{-i})$. The second branch is described with

$$\frac{1}{2M} \left(zW_{2M}^{-i}\right)^{-M} \sum_{k=0}^{2M-1} \breve{F}_k^{\text{e-e}}(z)\, \breve{H}_k^{\text{e-e}}(zW_{2M}^{-i})$$

because the delay element z^{-M} is before analysis stage. Term $\left(zW_{2M}^{-i}\right)^{-M}$ reduces to

$$\left(zW_{2M}^{-i}\right)^{-M} = z^{-M} \exp\left\{ -j\frac{2\pi i M}{2M} \right\} = (-1)^i z^{-M}.$$

Then two branches are added together to get the entry corresponding to the 2x oversampled even-stacked EMFB in Table 5.4.

Table 5.4: Transfer functions for even-stacked EMFB variants.

FB type	Transfer Function $T_i^{\text{even}}(z)$
CS EMFB	$\dfrac{z^{-M}}{2M} \displaystyle\sum_{k=0}^{2M-1} F_k^{\text{e-e}}(z) \left(H_k^{\text{e-e}}(zW_{2M}^{-i}) + \left(H_k^{\text{e-e}}(z^*W_{2M}^{i}) \right)^* \right)$ $+ (-1)^i \breve{F}_k^{\text{e-e}}(z) \left(\breve{H}_k^{\text{e-e}}(zW_{2M}^{-i}) + \left(\breve{H}_k^{\text{e-e}}(z^*W_{2M}^{i}) \right)^* \right)$
2xOS EMFB	$\dfrac{z^{-M}}{2M} \displaystyle\sum_{k=0}^{2M-1} F_k^{\text{e-e}}(z) H_k^{\text{e-e}}(zW_{2M}^{-i}) + (-1)^i \breve{F}_k^{\text{e-e}}(z) \breve{H}_k^{\text{e-e}}(zW_{2M}^{-i})$
2xOS EMFB real signal	$\dfrac{z^{-M}}{2M} \displaystyle\sum_{k=0}^{M} \left(F_k^{\text{e-re}}(z) H_k^{\text{e-re}}(zW_{2M}^{-i}) + \left(F_k^{\text{e-re}}(z^*) \right)^* \left(H_k^{\text{e-re}}(z^*W_{2M}^{i}) \right)^* \right.$ $\left. + (-1)^i \breve{F}_k^{\text{e-re}}(z) \breve{H}_k^{\text{e-re}}(zW_{2M}^{-i}) + (-1)^i \left(\breve{F}_k^{\text{e-re}}(z^*) \right)^* \left(\breve{H}_k^{\text{e-re}}(z^*W_{2M}^{i}) \right)^* \right)$
Type-I CMFB	$\dfrac{z^{-M}}{2M} \left(\displaystyle\sum_{k=0}^{M} F_k^{\text{c-e}}(z) H_k^{\text{c-e}}(zW_{2M}^{-i}) + (-1)^i \sum_{k=1}^{M-1} \breve{F}_k^{\text{c-e}}(z) \breve{H}_k^{\text{c-e}}(zW_{2M}^{-i}) \right)$
Type-II CMFB	$\dfrac{z^{-M}}{2M} \left(\displaystyle\sum_{k=0}^{M-1} F_k^{\text{c-e}}(z) H_k^{\text{c-e}}(zW_{2M}^{-i}) + (-1)^i \sum_{k=1}^{M} \breve{F}_k^{\text{c-e}}(z) \breve{H}_k^{\text{c-e}}(zW_{2M}^{-i}) \right)$
	PR condition: $T_i^{\text{even}}(z) = \delta[i]\, z^{-(\tau+M)}$, where $i = 0, 1, \dots, 2M-1$

The transfer functions satisfying the PR differ for the odd- and even-stacked MFBs by z^{-M} such that they can be, respectively, expressed as

$$T_i(z) = \delta[i]\, z^{-\tau}, \qquad\qquad i = 0, 1, \ldots, M - 1 \qquad (5.7)$$

$$T_i^{\text{even}}(z) = \delta[i]\, z^{-(\tau + M)}, \qquad\qquad i = 0, 1, \ldots, 2M - 1. \qquad (5.8)$$

The "two parallel FBs"-model cause an extra M sample delay to the even-stacked EMFB system, which is taken into account in (5.8). The different decimation factor is seen in the range of i.

5.2.2 Error Measures for Distortion and Aliasing

Transfer functions can be applied to measure the distortion and aliasing of a non-ideal filter bank. Two frequency domain figures of performance are commonly used. Firstly, the peak-to-peak distortion E_{pp} is the deviation of the distortion transfer function $T_0(e^{j\omega})$ from the unity. Formally, it can be given as

$$E_{\text{pp}} = \delta_0 + \delta_1, \qquad (5.9)$$

where $(1 - \delta_0) \le |T_0(e^{j\omega})| \le (1 + \delta_1)$ and $\delta_0, \delta_1 \ge 0$. The range of ω is $[0, \pi]$ for real and $[-\pi, \pi]$ for complex filter banks.

Secondly, the aliasing error E_{a} is the maximum value of the curve defined with the aliasing transfer functions. Due to different decimation factor of even-stacked EMFBs, we have

$$E_{\text{a}} = \begin{cases} \displaystyle\max_{\omega} \frac{1}{M} \Big(\sum_{i=1}^{M-1} |T_i(e^{j\omega})|^2 \Big)^{1/2} \\[4mm] \displaystyle\max_{\omega} \frac{1}{2M} \Big(\sum_{i=1}^{2M-1} |T_i^{\text{even}}(e^{j\omega})|^2 \Big)^{1/2} \end{cases}. \qquad (5.10)$$

In the continuation, we shall apply these equations for E_{pp} and E_{a} when verifying the prototype filter designs.

5.3 Numerical Verification of Perfect-Reconstruction Constraints

5.3.1 Feasibility of Arbitrary-Delay PR Condition

The formulation of the biorthogonal EMFB theory relied on the perfect reconstruction property of the prototype filter. Up to this point, there has not been a demonstration concerning the existence of such prototype filters. In this section, the subject is explored by optimizing a collection of prototype filters and then measuring E_{pp} and E_{a}. In these considerations the achieved stopband attenuation is equally important factor so that poor parameter combinations can be ranked out single-handedly. We select the ℓ_2-norm objective and $W_{\text{s}} = 1$ in these PR verifications. The weighting $1 - W_{\text{s}} = 0$ rules out the passband from the minimization problem. Other parameters are: $\rho_{\text{s}} = 1.0$, $M = 4$, and $N = LM - 1 = 8M - 1$. Now we let the adjustable delay parameter to have values from a set $D \in \{-(4M + 2), \ldots, 0, 1, 2\}$ and
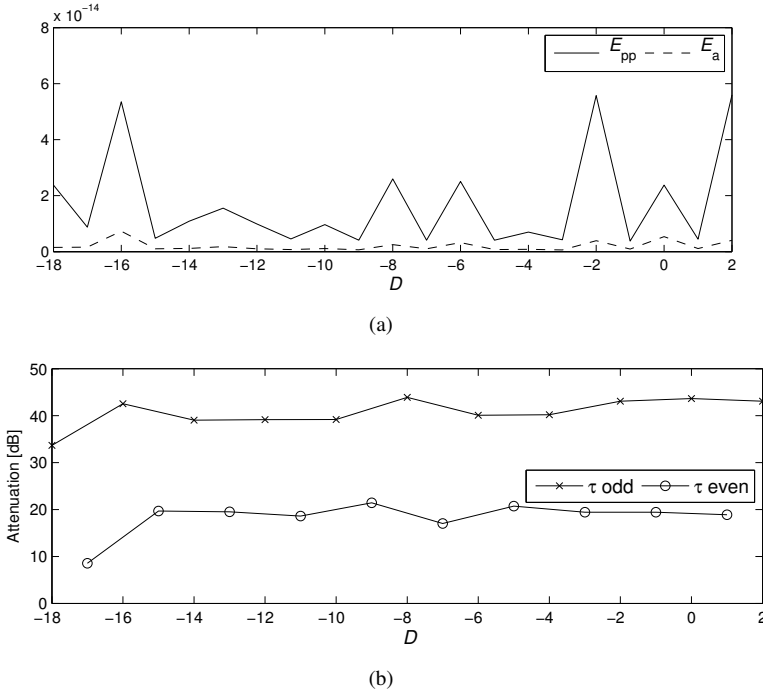
Figure 5.2: Checking the feasibility of the arbitrary-delay constraints by measuring the CMFB self-imperfection and one rule of thumb for τ selection: choose odd values. (a) Reconstruction and aliasing errors. (b) Attenuation (the highest sidelobe level).

solve the minimization problem for each value of D. The constrained optimization problem is thus

$$\underset{\mathbf{h}}{\text{minimize}} \quad F(\mathbf{h})$$
$$\text{subject to} \quad \mathbf{c}(\mathbf{h}, M, L, D) = \mathbf{0},$$

where the vector-valued $\mathbf{c}(\mathbf{h}, M, L, D)$ covers the equations $B^{\mathrm{I}} - B^{\mathrm{IV}}$ from Table 5.1. Alternatively, if the combination of parameters is suitable, we switch to constraint functions given in Table 5.2. The detailed expression for the objective function is postponed until Section 5.4.

Then we let the optimization routine to crunch numbers. This produces a collection of prototype filters with varying D and we compute the reconstruction and aliasing errors for each one using odd-stacked exponential modulation. The resulting E_{pp} and E_{a} curves are shown in Figure 5.2 (a) as a function of parameter D. Based on this single-shot experiment, we venture to say that the feasible region generated with the PR constraints is not empty. However, this is just the other side of the coin. If we also check the attenuation characteristics shown in Figure 5.2 (b), we observe that an even reconstruction delay is certainly miserable choice Intuitively, this seems to be one of the consequences of the PR constraint equations discussed in Section 5.3.2, for which further study is required. Secondly, when the delay parameter D gets values that are even multiples of M, the constraints are not so restrictive. The attenuations achieved with non-symmetric prototypes $\tau = N - 4M = 15$ and $\tau = N - 2M = 23$ are greater than with their closest odd τ neighbors.
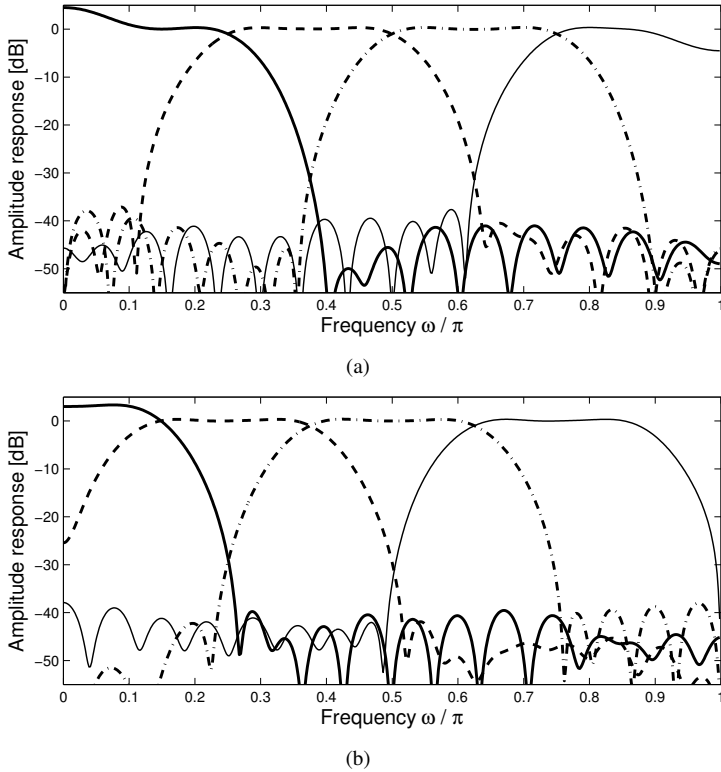
Figure 5.3: Difference between the alternative real-valued modulations when using a low-delay PR prototype with parameters $M = 4$, $N = 8M - 1$, and $D = -4M$ (odd-stacked over/undershoot not present in the paraunitary case) . (a) Odd-stacked $h_k^c[n]$. (b) Even-stacked $h_k^{c-e}[n]$.

One curious issue in arbitrary-delay designs is worth mentioning: the real-valued modulation of the CMFBs does not preserve the amplitude response of the prototype filter, whereas for the EMFBs, the modulation is based on using a pure frequency shift for the response of the prototype filter, thereby leaving the overall shape unaltered. Let us observe this from the $\tau = 15$ case of the previous example. Figure 5.3 (a) and (b) show the amplitude responses resulting from the odd- and even-stacked cosine-modulation, respectively. It is observed that the modulation sequence together with the prototype filter causes in the odd-stacked case a "noticeable" passband overshoot in the amplitude response of the lowpass analysis subfilter, whereas such a phenomenon does not occur in the even-stacked case.

5.3.2 Modification of Constraints to Cover Arbitrary-Length Designs

For the sake of completeness, we also check the reconstruction and aliasing errors when $N + 1$ is not an integer multiple of M. By tinkering the constraint equations $B^{\mathrm{I}} - B^{\mathrm{IV}}$, it is possible to design prototype filters suitable for this demonstration. The modifications for $\mathbf{c}(\mathbf{h}, M, L, D)$ are zero-padding and adjusting the delay parameter.

Arbitrary-Delay and Arbitrary-Length Constraints for $h[n]$

If the desired analysis-synthesis reconstruction delay is $\tau = N + D$ and $N \neq LM$ where $L = \lceil (N+1)/M \rceil$, we extend the prototype filter to LM-length vector:

$$\tilde{\mathbf{h}} = \left[\ \underbrace{h[0]\quad h[1]\cdots h[N]}_{\#(N+1)}\quad \underbrace{0\quad 0\cdots 0}_{\#\tilde{D}}\right]^{T}.$$

The length of the zero-pad $\tilde{D} = \lceil (N+1)/M \rceil M - (N+1)$. The modified constraint equation then becomes

$$\mathbf{c}(\tilde{\mathbf{h}}, M, \lceil (N+1)/M \rceil, D - \tilde{D}) = \mathbf{0}.$$

It is easier to tweak the optimization code in the described manner than find an analytic version of the arbitrary-length constraint equations.

Arbitrary-Length Constraints for Symmetric $h[n]$

If the symmetric prototype filter is of order $N \neq LM$ ($D = 0$ by definition), the objective function has $\lceil (N+1)/2 \rceil$ adjustable variables. In order to re-use the constraint function we expand the impulse response to full length using zero-padding:

$$\tilde{\mathbf{h}} = \left[\ \underbrace{h[0]\quad h[1]\cdots h[1]\quad h[0]}_{\#(N+1)}\quad \underbrace{0\quad 0\cdots 0}_{\#\tilde{D}}\right]^{T}.$$

Vector $\tilde{\mathbf{h}}$ is the argument for the modified constraint function

$$\mathbf{c}(\tilde{\mathbf{h}}, M, \lceil (N+1)/M \rceil, -\tilde{D}) = \mathbf{0}$$

that utilizes $B^{\mathrm{I}} - B^{\mathrm{IV}}$. Writing the optimization code with analytic gradients is somewhat tricky but it can be done ("MATLAB-proof" in [2]).

The reconstruction/aliasing error test is repeated with a symmetric prototype and $N \in \{4M - 3, \ldots, 8M + 1\}$. The results in Figure 5.4 (a) confirm that the arbitrary-length designs yield also negligible distortion measures. The stopband attenuation is shown in Figure 5.4 (b) as a function of $\tau = N$. The even and odd values of τ are plotted separately for apparent reason. It is now orderly to ponder this wide gap in results: in the even τ, even M case, the symmetric impulse response of the paraunitary prototype filter has a lonely center tap. In addition, the applied time-domain PR conditions force several impulse response coefficients to take on the value of zero. This is better observed with two design example impulse responses shown in Figure 5.5 (a) using $M = 16$ and an odd/even reconstruction delay. In the latter case, the PR constraints distort the smooth pulse shape thus causing a considerably poorer amplitude response, as shown in Figure 5.5 (b).

However, the question remains why the attenuation characteristics in Figure 5.4 (b) have a few downward steps when tracing the curves from left to right? These undesired downward steps occur even in redesigning the problematic Nth-order prototype filter using the "$(N-2)$th-order prototype initialization"-trickery§, which should result in the Nth-order prototype filter which has at least the same attenuation as its $(N-2)$th-order counterpart. The

---

§The initial impulse response of the Nth-order prototype filter for further optimization is generated based on the optimized impulse response $h[n]$ of the $(N-2)$th-order prototype by, first, using the shift $\tilde{n} = n + 1$, and, then, inserting zero-valued impulse response samples at both $\tilde{n} = 0$ and $\tilde{n} = N$.
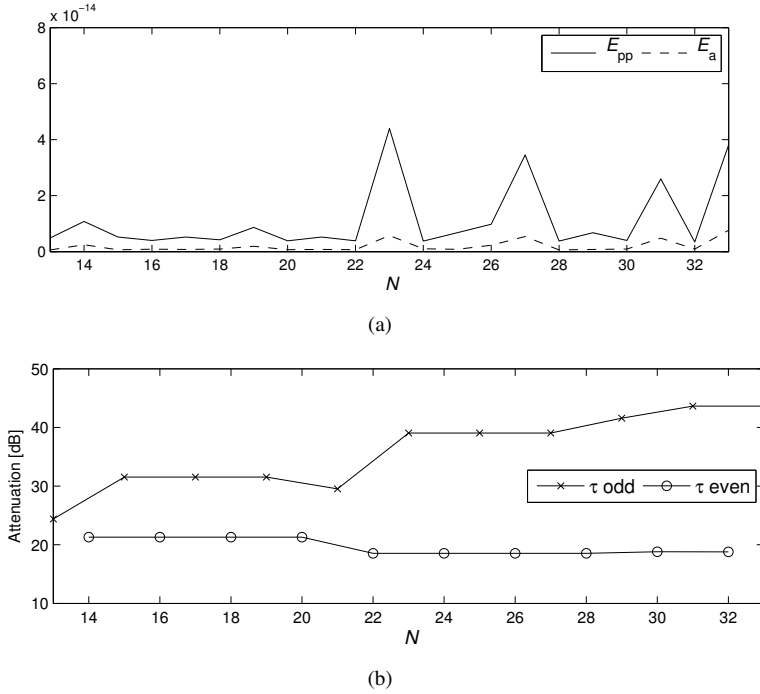
Figure 5.4: Checking the feasibility of the arbitrary-length constraints and highest sidelobe levels as a function of the filter order. (a) Reconstruction and aliasing errors. (b) Attenuation.

educated guess is that the non-linear PR constraints compel the general-purpose optimization algorithms, which are applied in this thesis, to drift to feasible local minima. Moreover, our constraint functions are hacked to handle arbitrary lengths by inserting dummy zeros, which from the programming point of view are computationally inefficient. Despite this, the main goal was to verify that the aliasing and reconstruction error levels become negligible so that we will follow the true path of the PR (and sometimes impractical) MFB theory.

In these numerical verifications of the PR feasibility, we have omitted the odd M case. The key reason is that it has been observed that for all existing MFB designs, there are no predetermined values for the even decimation factor M, whereas for M odd, the PR condition forces the prototype filter to have "zero-crossing" coefficients, as we shall see later on in equation (6.11) for an ELT window function. With this sidenote, we outline some rules of thumb to select the parameters:

- Select even M.

- Avoid such parameter combinations that results an even value for the analysis-synthesis reconstruction delay τ.

- For non-symmetric prototypes, favor filter lengths $N + 1 = LM$ and, depending on the parity of L, select D so that $\tau + 1 \mod 2M = 0$.

- For symmetric prototypes, favor filter lengths that are even multiples of M.
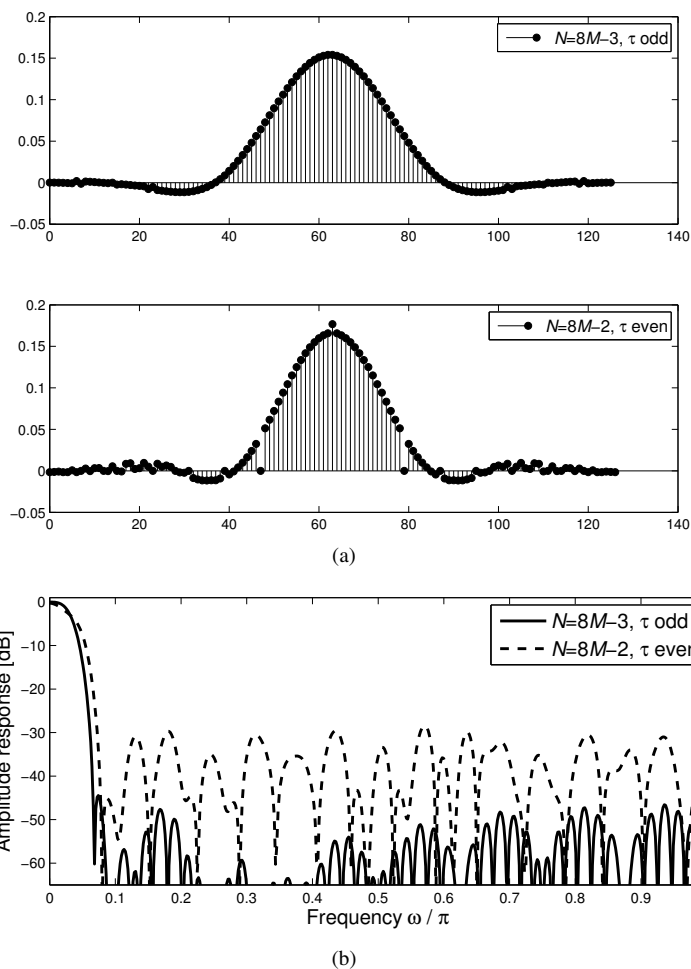
Figure 5.5: Comparison between two $M = 16$ prototype filters whose orders differ by one. (a) The impulse response of the $N = 8M - 2$ design is "punctured" by zero terms. (b) The matching amplitude responses.

5.4 Modulated Filter Bank Design Examples

This section covers in a fast-paced manner the applied objective functions - once again separately considering the arbitrary-delay and paraunitary MFB designs. The designer insights should guide in properly selecting the norm, stopband, and passband characteristics. For instance, ℓ_2-norm is preferable¶ in audio subband processing, where the rejection of distant aliasing components is important. Alas, even though the objective functions utilize these norms, the constraint functions may shape the sidelobe attenuation unexpectedly due to the non-linearities.

¶Although at least one notorious author advocates more sinister norms, such as $\ell_{2.4333}$, just to puzzle certain mathematically-oriented people.

5.4.1 Biorthogonal Design with ℓ_2-Norm Objective Function

For non-symmetric prototype filters, the integral of the objective function in (5.3) can be worked into the following closed form:

$$F(\mathbf{h}) = \mathbf{h}^T \left(W_s \mathbf{Q}_s + (1 - W_s) \mathbf{Q}_p \right) \mathbf{h} - 2(1 - W_s) \, \mathbf{p}_p^T \mathbf{h} + (1 - W_s) M \omega_p.$$

Here, the $(N + 1) \times (N + 1)$ matrices $\mathbf{Q}_s$ and $\mathbf{Q}_p$ have the Toeplitz structure and thus their 0th row elements are given by

$$\left[\mathbf{Q}_s \right]_{0,m} = \begin{cases} \pi - \omega_s & m = 0 \\ -\sin(m\omega_s)/m & m \neq 0 \end{cases}$$

$$\left[\mathbf{Q}_p \right]_{0,m} = \begin{cases} \omega_p & m = 0 \\ \sin(m\omega_p)/m & m \neq 0. \end{cases}$$

Other rows are produced by circular rotation. The vector term, in turn, has elements

$$\left[\mathbf{p}_p \right]_n = \begin{cases} \sqrt{M} \sin \left((n - \frac{N+D}{2}) \omega_p \right) / \left(n - \frac{N+D}{2} \right) & n \neq (N + D)/2 \\ \sqrt{M} \omega_p & n = (N + D)/2. \end{cases}$$

If the optimization algorithm utilizes gradients provided by the user, $\nabla F(\mathbf{h})$ can then be written as

$$\nabla F(\mathbf{h}) = \left(W_s \mathbf{Q}_s + (1 - W_s) \mathbf{Q}_p \right) \mathbf{h} - 2(1 - W_s) \, \mathbf{p}_p.$$

Generally speaking, the effort of programming these gradient functions in MFB design routines pays off in better optimization results compared with those being obtainable by computations with finite difference derivatives.

The biorthogonal ℓ_2-norm objective is now deployed by taking the passband under consideration. The parameter W_s is varied to visualize its effect to passband overshoot. The fixed parameters are: $M = 8$, $N = 8M - 1$, $D = -4M$, $\rho_s = 1.0$, and $\rho_p = 0.4$. The constraint functions also include the DC-leakage criterion with $\delta_{dc} = 10^{-4}$ so that the odd-stacked EMFB subfilters have the restriction $\left| \sum_n h_k^e[n] \right| \leq \delta_{dc}$ when $k = 0, 1, \ldots, 2M - 2$. Optimization results are shown in Figure 5.6. It can be seen that without passband weight the amplitude response has a tendency to overshoot near the passband edge. Better passband characteristics can be achieved at the expense of stopband attenuation.

5.4.2 Paraunitary Design with ℓ_2-Norm Objective Function

The objective function for prototype filters with symmetric impulse response can be expressed using the zero-phase frequency response as

$$F(\mathbf{h}) = (1 - W_s) \int_0^{\omega_p} \left| H(\omega) - \sqrt{M} \right|^2 d\omega + W_s \int_{\omega_s}^{\pi} H^2(\omega) d\omega.$$

This can be further expressed in terms of $\mathbf{p}_p$, $\mathbf{Q}_p$ and $\mathbf{Q}_s$ in the manner similar to that already performed for the non-symmetric prototype filter. The attractive feature is that now the zero-phase frequency response for N odd can be written as

$$H(\omega) = 2h[(N+1)/2]\cos(\omega/2) + 2h[(N+1)/2+1]\cos(3\omega/2) + \cdots + 2h[N]\cos(N\omega/2). \tag{5.11}$$
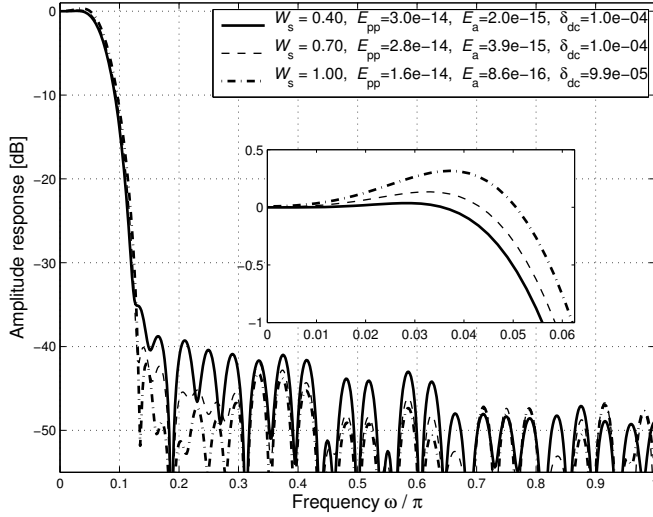
Figure 5.6: ℓ_2-norm design examples with non-symmetric prototype filters. Stopband attenuation can be traded to reduce passband overshoot.

Taking the above $H(\omega)$ into consideration allows us to halve the number of unknowns and, consequently, to end up with more accurate results with a reduced computation burden. For this purpose, the following formulae for the present paraunitary MFB design are used

$$
[\mathbf{Q}_s]_{n,m} = \begin{cases} 2\left(\pi - \omega_s - \frac{1}{2m+1}\sin\left((2m+1)\omega_s\right)\right) & n = m \\ -2\left(\frac{1}{n-m}\sin\left((n-m)\omega_s\right) + \frac{1}{m+n+1}\sin\left((m+n+1)\omega_s\right)\right) & n \neq m \end{cases}
$$

$$
[\mathbf{Q}_p]_{n,m} = \begin{cases} 2\left(\omega_p + \frac{1}{2m+1}\sin\left((2m+1)\omega_p\right)\right) & n = m \\ 2\left(\frac{1}{n-m}\sin\left((n-m)\omega_p\right) + \frac{1}{m+n+1}\sin\left((m+n+1)\omega_p\right)\right) & n \neq m \end{cases}
$$

$$
[\mathbf{p}_p]_n = \frac{2\sqrt{M}}{n+1/2}\sin\left((n+1/2)\omega_p\right).
$$

The coefficient vector contains the other half of the prototype impulse response in the following order: $\mathbf{h} = \left[h[(N+1)/2] \cdots h[N-1] \ h[N]\right]^T$. The objective function for prototype filters that have a single center tap (even N) also has the above described matrix formulation but we bypass the details as this option leads to quite poor FB performances.

The ℓ_2-norm paraunitary MFB design equations are then utilized to design a symmetric prototype with parameters $M = 8$, $N = 8M-1$, and $\rho_s = 1.1$. Figure 5.7 shows the amplitude responses of two optimized prototypes. The minimization problem for the first one includes passband objective ($\rho_p = 0.35$, $W_s = 0.5$) and the second one allocates all optimization efforts to minimize the stopband attenuation. The reconstruction and aliasing errors are computed with even-stacked EMFB subfilters. Figure 5.8 shows the amplitude responses of $f_k^{e-e}[n]$ generated with the $W_s = 1.0$ prototype filter. The subfilter impulse responses in Figure 5.9 demonstrate the symmetry property of the even-stacked EMFB. As seen in this figure, the real and imaginary parts of the subfilters are either symmetric or anti-symmetric, as is desired.
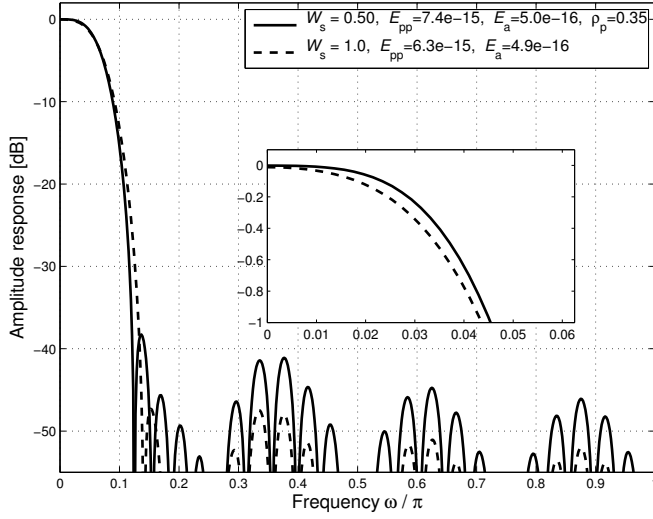
Figure 5.7: ℓ_2-norm design examples with symmetric prototype filters.
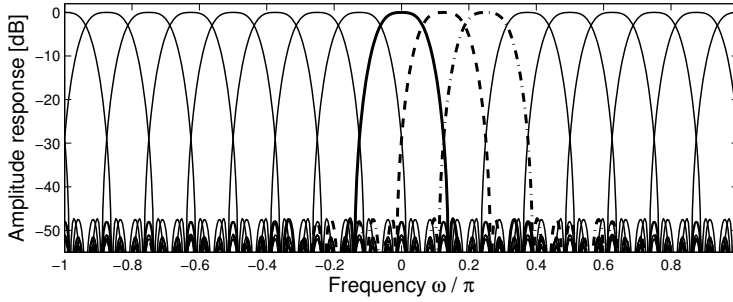
Figure 5.8: Even-stacked EMFB subchannelization using the symmetric prototype filter in Figure 5.7 (no passband shaping).

5.4.3 Biorthogonal Design with ℓ_∞-Norm Objective Function

As already mentioned, the frequency response of a non-symmetric FIR filter is not separable to the zero-phase frequency response and the phase term in the same manner as with symmetric filters. In order to overcome this obstacle we write the objective part of the minimization function as

$$\min_{\mathbf{h}} \max_{\omega \in \Omega_p \cup \Omega_s} W(\omega)\sqrt{|H(e^{j\omega}) - D(e^{j\omega})|^2},$$

where

$$D(e^{j\omega}) = \begin{cases} \sqrt{M}e^{-j(N+D)\omega/2} & \omega \in \Omega_p \\ 0 & \omega \in \Omega_s \end{cases}$$

$$W(\omega) = \begin{cases} 1 - W_s & \omega \in \Omega_p \\ W_s & \omega \in \Omega_s. \end{cases}$$

93
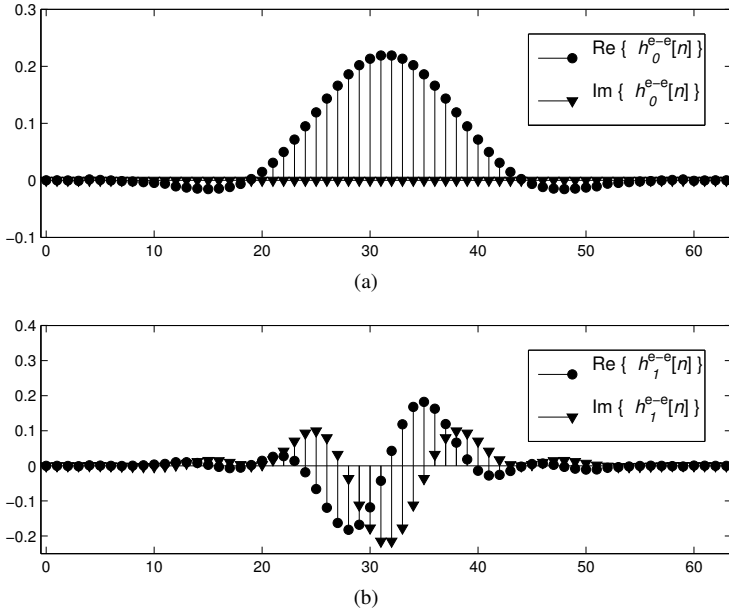
Figure 5.9: Even-stacked paraunitary EMFB has linear-phase subfilters.

Since the square root is a monotonically increasing function and its argument is always positive or zero, we opt to solve the following equivalent problem:

$$\min_{\mathbf{h}} \max_{\omega \in \Omega_\mathrm{p} \cup \Omega_\mathrm{s}} W^2(\omega)|H(e^{j\omega}) - D(e^{j\omega})|^2. \tag{5.12}$$

Alternatively, this problem can be re-written using a single matrix and three vectors as

$$|H(e^{j\omega}) - D(e^{j\omega})|^2 = \mathbf{h}^T \mathbf{Q}(\omega)\mathbf{h} - 2\,\mathbf{p}^T(\omega)\mathbf{h} + |D(e^{j\omega})|^2,$$

where $\mathbf{Q}(\omega)$ is the Toeplitz matrix with its 0th row defined by

$$\big[\mathbf{Q}(\omega)\big]_{0,m} = \cos(m\omega), \qquad m = 0, 1, \ldots, N.$$

The elements of the remaining vector $\mathbf{p}(\omega)$, in turn, are specified as

$$\big[\mathbf{p}(\omega)\big]_n = \begin{cases} \sqrt{M}\cos\big((n - \tfrac{N+D}{2})\omega\big) & \omega \in \Omega_\mathrm{p} \\ 0 & \omega \in \Omega_\mathrm{s}. \end{cases}$$

The bag of tricks for linear programming includes a method of writing ℓ_∞ problems as a set of linear constraints with the aid of additional variable. The idea is adaptable to the problem stated by equation (5.12) by expressing the objective function in the following manner not being exactly in the linear form:

$$\begin{aligned} \underset{\mathbf{h},\xi}{\text{minimize}} \quad & \xi \\ \text{subject to} \quad & \mathbf{h}^T \underbrace{W^2(\omega_i)\mathbf{Q}(\omega_i)}_{\mathbf{Q}_i} \mathbf{h} - 2\underbrace{W^2(\omega_i)\mathbf{p}^T(\omega_i)}_{\mathbf{p}_i^T} \mathbf{h} + W^2(\omega_i)|D(e^{j\omega_i})|^2 \leq \xi, \end{aligned}$$
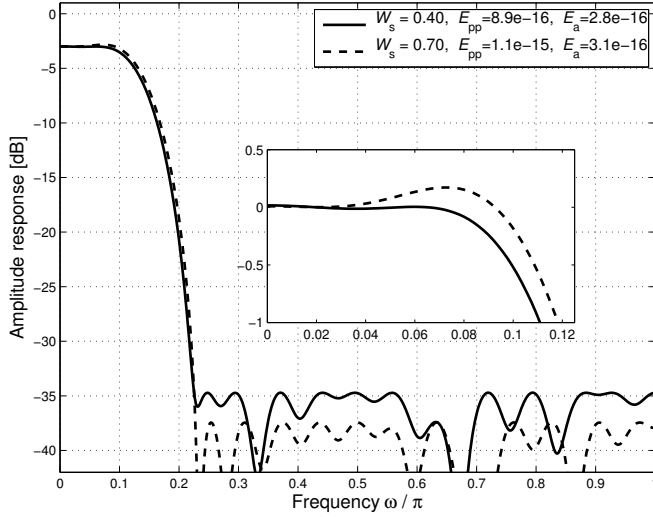
Figure 5.10: ℓ_∞-norm design examples with non-symmetric prototype filters.

where discretized frequency points ω_i form the set of grid points used in the optimization. These points are usually selected equidistantly and they should cover desnely enough both the passband Ω_p and Ω_s for guarenteeing the arrival at an accurate enough solution. In this respect, especially the sidelobe escape should be prevented. The problem can be streamlined into a non-linear minimization task with $N + 2$ variables

$$
\begin{aligned}
\underset{\mathbf{h},\xi}{\text{minimize}} \quad & \xi \\
\text{subject to} \quad & \mathbf{h}^T \mathbf{Q}_i \mathbf{h} - 2\mathbf{p}_i^T \mathbf{h} + (1 - W_\mathrm{s})^2 M - \xi \leq 0 \quad \omega_i \in \Omega_\mathrm{p} \\
& \mathbf{h}^T \mathbf{Q}_i \mathbf{h} - \xi \leq 0 \quad \quad \quad \quad \quad \quad \quad \omega_i \in \Omega_\mathrm{s}.
\end{aligned}
$$

Furthermore, we include in this optimization problem the constraints for the prototype filter guaranteeing the arrival at PR MFBs.

The design examples corresponding to biorthogonal ℓ_∞-norm prototype filters are shown in Figure 5.10. The parameters are $M = 4$, $N = 8M - 1$, $D = -4M$, $\rho_\mathrm{s} = 0.8$, and $\rho_\mathrm{p} = 0.5$. The stopband weights are given in the caption together with E_pp and E_a that are computed with even-stacked cosine modulation.

5.4.4 Paraunitary Design with ℓ_∞-Norm Objective Function

The reconstruction delay of the paraunitary MFB is set by the filter order and we can use the zero-phase response instead of $|H(e^{j\omega})|$. In contrast to the above, the desired frequency response $D(\omega)$ is now real-valued

$$
D(\omega) = \begin{cases} \sqrt{M} & \omega \in \Omega_\mathrm{p} \\ 0 & \omega \in \Omega_\mathrm{s} \end{cases}
$$

95

so that

$$F(\mathbf{h}) = \max_{\omega \in \Omega_{\mathrm{p}} \cup \Omega_{\mathrm{s}}} W(\omega)|H(\omega) - D(\omega)|$$

$$= \max_{\omega \in \Omega_{\mathrm{p}} \cup \Omega_{\mathrm{s}}} \{\omega \in \Omega_{\mathrm{p}} : (1 - W_{\mathrm{s}})|H(\omega) - \sqrt{M}|\} \cup \{\omega \in \Omega_{\mathrm{s}} : W_{\mathrm{s}}|H(\omega)|\}.$$

Assuming that the prototype filter order N is odd and $\mathbf{h} = \begin{bmatrix} h[0] & h[1] & \cdots & h[(N-1)/2] \end{bmatrix}^T$ we can write the zero-phase response with vector notation:

$$H(\omega) = 2 \sum_{n=0}^{(N-1)/2} h[n]\cos\big((N/2 - n)\omega\big) = \mathbf{c}^T(\omega)\,\mathbf{h},$$

where the $(N+1)/2$-length vector is $\big[\mathbf{c}^T(\omega)\big]_n = 2\cos\big((N/2 - n)\omega\big)$. Minimization of $F(\mathbf{h})$ leads to the following linearly constrained optimization problem:

$$\begin{aligned}
\underset{\mathbf{h},\xi}{\text{minimize}} \quad & \xi \\
\text{subject to} \quad & (1 - W_{\mathrm{s}})\mathbf{c}^T(\omega_i)\mathbf{h} - (1 - W_{\mathrm{s}})\sqrt{M} \le \xi & \omega_i \in \Omega_{\mathrm{p}} \\
& -(1 - W_{\mathrm{s}})\mathbf{c}^T(\omega_i)\mathbf{h} + (1 - W_{\mathrm{s}})\sqrt{M} \le \xi & \omega_i \in \Omega_{\mathrm{p}} \\
& W_{\mathrm{s}}\mathbf{c}^T(\omega_i)\mathbf{h} \le \xi & \omega_i \in \Omega_{\mathrm{s}} \\
& -W_{\mathrm{s}}\mathbf{c}^T(\omega_i)\mathbf{h} \le \xi & \omega_i \in \Omega_{\mathrm{s}}.
\end{aligned}$$

This can be further simplified into

$$\begin{aligned}
\underset{\mathbf{h},\xi}{\text{minimize}} \quad & \xi \\
\text{subject to} \quad & \mathbf{A} \begin{bmatrix} \mathbf{h} \\ \xi \end{bmatrix} \le \mathbf{b}
\end{aligned}$$

which is more suitable for the optimization software.

Three ℓ_∞ paraunitary design examples are shown in Figure 5.11. The fixed parameters are $M = 4$, $N = 8M - 1$, $\rho_{\mathrm{s}} = 0.8$, and $W_{\mathrm{s}} = 0.6$. The passband edge is varied as given in the caption. The MFB self-imperfection measures $(E_{\mathrm{pp}}, E_{\mathrm{a}})$ are computed with odd-stacked cosine modulation.

5.5 Two-Prototype Designs for 2x Oversampled EMFB

5.5.1 Intuitive Viewpoint to Design Problem Reformulation

At this point it is relevant to ask does the 2x oversampled EMFB provide any relaxation to the prototype filter design? The answer is found by dropping the single prototype assumption. In order to study this possibility we consider the simplest case of EMFB biorthogonality: the analysis and synthesis subfilters are generated from two distinct prototypes $h[n]$ and $f[n]$, respectively. The prototype filters are symmetric and have the same order $N = LM - 1$ with even L. The setup differs from the paraunitary condition due to modified objective function

$$\min_{\mathbf{h},\mathbf{f}} \big(\mathbf{h}^T \mathbf{Q}_h \mathbf{h} + \mathbf{f}^T \mathbf{Q}_f \mathbf{f}\big)$$
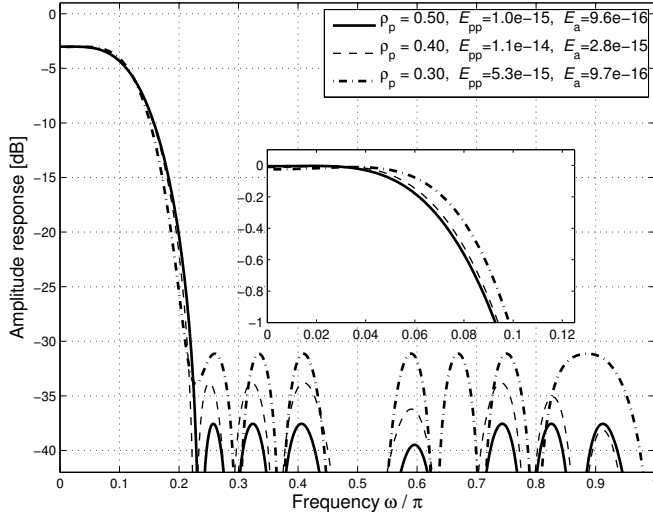
Figure 5.11: ℓ_∞-norm design examples with symmetric prototype filters.

when $\mathbf{Q}_h$ and $\mathbf{Q}_f$ utilize roll-off factors such that $\rho_h \neq \rho_f$.

The results of lapped transform theory [4] are here preferred over the polyphase domain analysis because the subfilter length is an even multiple of M. Further on, we switch to ELT-type of modulation as the arbitrary-delay parameter is left out for the time being. Then we use the subfilters of the EMFB as the basis functions of complex lapped transform matrices $\tilde{\mathbf{R}}$ (forward) and $\mathbf{R}$ (inverse). The connections given in equations (3.10) and (3.11) are applied in constructing these matrices:

$$\left[\tilde{\mathbf{R}}\right]_{n,k} = \left(h_k^e[LM - 1 - n]\right)^* = h[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$\left[\mathbf{R}\right]_{n,k} = f_k^e[n] = f[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],$$

such that the LT matrix dimensions are $LM \times 2M$. A flowgraph matching the above definition is shown in Figure 5.12. The transform matrices are divided into $M \times 2M$-sized blocks $\tilde{\mathbf{R}}_i$ and $\mathbf{R}_i$, where $\left[\tilde{\mathbf{R}}_i\right]_{n,k} = \left[\tilde{\mathbf{R}}\right]_{n+iM,k}$ and $\left[\mathbf{R}_i\right]_{n,k} = \left[\mathbf{R}\right]_{n+iM,k}$ with $i = 0, 1, \ldots, L - 1$. The first half of the shift-biorthogonality condition is then

$$\sum_{i=0}^{L-1-\ell} \mathbf{R}_i \tilde{\mathbf{R}}_{i+\ell}^H = \delta[\ell]\,\mathbf{I}, \tag{5.13}$$

when $\ell = 0, 1, \ldots, L - 1$ and the RHS comes from equation (3.8) with $D = 0$. The second half of the shift-biorthogonality is omitted since both prototype filters are symmetric and provide redundant PR constraints.

The exponential modulation part can be separated from the prototype filter like we did with the polyphase decomposition in Section 4.1.2:

$$\sum_{i=0}^{L-1-\ell} \mathbf{R}_i \tilde{\mathbf{R}}_{i+\ell}^H = \sum_{i=0}^{L-1-\ell} \mathbf{F}_i \mathbf{\Phi}_i^e \left(\mathbf{\Phi}_{i+\ell}^e\right)^H \mathbf{H}_{i+\ell} \tag{5.14}$$

97
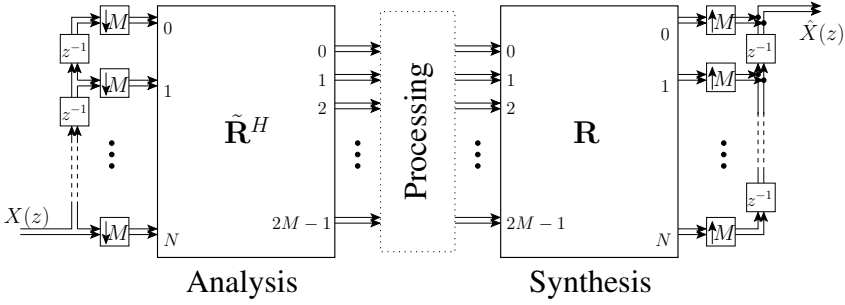
Figure 5.12: Complex lapped transform corresponding to the biorthogonal 2xOS EMFB.

where $\mathbf{F}_i = \text{diag}\{f[iM], f[iM+1], \ldots, f[iM+M-1]\}$ and $\mathbf{H}_i = \text{diag}\{h[iM], h[iM+1], \ldots, h[iM+M-1]\}$. (The latter one can be written here in natural order due to the symmetry of the prototype filters.) The exponential submatrices can be expressed with $M \times M$ (co)sine matrices that are defined as

$$\left[\mathbf{\Phi}_i^{\text{c}}\right]_{n,k} = 2\cos\left[\left(n+iM+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$\left[\mathbf{\Phi}_i^{\text{s}}\right]_{n,k} = 2\sin\left[\left(n+iM+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

and the substitution/simplification of the block matrix terms yields

$$\sum_{i=0}^{L-1-\ell} \mathbf{R}_i \tilde{\mathbf{R}}_{i+\ell}^H$$

$$= \frac{1}{4} \sum_{i=0}^{L-1-\ell} \mathbf{F}_i \left[\mathbf{\Phi}_i^{\text{c}} + j\,\mathbf{\Phi}_i^{\text{s}} \quad -(\mathbf{\Phi}_i^{\text{c}} - j\,\mathbf{\Phi}_i^{\text{s}})\mathbf{J}\right] \left[\begin{array}{c}\left(\mathbf{\Phi}_{i+\ell}^{\text{c}} - j\,\mathbf{\Phi}_{i+\ell}^{\text{s}}\right)^T \\ -\mathbf{J}\left(\mathbf{\Phi}_{i+\ell}^{\text{c}} + j\,\mathbf{\Phi}_{i+\ell}^{\text{s}}\right)^T\end{array}\right] \mathbf{H}_{i+\ell} \quad (5.15)$$

$$= \frac{1}{2} \sum_{i=0}^{L-1-\ell} \mathbf{F}_i \left(\mathbf{\Phi}_i^{\text{c}}\left(\mathbf{\Phi}_{i+\ell}^{\text{c}}\right)^T + \mathbf{\Phi}_i^{\text{s}}\left(\mathbf{\Phi}_{i+\ell}^{\text{s}}\right)^T\right)\mathbf{H}_{i+\ell}.$$

The imaginary part vanished from the shift-biorthogonality condition and we are left with the multiplication of real modulation matrices. These are then evaluated with even and odd shifts as with shift-orthogonality in [57]. But we need to be precise with sign reversions and thus it is conveniently handled with the roots of unity (cf. equation (4.10)). Let us begin with cosine modulation:

$$\left[\mathbf{\Phi}_i^{\text{c}}\left(\tilde{\mathbf{\Phi}}_{i+2s}^{\text{c}}\right)^T\right]_{n,m} = \sum_{k=0}^{M-1} \left[\mathbf{\Phi}_i^{\text{c}}\right]_{n,k}\left[\tilde{\mathbf{\Phi}}_{i+2s}^{\text{c}}\right]_{m,k}$$

$$= 4\sum_{k=0}^{M-1} \cos\left[\left(n+iM+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$\times \cos\left[\left(m+(i+2s)M+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right]$$

$$= S_{n-m-2s} + S_{n+m+2(i+s)M+M+1}.$$

98

The dimensions of the matrix product limit the possible values for λ such that

$$S_{n-m-2sM} = \begin{cases} (-1)^s 2M & \text{if } n = m \quad (\lambda = -s) \\ 0 & \text{otherwise} \end{cases}$$

and

$$S_{n+m+2(i+s)M+M+1} = \begin{cases} -(-1)^{i+s} 2M & \text{if } n = M - 1 - m \quad (\lambda = i + s + 1) \\ 0 & \text{otherwise.} \end{cases}$$

When the shift index is odd as in $S_{n-m-(2s+1)M}$, we can try substituting $\lambda = -s$, but then the non-zero condition is realized at $n = m + M$ - outside the $M \times M$ matrix boundaries. Alternatively, we can try $\lambda = -s - 1$ but again we miss the "target zone" with $n = m - M$. A similar reasoning for $S_{n+m+2(i+s+1)M+1}$ yields $n = -1 - m$ ($\lambda = i + s + 1$) and $n = 2M - 1 - m$ ($\lambda = i + s + 2$). Consequently, the odd shift of cosine block matrices contributes a zero term to the shift-biorthogonality in equation (5.15). The matrix notation of the above simplifies to

$$\Phi_i^c \left(\Phi_{i+2s}^c \right)^T = 2M \left((-1)^s \mathbf{I} - (-1)^{s+i} \mathbf{J} \right)$$
$$\Phi_i^c \left(\Phi_{i+2s+1}^c \right)^T = \mathbf{0}.$$

The matching product for the sine modulation matrices is handled by substituting the definition $2 \sum_{k=0}^{M-1} \sin(\frac{2\pi(k+1/2)\alpha}{2M}) = j \sum_{k=0}^{M-1} \left(W_{2M}^{-(k+1/2)\alpha} - W_{2M}^{(k+1/2)\alpha} \right)$. We can save ink with short-hand notations $n' = n + iM + (M+1)/2$ and $m' = m + (2i+\ell)M + (M+1)/2$. The n, mth element of the resulting matrix is thus

$$\left[\Phi_i^s \left(\tilde{\Phi}_{i+\ell}^s \right)^T \right]_{n,m} = j^2 \sum_{k=0}^{M-1} \left(W_{2M}^{-(k+1/2)n'} - W_{2M}^{(k+1/2)n'} \right) \left(W_{2M}^{-(k+1/2)m'} - W_{2M}^{(k+1/2)m'} \right)$$

$$= -\sum_{k=0}^{M-1} W_{2M}^{-(k+1/2)(n'+m')} + W_{2M}^{(k+1/2)(n'+m')} - W_{2M}^{-(k+1/2)(n'-m')} - W_{2M}^{(k+1/2)(n'-m')}$$

$$= S_{n'-m'} - S_{n'+m'} = S_{n-m-\ell M} - S_{n+m+(2i+\ell)M+M+1}.$$

Once the shift-index ℓ is divided into even and odd cases, we notice familiar terms from the above cosine matrix product. As can be seen from

$$\Phi_i^s \left(\Phi_{i+2s}^s \right)^T = 2M \left((-1)^s \mathbf{I} + (-1)^{s+i} \mathbf{J} \right)$$
$$\Phi_i^s \left(\Phi_{i+2s+1}^s \right)^T = \mathbf{0},$$

the only difference is the sign reversal of the crossdiagonal term.

On these grounds the PR constraints from the shift-biorthogonality condition in equation (5.13) reduces to even matrix shifts only. In addition, matrix $\mathbf{J}$, the nasty ol' bugger, disappears from the simplified form of equation (5.15):

$$\sum_{i=0}^{L-1-2s} \mathbf{R}_i \tilde{\mathbf{R}}_{i+2s}^H = (-1)^s 2M \sum_{i=0}^{L-1-2s} \mathbf{F}_i \mathbf{H}_{i+2s}.$$

The above is equated with the RHS of shift-biorthogonality condition (5.13) to get

$$\sum_{i=0}^{L-1-2s} \mathbf{F}_i \mathbf{H}_{i+2s} = \frac{1}{2M}\delta[s]\mathbf{I}, \tag{5.16}$$

where $s = 0, 1, \ldots, L/2 - 1$. In terms of impulse response coefficients the same prototype filter PR condition for 2x oversampled EMFB is

$$\sum_{i=0}^{L-1-2s} f[n + iM]h[n + (i + 2s)M] = \frac{1}{2M}\delta[s], \tag{5.17}$$

where $n = 0, 1, \ldots, M - 1$. The other half of the equations are redundant when $s = 0$, but we don't peel open the subject any further.

This example is sufficient to make our point here: 2x oversampled EMFB relaxes the PR constraints for biorthogonal two-prototype designs because the cross-diagonal terms (i.e., matrix $(-1)^{s+i}\mathbf{J}$ in our notation) is cancelled out from the product of exponentially modulated submatrices $\mathbf{\Phi}_i^{\mathrm{e}}\big(\mathbf{\Phi}_{i+\ell}^{\mathrm{e}}\big)^H$. It is stressed that such two-prototype designs do not satisfy PR with critical subsampling.

5.5.2 Advanced Design Types for Biorthogonal 2xOS EMFB

We obtained above an intuitive view how the antidiagonal terms in the constraint equations for the two-prototype EMFB vanish with 2x oversampling. This holds as well for arbitrary-length, arbitrary-delay designs and the concept can be generalized further. The analysis and synthesis subfilters can have different orders N_h and N_f, respectively. The adjustable delay parameter is also split in two which leads us to definitions

$$h_k^{\mathrm{e}}[n] = h[n] \exp\left[j\left(n - \frac{N_h + D_h + M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \tag{5.18}$$

$$f_k^{\mathrm{e}}[n] = f[n] \exp\left[j\left(n - \frac{N_f + D_f - M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \tag{5.19}$$

for the 2xOS odd-stacked EMFB subfilters. The analysis-synthesis reconstruction delay for such a system is $\tau = (N_h + N_f + D_h + D_f)/2$ which must be an integer number.

The parameter split provides new symmetry combinations for MFB design; we have four cases depending on the chosen delay parameter values:

- Both the analysis and synthesis prototype filters are symmetric;
  $D_h = 0$ and $D_f = 0$.

- Synthesis prototype is non-symmetric and analysis prototype is symmetric;
  $D_h = 0$ and $D_f \neq 0$.

- Analysis prototype is non-symmetric and synthesis prototype is symmetric;
  $D_h \neq 0$ and $D_f = 0$.

- Both the analysis and synthesis prototype filters are non-symmetric;
  $D_h \neq 0$ and $D_f \neq 0$.

The single-prototype PR constraints in Table 5.1 can be tinkered into the two-prototype 2xOS EMFB mode that utilizes parameters N_h, N_f, D_f, and D_h. It is not addressed here, but the implementation is included in [2] for curious souls. To back up our claims for two-prototype PR designs, we conclude this section with examples.

Our task for the optimization routine is to solve two problems with the following parameters:

(a) Symmetric prototypes; $M = 6$, $N_h = 33$, $N_f = 43$, $\rho_{h,s} = 1.0$, and $\rho_{f,s} = 1.2$.
    Reconstruction delay: $\tau = 38$.

(b) Non-symmetric prototypes; $M = 6$, $N_h = 54$, $N_f = 27$, $\rho_{h,s} = 0.8$, $\rho_{f,s} = 1.2$, $D_h = -11$, and $D_f = -6$.
    Reconstruction delay: $\tau = 32$.

In addition, we slightly tuned the objective function when compared with the previous design examples. Minimization criteria use the usual ℓ_2-norm but now with different roll-off values $\rho_{h,s}, \rho_{f,s}$ for the analysis and synthesis filters. The other modification is the inclusion of the passband ripple: We choose $W_{h,s} = W_{f,s} = 0.9$ and passband edge parameters as $\rho_{h,p}, \rho_{f,p} = 0.5$ because the slight passband shaping also stabilized the applied optimization routine.

Once the optimization routine produced a feasible solution, the reconstruction error and aliasing level was measured with the equations in Section 5.2.2 to verify the PR quality. The resulting prototype filter impulse responses are shown in Figure 5.13. When the subfilters of 2xOS odd-stacked EMFBs are generated from these two cases, we get the amplitude responses shown in Figure 5.14. Two subfilter curves per analysis and synthesis sides are drawn in the same subfigure. The distortion values for the oversampled EMFB are given below the amplitude response curves; both 2xOS EMFBs are practically PR subband systems.

It is stressed here that although we defined the two-prototype filter design for 2x oversampled EMFBs with odd-stacking, the PR is satisfied with all 2xOS EMFB-types discussed above. In the list below the numbers refer to the corresponding FB flowgraph.

- 2xOS odd-stacked EMFB for complex-valued signals (Figure 4.2)

- 2xOS odd-stacked EMFB for real-valued signals (Figure 4.4)

- 2xOS even-stacked EMFB for complex-valued signals (Figure 4.6 (a))

- 2xOS even-stacked EMFB for real-valued signals (Figure 4.7 (a))

Application of these prototype pairs with critically sampled MFBs leads to the distortion and loss of the PR property. This also raises questions: "Can we re-formulate the two-prototype CS optimization problem by utilizing our previous work and does the solver provide results with satisfactory stopband attenuation?" We leave that unanswered but some design methodology might work because in the simplest case there exists a biorthogonal window function pair with the desired property [58].

To summarize the 2xOS EMFB design: These biorthogonal MFB designs just illustrated the versatility of the EMFB concept. In order to get a good understanding how different parameters influence on the minimization results would require a systematic testing arrangement. This work is left for the future.
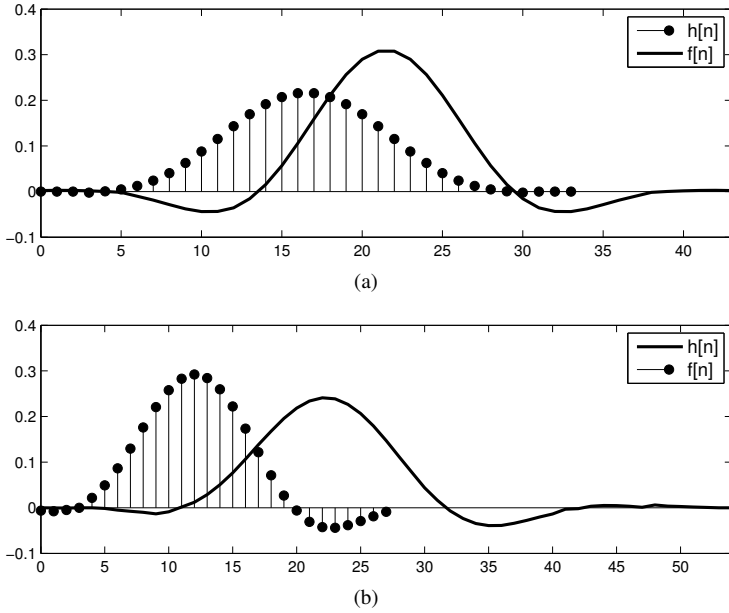
Figure 5.13: Different length two-prototype design examples: (a) Symmetric impulse responses. (b) Non-symmetric prototypes for the analysis and synthesis 2xOS EMFBs.
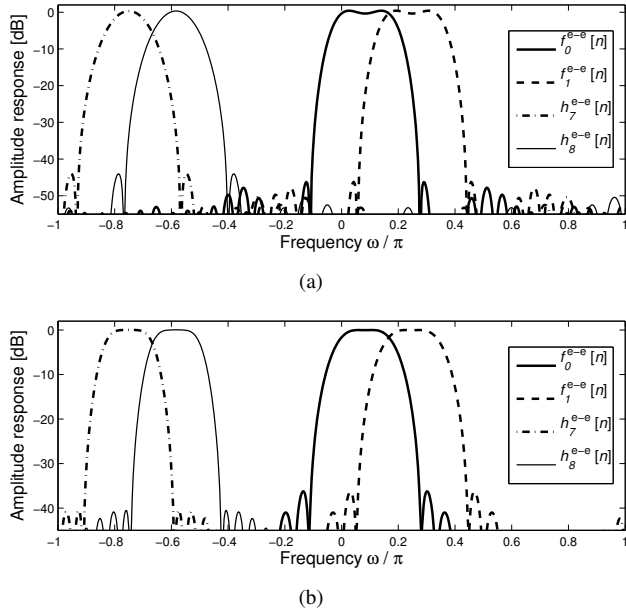
Figure 5.14: Amplitude responses of two-prototype design examples: (a) Symmetric with distortion $E_{\mathrm{pp}} = 1.0 \times 10^{-14}$, $E_{\mathrm{a}} = 9.6 \times 10^{-16}$. (b) Non-symmetric with distortion $E_{\mathrm{pp}} = 7.8 \times 10^{-15}$, $E_{\mathrm{a}} = 6.2 \times 10^{-16}$.

Chapter 6

Family of Parametrized Lapped Transforms

The alternative to generating PR modulated filter banks is to use parametrized window functions instead of optimized prototype filters. Choosing the parametrized prototypes is an appealing alternative in applications where highly frequency selective subfilters do not considerably improve the system performance. Why hit a nail with a sledgehammer? Secondly, we must find a balance between the contradicting requirements of time-frequency resolution; longer prototype filters increase the uncertainty in time-dimension.

The window functions for this purpose were originally introduced with extended lapped transform type of modulation and we review these subjects below. The motivation to discuss ELT modulation also stems from Chapter 8 where we introduce a version of the fast ELT structure with even-stacked modulation. In the algorithm derivation that utilizes butterfly rotations it is more convenient to use the re-defined subfilter equations.

By applying these prototype windows, one can generate LT matrices with odd-stacked cosine modulation and it is guaranteed that the shift-(bi)orthogonality is satisfied. In other words, all LTs discussed below are invertible transforms or, equivalently, M-channel perfect-reconstruction FBs. We begin gathering the prototype window equations found in [57, 58] and add one missing detail: adjustable extended lapped transform angles for odd M. Then the LT-concept is refined by applying the MFB theory from Chapter 4. This leads us to a family of LTs where we can pick a combination of transform properties that suits best for the application needs from three options:

- Critically sampled or 2x oversampled subsignals. In the latter case the subsignals are complex-valued.

- Real-valued or complex-valued input signal.

- Odd-stacked or even-stacked subchannelization. The even-stacked modulation produces (anti-)symmetric basis functions.

All parametrized window functions are symmetric and, consequently, the even-stacked transforms have basis functions with linear phase. We suggest a naming convention for these trans-

forms to clarify the somewhat confusing complex-valued LT terminology. A distinction is also made between adjustable and non-adjustable parametric window functions.

The second part of this chapter has more original content. There we develop a recursive algorithm for complex-valued modulated lapped transform (MLT) similar to the sliding DCT. In these subband systems, the transform-domain samples are computed every time a new input sample is available and also the subband signal is M (or $2M$) times oversampled. The sliding lapped transform (SLT) algorithms for MLT and ELT window functions can be realized as a bank of recursive filters. We derive the corresponding filter coefficients for both odd and even-stacked cases.

6.1 Window Functions for Generating Lapped Transforms

6.1.1 MLT and ELT without DC-Leakage

For a start we define the non-adjustable windows for two values of K

$$
h_K[n] = \begin{cases} \dfrac{1}{\sqrt{2M}} \sin\left[\left(n+\dfrac{1}{2}\right)\dfrac{\pi}{2M}\right] & K = 1 \\ \dfrac{1}{\sqrt{2M}}\left(-\dfrac{1}{2\sqrt{2}} + \dfrac{1}{2}\cos\left[\left(n+\dfrac{1}{2}\right)\dfrac{\pi}{2M}\right]\right) & K = 2, \end{cases} \tag{6.1}
$$

where $n = 0, 1, \ldots, 2KM - 1$. The shorter one is the MLT and the other the ELT window function. These equations were obtained from [57]. At the time of publishing the research was focused of odd-stacked modulation, i.e., the synthesis subfilters of a filter bank corresponding to the inverse MLT are produced with equation

$$
f_k^{\mathrm{mlt}}[n] = 2\,h_1[n]\cos\left[\left(n+\dfrac{M+1}{2}\right)\left(k+\dfrac{1}{2}\right)\dfrac{\pi}{M}\right]. \tag{6.2}
$$

The paraunitary relation of M-channel FBs provides the connection between the analysis and synthesis subfilters: $h_k^{\mathrm{mlt}}[n] = f_k^{\mathrm{mlt}}[2KM - 1 - n]$. It is well-known that MLT has zero DC-leakage, formally,

$$
\left|\sum_{n=0}^{2M-1} h_k^{\mathrm{mlt}}[n]\right| = \delta[k]\sqrt{M}.
$$

However, it is less widespread information that the window function $h_2[n]$ has also zero DC-leakage property, *with both even/odd-stacked modulation*. If we then define an even-stacked ELT where the synthesis subfilters are

$$
f_k^{\mathrm{elt-e}}[n] = c_k\,h_2[n]\cos\left[\left(n+\dfrac{M+1}{2}\right)\dfrac{k\,\pi}{M}\right]
$$

$$
\breve{f}_k^{\mathrm{elt-e}}[n] = -c_k\,h_2[n]\sin\left[\left(n+\dfrac{M+1}{2}\right)\dfrac{k\,\pi}{M}\right],
$$

where

$$
c_k = \begin{cases} \sqrt{2} & k = 0 \text{ or } k = M \\ 2 & \text{otherwise.} \end{cases}
$$

The parity of M resolves the number of subfilters in the first and second set. When M is odd(even) the even-stacked ELT subband configuration matches with Type-I(Type-II) CMFB

104

as given in Section 4.6. Similar to the odd-stacked configuration above, we can find equations that link the analysis and synthesis subfilters:

$$h_k^{\text{elt}-\text{e}}[n] = (-1)^k f_k^{\text{elt}-\text{e}}[n]$$
$$\breve{h}_k^{\text{elt}-\text{e}}[n] = (-1)^{k+1} \breve{f}_k^{\text{elt}-\text{e}}[n].$$

It is surprisingly troublesome to show the zero DC-leakage condition for the ELT. Here we resort to the infamous "proof-by-MATLAB"-methodology. Based on numerical evaluation the following holds when $h_2[n]$ is applied as the ELT-generating window function:

$$\left| \sum_{n=0}^{4M-1} h_k^{\text{elt}}[n] \right| = \delta[k]\sqrt{2M}, \qquad k = 0, 1, \ldots, M-1$$

$$\left| \sum_{n=0}^{4M-1} h_k^{\text{elt}-\text{e}}[n] \right| = \delta[k]\sqrt{2M}, \qquad k = 0, 1, \ldots$$

$$\left| \sum_{n=0}^{4M-1} \breve{h}_k^{\text{elt}-\text{e}}[n] \right| = 0, \qquad k = 1, 2, \ldots$$

where the range of k with even-stacked modulation depends on the parity of M.

To summarize the DC-leakage discussion: LTs generated from the non-adjustable window functions given in (6.1) have zero DC-leakage when

- <u>odd-stacked</u> cosine modulation is used with the MLT window function

- <u>odd/even-stacked</u> cosine modulation is used with the ELT window function.

Due to the simplicity of parametrized windows in (6.1) it is possible to develop a recursive algorithm to implement both odd and even-stacked transforms. These sliding transform algorithms can be used to implement efficiently M times oversampled subband systems. In the context of wavelet transforms these kind of systems are called time-invariant transforms but here the term is somewhat misleading. We will return to this topic in Section 6.2.

6.1.2 Adjustable ELT Angles

The fast ELT is an implementation algorithm for odd-stacked cosine-modulated filter bank. The prototype filter part is realized as a cascade of lattices, also called butterflies, where the term originates from the flowgraphs of FFT-algorithms [22]. The coefficients of the butterflies are cosine and sine terms of ELT angles $\theta_{i,n}$. The equations in [57] were given in adjustable angles. In the equations reviewed below, a variable γ is typically chosen within $[0, 1]$ and it controls the tradeoff between stopband attenuation and transition bandwidth.

The ELT angles <u>for even M</u> are given by

$$\theta_{0,n} = -\frac{\pi}{2} + \left(\frac{(1-\gamma)(n+(M+1)/2)}{M} + \gamma \right) \frac{(n+(M+1)/2)\pi}{4M} \qquad (6.3)$$

$$\theta_{1,M/2-1-n} = -\frac{\pi}{2} + \left(\frac{(1-\gamma)(n+1/2)}{M} + \gamma \right) \frac{(n+1/2)\pi}{4M}. \qquad (6.4)$$

They are connected with the prototype filter impulse response coefficients via

$$h_\gamma[n] = \cos\theta_{1,n}\cos\theta_{0,n} \tag{6.5}$$
$$h_\gamma[M-1-n] = \cos\theta_{1,n}\sin\theta_{0,n} \tag{6.6}$$
$$h_\gamma[n+M] = \sin\theta_{1,n}\cos\theta_{0,n} \tag{6.7}$$
$$h_\gamma[2M-1-n] = -\sin\theta_{1,n}\sin\theta_{0,n}, \tag{6.8}$$

though not yet properly scaled. These equations are valid for even M when we let $n = 0, 1, \ldots, M/2 - 1$.

The necessary modification <u>for odd M case</u> is just marginal; the second angle equation (6.4) is replaced with

$$\theta_{1,\lfloor M/2\rfloor - 1 - n} = -\frac{\pi}{2} + \left(\frac{(1-\gamma)(n+1)}{M} + \gamma\right)\frac{(n+1)\pi}{4M} \tag{6.9}$$

and equations (6.5-6.8) for computing $h_\gamma[n]$ can then be applied with $n = 0, 1, \ldots, \lfloor M/2\rfloor - 1$. The missing "gaps" have fixed values in order to satisfy the PR condition:

$$h_\gamma[M + \lfloor M/2\rfloor] = -1/\sqrt{2} \tag{6.10}$$
$$h_\gamma[\lfloor M/2\rfloor] = 0. \tag{6.11}$$

These fixed values also appear in ELT-structure-based prototype filter design for odd M [48, 20]. Or they can be obtained from the basic ELT equation. Namely, this adjustable ELT window matches with $K = 2$ case in equation (6.1) with parameter choice $\gamma = 1$. Smaller values for γ give better stopband attenuation at the expense of wider transition band.

The prototype window is symmetric $h_\gamma[n] = h_\gamma[4M - 1 - n]$ but we have to scale $h_\gamma[n]$ to meet the PR condition:

$$h[n] = h_\gamma[n]/\sqrt{2M}.$$

Then we may apply also the adjustable ELT window function with the CMFB and EMFB subfilter equations.

6.1.3 Modulated Lapped Biorthogonal Transform

The modulated lapped biorthogonal transform is not the first encountered case in this thesis where the forward and inverse LTs (or analysis and synthesis FBs) utilize different prototypes. We have already discussed this biorthogonality in the context of 2x oversampled arbitrary-delay EMFBs. However, now both window functions are symmetric, the transform has fixed length with reconstruction delay $\tau = 2M - 1$, and the PR condition holds also for the critically sampled MFBs. As expected, we may apply the exponential-modulation with the modulated lapped biorthogonal transform (MLBT) prototype windows to generate a biorthogonal critically sampled transform for subband processing of complex-valued signals. Or we can build a linear-phase biorthogonal transform with even-stacked cosine modulation.

The distinct prototype windows, forward $h[n]$ and inverse $f[n]$, of the MLBT give some degrees of freedom in choosing the transform properties when compared with the MLT. For example, we can shape the tails of the synthesis filter impulse response or have better frequency selectivity on the analysis side. However, tinkering of one prototype must be compensated with the other in order to have the PR conditions satisfied.

The equations for synthesis prototypes [58] come with two variables α and β:

$$f_{\alpha,\beta}[n] = \frac{1 - \cos\left(\left((n+1)/M\right)^\alpha \pi\right) + \beta}{2 + \beta}, \qquad (6.12)$$

where $n = 0, 1, \ldots, 2M - 1$. A good starting point for these parameters is $\alpha = 0.95$, $\beta = 0.2$ if one wants to improve the selectivity of, say, the analysis stage (the window functions are interchangeable for the PR's sake). The analysis window is defined as a function of $f_{\alpha,\beta}[n]$:

$$h_{\alpha,\beta}[n] = \frac{f_{\alpha,\beta}[n]}{f_{\alpha,\beta}^2[n] + f_{\alpha,\beta}^2[n+M]}, \qquad (6.13)$$

where we let $n = 0, 1, \ldots, M - 1$. The missing half is obtained via symmetry $h_{\alpha,\beta}[n] = h_{\alpha,\beta}[2M - 1 - n]$. The final prototypes are scaled in such a way that they are on the same level and PR holds with our definition for the modulating sequences:

$$h[n] = h_{\alpha,\beta}[n]/\sqrt{2M} \qquad f[n] = f_{\alpha,\beta}[n]/\sqrt{2M}.$$

Of course, switching the forward and inverse transform window functions with each other can be done. Finally, it is stressed that the MLBT prototypes work with all MFB types discussed in Chapter 4 using delay parameter value $D = 0$. The biorthogonal filter length is $N = 2M - 1$ and all resulting subband systems are perfectly reconstructing.

6.1.4 Basic Tools for Signal Decomposition into Uniform Subbands

It is now quite straightforward to draw the results from the biorthogonal EMFB theory and combine that with the concepts of modulated LTs. Because the LT window functions are symmetric, they can be used to generate paraunitary EMFBs. Equally well the ELT-phase definition applies with the exponential modulation:

$$f_k^{\text{celt}}[n] = h[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],$$

which imposes the restriction that the length of $h[n]$ is $2KM$, $K \in \mathbb{N}$. The argument of the exponential function produces odd-stacked subchannelization and we can apply the time-reversion relation in (4.36): $h_k^{\text{celt}}[n] = \left(f_k^{\text{celt}}[2KM - 1 - n]\right)^*$.

In analogy, the even-stacked complex-valued subfilters can be written as

$$f_k^{\text{celt}-\text{e}}[n] = h[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\frac{k\pi}{M}\right],$$

which obey the linear phase property; the real and imaginary parts of the subfilter impulse responses are either symmetric or anti-symmetric. As with the ELT-E, the connection $h_k^{\text{celt}-\text{e}}[n] = (-1)^k f_k^{\text{celt}-\text{e}}[n]$ provides the analysis subfilter equations but now $k \in [0, 2M - 1]$. Furthermore, if the subfilter model is fixed as in Figure 4.6, the other set of subfilters is generated with $\breve{h}_k^{\text{celt}-\text{e}}[n] = -j\, h_k^{\text{celt}-\text{e}}[n]$ and $\breve{f}_k^{\text{celt}-\text{e}}[n] = j\, f_k^{\text{celt}-\text{e}}[n]$.

To get ourselves organized with this family of modulated lapped transforms it is now orderly to give systematic names for the MLT, ELT and MLBT-based transforms. We suggest the following naming convention:

1. The window function name is used as the basic identifier of the transform.

Table 6.1: Summary of extended lapped transform variants.

LT type	Even/ Odd- Stacking	AFB Input Signal	Real/ Complex Subsignals	# of Subbands (M decimation factor)
ELT	Odd	Real	Real	M
ELT-E	Even	Real	Real	$M+1$
CELT	Odd	Complex	Real	$2M$
CELT-E	Even	Complex	Real	$2M$
CELT-C	Odd	Complex	Complex	$2M$
CELT-CE	Even	Complex	Complex	$2M$
ELT-C	Odd	Real	Complex	M
ELT-CE	Even	Real	Complex	$M+1$

2. Prefix C indicates the transform suitability for decomposing of complex-valued signals.

3. Suffix E selects the even-stacked subchannelization mode.

4. The optional operation mode when the subsignals are complex-valued is denoted with suffix C. In this case the LT-system is oversampled by a factor of two.

Note that these prefix/suffix-identifiers maintain the original LT definition. For example, MLT is real-valued and critically sampled transform in odd-stacked configuration.

By finding all combinations of the attached letters we get altogether 8 different transform types that are summarized in Table 6.1 for the ELT. The same kind of versioning can be done for the modulated lapped (biorthogonal) transform as well just by replacing the "ELT" acronym with "ML(B)T". As an example, we translate CMLT-CE as "2x oversampled even-stacked MLT for subband decomposition of complex-valued signals". We conclude the naming convention matter with illustrative examples of the LT frequency characteristics.

The first interesting case is DC-leakage-free transform defined with prototype window equation (6.1). The odd-stacked MLT with overlap $M = 6$ is shown in Figure 6.1 (a). The second zero DC-leakage alternative, i.e., ELT-E $M = 6$ has two sets of basis functions and for even M the high-pass filter belongs to the second set. The amplitude responses of these sub-filters are shown in Figure 6.1 (b). It can be seen that the highest sidelobe level of the MLT is below the basic ELT case, but one should also consider the transition bandwidth in comparing the selectivity of these parametrized transforms with small number of subbands.

The example with adjustable ELT angles is given for odd-stacked subband system that devours complex-valued input signals at leisure. Figure 6.2 (a) shows the amplitude responses of CELT for $M = 5$ and $\gamma = 0.6$. The first sidelobe goes down approximately 10 dBs when compared with the ELT in Figure 6.1 (b). However, now there is more mainlobe overlap with the adjacent filters.

The final LT amplitude response example is shown in Figure 6.2 (b). It is a biorthogonal transform with linear phase complex-valued basis functions, i.e., CMLBT-E. Thicker lines are the amplitude responses of the forward transform and the inverse transform basis functions are
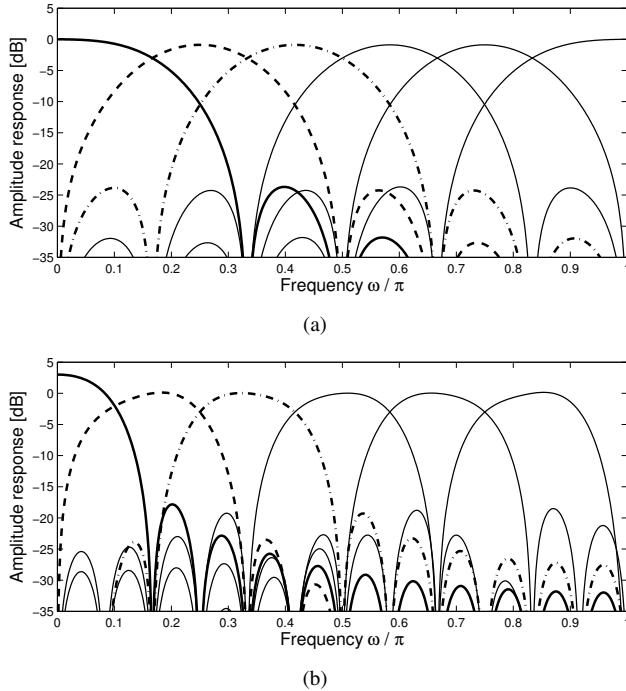
Figure 6.1: (a) MLT without DC-leakage, odd-stacked, $N = 11$, and $M = 6$. (b) First set of ELT-E basis functions without DC-leakage, even-stacked, $N = 23$, and $M = 6$. The high-pass filter belongs to the second set of subfilters.

shown with thin linetype. The negative frequency side basis functions are left out for clarity. Parameters for the biorthogonal window function are $\alpha = 0.85$ and $\beta = 0.2$. We see that this choice gives better frequency selectivity for the analysis filters when measured with sidelobe level.

6.2 Recursive Algorithms for Modulated Lapped Transforms

6.2.1 Oversampled Lapped Transforms by Factor $M/2M$

After refining the modulated LT concept to cover 2x oversampled subband processing for complex-valued signals, there is a temptation to oversample the 2x oversampled subband system. This somewhat sinister urge originates from the sliding transforms developed for Type-II DCT and DFT. Based on those recursive algorithms we might ask: "Can we find a similar algorithm for modulated LTs?" At least out of curiosity.

To better justify our cause, we might even resort to drawing support from wavelet signal processing applications. Namely, in [27, 49] time-invariant wavelet transforms is utilized in noise suppression problems. Translating the *time-invariance* into the language of FBs, it means that the decimation by two operations in tree-structured filter bank are left out after the high/low-pass splitting (See Figure 2.10 (c)). This turns the analysis system into a plain bank
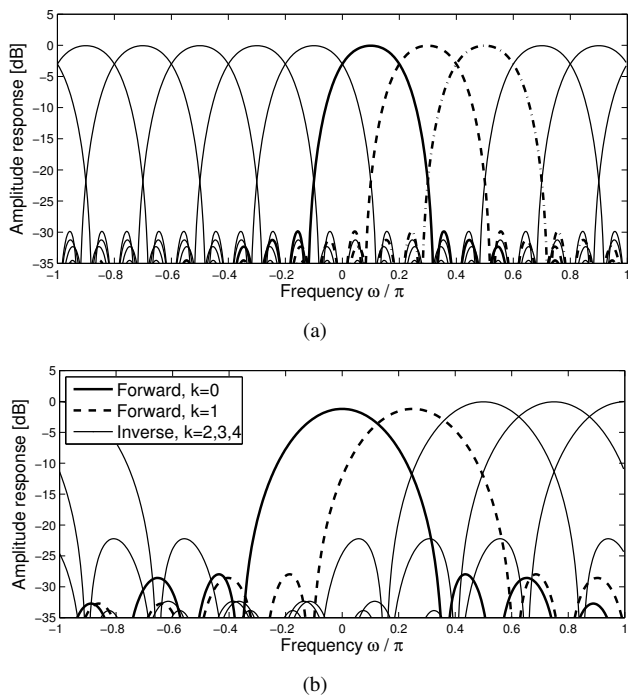
Figure 6.2: (a) Adjustable CELT, odd-stacked, $N = 19$, $M = 5$, and $\gamma = 0.6$. (b) Part of the CMLBT-E basis. Even-stacked, $N = 7$, $M = 4$, $\alpha = 0.85$, and $\beta = 0.2$.

of filters without downsampling operations at all. The redundancy of these oversampled FBs increases by the number of subbands. Signal enhancement is typically done with a subband threshold device that aims to preserve the significant subsignal components and cancel out noise. The signal synthesis is a reversal ("leaf-to-root"-procedure) of the analysis operations, as usual.

If our interest is to replace the wavelet transforming with a shift-orthogonal transform $\mathbf{P}$, we get an M *times oversampled* subband processing system shown in Figure 6.3 (a). By Mx oversampling we mean that transform-domain samples are now computed for every new input sample - not by shifts of M as in the critically sampled LT systems. Deliberately, we avoid speaking of time-invariant LTs because in the even-stacked configuration these are time varying subband systems even if we remove the decimation-interpolation operations. When the matrix $\mathbf{P}$ is a modulated transform, it can be implemented with block-transform-based fast algorithms, although the solution is not the most sophisticated. Hence there is a motivation to develop a more suitable algorithm for this purpose. Our Savior is the running-sum type of solution which can be implemented as a recursive filter (not IIR though!) with few coefficients per subchannel. This is possible due to the periodicity of the window functions in equation (6.1). The idea for the derivation of this recursive FB scheme is adopted from [50].

Before proceeding into the SLT filter structure one detail should be mentioned concerning the transform-domain adaptive filtering problems. As discussed in [34], an orthogonal transform can be plugged between the input delay line and adaptive filter weights to decorrelate the signal for faster convergence. In the basic setting, an $M \times M$ block transform is utilized

(a)

(b)
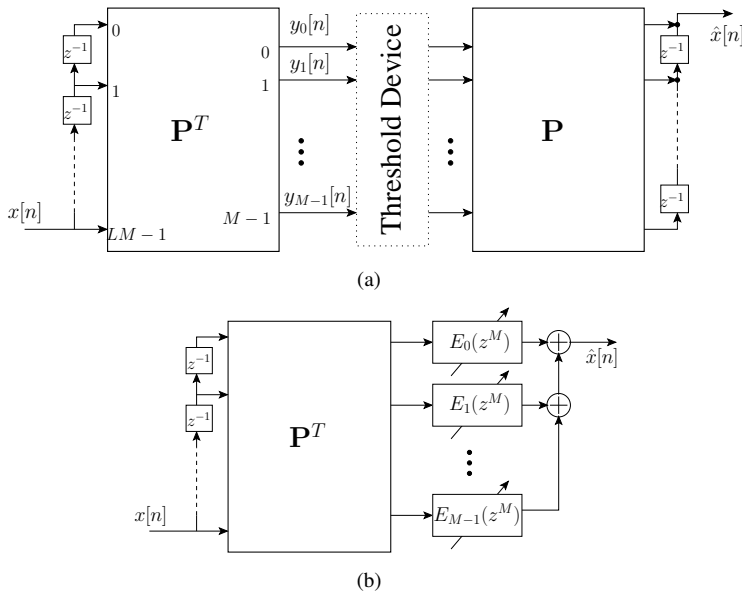
Figure 6.3: (a) M times oversampled LT system for noise suppression. (b) Transform-domain adaptive filtering system utilizing LT for signal decorrelation.

and there is one adaptable coefficient per subband. However, as shown in Figure 6.3 (b), if we have a lapped transform in this mode, then the transform overlap property comes into play. The forward transform is followed by a bank of adaptable filters $E_k(z^M)$ and the number of coefficients depend on the parameter K:

$$E_k(z^M) = \sum_{m=0}^{2K-1} e_k[m] z^{-mM}.$$

These M-spaced coefficients of the adaptive subfilters are a consequence of the lapped transform matrix shift-orthogonality. The practical value of LTs in this adaptive filtering setup is diminished by the increased number of adaptable coefficients per subband. What is gained in better decorrelation property over conventional block transforms is lost in slower adaptation due to "extra" coefficients. Then in our humble opinion, the system in Figure 6.3 (b) is rendered to be "of academic interest only". However, the SLTs may be applicable in subband adaptive filtering setup described in [71].

6.2.2 Sliding Lapped Transforms as Non-Decimated Filter Banks

In order to develop the SLT algorithms we again switch the viewpoint between LTs and FBs. The lapped transform system in Figure 6.3 (a) has a dual filter bank realization and the question is: "Can we exploit the periodicity of the MLT window and modulation sequence to find a recursive implementation for $M/2M$ times oversampled CMLTs?" Truly we may accomplish this and possibly the easiest way to do so is scheming with the subfilter equations. Note: The deduction chain is done only with the non-adjustable window $h_1[n]$ in (6.1) and the SLT filter

Figure 6.4: Parallel subfilter model for Mx/$2M$x oversampled SLTs: (a) CSLT or CSLT-C. (b) SLT or SLT-C. (c) CSLT-E or CSLT-CE. (d) SLT-E or SLT-CE.

Subband 0 (R)(R)(R)(R)(I)(I)(I)(I)
(C)(C)(C)(C)(C)(C)(C)(C)
••• (C)(C)(C)(C)(C)(C)(C)(C) •••
(C)(C)(C)(C)(C)(C)(C)(C)
Subband M (I)(I)(I)(I)(R)(R)(R)(R)
(a)

(R)(R)(R)(I)(I)(I)
(C)(C)(C)(C)(C)(C)
••• (C)(C)(C)(C)(C)(C) •••
(C)(C)(C)(C)(C)(C)
(R)(R)(R)(I)(I)(I)
(b)

Subband 0 (R)(R)(R)(R)(○)(○)(○)(○)
(R)(R)(R)(R)(R)(R)(R)(R)
••• (R)(R)(R)(R)(R)(R)(R)(R) •••
(R)(R)(R)(R)(R)(R)(R)(R)
Subband M (○)(○)(○)(○)(R)(R)(R)(R)
(c)

(R)(R)(R)(○)(○)(○)
(R)(R)(R)(R)(R)(R)
••• (R)(R)(R)(R)(R)(R) •••
(R)(R)(R)(R)(R)(R)
(R)(R)(R)(○)(○)(○)
(d)
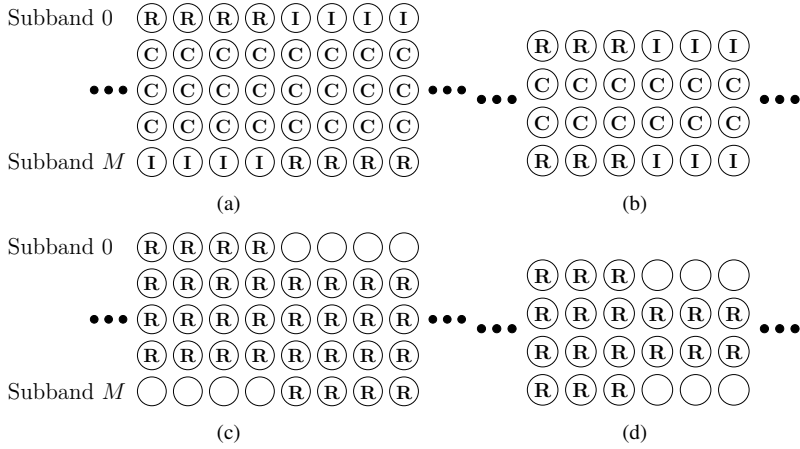
Figure 6.5: Data ordering has a period of $2M$ in subbands for $M/2M$x oversampled even-stacked transforms. (a) SLT-CE, $M = 4$. (b) SLT-CE, $M = 3$. (c) SLT-E, $M = 4$. (d) SLT-E, $M = 3$.
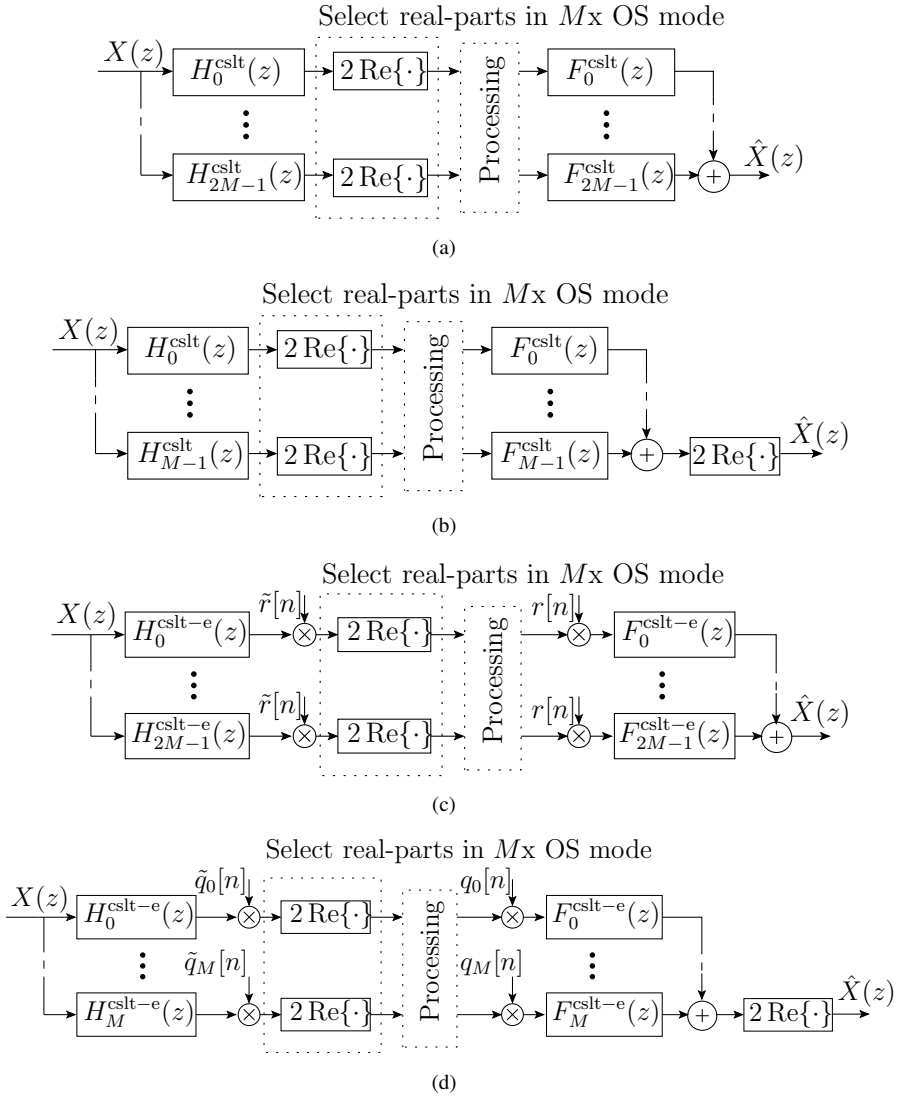
coefficients of the ELT are provided out of the blue. But in the continuation SLT generally refers to recursive algorithms generated from either of these two window functions.

Figure 6.4 shows the SLT-based M or $2M$ times oversampled subband systems implemented with a bank of parallel and independent subfilters. The prototype filter is here restricted to be $h_K[n]$ from equation (6.1) and the synthesis filter impulse responses are then

$$f_k^{\text{cslt}}[n] = \frac{1}{\sqrt{M}} h_K[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \qquad (6.14)$$

and

$$f_k^{\text{cslt}-\text{e}}[n] = \frac{1}{\sqrt{M}} h_K[n] \exp\left[j\left(n + \frac{M+1}{2}\right)\frac{k\pi}{M}\right], \qquad (6.15)$$

for odd and even-stacked modulation, respectively. The scaling term $1/\sqrt{M}$ is necessary due to leaving out the downsampling operations. The analysis filters can be written similarly by applying the subfilter equations from Section 6.1.1. It is reminded that the even-stacked LTs are time varying systems as there are PR-condition-induced phase rotation multipliers between the fixed subfilters. The multiplier sequence $r[n]$ is repeating in periods of $2M$ (i.e. M "1"s and M "j"s alternate) and thus should not be considered as multiplications in computational complexity evaluations:

$$r[n] = j^{(\lfloor n/M \rfloor \bmod 2)}.$$

In the case of SLT-E algorithm, the low/high-pass terms need to be downscaled

$$q_k[n] = \begin{cases} \dfrac{1}{\sqrt{2}} j^{(\lfloor n/M \rfloor \bmod 2)} & k = 0 \text{ or } M \\ j^{(\lfloor n/M \rfloor \bmod 2)} & k = 1, 2, \ldots, M-1. \end{cases}$$

The analysis side multipliers are complex conjugates of the above: $\tilde{r}[n] = r^*[n]$ and $\tilde{q}_k[n] = q_k^*[n]$. These $\pi/2$-rotations originate from the EMFB subsignal handling in Figure 7.5. Thus we end up with the four SLT-based subfilter configurations shown in Figure 6.4:

(a) CSLT/CSLT-C: Odd-stacked, $2M$ subbands, and $x[n]$ complex-valued

(b) SLT/SLT-C: Odd-stacked, M subbands, and $x[n]$ real-valued

(c) CSLT-E/CSLT-CE: Even-stacked, $2M$ subbands, and $x[n]$ complex-valued

(d) SLT-E/SLT-CE: Even-stacked, $M + 1$ subbands, and $x[n]$ real-valued

The choice between M and $2M$ oversampling is done by inclusion/exclusion of the real-part operation after the analysis filters in all cases (a-d). The subband organization in case (d) requires few extra words: Depending on the parity of M, the purely real and imaginary-valued samples are ordered differently in subbands. In Figure 6.5, we see examples of $2M$-length SLT-E/SLT-CE subsignal blocks for $M = 4$ and $M = 3$. The letters inside circle indicates whether the coefficient is real, imaginary, or complex-valued ($\mathbf{R,I,C}$). An empty circle is zero (placeholder) subsample. Subfigures (a) and (b) schematically display $2M$ times oversampled, whereas (c) and (d) M times oversampled data of the even-stacked ELT where the time index increases from left to right. As can be seen, the sequences (c) and (d) are the real parts of the ones above as expected on the basis of Figure 6.4 (d). This kind of data organization in subbands repeats in periods of $2M$. With these considerations in mind we develop next the running-sum algorithms for the analysis and synthesis subfilters.

6.3 Derivation of Sliding Transform Algorithm Using MLT Window

The computation of the forward and inverse transforms of the SLT-based systems in Figure 6.4 can be done with the block transform-based algorithms. However, that solution would not exploit the fact that transform-domain samples are computed every time a new input sample is available which is now under consideration. Below is provided the derivation of forward CSLT-C algorithm utilizing the $h_1[n]$ window function. The following developments can be applied as a template to verify also the inverse and odd-stacked versions with minor modifications. The SLT algorithms corresponding to the somewhat more burdensome $h_2[n]$ window function is then just manifested in Table 6.3.

A more convenient form for the kth subsignal of forward SLT-C is

$$y_k[n] = \sum_{i=0}^{2M-1} x[n-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i], \tag{6.16}$$

where the short-hand notations stand for

$$\phi^{\mathrm{s}}[i] = \sin\left[\left(i + \frac{1}{2}\right)\frac{\pi}{2M}\right] \tag{6.17}$$

$$\phi_k^{\mathrm{e}}[i] = -\frac{1}{M\sqrt{2}}\exp\left[j\left(i - \frac{M-1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right]. \tag{6.18}$$

Two handy definitions are also needed later on: $\theta_k = (k + 1/2)\frac{\pi}{M}$ and $\phi^{\mathrm{c}}[i] = \cos\left((i + 1/2)\frac{\pi}{2M}\right)$. In addition, the $4M$-periodicity of $\phi_k^{\mathrm{e}}[i]$ and $\phi^{\mathrm{s}}[i]$ provides us, say, $\phi_k^{\mathrm{e}}[i \pm 2M] = -\phi_k^{\mathrm{e}}[i]$.

Consecutive transform domain samples $y_k[n]$ and $y_k[n-1]$ use the same $(2M-2)$-length portion of the sequence $x[n]$ in the computation of the sum (6.16). The key question is now:

What is the relation between the past and present subband samples and can we apply it to develop a sliding algorithm for our purpose? The answer is found by using the running DCT [50] derivation as a guideline. Then we write first $y_k[n]$ in such a form where the previously known subsignal output appears:

$$
\begin{aligned}
y_k[n] &= \sum_{i=0}^{2M-1} x[n-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&= x[n]\,\phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0] + \Big[\sum_{i=1}^{2M} x[n-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i]\Big] - x[n-2M]\,\underbrace{\phi^{\mathrm{s}}[2M]}_{-\phi^{\mathrm{s}}[0]}\,\underbrace{\phi_k^{\mathrm{e}}[2M]}_{-\phi_k^{\mathrm{e}}[0]} \\
&= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n]-x[n-2M]\big) + \sum_{i=0}^{2M-1} x[n-(i+1)]\,\phi^{\mathrm{s}}[i+1]\,\phi_k^{\mathrm{e}}[i+1] \\
&= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n]-x[n-2M]\big) + e^{j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{s}}[i+1]\,\phi_k^{\mathrm{e}}[i].
\end{aligned}
$$

The range of summation is shifted by one in the last step above and $\phi_k^{\mathrm{e}}[i+1]$ is replaced by $e^{j\theta_k}\phi_k^{\mathrm{e}}[i]$. Now we are approaching the desired expression for $y_k[n-1]$, but first the sine-function $\phi^{\mathrm{s}}[i\pm1]$ should be expanded:

$$
\begin{aligned}
\phi^{\mathrm{s}}[i\pm1] &= \sin\Big[\Big(i\pm1+\frac{1}{2}\Big)\frac{\pi}{2M}\Big] \\
&= \cos\Big(\frac{\pi}{2M}\Big)\sin\Big[\Big(i+\frac{1}{2}\Big)\frac{\pi}{2M}\Big] \pm \sin\Big(\frac{\pi}{2M}\Big)\cos\Big[\Big(i+\frac{1}{2}\Big)\frac{\pi}{2M}\Big] \qquad (6.19) \\
&= \cos\Big(\frac{\pi}{2M}\Big)\phi^{\mathrm{s}}[i] \pm \sin\Big(\frac{\pi}{2M}\Big)\phi^{\mathrm{c}}[i].
\end{aligned}
$$

When (6.19) with plus is substituted inside the sum in the previous expression, we have

$$
\begin{aligned}
y_k[n] &= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n]-x[n-2M]\big) \\
&\quad + \cos\Big(\frac{\pi}{2M}\Big)e^{j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&\quad + \sin\Big(\frac{\pi}{2M}\Big)e^{j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{c}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n]-x[n-2M]\big) + \cos\big(\tfrac{\pi}{2M}\big)e^{j\theta_k}y_k[n-1] \\
&\quad + \sin\big(\tfrac{\pi}{2M}\big)e^{j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{c}}[i]\,\phi_k^{\mathrm{e}}[i].
\end{aligned}
$$

Now there is another summation where the term $\phi^{\mathrm{c}}[i]$ appears. Fortunately we can banish it by

repeating a similar operation for $y_k[n-2]$:

$$
\begin{aligned}
y_k[n-2] &= \sum_{i=0}^{2M-1} x[n-2-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&= x[n-2-(2M-1)]\,\underbrace{\phi^{\mathrm{s}}[2M-1]}_{-\phi^{\mathrm{s}}[-1]}\,\underbrace{\phi_k^{\mathrm{e}}[2M-1]}_{-\phi_k^{\mathrm{e}}[-1]} \\
&\quad + \Big[\sum_{i=-1}^{2M-2} x[n-2-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i]\Big] - x[n-2-(-1)]\,\phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1] \\
&= \phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1]-x[n-1]\big) + e^{-j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{s}}[i-1]\,\phi_k^{\mathrm{e}}[i].
\end{aligned}
$$

Then we expand $\phi^{\mathrm{s}}[i-1]$ as given above in (6.19)

$$
\begin{aligned}
y_k[n-2] &= \phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1]-x[n-1]\big) \\
&\quad + \cos\Big(\frac{\pi}{2M}\Big)e^{-j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{s}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&\quad - \sin\Big(\frac{\pi}{2M}\Big)e^{-j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{c}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&= \phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1]-x[n-1]\big) + \cos\Big(\frac{\pi}{2M}\Big)e^{-j\theta_k}\,y_k[n-1] \\
&\quad - \sin\Big(\frac{\pi}{2M}\Big)e^{-j\theta_k}\sum_{i=0}^{2M-1} x[n-1-i]\,\phi^{\mathrm{c}}[i]\,\phi_k^{\mathrm{e}}[i].
\end{aligned}
$$

Rearrangement of terms and multiplying both sides with $e^{j\theta_k}$ yields

$$
\begin{aligned}
\sin\Big(\frac{\pi}{2M}\Big)\sum_{i=0}^{2M-1} & x[n-1-i]\,\phi^{\mathrm{c}}[i]\,\phi_k^{\mathrm{e}}[i] \\
&= e^{j\theta_k}\phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1]-x[n-1]\big) \\
&\quad + \cos\Big(\frac{\pi}{2M}\Big)y_k[n-1] - e^{j\theta_k}y_k[n-2].
\end{aligned}
$$

The LHS of the above equation was the "term non-grata" in the recursive expression for $y_k[n]$. We substitute the last equation and simplify the coefficients, noting that the sign reversion

comes from $\phi^{\mathrm{s}}[-1] = -\phi^{\mathrm{s}}[0]$:

$$
\begin{aligned}
y_k[n] &= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n] - x[n-2M]\big) + \cos\left(\frac{\pi}{2M}\right)e^{j\theta_k}y_k[n-1] \\
&\quad + e^{j\theta_k}\Big[e^{j\theta_k}\phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1] - x[n-1]\big) \\
&\qquad\qquad + \cos\left(\frac{\pi}{2M}\right)y_k[n-1] - e^{j\theta_k}y_k[n-2]\Big] \\
&= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n] - x[n-2M]\big) \\
&\quad + e^{j2\,\theta_k}\phi^{\mathrm{s}}[-1]\,\phi_k^{\mathrm{e}}[-1]\big(x[n-2M-1] - x[n-1]\big) \\
&\quad + 2\,\cos\left(\frac{\pi}{2M}\right)e^{j\theta_k}y_k[n-1] - e^{j2\,\theta_k}y_k[n-2] \\
&= \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[0]\big(x[n] - x[n-2M]\big) \\
&\quad + \phi^{\mathrm{s}}[0]\,\phi_k^{\mathrm{e}}[1]\big(x[n-1] - x[n-2M-1]\big) \\
&\quad + 2\,\cos\left(\frac{\pi}{2M}\right)e^{j\theta_k}y_k[n-1] - e^{j2\,\theta_k}y_k[n-2].
\end{aligned}
$$

And we are done!

The difference equation above is an expression for $y_k[n]$ in terms of past subband samples. Then we may switch to z-transform domain notation for recursive filters. The complex-valued filter coefficients are

$$
\begin{aligned}
\tilde{b}_k[0] &= -\tfrac{1}{M\sqrt{2}}\sin\left(\tfrac{\pi}{4M}\right)\exp(-j(M-1)(k+\tfrac{1}{2})\tfrac{\pi}{2M}) \\
\tilde{b}_k[1] &= \tilde{b}_k[0]\,\exp(j\,(k+\tfrac{1}{2})\tfrac{\pi}{M}) \\
\tilde{a}_k[1] &= -2\cos\left(\tfrac{\pi}{2M}\right)\exp(j\,(k+\tfrac{1}{2})\tfrac{\pi}{M}) \\
\tilde{a}_k[2] &= \exp(j\,(k+\tfrac{1}{2})\tfrac{2\pi}{M}).
\end{aligned}
$$

The transfer function for the kth $h_1[n]$-based non-decimated CSLT subfilter is thus

$$
H_k^{\mathrm{cslt}}(z) = \frac{(1-z^{-2M})(\tilde{b}_k[0] + \tilde{b}_k[1]z^{-1})}{1 + \tilde{a}_k[1]\,z^{-1} + \tilde{a}_k[2]\,z^{-2}} = \frac{(1-z^{-2M})\tilde{B}_k(z)}{\tilde{A}_k(z)}.
$$

The equations for the other subfilters of CSLTs can be verified similarly and the results are gathered in Table 6.2. The odd and even-stacked coefficients can be given compactly by reversing few signs in the definitions as explained in the caption. The superscript identifier of the subfilters is thus left out and the coefficients for the forward transform filters are marked with tilde.

We wash our hands over this matter with Table 6.3 which shows the transfer functions of the $h_2[n]$-based SLTs. The equations for both the odd and even-stacked modulation are given, even though the realization of such a system might require further consideration. The order of actual filtering operations may be different from the z-transform domain polynomial form given in Table 6.3. This can be influenced by the known fixed point implementation problems with IIR filters having poles close to the unit circle. But the functionality of the derived recursive algorithms have been tested with floating point arithmetic. In the PR testing using the analysis-synthesis configuration, there was reconstruction error level higher than the normal numerical imprecision with block-by-block algorithms without feedback loop. It might be wiser to analyze the reconstruction noise of SLTs accompanied with the comparisons

Table 6.2: Non-decimated MLTs realized as recursive filters. The upper and lower signs correspond to odd and even-stacked modulation, respectively.

Analysis and Synthesis Filters	
Analysis:	Synthesis:
$H_k(z) = \pm \dfrac{(1 \mp z^{-2M})\tilde{B}_k(z)}{\tilde{A}_k(z)}$	$F_k(z) = \dfrac{(1 \mp z^{-2M})B_k(z)}{A_k(z)}$
Coefficients	
$\theta_k = \begin{cases} \left(k + \frac{1}{2}\right)\frac{\pi}{M} & \text{odd-stacked} \\ \frac{k\,\pi}{M} & \text{even-stacked} \end{cases}$	$C = -1/(M\sqrt{2})$
$\tilde{a}_k[1] = a_k[1] = -2\cos\left(\frac{\pi}{2M}\right)\exp(j\,\theta_k)$	$\tilde{a}_k[2] = a_k[2] = \exp(j\,2\theta_k)$
$\tilde{b}_k[0] = C\,\sin(\frac{\pi}{4M})\exp(-j(M-1)\theta_k/2)$	$\tilde{b}_k[1] = \tilde{b}_k[0]\exp(j\,\theta_k)$
$b_k[0] = -C\,\sin(\frac{\pi}{4M})\exp(j(M+1)\theta_k/2)$	$b_k[1] = b_k[0]\exp(j\,\theta_k)$

against the residual error in sliding DCT before making any further conclusions. However, in relatively short simulations with periodic resets, the accumulated inherent round-off error is negligible compared with the signal level.

Table 6.3: Non-decimated ELTs realized as recursive filters.

Analysis and Synthesis Filters

Analysis:
$$H_k(z) = \frac{\left((1 - z^{-4M})\tilde{D}_k(z)\right)/\tilde{C}_k(z) + (1 - z^{-4M})\tilde{B}_k(z)}{\tilde{A}_k(z)}$$

Synthesis:
$$F_k(z) = \frac{\left((1 - z^{-4M})D_k(z)\right)/C_k(z) + (1 - z^{-4M})B_k(z)}{A_k(z)}$$

Coefficients

$$\theta_k = \begin{cases} \left(k + \frac{1}{2}\right)\frac{\pi}{M} & \text{odd-stacked} \\ \frac{k\,\pi}{M} & \text{even-stacked} \end{cases}$$

$\tilde{a}_k[1] = a_k[1] = -2\cos\left(\frac{\pi}{2M}\right)\exp(j\,\theta_k)$ $\qquad$ $\tilde{a}_k[2] = a_k[2] = \exp(j\,2\theta_k)$

$\tilde{b}_k[0] = \frac{1}{2M\sqrt{2}}\left(\cos\left(\frac{\pi}{4M}\right) - \frac{1}{\sqrt{2}}\right)\exp(-j(M-1)\theta_k/2)$ $\qquad$ $\tilde{b}_k[1] = -\tilde{b}_k[0]\exp(j\,\theta_k)$

$b_k[0] = \frac{1}{2M\sqrt{2}}\left(\cos\left(\frac{\pi}{4M}\right) - \frac{1}{\sqrt{2}}\right)\exp(j(M+1)\theta_k/2)$ $\qquad$ $b_k[1] = -b_k[0]\exp(j\,\theta_k)$

$\tilde{c}_k[1] = c_k[1] = -\exp(j\,\theta_k)$ $\qquad$ $\tilde{d}_k[0] = d_k[0] = 0$

$\tilde{d}_k[1] = \frac{1}{2M}\left(\cos\left(\frac{\pi}{2M}\right) - 1\right)\exp(-j(M-3)\theta_k/2)$

$d_k[1] = \frac{1}{2M}\left(\cos\left(\frac{\pi}{2M}\right) - 1\right)\exp(j(M+3)\theta_k/2)$

7

Chapter

Fast Algorithms for Biorthogonal Modulated Filter Banks

The first step needed to find a block-transform-based algorithms for MFBs is already taken. Namely, the derivation of PR condition for odd-stacked CMFB in Section 4.1 contains a suitable equation, where the modulation block is extracted from the polyphase factored prototype filter. All that is left is to identify the basis functions of the DCT/DST/DFT from the modulation block and position the corresponding fast algorithm into the MFB implementation structure. Even though this seems a straightforward method when expressed verbally, the complications hide in various modulation options and parameter combinations. The important paraunitary case receives later a special treatment in Chapter 8, where the MFB implementation with fast ELT algorithm is further elaborated to cover even-stacked modulation.

A road map for the various MFB in this section: We begin in Section 7.1 with the polyphase factoring of the biorthogonal MFB prototype filter combined with odd/even-stacked modulation. Then Sections 7.2, 7.3, and 7.4 provide the block-transform-based algorithms for odd-stacked 2xOS EMFB, CMFB, and CS EMFB, respectively. The difference between the EMFB versions are that in the CS case DCTs/DSTs can be utilized, whereas 2xOS case is handled with the FFT algorithm. A similar distinction explains the 2xOS and CS even-stacked EMFB versions provided in Section 7.5 and 7.7, respectively. The biorthogonal even-stacked CMFB algorithm is tackled in Section 7.6 for both possible subband configurations discussed in Section 4.6. Moreover, since the 2xOS EMFB algorithms have a burden of complex-valued phase tweaking multipliers for subsignals, it is desirable to embed these exponential coefficients in the FFT itself. In Section 7.8 we introduce such modifications for radix-2 FFT and, when accompanied with circular shifting, the 2xOS EMFBs can be computed with fewer arithmetic operations. In addition, we discuss the per-subband processing EMFB algorithm for such applications where only a restricted set of subsignals are required. In Section 7.9, the recursive Goertzel-algorithm is merged with "windowing" using the biorthogonal prototype. The computational complexity of the relevant block-transform-based algorithms is evaluated in Section 7.10. Lastly, the final twist is to compute the CS EMFB's modulation part with two consecutive subsample vectors using a single FFT realization (Section 7.11).

7.1 Polyphase Domain Factoring for Prototype Filter

All block-transform-based biorthogonal MFBs discussed here share a common element: the prototype filter is implemented as a polyphase block with $2M$-branches operating at $1/M$th rate of the input signal rate. A more intuitive description for this computation is to say that the prototype filter (on the analysis side) windows $(N + 1)$-length portion of the signal, folds the data, and then directs the result to the modulation block. Synthesis side prototype filtering is a reverse operation including an overlap-add procedure at shifts of M. However, the prototype filter part is not exactly the same as in basic down/upsampling filters in Figure 2.8. Let us return to decomposition in (4.7), but generalize the formula to include also complex-valued subfilters:

$$\boldsymbol{F}_{\mathrm{p}}(z)\,\boldsymbol{H}_{\mathrm{p}}(z) = \begin{bmatrix} \mathbf{I} & z^{-1}\mathbf{I} \end{bmatrix} \boldsymbol{G}(z)\boldsymbol{\Phi}\tilde{\boldsymbol{\Phi}}^{H}\tilde{\boldsymbol{G}}(z) \begin{bmatrix} z^{-1}\mathbf{I} \\ \mathbf{I} \end{bmatrix}. \tag{7.1}$$

In the above, the modulation sequences in $\tilde{\boldsymbol{\Phi}}$, $\boldsymbol{\Phi}$ and their dimensions are fixed separately for each MFB type later. Here we turn our attention to $2M \times 2M$ diagonal matrices $\tilde{\boldsymbol{G}}(z)$ and $\boldsymbol{G}(z)$, where M is the decimation factor. Similarly as with the CMFB PR proof in Section 4.1, the $2M$-phase prototype filter components are first gathered as follows:

$$\tilde{G}_{\ell}(z) = \sum_{m} h[2M - 1 - \ell + 2mM]z^{-m}$$

$$G_{\ell}(z) = \sum_{m} h[\ell + 2mM]z^{-m}.$$

In Section 4.1.2 the $4M$-periodicity of the odd-stacked modulation was handled by embedding the sign reversion in the prototype filter coefficients so that the decomposition could utilize a half-length modulation generating sequence. In turn, the even-stacked modulation is $2M$-periodic and the prototype filter coefficients need no such tweaking. Therefore our generalized MFB decomposition selects either one of the two cases in definitions

$$\begin{bmatrix} \tilde{\boldsymbol{G}}(z) \end{bmatrix}_{\ell,\ell} = \begin{cases} \tilde{G}_{\ell}(-z^2) & \text{odd-stacked} \\ \tilde{G}_{\ell}(z^2) & \text{even-stacked} \end{cases} \tag{7.2}$$

and

$$\begin{bmatrix} \boldsymbol{G}(z) \end{bmatrix}_{\ell,\ell} = \begin{cases} G_{\ell}(-z^2) & \text{odd-stacked} \\ G_{\ell}(z^2) & \text{even-stacked} \end{cases} \tag{7.3}$$

depending on the MFB "stackedness".

When drawing the algorithm flowgraphs, the natural ordering of the polyphase components is favored, although this choice extends the input/output delay lines to have $2M - 1$ delay elements. As an example, the prototype filter part for even-stacked case is shown in Figure 7.1. If desired, we can always return to the basic M-channel FB polyphase matrix configuration by merging the kth and $(k + M)$th branch as in Figure 4.1. Then we can proceed to hunt for block transform matrices contained in the shortened ($2M$-length) modulation sequences.
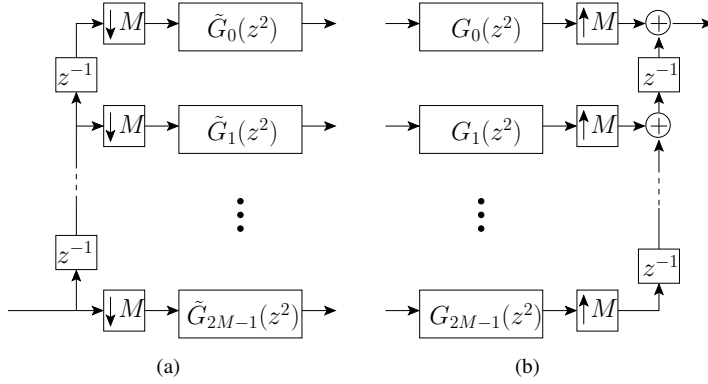
Figure 7.1: $2M$-phase even-stacked decomposition for the prototype filter where M is the decimation factor: (a) Analysis. (b) Synthesis.

7.2 FFT-Based Implementation for Odd-Stacked Biorthogonal EMFB

The subfilters of the odd-stacked EMFB are generated with exponential sequences defined in (4.27) and (4.28). A $2M$-length portion is selected for the columns of the synthesis side modulation matrix

$$\left[\mathbf{\Phi}_{\mathrm{e}}\right]_{n,k} = \exp\left[j\left(n - \frac{N+D-M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right],$$

where $n, k = 0, 1, \ldots, 2M - 1$. The parameters of the argument can be re-arranged so that

$$\left[\mathbf{\Phi}_{\mathrm{e}}\right]_{n,k} = \exp\left[j\left(nk + \frac{n}{2} - \left(k + \frac{1}{2}\right)\frac{N+D-M}{2}\right)\frac{\pi}{M}\right] = v_n W_{2M}^{-nk} w_k.$$

In the above the complex multipliers are denoted with $v_n = \exp\left[j\frac{\pi n}{2M}\right]$ and $w_k = \exp\left[-j\left(k + \frac{1}{2}\right)\frac{(N+D-M)\pi}{2M}\right]$. The remaining term W_{2M}^{-nk} is covered with the basis functions of the inverse DFT matrix $\left[\mathbf{\Psi}_{2M}\right]_{n,k} = \frac{1}{\sqrt{2M}}W_{2M}^{-nk}$. It is assumed here that the pair of forward-inverse DFT matrices are normalized - which is not usually true for the FFT algorithm. (When necessary, the radix-2 FFT can be normalized by customizing the algorithm, e.g. with bit-shifting. Or we can also multiply the scaling term into the w_k-coefficients if the computations are handled with floating point numbers.) The odd-stacked EMFB modulation matrix then factors into two diagonal matrices and the DFT:

$$\mathbf{\Phi}_{\mathrm{e}} = \sqrt{2M}\mathrm{diag}\{v_0, v_1, \ldots, v_{2M-1}\}\,\mathbf{\Psi}_{2M}\,\mathrm{diag}\{w_0, w_1, \ldots, w_{2M-1}\} = \sqrt{2M}\mathbf{V}\mathbf{\Psi}_{2M}\mathbf{W}.$$

The analysis side $\tilde{\mathbf{\Phi}}_{\mathrm{e}}$ is handled similarly. A rearrangement of the argument yields

$$\begin{aligned}
\left[\tilde{\mathbf{\Phi}}_{\mathrm{e}}\right]_{n,k} &= \exp\left[-j\left(2M - 1 - n - \frac{N+D+M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \\
&= \exp\left[j\left(nk + \frac{n}{2} + \left(k + \frac{1}{2}\right)\frac{N+D-3M+2}{2}\right)\frac{\pi}{M}\right] = v_n W_{2M}^{-nk}\tilde{w}_k
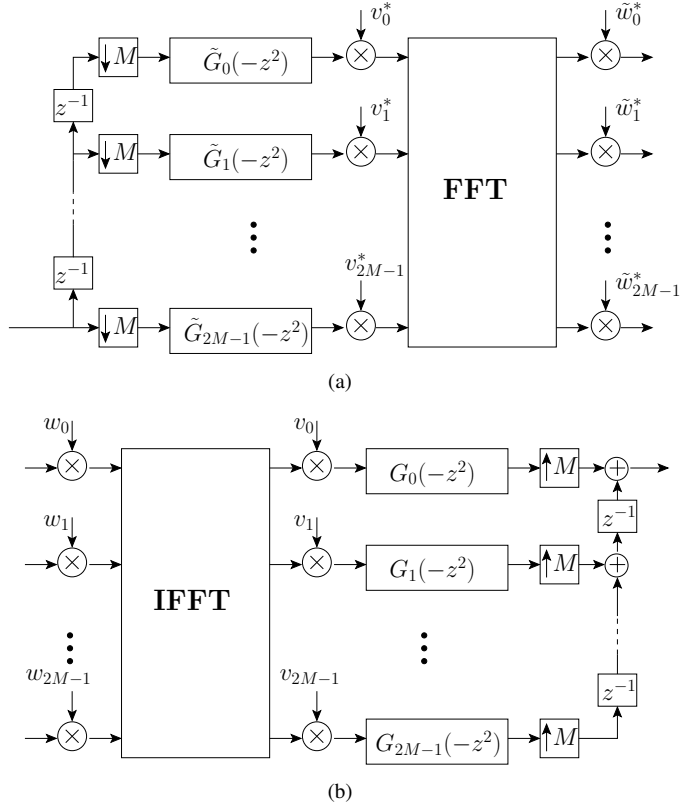\end{aligned}$$

(a)

(b)

Figure 7.2: FFT-based implementation for biorthogonal odd-stacked EMFB. (a) AFB. (b) SFB.

where $\tilde{w}_k = \exp\left[j\left(k + \frac{1}{2}\right)\frac{(N+D-3M+2)\pi}{2M}\right]$. Once we take the hermitian transpose we get

$$\tilde{\boldsymbol{\Phi}}_{\mathrm{e}}^H = \sqrt{2M}\,\mathrm{diag}\{\tilde{w}_0^*, \tilde{w}_1^*, \ldots, \tilde{w}_{2M-1}^*\}\, \boldsymbol{\Psi}_{2M}^H\, \mathrm{diag}\{v_0^*, v_1^*, \ldots, v_{2M-1}^*\}$$
$$= \sqrt{2M}\,\tilde{\mathbf{W}}^*\boldsymbol{\Psi}_{2M}^H\mathbf{V}^*.$$

When the above matrix factoring is combined with the polyphase prototype filter (odd-stacked case) we arrive to the algorithm shown in Figure 7.2. This implementation structure utilizes in a natural manner 2xOS subsignals which makes it suitable for two-prototype designs discussed in Section 5.5. The subband system is again critically sampled if we just select the real parts of the AFB subsamples and upscale them by a factor of 2 to satisfy PR.

7.3 DCT-Based Implementation for Odd-Stacked CMFB

The odd-stacked CMFB was a milestone in the discrete FB theory and enjoyed wide interest back in the days. Consequently, there are now several optional algorithms available. There are several lattice-based implementations for paraunitary designs [56, 48, 68]. However, the nearly-perfect-reconstruction or arbitrary-delay designs can not be mapped into rotation angles and the prototype filter coefficients are directly implemented as a set of polyphase subfilters

[42, 101]. A fast block transform handling the burden of modulation is a common feature in all of these CMFB variants. The designer/programmer should make the algorithm selection based on the accelerated features of the implementation platform. In the following, the provided arbitrary-delay CMFB algorithm is also used as a building block in the odd-stacked EMFB.

The modulation sequence for odd-stacked CMFB is real-valued and thus the phase rotation with complex multipliers is not an option. Instead, a strategy that works here is to identify the basis functions of Type-III/IV DCT, extend the basis into $2M$-length sequences and apply circular shifting to match the modulation phase. Depending on the parity of $N + D - M$ we consider two cases:

(a) $N + D - M$ odd

If we look at the SFB modulation matrix $\mathbf{\Phi}_c$ in (4.4), the time index n is accompanied with a fractional term $(N + D - M)/2$. The trick is to add and subtract $1/2$ so that

$$\left[\mathbf{\Phi}_c\right]_{n,k} = 2\cos\left[\left(n + \frac{1}{2} - \underbrace{\frac{N + D - M + 1}{2}}_{\text{integer}}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right].$$

We notice a resemblance with $M \times M$ DCT-IV definition; yet there is still an offset of $(N + D - M + 1)/2$ samples and the need to double the length of the basis. The latter is addressed by denoting $\left[\mathbf{C}_{\text{IV}}\right]_{n,k} = \sqrt{2/M}\cos\left[(n + 1/2)(k + 1/2)\frac{\pi}{M}\right]$ and multiply it with a mapping matrix $\mathbf{J}_c = \begin{bmatrix} \mathbf{I}_M \\ -\mathbf{J}_M \end{bmatrix}$ which generates a continuous sinusoidal sequence. Odd-stacked modulation is $4M$-periodic, but we want to limit the circular-shift index d within $[0, 2M - 1]$. This way the parameter d is always smaller than the number of the prototype filter polyphase components. Therefore, the offset is partitioned into multiple of $2M$ term and the remaining fraction:

$$d = (N + D - M + 1)/2 - 2rM,$$

where $r = \left\lfloor \frac{N+D-M+1}{2} \frac{1}{2M} \right\rfloor$. In addition, the $4M$-periodicity necessitates the reversion of signs with an array of multipliers

$$s_\ell = \begin{cases} -(-1)^r & \ell < d \\ (-1)^r & \ell \geq d. \end{cases}$$

These ± 1 terms can be multiplied with the polyphase components of the prototype filter.

For these MFB flowgraphs we introduce a circular-shift function

$$\text{CircularShift}(\mathbf{A}, d) = \left(\mathbf{S}_{-d} + \mathbf{S}_{2M-d}\right)\mathbf{A}.$$

with the aid of the shift-matrix definition for $\mathbf{S}_c$ in (3.5). Because d can have only positive values, it rotates the columns of $\mathbf{A}$ downwards. Then the synthesis side modulation matrix can be rewritten as

$$\mathbf{\Phi}_c = \sqrt{2M}\,\text{diag}\{s_0, s_1, \ldots, s_{2M-1}\}\,\text{CircularShift}(\mathbf{J}_c\mathbf{C}_{\text{IV}}, d).$$

This suffices for the synthesis CMFB algorithm and the flowgraph is shown in Figure 7.3 (b).

(a)

(b)
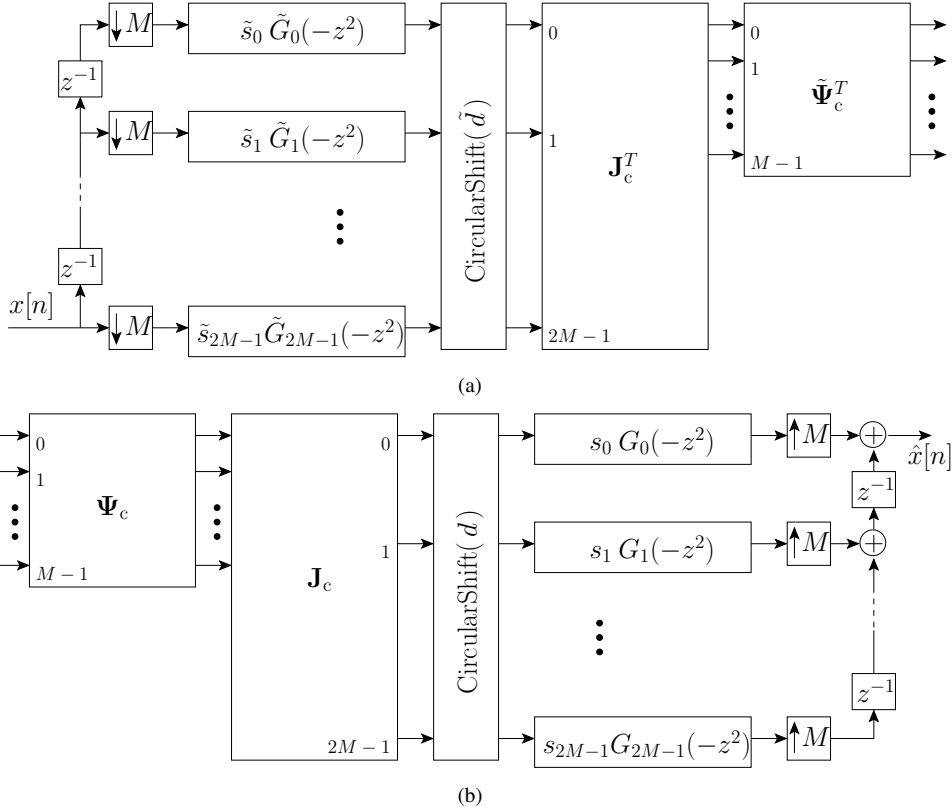
Figure 7.3: "DCT Type-III/IV"-based implementation for biorthogonal odd-stacked CMFB. (a) AFB. (b) SFB.

On the analysis side, we manipulate the argument of the modulating cosine and use the formula $\cos(-\alpha) = \cos(\alpha)$:

$$\begin{aligned}
\left[\tilde{\mathbf{\Phi}}_c\right]_{n,k} &= 2\cos\left[\left(2M - 1 - n - \frac{N + D + M}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right] \\
&= 2\cos\left[\left(n + \frac{1}{2} + \frac{N + D - 3M + 1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right].
\end{aligned}$$

Again we pick the forward DCT-IV basis and the integer-valued offset. Proceeding as above gives parameters

$$\begin{aligned}
\tilde{r} &= \left\lfloor \frac{N + D - 3M + 1}{2}\frac{1}{2M} \right\rfloor \\
\tilde{d} &= (N + D - 3M + 1)/2 - 2\tilde{r}M.
\end{aligned}$$

The sign reversion for the analysis side is denoted with

$$\tilde{s}_\ell = \begin{cases} (-1)^{\tilde{r}} & \ell < 2M - \tilde{d} \\ -(-1)^{\tilde{r}} & \ell \geq 2M - \tilde{d}. \end{cases}$$

Table 7.1: Parameters for the odd-stacked CMFB algorithm in Figure 7.3.

	$N + D - M$ even	$N + D - M$ odd
$\mathbf{\Psi}_{\mathrm{c}}$	$\mathbf{C}_{\mathrm{III}}$	$\mathbf{C}_{\mathrm{IV}}$
$\mathbf{J}_{\mathrm{c}}$	$\begin{bmatrix} \sqrt{2} & \mathbf{0}_{1\times M-1} \\ \mathbf{0}_{M-1\times 1} & \mathbf{I}_{M-1} \\ \mathbf{0}_{1\times M} & \\ \mathbf{0}_{M-1\times 1} & -\mathbf{J}_{M-1} \end{bmatrix}$	$\begin{bmatrix} \mathbf{I}_M \\ -\mathbf{J}_M \end{bmatrix}$
Analysis	$\tilde{r} = \left\lfloor \dfrac{N + D - 3M + 2}{4M} \right\rfloor$ $\tilde{d} = \dfrac{N + D - 3M + 2}{2} - 2\tilde{r}M$	$\tilde{r} = \left\lfloor \dfrac{N + D - 3M + 1}{4M} \right\rfloor$ $\tilde{d} = \dfrac{N + D - 3M + 1}{2} - 2\tilde{r}M$
	$\tilde{s}_\ell = \begin{cases} (-1)^{\tilde{r}} & \ell < 2M - \tilde{d} \\ -(-1)^{\tilde{r}} & \ell \geq 2M - \tilde{d}. \end{cases}$	
Synthesis	$r = \left\lfloor \dfrac{N + D - M}{4M} \right\rfloor$ $d = \dfrac{N + D - M}{2} - 2rM$	$r = \left\lfloor \dfrac{N + D - M + 1}{4M} \right\rfloor$ $d = \dfrac{N + D - M + 1}{2} - 2rM$
	$s_\ell = \begin{cases} -(-1)^r & \ell < d \\ (-1)^r & \ell \geq d. \end{cases}$	

All pieces together writes as

$$\tilde{\mathbf{\Phi}}_{\mathrm{c}}^T = \sqrt{2M}\,\mathbf{C}_{\mathrm{IV}}^T \mathbf{J}_{\mathrm{c}}^T\,\text{CircularShift}(\text{diag}\{\tilde{s}_0, \tilde{s}_1, \ldots, \tilde{s}_{2M-1}\}, \tilde{d}).$$

The matching analysis CMFB algorithm is shown in Figure 7.3 (a).

(b) $N + D - M$ **even**

When this combination of parameters is realized, the offset term $(N + D - M)/2$ is already an integer and the modulation is handled with Type-III DCT. The basis functions of the (inverse) $M \times M$ DCT are

$$\left[\mathbf{C}_{\mathrm{III}}\right]_{n,k} = \begin{cases} \sqrt{2/M}\,\cos\left[(2k+1)\frac{\pi n}{2M}\right] & n \neq 0 \\ \sqrt{1/M} & n = 0. \end{cases}$$

If we extend this to $2M$-length continuous sinusoidal waveform, the $n = 0$ coefficients need scaling by $\sqrt{2}$. There is also a one sample "gap" at $n = M$ because $\cos\left[(2k+1)\frac{\pi}{2}\right] = 0$, $k \in \mathbb{Z}$. Then the mapping matrix is defined as

$$\mathbf{J}_{\mathrm{c}} = \begin{bmatrix} \sqrt{2} & \mathbf{0}_{1\times M-1} \\ \mathbf{0}_{M-1\times 1} & \mathbf{I}_{M-1} \\ \mathbf{0}_{1\times M} & \\ \mathbf{0}_{M-1\times 1} & -\mathbf{J}_{M-1} \end{bmatrix}$$

and the parameters for the circular-shift/sign reversion are

$$r = \left\lfloor \frac{N + D - M}{2} \frac{1}{2M} \right\rfloor$$
$$d = (N + D - M)/2 - 2rM.$$

The analysis side modulation matrix brings no surprises. The matrix decomposition in Figure 7.3 applies with parameters given in Table 7.1.

7.4 DCT/DST-Based Algorithm for CS Odd-Stacked EMFB

The author's earlier approach to generate paraunitary odd-stacked EMFB was mentioned in the introductory chapter: two real MFBs (ELT-based CMFB and SMFB) were used as a building block of the EMFB for complex-valued signals. The same idea applies with biorthogonal critically sampled EMFB; we just need to decompose the subfilters of the SMFB so that the modulation can be computed with Type-III/IV DST algorithms. In analogy with the cosine modulation, the choice of the DST type depends on the parity of $N + D - M$.

7.4.1 Biorthogonal SMFB when $N + D - M$ Odd

The elements of synthesis side modulation matrix are

$$\left[\mathbf{\Phi}_{\mathrm{s}} \right]_{n,k} = 2 \sin \left[\left(n - \frac{N + D - M}{2} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$
$$= 2 \sin \left[\left(n + \frac{1}{2} - \underbrace{\frac{N + D - M + 1}{2}}_{\text{integer}} \right) \left(k + \frac{1}{2} \right) \frac{\pi}{M} \right]$$

and the matrix factors into

$$\mathbf{\Phi}_{\mathrm{s}} = \sqrt{2M} \mathrm{diag}\{s_0, s_1, \ldots, s_{2M-1}\} \, \mathrm{CircularShift}(\mathbf{J}_{\mathrm{s}} \mathbf{S}_{\mathrm{IV}}, d).$$

where $\left[\mathbf{S}_{\mathrm{IV}} \right]_{n,k} = \sqrt{2/M} \sin \left[(n + 1/2)(k + 1/2) \frac{\pi}{M} \right]$ and $\mathbf{J}_{\mathrm{s}} = \begin{bmatrix} \mathbf{I}_M \\ \mathbf{J}_M \end{bmatrix}$. Parameters s_ℓ, r, and d are identical with the odd $N + D - M$ case of the biorthogonal CMFB above. This holds also with the analysis side algorithm:

$$\tilde{\mathbf{\Phi}}_{\mathrm{s}}^T = \sqrt{2M} \mathbf{S}_{\mathrm{IV}}^T \mathbf{J}_{\mathrm{s}}^T \, \mathrm{CircularShift}(\mathrm{diag}\{\tilde{s}_0, \tilde{s}_1, \ldots, \tilde{s}_{2M-1}\}, \tilde{d}).$$

7.4.2 Biorthogonal SMFB when $N + D - M$ Even

Based on the biorthogonal CMFB, the reader may anticipate that in this case DST-III has the responsibility of the SMFB number crunching. Without going into details, the $\mathbf{J}_{\mathrm{s}}$ is given in Table 7.2. $\mathbf{S}_{\mathrm{III}}$ is the inverse transform matrix of Type-III DST.

Table 7.2: Modulation components for the odd-stacked SMFB algorithm. Circular-shift and sign reversion parameters are identical with the CMFB in Table 7.1.

		$N + D - M$ even	$N + D - M$ odd
$\mathbf{\Psi}_\mathrm{s}$		$\mathbf{S}_{\mathrm{III}}$	$\mathbf{S}_{\mathrm{IV}}$
$\mathbf{J}_\mathrm{s}$		$\begin{bmatrix} \mathbf{0}_{1\times M} \\ \mathbf{I}_{M-1} \quad \mathbf{0}_{M-1\times 1} \\ \mathbf{0}_{1\times M-1} \quad \sqrt{2} \\ \mathbf{J}_{M-1} \quad \mathbf{0}_{M-1\times 1} \end{bmatrix}$	$\begin{bmatrix} \mathbf{I}_M \\ \mathbf{J}_M \end{bmatrix}$

7.4.3 Critically Sampled Biorthogonal EMFB with CMFB and SMFB

Now we have all ingredients ready for the critically sampled biorthogonal EMFB which utilizes cosine/sine-modulated FBs as algorithm components. The exponential modulation matrices can be written as

$$\tilde{\mathbf{\Phi}}_{\mathrm{e}}^{H} = \frac{1}{2} \begin{bmatrix} \mathbf{I}_M & -\mathbf{I}_M \\ (-1)^{N+D-M}\mathbf{J}_M & (-1)^{N+D-M}\mathbf{J}_M \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{\Phi}}_{\mathrm{c}}^{T} \\ j\,\tilde{\mathbf{\Phi}}_{\mathrm{s}}^{T} \end{bmatrix}$$

and

$$\mathbf{\Phi}_{\mathrm{e}} = \frac{1}{2} \begin{bmatrix} \mathbf{\Phi}_{\mathrm{c}} & j\,\mathbf{\Phi}_{\mathrm{s}} \end{bmatrix} \begin{bmatrix} \mathbf{I}_M & (-1)^{N+D-M}\mathbf{J}_M \\ \mathbf{I}_M & -(-1)^{N+D-M}\mathbf{J}_M \end{bmatrix},$$

where the $(-1)^{N+D-M}$-term originates from (4.29) and (4.30). The flowgraph of the analysis EMFB algorithm is shown in Figure 7.4 (a). The output subsamples from the analysis polyphase filters are split into real and imaginary parts. The real part is processed with cosine modulation block and, conversely, the imaginary part is handled with sine modulation. Modulation block outputs are combined into real-valued subsignals of the critically sampled EMFB. The order of the negative side subsignals is reversed. The SFB performs the corresponding inverse operations as shown in Figure 7.4 (b).

A quick verification of the factoring is in order the check that $\mathbf{\Phi}_{\mathrm{e}}$ is equivalent with the definition in Section 7.2. Firstly, the block matrix multiplication produces

$$\mathbf{\Phi}_{\mathrm{e}} = \frac{1}{2} \begin{bmatrix} \mathbf{\Phi}_{\mathrm{c}} + j\,\mathbf{\Phi}_{\mathrm{s}} & (-1)^{N+D-M}(\mathbf{\Phi}_{\mathrm{c}} - j\,\mathbf{\Phi}_{\mathrm{s}})\mathbf{J} \end{bmatrix}.$$

Matrix multiplication by $\mathbf{J}$ from the right reverses the order of columns so that we can substitute the elements and simplify

$$
[\mathbf{\Phi}_{\mathrm{e}}]_{n,k} = \begin{cases} \frac{1}{2}\left([\mathbf{\Phi}_{\mathrm{c}}]_{n,k} + j\,[\mathbf{\Phi}_{\mathrm{s}}]_{n,k} \right) & k < M \\ \frac{(-1)^{N+D-M}}{2}\left([\mathbf{\Phi}_{\mathrm{c}}]_{n,2M-1-k} - j\,[\mathbf{\Phi}_{\mathrm{s}}]_{n,2M-1-k} \right) & k \geq M \end{cases}
$$
$$
= \begin{cases} \exp\left[j\left(n - \dfrac{N+D-M}{2} \right)\left(k + \dfrac{1}{2} \right)\dfrac{\pi}{M} \right] & k < M \\ (-1)^{N+D-M}\exp\left[-j\left(n - \dfrac{N+D-M}{2} \right)\left(2M-1-k + \dfrac{1}{2} \right)\dfrac{\pi}{M} \right] & k \geq M. \end{cases}
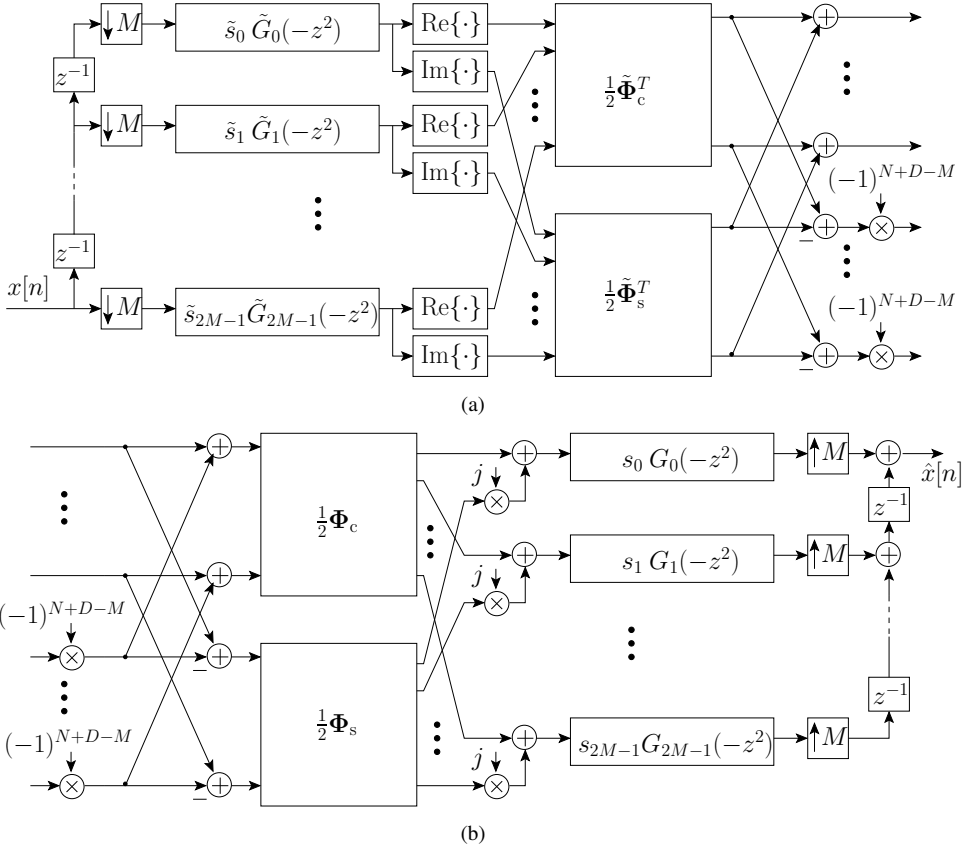$$

Figure 7.4: Critically sampled odd-stacked biorthogonal EMFB composed of CMFB and SMFB algorithms. (a) AFB. (b) SFB.

The positive frequency side is already in the desired form. When $k \geq M$, $\big[\mathbf{\Phi}_{\mathrm{e}}\big]_{n,k}$ becomes

$$
\begin{aligned}
&\big[\mathbf{\Phi}_{\mathrm{e}}\big]_{n,k} \\
&= (-1)^{N+D-M} \exp\big[-j\big(n - \tfrac{N+D-M}{2}\big)\big(-k - \tfrac{1}{2}\big)\tfrac{\pi}{M}\big] \exp\big[-j\big(n - \tfrac{N+D-M}{2}\big)2\pi\big] \\
&= (-1)^{N+D-M} \exp\big[j\big(n - \tfrac{N+D-M}{2}\big)\big(k + \tfrac{1}{2}\big)\tfrac{\pi}{M}\big] \underbrace{e^{-j2\pi n}}_{=1} \underbrace{e^{j(N+D-M)\pi}}_{=(-1)^{N+D-M}}
\end{aligned}
$$

Thus the elements of the odd-stacked CMFB and SMFB modulation matrices constitute implementation structure for CS odd-stacked EMFB.

7.5   FFT-Based Algorithm for Even-Stacked EMFB

In the MFB theory (cf. Section 4.5) we first discussed the even-stacked EMFB as a time varying FB obtained from odd-stacked EMFB via frequency shifting. This model was modified into a two parallel FBs to analyze it with polyphase domain matrices. The former approach

is more suitable for the FFT-based algorithm derivation, but instead of frequency shifting with demodulation multiplier, equation (4.42) should be considered. There we provided the formula for the fixed even-stacked subfilters $f_k^{\mathrm{e-e}}[n]$. Moreover, it was additionally mentioned that $f_k^{\mathrm{e-e}}[n] = j\,\breve{f}_k^{\mathrm{e-e}}[n]$; the two sets of synthesis subfilters are related by interchanging the real and imaginary parts. Ergo, we just flip the real and imaginary parts of the subsamples (when subsample time index m is odd) for the same effect. Then the remaining task is to decompose the modulation block.

The modulation matrices for the even-stacked EMFB are

$$\left[\mathbf{\Phi}_{\mathrm{e-e}}\right]_{n,k} = \exp\left[j\left(n - \frac{N+D-M}{2}\right)\frac{k\pi}{M}\right]$$

and

$$\left[\tilde{\mathbf{\Phi}}_{\mathrm{e-e}}\right]_{n,k} = \exp\left[-j\left(2M - 1 - n - \frac{N+D+M}{2}\right)\frac{k\pi}{M}\right].$$

As demonstrated with odd-stacked modulation, the DFT basis can be extracted and the matrices can be given as

$$\mathbf{\Phi}_{\mathrm{e}} = \sqrt{2M}\,\mathbf{\Psi}_{2M}\,\mathrm{diag}\{w_0, w_1, \ldots, w_{2M-1}\} = \sqrt{2M}\,\mathbf{\Psi}_{2M}\,\mathbf{W}.$$

and

$$\tilde{\mathbf{\Phi}}_{\mathrm{e-e}}^H = \sqrt{2M}\,\mathrm{diag}\{\tilde{w}_0^*, \tilde{w}_1^*, \ldots, \tilde{w}_{2M-1}^*\}\,\mathbf{\Psi}_{2M}^H = \sqrt{2M}\,\tilde{\mathbf{W}}^*\,\mathbf{\Psi}_{2M}^H$$

where $w_k = \exp\left[\frac{-j(N+D-M)k\pi}{2M}\right]$ and $\tilde{w}_k = \exp\left[\frac{j(N+D-3M+2)k\pi}{2M}\right]$. The other component of the fixed part is the prototype filter as defined in (7.2) and (7.3) for even-stacked modulation. Due to $2M$-periodicity there are no sign reversion modifications.

The time varying property is incorporated into the subband system with a sequence $q[m] = \{\ldots, 1, j, 1, j, \ldots, \}$. This applies on the synthesis side and effectively interchanges the real and imaginary parts when the subsignals are 2x oversampled. If the even-stacked EMFB is CS, the previous reduces to alternating purely real/imaginary-valued subsignals which are then processed by the fixed part of the SFB. Perfect-reconstruction condition is satisfied when we use $\tilde{q}[m] = q^*[m]$ on the analysis side. This can be deduced from the connection between $h_k^{\mathrm{e-e}}[n]$ and $\breve{h}_k^{\mathrm{e-e}}[n]$. The flowgraph of the even-stacked EMFB is shown in Figure 7.5, where the DFT basis is replaced with FFT.

It is also possible to express the time varying system with polyphase matrices. In order to do so, subsignals $y_k[m]$ are divided into even and odd samples by denoting $Y_{k,\mathrm{even}}(z) = \mathcal{Z}\{y_k[2m]\}$ and $Y_{k,\mathrm{odd}}(z) = \mathcal{Z}\{y_k[2m+1]\}$. An operator $Q\{\cdot\}$ combines these two sequences

$$Q\{Y_k(z)\} = Y_{k,\mathrm{even}}(z^2) + j\,z^{-1}Y_{k,\mathrm{odd}}(z^2).$$

The above is the z-transform domain equivalent to $q[m]Y_k[m]$. If $Q\{\cdot\}$ operates on each element of the argument vector, the signal synthesis becomes

$$\hat{X}(z) = \sqrt{2M}\boldsymbol{c}^T(z)\begin{bmatrix}\mathbf{I} & z^{-M}\mathbf{I}\end{bmatrix}\boldsymbol{G}(z^M)\,\mathbf{\Psi}_{2M}\,\mathbf{W}\,Q\{\boldsymbol{y}(z^M)\},$$

where we applied (2.24) and (7.1).

The analysis side operator $\tilde{Q}\{\cdot\}$ is a conjugated version of $Q\{\cdot\}$. Then we just write the algorithm components in reverse order:

$$\boldsymbol{y}(z) = \sqrt{2M}\tilde{Q}\{\tilde{\mathbf{W}}\mathbf{\Psi}_{2M}^H\tilde{\boldsymbol{G}}(z)\begin{bmatrix}z^{-1}\mathbf{I} \\ \mathbf{I}\end{bmatrix}\boldsymbol{x}_{\mathrm{p}}(z)\}.$$
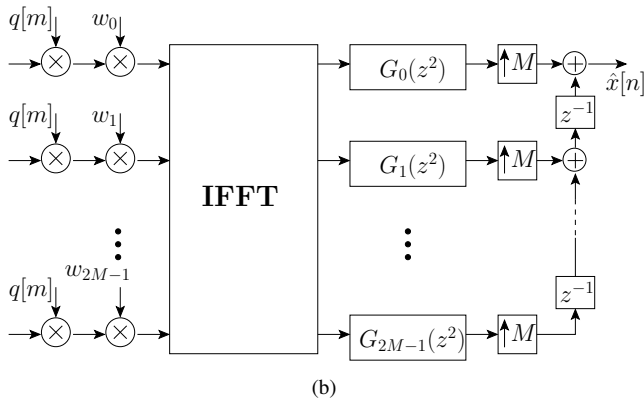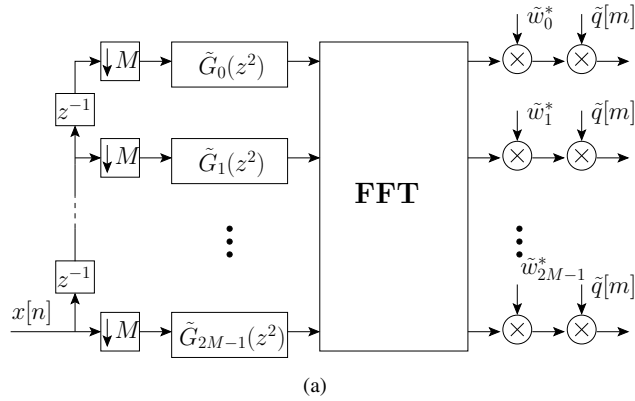
Figure 7.5: FFT-based implementation for biorthogonal even-stacked EMFB. (a) AFB. (b) SFB.
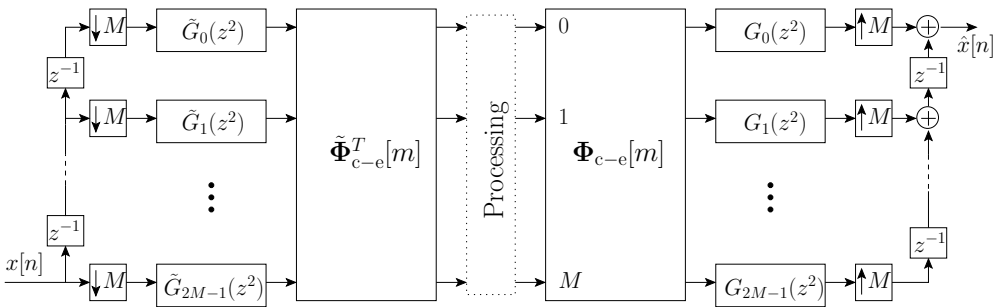

Figure 7.6: Biorthogonal even-stacked CMFB with time varying modulation.

At this point it is apparent that the FFT-based algorithm is much easier to grasp from the flowgraph than using the polyphase domain formulae. As such, the described algorithm operates in 2xOS mode; critical subsampling needs the real-part operator after the AFB and scaling by 2.
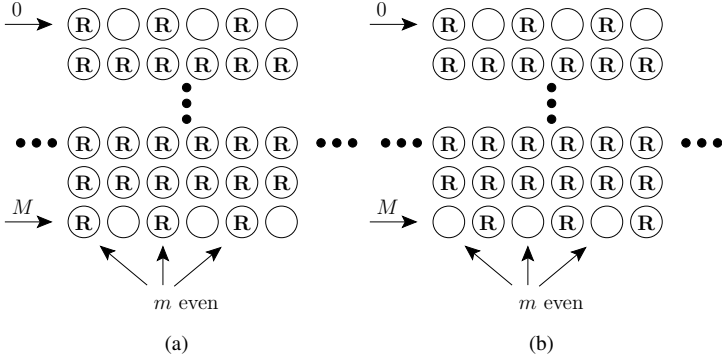
Figure 7.7: Critically sampled (with decimation factor M) even-stacked CMFB has place-holder zero subsamples in low/high-pass subchannels. The exact ordering depends on the modulation sequence parameters. (a) $N + D - M$ even. (b) $N + D - M$ odd.

7.6 DCT/FFT-Based Algorithms for Even-Stacked CMFB

The implementation algorithm for even-stacked CMFB presented here is slightly different from the FBs in Figures 4.8 and 4.9. Though the fixed FB system has $2M$ subfilters they effectively divide the frequency range $[0, \pi]$ (real-valued input signal) into $M + 1$ subbands. Let us first consider a general even-stacked CMFB model in Figure 7.6 which encloses both Type-I and Type-II subband configurations. The familiar component is the prototype filter implemented as even-stacked polyphase filter for data windowing and folding. The modulation functionality is handled with time varying matrices $\tilde{\boldsymbol{\Phi}}_{c-e}[m]$ (analysis) and $\boldsymbol{\Phi}_{c-e}[m]$ (synthesis) which have dimensions $2M \times (M + 1)$. We concentrate our factoring efforts on the SFB modulation matrix and justify certain choices in algorithm derivation. The previous discussion with odd-stacked modulation contains tools for addressing the analysis side in an analytic manner. However, there is another route to find the AFB implementation structure: Once the synthesis side factoring is obtained, we just write the inverse operations in reverse order. And that's it!

In a time varying model for even-stacked synthesis CMFB, the parity of subsignal time index m chooses between cosine and sine modulation:

$$\boldsymbol{\Phi}_{c-e}[m] = \begin{cases} \boldsymbol{\Phi}_{\text{even}} & m \text{ even} \\ \boldsymbol{\Phi}_{\text{odd}} & m \text{ odd}. \end{cases}$$

The elements of these alternating matrices are $2M$-length sequences from the even-stacked MFB subfilter definition:

$$\left[\boldsymbol{\Phi}_{\text{even}}\right]_{n,k} = c_k \cos\left[\left(n - \frac{N + D - M}{2}\right)\frac{k\pi}{M}\right] \tag{7.4}$$

$$\left[\boldsymbol{\Phi}_{\text{odd}}\right]_{n,k} = -c_k \sin\left[\left(n - \frac{N + D - M}{2}\right)\frac{k\pi}{M}\right], \tag{7.5}$$

where $c_k = \sqrt{2}$ when $k = 0$ or $k = M$. Otherwise the scaling term $c_k = 2$ is applied with the bandpass subchannels. It should be noted that if $k = 0$ then the 0th column of $\boldsymbol{\Phi}_{\text{odd}}$ is a zero vector. Similarly, when $k = M$ then the Mth column of either $\boldsymbol{\Phi}_{\text{even}}$ or $\boldsymbol{\Phi}_{\text{odd}}$ is zero depending on the parity of $N + D - M$. If we observe this from the subband processing
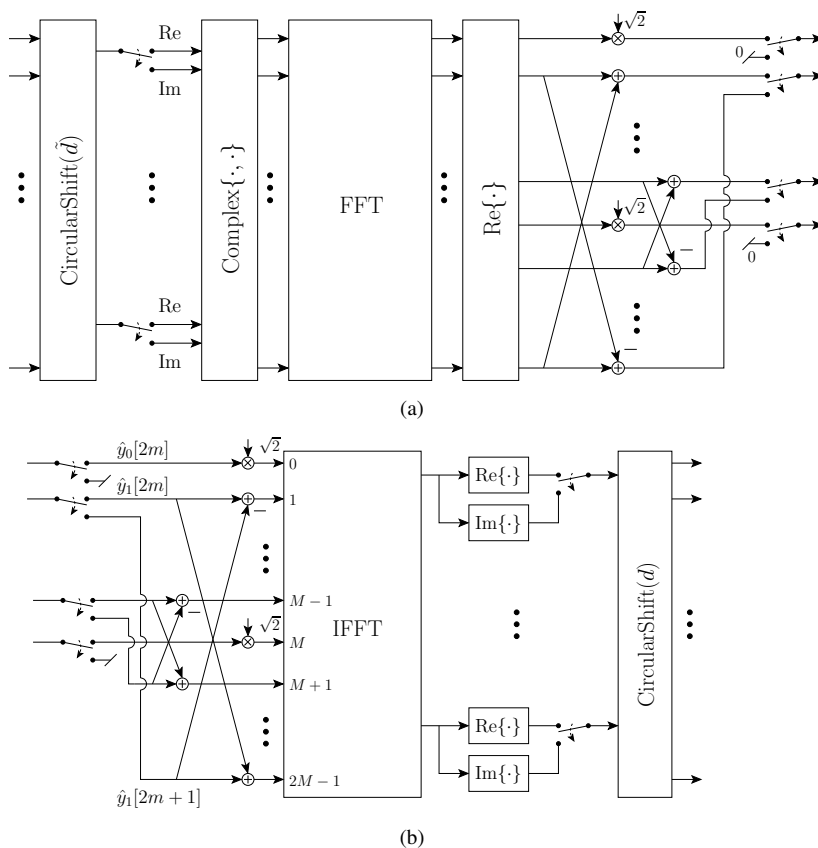
(a)

(b)

Figure 7.8: Modulation block for Type-I biorthogonal even-stacked CMFB. The alternating cosine- and sine modulation is combined into FFT-based algorithm. (a) Modulation for the AFB. (b) Modulation for the SFB.

algorithm viewpoint, the arrangement interleaves the low/high-pass subsignals with phantom (or placeholder) samples. This is schematically shown in Figure 7.7. The reason behind this frequency ordering is to match the subsignals of the cosine- and sine-modulated subfilters having the same center frequency. In signal synthesis, these zero samples can be ignored and to emphasize this fact we "hardwire" them into the presented even-stacked CMFB algorithms.

The task is again to implement modulation matrices with fast algorithms. At first glance on the modulation sequence, we might notice the phase shifted Type-I DCT/DST basis when $N + D - M$ is even. Though the possibility of constructing such MFB algorithm is recognized here, there is an opportunity for the FFT-based algorithm. Besides, with a suitably modified $2M$-point FFT algorithm we can avoid the notationally "ugly" scaling terms associated with the DCT-I definition. With this sidenote we can start breaking $\mathbf{\Phi}_{c-e}[m]$ apart.

7.6.1 Modulation for Biorthogonal Even-Stacked CMFB when $N + D - M$ Even

This case matches with the Type-I even-stacked CMFB subband configuration. Now the offset term in (7.4) and (7.5) is an integer without add-subtract manipulations. Again the shift parameter d is the remainder when the multiple of $2M$ term is subtracted from the offset:

$$d = (N + D - M)/2 - 2rM,$$

where $r = \left\lfloor \frac{N+D-M}{2} \frac{1}{2M} \right\rfloor$. What remains here is the mapping of the subsignals into a weighted linear combination of cosine/sine sequences utilizing the FFT algorithm. When m is even, the set of subfilters includes the half-bandwidth low and high-pass subfilters with indices $k = 0$ and $k = M$, respectively. With scaling correction by $\sqrt{2}$, the matching modulation is directly obtained from the 0th/Mth subchannels of the $2M$-point IFFT. Then we can write $2\cos[nk\pi/M] = W_{2M}^{-nk} + W_{2M}^{nk}$ and remember the conjugate symmetry property of the DFT basis: real input to kth and $2M - k$th generates the desired cosine sequence when $k \in \{1, 2, \ldots, M - 1\}$.

When m is odd, there are $M - 1$ bandpass subchannels. Again the conjugate symmetry property is useful to produce the desired sine-sequence, but it is required to be purely imaginary-valued: $-2j\sin[nk\pi/M] = -W_{2M}^{-nk} + W_{2M}^{nk}$. This sequence is generated with $2M$-point IFFT with real input to kth and $2M - k$th subchannel, but reversing the sign of the kth band. The catch in this arrangement is to combine each alternating cosine/sine modulation into a single FFT computation.

Figure 7.8 (b) displays the above described algorithm for the SFB. The input buffering maps a pair of subsignal vectors into $2M$-length real-valued data for the IFFT. Because of the DFT linearity, the desired modulation components are separable (real/imaginary-split) and are interleaved before processing with circular-shift. The forward algorithm in Figure 7.8 (a) is essentially composed of transposed operations in reverse order. A correct modulation phase offset for the AFB is defined as $\tilde{r} = \left\lfloor \frac{N+D-3M+2}{2} \frac{1}{2M} \right\rfloor$ and $\tilde{d} = (N+D-3M+2)/2-2\tilde{r}M$. If we are faithful to the CMFB definition, the FFT/IFFT-pair needs to be normalized. The scaling correction can be pre-multiplied into the prototype filter coefficients.

7.6.2 Modulation for Biorthogonal Even-Stacked CMFB when $N + D - M$ Odd

The other case, Type-II even-stacked CMFB, can be realized with DCT/DST-II transforms. For instance, the basis functions of DCT-II inverse transform are

$$[\mathbf{C}_{\mathrm{II}}]_{n,k} = \begin{cases} 1/\sqrt{M} & k = 0 \\ \sqrt{\frac{2}{M}} \cos\left[\left(n + \frac{1}{2}\right)\frac{k\pi}{M}\right] & 0 < k \le M - 1. \end{cases} \tag{7.6}$$

When the "add-subtract-by-1/2"-trick is applied with the cosine modulation matrix, we obtain the desired form where the offset term $\frac{N+D-M+1}{2}$ is an integer. For the modulation phase adjustment this is rewritten as $\frac{N+D-M+1}{2} = 2rM + d$, where r is an integer and the remainder $d \in \{0, 1, \ldots, 2M - 1\}$. Parameter d is the argument of the synthesis side circular-shift function. The mapping matrix that extends the DCT-II into $2M$-length sequence is

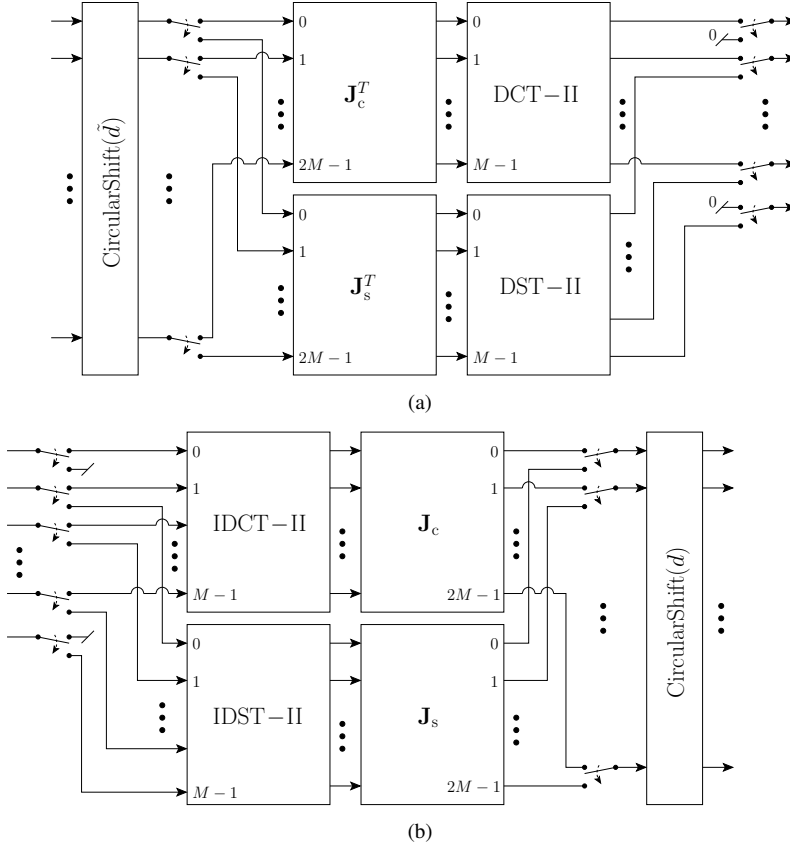$$\mathbf{J}_{\mathrm{c}} = \begin{bmatrix} \mathbf{I} \\ \mathbf{J} \end{bmatrix}.$$

Figure 7.9: Modulation block for Type-II biorthogonal even-stacked CMFB. The subband samples $k \in [0, M-1]$ are computed with DCT-II when the subsignal time index is even. DST-II handles the other case and the subband indices are then $k \in [1, M]$. (a) Modulation for the AFB. (b) Modulation for the SFB.

The final twist in the factoring of the $\boldsymbol{\Phi}_{\text{even}}$ is to scale the 0th subband with $c_0 = \sqrt{2}$. But this is already built in the DCT-II basis definition and we can write

$$\boldsymbol{\Phi}_{\text{even}} = \begin{bmatrix} \text{CircularShift}\{\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}}, d\} & \mathbf{0}_{2M \times 1} \end{bmatrix},$$

where the effectively zero Mth column of $\boldsymbol{\Phi}_{\text{even}}$ can be left out in the implementation. Depending on the implementation algorithm of the fast DCT-II (and DST-II below), scaling terms might be required to normalize the analysis-synthesis pair to level with the CMFB subfilter definition. The basis functions in (7.6) need correction with $\sqrt{2M}$-multiplier.

The sine modulation in $\boldsymbol{\Phi}_{\text{odd}}$ is covered with DST-II basis:

$$[\mathbf{S}_{\text{II}}]_{n,k} = \begin{cases} \sqrt{\frac{2}{M}} \sin\left[\left(n + \frac{1}{2}\right)\frac{(k+1)\pi}{M}\right] & 0 \le k < M-1. \\ (-1)^n/\sqrt{M} & k = M-1. \end{cases} \tag{7.7}$$

The subband configuration of the DST-II corresponds with columns $1, 2, \ldots, M$ in (7.5). Ex-

tending the sine modulation sequence requires its own mapping matrix

$$
\mathbf{J}_{\mathrm{s}} = \left[\begin{array}{c} -\mathbf{I} \\ \mathbf{J} \end{array}\right]
$$

to match the sign and make the waveforms continuous. The circular-shift parameter d is given above with the DCT-II. If we similarly ignore the 0th column of $\mathbf{\Phi}_{\mathrm{odd}}$, the modulation matrix is

$$
\mathbf{\Phi}_{\mathrm{odd}} = \left[\mathbf{0}_{2M \times 1} \quad \mathrm{CircularShift}\{\mathbf{J}_{\mathrm{s}}\mathbf{S}_{\mathrm{II}}, d\}\right]
$$

when m odd.

A slight modification for the sine modulation is to replace DST-II with DCT-II algorithm. This may come handy, because usually a fine-tuned version of the DCT-II is more readily available in signal processing software/hardware libraries. In that case, the connection

$$
\mathbf{S}_{\mathrm{II}} = \mathrm{diag}\{1, -1, 1, -1, \ldots\}\mathbf{C}_{\mathrm{II}}\mathbf{J} \tag{7.8}
$$

allows us to rely solely on the fast DCT-II algorithm in the Type-II even-stacked CMFB implementation.

The synthesis modulation block is shown in Figure 7.9 (b). The samples of the CMFB subbands $k \in \{0, 1, \ldots, M-1\}$ are computed with the DCT-II path when m even. Conversely, DST-II processes the odd m samples and then $k \in \{1, 2, \ldots, M\}$. The $\mathbf{J}_{\mathrm{c}}/\mathbf{J}_{\mathrm{s}}$-extended sequences are interleaved and yet shifted by d before the polyphase filters.

The flowgraph in Figure 7.9 (a) describes the analysis side de-modulation. It's building blocks are covered in the discussion above or can be obtained by matrix inversion. The shift-parameter can be derived in analogy with the odd-stacked case in Section 7.3: $\tilde{r} = \left\lfloor \frac{N+D-3M+1}{2} \frac{1}{2M} \right\rfloor$ and $\tilde{d} = (N + D - 3M + 1)/2 - 2\tilde{r}M$. Then we may proceed to apply the above with even-stacked EMFB algorithm.

7.7 DCT/DST-Based Algorithm for CS Even-Stacked EMFB

The last of these block-transform-based algorithms for biorthogonal MFBs is the critically sampled even-stacked EMFB. In particular, we consider case $N + D - M$ odd where the implementation can be grounded on DCT/DST-II-based algorithms. The SFB modulation matrix is generated with the following factoring

$$
\mathbf{\Phi}_{\mathrm{e-e}} = \mathrm{CircularShift}\left\{\sqrt{M/2}\left[\mathbf{J}_{\mathrm{c}}\mathbf{C}_{\mathrm{II}} \quad j\,\mathbf{J}_{\mathrm{s}}\mathbf{S}_{\mathrm{II}}\right]\mathbf{J}_{\mathrm{cs}}, d\right\}.
$$

The $2M$-length modulation sequences $\mathbf{J}_{\mathrm{c}}\mathbf{C}_{\mathrm{II}}$ and $\mathbf{J}_{\mathrm{s}}\mathbf{S}_{\mathrm{II}}$ are familiar from the even-stacked CMFB algorithm in Section 7.6 (b):

$$
\sqrt{M/2}\left[\mathbf{J}_{\mathrm{c}}\mathbf{C}_{\mathrm{II}}\right]_{n,k} = \begin{cases} 1/\sqrt{2} & k = 0 \\ \cos\left[(n+1/2)\frac{k\pi}{M}\right] & 0 < k \le M-1 \end{cases}
$$

and

$$
\sqrt{M/2}\left[\mathbf{J}_{\mathrm{s}}\mathbf{S}_{\mathrm{II}}\right]_{n,k} = \begin{cases} -\sin\left[(n+1/2)\frac{k\pi}{M}\right] & 0 \le k < M-1 \\ -(-1)^n/\sqrt{2} & k = M-1. \end{cases}
$$

A mapping matrix $\mathbf{J}_{\text{cs}}$ combines the cosine and sine sequences into exponential modulation

$$
\mathbf{J}_{\text{cs}} = \begin{bmatrix} \text{diag}\{\sqrt{2}, \underbrace{1, \ldots, 1}_{\#(M-1)}\} & -\mathbf{S}_{-1}\mathbf{J} \\ -\mathbf{S}_1 & -\text{diag}\{\underbrace{1, \ldots, 1}_{\#(M-1)}, \sqrt{2}\}\mathbf{J} \end{bmatrix}.
$$

In block-by-block multiplication $-\mathbf{S}_1$ moves the columns of $j\,\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}$ right by one position (in a non-circular manner) and reverses the signs of the elements. Similarly, $\mathbf{S}_{-1}\mathbf{J}$ shifts $\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}}$ left and then reverses the order of columns. Scaling term $\sqrt{2}$ in diagonal matrices effect on columns $k = 0$ and $k = M$. This equates as

$$
\begin{aligned}
&\big[[\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}} \quad j\,\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}]\mathbf{J}_{\text{cs}}\big]_{n,k} \\
&= \begin{cases} \sqrt{2}\big[\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}}\big]_{n,0} & k = 0 \\ \big[\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}}\big]_{n,k} - j\,\big[\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}\big]_{n,k-1} & 0 < k < M \\ -\sqrt{2}\big[\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}\big]_{n,2M-1-M} & k = M \\ -\big(\big[\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}}\big]_{n,2M-1-k+1} + j\,\big[\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}\big]_{n,2M-1-k}\big) & M < k \le 2M-1 \end{cases}
\end{aligned}
$$

in terms of extended modulation sequence matrices. The cosine/sine-elements can now be substituted and after some simplification the four cases are

$$
\begin{aligned}
&\sqrt{M/2}\big[[\mathbf{J}_{\text{c}}\mathbf{C}_{\text{II}} \quad j\,\mathbf{J}_{\text{s}}\mathbf{S}_{\text{II}}]\mathbf{J}_{\text{cs}}\big]_{n,k} \\
&= \begin{cases} 1 & k = 0 \\ \exp\big[j(n+1/2)\frac{k\pi}{M}\big] & 0 < k < M \\ j\,(-1)^n & k = M \\ -\exp\big[-j(n+1/2)\frac{(2M-k)\pi}{M}\big] & M < k \le 2M-1. \end{cases}
\end{aligned} \tag{7.9}
$$

We can get rid of the conditions for k altogether by noticing that $\exp\big[j(n+1/2)\frac{0\pi}{M}\big] = 1$ (when $k = 0$) and $\exp\big[j(n+1/2)\pi\big] = e^{j\pi n}e^{j\pi/2} = (-1)^n\,j$ (when $k = M$). The negative frequency side is not problematic either

$$
-\exp\big[-j(n+1/2)\tfrac{(2M-k)\pi}{M}\big] = -\exp\big[j(n+1/2)\tfrac{k\pi}{M}\big]\underbrace{e^{-j2\pi n}}_{=1}\underbrace{e^{-j\pi}}_{=-1}.
$$

As a consequence, the different conditions in (7.9) can be merged. The last detail in modulation matrix generation is the circular-shift parameter $d = (N + D - M + 1)/2 - 2rM$ where r is an integer (cf. Section 7.6 (b)). Processing the exponential sequence with the circular-shift function causes a phase shift:

$$
\begin{aligned}
&\exp\big[j\big(n - d + 1/2\big)\tfrac{k\pi}{M}\big] \\
&= \exp\big[j\big(n - \tfrac{N+D-M}{2} - 1/2 + 1/2\big)\tfrac{k\pi}{M}\big]\,\exp\big[j2rM\tfrac{k\pi}{M}\big] \\
&= \exp\big[j\big(n - \tfrac{N+D-M}{2}\big)\tfrac{k\pi}{M}\big]\underbrace{\exp\big[j2\pi kr\big]}_{=1} = \big[\mathbf{\Phi}_{\text{e-e}}\big]_{n,k}.
\end{aligned}
$$

Therefore, the CS even-stacked EMFB can be build around a pair of DCT/DST-II transforms.
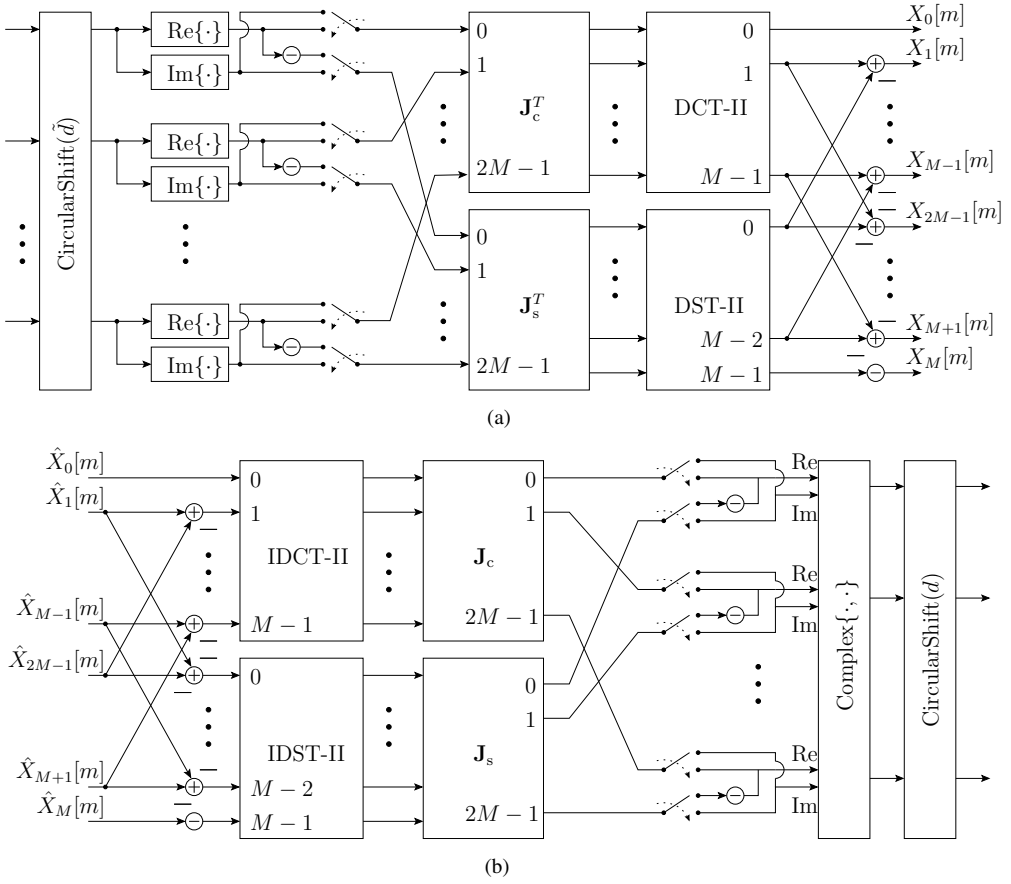
Figure 7.10: Modulation block for critically sampled even-stacked EMFB based on DCT/DST-II algorithms. (a) Analysis. (b) Synthesis.

The synthesis modulation block is shown in Figure 7.10 (b). Before the circular-shift function, there is an operation that interchanges the real-imaginary parts of the samples when the subsignal index m is odd. This takes into account the time varying property of the even-stacked EMFB. (The multiplier sequence $q[m]$ has the same role in the FFT-based even-stacked EMFB as shown in Figure 7.5 (b).) Note also the reverse ordering of the negative frequency side subsignals in Figure to make the flowgraph aesthetically more pleasing. The DCT/DST-II pair can be replaced with the combined algorithm described above.

The analysis side modulation is generated with the algorithm shown in Figure 7.10 (a). Again we can argue that the structure is a mirror image of the synthesis side algorithm to make the pair perfectly invertible. The modulation block for CS even-stacked analysis EMFB inherits the circular-shift parameter $\tilde{d}$ from the CMFB algorithm. But once more, $\tilde{r} = \left\lfloor \frac{N+D-3M+1}{2} \frac{1}{2M} \right\rfloor$ and it limits $\tilde{d} \in \{0, 1, \ldots, 2M - 1\}$ with $\tilde{d} = (N + D - 3M + 1)/2 - 2\tilde{r}M$.

7.8 Modified FFT Algorithm for Oversampled EMFBs

7.8.1 Removing the Phase Rotating Multipliers of Even-Stacked EMFB

The general FFT-based implementation algorithm for even-stacked 2xOS EMFB shown in Figure 7.5 has two arrays of complex-valued multipliers for the analysis and synthesis sides $\tilde{w}_k = \exp\left[\frac{j(N+D-3M+2)k\pi}{2M}\right]$ and $w_k = \exp\left[\frac{-j(N+D-M)k\pi}{2M}\right]$, respectively. Our objective is now to reduce the computational complexity by modifying a FFT to embed these multiplications in the transform algorithm. Here we restrict to the most interesting case where M is a power of two (radix-2 transforms are applicable) and $N + D - M$ is odd. These parameter values cover the paraunitary and the best biorthogonal designs given in Table 5.2.

The basis functions of the DFT does not directly contain the required EMFB phases so we tweak the matrix a bit. Let us then consider an $M \times M$ matrix $\boldsymbol{\Theta}$ whose elements are

$$[\boldsymbol{\Theta}]_{n,k} = \frac{1}{\sqrt{M}} W_M^{-(n+1/2)k}.$$

It is orthogonal ($\boldsymbol{\Theta}^H \boldsymbol{\Theta} = \mathbf{I}$), but in the algorithm derivation we assume that the scaling term $\frac{1}{\sqrt{M}}$ of the forward matrix is combined with the one on the inverse side. (Downscaling by M can be implemented with bit shifting when using fixed point arithmetic.) The M-point forward transform of vector $\mathbf{x}$ is $\mathbf{X} = \boldsymbol{\Theta}_M^H \mathbf{x}$ and can be given for each element $[\mathbf{X}]_k = X_k$ as

$$X_k = \sum_{n=0}^{M-1} x_n W_M^{(n+1/2)k}.$$

The modified FFT algorithm derivation follows the example given in [73] for decimation in frequency algorithm. First the sum is written in two parts and simplified:

$$
\begin{aligned}
X_k &= \sum_{n=0}^{M/2-1} x_n W_M^{(n+1/2)k} + \sum_{n=0}^{M/2-1} x_{n+M/2} W_M^{(n+M/2+1/2)k} \\
&= \sum_{n=0}^{M/2-1} x_n W_M^{(n+1/2)k} + \underbrace{W_M^{kM/2}}_{(-1)^k} x_{n+M/2} W_M^{(n+1/2)k} \\
&= \sum_{n=0}^{M/2-1} \left(x_n + (-1)^k x_{n+M/2}\right) W_M^{(n+1/2)k}.
\end{aligned}
$$

Then we consider the even and odd terms of X_k separately and apply the fact that $W_M^2 = W_{M/2}$:

$$X_{2k} = \sum_{n=0}^{M/2-1} \left(x_n + x_{n+M/2}\right) W_{M/2}^{(n+1/2)k}$$

$$X_{2k+1} = \sum_{n=0}^{M/2-1} \left((x_n - x_{n+M/2})W_M^{n+1/2}\right) W_{M/2}^{(n+1/2)k}$$
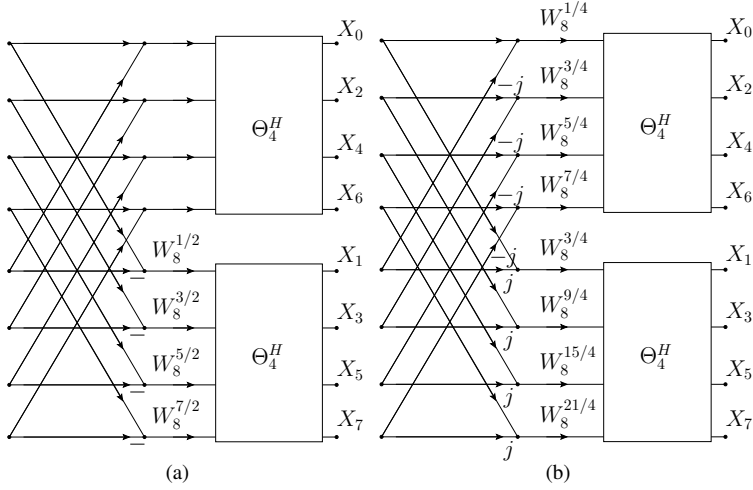
Figure 7.11: Modified decimation-in-frequency FFTs for EMFB when $M = 8$. (a) Even-stacked. (b) Odd-stacked.

The multiplier $W_M^{n+1/2}$ is called in some occurrences as the twiddle factor. By denoting the intermediate sequences with $g_{0,n} = x_n + x_{n+M/2}$ and $g_{1,n} = (x_n - x_{n+M/2})W_M^{n+1/2}$ it can be observed that

$$X_{2k} = \sum_{n=0}^{M/2-1} g_{0,n} W_{M/2}^{(n+1/2)k}$$

$$X_{2k+1} = \sum_{n=0}^{M/2-1} g_{1,n} W_{M/2}^{(n+1/2)k}$$

are equivalent to two independent $\Theta_{M/2}^H$ transforms.

The above derivation is shown schematically in Figure 7.11 (a) for $M = 8$. It should be noted that repeated application of the above procedure for $\Theta_{M/2} \to \Theta_2$ permutes the output. Thus the modified forward FFT algorithm is equipped with output re-ordering. An in-place version of this algorithm can be found in Algorithm 7.1.

The last step is to correct the phases of Θ_{2M} matrix to match with the even-stacked EMFB modulation matrices $\tilde{\mathbf{\Phi}}_{\mathrm{e-e}}$ and $\mathbf{\Phi}_{\mathrm{e-e}}$. On the analysis side the modulation block operates with $2M$-length vectors $\mathbf{X} = \tilde{\mathbf{\Phi}}_{\mathrm{e-e}}^H \mathbf{x}$, where

$$\left[\tilde{\mathbf{\Phi}}_{\mathrm{e-e}}^H\right]_{n,k} = \frac{1}{\sqrt{2M}} W_{2M}^{nk} W_{2M}^{n(N+D-3M+2)/2}$$

$$= \frac{1}{\sqrt{2M}} W_{2M}^{n(k+(N+D-3M+2)/2)}$$

$$= \frac{1}{\sqrt{2M}} W_{2M}^{n(k+1/2+(N+D-3M+1)/2)}.$$

Since $(N + D - 3M + 1)/2$ was assumed to be an integer, the term can be interpreted as a row-shifting for matrix Θ_{2M}^H. Equivalently we can write

$$\mathbf{X} = \Theta_{2M}^H \, \mathrm{CircularShift}(\mathbf{x}, \tilde{d}),$$

141

Algorithm 7.1 Modified FFT algorithm to compute $\mathbf{X} = \boldsymbol{\Theta}_M^H \mathbf{x}$.

Decimation-in-frequency:
$r = \log_2 M$
$N = 1$
for $k = r - 1 \rightarrow 0$ **do**
   $s = 0$
   $M_k = 2^k$
   for $\ell = 1 \rightarrow N$ **do**
     for $m = s \rightarrow s + M_k - 1$ **do**
       $t = x_m + x_{m+M_k}$
       $x_{m+M_k} = (x_m - x_{m+M_k})W_{2M_k}^{m-s+1/2}$
       $x_m = t$
     end for
     $s = s + 2M_k$
   end for
   $N = 2N$
end for
Output permutation:
$\ell = 0$
for $k = 1 \rightarrow M - 1$ **do**
   $s = M/2$
   while $\ell \geq s$ **do**
     $\ell = \ell - s$
     $s = s/2$
   end while
   $\ell = \ell + s$
   if $k < \ell$ **then**
     $t = x_k$
     $x_k = x_\ell$
     $x_\ell = t$
   end if
end for

---

where $\tilde{d} = (N + D - 3M + 1)/2 \bmod 2M$ due to the $2M$-periodicity of the modulation. On the synthesis side the transform is $\hat{\mathbf{x}} = \boldsymbol{\Phi}_{\mathrm{e-e}} \hat{\mathbf{X}}$ and again we extract the modified DFT basis:

$$
\begin{aligned}
\left[\boldsymbol{\Phi}_{\mathrm{e-e}}\right]_{n,k} &= \frac{1}{\sqrt{2M}} W_{2M}^{-nk} W_{2M}^{k(N+D-M)/2} \\
&= \frac{1}{\sqrt{2M}} W_{2M}^{-(n-(N+D-M)/2)k} \\
&= \frac{1}{\sqrt{2M}} W_{2M}^{-(n+1/2-(N+D-M+1)/2)k}.
\end{aligned}
$$

This time the shifting effects on the columns of $\boldsymbol{\Theta}_{2M}$ so that the even-stacked modulation can be computed as

$$
\hat{\mathbf{x}} = \mathrm{CircularShift}\left(\boldsymbol{\Theta}_{2M} \hat{\mathbf{X}}, d\right),
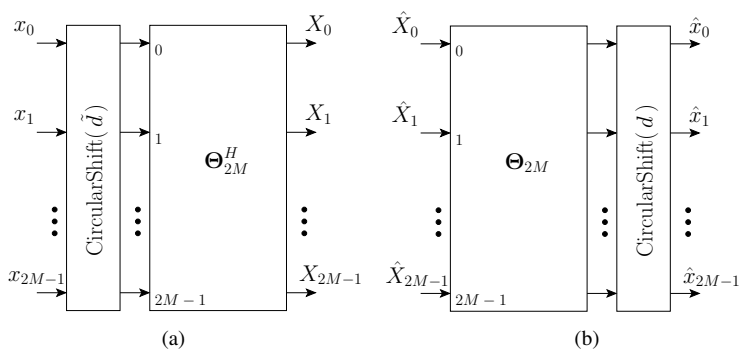$$

Figure 7.12: Alternative even-stacked EMFB modulation block implementation when M is a power of two and $N + D - M$ is odd. (a) Analysis $\tilde{\Phi}_{\text{e-e}}^{H}$. (b) Aynthesis $\Phi_{\text{e-e}}$.

where $d = (N+D-M+1)/2 \bmod 2M$. The flowgraphs of the modulation blocks are shown in Figure 7.12. This tweak then reduces the number of complex-valued multipliers by $2M$ at the cost of data shifting and non-standard FFT.

7.8.2 Modified FFT for Odd-Stacked EMFB

The above restrictions (M is a power of two and $N + D - M$ odd) hold also in the odd-stacked case. But now the initial matrix $\breve{\Theta}$ has also a frequency shift:

$$\left[\breve{\Theta}\right]_{n,k} = \frac{1}{\sqrt{M}} W_M^{-(n+1/2)(k+1/2)}.$$

When compared with the previous even-stacked case, this causes some complications to the algorithm under the guise of extra multiplications. Also some real-imaginary terms interchanging in necessary. Fortunately, these terms appear only in the first stage of the FFT butterfly computations.

When the above derivation is repeated for $\breve{\Theta}$ having odd-stacked center frequencies, we now encounter a multiplier $j(-1)^{k+1}$:

$$X_k = \sum_{n=0}^{M/2-1} x_n W_M^{(n+1/2)(k+1/2)}$$

$$= \sum_{n=0}^{M/2-1} x_n W_M^{(n+1/2)(k+1/2)} + \sum_{n=0}^{M/2-1} x_{n+M/2} W_M^{(n+M/2+1/2)(k+1/2)}$$

$$= \sum_{n=0}^{M/2-1} \Big(x_n + \underbrace{W_M^{(k+1/2)M/2}}_{j(-1)^{k+1}} x_{n+M/2}\Big) W_M^{(n+1/2)(k+1/2)}.$$

This term can then be implemented by interchanging the real and imaginary terms of $x_{n+M/2}$ and reversing the signs according to the parity of k.

143

Algorithm 7.2 Modified odd-stacked FFT algorithm to compute $\mathbf{X} = \breve{\boldsymbol{\Theta}}_M^H \mathbf{x}$.

Decimation-in-frequency:
$r = \log_2 M$
$N = 1$
for $k = r - 1 \to 0$ **do**
   $s = 0$
   $M_k = 2^k$
   for $\ell = 1 \to N$ **do**
     for $m = s \to s + M_k - 1$ **do**
       if $k = r - 1$ **then**
         $t = (x_m - j\, x_{m+M_k})W_{2M_k}^{(m-s)/2+1/4}$
         $x_{m+M_k} = (x_m + j\, x_{m+M_k})W_{2M_k}^{3(m-s)/2+3/4}$
       else
         $t = x_m + x_{m+M_k}$
         $x_{m+M_k} = (x_m - x_{m+M_k})W_{2M_k}^{m-s+1/2}$
       end if
       $x_m = t$
     end for
     $s = s + 2M_k$
   end for
   $N = 2N$
end for
Output permutation:
As in Algorithm 7.1

The next step is to gather separately the even and odd subsamples. This yields

$$
X_{2k} = \sum_{n=0}^{M/2-1} \left(x_n + j(-1)^{2k+1}x_{n+M/2}\right)W_M^{(n+1/2)(2k+1/2)}
$$

$$
= \sum_{n=0}^{M/2-1} \left((x_n - j\,x_{n+M/2})W_M^{(n/2+1/4)}\right)W_{M/2}^{(n+1/2)k} = \sum_{n=0}^{M/2-1} \breve{g}_{0,n}W_{M/2}^{(n+1/2)k}
$$

and

$$
X_{2k+1} = \sum_{n=0}^{M/2-1} \left(x_n + j(-1)^{2k+2}x_{n+M/2}\right)W_M^{(n+1/2)(2k+1+1/2)}
$$

$$
= \sum_{n=0}^{M/2-1} \left((x_n + j\,x_{n+M/2})W_M^{(3n/2+3/4)}\right)W_{M/2}^{(n+1/2)k} = \sum_{n=0}^{M/2-1} \breve{g}_{1,n}W_{M/2}^{(n+1/2)k}.
$$

In subsequence $\breve{g}_{0,n}$ we observe the extra twiddle factors $W_M^{(n/2+1/4)}$. However, the left-over multiplier $W_{M/2}^{(n+1/2)k}$ is identical with the $\boldsymbol{\Theta}_{M/2}^H$ transform matrix. As an example, a schematic flowgraph is shown in Figure 7.11 (b) for $M = 8$. That generalizes to modified odd-stacked FFT algorithm given in Algorithm 7.2. The FFT bin permutation is identical with Algorithm 7.1.

The final detail is to include the modulation phase term into the play. It is again reminded that the odd-stacked EMFB modulation has $4M$ periodicity which requires sign reversions that can again be embedded in the polyphase filters. (The derivation is skipped.) On the analysis side we have parameters $\tilde{r} = \lfloor \frac{(N+D-3M+1)/2}{2M} \rfloor$ and $\tilde{d} = (N + D - 3M + 1)/2 - 2\tilde{r}M$. They define the diagonal matrix $\tilde{\mathbf{S}} = \mathrm{diag}\{\tilde{s}_0, \tilde{s}_1, \ldots, \tilde{s}_{2M-1}\}$ where

$$\tilde{s}_k = \begin{cases} (-1)^{\tilde{r}} & k < 2M - \tilde{d} \\ -(-1)^{\tilde{r}} & k \geq 2M - \tilde{d}. \end{cases}$$

The odd-stacked EMFB matrix multiplication $\mathbf{X} = \tilde{\boldsymbol{\Phi}}_{\mathrm{e}}^H \mathbf{x}$ is thus alternatively expressed as

$$\mathbf{X} = \breve{\boldsymbol{\Theta}}_{2M}^H \, \mathrm{CircularShift}\big(\tilde{\mathbf{S}}\,\mathbf{x}, \tilde{d}\big).$$

On the synthesis side the sign reversing and the modulation sequence shift parameters are $r = \lfloor \frac{(N+D-M+1)/2}{2M} \rfloor$ and $d = (N + D - M + 1)/2 - 2rM$. The non-zero elements of the diagonal sign reversion matrix are

$$\big[\mathbf{S}\big]_{k,k} = s_k = \begin{cases} (-1)^r & k \geq d \\ -(-1)^r & k < d \end{cases}$$

and we may plug the modified odd-stacked FFT algorithm into the synthesis algorithm:

$$\hat{\mathbf{x}} = \boldsymbol{\Phi}_{\mathrm{e}} \, \hat{\mathbf{X}} = \mathbf{S} \, \mathrm{CircularShift}\big(\breve{\boldsymbol{\Theta}}_{2M} \, \hat{\mathbf{X}}, d\big).$$

A final remark regarding to the presented modified FFT algorithms is the computation of the inverse transform. One way to achieve this is to reverse the order of computation of the forward algorithm: 1) start with the bin permutation loop (its own inverse); 2) reverse the loop indices; 3) conjugate the twiddle factors; and 4) in the odd-stacked case the last lattice of the inverse stage restores the real-imaginary parts swapping done on the forward side.

7.9   EMFB Subsignal Computations with Goertzel-Algorithm

All MFB implementation algorithms discussed above operated with a full set of subchannels. Nevertheless, if there is a need to compute a small number of subsignals (with respect to M), the block-transform-based algorithms with polyphase filters are cubersome. Especially so, if the block sample phase is slightly advanced or delayed and, consequently, the polyphase filters contain intermediate data that needs to be flushed. One possible future application might be in the internet of things communications, where in some scenarios low-rate sensory information is transmitted in asynchronous manner, in single or few subcarriers of a subcarrier system. Another application might be in radar systems where the traced signal of interest is known to have a relatively narrow band.

For this purpose, a recursive per-subchannel biorthogonal EMFB algorithm is developed using the Goertzel-DFT for modulation.

7.9.1 Review on Goertzel-Algorithm

Goertzel-algorithm is usually described as a recursive filter structure to compute the kth coefficient ($k \neq 0$) of the M-point DFT [73]. The filter polynomial is

$$G_k(z) = \frac{1 - e^{-j2\pi k/M} z^{-1}}{1 - 2\cos\left(\frac{2\pi k}{M}\right) z^{-1} + z^{-2}} = \frac{1 + w(k)\, z^{-1}}{1 - c(k) z^{-1} + z^{-2}}$$

which is at the first glance more costly in terms of multiplications than the vector inner product with complex numbers. But the point in reducing the complexity is the real-valued coefficient of the feedback loop. The M-sample sequence is filtered by the denominator polynomial and only the last two samples are latched to the feedforward part. In other words, the kth analysis Goertzel-filter maps an M-length sequence into a single complex-valued subsample y_k using direct form II configuration. Depending on the implementation of the recursive filter, it might be necessary to pad the data with one extra zero sample.

For the sake of clarity we represent the basic Goertzel-algorithm here also as a difference equation. Let x_n be the M-length input sequence and we need to compute the forward DFT in kth bin. The intermediate sequence $y_k[n]$ is first initialized properly and then computation proceeds until we reach the Mth term:

$$\begin{aligned}
y[-1] &= 0 \\
y[0] &= x_0 \\
y[1] &= c(k)\, y[0] - y[-1] + x_1 \\
&\vdots \\
y[M-1] &= c(k)\, y[M-2] - y[M-3] + x_{M-1} \\
y[M] &= c(k)\, y[M-1] - y[M-2].
\end{aligned} \tag{7.10}$$

The output of the algorithm is then produced from the last two terms of $y_k[n]$:

$$y_k = \sum_{n=0}^{M-1} x_n W_M^{nk} = y[M] + w(k)\, y[M-1]. \tag{7.11}$$

Then we define an operator $\tilde{G}_{\gamma, P}\{\cdot\}$ for the computations given in the set of equations (7.10) combined with the RHS of (7.10). Parameter γ is the angle of the cosine and complex exponential terms in the transfer function. The other parameter P in the operator form of the Goertzel recursion refers to index when recursion $y_k[n]$ stops. The kth DFT bin then matches with $y_k = \tilde{G}_{2\pi k/M, M}\{(x_n)\}$ where the argument (x_n) is the input sequence.

The 0th Goertzel-filter reduces to a trivial multiplier-free feedback filter. This special case is omitted in the discussion below, but it should be noticed with the low-pass subfilter pair of the even-stacked EMFB. The filter coefficients are either purely real or imaginary-valued depending on the phase flip index. Thus $k = 0$ case must be treated separately.

The inverse filter generates an M-length sequence from the kth subband coefficient. The filtered data can be combined with other outputs if several inverse Goertzel-filters run in parallel. The trick to reduce complexity is to compute first the feedforward section. This results in a M-length sequence $(y_k,\, y_k w(k), 0, \ldots, 0)$. Then the computation is handled with one exponential rotation (implementable with three real multiplications and additions) plus the feedback loop complexity: $2M$ multiplications due to real-valued coefficient and complex-valued data.
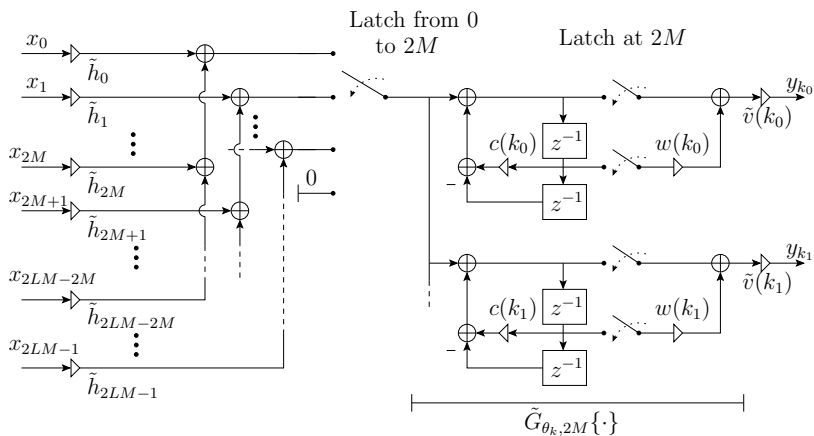
Figure 7.13: Analysis Goertzel-EMFB. Handling the modulation with the direct form II Goertzel-filters. The parallel-to-serial latch adds a trailing zero as the $2M$th (extra) sample if we initialize the delay elements of the recursive filter with zero-values.

The required set of difference equations on the synthesis side can be written as

$$
\begin{aligned}
\hat{x}_{-1} &= 0 \\
\hat{x}_0 &= \hat{y}_k \\
\hat{x}_1 &= c(k)\,\hat{x}_0 - \hat{x}_{-1} + w(k)\,\hat{y}_k \\
\hat{x}_2 &= c(k)\,\hat{x}_1 - \hat{x}_0 \\
&\vdots \\
\hat{x}_{M-1} &= c(k)\,\hat{x}_{M-2} - \hat{x}_{M-3}
\end{aligned}
$$

Using the operator notation the sequence resulting from the inverse transform of $\hat{y}_k$ is thus

$$
(\hat{x}_n)_{n=0}^{M-1} = G_{2\pi k/M,M}\{\hat{y}_k\}.
$$

7.9.2 Adapting Goertzel-Algorithm for Odd and Even-Stacked EMFB

The recursive EMFB-algorithm discussed below is a biorthogonal generalization of the paraunitary version presented in [9]. The task is now to modify the kth analysis side sum

$$
y_k = \sum_{n=0}^{N} x_n\, h[N-n]\, \exp\left[j(N - n - (N + D + M)/2)\theta_k\right]
$$

to make it suitable for $2M$-length Goertzel-DFT. We may work out both the even and odd-stacked cases simultaneously by setting

$$
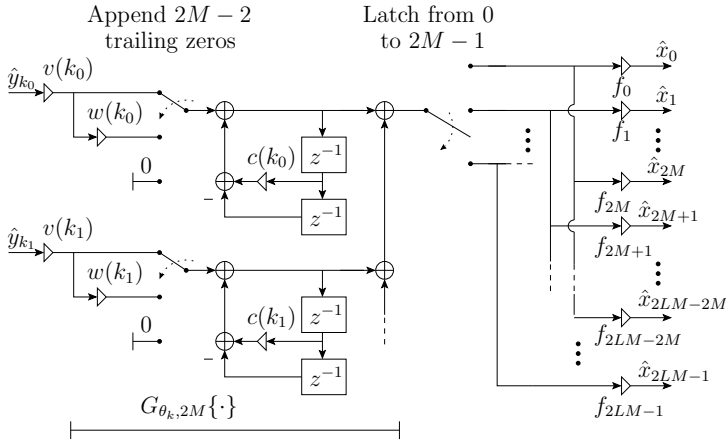\theta_k = \begin{cases} (k+1/2)\pi/M & \text{if odd-stacked} \\ k\pi/M & \text{if even-stacked.} \end{cases}
$$

147

Figure 7.14: Synthesis EMFB build around the direct form I Goertzel-filters.

(a)

(b)
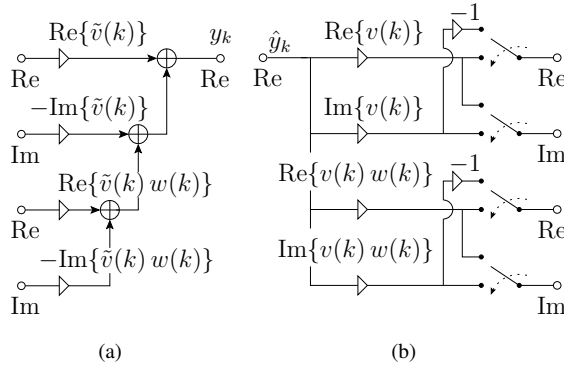
Figure 7.15: (a) Redundant multipliers removed for the critically sampled (real-valued subsignals) analysis side, odd-stacked EMFB. (b) Alternating phase flipping with reduced number of multipliers for the critically sampled even-stacked EMFB, synthesis side.

The exponential modulating sequences have non-DFT phase offset to match the PR requirements. The phase rotation multiplier can be pulled apart from the sum thus adding an extra coefficient $\tilde{v}(k)$ to Goertzel-algorithm:

$$
\begin{aligned}
y_k &= \sum_{n=0}^{N} x_n \, h[N-n] \, \exp\left[j(N-n-(N+D+M)/2)\theta_k\right] \\
&= \sum_{n=0}^{N} x_n \, h[N-n] \, \exp\left[-j(n+(-N+D+M)/2)\theta_k\right] \\
&= \underbrace{\exp\left(j(N-D-M)\theta_k/2\right)}_{\tilde{v}(k)} \sum_{n=0}^{N} x_n \, h[N-n] \, \exp\left[-j\,n\,\theta_k\right]
\end{aligned}
$$

Secondly, for derivation purposes we would like to operate with multiple of $2M$-length sequences. Then we denote by h_n the terms of reversed impulse response padded with trailing zeros:

$$(h_n)_{n=0}^{2LM-1} = (h[N], h[N-1], \ldots, h[0], \underbrace{0, 0, \ldots, 0}_{\#(2LM-N-1)}),$$

where $L = \lceil \frac{N+1}{2M} \rceil$. The zero coefficients can be omitted in the actual computations. Now the substitution of h_n allows to replace the summation upper limit and then split the summation index:

$$y_k = \tilde{v}(k) \sum_{n=0}^{2LM-1} x_n h_n \exp\left[-j\, n\, \theta_k\right]$$

$$= \tilde{v}(k) \sum_{n=0}^{2M-1} \sum_{\ell=0}^{L-1} x_{n+2\ell M} h_{n+2\ell M} \exp\left[-j\,(n+2\ell M)\theta_k\right]$$

The exponential term can be moved outside the inner sum once we compensate the $4M$-periodicity of the odd-stacked modulation. This is handled with $(\pm 1)^\ell$-term where the upper(lower) sign corresponds to even(odd)-stacked modulation. That is

$$y_k = \tilde{v}(k) \sum_{n=0}^{2M-1} \exp\left[-j\, n\, \theta_k\right] \sum_{\ell=0}^{L-1} x_{n+2\ell M} h_{n+2\ell M} (\pm 1)^\ell$$

$$= \tilde{v}(k) \sum_{n=0}^{2M-1} \exp\left[-j\, n\, \theta_k\right] \tilde{u}_n \tag{7.12}$$

where (u_n) is the sequence

$$(u_n)_{n=0}^{2M-1} = \Big(\sum_{\ell=0}^{L-1} x_{n+2\ell M} h_{n+2\ell M} (\pm 1)^\ell \Big)_{n=0}^{2M-1}.$$

The summation in (7.12) is almost compatible with the Goertzel-DFT algorithm. If we use coefficients $c(k) = 2\cos(\theta_k)$ and $w(k) = -\exp(-j\theta_k)$ in Goertzel filtering of sequence $(\tilde{u}_n)$ the result matches up to the sign:

$$\sum_{n=0}^{2M-1} \exp[-j\, n\, \theta_k]\tilde{u}_n = \begin{cases} \tilde{G}_{\theta_k, 2M}\{(\tilde{u}_n)\} & \text{if even-stacked} \\ -\tilde{G}_{\theta_k, 2M}\{(\tilde{u}_n)\} & \text{if odd-stacked.} \end{cases}$$

To further polish the sum representation we embed the sign reversions in the prototype filter coefficients $\tilde{h}_n = (\pm 1)^{\lfloor n/(2M) \rfloor + 1} h_n$ yielding

$$y_k = \tilde{v}(k)\, \tilde{G}_{\theta_k, 2M}\Big\{ \Big(\sum_{\ell=0}^{L-1} x_{n+2\ell M} \tilde{h}_{n+2\ell M} \Big) \Big\}. \tag{7.13}$$

Equation (7.13) describes the analysis Goertzel-EMFB for the kth subchannel and the flowgraph is shown in Figure 7.13. Data windowing can be shared by parallel Goertzel-filters marked with frequency indices k_0, k_1, etc. If the algorithm is operated in critically sampled

mode with real-valued subsignals, the phase rotation coefficients can be combined to remove unnecessary multiplications. The simplification described in Figure 7.15 (a) applies for odd-stacked EMFB.

The synthesis Goertzel-EMFB is a reverse procedure for the above as shown in Figure 7.14. Let S be the set of subband indices involved in reconstruction $S = \{k_0, k_1, \ldots k_p\}$. Then matching to these subbands we have the phase correction terms ($k \in S$)

$$v(k) = \exp\left(-j(N + D - M)\theta_k/2\right)$$

that multiply the subsignal data $\hat{y}_k$ before the bank of inverse Goertzel-algorithms

$$\left(u_{n,k}\right)_{n=0}^{2M-1} = G_{\theta_k, 2M}\{v(k)\hat{y}_k\}.$$

The output signals of the parallel Goertzel-filters are combined before the unfolding stage:

$$\left(\hat{u}_n\right)_{n=0}^{2M-1} = \left(\sum_{k_i \in S} u_{n,k_i}\right)_{n=0}^{2M-1}.$$

The incorporation of the sign reversions for the odd-stacked modulation is again handled with the prototype filter coefficients:

$$f_n = \begin{cases} h[n] & 0 \leq n \leq N, \text{ even-stacked} \\ (-1)^{\lfloor n/(2M)\rfloor + 1}h[n] & 0 \leq n \leq N, \text{ odd-stacked} \\ 0 & N < n \leq 2LM - 1. \end{cases}$$

The prototype filter is extended to multiple of $2M$ length so that we can write the unfolding of the modulation sequence simply as

$$\left(\hat{x}_n\right)_{n=0}^{2LM-1} = \left(\hat{u}_{(n \bmod 2M)}f_n\right)_{n=0}^{2LM-1}.$$

In typical signal reconstruction, the synthesis side needs an M-shift overlap-add buffer. Secondly, the even-stacked EMFB must also be equipped with the phase flipping operation (multiply by $-j$ every other $\hat{y}_k[n]$) if the algorithm is applied analogously with the M-decimated PR system. Again we can snip extra multiplications in the critically sampled case as demonstrated in Figure 7.15 (b). (This time with even-stacking including the phase flip.)

7.9.3 Slowing Down Feedback Loop of Goertzel-EMFB

The periodic properties of the exponential modulation allow us to shorten the $2M$-length sequence generated with the feedback loop by half. This has two advantages: The high-rate recursive structure of the Goertzel-EMFB may apply a lower clock rate and $2M$ real multiplications (complex-valued data) per branch are traded into less complex operations. Depending on the parity of k, the signs of the folded sequences must be reversed. Furthermore, the odd-stacked modulation requires the interchanging of the real and imaginary parts to maintain correct modulation waveforms. All the above is contained in the folding coefficient definition

$$\tilde{r} = \begin{cases} (-1)^k & \text{even-stacked} \\ j(-1)^k & \text{odd-stacked}. \end{cases}$$
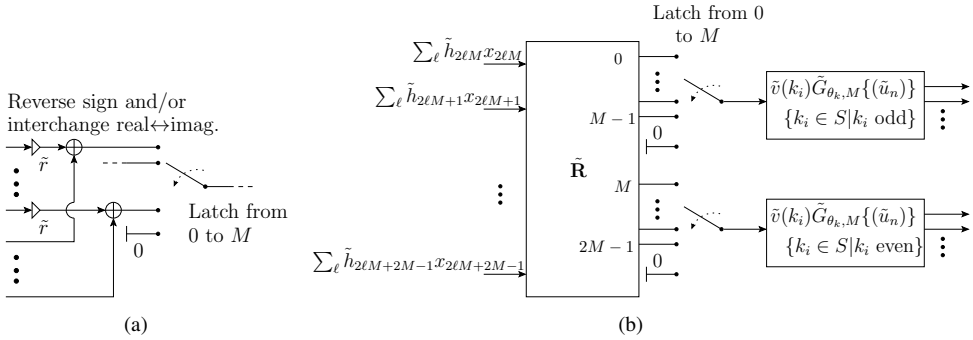
Figure 7.16: (a) Trading $2M$ real multiplications of the analysis Goertzel-EMFB into less complex arithmetic operations. (b) If the subfilter set contains a mix of even and odd indices, the algorithm has to branch due to sign differences.
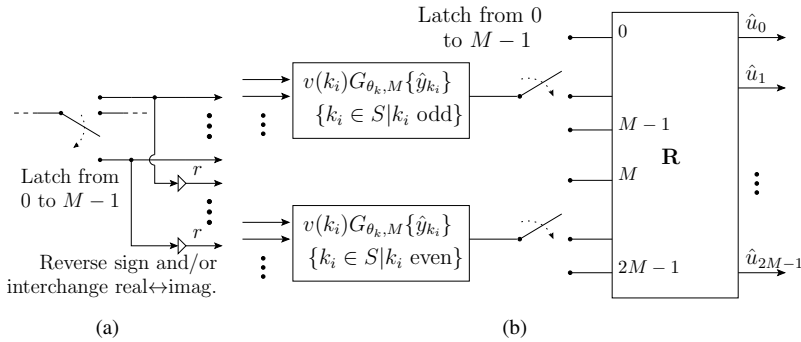

Figure 7.17: (a) Goertzel-EMFB synthesis side unfolding of the M-length sequence to double. (b) Handling the set of odd and even k_i in separate blocks and combining them with $\mathbf{R}$.

In Figure 7.16 (a) shown how the coefficients are applied to concatenate the input sequence of analysis Goertzel-EMFB by half.

If the subfilter index set contains both even and odd numbers then the algorithm has to branch as shown in Figure 7.16 (b). Matrix $\tilde{\mathbf{R}}$ maps then the input data into two M-length sequences when it is defined as

$$
\tilde{\mathbf{R}} = \begin{cases} \begin{bmatrix} -j\,\mathbf{I} & \mathbf{I} \\ j\,\mathbf{I} & \mathbf{I} \end{bmatrix} & \text{if odd-stacked} \\[2em] \begin{bmatrix} -\mathbf{I} & \mathbf{I} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} & \text{if even-stacked.} \end{cases}
$$

The upper (lower) matrix block corresponds to odd (even) subband indices.

On the synthesis side the coefficient applied in unfolding can be reused: $r = \tilde{r}$. The first unfolding operation replicates the M-length sequence generated with inverse Goertzel-DFT to double length and the other half is "multiplied" with r. This is schemed in Figure 7.17 (a). The

issue of odd and even k_i must be addressed also here and now the mapping can be written as

$$\mathbf{R} = \begin{cases} \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ -j\,\mathbf{I} & j\,\mathbf{I} \end{bmatrix} & \text{if odd-stacked} \\[1em] \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{bmatrix} & \text{if even-stacked.} \end{cases}$$

In this case the left (right) subblock of $\mathbf{R}$ corresponds with the odd (even) values of k_i. The unfolding system without the prototype filter part is shown in Figure 7.17 (b).

7.10   Computational Complexity of EMFB Algorithms

In MFB algorithm selection, the required number of arithmetic operations is one measure of merit. Often there are other practical issues like the availability of off-the-self block transform algorithms and the quality or efficiency of those implementations. In addition, the block transforms tailored for integer arithmetic (e.g., DCTs used in image processing [76]) are quite different transform realizations versus general purpose algorithms.

One complicating issue is that in our benchmarking we have a mixture of complex and real-valued block transforms. The computing architecture might have a support for complex arithmetic or bit reverse indexing available, which may tip the scales unexpectedly. Therefore we uncouple ourselves from the real-world implementation platforms by measuring the complexity with real-valued additions and multiplications. The decimation factor M is restricted to be a power of 2.

The number of arithmetic operations for M-point DCT-II, DCT-IV, and split-radix fast Fourier transform (SRFFT) are given in Table 7.3. The formulae contain an assumption that each complex-valued multiplier is implemented with three plain additions and multiplications [57]. The sine-transforms corresponding to the two DCTs have an identical complexity figure. The SRFFT is used with 2xOS EMFBs as well as a sub-component of the DCT algorithms.

The modified FFT for even-stacked EMFB has the same complexity as radix-2 FFT [73]. In terms of complex-valued multiplications (additions), the operation count for M-bin radix-2 FFT is $\mu_{\mathrm{c}} = (M/2)\log_2(M)$ ($\alpha_{\mathrm{c}} = M\log_2(M)$). These numerics ignore the small performance gain of multiplierless small FFTs. When measured with real-valued operations the matching complexity formulae are $\mu_{\mathrm{r}} = (3M/2)\log_2(M)$ and $\alpha_{\mathrm{r}} = 2M\log_2(M) + (3M/2)\log_2(M) = (7M/2)\log_2(M)$.

The prototype filter portion of the complexity is omitted in this evaluation since it only contributes a similar constant term. Secondly, we assume that all EMFBs are operated in CS mode. The algorithm identification number below matches with the entries in Table 7.3.

1. Figure 7.2: Odd-stacked EMFB with $2M$-bin SRFFT and $4M$ exponential multipliers (real-valued subsignals are taken into account)

2. Figure 7.4: Odd-stacked CS EMFB with Type-IV DCT and DST, the number of additions includes the cross-connection matrix and the analysis side folding with $\mathbf{J}_{\mathrm{c}}^T$ and $\mathbf{J}_{\mathrm{s}}^T$

3. Algorithm 7.2: Odd-stacked EMFB using the modified radix-2 FFT. The modulation requires $2M$-bin algorithm and, additionally, M complex-valued coefficients during the first iteration step.

Table 7.3: Arithmetic complexity of various EMFB algorithms (excluding the prototype filter).

Transform / Id. number EMFB	the number of real multiplications	the number of real additions
SRFFT	$M(\log_2(M) - 3) + 4$	$3M(\log_2(M) - 1) + 4$
radix-2 FFT	$(3M/2)\log_2(M)$	$(7M/2)\log_2(M)$
DCT-II	$(M/2)\log_2 M$	$(3M/2)\log_2 M - M + 1$
DCT-IV	$(M/2)\log_2(M) + M$	$(3M/2)\log_2(M)$
Biorthogonal Odd-Stacked EMFB Modulation		
Algorithm #1	$2M(\log_2(M) + 3) + 4$	$2M(3\log_2(M) + 4) + 4$
Algorithm #2	$M(\log_2(M) + 3)$	$M(3\log_2(M) + 7)$
Algorithm #3	$3M(\log_2(M) + 2)$	$M(7\log_2(M) + 10)$
Biorthogonal Even-Stacked EMFB Modulation		
Algorithm #4	$2M\log_2(M) + 4$	$2M(3\log_2(M) + 1) + 4$
Algorithm #5	$M(\log_2(M) + 1) + 2$	$M(3\log_2(M) + 5) - 1$
Algorithm #6	$3M(\log_2(M) + 1)$	$7M(\log_2(M) + 1)$

4. Figure 7.5: Even-stacked EMFB with $2M$-bin SRFFT and $2M$ exponential multipliers (real-valued subsignals are taken into account)

5. Figure 7.10: Even-stacked CS EMFB with Type-II DCT and DST, the number of additions includes the cross-connection matrix and the analysis side folding with $\mathbf{J}_c$ and $\mathbf{J}_s$

6. Algorithm 7.1: Odd-stacked EMFB using the modified radix-2 FFT with $2M$ bins.

In these complexity evaluations the $M\log_2 M$-term dominates the number of multiplications and additions. The algorithm ranking order (for critically sampled systems) is thus: 1) DCT/DST-based algorithms; 2) EMFBs using split-radix FFT; and 3) modified radix-2 FFTs. The complexity advantage of the SRFFT over the radix-2 version raises a question: Is it possible to modify the SRFFT algorithm to accommodate directly the EMFB-type of modulation?

These estimates of arithmetic complexity provide also some clue when it's advantageous to choose the Goertzel-EMFB over the block-transform-based algorithm. For instance, the 2xOS odd-stacked EMFB modulation with $M = 256$ requires 5636 multiplications. In the Goertzel-algorithm the feedback loop coefficients are real-valued, which count as two multipliers with complex-valued data. If there are I subsignals, we use the slower (the number of iterations is M) Goertzel-EMFB, and include the two complex multipliers, then the multiplicative complexity is $(2M+6)I$. When $I = 10$, the algorithm costs 5180 multiplications. In this particular case, selecting a larger number of subchannels starts to favor the use of block transforms in computation. However, it is worth mentioning that the Goertzel-algorithm is quite compact. The feedback term and intermediate data fit into, say, the registers of a signal processor.

In spite of the algorithm "ranking" above, that is based on the arithmetic figures of merit, their referential nature is still stressed. Modern multicore processors may process several threads in parallel, which makes code optimization and thus algorithm evaluation much more difficult. It is also possible that the memory access becomes the bottleneck of the algorithm and some extra arithmetic operations for the idle processor actually improves the throughput, if it leads to a smaller memory footprint.

7.11 Yet Another Modulation Block for CS EMFBs

Partly motivated from the arithmetic complexity comparison of the CS EMFB algorithms, where the EMFB with SRFFT outperformed the Modified radix-2 FFT-based algorithm. However, we may find a better use for the orthogonal matrices Θ and $\check{\Theta}$ encountered in Section 7.8. Since the transform matrices are complex-valued and the subsignals in CS mode purely real-valued, there is a nagging thought that these modulation blocks are not fully utilized. In the algorithm proposed in this section, we combine two consecutive subsignal vectors, say, $\mathbf{X}[2m]$ and $\mathbf{X}[2m+1]$ in a suitable manner and then process them simultaneously in the modulation block. The advantage of the algorithms discussed in the sequel is the reduction of the number of multipliers used by the FFTs by half. This comes with the cost of "cheaper" elementary operations and, in addition, increases the analysis-synthesis reconstruction delay due to the buffering. As mentioned above, the signal processor may handle these computations at her leisure, if the intermediate data swapping from the memory can be reduced.

The $2M \times 2M$ matrix Θ corresponding to the modified FFT for the even-stacked EMFB has the following elements $[\Theta]_{n,k} = (1/\sqrt{2M})W_{2M}^{-(n+1/2)k}$. After some visual waveform observation, one might find a lead that helps to construct the algorithm. Namely, for each matrix column k we have a conjugate symmetry property

$$[\Theta]_{n,k} = [\Theta^*]_{2M-1-n,k},$$

where $n = 0, 1, \ldots, M - 1$. When this connection is applied in, the author happily acknowledges, somewhat intuitive manner, the trial and error methodology provides the flowgraph shown in Figure 7.18 (b). The task at hand is the verification of the structure. That is, two consecutive subsamples $X_k[2m] = a$ and $X_k[2m+1] = b$ combined at the input are observed as separable vectors $a\,\boldsymbol{\theta}_k$ and $j\,b\,\boldsymbol{\theta}_k$ at the output switching lines, where $\boldsymbol{\theta}_k$ is the kth column of Θ. The parameter d, as well as $\tilde{d}$ for the analysis side, for the circular shifting blocks were discussed in Section 7.8.

The first step is to simplify the product of cross-connection and modulation matrices:

$$\acute{\Theta} = \frac{\sqrt{2M}}{2} \begin{bmatrix} \mathbf{I}_M & -\mathbf{J}_M \\ \mathbf{J}_M & \mathbf{I}_M \end{bmatrix} \Theta.$$

The constant scaling term $\sqrt{2M}/2$ is included for the sake of the derivation below. Element-wise, the product yields

$$[\acute{\Theta}]_{n,k} = \sqrt{2M}/2\big([\Theta]_{n,k} - [\Theta]_{2M-1-n,k}\big)$$

$$[\acute{\Theta}]_{n+M,k} = \sqrt{2M}/2\big([\Theta]_{M-1-n,k} + [\Theta]_{n+M,k}\big)$$

$$[\acute{\Theta}]_{n,k+M} = \sqrt{2M}/2\big([\Theta]_{n,k+M} - [\Theta]_{2M-1-n,k+M}\big)$$

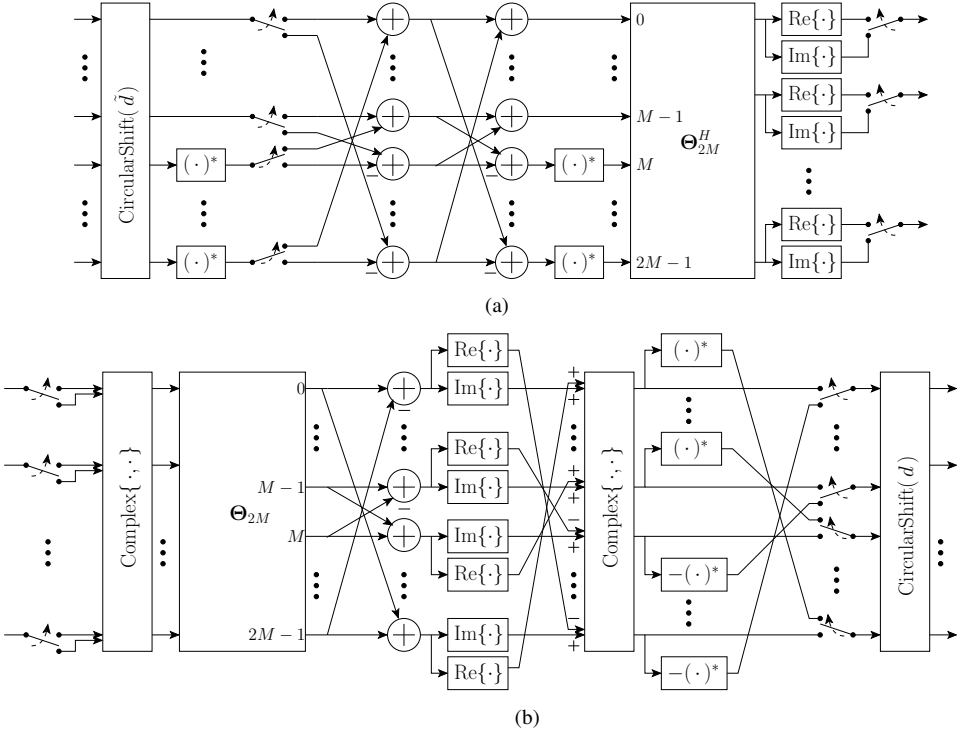$$[\acute{\Theta}]_{n+M,k+M} = \sqrt{2M}/2\big([\Theta]_{M-1-n,k+M} + [\Theta]_{n+M,k+M}\big),$$

Figure 7.18: Even-stacked EMFB modulation blocks for simultaneous processing of CS sub-signals $\mathbf{X}[2m]$ and $\mathbf{X}[2m+1]$. (a) Analysis. (b) Synthesis.

where $n, k = 0, 1, \ldots, M-1$. In addition, we may simplify the expression with respect to k and substitute the roots of unity terms:

$$
\begin{aligned}
\left[\acute{\boldsymbol{\Theta}}\right]_{n,k} &= \left(W_{2M}^{-(n+1/2)k} - W_{2M}^{-(2M-1-n+1/2)k}\right)/2 \\
\left[\acute{\boldsymbol{\Theta}}\right]_{n+M,k} &= \left(W_{2M}^{-(M-1-n+1/2)k} + W_{2M}^{-(n+M+1/2)k}\right)/2,
\end{aligned}
$$

where $n = 0, 1, \ldots, M-1$. The matrix elements can then be expressed in terms of sines and cosines:

$$
\begin{aligned}
\left[\acute{\boldsymbol{\Theta}}\right]_{n,k} &= \left(W_{2M}^{-(n+1/2)k} - \underbrace{W_{2M}^{-2Mk}}_{=1} W_{2M}^{(n+1/2)k}\right)/2 \\
&= j \sin\left(2\pi(n+1/2)k/(2M)\right) \\
\left[\acute{\boldsymbol{\Theta}}\right]_{n+M,k} &= \left(\underbrace{W_{2M}^{-Mk}}_{=(-1)^k} W_{2M}^{(n+1/2)k} + W_{2M}^{-Mk} W_{2M}^{-(n+1/2)k}\right)/2 \\
&= (-1)^k \cos\left(2\pi(n+1/2)k/(2M)\right).
\end{aligned}
$$

Let us then proceed with input $X_k[2m] = a$ and $X_k[2m+1] = b$ $(a, b \in \mathbb{R})$ to find out what the system actually does. The first breakpoint is after the cross-connection lattice, where $\mathbf{t}_0$

and $\mathbf{s}_0$ are M-length temporary variables

$$\left[\begin{array}{c} \mathbf{t}_0 \\ \mathbf{s}_0 \end{array} \right] = \acute{\mathbf{\Theta}} \big[\underbrace{0\; 0 \cdots 0}_{\#(k)}\, (a + j\,b)\; 0 \cdots 0 \big]^T,$$

whose elements are

$$\big[\mathbf{t}_0\big]_n = (-b + j\,a) \sin\big(2\pi(n + 1/2)k/(2M)\big)$$
$$\big[\mathbf{s}_0\big]_n = (a + j\,b)(-1)^k \cos\big(2\pi(n + 1/2)k/(2M)\big).$$

Further on, the real and imaginary parts are selected to form a second pair of complex-valued temporary vectors $\mathbf{t}_1$ and $\mathbf{s}_1$:

$$\begin{aligned} \big[\mathbf{t}_1\big]_n &= \mathrm{Re}\big\{\big[\mathbf{s}_0\big]_{M-1-n}\big\} + j\,\mathrm{Im}\big\{\big[\mathbf{t}_0\big]_n\big\} \\ &= a\big((-1)^k \cos\big(2\pi(M - 1 - n + 1/2)k/(2M)\big) + j\,\sin\big(2\pi(n + 1/2)k/(2M)\big)\big) \\ \big[\mathbf{s}_1\big]_n &= -\mathrm{Re}\big\{\big[\mathbf{t}_0\big]_{M-1-n}\big\} + j\,\mathrm{Im}\big\{\big[\mathbf{s}_0\big]_n\big\} \\ &= b\big(\sin\big(2\pi(M - 1 - n + 1/2)k/(2M)\big) + j(-1)^k \cos\big(2\pi(n + 1/2)k/(2M)\big)\big). \end{aligned}$$

Thus we have separated two M-length complex-valued vectors scaled with scalars a and b. It remains to show that the extension to $2M$-length results the desired output. This requires the substitution of certain identities and equations listed below:

$$\begin{aligned} \cos(\alpha) &= \cos(-\alpha) \\ \sin(\alpha) &= -\sin(-\alpha) \\ \cos\big(2\pi(n + M + 1/2)k/(2M)\big) &= \cos\big(2\pi(n + 1/2)k/(2M) + k\pi\big) \\ &= \cdots = (-1)^k \cos\big(2\pi(n + 1/2)k/(2M)\big) \\ \sin\big(2\pi(M - 1 - n + 1/2)k/(2M)\big) &= \sin\big(2\pi(M - 1 - n + 1/2)k/(2M) - 2\pi k\big) \\ &= \cdots = -\sin\big(2\pi(n + M + 1/2)k/(2M)\big) \end{aligned}$$

The temporary output vector $\mathbf{t}_2$ (before the circular shifting) matching the even subsignal time indices is

$$\begin{aligned} \big[\mathbf{t}_2\big]_n &= \big[\mathbf{t}_1\big]_n \\ &= a\big((-1)^k(-1)^k \cos\big(-2\pi(n + 1/2)k/(2M)\big) \\ &\quad + j\,\sin\big(2\pi(n + 1/2)k/(2M)\big)\big) = aW_{2M}^{-(n+1/2)k} \end{aligned}$$

$$\begin{aligned} \big[\mathbf{t}_2\big]_{n+M} &= \big[\mathbf{t}_1^*\big]_{M-1-n} \\ &= a\big((-1)^k \cos\big(2\pi(n + 1/2)k/(2M)\big) \\ &\quad - j\,\sin\big(2\pi(M - 1 - n + 1/2)k/(2M)\big)\big) \\ &= a\big(\cos\big(2\pi(n + M + 1/2)k/(2M)\big) \\ &\quad + j\,\sin\big(2\pi(n + M + 1/2)k/(2M)\big)\big) = aW_{2M}^{-(n+M+1/2)k}, \end{aligned}$$

where $n = 0, 1, \ldots, M-1$. Similarly, we may produce the case when the subsignal time index is odd. A kth subchannel input yields vector $\mathbf{s}_2$ having elements

$$
\begin{aligned}
\left[\mathbf{s}_2\right]_n &= -\left[\mathbf{s}_1^*\right]_{M-1-n} \\
&= -b\big(\sin\left(2\pi(n+1/2)k/(2M)\right) \\
&\quad - j(-1)^k \cos\left(2\pi(M-1-n+1/2)k/(2M)\right)\big) \\
&= -b(-j)j\big(\sin\left(2\pi(n+1/2)k/(2M)\right) \\
&\quad - j(-1)^{2k} \cos\left(2\pi(-n-1/2)k/(2M)\right)\big) \\
&= j\,b\big(j \sin\left(2\pi(n+1/2)k/(2M)\right) \\
&\quad + \cos\left(2\pi(n+1/2)k/(2M)\right)\big) = j\,bW_{2M}^{-(n+1/2)k}
\end{aligned}
$$

$$
\begin{aligned}
\left[\mathbf{s}_2\right]_{n+M} &= \left[\mathbf{s}_1\right]_n \\
&= b\big(\sin\left(2\pi(M-1-n+1/2)k/(2M)\right) \\
&\quad + j(-1)^k \cos\left(2\pi(n+1/2)k/(2M)\right)\big) \\
&= b(-j)j\big(-\sin\left(2\pi(n+M+1/2)k/(2M)\right) \\
&\quad + j \cos\left(2\pi(n+M+1/2)k/(2M)\right)\big) \\
&= j\,b\big(j \sin\left(2\pi(n+M+1/2)k/(2M)\right) \\
&\quad + j(-j) \cos\left(2\pi(n+M+1/2)k/(2M)\right)\big) = j\,bW_{2M}^{-(n+M+1/2)k}.
\end{aligned}
$$

By combining the above M-length vectors, we have just the desired alternating output vectors for even-stacked EMFB: $\mathbf{t}_2 = a\boldsymbol{\theta}_k$ and $\mathbf{s}_2 = j\,b\boldsymbol{\theta}_k$. Naturally, in the actual EMFB processing the modulation block generates a linear combination of weighted $\boldsymbol{\theta}_k$ vectors, aligns the correct modulation phase with circular shifting and then passes the turn to the polyphase filters.

On the grounds of the other analysis-synthesis systems discussed in this chapter, we should expect an analogical flowgraph for the analysis side, that "mirrors" the algorithm presented in Figure 7.18 (b). Bluntly, the modulation block in Figure 7.18 (a) does just that. The polyphase filters generate complex-valued input vectors for the system. Then the sequence of operations is: 1) circular shifting and conjugating the lower half of the data vector; 2) buffering two consecutive data vectors and combining them with a cross-connection lattice; 3) a second lattice and conjugation of the lower half; and 4) forward transform matrix and alternating output subsample selection, i.e., the last stage in computing $\mathbf{X}[2m]$ and $\mathbf{X}[2m+1]$. The algorithm verification is omitted in this thesis.

Furthermore, the same kind of combined processing can be adopted to the CS odd-stacked EMFB algorithm. The corresponding modulation matrix (cf. Section 7.8) is defined as $\left[\breve{\boldsymbol{\Theta}}\right]_{n,k} = (1/\sqrt{2M})W_{2M}^{-(n+1/2)(k+1/2)}$. and its elements have a conjugate symmetry property

$$
\left[\breve{\boldsymbol{\Theta}}\right]_{n,k} = -\left[\breve{\boldsymbol{\Theta}}^*\right]_{2M-1-n,k},
$$

where $n = 0, 1, \ldots, M-1$. The second difference is the fixed nature of the modulation block. If we denote with $\breve{\boldsymbol{\theta}}_k$ the columns of the modulation matrix and do a similar step-by-step verification, the result of two consecutive subsamples $X_k[2m] = a$ and $X_k[2m+1] = b$ must be separable vectors: $a\,\breve{\boldsymbol{\theta}}_k$ and $b\,\breve{\boldsymbol{\theta}}_k$. All that is left, is the author's request for leap of faith from the reader: the flowgraph in Figure 7.19 (b) succeeds is such operation. In addition, the analysis side counterpart is shown in Figure 7.19 (a).
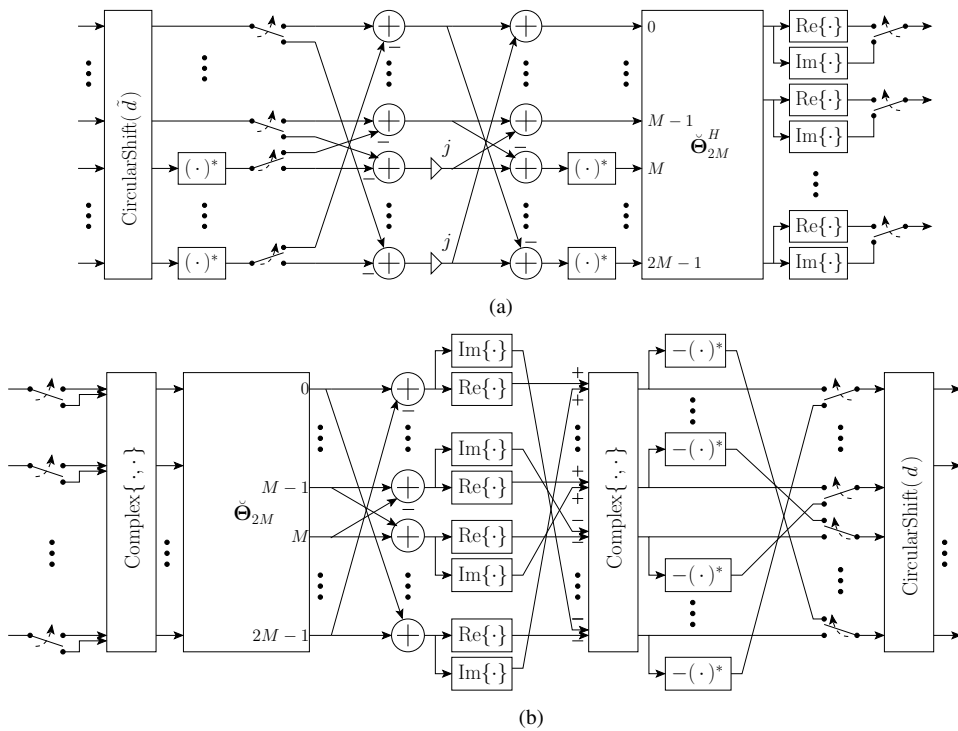
Figure 7.19: Odd-stacked EMFB modulation blocks for simultaneous processing of CS sub-signals $\mathbf{X}[2m]$ and $\mathbf{X}[2m+1]$. (a) Analysis. (b) Synthesis.

Chapter 8

Chapter **8**

Fast ELT Algorithms for Paraunitary CMFBs

8.1 Inherently PR Structure Using Orthogonal Rotations

Fast ELT algorithm was presented by H.S. Malvar as an efficient implementation structure for odd-stacked paraunitary CMFBs. When accompanied with SMFB, the algorithm can be used as a building block for complex transmultiplexer [5, 11] or in subband-based equalization [3, 6]. The flowgraph of the forward and inverse transform algorithms are shown in Figure 8.1. The modulation sequence generation is the responsibility of Type-IV DCT and the prototype filter part consists of $\mathbf{D}_m$ matrices in cascade separated with delays. The elements of $\mathbf{D}_m$ are cosines and sines of ELT angles $\theta_{n,m}$ that are located on the diagonals

$$\mathbf{D}_m = \begin{bmatrix} -\mathbf{C}_m & \mathbf{S}_m\mathbf{J} \\ \mathbf{J}\mathbf{S}_m & \mathbf{J}\mathbf{C}_m\mathbf{J} \end{bmatrix}$$

where $\mathbf{C}_m = \operatorname{diag}\{\cos(\theta_{0,m}), \cos(\theta_{1,m}), \ldots, \cos(\theta_{M/2-1,m})\}$ and $\mathbf{S}_m = \operatorname{diag}\{\sin(\theta_{0,m}), \sin(\theta_{1,m}), \ldots, \sin(\theta_{M/2-1,m})\}$. Due to the X-shaped diagonal form, $\mathbf{D}_m$ has a poetic name: the butterfly matrix. This regular construction of the fast ELT algorithm dictates the basis function (subfilter) length; it is $2KM$ where the number of butterfly matrices $K \in \mathbb{N}$ and M is an even integer.

There are several reasons why the orthogonal rotations are appealing for implementation:

- The orthogonal rotations are normalized, i.e., each butterfly matrix preserves the vector norm: $||\mathbf{D}_m\mathbf{x}||_2^2 = \mathbf{x}^T\mathbf{D}_m^T\mathbf{D}_m\mathbf{x} = \mathbf{x}^T\mathbf{x} = ||\mathbf{x}||_2^2$.

- The direct computation of one orthogonal rotation (four multiplications, two additions) can be replaced with three step lifting factoring. One possible version is

$$
\begin{aligned}
\mathbf{D}(\theta) &= \begin{bmatrix} -\cos(\theta) & \sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 \\ \cot(\theta/2) & 1 \end{bmatrix} \begin{bmatrix} 1 & -\sin(\theta) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \cot(\theta/2) & -1 \end{bmatrix}
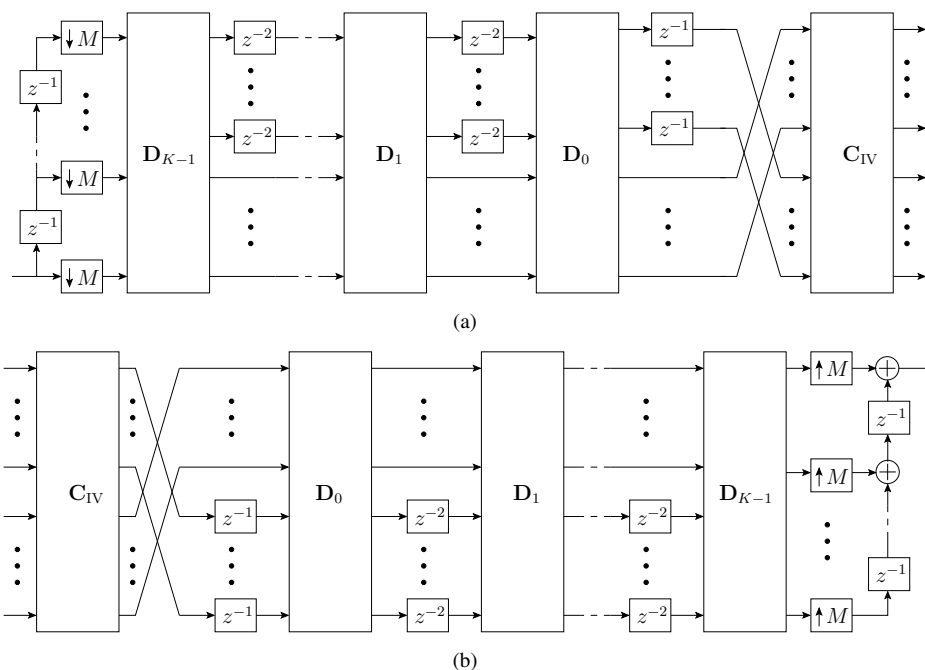\end{aligned}
\tag{8.1}
$$

Figure 8.1: Fast ELT Algorithm: (a) Analysis. (b) Synthesis.

This factoring trades one multiplication to addition operation. There are also some common sense issues how to make these rotations "well behaved" from the implementation point of view. We return to these considerations in Section 8.7.

- If we combine an (integer) quantization operator with the lifting multipliers, it is straight-forward to create an integer fast ELT (IntELT) algorithm with perfect-reconstruction property. Naturally, it is assumed that the applied DCT-IV satisfies the same integer-to-integer PR condition [76].

- Rotations are common operations in various signal processing methods and there are specialized realizations for these creatures; CORDIC [102] and SOPOT [25].

- If the implementation platform is utilizing floating point arithmetic, yet another trick is to scale the butterfly cascade so that some multipliers become plus or minus unity. This discussion is also postponed until Section 8.7.

The other consequence from orthogonal-rotation-based construction is to provide alternative methods for the prototype filter design. The inherently PR structure implies unconstrained optimization algorithms if the butterfly angles $\theta_{n,m}$ are directly applied as problem variables. However, the formulation produces non-quadratic objective function with numerous local minima and off-the-shelf solvers have problems in finding the global minimum as K increases.

The purpose of this fast ELT review is to pave way into the derivation of the even-stacked version of the algorithm. The even-stacked fast ELT enjoys the advantages of orthogonal rotations and may utilize the same ELT angles as the odd-stacked version. But the main point is that
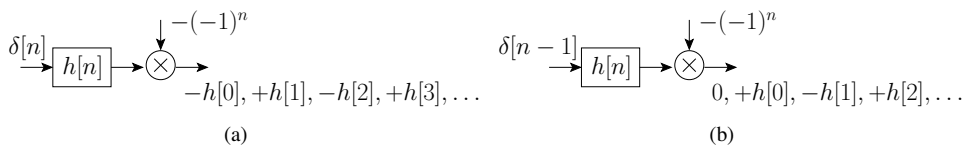
Figure 8.2: Preparation for even-stacked fast ELT algorithm derivation; we require a notational convention for a time varying impulse response. (a) Sign reversion of even terms. (b) Sign reversion of odd terms.

the linear phase filters are generated by even-stacked modulation of the symmetric prototype filter.

The starting point is the odd-stacked fast ELT algorithm. Below we introduce few modifications to this structure that are just elementary operations. We claim then that the modified subband system has the desired even-stacked property. Proving this is more tricky issue due to the recursive construction of the algorithm. Thus our reasoning proceeds in steps outlined below:

- First we introduce the flowgraph of the even-stacked fast ELT algorithm to present our goal.

- Then we provide a formula (expressed with nested for-loops) that can be used to map a $2KM$-length PR prototype filter $h[n]$ into ELT angles. Partly the reason to do so is to complete the matter in a book, which has greatly inspired this thesis [57], where these equations are omitted. Another point is that it provides the polyphase domain equality between the cascaded butterfly matrices and matrix equations expressed in terms of $h[n]$. We take the existence of this connection as granted on the basis of Malvar's work.

- The polyphase matrix mayhem starts with the verification of the modulation phase for the odd-stacked subfilters: this involves the elementwise polyphase matrix multiplication with the DCT-IV matrix. This step is just to ensure that the applied definition is correctly phrased as we need it below. The emphasis is on the synthesis stage.

- Then we define the modified butterfly cascade for the even-stacked case. At this stage it is unknown how the even-stacked polyphase matrix is expressed in terms of the prototype filter coefficients.

- We make an assumption where the prototype filter polyphase matrices of the even and odd-stacked cases are connected. The connection is proved with mathematical induction for arbitrary K.

- From the previous step we have an access to the elements of the even-stacked polyphase matrix thorough the connection proved above. If we further consider the alternating cosine and sine modulation separately, the polyphase equations for the synthesis filters are observed.

- All subcomponents of the inverse fast ELT algorithms are reversible and the corresponding forward algorithm is found trivially. Then considering the inverse algorithm in detail is sufficient for our purposes.
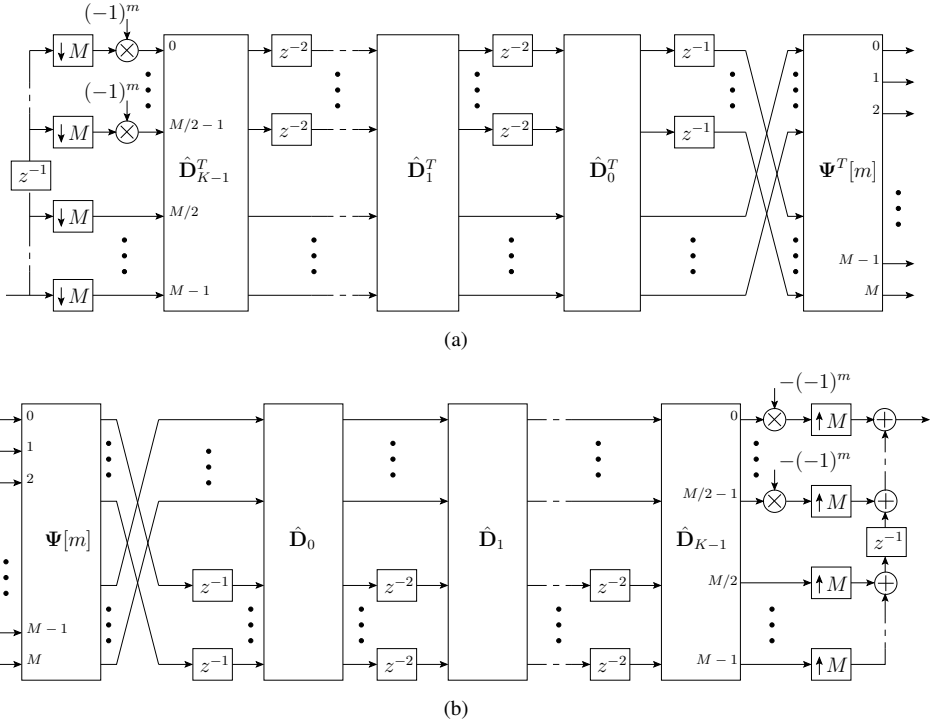
(a)

(b)

Figure 8.3: Fast ELT Algorithm in even-stacked configuration: (a) Analysis. (b) Synthesis.

One algorithm feature needs notational clarification that is convenient to address in advance. Let us consider the simple time varying system shown in Figure 8.2. There we have a causal FIR filter $h[n]$ whose output is multiplied with a alternating sequence of ± 1s. When the excitation is not delayed we get

$$
\begin{aligned}
\mathcal{Z}\{\delta[n] * \left(-(-1)^n h[n]\right)\} &= -h[0] + h[1]z^{-1} - h[2]z^{-2} + \dots \\
&= -\left(h[0] + h[1](-z)^{-1} + h[2]\left((-z)^2\right)^{-1} + h[3](-z)^{-3}\right) = -H(-z).
\end{aligned}
$$

Similarly, one sample delay in excitation produces the impulse

$$
\begin{aligned}
\mathcal{Z}\{\delta[n-1] * \left(-(-1)^n h[n]\right)\} &= 0 + h[0]z^{-1} - h[1]z^{-2} + h[2]z^{-3} - \dots \\
&= z^{-1}H(-z).
\end{aligned}
$$

The pattern repeats: $\mathcal{Z}\{\delta[n-2] * \left(-(-1)^n h[n]\right)\} = -z^{-2}H(-z)$ and $\mathcal{Z}\{\delta[n-3] * \left(-(-1)^n h[n]\right)\} = z^{-3}H(-z)$. The conclusion is that when the system impulse response of the described system is divided in two phases (n odd or even), the z-transform domain equations must include this sign reversion. Below we emphasize the application of this property with an "operator" $\mathcal{M}\{\cdot\}$. Otherwise if we just plug the sign reversions (using the Stetson-Harrison method) into the polyphase matrix formulae it would surely be raising eyebrows.
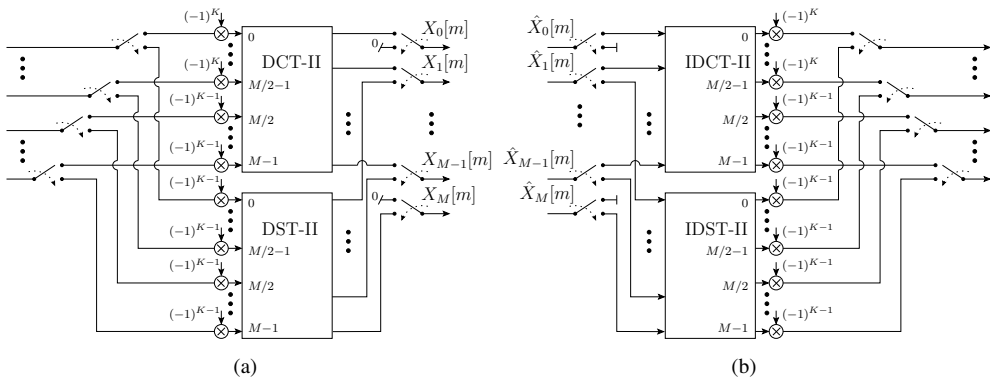
Figure 8.4: Modulation block $\mathbf{\Psi}[m]$ for the even-stacked fast ELT algorithm: (a) Analysis. (b) Synthesis.

8.2 Even-Stacked Fast ELT Algorithm

At first glance the even-stacked fast ELT in Figure 8.3 has the same general form as in the odd-stacked configuration; there are the modulation block and the cascade of butterfly matrices. The more subtle change is the number of subbands which is now $M+1$. The even-stacked ELT modulation has the same subband ordering as the Type-II even-stacked CMFB (See Figure 4.9). The $M \times (M+1)$ matrix function $\mathbf{\Psi}[m]$ flips between the alternating modulation sequences as a function of m. When m is even the high-pass component of the modulation vanishes. Similarly, the low-pass subband is set to zero when m is odd. The generated zero subsamples can be viewed as placeholders as shown in Figure 7.7 (b). In this arrangement the natural frequency ordering of the consecutive subsignal samples is preserved and we may say that the high/low-pass subchannels produce data at rate $1/(2M)$.

The actual definition for $\mathbf{\Psi}[m]$ can be presented in terms of $M \times M$ transform matrices $\mathbf{C}_{\mathrm{II}}$ and $\mathbf{S}_{\mathrm{II}}$ given in (7.6) and (7.7):

$$
\mathbf{\Psi}[m] = \begin{cases} (-1)^K \begin{bmatrix} \mathbf{I}_{M/2 \times M/2} & \\ & -\mathbf{I}_{M/2 \times M/2} \end{bmatrix} \begin{bmatrix} \mathbf{C}_{\mathrm{II}} & \mathbf{0}_{M \times 1} \end{bmatrix} & m \text{ even} \\[4mm] \begin{bmatrix} \mathbf{0}_{M \times 1} & (-1)^{K-1} \mathbf{S}_{\mathrm{II}} \end{bmatrix} & m \text{ odd} \end{cases}
$$

The multiplier $(-1)^{K-1}$ is included just to get one-to-one match with the subfilter definitions but reversing the signs of the other half at the DCT-II output is necessary to get continuous modulation waveforms. This can be observed below as we fight our way from the polyphase matrices into subfilter equations.

The above information is presented in Figure 8.4 as flowgraphs for both analysis and synthesis stages. Multipliers equal to -1 (that are known once the design parameter K is fixed) can be embedded in the DCT/DST-II computation and should not be regarded as a multiplication in the implementation complexity evaluation. Mainly the idea in this modulation block representation is to support the algorithm derivation.

The two other modifications are an array of multipliers that produce sequence of ± 1s next to the $(K-1)$th butterfly stage and each butterfly matrix now wears a hat. This decoration

Algorithm 8.1 Mapping prototype filter coefficients into ELT angles.

Initialization:
$$h^{(K-1)}[n] = (-1)^{K-1}(-1)^{\lfloor (n+3M/2)/(2M) \rfloor} h[n]$$

Iteration:
for $m = K - 1 \to 0$ **do**
  for $n = 0 \to M/2 - 1$ **do**
$$\theta_{n,m} = \tan^{-1}\left(\frac{h^{(m)}[n + 2mM + M]}{h^{(m)}[2(m+1)M - 1 - n]} \right)$$
    if $h^{(m)}[2(m+1)M - 1 - n] < 0$ **then**
      if $h^{(m)}[n + 2mM + M] < 0$ **then**
        $\theta_{n,m} = \theta_{n,m} - \pi$
      else
        $\theta_{n,m} = \theta_{n,m} + \pi$
      end if
    end if
    $c_{n,m} = \cos(\theta_{n,m})$
    $s_{n,m} = \sin(\theta_{n,m})$
    for $\ell = 0 \to 2m - 1$ **do**
      $h^{(m-1)}[n + \ell M] = c_{n,m}\, h^{(m)}[n + \ell M] - s_{n,m}\, h^{(m)}[M - 1 - n + \ell M]$
      $h^{(m-1)}[M - 1 - n + \ell M] = -\big(s_{n,m}\, h^{(m)}[n + (\ell+2)M]$
      $\qquad\qquad\qquad\qquad\qquad\qquad + c_{n,m}\, h^{(m)}[M - 1 - n + (\ell+2)M]\big)$
    end for
  end for
end for

---

marks a small change in the definition; we just move the minus sign to the left bottom submatrix

$$\hat{\mathbf{D}}_m = \begin{bmatrix} \mathbf{C}_m & \mathbf{S}_m \mathbf{J} \\ -\mathbf{J}\mathbf{S}_m & \mathbf{J}\mathbf{C}_m \mathbf{J} \end{bmatrix}.$$

The rotation matrix is no longer orthogonal but the inverse matrix (for $\hat{\mathbf{D}}_m$) is simply its own transpose as we can easily verify

$$\hat{\mathbf{D}}_m \hat{\mathbf{D}}_m^{-1} = \hat{\mathbf{D}}_m \hat{\mathbf{D}}_m^T = \begin{bmatrix} \mathbf{C}_m & \mathbf{S}_m \mathbf{J} \\ -\mathbf{J}\mathbf{S}_m & \mathbf{J}\mathbf{C}_m \mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{C}_m & -\mathbf{S}_m \mathbf{J} \\ \mathbf{J}\mathbf{S}_m & \mathbf{J}\mathbf{C}_m \mathbf{J} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{C}_m \mathbf{C}_m + \mathbf{S}_m \mathbf{S}_m & -\mathbf{C}_m \mathbf{S}_m \mathbf{J} + \mathbf{S}_m \mathbf{J}\mathbf{J}\mathbf{C}_m \mathbf{J} \\ -\mathbf{J}\mathbf{S}_m \mathbf{C}_m + \mathbf{J}\mathbf{C}_m \mathbf{J}\mathbf{J}\mathbf{S}_m & \mathbf{J}\mathbf{S}_m \mathbf{S}_m \mathbf{J} + \mathbf{J}\mathbf{C}_m \mathbf{J}\mathbf{J}\mathbf{C}_m \mathbf{J} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \mathbf{I}.$$

The simplifications above are possible because the product of two diagonal matrices is a diagonal matrix and then we can apply the sum of squares for sines and cosines. We follow the code of conduct in our notations and use the transposed version of $\mathbf{D}_m$ in the forward even-stacked fast ELT algorithm.

8.3 Mapping Prototype Filter into ELT Angles

The necessary requirement for the ELT angle recursion is the PR condition for the prototype filter. From the earlier derivation in Section 4.2 we had the equation

$$\sum_{i=0}^{2K-1-2m} h[\ell + iM]h[\ell + iM + 2mM] = \frac{1}{2M}\delta[m],$$

where $n = 0, 1, \ldots, M/2 - 1$ and $m = 0, 1, \ldots, K - 1$. Actually, the scaling of the prototype filter is not an issue because the impulse response coefficients are applied as a ratio of two terms in the mapping algorithm. Secondly, the conditional statements in the recursion can be avoided if the four quadrant arctan function is used instead of the single argument function. The recursion below utilizes the single argument arctan and corrects the ELT angles with $\pm\pi$ depending on the quadrant defined by the two terms. This just to emphasize that recursion would fail without angle tweaking.

The ELT angle recursion is described with the listing in Algorithm 8.1. In the initialization stage we substitute the selectively sign reversed coefficients of the prototype filter into variable $h^{(K-1)[n]}$ where the superscript indicates $(K - 1)$th temporary impulse response.

The outer loop of iteration proceeds from $K - 1 \to 0$, that is, we solve the ELT angles in order $\theta_{n,K-1}, \ldots, \theta_{n,1}, \theta_{n,0}$. Each iteration begins with the computation of mth set of angles from the temporary impulse response coefficients. The angles are corrected as with four quadrant arctan function before computation of the cosine/sine-terms. Then algorithm proceeds to "pealing off" the current butterfly stage from the impulse response. Then the residual impulse response shortens by $2M$ samples and it is denoted with $h^{(m-1)}[n]$. Eventually the recursion terminates when $m = 0$ and the remaining $\theta_{n,0}$ are computed from the coefficients of the $2M$-length truncated impulse response $h^{(0)}[n]$.

By using the Algorithm 8.1 we obtain the angles for the fast ELT shown in Figure 8.1. Note that in the odd-stacked case the butterfly matrices are symmetric and we thus we leave out the transpose in the representation for the analysis algorithm.

8.4 Polyphase Matrices for Odd-Stacked Fast ELT

The polyphase domain matrices $\boldsymbol{H}_\mathrm{p}(z)$ and $\boldsymbol{F}_\mathrm{p}(z)$ for the flowgraphs in Figure 8.1 are straightforward to write. On the synthesis stage we just start from the Type-IV DCT matrix and left-multiplying it with the butterfly stages gives

$$\boldsymbol{F}_\mathrm{p}(z) = \prod_{i=1}^{K-1} \left(\begin{bmatrix} -\mathbf{C}_i & \mathbf{S}_i\mathbf{J} \\ \mathbf{J}\mathbf{S}_i & \mathbf{J}\mathbf{C}_i\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & z^{-2}\mathbf{I} \end{bmatrix} \right) \begin{bmatrix} -\mathbf{C}_0 & \mathbf{S}_0\mathbf{J} \\ \mathbf{J}\mathbf{S}_0 & \mathbf{J}\mathbf{C}_0\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ z^{-1}\mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{C}_{\mathrm{IV}}.$$

The handy notation for the above is $\boldsymbol{F}_\mathrm{p}(z) = \boldsymbol{F}_\theta(z)\mathbf{C}_{\mathrm{IV}}$ where the matrix $\boldsymbol{F}_\theta(z)$ gathers the polyphase components that corresponds to the prototype filter. The analysis polyphase matrix is constructed in reverse order using the causal inverse for the delay terms:

$$\boldsymbol{H}_\mathrm{p}(z) = \mathbf{C}_{\mathrm{IV}} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ z^{-1}\mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} -\mathbf{C}_0 & \mathbf{S}_0\mathbf{J} \\ \mathbf{J}\mathbf{S}_0 & \mathbf{J}\mathbf{C}_0\mathbf{J} \end{bmatrix} \prod_{i=1}^{K-1} \begin{bmatrix} z^{-2}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} -\mathbf{C}_i & \mathbf{S}_i\mathbf{J} \\ \mathbf{J}\mathbf{S}_i & \mathbf{J}\mathbf{C}_i\mathbf{J} \end{bmatrix}$$

where the transposes can be omitted due to the symmetry of the $\mathbf{D}_m$ matrices and $\mathbf{C}_{\mathrm{IV}}$. Note also the subtle thing in the cross-diagonal term on the analysis side:

$$\begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} z^{-1}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ z^{-1}\mathbf{I} & \mathbf{0} \end{bmatrix}$$

so that the delay term z^{-1} belongs to the left lower block. The PR property of the ELT is simple to verify; just combine the innermost terms in the polyphase matrix product $\boldsymbol{F}_{\mathrm{p}}(z)\boldsymbol{H}_{\mathrm{p}}(z)$ until only a diagonal matrix is left:

$$\boldsymbol{F}_{\mathrm{p}}(z)\boldsymbol{H}_{\mathrm{p}}(z) = z^{-(2K-1)}\mathbf{I}.$$

This is a special case of the biorthogonal definition in (2.25).

The next obstacle is to state the prototype filter polyphase matrix in terms of the impulse response coefficients $h[n]$. We make a silent assumption that the recursive equations truly provide the coefficients $\theta_{n,m}$ for the fast ELT on the grounds of Malvar's work. However, once these equations are given we must check that the subfilters are properly generated by the modulation. This consideration is done for the synthesis side only because the paraunitary PR condition dictates the analysis side once $\boldsymbol{F}_{\mathrm{p}}(z)$ is fixed. Now it is convenient to write $\boldsymbol{F}_\theta(z)$ with four blocks:

$$\boldsymbol{F}_\theta(z) = \begin{bmatrix} z^{-1}\boldsymbol{F}_{00}(z^2)\mathbf{J} & \boldsymbol{F}_{01}(z^2) \\ z^{-1}\boldsymbol{F}_{10}(z^2) & \boldsymbol{F}_{11}(z^2)\mathbf{J} \end{bmatrix}, \tag{8.2}$$

where the diagonal $M/2 \times M/2$ matrices are defined as

$$\begin{aligned} \left[\boldsymbol{F}_{00}(z)\right]_{\ell,\ell} &= -\sqrt{2M}\sum_m (-1)^m h[\ell + M + 2mM]z^{-m} \\ \left[\boldsymbol{F}_{01}(z)\right]_{\ell,\ell} &= \sqrt{2M}\sum_m (-1)^m h[\ell + 2mM]z^{-m} \\ \left[\boldsymbol{F}_{10}(z)\right]_{\ell,\ell} &= -\sqrt{2M}\sum_m (-1)^m h[\ell + 3M/2 + 2mM]z^{-m} \\ \left[\boldsymbol{F}_{11}(z)\right]_{\ell,\ell} &= -\sqrt{2M}\sum_m (-1)^m h[\ell + M/2 + 2mM]z^{-m}. \end{aligned} \tag{8.3}$$

If our formulation is correct, the polyphase matrix elements should now compose the subfilter transfer function

$$F_k^{\mathrm{elt}}(z) = \sum_{n=0}^{2KM-1} 2\,h[n]\cos\left[\left(n + \frac{M+1}{2}\right)\left(k + \frac{1}{2}\right)\frac{\pi}{M}\right]z^{-n}. \tag{8.4}$$

The kth subfilter can be obtained by summing the terms of the kth column of the "M-interpolated" polyphase matrix. That is

$$F_k^{\mathrm{elt}}(z) = \sum_{\ell=0}^{M-1} z^{-\ell}\left[\boldsymbol{F}_{\mathrm{p}}(z^M)\right]_{\ell,k} = \left[\boldsymbol{F}_\theta(z^M)\mathbf{C}_{\mathrm{IV}}\right]_{\ell,k}$$

which must be then be equal to (8.4). The verification of this connection requires a certain degree of stubbornness and my apologies for the headaches the index-trickery may cause.

Let us start with a technical matrix split which shortens the sum notations below. If the matrix $\mathbf{C}_{\mathrm{IV}}$ is partitioned into two $M/2 \times M$ matrices

$$\mathbf{C}_{\mathrm{IV}} = \left[\begin{array}{c} \mathbf{C}_{\mathrm{IV},0} \\ \mathbf{C}_{\mathrm{IV},1} \end{array} \right]$$

we can move these subblocks next to the polyphase components using the distributive law

$$\boldsymbol{F}_{\mathrm{p}}(z) = \boldsymbol{F}_\theta(z)\mathbf{C}_{\mathrm{IV}} = \left(\left[\begin{array}{cc} z^{-1}\boldsymbol{F}_{00}(z^2)\mathbf{J} & \mathbf{0} \\ z^{-1}\boldsymbol{F}_{10}(z^2) & \mathbf{0} \end{array} \right] + \left[\begin{array}{cc} \mathbf{0} & \boldsymbol{F}_{01}(z^2) \\ \mathbf{0} & \boldsymbol{F}_{11}(z^2)\mathbf{J} \end{array} \right] \right) \left[\begin{array}{c} \mathbf{C}_{\mathrm{IV},0} \\ \mathbf{C}_{\mathrm{IV},1} \end{array} \right]$$

$$= \left[\begin{array}{c} z^{-1}\boldsymbol{F}_{00}(z^2)\mathbf{J}\mathbf{C}_{\mathrm{IV},0} \\ z^{-1}\boldsymbol{F}_{10}(z^2)\mathbf{C}_{\mathrm{IV},0} \end{array} \right] + \left[\begin{array}{c} \boldsymbol{F}_{01}(z^2)\mathbf{C}_{\mathrm{IV},1} \\ \boldsymbol{F}_{11}(z^2)\mathbf{J}\mathbf{C}_{\mathrm{IV},1} \end{array} \right].$$

From hereon its just plug'n'play:

$$\sum_{\ell=0}^{M-1} z^{-\ell}\big[\boldsymbol{F}_{\mathrm{p}}(z^M)\big]_{\ell,k} = \sum_{\ell=0}^{M/2-1} z^{-\ell}\big(z^{-M}\big[\boldsymbol{F}_{00}(z^{2M})\mathbf{J}\mathbf{C}_{\mathrm{IV},0}\big]_{\ell,k} + \big[\boldsymbol{F}_{01}(z^{2M})\mathbf{C}_{\mathrm{IV},1}\big]_{\ell,k}\big)$$

$$+ \sum_{\ell=M/2}^{M-1} z^{-\ell}\big(z^{-M}\big[\boldsymbol{F}_{10}(z^{2M})\mathbf{C}_{\mathrm{IV},0}\big]_{\ell-M/2,k} + \big[\boldsymbol{F}_{11}(z^{2M})\mathbf{J}\mathbf{C}_{\mathrm{IV},1}\big]_{\ell-M/2,k}\big)$$

The DCT-IV is substituted back elementwise with $\big[\mathbf{C}_{\mathrm{IV},0}\big]_{n,m} = \big[\mathbf{C}_{\mathrm{IV}}\big]_{n,m}$ and $\big[\mathbf{C}_{\mathrm{IV},1}\big]_{n,m} = \big[\mathbf{C}_{\mathrm{IV}}\big]_{n+M/2,m}$. Then we get rid of $\mathbf{J}$s by reversing the indexing order and merge the two sums:

$$\sum_{\ell=0}^{M-1} z^{-\ell}\big[\boldsymbol{F}_{\mathrm{p}}(z^M)\big]_{\ell,k} = \sum_{\ell=0}^{M/2-1} z^{-\ell}\big(z^{-M}\big[\mathbf{C}_{\mathrm{IV}}\big]_{M/2-1-\ell,k}\big[\boldsymbol{F}_{00}(z^{2M})\big]_{\ell,\ell}$$

$$+ \big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell+M/2,k}\big[\boldsymbol{F}_{01}(z^{2M})\big]_{\ell,\ell}\big) + z^{-(\ell+M/2)}\big(z^{-M}\big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell,k}\big[\boldsymbol{F}_{10}(z^{2M})\big]_{\ell,\ell} \tag{8.5}$$

$$+ \big[\mathbf{C}_{\mathrm{IV}}\big]_{M-1-\ell,k}\big[\boldsymbol{F}_{11}(z^{2M})\big]_{\ell,\ell}\big)$$

The cosine sequence of the $M \times M$ DCT-IV is $4M$-periodic and thus in a functional form where we index outside is bounds the identity $\big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell,k} = -\big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell\pm 2M,k}$ is useful. In addition, we apply the short-hand notation $\xi = (k+1/2)\pi/M$ below and the minus-signs cancel out once we substitute the prototype filter coefficients from the equation (8.3):

$$\sum_{\ell=0}^{M-1} z^{-\ell}\big[\boldsymbol{F}_{\mathrm{p}}(z^M)\big]_{\ell,k} = \sum_{\ell=0}^{M/2-1} z^{-\ell}\big(-z^{-M}\big[\mathbf{C}_{\mathrm{IV}}\big]_{M/2-1-\ell-2M,k}\big[\boldsymbol{F}_{00}(z^{2M})\big]_{\ell,\ell}$$

$$+ \big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell+M/2,k}\big[\boldsymbol{F}_{01}(z^{2M})\big]_{\ell,\ell}\big) + z^{-(\ell+M/2)}\big(-z^{-M}\big[\mathbf{C}_{\mathrm{IV}}\big]_{\ell+2M,k}\big[\boldsymbol{F}_{10}(z^{2M})\big]_{\ell,\ell}$$

$$- \big[\mathbf{C}_{\mathrm{IV}}\big]_{M-1-\ell-2M,k}\big[\boldsymbol{F}_{11}(z^{2M})\big]_{\ell,\ell}\big)$$

$$= 2\sum_{\ell=0}^{M/2-1} z^{-\ell}\sum_{m=0}^{K-1}(-1)^m\Big(\cos\Big[\Big(-\frac{3M}{2}-\ell-\frac{1}{2}\Big)\xi_k\Big]h[\ell+M+2mM]z^{-2mM-M}$$

$$+ \cos\Big[\Big(\ell+\frac{M+1}{2}\Big)\xi_k\Big]h[\ell+2mM]z^{-2mM}$$

$$+ \cos\Big[\Big(\ell+\frac{3M}{2}+\frac{M+1}{2}\Big)\xi_k\Big]h[\ell+\frac{3M}{2}+2mM]z^{-2mM-3M/2}$$

$$+ \cos\Big[\Big(-M-\ell-\frac{1}{2}\Big)\xi_k\Big]h[\ell+M/2+2mM]z^{-2mM-M/2}\Big)$$

If we look at the argument of the cosines we observe that there are still two that contain $-\ell$ term. This poses no difficulties as the trigonometric identity $\cos(\alpha) = \cos(-\alpha)$ handles it. also the delay term $z^{-\ell}$ is moved inside the second sum:

$$
\sum_{\ell=0}^{M-1} z^{-\ell} \big[\boldsymbol{F}_{\mathrm{p}}(z^M) \big]_{\ell,k}
$$

$$
= 2 \sum_{\ell=0}^{M/2-1} z^{-\ell} \sum_{m=0}^{K-1} (-1)^m \Big(\cos\Big[\Big(\ell + M + \frac{M+1}{2}\Big)\xi_k\Big] h[\ell + M + 2mM] z^{-2mM-M}
$$

$$
+ \cos\Big[\Big(\ell + \frac{M+1}{2}\Big)\xi_k\Big] h[\ell + 2mM] z^{-2mM}
$$

$$
+ \cos\Big[\Big(\ell + \frac{3M}{2} + \frac{M+1}{2}\Big)\xi_k\Big] h[\ell + \frac{3M}{2} + 2mM] z^{-2mM-3M/2}
$$

$$
+ \cos\Big[\Big(\ell + \frac{M}{2} + \frac{M+1}{2}\Big)\xi_k\Big] h[\ell + M/2 + 2mM] z^{-2mM-M/2} \Big)
$$

The four components of the sum can be merged when we let $\ell = 0, 1, \ldots, 2M - 1$:

$$
\sum_{\ell=0}^{M-1} z^{-\ell} \big[\boldsymbol{F}_{\mathrm{p}}(z^M) \big]_{\ell,k} = 2 \sum_{\ell=0}^{2M-1} \sum_{m=0}^{K-1} (-1)^m \cos\Big[\Big(\ell + \frac{M+1}{2}\Big)\xi_k\Big] h[\ell + 2mM] z^{-(\ell+2mM)}
$$

The $4M$-periodic property of the modulating sequence is again needed. The trick is to replace ℓ with $\ell + 2mM$ and then compensate the sign reversion. This is accomplished with substitution

$$
\cos\Big[\Big(\ell + \frac{M+1}{2}\Big)\Big(k + \frac{1}{2}\Big)\frac{\pi}{M}\Big] = (-1)^m \cos\Big[\Big(\ell + 2mM + \frac{M+1}{2}\Big)\Big(k + \frac{1}{2}\Big)\frac{\pi}{M}\Big].
$$

As can be seen below, the $(-1)^m$ multipliers cancel out each other and finally we can combine the summation range $n \to \ell + 2mM$:

$$
\sum_{\ell=0}^{M-1} z^{-\ell} \big[\boldsymbol{F}_{\mathrm{p}}(z^M) \big]_{\ell,k}
$$

$$
= 2 \sum_{\ell=0}^{2M-1} \sum_{m=0}^{K-1} (-1)^{2m} \cos\Big[\Big(\ell + 2mM + \frac{M+1}{2}\Big)\Big(k + \frac{1}{2}\Big)\frac{\pi}{M}\Big] h[\ell + 2mM] z^{-(\ell+2mM)}
$$

$$
= \sum_{n=0}^{2KM-1} 2\, h[n] \cos\Big[\Big(n + \frac{M+1}{2}\Big)\Big(k + \frac{1}{2}\Big)\frac{\pi}{M}\Big] z^{-n} = F_k^{\mathrm{elt}}(z).
$$

Everything goes according to my devious plan... (Eevertti, the Kerberos of PIF approves).

8.5 Connection of Even and Odd-Stacked Butterfly Cascades

The fixed prototype filter part of both even and odd-stacked fast ELT algorithms are almost the same - there is only a sign difference in the definition of the butterfly matrices. But the

modification is necessary for the correct generation of the even-stacked subfilters where the modulating sequence is $2M$-periodic. The time varying property of the even-stacked modulation is put aside for the moment. Let us consider the polynomial matrix

$$\hat{\boldsymbol{F}}_\theta(z) = \prod_{i=1}^{K-1} \left(\begin{bmatrix} \mathbf{C}_i & \mathbf{S}_i \mathbf{J} \\ -\mathbf{J}\mathbf{S}_i & \mathbf{J}\mathbf{C}_i\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & z^{-2}\mathbf{I} \end{bmatrix} \right) \begin{bmatrix} \mathbf{C}_0 & \mathbf{S}_0\mathbf{J} \\ -\mathbf{J}\mathbf{S}_0 & \mathbf{J}\mathbf{C}_0\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ z^{-1}\mathbf{I} & \mathbf{0} \end{bmatrix}$$

which is obtained from the inverse transform in Figure 8.4. This sign modified prototype filter polyphase matrix has a similar diagonal structure as above in equation (8.3):

$$\hat{\boldsymbol{F}}_\theta(z) = \begin{bmatrix} z^{-1}\hat{\boldsymbol{F}}_{00}(z^2)\mathbf{J} & \hat{\boldsymbol{F}}_{01}(z^2) \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{bmatrix}.$$

Unfortunately the representation of the submatrices in terms of the impulse response coefficients is unknown. However, if we can find the blockwise connection between $\hat{\boldsymbol{F}}_\theta(z)$ and $\boldsymbol{F}_\theta(z)$ in (8.2), it will provide the necessary equations to be combined with the time varying elements. The fast ELT structure is recursive by nature and thus a favourite playground for induction proof.

Claim: The prototype filter submatrices of the even and odd-stacked fast ELT structure have connection

$$\begin{aligned} \hat{\boldsymbol{F}}^i_{00}(z^2) &= (-1)^{i-1}\boldsymbol{F}^i_{00}(-z^2) \\ \hat{\boldsymbol{F}}^i_{01}(z^2) &= (-1)^i \boldsymbol{F}^i_{01}(-z^2) \\ \hat{\boldsymbol{F}}^i_{10}(z^2) &= (-1)^{i-1}\boldsymbol{F}^i_{10}(-z^2) \\ \hat{\boldsymbol{F}}^i_{11}(z^2) &= (-1)^i \boldsymbol{F}^i_{11}(-z^2) \end{aligned} \qquad (8.6)$$

where i is the number of butterfly stages in the cascade.
Proof:

1° The base case
   When $i = 1$ we have one butterfly matrix. The odd-stacked case $\hat{\boldsymbol{F}}^1(z)$ is

$$\begin{aligned} \boldsymbol{F}^1_\theta(z) &= \begin{bmatrix} z^{-1}\boldsymbol{F}^1_{00}(z^2)\mathbf{J} & \boldsymbol{F}^1_{01}(z^2) \\ z^{-1}\boldsymbol{F}^1_{10}(z^2) & \boldsymbol{F}^1_{11}(z^2)\mathbf{J} \end{bmatrix} \\ &= \begin{bmatrix} -\mathbf{C}_0 & \mathbf{S}_0\mathbf{J} \\ \mathbf{J}\mathbf{S}_0 & \mathbf{J}\mathbf{C}_0\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ z^{-1}\mathbf{I} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} z^{-1}\mathbf{S}_0\mathbf{J} & -\mathbf{C}_0 \\ z^{-1}\mathbf{J}\mathbf{C}_0\mathbf{J} & \mathbf{J}\mathbf{S}_0 \end{bmatrix} \end{aligned}$$

where we can arrange $\mathbf{J}$ into the lower right block by substituting the identity matrix $\mathbf{I} = \mathbf{J}\mathbf{J}$. That is: $\mathbf{J}\mathbf{S}_0 = \mathbf{J}\mathbf{S}_0\mathbf{J}\mathbf{J} = \boldsymbol{F}^1_{11}(z^2)\mathbf{J}$. Then the even-stacked matrix $\hat{\boldsymbol{F}}^1(z)$ is written as in the cascade definition and we manipulate the terms to back up our claim

$$\begin{aligned} \hat{\boldsymbol{F}}^1_\theta(z) &= \begin{bmatrix} z^{-1}\hat{\boldsymbol{F}}^1_{00}(z^2)\mathbf{J} & \hat{\boldsymbol{F}}^1_{01}(z^2) \\ z^{-1}\hat{\boldsymbol{F}}^1_{10}(z^2) & \hat{\boldsymbol{F}}^1_{11}(z^2)\mathbf{J} \end{bmatrix} \\ &= \begin{bmatrix} z^{-1}\mathbf{S}_0\mathbf{J} & \mathbf{C}_0 \\ z^{-1}\mathbf{J}\mathbf{C}_0\mathbf{J} & -(\mathbf{J}\mathbf{S}_0\mathbf{J})\mathbf{J} \end{bmatrix} = \begin{bmatrix} z^{-1}(-1)^{1-1}\boldsymbol{F}^1_{00}(-z^2)\mathbf{J} & (-1)^1\boldsymbol{F}^1_{01}(-z^2) \\ z^{-1}(-1)^{1-1}\boldsymbol{F}^1_{10}(-z^2) & (-1)^1\boldsymbol{F}^1_{11}(-z^2)\mathbf{J} \end{bmatrix}. \end{aligned}$$

The induction base is thus solid and we could proceed to the induction step. Nevertheless, expanding of the $i = 2$ matrix product may enlighten the induction procedure.

When $i = 2$ the odd-stacked butterfly stages $\{0, 1\}$ can be combined into matrix

$$\boldsymbol{F}_\theta^2(z) = \begin{bmatrix} -\mathbf{C}_1 & \mathbf{S}_1\mathbf{J} \\ \mathbf{J}\mathbf{S}_1 & \mathbf{J}\mathbf{C}_1\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & z^{-2}\mathbf{I} \end{bmatrix} \boldsymbol{F}_\theta^1(z)$$

$$= \begin{bmatrix} -\mathbf{C}_1 & z^{-2}\mathbf{S}_1\mathbf{J} \\ \mathbf{J}\mathbf{S}_1 & z^{-2}\mathbf{J}\mathbf{C}_1\mathbf{J} \end{bmatrix} \begin{bmatrix} z^{-1}\mathbf{S}_0\mathbf{J} & -\mathbf{C}_0 \\ z^{-1}\mathbf{J}\mathbf{C}_0\mathbf{J} & \mathbf{J}\mathbf{S}_0 \end{bmatrix}$$

$$= \begin{bmatrix} z^{-1}(-\mathbf{C}_1\mathbf{S}_0 + z^{-2}\mathbf{S}_1\mathbf{C}_0)\mathbf{J} & \mathbf{C}_1\mathbf{C}_0 + z^{-2}\mathbf{S}_1\mathbf{S}_0 \\ z^{-1}\mathbf{J}(\mathbf{S}_1\mathbf{S}_0 + z^{-2}\mathbf{C}_1\mathbf{C}_0)\mathbf{J} & \mathbf{J}(-\mathbf{S}_1\mathbf{C}_0 + z^{-2}\mathbf{C}_1\mathbf{S}_0)\mathbf{J}\mathbf{J} \end{bmatrix}$$

so that the submatrices are

$$\boldsymbol{F}_{00}^2(z^2) = -\mathbf{C}_1\mathbf{S}_0 + z^{-2}\mathbf{S}_1\mathbf{C}_0$$
$$\boldsymbol{F}_{01}^2(z^2) = \mathbf{C}_1\mathbf{C}_0 + z^{-2}\mathbf{S}_1\mathbf{S}_0$$
$$\boldsymbol{F}_{10}^2(z^2) = \mathbf{J}(\mathbf{S}_1\mathbf{S}_0 + z^{-2}\mathbf{C}_1\mathbf{C}_0)\mathbf{J}$$
$$\boldsymbol{F}_{11}^2(z^2) = \mathbf{J}(-\mathbf{S}_1\mathbf{C}_0 + z^{-2}\mathbf{C}_1\mathbf{S}_0)\mathbf{J}.$$

The equations above are then compared against the blocks of the even-stacked matrix which are

$$\hat{\boldsymbol{F}}_\theta^2(z) = \begin{bmatrix} \mathbf{C}_1 & z^{-2}\mathbf{S}_1\mathbf{J} \\ -\mathbf{J}\mathbf{S}_1 & z^{-2}\mathbf{J}\mathbf{C}_1\mathbf{J} \end{bmatrix} \begin{bmatrix} z^{-1}\mathbf{S}_0\mathbf{J} & \mathbf{C}_0 \\ z^{-1}\mathbf{J}\mathbf{C}_0\mathbf{J} & -\mathbf{J}\mathbf{S}_0 \end{bmatrix}$$

$$= \begin{bmatrix} z^{-1}(\mathbf{C}_1\mathbf{S}_0 + z^{-2}\mathbf{S}_1\mathbf{C}_0)\mathbf{J} & \mathbf{C}_1\mathbf{C}_0 - z^{-2}\mathbf{S}_1\mathbf{S}_0 \\ z^{-1}\mathbf{J}(-\mathbf{S}_1\mathbf{S}_0 + z^{-2}\mathbf{C}_1\mathbf{C}_0)\mathbf{J} & \mathbf{J}(-\mathbf{S}_1\mathbf{C}_0 - z^{-2}\mathbf{C}_1\mathbf{S}_0)\mathbf{J}\mathbf{J} \end{bmatrix}.$$

Collecting the terms produces

$$\hat{\boldsymbol{F}}_{00}^2(z^2) = \mathbf{C}_1\mathbf{S}_0 + z^{-2}\mathbf{S}_1\mathbf{C}_0 = (-1)^{2-1}\boldsymbol{F}_{00}^2(-z^2)$$
$$\hat{\boldsymbol{F}}_{01}^2(z^2) = \mathbf{C}_1\mathbf{C}_0 - z^{-2}\mathbf{S}_1\mathbf{S}_0 = (-1)^2\boldsymbol{F}_{01}^2(-z^2)$$
$$\hat{\boldsymbol{F}}_{10}^2(z^2) = \mathbf{J}(-\mathbf{S}_1\mathbf{S}_0 + z^{-2}\mathbf{C}_1\mathbf{C}_0)\mathbf{J} = (-1)^{2-1}\boldsymbol{F}_{10}^2(-z^2)$$
$$\hat{\boldsymbol{F}}_{11}^2(z^2) = \mathbf{J}(-\mathbf{S}_1\mathbf{C}_0 - z^{-2}\mathbf{C}_1\mathbf{S}_0)\mathbf{J} = (-1)^2\boldsymbol{F}_{11}^2(-z^2)$$

which is just the desired form. Listing the particular polyphase matrices for $i > 2$ will not offer any better insight for our purpose. At least the $i = 2$ case reveals why this indirect approach is necessary to gain access into the polyphase matrices represented with the prototype filter impulse response coefficients.

2° Induction assumption
   The even and odd-stacked fast ELT connection in equation (8.6) holds for some $i \in \mathbb{N}$.

3° Induction step
   The polyphase matrices for the odd-stacked case are first written when $i + 1$ and expressed with the submatrices of $\boldsymbol{F}^i(z)$:

$$\boldsymbol{F}_\theta^{i+1}(z) = \begin{bmatrix} -\mathbf{C}_{i+1} & z^{-2}\mathbf{S}_{i+1}\mathbf{J} \\ \mathbf{J}\mathbf{S}_{i+1} & z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J} \end{bmatrix} \boldsymbol{F}_\theta^i(z)$$

$$= \begin{bmatrix} -\mathbf{C}_{i+1} & z^{-2}\mathbf{S}_{i+1}\mathbf{J} \\ \mathbf{J}\mathbf{S}_{i+1} & z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J} \end{bmatrix} \begin{bmatrix} z^{-1}\boldsymbol{F}_{00}^i(z^2)\mathbf{J} & \boldsymbol{F}_{01}^i(z^2) \\ z^{-1}\boldsymbol{F}_{10}^i(z^2) & \boldsymbol{F}_{11}^i(z^2)\mathbf{J} \end{bmatrix}$$

$$= \begin{bmatrix} z^{-1}\boldsymbol{F}_{00}^{i+1}(z^2)\mathbf{J} & \boldsymbol{F}_{01}^{i+1}(z^2) \\ z^{-1}\boldsymbol{F}_{10}^{i+1}(z^2) & \boldsymbol{F}_{11}^{i+1}(z^2)\mathbf{J} \end{bmatrix}$$

where

$$\boldsymbol{F}_{00}^{i+1}(z^2) = -\mathbf{C}_{i+1}\boldsymbol{F}_{00}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\boldsymbol{F}_{10}^i(z^2)\mathbf{J}$$
$$\boldsymbol{F}_{01}^{i+1}(z^2) = -\mathbf{C}_{i+1}\boldsymbol{F}_{01}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\boldsymbol{F}_{11}^i(z^2)\mathbf{J}$$
$$\boldsymbol{F}_{10}^{i+1}(z^2) = \mathbf{J}\mathbf{S}_{i+1}\boldsymbol{F}_{00}^i(z^2)\mathbf{J} + z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J}\boldsymbol{F}_{10}^i(z^2)$$
$$\boldsymbol{F}_{11}^{i+1}(z^2) = \mathbf{J}(\mathbf{S}_{i+1}\boldsymbol{F}_{01}^i(z^2)\mathbf{J} + z^{-2}\mathbf{C}_{i+1}\mathbf{J}\boldsymbol{F}_{11}^i(z^2)).$$

Then we write open $\hat{\boldsymbol{F}}_\theta^{i+1}(z)$ which yields

$$\hat{\boldsymbol{F}}_\theta^{i+1}(z) = \begin{bmatrix} \mathbf{C}_{i+1} & z^{-2}\mathbf{S}_{i+1}\mathbf{J} \\ -\mathbf{J}\mathbf{S}_{i+1} & z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J} \end{bmatrix} \begin{bmatrix} z^{-1}\hat{\boldsymbol{F}}_{00}^i(z^2)\mathbf{J} & \hat{\boldsymbol{F}}_{01}^i(z^2) \\ z^{-1}\hat{\boldsymbol{F}}_{10}^i(z^2) & \hat{\boldsymbol{F}}_{11}^i(z^2)\mathbf{J} \end{bmatrix}$$
$$= \begin{bmatrix} z^{-1}(\mathbf{C}_{i+1}\hat{\boldsymbol{F}}_{00}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{10}^i(z^2)\mathbf{J})\mathbf{J} & \cdots \\ z^{-1}(-\mathbf{J}\mathbf{S}_{i+1}\hat{\boldsymbol{F}}_{00}^i(z^2)\mathbf{J} + z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{10}^i(z^2)) & \\ \mathbf{C}_{i+1}\hat{\boldsymbol{F}}_{01}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{11}^i(z^2)\mathbf{J} \\ \mathbf{J}(-\mathbf{S}_{i+1}\hat{\boldsymbol{F}}_{01}^i(z^2)\mathbf{J} + z^{-2}\mathbf{C}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{11}^i(z^2))\mathbf{J} \end{bmatrix}.$$

Considering next the four submatrices separately we first apply the induction assumption and then arrange the signs so that the derived formulae match with (8.6) when the recursion index is $i + 1$:

$$\begin{aligned}
\hat{\boldsymbol{F}}_{00}^{i+1}(z^2) &= \mathbf{C}_{i+1}\hat{\boldsymbol{F}}_{00}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{10}^i(z^2)\mathbf{J} \\
&= \mathbf{C}_{i+1}\big((-1)^{i-1}\boldsymbol{F}_{00}^i(-z^2)\big) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\big((-1)^{i-1}\boldsymbol{F}_{10}^i(-z^2)\big)\mathbf{J} \\
&= (-1)^i\big(-\mathbf{C}_{i+1}\boldsymbol{F}_{00}^i(-z^2) + (-z^{-2})\mathbf{S}_{i+1}\mathbf{J}\boldsymbol{F}_{10}^i(-z^2)\mathbf{J}\big) \\
&= (-1)^{(i+1)-1}\boldsymbol{F}_{00}^{i+1}(-z^2) \\
\hat{\boldsymbol{F}}_{01}^{i+1}(z^2) &= \mathbf{C}_{i+1}\hat{\boldsymbol{F}}_{01}^i(z^2) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{11}^i(z^2)\mathbf{J} \\
&= \mathbf{C}_{i+1}\big((-1)^i\boldsymbol{F}_{01}^i(-z^2)\big) + z^{-2}\mathbf{S}_{i+1}\mathbf{J}\big((-1)^i\boldsymbol{F}_{11}^i(-z^2)\big)\mathbf{J} \\
&= (-1)^{i+1}\big(-\mathbf{C}_{i+1}\boldsymbol{F}_{01}^i(-z^2) + (-z^{-2})\mathbf{S}_{i+1}\mathbf{J}\boldsymbol{F}_{11}^i(-z^2)\mathbf{J}\big) \\
&= (-1)^{i+1}\boldsymbol{F}_{01}^{i+1}(-z^2) \\
\hat{\boldsymbol{F}}_{10}^{i+1}(z^2) &= -\mathbf{J}\mathbf{S}_{i+1}\hat{\boldsymbol{F}}_{00}^i(z^2)\mathbf{J} + z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{10}^i(z^2) \\
&= -\mathbf{J}\mathbf{S}_{i+1}\big((-1)^{i-1}\boldsymbol{F}_{00}^i(-z^2)\big)\mathbf{J} + z^{-2}\mathbf{J}\mathbf{C}_{i+1}\mathbf{J}\big((-1)^{i-1}\boldsymbol{F}_{10}^i(-z^2)\big) \\
&= (-1)^i\big(\mathbf{J}\mathbf{S}_{i+1}\boldsymbol{F}_{00}^i(-z^2)\mathbf{J} + (-z^{-2})\mathbf{J}\mathbf{C}_{i+1}\mathbf{J}\boldsymbol{F}_{10}^i(-z^2)\big) \\
&= (-1)^{(i+1)-1}\boldsymbol{F}_{10}^{i+1}(-z^2) \\
\hat{\boldsymbol{F}}_{11}^{i+1}(z^2) &= \mathbf{J}(-\mathbf{S}_{i+1}\hat{\boldsymbol{F}}_{01}^i(z^2)\mathbf{J} + z^{-2}\mathbf{C}_{i+1}\mathbf{J}\hat{\boldsymbol{F}}_{11}^i(z^2)) \\
&= \mathbf{J}\big(-\mathbf{S}_{i+1}\big((-1)^i\boldsymbol{F}_{01}^i(-z^2)\big)\mathbf{J} + z^{-2}\mathbf{C}_{i+1}\mathbf{J}\big((-1)^i\boldsymbol{F}_{11}^i(-z^2)\big)\big) \\
&= (-1)^{i+1}\mathbf{J}\big(\mathbf{S}_{i+1}\boldsymbol{F}_{01}^i(-z^2)\mathbf{J} + (-z^{-2})\mathbf{C}_{i+1}\mathbf{J}\boldsymbol{F}_{11}^i(-z^2)\big) \\
&= (-1)^{i+1}\boldsymbol{F}_{11}^{i+1}(-z^2)
\end{aligned}$$

The claim is valid on the grounds of $1° - 3°$.

171

By combining the above proven connection with the submatrix equations in (8.3) yields

$$
\begin{aligned}
\left[\hat{\boldsymbol{F}}_{00}(z^2)\right]_{\ell,\ell} &= \sqrt{2M}(-1)^{K-1}\left(-\sum_{m}(-1)^m h[\ell+M+2mM](-z^2)^{-m}\right) \\
&= \sqrt{2M}(-1)^{K}\sum_{m} h[\ell+M+2mM]z^{-2m} \\
\left[\hat{\boldsymbol{F}}_{01}(z^2)\right]_{\ell,\ell} &= \sqrt{2M}(-1)^{K}\sum_{m}(-1)^m h[\ell+2mM](-z^2)^{-m} \\
&= \sqrt{2M}(-1)^{K}\sum_{m} h[\ell+2mM]z^{-2m} \\
\left[\hat{\boldsymbol{F}}_{10}(z^2)\right]_{\ell,\ell} &= \sqrt{2M}(-1)^{K-1}\left(-\sum_{m}(-1)^m h[\ell+3M/2+2mM](-z^2)^{-m}\right) \\
&= \sqrt{2M}(-1)^{K}\sum_{m} h[\ell+3M/2+2mM]z^{-2m} \\
\left[\hat{\boldsymbol{F}}_{11}(z^2)\right]_{\ell,\ell} &= \sqrt{2M}(-1)^{K}\left(-\sum_{m}(-1)^m h[\ell+M/2+2mM](-z^2)^{-m}\right) \\
&= \sqrt{2M}(-1)^{K-1}\sum_{m} h[\ell+M/2+2mM]z^{-2m},
\end{aligned}
\tag{8.7}
$$

where $m = 0, 1, \ldots, K-1$. Then we are ready to finalize the derivation for even-stacked fast ELT algorithm.

8.6 Subfilters of Even-Stacked fast ELT

The subfilter impulse response equations that we are trying to reproduce with the modified inverse ELT algorithm are

$$
f_k^{\text{elt}-\text{e}}[n] = c_k\, h[n] \cos\left[\left(n+\frac{M+1}{2}\right)\frac{k\pi}{M}\right], \qquad k \in [0, M-1] \tag{8.8}
$$

and

$$
\breve{f}_k^{\text{elt}-\text{e}}[n] = -c_k\, h[n] \sin\left[\left(n+\frac{M+1}{2}\right)\frac{k\pi}{M}\right], \qquad k \in [1, M] \tag{8.9}
$$

where

$$
c_k = \begin{cases} \sqrt{2} & k = 0 \text{ or } k = M \\ 2 & \text{otherwise.} \end{cases}
$$

The polyphase matrix is then also defined for two phases so that $\boldsymbol{F}_{\mathrm{p}}(z)$ and $\breve{\boldsymbol{F}}_{\mathrm{p}}(z)$ alternate according to the subsignal time index parity. It is reminded that we introduced above the operator $\mathcal{M}\{\cdot\}$ which reverses the sign of the complex variable z of its argument transfer function and, in addition, reverses the sign of the argument if the time-phase is even. In this way we handle the $-(-1)^m$ multiplier that operates on the half diagonal (output lines $0, 1, \ldots, M/2 - 1$ before the interpolation if Figure 8.3 (b)). Then $\mathcal{M}\{\cdot\}$ operates the top submatrices of the polyphase matrix. The modulation terms are obtained from Figure 8.4 (b) and we write the elements of $\mathbf{C}_{\mathrm{II}}$ and $\mathbf{S}_{\mathrm{II}}$ once more for easy reference:

$$
\left[\mathbf{C}_{\mathrm{II}}\right]_{n,k} = \frac{c_k}{\sqrt{2M}} \cos\left[\left(n+\frac{1}{2}\right)\frac{k\pi}{M}\right]
$$

$$\left[\mathbf{S}_{\mathrm{II}}\right]_{n,k} = \frac{c_{k+1}}{\sqrt{2M}} \sin\left[\left(n+\frac{1}{2}\right)\frac{(k+1)\pi}{M}\right].$$

Then we gather all this knowledge of the even-stacked fast ELT algorithm and consider the two polyphase matrix cases:

$1°$ when m is even

$$\begin{aligned}
\boldsymbol{F}_{\mathrm{p}}(z) &= \left[\begin{array}{cc} \mathcal{M}\{z^{-1}\hat{\boldsymbol{F}}_{00}(z^2)\mathbf{J}\} & \mathcal{M}\{\hat{\boldsymbol{F}}_{01}(z^2)\} \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\left[\begin{array}{cc} (-1)^K\mathbf{I} & \mathbf{0} \\ \mathbf{0} & (-1)^{K-1}\mathbf{I} \end{array}\right]\mathbf{C}_{\mathrm{II}} \\
&= (-1)^{K-1}\left[\begin{array}{cc} -(-z)^{-1}\hat{\boldsymbol{F}}_{00}((-z)^2)\mathbf{J} & -\hat{\boldsymbol{F}}_{01}((-z)^2) \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\left[\begin{array}{cc} -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{array}\right]\mathbf{C}_{\mathrm{II}} \\
&= (-1)^{K-1}\left[\begin{array}{cc} -z^{-1}\hat{\boldsymbol{F}}_{00}(z^2)\mathbf{J} & -\hat{\boldsymbol{F}}_{01}(z^2) \\ -z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\mathbf{C}_{\mathrm{II}}
\end{aligned}$$

$2°$ when m is odd

$$\begin{aligned}
\breve{\boldsymbol{F}}_{\mathrm{p}}(z) &= (-1)^{K-1}\left[\begin{array}{cc} \mathcal{M}\{z^{-1}\hat{\boldsymbol{F}}_{00}(z^2)\mathbf{J}\} & \mathcal{M}\{\hat{\boldsymbol{F}}_{01}(z^2)\} \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\mathbf{S}_{\mathrm{II}} \\
&= (-1)^{K-1}\left[\begin{array}{cc} (-z)^{-1}\hat{\boldsymbol{F}}_{00}((-z)^2)\mathbf{J} & \hat{\boldsymbol{F}}_{01}((-z)^2) \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\mathbf{S}_{\mathrm{II}} \\
&= (-1)^{K-1}\left[\begin{array}{cc} -z^{-1}\hat{\boldsymbol{F}}_{00}(z^2)\mathbf{J} & \hat{\boldsymbol{F}}_{01}(z^2) \\ z^{-1}\hat{\boldsymbol{F}}_{10}(z^2) & \hat{\boldsymbol{F}}_{11}(z^2)\mathbf{J} \end{array}\right]\mathbf{S}_{\mathrm{II}}
\end{aligned}$$

All that is left is to substitute $z^M \to z$ and sum the columns of the (synthesis) polyphase matrices to recover the transfer functions of the inverse stage. The derivation is similar with the odd-stacked version in Section 8.4; we just replace DCT-IV with type-II transforms and odd-stacked polyphase matrix with the even-stacked one. Thus the equation (8.5) can be recycled and the even-stacked cosine modulation phase produces

$$\begin{aligned}
\sum_{\ell=0}^{M-1} z^{-\ell}\left[\boldsymbol{F}_{\mathrm{p}}(z^M)\right]_{\ell,k} &= (-1)^{K-1}\sum_{\ell=0}^{M/2-1} z^{-\ell}\left(-z^{-M}\left[\mathbf{C}_{\mathrm{II}}\right]_{M/2-1-\ell,k}\left[\hat{\boldsymbol{F}}_{00}(z^{2M})\right]_{\ell,\ell}\right. \\
&\left. -\left[\mathbf{C}_{\mathrm{II}}\right]_{\ell+M/2,k}\left[\hat{\boldsymbol{F}}_{01}(z^{2M})\right]_{\ell,\ell}\right) + z^{-(\ell+M/2)}\left(-z^{-M}\left[\mathbf{C}_{\mathrm{II}}\right]_{\ell,k}\left[\hat{\boldsymbol{F}}_{10}(z^{2M})\right]_{\ell,\ell}\right. \\
&\left. +\left[\mathbf{C}_{\mathrm{II}}\right]_{M-1-\ell,k}\left[\hat{\boldsymbol{F}}_{11}(z^{2M})\right]_{\ell,\ell}\right)
\end{aligned}$$

The cosine sequence of the $M \times M$ DCT-II is $2M$-periodic and thus $\left[\mathbf{C}_{\mathrm{II}}\right]_{\ell,k} = \left[\mathbf{C}_{\mathrm{II}}\right]_{\ell\pm 2M,k}$. This can be applied to match the modulation phase with the ELT definition:

$$
\sum_{\ell=0}^{M-1} z^{-\ell}\left[\boldsymbol{F}_{\mathrm{p}}(z^M)\right]_{\ell,k} = (-1)^{K-1} \sum_{\ell=0}^{M/2-1} z^{-\ell}\Big(-z^{-M}\left[\mathbf{C}_{\mathrm{II}}\right]_{M/2-1-\ell-2M,k}\left[\hat{\boldsymbol{F}}_{00}(z^{2M})\right]_{\ell,\ell}
$$
$$
- \left[\mathbf{C}_{\mathrm{II}}\right]_{\ell+M/2,k}\left[\hat{\boldsymbol{F}}_{01}(z^{2M})\right]_{\ell,\ell}\Big) + z^{-(\ell+M/2)}\Big(-z^{-M}\left[\mathbf{C}_{\mathrm{II}}\right]_{\ell+2M,k}\left[\hat{\boldsymbol{F}}_{10}(z^{2M})\right]_{\ell,\ell}
$$
$$
+ \left[\mathbf{C}_{\mathrm{II}}\right]_{M-1-\ell-2M,k}\left[\hat{\boldsymbol{F}}_{11}(z^{2M})\right]_{\ell,\ell}\Big)
$$
$$
= (-1)^K c_k \sum_{\ell=0}^{M/2-1}\Big(z^{-\ell}
$$
$$
\times \sum_{m=0}^{K-1} \cos\left[\left(-\frac{3M}{2}-\ell-\frac{1}{2}\right)\frac{k\pi}{M}\right](-1)^K h[\ell+M+2mM]z^{-2mM-M}
$$
$$
+ \cos\left[\left(\ell+\frac{M+1}{2}\right)\frac{k\pi}{M}\right](-1)^K h[\ell+2mM]z^{-2mM}
$$
$$
+ \cos\left[\left(\ell+\frac{3M}{2}+\frac{M+1}{2}\right)\frac{k\pi}{M}\right](-1)^K h[\ell+\frac{3M}{2}+2mM]z^{-2mM-3M/2}
$$
$$
- \cos\left[\left(-M-\ell-\frac{1}{2}\right)\frac{k\pi}{M}\right](-1)^{K-1} h[\ell+M/2+2mM]z^{-2mM-M/2}\Big)
$$

If we look at the argument of the two cosine terms, the minus sign of ℓ can be reversed because $\cos(\alpha) = \cos(-\alpha)$. In addition, the delay term $z^{-\ell}$ is moved inside the second sum and the $(-1)^K$ multipliers vanish due to $(-1)^K(-1)^K = (-1)^{2K} = 1$:

$$
\sum_{\ell=0}^{M-1} z^{-\ell}\left[\boldsymbol{F}_{\mathrm{p}}(z^M)\right]_{\ell,k} = (-1)^{2K} c_k \sum_{\ell=0}^{M/2-1}\Big(z^{-\ell}
$$
$$
\times \sum_{m=0}^{K-1} \cos\left[\left(\ell+M+\frac{M+1}{2}\right)\frac{k\pi}{M}\right]h[\ell+M+2mM]z^{-2mM-M}
$$
$$
+ \cos\left[\left(\ell+\frac{M+1}{2}\right)\frac{k\pi}{M}\right]h[\ell+2mM]z^{-2mM}
$$
$$
+ \cos\left[\left(\ell+\frac{3M}{2}+\frac{M+1}{2}\right)\frac{k\pi}{M}\right]h[\ell+\frac{3M}{2}+2mM]z^{-2mM-3M/2}
$$
$$
+ \cos\left[\left(\ell+\frac{M}{2}+\frac{M+1}{2}\right)\frac{k\pi}{M}\right]h[\ell+M/2+2mM]z^{-2mM-M/2}\Big)
$$

The four terms of the sum can be combined when we let $\ell = 0, 1, \ldots, 2M-1$:

$$
\sum_{\ell=0}^{M-1} z^{-\ell}\left[\boldsymbol{F}_{\mathrm{p}}(z^M)\right]_{\ell,k} = \sum_{\ell=0}^{2M-1}\sum_{m=0}^{K-1} c_k\, h[\ell+2mM] \cos\left[\left(\ell+\frac{M+1}{2}\right)\frac{k\pi}{M}\right]z^{-(\ell+2mM)}
$$

Since the modulating sequence is $2M$-periodic we can add an arbitrary integer multiple of $2M$ to ℓ without changing the value of the cosine-term. Quite natural choice is $2mM$ so that we can replace the double sum with a single summation when the index ranges $n =$

$0, 1, \ldots, 2KM - 1$:

$$\sum_{\ell=0}^{M-1} z^{-\ell} \big[\boldsymbol{F}_{\mathrm{p}}(z^M) \big]_{\ell,k}$$

$$= \sum_{\ell=0}^{2M-1} \sum_{m=0}^{K-1} c_k\, h[\ell + 2mM] \cos\left[\left(\ell + 2mM + \frac{M+1}{2} \right) \frac{k\pi}{M} \right] z^{-(\ell + 2mM)}$$

$$= \sum_{n=0}^{2KM-1} c_k\, h[n] \cos\left[\left(n + \frac{M+1}{2} \right) \frac{k\pi}{M} \right] z^{-n} = F_k^{\mathrm{elt-e}}(z).$$

Sine modulation is analogical with the above

$$\sum_{\ell=0}^{M-1} z^{-\ell} \big[\breve{\boldsymbol{F}}_{\mathrm{p}}(z^M) \big]_{\ell,k} = (-1)^{K-1} \sum_{\ell=0}^{M/2-1} z^{-\ell} \Big(- z^{-M} \big[\mathbf{S}_{\mathrm{II}} \big]_{M/2-1-\ell,k} \big[\hat{\boldsymbol{F}}_{00}(z^{2M}) \big]_{\ell,\ell}$$

$$+ \big[\mathbf{S}_{\mathrm{II}} \big]_{\ell+M/2,k} \big[\hat{\boldsymbol{F}}_{01}(z^{2M}) \big]_{\ell,\ell} \Big) + z^{-(\ell+M/2)} \Big(z^{-M} \big[\mathbf{S}_{\mathrm{II}} \big]_{\ell,k} \big[\hat{\boldsymbol{F}}_{10}(z^{2M}) \big]_{\ell,\ell}$$

$$+ \big[\mathbf{S}_{\mathrm{II}} \big]_{M-1-\ell,k} \big[\hat{\boldsymbol{F}}_{11}(z^{2M}) \big]_{\ell,\ell} \Big)$$

The sine-sequence of the $M \times M$ DST-II is $2M$-periodic and thus $\big[\mathbf{S}_{\mathrm{II}} \big]_{\ell,k} = \big[\mathbf{S}_{\mathrm{II}} \big]_{\ell \pm 2M, k}$. Substitution yields

$$\sum_{\ell=0}^{M-1} z^{-\ell} \big[\breve{\boldsymbol{F}}_{\mathrm{p}}(z^M) \big]_{\ell,k} = (-1)^{K-1} \sum_{\ell=0}^{M/2-1} z^{-\ell} \Big(- z^{-M} \big[\mathbf{S}_{\mathrm{II}} \big]_{M/2-1-\ell-2M,k} \big[\hat{\boldsymbol{F}}_{00}(z^{2M}) \big]_{\ell,\ell}$$

$$+ \big[\mathbf{S}_{\mathrm{II}} \big]_{\ell+M/2,k} \big[\hat{\boldsymbol{F}}_{01}(z^{2M}) \big]_{\ell,\ell} \Big) + z^{-(\ell+M/2)} \Big(z^{-M} \big[\mathbf{S}_{\mathrm{II}} \big]_{\ell+2M,k} \big[\hat{\boldsymbol{F}}_{10}(z^{2M}) \big]_{\ell,\ell}$$

$$+ \big[\mathbf{S}_{\mathrm{II}} \big]_{M-1-\ell-2M,k} \big[\hat{\boldsymbol{F}}_{11}(z^{2M}) \big]_{\ell,\ell} \Big)$$

$$= (-1)^{K-1} c_{k+1} \sum_{\ell=0}^{M/2-1} \Bigg(z^{-\ell}$$

$$\times \sum_{m=0}^{K-1} - \sin\left[\left(- \frac{3M}{2} - \ell - \frac{1}{2} \right) \frac{(k+1)\pi}{M} \right] (-1)^K h[\ell + M + 2mM] z^{-2mM-M}$$

$$+ \sin\left[\left(\ell + \frac{M+1}{2} \right) \frac{(k+1)\pi}{M} \right] (-1)^K h[\ell + 2mM] z^{-2mM}$$

$$+ \sin\left[\left(\ell + \frac{3M}{2} + \frac{M+1}{2} \right) \frac{(k+1)\pi}{M} \right] (-1)^K h[\ell + \frac{3M}{2} + 2mM] z^{-2mM-3M/2}$$

$$+ \sin\left[\left(- M - \ell - \frac{1}{2} \right) \frac{(k+1)\pi}{M} \right] (-1)^{K-1} h[\ell + M/2 + 2mM] z^{-2mM-M/2} \Bigg)$$

Again we encounter the problem where the variable ℓ has the "wrong indexing direction" in the argument of the sine function. The identity $\sin(\alpha) = -\sin(-\alpha)$ saves our skins and this

also sets the signs so that $(-1)^{2K-1} = -1$ is the common factor:

$$\sum_{\ell=0}^{M-1} z^{-\ell} \left[\breve{\boldsymbol{F}}_{\mathrm{p}}(z^M)\right]_{\ell,k} = (-1)^{2K-1} c_{k+1} \sum_{\ell=0}^{M/2-1} \Bigg(z^{-\ell}$$

$$\times \sum_{m=0}^{K-1} \sin\left[\left(\ell + M + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + M + 2mM] z^{-2mM-M}$$

$$+ \sin\left[\left(\ell + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + 2mM] z^{-2mM}$$

$$+ \sin\left[\left(\ell + \frac{3M}{2} + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + \frac{3M}{2} + 2mM] z^{-2mM-3M/2}$$

$$+ \sin\left[\left(\ell + M/2 + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + M/2 + 2mM] z^{-2mM-M/2} \Bigg).$$

Then we have succeeded in matching the modulation phase with the even-stacked sine modulation definition. To polish the result we merge the four components

$$\sum_{\ell=0}^{M-1} z^{-\ell} \left[\breve{\boldsymbol{F}}_{\mathrm{p}}(z^M)\right]_{\ell,k}$$

$$= -c_{k+1} \sum_{\ell=0}^{2M-1} \sum_{m=0}^{K-1} \sin\left[\left(\ell + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + 2mM] z^{-(\ell+2mM)}.$$

On the grounds of the $2M$-periodicity of the sine modulation we replace ℓ with $\ell + 2mM$. Then the double sum simplifies into

$$\sum_{\ell=0}^{M-1} z^{-\ell} \left[\breve{\boldsymbol{F}}_{\mathrm{p}}(z^M)\right]_{\ell,k}$$

$$= -c_{k+1} \sum_{\ell=0}^{2M-1} \sum_{m=0}^{K-1} \sin\left[\left(\ell + 2mM + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] h[\ell + 2mM] z^{-(\ell+2mM)}$$

$$= \sum_{n=0}^{2KM-1} -c_{k+1} \, h[n] \sin\left[\left(n + \frac{M+1}{2}\right)\frac{(k+1)\pi}{M}\right] z^{-n} = \breve{F}_{k+1}^{\mathrm{elt-e}}(z),$$

where we let $k = 0, 1, \ldots, M-1$ because the polyphase matrix has M columns. The even-stacked inverse fast ELT thus generates the correct subfilter impulse responses using the implementation structure in Figure 8.3. The forward transform is an mirror image of the synthesis stage but includes one feature worth of clarification. When the terms that correspond to the prototype filter part are merged in the polyphase matrix product $\boldsymbol{F}_{\mathrm{p}}(z)\boldsymbol{H}_{\mathrm{p}}(z)$, it causes an odd delay $2K - 1$ for the subsamples (without processing delay). Thus PR requires that the sign flipping multipliers have the opposite phase in the forward and inverse stages.

8.7 Realization Tips for Fast ELT Algorithms

8.7.1 Cascade of Scaled Butterfly Matrices

We conclude the fast ELT topic with selected implementation considerations. The focus is solely on the prototype filter realization; the fast computation of DCTs is a vast subject which is addressed, e.g., in [76]. The first enhancement for the fast ELT structure is the scaling of the butterfly cascade to reduce the number of multiplications. For simplicity we pick $M = 2$ butterfly cascade where the coefficients are denoted with $c_m = \cos \theta_m$ and $s_m = \sin \theta_m$. Then the prototype filter part for odd-stacked algorithm is

$$\boldsymbol{F}_\theta(z) = \begin{bmatrix} -c_{K-1} & s_{K-1} \\ s_{K-1} & c_{K-1} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-2} \end{bmatrix} \cdots \begin{bmatrix} -c_1 & s_1 \\ s_1 & c_1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-2} \end{bmatrix}$$
$$\times \begin{bmatrix} -c_0 & s_0 \\ s_0 & c_0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ z^{-1} & 0 \end{bmatrix}$$

It was suggested in [57] that c_m are pulled out from the butterfly matrices so that

$$\begin{bmatrix} -c_m & s_m \\ s_m & c_m \end{bmatrix} = c_m \begin{bmatrix} -1 & s_m/c_m \\ s_m/c_m & 1 \end{bmatrix}$$

for each $m > 0$. The scaling terms are gathered at the 0th butterfly stage:

$$\prod_{m=1}^{K-1} c_m \begin{bmatrix} -c_0 & s_0 \\ s_0 & c_0 \end{bmatrix} = \begin{bmatrix} -c_0' & s_0' \\ s_0' & c_0' \end{bmatrix} = \mathbf{D}_0'$$

This matrix can be further expanded

$$\mathbf{D}_0' = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \mathrm{diag}\{-c_0' + s_0', s_0', c_0' + s_0'\} \begin{bmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix}$$

which trades one multiplication into addition.

The problem with this kind of scheme are scaling terms $\cos \theta_m \approx 0$ causing large coefficient absolute values: $|\sin \theta_m / \cos \theta_m| \gg 1$. This may lead to unnecessarily strict signal scaling policy to avoid clipping or overflow conditions.

A simple fix is to select the mth scaling term with

$$d_m = \begin{cases} \cos \theta_m & \text{if } |\cos \theta_m| \geq |\sin \theta_m| \\ \sin \theta_m & \text{otherwise.} \end{cases}$$

Plainly, we select the one with the larger absolute value as the scaling term so that $1/\sqrt{2} \leq |d_m| \leq 1$. Depending on the value of θ_m, the two cases are

$$\mathbf{D}_m' = \begin{bmatrix} -1 & s_m/d_m \\ s_m/d_m & 1 \end{bmatrix} \quad \text{or} \quad \mathbf{D}_m' = \begin{bmatrix} -c_m/d_m & 1 \\ 1 & c_m/d_m \end{bmatrix}.$$

If this alternative scaling scheme is adopted it guarantees that $-1 \leq s_m/d_m \leq 1$ or $-1 \leq c_m/d_m \leq 1$ and reins the signal level effectively. The scaling compensation with the 0th butterfly is $c_0' = c_0 \prod_{m=1}^{K-1} d_m$ and $s_0' = s_0 \prod_{m=1}^{K-1} d_m$ and this does not change the implementation; the 3 multiplier/ 3 additions factoring is directly applicable. The drawback of this modified scaling is added control logic; the algorithm now contains two types of scaled rotations and in software-based computations these translate into if-conditions.
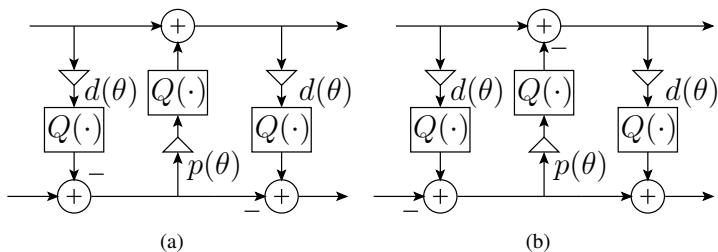
Figure 8.5: ELT rotation matrix (case 2) implemented with lifting steps including the quantization operator $Q(\cdot)$: (a) Analysis. (b) Synthesis.

8.7.2 Mapping Butterfly Matrices into Lifting Steps

The other butterfly implementation option discussed here leads in fact to a systematic way to map any invertible transform which can be represented with orthogonal rotations into an integer-to-integer transform which guarantees the perfect reconstruction property. The tedious task is just to map all 2×2 subrotations into three-step lifting lattices. In the odd-stacked fast ELT structure the butterfly matrices have all identical sign configuration. Then it sufficient to factorize single matrix

$$\mathbf{D}(\theta) = \left[\begin{array}{cc} -\cos\theta & \sin\theta \\ \sin\theta & \cos\theta \end{array} \right]$$

in our considerations and then the result is applicable for the butterfly matrix cascade. The three matrix decomposition for $\mathbf{D}(\theta)$ in (8.1) is useful but requires some tweaking. In this case the $\cot(\theta/2)$-term is the troublemaker: $\lim_{\theta \to 0^+} \cot\theta = \infty$. To keep the lifting multipliers within $[-1, 1]$ we selectively add π to the rotation angle:

$$d(\theta) = \left\{ \begin{array}{ll} \cot(\theta/2) & \text{if } (\theta/\pi - 1/2) \bmod 2 < 1 \\ \cot((\theta + \pi)/2) & \text{otherwise} \end{array} \right.$$

and

$$p(\theta) = \left\{ \begin{array}{ll} \sin(\theta) & \text{if } (\theta/\pi - 1/2) \bmod 2 < 1 \\ \sin(\theta + \pi) & \text{otherwise.} \end{array} \right.$$

The "otherwise"-condition can be compensated by reversing the sign of the data before/after the lifting steps. The other option is to write two alternative versions of the factorization for $\mathbf{D}(\theta)$ that embed the sign reversion in add/sub-operations. Depending on the value of ELT angle, we consider two cases:
Case 1: $(\theta/\pi - 1/2) \bmod 2 \geq 1$

$$\mathbf{D}(\theta) = \left[\begin{array}{cc} 1 & 0 \\ d(\theta) & -1 \end{array} \right] \left[\begin{array}{cc} -1 & p(\theta) \\ 0 & 1 \end{array} \right] \left[\begin{array}{cc} 1 & 0 \\ d(\theta) & -1 \end{array} \right]$$

Case 2: $(\theta/\pi - 1/2) \bmod 2 < 1$

$$\mathbf{D}(\theta) = \left[\begin{array}{cc} 1 & 0 \\ d(\theta) & 1 \end{array} \right] \left[\begin{array}{cc} 1 & -p(\theta) \\ 0 & 1 \end{array} \right] \left[\begin{array}{cc} 1 & 0 \\ d(\theta) & -1 \end{array} \right].$$

The ELT butterflies are symmetric matrices and thus the above lifting steps implement either analysis or synthesis side prototype filter. *But integer-to-integer realization necessitates the*

178

use of exactly reverse procedure to guarantee PR. In order to illustrate the complementary sign reversions needed, Figure 8.5 shows the rotation pair of Case 2 that meets this requirement.

Chapter 9

Conclusion

The essence of this thesis was to present a sound theoretical foundation for the biorthogonal exponentially modulated filter bank. For this purpose it was necessary to start by gathering mathematical tools for concepts such as downsampling, interpolation, aliasing, and polyphase decomposition. All of these are key elements in rigorous addressing of multirate system design problems. Especially polyphase domain analysis proved to be very effective and intuitive way to study the perfect reconstruction property of various multichannel filter banks.

Previously the odd-stacked paraunitary EMFB was incorporated into the existing FB framework by Ari Viholainen in his doctoral thesis. In this work the EMFB was generalized to biorthogonality where the analysis-synthesis reconstruction delay is an adjustable design parameter. This added functionality allows the subband processing of complex-valued signals with the EMFB having characteristics such as critical sampling, odd-stacked subchannelization, low reconstruction delay, and high selectivity. Yet proving the perfect reconstruction property was the weight on the author's shoulders.

The deduction chain on the PR of the biorthogonal EMFB was the following: First we redefined the biorthogonal odd-stacked CMFB in such a way that the modulation phase suits better for our needs to generalize the MFB concept later on. Then the polyphase domain analysis was applied to express the PR requirement in terms of the prototype filter coefficients and it was assumed that its application in optimization produces feasible results. Then the product of biorthogonal odd-stacked CMFB polyphase matrices is the pseudo-circulant matrix needed for polyphase domain PR condition. The next step was to state the SMFB equations ("the companion FB" for the CMFB) and use the subfilter connection to show that the SMFB is an invertible subband system. From the previous two the polyphase matrices of the EMFB could be constructed and this complex-valued subband system inherits the properties of its submatrices. Thus the EMFB satisfies the PR condition in both critically sampled and 2x oversampled modes, if the prototype filter is properly designed. For this reason, we made a hasty excursion into the MFB prototype filter optimization field and demonstrated the EMFB concept with design examples.

The flexibility of the EMFB provides an easy access to find the other MFB variants. When the real-part operation is left out after the analysis filters, the subband system becomes oversampled by a factor of two. This can be interpreted to provide phase information of the localized frequency component and might be advantageous feature, say, in subband equalization

problems. Secondly, the EMFB was converted into the even-stacked subchannel configuration with invertible frequency shifting scheme. When the subfilters are generated with symmetric prototype filters, the even-stacked EMFB offers linear phase subband system at our disposal. In addition, the application of the even-stacked EMFB in critically sampled subband processing of real-valued signals reduces it to the even-stacked CMFB.

The consideration of the MFB prototype filter optimization problems did not solely aim to verify the correct formulation of the design constraints. Namely, the secondary motivation was to present the transfer functions of the discussed MFB variants and ways to measure the FB realization imperfections: residual distortion and aliasing. In this context, a novel two-prototype design for 2x oversampled EMFB was introduced. Relaxation of certain constraints due to the oversampling brings extra degrees of freedom for the truly biorthogonal subfilter design. The matter was covered somewhat superficially because the two-prototype impulse response expression for the PR requirements is quite irksome. Thus we suggest for any curious souls to check [2], where one can find the design optimization routines.

High selectivity filter banks are not always the optimal choice in subband system design. If a more precise time localization becomes an issue, then the parametrized lapped transforms might be an adequate solution. It was shown how the modulation options of the EMFB can be combined with the LT window functions to define a transform family, resulting in and extended set of generalized LTs with the options for linear phase basis functions and/or suitability complex-valued signals. It was further shown how the simplest parametric MLT and ELT windows can be realized by non-decimated sliding transform algorithms that can be realized as a bank of parallel recursive filters.

In this thesis the MFB implementation algorithms were given a strong emphasis. If we consider the biorthogonal case, the straightforward procedure is to separate the prototype filter part from the polyphase matrices and start tweaking with the modulation block to extract the desired fast transform basis. The 2x oversampled even/odd-stacked EMFBs were conveniently handled with the array of complex-valued multipliers and FFT. But the critically sampled MFBs gave more challenges to try one's wit. In addition, we introduced the per-subband algorithm for even/odd-stacked biorthogonal EMFBs, where the computational burden is handled by the Goertzel-DFT.

The above discussed fast implementation structures are applicable with paraunitary prototype filter designs as well since the underlying theme was: paraunitary is a special case of biorthogonality. However, when the prototype filter length is an even multiple of the decimation factor, the fast ELT algorithm becomes available for use. The original fast ELT realized odd-stacked CMFB; here we introduced a novel even-stacked version. Due to time varying property and order recursive structures the verification of the flowgraph required some effort. But in the author's humble opinion, such steps were necessary to prove that the even-stacked fast ELT algorithm behaves as promised.

During the writing process of this thesis, the algorithms and auxiliary functions used in the development have evolved into a computing toolbox [2], which is a feat itself. More importantly, after having a solid theoretical foundation to stand on, it makes possible to envision directions to future work [15, 16]. There are a few unexplored paths in algorithm development, including PR integer-2-integer versions. The two-prototype optimization problem for 2xOS EMFBs might enjoy a different formulation using aliasing and reconstruction errors as constraints - possibly in a nearly-perfect-reconstruction sense. The oversampled EMFBs may yet prove to be useful in specialized acoustic signal processing tasks and, additionally, it would be interesting to test how the two-directional versions compare with plain 2-D CMFBs.

Bibliography

[1] S. Akkarakaran and P. P. Vaidyanathan, "Results on principal component filter banks: Colored noise suppression and existence issues," *IEEE Transactions on Information Theory*, pp. 1003–1020, Mar. 2001.

[2] J. Alhava, *Modulated filter bank and lapped transform toolbox for MATLAB*, Download: http://www.students.tut.fi/~jalhava/lt/intro.html.

[3] J. Alhava and M. Renfors, "Adaptive sine-modulated/ cosine-modulated filter bank equalizer for transmultiplexers," in *Proc. European Conf. on Circuit Theory and Design*, vol. 3, Espoo, Finland, Aug. 2001, pp. 337–340.

[4] ——, "Complex lapped transforms and modulated filter banks," in *Proc. Int. TICSP Workshop on Spectral Methods and Multirate Signal Processing*, Tolouse, France, Sept. 2002, pp. 87–94.

[5] ——, "Exponentially-modulated filter bank-based transmultiplexer," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 4, Bangkok, Thailand, May 2003, pp. 233–236.

[6] ——, "Exponentially-modulated filter bank transmultiplexer with fine-coarse adaptive filtering," in *Proc. Int. Symposium on Communications, Control and Signal Processing*, St Julians, Malta, March 2008, pp. 68–72.

[7] ——, "Prototype filter design for arbitrary delay exponentially-modulated filter bank," in *Proc. Int. Symposium on Communications, Control and Signal Processing*, St Julians, Malta, March 2008, pp. 1111 – 1114.

[8] ——, "Biorthogonal exponentially-modulated filter bank," *EURASIP Journal on Signal Processing*, vol. 97, pp. 195–206, April 2014.

[9] ——, "Recursive algorithms for modulated lapped transforms," *IEEE Journal on Circuits and Systems I*, vol. 61, pp. 191–201, Jan. 2014.

[10] ——, "Integer-to-integer complex extended lapped transform," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 8, May 2015, pp. 2612–2615.

[11] J. Alhava, A. Viholainen, and M. Renfors, "Efficient implementation of complex exponentially-modulated filter banks," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 4, Bangkok, Thailand, May 2003, pp. 157–160.

[12] F. Argenti and E. D. Re, "Design of biorthogonal m-channel cosine-modulated FIR/IIR filter banks," *IEEE Trans. Signal Process.*, vol. 48, pp. 876–881, Mar. 2000.

[13] M. G. Bellanger and J. L. Daguet, "TDM-FDM transmultiplexer: Digital polyphase and FFT," *IEEE Trans. Commun.*, vol. 22, pp. 1199–1204, Sept. 1974.

[14] J. J. Benedetto, C. Heil, and D. F. Walnut, "Gabor systems and the Balian-Low theorem," in *Gabor analysis and algorithms*. New York, USA: Springer, 1998, pp. 85–122.

[15] T. Bergman, L. Eloranta, V. Vartiainen, M. Luukko, and M. Maunula, "Takaisin työttömyyskortistoon," in *Yet unnamed 7" by Pää Kii*, 2016.

[16] T. Bergman and tyypit, "Rautamies ruostuu," in *Töölö palaa ja kuolee EP by Kytänsoittajat*. Blast of Silence, 2014.

[17] B. Boashash, Ed., *Time-Frequency Signal Analysis and Processing*, 1st ed. Elsevier, 2003.

[18] H. Bölcskei and F. Hlawatsch, "Oversampled cosine modulated filter banks with perfect reconstruction," *IEEE Trans. Circuits Syst.*, vol. 45, pp. 1057–1071, Aug. 1998.

[19] H. Bölcskei, F. Hlawatsch, and H. G. Feichtinger, "Frame-theoretic analysis of oversampled filter banks," *IEEE Trans. Signal Process.*, vol. 46, pp. 3256–3268, Dec. 1998.

[20] R. Bregović and T. Saramäki, "A systematic technique for designing linear-phase FIR prototype filter for perfect-reconstruction cosine-modulated and modified DFT filterbanks," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3193–3201, Aug. 2005.

[21] V. Britanak, "Discrete cosine and sine transforms," in *The Transform and Data Compression Handbook*, K. R. Rao, Ed. CRC Press, 2001, ch. 4.

[22] C. S. Burrus and T. W. Parks, *DFT/FFT and Convolution Algorithms: Theory and Implementation*. New York, USA: Wiley, 1985.

[23] P. Cassereau, "A new class of optimal unitary transforms for image processing," Master's thesis, Mass. Inst. Tech., Cambridge, USA, May 1985.

[24] "Information technology - digital compression and coding of continuous-tone still images - requirements and guidelines," *CCITT Rec. T.81 (1992 E)*, pp. 1–182, 1992.

[25] S. C. Chan and P. M. Yiu, "Multiplierless discrete sinusoidal and lapped transforms using sum-of-powers-of-two (SOPOT) coefficients," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. II, Sydney, Australia, May 2001, pp. 13–16.

[26] W. Chang, "High-speed multichannel data transmission with bandlimited orthogonal signals," *The Bell System Technical Journal*, pp. 1775–1796, 1966.

[27] R. Coifman and D. Donoho, "Translation-invariant de-noising," in *Wavelets and Statistics*. Springer-Verlag Lecture Notes, 1995.

[28] R. Coifman and Y. Meyer, "Remarques sur l'analyse de Fourier à fenêtre," *C. R. Acad. Sci. Paris*, vol. I, pp. 259–261, 1991.

[29] C. Creusere and S. Mitra, "A simple method for designing high-quality prototype filters for M-band pseudo QMF banks," *IEEE Trans. Signal Process.*, vol. 43, no. 4, pp. 1005–1007, Apr. 1995.

[30] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1983.

[31] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. XLI, pp. 909–996, 1988.

[32] ——, "The wavelet transform: A method for time-frequency localization," in *Advances in Spectrum Analysis and Array Processing*. Prentice-Hall, 1991, pp. 22–38.

[33] R. L. de Queiroz, T. Q. Nguyen, and K. R. Rao, "The GenLOT: Generalized linear-phase lapped orthogonal transform," *IEEE Trans. Signal Process.*, vol. 44, no. 3, pp. 497–507, Mar. 1996.

[34] P. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*. Boston, USA: Kluwer Academic Publishers, 1997.

[35] R. D. Dony, "Karhunen-loève transform," in *The Transform and Data Compression Handbook*, K. R. Rao, Ed. CRC Press, 2001, ch. 1.

[36] D. Esteban and C. Galand, "Application of quadrature mirror filters to split-band voice coding schemes," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Hartford, CT, USA, May 1977, pp. 191–195.

[37] B. Farhang-Boroujeny, "OFDM versus filter bank multicarrier," *IEEE Signal Processing Magazine*, vol. 28, pp. 92–112, May 2011.

[38] N. J. Fliege, "Computational efficiency of MDFT filter banks," in *Proc. 27th Asilomar Conference on Signals, Systems, and Computers*, Asilomar, CA, USA, 1993.

[39] Y.-T. Fong and C. Kok, "Iterative least squares design of DC-leakage free paraunitary cosine-modulated filter banks," *IEEE Trans. Circuits Syst. II*, vol. 50, pp. 238–243, May 2003.

[40] M. B. Furtado, P. S. R. Diniz, and S. L. Netto, "Numerically efficient optimal design of cosine-modulated filter banks with peak-constrained least-squares behavior," *IEEE Trans. Circuits Syst.*, vol. 52, pp. 597–608, Mar. 2005.

[41] D. Gabor, "Theory of communication," *J. IEE*, vol. 93, no. III, pp. 429–457, 1946.

[42] P. N. Heller, T. Karp, and T. Q. Nguyen, "A general formulation of modulated filter banks," *IEEE Trans. Signal Process.*, vol. 47, no. 4, pp. 986–1002, Apr. 1999.

[43] A. K. Jain, *Fundamentals of digital image processing*. Eaglewood Cliffs, NJ, USA: Prentice-Hall, 1989.

[44] J. D. Johnston, "A filter family designed for use in quadrature mirror filter banks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Denver, CO, USA, Apr. 1980, pp. 291–294.

[45] H. Karhunen, "Über lineare methoden in der wahrscheinlich-keitsrechnung," *Ann. Acad. Science Fenn.*, vol. Ser. A.I., no. 37, 1947.

[46] T. Karp and N. J. Fliege, "Modified DFT filter banks with perfect reconstruction," *IEEE Trans. Circuits Syst. II*, vol. 46, pp. 1404–1414, Nov. 1999.

[47] T. Karp, A. Mertins, and G. Schuller, "Recent trends in the design of biorthogonal modulated filter banks," in *Proc. of TICSP Workshop on Transforms and Filter Banks*, Tampere, FINLAND, 1998.

[48] R. D. Koilpillai and P. P. Vaidyanathan, "Cosine-modulated FIR filter banks satisfying perfect reconstruction," *IEEE Trans. Signal Process.*, vol. 40, pp. 770–783, Apr. 1992.

[49] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus, and R. O. Wells, "Nonlinear processing of a shift-invariant DWT for noise reduction," in *Proc. of SPIE*, vol. 2491, 1995, pp. 640–651.

[50] Z. Lin, J. Chen, and R. W. McCallum, "Transform domain adaptive filtering using recursive running DCT and its application in surface recording of small intestine," in *Proc. IEEE Southeastcon '93*, Charlotte, NC, USA, April 1993.

[51] Q.-G. Liu, B. Champagne, and D. K. C. Ho, "Simple design of oversampled uniform DFT filter banks with applications to subband acoustic echo cancellation," *Signal Processing*, vol. 80, pp. 831–847, May 2000.

[52] M. Loève, "Fonctions aléatoires de seconde ordre," in *Processus Stochastiques et Mouvement Brownien*. Paris, France: Hermann, 1948.

[53] Y. Luo, S. C. Chan, and K. L. Ho, "Theory and design of arbitrary-length biorthogonal cosine-modulated filter banks," in *Proc. IEEE Int. Symp. on Circuits and Systems*, Hong Kong, June 1997, pp. 2429–2432.

[54] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[55] ——, *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

[56] H. S. Malvar, "Extended lapped transforms: Properties, applications, and fast algorithms," *IEEE Trans. Signal Process.*, vol. 40, pp. 2703–2714, Nov. 1992.

[57] ——, *Signal Processing with Lapped Transforms*. Norwood, MA, USA: Artech House, 1992.

[58] ——, "Biorthogonal and nonuniform lapped transforms for transform coding with reduced blocking and ringing artifacts," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 1043–1053, Apr. 1998.

[59] ——, "A modulated complex lapped transform and its applications to audio processing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1999, pp. 1421–1424.

[60] H. S. Malvar and D. H. Staelin, "The LOT: Transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, pp. 553–559, Apr. 1989.

[61] J. S. Mao, S. C. Chan, and K. L. Ho, "Theory and design of causal stable IIR PR cosine-modulated filter banks," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 3, Orlando, USA, May 1999, pp. 427–430.

[62] MATLAB, *Optimization toolbox, version 6.4*, `http://mathworks.com`, 2013.

[63] A. Mertins, "Subspace approach for the design of cosine-modulated filter banks with linear-phase prototype filter," *IEEE Trans. Signal Process.*, vol. 46, no. 10, pp. 2812–2818, Oct. 1998.

[64] A. Mertins, T. Karp, and J. Kliewer, "Design of perfect reconstruction integer modulated filter banks," in *Proc. IEEE Symp. Signal Processing and Its Applications*, Brisbane, Australia, Aug. 1999, pp. 591–594.

[65] T. Q. Nguyen, "A class of generalized cosine-modulated filter bank," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 2, San Diego, CA, USA, May 1992, pp. 943–946.

[66] ——, "Near-perfect-reconstruction pseudo-QMF banks," *IEEE Trans. Signal Process.*, vol. 42, no. 1, pp. 65–76, Jan. 1994.

[67] ——, "Digital filter bank design quadratic constrained formulation," *IEEE Trans. Signal Process.*, vol. 43, no. 9, pp. 2103–2108, Sept. 1995.

[68] T. Q. Nguyen and R. D. Koilpillai, "The theory and design of arbitrary-length cosine-modulated filter banks and wavelets, satisfying perfect reconstruction," *IEEE Trans. Signal Process.*, vol. 44, pp. 473–483, Mar. 1996.

[69] T. Q. Nguyen, T. I. Laakso, and T. E. Tuncer, "On perfect-reconstruction allpass-based cosine-modulated IIR filter banks," in *Proc. IEEE Int. Symp. on Circuits and Systems*, London, UK, May 1994, pp. 33–36.

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

[71] S. S. Pradhan and V. U. Reddy, "A new approach to subband adaptive filtering," *IEEE Trans. Signal Process.*, vol. 47, pp. 655–664, Mar. 1999.

[72] J. P. Princen and A. B. Bradley, "Analysis/synthesis filter bank design based on time-domain aliasing cancellation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 5, pp. 1153–1161, Oct. 1986.

[73] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[74] T. A. Ramstad and J. P. Tanem, "Cosine-modulated analysis-synthesis filterbank with critical sampling and perfect reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 3, Toronto, ON, CANADA, Apr. 1992, pp. 1789–1792.

[75] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video, and Audio Coding*. Englewood Cliffs, New Jersey, USA: Prentice-Hall, 1996.

[76] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*. San Diego, CA, USA: Academic, 1990.

[77] J. H. Rothweiler, "Polyphase quadrature filters – a new subband coding technique," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Boston, MA, USA, Apr. 1983, pp. 1280–1283.

[78] B. R. Saltzberg, "Performance of an efficient parallel data transmission system," *IEEE Transactions on Communications Technology*, no. 6, pp. 805–811, 1967.

[79] S. D. Sandberg and M. A. Tzannes, "Overlapped discrete multitone modulation for high speed copper wire communications," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 9, pp. 1571–1585, 1995.

[80] T. Saramäki, "Designing prototype filters for perfect-reconstruction cosine-modulated filter banks," in *Proc. IEEE Int. Symp. on Circuits and Systems*, San Diego, USA, May 1992, pp. 1605–1608.

[81] ——, "Finite impulse reponse filter design," in *Digital Signal Processing Handbook*. New York, USA: John Wiley and Sons, 1993, ch. 4, pp. 155–277.

[82] ——, "A generalized class of cosine-modulated filter banks," in *Proc. Int. Workshop on Transforms and Filter Banks*, Tampere, Finland, Feb. 1998, pp. 336–365.

[83] ——, *Part V: Multirate Filter Banks* in the lecture note for the TUT course entitled 80558 Multirate Signal Processing, Download: http://www.cs.tut.fi/˜ts/multi50.pdf, multi51.pdf, multi52.pdf, partv3.pdf, and book01_cor.pdf, 2000.

[84] T. Saramäki and R. Bregović, "An efficient approach for designing nearly perfect-reconstruction cosine-modulated and modified DFT filter banks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. VI, Salt Lake City, USA, May 2001, pp. 3617–3620.

[85] G. D. T. Schuller and T. Karp, "Modulated filter banks with arbitrary system delay: Efficient implementations and the time-varying case," *IEEE Trans. Signal Process.*, vol. 48, no. 3, pp. 737–748, Mar. 2000.

[86] C. Siclet and P. Siohan, "Design of OFDM/OQAM systems based on biorthogonal modulated filter banks," in *Proc. IEEE Global Communications Conference*, San Francisco, CA, Dec. 2000, pp. 701–705.

[87] P. Siohan, C. Siclet, and N. Lacaille, "Analysis and design of OFDM/OQAM systems based on filterbank theory," *IEEE Trans. Signal Process.*, vol. 50, no. 5, pp. 1170 – 1183, May 2002.

[88] M. J. T. Smith and T. P. Barnwell III, "Exact reconstruction for tree-structured subband coders," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 3, pp. 314–327, Mar. 1987.

[89] T. Springer, "Sliding FFT computes frequency spectra in real time," *EDN Magazine*, pp. 161–170, Sept. 1988.

[90] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Wellesley, MA, USA: Wellesley-Cambridge Press, 1996.

[91] J. Tsui and C.-H. Cheng, *Digital Techniques for Wideband Receivers*, 3rd ed. Artech House, 1995.

[92] P. P. Vaidyanathan, "Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial," *Proc. of the IEEE*, vol. 78, pp. 56–93, Jan. 1990.

[93] ——, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.

[94] P. P. Vaidyanathan and P.-Q. Hoang, "Lattice structures for optimal design and robust implementation of two-channel perfect reconstruction filter banks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 1, pp. 81–94, Jan. 1988.

[95] P. P. Vaidyanathan and P. Q. Hoang, "Lattice structures for optimal design and robust implementation of two-channel perfect reconstruction QMF banks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, pp. 81–94, Jan. 1988.

[96] R. G. Vaughan, N. L. Scott, , and D. R. White, "The theory of bandpass sampling," *IEEE Trans. Signal Process.*, vol. 39, pp. 1973 – 1984, Sept. 1991.

[97] M. Vetterli and Z. Cvetković, "Oversampled FIR filter banks and frames in $\ell^2(\mathbb{Z})$," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 3, May 1996, pp. 1530–1533.

[98] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*. Englewood Cliffs, New Jersey, USA: Prentice Hall PTR, 1995.

[99] A. Viholainen, "Modulated filter bank design for communications signal processing," Ph.D. dissertation, Tampere University of Technology, Tampere, Finland, 2004.

[100] A. Viholainen, J. Alhava, and M. Renfors, "Efficient implementation of 2x oversampled exponentially-modulated filter banks," *IEEE Trans. Circuits Syst.*, vol. 53, pp. 1138–1142, Oct. 2006.

[101] ——, "Efficient implementation of complex modulated filter banks using cosine and sine modulated filter banks," *EURASIP journal on advances in signal processing*, April 2006.

[102] J. E. Volder, "The CORDIC trigonometric computing techniques," *IRE Trans. Electron. Comput.*, vol. 8, pp. 330–334, Sept. 1959.

[103] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing by using the discrete Fourier transform," *IEEE Trans. Commun.*, vol. 19, pp. 628–634, Oct. 1971.

[104] J. W. Woods and P. O'Neil, "Subband coding of images," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 1278–1288, Oct. 1986.

[105] H. Xu, W.-S. Lu, and A. Antoniou, "Efficient iterative design method for cosine-modulated QMF banks," *IEEE Trans. Signal Process.*, vol. 44, no. 7, pp. 1657–1668, July 1996.

[106] Y. Zeng, L. Cheng, G. Bi, and A. C. Kot, "Integer DCTs and fast algorithms," *IEEE Trans. Signal Process.*, vol. 49, no. 11, pp. 2774–2782, Nov. 2001.