Perfect Reconstruction Filter Banks Based on Lifting and Other Approaches

Zhang Lei

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2007

Abstract

Perfect reconstruction filter banks have found applications in a large number of topics in modern digital signal processing theory. This work is an endeavor towards better understanding the nature of perfect reconstruction filter banks both from 2-channel to M-channel, and both from one dimension to multiple dimension.

In the case of one dimension, we first investigate 2-channel linear phase perfect reconstruction filter banks (LPPRFB). We propose different approaches to factorize LPPRFBs of different natures. To tackle a class of singular FBs, we also propose a lifting-like structure. To show the effectiveness of the proposed factorizations and designs, a number of examples are presented. The extension of the classic lifting scheme is discussed as well.

The idea of lifting factorization is extended to M-channel case. After studying the lifting building blocks for M-channel case, we demonstrate the factorization of M-channel PRFB, which is based on the extended Euclidean algorithm. The lifting-based design is applicable for both general PRFB and those with linear phase.

The scope is changed to multiple dimensions (nD) then. We briefly review the essential concepts in nD systems. For nD systems, the variety of downsampling matrices is a significant difference from 1D system. It is interesting to ask whether for a given downsampling ratio, the number of different downsampling patterns is infinite. We investigate this issue and enumerate as well as parameterize all such patterns. The generation of alias-free ideal filters is studied.

nD problems can be translated into the language of algebraic geometry. The sound foundation of algebraic geometry provides us many powerful tools to address nD problems, such as Groebner basis and resultants. By using these tools, we study the relations between various zeros and various polynomial invertibilities. This leads us to some insights, especially when linear phase is under consideration. We disprove an approach to construct PRFB in the literature based on our derived results on resultant. The way to characterizing nD LPPRFB and to checking whether a 1D-to-nD transform can retain PR is also studied.

We then narrow our focus to nD 2-channel LPPRFB. Based on the study of their nature, a classification is presented. For one type, we propose a completion approach based on Groebner basis. For the other type, the lifting scheme is used for its factorization via a proposed conjecture. Some design examples for this type of LPPRFB are also presented.

Acknowledgments

I would like to thank my supervisor, Dr. Anamitra Makur, for his continuing support and guidance during my candidature period. The discussions and meetings with him are always inspiring. It is his encouragement and timely help that keep me working towards the completion of this thesis. Aside from being an excellent supervisor, he is also a sincere and hospitable friend.

I would like to express my gratitude towards School of Electrical and Electronics Engineering, Nanyang Technological University (NTU) for providing me the financial support as well as a convenient environment that facilitate my research very much. Without them, almost nothing is possible.

Of course, I also appreciate those who ever helped me during these years. Thank Dr. Lin Zhiping for his fruitful discussion and illumination in multidimensional system. Thank Dr. Alex Kot for his help and encouragement. Thank Dr. Zhu Ce, Dr. Chau Lap-Pui for their help during conference. I am grateful to my friends and colleagues at NTU, who I learned a lot from and I spent wonderful years with.

Finally, I would give my heartiest appreciation to my parents and Wei Juanjuan for their constant love and unconditional support. Without them, absolutely nothing is possible.

Contents

List of Figures

List of Tables

List of Abbreviations

deglex	Degree lexicographical ordering
degrevlex	Degree reverse lexicographical ordering
DOF	Degree of freedom
DSP	Digital signal processing
EEA	Extended Euclidean algorithm
FB	Filter bank
FI	FIR invertible
FIR	Finite impulse response
FPD	Fundamental parallelepiped
GB	Groebner basis
gcd	the greatest common divisor
GLS	Generalized lifting scheme
GMT	Generalized McClellan transform
HVS	Human visual system
IIR	Infinite impulse response
LHS	Left hand side
LP	Linear phase
LS	Lifting scheme
MFR	Magnitude frequency responses
MIP	Mirror image pair
PI	Polynomial invertible
PR	Perfect reconstruction
PU	Paraunitary

PSNR	Peak signal-to-noise ratio
PSO	Particle swarm optimization
RHS	Right hand side
SPT	Signed power of two
VM	Vanishing moment

Chapter 1

Introduction

Multirate digital signal processing (DSP) becomes increasingly important with the development of the multimedia and internet. The central idea in this field is to analyze a signal at different spectral components, which can be achieved by decomposing the signal into different subbands of frequency. This is useful because most applications related to human race, who are more sensitive to some frequencies than the others due to the features of the human sensory system. This provides the possibility to effectively utilize the limited resource of implementation and transportation. For example, in a coding application, these important frequencies can be coded with a finer resolution, while using a coarser resolution for the others. It is already known that the above approach can achieve a better compression ratio than equally allocating the resolution. A filter bank (FB) is just such a device serving to isolate different frequency components in a signal. Furthermore, FBs have close connection with many popular time-frequency representation tools, e.g., block transform, wavelet transform and lapped transform. Therefore, FBs have found applications in almost every area of modern DSP, including image and video compression, digital audio processing, adaptive and statistical signal processing, and communications [1–4]. For further details on history, applications of multirate systems and FBs etc., the interested readers may refer to the following excellent textbooks on the topic [3–6].

The present thesis aims to investigate the design of FB that has the perfect reconstruction property, especially through a popularly-used structure, i.e. the lifting scheme (LS). The motivation and objective is explained firstly, followed by a literature review. Then the organization and the contributions of subsequent chapters is outlined and the notations to be used later are established. To make the thesis self-contained, the fundamentals that are useful in the following part of thesis are briefly reviewed in the second chapter, which includes the essential principles of multirate systems and FBs, the lifting scheme, multidimensional systems and signals and so on.

1.1 Motivation and objectives

Structures, like lattices and ladders (popularly known as lifting structure now), have an important influence on image compression, or other subband coding. For the same wavelet or filter bank, the performance (e.g. robustness to quantization, computational complexity) of implementing with different structures generally varies from structure to structure, sometimes even drastically. This is not difficult to understand, since different structures have different properties, which are suitable for different situations.

Lattice, as a structure emerged for tens of years, has been studied from various aspects and it is famous for its usage in fast Fourier transform. Though studied for a long time, new results on lattices keep coming out even nowadays. Lifting, while of much younger age than lattices, is also a structure of this kind, with proved advantages in efficient computation, in-place implementation, insensitivity to quantization, and other potential advantages yet to be explored. This work is an effort towards better understanding the nature of lifting, which is basically the motivation.

The LS, earlier known as ladder structure [7–9], was introduced in a systematic manner by Sweldens in [10] and applied to factorize the wavelet transform into lifting step in [11]. Since the scheme provides above mentioned advantages, it has been widely applied and is even adopted into the JPEG2000 standard. Despite the fact that the Euclidean algorithm in [11] can solves all PR FBs (complementary pairs, as addressed in [11]), when it comes to a particular context, for example, a linear phase perfect reconstruction filter bank, following the solution as in [11] do not necessarily lead to a factorization that is optimal in some sense, as demonstrated later. To find such an optimal factorization is one of our objectives. Some singular cases that are noted in literature also need to be taken care of.

M-channel PRFB is also of interest due to subband coding. In [8, 9, 12–14], the nature of M-channel LPPRFB has been studied thoroughly. Based on these results, LS is considered for multichannel case as well. By exploiting the symmetry of the construction matrix, a lifting-based approach of extending the linear phase (LP) property is obtained.

In [8,9], the LS was further considered for multivariate case but with a relatively simple setting. These only consider the construction side, which is similar to that for M-channel LPPRFB. This motivates us to explore the problems of factorization in multidimensional (nD) PRFB.

Actually, using lifting for the design of multidimensional filter is not difficult to find in the literature, though the LS is less popular in nD than in 1D. For example, a novel design of filter banks with hourglass-shaped passband support has been presented in [15]. However, to the best of our knowledge, there is no work on lifting-based factorization of nD LPPRFB. The lack of lifting factorization in nD case motivates us to investigate its feasibility.

nD problem is quite different from 1D problem due to factors like resampling matrices, nature of multivariate polynomials, etc. Therefore, before we solve the above lifting based factorization problems, we need to further understand those differences, including multidimensional resampling and the relationship between zeros and PR or LPPR properties.

Considering resampling, there exists a good knowledge about multidimensional resampling in the literature [16–22]. In the existing literature, although the fact that different resamplings may correspond to the same pattern is noted, enumeration and parametrization of distinct patterns are not addressed systematically. This motivates us to investigate the nature of resampling matrices and to enumerate and parameterize them in a systematic way.

Considering zeros in nD, their difference from 1D zeros has been noted in [23,24]. Zeros are no longer equivalent to common factors as in 1D case. Requirements of PR property need to change accordingly. To establish the relations between zeros and PRFB or LPPRFB naturally becomes an objective of this work. We derive some results by using powerful tools from algebraic geometry, such as Groebner basis and resultants.

In the above, the motivation and objective of the thesis has been briefly explained. Most of them are more or less related to the lifting structure. However, during the process of seeking the solution to these lifting related problems, we do obtain some other novel and interesting results that are not directly related to the LS. These, too, are included into this work.

1.2 Organization and contributions of the thesis

As explained, the work in this thesis is motivated by the excellent features provided by lifting scheme. We try to establish an extensive relation between the studied filter banks and lifting schemes. To achieve this objective, we organize the thesis as follows, where the contributions are highlighted as well.

In Chapter 2, to make the thesis self-contained, some fundamentals used extensively in the following chapters are reviewed. Brief reviews cover basic principles of 1D and nD filter banks, different polynomials, lifting structure and so on.

In Chapter 3, we study the relation between lifting structure and 2-channel LPPRFB. Firstly, we review the factorization of 2-channel PRFB based on lifting scheme and limitations of the factorization when LP is considered. To understand the problems well, the classification of 2-channel LPPRFB is introduced. Since lifting structure can impose PR property, we want to extend it to the LP property. Due to different natures of the only two possible types of LPPRFBs, we use different approaches to study their relation with LP and lifting. Some novel results on the LPPRFB factorization and construction are presented. Specifically, for a class of singular LPPRFBs, a lifting-like structure is proposed. We also give the necessary theoretical proof there. The theoretic results are illustrated by a number of examples, including factorization of existing filter banks from MATLAB and literature, and further applying them in image compression to evaluate the advantage of the lifting based factorization. The scope of lifting scheme is generalized towards the end of the chapter. This generalization brings insight into this scheme and is beneficial to further investigation.

In Chapter 4, we extend the lifting scheme to M-channel PRFB, in both aspects of factorization and construction. Firstly, we will uncover the mathematical essence of the M-channel lifting, and give the description of lifting building blocks from the matrix viewpoint. We also give the proof for the existence of lifting factorization in M-channel case after reviewing some existing approaches. Results of the 4-channel PRFB constructed by lifting are also presented. Finally, we narrow our focus to M-channel LPPRFB and propose an approach to construct it.

From Chapter 5 onwards, we extend our interest to multidimensional (nD) FBs. In Chapter 4, we first review and classify the existing works in literature on nD FBs. We also review some fundamentals that distinguish nD problems from 1D problems. One essential difference is the manner of resampling in nD. Whether an infinite number of possible resampling matrices correspond to a finite number of resampling methods is of interest. This is addressed for 2D case in Chapter 5.3, where some positive results are obtained. Another difference is the choice of alias-free filter for a resampling matrix, which is less complicated in 1D. We propose an approach to obtain an ideal filter that is alias-free for a given resampling matrix.

In Chapter 6, we relate the nD PRFB to the problems of polynomial invertibility in algebraic geometry. The first step to this goal is to investigate the relation between types of zeros in nD and various nD PRFB. For LPPRFB, a stronger condition is obtained, which can be translated in the language of results as done in the succeeding section. Resultant is a powerful tool to find common zeros of given polynomials. Based on the resultants we disprove the argument on constructional approach from a well cited work. Being another important tool in algebraic geometry, Groebner basis deserves a brief introduction. Applications of Groebner basis in nD PRFB are also reviewed. A following approach is based on this. We apply the result on zeros to McClellan transform and its extension and reach the same conclusion that is drawn in other works. This provides another novel angle to consider how to secure PR property when using 1D-to-nD transform.

In Chapter 7, we specifically investigate the factorization and design of nD 2-channel LPPRFB. Similar to 1D, we first classify FBs according to the symmetry natures into two types. The classification is elaborated for quincunx downsampled 2-channel LPPRFB. For PR completion, we propose a symmetry extension for one type while the other type can be taken care of similarly by the approach proposed in another work. For factorization, by combining the result from Chapter 5 and a conjecture, the algorithm to factorize a type of 2-channel LPPRFB is proposed. We prove the conjecture in some special cases. The lifting based structure is also applied for the design of this type of FBs and examples are presented. For other type of FBs, the existing design approach is reviewed. We also prove the completeness of this structure under special conditions.

In the last chapter, the results obtained and presented in this thesis are summarized. The possible extensions for future research of the presented work and some open problems are addressed there.

1.3 Notation

Bold-faced quantities denote matrices and vectors, as in $\mathbf{A}$, $\mathbf{H}(z)$, $X(\mathbf{z})$ etc. Sometimes they will be followed by some subscripts which denote their dimensions. For example, $\mathbf{A}_{m\times n}$ denotes a matrix with m rows and n columns. The subscripts will be omitted if they are clear from the context. The superscript T means transposition, e.g. $\mathbf{A}^T$.

For the power of a vector, this notation is followed,

$$\mathbf{z}^{\mathbf{n}} = z_0^{n_0} z_1^{n_1} \cdots z_{D-1}^{n_{D-1}},$$

where $\mathbf{z} = [z_0, \cdots, z_{D-1}]^T$ and $\mathbf{n} = [n_0, \cdots, n_{D-1}]^T$. $\mathbf{z}^{-1} = z_0^{-1} z_1^{-1} \cdots z_{D-1}^{-1}$ is also used if no ambiguity is caused.

As mentioned, $S(H(\mathbf{z}))$ denotes the size of the polynomial $H(\mathbf{z})$ along each dimension, or $L(H)$ for 1D case.

The proofs for the theorems are provided in detail unless they are available elsewhere.

Chapter 2

Overview

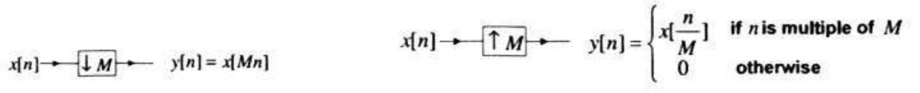
This chapter reviews some basic definitions and important concepts that are used extensively throughout the thesis.

2.1 Filter bank and its polyphase representation

A filter bank is a collection of band-pass filters that separate an input signal into several spectral components, each one carrying a single frequency subband of the original signal.

The essential components for a FB are resamplers, namely downsampler and upsampler. In different literature, different names may be used, such as decimator or subsampler for downsampler and interpolator or expander for upsampler. Whatever their names are, they can be depicted as Figure 2.1, where the relation between input and output signals are also described.

Generally, a filter bank is comprised of an analysis bank and a synthesis bank. The former one is a set of analysis filters $H_k(z)$ which splits a signal into M subband signals $x_k[n]$ as shown in Figure 2.2. These are followed by m_k-fold downsampler that retains 1 sample of every m_k input samples. According to the application, an appropriate processing will be performed to these subband signals. A synthesis bank consists of M synthesis filters $F_k(z)$, which combines M subband signals. These are preceded by m_k-fold expander that introduces $m_k - 1$ zero samples between every pair of input samples. Figure 2.3 shows the frequency response of a typical analysis bank in a uniform FB.



(a) M-fold downsampler (b) M-fold upsampler

Figure 2.1: Resamplers

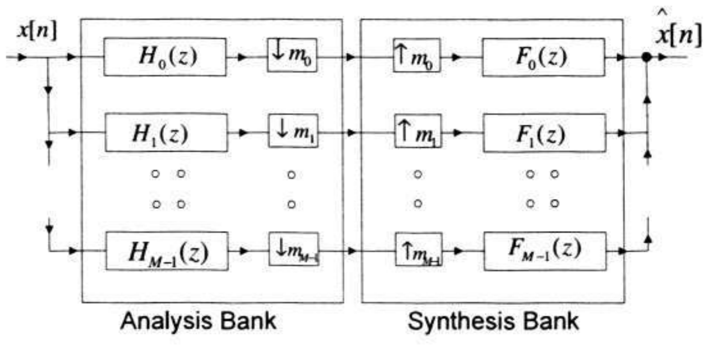


Figure 2.2: Analysis and synthesis filter banks

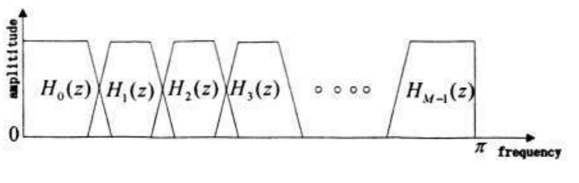


Figure 2.3: Frequency response of an analysis filter bank

Consider all $m_k = M$. The analysis filters can always be represented in a polyphase manner as follows:

$$H_i(z) = \sum_{k=0}^{M-1} z^{-k} E_{ik}(z^M) \quad i = 0 \cdots M-1 \tag{2.1}$$

Similarly, we can express the synthesis filters as:

$$F_i(z) = \sum_{k=0}^{M-1} z^{-(M-1-k)} R_{ik}(z^M) \quad i = 0 \cdots M-1 \tag{2.2}$$

Using such representation, we can rewrite the analysis filters as well as the synthesis filters.

$$\mathbf{H}(z) \triangleq [H_0(z), H_1(z), \cdots, H_{M-1}(z)]^T = \mathbf{E}(z^M)\mathbf{d}(z) \tag{2.3}$$

$$\mathbf{F}(z) \triangleq [F_0(z), F_1(z), \cdots, F_{M-1}(z)] = \widetilde{\mathbf{d}(z)}\mathbf{R}(z^M) \tag{2.4}$$

where $\mathbf{d}(z) = [1, z^{-1}, z^{-2}, \cdots, z^{M-1}]^T$, $\widetilde{\mathbf{d}(z)} = [z^{M-1}, \cdots, z^{-2}, z^{-1}, 1]$, $\{\mathbf{E}(z)\}_{ij} = E_{ij}(z)$ and $\{\mathbf{R}(z)\}_{ij} = R_{ji}(z)$. By exploiting noble identities [3] of Figure 2.4, an

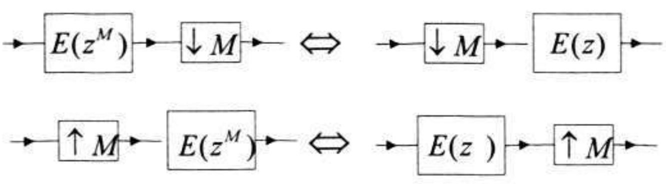


Figure 2.4: Noble identities

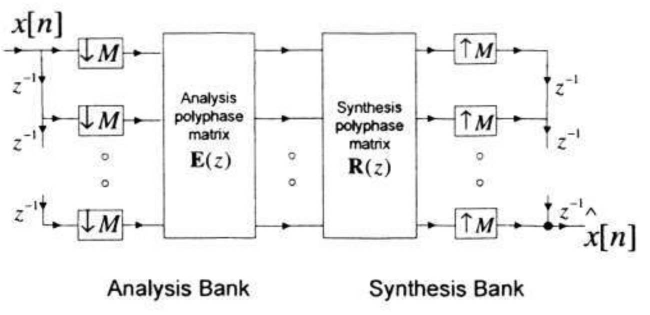


Figure 2.5: The polyphase representation of the filter bank

efficient representation of FBs can be obtained in polyphase domain as shown in Figure 2.5. Polyphase matrix is a significant invention and important advancement in multirate signal processing. This permits great simplification of theoretical results, because the polyphase representation allows us to study the FB via matrices rather than individual filters. Polyphase representation also brings implementation efficiency of FB, since the polyphase filters are operating at a lower rate than filters in Figure 2.2. However, one should be aware that the implementation advantage only exists when a high speed operation is more expensive than the equivalent low speed operation.

2.2 The classification of filter banks

Filter banks can be categorized according to different features, such as downsampling ratio, properties of polyphase matrix, filter types etc. Usually, research is conducted within a certain class of FBs. Hence the classification is introduced here.

If the downsampling ratios are identical for each channel, the system is named as uniform FB, and otherwise as non-uniform FB. In literature, most works deal with a uniform FB. Even for non-uniform FB, a class of this kind is achieved by tree-cascaded uniform FBs. In this thesis, we also focus on uniform FB. We assume that the number of channels is N and the downsampling ratio is M. The FBs are classified as undersampled if $N < M$, maximally sampled if $N = M$, or oversampled FBs if $N > M$.

The reconstructed signal $\hat{x}[n]$, in general, suffers from aliasing error and amplitude and phase distortion due to the fact that all the involved filters are not ideal. If we choose the filters that satisfy

$$\mathbf{E}(z)\mathbf{R}(z) = c\mathbf{I}z^{-n_0}, \tag{2.5}$$

we can prove that [3] the reconstructed signal is therefore a scaled and time-delayed version of the original signal $x[n]$. This property is the so-called perfect reconstruction (PR), and the corresponding filter bank is called Perfect Reconstruction FB

(PRFB) or biorthogonal FB. Obviously, $\mathbf{E}(z)$ and $\mathbf{R}(z)$ are not necessarily square matrices. For example, if dealing with oversampled system, both $\mathbf{E}(z)$ and $\mathbf{R}(z)$ are rectangular matrices. However, when considering maximally sampled system, which is of major interest in this thesis, they are indeed square matrices.

If we focus on filter banks with FIR filter, it is necessary and sufficient for such a PRFB to have an analysis polyphase matrix with a special determinant as follows:

$$det(\mathbf{E}(z)) = cz^{-r} \tag{2.6}$$

The above condition will be referred as PR condition in future.

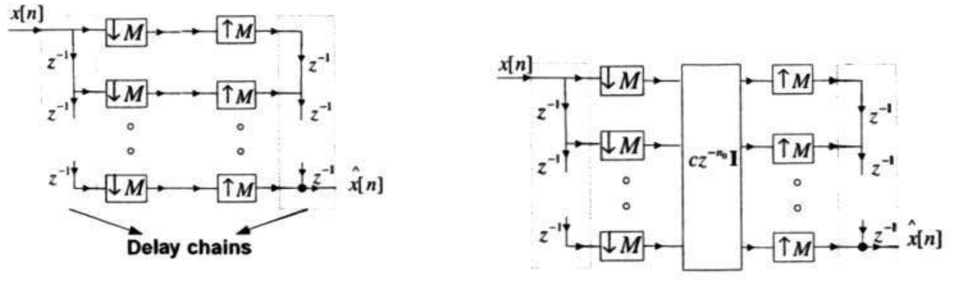
Since PR condition is significant in most applications of FBs, it deserves another explanation from a viewpoint of delay chain. A delay chain system can be regarded as a trivial FB as depicted in Figure 2.6(a). Obviously, it is PR since signals are kept intact except being delayed. Hence it is still PR when we insert more delays to each channel as done in Figure 2.6(b). That is why the PR condition appears in the form of (2.5). In fact, due to the periodicity of the resamplers, the PR condition can be relaxed as [3]

$$\mathbf{E}(z)\mathbf{R}(z) = cz^{-n_0} \begin{bmatrix} \mathbf{0} & \mathbf{I}_{M-r} \\ z^{-1}\mathbf{I}_r & \mathbf{0} \end{bmatrix} \tag{2.7}$$

for some r in $0 \leq r \leq M-1$. The PR condition for an FIR analysis bank in (2.6) is still valid.

Furthermore, if matrices $\mathbf{E}(z)$ and $\mathbf{R}(z)$ are paraunitary, namely $\mathbf{E}(z) = \mathbf{R}^{\mathbf{H}}(z^{-1}) = \mathbf{R}^{T*}(z^{-1})$, the corresponding FB is called Paraunitary FB (PUFB) or orthogonal FB.

Due to the presence of quantization error and the viewer's sensitivity to the edge of an object in an image, linear phase (LP) filters are often desirable. This brings us LPFB, which is quite useful in image and video processing and some other applications. They are featured by all filters having LP property. Strictly speaking,



(a) (b)

Figure 2.6: Delay chain systems

a digital filter is said to have LP if the phase response $\phi(\omega)$ is linear in ω. In z-domain, the LP property requires a filter, say $H(z)$, to satisfy

$$H(z) = \pm z^{-c} H^*(z^{-1}), \tag{2.8}$$

for some integer c. LP can only be exactly achieved by finite impulse response (FIR) filters. Hence LP property also translates into symmetry or antisymmetry of filter coefficients in time (spatial) domain. When dealing with real coefficients, we can omit the conjugation. Eq. (2.8) is valid even for multivariate case with proper choice of a variable as the main variable. This also corresponds to symmetry or antisymmetry of coefficients in higher dimensional space.

According to the filter types, FBs are categorized as FIR FBs, IIR FBs, and hybrid FBs. It is observed that a large part of the works in literature deal with FBs that use FIR filters. This is because this category is of great interest in practice. The FBs studied in this thesis are also mostly FIR FBs.

2.3 Lifting structure

Lifting structure has two basic forms, as depicted in Figure 2.7. Most of the time, a dual lifting step is needed to make the system sophisticated enough. A dual lifting

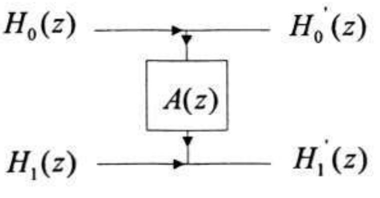


Figure 2.7: A lifting structure

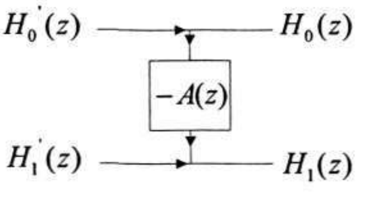


Figure 2.8: The inverse structure of lifting

is simply another lifting step yet with an opposite lifting direction. Here for the sake of brevity, only a lifting step is depicted. The relation between the input and output is easy to derive.

$$\begin{cases} H_0'(z) = H_0(z) \\ H_1'(z) = H_0(z)A(z) + H_1(z) \end{cases} \tag{2.9}$$

The attraction of lifting scheme is mostly relied on its simple inverse, which can be obtained simply by mirror-flipping the structure and multiplying the building blocks with -1, as depicted in Figure 2.8. By doing so, we can reconstruct the original signal perfectly.

The advantages of lifting scheme [11] includes allowing a faster implementation of the wavelet transform, and allowing a fully in-place calculation of the wavelet transform. Nevertheless, it provides the feature that the synthesis bank is nothing but the mirror-flipping and sign-changed version of analysis bank. This greatly facilitates the design of a whole FB system. The factorization of any PRFB into lifting steps is based on the Euclidean algorithm which deals with the division of two univariate polynomials. However, when applying Euclidean algorithm, the choice of divisor polynomials is highly non-unique. To seek a good strategy for picking the best lifting step is hence a difficult job [11]. This partially motivates our work in this thesis.

2.4 Polynomials and Laurent polynomials

A polynomial is a mathematical expression involving a sum of powers in one or more variables multiplied by coefficients. A univariate polynomial, i.e. a polynomial in one variable, is given by

$$F(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} \cdots a_n z^{-n} \tag{2.10}$$

According to the field where operations, such as addition, subtraction and so on, for coefficients are defined, the polynomial also belongs to the corresponding field. For example, if the coefficients a_i are defined over the field of rational numbers $\mathbb{Q}$, real numbers $\mathbb{R}$ or complex numbers $\mathbb{C}$, we denote the polynomials as $F(z) \in \mathbb{Q}[z]$, $F(z) \in \mathbb{R}[z]$ or $F(z) \in \mathbb{C}[z]$ respectively. Sometimes, an arbitrary field $K[z]$ may be used for convenience.

Here we further restrict the scope the polynomials to the ones with $a_0 \neq 0$ and $a_n \neq 0$. The highest power of a polynomial is called its order, or sometimes its degree. Usually, z^{-1} instead of z is chosen as the main variable for polynomials in z-domain. Therefore, the degree of the polynomial $F(z)$ is denoted as $deg(F(z)) = n$.

Cases with $a_0 = 0$ or polynomials having both positive and negative powers are taken care of by an extension of polynomials, which is called Laurent polynomial. Formally, a univariate Laurent polynomial is given by

$$F(z) = a_m z^{-m} + a_{m+1} z^{-(m+1)} + \cdots a_n z^{-n} \tag{2.11}$$

where integers m, n satisfy $m \leq n$. When $m = n$, it is called a monomial. For Laurent polynomials, the degree is defined as the difference of the powers at both ends. For example, $deg(F(z)) = n - m$. Obviously, polynomials can be regarded as a special case of Laurent polynomials. Similar notation on fields also applies to Laurent polynomials.

For multidimensional (nD) system, multivariate polynomials or Laurent polynomials are used. Its definition is similar except that the arrangement of terms may vary with the chosen term ordering. This is further addressed in Chapter 6.3.

For some case, it is desired to describe the polynomials in every dimension. Therefore, the size for a filter is defined as the lengths of corresponding Laurent polynomial in every dimension. For example, the size of $H(z_1, z_2) = 1 + z_1 + z_1^2 + (1+z_1+z_1^2)z_2$ is $S(H) = 3\times 2$. If $S(H) = h_1 \times h_2$ and $S(G) = g_1 \times g_2$, the inequality $S(H) \geq S(G)$ indicates $h_1 \geq g_1$ and $h_2 \geq g_2$. Obviously, for 1D the size reduces to the length of a given polynomial, denoted as $L(H)$.

In the following, either polynomials or Laurent polynomials are used, whichever is more convenient.

2.5 Multidimensional signals and systems

2.5.1 nD signals and their transforms

As notated in Chapter 1, all vectors, matrices and sets are expressed in boldface letters. A discrete nD signal $x[\mathbf{n}]$ is a function of M-component vector

$$\mathbf{n} = [n_0, n_1 \cdots n_{M-1}]^T.$$

Correspondingly, the Fourier transform and its inverse of $x[\mathbf{n}]$ can be expressed as

$$X(\boldsymbol{\omega}) = \sum_{n_0=0}^{N_0-1}\sum_{n_1=0}^{N_1-1}\cdots\sum_{n_{M-1}=0}^{N_{M-1}-1} x[\mathbf{n}]e^{-j\boldsymbol{\omega}^T\mathbf{n}} \tag{2.12}$$

$$x[\mathbf{n}] = \frac{1}{(2\pi)^M}\int_{\omega_0=-\pi}^{\pi}\cdots\int_{\omega_{M-1}=-\pi}^{\pi} X(\boldsymbol{\omega})e^{j\boldsymbol{\omega}^T\mathbf{n}}d\omega_0\cdots d\omega_{M-1} \tag{2.13}$$

Here $\boldsymbol{\omega} = [\omega_0, \omega_1 \cdots \omega_{M-1}]^T$. These Fourier transform pairs also have some properties such as linearity, the equivalence between convolution and multiplication, etc. [25]. We only list the properties of our interest. Here we assume $x[\mathbf{n}]$ is a filter with support size $N_0 \times N_1 \times \cdots \times N_{M-1}$ and a Fourier transform $X(\boldsymbol{\omega})$.

Property 1. Separable sequence

$$\begin{aligned} & x[n_0, n_1, \cdots, n_{M-1}] = x_0[n_0] \cdot x_1[n_1] \cdots x_{M-1}[n_{M-1}] \\ \Leftrightarrow \quad & X(\omega_0, \omega_1, \cdots, \omega_{M-1}) = X_0(\omega_0) \cdot X_1(\omega_1) \cdots X_{M-1}(\omega_{M-1}) \end{aligned} \tag{2.14}$$

Property 2. Symmetry properties

(a)
$$x^*[\mathbf{n}] \Leftrightarrow X^*(\bar{\omega}) \tag{2.15}$$

(b)
$$x[\mathbf{n}] \text{ is real} \Leftrightarrow X(\boldsymbol{\omega}) = X^*(-\boldsymbol{\omega}) \tag{2.16}$$

$$X_R(\boldsymbol{\omega}) = X_R(-\boldsymbol{\omega}), \mid X(\boldsymbol{\omega}) \mid = \mid X(-\boldsymbol{\omega}) \mid$$

$$X_I(\boldsymbol{\omega}) = -X_I(-\boldsymbol{\omega}), \theta_x[\boldsymbol{\omega}] = -\theta_x(-\boldsymbol{\omega})$$

(c)
$$x[n_0, n_1, \cdots - n_i, \cdots n_{M-1}] \Leftrightarrow X(\omega_0, \omega_1, \cdots - \omega_i, \cdots \omega_{M-1}) \tag{2.17}$$

(d)
$$x[\mathbf{n}] \text{ is real and even} \Leftrightarrow X(\boldsymbol{\omega}) \text{ is real and even} \tag{2.18}$$

(e)
$$x[\mathbf{n}] \text{ is real and odd} \Leftrightarrow X(\boldsymbol{\omega}) \text{ is pure imaginary and odd} \tag{2.19}$$

The first property, in fact, is referred as separability for nD FBs, as mentioned in the beginning of this chapter. We address this issue in detail in a following section. The second property is about the symmetry relation between filters in original domain and their responses in transformed domain. This is useful if we want to incorporate some symmetry on their frequency response. For example, if a filter has a frequency response symmetry with respect to the origin, its filter coefficients must be real.

Similarly, z-transform for an nD signal $x[\mathbf{n}]$ is defined as

$$X(\mathbf{z}) = \sum_{n_0} \sum_{n_1} \cdots \sum_{n_{M-1}} x[\mathbf{n}] z_0^{-n_0} z_1^{-n_1} \cdots z_{M-1}^{-n_{M-1}} = \sum_{\mathbf{n} \in \mathcal{N}} x[\mathbf{n}] \mathbf{z}^{-\mathbf{n}} \tag{2.20}$$

The properties of z-transform are similar to those of nD-FT, though with a slightly different expression [25]. Some properties of interest are given as follows.

Property 1. Linear mapping of variables

$$x[n_0, n_1] = y[m_0, m_1] \mid_{m_0 = l_{00} n_0 + l_{01} n_1, m_1 = l_{10} n_0 + l_{11} n_1} \Leftrightarrow Y(z_0, z_1) = X(z_0^{l_{00}} z_1^{l_{10}}, z_0^{l_{01}} z_1^{l_{11}}) \tag{2.21}$$

Property 2. Symmetry property

$$x[-\mathbf{n}] \Leftrightarrow X(\mathbf{z}^{-1}) \tag{2.22}$$

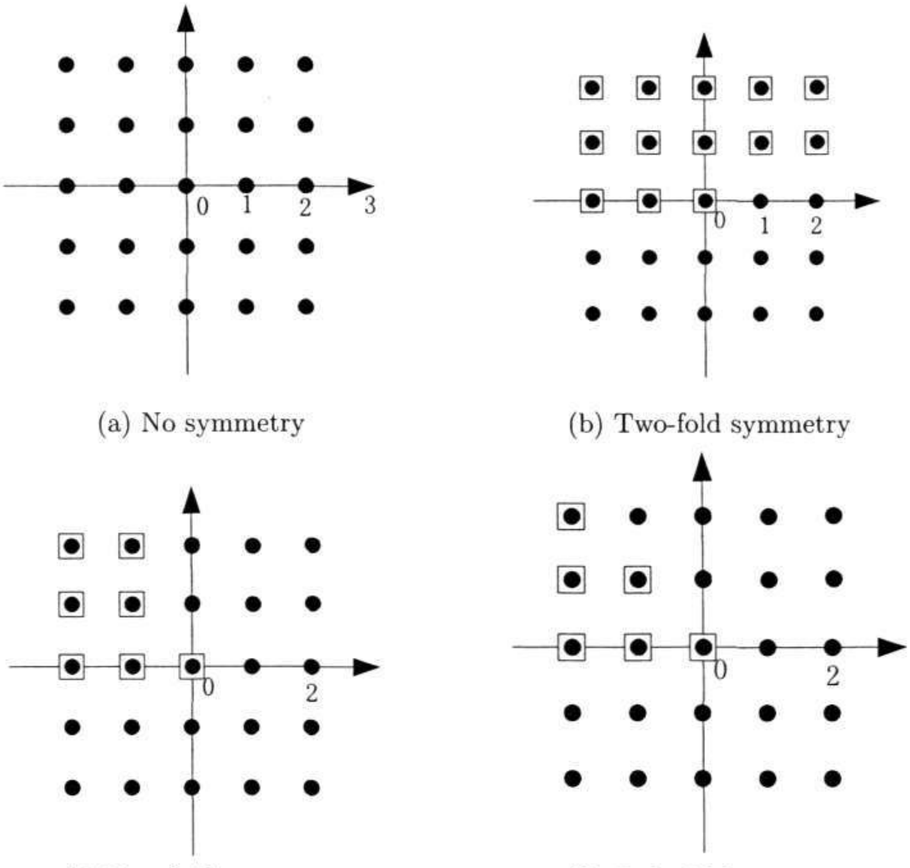
Here, for the sake of brevity, the first property is described in the form of 2D, while the nD case can be derived accordingly. These two properties are important in the following sense. The first property establishes the relation between the signals before and after a resampling procedure while resampling is a crucial element for an nD FB. For the second one, it translates the symmetry in the original domain to the symmetry in z-domain. As we see later, this symmetry is related to linear phase for a filter.

2.5.2 Linear phase and symmetries in 2D

As we mentioned, linear phase property is critical in the applications including image processing, due to its tendency to preserve the shape of the signal component in the passband region of the filter. For 1D filter, linear phase property is characterized by the symmetry of its support. However, in nD, symmetry may have different types, including point-wise symmetry, line-wise symmetry, plane-wise symmetry, etc. For the sake of a clear illustration, the scope is restricted to 2D case as the higher dimensional cases can be obtained similarly. For 2D system, a digital filter $h[n_0, n_1]$ is said to have linear phase when its frequency response $H(\omega_0, \omega_1)$ has the phase component linear in both ω_0 and ω_1. Furthermore, if the phase component is the constant 0, namely

$$H(\omega_0, \omega_1) = H^*(\omega_0, \omega_1), \tag{2.23}$$

it is said to have zero phase. This is equivalent to requiring $h[n_0, n_1] = h^*(-n_0, -n_1)$, or $h[n_0, n_1] = h[-n_0, -n_1]$ if we only consider real coefficients. The above symmetry makes approximately half of points of $h[n_0, n_1]$ related to the others, as depicted



(a) No symmetry

(b) Two-fold symmetry

(c) Four-fold symmetry

(d) Eight-fold symmetry

Figure 2.9: Independent samples of signals under different symmetries

in Figure 2.9(b). Hence we address such symmetry as two-fold symmetry. In fact, more restricted symmetries are also of interest. The four-fold symmetry is given by (as illustrated in Figure 2.9(c))

$$h[n_0, n_1] = h[-n_0, n_1] = h[n_0, -n_1] = h[-n_0, -n_1] \tag{2.24}$$

This symmetry can be translated into frequency domain according to Property 2 of Fourier transform introduced in the previous section,

$$H(\omega_0, \omega_1) = H(-\omega_0, \omega_1) = H(\omega_0, -\omega_1) = H(-\omega_0, -\omega_1) \tag{2.25}$$

Sometimes, even an eight-fold symmetry is needed, which can be described as (as illustrated in Figure 2.9(d))

$$h[n_0, n_1] = h[-n_0, n_1] = h[n_0, -n_1] = h[-n_0, -n_1] = h[n_1, n_0] \tag{2.26}$$

or in frequency domain

$$H(\omega_0, \omega_1) = H(-\omega_0, \omega_1) = H(\omega_0, -\omega_1) = H(-\omega_0, -\omega_1) = H(\omega_1, \omega_0) \tag{2.27}$$

The symmetry should be taken into account during filter design. When a special shape of filter is needed, the symmetry of the filter coefficients can be exploited to reduce the design space. This is especially desired since a reduced design space usually means a reduced number of parameters to search and a reduced implementation cost. For example, if we want to design a linear phase filter with diamond-shaped magnitude response, we can consider incorporating 4-fold or 8-fold symmetry to the filter coefficients. This leads to approximately a 75% or 87.5% saving in the number of coefficients. Conversely, for some filter structures, it may be impossible to achieve some symmetric filter shape due to the structure's lack of certain symmetry.

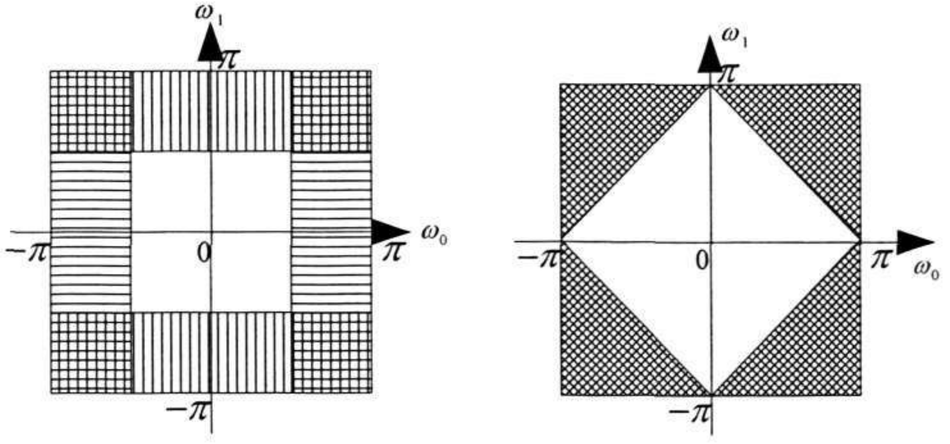
2.5.3 Separability and non-separability

A multidimensional filter is said to be separable if

$$H(\mathbf{z}) = H_0(z_0)H_1(z_1)\cdots H_{M-1}(z_{M-1}) \tag{2.28}$$

This equivalently indicates that the transfer function is a product of 1D transfer functions along each dimension. A most popular example is the tree-structured filtering used for wavelet transform. In Figure 2.10(a), four rectangular ideal filters are depicted. Assume that we have an ideal 1D low pass filter $H(z)$ and an ideal high pass filter $G(z)$, both of which stop at frequency $\frac{\pi}{2}$. Filters in Figure 2.10(a) can be obtained as follows. The filter without grating is the product of $H(z_0)$ and $H(z_1)$.

Hence this filter is actually performing low-pass filtering along both horizontal and vertical directions to the input signal. Similarly, the filter with horizontal grating is the product of $G(z_0)$ and $H(z_1)$, the filter with vertical grating is the product of $H(z_0)$ and $G(z_1)$, and the filter with cross grating is the product of $G(z_0)$ and $G(z_1)$. We also depict two non-separable filters in Figure 2.10(b). For these two filters, none of them can be expressed as a product of filters of only z_0 or z_1.



(a) Rectangular filters (b) Diamond filters

Figure 2.10: Frequency responses for ideal filters

These two kinds of filters also differ in terms of design freedom. For example, to obtain a 2D filter of size $N_0 \times N_1$, the greatest number of free parameters for a separable filter is $N_0 + N_1$, yet it is $N_0 \cdot N_1$ for a nonseparable filter. However, a greater freedom of design may translate to expensive implementation. Aside from the greater design freedom, nonseparable filters are also favored in the following sense. As already known [26], human visual system (HVS) could better resolve frequencies along the horizontal and vertical directions. This is reflected in Figure 2.11 [26], which plots the points for which the corresponding spatial frequencies can no longer be resolved. Evidently, the area that is considerably sensitive to HVS can be approximated by a diamond shape as depicted in the figure. Compared to the rectangular filters in Figure 2.10(a), the diamond shaped filter in Figure 2.10(b) obviously is a much better match to HVS. That is another reason why nonseparable filters such as diamond filters are favored at the cost of an expensive implementation.

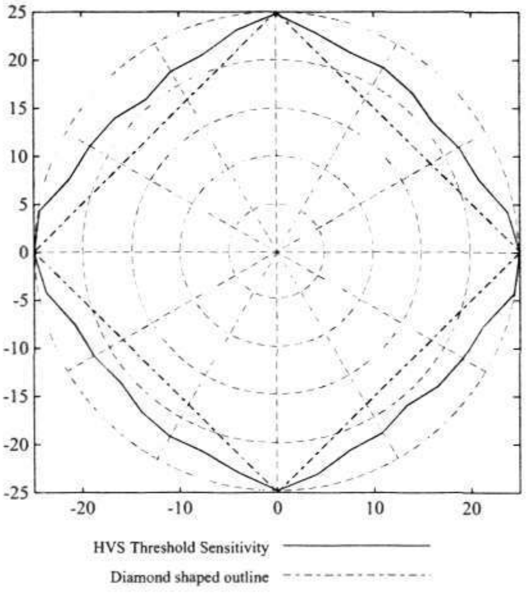


Figure 2.11: Resolvability threshold for sinusoidal gratings, cycles per degree

Chapter 3

The lifting based 2-channel linear phase filter bank

3.1 Introduction

Perfect-reconstruction (PR) filter banks (FB) with linear phase (LP) have been extensively used in the fields of image and signal processing. In the recent decade, there has been a number of reported works on 2-channel LPPRFB, such as [27–34]. Generally speaking, these works fall into two classes. For one class, as done in [30–34], the FBs are developed by solving the linear equations from various constraints as needed. Since the desired properties, like LP and PR, are embedded into the linear equations, unconstrained optimization can always be used for the chosen free parameters. Therefore, these FBs usually have a better performance in terms of stop-band attenuation, transition band etc. During implementation, these FBs have to be adapted to some specific structure. However, such adaption (such as limited precision) usually leads to the loss of the obtained properties. Preferably, the other class of approaches are to combine the choice of structure and the design of FBs. In [27] Nguyen et al developed the lattice structure, which is elicited from the linear prediction coder. This work established the framework of 2-channel LPPRFB for many succeeding works. However, their solution is incomplete for type-A (as explained later) and redundant for type-B (more design parameters than the degree of freedom), where type-A and type-B are two possible types of 2-channel LPPRFB. The lattice structure proposed in [28] is novel in the sense that it employed the Chebyshev representation of polynomials to factorize a FB. But for a given analysis bank constructed by this method, its inverse has to be recalculated which, may have similar form. Besides, the scaling in the inverse structure may be totally different from its counterpart and expensive to implement. Another lattice structure is also reported in [29]. However, similar problem exists since the inverse structure needs to be determined. In this chapter we also want to explore the relation between LPPRFB and a specific structure, the lifting scheme, which has a direct inverse from its analysis bank.

For a specific structure, two things are of interest. One is whether we can impose desired properties to the structure and how. The other is whether it is complete. Now we explain further on the former point. For design optimization and implementation, any structurally imposed property of a FB is preferred. A structural property is valid even with change of parameters (for design) or approximation of parameters (for implementation). The lattice structure of [27] employing a matrix $\begin{bmatrix} 1 & a \\ a & 1 \end{bmatrix}$ is such an example. It is structurally invertible (alternatively, structurally PR) with an inverse of $\frac{1}{1-a^2}\begin{bmatrix} 1 & -a \\ -a & 1 \end{bmatrix}$ whatever we choose for a, if we neglect the scaling $\frac{1}{1-a^2}$ and the exception of $a = 1$. Therefore, a in $\begin{bmatrix} 1 & a \\ a & 1 \end{bmatrix}$ may be used as a design parameter during optimization. Further, the obtained value of a may be approximated to any limited precision during implementation.

In case of LPPRFB's, it is desirable that the FB is structurally LP as well as structurally PR. A linear phase filter is structurally LP if realized in direct form, but usually not in other forms. For example, direct form implementation of $1 - \frac{13}{6}z^{-1} + \frac{133}{36}z^{-2} - \frac{13}{6}z^{-3} + z^{-4}$ is structurally LP, since coefficient symmetry may be imposed even with finite precision. But its factorized (cascade) form $\left(1 - \frac{2}{3}z^{-1} + \frac{4}{9}z^{-2}\right)\left(1 - \frac{3}{2}z^{-1} + \frac{9}{4}z^{-2}\right)$ is not structurally LP. If all coefficients are expressed using one decimal point precision, then

$$\left(1-0.7z^{-1}+0.4z^{-2}\right)\left(1-1.5z^{-1}+2.3z^{-2}\right)=1-2.2z^{-1}+3.75z^{-2}-2.21z^{-3}+0.92z^{-4}$$

is no longer LP. Direct form design or implementation of LPPRFB is structurally LP but typically not structurally PR. The above lattice structure is structurally PR as well as LP. However, the examples from [27] show that this lattice is more suitable for the FB with filters of the same length rather than those of different lengths.

Lifting structure, as depicted in Figure 3.1, is another popular structure born with such structurally PR property. In this chapter, how to factorize 2-channel LPPRFB based on lifting and how to incorporate LPPR properties to the implementation structure of a FB is investigated. For type-A and type-B FB, different approaches are used to achieve this. The factorization also includes the singular type-A filter bank, which is based on a novel generalized lifting structure. It is shown that for type-B, the proposed lifting factorization is most suitable, since it can structurally guarantee linear phase and perfect reconstruction. For type-A, if filters are of equal length, the proposed lifting factorization is proved to be equivalent to lattice factorization. Therefore, the proposed lifting-lattice mixed structure is most suitable for type-A filter banks. We apply the factorization to existing filter banks, and illustrate its advantages towards efficient implementation and reduced word-length implementation. Some design results based on the proposed factorization are also presented.

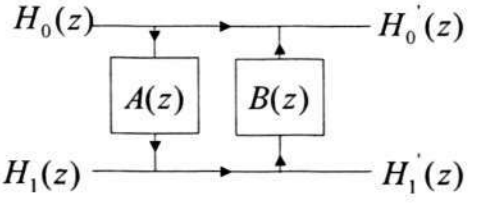


Figure 3.1: Lifting structure

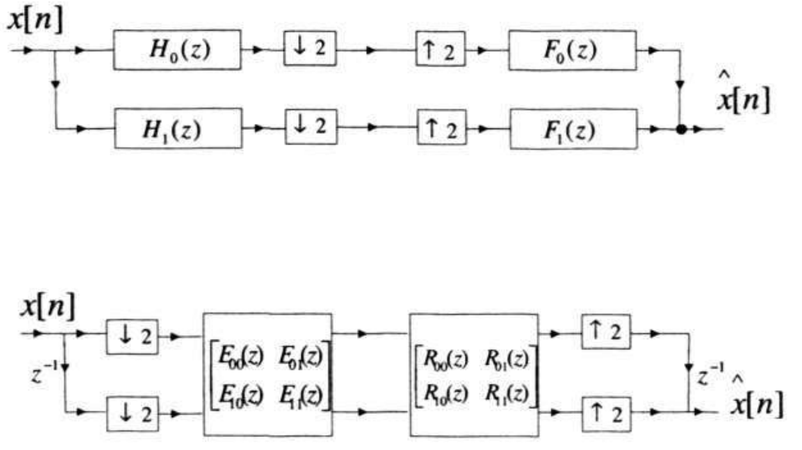


Figure 3.2: The 2-channel filter bank and its polyphase representation

3.1.1 Lifting factorization for 2-channel PRFB

Let us briefly introduce how a 2-channel PRFB is factorized into lifting steps according to Euclidean algorithm. We consider $[H_0(z), H_1(z)]$, a 2-channel PRFB, as depicted in Figure 3.2. Let $E_{i0}(z)$, $E_{i1}(z)$ be the polyphase components of $H_i(z)$. Let $\mathbf{E}(z)$ be the analysis polyphase matrix. Without loss of generality, we assume that $c = 1$ and $n = 0$ in (2.6). This can be achieved by appropriate scaling and shifting. Therefore $\mathbf{E}(z)$ satisfies the relation

$$det(\mathbf{E}(z)) = det \begin{bmatrix} E_{00}(z) & E_{01}(z) \\ E_{10}(z) & E_{11}(z) \end{bmatrix} = 1 \tag{3.1}$$

Theorem 3.1 *[35] (Euclidean algorithm for Laurent polynomials) Take two Laurent polynomials $a(z)$ and $b(z) \neq 0$ with $deg(a(z)) \geq deg(b(z))$. Initiate variables as $a_0(z) = a(z)$ and $b_0(z) = b(z)$ and iterate the following steps starting from $i = 0$:*

$$\begin{aligned} a_{i+1}(z) &= b_i(z) \\ b_{i+1}(z) &= a_i(z)\%b_i(z) \end{aligned} \tag{3.2}$$

Then $a_n(z) = gcd(a(z), b(z))$ where n is the smallest number for which $b_n(z) = 0$.

The proof can be found in [35]. By the way, the Euclidean algorithm is also applicable to matrices. The approaches to extract the gcrd and gcld from two univariate polynomial matrices have been studied extensively, e.g. by B.D.O. Anderson [36].

Here %, similar to that in C language notation, is the operation to get the remainder of division. For example if in (3.2) the quotient is $q_i(z)$, then (3.2) can be written as:

$$\left[\begin{array}{cc} a_i(z) & b_i(z) \end{array}\right] = \left[\begin{array}{cc} a_{i+1}(z) & b_{i+1}(z) \end{array}\right] \left[\begin{array}{cc} q_i(z) & 1 \\ 1 & 0 \end{array}\right] \tag{3.3}$$

Applying the above algorithm on $E_{00}(z)$ and $E_{01}(z)$, and noting from (3.1) that the gcd of $E_{00}(z)$ and $E_{01}(z)$ is a constant, we can always express them in the following manner:

$$\left[\begin{array}{cc} E_{00}(z) & E_{01}(z) \end{array}\right] = \left[\begin{array}{cc} K & 0 \end{array}\right] \prod_{i=0}^{n-1} \left[\begin{array}{cc} q_i(z) & 1 \\ 1 & 0 \end{array}\right] \tag{3.4}$$

We can construct another filter $H_1'(z)$ based on the filter $H_0(z)$,

$$\mathbf{E}'(z) = \left[\begin{array}{cc} E_{00}(z) & E_{01}(z) \\ E_{10}'(z) & E_{11}'(z) \end{array}\right] = \left[\begin{array}{cc} K & 0 \\ 0 & 1/K \end{array}\right] \prod_{i=0}^{n-1} \left[\begin{array}{cc} q_i(z) & 1 \\ 1 & 0 \end{array}\right] \tag{3.5}$$

Since we can guarantee n to be an even number by multiplying z to $H_0(z)$ and $-z^{-1}$ to $H_1'(z)$ when necessary without destroying the PR, the determinant of $\mathbf{E}'(z)$ is 1 according to (3.5). It means $H_0(z)$ and $H_1'(z)$ are also PR pair. To bridge $H_1'(z)$ to the original $H_1(z)$, the following theorem is needed. Note that a similar statement also appears in Proposition 4.6 from [37].

Theorem 3.2 *If both $[H_0(z), H_1(z)]$ and $[H_0(z), H_1'(z)]$ are PRFB, we can always relate $H_1(z)$ and $H_1'(z)$ with some Laurent polynomial $S(z)$ as*

$$H_1(z) = H_1'(z) + H_0(z)S(z^2).$$

Proof: With the definition of polyphase components, we can easily translate the PR condition in (2.5) as follows:

$$\begin{cases} H_0(-z)H_1(z) - H_0(z)H_1(-z) = 2z^{-1} \\ H_0(-z)H_1'(z) - H_0(z)H_1'(-z) = 2z^{-1} \end{cases}$$
$$\Rightarrow H_0(-z)H_1(z) - H_0(z)H_1(-z) = H_0(-z)H_1'(z) - H_0(z)H_1'(-z)$$
$$\Rightarrow \frac{H_1(z)-H_1'(z)}{H_0(z)} = \frac{H_1(-z)-H_1'(-z)}{H_0(-z)}$$

if we denote the left side of the last step as $K(z)$, we will find $K(z) = K(-z)$, which means $K(z)$ only includes even powers of z^{-1}. If we define $K(z) = S(z^2)$ we will reach the conclusion in the theorem, i.e., $H_1(z) = H_1'(z) + H_0(z)S(z^2)$. ∎

If we express the above theorem in a polyphase matrix manner, it will appear as

$$\mathbf{E}^{\triangle}(z) = \begin{bmatrix} 1 & S(z) \\ 0 & 1 \end{bmatrix} \mathbf{E}'(z). \tag{3.6}$$

With the observation

$$\begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & q_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ q_i(z) & 1 \end{bmatrix} \tag{3.7}$$

and Theorem 3.2, we can express the original polyphase matrix with lifting steps as follows:

$$\mathbf{E}^{\triangle}(z) = \prod_{i=1}^{m} \begin{bmatrix} 1 & s_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & 1/K \end{bmatrix} \tag{3.8}$$

The above result follows from combining (3.5) and (3.6), setting $m = n/2+1$,$t_m(z) = 0$ and $s_m(z) = K^2S(z)$.

Note that in (3.8), the scaling matrix is actually moved to the most right using the following equivalence,

$$\begin{aligned} \begin{bmatrix} 1 & 0 \\ A(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K^{-1} \end{bmatrix} &= \begin{bmatrix} K & 0 \\ 0 & K^{-1} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A(z)K^2 & 1 \end{bmatrix} \\ \begin{bmatrix} 1 & A(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & K^{-1} \end{bmatrix} &= \begin{bmatrix} K & 0 \\ 0 & K^{-1} \end{bmatrix} \begin{bmatrix} 1 & A(z)K^2 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3.9}$$

The above factorization is only valid for 2-channel FB. Factorization of multiple channels is elaborated in the next chapter. The approach is similarly based on the so-called extended Euclidean algorithm.

The problem with the lifting scheme is, as declared in [11], its high non-uniqueness leading to many inefficient solutions. It's interesting to find out a strategy to pick the best factorization scheme. Take the factorization of Bior2.2 (a biorthogonal wavelet in MATLAB) for an example. The following factorizations are both legal results from the Euclidean algorithm in [11].

$$\mathbf{E}(z)=\begin{bmatrix}\frac{7\sqrt{2}}{8} & 0\\ 0 & -\frac{8}{7\sqrt{2}}\end{bmatrix}\begin{bmatrix}1 & \frac{16}{49}\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ -\frac{7}{16}z^{-1} & 1\end{bmatrix}\begin{bmatrix}1 & \frac{2}{7}z\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ -0.5 & 1\end{bmatrix} \tag{3.10}$$

$$\mathbf{E}(z)=\begin{bmatrix}1 & 0\\ \frac{2}{7} & 1\end{bmatrix}\begin{bmatrix}\frac{7\sqrt{2}}{8} & 0\\ 0 & -\frac{8}{7\sqrt{2}}\end{bmatrix}\begin{bmatrix}1 & \frac{16}{7}\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ -\frac{z}{16} & 1\end{bmatrix}\begin{bmatrix}1 & -2\\ 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\ \frac{-z^{-1}}{2} & 1\end{bmatrix} \tag{3.11}$$

These structures are not satisfactory in more than one way. First, when implementing the structure with finite precision, the LP property is no longer guaranteed. Therefore these structure are not structurally LP. Further, these structures are not efficient since the number of coefficients are not less than that of the direct form. As a result, there are many works on the lifting implementation of some specific FBs, especially 9/7 wavelets [38–40]. In this work we aim to find a lifting factorization that is both structurally PR and structurally LP, unique and efficient, applicable to all 2-channel LPPRFB and not just any specific one.

A complete approach is presented in this thesis for the factorization of 2-channel LPPRFB. To achieve structurally LP, each lifting block is made LP (symmetric). For 2-channel LPPRFB, the low pass and high pass filters are symmetric and related to each other due to the PR property. We exploit the symmetry in the form of PR condition and prove that we can enforce such symmetry to each lifting block. Further, explicit expression for the lifting blocks are provided. The proposed structure has many desirable properties inherited from lifting and it is useful both in design and implementation.

3.2 LPPRFB and lifting-based factorization

3.2.1 Characteristics of 2-channel LPPRFB

To solve the problem of factorization, it is necessary to understand the characteristics of 2-channel LPPRFBs first.

We start from the PR condition,

$$H_0(z)H_1(-z) - H_0(-z)H_1(z) = z^c \tag{3.12}$$

for certain integer c. Considering the relation between filters and its polyphase components,

$$\begin{aligned} E_{00}(z^2) &= \frac{H_0(z)+H_0(-z)}{2} \\ E_{01}(z^2) &= \frac{H_0(z)-H_0(-z)}{2z^{-1}}, \end{aligned} \tag{3.13}$$

the PR condition appears as

$$E_{00}(z)E_{11}(z) - E_{01}(z)E_{10}(z) = z^{n_0} \tag{3.14}$$

where $c = 2n_0 + 1$. Therefore, c in Eq. (3.12) should be an odd number. Further consider the constraints on permitted length and symmetry for linear phase filters. It is known that there are only two types of 2-channel LPPRFB, type-A and type-B, as defined in [27]. Simply speaking, type-A FB is comprised of two even-length filters with one symmetric and the other antisymmetric, while type-B FB is comprised of two odd-length filters with both symmetric. For detailed derivation, the interested reader can refer to [27] or the nD counterpart for quincunx downsampled LPPRFB in Chapter 7.1.

Since a filter is symmetric/antisymmetric, we define the power of z at its mid-point as the symmetry center. If the length of the filter is even, the symmetry center will be of the form $n + \frac{1}{2}$ where n is integer. For example, a LP filter with terms from z^0 to z^{-3} has symmetry center at $z^{-1.5}$. We further find that the distance of symmetry centers of both filters are not arbitrary. It can be described by the following theorem.

	Type-A	Type-B
length	$(2N_0, 2N_1)$	$(2N_0+1, 2N_1+1)$
symmetry	symmetric & antisymmetric	both are symmetric
length combination	both N_0 and N_1 are even or odd	N_0 is even and N_1 is odd or vice versa
degree of freedom	$\frac{N_0+N_1}{2}$	$\frac{N_0+N_1+3}{2}$
distance of symmetry centers	even	odd

Table 3.1: Properties of 2-channel LPPRFB

Theorem 3.3 *For type-A FB's, it is necessary to have the difference of symmetry centers of two filters to be an even number. For type-B FB's, it is necessary to have the difference of symmetry centers of two filters to be an odd number.*

For proof, please refer to Appendix A.

Apart from aforementioned distinct features, the two types are different from each other in other aspects. We summarize some properties of these two types in Table 3.1, where N_0 and N_1 are the numbers of (anti)symmetric coefficients for the analysis filters $H_0(z)$ and $H_1(z)$ respectively.

Though the derivation mentioned above from [27] is not easy to extend to multivariate case or nD case, since the length combinations are tedious to enumerate when the dimension is higher, we show in a later chapter that actually the classification is still applicable in nD case.

Note that there are two kinds of operation that can move the symmetry centers without destroying PR. They are (a) inserting the same amount of delays to both channels, and (b) inserting even number of delays to either channel. Therefore, a given 2-channel LPPRFB can be always aligned as follows without loss of generality. The simplified expressions are used in succeeding sections.

An arbitrary type-B FB consists of symmetric filters $H_0(z), H_1(z)$ with lengths $2N_0+1, 2N_1+1$ and symmetry centers z^{c_0}, z^{c_1} respectively. With two operations defined above, these symmetry centers may be moved to $c_0 = 0$ and $c_1 = -1$, so that the filters may be expressed as follows.

$$\begin{aligned} H_0(z) &= \left(\sum_{n=1}^{N_0} h_0[n](z^{-n} + z^n) + h_0[0]\right) \\ H_1(z) &= z^{-1}\left(\sum_{n=1}^{N_1} h_1[n](z^{-n} + z^n) + h_1[0]\right) \end{aligned} \tag{3.15}$$

Similarly, an arbitrary type-A FB consists of symmetric filter $H_0(z)$ and antisymmetric filter $H_1(z)$ with lengths $2N_0, 2N_1$ and symmetry centers z^{c_0}, z^{c_1}. If the symmetry centers are moved to $c_0 = c_1 = -\frac{1}{2}$, the filters may be expressed as follows.

$$\begin{aligned} H_0(z) &= z^{-\frac{1}{2}}\left(\sum_{n=1}^{N_0} h_0[n](z^{n-\frac{1}{2}} + z^{-n+\frac{1}{2}})\right) \\ H_1(z) &= z^{-\frac{1}{2}}\left(\sum_{n=1}^{N_1} h_1[n](z^{n-\frac{1}{2}} - z^{-n+\frac{1}{2}})\right) \end{aligned} \tag{3.16}$$

3.2.2 Main results for factorization

Since our objective is a structurally LP lifting based factorization or construction, a given LPPRFB should retain the LP and PR property and the locations of symmetry centers after it is updated by a lifting step (or a dual lifting step). For the polyphase implementation, it is also required that the lifting block is a function of z^2. The following theorems provide the basis for such a lifting factorization.

Theorem 3.4 *Any 2-Channel LPPRFB with filters $H_0(z), H_1(z)$ with half lengths N_0, N_1 such that $N_0 > N_1$, may always be reduced to another 2-channel LPPRFB with filters $H_0'(z), H_1(z)$ using a lifting step of the following form,*

$$\begin{bmatrix} H_0'(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} 1 & B(z) \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} \tag{3.17}$$

such that the length of $H_0'(z)$ is at least 4 less than the length of $H_0(z)$.

Proof: For type-A FB, the lifting step is chosen as $B(z) = b(z^p - z^{-p})$ with $b = -\frac{h_0[N_0]}{h_1[N_1]}$, where $p = N_0 - N_1$ is the half-length difference of two filters. Note that $h_1[N_1]$ is not zero since the length of $H_1(z)$ is defined to be $2N_1$ (otherwise it is a trivial case). From Eq. (3.17), the new filter is given by $H_0'(z) = H_0(z) + B(z)H_1(z)$. It follows that the coefficients of z^{-N_0} and z^{N_0-1} in $H_0'(z)$ can be reduced to zero. The above strategy is also applicable for type-B FB but with $B(z) = bz(z^p + z^{-p})$ rather than that for type-A. It is also easy to verify that such choice can be directly adapted to polyphase implementation.

More interestingly, this step can simultaneously eliminate two more items in $H_0'(z)$, as shown below.

Recall the PR condition given by Eq. (3.12). The left hand side (LHS) of the above equation is a polynomial of z. By equating terms with the same power of z, a set of equations can be obtained. Note that the LHS of Eq. (3.12) is an odd function. Hence it contains only the terms with odd power of z. We only focus on the first one (as well as the last one, due to the symmetry), which has the highest (odd) order in the LHS.

For a type-A FB as given in Eq. (3.16), the term with the highest order appears as follows.

$$[(h_0[N_0]h_1[N_1-1] - h_0[N_0-1]h_1[N_1]]\, z^{N_0+N_1-1} = 0 \tag{3.18}$$

This is because the right hand side of Eq. (3.12) is a monomial containing z^{-1}, as indicated earlier. Similarly for a type-B FB given in Eq. (3.15), the highest order PR equation is

$$[h_0[N_0]h_1[N_1-1] - h_0[N_0-1]h_1[N_1]]\, z^{N_0+N_1-2} = 0 \tag{3.19}$$

Note that, apart from the difference in the order, this equation is the same as Eq. (3.18). Hence, we don't distinguish them in the following.

The above lifting step eliminates two more terms in $H_0'(z)$. In fact the bonus relies on Eq. (3.18) and Eq. (3.19). One more step from Eq. (3.18) or Eq. (3.19) gives

$$h_0[N_0-1] = \frac{h_0[N_0]h_1[N_1-1]}{h_1[N_1]} \tag{3.20}$$

It follows that for type-A the coefficients of $z^{-(N_0-1)}$ and z^{N_0-2} (z^{N_0-1} for type-B) in $H_0'(z)$ are also reduced to zero. ∎

Note that the symmetry center of $H_0'(z)$ remains the same as that of $H_0(z)$. After one step of the above reduction, the long filter will be reduced by 4 taps. We can update the corresponding PR conditions. For the reduced FB, an identical equation as Eq. (3.18) may again be derived, however, with different impulse response coefficients. Note that the above reduction also relies on the length difference of the filters, since otherwise we may have some other terms aside from $\frac{h_0[N_0]h_1[N_1-1]}{h_1[N_1]}$. These terms may prohibit the above cancelation.

Remark: Indeed similar result as in Theorem 1 can be achieved by Euclidean algorithm (EA) [39]. For the first stage, EA is executed on two polyphase components of one original filter assuming that the non-uniqueness problem we mentioned in the beginning has been properly solved. By transposing and reverse construction, another filter can be obtained. Then this generated filter is bridged to the original counterpart (the other original filter) by a lifting step derived from PR condition. Further as stated in section 3.1, the non-uniqueness of EA does not guarantee an efficient and/or LP factorization. However, the approach here is based on the PR equations and involves both filters. Every reduction is explicitly guaranteed by a PR equation. No middle results are generated. The factors obtained are LP. Consequently, this approach is more efficient and succinct.

Equation (3.17) provides a reduction step when the lengths of the filters are unequal. In case of type-A the filter lengths may become equal. In such a case, consider the filter $H_0(z)$. It's polyphase components may be written as

$$\begin{bmatrix} E_{00}(z) \\ E_{01}(z) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N_0} h_{0,i} z^{i-1} \\ \sum_{i=1}^{N_0} h_{0,N_0+1-i} z^{i-1} \end{bmatrix} = \begin{bmatrix} [h_{0,i}] \\ [h_{0,N_0+1-i}] \end{bmatrix}_{1\le i\le N_0} \tag{3.21}$$

where $[h_{0,i}]$ denotes the sum $\sum_{i=1}^{N_0} h_{0,i} z^{i-1}$. Comparing with (3.16), one obtains $h_{01} = h_0[N_0]$, $h_{0,N_0} = h_0[N_0-1]$ etc. Since $H_0(z)$ is even-length and symmetric, the coefficients of the polyphase components $E_{01}(z)$ are in reversed order of those of $E_{00}(z)$. It should be mentioned that we assume $h_{0,N_0} \neq h_{01}$ (i.e. $h_0[N_0] \neq h_0[N_0-1]$) here, and its exception (namely the mentioned singularity) will be treated later.

The following theorem resorts to Euclidean algorithm [11] on these polyphase components to obtain a reduction.

Theorem 3.5 *$H_0(z)$ with length $2N_0$ that belongs to any type-A 2-Channel LP-PRFB, may always be reduced to $H_0'(z)$ which belongs to another type-A 2-Channel LPPRFB, using a lifting step of the following form,*

$$\begin{bmatrix} [h'_{0,i}] \\ [h'_{0,N_0-i}] \end{bmatrix}_{1\leq i\leq N_0-1} = \begin{bmatrix} z & 0 \\ 0 & \frac{(h_{0,N_0}^2-h_{01}^2)}{h_{0,N_0}^2} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{-h_{0,N_0}h_{01}}{(h_{0,N_0}^2-h_{01}^2)} & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{-h_{01}}{h_{0,N_0}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} [h_{0,i}] \\ [h_{0,N_0+1-i}] \end{bmatrix}_{1\leq i\leq N_0} \tag{3.22}$$

such that the length of $H_0'(z)$ is at least 2 less than the length of $H_0(z)$.

Here $h'_{0,i} = (h_{0,N_0}h_{0,i+1} - h_{01}h_{0,N_0-i})$ for $i = 1, 2, \cdots N_0-1$. This theorem may be easily verified, so the proof is not included. The lifting step (third matrix from left) in the above equation is designed to cancel the first term of $E_{00}(z)$, while the dual lifting step (second matrix) cancels the last term of $E_{01}(z)$. The diagonal matrix re-aligns and scales the result so that the polyphase components are mirror images of each other. Note that $h_{0,N_0} \neq 0$ due to length definition, and $h_{0,N_0}^2 - h_{01}^2 \neq 0$ as assumed earlier.

3.3 Lifting structure for type-B FB

3.3.1 The procedure of factorization

According to Table 3.1 a type-B FB always has filters of different length, say $2N_0+1$ and $2N_1+1$. Without loss of generality, $N_0 > N_1$ is assumed. Using the technique introduced in Theorem 1, we can reduce the length of the filters alternately. During the reduction, the symmetry centers can also be retained by the choice of the quotient polynomial $B(z)$ in Eq. (3.17). Assuming N_0 to be even, finally the filters can be reduced to 1 taps ($\Leftrightarrow N_0 = 0$) and 3 tap ($\Leftrightarrow N_1 = 1$) respectively. For this stage, if we express them as follows,

$$\begin{aligned} H_0(z) &= h_0[0]z^{-1} \\ H_1(z) &= h_1[1](z^1 + z^{-1}) + h_1[0] \end{aligned} \tag{3.23}$$

the PR condition reduces to a single equation, i.e. $h_0[0]h_1[0] = \alpha$. Hence the following can be easily verified.

$$\begin{bmatrix} 1 & 0 \\ \frac{-h_1[1]\cdot(1+z^{-2})}{h_0[0]} & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{h_0[2]\cdot(1+z^2)}{h_1[1]} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} \alpha(h_1[0])^{-1} \\ h_1[0]\cdot z^{-1} \end{bmatrix} \tag{3.24}$$

Now we present the complete form of the above factorization for a given type-B FB. To describe the lifting factorization clearly, we introduce some notations. When reducing $H_0(z)$ via $H_1(z)$ we denote the quotient polynomial as $B(z)$. When reducing $H_1(z)$ via $H_0(z)$, it is $A(z)$. We also number the $A(z)$'s and $B(z)$'s inversely (i.e. $A_1(z)$ is for the last step, etc.), and denote $h_1[0]$ of Eq. (3.23) as K to obtain the following.

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \left(\prod_{n=1}^{\frac{N_1+1}{2}} \begin{bmatrix} 1 & -B_n(z) \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -A_n(z) & 1 \end{bmatrix} \right) \begin{bmatrix} \alpha K^{-1} \\ K z^{-1} \end{bmatrix} \tag{3.25}$$

Here α is the same as in (3.24), $A_n(z) = a_n(1 + z^{-2})$ when $n \leq \frac{N_1+1}{2}$ and $B_n(z) = b_n(1 + z^2)$ when $n < \frac{N_1+1}{2}$. Note that

$$\begin{pmatrix} 1 & B_1(z) \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & B_2(z) \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & B_1(z) + B_2(z) \\ 0 & 1 \end{pmatrix}. \tag{3.26}$$

Since $H_0(z)$ may be longer than $H_1(z)$ by more than 4 taps (correspondingly $N_0 > N_1 + 2$), the first lifting step may contain more than one section. In a general manner, it can be expressed as

$$B_{\frac{N_1+1}{2}}(z) = b^{(1)}_{\frac{N_1+1}{2}}(1 + z^2) + \cdots + b^{(\frac{p+1}{2})}_{\frac{N_1+1}{2}}(z^{-p+1} + z^{p+1}) \tag{3.27}$$

Figure 3.3 shows the corresponding polyphase form of analysis FB (we simplify the figure by assuming $N_0 = N_1 + 1$ and $\alpha = 1$). Its synthesis counterpart can be achieved simply by changing the sign of every lifting block and mirror flipping them. It's easy to extend such factorizations to cases with either $N_0 < N_1$ or N_0 to be odd number. Considering that each $A_n(z)$, $B_n(z)$ requires one multiplier except $B_{\frac{N_1+1}{2}}(z)$ requires $\frac{N_0-N_1-1}{2}$ multiplications, and that one multiplier of $\begin{bmatrix} K^{-1} & 0 \\ 0 & K \end{bmatrix}$ may be absorbed elsewhere, this structure requires a total of $\frac{N_0+N_1+3}{2}$ multiplications.

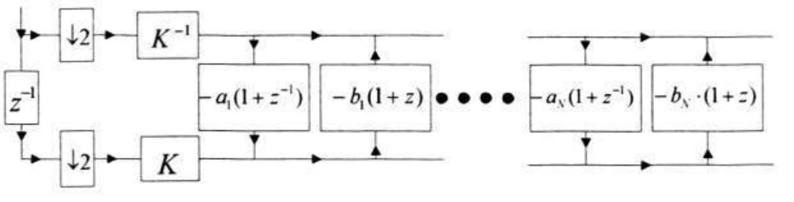


Figure 3.3: The type-B analysis filter bank with lifting

Note that the factorization of [11] may also lead to a structurally LP form, such as some examples given in [11]. However, there is no systematic approach to reach such factorization in [11], while the proposed factorization is general and explicit. We can declare that given an arbitrary type-B FB (not only some special wavelets), we can always factorize it into a succinct manner. With these factorizations we impose the LP property to the structure just as what we have done to PR property using lifting. We also obtain some other benefits. This will be illustrated in section 3.5.

3.3.2 The lifting design of type-B FB

Based on the factorization, we have the following structure to construct a type-B FB.

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \left(\prod_{n=1}^{N} \begin{bmatrix} 1 & B_n(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_n(z) & 1 \end{bmatrix} \right) \begin{bmatrix} \alpha K^{-1} \\ Kz^{-1} \end{bmatrix} \tag{3.28}$$

where

$$A_n(z) = a_1^{[n]}(1 + z^{-2}) + \cdots + a_{i_n}^{[n]}(z^{2(i_n-1)} + z^{-2i_n}) \tag{3.29}$$

and

$$B_n(z) = b_1^{[n]}(z^2 + 1) + \cdots + b_{i'_n}^{[n]}(z^{2i'_n} + z^{-2(i'_n-1)}) \tag{3.30}$$

.

We should point out here that to cover all possible filter banks of a fixed length the following restrictions should be satisfied, $i_n = i'_n = 1$ $(1 \le n \le N-1)$ and $i_N = 1$ if $B_N(z) \neq 0$. This can be demonstrated by a simple example.

Example 3.1 *If we construct a filter bank using the following structure,*

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ c(1+z^{-2}) & 1 \end{bmatrix} \begin{bmatrix} 1 & a(1+z^2) + b(z^{-2}+z^4) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ z^{-1} \end{bmatrix} \tag{3.31}$$

the filters can be denoted by parameters a, b, c as

$$\begin{cases} H_0(z) &= b\cdot(z^3+z^{-3}) + 0\cdot(z^2+z^{-2}) + a\cdot(z+z^{-1}) + 1 \\ H_1(z) &= bc\cdot(z^3+z^{-5}) + 0\cdot(z^2+z^{-4}) + (a+b)c\cdot(z+z^{-3}) \\ & \quad +c\cdot(1+z^{-2}) + (1+2ac)\cdot z^{-1} \end{cases} \tag{3.32}$$

No matter what values a, b, c are, filter $H_0(z)$ doesn't contain the items of z^2 and z^{-2} and filter $H_1(z)$ doesn't contain the items of z^2 and z^{-4}, which means the structure is not complete due to additional constraint. To explain this more generally, we study it from a viewpoint of degree of freedom.

3.3.3 A study from degree of freedom

The degree of freedom (DOF) is the total number of parameters minus the number of constraint equations among the parameters. In [27], the DOF has been calculated by following the above definition. This obtained value of DOF has been listed in Table 3.1. We refer to this value as the canonical DOF in the following, since this derivation of the DOF is widely accepted and used. When the DOF is less than the canonical DOF, the scheme will prove to be incomplete, which means the structure cannot cover all possible cases. When the DOF is more than the canonical DOF, the scheme will prove to be over-complete, which means the structure may lead to some extraneous results that conflict with the constraints. Only when the DOF is equal to the canonical DOF, we can regard the structure as a complete structure.

Assume that we construct a type-B FB via Eq. (3.28) with each $A_i(z)$, $B_i(z)$ as in Figure 3.3. We define $L(F)$ as the length of the filter $F(z)$. So $L(A_i) = 3$ and $L(B_i) = 3$. We can verify that for all positive N in Eq. (3.28), the lengths for the designed filters are $L_N(H_0) = 4N + 1$ and $L_N(H_1) = 4N - 1$. Since the PR constraints and the LP constraints are structurally satisfied, the parameters have no other constraints. Moreover, these parameters cannot be absorbed into other existing parameters. Therefore the number of parameters is equal to the DOF of our scheme. For $[H_0(z), H_1(z)]$, $\text{DOF}_{\text{proposed}} = 2N + 1$ (considering one freedom brought by K). This can be obtained by observing the number of parameters in Figure 3.3. According to Table 3.1, the canonical DOF for $[H_0(z), H_1(z)]$ is $\text{DOF}_{\text{can}} = \frac{(4N+1)+(4N-1)}{4} + 1 = 2N + 1$. It is indeed equal to that of the proposed scheme. We can declare that the structure in Eq. (3.28) is complete. This agrees with the factorization results of section 3.3.

Now assume that from a type-B FB $[H_0(z), H_1(z)]$, denoted by $\mathbf{H}$, we cascade another lifting and dual lifting pair $[A(z), B(z)]$ as in Eq. (3.29) and (3.30) and get the updated FB $[H_0'(z), H_1'(z)]$, denoting it by $\mathbf{H}'$. We denote the length difference of $[H_0(z), H_1(z)]$ by $4n + 2$, with $n = 0$ here. Since $L(H_0) > L(H_1)$, we have

$$\begin{cases} L(H_0') = L(H_0) + (4n_a - 2) + (4n_b - 2) \\ L(H_1') = L(H_0) + (4n_a - 2) \end{cases}$$

The canonical DOF of $\mathbf{H}'$ is $\text{DOF}_{\text{can}}(\mathbf{H}') = \frac{L(H_0') + L(H_1')}{4} + 1$, and similarly $\text{DOF}_{\text{can}}(\mathbf{H})$ is found. Since $A(z)$ and $B(z)$ has n_a and n_b parameters, the DOF of the new scheme is $\text{DOF}_{\text{new}}(\mathbf{H}') = \text{DOF}_{\text{new}}(\mathbf{H}) + n_a + n_b$. To make the new scheme complete, we require

$$\begin{aligned}
&\mathrm{DOF}_{\mathrm{can}}(\mathbf{H}') - \mathrm{DOF}_{\mathrm{can}}(\mathbf{H}) = \mathrm{DOF}_{\mathrm{new}}(\mathbf{H}') - \mathrm{DOF}_{\mathrm{new}}(\mathbf{H}) \\
\Longleftrightarrow &\frac{(L(H_0') + L(H_1')) - (L(H_0) + L(H_1))}{4} = n_a + n_b \\
\Longleftrightarrow &\left((2n_a - 1) + n_b - \frac{1}{2}\right) + n + \frac{1}{2} = n_a + n_b
\end{aligned}$$

The only solution for the above equation, $n_a + n = 1$, for non-negative n and positive n_a is $n = 0$ and $n_a = 1$. We have no constraint on the last section, or n_b, and we can choose any $B(z)$ of the form of Eq. (3.29) and retain the completeness of the structure. But if we want to cascade one more such section, it may be shown in a similar way that only $n_b = 1$ keeps the updated FB complete. This is because we make the length difference $n > 0$ for this stage, and this excludes any solution other than $n_b = 1$ for the completeness equation.

Therefore, we verify the completeness of Eq. (3.28) for type-B with the above constraints from a viewpoint of DOF. This agrees with the result we obtained during the factorization. Under such condition the structure is canonical in the sense that the number of parameters equals to the degree of freedom.

Since we impose the PR as well as LP property to the structure, we can use the unconstrained routine to optimize the parameters. Alternately, we can use finite precision parameters during design phase. For example, we can directly design the FB within the field of signed power of two numbers (SPT).

3.4 Lifting structure for type-A FB

3.4.1 Lifting based factorization

In [27], type-A FB is related with lattice structure. However, this structure is incomplete in the following situation: a) When the first two coefficients of $H_0(z)$ or $H_1(z)$ are identical, it will lead to a singular lattice structure with lattice coefficient to be 1. Such singularity is also reported in [41]. b) When two filters are of different lengths, they may fall into the above singular case when we seek the corresponding MIP (mirror image pair). Now we show these can be overcome when using lifting.

Firstly, we solve the latter problem, which has unequal-length filters. Due to the constraints listed in the Table 3.1, the length difference of two filters is a multiple of 4 and for the unequal part Theorem 3.4 is applicable. We can reduce the length of the longer filter by 4 taps in every iteration of lifting factorization. Hence the long filter can always been reduced up to the length of the short filter. We also obtain the corresponding lifting blocks $B(z)$ from these steps. Since the filters remain LP after the reduction, it follows that $B(z)$ is LP. However, when the filters are reduced to the same length ($N_0 = N_1$ in Eq. (3.16)), such factorization can no longer be extended.

At this stage, we resort to Theorem 3.5, a solution relying on the Euclidean algorithm [11]. By iterating the technique in Theorem 3.5, the following result can be reached.

$$\begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \left(\prod_{N_0-1}^{i=1} \begin{bmatrix} z & 0 \\ 0 & k_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -B_i & 1 \end{bmatrix} \begin{bmatrix} 1 & -A_i \\ 0 & 1 \end{bmatrix} \right) \begin{bmatrix} E_{00}(z) \\ E_{01}(z) \end{bmatrix} = \begin{bmatrix} a \\ 0 \end{bmatrix} \tag{3.33}$$

where at each stage $A_i = \frac{h_0[N]}{h_0[N-1]}$, $B_i = \frac{h_0[N]h_0[N-1]}{h_0^2[N-1]-h_0^2[N]}$ and $k_i = \frac{h_0^2[N-1]-h_0^2[N]}{h_0^2[N-1]}$. Since the high pass filter $H_1(z)$ is complementary (alternatively PR) to $H_0(z)$, we can generate a complementary high pass filter $H_1^r(z)$ via $H_0(z)$ and bridge it to $H_1(z)$ via another lifting step (section 4 of [11]). In conclusion, we may factorize an arbitrary type-A FB (after reducing the filters to the same length) into lifting steps as

$$\begin{aligned} \begin{bmatrix} E_{00}(z) & E_{01}(z) \\ E_{10}(z) & E_{11}(z) \end{bmatrix} = & \begin{bmatrix} 1 & 0 \\ D(z) & 1 \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & \eta a^{-1} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \\ & \cdot \left(\prod_{i=1}^{N_0-1} \begin{bmatrix} z^{-1} & 0 \\ 0 & k_i^{-1} \end{bmatrix} \begin{bmatrix} 1 & B_i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_i & 1 \end{bmatrix} \right) \end{aligned} \tag{3.34}$$

where η is a scaling factor that guarantees the identical determinant as that of FB $[H_0(z), H_1(z)]$. If we assume the polyphase matrix of FB $[H_0(z), H_1(z)]$ satisfies

$det(\mathbf{E}(z)) = z^{N_0-1}$, we have $\eta = \prod_{N_0-1}^{i=1} k_i$. Dual lifting block $D(z)$ acts as the bridge mentioned above and can also be uniquely determined by the given FB. Since the filters have been assumed to be the same length for this stage, $D(z)$ is nothing but a scaling. This scaling, according to [11], can be chosen as $D(z) = -\frac{\eta}{2a^2}$.

3.4.2 Equivalence between the proposed structure and lattice structure

Note that, three parameters A_i,B_i,k_i are actually inter-dependant and in this sense Eq. (3.34) is not strictly structurally LP. They are related by

$$B_i = \frac{A_i}{1-A_i^2} \quad \text{and} \quad k_i = 1 - A_i^2$$

Substituting them into the building section in (3.34), we obtain

$$\begin{bmatrix} z^{-1} & 0 \\ 0 & k_i^{-1} \end{bmatrix} \begin{bmatrix} 1 & B_i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_i & 1 \end{bmatrix} = \begin{bmatrix} \frac{z^{-1}}{1-A_i^2} & \frac{A_i z^{-1}}{1-A_i^2} \\ \frac{A_i}{1-A_i^2} & \frac{1}{1-A_i^2} \end{bmatrix} \tag{3.35}$$

Furthermore, if we consider

$$\begin{bmatrix} 1 & 0 \\ -\frac{\eta}{2a^2} & 1 \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & \frac{\eta}{a} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} a & a \\ -\frac{\eta}{a} & \frac{\eta}{a} \end{bmatrix} \tag{3.36}$$

as well as

$$\begin{bmatrix} \alpha & \alpha \\ \beta & -\beta \end{bmatrix} = \frac{\alpha+\beta}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & \frac{\alpha-\beta}{\alpha+\beta} \\ \frac{\alpha-\beta}{\alpha+\beta} & 1 \end{bmatrix} \tag{3.37}$$

we may find the equivalence between the proposed structure and the lattice structure from [27]. Actually, without the singular cases introduced in the beginning of this section, the lattice structure is more suitable for type-A than the proposed lifting structure, since the lattice structure can structurally retain the LP property when its coefficients are in any field (including integers) while the lifting structure can only keep it for real or rational numbers.

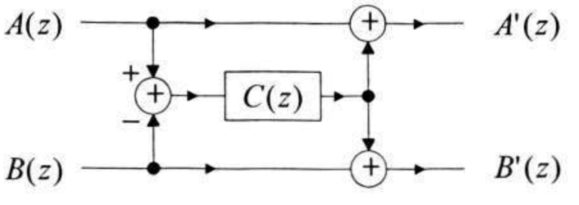


Figure 3.4: A structure for singular type-A FB

3.4.3 The treatment for singular type-A LPPRFB

Now we treat the other special case when $h_{N_0} = h_{N_0-1}$. This is regarded as a singular case in [41], which is beyond the capability of the above factorization as well as lattice factorization. To tackle it, a structure, as depicted in Figure 3.4, is proposed. This structure, similar to lifting scheme, has an inverse that can be obtained by simply changing the sign of $C(z)$ and mirror flipping. Therefore, it can be regarded as a generalized lifting scheme (GLS), as introduced in a latter section. Here we choose the GLS in Figure 3.4 with the lifting step $C(z) = \frac{h_{01}}{h_{02}-h_{0,N_0-1}}(z - z^{-1})$. When we apply it to the low-pass filter in Eq. (3.21), it's not difficult to verify that after applying such a lifting-like structure we will return to a similar point as shown in Eq. (3.21) without singularity,

$$\begin{bmatrix} [h'_{0,i}] \\ [h'_{0,N_0-i}] \end{bmatrix} = \begin{bmatrix} 1 + C(z) & -C(z) \\ C(z) & 1 - C(z) \end{bmatrix} \begin{bmatrix} [h_{0,i}] \\ [h_{0,N_0+1-i}] \end{bmatrix} \tag{3.38}$$

Then the normal lifting factorization can be continued. If further $h_{02} = h_{0,N_0-1}$, we can choose

$$C(z) = \frac{h_{01}}{h_{03} - h_{0,N_0-2}}(z^2 - z^{-2}) + \frac{h_{02}}{h_{03} - h_{0,N_0-2}}(z - z^{-1}) \tag{3.39}$$

to remove the 'double' singularity. This can be easily extended to more singular cases with a similar proof.

3.4.4 Type-A LPPRFB design based on a hybrid structure

The lifting structure can be used to take care of the unequal part and the singular part as we have shown in the above, and the lattice structure may be used for the equal part. This leads to a hybrid factorization, which is most effective. Considering that the structure has $N_1 - 1$ lattice sections and each such section may be realized using 1 multiplier [3], and $B(z)$, the last dual lifting steps requires $\frac{N_0-N_1}{2}$ multipliers, and two multipliers for scaling, a total of $\frac{N_0-N_1}{2} + N_1 + 1$ multipliers are needed in this factorization. An example is given in section 3.5.

In Figure 3.5 we depict the corresponding analysis bank of a type-A LPPRFB. Note that we absorb the bridge block $D(z)$ into the last dual lifting step $B(z)$.

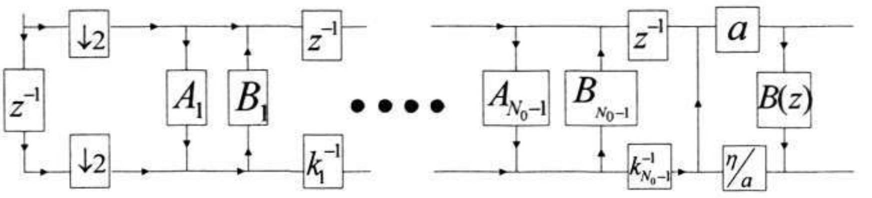


Figure 3.5: The type-A analysis filter bank with lifting

Consequently, for design purpose the following structure is suggested,

$$\mathbf{E}(z)=\begin{bmatrix} 1 & 0 \\ B(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & \alpha_0 \\ \alpha_0 & 1 \end{bmatrix} \left(\prod_{i=1}^{N} \begin{bmatrix} 1 & \alpha_i \\ \alpha_i z^{-1} & z^{-1} \end{bmatrix} \right) \begin{bmatrix} 1+C(z) & -C(z) \\ C(z) & 1-C(z) \end{bmatrix}$$

where $B(z) = f_1(z - z^{-1}) + \cdots + f_M(z^M - z^{-M})$ and $C(z) = c(z - z^{-1})$, and the rightmost section involving $C(z)$ is used only if singular FBs are to be included.

Remark: In [39], a structure for designing type-A LPPRFB is also proposed. In fact, the structure there is only applicable to a small subclass of type-A LPPRFB, which is comprised of Haar wavelet and its derived FBs by cascading symmetric lifting steps as indicated in Theorem 1. Our proposed approach is generally applicable to all type-A FB. With such structure we can enforce PR as well as LP properties to the FB we design. Hence we can use the unconstrained optimization to seek the desired FB.

3.5 Examples and applications

3.5.1 Factorization examples

We apply the proposed factorization to the following FBs: Bior2.2, namely the FB in Eq. (3.10) and Eq. (3.11) (FB1); popular CDF 9/7 wavelet (FB2) from [42]; 2 FBs from [43], where they are regarded as wavelets for image compression (FB3 & FB4). In [43], they are labeled as Filter 2 and Filter 3 respectively. The coefficients for first two FBs are easily available, hence we only list the effective coefficients for last two FBs in Table 3.2. In the following $\gamma(z) = 1 + z$ and Table 3.4 gives the lifting coefficients.

$$\text{FB1: } \mathbf{E}(z) = \begin{bmatrix} 1 & -\gamma(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{1}{4}\gamma(z^{-1}) & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

$$\text{FB2: } \mathbf{E}(z) = \left(\prod_{i=1}^{2} \begin{bmatrix} 1 & a_{2i-1}\gamma(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ a_{2i}\gamma(z^{-1}) & 1 \end{bmatrix} \right) \begin{bmatrix} a_5 & 0 \\ 0 & a_5^{-1} \end{bmatrix}$$

$$\text{FB3: } \mathbf{E}(z) = \left(\prod_{i=1}^{3} \begin{bmatrix} 1 & 0 \\ a_{2i-1}\gamma(z^{-1}) & 1 \end{bmatrix} \begin{bmatrix} 1 & a_{2i}\gamma(z) \\ 0 & 1 \end{bmatrix} \right) \begin{bmatrix} a_7 & 0 \\ 0 & a_7^{-1} \end{bmatrix}$$

$$\text{FB4: } \mathbf{E}(z) = \begin{bmatrix} 1 & 0 \\ a_1(z-1) & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & a_2 \\ a_2 & 1 \end{bmatrix} \left(\prod_{i=3}^{4} \begin{bmatrix} 1 & a_i \\ a_i z^{-1} & z^{-1} \end{bmatrix} \right)$$

First three FBs are all type-B and are factorized in a similar manner while the last FB is of type-A and is factorized in a hybrid way. The last factorization consists of 3 lattice steps and 1 lifting step. These factorizations are all canonical in the sense that their number of free parameters is equal to the degree of freedom of that FB. For comparison, we list the specifications of the FBs before and after lifting

	h_0 (FB3)	h_1 (FB3)	h_0 (FB4)	h_1 (FB4)
0	0.767245	0.832848	0.788486	0.615051
1	0.383269	-0.448109	0.047699	-0.133389
2	-0.06888	-0.069163	-0.129078	-0.067237
3	-0.033475	0.109737		-0.006724
4	0.047282	0.006292		0.018914
5	0.003759	-0.014182		
6	-0.008473			

Table 3.2: Filter coefficients for some existing FBs

factorization in Table 3.3. As shown in [3], lattice structure $\begin{bmatrix} 1 & a \\ a & 1 \end{bmatrix}$ as in FB4 factorization, needs 1 multiplier and 3 adders to implement. Note that the dynamic range is defined as the quotient of the maximum coefficient and the minimum coefficient. According to this table the proposed factorization is typically capable of a more efficient implementation. For FB1, compared to the factorizations of Eq. (3.10) and Eq. (3.11), the proposed factorization is more efficient. For FB2, the proposed dynamic range of 17 is better than the dynamic range of 30 from the factorization of [40]. For FB4, only 4 multipliers are needed to implement in proposed structure while 8 multipliers are needed if implementing in direct structure.

		FB1	FB2	FB3	FB4
No. of Coeffs.	Direct	5	9	13	8
	Lifting	3	5	7	4
Dynamic Range	Direct	6	36	21	221
	Lifting	6	17	11	20

Table 3.3: Comparison of lifting factorizations

3.5.2 The factorization of singular type-A FB

In [41], Vetterli et al gave an example of type-A FB that is singular for lattice factorization.

$$H_0(z)= 1+z^{-1}+2z^{-2}+3z^{-3}+3z^{-4}+2z^{-6}+z^{-7}+z^{-8}$$

$$H_1(z)=-1-z^{-1}-4z^{-2}-5z^{-3}+5z^{-4}+4z^{-6}+z^{-7}+z^{-8}$$

	FB2	FB3	FB4
a_1	0.58613434191	0.59742875528	0.14653358552
a_2	0.66806717120	1.54015786729	0.45537754123
a_3	-0.0700180094	0.11059029412	-9.6890747365
a_4	-1.2001710166	-0.1903980712	-2.7060935152
a_5	1.14960439886	-0.1972320984	
a_6		-2.1812627017	
a_7		1.01651054891	

Table 3.4: Coefficients of factorization

Using our proposed singularity treatment we can factorize it as

$$\mathbf{E}(z)=\begin{bmatrix}1 & 1\\-1 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\0 & z^{-1}\end{bmatrix}\begin{bmatrix}3 & 4\\4 & 3\end{bmatrix}\begin{bmatrix}1+(z^{-1}-z) & (z^{-1}-z)\\-(z^{-1}-z) & 1-(z^{-1}-z)\end{bmatrix}$$

Here we choose $C(z) = (z^{-1} - z)$. Another existing factorization from [28] is

$$\mathbf{E}(z)=\begin{bmatrix}1+z^{-1} & -1+z^{-1}\\-1+z^{-1} & 1+z^{-1}\end{bmatrix}\begin{bmatrix}\frac{7}{2}z^{-1} & 0\\0 & 1\end{bmatrix}\begin{bmatrix}1 & 0\\-1+z^{-2} & \frac{-1}{2}z^{-1}\end{bmatrix}\begin{bmatrix}1 & 1\\-1 & 1\end{bmatrix}$$

For comparison, the overall arithmetic complexity of the factorization of [28] is 2 multipliers, 8 adders. Specifically the block $\begin{bmatrix}1+z^{-1} & -1+z^{-1}\\-1+z^{-1} & 1+z^{-1}\end{bmatrix}$ only needs 4 adders as shown in [28]. The proposed factorization needs 1 multiplier, 9 adders as shown in Figure 3.6. Though the advantage is not overwhelming, we may conclude that the proposed factorization scheme is capable of the singularity treatment in a slightly better way.

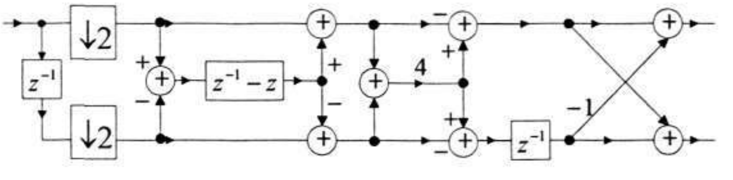


Figure 3.6: Factorization of the singular type-A FB

3.5.3 Reduced word-length implementation

Due to being structurally PR as well as LP, we can further reduce the implementation requirements stated above by representing the lifting or lattice coefficients

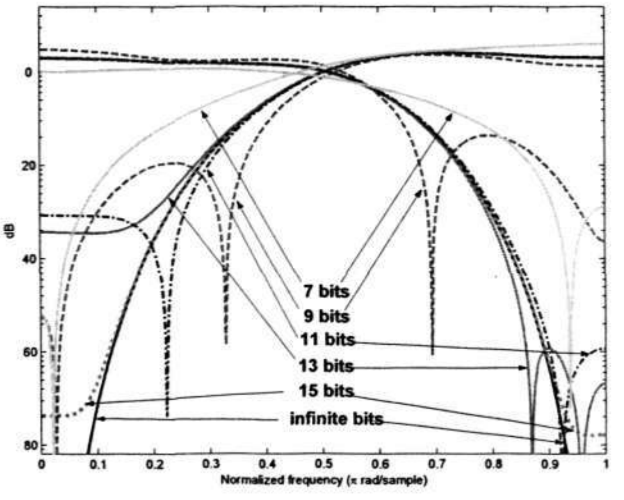


Figure 3.7: Frequency response of FB3

at $z = -1$. [45] also retains VM for FBs with reduced word-length. [45] relates the filter coefficients to a free design parameter derived from the constraints such as PR and VM. By adjusting this free parameter, the rationalized coefficients are obtained. However, this solution only involves the impulse response of the filter bank. When implementing the designed filter bank with some structure, the vanishing moment

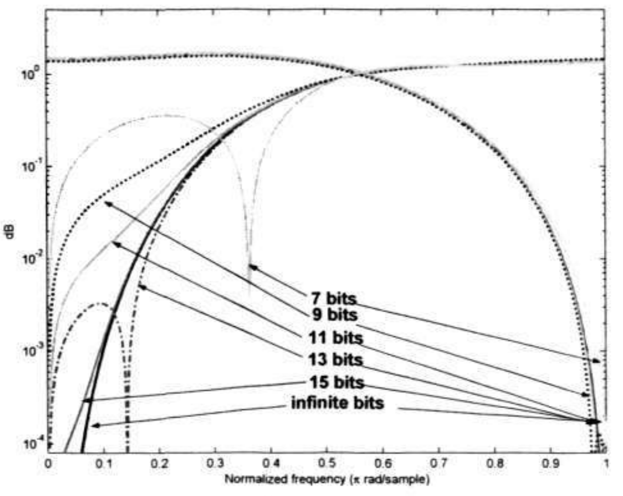


Figure 3.8: Frequency response of FB4

with shorter word-length while still retaining LP and PR properties. This is quite useful when the resource for implementation is limited or we prefer lower cost. Here FB2, FB3 and FB4 in subsection 3.5.1 are used for demonstration. We use from infinite bits (double precision floating point) to 7 bits (sign bit included) to denote the lifting and lattice coefficients. For the last two FBs we depict their frequency response for different bits in Figure 3.7 and Figure 3.8. We find the benefits of reduced implementation requirements are only at the cost of marginal deterioration of the frequency response. Also, we compare the performance of these reduced word-length FBs to that of infinite precision for image coding in Table 3.5. The results are the average difference of coding performance from the infinite bits case in terms of PSNR (dB) for several images with 5-level wavelet based zero-tree compression at the rate of 1bpp. A uniform quantizer is employed during compression. For entropy coding, histogram adaptation with escape codes is performed. For more information on the compression technique, please refer to [44].

As we can see from the table, the performance is acceptable for the precision with more than 7 bits (inclusive for FB4). For example, in Figure 3.9, two images are the coding result of 'Barbara' with FB3 in 9bits and 15 bits at a ratio of 1bpp respectively. As seen in the figure, the difference is not visually obvious to humans.

	15 bits	13 bits	11 bits	9 bits	7 bits
FB2	0.0019	0.0058	0.0332	0.5289	0.9567
FB3	0.0009	0.2347	0.3411	0.6086	2.2579
FB4	0.0013	0.0102	0.0479	0.0908	0.1100

Table 3.5: Average PSNR change in dB for different precisions

Note that during the reduction of precision, the vanishing moment (VM) of the FB is also one desirable property that we want to retain for applications such as compression. This property can be regarded as the zeros for the low-pass filter $H_0(z)$

(a) coefficients in 15 bits

(b) coefficients in 9 bits

Figure 3.9: Reconstructed images of FB3 at 1bpp

may no longer be retained due to the limited precision of the structure. However, for a specific 9/7 filters, a lifting structure that can retain all properties (including VM) with reduced word-length is presented in [46].

Another way to achieve VM for reduced word-length in a generic manner is proposed here, where precision reduction is performed on the lifting structure which can be directly used for implementation. Due to the nature of type-A FB, one VM will be enforced automatically. However this is not true for type-B. This is the reason for the different performance between these two types of FBs in Table 3.5.

VM is retained for some reduced word-length FBs in this work in the following way. For example, for FB1, if the coefficients $-\frac{1}{2}$, $\frac{1}{4}$, $\sqrt{2}$ in reduced word-length are denoted by a, b, c, VM for the low pass filter is retained if $4abc^2 + 2a + c^2 = 0$ is satisfied. This condition is achieved by putting $z = -1$. In the above, the reduced word-length coefficients of FB2 and FB3 are chosen in a similar fashion to retain the VM. However, for the reduce precision case, the adjustment of the coefficient for VM generally contradicts the aim of restraining the deterioration of frequency response. Hence, to find out the optimum tradeoff is interesting and worth further study.

3.5.4 Lifting design of 2-channel LPPRFB

Using the structure of analysis bank in (3.28), we have designed type-B FBs for $N = 1$ to 4 (filter lengths 5/3 to 17/15). The filter banks are designed subject to the minimum of the cost function involving the filter shape as follows.

$$\begin{aligned}\Phi = &\int_{0}^{0.5\pi-\varepsilon} \mid 1 - H_0(e^{j\omega}) \mid^2 \cdot W_0 d\omega + \int_{0.5\pi+\varepsilon}^{\pi} \mid H_0(e^{j\omega}) \mid^2 \cdot W_1 d\omega + \\ &\int_{0.5\pi+\varepsilon}^{\pi} \mid 1 - H_1(e^{j\omega}) \mid^2 \cdot W_0 d\omega + \int_{0}^{0.5\pi-\varepsilon} \mid H_1(e^{j\omega}) \mid^2 \cdot W_1 d\omega\end{aligned} \tag{3.40}$$

Here ε is the parameter that has an effect on the desired transition bandwidth, and W_1 and W_0 are the weighting factors with $W_1 + W_0 = 1$. In our design example, we choose transition band of $\varepsilon = 0.2\pi$ and equal weight for pass band and stop band error. We depict the frequency responses for different lengths in Figure 3.10.

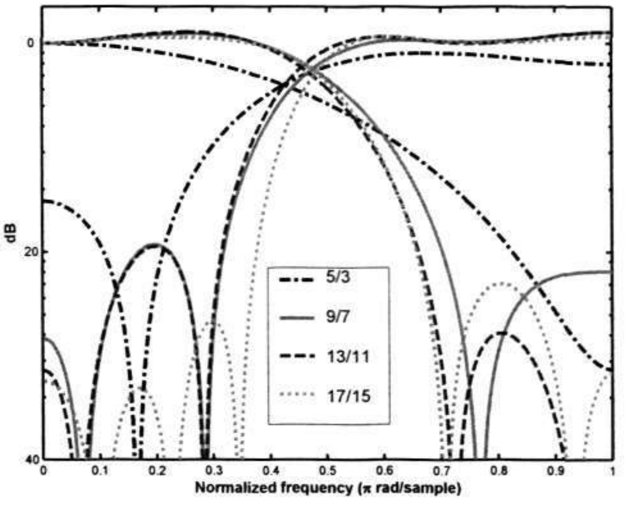
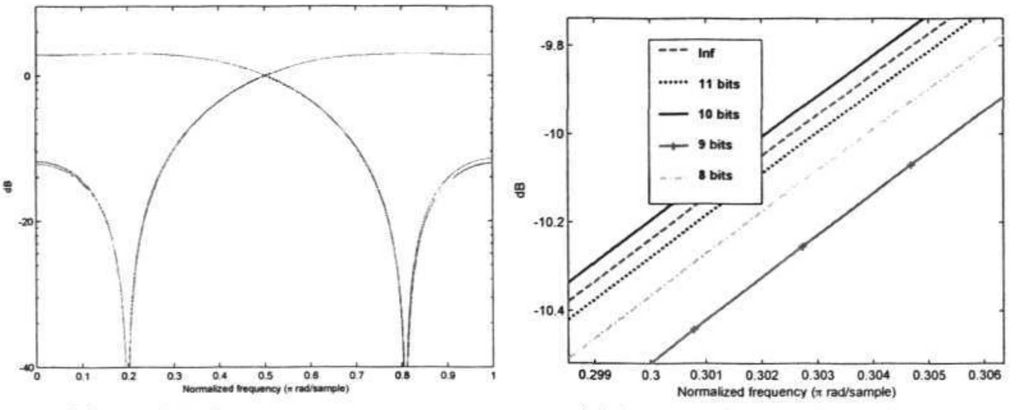


Figure 3.10: Frequency response of the designed FBs for different length

The proposed lifting structure is also applicable to directly design a FB with desired precision. This is similar to the example given in subsection 3.5.3. However, instead of truncating an existing FB, the lifting coefficients of FB with assigned bits are optimized according to some criterion. For the following example, we still choose the same cost function as above, which involves the filter shape. Since the coefficients are not relaxed to be real number any more, they may be discrete. A particle swarm optimization (PSO) based optimization routine is employed. The optimization result is given in Table 3.6. As we can see from Figure 3.11, the frequency responses are quite close for different precisions. It is reasonable to believe that the performance will also be close, if we impose some other desired property during the optimization.



(a)The full frequency response (b)A zoomed part of left figure

Figure 3.11: Frequency response of the designed FBs for different precision

	a_5	a_4	a_3	a_2	a_1
Inf. bits	1	0.14748438	-0.3298116	0.32645567	-0.1420073
11 bits	1023/1024	147/1024	-335/1024	337/1024	-149/1024
10 bits	511/512	43/256	-11/32	161/512	-31/256
9 bits	255/256	15/128	-5/16	45/128	-43/256
8 bits	127/128	19/128	-43/128	21/64	-17/128

Table 3.6: Lifting coefficients for the designed FBs

We can also design the FB in the following way, similar to [28]. First design a (high pass or low pass) filter, say $H_i(z)$. Note that the designed filter should not have zeros at both $z = k$ and $z = -k$ with $k \neq 0$. Otherwise the corresponding polyphase components will have zeros at $\pm\sqrt{k}$, which can be easily obtained by Eq.

(3.13). This indeed destroys the PR condition in Eq. (3.14). Fortunately, there are still many approaches to find such filters, such as REMEZ routine. After we design a LP filter using REMEZ method, we can use lifting to find its PR counterpart $H_{|1-i|}(z)$. Note that the obtained lifting structure is fixed. Generally, $H_{|1-i|}(z)$ is not as good as $H_i(z)$ in term of filter shape etc. (as depicted in Figure 3.12). To achieve a practical FB with a better filter $H_{|1-i|}(z)$, we cascade one more lifting structure, which has a symmetric polynomial and may have more than one free parameters. We give a type-A example in Figure 3.12, which has a fixed high pass filter of 8 taps and a corresponding low pass filter of the same length. We refine the low pass filter with 3 similar lifting structures of different length. They have 1/2/3 free parameters in the lifting structure respectively, and result in low pass filters of 12/16/20 taps (denoted as LoP1-LoP3) respectively. As seen from the figure, we can get a comparable low-pass filter with only 2 free parameters and a better one with 3. We may further refine the low pass filter when necessary.

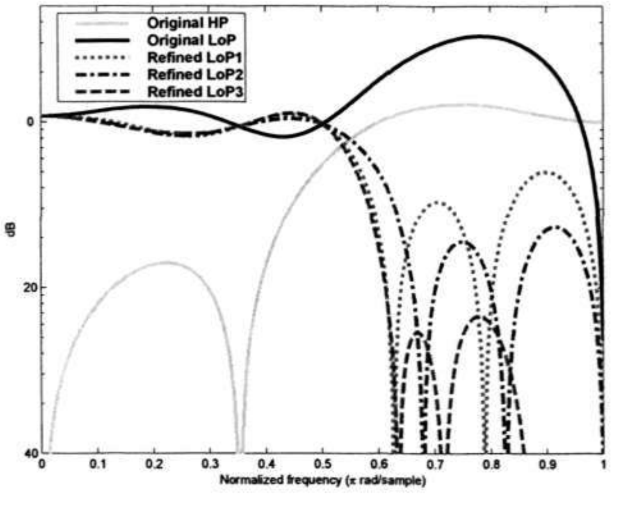


Figure 3.12: Pre-designed high pass filter and its paired low pass filters

3.6 Generalized lifting scheme

In fact, when dealing with singular type-A LPPRFB, we have already introduced the generalized lifting scheme (GLS). Here we make an effort to extend the essential idea behind lifting scheme to a wider range by exploring it in a systematic way. Though the following discussion is not complete, being a solution from another viewpoint, it brings us some insight into the lifting scheme.

The reason why lifting scheme is favored, to some extent, is its simple inverse. Namely, the inverse structure is similar to the designed analysis bank and can be obtained immediately after the analysis bank is designed. This idea is extended to a wider range of other forms, beyond only ladder structures used for classic lifting scheme. Now we can formulate this problem as to find functions $f_1, \cdots f_4$ of B, such that

$$\begin{bmatrix} f_1(B) & f_2(B) \\ f_3(B) & f_4(B) \end{bmatrix}^{-1} = \begin{bmatrix} f_1(-B) & f_2(-B) \\ f_3(-B) & f_4(-B) \end{bmatrix} \tag{3.41}$$

As we know,

$$\begin{bmatrix} f_1 & f_2 \\ f_3 & f_4 \end{bmatrix}^{-1} = \frac{1}{f_1 f_4 - f_2 f_3} \begin{bmatrix} f_4 & -f_2 \\ -f_3 & f_1 \end{bmatrix} \tag{3.42}$$

Therefore, to satisfy the condition in Eq. (3.41), sufficient conditions can be derived as

$$f_1 f_4 - f_2 \cdot f_3 = 1 \tag{3.43}$$

$$f_1(-B) = f_4(B) \tag{3.44}$$

$$f_{2,3}(-B) = -f_{2,3}(B) \tag{3.45}$$

Immediately, we can derive the following three cases, where Eq. (3.41) is satisfied.

Case A $f_1(B) = f_4(B) = 1$, and either $f_2(B) = 0$ or $f_3(B) = 0$, e.g. $\begin{bmatrix} 1 & 0 \\ f_3(B) & 1 \end{bmatrix}$ or $\begin{bmatrix} 1 & f_2(B) \\ 0 & 1 \end{bmatrix}$

Case B $f_1(B) = 1 \pm B$, $f_4 = 1 \mp B$, $f_2(B) = \pm kB$ and $f_3(B) = \mp k^{-1}B$, e.g. $\begin{bmatrix} 1+B & kB \\ -k^{-1}B & 1-B \end{bmatrix}$

Case C $f_1(B) = \frac{1}{1 \pm B}$, $f_4 = \frac{1}{1 \mp B}$, $f_3(B) = \frac{1}{2}\left(\frac{1}{1-B} - \frac{1}{1+B}\right)$ and $f_2(B) = B$, e.g. $\begin{bmatrix} \frac{1}{1+B} & B \\ \frac{1}{2}\left(\frac{1}{1-B} - \frac{1}{1+B}\right) & \frac{1}{1-B} \end{bmatrix}$

Remark: Obviously, Case A is actually the classical lifting we introduced in Chapter 2.3. Case B is the lifting-like structure introduced to address the singularity of type-A LPPRFB when parameter $k = 1$. The parameter k brings another degree of freedom, which may improve the filter performance. Case C is of less interest due to its IIR nature. All of them share the feature that whatever (such as polynomial in z) is chosen for B, the PR condition can be met exactly. It may be interesting to explore more sophisticated structures that can satisfy Eq. (3.41).

If we regard the lifting matrix as the sum of two matrices, i.e., $\mathbf{I} + \mathbf{B}$, PR condition requires,

$$(\mathbf{I} + \mathbf{B})(\mathbf{I} - \mathbf{B}) = \mathbf{I} \tag{3.46}$$

This can translate into the requirement $\mathbf{B} \cdot \mathbf{B} = 0$. In fact, this kind of matrix is called nilpotent matrix in literature [47]. However, for multiple channel case, a nilpotent matrix can mean more than $\mathbf{B} \cdot \mathbf{B} = 0$. A nilpotent matrix with order q is an $M \times M$ matrix such that

$$\mathbf{B}^q = \mathbf{0} \tag{3.47}$$

for some positive integer q ($1 \leq q \leq M$, size of $\mathbf{B}$). Hence

$$(\mathbf{I} + \mathbf{B})^{-1} = \mathbf{I} + \sum_{i=1}^{q-1} (-\mathbf{B})^i \tag{3.48}$$

Therefore, by varying the order q the synthesis (inverse) structure can have a different length from the analysis structure. This approach has been reported in [48], where the details of construction are given.

3.7 Summary

In this chapter, we first study the relation between lifting structure and 2-channel LPPRFB. For type-B, an effective lifting structure is presented. We prove it to be complete by factorizing an arbitrary type-B FB. For type-A we propose a lifting factorization which is partly equivalent to lattice factorization. The singular type-A is also included in the factorization. For both types we can impose the LPPR property to the structure. We demonstrate the factorization by some examples. We illustrate the structural properties using reduced word-length on some FBs. We also show how the vanishing moment may be retained in reduced word-length realization. As an application, we demonstrate the use of the lifting structure for reduced word-length FBs in image coding. We show that the performance of the FBs are similar while we can realize their lifting coefficients with less bits. Lastly, we design some FBs of both types using the proposed factorization. Design is done for full precision as well as in reduced word-length to show no loss of performance. Further, design of a FB with a pre-designed filter is also illustrated. In conclusion, the proposed factorizations are useful in a multitude of ways. We also investigate the so-called generalized lifting scheme and obtain some insights.

Chapter 4

M-channel PRFB based on lifting scheme

In Chapter 2, the factorization of 2-channel PRFB and LPPRFB have been introduced in detail. It is natural to consider extending the lifting scheme to multiple channel or M-channel PRFB. There are many applications of this kind in digital communications. For an instance, cosine modulated FBs are frequently used in DSL technology. For M-channel FBs, more flexibility of design are allowed due to the increased degree of freedom. For example, linear phase and paraunitary properties can coexist for an M-channel FB while this is forbidden for a 2-channel FB. Therefore, M-channel filter banks are desired in some circumstances. However, beside these advantages, M-channel PRFB design tasks are more complicated. In this chapter, lifting scheme is extended to the case of M-channel PRFB. We first study the lifting building blocks in the context of M-channel case. Then the factorization of M-channel PRFB is investigated. Finally, we apply the lifting structure to design M-channel FBs, specifically incorporating PR and LP properties respectively. This topic has been solved well in a language of lattice factorization by a number of pioneering works [12–14, 49–55]. In [49–55], this kind of problems has been studied completely in both theory and application. Novel and simplified lattice structure is also proposed and studied there. The factorization problems of M-channel PR FB have been hence simplified significantly, while the completeness is still guaranteed. Readers who are interested in how to solve the problems using lattice structure are strongly encouraged to read those excellent works. Here we mostly focus on the solutions based on lifting.

Now we start with the building blocks for M-channel lifting scheme.

4.1 Building blocks for M-channel lifting scheme

4.1.1 M-channel lifting scheme

Lifting scheme for 2-channel is fundamentally based on the following fact from matrix multiplication,

$$\begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & a+b \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 0 \\ a & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ b & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ a+b & 1 \end{bmatrix}$$

When we choose $b = -a$, we can find that the product equals to identity. We address this property as the sign changing inverse in the following. From these equations, we can easily extend them to multiple channel case.

$$\begin{aligned} &\begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times n} \\ \mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix}\begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times n} \\ \mathbf{B}_{n\times m} & \mathbf{I}_n \end{bmatrix} = \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times n} \\ (\mathbf{A}+\mathbf{B})_{n\times m} & \mathbf{I}_n \end{bmatrix} \\ \text{and } &\begin{bmatrix} \mathbf{I}_m & \mathbf{A}_{n\times m} \\ \mathbf{0}_{m\times n} & \mathbf{I}_n \end{bmatrix}\begin{bmatrix} \mathbf{I}_m & \mathbf{B}_{n\times m} \\ \mathbf{0}_{m\times n} & \mathbf{I}_n \end{bmatrix} = \begin{bmatrix} \mathbf{I}_m & (\mathbf{A}+\mathbf{B})_{n\times m} \\ \mathbf{0}_{m\times n} & \mathbf{I}_n \end{bmatrix} \end{aligned} \tag{4.1}$$

Similarly if we choose $\mathbf{A}_{n\times m} = -\mathbf{B}_{n\times m}$, we can get the inverse. For example, when $M = m + n = 4$, only three kinds of matrices and their transposed versions have this property. They are:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ a_1 & 1 & 0 & 0 \\ a_2 & 0 & 1 & 0 \\ a_3 & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a_1 & a_2 & 1 & 0 \\ a_3 & a_4 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ a_1 & a_2 & a_3 & 1 \end{bmatrix} \tag{4.2}$$

Different from the situation with 2-channel FB, in the multiple channel case we may allow permutations. For example, we can swap arbitrary columns together with the corresponding rows and still keep the property of sign changing inverse. This can be proved by the following equations.

Since $\mathbf{P}^T \cdot \mathbf{P} = \mathbf{I}$

$$\Rightarrow \mathbf{P}^T \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times n} \\ \mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix} \mathbf{P} \cdot \mathbf{P}^T \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times n} \\ -\mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix} \mathbf{P} = \mathbf{I}_{m+n}$$

where $\mathbf{P} = \prod_i \mathbf{P}_i$ is a combination of permutation matrices. A permutation matrix $\mathbf{P}_i$ is obtained by permuting the rows of an $n \times n$ identity matrix according to some permutation of the row vectors or the column vectors. For example, for $n = 3$, matrices $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ are some possible permutation matrices.

Note that the above result is valid even if the row permutation matrix is different from the column permutation matrix. However, in such a case the inverse is not merely sign changing but some components are swapped, too.

For lifting structure, a matrix should have diagonal entries as 1's, and the so-called sign changing inverse. It is desirable to know if there exist other such matrices apart from that in (4.1). The following theorem addresses this issue.

Theorem 4.6 *It is necessary and sufficient for all matrices with 1's in the diagonal and any arbitrary values off the diagonal to have the exact form or permutated form as that in (4.1) if their inverse can be acquired by changing the signs of all non-diagonal entries.*

Proof: The sufficient part is obvious from (4.1). We show below the necessary part. We can express the simple inverse property as:

$$(\mathbf{I} + \mathbf{B}) \cdot (\mathbf{I} - \mathbf{B}) = \mathbf{I}, \tag{4.3}$$

where $\mathbf{B}$ is a matrix with 0's in the diagonal. Due to $\mathbf{P}^T \cdot \mathbf{P} = \mathbf{I}$, equivalently we require

$$\mathbf{P}^T(\mathbf{I} + \mathbf{B})\mathbf{P} \cdot \mathbf{P}^T(\mathbf{I} - \mathbf{B})\mathbf{P} = \mathbf{I}, \tag{4.4}$$

We don't restrict the permutation matrix $\mathbf{P}$ to have certain form. Then the above equation should hold for all possible $\mathbf{P}$'s. With some $m, n, \mathbf{P}$, we can always rewrite the above matrix as

$$\mathbf{P}^T(\mathbf{I}+\mathbf{B})\mathbf{P} = \mathbf{H} = \left(\begin{bmatrix} \mathbf{I}_m & 0 \\ \mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{m\times m} & \mathbf{L}_{m\times n} \\ 0 & 0 \end{bmatrix} \right)$$

and $\mathbf{K}_{m\times m}$, $\mathbf{L}_{m\times n}$ are determined by $\mathbf{B}$ and $\mathbf{P}$. Assume $\mathbf{H}$ has a simple inverse as we stated above, namely:

$$\left(\begin{bmatrix} \mathbf{I}_m & 0 \\ \mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{m\times m} & \mathbf{L}_{m\times n} \\ 0 & 0 \end{bmatrix} \right)^{-1} = \left(\begin{bmatrix} \mathbf{I}_m & 0 \\ -\mathbf{A}_{n\times m} & \mathbf{I}_n \end{bmatrix} + \begin{bmatrix} -\mathbf{K}_{m\times m} & -\mathbf{L}_{m\times n} \\ 0 & 0 \end{bmatrix} \right) \quad (4.5)$$

It is equivalent to:

$$\begin{aligned} &\left(\begin{bmatrix} \mathbf{I}_m & 0 \\ \mathbf{A} & \mathbf{I}_n \end{bmatrix} + \begin{bmatrix} \mathbf{K} & \mathbf{L} \\ 0 & 0 \end{bmatrix} \right) \cdot \left(\begin{bmatrix} \mathbf{I}_m & 0 \\ -\mathbf{A} & \mathbf{I}_n \end{bmatrix} + \begin{bmatrix} -\mathbf{K} & -\mathbf{L} \\ 0 & 0 \end{bmatrix} \right) \\ =&\mathbf{I}_{m+n} + \begin{bmatrix} -\mathbf{K} & -\mathbf{L} \\ -\mathbf{A}\cdot\mathbf{K} & -\mathbf{A}\cdot\mathbf{L} \end{bmatrix} + \begin{bmatrix} \mathbf{K}-\mathbf{L}\cdot\mathbf{A} & \mathbf{L} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{K}\cdot\mathbf{K} & -\mathbf{K}\cdot\mathbf{L} \\ 0 & 0 \end{bmatrix} \\ =&\mathbf{I}_{m+n} + \begin{bmatrix} -\mathbf{K}\cdot\mathbf{K}-\mathbf{L}\cdot\mathbf{A} & -\mathbf{K}\cdot\mathbf{L} \\ -\mathbf{A}\cdot\mathbf{K} & -\mathbf{A}\cdot\mathbf{L} \end{bmatrix} \end{aligned}$$

If we want the assumption to hold, we need the second item in the last step of the above equation to be zero. Given arbitrary $\mathbf{A}_{n\times m}$, $\mathbf{K}_{m\times m}$ and $\mathbf{L}_{m\times n}$, the only solutions are 1) $\mathbf{K}$=0 and $\mathbf{L}$=0 for arbitrary $\mathbf{A}$, and 2) $\mathbf{K}$=0 and $\mathbf{A}$=0 for arbitrary $\mathbf{L}$. In fact, we can regard these two solutions as the permutation of each other. ∎

With the above theorem we can conclude that if we want to construct an M-channel FB via lifting steps, we must have the building blocks of the form as appeared in (4.1) or their permutated version.

4.1.2 Lifting based factorization for triangular matrix

Lemma 4.1 *Any lower or upper triangular matrix with all diagonal entries equal to 1 can be expressed as the multiplication of lifting matrices of the form in (4.1).*

Proof: The easiest way is to split the original matrix according to the rows and pre-multiply them from the bottom row. It can be depicted as Fig 4.1 and the

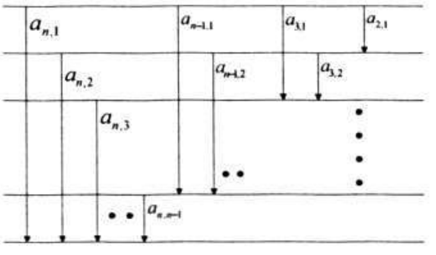


Figure 4.1: The diagram of lifting based factorization

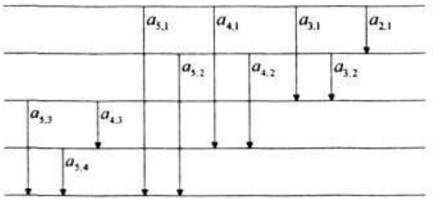


Figure 4.2: One possible lifting based factorization for 5-channel

following equation:

$$\begin{bmatrix} 1 & & & & \\ a_{2,1} & 1 & & & \\ \vdots & & \ddots & & \\ a_{n-1,1} & a_{n-1,2} & \cdots & 1 & \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n-1} & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ a_{2,1} & 1 & & & \\ 0 & 0 & 1 & & \\ \vdots & & & \ddots & 1 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ \vdots & & \ddots & & \\ 0 & 0 & \cdots & 1 & \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n-1} & 1 \end{bmatrix}$$

But row-wise is not the only possible way to factorize. We can factorize it differently. For example when $n = 5$, we can factorize it in the following way (also depicted in Figure 4.2):

$$\begin{bmatrix} 1 & & & & \\ a_{2,1} & 1 & & & \\ a_{3,1} & a_{3,2} & 1 & & \\ a_{4,1} & a_{4,2} & a_{4,3} & 1 & \\ a_{5,1} & a_{5,2} & a_{5,3} & a_{5,4} & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ a_{2,1} & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ a_{3,1} & a_{3,2} & 1 & & \\ a_{4,1} & a_{4,2} & & 1 & \\ a_{5,1} & a_{5,2} & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & a_{4,3} & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & a_{5,3} & a_{5,4} & 1 \end{bmatrix}$$

The upper triangular case may be proved in a similar fashion. ∎

This lemma further relax the requirement for matrices that have the sign changing inverse property. We can consider the triangular matrix during factorization instead of the more restricted version of Eq. (4.1).

4.2 M-channel PRFB factorization based on lifting

To tackle the problem of factoring multiple but univariate polynomials, the extended Euclidean algorithm is used popularly.

Definition 4.1 *The extended Euclidean algorithm (EEA) is an algorithm to determine the greatest common divisor among more than two polynomials.*

For PR condition, all polyphase components of each filter in the bank should not have any common zero, otherwise the determinant of polyphase matrix may vanish at these zeros, which contradicts the PR condition. Therefore, the following lemma can be obtained by applying the EEA.

Lemma 4.2 *For a vector* $\mathbf{p}(z)$*, whose elements are all coprime, there exist a finite number of* M*-channel lifting steps, which reduce* $\mathbf{p}(z)$ *to the vector* $\mathbf{e}_1 = [1, 0, \cdots, 0]^T$.

Proof: As previously introduced, according to Euclidean algorithm, for two coprime polynomials, say a and b, there exist polynomials q and r, s.t. $a = qb + r$ and $S(r) < S(b)$. Hence we can choose the element with the lowest order in $\mathbf{p}(z)$ to practise this algorithm, i.e.

$$\begin{bmatrix} p_1'(z) \\ p_2(z) \\ p_3'(z) \\ \vdots \\ p_M'(z) \end{bmatrix} = \begin{bmatrix} 1 & -q_1(z) & & & \\ & 1 & & & \\ & -q_3(z) & 1 & & \\ & \vdots & & \ddots & \\ & -q_M(z) & \cdots & & 1 \end{bmatrix} \begin{bmatrix} p_1(z) \\ p_2(z) \\ p_3(z) \\ \vdots \\ p_M(z) \end{bmatrix} \tag{4.6}$$

Here we assume $p_2(z)$ is the element with the lowest order. After this step, $p_2(z)$ is no longer the lowest degree element. Therefore, another round of reduction can be practised. Since all the elements are coprime between each other, their common factor is 1. Therefore, the above procedure can be repeated until

$$\begin{bmatrix} 0 \\ k \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \prod_i \mathbf{Q}_i \begin{bmatrix} p_1(z) \\ p_2(z) \\ p_3(z) \\ \vdots \\ p_M(z) \end{bmatrix} \tag{4.7}$$

Again we assume $p_2(z)$ is the last item to be reduced. With two more steps as follows, the lemma is obtained.

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & & & \\ -k & 1 & & & \\ & 0 & 1 & & \\ & \vdots & & \ddots & \\ & 0 & \cdots & & 1 \end{bmatrix} \begin{bmatrix} 1 & 1/k & & & \\ & 1 & & & \\ & 0 & 1 & & \\ & \vdots & & \ddots & \\ & 0 & \cdots & & 1 \end{bmatrix} \begin{bmatrix} 0 \\ k \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{4.8}$$

Note that matrices post multiplied as above are all the so-called M-channel lifting building blocks as explained in the previous section. Also note that if the vector to be reduced is a column vector, the matrices for reduction need to be applied in pre-multiplication. ∎

Remark: The approach of EEA also appeared in [8, 9, 56] and is closely linked to the completion problem by Bose in Chapter 1 of [57]. The algorithm to reduce the order of a given matrix presented in the lemma is also used extensively in multivariate control and with a more generic setting that does not require the matrix to be square. For example, Rosenbrock [58] and Wolovich [59] both proposed similar approaches in their works.

As mentioned, for a M-channel PRFB, its polyphase matrix $\mathbf{E}(z)$ should have a determinant of 1 (it's no different to requiring the determinant being a monomial if Laurent polynomials are employed). Therefore, every column and every row can form a vector we mentioned in Lemma 4.2. Therefore, by following the above steps for the first column and the first row respectively, we can obtain

$$\begin{bmatrix} 1 & 0 \cdots 0 \\ 0 & \\ \vdots & \mathbf{E}_{M-1}(z) \\ 0 & \end{bmatrix} = \prod_i \mathbf{L}_i^{(M)}(z)\mathbf{E}(z)\prod_j \mathbf{R}_j^{(M)}(z) \tag{4.9}$$

Obviously, $\mathbf{E}_{M-1}(z)$ is still a matrix with a determinant of 1. Hence Eq. (4.9) can be repeated to obtain another set of matrices $\mathbf{L}_i^{(M-1)}(z)$, $\mathbf{R}_i^{(M-1)}(z)$. To perform in the original size, we can set these matrices as

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & \mathbf{L}^{(M-1)}(z) & \text{or} & \mathbf{R}^{(M-1)}(z) \\ 0 & & & \end{bmatrix} \tag{4.10}$$

Continuing this way, a PRFB with polyphase matrix $\mathbf{E}(z)$ with a determinant of 1 can be reduced to an identity by the following steps,

$$\mathbf{I} = \prod_i \mathbf{L}_i(z)\mathbf{E}(z)\prod_j \mathbf{R}_j(z) \tag{4.11}$$

Since all $\mathbf{L}(z)$'s and $\mathbf{R}(z)$'s are lifting building blocks, their inverse can be simply obtained. Hence it is easy to rewrite Eq. (4.11) as

$$\mathbf{E}(z) = \prod_i \mathbf{L}'_i(z) \prod_j \mathbf{R}'_j(z) \tag{4.12}$$

where $\mathbf{L}'_i(z)$ and $\mathbf{R}'_j(z)$ are corresponding inverse matrices in reverse order.

This type of approaches are reported in [8,9,56,60]. The problems of construction are also addressed in [56,60]. Although we can constrain the matrices that can be used for lifting construction with Eq. (4.1), when number of channels is large, especially when we take into account possible permutations, the possible structures are numerous and highly non-unique. In [56] and [60], Amaratunga et al proposed a lifting scheme for M-channel FB based on the Householder matrix from [3]. By replacing every building block of a PUFB or PRFB with lifting steps, when these blocks are proved to be complete for construction in [3], they managed to present a generalized scheme for M-channel FB. The problem is that some of the lifting coefficients are determined by other design parameters and therefore are not free. This will lead to the loss of PU property when we consider coefficient quantization.

In [61,62], the more special case that the PRFB being also orthogonal is investigated. The authors further rely their approach on the factorization of unimodular matrix. Since we are less interested in the orthogonal FB, we do not go further into their work.

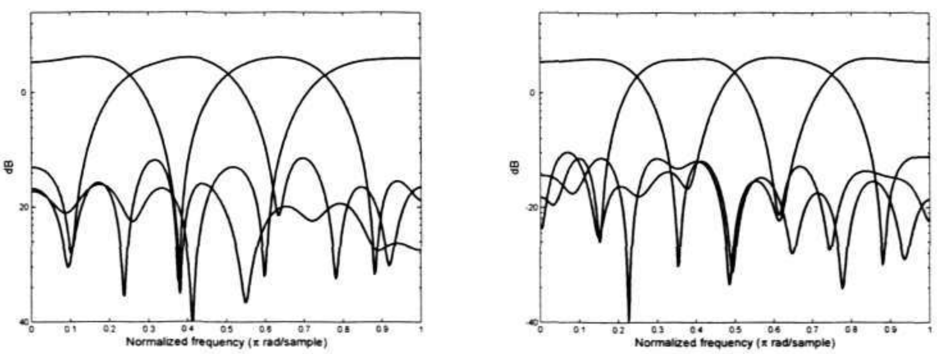
4.3 M-channel PRFB design based on lifting

4.3.1 Design examples of M-channel PRFB

Firstly, we construct some PRFBs using the structurally PR property of lifting. We apply three possible structures in designing 4-channel PRFB and compare their performance.

To obtain a satisfactory filter bank, structures as listed in Eq. (4.13) are cascaded and the parameters optimized using the unconstrained routines in MATLAB. Here $a_i(z) = m_i + n_i z^{-1}$ is chosen. In fact, the last structure is to some extent equivalent to the first one, which has been verified by their similar performance. We construct 4-channel PRFB based on these three types of lifting building blocks and their corresponding dual lifting steps. For the middle structure, we cascade 2 sections and for the other structures we cascade 3 such sections. Figure 4.3 is the frequency response for two of these FBs. The number of free parameters is 37 and 33, and the filter lengths are {27,24,24,24} and {20,20,16,16} for the first (or the last) structure and the middle structure respectively. There are possibly some redundant parameters.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ a_1(z) & 1 & 0 & 0 \\ a_2(z) & 0 & 1 & 0 \\ a_3(z) & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a_1(z) & a_2(z) & 1 & 0 \\ a_3(z) & a_4(z) & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ a_1(z) & a_2(z) & a_3(z) & 1 \end{bmatrix} \tag{4.13}$$



FB constructed by the middle structure   FB constructed by the first structure

Figure 4.3: The frequency response of lifting-based 4-channel filter banks

4.3.2 Incorporation of LP property

M-channel LPPRFB has been investigated for many years. Excellent pioneering works based on lattice structure for this kind of FBs have been reported in [12–14, 49–55].

Due to the difficulty of characterizing M-channel LPPRFB, all investigations are restricted to a subclass of FBs of this kind. For example, in [13, 14], the authors require the lengths of filters in analysis bank being in the form of $L_i = K_i M + \beta$, where integers $0 \leq \beta < M$ and $K_i \geq 1$.

According to the example in the previous section, one may think of constructing an LPPRFB by setting linear phase lifting steps, i.e. $a_i(z) = m_i + m_i z^{-1}$ in Eq. (4.13). However, such approach does not necessarily result in a LPPRFB in M-channel, when $M > 2$. For example, it is easy to verify that 3-channel FB $\mathbf{H}(z) = \mathbf{E}(z^M)\mathbf{d}(z)$ is not linear phase, where

$$\mathbf{E}(z) = \begin{bmatrix} 1 & 1+z^{-1} & 2+2z^{-1} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{d}(z) = \begin{bmatrix} 1 \\ z^{-1} \\ z^{-2} \end{bmatrix} \tag{4.14}$$

The problem with the above construction is the alignment of lifting steps. Recall the 2-channel case, where the symmetry center of lifting step is chosen carefully to align with the difference between symmetry centers of both channel. If we take this fact into account, it leads us to some positive results.

Similar to 2-channel, we focus on a class of M-channel LPPRFB, which starts from a trivial delay chain system. This corresponds to the type-B system as we define for 2-channel case. An LP filter $F(z)$, which is centered at z^c, should satisfy,

$$F(z^{-1}) = \pm z^{-2c} F(z) \tag{4.15}$$

Here $\pm$ corresponds to symmetry or antisymmetry of the filter. Therefore, for a filter bank, if all the filters have LP property and if the filters are centered at

$[1, z^{-1}, z^{-2}, \cdots]$ respectively, the following should be obtained,

$$\begin{bmatrix} H_0(z^{-1}) \\ \vdots \\ H_{M-1}(z^{-1}) \end{bmatrix} = \mathbf{D} \begin{bmatrix} 1 & & & \\ & z^2 & & \\ & & \ddots & \\ & & & z^{2M-2} \end{bmatrix} \begin{bmatrix} H_0(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix} \tag{4.16}$$

where $\mathbf{D}$ is a diagonal matrix comprised of $+1$ or -1 according to the symmetry of the corresponding filter.

Note that if we start from a delay chain (as described in Chapter 1), the requirement of symmetry centers can be satisfied, i.e. the filters are indeed centered at $[1, z^{-1}, z^{-2}, \cdots]$ respectively. The delay chain system also indicates that all filters are symmetric (rather than anti-symmetric), since an antisymmetric filter with odd length should have a zero center. Therefore, the matrix $\mathbf{D}$ corresponding to the symmetry is actually an identity matrix here.

Now we extend the delay chain system using a lifting scheme,

$$\begin{bmatrix} H'_0(z) \\ \vdots \\ H'_{M-1}(z) \end{bmatrix} = (\mathbf{I} + \mathbf{A}(z)) \begin{bmatrix} H_0(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix} \tag{4.17}$$

To keep LP property, applying Eq. (4.16) on both $H_i(z)$ and $H'_i(z)$, and using Eq. (4.17), the requirement on $\mathbf{A}(z)$ is,

$$(\mathbf{I} + \mathbf{A}(z^{-1})) \begin{bmatrix} 1 & & & \\ & z^2 & & \\ & & \ddots & \\ & & & z^{2M-2} \end{bmatrix} = \begin{bmatrix} 1 & & & \\ & z^2 & & \\ & & \ddots & \\ & & & z^{2M-2} \end{bmatrix} (\mathbf{I} + \mathbf{A}(z)) \tag{4.18}$$

The requirement in Eq. (4.18) can be translated as follows. Every lifting step should be linear phase, and the corresponding symmetry center is properly chosen to offset the difference of symmetry centers of channels involved in the lifting step. For example, the following lifting blocks are appropriate choices for the extension,

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ k_3 z^{-2}\gamma(z^{-2}) & k_2 z^{-1}\gamma(z^{-2}) & k_1\gamma(z^{-2}) & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & k'_3 z^2\gamma(z^2) \\ 0 & 1 & 0 & k'_2 z\gamma(z^2) \\ 0 & 0 & 1 & k'_1\gamma(z^2) \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\gamma(z) = 1 + z$.

By cascading the above structures and optimizing the parameters, the following 4-channel FB has been obtained. The lengths of filters are 9,9,9,11 respectively. The frequency responses of the filters are depicted in Figure 4.4.

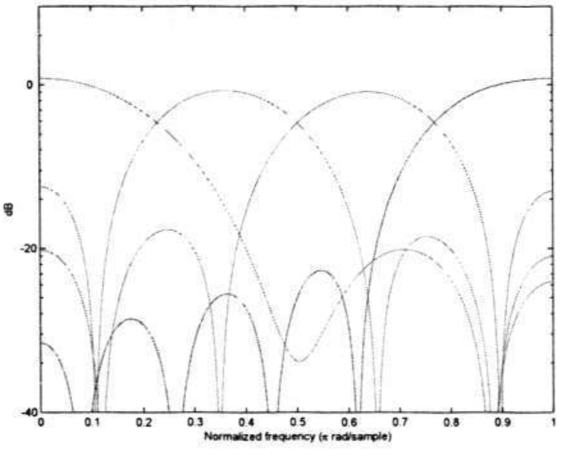


Figure 4.4: Lifting-based 4-channel filter banks with linear phase

After proposing the above approach, we have found a similar approach reported in [8]. The authors start from a trivial FB called seed FB, and use right-hand side extension or left-hand side extension to extend LP property for a longer FB. Our approach is similar to their left-hand side extension.

4.4 Summary

In this chapter, the relation between lifting scheme and M-channel PRFB is investigated. We first study lifting building blocks in M-channel case and find the building block that is necessary to be a direct extension of 2-channel building block in the sense that M-channel building block is the block matrix version of 2-channel building block or its permutation. We also see how to express an M-channel triangular matrix with lifting steps defined above. Towards factorization side, we introduce how to factorize a PRFB into lifting steps using extended Euclidean algorithm. Towards construction side, we first give an example of a PRFB design based on lifting steps. Further, we show how to incorporate LP property on it. Though the performance of designed FB in the terms of frequency response is not fabulous enough, being an alternative and with some inherited advantages from lifting scheme, these approaches still may be useful.

Chapter 5

Resampling patterns and ideal resampling filters for 2D PRFB

5.1 Multidimensional filter banks

From this chapter on, our focus is extended to the FB in higher dimension, i.e. multidimensional (nD) filter banks (FBs). nD FBs have a potentially wide range of application in modern signal processing, such as image processing and deconvolution for communication. Therefore, this topic has attracted a lot of attention from researchers in the last decade. However, most existing nD systems are separable which means the transfer function is actually the tensor product of the transfer functions along every dimension. These structures lead to separable sampling and separable filtering, which are indeed favorable in implementation. However, people are increasingly willing to trade a better performance with an increased complexity especially when implementation becomes more and more inexpensive. Nonseparable filter bank is such a case in point. Though it requires more resource in terms of both memory and computation, the better performance over its separable counterpart still makes it attractive. This is also the motivation of our research introduced in the following chapters.

Generally speaking, the problem of nD FBs are not a direct extension of their 1D counterparts, since the nature of the problems may be thoroughly different when the number of variables changes from one to more than one. However, the similarity between 1D and nD problems indeed exists under some special circumstances. Fortunately, many excellent works in literature help us a lot to realize these differences and similarities. Before we get down to our topics, we first have a look at existing works of this kind.

A great percentage of current research works on nD FB may fall into one of the following five classes.

- In the first class of works such as [16–22], nD sampling, which is crucial to nD FB, is studied. They are focused on the issues related to resampling matrix, such as the commutativity between downsamplers and upsamplers.

- The second class, e.g. [63–67], parallel to the mainstream of 1D work on PRFB, mainly studies the nD FB from a viewpoint of matrix theory and achieves some similar results. These concentrate on the properties of nD FB, such as perfect reconstruction property, paraunitary property, linear phase, etc.

- The third class of work, including [68–72], mainly studies how to obtain an nD FB from a 1D prototype, especially from an approach called McClellan transform. The studied transforms usually can bring a part of, if not entire, desired properties from 1D to nD, such as symmetry.

- There is another class of work studying the directional frequency response of each subband filter. To discriminate different directions by different filters in an nD (usually 2D) FB is important in some applications. Contourlet [73] is a case in point. Other similar works appear in [74–77] etc.

- In the last class of works, problems of nD FB are formulated into the language of polynomials. Especially, Groebner basis is a main tool for them to tackle the problems. [23, 78–86] are all works of this class, where [86] contains a comprehensive survey of them.

What we focus in the thesis and what we investigate in the following chapters are within the scopes of the first and last categories. We also cover some scope of the third class.

In this chapter, a study on how to enumerate and parameterize resampling patterns under a given ratio is presented after some preliminaries are reviewed to make it self-contained. We also propose a new approach to obtain an ideal filter shape for a given downsampling matrix such that the resulting filter bank can be alias-free.

5.2 Resampling of a multidimensional signal

Two essential building blocks of a multirate system are the downsampler for the analyzed signal and the upsampler for the synthesized signal. For 1D signal, uniform resampling is intuitively simple and hence popular. It has been thoroughly studied in many publications such as [3]. However, they become exponentially varied and complicated with the dimension when higher dimensional signal is considered, since there are many ways to choose the sampling lattice. This born difficulty has attracted quite a few research concentration in recent decades [16–22]. For the sake of an illustrative delivery, we address the problem mainly for the 2D case. Since upsampling is just the reciprocal counterpart of downsampling, in the following we limit our study to downsampling matrices without loss of generality.

In 2D systems, the downsampler is a 2×2 non-singular integer matrix, denoted as $\mathbf{M}$. The downsampling matrix $\mathbf{M}$ keeps those samples that are on the lattice generated by $\mathbf{M}$. The lattice generated is the set of integer vectors $\mathbf{m}$ such that $\mathbf{m} = \mathbf{Mn}$ for some integer vector $\mathbf{n}$. For example, in Figure 5.1 three possible lattices are shown. They might correspond to following downsampling matrices $\mathbf{M}=\begin{pmatrix}1 & 0\\ 0 & 2\end{pmatrix}$, $\mathbf{M}=\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}$, $\mathbf{M}=\begin{pmatrix}2 & 0\\ 0 & 1\end{pmatrix}$ respectively. Be aware that, the usage of 'might' above actually indicates that there exists an infinite number of different downsampling matrices for each lattice in Figure 5.1. In the following, we regard different downsampling matrices having the same lattice as the same **downsampling pattern**.

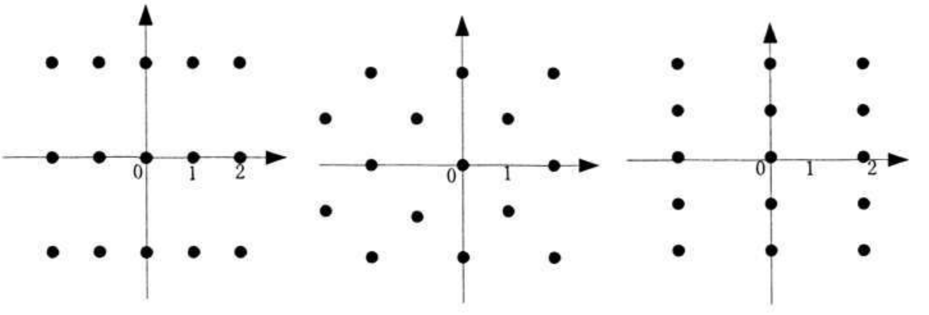


Figure 5.1: Three possible lattices for $|\mathbf{M}|= 2$

It is well-known that any two matrices of the same pattern are related by right-multiplication of an integer unimodular matrix whose determinant is ± 1 [3]. It can be explained in the following way. Given downsampling matrices $\mathbf{M}$ and $\hat{\mathbf{M}}$ with $\hat{\mathbf{M}} = \mathbf{M} \cdot \mathbf{V}$, where $det(\mathbf{V}) = 1$, any point, say $\mathbf{m}$, on the lattice generated by $\mathbf{V}$ can be expressed as $\mathbf{m} = \mathbf{Mn}$. When downsampled with $\hat{\mathbf{M}}$,

$$\mathbf{m}' = \hat{\mathbf{M}}\mathbf{n} = \mathbf{MVn} = \mathbf{Mn}' \tag{5.1}$$

where $\mathbf{n}' = \mathbf{Vn}$. Since $det(\mathbf{V}) = 1$, $\mathbf{n}'$ is nothing but a rearrangement of $\mathbf{n}$, and they correspond to the same set of samples. Therefore, the set $\mathbf{m}'$ is the same as $\mathbf{m}$, i.e. they are the same lattice. However, when a unimodular matrix is post multiplied, the above approach is invalid. Generally speaking, matrices which are related by post (or left) multiplication of an integer unimodular matrix may correspond to different lattices.

For a given downsampling matrix $\mathbf{M}=\begin{pmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{pmatrix}$ and a given lattice $x[\mathbf{n}]$ ($\mathbf{n} = [n_1, n_2]^T, n_1, n_2$ are integers), the downsampled lattice is $y[\mathbf{n}] = x[\mathbf{Mn}]$. Corresponding downsampling pattern can be obtained by repeating vectors from $\mathbf{M}$ [3]. An example is shown in Figure 5.2. The vertices form a set, which is a downsampling pattern as defined above. If there exist integer matrix $\mathbf{M}'$ and unimodular integer matrix $\mathbf{V}$ such that, $\mathbf{M} = \mathbf{M}'\mathbf{V}$, $\mathbf{M}'$ is regarded equivalent to $\mathbf{M}$, denoted as $\mathbf{M}' \leftrightarrow \mathbf{M}$. Downsampling ratio is defined as $m = |\mathbf{M}|$. Therefore, for the example in Figure 5.2, $m = 3$. The shaded area in Figure 5.2 is also addressed as the fundamental parallelepiped, or FPD($\mathbf{M}$) in literature. Mathematically,

$$\text{FPD}(\mathbf{V}) = \text{set of all points } \mathbf{Vx} \text{ with } \mathbf{x} \in [0,1)^M \tag{5.2}$$

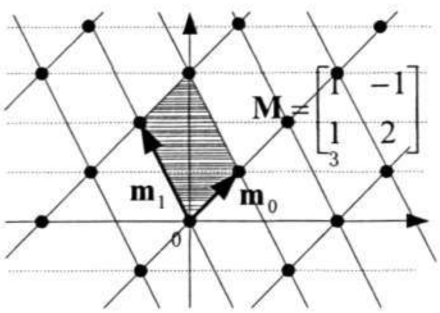


Figure 5.2: The lattice generated by downsampling matrix $\mathbf{M}$

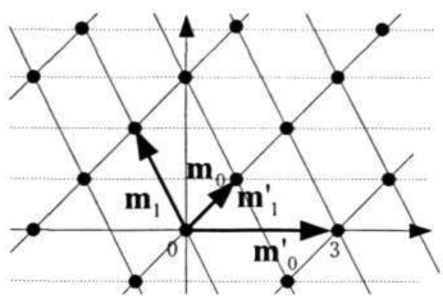


Figure 5.3: The same lattice for different downsampling matrices

5.3 Downsampling patterns enumeration and parametrization

Resampling matrices of the same pattern may share many properties in common [3]. For example, the filter bank that is alias-free for a downsampling matrix is also alias-free for other downsampling matrices of the same pattern after necessarily rearranging samples [3]. Exploration of infinite number of downsampling matrices can be reduced to exploration of some representatives of distinct patterns. It is interesting to ask whether under a given resampling ratio, the number of different lattices is finite and how to enumerate and parameterize them. The answer can also facilitate other related research. This is precisely the scope of this section and it is motivated by the lack of any such existing enumeration in 2D. Since upsampling is the reciprocal counterpart of downsampling, we limit our study to downsampling matrices without loss of generality.

According to the different types of m, the discussion has been divided into two sections. One is for downsampling ratio m being prime. The other is for m being composite.

5.3.1 Downsampling patterns with prime downsampling ratio

For a given downsampling matrix $\mathbf{M}=\begin{pmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{pmatrix}$, the following equivalence is always obtainable.

Lemma 5.3 $\mathbf{M} \leftrightarrow \begin{pmatrix} m & n \\ 0 & 1 \end{pmatrix}$ *or* $\mathbf{M} \leftrightarrow \begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$ *for some n, where $|\mathbf{M}|= m$ is a prime number and n is an integer.*

Lemma 5.3 provides normalization of arbitrary downsampling matrix. This can be geometrically justified as follows. As we explained above, the pattern is generated by repeating the parallelogram formed by vectors in the downsampling matrix. Conversely for this generated lattice, the forming vectors can be chosen as needed. Here since m is prime, they can always be chosen to be $\{\begin{bmatrix} m \\ 0 \end{bmatrix}, \begin{bmatrix} n \\ 1 \end{bmatrix}\}$ or $\{\begin{bmatrix} 1 \\ n \end{bmatrix}, \begin{bmatrix} 0 \\ m \end{bmatrix}\}$.

For example, we have the downsampling matrix $\mathbf{M} =[\mathbf{m}_0, \mathbf{m}_1]=\begin{pmatrix} 1 & -1 \\ 1 & 2 \end{pmatrix}$ in Figure 5.2. From this lattice we can also get another two vectors $\mathbf{m}_0'$ and $\mathbf{m}_1'$ shown in Figure 5.3. If we choose the vectors as above, we obtain $\mathbf{M}'=[\mathbf{m}_0', \mathbf{m'}_1]=\begin{pmatrix} 3 & 1 \\ 0 & 1 \end{pmatrix}$. The old and the new downsampling matrices are related by $\mathbf{M}\begin{pmatrix} 2 & 1 \\ -1 & 0 \end{pmatrix}=\mathbf{M}'$. In Figure 5.3, the dashed lines denote the new parallelogram generated by $\mathbf{M}'$. Note that half of the dashed lines is overlapped by the original lines generated by $\mathbf{M}$. Although the parallelograms are different, the resampled points are the same, namely the pattern is the same.

Lemma 5.4 $\begin{pmatrix} 1 & 0 \\ m+n & m \end{pmatrix} \leftrightarrow \begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$ *and* $\begin{pmatrix} m & m+n \\ 0 & 1 \end{pmatrix} \leftrightarrow \begin{pmatrix} m & n \\ 0 & 1 \end{pmatrix}$.

The corresponding unimodular matrices for the above equivalences are trivially $\mathbf{V}=\begin{pmatrix}1&0\\1&1\end{pmatrix}$ and $\mathbf{V}=\begin{pmatrix}1&1\\0&1\end{pmatrix}$.

This lemma further reduces our choice of n in Lemma 5.3. It is sufficient to require integer n being within $[0, m-1]$ in order to represent all downsampling patterns. Up to this stage, the number of patterns is reduced to $2m$.

Lemma 5.5 *For a given $n \in [1, m-1]$ there always exists n' such that* $\begin{pmatrix}1&0\\n&m\end{pmatrix}\leftrightarrow\begin{pmatrix}m&n'\\0&1\end{pmatrix}$.

Proof: In a Galois field $GF(m)$, every element, except zero, has an inverse modulo m. Hence, if $b \in GF(m)(b \neq 0)$, then its inverse is defined as b^{-1} and $bb^{-1} = 1$ [87]. All integers that are less than a prime number m form such a Galois field, and every integer $n < m$ has corresponding inverse n' satisfying $n \cdot n' = 1 \pmod{m}$.

For matrices in Lemma 5.5, they are related by

$$\begin{pmatrix}1&0\\n&m\end{pmatrix} = \begin{pmatrix}m&n'\\0&1\end{pmatrix}\begin{pmatrix}\frac{1-n'n}{m}&-n'\\n&m\end{pmatrix}.$$

Since n' is the inverse of n as defined above, $\frac{1-n'n}{m}$ is an integer. Hence the unimodular matrix in the right hand side is also an integer matrix. This justifies the equivalence in this lemma. ∎

Lemma 5.5 rules out the remaining redundancy in the downsampling matrices. According to it, matrices in the form of $\begin{pmatrix}1&0\\n&m\end{pmatrix}$ can be adopted into matrices in the form of $\begin{pmatrix}m&n\\0&1\end{pmatrix}$ or vice versa, except the case with $n = 0$. Therefore up to this step, the number of representative matrices is reduced to $m+1$. Next, we show that these $m+1$ matrices are indeed distinct.

Lemma 5.6 *The aforementioned $m+1$ downsampling patterns are distinct from each other.*

Proof: Firstly, by contradiction assume there exist $n_1, n_2 \in [0, m-1]$ such that $\begin{pmatrix}1&0\\n_1&m\end{pmatrix}\leftrightarrow\begin{pmatrix}1&0\\n_2&m\end{pmatrix}$. The corresponding unimodular matrix should be expressible as,

$$\mathbf{V} = \begin{pmatrix}1&0\\n_1&m\end{pmatrix}^{-1}\begin{pmatrix}1&0\\n_2&m\end{pmatrix} = \begin{pmatrix}1&0\\\frac{n_2-n_1}{m}&1\end{pmatrix}$$

It can not be an integer matrix unless $n_1 = n_2$.

Secondly, by contradiction assume that there exist $n \in [0, m-1]$ such that $\begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix} \leftrightarrow \begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}$. The corresponding unimodular matrix should be expressible as,

$$\mathbf{V} = \begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}^{-1} \begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} m & 0 \\ -n_1 & \frac{1}{m} \end{pmatrix}$$

It can not be an integer matrix at all.

Hence both the above assumptions are invalid. ∎

Naturally, the following theorem can be obtained based on the above four lemmas.

Theorem 5.7 *For a downsampling matrix with a prime downsampling ratio m, there only exist $m+1$ possible different downsampling patterns. The corresponding downsampling matrices can be parameterized as $\begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$ ($n \in [0, m-1]$) and $\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}$.*

Hence the downsampling patterns in Figure 5.1 are the only 3 possible lattices for downsampling ratio $m = 2$, and the quincunx matrix is the only non-separable one (whose representative matrix as in Theorem 5.7 is not diagonal).

Remark: As a matter of fact, Theorem 5.7 can be also interpreted in the following way. First, connect samples with parallel straight lines. Downsampling is performed by retaining only 1 line of samples out of consecutive m lines and discarding the others. For different patterns, these lines incline in different possible angles. Note that this explanation is only valid for prime downsampling ratio.

5.3.2 Downsampling patterns with composite downsampling ratio

When the downsampling ratio is composite, though the above conclusion is no longer valid, a similar approach can be followed. The following lemma is necessary to relate this problem to the problem having been solved.

Lemma 5.7 *Given an integer matrix $\mathbf{A}$, it can be factorized into integer matrices as $\mathbf{A} = \prod_{i=1}^{k} \mathbf{A}_i$, if and only if $d = \prod_{i=1}^{k} d_i$, where $d = det(\mathbf{A})$ and $d_i = det(\mathbf{A}_i)$.*

Proof: ($\Leftarrow$) If there exists a factorization $\mathbf{A} = \prod_{i=1}^{k} \mathbf{A}_i$, equation $d = \prod_{i=1}^{k} d_i$ can be easily obtained by taking the determinant of both sides.

($\Rightarrow$) As we already know, any $M \times M$ integer matrix $\mathbf{M}$ can be factorized as $\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}$ [3], [88], where $\mathbf{U}$ and $\mathbf{V}$ are integer unimodular matrices and $\mathbf{\Lambda}$ is a diagonal integer matrix. For our case, $\mathbf{\Lambda}$ is expressible as $\mathbf{\Lambda}=\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$. Since λ_1 and λ_2 are also integers, $\mathbf{\Lambda}$ can be further decomposed as

$$\mathbf{\Lambda}=\prod_{i \in S_1} \begin{pmatrix} d_i & 0 \\ 0 & 1 \end{pmatrix} \prod_{j \in S_2} \begin{pmatrix} 1 & 0 \\ 0 & d_j \end{pmatrix}$$

where $\lambda_{1(2)} = \prod_{i \in S_1(S_2)} d_i$ and $S_1 \cup S_2 = \{1, 2, \cdots k\}$. Matrices $\mathbf{A_i}$ are readily obtained. ∎

Remark: Note that λ_1 must be chosen in such a way that it is a factor of λ_2 [3], [88]. Therefore, if the determinant of downsampling matrix $\mathbf{M}$ is $d = a^2 b$, $\lambda_1 = a$ and $\lambda_2 = ab$. Obviously, this factorization sometimes is not unique. For example, when $d = 4$, both $a = 2, b = 1$ and $a = 1, b = 4$ are legal. But $a = 1, b = 6$ is uniquely legal when $d = 6$. Obviously, we can obtain factor matrices $\mathbf{A}_i$ with determinants as prime numbers by choosing all d_i to be prime.

Based on this, the total number of downsampling patterns is given as follows.

Theorem 5.8 *For a given composite downsampling ratio m, if we assume that $m = p_1 p_2 \cdots p_t$, where $p_i \neq 1$ $(i = 1, \cdots t)$ is prime, the total number of downsampling patterns is at most $\prod_{i=1}^{t}(p_i + 1)$. When multiplicity of some prime factor does not exist, the number of patterns is exact. They can be parameterized as $\mathbf{M}=\prod_{i=1}^{t} \mathbf{M}_i(p_i)$, where*

$$\mathbf{M}_i(p_i)=\begin{pmatrix} 1 & 0 \\ n_i & p_i \end{pmatrix} \text{ or } \begin{pmatrix} p_i & 0 \\ 0 & 1 \end{pmatrix} \quad \text{with} \quad n_i = 0, \cdots, p_i - 1$$

Proof: The total number is a consequence of the proposed parametrization. Therefore the parametrization is considered first. For a given downsampling matrix $\mathbf{M}$, as we demonstrated in Lemma 5.7 and its following remark, we can always factor

$\mathbf{M}$ as $\mathbf{M} = \mathbf{U}\mathbf{M}'_1 \cdots \mathbf{M}'_t\mathbf{V}$ with all matrices $\mathbf{M}'_i$ having prime determinants. Assume that $m = |\mathbf{M}| = p_1p_2\cdots p_t$. To make the factorization as unique as possible, we further require $p_1 \leq p_2 \cdots \leq p_t$. Of course, other method of sorting p_i is also applicable once it is consistent in the following. Since $\mathbf{M}'_i$ are diagonal and hence commutative, we can rearrange them as $\mathbf{M} = \mathbf{U}\mathbf{M}^\circ_1 \cdots \mathbf{M}^\circ_t\mathbf{V}$ such that $det(\mathbf{M}^\circ_i) = p_i$. We start from the leftmost item. Since $\mathbf{U}$ is an integer unimodular matrix and $\mathbf{M}^\circ_1$ is an integer matrix with prime determinant, according to Theorem 5.7, $\mathbf{U}\mathbf{M}^\circ_1$ can be always expressed as $\mathbf{M}_1\mathbf{U}_1$ where $\mathbf{M}_1$ is identically in the form of $\begin{pmatrix} 1 & 0 \\ n_1 & p_1 \end{pmatrix}$ or $\begin{pmatrix} p_1 & 0 \\ 0 & 1 \end{pmatrix}$ with $n_1 = 0, \cdots, p_1 - 1$ and $\mathbf{U}_1$ is also an integer unimodular matrix. Iteratively $\mathbf{U}_1\mathbf{M}^\circ_2 \leftrightarrow \mathbf{M}_2\mathbf{U}_2$ and so forth. Finally the downsampling matrix can be written as $\mathbf{M} = \mathbf{M}_1 \cdots \mathbf{M}_t\mathbf{U}_t\mathbf{V}$. It is indeed one of patterns in the proposed parametrization.

We now consider the case without multiplicity of prime factors. When $m = p_1p_2$ and prime numbers $p_1 \neq p_2$, corresponding patterns can be obtained as follows. List $p_1 + 1$ matrices on p_1 column-wise and list $p_2 + 1$ matrices on p_2 row-wise. The desired downsampling matrices can be obtained by post multiplying a matrix from the column to a matrix from the row and therefore a table with $(p_1+1)\times(p_2+1)$ cells can be obtained as such. Such a table is shown in Figure 5.4. The downsampling matrix under the proposed parametrization can be expressed as $\begin{pmatrix} p & 0 \\ n_p & m/p \end{pmatrix}$, where $p = 1, p_1, p_2, p_1p_2$ and $n_p \in [0, m/p - 1]$. Note that the general form is only applicable when p and m/p are coprime. All together, the above matrix correspond to $(p_1 + 1) \times (p_2 + 1)$ downsampling matrices with different choice of p. We next show, these $(p_1 + 1) \times (p_2 + 1)$ choices are indeed distinct and correspond to different patterns. Given p and p' of aforementioned form, if they are equivalent, the unimodular matrix should be expressible as

$$\mathbf{V} = \begin{pmatrix} p & 0 \\ n_p & m/p \end{pmatrix}^{-1} \begin{pmatrix} p' & 0 \\ n'_p & m/p' \end{pmatrix} = \begin{pmatrix} \frac{p'}{p} & 0 \\ \frac{-n_pp' + n'_pp'}{m} & \frac{p}{p'} \end{pmatrix}$$

$\mathbf{V}$ can be integer if and only if $p = p'$ and $n_p = n'_p$ for $n_p, n'_p \in [1, m/p]$, which indicates that the proposed parametrization is complete and irreducible.

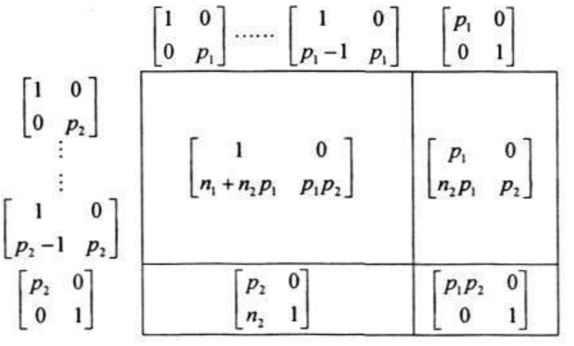


Figure 5.4: Downsampling matrix that is comprised of two prime matrices

In fact, in above proof p_1 and p_2 are not necessarily prime. Instead, coprimeness between p_1 and p_2 is sufficient to obtain a similar result. Therefore, as long as no multiplicity of factors exists, the parameterized number is exact as above. Due to limited space, details are not included.

However, the parametrization may repeat some patterns when multiplicity happens. That is why the term 'at most' is used in the theorem. Under such circumstance, the theorem provides an upper bound to the number of downsampling patterns. ∎

5.3.3 Examples of downsampling patterns

Apart from the $m = 2$ example shown in the previous section, more examples to illustrate Theorem 5.7 and Theorem 5.8 are given in this section.

For downsampling ratio $m = 3$, according to Theorem 5.7, there exists 4 different downsampling patterns. These are shown in Figure 5.5. The remark of Theorem 5.7 involving straight line interpretation is also reflected in this figure.

For downsampling ratio $m = 4$, according to Theorem 5.8, since $m = 2 \times 2$, there should exist at most 9 different downsampling patterns. Actually, given that both factors of m are 2, there are some identical patterns. After ruling out the repetition, there exists 7 different patterns and the corresponding downsampling matrices can

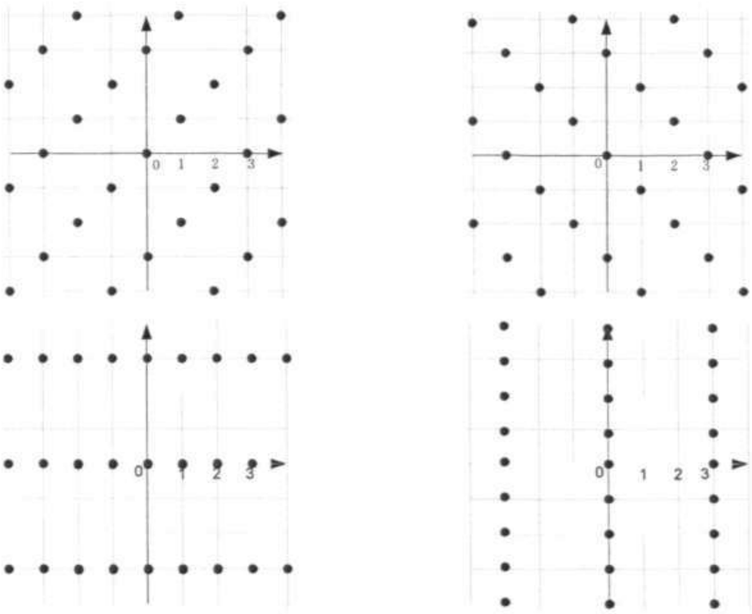


Figure 5.5: Four patterns for $|\mathbf{M}| = 3$

be parameterized by the product of 2 matrices with determinant of 2, as shown in Table 5.1. Possible choices for $\mathbf{M_1}$ and $\mathbf{M_2}$ as in Theorem 5.8 are listed in the first column and in the first row respectively. Different circled letters are also used for indicating distinct patterns. Note that 3 different matrices correspond to the same patterns ⓒ . These patterns are depicted in Figure 5.6.

	$\begin{pmatrix}1&0\\0&2\end{pmatrix}$	$\begin{pmatrix}1&0\\1&2\end{pmatrix}$	$\begin{pmatrix}2&0\\0&1\end{pmatrix}$
$\begin{pmatrix}1&0\\0&2\end{pmatrix}$	$\begin{pmatrix}1&0\\0&4\end{pmatrix}$ ⓐ	$\begin{pmatrix}1&0\\2&4\end{pmatrix}$ ⓑ	$\begin{pmatrix}2&0\\0&2\end{pmatrix}$ ⓒ
$\begin{pmatrix}1&0\\1&2\end{pmatrix}$	$\begin{pmatrix}1&0\\1&4\end{pmatrix}$ ⓓ	$\begin{pmatrix}1&0\\3&4\end{pmatrix}$ ⓔ	$\begin{pmatrix}2&0\\2&2\end{pmatrix}$ ⓒ
$\begin{pmatrix}2&0\\0&1\end{pmatrix}$	$\begin{pmatrix}2&0\\0&2\end{pmatrix}$ ⓒ	$\begin{pmatrix}2&0\\1&2\end{pmatrix}$ ⓕ	$\begin{pmatrix}4&0\\0&1\end{pmatrix}$ ⓖ

Table 5.1: Parametrization of downsampling patterns when $m = 4$

Similarly, the parametrization for downsampling ratio $m = 6$ is given in Table 5.2. Since the factors are different, there are exactly 12 different patterns. Some readers may be curious about the consequence of interchanging $\mathbf{M_1}$ and $\mathbf{M_2}$. The

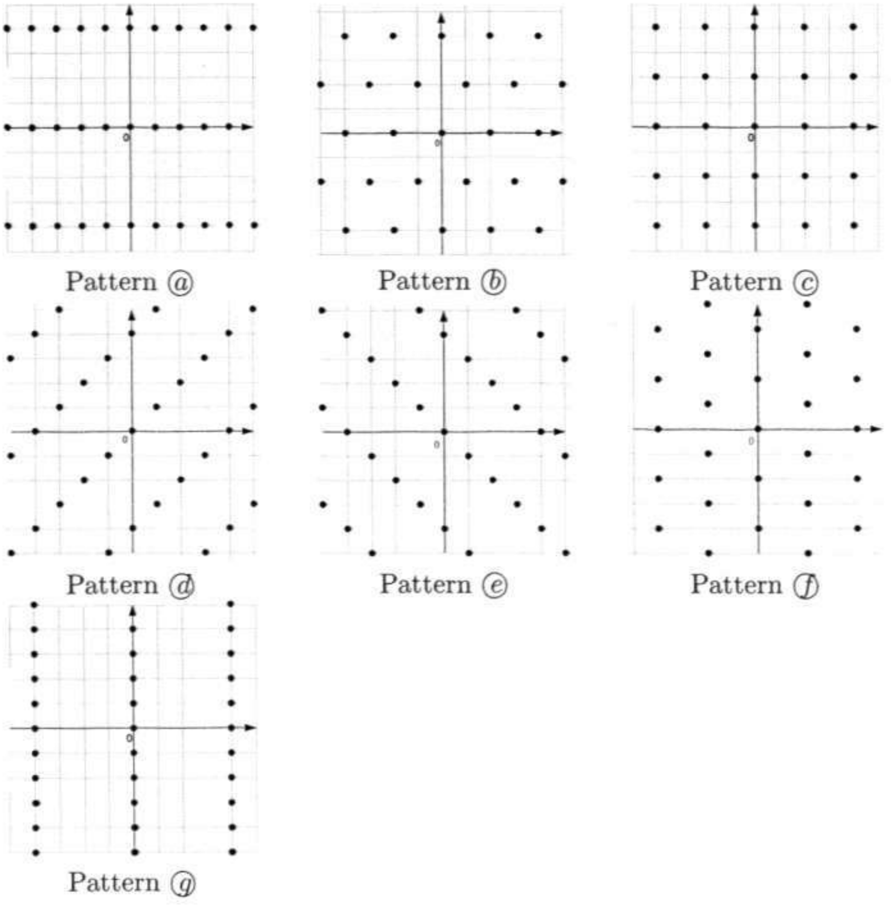


Figure 5.6: Distinct patterns for $|\mathbf{M}|= 4$

resulting patterns are exactly the same as those in Table 5.2, yet in a different arrangement. Recalling the proof of Theorem 5.8, this equivalence is, in fact, guaranteed by the commutativity of diagonal matrices.

	$\begin{pmatrix}1&0\\0&3\end{pmatrix}$	$\begin{pmatrix}1&0\\1&3\end{pmatrix}$	$\begin{pmatrix}1&0\\2&3\end{pmatrix}$	$\begin{pmatrix}3&0\\0&1\end{pmatrix}$
$\begin{pmatrix}1&0\\0&2\end{pmatrix}$	$\begin{pmatrix}1&0\\0&6\end{pmatrix}$ ⓐ	$\begin{pmatrix}1&0\\2&6\end{pmatrix}$ ⓑ	$\begin{pmatrix}1&0\\4&6\end{pmatrix}$ ⓒ	$\begin{pmatrix}3&0\\0&2\end{pmatrix}$ ⓓ
$\begin{pmatrix}1&0\\1&2\end{pmatrix}$	$\begin{pmatrix}1&0\\1&6\end{pmatrix}$ ⓔ	$\begin{pmatrix}1&0\\3&6\end{pmatrix}$ ⓕ	$\begin{pmatrix}1&0\\5&6\end{pmatrix}$ ⓖ	$\begin{pmatrix}3&0\\3&2\end{pmatrix}$ ⓗ
$\begin{pmatrix}2&0\\0&1\end{pmatrix}$	$\begin{pmatrix}2&0\\0&3\end{pmatrix}$ ⓘ	$\begin{pmatrix}2&0\\1&3\end{pmatrix}$ ⓙ	$\begin{pmatrix}2&0\\2&3\end{pmatrix}$ ⓚ	$\begin{pmatrix}6&0\\0&1\end{pmatrix}$ ⓛ

Table 5.2: Parametrization of downsampling patterns when $m = 6$

For greater downsampling ratio, the previous results should be utilized to avoid the repeating of patterns. For example, when $m = 12$, the solutions for $m = 3$ and $m = 4$ should be used. We list the numbers of different patterns for some downsampling ratio in Table 5.3. In the result, the repeating cases have been ruled out by exhaustive checking.

m	2	3	4	5	6	7	8	9	10	11	12
♯ of Pat.	3	4	7	6	12	8	15	13	18	12	28
m	14	15	16	18	20	30	32	42	50	64	67
♯ of Pat.	24	24	31	39	42	72	63	96	93	127	68

Table 5.3: Numbers of different patterns for some downsampling ratio

5.3.4 Enumeration of nonseparable patterns

In many applications, people are willing to trade a better performance with an increased complexity especially when the cost of implementation is becoming more and more inexpensive. Nonseparable filter bank is such a case in point. Though it requires more resource in terms of both memory and computation, the better performance over its separable counterpart still makes it attractive to many researchers. For this kind of filter bank, the nonseparable downsampling patterns are essential. We further propose a specific approach to parameterizing all nonseparable patterns.

Lemma 5.8 *Patterns with downsampling matrices* $\mathbf{M}=\begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$ *or* $\mathbf{M}=\begin{pmatrix} m & n \\ 0 & 1 \end{pmatrix}$ *for any integer* m *and* $n = 1, \cdots, m-1$ *are nonseparable.*

Proof: According to this lemma, we can not find an integer unimodular matrix $\mathbf{U}$ such that $\begin{pmatrix} m & n \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \cdot \mathbf{U}$ with integers a and b satisfying m=ab. Obviously, if such matrix exists, it can be expressed as

$$\mathbf{U}=\begin{pmatrix} \frac{1}{a} & 0 \\ 0 & \frac{1}{b} \end{pmatrix}\begin{pmatrix} m & n \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} \frac{m}{a} & \frac{n}{a} \\ 0 & \frac{1}{b} \end{pmatrix}.$$

To make the last entry of $\mathbf{U}$ integer, b has to be 1. Then from $m = ab$, $a = m$. This choice disables the entry $\frac{n}{a}$ from integer since $n \in [1, m-1]$. Similar proof goes for $\begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$. ∎

Remark: From Theorem 5.7 and Lemma 5.8, the number of distinct nonseparable patterns for prime m is $m-1$ and these patterns can be parameterized as $\begin{pmatrix} 1 & 0 \\ n & m \end{pmatrix}$, $n \in [1, m-1]$.

Theorem 5.9 *For a downsampling ratio $m = p_1 p_2$, where prime numbers $p_1 \neq p_2 \neq 1$, its corresponding downsampling patterns that are nonseparable can be parameterized as*

$$\mathbf{M}=\begin{pmatrix} 1 & 0 \\ n & p_1p_2 \end{pmatrix} \quad or \quad \begin{pmatrix} p_1p_2 & n \\ 0 & 1 \end{pmatrix} \quad with\ n = 1, \cdots p_1p_2 - 1 \ .$$

Proof: From Lemma 5.8, both $\mathbf{M}=\begin{pmatrix} 1 & 0 \\ n & p_1p_2 \end{pmatrix}$ and $\begin{pmatrix} p_1p_2 & n \\ 0 & 1 \end{pmatrix}$ are nonseparable. As shown in Figure 5.4, the patterns within the first p_1 rows and the first p_2 columns are of the form

$$\begin{pmatrix} 1 & 0 \\ n_1 & p_1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ n_2 & p_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ n_1 + n_2p_1 & p_1p_2 \end{pmatrix}.$$

Since p_1 and p_2 are coprime, for $n_1 \in [0, p_1 - 1]$ and $n_2 \in [0, p_2 - 1]$, $n_1 + n_2p_1 \in [0, p_1p_2 - 1]$. Thus, all these p_1p_2 patterns may be parameterized in the form of $\begin{pmatrix} 1 & 0 \\ n & p_1p_2 \end{pmatrix}$. For $n \in [1, p_1p_2 - 1]$, they are nonseparable.

This leaves $p_2 - 1$ matrices in the last row of the table (except first and last entries, which are separable) and $p_1 - 1$ matrices in the last column (except first and last entries). Downsampling matrices in the last column can be expressed as $\begin{pmatrix} 1 & 0 \\ n' & p_1 \end{pmatrix} \begin{pmatrix} p_2 & 0 \\ 0 & 1 \end{pmatrix}$ where $n' \in [1, p_1 - 1]$. We now show each such matrix is equivalent to a matrix of the form given in the theorem, or that there exists an integer unimodular matrix $\mathbf{U}$ such that

$$\begin{pmatrix} 1 & 0 \\ n' & p_1 \end{pmatrix} \begin{pmatrix} p_2 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} p_1p_2 & n^\circ \\ 0 & 1 \end{pmatrix} \mathbf{U}.$$

It follows that

$$\mathbf{U}=\begin{pmatrix} \frac{1}{p_1p_2} & \frac{-n^\circ}{p_1p_2} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ n' & p_1 \end{pmatrix} \begin{pmatrix} p_2 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1-n'n^\circ}{p_1} & \frac{-n^\circ}{p_2} \\ p_2n' & p_1 \end{pmatrix}.$$

The task is reduced to showing that there exists such n°. Following Bezout Identity, we can always find integers k, j for the given coprimes p_1, p_2 such that $p_2k - p_1j = 1$. Following the proof of Lemma 5.5, we can always find an integer n^* within $[1, p_1 - 1]$ such that $\frac{1-n'n^*}{p_1}$ is an integer. Now choose $n^\circ = n^* + p_1(n^*j)$. It follows that $\frac{1-n'n^\circ}{p_1}$ is indeed an integer. Further, since $n^*p_2k - n^*p_1j = n^*$, this choice of n° makes sure that $\frac{n^\circ}{p_2} = n^*k$ is also an integer.

Similar approach may be used for the matrices in the last row. ∎

Remark: For composite downsampling ratio, the above parametrization is complete yet redundant, which means repeating may happen. For example, for any integer m, (not necessarily prime)

$$\begin{pmatrix} m & m-1 \\ 0 & 1 \end{pmatrix} \leftrightarrow \begin{pmatrix} 1 & 0 \\ m-1 & m \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} m & 1 \\ 0 & 1 \end{pmatrix} \leftrightarrow \begin{pmatrix} 1 & 0 \\ 1 & m \end{pmatrix} .$$

As emphasized in the theorem, m is necessary to be the product of only two distinct prime numbers. Whether this can be extended to the more generic case is for further research.

5.4 Alias-free downsampling filters

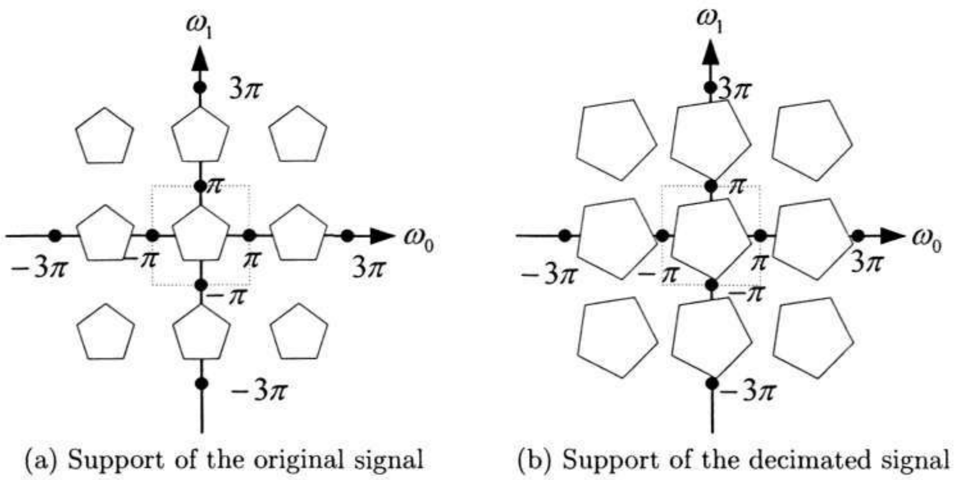
Just like the case in 1D, the downsampler in 2D will also lead to stretching of the signal in the frequency domain. To avoid overlapping between the neighboring frequency components, which is called aliasing effect, we should apply the downsampling filter before the signal is downsampled. Correspondingly, in the synthesis side, we have many images of the original frequency components due to the upsampling and we always use upsampling filter to pick up the desired one. The filters in both sides are closely related by the perfect reconstruction constraint. Especially for the paraunitary filter banks, the criterions and the process to choose them are similar. Hence we just focus on one of them, the downsampling filter. In this section, we study how to choose an alias-free filter. Before that, we need to understand the aliasing effects well.

5.4.1 Aliasing effects created by downsampling

If we let $x[\mathbf{n}]$ be a M-dimensional signal with Fourier transform $X(\boldsymbol{\omega})$ and the sequence $y[\mathbf{n}] = x[\mathbf{M} \cdot \mathbf{n}]$, we have the following relation in the frequency domain

$$Y(\boldsymbol{\omega}) = \frac{1}{det(\mathbf{M})} \sum_{\mathbf{k} \in \mathcal{N}(\mathbf{M})^T} X\left(\mathbf{M}^{-T}(\boldsymbol{\omega} - 2\pi\mathbf{k})\right) \tag{5.3}$$

where $\mathcal{N}(\mathbf{M}^T)$ is the set of integers of the form $\mathbf{M}^T\mathbf{x}$ with $\mathbf{x} \in [0,1)^M$ and there are $det(\mathbf{M})$ elements in this set. Equation (5.3) suggests that $Y(\boldsymbol{\omega})$ is a sum of the stretched versions of $X(\boldsymbol{\omega})$. We demonstrate this graphically in Figure 5.7. Note that the pentagons in Fig 5.7(b) are actually the sum of alias components and the periodic version of that in the origin. Here the downsampling matrix we choose is the popular quincunx matrix, i.e. $\mathbf{M} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$.



(a) Support of the original signal (b) Support of the decimated signal

Figure 5.7: Illustrating the frequency domain effects of downsampling

Intuitively, $Y(\boldsymbol{\omega})$ should be a periodic function, just like its 1D counterpart. Fortunately, it indeed is. This is in fact owing to the periodicity of downsampling lattice. Let us first recall its 1D counterpart, i.e. Eq. (4.1.4) in [3],

$$Y(\omega) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(\frac{\omega - 2\pi k}{M}\right) \tag{5.4}$$

Since $X(\omega)$ itself is periodic with period 2π, the following derivation tells us, $Y(\omega)$ is also a periodic function with period 2π.

$$\begin{aligned} Y(\omega-2\pi) &= \tfrac{1}{M}\textstyle\sum_{k=0}^{M-1} X(\frac{\omega-2\pi-2\pi k}{M}) \\ &= \tfrac{1}{M}\textstyle\sum_{k=1}^{M-1} X(\frac{\omega-2\pi k}{M}) + X(\frac{\omega-2\pi-2\pi(M-1)}{M}) \\ &= \tfrac{1}{M}\textstyle\sum_{k=1}^{M-1} X(\frac{\omega-2\pi k}{M}) + X(\frac{\omega}{M}-2\pi) \\ &= \tfrac{1}{M}\textstyle\sum_{k=0}^{M-1} X(\frac{\omega-2\pi k}{M}) = Y(\omega) \end{aligned} \tag{5.5}$$

Note that the set of integers $\{0, 1, \cdots, M-1\}$, corresponding to $X(\frac{\omega}{M})$, $X(\frac{\omega-2\pi}{M})$, $\cdots$, $X(\frac{\omega-2\pi(M-1)}{M})$, have the property that the set remains unchanged after an integer shift and modulo M operation. It is this essential fact that makes $Y(\omega)$ periodic. Fortunately, similar fact also exists in the nD case. As proved in [3], we can express an arbitrary integer vector $\mathbf{n}$ as,

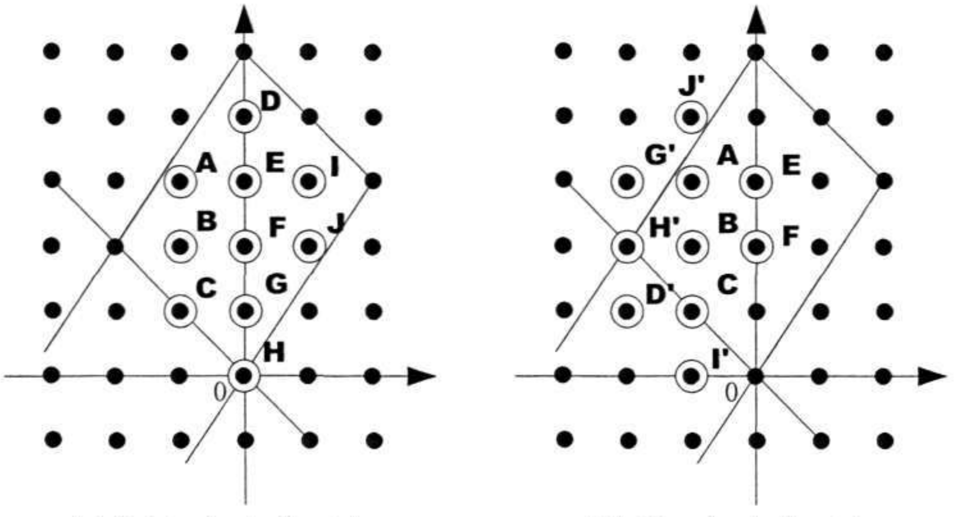
$$\mathbf{n} = \mathbf{k} + \mathbf{M}\cdot\mathbf{n}_0 \tag{5.6}$$

where $\mathbf{k} \in \mathcal{N}(M)$ and $\mathbf{n}_0$ is an integer vector. Under this definition, we can regard that $\mathbf{k}$ is equal to $\mathbf{n}$ modulo $\mathbf{M}$. Consequently, the following equivalence between two sets can be obtained,

$$\mathcal{N}(\mathbf{M}) \Longleftrightarrow \mathcal{N}(\mathbf{M}) + \mathbf{k} \mod \mathbf{M} \tag{5.7}$$

for any integer vector $\mathbf{k}$ denoting a shift. In Figure 5.8, we show an example of this equivalence with a shift $\mathbf{k} = [-1, 0]^T$ and a downsampling matrix $\mathbf{M} = \begin{bmatrix} 2 & -2 \\ 3 & 2 \end{bmatrix}$. Figure 5.8(a) shows the set $\mathcal{N}(\mathbf{M})$ consisting of integer vectors $\mathbf{A}, \mathbf{B}, \cdots, \mathbf{J}$. Figure 5.8(b) shows the set after a shift by $\mathbf{k}$. Points $\mathbf{D}', \mathbf{G}', \mathbf{H}', \mathbf{I}', \mathbf{J}'$ move to $\mathbf{D}, \mathbf{G}, \mathbf{H}, \mathbf{I}, \mathbf{J}$ respectively after modulo $\mathbf{M}$ operation, resulting in an equivalent set. This actually indicates the period of 2π in each frequency dimension.

In the example of Figure 5.7, the aliasing effect fortunately does not happen. However, it is not typically true. To avoid aliasing we have to use a downsampling filter to constrain the spectrum of the signal within the range of the symmetric



(a) Original set of points (b) Moved set of points

Figure 5.8: Equivalence of sets of points modulo a downsampling matrix

parallelepiped, or $SPD(\pi\mathbf{M}^{-1})$. This can be proved as follows. Actually a similar proof can also be found in [3].

The region of support given by $SPD(\pi\mathbf{M}^{-1})$ should be interpreted as the region

$$\omega = \pi\mathbf{M}^{-1}\mathbf{x}, \quad \mathbf{x} \in (-1,1)^M \tag{5.8}$$

Its shifted version should be

$$\omega = \pi\mathbf{M}^{-1}(\mathbf{x} - 2\mathbf{k}), \quad \mathbf{k} \in \mathcal{N}(\mathbf{M}^T) \tag{5.9}$$

If two regions shifted differently have overlapping, we can express them as

$$\omega_1 = \omega_2 \iff \pi\mathbf{M}^{-1}(\mathbf{x_1} - 2\mathbf{k_1}) = \pi\mathbf{M}^{-1}(\mathbf{x_2} - 2\mathbf{k_2}) \tag{5.10}$$

Since $\mathbf{M}$ is nonsingular, we equivalently require

$$\mathbf{x}_1 - \mathbf{x}_2 = -2(\mathbf{k}_1 - \mathbf{k}_2) \tag{5.11}$$

Since $\mathbf{x}_1, \mathbf{x}_2 \in (-1,1)^M$, we can derive that $\mathbf{x}_1 - \mathbf{x}_2 \in (-2,2)^M$. We also know $\mathbf{k}_1 - \mathbf{k}_2$ have to be integer, this implies $\mathbf{k}_1 - \mathbf{k}_2 = 0$, namely $\mathbf{k}_1 = \mathbf{k}_2$, and naturally leads to $\mathbf{x}_1 = \mathbf{x}_2$. This overlapping only happens when the regions are not distinct.

Therefore, the support region of $SPD(\pi\mathbf{M}^{-1})$ doesn't overlap that of its legal shifted versions.

In fact the above conclusion can be relaxed slightly by a fixed shift and periodicity, namely if the support of $X(\boldsymbol{\omega})$ is restricted to

$$\boldsymbol{\omega} = \mathbf{c} + \pi\mathbf{M}^{-T}\mathbf{x} + 2\pi\mathbf{m} \quad \mathbf{x} \in (-1,1)^M, \mathbf{m} \in \mathcal{N} \tag{5.12}$$

it can be decimated by $\mathbf{M}$ without causing aliasing. The above conclusion gives us the guideline for the downsampling filter. For a multidimensional filter bank with a downsampling matrix $\mathbf{M}$, we can choose the filter with passband $SPD(\pi\mathbf{M}^{-T})$ as the low pass filter and choose its shifted version as regions of support for the filters of the other channels.

5.4.2 The approach to identify ideal alias-free filters

Similar to the downsampling matrix, the support of downsampling filters are also non-unique. As we mentioned in section 5.3, for different downsampling matrices of the same pattern, the same ideal filters can be applied while necessarily rearranging the input samples. Sometimes, the rearrangement may not be desired. Anyway, to have more choices in different shapes of ideal filters is always useful when certain special application is considered. Therefore, we investigate the ideal filters that are alias-free under different downsampling matrices. Interesting shapes are obtained by our proposed approach.

Instead of original frequency domain, we can consider this issue in the downsampled frequency domain. This will be described in detail for the 2-D case. In the downsampled domain, the filter may have various supports. As shown above, the filters in downsampled domain, $Y(\boldsymbol{\omega})$ is periodic with period 2π in both frequency directions. Alias-free indicates no overlapping between the neighboring spectral blocks. We can avoid the overlapping in the following way. Given a region of

$[-\pi, \pi]^2$ in downsampled domain, we can arbitrarily divide it into four subregions as depicted in Figure 5.9(a). These four blocks can be put into corresponding quadrants as in Figure 5.9(b). The shaded region in Figure 5.9(b) denotes the subband spectrum. This figure also shows that if we take into account the periodicity, the region of support will thoroughly span the whole plane of ω_0 and ω_1. This is the extreme case for non-overlapping support.

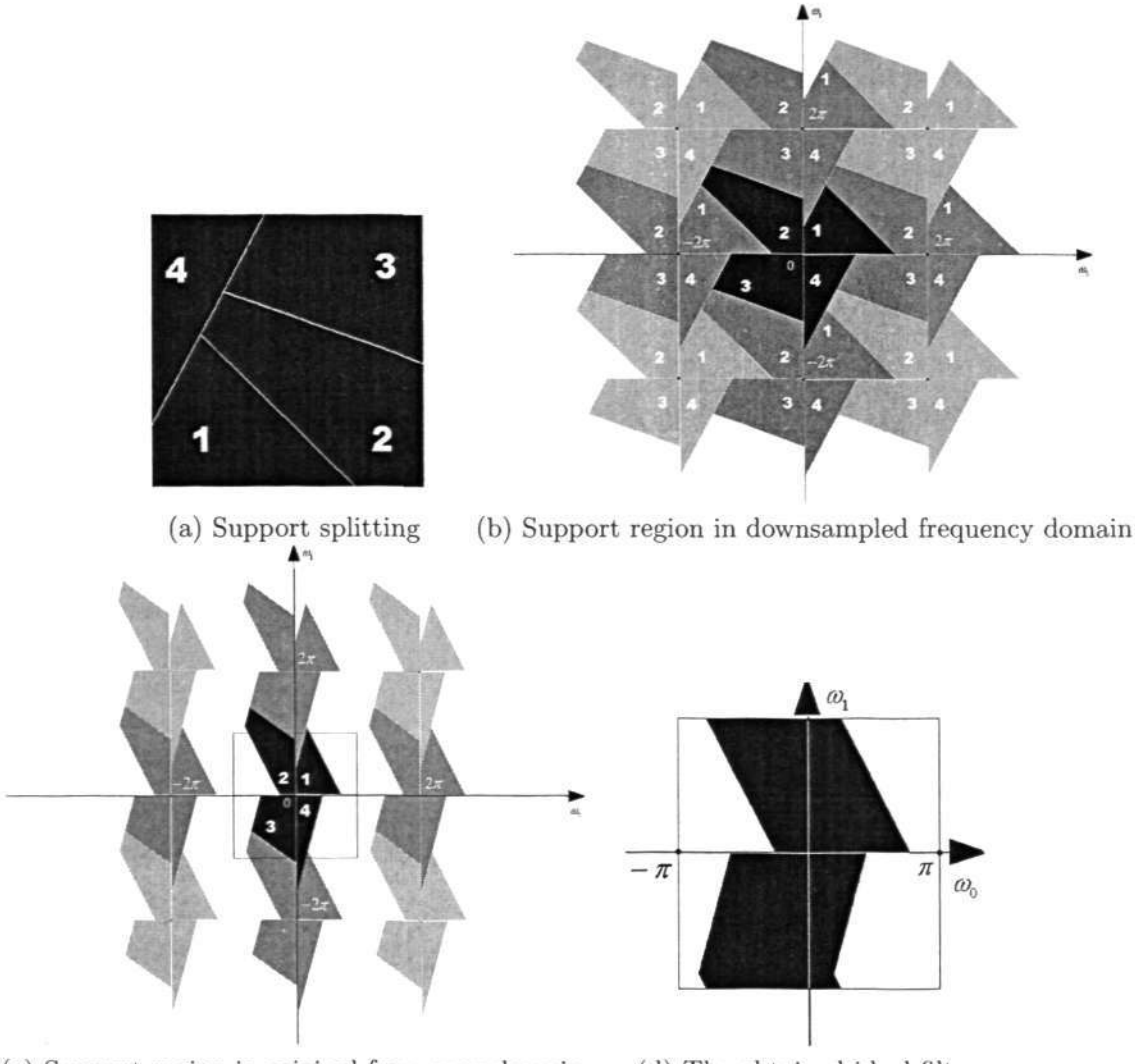
Now we find the ideal filter that produces the subband spectrum such as Figure 5.9(b). The filter shape depends on the downsampling matrix. Let us take $\mathbf{M} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$. The original $[-\pi\pi]^2$ spectrum will be stretched by $\mathbf{M}$ (in this example, stretched by 2 in ω_0 and no change in ω_1) to obtain the subband spectrum. Conversely, the subband spectrum of Figure 5.9(b) may be compressed to obtain the spectrum in original frequency domain as shown in Figure 5.9(c). Finally, the ideal filter shape is shown in Figure 5.9(d). Therefore, using such filters will not lead to aliasing effect even if the signal is downsampled.

Since the shape of filters are related to the downsampling matrix, we can obtain different filter shapes from the same support-splitting. We demonstrate this in Figure 5.10 and Figure 5.11. In Figure 5.10(a), we demonstrate a way to split a square filter support. We shift each component to the corresponding quadrant, as done above. The filter support in downsampled domain is depicted in Figure 5.10(b). Here we neglect the neighboring periodic filters. For the rest four subfigures we present possible filters of the corresponding downsampling matrix. They are located in a range of $[-\pi, \pi]^2$ and centered at $(0, 0)$. Similar arrangement also goes for Figure 5.11. In these two figures some filters with familiar support are found, such as the diamond filter (Figure 5.10(e)).

The proposed approach is different from the existing approaches which has to check whether a given filter is alias-free. The proposed filters are generated from the downsampled domain and are designed to be alias-free. All such filters are possible choices for a nD PRFB system.

5.5 Summary

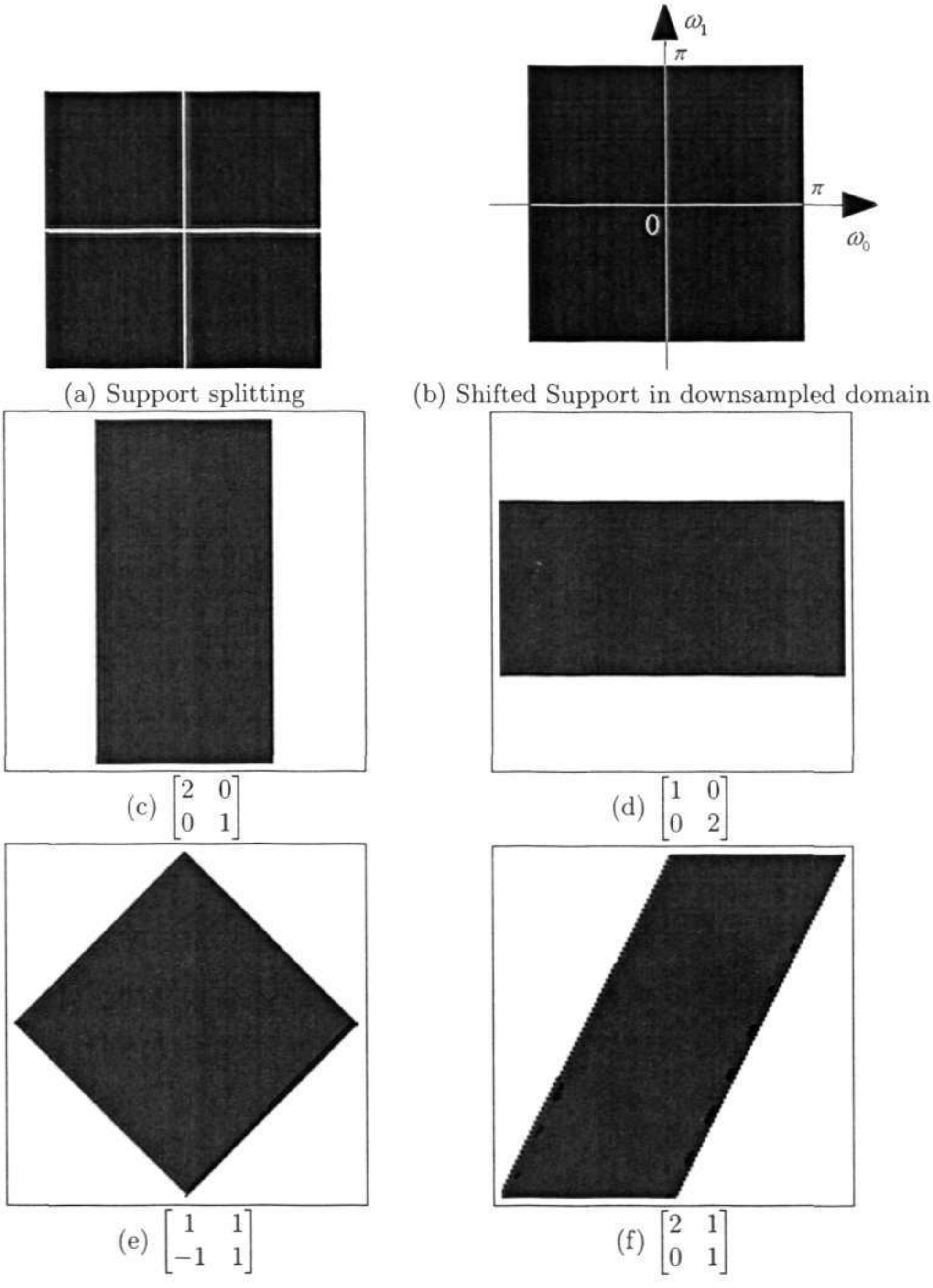
In this chapter, we first review the fundamentals of nD signals and systems. The differences between the 1D system and nD system are emphasized. It is followed by a detailed study on downsampling patterns for arbitrary downsampling ratios. We show that for a downsampling ratio, the number of downsampling patterns is finite. Specifically, when the downsampling ratio m is a prime number, there exists $m + 1$ different patterns and $m - 1$ of them are nonseparable. Both these cases are completely parameterized. When the downsampling ratio m is composite and can be expressed as the product of prime numbers $m = p_1 p_2 \cdots p_t$, where prime numbers $p_i \neq 1$ $(i = 1, \cdots t)$, there exists at most $\prod_{i=1}^{t}(p_i + 1)$ patterns. When multiplicity doesn't exist, the number is exact. These patterns are also completely parameterized. Further, for $t = 2$ with $p_1 \neq p_2$, a parametrization for all nonseparable patterns has been proposed. We also propose an approach to obtain ideal filters for alias-free nD PRFBs.



(a) Support splitting (b) Support region in downsampled frequency domain

(c) Support region in original frequency domain (d) The obtained ideal filter

Figure 5.9: Illustrating non-overlapping support



(a) Support splitting

(b) Shifted Support in downsampled domain

(c) $\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$

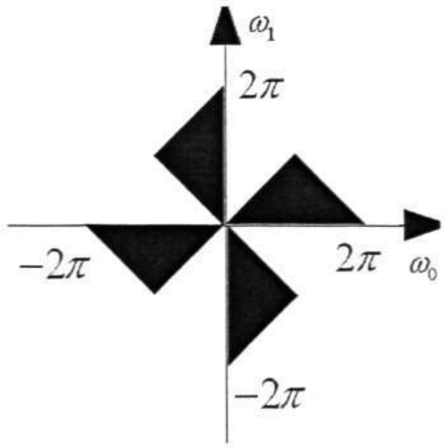
(d) $\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$

(e) $\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$

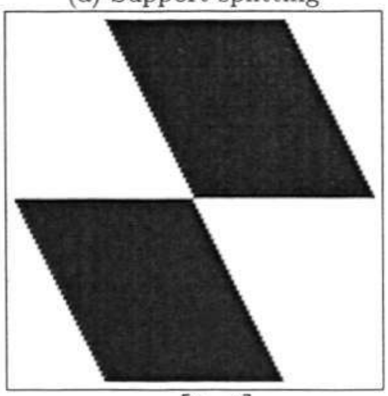
(f) $\begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix}$

Figure 5.10: A possible support splitting and the corresponding filters

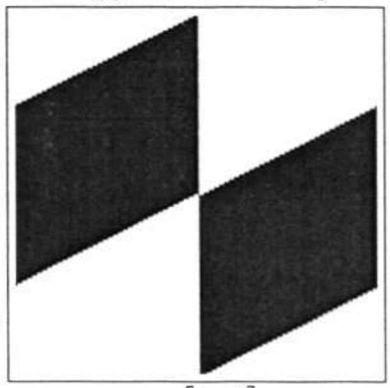
(a) Support splitting



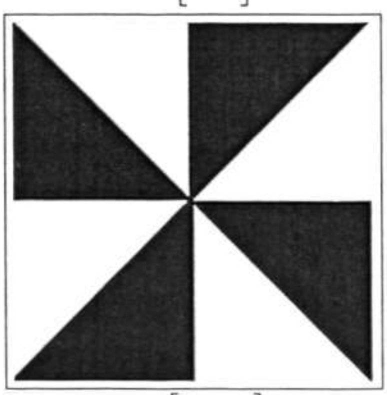
(b) Shifted Support in downsampled domain



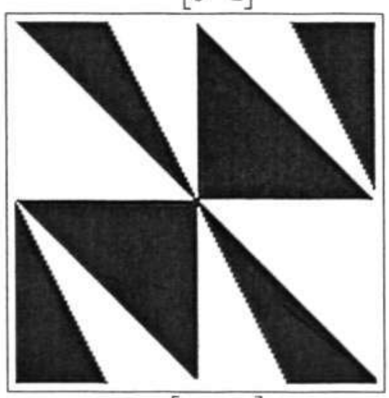
(c) $\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$



(d) $\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$



(e) $\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$



(f) $\begin{bmatrix} 2 & -1 \\ 0 & 1 \end{bmatrix}$

Figure 5.11: Another possible support splitting and the corresponding filters

Chapter 6

Polynomial invertibility and nD PRFB

FIR filters are actually polynomials in z-domain. Hence many filter problems can be re-formulated into problems in polynomial. So does the problems in filter banks. For example, as we show in Chapter 2, to find the lifting factorization for a given filter bank, a popular approach in elementary algebra, the Euclidean algorithm can be applied. In this chapter, we deal with the problems in a similar manner. However, when the number of variables in the system increase to more than one, elementary algebra is not appropriate any more. When there is more than one variable, geometric considerations enter and are important in understanding the phenomenon. Therefore, to solve the problems in nD PRFB, we resort to algebraic geometry, which is a branch of mathematics.

As the name suggests, it combines abstract algebra, especially commutative algebra, with geometry. In classical algebraic geometry, the main objects of interest are the vanishing sets of collections of polynomials, meaning the set of all points that simultaneously satisfy one or more polynomial equations. This can be regarded as the mathematical prototype for many practical engineering problems. Algebraic geometry is hence important to modern engineering disciplines. It has developed tremendously since the 19th century, especially after the theory and computational method of Groebner basis (GB) was introduced and developed by Buchberger in 1965 [89]. Actually an analogous concept was developed independently by Hironaka [90] in 1964, who named them standard bases. Several years after the development for construction of GB, it has been studied extensively in the context of many fields including control theory, coding theory and filter design. Self-contained tutorials can be found in [91], [92] and [93].

In this chapter, we investigate the problems of polynomial invertibility in algebraic geometry and extend the results to the applications of nD PRFB. For some cases, algebraic geometry is applied for the first time. Though whether the results can be extended generally is still under investigation, the application of algebraic geometry in these aspects does gain us some insight.

To elaborate, we first establish the relation between polynomial invertibility and various zeros. This is further extended to various nD PRFBs. Consequently we derive some results that can be used as criteria for PR checking. Especially for linear phase case, an essential result is derived which is used later for disproving the approach to complete a nD LPPRFB for a given FB provided in a well-cited paper. This is done in the second section where we review an important concept, resultants, in algebraic geometry. The theory of resultants is also applied for LPPR condition checking. Groebner basis (GB), another tool apart from resultants, is also popularly used to check the corresponding zeros. A brief introduction on GB's concept and application in nD PRFB is provided in section 6.3. We also demonstrate the application of results obtained in section 6.1 on McClellan transform for nD PRFB with a positive example and a negative example. Finally, a summary follows.

6.1 Zeros and nD PRFB

To design an nD FB directly is always a difficult problem since the corresponding factorization theorems are much less ready than those in 1D. Hence, people sometimes use a similar approach as we do in subsection 3.5.4, namely to design a filter first and then complete the nD PRFB by some constructional method. For these FBs, at least one of the filters is entirely under our control, and others can be further refined using schemes like lifting. This kind of approach is called PR completion. Being a filter in a PRFB, it has to bear some extra constraints. For example, for a 1D LPPRFB as we explained in subsection 3.5.4, a filter cannot have zeros at both $z = k$ and $z = -k$ with $k \neq 0$, otherwise the PR condition is destroyed. Similar facts remain true for nD case but with a more complicated setup. Before considering the relation between PR and invertibility, we start the investigation with a glance at classification of nD zeros.

6.1.1 Classification of nD zeros

Common zeros for 2D (or nD) are different in nature from those of 1D case. For univariate polynomial, a common zero indicate equivalently a common factor between the polynomials. However, this is not the case in nD. Common factor should still be avoided though, for consideration of PR.

We define three classes of common zeros in the context of PRFB. Only 2D case is demonstrated for the sake of brevity, nD case can be achieved similarly.

Definition 6.2 *Consider polynomials $A(x, y)$ and $B(x, y)$, which are devoid of common factors (up to a trivial monomial). They have a common zero at (x_0, y_0):*

- *if $x_0 \neq 0$ and $y_0 \neq 0$, the zero is called a nontrivial zero.*
- *if $x_0 = 0$ and $y_0 =$* ***any value*** *or vice versa, the zero is called a strong zero.*
- *if $x_0 = 0$ and $y_0 = k \neq 0$ or vice versa, the zero is called a weak zero.*

The latter two classes of zeros seem alike. It may be illuminative to give a simple example.

Example 6.2 *For polynomials* $A(x,y) = x(1+xy+x)$ *and* $B(x,y) = x(1+x+y)$, $(1,-2)$ *is a nontrivial zero,* $(-1,0)$ *is a weak zero, and* $(0,\times)$ *is a strong zero where* $\times$ *denotes an arbitrary value.*

Obviously, strong zero exists when $A(x,y)$ and $B(x,y)$ have a common factor in the form of $x^{c_x}y^{c_y}$, where integers c_x and c_y are not simultaneously zero. Different from 1D case, two relatively prime multivariate polynomials do not have any common factors but nontrivial common zeros may be present. In the above example, the polynomials are not relatively prime. Excluding the common factor x, they do not share common factors while they still share zeros at $(1,-2)$ and $(-1,0)$. Mathematically speaking, in nD, absence of common factor other than monomials does not necessarily suggest absence of common zeros defined above. These two situations are referred as weak coprimeness and strong coprimeness respectively [24]. To some extent, this is the root of many difficulties that appear in nD problems.

In fact, classification of zeros have appeared in literature. Weak zeros are also used in [23] but they include both weak zeros and strong zeros of our definition, which is complementary to non-trivial zero as defined in [80].

6.1.2 Polynomial invertibility and nD zeros

In algebraic geometry, it has been shown that polynomial invertibility is actually tied to various common zeros of polynomials, as shown in the following two theorems. Note that here we use polynomials rather than Laurent polynomials, since we care more about zeros rather than poles.

Definition 6.3 *Polynomials* $A(x,y)$ *and* $B(x,y)$ *are polynomial invertible (PI) if there exist some polynomials* $M(x,y)$ *and* $N(x,y)$ *such that*

$$A(x,y)\cdot M(x,y) + B(x,y)\cdot N(x,y) = 1 \tag{6.1}$$

For convenience, $M(x,y)$ and $N(x,y)$ are named as PI invert polynomials in the following.

Theorem 6.10 *[23] Polynomials $A(x,y)$ and $B(x,y)$ are PI if and only if they do* not *have any common zero.*

Remark: This theorem can be proved in light of Quillen-Suslin theorem [94]. Note that the polynomial invertibility can be trivially extended to the case that $A(x,y)$ and $B(x,y)$ *only* has some strong common zeros, which means they share some common factor in the form of $x^{c_x}y^{c_y}$ for some integers c_x and c_y and the right-hand side (RHS) of Eq. (6.1) has to be changed to $x^{c_x}y^{c_y}$. Essentially, this is the same as Eq. (6.1) after extracting such common factor. The polynomials $M(x,y)$ and $N(x,y)$ are still obtainable. For the latter case, we name it as generalized PI. For example, $A = (y+1)x+1$ and $B = y+1$ are PI since

$$((y+1)x+1)\cdot 1-(y+1)\cdot x = 1.$$

$A = (y+1)x^2+x$ and $B = xy+x$ are generalized PI since

$$((y+1)x^2+x)\cdot 1-(xy+x)\cdot x = x.$$

Obviously, PR condition for 2-channel FB has a similar appearance as Eq. (6.1). If the filter $F(x,y)$ with polyphase components $A(x,y)$ and $B(x,y)$ satisfies Eq. (6.1), there exist filters that can constitute a PRFB with it, such as the filter having $M(x,y)$ and $N(x,y)$ as polyphase components. In this way, we relate a problem of nD PRFB completion to a problem of polynomial invertibility in algebraic geometry.

However, the strict requirement of LHS of Eq. (6.1) being 1 is far from necessary (not considering the generalized PI), since in practice filters can be made causal by inserting delays. Recall that in 1D, in Eq. (3.12) c is also not necessary to be zero. Consequently, FIR invertibility is considered.

Definition 6.4 *Polynomials $A(x,y)$ and $B(x,y)$ are FIR invertible (FI) if there exist some polynomials $M(x,y)$ and $N(x,y)$ such that*

$$A(x,y) \cdot M(x,y) + B(x,y) \cdot N(x,y) = x^n y^m, \tag{6.2}$$

where integers n and m are non-negative. Similarly, $M(x,y)$ and $N(x,y)$ are named as FI invert polynomials.

Note that FI is more general than generalized PI since $x^n y^m$ is not necessarily a common factor of A and B.

Theorem 6.11 *[23] Polynomials $A(x,y)$ and $B(x,y)$ are FIR invertible if and only if they do not have nontrivial zeros.*

Remark: This theorem can be proved in the light of Hilbert's Nullstellensatz theorem [91]. A proof of this kind can be found in [23]. Obviously, polynomial invertibility is a special case of FIR invertibility. If the polyphase components of $F(z_1, z_2)$ are FI, there always exist some filters to make them PR. It is very likely that the polyphase components are not PI but FI.

Following above theorems, we translate the problem of checking the possibility of PR completion for a given filter into the problem of checking the nature of their common zeros. As we mentioned, common zeros are indeed the major objects of interest in classical algebraic geometry. There exists many ways to obtain common zeros. Among them, two methods are generally employed. One method is GB. Since GB does not change the variety of the given polynomials, the obtained GB retains the same common zeros as the given polynomials. Generally, common zeros can easily be solved from their reduced GB. The other method uses resultant. As shown later in section 6.2, two facts derived from resultants can be used for checking common zeros.

6.1.3 Linear phase property and nD zeros

Linear phase is of special concern in this thesis as we explain in the beginning. Hence we further investigate the requirement on zeros if LPPRFB is desired.

Before we get down to the relation between LPPRFB and zeros, we describe a result of the next chapter. Similar to 1D case, nD 2-channel LPPRFBs are also classified into two types. These types have similar symmetry nature as 1D counterparts, namely for one type both filters are symmetric, and for the other type, filters have opposite symmetry. This is addressed in detail in the next chapter. Here we only mention the above result without proof to facilitate the following delivery. The following theorem is valid for both types, though the proof is only given specifically for the first type. For this type, both polyphase components of a given filter should be self-symmetric. A polyphase component, say $A(x, y)$, is self-symmetric if there exists non-negative integers j, k such that

$$A(x, y) = x^j y^k A(x^{-1}, y^{-1}). \tag{6.3}$$

Holding this, the following theorem can be easily proved.

Theorem 6.12 *LP polynomials $A(x, y)$ and $B(x, y)$ can not have LP PI invert polynomials (except trivial case of constants).*

Proof: By contradiction, if there exist LP invert polynomials $M(x, y)$ and $N(x, y)$ such that

$$A(x, y) \cdot M(x, y) + B(x, y) \cdot N(x, y) = 1 \tag{6.4}$$

then, from Eq. (6.3),

$$x^{j_a} y^{k_a} A(x^{-1}, y^{-1}) \cdot x^{j_m} y^{k_m} M(x^{-1}, y^{-1}) + x^{j_b} y^{k_b} B(x^{-1}, y^{-1}) \cdot x^{j_n} y^{k_n} N(x^{-1}, y^{-1}) = 1. \tag{6.5}$$

In order to cancel all terms, it is necessary that $j_a + j_m = j_b + j_n$ and $k_a + k_m = k_b + k_n$. So from (6.5),

$$x^{j_a+j_m} y^{k_a+k_m} \cdot \underbrace{\left(A(x^{-1}, y^{-1}) \cdot M(x^{-1}, y^{-1}) + B(x^{-1}, y^{-1}) \cdot N(x^{-1}, y^{-1})\right)}_{=1 \text{ from } (6.4)} = 1 \quad (6.6)$$

$$\Longrightarrow \quad x^{j_a+j_m} y^{k_a+k_m} = 1 \quad (6.7)$$

Since j_a, k_a etc. are non-negative, the only case when Eq. (6.7) holds is $j_a = j_m = 0$, $k_a = k_m = 0$. So does j_b, j_n, k_b, k_n. This corresponds to the trivial case that all of the polynomials are constants. Hence, such LP PI invert polynomials generally do not exist. ∎

Theorem 6.12 is also valid for the other type of nD 2-channel LPPRFB. A similar proof is not difficult to obtain. Another intuitive viewpoint of the theorem may be as follows. The theorem results simply because it is impossible to have a polynomial $T(x, y)$ on the LHS of Eq. (6.4) to be a symmetric polynomial (all the terms with non-negative degree) with a center in (x^0, y^0), though the trivial case of constants is possible. Now we summarize the above conclusions in the Venn diagram of Figure 6.1.

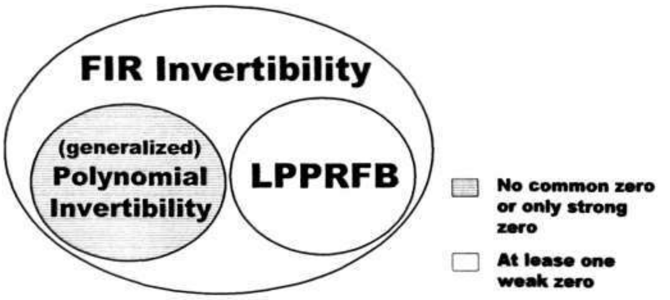


Figure 6.1: The relation between invertibilities and LPPR

Corollary 6.1 *At least one weak zero is necessary for the polyphase components of the filters of a 2-D 2-channel LPPRFB.*

Remark: This corollary can be easily derived from Theorem 6.12. However, as shown by examples in the following section, generally they have more than one common zeros, in both forms of $(0, y_i)$ and $(x_i, 0)$. This is even true when Laurent polynomials are employed, since Laurent polynomials only change the situation of strong zeros. Also note that the theorem is also applicable to higher dimension, yet not valid for 1D case, since in 1D there is no weak zero.

The result in Corollary 6.1 is in fact a necessary condition for LPPRFB conditions to hold. In the following, e.g. the next section and the next chapter, this corollary is applied to the occasions including resultants, LPPRFB factorizations and so on. However, to obtain more meaningful result, stronger and necessary results on the relation between zeros and LPPRFB are desired. This is of further interest.

6.2 Resultants and nD LPPRFB

6.2.1 Resultant of two polynomials

Resultants are an useful tool for checking the coprimeness between two polynomials [91]. Given two polynomials $F(z_1, z_2)$ and $G(z_1, z_2)$, we write

$$\begin{array}{lcl} F(z_1, z_2) = a_0(z_2) + a_1(z_2)z_1^{-1} + \cdots + a_l(z_2)z_1^{-l} & , & a_l \neq 0 \\ G(z_1, z_2) = b_0(z_2) + b_1(z_2)z_1^{-1} + \cdots + b_m(z_2)z_1^{-m} & , & b_m \neq 0 \end{array} . \tag{6.8}$$

Here z_1 is treated as the main variable. For the sake of simplicity, when z_2 is the main variable, the same coefficients a_i or b_i are used except that they are functions of z_1.

The resultant of F and G with respect to z_1 is defined as

$$Res(F, G, z_1) = det \begin{bmatrix} a_l & & & & b_m & & & \\ a_{l-1} & a_l & & & b_{m-1} & b_m & & \\ & a_{l-1} & \ddots & & & b_{m-1} & \ddots & \\ \vdots & & \ddots & a_l & \vdots & & \ddots & b_m \\ & \vdots & & a_{l-1} & & \vdots & & b_{m-1} \\ a_0 & & & & b_0 & & & \\ & a_0 & & \vdots & & b_0 & & \vdots \\ & & \ddots & & & & \ddots & \\ & & & a_0 & & & & b_0 \end{bmatrix} \quad (6.9)$$

where the above matrix (called Sylvester matrix) is $(m+l) \times (m+l)$, and its two parts are not aligned since in general $l \neq m$. Also note that the definition is for polynomials of z_1^{-1} (all items with non-positive degree). For FIR polynomials (some items with positive degree), some monomial can be multiplied to align the degree. In different texts (e.g. [24] and [57]), the Sylvester matrix may be a permutated version. Since the permutation does not change the determinant except the sign, the resultants under different definitions are the same up to the sign.

We list some facts about resultant, which can be found or easily derived from [91].

Fact 6.1 *$F(z_1, z_2)$ and $G(z_1, z_2)$ have a common factor $C(z_1, z_2)$ which has a positive degree in z_1^{-1} (or z_2^{-1}) if and only if $Res(F, G, z_1) \equiv 0$ (or $Res(F, G, z_2) \equiv 0$).*

Remark: The fact provides a criterion for judging the coprimeness between two given polynomials. Actually, there is a systematical way to do so or extract their gcd if they are not coprime. A detailed exposition on the primitive factorization method for extraction of gcd can be found in [95], where polynomials are not restricted to be bivariate. However, it is shown in [96], a paper co-authored by the same author, that for the matrix case, existence of the related primitive factorization as well as construction in the ground field are only guaranteed for bivariate case.

Fact 6.2 *If $Res(F,G,z_1)$ ($Res(F,G,z_2)$) vanishes at $z_2^{\{i\}}$ ($z_1^{\{j\}}$) for some i (j),* ***either*** *$F(z_1,z_2)$ and $G(z_1,z_2)$ have some common zeros at $(z_1^{\{j\}}, z_2^{\{i\}})$ for some $z_1^{\{j\}}$ ($z_2^{\{i\}}$),* ***or*** *$a_l(z_2)$ ($a_l(z_1)$) in $F(z_1,z_2)$ and $b_m(z_2)$ ($b_m(z_1)$) in $G(z_1,z_2)$ have common roots at $z_2 = z_2^{\{i\}}$ ($z_1 = z_1^{\{j\}}$).*

Therefore, primeness between two given polynomials can be determined by checking their resultants. This is elaborated in the next section.

6.2.2 Resultants for LPPRFB

As we mentioned, to check whether two polynomials $A(x,y)$ and $B(x,y)$ have common zeros can be reduced to the task of checking the nature of their resultants. Again, the investigation here is restricted to LPPRFBs.

For a filter of the same type (addressed as type-B later) as that in subsection 6.1.3, if expressed in the form of Eq. (6.8), the following symmetry may be found

$$a_l(z_2) = z_2^{n_a} a_0(z_2^{-1}) \text{ and } b_m(z_2) = z_2^{n_b} b_0(z_2^{-1}) \tag{6.10}$$

for some positive integers n_a and n_b. Similar symmetry also exists for the case when z_2 is treated as main variable. For a filter of the other type (addressed as type-A later), the cross symmetry may be found

$$a_l(z_2) = \pm z_2^{n_a} b_0(z_2^{-1}) \text{ and } b_m(z_2) = \pm z_2^{n_b} a_0(z_2^{-1}) \tag{6.11}$$

for some positive integers n_a and n_b. Here $+$ is for the symmetric filter and $-$ is for the antisymmetric filter.

According to Eq. (6.9), the determinant can be expanded by the first row (for instance, z_1 is the main variable and type-B FB is considered):

$$Res(F,G,z_1) = a_l(z_2) \cdot \mathbf{M}_{11}(z_2) + (-1)^{m+1} b_m(z_2) \cdot \mathbf{M}_{1,m+1}(z_2) \tag{6.12}$$

Here $\mathbf{M}_{i,j}$ is corresponding minor of the Sylvester matrix. The determinant can also be expanded by the last row as

$$\begin{aligned} Res(F,G,z_1) &= (-1)^m a_0(z_2)\cdot \mathbf{M}_{m+l,m}(z_2) + b_0(z_2)\cdot \mathbf{M}_{m+l,m+l}(z_2) \\ &= a_l(z_2^{-1})\left((-1)^m z_2^{-n_a}\cdot \mathbf{M}_{m+l,m}(z_2)\right) + b_m(z_2^{-1})\left(z_2^{-n_b}\cdot \mathbf{M}_{m+l,m+l}(z_2)\right) \end{aligned} \tag{6.13}$$

Therefore, if $a_0(z_2)$ and $b_0(z_2)$ have a common factor $1+kz_2^{-1}$, the resultant $Res(F,G,z_1)$ must have this factor and furthermore, according to Eq. (6.12) and Eq. (6.13), $Res(F,G,z_1)$ must also have factor $1+kz_2$. If only weak zeros of resultants are considered, $Res(F,G,z_1)$ must have the coupled factor $(1+kz_2^{-1})(k+z_2^{-1})$.

As indicated in Theorem 6.12, for 2-D 2-channel LPPRFB, weak zero(s) is needed, which means $a_0(z_1)$ and $b_0(z_1)$ and/or $a_0(z_2)$ and $b_0(z_2)$ have common zero(s). As indicated in Theorem 6.11, only weak zeros and strong zeros are allowed for PR. Hence, if $F(z_1,z_2)$ and $G(z_1,z_2)$ have common zero at $(z_1^i,0)$, z_1^i must be the common zero of $a_0(z_1)$ and $b_0(z_1)$. Considering Fact 6.2, it is not difficult to see that for every couple of roots of the resultant $Res(F,G,z_2)$, the prototype root must be found in $a_0(z_1)$ and $b_0(z_1)$. The same happens to $Res(F,G,z_1)$ and zeros of $(0,z_2^i)$. We summarize the above in the following fact.

Fact 6.3 *A 2-D LP filter with polyphase components $F(z_1,z_2)$ and $G(z_1,z_2)$ as in Eq. (6.8), can have a 2-D LPPR counterpart, if and only if at least one of the corresponding resultants $Res(F,G,z_1)$ and $Res(F,G,z_2)$ is a coupled polynomial (the product of only coupled factors $(1+k_iz_2^{-1})(k_i+z_2^{-1})$ and monomial), and the roots $-k_i$ (or $-1/k_i$) are also the roots of the corresponding a_0 and b_0. The other resultant may be a monomial (though this happens only for some special case).*

This fact provides another criterion to check whether an LP filter have LPPR counterpart. Note that this is also applicable to type-A LPPRFB. The proof can follow the above explanation for type-B.

Example 6.3 *Given two polynomials, which can be regarded as the polyphase components of an LP filter for a type-B FB,*

$$g_{00}(x,y) = (6+5x+x^2) + (5+7x+5x^2)y + (1+5x+6x^2)y^2$$

and

$$g_{01}(x,y) = (2+x) + (1+2x)y,$$

by observation, their common zeros are located at $(-2,0)$ *or* $(0,-2)$, *both of which are the so-called weak zeroes. We can find their FI invert polynomials* $g_{10}(x,y)$ *and* $g_{11}(x,y)$ *such that*

$$g_{00}g_{11} - g_{01}g_{10} = -7xy$$

where one such LP solution is $g_{10}(x,y) = (1+3x) + (3+x)y$ *and* $g_{11}(x,y) = 1$.

Corresponding resultants $Res(g_{00}, g_{01}, y) = 7x(1+2x)(2+x)$ *and* $Res(g_{00}, g_{01}, x) = 7y(1+2y)(2+y)$, *are indeed coupled polynomials. Further, root of* $Res(g_{00}, g_{01}, y)$ *which is* $x = -2$ *is also root of* $a_0(x) = 6+5x+x^2$ *and* $b_0 = 2+x$. *Similar fact happens for* $Res(g_{00}, g_{01}, x)$, *too.*

Example 6.4 *Given two polynomials, which can be regarded as the polyphase components of an LP filter for a type-A FB,*

$$f_{00}(x,y) = (144+516y+42y^2)x^2 + (66+480y+462y^2)x + 6 + 156y + 1008y^2$$

and

$$f_{01}(x,y) = (1008+156y+6y^2)x^2 + (462+480y+66y^2)x + 42 + 516y + 144y^2,$$

having common zeros at $(-\frac{1}{3},0)$, $(-\frac{1}{8},0)$, $(0,-\frac{1}{12})$. *There exist* $f_{10}(x,y)$ *and* $f_{11}(x,y)$ *such that*

$$f_{00}f_{11} - f_{01}f_{10} = -829440x^2y^2$$

where

$$f_{10}(x,y) = (-28y^2 - 344y - 96)x^2 + (-140y^2 - 48y + 20)x + 672y^2 + 104y + 4$$

and

$$f_{11}(x,y) = (-4y^2 - 104y - 672)x^2 + (-20y^2 + 48y + 140)x + 96y^2 + 344y + 28.$$

Their resultants are $Res(f_{00}, f_{01}, x) = -3224862720y^3(y+12)(1+12y)$ *and* $Res(f_{00}, f_{01}, y) = -134369280x(8+x)(1+8x)(3+x)^2(3x+1)^2$ *respectively, satisfying Fact 6.3.*

Example 6.5 *Given two polynomials from another type-A FB,*

$$f_{00}(x,y) = (4+8x) + (100+140x)y + (24+12x)y^2$$

and

$$f_{01}(x,y) = (12+24x) + (140+100x)y + (8+4x)y^2,$$

their Groebner basis is $g_0 = x + 3.75y + 0.5$ *and* $g_1 = y^2$. *Consequently they have a common zero in* $(-\frac{1}{2}, 0)$. *There exist* $f_{01}(x,y)$ *and* $f_{11}(x,y)$ *such that*

$$f_{00}f_{11} - f_{01}f_{10} = -9216xy^2$$

where

$$f_{00}(x,y) = (-6+12x) + (-70+50x)y + (-4+2x)y$$

and

$$f_{01}(x,y) = (-2+4x) + (-50+70x)y + (-12+6x)y^2.$$

Their resultants are $Res(f_{00}, f_{01}, x) = 9216y^2$ *and* $Res(f_{00}, f_{01}, y) = -393216(1+2x)^2(2+x)^2$ *respectively.*

6.2.3 Comments on an LPPRFB construction

In Section 7 of [79], for the 2-D case with quincunx sampling, given a filter $g_0(x, y)$ with polyphase components $g_{00}(x, y)$ and $g_{01}(x, y)$, Basu proposes an approach of LPPRFB completion (that is, of finding another FIR filter $g_1(x, y)$ such that the filters constitute an LPPRFB). Detailed proof of this approach is found in Chapter 10-"On the structure of linear phase perfect reconstruction quincunx filter banks" of [57] (by the same author). In this approach, the solution obtained involves $Res^{-1}(g_{00}, g_{01}, y)$, the inverse of a resultant of the polyphase components of the given filter. It is assumed that for PR, $g_{00}(x, y)$ and $g_{01}(x, y)$ do not share *any* common zero (except possibly $(0, 0)$; similar requirements also appear in [79], [78], and [97]). Consequently, according to Theorem 10.10 in Chapter 10 of [57], the resultants are monomials, hence this approach gives an FIR solution.

However, Theorem 10.10 is obviously incorrect according to our examples 6.3, 6.4 and 6.5. Even if $g_{00}(x, y)$ and $g_{01}(x, y)$ do not share any common zero, their resultant may be a polynomial.

Furthermore, as suggested in Theorem 6.12, for LPPRFB, polynomial invertibility is not possible except for trivial case. Consequently, $g_{00}(x, y)$ and $g_{01}(x, y)$ must share common zeros of the form $(x_i, 0)$ or $(0, y_i)$ and possibly some of the form $(0, 0)$. This *disproves* the assumption mentioned above. The corresponding resultants, from Fact 6.3, are *not* necessarily (and, generally not) a monomial. Indeed, the following case is possible in that the resultant $Res(A, B, x)$ is a polynomial while $Res(A, B, y)$ is a monomial (only one common zero of the form $(x_i, 0)$, as demonstrated in Example 6.5), and the approach of [79] may succeed. However, this is *just* a special case. In general, both resultants are polynomials and the approach does not provide FIR solution.

To summarize, the assumption in [79] and in Chapter 10 of [57] is incorrect and therefore the solutions are by no means a complete approach and do not guarantee

FIR filter bank.

We believe, apart from Theorem 6.12, a number of so-called weak zeros are in fact needed to obtain a good frequency response, and therefore, both resultants need to be polynomials in a good design.

6.3 Groebner basis and its applications

To find the common zeros for a given set of polynomials, apart from resultants, Groebner basis is another popularly used tool. However, the applications of Groebner basis extend further than merely solving polynomial equations. It can also be utilized for a large number of other algorithmic problems including Hermann's original membership problem, the decision of polynomials ideal congruence, etc. In this section, we review the concept and usage of Groebner basis, specifically in the field of nD PRFB. A similar and comprehensive survey can be found in [86,98]. Systematic exposures to GB can be found in [91] and [92].

6.3.1 Introduction to Groebner basis

To obtain an instant picture of Groebner basis, let us have a look at the following example (the same one as that in previous section). Thanks to the efforts from pioneers in developing GB, various computational tools that can calculate GB have been easily available nowadays. SINGULAR is a software kit that provides functions for solving the problems involving singularity theory and algebraic geometry. All examples given in the thesis involving GB are calculated by this excellent tool. It can be obtained from [99].

Example 6.6 *Given polynomials in x, y,*

$$g_{00}(x,y) = (6 + 5x + x^2) + (5 + 7x + 5x^2)y + (1 + 5x + 6x^2)y^2$$

and

$$g_{01}(x,y) = (2+x) + (1+2x)y,$$

their Groebner bases are $g_{00}^G = x+y+2$ *and* $g_{01}^G = y^2+2y$. *Obviously, GB polynomials are much simpler compared to the prototype polynomials. Moreover, since GB don't change the variety of given polynomials, the two set of polynomials have the same set of common zeros [91]. By setting equations,*

$$\begin{aligned} g_{00}^G &= x+y+2 = 0 \\ g_{01}^G &= y^2+2y = 0 \end{aligned}$$

we can solve the common zeros at $(-2,0)$ *or* $(0,-2)$. *It is easy to very that they are indeed common zeros for* g_{00} *and* g_{01}.

We may also find that

$$\begin{cases} (6xy^2 - 6y^3 + 5xy - 12y^2 + x - 4y + 3) \cdot g_{00}^G + (6y^2 + 12y + 5) \cdot g_{01}^G = g_{00} \\ (2y+1) \cdot g_{00}^G + (-2) \cdot g_{01}^G = g_{01} \end{cases} \tag{6.14}$$

In the above example, we may feel familiar with the Eq. (6.14). It reminds us of the standard basis in a linear space. This similarity indeed can give us a sense of their relation. Instead of a linear subspace, here it is an ideal that is generated by the given polynomials. An ideal I is defined as a subset in a ring of n-variable polynomials over a field K, which is denoted as $K[x_1, \cdots x_n]$. The ideal I should satisfy,

- $a, b \in I \Longrightarrow a + b \in I$;
- $a \in I, f \in K[x_1, \cdots, x_n] \Longrightarrow fa \in I$.

According to Hilbert's basis theorem [35], an ideal is always finitely generated, which means any polynomial in this ideal can be expressed as the combination of finite fixed polynomials, as in Eq. (6.14). One choice of these finite fixed polynomials is indeed GB. Now may be a proper time to give a formal definition of GB.

Definition 6.5 *The polynomials $g_1, \cdots g_t$ form a Groebner basis of an ideal I if and only if one of the follwing equivalent conditions holds:*

(i) $f \in I \Longleftrightarrow$ *the residue after division of f by $g_1, \cdots g_t$ is equal to* 0*;*

(ii) $f \in I \Longleftrightarrow f = \sum h_i g_i$ *and the highest monomial of f is equal to the highest of the products of the highest monomials of h_i and g_i;*

(iii) *The ideal $L(I)$ generated by the highest terms of the elements of I is also generated by the highest terms of the polynomials $g_1, \cdots g_t$.*

GB is similar to the basis of a linear subspace. Similarly, GB for a given ideal is not unique. To make it unique, some criterions [35] are introduced to the choice of polynomials $g_1, \cdots g_t$. This results in the so-called reduced GB, which is unique for an ideal [35]. In fact, the GB we obtain for Example 6.6 and those obtained by SINGULAR are actually reduced GBs. In the following, we always mean reduced GB when using GB unless stated otherwise. We summarize the similarities between the linear space and polynomials in Table 6.1, though the analogy may not be strict.

Vector space	Entire space	Subspace	basis of subspace	standard basis
Polynomials	$K[x_1, \cdots x_n]$	Ideal I	Groebner basis	reduced GB

Table 6.1: Similarities between polynomial space and vector space

Another thing that has effect on the obtained GB is monomial ordering, namely, a ranking system for a given multivariate monomial within a set. For example, we can use lexicographical ordering where $a^2bc > ab^3 > ab > a > b^3 > bc^4 > b > c > \cdots$, or degree lexicographical ordering (deglex) where $bc^4 > a^2bc > ab^3 > b^3 > ab > a > b > c > \cdots$, or degree reverse lexicographical ordering (degrevlex) $bc^4 > ab^3 > a^2bc > b^3 > ab > c > b > a > \cdots$. There is almost an infinite number of way to define orderings. However, whatever ordering is chosen for a given problem, we should stick to the ordering, since different ordering usually leads to different results of GB for the same polynomials. For example, the GB in aforementioned example is obtained under lexicographical ordering, while it is $g_{00}^G = x + y + 2$ and $g_{01}^G = 2y^2 - x + 3y - 2$ under degree lexicographical ordering. In fact, these three orderings described above are sufficient for almost all practical purposes and as noted in [92], the choice of degrevlex results in succinct result of Groebner basis computation and is therefore preferred. It is also noted by the same book that for bivariate case, deglex and degrevlex are the same orders.

This completes the introduction to GB. With the above knowledge of GB, we are able to turn to the corresponding applications of GB now.

6.3.2 Applications of Groebner basis

There is a class of problems involving solving the corresponding synthesis polyphase matrix, say $\mathbf{B}(\mathbf{z})$ with dimension $p \times q$, for a filter bank with analysis polyphase matrix $\mathbf{A}(\mathbf{z})$ with dimension $q \times p$, such that

$$\mathbf{B}(\mathbf{z})\mathbf{A}(\mathbf{z}) = \mathbf{I} \tag{6.15}$$

These problems have been tackled by approaches that are based on Groebner basis. The detailed solution first appeared in [100]. Actually, such approach makes more sense when $p < q$, i.e. oversampled filter banks. This is because for critically sampled FB, if the system employs FIR filters and satisfy the PR condition that is similar to Eq. (2.6), its inverse can be easily obtained by its adjoint matrix and the determinant. Hence, we do not elaborate it but refer the interested readers to [86, 100–103].

In fact, in nD, to obtain an analysis polyphase matrix $\mathbf{A}(\mathbf{z})$ that is PR compliant is difficult. We put emphasis on the works of this kind, where the following problem is investigated. For a given column of polynomials $\mathbf{a}(\mathbf{z})$, using Groebner basis, another column of polynomials $\mathbf{b}(\mathbf{z})$ can be obtained, s.t.

$$\mathbf{b}^T(\mathbf{z})\mathbf{a}(\mathbf{z}) = 1 \tag{6.16}$$

Similar approaches to tackle this problem can be found in [23, 85, 104]. These approaches only have some subtle difference, though the major ideas are essentially the same as what we describe in the following procedure.

(i) Multiply a monomial $\mathbf{z}^{\mathbf{c}}$ to all polynomials in $\mathbf{a}(\mathbf{z})$ with a proper power, s.t. no polynomial contains any term with negative power.

(ii) Check the Groebner basis for polynomials $\{a_1(\mathbf{z}), \cdots, a_N(\mathbf{z}), 1 - \mathbf{z}^{\mathbf{1}^T}x\}$, where x is an introduced variable.

(iii) If the GB produced in previous step is $\{1\}$, trace back the GB algorithm to find out $\mathbf{b}'(\mathbf{z}, x)$, s.t.

$$a_1(\mathbf{z})b'_1(\mathbf{z}, x) + \cdot + a_N(\mathbf{z})b'_N(\mathbf{z}, x) + (1 - \mathbf{z}^{\mathbf{1}}x)b'_{N+1}(\mathbf{z}, x) = 1. \tag{6.17}$$

The corresponding $\mathbf{b}(\mathbf{z})$ can be obtained by setting $x = \mathbf{z}^{-\mathbf{1}^T}$.

(iv) Otherwise, no such $\mathbf{b}(\mathbf{z})$ exists.

Here the introduction of variable x is also addressed as Rabinowitsch trick [105]. This trick is well used in system theory, e.g. [106]. It extends solutions to the field of Laurent polynomials. As we mentioned before, generally the RHS of the constraint Eq. (6.16) is not necessarily 1, but some monomial involving only $\mathbf{z}$. This is quite similar to the proof of Theorem 6.11. In [23, 85, 104], this variable is introduced in different manners.

The above result can be easily used for the aforementioned problem of PR completion, namely, for a given filter, to find the complementary filter to constitute a PRFB. Obviously, this is most applicable to the 2-channel case. For M-channel case ($M \geq 3$), this approach can not be applied directly, since nD factorization is necessary for them. nD factorization is difficult and much less amenable to use than 1D factorization, as we show later in Chapter 6. In [104], the above result is indeed applied for the M-channel FB, but only for the oversampled (or so-called nonsubsampled) PRFB depicted in Figure 6.2.

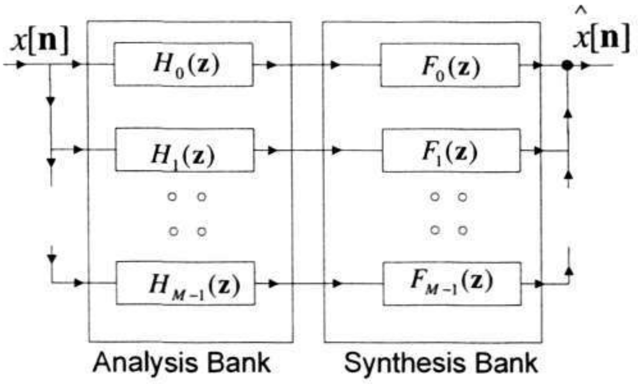


Figure 6.2: An nD nonsubsampled filter bank

In [80], a slightly different approach with an emphasis on linear phase is proposed. It is assumed that a solution to the PR completion problems exists if the given polyphase components are devoid of nontrivial zeros. This is equivalent to the checking in Step (ii) of the above procedure. The approach, in fact, only takes care of one subclass of LPPRFBs. We elaborate on this in the next chapter and propose a remedy.

6.4 Polynomial invertibility and McClellan transform

One possible application of Theorem 6.11 is to check whether a transform from 1D filters to nD filters can retain the PR property. This can be accomplished by checking their common zeros before and after the transform. Now we investigate such a transform, namely, the McClellan transform.

This transform was proposed by McClellan in 1973 and has since been a popular method for designing 2D filters [107]. A 1D zero phase filter is transformed into a 2D zero phase filter as we illustrate in the following example. Initially, the transform is only applicable to zero phase filters of odd length. It is then extended by some researchers to filters of even length, such as [70]. Since FB is a collection of filters, it is natural to consider extending the transform from filters to FBs. The following two examples illustrate this. One has a positive outcome and the other has a negative outcome.

The transforms are generally performed in original domain instead of polyphase domain. For the sake of convenience, note that under quincunx downsampling the fact that polyphase components $H_0(\mathbf{z})$ and $H_1(\mathbf{z})$ do not have nontrivial common zeros is equivalent to the fact that the original filter $H(\mathbf{z})$ does not simultaneously have zeros at (m_0, n_0) as well as at $(-m_0, -n_0)$, where $m_0, n_0 \neq 0$ [79].

Example 6.7 *McClellan transform for filters of odd length.*

Given an odd length LP filter $H^{1D}(z) = F(\frac{z+z^{-1}}{2})$, the corresponding 2-D filter can be obtained by $H^{2D}(\mathbf{z}) = F(\frac{z_1+z_1^{-1}+z_2+z_2^{-1}}{4})$. This is a well-established technique to transform a 1-D LP filter of odd length into 2-D. For PRFB, we further require the initial 1D filter be devoid of zeros simultaneously at $z = m$ and $z = -m$ with $m \neq 0$. We call it the PR constraint in the following. We next show the transform keeps this property intact from 1D to 2D.

Since

$$H^{1D}(z) = F(\frac{z+z^{-1}}{2}) = k\prod_i (z + z^{-1} + a_i),$$

the PR constraint indicates that there are no a_i and a_j such that $a_i = -a_j$ $(i \neq j)$. When it is transformed into 2D,

$$H^{2D}(\mathbf{z}) = F(\tfrac{z_1+z_1^{-1}+z_2+z_2^{-1}}{4}) = k'\textstyle\prod_i (z_1 + z_1^{-1} + z_2 + z_2^{-1} + 2a_i) \qquad (6.18)$$

By contradiction, assume that $H^{2D}(\mathbf{z})$ has zeros at both (m_0, n_0) and $(-m_0, -n_0)$. This means for some a_i and a_j

$$\begin{cases} m_0 + m_0^{-1} + n_0 + n_0^{-1} + 2a_i = 0 \\ -m_0 - m_0^{-1} - n_0 - n_0^{-1} + 2a_j = 0 \end{cases} \qquad (6.19)$$
$$\Longrightarrow \quad a_i = -a_j$$

This contradicts the aforementioned PR constraint. Hence after transformation, the filter is still devoid of zeros that are PR forbidden.

The above argument can be extended to the case of generalized McClellan transform without difficulty. Namely, we can make the following variable substitution,

$$\frac{z + z^{-1}}{2} \longrightarrow \frac{z_1 + z_1^{-1} + kz_2 + kz_2^{-1}}{2 + 2k} \qquad (6.20)$$

The result presented above is more or less intuitive, while the following negative example needs further observation.

Example 6.8 *Extended McClellan transform for filters of even length.*

Given an even length LP filter $H^{1D}(z) = (1 + z)F(\frac{z+z^{-1}}{2})$, the corresponding 2-D filter can be obtained by $H^{2D}(\mathbf{z}) = (1 + z_1)(1 + z_2)F(\frac{z_1+z_1^{-1}+z_2+z_2^{-1}}{4})$. This is an extension of McClellan transform to even length LP filters [70]. However, the transform can not retain the condition on zeros.

For the initial 1D filter,

$$H^{1D}(z) = k(1 + z)\prod_i (z + z^{-1} + a_i),$$

$z = -1$ is always a zero. This generally holds because of symmetry and length. Hence for PR constraint, we further require that $a_i \neq 2$. Use the extended McClellan transform to obtain

$$H^{2D}(\mathbf{z}) = k'(1 + z_1)(1 + z_2)\mathbf{z}^c \prod_i (z_1^2 z_2 + z_1 z_2^2 + z_1 + z_2 + 2a_i z_1 z_2). \qquad (6.21)$$

The first two factors introduce zeros at $(-1, x)$ and $(x, -1)$, where x indicates arbitrary legal value. If we can find zeros at $(1, -x)$ or $(-x, 1)$, the PR property *is no longer satisfied according to Theorem 6.11. Indeed, these kind of zeros exist. By simply letting $x = 1$, we obtain a couple of zeros $(-1, 1)$ and $(1, -1)$ that will destroy the PR property. Hence the extended approach of McClellan transform is not applicable for 2-D LPPRFB. This agrees with the conclusion from [71].*

6.5 Summary

In this chapter, 2-channel M-D LPPRFB is investigated from a viewpoint of algebraic geometry. PR is possible only if there is no nontrivial common zero, and when PR is guaranteed, LPPR property is possible only if the polynomials have at least one common zero of type weak zero. An application of this result is shown in checking whether PR property can be retained after transforming from a 1D filter to a 2D filter. The above result on zeros and LPPRFB can also be expressed in the form of resultant theory. A criterion to check LPPRFB in terms of resultants is developed. This criterion also disproves the approach from an existing work. Another useful tool to check common zero is Groebner basis. As mentioned in our introduction, GB has a much wider range of applications rather than merely checking zeros. We demonstrate its application in the problem of 2-channel nD PRFB completion.

Chapter 7

Design and construction of 2-channel nD LRPRFBs

Results in Chapter 6 are more or less theoretical. How to incorporate them into the applications of design and construction of FB is surely of interest. This chapter is an effort towards it. Accordingly, thanks to the results obtained in Chapter 6, we can now step further in the investigation of nD LPPRFB. In this chapter, we mainly focus on 2-channel nD LPPRFB, which can soundly incorporate results from last chapter. The extension to M-channel case still needs more effort.

We outline the chapter as follows. In the first section, we extend the classification of 2-channel LPPRFB to nD case and give a detailed description for 2D case. Further, we review the approach of PR completion in [80], which only takes care of a subclass of 2-channel LPPRFB and hence is incomplete. For the other subclass, the remedy of PR completion is proposed and proved there. The extensions of the proposed approach are also discussed. In the third section, lifting based factorization for type-B LPPRFB is presented. This factorization is based on a proved lifting-related theorem, as well as on a conjecture which can only be proved for some special cases of lower dimension. Note that the proofs are established using the results we obtain in the previous chapter. Lifting structures are employed to design LPPRFBs, and their comparison with McClellan transform are presented. For type-A FB, similar to 1D, the lattice structure is found to be more suitable for both factorization and design. This is discussed in section 4. Finally, a summary is followed.

7.1 2-channel nD LPPRFB

7.1.1 Linear phase condition in nD

Here we assume, without loss of generality, that only polynomials are considered, which means all the terms of a polynomial have non-negative powers. For Laurent polynomials, they can be calibrated by multiplying with some monomials. In spatial domain, the support of a 2D filter can always be regarded as a rectangle, if necessary padded with zeros. The size of this rectangle is treated as the size of this filter in the following. The size of a filter is defined as in Chapter 1.

For polynomials linear phase property, namely symmetry or antisymmetry, is related to its size. For example, if a filter is of size $S(H) = (c_1 + 1) \times (c_2 + 1)$, it has linear phase iff $H(z_1, z_2) = \pm z_1^{c_1} z_2^{c_2} H(z_1^{-1}, z_2^{-1})$.

For an nD 2-channel LPPRFB ($M \geq 2$ is the number of dimensions), we denote the filters in the analysis bank with $H_0(\mathbf{z})$ and $H_1(\mathbf{z})$ as depicted in Figure 7.1. Here $\mathbf{M}$ is the downsampling matrix.

The PR condition can be obtained from a viewpoint of polyphase components as shown in Appendix B. Alternatively, similar to 1D, the alias term and transfer function for the system in Figure 7.2 may be derived [65].

$$A(\mathbf{z}) = \frac{1}{2}\left(H_0(-\mathbf{z})F_0(\mathbf{z}) + H_1(-\mathbf{z})F_1(\mathbf{z})\right) \tag{7.1}$$

$$T(\mathbf{z}) = \frac{1}{2}\left(H_0(\mathbf{z})F_0(\mathbf{z}) + H_1(\mathbf{z})F_1(\mathbf{z})\right) \tag{7.2}$$

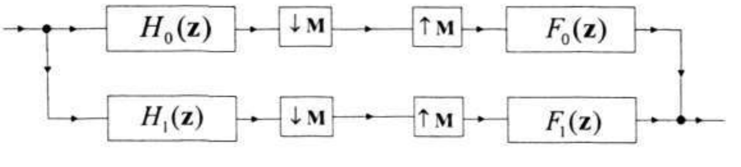


Figure 7.1: nD 2-channel filter bank

To achieve PR, it is necessary that the alias term be canceled. This can be satisfied by the following choice

$$F_0(\mathbf{z}) = z_1 H_1(-\mathbf{z}) \quad , \quad F_1(-\mathbf{z}) = z_1 H_0(\mathbf{z}) \tag{7.3}$$

After alias cancelation, the transfer function $T(\mathbf{z})$ can be expressed as

$$T(\mathbf{z}) = \frac{z_1}{2}\left(H_0(\mathbf{z})H_1(-\mathbf{z}) - H_0(-\mathbf{z})H_1(\mathbf{z})\right) \tag{7.4}$$

To realize the PR condition, we further require $T(\mathbf{z})$ to be equal to a monomial, i.e.

$$T'(\mathbf{z}) = H_0(\mathbf{z})H_1(-\mathbf{z}) - H_0(-\mathbf{z})H_1(\mathbf{z}) = \mathbf{z}^{\mathbf{c}} \tag{7.5}$$

for some integer vector $\mathbf{c} = [c_1, c_2, \cdots, c_M]^T$. Obviously,

$$T'(\mathbf{z}) = \mathbf{z}^{\mathbf{c}} \Leftrightarrow T'(\mathbf{z}^{-1}) = \mathbf{z}^{-\mathbf{c}} \Rightarrow T'(\mathbf{z}) = \mathbf{z}^{2\mathbf{c}}T'(\mathbf{z}^{-1})$$

Hence a *necessary* condition can be achieved that $T'(\mathbf{z})$ is generally symmetric and of odd length along every dimension. This is useful for the following derivation.

Filters $H_i(\mathbf{z})$ may have symmetry or antisymmetry. Hence there are four combinations for $[H_0(\mathbf{z}), H_1(\mathbf{z})]$. Now we show the impossibility of both filters being antisymmetric.

If a filter $H(\mathbf{z})$ is antisymmetric, it indicates the following, $H(-\mathbf{1}) = -H(\mathbf{1})$. If Eq. (7.5) takes this into account,

$$\begin{aligned} & H_0(\mathbf{1})H_1(-\mathbf{1}) - H_0(-\mathbf{1})H_1(\mathbf{1}) \\ = \;& -H_0(\mathbf{1})H_1(\mathbf{1}) + H_0(\mathbf{1})H_1(\mathbf{1}) = 0 \neq \mathbf{1}^{\mathbf{c}} \end{aligned} \tag{7.6}$$

PR condition is no longer satisfied under this case. Therefore, it only leaves three possible combinations, two of which are both comprised of a symmetric filter and an antisymmetric filter and hence can be regarded equivalent. Parallelling to the 1D literature [27], the type with filters having different symmetry is defined as type-A, and the type with filters having both symmetric is defined as type-B 2-channel LPPRFB respectively.

Extending the above results, we further characterize these two types of 2-channel LPPRFB when the dimension is 2 and quincunx downsampling is applied, which is of a significant application interest.

7.1.2 Linear phase condition in 2D for quincunx downsampling

Here quincunx matrix, which is defined as $\mathbf{Q} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$, is chosen for resampling procedures (including downsampling and upsampling). As we already know from previous chapters, the corresponding lattice of quincunx matrix is the only non-separable lattice when decimation ratio is 2. This also explains the motivation of our choice here. For the remaining separable lattices, 1D results can be used directly [27]. In the following, we only focus on quincunx-downsampled PRFBs, as depicted in Figure 7.2.

For filters $H_0(z_1, z_2)$ and $H_1(z_1, z_2)$, to satisfy the aforementioned necessary condition for $T'(z_1, z_2)$, the lengths of the same dimension should be of the same nature. Namely, both horizontal lengths are even or odd. So do the vertical lengths. Consequently, only 4 length combinations are possible for two filters $H_0(z_1, z_2)$ and $H_1(z_1, z_2)$. Together with 4 different symmetry combinations, there are 16 possible cases. Further considering the symmetry constraint and the fact that $T'(z_1, z_2)$ cannot have zeros at (x_i, y_i) ($x_i \neq 0$ and $y_i \neq 0$) according to Eq. (7.5), only 5 out of these 16 combinations are possible. The result is listed in Table 7.1. Here are some explanatory notes. In case ① $T'(z_1, z_2)$ has zeros at $(1, -1)$ and $(-1, 1)$ (as shown in subsection 7.1.1), which conflicts the PR condition in Eq. (7.5). In case ②

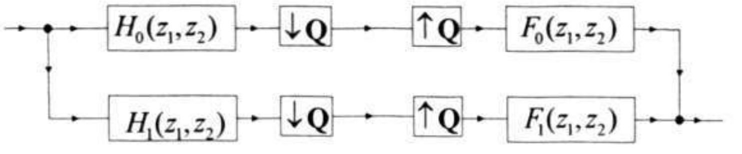


Figure 7.2: 2D 2-channel filter bank with quincunx resampling

$T'(z_1, z_2)$ is antisymmetric, which conflicts the symmetry requirement. In case ③ $T'(z_1, z_2)$ has zeros at $(1, 1)$ and $(-1, -1)$ (already shown in subsection 7.1.1), which contradicts the PR condition in Eq. (7.5). Cases ④and ⑤are possible. As said, cases ④and ⑤are defined as type-A and type-B 2D 2-channel LPPRFB respectively. The results agree with that in [79], where they are defined as type-K_1 and type-K_0 respectively.

$H_0(z_1, z_2) \diamond H_1(z_1, z_2)$	S◇S	S◇A	A◇S	A◇A
E×E ◇ E×E	× ①	× ②	× ②	× ③
E×O ◇ E×O	× ②	√ ④	√ ④	× ②
O×E ◇ O×E	× ②	√ ④	√ ④	× ②
O×O ◇ O×O	√ ⑤	× ②	× ②	× ③

Table 7.1: The possible cases for 2-D LPPRFB.

As we show, the symmetry is related to the filter size. So are the symmetry centers. By combining the above results, it is not difficult to obtain the following facts. Let $z_1^m z_2^n$ be the difference (or quotient) of symmetry centers of two filters. Then for type-A, m and n are both even or both odd, but for type-B one of m and n is even while the other is odd. This can simply explain why one filter has to be essentially delayed to obtain PRFB using McClellan transform in [72].

In polyphase domain, the PR condition of Eq. (7.5) can be translated as

$$E_{00}(z_1, z_2)E_{11}(z_1, z_2) - E_{01}(z_1, z_2)E_{10}(z_1, z_2) = \mathbf{z}^{\mathbf{c}'} \tag{7.7}$$

for some integer vector $\mathbf{c}' = [c_1', c_2']^T$. Here $E_{i,j}$ $(i, j = 0, 1)$ is a polyphase component of filter H_i. For type-A, E_{00} and E_{01} are cross-symmetric and E_{10} and E_{11} are cross-antisymmetric, or vice versa. They may be of the same size. For type-B, all the polyphase components are symmetric. The polyphase components must be of different sizes.

These properties easily remind us of those in 1D case. For comparison, we also summarize the above facts in a table (Table 7.2) that is similar to Table 3.1.

	Type-A	Type-B
size nature	(E×O,E×O) or (O×E,O×E)	(O×O,O×O)
size	may the same or different	must be different
symmetry of filters	sym. & antisym.	both are sym.
symmetry of polyphase components	cross-sym. & cross-antisym.	both are self-sym.
Diff. of sym. centers	$z_1^m z_2^n$, $m+n$ even	$m+n$ odd

Table 7.2: The properties for 2D 2-channel LPPRFB under quincunx downsampling

7.2 PR completion using symmetry extension

As before, the problem of interest is formulated as follows. Given the polyphase components $[E_{00}(\mathbf{z}), E_{01}(\mathbf{z})]$ of an LP filter $E_0(\mathbf{z})$, if they are devoid of any non-trivial common zeros [80], another LP filter $E_1(\mathbf{z})$ has to be found with polyphase components $[E_{10}(\mathbf{z}), E_{11}(\mathbf{z})]$ such that Eq. (7.7) can be satisfied. This is known as the symmetric completion problem.

The problem can be further divided into two stages. The first stage is to seek a filter that is PR complementary to the given filter. The second stage is to incorporate LP property to the obtained filter without destroying PR condition. For the first problem, GB can be applied as mentioned in the last chapter. The second task is solved by the so-called symmetry extension. We now describe the algorithms for type-A and type-B in detail.

7.2.1 Symmetry extension for type-B LPPRFB

Algorithm D of [80] parameterizes LPPR complementary filter only for type-B LP-PRFB. We list this algorithm without proof, which can be found in [80] or can be derived as done for type-A LPPRFB in the following subsection.

Assume that the given filter has self-symmetric polyphase components, $E_{00}(\mathbf{z}) = \mathbf{z}^{\mathbf{m}_{00}} E_{00}(\mathbf{z}^{-1})$ and $E_{01}(\mathbf{z}) = \mathbf{z}^{\mathbf{m}_{01}} E_{01}(\mathbf{z}^{-1})$.

Algorithm 7.1 *[80]*

(i) *Calculate the Groebner basis of the ideal generated by* $E_{00}(\mathbf{z})$ *and* $E_{01}(\mathbf{z})$.

(ii) *Within the ideal mentioned above, choose a monomial with minimum degree, say* $\mathbf{z}^{\mathbf{s}}$.

(iii) *Express the monomial with* $E_{00}(\mathbf{z})$ *and* $E_{01}(\mathbf{z})$ *as*

$$E_{00}(\mathbf{z})E'_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E'_{10}(\mathbf{z}) = \mathbf{z}^{\mathbf{s}}$$

(iv) *Choose* $\mathbf{m_{10}}$ *and* $\mathbf{m_{11}}$ *such that* $\mathbf{z}^{2\cdot\mathbf{s}} = \mathbf{z}^{\mathbf{m}_{00}+\mathbf{m}_{11}} = \mathbf{z}^{\mathbf{m}_{01}+\mathbf{m}_{10}}$. *Let* $E''_{10}(\mathbf{z}) = \mathbf{z}^{\mathbf{m}_{10}}E'_{10}(\mathbf{z}^{-1})$ *and* $E''_{11}(\mathbf{z}) = \mathbf{z}^{\mathbf{m}_{11}}E'_{11}(\mathbf{z}^{-1})$.

(v) *The desired polyphase components* $\{E_{10}(\mathbf{z}), E_{11}(\mathbf{z})\}$ *for a LP and PR complementary filter can be found as*

$$\begin{aligned} E_{10}(\mathbf{z}) &= \tfrac{1}{2}\left(E'_{10}(\mathbf{z}) + E''_{10}(\mathbf{z})\right) \\ E_{11}(\mathbf{z}) &= \tfrac{1}{2}\left(E'_{11}(\mathbf{z}) + E''_{11}(\mathbf{z})\right) \end{aligned} \tag{7.8}$$

7.2.2 Symmetry extension for type-A LPPRFB

The algorithm proposed here is motivated by the previous approach for type-B LPPRFB. According to Table 7.2, a filter in type-A LPPRFB can be either symmetric or antisymmetric. Here we assume that the given filter is symmetric. The antisymmetric case can be similarly achieved. Now, the polyphase components of the given filter are cross symmetric, i.e. $E_{00}(\mathbf{z}) = \mathbf{z}^{\mathbf{m}_0}E_{01}(\mathbf{z}^{-1})$. Using the following algorithm, we can find its counterpart filter that is both LP and PR complementary.

Algorithm 7.2 (i) *Calculate the Groebner basis of the ideal generated by* $E_{00}(\mathbf{z})$ *and* $E_{01}(\mathbf{z})$.

(ii) *Within the ideal mentioned above, choose a monomial with minimum degree, say* $\mathbf{z}^{\mathbf{s}}$.

(iii) *Express the monomial with* $E_{00}(\mathbf{z})$ *and* $E_{01}(\mathbf{z})$ *as*

$$E_{00}(\mathbf{z})E'_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E'_{10}(\mathbf{z}) = \mathbf{z}^{\mathbf{s}}$$

(iv) *Choose* $\mathbf{m_1}$ *such that* $\mathbf{z}^{2\cdot\mathbf{s}} = \mathbf{z}^{\mathbf{m_0}+\mathbf{m_1}}$. *Let* $E''_{10}(\mathbf{z}) = -\mathbf{z}^{\mathbf{m_1}}E'_{11}(\mathbf{z}^{-1})$ *and* $E''_{11}(\mathbf{z}) = -\mathbf{z}^{\mathbf{m_1}}E'_{10}(\mathbf{z}^{-1})$.

(v) *The desired solution set* $\{E_{10}(\mathbf{z}), E_{11}(\mathbf{z})\}$ *can be found as*

$$\begin{array}{l} E_{10}(\mathbf{z}) = \frac{1}{2}\left(E'_{10}(\mathbf{z}) + E''_{10}(\mathbf{z})\right) \\ E_{11}(\mathbf{z}) = \frac{1}{2}\left(E'_{11}(\mathbf{z}) + E''_{11}(\mathbf{z})\right) \end{array} \tag{7.9}$$

Note that the feasibility of the second step in Algorithm 7.2 is guaranteed by the fact that $E_{00}(\mathbf{z})$ and $E_{01}(\mathbf{z})$ are devoid of nontrivial common zero [91]. This is a typical problem, called membership problem, that can be solved by Groebner basis [91]. The minimum degree of the monomial is, in fact, chosen to avoid ambiguity. If another monomial is chosen, the algorithm is still valid, though it may lead to a different result.

Proof: For LP condition,

$$\begin{aligned} E_{10}(\mathbf{z}) &= \tfrac{1}{2}\left(E'_{10}(\mathbf{z}) + E''_{10}(\mathbf{z})\right) \\ &= \tfrac{1}{2}\left(-\mathbf{z}^{\mathbf{m_1}}E''_{11}(\mathbf{z}^{-1}) - \mathbf{z}^{\mathbf{m_1}}E'_{11}(\mathbf{z}^{-1})\right) \\ &= -\mathbf{z}^{\mathbf{m_1}}\cdot\tfrac{1}{2}\left(E''_{11}(\mathbf{z}^{-1}) + E'_{11}(\mathbf{z}^{-1})\right) \\ &= -\mathbf{z}^{\mathbf{m_1}}E_{11}(\mathbf{z}^{-1}) \end{aligned}$$

They are indeed cross-antisymmetric, as required in Table 7.2.

For PR condition,

$$\begin{aligned} & E_{00}(\mathbf{z})E_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E_{10}(\mathbf{z}) \\ = \;& E_{00}(\mathbf{z})\cdot\tfrac{1}{2}\left(E'_{11}(\mathbf{z}) + E''_{11}(\mathbf{z})\right) - E_{01}(\mathbf{z})\cdot\tfrac{1}{2}\left(E'_{10}(\mathbf{z}) + E''_{10}(\mathbf{z})\right) \\ = \;& \tfrac{1}{2}\mathbf{z}^{\mathbf{s}} + \tfrac{1}{2}\left(E_{00}(\mathbf{z})E''_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E''_{10}(\mathbf{z})\right) \\ = \;& \tfrac{1}{2}\mathbf{z}^{\mathbf{s}} + \tfrac{1}{2}\left(\mathbf{z}^{\mathbf{m_0}}E_{01}(\mathbf{z}^{-1})\cdot(-\mathbf{z}^{\mathbf{m_1}})E'_{10}(\mathbf{z}^{-1}) - \mathbf{z}^{\mathbf{m_0}}E_{00}(\mathbf{z}^{-1})\cdot(-\mathbf{z}^{\mathbf{m_1}})E'_{11}(\mathbf{z}^{-1})\right) \\ = \;& \tfrac{1}{2}\mathbf{z}^{\mathbf{s}} + \tfrac{1}{2}\mathbf{z}^{\mathbf{m_0}+\mathbf{m_1}}\left(E_{00}(\mathbf{z}^{-1})E'_{11}(\mathbf{z}^{-1}) - E_{01}(\mathbf{z}^{-1})E'_{10}(\mathbf{z}^{-1})\right) \\ = \;& \tfrac{1}{2}\mathbf{z}^{\mathbf{s}} + \tfrac{1}{2}\mathbf{z}^{\mathbf{m_0}+\mathbf{m_1}-\mathbf{s}} = \mathbf{z}^{\mathbf{s}} \end{aligned}$$

∎

Example 7.9 *We now show a construction example. $H_0(\mathbf{z})$ is the given filter with (quincunx-downsampled) polyphase components*

$$E_{00}(\mathbf{z}) = 3 + 9z_1 + 36z_2 + 12z_1z_2$$

and

$$E_{01}(\mathbf{z}) = 12 + 36z_1 + 9z_2 + 3z_1z_2$$

Following the first three steps of Algorithm 7.2, we can find

$$E_{00}(\mathbf{z}) \cdot (z_2 + 12) - E_{01}(\mathbf{z}) \cdot (4z_2 + 3) = 360z_2.$$

Here z_2 is the chosen monomial and hence $\mathbf{z}^{\mathbf{s}} = z_2$. By observing the symmetry, $\mathbf{z}^{\mathbf{m_0}} = z_1z_2$. Therefore, $\mathbf{z}^{\mathbf{m_1}} = z_2^2/z_1z_2 = z_2/z_1$.

According to the last step,

$$\begin{aligned} E_{10}(\mathbf{z}) &= \tfrac{1}{2}\left((4z_2+3) - z_2/z_1 \cdot (z_2^{-1} + 12)\right) \\ &= \tfrac{1}{2z_1}\left(-1 + 3z_1 - 12z_2 + 4z_1z_2\right) \\ E_{11}(\mathbf{z}) &= \tfrac{1}{2}\left((z_2+12) - z_2/z_1 \cdot (4z_2^{-1} + 3)\right) \\ &= \tfrac{1}{2z_1}\left(-4 + 12z_1 - 3z_2 + z_1z_2\right) \end{aligned}$$

It is easy to verify that

$$E_{00}(\mathbf{z})E_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E_{10}(\mathbf{z}) = 360z_2.$$

It is not difficult to verify that if the given filter is antisymmetric, the algorithm is still valid as long as the symmetric parts chosen in step (iv) are changed to $E''_{10}(\mathbf{z}) = \mathbf{z}^{\mathbf{m_1}} E'_{11}(\mathbf{z}^{-1})$ and $E''_{11}(\mathbf{z}) = \mathbf{z}^{\mathbf{m_1}} E'_{10}(\mathbf{z}^{-1})$.

7.2.3 Comments on generalizing symmetry extension approach

7.2.3.1 Extension to other downsampling matrices

By observing Algorithm 7.1 and 7.1, it may be found that the algorithms have nothing to do with downsampling matrices, which means they are generally applicable to all 2-D 2-channel LPPRFBs. This is certainly true, though we should be cautious about the fact that the polyphase components and their symmetry from LP depends on the downsampling matrix. This can be illustrated by a counterexample.

Example 7.10 *Given two self-symmetric polyphase components*

$$E_{00}(\mathbf{z}) = 1 + 2z_1 + 2z_2 + z_1 z_2 \text{ and } E_{01}(\mathbf{z}) = 3$$

If they are downsampled by quincunx matrix, by tracing back, the original filter

$$\begin{aligned} H_0(\mathbf{z}) &= E_{00}(z_1 z_2, z_1 z_2^{-1}) + z_1 E_{01}(z_1 z_2, z_1 z_2^{-1}) \\ &= 1 + 3z_1 + z_1^2 + 2z_1 z_2 + 2z_1 z_2^{-1} \end{aligned}$$

is obviously an LP filter. However, if we assume the downsampling matrix as $\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$, *the original filter*

$$\begin{aligned} H_0(\mathbf{z}) &= E_{00}(z_1, z_2^2) + z_2 E_{01}(z_1, z_2^2) \\ &= 1 + 2z_1 + 2z_2^2 + z_1 z_2^2 + 3z_2 \end{aligned}$$

is not an LP filter. Therefore, even if we find counterparts for $E_{00}(\mathbf{z})$ and $E_{01}(\mathbf{z})$ by using the above-mentioned algorithm, they can only form an LPPRFB under certain downsampling matrices.

7.2.3.2 Extension to higher dimension and multiple channel case

Similar to Algorithm 7.1 [80], the proposed Algorithm 7.2 is actually valid for higher dimensional case, since the vector $\mathbf{z}$ is not necessary to have only two components.

Up to now, the application of GB in nD PRFB has been restricted to 2-channel case or a nonsubsampled M-channel case. That is because, for multiple-channel case, the PR counterpart of a given filter is a set of filters unlike just one filter of 2-channel. This requires sophisticated factorization of M-D, which is much less developed and usable than that in 1D. We may illustrate the difficulty by the following example.

Example 7.11 *Here, we actually discuss the extension of PR completion by a GB approach rather than LPPR completion. Note that the LPPR conditions for multiple channel and multidimensional case are quite complicated and hence remain open.*

For 3-channel case, given a filter $H_0(\mathbf{z})$ with polyphase components $E_{00}(\mathbf{z})$, $E_{01}(\mathbf{z})$ and $E_{02}(\mathbf{z})$, by calculating their Groebner basis and tracing back, polynomials $A_0(\mathbf{z})$, $A_1(\mathbf{z})$ and $A_2(\mathbf{z})$ can be obtained such that

$$E_{00}(\mathbf{z}) \cdot A_0(\mathbf{z}) + E_{01}(\mathbf{z}) \cdot A_1(\mathbf{z}) + E_{02}(\mathbf{z}) \cdot A_2(\mathbf{z}) = \mathbf{z}^{\mathbf{s}}. \tag{7.10}$$

From the constraint of PR, if there exist $H_1(\mathbf{z})$ and $H_2(\mathbf{z})$ forming a PRFB with $H_0(\mathbf{z})$, their polyphase components must satisfy

$$\begin{vmatrix} E_{00}(\mathbf{z}) & E_{01}(\mathbf{z}) & E_{02}(\mathbf{z}) \\ E_{10}(\mathbf{z}) & E_{11}(\mathbf{z}) & E_{12}(\mathbf{z}) \\ E_{20}(\mathbf{z}) & E_{21}(\mathbf{z}) & E_{22}(\mathbf{z}) \end{vmatrix} = \mathbf{z}^{\mathbf{s}}.$$

By equating the above two equations, we obtain

$$A_0(\mathbf{z}) = E_{11}(\mathbf{z})E_{22}(\mathbf{z}) - E_{12}(\mathbf{z})E_{21}(\mathbf{z}) \tag{7.11}$$

$$A_1(\mathbf{z}) = E_{12}(\mathbf{z})E_{20}(\mathbf{z}) - E_{10}(\mathbf{z})E_{22}(\mathbf{z}) \tag{7.12}$$

$$A_2(\mathbf{z}) = E_{10}(\mathbf{z})E_{21}(\mathbf{z}) - E_{11}(\mathbf{z})E_{20}(\mathbf{z}) \tag{7.13}$$

Eq. (7.11)$\times E_{20}(\mathbf{z})$+Eq. (7.13)$\times E_{22}$ gives

$$A_0 \cdot E_{20} + A_2 \cdot E_{22} = E_{10}E_{21}E_{22} - E_{12}E_{21}E_{20}$$

By a further observation, we obtain,

$$A_0 \cdot E_{20} + A_2 \cdot E_{22} + A_1 \cdot E_{21} = 0 \tag{7.14}$$

Similarly,

$$A_0 \cdot E_{10} + A_2 \cdot E_{12} + A_1 \cdot E_{11} = 0 \tag{7.15}$$

We can summarize Eq. (7.10), Eq. (7.14) and Eq. (7.15) in a matrix form as

$$\begin{bmatrix} E_{00} & E_{01} & E_{02} \\ E_{10} & E_{11} & E_{12} \\ E_{20} & E_{21} & E_{22} \end{bmatrix} \begin{bmatrix} A_0 \\ A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} \mathbf{z}^{\mathbf{s}} \\ 0 \\ 0 \end{bmatrix} \tag{7.16}$$

Eq. (7.16) indicates the indeterminate polynomials $E_{1,i}(\mathbf{z})$ and $E_{2,i}(\mathbf{z})$ are actually syzygy modules for the GB $A_i(\mathbf{z})$ of the given polynomials $E_{0,i}$ with $i = 0, 1, 2$. Obviously, the above approach is difficult to extend to cases with more channels, since the relations between the generated GB and the indeterminate polynomials are hard to establish. To apply the Groebner basis to multiple channel nD FB is hence of interest and worth further study.

7.3 Lifting factorization and design of type-B nD LPPRFB

Lifting scheme, as introduced in Chapter 3, can structurally incorporate PR property and is hence increasingly favored. Here we investigate its application in nD LPPRFB. However, for the nD case the Euclidean algorithm does not work and the factorization result can not be extended directly. Here a different approach based on a conjecture is presented.

7.3.1 Lifting factorization algorithm

The factorization is performed in the polyphase domain as in Eq. (7.7). Firstly, an important theorem is introduced.

Theorem 7.13 *For a given filter $H_0(\mathbf{z})$, if there exist two different filters $H_1(\mathbf{z})$ and $H_1'(\mathbf{z})$ such that both of them can constitute a PRFB with $H_0(\mathbf{z})$, their polyphase components must be related by some (Laurent) polynomial $K(\mathbf{z})$ as*

$$\begin{aligned} E_{10}(\mathbf{z}) &= E_{00}(\mathbf{z}) \cdot K(\mathbf{z}) + E_{10}'(\mathbf{z}) \\ E_{11}(\mathbf{z}) &= E_{01}(\mathbf{z}) \cdot K(\mathbf{z}) + E_{11}'(\mathbf{z}) \end{aligned} \tag{7.17}$$

Proof: Note that for simplicity the Laurent polynomials are employed. Hence the RHS of Eq. (7.7) can be 1, when the common monomial $\mathbf{z}^{\mathbf{c}'}$ is absorbed by the Laurent polynomials $E_{00}(\mathbf{z})$ and $E_{01}(\mathbf{z})$. If both $H_1(\mathbf{z})$ and $H_1'(\mathbf{z})$ are PR counterparts of $H_0(\mathbf{z})$, the following is satisfied.

$$E_{00}(\mathbf{z})E_{11}(\mathbf{z}) - E_{01}(\mathbf{z})E_{10}(\mathbf{z}) = 1 \tag{7.18}$$

$$E_{00}(\mathbf{z})E_{11}'(\mathbf{z}) - E_{01}(\mathbf{z})E_{10}'(\mathbf{z}) = 1 \tag{7.19}$$

They can be expressed in matrix form

$$\begin{vmatrix} E_{00} & E_{01} \\ E_{10} & E_{11} \end{vmatrix} = 1 \quad , \quad \begin{vmatrix} E_{00} & E_{01} \\ E_{10}' & E_{11}' \end{vmatrix} = 1 \tag{7.20}$$

Hence it may be verified that

$$\begin{bmatrix} E_{00} & E_{01} \\ E_{10} & E_{11} \end{bmatrix} \cdot \begin{bmatrix} E_{00} & E_{01} \\ E_{10}' & E_{11}' \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ E_{10}E_{11}' - E_{11}E_{10}' & 1 \end{bmatrix} \tag{7.21}$$

Finally, letting $K(\mathbf{z}) = E_{10}E_{11}' - E_{11}E_{10}'$, we have

$$\begin{bmatrix} E_{00} & E_{01} \\ E_{10} & E_{11} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ K & 1 \end{bmatrix} \cdot \begin{bmatrix} E_{00} & E_{01} \\ E_{10}' & E_{11}' \end{bmatrix} \tag{7.22}$$

This is indeed what is shown in the theorem. ∎

In Eq. (7.22), it is shown that the PR counterparts for a given filter are related by a lifting step. Conversely, all the PR counterparts can be bridged from a special PR counterpart with a lifting step as long as no extra condition is imposed on $K(\mathbf{z})$. This theorem is valid for both type-A and type-B LPPRFBs and even for higher dimensional case, since in the proof the dimension of variable $\mathbf{z}$ is not restricted. To factorize an arbitrary type-B FB, the following conjecture is needed.

Conjecture 7.1 *For LP polyphase components $E_{00}(\mathbf{z})$ and $E_{01}(\mathbf{z})$, if they have LPPR counterparts (or as mentioned previously, LP PI invert polynomials), there must exist some $E_{10}(\mathbf{z})$ and $E_{11}(\mathbf{z})$ among these invert polynomials such that $S(E_{11}) \leq S(E_{00})$ and $S(E_{10}) \leq S(E_{01})$.*

Remark: The conjecture can be proved for some special cases. For $S(E_{00}) = 2 \times 2$ and $S(E_{01}) = 1 \times 1$, without loss of generality assume $E_{00} = 1 + az_1^{-1} + (a + z_1^{-1})z_2^{-1}$ and $E_{01} = 1$, and the PR counterparts are trivially $E_{10} = 1$ and $E_{11} = 0$. We next show the proof for another case when $S(E_{00}) = 3 \times 3$ and $S(E_{01}) = 2 \times 2$.

Without loss of generality, we assume

$$E_{01}(\mathbf{z}) = 1 + az_1^{-1} + (a + z_1^{-1})z_2^{-1}. \tag{7.23}$$

As shown in Theorem 6.12, at least one weak zero is necessary for LPPR condition. For E_{01}, there are two weak zeros $(-\frac{1}{a}, 0)$ and $(0, -\frac{1}{a})$. Note that here we regard $E_{01}(\mathbf{z})$ and $E_{00}(\mathbf{z})$ as polynomials of z_1^{-1} and z_2^{-1} rather than z_1 and z_2. Assuming that E_{01} and E_{00} have at least one common zero, say $(-\frac{1}{a}, 0)$, we can express E_{00} as

$$E_{00}(\mathbf{z}) = 1 + (a+b)z_1^{-1} + abz_1^{-2} + (c + kz_1^{-1} + cz_1^{-2})z_2^{-1} + (ab + (a+b)z_1^{-1} + z_1^{-2})z_2^{-2} \tag{7.24}$$

Here a, b, c, k are arbitrary legal coefficients. Symmetry has been incorporated in the above assumption.

By setting $E_{01} = 0$ and assuming $a \neq \pm 1$ (the exception can be treated similarly), we obtain $z_2^{-1} = -\frac{(1+az_1^{-1})}{a+z_1^{-1}}$. Substituting this into $E_{00} = 0$, the following can be obtained.

$$\frac{(1+az_1^{-1})\left((-c+a+b)z_1^{-2} + (2ab - k + 2)z_1^{-1} - c + a + b\right)}{a + z_1^{-1}} = 0 \tag{7.25}$$

Since from Theorem 3 we know that common zeros other than $(-\frac{1}{a}, 0)$ and $(0, -\frac{1}{a})$ are not allowed, for Eq. (7.25) only solutions are $z_1^{-1} = -\frac{1}{a}$ or 0. Holding this, from Eq. (7.25) the following condition must be satisfied.

$$-c + a + b = 0 \tag{7.26}$$

$$2ab - k + 2 \neq 0 \tag{7.27}$$

Therefore, we can find PR counterparts for E_{00} and E_{01} as $E_{11} = 1$ and $E_{10} = 1 + bz_1^{-1} + (b + z_1^{-1})z_2^{-1}$ such that

$$E_{00} \cdot E_{11} - E_{10} \cdot E_{01} = (k - 2ab - 2) \cdot z_1^{-1} z_2^{-1} \neq 0 \tag{7.28}$$

Indeed $S(E_{11}) \leq S(E_{00})$ and $S(E_{10}) \leq S(E_{01})$.

Unfortunately, the general case cannot be proved by the above mentioned approach. Based on this conjecture and Theorem 7.13, the factorization of 2-D 2-channel type-B LPPRFB is proposed. It is described in the following algorithm.

Algorithm 7.3 *An algorithm for factorizing a 2-D 2-channel type-B LPPRFB into lifting steps.*

Input: A 2-D 2-channel type-B LPPRFB with polyphase components

$$\{E_{00}(\mathbf{z}), E_{01}(\mathbf{z}), E_{10}(\mathbf{z}), E_{11}(\mathbf{z})\}.$$

Output: Lifting blocks $K_i(\mathbf{z})$ $(i = 1 \cdots L)$ that can factorize the given FB.

(i) *Choose the smaller size (either in one or all dimensions) filter out of given two filters, for example $H_0(\mathbf{z})$. let $L = 1$, $A(\mathbf{z}) = H_0(\mathbf{z})$ and $B(\mathbf{z}) = H_1(\mathbf{z})$.*

(ii) *Find a PR counter parts for polyphase components of $A(\mathbf{z})$ such that the size requirement in Conjecture 7.1 is satisfied. The resulting polyphase components can be denoted as $B_0'(\mathbf{z})$ and $B_1'(\mathbf{z})$ (and correspondingly the filter as $B'(\mathbf{z})$).*

(iii) *Since $B(\mathbf{z})$ and $B'(\mathbf{z})$ are of different sizes but both complementary to $A(\mathbf{z})$, according to Theorem 7.13, there exists a filter such that they are related. The filter can be expressed as $K_L(\mathbf{z})$.*

(iv) *If $S(A(\mathbf{z})) > 1 \times 1$, let $L = L + 1$, $B(\mathbf{z}) = A(\mathbf{z})$, and $A(\mathbf{z}) = B'(\mathbf{z})$ and repeat step 2, otherwise cease.*

When the algorithm ceases, both $A(\mathbf{z})$ and $B(\mathbf{z})$ should be monomials. By inserting delays (as done in 1D) to H_0 and H_1 without destroying PR, it is possible that after the last step of Algorithm 7.3, $A(\mathbf{z}) = c_1$ and $B(\mathbf{z}) = c_2 z_1^{-1}$ with real coefficients c_1 and c_2. Also note that PR counterparts in step (ii) can be obtained using GB. The factorization for a possible 2D type-B LPPRFB is shown in Figure 7.3. The alternation of the directions of lifting blocks reflect the filters exchanging in Step (iv) of Algorithm 7.3.

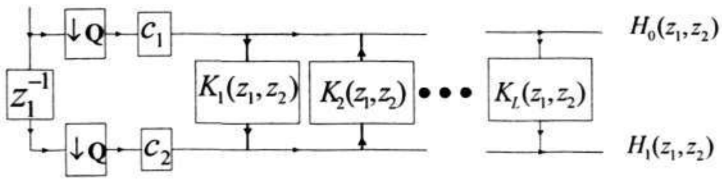


Figure 7.3: 2D 2-channel filter bank lifting factorization

7.3.2 Examples of lifting factorization and design

Example 7.12 *An LPPRFB from [80] can be decomposed into polyphase components under quincunx decimation as*

$$\begin{aligned}
E_{00}(\mathbf{z}) &= -0.075(z_1^3 z_2 + z_1^2 z_2^2 + z_1^2 + z_1 z_2) - 0.0375(z_1^3 + z_1 z_2^2 + z_1 + z_1^3 z_2^2) + 0.85 z_1^2 z_2,\\
E_{01}(\mathbf{z}) &= 0.125(z_1^2 z_2 + z_1^2 + z_1 z_2 + z_1),\\
E_{10}(\mathbf{z}) &= -0.3(z_1^2 z_2 + z_1^2 + z_1 z_2 + z_1), \qquad E_{11} = z_1
\end{aligned} \tag{7.29}$$

Note that they share an unnecessary common factor z_1. After extracting this common factor, E_{00} and E_{01} do share some weak zeros at $(0,-1)$ and $(-1,0)$. This agrees with Corollary 6.1. Following Algorithm 7.3, the FB can be factorized as

$$\begin{bmatrix} E_{00} & E_{01} \\ E_{10} & E_{11} \end{bmatrix} = \begin{bmatrix} z_1^2 z_2 & 0.125 z_1 C(\mathbf{z}) \\ 0 & z_1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -0.3C(\mathbf{z}) & 1 \end{bmatrix} \tag{7.30}$$

where $C(\mathbf{z}) = 1+z_1+z_2+z_1 z_2$.

Similar to the 1D result, the above factorization also has a kernel lifting block $C(\mathbf{z})$, which appears in the form of $(1+z)$ in 1D lifting factorization. This easily reminds us of the McClellan transform. Indeed, this can be regarded as an approach to implement the McClellan transform. It is also known that the transform can be generalized by replacing the kernel lifting block by some other zero phase 2D polynomial, which is called generalized McClellan transform (GMT).

Based on the above factorization result (though the completeness depends on Conjecture 7.1), type-B LPPRFBs can still be designed by corresponding lifting construction as

$$\begin{bmatrix} E_{00} & E_{01} \\ E_{10} & E_{11} \end{bmatrix} = \prod_{i=1}^{N} \begin{bmatrix} 1 & k_{2i-1}C(\mathbf{z}, m_{2i-1}) \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ k_{2i}C(\mathbf{z}^{-1}, m_{2i}) & 1 \end{bmatrix} \tag{7.31}$$

where $C(\mathbf{z}, m) = 1+mz_1+mz_2+z_1z_2$ and is slightly different from that defined in Example 7.12. We demonstrate the change of sizes of filter supports during lifting in Figure 7.4. As emphasized, two filters have their symmetry centers differed by one unit. The proposed approach is better than GMT in terms of filter performance, as shown below.

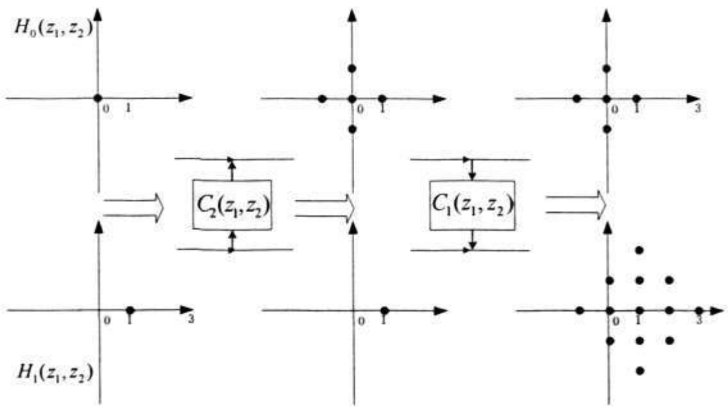
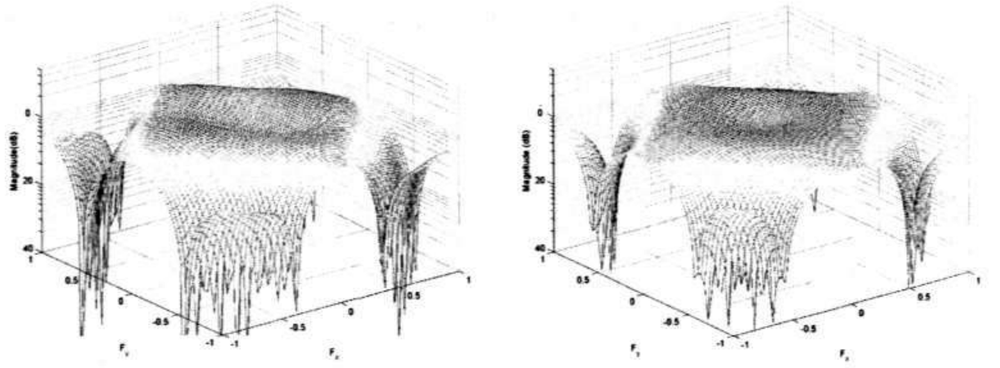


Figure 7.4: Support change during 2D lifting

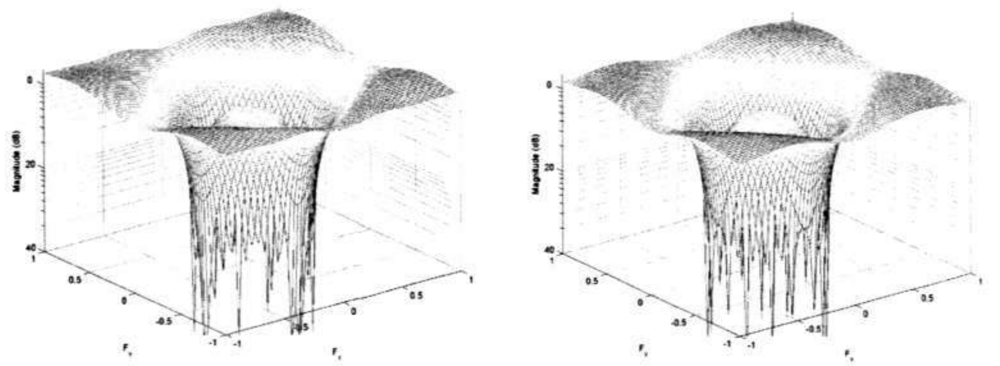
Example 7.13 *Two 2D type-B FBs of the same size are designed by GMT and the proposed construction respectively. High pass filters are of size* 15×15 *and low pass filters are* 17×17*. The magnitude frequency responses (MFR) of low pass filters are shown in Figure 7.5 and MFR of high pass filters are shown in Figure 7.6.*



(a) proposed 2x2 kernel based lifting

(b) generalized McClellan transform based lifting

Figure 7.5: Frequency response of low pass filters by different FB design methods



(a) proposed 2x2 kernel based lifting

(b) generalized McClellan transform based lifting

Figure 7.6: Frequency response of high pass filters by different FB design methods

Note that all these lifting based design use Eq. (7.31) or the structure of Figure 7.3. However, they differ in the kernel block. $C(\mathbf{z}) = 1 + z_1 + z_2 + z_1 z_2$ such as in Example 7.12 is termed McClellan transform based lifting. $C(\mathbf{z}) = 1 + mz_1 + mz_2 + z_1 z_2$ is termed GMT based lifting. $C(\mathbf{z}, m_i) = 1 + m_i z_1 + m_i z_2 + z_1 z_2$ is the proposed kernel.

The FBs are optimized under the same cost function that minimizes the difference of the MFR between the designed FB and the ideal filters. The diamond filter and its complement are chosen as the ideal low pass and the ideal high pass respectively. For comparison, the optimized values of the cost function for the proposed approach and GMT are 7246.75 and 7580.79 respectively.

The slightly better performance of the proposed method can be explained as follows. In fact, the proposed lifting construction can be regarded as an extended GMT. For GMT, every 1D zero phase polynomial (in polyphase domain) $s(z) = (1+z)/2$ is identically replaced by $C(\mathbf{z}) = 1 + mz_1 + mz_2 + z_1z_2$, while for the proposed method, every 1D zero phase polynomial $s_i(z)$ may have different transform $C(\mathbf{z}, m_i) = 1 + m_iz_1 + m_iz_2 + z_1z_2$. This increases the degree of freedom in the design. Therefore it is naturally better.

7.3.3 Lifting based refinement and comparison with existing approaches

It is possible that the complementary filter obtained by the PR completion approach introduced in section 7.2 is not satisfactory enough. In 1D, as shown in examples, we apply the lifting scheme to refine the obtained filter. This approach can also be followed in 2D case, as illustrated in the following equation.

$$\begin{bmatrix} H_0'(\mathbf{z}) \\ H_1(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} 1 & C(\mathbf{z}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} H_0(\mathbf{z}) \\ H_1(\mathbf{z}) \end{bmatrix} \tag{7.32}$$

Here $H_i(\mathbf{z})$ $(i = 0, 1)$ is the original filters, $H_0'(\mathbf{z})$ is the refined filter and $C(\mathbf{z})$ is a zero-phase filter centering in the origin.

By varying the polynomial for $C(\mathbf{z})$, we can obtain a class of $H_0(\mathbf{z})$'s that are all complementary to the given filter $H_1(\mathbf{z})$. This is easily verified by applying Theorem 7.13. This non-uniqueness of PR counterparts (or synthesis filter) is also observed in [85] and [23]. Respectively, they proposed different approaches based on Syzygy and extension of pseudo-inverse to refine the obtained filter. These methods are next shown equivalent to the lifting characterization of Theorem 7.13.

7.3.3.1 The equivalence of Syzygy method and lifting method

In [85], if $G(\mathbf{z})$ (with polyphase components $G_0(\mathbf{z})$ and $G_1(\mathbf{z})$) is a PR counterpart of a given $H(\mathbf{z})$ (with polyphase components $H_0(\mathbf{z})$ and $H_1(\mathbf{z})$) then

$$[H_0(\mathbf{z})\ H_1(\mathbf{z})] \cdot \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} = 1. \tag{7.33}$$

Note that in order to keep the notation consistent with that in Eq. (7.7), the indices of $G(\mathbf{z})$ are reversed and an extra minus sign is placed. Another PR counterpart $G'(\mathbf{z})$ of $H(\mathbf{z})$ can be characterized as,

$$\begin{bmatrix} G_1'(\mathbf{z}) \\ -G_0'(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} + u(\mathbf{z}) \begin{bmatrix} S_0(\mathbf{z}) \\ S_1(\mathbf{z}) \end{bmatrix} \tag{7.34}$$

where $[S_0(\mathbf{z})\ S_1(\mathbf{z})]^T$ is a Syzygy basis of $[H_0(\mathbf{z})\ H_1(\mathbf{z})]^T$ such that

$$[H_0(\mathbf{z})\ H_1(\mathbf{z})] \cdot \begin{bmatrix} S_0(\mathbf{z}) \\ S_1(\mathbf{z}) \end{bmatrix} = 0$$

and $u(\mathbf{z})$ is an arbitrary polynomial. Since $H_0(\mathbf{z})$ and $H_1(\mathbf{z})$ are coprime for the sake of PR, their Syzygy basis is just $S_0(\mathbf{z}) = H_1(\mathbf{z})$ and $S_1(\mathbf{z}) = -H_0(\mathbf{z})$. Substituting these into Eq. (7.34), we obtain

$$\begin{bmatrix} G_1'(\mathbf{z}) \\ -G_0'(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} G_1(\mathbf{z}) + u(\mathbf{z})H_1(\mathbf{z}) \\ -G_0(\mathbf{z}) - u(\mathbf{z})H_0(\mathbf{z}) \end{bmatrix} \tag{7.35}$$

Hence this characterization approach based on Syzygy method is obviously equivalent to the lifting characterization shown in Eq. (7.17).

7.3.3.2 The equivalence of pseudo-inverse extension and lifting method

If we assume similarly that $G(\mathbf{z})$ is a PR counterpart of a given $H(\mathbf{z})$, also satisfying Eq. (7.33), Theorem 3 in [23] shows that $G'(\mathbf{z})$ also satisfies Eq. (7.33) if and only if $G'(\mathbf{z})$ can be expressed as

$$\begin{bmatrix} G_1'(\mathbf{z}) \\ -G_0'(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} + \left(\mathbf{I} - \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} [H_0(\mathbf{z})\ H_1(\mathbf{z})]\right) \cdot \begin{bmatrix} S_0(\mathbf{z}) \\ S_1(\mathbf{z}) \end{bmatrix} \tag{7.36}$$

where $\mathbf{I}$ is an identity matrix and $S_0(\mathbf{z})$ as well as $S_1(\mathbf{z})$ are arbitrary polynomials (or FIR if necessary).

In fact, this is a widely used approach to parameterize all possible PR synthesis FBs or pseudo-inverse extension under oversampled situation, and is initiated in [108]. Next we show that in the circumstance of our special case mentioned above, it is equivalent to our lifting parametrization.

From Eq. (7.36) and Eq. (7.33),

$$\begin{aligned}\begin{bmatrix} G'_1(\mathbf{z}) \\ -G'_0(\mathbf{z}) \end{bmatrix} &= \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} + \begin{bmatrix} -H_1(\mathbf{z})G_0(\mathbf{z}) & -H_1(\mathbf{z})G_1(\mathbf{z}) \\ H_0(\mathbf{z})G_0(\mathbf{z}) & H_0(\mathbf{z})G_1(\mathbf{z}) \end{bmatrix} \cdot \begin{bmatrix} S_0(\mathbf{z}) \\ S_1(\mathbf{z}) \end{bmatrix} \\ &= \begin{bmatrix} G_1(\mathbf{z}) \\ -G_0(\mathbf{z}) \end{bmatrix} + \begin{bmatrix} -H_1(\mathbf{z})\left(G_0(\mathbf{z})S_0(\mathbf{z}) + G_1(\mathbf{z})S_1(\mathbf{z})\right) \\ H_0(\mathbf{z})\left(G_0(\mathbf{z})S_0(\mathbf{z}) + G_1(\mathbf{z})S_1(\mathbf{z})\right) \end{bmatrix} \end{aligned} \tag{7.37}$$

The equivalence may seem obvious if we choose $K(\mathbf{z}) = -\left(G_0(\mathbf{z})S_0(\mathbf{z}) + G_1(\mathbf{z})S_1(\mathbf{z})\right)$ in Eq. (7.17). Note that $G_0(\mathbf{z})$ and $G_1(\mathbf{z})$ are coprime due to PR, hence the ideal (as defined in [91]) generated by them $< G_0, G_1 >= k[z_1, z_2]$, which means by varying the coefficient polynomials $S_0(\mathbf{z})$ and $S_1(\mathbf{z})$, we can obtain any polynomial in $k[z_1, z_2]$. This agrees with the fact that there is no constraint on the lifting block $K(\mathbf{z})$ in Eq. (7.17).

As we show above, both the approaches to extend the special solution based on Syzygy and based on pseudo-inverse are actually equivalent to our lifting characterization (or parametrization). However, the equivalence is only provably valid in 2-D 2-channel cases. Whether this can be extended to 2-D multichannel or even M-D multichannel cases is of further interest.

7.4 Lattice based design of type-A nD LPPRFB

In 1D, we argue that lattice structure is more suitable to implement type-A LP-PRFB. The argument is also valid in nD. In [64], a lattice structure is proposed for the case of nD type-A LPPRFB as

$$\mathbf{E}(\mathbf{z}) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{R_0} \prod_{j=1}^{k} \prod_{i=1}^{M} \begin{bmatrix} 1 & 0 \\ 0 & z_i^{-1} \end{bmatrix} \mathbf{R}_{ji} \tag{7.38}$$

where $\mathbf{R} = \begin{bmatrix} 1 & a \\ a & 1 \end{bmatrix}$ with a design parameter a. By this cascade structure, a type-A LPPRFB can be designed. A design example is provided by the same authors in [109].

To prove the above parametrization to be complete is difficult and remains an open problem. However, for the simple and special case when the filter sizes are of 4×2, which is the most simple case of 2D, proof of the completeness is trivially obtained as follows. Assume the given filter $H_0(\mathbf{z})$ has polyphase components

$$H_{00}(\mathbf{z}) = a + bz_1^{-1} + (c + dz_1^{-1})z_2^{-1}$$

and

$$H_{01}(\mathbf{z}) = d + cz_1^{-1} + (b + az_1^{-1})z_2^{-1}.$$

According to Theorem 6.12, at least one weak zero is necessary for $H_{00}(\mathbf{z})$ and $H_{01}(\mathbf{z})$. Assume the common zero is in the form of $(k, 0)$. Substituting this common zero into the polyphase components, we obtain,

$$\begin{aligned} a + bk &= 0 \\ d + ck &= 0 \end{aligned} \tag{7.39}$$

Considering $k \neq 0$, this indicates $a \cdot c = d \cdot b$. With this condition, we have the following lattice reduction

$$\begin{bmatrix} 1 & \frac{-c}{b} \\ \frac{-c}{b} z_2 & z_2 \end{bmatrix} \begin{bmatrix} H_{00}(\mathbf{z}) \\ H_{01}(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} (a - \frac{dc}{b}) + (b - \frac{c^2}{b})z_1^{-1} \\ (c - \frac{c^2}{b}) + (a - \frac{dc}{b})z_1^{-1} \end{bmatrix} \tag{7.40}$$

To this step, the problem reduces to 1D factorization which has been completely solved as in Chapter 3.4. Using a similar technique as there, the other filter $H_1(\mathbf{z})$ can be also related to these lattice coefficients. Hence, Theorem 6.12 guarantees the lattice factorization for this special case. For the general case, we are still unable to validate completeness. We need more understanding of the relation between weak zeros and LPPRFB, which is a possible and interesting future extension.

7.5 Summary

In this chapter, we specifically investigate the applications of results, which are obtained in previous chapters, in the fields of nD 2-channel LPPRFB. As shown, the linear phase property brings similar constraints to nD FBs as in 1D case. The permissible LPPRFBs can also be classified as two types (type-A and type-B) according to the different symmetry natures. The detailed requirements on lengths and symmetry centers are derived for the special case of 2-D quincunx downsampled FBs. We demonstrate a class of approaches using symmetry extension for problems of PR completion. These approaches allow us to find a PR complementary filter for a given filter based on Groebner basis, and the LP property is further incorporated by symmetry extension. The extension of these approaches are discussed and some interesting examples are presented. We also establish the relation between lifting structure and an nD type-B LPPRFB based on a theorem and a conjecture. With the results obtained previously, the conjecture can only be proved for some special cases. For nD type-A LPPRFBs, we refer to an existing design approach and prove its completeness for some special case.

Chapter 8

Conclusions and recommendations

8.1 Conclusions

In this thesis, we have studied perfect reconstruction filter banks in both 2-channel and multiple channel cases, as well as in both 1D and nD cases, especially using a lifting-based approach. Perfect reconstruction plays a key role in the theory and applications of filter banks. The purpose of this thesis is to investigate the characterization, factorization and design of these PRFBs. Like most of works in FB, we concentrate on FIR FBs that are widely used for its ease in implementation.

In 1D, we first focussed on 2-channel case. To relate the lifting scheme with PRFBs, we review the factorization that is based on Euclidean algorithm for two polynomials. The non-unique choice during the polynomials division motivates our work. We classify 2-channel LPPRFBs as two types according to their different natures and apply different lifting steps to factorize them, such that the LP property can be structurally incorporated. Specifically, for a class of singular LPPRFBs, a lifting-like structure is proposed. The factorization has been demonstrated by some examples. The results shows that the proposed lifting factorization is useful for implementing FBs with reduced word-length. Further, we use the obtained structure to design LPPRFBs that can meet the desired specifications. We have attempted to investigate the essence of lifting scheme. This leads us to some insightful and interesting results.

We further extended the idea of lifting scheme to M-channel case. The lifting based building blocks are investigated. We showed that only block based lifting matrix or its permutation can have the so-called sign changing inverse. We study how to extend the Euclidean algorithm to multiple channel and apply it to factorize a M-channel PRFB. On the design side, we present results from both PRFB and those with linear phase.

Before we moved to the scope of nD PRFB, the relevant literature is reviewed and categorized. Some basic relationship and concepts, including transform, symmetry, nonseparability and sampling in nD, are studied. We further study the enumeration and parametrization of downsampling patterns for a given downsampling ratio in a 2D PRFB system. The results give us an explicit way to enumerate and parameterize the downsampling matrices exhaustively. We further extend the result to nonseparable downsampling. How to generate ideal filters for a given downsampling matrix, such that they can form an alias-free FB, is also investigated.

The problems of nD PRFB are further investigated from a viewpoint of algebraic geometry. The relations between various zeros and various polynomial invertibility are studied and presented. From the results, we can not only disprove an existing approach from others' work to construct 2D 2-channel LPPRFB, but we can apply these as well to factorize a given 2D type-B LPPRFB together with a conjecture. The result can be also applied to check whether PR property can be retained after a 1D-to-nD transform, such as McClellan transform. These approaches originated from algebraic theory are insightful and not found in literature.

As essential tools to solve the above problems, resultants and Groebner basis are studied. We propose an approach to solve the PR completion problem for a class of LPPRFB based on Groebner basis. The relation between this class of LPPRFB and lattice structure is also investigated. There should be many other ways to utilize these powerful tools than the ways used and mentioned in this work. For example, GB can be considered in the design of M-channel LPPRFB. But this really needs some efforts and inspiration.

8.2 Recommendations for further research

During our above-presented study of PRFBs, we think the following directions may be interesting for further investigation.

- One future work is to further investigate the relation between various zeros and various properties in nD system. According to our study, a weak zero is necessary for a nD 2-channel LPPRFB. This result is not strong enough to justify the lifting based factorization. That is why we further propose a conjecture to tackle this problem. According to our observation, an LPPRFB is very likely to need more weak zeros. However, to explore the correctness of this argument needs an in-depth investigation in algebraic geometry. The similar study can also be conducted between various zeros and paraunitary property and so on. Different from matrix theory based approaches, approaches from algebraic geometry always discover the essence of the problems and give us more insight. It is surely worth further researching. The investigation can be even extended to M-channel case. Zeros may have more variations then.

- It might be noted that filters used in this work are mostly FIR filters. However, IIR filter is indeed preferred in some circumstance for its effectiveness of computation. Sometimes IIR filter may even be the only practical choice for a desired frequency response. Furthermore, for IIR problems related to PRFB, intuitively we would have a larger space to search, both in 1D and nD. once a suitable ring is chosen, the tools such as GB and resultants can be used directly in IIR. For LS, lifting steps now have more choices. This surely brings us more freedom of design. Pioneering works can be found in [81, 110] and a good review on nD IIR problems can be found in [86]. When IIR is considered, poles consequently come into the scene. Similar to the results we obtained for zeros, there must be some poles to avoid and some to incorporate. This also need further study.

- In this work, only critically sampled systems are considered. However, it has been proved by many convincing experiments that an oversampled system actually has more design freedom and better noise immunity than critically sampled FBs. In fact, an oversampled system would resample input signals of certain time slot (or location) more than once. To some extent, it is why the system is so named. When the signal has more than one dimension, such redundant resampling can be executed more sophisticatedly. We may even combine a critically sampled system with an oversampled one by appropriately allocating the signals into dimensions. After all, to extend the oversampled system to the multivariate situation is interesting.

- Another possible future work is to investigate the PRFB from a viewpoint of group theory, just as done in [111] and [112]. This is different from the traditional derivation from the viewpoint of signal processing. Many results derived in this novel way agree with what has been reached before, while some reveal the limits of the lattice and lifting structures. Some such results indeed fall within our interest in this work.

Author's publication

1) Lei Zhang, Anamitra Makur, Enumeration of downsampling lattices in two-dimensional multirate systems, *accepted by IEEE Trans. on Signal Processing*, 2007.

2) Lei Zhang, Anamitra Makur, Two-dimensional antisymmetric linear phase filter bank construction using symmetric completion, p57-60, Vol. 54, Issue 1, *IEEE Trans. on Circuits and Systems II*, Jan. 2007.

3) Lei Zhang, Anamitra Makur, Zhiming Xu, On lifting factorization for 2-D LPPRFB, p2145-2148, *Proceedings ICIP*, Oct. 2006.

4) Lei Zhang, Anamitra Makur, Enumeration and parametrization of distinct downsampling patterns in two-dimensional multirate systems, p373-376, Vol. 3, *Proceedings ICASSP*, May 2006, (awarded the Student Paper Award).

5) Lei Zhang, Anamitra Makur, Comments on "Multi-dimensional Filter banks and Wavelets–A System Theoretic Perspective", p699-704, Vol. 343, Issue 7, *Journal of the Franklin Institute*, Nov. 2006.

6) Lei Zhang, Anamitra Makur, Ce Zhu, Design of 2-channel linear phase filter bank: a lifting approach, p4301 - 4304, Vol. 5, *Proceedings ISCAS*, May 2005.

7) Lei Zhang, Anamitra Makur, Structurally linear phase factorization of 2-channel filter banks based on lifting, p609-612, Vol. 4, *Proceedings ICASSP*, Mar. 2005.

8) Lei Zhang,Anamitra Makur, Multidimensional perfect reconstruction filter banks–an approach of algebraic geometry, *submitted to Multidimensional Systems and Signal Processing.*

References

[1] M. J. T. Smith; S. L. Eddlins. Analysis/synthesis techniques for subband image coding. *IEEE Trans. on Acoust., Speech, and Signal Processing*, 38:1446–1456, Aug. 1990.

[2] P.P. Vaidyanathan. Filter banks in digital communications. *IEEE Circuits System Magazine*, 1:4–25, 2nd Quarter 2001.

[3] P.P.Vaidyananthan. *Multirate Systems and Filter Banks.* Englewood Cliffs, NJ: Prentice Hall, 1993.

[4] J.Kovacevic M.Vetterli. *Wavelets and Subband Coding.* Englewood Cliffs, NJ: Prentice Hall, 1995.

[5] T.Q.Nguyen G.Strang. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 1996.

[6] S.Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1998.

[7] A.W.M. van den Enden F.A.M.L. Bruekers. New networks for perfect inversion and perfect reconstruction. *IEEE Journal on Selected Areas in Communications*, 10(1):129 – 137, Jan. 1992.

[8] T.A.C.M. Kalker, I.A. Shah. On Ladder Structures and Linear Phase Conditions for Multidimensional Bi-Orthogonal Filter Banks. *Research report, Philips research laboratories, Eindhoven*, 1993.

[9] T.A.C.M. Kalker, I. Shah. Ladder Structures for multidimensional linear phase perfect reconstruction filter banks and wavelets. *In Proceedings of the SPIE Conference on Visual Communications and Image Processing (Boston)*, pages 12–20, 1992.

[10] W. Sweldens. The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions. *Wavelet Applications in Signal and Image Processing III*, pages 68–79, 1995.

[11] W. Sweldens I. Daubechies. Factoring wavelet transforms into lifting steps. *J.Fourier Anal. Appl.*, 4(3):247–269, May 1998.

[12] T.D. Tran, R.L. de Queiroz, T.Q. Nguyen. Linear-phase perfect reconstruction filter banks: lattice structure, design and application in image coding. *IEEE. Trans. on Signal processing*, 48(1):133–146, Jan. 2000.

[13] T.D. Tran, T.Q. Nguyen. On M-channel linear phase FIR filter banks and application in image compression. *IEEE. Trans. on Signal processing*, 45(9):2175–2187, Sep. 1997.

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

[15] K.S.C. Pun, T.Q. Nguyen. A novel and efficient design of multidimensional PR two-channel filter banks with hourglass-shaped passband support. *IEEE Signal Processing Letters*, 11(3):345– 348, Mar. 2001.

[16] T. Chen, P.P Vaidyanathan. The role of integer matrices in multidimensional multirate systems. *IEEE Trans. on Signal Processing*, 41(3):1035–1047, Mar. 1993.

[17] T. Chen, P.P Vaidyanathan. Commutativity of D-dimensional decimation and expansion matrices, and application to rational decimation systems. *Proceedings ICASSP*, 4:637–640, Mar. 1992.

[18] T. Chen, P.P Vaidyanathan. Least common right/left multiples of integer matrices and applications to multidimensional multirate systems. *Proceedings ISCAS*, 2:935–938, May 1992.

[19] B.L. Evans, R.H. Bamberger, J.H. McClellan. Rules for Multidimensional Multirate structures. *IEEE Trans. on Signal Processing*, 42(4):762–770, Apr. 1994.

[20] A.A.C.M. Kalker. Commutativity of Up/Down Sampling. *Electronics Letters*, 28(6):567–569, Mar. 1992.

[21] J. Kovacevic, M. Vetterli. The commutativity of up/downsampling in two dimensions. *IEEE Trans. on Information Theory*, 37(3(II)):695–698, May 1991.

[22] R.A. Gopinath, C.S. Burrus. On Upsampling, Downsampling and Rational Sampling Rate Filter Banks. *IEEE Trans. on Signal Processing*, 42(4):812–824, Apr. 1994.

[23] J. Zhou, M. N. Do. Multidimensional Multichannel FIR Deconvolution Using Groebner Bases. *IEEE. Trans. on Image Processing*, 15(10):2998–3007, Oct. 2006.

[24] N.K. Bose. *Multidimensional Systems Theory and Applications.* Kluwer Academic Publishers, 2003.

[25] J.S. Lim. *Two-Dimensional Signal and Image Processing.* Prentice-Hall Signal Processing Series, 1990.

[26] T. O'Rourke, R. Stevenson. Human visual system based wavelet decomposition for image compression. *J. Vis. Commun. Image Represent.*, 6:109–121, Jun. 1995.

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

[28] D. Pinchon, S. Siohan. Analysis, design, and implementation of two-channel linear phase filter banks: a new approach. *IEEE Trans. on Signal Processing*, 46(7):1814–1825, Jul. 1998.

[29] C. K. Goh, Y. C. Lim. Novel Approach for the Design of Two Channel Perfect Reconstruction Linear Phase FIR Filter Banks. *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, 45(8):1141–1146, Aug. 1998.

[30] H. Kiya, M. Yae, M. Iwahashi. A linear phase two-channel filter bank allowing perfect reconstruction [image coding]. *Proceedings ISCAS*, 2:951–954, May 1992.

[31] M. Li, C.-W. Kok. Linear phase filter bank design using LMI-based H_∞ optimization. *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, 50(3):143–149, Mar. 2003.

[32] T. Nagai, M. Ikehara. A new approach to the design of linear-phase biorthogonal filter banks and wavelets. *Conference Record of the Thirtieth Asilomar Conference on Signals, Systems and Computers*, 2:978 – 982, Nov. 1996.

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

[34] A.N. Willson P. Saghizadeh. A new approach to the design of critically sampled M-channel uniform-band perfect-reconstruction linear-phase FIR filter banks. *IEEE Trans. on Signal Processing*, 46(6):1544–1557, Jun. 1998.

[35] V.V. Prasolov. *Polynomials*. Springer-Verlag Berlin Heidelberg Inc, 2004.

[36] R. Bitmead, S.Y. Kung, B. Anderson, T. Kailath. Greatest common divisor via generalized Sylvester and Bezout matrices. *IEEE Trans. on Automatic Control*, 23(6):1043 – 1047, Dec. 1978.

[37] M. Vetterli, C. Herley. Wavelets and filter banks: theory and design. *IEEE Trans. on Signal Processing*, 40(9):2207–2232, Sep. 1992.

[38] G. Dillen, B. Georis, J.D. Legat, O. Cantineau. Combined line-based architecture for the 5-3 and 9-7 wavelet transform of JPEG2000. *IEEE Trans. on Circuits and Systems for Video Technology*, 13(9):944–950, Sep. 2003.

[39] M.D. Adams, R.K. Ward. Symmetric-extension-compatible reversible integer-to-integer wavelet transforms. *IEEE Trans. on Signal Processing*, 51(10):2624–2636, Oct. 2003.

[40] G. Zhong, L. Cheng, H. Chen. A simple 9/7-tap wavelet filter based on lifting scheme. *Proceedings ICIP*, 2:249 – 252, Oct. 2001.

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

[42] A. Cohen, I. Daubechies, J. C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Commun. Pure Appl. Math.*, 45:485–560, 1992.

[43] J. Villasenor, B. Belzer, J. Liao. Wavelet Filter Evaluation for Image Compression. *IEEE Trans. on Image Processing*, 2:1053–1060, Aug. 1995.

[44] R. Heasman G. Davis, J. Danskin. WAVELET IMAGE COMPRESSION CONSTRUCTION KIT 0.3. Technical report, 1997. http://www.geoffdavis.net/dartmouth/wavelet/wavelet.html.

[45] D. B. H. Tay. Rationalizing the Coefficients of Popular Biorthogonal Wavelet Filters. *IEEE Trans. on Circuits and Systems for Video Technology*, 10(6):998 – 1005, Sep. 2000.

[46] D.B.H. Tay. A class of lifting based integer wavelet transform. *Proceedings ICIP*, 1:602–65, Oct. 2001.

[47] A. Yaqub, H.G. Moore, editor. *A First Course in Linear Algebra: With Applications*. Elsevier, 1998.

[48] G. Schuller, W. Sweldens. Filter bank design using nilpotent matrices. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 51–54, Oct. 1999.

[49] Lu Gan and Kai-Kuang Ma. On Minimal Lattice Factorization of Symmetric-Antisymmetric Multifilter Banks. *IEEE Trans. on Signal Processing*, 53(2):606–621, Feb. 2005.

[50] Lu Gan and Kai-Kuang Ma. Time-domain Oversampled Lapped Transforms: Theory, Structure and Application in Image Coding. *IEEE Trans. on Signal Processing*, 52(10):2762 2775, Oct. 2004.

[51] Lu Gan and Kai-Kuang Ma. On Simplified Order-one Factorizations of Paraunitary Filterbanks. *IEEE Trans. on Signal Processing*, 52(3):674–686, Mar. 2004.

[52] Lu Gan and Kai-Kuang Ma. A Simplified Lattice Factorization for Linear-phase Paraunitary Filter Banks with Pairwise Mirror Image Frequency Responses. *IEEE Trans. on Circuits and Systems II*, 51(1):3–7, Jan. 2004.

[53] Lu Gan and Kai-Kuang Ma. Oversampled Linear-phase Perfect Reconstruction Filterbanks: Theory, Lattice Structure and Parameterization. *IEEE Trans. on Signal Processing*, 5(3):744–759, Mar. 2003.

[54] Lu Gan, Kai-Kuang Ma, Truong Q. Nguyen, Trac D. Tran, Ricardo L. de Queiroz. On the Completeness of the Lattice Factorization for Linear-phase Perfect Reconstruction Filter Banks. *IEEE Signal Processing Letters*, 9(4):133–136, Apr. 2002.

[55] Lu Gan and Kai-Kuang Ma. A Simplified Lattice Factorization for Linear-Phase Perfect Reconstruction Filter Bank. *IEEE Signal Processing Letters*, 8(7):207–209, Jul. 2001.

[56] K.S. Amaratunga Y.-J. Chen. M-channel lifting factorization of perfect reconstruction filter banks and reversible m-band wavelet transforms. *IEEE Trans. on Circuits and Systems II*, 50(12):963 – 976, Dec. 2003.

[57] K. Galkowski, J. Wood (editors). *Multidimensional Signals, Circuits and Systems*. Taylor and Francis, 2001.

[58] H.H. Rosenbrock. *State-space and multivariable theory*. John Wiley and Sons Inc., New York, 1970.

[59] W. A. Wolovich. The determination of state-space representations for linear multivariable systems. *Automatica*, 9(1):97–106, Jan. 1973.

[60] K.S. Amaratunga Y.-J. Chen, S. Oraintara. M-channel lifting-based design of paraunitary and biorthogonal filter banks with structural regularity. *Proceedings ISCAS*, pages IV221–IV224, May 2003.

[61] T. Lin, P. Hao, S. Xu. Factoring M-band wavelet transforms into reversible integer mappings and lifting steps. *Proceedings ICASSP*, 4:629–632, Mar. 2005.

[62] T. Lin, P. Hao, S. Xu. Matrix Factorization for Reversible Integer Implementation of Orthonormal M-Band Wavelet Transforms. *Signal Processing, Elsevier*, 86(8):2085–2093, Aug. 2006.

[63] G. Karlsson, M. Vetterli. Theory of Two-dimensional Multirate Filter Banks. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 38(6):925–937, Jun. 1990.

[64] J. Kovacevic, M. Vetterli. Nonseparable two- and three-dimensional wavelets. *IEEE Trans. on Signal Processing*, 43(5):1269–1273, May 1995.

[65] Y.P. Lin, P.P. Vaidyanathan. Theory and design of two-dimensional filter banks: a review. *Multidimensional Systems and Signal Processing*, 7(3-4):263–330, July/Oct. 1996.

[66] E. Viscito, J.P. Allebach. The analysis and design of multidimensional FIR perfectreconstruction filter banks for arbitrary sampling lattices. *IEEE Trans. on Circuits and Systems*, 38(1):29–41, Jan. 1991.

[67] T. Chen, P.P Vaidyanathan. Recent Developments in Multidimensional Multirate Systems. *IEEE Trans. on Circuits and Systems for Video Technology*, 3(2):116–137, Apr. 1993.

[68] T. Chen, P.P. Vaidyanathan. Multidimensional multirate filters derived from one dimensional filters. *Electronics Letters*, 27(3):225–228, Jan. 1991.

[69] D.B.H. Tay, N.G. Kingsbury. Flexible Design of Multidimensional Perfect Reconstruction FIR 2-Band Filters using Transformations of Variables. *IEEE Trans. on Image Processing*, 2(4):466–480, Oct. 1993.

[70] R.M. Mersereau, W.F.G. Mecklenbrauker, T.F. Quatieri, Jr. McClellan transformations for two-dimensional digital filtering, I: Design. *IEEE Trans. on Circuits and Systems*, 23:405–414, Jul. 1976.

[71] D. Wei, B.L. Evans, A.C. Bovik. Loss of Perfect Reconstruction in Multidimensional Filterbanks and Wavelets Designed via Extended McClellan Transformations. *IEEE Singal Processing Letter*, 4(10):295–297, Oct. 1997.

[72] K. Kurosawa. On McClellan transform and 2-D QMF banks. *Proceedings ISCAS*, 2:505–508, May 1994.

[73] M.N. Do, M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *IEEE Trans. on Image Processing*, 14(12):2091–2106, Dec. 2005.

[74] R.H. Bamberger, M.J.T. Smith. A filter bank for the directional decomposition of images: Theory and design. *IEEE Trans. on Signal Processing*, 40(4):882–893, Apr. 1992.

[75] Y. Lu, M.N. Do. The finer directional wavelet transform. *Proceedings ICASSP*, 4:573–576, Mar. 2005.

[76] A.L. da Cunha, M.N. Do. Biorthogonal filter banks with directional vanishing moments. *Proceedings ICASSP*, 4:553–556, Mar. 2005.

[77] T.T. Nguyen, S. Oraintara. A Class of Multiresolution Directional Filter Bank. *IEEE Trans. on Signal Processing*, 55(3):949–961, Mar. 2007.

[78] S. Basu. New results in multidimensional linear phase filter bank design. *Proceedings ISCAS*, 5:1–4, 1999.

[79] S. Basu. Multi-dimensional filter banks and wavelets - a system theoretic perspective. *Journal of the Franklin Institute*, 335B(8):1367–1409, Nov. 1998.

[80] C. Charoenlarpnopparut, N.K. Bose. Multidimensional FIR filter bank design using Gröbner bases. *IEEE Trans. on Circuits and Systems II: Express Briefs*, 46(12):1475–1486, Dec. 1999.

[81] C. Charoenlarpnopparut, N.K. Bose. Gröbner Bases for Problem Solving in Multidimensional Systems. *Multidimensional Systems and Signal Processing*, 12(3-4):365–376, Jul./Oct. 2001.

[82] T. Kalker, H. Park, M. Vetterli. Groebner basis techniques in multidimensional multirate systems. *Proceedings ICASSP*, 4:2121–2124, May 1995.

[83] H. Park. Parametrized Family of 2-D Non-factorable FIR Lossless Systems and Gröbner Bases. *Multidimensional Systems and Signal Processing*, 12(3-4):345–364, Jul./Oct. 2001.

[84] H. Park, T. Kalker, M. Vetterli. Gröbner Bases and Multidimensional FIR Multirate Systems. *Multidimensional Systems and Signal Processing*, 8(1-2):11–30, Jan. 1997.

[85] H. Park. Optimal design of synthesis filters in multidimensional perfectreconstruction FIR filter banks using Grobner bases. *IEEE Trans. on Circuits and Systems I: Regular Papers*, 49(6):843–851, Jun. 2002.

[86] Z Lin, L. Xu, Q. Wu. Appliation of Gröbner bases to signals and image processing: a survey. *Linear Algebra and its Application*, 391:169–202, Nov. 2004.

[87] J.G. Proakis. *Digital Communications.* McGraw-Hill, 4th edition, 2001.

[88] H.J.S. Smith. On System Of Linear Indeterminate Equations and Conguences. *Philos. Trans. Royal Soc. London*, 151:293–326, 1861.

[89] B. Buchberger. An algorithm for finding the bases elements of the residue class ring modulo a zero dimensional polynomial ideal. *Ph.D thesis, Univer. of Innsbruck*, 1965.

[90] H. Hironaka. Resolution of singularities of an algebraic variety over a field of characteristic zero: I, II. *Annals of Mathematics*, 79(2), 1964.

[91] D. Cox, J. Little, D. O'Shea. *Ideals, Varieties, and Algorithms.* Springer-Verlag New York Inc, 2nd edition, 1996.

[92] W. W Adams, P. Loustaunau. *An introduction to Gröbner bases.* Providence, R.I. : American Mathematical Society, 1996.

[93] B. Buchberger. Gröbner Bases and Systems Theory. *Multidimensional Systems and Signal Processing*, 12(3/4):223–251, Oct. 1998.

[94] D. Youla, G. Gnavi. Notes on n-Dimensional System Theory. *IEEE Trans. on Circuits and Systems*, 26(2):105–111, Feb. 1979.

[95] N.K. Bose. *Applied multidimensional systems theory.* New York : Van Nostrand Reinhold, 1982.

[96] J. P. Guiver, N. K. Bose. Polynomial matrix primitive factorization over arbitrary coefficient field and related results. *IEEE Trans. Circuits and System*, 29:649–657, Oct. 1982.

[97] S. Basu. On multidimensional linear phase perfect reconstruction filter banks. *Proceedings ISCAS*, 2:145–148, 1994.

[98] Z.P. Lin, L. Xu, Q.H. Wu. Design of multidimensional filter banks using Gröbner bases: a survey. *Proceedings ISCAS*, 3:277–280, May 2004.

[99] H. Schönemann G.-M. Greuel, G. Pfister. SINGULAR 3.0. A Computer Algebra System for Polynomial Computations, Centre for Computer Algebra, University of Kaiserslautern, 2005. `http://www.singular.uni-kl.de`.

[100] N.K. Bose, C. Charoenlarpnopparut. Multivariate Matrix Factorization: New Results. In *Proceedings, Mathematical Theory of Networks and Systems Symposium (editors: A. Beghi, L. Finesso, G. Poeco)*, pages 97–100, Jul. 1998.

[101] A. Logar, B. Sturmfels. Algorithms for the Quillen-Suslin theorem. *Journal of Algebra*, 145:231–239, Jan. 1992.

[102] H-J. Park. A Computational Theory of Laurent Polynomial Rings and Multidimensional FIR Systems. Technical Report UCB/ERL M95/39, EECS Department, University of California, Berkeley, 1995.

[103] R. Rajagopal, L. Potter. Multi-channel multi-variate equalizer design. *Multidimensional Systems and Signal Processing* , 14:105–118, Jan.-Jul. 2003.

[104] J. Zhou, M. N. Do. Multidimensional oversampled filter banks. *SPIE, Wavelets XI*, 5914(1):591424.1–591424.12, Sep. 2005.

[105] J. L. Rabinowitsch. Zum Hilbertschen Nullstellensatz. *Mathematische Annalen*, 102(1):520, Dec. 1930.

[106] L. Xu, O. Saito, K. Abe. Output feedback stabilizability and stabilization algorithms for 2D systems. *Multidimensional Systems and Signal Processing*, 5(1):41–60, Jan. 1994.

[107] J.H. McClellan. The design of two-dimensional digital filters by transformation. *Proceedings 7th Annual Princeton Conference on Information Sciences and Systems, Princeton, N.J.*, pages 247–251, Mar. 1973.

[108] H. Bolcskei, F. Hlawatsch, H.G. Feichtinger. Frame-theoretic analysis of oversampled filter banks. *IEEE Trans. on Signal Processing*, 46(12):3256–3268, Dec. 1998.

[109] M. Vetterli, J. Kovacevic. Perfect reconstrution filter banks for HDTV representation and coding. *Singnal processing: Image Communication*, 2:349–363, Oct. 1990.

[110] W. Wang, T. Yoshikawa, Y. Takei. Design of IIR Orthogonal Wavelet Filter Banks Using Lifting Scheme. *IEEE Trans. on Signal Processing*, 54(7):2616–2624, Jul. 2006.

[111] A. Klappenecker. On some limits of lattice and lifting structures. *Wavelet Applications in Signal and Image Processing VII*, pages 674–681, Oct. 1999.

[112] A. Klappenecker, M. Holschneider, K. Flornes. Two-channel Perfect Reconstruction FIR Filter Banks over Commutative Rings. *Applied and Computational Harmonic Analysis*, 8:113–121, Jan. 2000.

Appendix A

Proof on symmetry centers

Assume that z^{c_0} and z^{c_1} are the symmetry centers for $H_0(z)$ and $H_1(z)$ respectively. Take into account four facts:

Fact A.4 $H(z)_{[S,c_0]} \cdot G(z)_{[S,c_1]} = F(z)_{[S,c_0+c_1]}$ *(where* $H(z)_{[S,c_0]}$ *suggests* $H(z)$ *is a symmetric filter with a center in* z^{c_0}*)*

Fact A.5 $H(z)_{[A,c_0]} \cdot G(z)_{[A,c_1]} = F(z)_{[S,c_0+c_1]}$ *(where subscript* $_A$ *denotes antisymmetric filter)*

Fact A.6 $H(z)_{[A,c_0]} \Rightarrow H(-z)_{[S,c_0]}$ *and vice versa for even length;* $H(z)_{[S,c_0]} \Rightarrow H(-z)_{[S,c_0]}$ *for odd length*

Fact A.7 $H(z)_{[S,c_0]} + G(z)_{[S,c_0]} = F(z)_{[S,c_0]}$.

From these facts we can conclude that the left part of Eq. (3.12) is a symmetric polynomial with a center in $z^{c_0+c_1}$. This holds for both type-A and type-B.

According to Eq. (3.12) and Eq. (3.14), it is necessary to require $c_0+c_1 = 2n_0-1$, where n_0 is an integer. For type-A, c_0 and c_1 are not integers but integers plus $\frac{1}{2}$. If we denote them as $c'_0 + \frac{1}{2} = c_0$ and $c'_1 + \frac{1}{2} = c_1$, we will find that $c'_0 + c'_1 = 2r$. It is equivalent to saying the difference between c'_0 and c'_1 is an even number since they are both integers. It is the same with c_0 and c_1. Similarly, for type-B, c_0 and c_1 are integers and have a sum of odd number. This is equivalent to saying the difference between c_0 and c_1 is an odd number. ∎

Appendix B

PR condition for nD 2-channel LPPRFB

Assume two filters in an nD 2-channel LPPR analysis bank are denoted by $H_0(\mathbf{z})$ and $H_1(\mathbf{z})$. Similar to 1D notation, their polyphase components are expressed as $E_{i,j}(\mathbf{z})$ $(i, j = 0, 1)$ respectively. The relation between the filters and their polyphase components can be expressed as:

$$H_i(\mathbf{z}) = \mathbf{z}^{\mathbf{k}_0} E_{i,0}(\mathbf{z}^{\mathbf{M}}) + \mathbf{z}^{\mathbf{k}_1} E_{i,1}(\mathbf{z}^{\mathbf{M}}) \tag{B.1}$$

Here $\mathbf{M}$ is the downsampling matrix, $\mathbf{k}_0$ and $\mathbf{k}_1$ are the only two integer points within $FPD(\mathbf{M})$ since we are considering critically downsampled case with downsampling ratio $m = \det(\mathbf{M}) = 2$.

In fact, $\mathbf{k}_i$ can also be chosen as any other point of the same phase. This choice will lead to a shift of $E_{i,j}(\mathbf{z}^{\mathbf{M}})$. For example,

$$\begin{aligned} H(z_1, z_2) = &\ 1 + 2z_1 + 3z_2 + 4z_1 z_2 \\ = &\ E_0(z_1 z_2, z_1 z_2^{-1}) + z_2 E_1(z_1 z_2, z_1 z_2^{-1}) \\ = &\ z_1 z_2 E_0'(z_1 z_2, z_1 z_2^{-1}) + z_1 E_1'(z_1 z_2, z_1 z_2^{-1}) \end{aligned} \tag{B.2}$$

where $E_0(z_1, z_2) = 1 + z_1$, $E_1(z_1, z_2) = 1 + z_2$ and $E_0'(z_1, z_2) = z_1(1 + z_1)$, $E_1(z_1, z_2) = 1 + z_2^{-1}$. The essential reason behind this is the nonunique choice of delay chain system of Figure 2.6 in nD.

Therefore, we can always choose $\mathbf{k}_0$ as the origin (in fact, according to the definition of $FPD(\mathbf{M})$ in Eq. (5.2), it should be) and choose $\mathbf{k}_1$ as one of the origin's closest neighboring points, which is one unit away, e.g. $z_1, z_2, z_1^{-1}, z_2^{-1}$ etc. Now we want to show that at least one of these closest neighboring points is of the other phase. This can be proved by contradiction. Assume all the aforementioned points are of the same phase, i.e. there exists integer vector $\mathbf{n}$ s.t.

$$\pm\mathbf{e}_i = \mathbf{M} \cdot \mathbf{n} \Longleftrightarrow \mathbf{n} = \mathbf{M}^{-1} \cdot \pm\mathbf{e}_i \tag{B.3}$$

where $i \in [1, M]$ and $\mathbf{e}_i$ is a zero-vector with only the i^{th} element being 1, i.e. $\mathbf{e}_i = [0, \cdots, 1, \cdots, 0]^T$. It is known that $\mathbf{M}^{-1} = \frac{\text{adj}(\mathbf{M})}{det(\mathbf{M})}$. To make $\mathbf{n}$ an integer vector, Eq. (B.3) requires all the elements of $\mathbf{M}$ are multiples of $det(\mathbf{M})$, i.e. 2. This is obviously impossible if all the elements of $\mathbf{M}$ are integer. Therefore, the assumption that all the aforementioned points are of the same phase is not true. Hence, for any downsampling matrix $\mathbf{M}$ with downsampling ratio 2, we can always find an integer k such that the filter can be decomposed as

$$H_i(\mathbf{z}) = E_{i,0}(\mathbf{z}^{\mathbf{M}}) + \mathbf{z}^{\mathbf{e}_k} E_{i,1}(\mathbf{z}^{\mathbf{M}}) \tag{B.4}$$

According to its definition, $(-\mathbf{z})^{\mathbf{e}_k} = -\mathbf{z}^{\mathbf{e}_k}$. Therefore, the relation between filters and their polyphase components can be established as,

$$\begin{aligned} E_{i,0}(\mathbf{z}^{\mathbf{M}}) &= \frac{H_i(\mathbf{z})+H_i(-\mathbf{z})}{2} \\ E_{i,1}(\mathbf{z}^{\mathbf{M}}) &= \frac{H_i(\mathbf{z})-H_i(-\mathbf{z})}{2\mathbf{z}^{\mathbf{e}_k}} \end{aligned} \tag{B.5}$$

Substituting this into the PR condition for a delay chain system,

$$E_{00}(\mathbf{z})E_{11}(\mathbf{z}) - E_{10}(\mathbf{z})E_{01}(\mathbf{z}) = \mathbf{z}^{\mathbf{c}} \tag{B.6}$$

we can obtain the PR condition in Eq. (7.5).