Wavelet Footprints and Frames for Signal Processing and Communication

Pier Luigi Dragotti

April 26, 2002

"From the moment I picked your book up until I laid it down I was convulsed with laughter. Someday I intend reading it. "

Groucho Marx

Abstract

Linear transforms and expansions are fundamental mathematical tools of signal processing. In particular, the wavelet transform has played an important role in several signal processing tasks, compression being a prime example. A signal can be represented in a basis or a frame. Frames, which are an extension of bases to overcomplete sets of vectors, are sometimes preferred to bases because of their greater design freedom and recently they have had an important impact in signal processing.

This thesis focuses on the design of new bases and frames for signal processing and communications. The first contribution of the thesis is the exploration of the use of oversampled filter banks, which represent a possible way to implement frames, to robust communications. Normally, transforms are used as a pure source coding method and a reliable communication is obtained with a channel coder which follows the source coder. In this case, we use frames to achieve efficient source compression and robustness to trasmission errors at the same time.

In the context of pure source transform coding, we investigate the performance of the wavelet transform and study the dependency of the wavelet coefficients across scales. This analysis leads to the design of a new expansion which provides an efficient representation of piecewise smooth signals. We call *footprints* the elements of this expansion. The main property of footprints is that they efficiently characterize the singular structures of the signal, which usually carry the main information. We show that algorithms based on footprints outperform wavelet methods in different applications such as denoising, compression and deconvolution.

Finally, we study a particular source coding technique called multiple description coding. This technique is used for data transmission over unreliable networks. In multiple description coding, the coder generates several different descriptions of the same signal and the decoder can produce a useful reconstruction of the source with any received subset of these descriptions. We study the problem of multiple description coding of stationary Gaussian sources with memory. First, we compute the multiple description rate distortion region for these sources. Then, we develop an algorithm for the design of optimal critically sampled filter banks for multiple description coding of Gaussian sources.

Sommario

Le trasformazioni e gli sviluppi lineari rappresentano uno strumento di vitale importanza nell'elaborazione dei segnali. La trasformata wavelet, in particolare, ha svolto un ruolo fondamentale im molte applicazioni come, ad esempio, la compressione dei dati. Un segnale può essere rappresentato attraverso una base o una *frame*. Le frame sono una estensione del concetto di base. Una base è un insieme di vettori che copre un certo spazio (in questo caso l'insieme dei vettori è completo), mentre la frame è un insieme di vettori che contiene almeno una base (in questo secondo caso l'insieme dei vettori è "più" che completo). Le frame sono talvolta preferite alle basi proprio per la loro generalità che ne garantisce un uso più efficace. A ciò si deve il notevole impatto che le frame stanno avendo nell'elaborazione dei segnali. Questo insieme di considerazioni giustificano l'interesse a proseguire nella ricerca di nuove basi e nuove frame per affrontare, in modo efficiente, i problemi legati all'elaborazione e alla trasmissione dei dati.

Il primo contributo di questa tesi consiste nello studio dei banchi di filtri sovracampionati e nel loro utilizzo per i problemi di comunicazione affidabile. I banchi di filtri sovracampionati rappresentano uno dei modi possibili di realizzare le frame. Generalmente, le trasformate sono utilizzate per la codifica di sorgente mentre la codifica di canale è garantita da un secondo codificatore. In questo caso, le frame sono utilizzate per garantire, allo stesso tempo, un' efficace compressione ed una trasmissione affidabile dei dati.

Nell' ambito della pura codifica di sorgente tramite trasformate, le prestazioni della trasformata wavelet sono state analizzate con particolare riferimento alla dipendenza esistente tra i coefficienti wavelet a diverse scale di risoluzione. Questa analisi ha condotto alla realizzazione di una nuova espansione per la rappresentazione di segnali regolari a tratti (piecewise smooth signals). Gli elementi di questo nuovo sviluppo sono stati chiamati *footprints*. Una delle principali caratteristiche dei footprints consiste nel rappresentare efficacemente le discontinuità di un segnale. Sono proprio le discontinuità che, generalmente, contengono la parte saliente dell'informazione. Inoltre, si mostra che, in molte applicazioni, gli algoritmi basati su questa nuova trasformata offrono prestazioni superiori ai corrispondenti algoritmi basati sulla trasformata wavelet. Le applicazioni considerate sono : la compressione dati, la rimozione di rumore nei segnali (denoising) e l'operazione di filtraggio inverso (deconvolution).

Infine, questa tesi affronta una particolare tecnica di codifica di sorgente chiamata codifica a descrizione multipla. Si tratta di una tecnica generalmente usata per trasmettere dati su sistemi di comunicazione non affidabili. Nella codifica a descrizione multipla, il codificatore produce molteplici descrizioni del medesimo segnale e il decodificatore può ottenere un'utile ricostruzione del segnale con ogni sottoinsieme di descrizioni ricevute. Oggetto di studio è la codifica a descrizione multipla di processi gaussiani stazionari con memoria. Dapprima si calcola la regione tasso compressione-distorsione (rate distortion region) per questo tipo di processi. Quindi, viene sviluppato un algoritmo per la costruzione di banchi di filtri sottocampionati per la codifica a descrizione multipla di sorgenti gaussiane.

Acknowledgments

My deepest gratitude goes to my advisor Martin Vetterli. His enthusiasm, originality, dedication to work and broad knowledge have boosted my research and have had a profound impact on my vision of the scientific world. Not only is Martin a great scientific guidance, he is also a great advisor who puts his outstanding human qualities at the service of his students. I thank him for giving always the impression that I was an important person and for being always ready to listen and to understand the needs of a graduate student.

Jelena and Vivek made my stay at Bell Labs a wonderful and, hard to believe, relaxing experience. I thank them for that and for opening my mind to the world of frames. I also have to thank Vivek for hosting me at his place for longer than expected (my 'a few days' stay ended up in an endless more than two weeks permanence). However, I have to say that his sofa was much more comfortable than expected.

Several colleagues and friends have contributed in different ways to this work. In particular, I would like to thank Minh Do, Sergio Servetto, Rahul Shukla, Michael Gastpar, Gianpaolo Evangelista, Paolo Prandoni, Zoran Pečenović, Vladan Velisavljević, Loïc Baboulaz, David Hasler, Pina Marziliano, Claudio Weidmann, Jerôme Lebrun and Andrea Ridolfi. Max, Arrate, Catherine, Michal, Magdalena, Nicu, Oscar, Arnaud and Thomas had more impact on my night life.

My Italian friends Tomaso, Diego, Antonio, Peppe, Matteo, Fulvio e Carlo made my travels back home always a great moment. Matteo honoured me with a visit in Lausanne, but I still do not know how he managed to pass the swiss customs with no documents (except for a driving license) pretending that he was a lawyer and that he had to talk to one of his swiss clients! Daniele and Superman keep organizing the 'end of the year' soccer match in Naples and this is a great occasion for 'emigrants' to find their roots again. After all Maradona was one of us.

Finally, I would like to thank my 'two' families for their continuous support. My parents and my sister represent a constant presence in my life, eventhough I know they would love to *physically* see me more often. Claudia has always given me the needed comfort and had an infinite patience with me. I thank her for recalling me, from time to time, that it is not necessarily normal to spend a week-end to compute an integral. She also had a subtle way to make fun of me each time I was talking about my research and was pronouncing 'common'

words such as multiple description coding, matlab, wavelet footprints, vector quantization. This helped me to remember that after all research is nothing more than research, that is, a game.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

At the heart of many signal processing and communication tasks lies a representation problem: what is the best way to look at data, what is the best way to approximate a signal, what is the best representation to perform a certain task.

Usually signals are represented through a set of vectors $\{\varphi_i\}_{i\in\mathcal{I}}$ which form a basis. In broad terms, a basis is a set of linearly independent vectors that cover a space. For example, any set of two linearly independent vectors forms a basis for the plane, or R^2. In general, given a set of basis vectors $\{\varphi_i\}$, one can represent any signal x as a linear combination of the $\{\varphi_i\}$'s:

$$x = \sum_{i\in\mathcal{I}} \alpha_i \varphi_i.$$

In engineering, two types of bases have played an important role: the Fourier series and the wavelet basis. Fourier analysis is widely used in many areas of engineering especially to analyze stationary signals. Wavelets have become popular especially for their ability to analyze more complex signals (i.e., signals with transient behaviours). Then, the choice between Fourier and wavelet transform depends on the type of signal and on the kind of application. For instance, in many applications the target is sparse representation. That is, a good basis for a signal is the one in which only few coefficients are different from zero or bigger than a small threshold. Now, if the target is sparse representation and the signal contains transient behaviours such as impulses, then Fourier bases perform poorly, while wavelets are the right representation of these objects. At the same time, Fourier bases are more suited to approximate signals with high frequency sinusoids.

These first considerations seem to indicate that while one particular representation is not enough, the freedom to choose between a few different bases might be the solution. With a simple example, we will soon show that this is not the case and that in many situations the best signal representation is given by overcomplete expansions, that is, by sets of vectors which contain more than the minimum number of elements necessary to form a basis. These sets of vectors are more formally called *frames*.

Consider again the problem of sparse representations (this example will be more formally explored in the next section). Assume that a signal x is well represented in one basis (e.g. Fourier basis) and that a second signal y is well represented in the other basis (e.g. wavelet basis). The signal $x+y$ might not be well represented in either basis. But the union of these two bases, which leads to a frame, can give a compact representation of $x + y$. Thus, by extending bases into frames, it is possible to obtain sparser signal representations.

Now, because the elements $\{\varphi_i\}$ of a frame are linearly dependent, there are infinitely many ways to express a signal as a linear combination of the φ_i's. Because of this, the search for the *right* representation (i.e., sparse representation in the example above) can be computationally complex and, in some cases, computational complexity associated to a frame is a key issue. At the same time, the freedom to represent a signal in different ways is an attractive property of frames. In fact, different representations can be good for different applications.

Assume that our target is error resilience (for instance, in robust communication for wireless or packet transmission). In this case, sparse signal representation is not a good solution. In fact, if a signal is given by the combination of only few elements of the frame, the error due to the loss of some of these elements can be very high. However, if each φ_i contributes to the representation of the signal, that is, if the energy of the signal is spread over all the frame elements, then the loss of some elements is much less dramatic. Clearly, this resilience increases with the number of elements in a frame, since the energy of the signal is spread over a larger number of elements. Therefore a basis, which contains the minimum number of elements necessary to cover a space, is less resilient to losses than frames.

Now, consider, again the signal $x+y$ introduced before, and the frame given by the union of two bases, in our example wavelet and Fourier bases. We have seen that this frame can give a sparse representation of $x+y$. At the same time, using the fact that there are infinitely many ways to express $x+y$ in terms of the φ_i's, we can choose a representation in which $x + y$ is given by the combination of *all* the elements of that frame. In this way, we achieve error resilience as explained above. Thus, by changing signal representation we can solve different problems.

These two examples shows that frames can be more powerful than bases and that frames represent a flexible tool. These are some of the reasons for the recent success of frames in applications and for our interest in frames. The examples highlighted before are explored in more details in the next section. Some historical notes about frames are given in Section 1.3. Finally, the outline of the thesis is given in Section 1.4.

1.2 A simple example

We now explore the previous examples more formally. First, we pursue sparse representations. We consider finite dimensional spaces (i.e. $\mathbb{R}^N$ or $\mathbb{C}^N$) and denote the space by $\mathbb{H}^N$. We define vectors as columns with the inner product

$$\langle x, y \rangle = \sum_{i=0}^{N-1} x_i^* y_i.$$

The two bases that we consider are the spike and the Fourier basis. The spike basis is given by vectors $\{\delta_i\}_{i=0,..,N-1}$ where δ_i is a vector with a 1 in the i-th position and zero elsewhere. The Fourier basis is given by $\{f_i\}_{i=0,..,N-1}$ where

$$f_i = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 \\ e^{2\pi i/N} \\ \vdots \\ e^{2\pi i(N-1)/N} \end{pmatrix}.$$

Clearly, both $\{\delta_i\}_{i=0,..,N-1}$ and $\{f_i\}_{i=0,..,N-1}$ are orthonormal bases for $\mathbb{H}^N$. Therefore, any vector $x \in \mathbb{H}^N$ can be written as $x = \sum_{i=0}^{N-1} \langle x, \delta_i \rangle \delta_i$ or $x = \sum_{i=0}^{N-1} \langle x, f_i \rangle f_i$.

Now, consider a signal x with one component well represented in one basis (e.g., spike basis) and a second one well represented in the other basis (e.g. Fourier basis), that is, assume that x is composed of a pure spike and a pure sinusoid:

$$x = \delta_m + f_l. \tag{1.1}$$

Clearly, x does not have a sparse representation in either the Fourier or the spike basis. However, the overcomplete set of vectors $\mathcal{F}$ given by the union of the Fourier and spike bases:

$$\mathcal{F} = \{\varphi_i\}_{i=0,..,2N-1} = \{\delta_i\}_{i=0,..,N-1} \cup \{f_i\}_{i=0,..,N-1}$$

can provide a sparse representation of x. In fact, there is one representation that uses only 2 non-zero coefficients, namely the one given by (1.1). Because, the vectors of this frame are linearly dependent, we can find other ways to express x. But, one can easily convince oneself that the one in (1.1) is the sparsest possible representation of x with that frame. It is also true that one can come up with other frames or ad-hoc bases which allow even sparser representations. Still, this example shows the advantage of going from bases to frames and that redundant expansions allow sparser representations.

Assume, now, that we want to pursue resilience to erasures. That is, assume that x is given by a linear combination of the φ_i's: $x = \sum_{i=0}^{2N-1} \alpha_i \varphi_i$ and that some of the coefficients α_i of this representation can be lost. These losses mimic errors that can occur in an unreliable communication medium.[1]

[1] Of course, we are depicting a very simplified scenario.

Our new target is to find a representation of x that minimizes the error due to these erasures. Clearly, the representation in (1.1) is not good in this setting, since the loss of one of the two non-zero coefficients would be catastrophic. Instead, consider the following representation of x:

$$x = \frac{1}{2} \sum_{i=0}^{N-1} \langle x, \delta_i \rangle \delta_i + \frac{1}{2} \sum_{i=0}^{N-1} \langle x, f_i \rangle f_i.$$

In this case, the loss of some coefficients is less dramatic. In particular, in the case of only one loss, it is always possible to reconstruct x exactly. This is because the remaining $2N-1$ vectors still cover the entire space. Assume, for instance, that we lose one coefficient of the spike basis, then we can use the coefficients $\langle x, f_i \rangle$ of the Fourier basis to reconstruct x and viceversa.[2] Clearly, bases do not perform well in this case. This is because in the case of loss, the remaining elements of the basis do not cover the space anymore.

Therefore, frames can be more powerful than bases not only for sparse representations but also for error resilient representations. Moreover, this second example shows that one can use the same frames for different applications.

1.3 Historical notes

Frame were introduced by Duffin and Schaeffer [44] (1952) in the context of non-harmonic Fourier series (i.e., expansions of function in $L_2([0,1])$ in complex exponentials $\exp(j\lambda_n x)$, where $\lambda_n \neq 2\pi n$). The theory then laid largely dormant until 1986 with the publication of work by Daubechies, Grossman and Meyer [25]. The mathematics of frames can be found in the works mentioned above. In addition, the paper [23] and the book [24] by Daubechies offer an excellent introduction into frames, and in particular, wavelet and Gabor frames.

Frames, or redundant representations, have been used in different areas under different guises. Perfect reconstruction oversampled filter banks are equivalent to frames in $l_2(\mathbb{Z})$. The authors in [22, 9] describe and analyze such frames. Frames show resilience to additive noise as well as numerical stability of reconstruction [24]. They have also demonstrated resilience to quantization [57]. Frames provide a greater freedom to capture significant signal characteristics and this have been exploited in several works [5, 6, 100]. In communication, frames have been used to design unitary space-time constellations for multiple-antenna wireless systems [65] and to increase resilience to losses in a packet-based communication system [56, 55].

In compression, the idea is to form a frame with the collection of different bases and to search for the best basis among that collection. This gave rise to wavelet packets [16, 88]. The advantage of this approach is that the search for the best basis is less complex than a search in a general frame. In the case of general frames, the matching pursuit algorithm [79] is the most used method.

[2] In the next chapter, we will also consider coefficient quantization and it will be shown that this kind of representation is indeed optimal in the case of one erasure.

Matching pursuit is a greedy search algorithm for an M-term approximation of an input signal. This method is related to gain-shape vector quantization [50] and to projection pursuit in statistics [49].

Frames have had an important role in analyzing performance of oversampled analog-to-digital conversion. In this context, oversampling is used to reduce the noise due to the analog-to-digital converter. Such oversampling has been used for years, but the frame based interpretation and analysis has lead to new insight and performance bounds [96].

Finally, overcomplete representations are becoming again popular in image analysis. In this case, the main goal is to come up with representations having good directionality properties. In this way, it is possible to catch and analyze curves and contours in images [11, 30].

1.4 Thesis outline and contribution

The focus of this thesis is on redundant representations or frames with applications to either communication or signal processing problems.

In the next chapter we focus on oversampled filter banks and on their application in communication. In particular, we consider communication systems with erasures and study the use of frames to mitigate the effect of erasures. In that chapter, basic properties of frames and oversampled filter banks are reviewed. Then the effect of erasures on the structure of filter banks is studied and several filter bank examples are discussed. Finally, we show that filter banks implementing uniform tight frames and newly defined strongly uniform tight frames perform the best in this context.

In Chapters 3 and 4, we introduce a new overcomplete expansion for compact representation of piecewise smooth signals. The elements of this expansion are called *footprints*. Footprints are scale-space vectors built from the wavelet transform and their main characteristics is to exploit the dependency across scales of the wavelet coefficients. In Chapter 3, we present the footprint expansion and propose algorithms to efficiently search for the sparsest footprint representation of a signal. We also study the use of matching pursuit to approximate a signal with footprints and show that in some cases matching pursuit performs as well as the best search algorithm. Applications, namely compression, denoising and deconvolution, are discussed in Chapter 4. We compare footprints based methods with wavelet algorithms and see that footprints are consistently superior to wavelet algorithms.

In Chapters 5 and 6, we go back to communication problems. In particular, we study a source coding technique called multiple description coding. Multiple description coding is used for data transmission over unreliable networks in particular packet switched networks.

In Chapter 5, we review some fundamental performance bounds of multiple description coders and present a new multiple description rate-region for stationary Gaussian sources.

Chapter 6 studies some practical multiple description systems. In particular, we review the multiple description scalar quantizer [102] and the multiple description transform coder [83, 110, 53, 54]. Then we propose the use of two-channel critically sampled filter banks to implement multiple description coders. In this case, we make statistical assumptions on the source to be coded and the transform is designed to introduce statistical redundancy (correlation) in the signal representation. This redundancy is used to combat channel impairments. Therefore, eventhough we do not use frames in an explicit way, the idea of using transforms which introduce statistical redundancy to increase error resilience is in spirit similar to the idea of using the redundancy introduced by frames to allow robust transmission. In other words, frames introduce deterministic redundancy, critically sampled filter banks for multiple description coding introduce statistical redundancy.

Finally, we conclude in Chapter 7 with a summary and an outlook on future research.

Chapter 2

Frames and Oversampled Filter Banks for Communications

2.1 Introduction

The use of frame expansions in signal processing has recently become quite popular. The basics of frames can be found in several sources, in particular, in the original work by Duffin and Schaeffer [44] and in the paper [23] and the book [24] by Daubechies which offer an excellent introduction into frames, and in particular, wavelet and Gabor frames. Perfect reconstruction oversampled filter banks which are equivalent to a particular class of frames in $l_2(\mathbb{Z})$ were described and analyzed by Cvetković and Vetterli in [22] (See also [9]).

In this chapter, our aim is to exploit the redundancy present in a frame representation to reduce the effect of losses in a communication system. In a previous work [55], the frame elements belonged to $\mathbb{R}^N$ (or $\mathbb{C}^N$) and can be seen as filters in a block-transform filter bank. Here, we investigate frames with elements in $l_2(\mathbb{Z})$; they can be seen as filters in a general, oversampled filter bank.

We consider the communication model depicted in Figure 2.1. An input sequence $x[n]$ is fed through an M-channel finite-impulse response (FIR) filter bank followed by downsampling by N ($N < M$). The M output sequences are then separately scalar quantized with uniform scalar quantizers and sent over M different channels. Each channel either works perfectly or not at all. The decoder receives only $M - e$ of the quantized output sequences, where e is the number of erasures during the transmission. We assume there are no more than $M-N$ erasures. The reconstruction process is linear. We wish to find properties of the filter banks that minimize the mean square error (MSE) between the

[0]This chapter includes research conducted jointly with Jelena Kovačević and Vivek Goyal [36, 74]

input and the reconstructed sequences. We do not make any assumptions on the input source. Rather, a statistical model for the quantization error makes the reconstruction quality depend only on the characteristics of the filter bank.

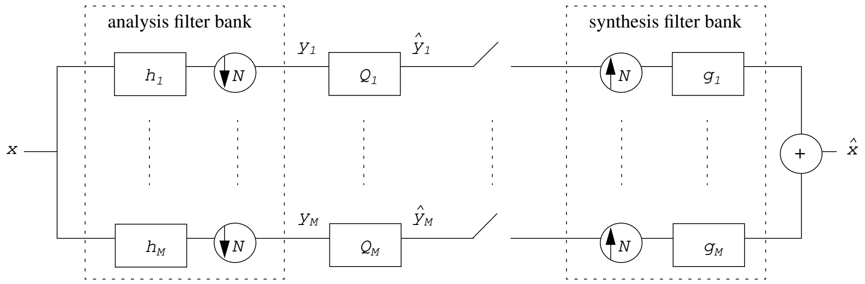

Figure 2.1: Abstraction of a lossy network with a frame expansion implemented by an oversampled FIR filter bank. An input sequence $x[n]$ is fed through an M-channel finite-impulse response (FIR) filter bank followed by downsampling by N ($N < M$). The M output sequences are then separately scalar quantized with uniform scalar quantizers and sent over M different channels. Each channel either works perfectly or not at all. The decoder receives only $M - e$ of the quantized output sequences, where e is the number of erasures during the transmission. We assume there are no more than $M - N$ erasures. The reconstruction process is performed by the synthesis filter bank. The choice of synthesis filters depends on which channels are received.

In the next section, we go through the basics of frame expansions in $\mathbb{C}^N$ and $l_2(\mathbb{Z})$. In Section 2.3, we introduce the notion of strongly uniform frames and discuss several examples. We then quantize the frame coefficients and find the MSE (Section 2.4). Finally, in Section 2.5 we let some coefficients be erased (mimicking the losses in a network) and discuss the effect on both the structure of the frame and the MSE. We draw conclusions in Section 2.6.

2.2 Frame Expansions and Oversampled Filter Banks

This review of frames and oversampled filter banks is based on [22, 24, 55]. For convenience we start with the finite-dimensional case, that is, we consider only frames in $\mathbb{C}^N$.

A family of M vectors $\Phi = \{\varphi_i\}_{i=1}^M \in \mathbb{C}^N$ constitutes a frame if for any vector $x \in \mathbb{C}^N$ there exist two constants $A > 0$ and $B < \infty$ such that:

$$A\|x\|^2 \leq \|y\|^2 = \sum_{i=1}^{M} |\langle x, \varphi_i \rangle|^2 \leq B\|x\|^2. \tag{2.1}$$

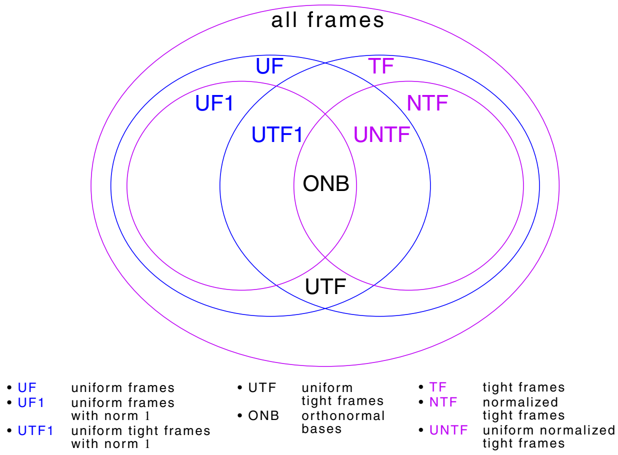

Figure 2.2: Frames at a glance. Note that here we denote by UF uniform frames with the same norm, not necessarily 1. When the norm is 1, we say so explicitly. For instance, UF1 denotes uniform frames with norm 1.

When $A = B$ the frame is *tight* (TF). If $A = B = 1$, the frame is *normalized tight* (NTF). A frame is *uniform* (UF) if all its elements have norm 1,[1] $\|\varphi_i\| = 1$. For a UTF, the frame bound A gives the *redundancy ratio* (it is M/N in this case). A UTF which is also normalized, that is, with $A = 1$, is an orthonormal basis (ONB). Figure 2.2 helps to clarify the various possible sets of frames. Moreover, the following theorem tells us that every tight frame can be seen as a projection of an orthonormal basis from a larger space.

Theorem 2.1 (Naimark [2]) [2] *A family $\{\varphi_i\}_{i\in I}$ in a Hilbert space H is a normalized tight frame for H if and only if there is a larger Hilbert space $H \subset K$ and an orthonormal basis $\{e_i\}_{i\in I}$ for K so that the orthogonal projection P of K onto H satisfies: $Pe_i = \varphi_i$, for all $i \in I$.*

If we denote vectors as columns then we can define the *frame operator* F associated with Φ as an $M \times N$ matrix with the ith row equal to φ_i^*.[3] That is:

$$F = \begin{pmatrix} \varphi_{11}^* & \cdots & \varphi_{1N}^* \\ \vdots & \ldots & \vdots \\ \varphi_{M1}^* & \cdots & \varphi_{MN}^* \end{pmatrix}, \tag{2.2}$$

We then define an output vector with $y = Fx$. The properties of a frame can be conveniently defined using its frame operator F. For example, (2.1) can be

[1] Actually, the definition of a UF is more general; the norm is allowed to be $c \neq 1$. In this work, however, we consider only UF with norm 1.

[2] This theorem has been rediscovered by several people in recent years. See for instance: [24, 64, 94].

[3] The superscript * denotes the Hermitian transpose.

rewritten as

$$AI \leq F^*F \leq BI. \qquad (2.3)$$

It follows that F^*F is invertible (Lemma 3.2.2 in [24]), and furthermore

$$B^{-1}I \leq (F^*F)^{-1} \leq A^{-1}I. \qquad (2.4)$$

Then, the *dual frame* of Φ is a frame defined as $\tilde{\Phi} = \{\tilde{\varphi}_i\}_{i=1}^{M}$, where

$$\tilde{\varphi}_i = (F^*F)^{-1}\varphi_i, \qquad (2.5)$$

for $i = 1, \ldots, M$. Noting that $\tilde{\varphi}_i^* = \varphi_i^*(F^*F)^{-1}$ and stacking $\tilde{\varphi}_1^*, \tilde{\varphi}_2^*, \ldots, \tilde{\varphi}_M^*$ in a matrix, the frame operator associated with $\tilde{\Phi}$ is

$$\tilde{F} = F(F^*F)^{-1}. \qquad (2.6)$$

Since $\tilde{F}^*\tilde{F} = (F^*F)^{-1}$, (2.4) shows that B^{-1} and A^{-1} are frame bounds for $\tilde{\Phi}$. Vector x can be reconstructed from y using the so called pseudo-inverse $F^\dagger$.

$$F^\dagger = \tilde{F}^*. \qquad (2.7)$$

Finally, a frame is tight if and only if $F^*F = AI_N$, where I_N is the $N \times N$ identity matrix while a UTF satisfies $F^*F = \frac{M}{N}I_N$.

If we call λ_i the eigenvalues of F^*F, then:

1. the sum of the eigenvalues of F^*F equals the sum of the lengths of the frame vectors: $\sum_{i=1}^{N}\lambda_i = tr(F^*F) = tr(FF^*) = \sum_{k=1}^{M}\varphi_k^*\varphi_k = \sum_{k=1}^{M}\|\varphi_k\|^2$,

2. for a UF the sum of the eigenvalues equals M,

3. for a TF, F^*F has eigenvalue A with multiplicity N,

4. for a UTF, F^*F has eigenvalue $\frac{M}{N}$ with multiplicity N.

Consider now the filter bank shown in Figure 2.1. The input into the filter bank is a square-summable infinite sequence $x[n] \in l_2(\mathbb{Z})$. Call $H_i(\omega) = [H_{i1}(\omega), H_{i2}(\omega), ...H_{iN}(\omega)]^*$ the polyphase representation of the ith analysis filter where

$$H_{ij}(\omega) = \sum_{n=-\infty}^{\infty} h_i[nN - j]e^{-jn\omega}.$$

Call $H(\omega)$ the corresponding $M \times N$ polyphase analysis matrix, which is a matrix whose ith row equals $H_i^*(\omega)$. The following result establishes the equivalence between frames in $l_2(\mathbb{Z})$ and polyphase matrices:

Proposition 2.1 (Cvetković and Vetterli [22]) *A filter bank implements a frame decomposition in $l_2(\mathbb{Z})$ if and only if its analysis polyphase matrix is of full rank on the unit circle.*

Moreover, it is of interest to note that when the filters $h_i[n]$, $i = 1, \dots, M$, are all of length N, then, each polyphase sequence is of length 1. Thus, $H(\omega)$ becomes a constant. In this case, it is easy to relate the polyphase matrix to the frame operator F. In fact, the filter bank as given in Figure 2.1 implements a finite-dimensional frame expansion as we explained earlier. In other words

$$\varphi_{in} \;=\; h_i^*[N-n], \tag{2.8}$$

for $i = 1, \dots, M$ and $n = 1, \dots, N$ and the polyphase matrix reduces to $H(\omega) = FJ$, with J an antidiagonal matrix.[4]

Many properties of frames implemented with filter banks can be stated using the polyphase matrix in much the same way as F was used to define properties in the finite dimensional case. For instance as in the finite-dimensional case, the dual frame is represented by

$$\tilde{H}(\omega) \;=\; H(\omega)(H^*(\omega)H(\omega))^{-1}, \tag{2.9}$$

while the pseudo-inverse is

$$H^\dagger(\omega) \;=\; \tilde{H}^*(\omega). \tag{2.10}$$

We now revisit briefly the definition of a UF. The frame is uniform if $\|h_i[n]\| = 1$ for $i = 1, \dots, M$ or, using Parseval's relation, if

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{k=1}^{N}|H_{ik}(\omega)|^2 d\omega \;=\; 1,$$

for $i = 1, \dots, M$. This leads us to define a more restrictive condition:

Definition 2.1 (*Strongly uniform frame*) *A frame expansion in $l_2(\mathbb{Z})$ implemented by an $M \times N$ polyphase matrix $H(\omega)$ is strongly uniform*[5] *if*

$$\sum_{k=1}^{N}|H_{ik}(\omega)|^2 \;=\; 1, \tag{2.11}$$

for $i = 1, \dots, M$ and $\omega \in [-\pi, \pi]$.

In other words, a strongly uniform frame is implemented by a filter bank which is uniform for each fixed ω. Clearly, strongly uniform frames are a subset of uniform frames. If $H(\omega) = FJ$ and F is uniform, then the corresponding frame is strongly uniform. Moreover, a square paraunitary matrix is automatically strongly uniform. Further examples of strongly uniform frames will be shown in the next section.

In the finite dimensional case, we have seen that a frame is tight if and only if $F^*F = A \cdot I_N$. If we are dealing with infinite sequences, analogous results can be formulated. The following is known:

[4] The matrix J just reverses the order of columns of F.

[5] As before, when we say "strongly uniform", we will mean "strongly uniform with norm 1".

Proposition 2.2 (Cvetković and Vetterli [22]) *A filter bank implements a tight frame expansion in $l_2(\mathbb{Z})$ if and only if $H^*(\omega)H(\omega) = AI_N$.*

Proposition 2.3 (Vaidyanathan [101]) *An $M \times N$ polyphase matrix $H(\omega)$ represents a tight frame if and only if it has the following decomposition: $H(\omega) = U(\omega)F$, where $U(\omega)$ is an $M \times M$ paraunitary matrix*[6] *and F is an $M \times N$ matrix such that $F^*F = AI_N$.*

Proposition 2.4 (Cvetković [21]) *For a frame associated with an FIR filter bank, with the polyphase analysis matrix $H(\omega)$, its dual frame (2.9) consists of finite length vectors if and only if $H^*(\omega)H(\omega)$ is unimodular.*

Here *unimodular* means that the determinant of $H^*(\omega)H(\omega)$ is ± 1. This result leads us to formulate the following useful property of TFs:

Corollary 2.1 *Given an FIR analysis polyphase matrix $H(\omega)$ corresponding to a TF, the synthesis polyphase matrix $G(\omega)$ corresponding to the pseudo-inverse as in (2.10), is FIR as well.*

Proof: Using Proposition 2.2, we know that $H^*(\omega)H(\omega) = AI_N$. Since $H(\omega)$ is FIR, $H(\omega)/\sqrt{A}$ is FIR as well. Thus, $(H(\omega)/\sqrt{A})^*(H(\omega)/\sqrt{A}) = I_N$ is unimodular. By Proposition 2.4, the dual frame (synthesis polyphase matrix) to $H(\omega)/\sqrt{A}$ is FIR as well. Since scaling does not affect the FIR property, the dual frame (synthesis polyphase matrix) to $H(\omega)$ is FIR.

$\square$

As for the eigenvalues, if we call $\lambda_i(\omega)$ the spectral eigenvalues of $H^*(\omega)H(\omega)$, then:

1. the integral sum of the spectral eigenvalues of $H^*(\omega)H(\omega)$ equals the sum of the filters' norms: $\frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{i=1}^{N}\lambda_i(\omega)d\omega = \sum_{i=1}^{M}\|h_i[n]\|^2$;

2. for a UF, the integral sum of the eigenvalues equals M;

3. for a TF, $H^*(\omega)H(\omega)$ has eigenvalues constant over the unit circle and equal to A with multiplicity N: $\lambda_i(\omega) = A, \quad i = 1, ..., N$;

4. for a UTF, $H^*(\omega)H(\omega)$ has eigenvalues constant over the unit circle and equal to $\frac{M}{N}$ with multiplicity N.

[6] A square matrix $H(\omega)$ is called *paraunitary* if $H^*(\omega)H(\omega) = H(\omega)H^*(\omega) = cI, c \neq 0$. Moreover, any paraunitary matrix can be decomposed into a sequence of elementary matrices such as rotations and delays [101].

2.3 Examples of Uniform and Strongly Uniform Frames

Oversampled filter banks are sometimes preferred to classical critically downsampled filter banks for their greater design freedom. However, this freedom makes the actual design difficult.

One of the most used families of oversampled filter banks is the nondownsampled one. They are obtained by eliminating the downsampling in the filter bank scheme. If the analysis and synthesis filters are power complementary (i.e. with FIR filters, the synthesis filters are the time reversed versions of the analysis ones) then the corresponding frame is tight and uniform but not strongly uniform.

It will be shown in next sections that strongly uniform tight frames constitute an important class of frames. We propose the following factorization to design polyphase matrices corresponding to strongly uniform tight frames:

$$H(\omega) = FU(\omega), \tag{2.12}$$

where F is a uniform tight frame in $\mathbb{C}^N$ and $U(\omega)$ is an $N \times N$ paraunitary matrix. It is easy to see that such a polyphase matrix corresponds to a strongly uniform tight frame.

Note the difference between this factorization and the one in Proposition 2.3 ($H(\omega) = U(\omega)F$). The order of the elements is reversed, so in this last factorization, the paraunitary matrix has size $M \times M$, while in our factorization it has size $N \times N$ ($N < M$). This is not surprising since the family of polyphase matrices with the factorization $H(\omega) = U(\omega)F$ represents the more general class of tight frames and not the restricted class of strongly uniform tight frames.

Although we cannot claim that our factorization includes all possible strongly uniform tight frames, we can state the following:

Theorem 2.2 *Define an equivalence relation by bundling a frame (implemented with an FIR oversampled filter bank) with all frames that result from rigid rotations of its elements as well as negation or shift of some individual ones (i.e. $h_i[n] \to -h_i[n-k]$ $k \in \mathbb{Z}$). When $M = N+1$, there is a single equivalence class for all strongly uniform tight frames.*

Proof: See Appendix 2.A.1.

□

Since a UTF F in $\mathbb{C}^N$ can be seen as a strongly uniform tight frame in $l_2(\mathbb{Z})$ (that is, $H(\omega) = FJ$), Theorem 2.2 basically says that the factorization in (2.12) essentially includes all the possible strongly uniform tight frames when $M = N+1$ (up to a shift or negation of some individual elements). Also, when $H(\omega) = FJ$, this theorem reduces to Theorem 2.6 from [55].

Unfortunately, when M exceeds $N+1$, there are uncountably many equivalence classes of the type described above; thus, we cannot systematically obtain all uniform tight frames. However, at least for $N = 2$, UTFs still have a simple characterization given by Theorem 2.7 from [55]:

Theorem 2.3 (Goyal, Kovačević and Kelner [55]) *The following are equivalent:*

1. $\{\varphi_k = (\cos\alpha_k, \sin\alpha_k)\}_{k=1}^{M}$ *is a uniform tight frame.*
2. $\sum_{k=1}^{M} z_k = 0$ *where* $z_k = e^{j2\alpha_k}$ *for* $k = 1, 2, \ldots, M$.

Thus, a simple combination of our factorization (2.12) together with the complete characterization of UTFs for $N = 2$ given by the above theorem, produces a useful factorization (although probably not complete) of SUTFs.

2.4 Quantized Oversampled Filter Banks

In this section we will analyze the effect of quantization on the performance of the system. For the moment we assume that there are no erasures during transmission. We want the reconstruction operator to be linear, that is, we want it to be implemented by a synthesis filter bank. The reconstruction operator that we will use is the pseudo-inverse (2.10).

We will assume that the quantization error can be treated as additive white noise with variance $\sigma^2 = \Delta^2/12$, where Δ represents the step size of the quantizer and each quantizer has the same step size. We further assume that the noise sequences generated by two different channels are pairwise uncorrelated. This can be expressed as:

$$\hat{y}_i[n] = y_i[n] + w_i[n], \tag{2.13}$$

for $i = 1, \ldots, M$, and

$$E[w_i[n]w_j^*[n-m]] = \sigma^2 \delta_{ij}\delta[m]. \tag{2.14}$$

Under this assumption (input sequences corrupted by additive white noise), the pseudo-inverse in (2.7) and (2.10) is the best linear reconstruction operator in the mean square sense [24]. Moreover, in Appendix 2.A.2 we show that the MSE due to quantization is:

$$\begin{aligned} MSE &= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} tr((H^*(\omega)H(\omega))^{-1})d\omega & (2.15)\\ &= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi}\sum_{k=1}^{N}\frac{1}{\lambda_k(\omega)}d\omega, & (2.16)\end{aligned}$$

where $\lambda_k(\omega)$, $k = 1, \ldots, N$ denote the spectral eigenvalues of $H^*(\omega)H(\omega)$. We will be using the above two expressions interchangeably. Recall that the integral sum of the eigenvalues is constant and if we are encoding with a uniform frame, it is equal to M. Thus, we want to minimize the MSE given the constraint that the integral sum of the eigenvalues is constant. This occurs when the eigenvalues are equal and constant over ω which is true if and only if the original frame is tight. The above discussion and computation can be summarized by the following theorem:

Theorem 2.4 *When encoding with a filter bank implementing a uniform frame and decoding with the pseudo-inverse under the noise model (2.13)-(2.14), the MSE is minimum if and only if the frame is tight. Then:*

$$MSE_0 \;=\; \frac{N}{M}\sigma^2. \tag{2.17}$$

2.5 Introducing Erasures

Here we consider the effect of erasures on the structure of the frame and on the MSE. We denote by E the index set of erasures and by $H_E(\omega)$ the polyphase matrix after $e = |E|$ erasures. $H_E(\omega)$ is an $(M-e) \times N$ matrix obtained by deleting the E-numbered rows from the $M \times N$ polyphase matrix $H(\omega)$. The first question to be answered is under which conditions $H_E(\omega)$ still represents a frame. We then study the effect of erasures on the MSE.

2.5.1 Effect of Erasures on the Structure of a Frame

Our aim is to use the pseudo-inverse operator to reconstruct after e erasures. The pseudo-inverse matrix is defined only if the matrix $H_E(\omega)$ is still a frame, that is, if and only if it is still of full rank on the unit circle. This leads to the following definition:

Definition 2.2 *An oversampled filter bank which implements a frame expansion represented by a polyphase matrix $H(\omega)$ is said to be robust to $e = |E|$ erasures if and only if for any index set E of erasures, $H_E(\omega)$ is of full rank on the unit circle.*

Let us consider first the case where there is only one erasure.

Theorem 2.5 *An oversampled filter bank which implements a uniform tight frame is robust to one erasure if and only if*

$$\sum_{k=1}^{N} |H_{ik}(\omega)|^2 \;<\; \frac{M}{N},$$

for $i = 1, \dots, M$ and for all ω.

Proof: See Appendix 2.A.3.

□

Recall that by definition a strongly uniform frame is such that: $\sum_{j=1}^{N}|H_{ij}(\omega)|^2 = 1,\ i = 1,...M$, for all ω. Thus, as a consequence of the previous theorem we can state:

Corollary 2.2 *Any oversampled filter bank which implements a strongly uniform tight frame is robust to one erasure.*

The result of Theorem 2.5 does not reveal anything about the existence of filter banks which are robust to more than one erasure.

In [55], it has been shown that a complex harmonic frame in $\mathbb{C}^N$ or a real harmonic frame in $\mathbb{R}^N$ is robust to e erasures ($e \leq M-N$). A complex harmonic tight frame is given by:

$$\varphi_{ki} = \frac{1}{\sqrt{N}} W_M^{(k-1)(i-1)}, \qquad i = 1,2,...,N, \quad k = 1,2,...,M,$$

where $W_M = e^{j2\pi/M}$. (A real harmonic tight frame could be defined similarly). The following theorem guarantees the existence of at least one family of strongly uniform tight frames in $l_2(\mathbb{Z})$ which are robust to e erasures ($e \leq M-N$):

Theorem 2.6 *Consider an oversampled filter bank with polyphase analysis matrix $G(\omega) = FU(\omega)$, where F is a complex harmonic frame in $\mathbb{C}^N$ or a real harmonic frame in $\mathbb{R}^N$ and $U(\omega)$ is an $N \times N$ polyphase matrix nonsingular on the unit circle ($\det(U(\omega)) \neq 0$). This filter bank is robust to e erasures ($e \leq M-N$).*

Proof: See Appendix 2.A.4.

□

If $U(\omega)$ is a paraunitary matrix, the resulting oversampled filter bank $G(\omega) = FU(\omega)$ represents a strongly uniform tight frame robust to e erasures ($e \leq M-N$).

2.5.2 Effect of Erasures on the MSE

In the previous section, it has been shown that it is possible to design oversampled filter banks which are robust up to $M-N$ erasures. We assume such filter banks for the rest of the chapter.

Now, we want to compute the effect of the erasures on the MSE. Call $H(\omega)$ the polyphase matrix related to the original frame and $H_E(\omega)$ the polyphase matrix after $e = |E|$ erasures. The reconstruction uses the dual polyphase matrix $H_E^{\dagger}(\omega)$ and the quantization model is the one proposed in (2.13)-(2.14). Under these assumptions the mean square error is equivalent to the one determined in (2.15)-(2.16):

$$MSE_E = \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} tr((H_E^*(\omega)H_E(\omega))^{-1})d\omega \tag{2.18}$$

$$= \frac{\sigma^2}{2\pi N}\sum_{k=1}^{N}\int_{-\pi}^{\pi}\frac{1}{\lambda_k(H_E^*(\omega)H_E(\omega))}d\omega, \tag{2.19}$$

where $\lambda_k(H_E^*(\omega)H_E(\omega))$, $k = 1, ..., N$, are the spectral eigenvalues of $H_E^*(\omega)H_E(\omega)$.

However, our target is to express the mean square error in terms of the original frame and to find properties that the original frame operator has to satisfy to minimize the distortion. Consider first a strongly uniform frame and $e = 1$:

Theorem 2.7 *Consider encoding with a strongly uniform frame and decoding with linear reconstruction. The MSE averaged over all possible erasures of one channel is minimum if and only if the original frame is tight. Moreover a tight frame minimizes the maximum distortion caused by one erasure. The MSE is given by:*

$$MSE_1 = \left(1 + \frac{1}{M-N}\right) MSE_0. \tag{2.20}$$

where MSE_0 is given by (2.17).

Proof: See Appendix 2.A.5

□

It is hard to extend the result of this theorem to the case of more than one erasure. However, it is possible to compute the MSE with $e > 1$ when the original frame is strongly uniform and tight:

$$MSE_E = \left(1 + \frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{i=1}^{e}\frac{\mu_i(\omega)}{M - N\mu_i(\omega)}d\omega\right) MSE_0, \tag{2.21}$$

where $\mu_i(\omega)$ are the spectral eigenvalues of $T^*(\omega)T(\omega)$ and $T(\omega)$ is the $N \times e$ polyphase matrix with columns $\{H_i(\omega)\}_{i\in E}$.

Since (2.21) is similar to (2.16), and the spectral sum of the e eigenvalues of $T(\omega)$ is constrained to be a constant, the minimum in (2.21) occurs when all the eigenvalues are equal and constant, which is true when $T(\omega)$ is tight.

When any erasure event is possible - meaning any combination of switches may be open in Figure 2.1 - it is not possible to make $T(\omega)$ always correspond to a tight frame. There are situations in which the number of "physical" channels (separate transmission media) is less than the number of branches in the analysis filter bank. In this case, there may be sets of branches that are each completely lost or completely received and then it may be possible for the erased vectors to form a tight frame. Therefore, in this situation it is possible to minimize (2.21).

2.6 Conclusions

In this chapter, we have investigated the use of frame expansions to mitigate the effect of erasures in a communication systems. We have focused on the case of frames implemented with oversampled filter banks and we have seen that oversampled filter banks associated with uniform and strongly uniform tight frames perform the best in this context. In addition, we have presented practical ways to design oversampled filter banks robust to one or more erasures.

Appendix 2.A Proofs

2.A.1 Proof of Theorem 2.2

Given a strongly uniform tight frame represented by the polyphase matrix $K(\omega)$, all the other polyphase matrices related to the same equivalent class are obtained as follows:

$$H(\omega) \;=\; \Sigma\, K(\omega)\, U(\omega), \tag{2.22}$$

where $U(\omega)$ is an $N \times N$ paraunitary matrix, $\Sigma = diag(\sigma_1, \sigma_2, \dots, \sigma_M)$ and $\sigma_i \;=\; \pm\, e^{-jl\omega}$, $l \in mathbbZ$, $i = 1, \dots, M$. This equivalence class preserves tightness, uniformity, as well as strong uniformity. Thus, if $K(\omega)$ is strongly uniform and tight, so is $H(\omega)$.

Now, let $H(\omega)$ be a polyphase matrix associated with a SUTF with $M = N+1$. By Theorem 2.1, one can see that $H(\omega)$ consists of the first N columns of a scaled $(N+1) \times (N+1)$ paraunitary matrix $\bar{H}(\omega)$. Each row (or column) of $\bar{H}(\omega)$ is of norm $\sqrt{N+1/N}$, that is:

$$\sum_{k=1}^{N+1} |\bar{H}_{ik}(\omega)|^2 \;=\; \frac{N+1}{N}, \tag{2.23}$$

for $i = 1, 2, \dots, N+1$. Moreover, since our frame is strongly uniform we have:

$$\sum_{k=1}^{N} |\bar{H}_{ik}(\omega)|^2 \;=\; 1, \tag{2.24}$$

for $i = 1, 2, \dots, N+1$. Subtracting (2.24) from (2.23) we obtain:

$$|\bar{H}_{i,N+1}(\omega)|^2 \;=\; \frac{1}{N}.$$

Since $H(\omega)$ is realized with FIR filters, it is formed only of Laurent polynomial elements. This implies that $\bar{H}_{i,N+1}(\omega)$ must be a monomial : $\bar{H}_{i,N+1}(\omega) = \pm N^{-1/2} e^{-jl\omega}$ $l \in \mathbb{Z}$. Without loss of generality we assume that $\bar{H}_{i,N+1}(\omega) = \pm N^{-1/2}$. That is, the last column of $\bar{H}(\omega)$ is $(\pm N^{-1/2}, \pm N^{-1/2}, ..., \pm N^{-1/2})$ for some choice of signs.

Any given choice of signs in $\bar{H}_{i,N+1}(\omega)$ determines a subspace. Thus the span of the other N subspaces (each subspace is related to one of the channels) must be the orthogonal complement to this subspace. Since orthonormal bases for a subspace are unitarily equivalent, the possible tight frames corresponding to a single choice of signs are in the same equivalence class. Flipping signs yields frames in the same equivalence class.

2.A.2 Derivation of (2.16)

We now find the error of the reconstruction after the frame coefficients have been quantized:

$$\begin{aligned}
MSE &= E[\|x[n]-\hat{x}[n]\|^2] \\
&= \frac{1}{2\pi N}\int_{-\pi}^{\pi} E[\|X(\omega)-\hat{X}(\omega)\|^2]d\omega \\
&= \frac{1}{2\pi N}\int_{-\pi}^{\pi} E[(H^\dagger(\omega)W(\omega))^*(H^\dagger(\omega)W(\omega))]d\omega \\
&= \frac{1}{2\pi N}\int_{-\pi}^{\pi} E[\sum_{i,j=1}^{M} W_i^*(\omega)W_j(\omega)\tilde{H}_i^*(\omega)\tilde{H}_j(\omega)]d\omega \\
&= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} \sum_{i=1}^{M} \tilde{H}_i^*(\omega)\tilde{H}_i(\omega)d\omega \\
&= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} tr(\tilde{H}^*(\omega)\tilde{H}(\omega))d\omega \\
&= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} tr((H^*(\omega)H(\omega))^{-1})d\omega \\
&= \frac{\sigma^2}{2\pi N}\int_{-\pi}^{\pi} \sum_{k=1}^{N}\frac{1}{\lambda_k(\omega)}d\omega.
\end{aligned}$$

2.A.3 Proof of Theorem 2.5

Assume that the erased channel is $H_i(\omega)$. Call $H_{(i)}(\omega)$ the polyphase matrix after one erasure. Then:

$$\begin{aligned}
H_{(i)}^*(\omega)H_{(i)}(\omega) &= H^*(\omega)H(\omega) - H_i(\omega)H_i^*(\omega) \\
&= \frac{M}{N}I_N - H_i(\omega)H_i^*(\omega).
\end{aligned} \tag{2.25}$$

$H_{(i)}(\omega)$ is a frame if and only if $H_{(i)}^*(\omega)H_{(i)}(\omega)$ is of full rank on the unit circle. That means that $(H_{(i)}^*(\omega)H_{(i)}(\omega))^{-1}$ must exist on the unit circle. The identity:

$$(A-BCD)^{-1} = A^{-1} + A^{-1}B(C^{-1}-DA^{-1}B)^{-1}DA^{-1}, \tag{2.26}$$

with $A=(M/N)I_N$, $B=H_i(\omega)$, $C=1$, and $D=H_i^*(\omega)$ yields:

$$(H_{(i)}^*(\omega)H_{(i)}(\omega))^{-1} = \tag{2.27}$$

$$= \frac{N}{M}I_N + \tag{2.28}$$

$$+ \frac{N}{M}I_N H_i(\omega)\left(1-H_i^*(\omega)\frac{N}{M}I_N H_i(\omega)\right)^{-1} H_i^*(\omega)\frac{N}{M}I_N \tag{2.29}$$

$$= \frac{N}{M}I_N + \frac{N^2}{M^2}\left(1-\frac{N}{M}H_i^*(\omega)H_i(\omega)\right)^{-1} H_i(\omega)H_i^*(\omega). \tag{2.30}$$

Thus, the matrix is invertible if and only if:

$$1 - \frac{N}{M} H_i^*(\omega) H_i(\omega) \;\neq\; 0$$

for all ω. The desired inequality now follows from the fact that the frequency response of each filter is continuous (since we are only considering FIR filters) and the frame is uniform. The continuity of the filters implies that $\sum_{k=1}^{N} |H_{ik}(\omega)|^2 < M/N$, for all ω or $\sum_{k=1}^{N} |H_{ik}(\omega)|^2 > M/N$, for all ω. However, since the frame is uniform, that is, $(1/2\pi) \int_{-\pi}^{\pi} \sum_{k=1}^{N} |H_{ik}(\omega)|^2 = 1$, then $\sum_{k=1}^{N} |H_{ik}(\omega)|^2 < M/N$, for all ω.

2.A.4 Proof of Theorem 2.6

First note that if a finite set of channels has a subset that is a frame, then the original set of channels is also a frame. Thus it suffices to consider subsets with N channels; since all of these will be shown to be frames, larger subsets are also frames.

Let us call $H_E(\omega)$ the $N \times N$ polyphase matrix after $e = M - N$ erasures. $H_E(\omega)$ is a frame if and only if $\det(H_E(\omega)) \neq 0$ on the unit circle. Now, we know that $\det(F_E) \neq 0$ for any subset of $e = (M - N)$ erasures [55] and since: $H_E(\omega) = F_E U(\omega)$,

$$\det(H_E(\omega)) \;=\; \det(F_E)\det(U(\omega)) \;\neq\; 0$$

for all ω.

2.A.5 Proof of Theorem 2.7

As in the proof of Theorem 2.5, assume that the erased channel is $H_i(\omega)$. Call $H_{(i)}(\omega)$ the polyphase matrix after one erasure. Then (2.25) holds. According to (2.18), the average MSE with one erasure is:

$$\overline{MSE}_1 \;=\; \frac{\sigma^2}{2\pi MN} \sum_{i=1}^{M} \int_{-\pi}^{\pi} tr((H_{(i)}^*(\omega) H_{(i)}(\omega))^{-1}) d\omega. \tag{2.31}$$

Call

$$\begin{aligned} v(\omega) &= H^*(\omega) H(\omega), \\ v_i(\omega) &= H_i^*(\omega)(H^*(\omega)H(\omega))^{-1} H_i(\omega). \end{aligned}$$

Note that $v(\omega)$ is an $N \times N$ matrix, while $v_i(\omega)$ is a scalar. With that, (2.25) can be rewritten as

$$v_{(i)}(\omega) \;=\; H_{(i)}^*(\omega) H_{(i)}(\omega) \;=\; v(\omega) - H_i(\omega) H_i^*(\omega).$$

We now find

$$\begin{aligned} v_{(i)}(\omega)^{-1} &= v(\omega)^{-1} \;+ \\ &+\; v(\omega)^{-1} H_i(\omega)\left(1 - v_i(\omega)\right)^{-1} H_i^*(\omega) v(\omega)^{-1}, \end{aligned}$$

where we used (2.26) with $A = v(\omega)$, $B = H_i(\omega)$, $C = 1$, and $D = H_i^*(\omega)$. Taking the trace of both sides gives

$$\begin{aligned} tr(v_{(i)}(\omega)^{-1}) &= tr(v(\omega)^{-1}) + \\ &+ (1 - v_i(\omega))^{-1}\, tr(v(\omega)^{-1} H_i(\omega) H_i^*(\omega) v(\omega)^{-1}) \\ &= tr(v(\omega)^{-1}) + \\ &+ (1 - v_i(\omega))^{-1}\, tr(H_i^*(\omega) v(\omega)^{-2} H_i(\omega)) \\ &= tr(v(\omega)^{-1}) + \frac{H_i^*(\omega) v(\omega)^{-2} H_i(\omega)}{1 - v_i(\omega)}, \end{aligned}$$

since both $H_i^*(\omega)v(\omega)^{-2}H_i(\omega)$ and $1 - v_i(\omega)$ are scalars and the trace of the product is invariant to the cyclic permutation of the factors. The average MSE becomes

$$\begin{aligned} \overline{MSE}_1 &= \frac{\sigma^2}{2\pi N} \int_{-\pi}^{\pi} tr(v(\omega)^{-1}) d\omega + \\ & \frac{\sigma^2}{2\pi M N} \sum_{i=1}^{M} \int_{-\pi}^{\pi} \frac{H_i^*(\omega) v(\omega)^{-2} H_i(\omega)}{1 - v_i(\omega)} d\omega. \end{aligned}$$

The first term of the above equation is minimized if and only if the frame is tight (since $tr(v(\omega)^{-1}) = tr((H^*(\omega)H(\omega))^{-1}) = \sum_{k=1}^{N} 1/\lambda_k(\omega)$. We show now that the second term is minimized as well if and only if the frame is tight. We can say that

$$\sum_{i=1}^{M} \frac{H_i^*(\omega) v(\omega)^{-2} H_i(\omega)}{(1 - v_i(\omega))} \geq \sum_{i=1}^{M} \frac{v_i(\omega)^2}{1 - v_i(\omega)}. \tag{2.32}$$

Here we used Lemma A.1 from [55] which is valid for SUFs and allows us to exploit the following inequality:

$$H_i^*(\omega) v(\omega)^{-2} H_i(\omega) \geq \left(H_i^*(\omega) v(\omega)^{-1} H_i(\omega)\right)^2 = v_i(\omega)^2.$$

Since we have the following constraint:

$$\sum_{i=1}^{M} v_i(\omega) = N,$$

the equality and minimization of (2.32) occur when each term of $\sum_{i=1}^{M} \frac{v_i(\omega)^2}{1 - v_i(\omega)}$ contributes equally and when they are constant over ω. This happens if and only if the original frame is a SUTF. The minimax optimality is clear because the average MSE is minimized while keeping every term in (2.31) equal. Obviously, the maximum term of (2.31) cannot be smaller than the mean.

Chapter 3

Wavelet Footprints: Theory

3.1 Introduction

The design of a complete or overcomplete expansion that allows for compact representation of arbitrary signals is a central problem in signal processing and its applications. Parsimonious representation of data is important for compression [34]. Furthermore, achieving a compact representation of a signal also means intimate knowledge of the signal features and this can be useful for many other tasks including: denoising, classification and interpolation. From a computational analysis point of view, one can say that the problem is to build a dictionary $\mathcal{D} = \{f_i\}_{i \in I}$ of elementary functions which can well approximate any signal in a given functional class $\mathcal{F}$ with the superposition of few of its elements.

The design of a dictionary with good approximation properties, however, is not the only important element. Together with $\mathcal{D}$, one also needs to develop a fast algorithm that can efficiently find the sparsest representation of any signal $g \in \mathcal{F}$ in terms of the elements of $\mathcal{D}$. When $\mathcal{D} = \{f_i\}_{i \in I}$ is a basis, there is a unique way to express g as a linear combination of the f_i's, and this representation can be easily found computing the inner products between g and the duals of f_i's (clearly, the dual elements coincides with f_i if the dictionary is orthonormal). Despite this nice property, overcomplete dictionaries are usually preferred to basis expansions. Overcomplete dictionaries are more flexible, they can better adapt to the characteristics of the signal under consideration and this allows for sparser signal representations. Examples of overcomplete dictionaries include best basis methods or adaptive wavelet packets [16, 88]. In the case of overcomplete bases, however, it is more difficult to develop fast algorithms that find the right sparse representation of a signal in $\mathcal{F}$. Because the elements of $\mathcal{D}$ are linearly dependent, there are infinitely many ways to express g as a linear combination of the f_i's, In few cases, it is possible to arrive at sparse signal representations with linear complexity algorithms [33, 35, 46]. But, in general, the search for the sparsest signal representation is an NP-complete problem [27].

[0]This and the next chapter include research conducted jointly with Martin Vetterli [39, 41, 42, 43]

Note that techniques based on the Singular Value Decomposition (SVD) and pseudo-inverse do not yield compact signal representations [51]. Other methods like basis pursuit [12] are usually computationally intensive; matching pursuit [79], which is a greedy iterative algorithm, is computational efficient but does not converge in a finite number of iterations in general. Orthogonalized matching pursuit [28, 85] does converge in a finite number of steps, but is computationally heavy.

In this and in the next chapter, we focus on the class of piecewise smooth functions. In particular we will mostly consider piecewise polynomial signals. We propose a new representation of these functions in terms of objects which we call *footprints* and which make up an overcomplete dictionary of atoms. The footprints dictionary is built from the wavelet transform. Given a signal of interest, we first perform the wavelet transform of this signal and then the wavelet coefficients are expressed in terms of footprints. Together with the scaling coefficients, footprints can represent any piecewise polynomial signals. The main property of footprints is that they characterize efficiently the singular structures of the signal, which usually carry the main information. Wavelets are also efficient at representing singularities, however the wavelet coefficients generated by a singularity are dependent across scales. By constructing the footprint expansion on the wavelet transform we remove this dependency completely. Thus, by representing any discontinuity with the combination of a few footprints, we can get a sparser representation of the signal under consideration.

Eventhough the footprint expansion is overcomplete, it can be made locally orthogonal and this allows the use of fast algorithms to find the right sparse decomposition of the signal in terms of footprints. Alternatively, it is also possible to use matching pursuit. We show that there are situations in which matching pursuit with footprints can attain the sparsest signal representation with a finite number of iterations.

This chapter is organized as follows. Section 3.2 is meant to build up intuition about footprints. We analyze the dependency across scales of the wavelet coefficients generated by discontinuities and demonstrate a decomposition of a piecewise smooth signal into a piecewise polynomial signal and a regular residual (Theorem 3.1). This theorem will be invoked each time we will move from piecewise polynomial to piecewise smooth signals. In Section 3.3, we present the footprint expansion. We show that in the case of piecewise constant signals and Haar wavelets, footprints form a biorthogonal basis (Proposition 3.1). In the general case of piecewise polynomial signals, footprints form an overcomplete dictionary. Moreover, we show that footprints are in some cases unconditional for the class of piecewise constant and piecewise polynomial signals (Propositions 3.2, 3.4). In Section 3.4, we develop algorithms to efficiently represent piecewise polynomial signals in terms of footprints (Algorithms 3.1, 3.2) and show situations in which matching pursuit is optimal (Theorem 3.2). Finally, we draw conclusions in Section 3.5.

3.2 Dependency of the wavelet coefficients across scales

In wavelet based signal processing, it is usually important to exploit the dependency across scales of the wavelet coefficients and several efforts have been made in this direction recently, see for instance [8, 20, 78, 80, 93]. The singular structures of a signal often carry critical information and thus their efficient characterization is crucial in many signal processing tasks.

In this section, we review some of the properties of the wavelet transform, namely its ability to characterize the local regularity of a function and then we focus on the analysis of the dependency of the wavelet coefficients generated by discontinuities.

Our interest is in piecewise smooth signals, that is, in signals which are made of regular pieces. The regularity of a function is usually measured with the Lipschitz exponent [77, 78].[1]

Definition 3.1

- *A function $f(t)$ is pointwise Lipschitz $\alpha \geq 0$ at ν, if there exist a $K > 0$, and a polynomial $p_\nu(t)$ of degree $m = \lfloor \alpha \rfloor$ such that*

$$\forall t \in \mathbb{R}, |f(t) - p_\nu(t)| \leq K|t - \nu|^\alpha. \tag{3.1}$$

- *A function $f(t)$ is uniformly Lipschitz α over $[a, b]$ if it satisfies (3.1) for all $\nu \in [a, b]$, with a constant K that is independent of ν.*

- *The Lipschitz regularity of $f(t)$ at ν or over $[a, b]$ is the supremum of the α such that $f(t)$ is Lipschitz α.*

Therefore, we define a piecewise smooth function $f(t)$, $t \in [0, T]$ with $K + 1$ pieces, as follows

$$f(t) = \sum_{i=0}^{K} f_i(t) \mathbf{1}_{[t_i, t_{i+1}[}(t), \tag{3.2}$$

where $t_0 = 0$, $t_{K+1} = T$ and $f_i(t)$ is uniformly Lipschitz α over $[0, T]$. Such signals are interesting, because many signals encountered in practice can be modeled as piecewise smooth. For example, in Figure 3.2 we show a line of the image 'Peppers' (Figure 3.1), as you can see this signal is very close to a piecewise smooth signal.

Consider now an orthonormal wavelet series with scale and shift parameters m and n, respectively. We use the convention that small scales correspond to large, negative m, that is:

$$\psi_{m,n}(t) = \frac{1}{2^{m/2}} \psi(2^{-m} t - n) \qquad m, n \in \mathcal{Z}. \tag{3.3}$$

[1] The so defined Lipschitz exponent is sometimes called Hölder exponent.

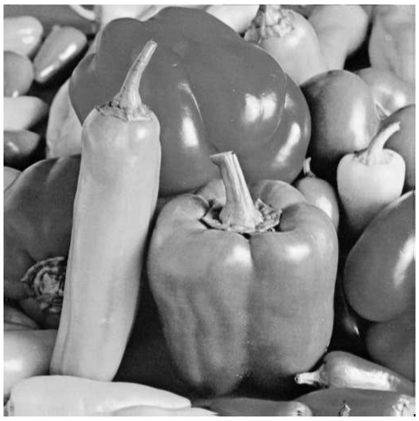

Figure 3.1: The image 'Peppers'.

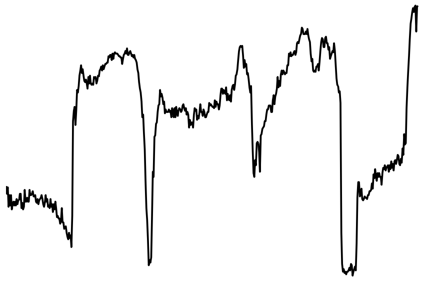

Figure 3.2: One line of 'Peppers'.

Moreover, assume that the wavelet has k vanishing moments, that is

$$\int_{-\infty}^{\infty} t^d \psi(t) dt = 0, \qquad d = 0, 1, ..k-1.$$

Then, it follows that the wavelet coefficients of a function which is uniformly Lipschitz $\alpha \leq k$ on an interval $[a, b]$ decay across scales as $2^{m(\alpha+1/2)}$ [77, 78], where m is the scale as defined in (3.3). The converse is also true: If the wavelet coefficients of a function $f(t)$ satisfies $2^{m(\alpha+1/2)}$ on an interval $[a, b]$ and $\alpha < k$ is not an integer then $f(t)$ is uniformly Lipschitz α on $[a+\epsilon, b-\epsilon]$, for any $\epsilon > 0$ [77, 78]. The (local) decay property of the wavelet coefficients is at the heart of the success of the wavelet transform in several applications. Now, because of this decay property, larger wavelet coefficients tend to be around the singular parts of a signal, that is, around points with small Lipschitz coefficients. These wavelet coefficients gather most of the energy of the original signal and for this reason we are interested in modeling their behaviour across scales. For instance, given the signal (3.2), we are interested in studying the wavelet coefficients related to the break-points t_i $i = 1, 2.., K$.

To begin our analysis, we start by considering a particular sub-class of piecewise smooth signals, namely piecewise polynomial signals. A function $p(t)$ $t \in [0, T]$ is piecewise polynomial with $K+1$ pieces if

$$p(t) = \sum_{i=0}^{K} p_i(t) \mathbf{1}_{[t_i, t_{i+1}[}(t), \tag{3.4}$$

where $t_0 = 0$, $t_{K+1} = T$ and $p_i(t) = \sum_{d=0}^{D} a_i^{(d)} t^d$ $\;i = 0, 1, ..K$ are polynomials of maximum degree D. Piecewise polynomial signals have a finite number of degrees of freedom and are easier to analyze. However, despite their simplicity they can be used to efficiently approximate piecewise smooth functions. In fact, if the piecewise polynomial approximation is chosen properly, then the approximation error shows interesting regularity properties.

Theorem 3.1 Given is a piecewise smooth signal $f(t)$ defined as in Eq. (3.2), that is, with pieces of Lipschitz regularity α. Then, there exists a piecewise polynomial signal $p(t)$ with pieces of maximum degree $d = \lfloor \alpha \rfloor$ such that the difference signal $r_\alpha(t) = f(t) - p(t)$ is uniformly Lipschitz α over $[0, T]$.

Proof: See Appendix 3.A.1.

□

Theorem 3.1 indicates a practical way to deal with piecewise smooth signals. It shows that any piecewise smooth signal $f(t)$ can be expressed as the sum of a piecewise polynomial signal and a residual which is uniformly Lipschitz α. That is

$$f(t) = p(t) + r_\alpha(t).$$

Now, since the residual is regular, it can be well represented with wavelets (the wavelet decomposition of $r_\alpha(t)$ results in small coefficients with fast decay across scales). Therefore, the only elements we need to analyze are discontinuities in the piecewise polynomial function and, in particular, the dependency across scales of the wavelet coefficients generated by these piecewise polynomial discontinuities.[2]

We start by considering the simple case of piecewise constant functions with only one discontinuity at location t_1 (i.e. $p(t) = a_0^{(0)} \mathbf{1}_{[0,t_1[}(t) + a_1^{(0)} \mathbf{1}_{[t_1,T]}(t)$) and a wavelet series with one vanishing moment and compact support. The decomposition of this signal in the wavelet basis results in zero wavelet coefficients except for the coefficients in the cone of influence of t_1. Recall that the cone of influence of t_1 in the scale-space plane is the set of points (m, n) such that t_1 is included in the support of $\psi_{m,n}(t)$ (see figure 3.3. Now, in this case, the wavelet coefficients in this cone of influence are dependent: they have only one degree of freedom. This can be easily shown recalling that a wavelet with k vanishing moments and fast decay can be written as the k^{th} order derivative of a function θ which has also a fast decay [77]. Thus, the following conditions are true: $\psi(t) = (-1)^k \frac{d^k\theta(t)}{dt^k}$ and $\psi_{m,n}(t) = (-1)^k 2^{km} \frac{d^k\theta_{m,n}(t)}{dt^k}$, where $\theta_{m,n}(t) = \frac{1}{2^{m/2}}\theta(2^{-m}t - n)$. Since the k^{th} derivative of a function is well defined in the sense of distributions, it follows that

$$\begin{aligned}
\langle p(t), \psi_{m,n}(t) \rangle &= -2^m \int_{-\infty}^{\infty} \frac{dp(t)}{dt} \theta_{m,n}(t) dt \\
&= -2^m \int_{-\infty}^{\infty} (a_1^{(0)} - a_0^{(0)}) \delta(t - t_1) \theta_{m,n}(t) dt.
\end{aligned}$$

where we used integration by parts to move the derivative from $\theta(t)$ to $p(t)$. That is $\langle p(t), \frac{d\theta(t)}{dt} \rangle = -\langle \frac{dp(t)}{dt}, \theta(t) \rangle$. Thus, if the wavelet has compact support, $\langle p(t), \psi_{m,n}(t) \rangle$ is equal to zero if $\psi_{m,n}(t)$ does not overlap t_1, and $\langle p(t), \psi_{m,n}(t) \rangle$ depends only on the difference $a_1^{(0)} - a_0^{(0)}$ otherwise. This means that, for a step edge, the wavelet behaviour across scales is deterministic. If one knows the

[2] For simplicity, we call *piecewise polynomial discontinuity* a singularity between two polynomials.

value of a single non-zero wavelet coefficient in the cone of influence of t_1, one can derive from it all the other wavelet coefficients in that cone of influence.

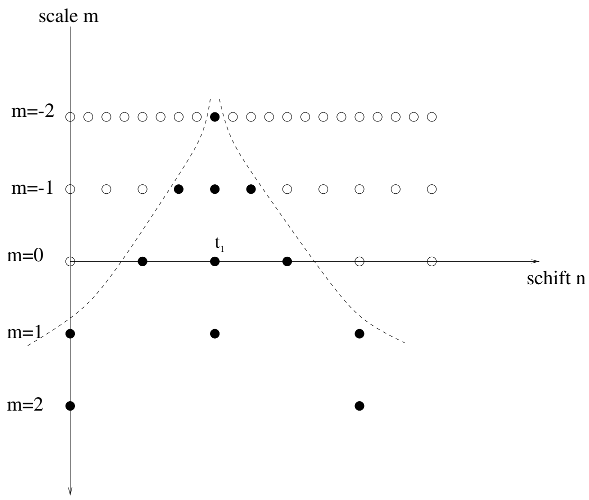

Figure 3.3: The cone of influence of t_1 consists of the scale-space points m, n such that t_1 is included in the support of $\psi_{m,n}(t)$.

This discussion generalizes to the case of piecewise polynomial signals with polynomials of maximum degree D. Consider the case of a piecewise polynomial function with one discontinuity at t_1 and polynomials $p_i(t) = \sum_{d=0}^{D} a_i^{(d)} t^d$, $i = 0, 1$. Compute the wavelet decomposition with a wavelet having $D+1$ vanishing moments and compact support. Again, the non-zero wavelet coefficients are only in the cone of influence of t_1 and we have

$$\begin{aligned} \langle p(t), \psi_{m,n}(t)\rangle = (-1)^k 2^{mk} \int_{-\infty}^{\infty} \frac{d^k p(t)}{dt^k} \theta_{m,n}(t) dt\Big|_{k=D+1} = \\ (-1)^{(D+1)} 2^{m(D+1)} \int_{-\infty}^{\infty} \sum_{d=0}^{D} c_d \delta^{(d)}(t-t_1)\theta_{m,n}(t) dt, \end{aligned} \tag{3.5}$$

where $\delta^{(d)}(t)$ is the d^{th} derivative of the Dirac δ-function and the coefficients c_d depend on the differences $(a_1^{(d)} - a_0^{(d)})$, $d = 0, 1, .., D$.[3] Thus, in the more general case, the wavelet coefficients in the cone of influence of t_1 have only $D+1$ degrees of freedom and one can determine all these wavelet coefficients by knowing only $D+1$ non-zero coefficients in that cone of influence.

In summary, the above analysis indicates that piecewise polynomial signals are well represented by wavelets, but that it is possible to model piecewise polynomial discontinuities in a more efficient way. In the next section, we present a new way to express discontinuities in piecewise polynomial signals. Together, with Theorem 3.1, this will lead to efficient algorithms to represent piecewise smooth signals. Although, we could perform this analysis in continuous time, we concentrate on the discrete-time case. This is because our final target is to

[3] To be more precise, $c_d = \sum_{i=0}^{d} t_1^{d-i}(a_1^{(D-i)} - a_0^{(D-i)})$.

develop efficient algorithms that act on discrete-time signals.

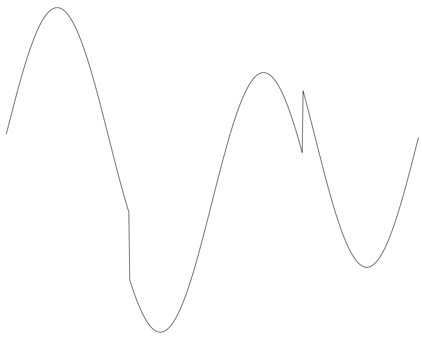

Figure 3.4: The HeaviSine function in the time domain. This function is made of three different sinusoids.

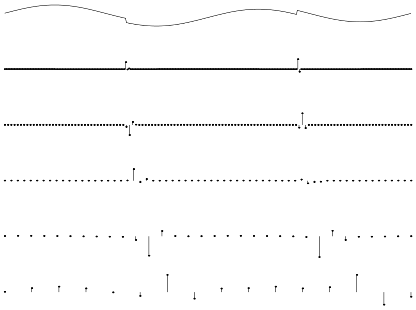

Figure 3.5: Time domain (top) and wavelet domain (bottom) representation of the HeaviSine function. Wavelet used: Daubechies with three vanishing moments.

Figure 3.6: Piecewise linear approximation of the HeaviSine function. The polynomials are chosen to guarantee continuity at the border and first order differentiability inside the interval. See Theorem 3.1 and Corollary 3.1

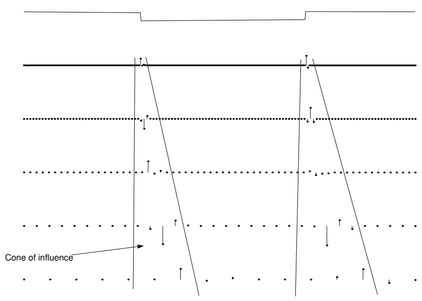

Figure 3.7: Time domain (top) and wavelet domain (bottom) representation of the piecewise linear signal. Scaling coefficients are omitted. The wavelet coefficients in the two cones of influence have only 2 degrees of freedom.

Before concluding this section, we want to analyze the border effects. Since our signals are defined on a finite interval $[0, T]$, we need to extend them outside this interval in order to perform a wavelet decomposition. Several extensions are possible [77] and some of them guarantees regularity of the signal at the borders [15]. In our formulation, we make a periodic extension. That is, we assume that they are T-periodic and that, on the period $[0, T]$, they are given by Eqns. (3.2),(3.4). Now this extension creates an artificial discontinuity at

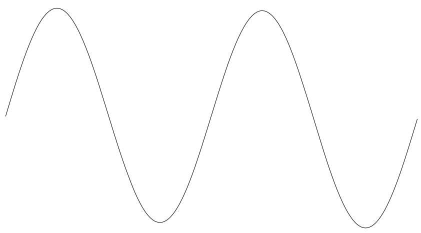

Figure 3.8: The residual $r_\alpha(t)$ in the time domain.

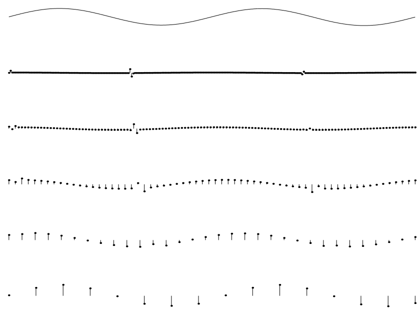

Figure 3.9: The residual $r_\alpha(t)$ in the time and wavelet domain.

$t = m{\cdot}T$, $m \in \mathbb{Z}$ and Theorem 3.1 does not guarantee that the periodic extension of $r_\alpha(t)$ is regular in $t = m \cdot T$. However, using higher order polynomials, one can easily generalize the result of Theorem 3.1 and guarantee regularity of $r_\alpha(t)$ over all $\mathbb{R}$:

Corollary 3.1 *Given is a T-periodic piecewise smooth signal $f(t)$ defined as in Eq. (3.2). There exists a T-periodic piecewise polynomial signal $p(t)$ with pieces of maximum degree $p+1$ ($p = \lfloor \alpha \rfloor$) such that the difference signal $r_\alpha(t) = f(t) - p(t)$ is uniformly Lipschitz α over $\mathbb{R}$.*

Proof: See Appendix 3.A.2.

□

Figures 3.4-3.9 show an example of approximation of a piecewise smooth function with a piecewise polynomial signal. In that example, the 'Heavisine' function is approximated with a piecewise linear signal. The 'Heavisine' function is made of three different sinusoids. The approximation is performed to guarantee first order differentiability inside $[0, T]$ and continuity at the border.

3.3 Footprint dictionaries

We now move from continuous-time to discrete-time signals and introduce the notion of footprints which are finite scale-space vectors containing all the wavelet coefficients generated by particular polynomial discontinuities.[4] We show that any piecewise polynomial discontinuity is specified by the linear combination of a few footprints, and that footprints can be interpreted as an overcomplete expansion with good approximation properties.

[4]In continuous time, one can define footprints equivalently, but they are of infinite dimension and so of little computational value.

3.3.1 Preliminaries

For our discussion, we need to introduce two discrete-time wavelet operators. The first one is an orthonormal discrete-time wavelet decomposition with J levels.[5] This decomposition can be efficiently implemented with a critically subsampled octave-band filter bank [108]. Let $\psi_{jl}[n]$ denote the wavelet function at scale j and shift l and $\phi_{Jl}[n]$ the scaling function at shift l. This wavelet operator is linear and periodically shift-variant with period 2^J. The other operator is the wavelet frame obtained by shifting out (with corresponding equivalent filters) the subsamplers in the filter bank [108]. In this case, we denote the wavelet functions at scale j and shift l with $\tilde{\psi}_{jl}[n]$ and the scaling function at shift l with $\tilde{\phi}_{Jl}[n]$. This frame is shift invariant.

The discrete-time signals we consider are N dimensional vectors defined over the interval $[0, N-1]$. Now, the wavelet operators defined above act in $l_2(\mathbb{Z})$, so we need to modify them to act on $[0, N-1]$. As anticipated in the previous section, we use a periodic extension [77], so the wavelet basis becomes

$$\psi_{jl}^{per}[n] = \sum_{k=-\infty}^{\infty} \psi_{jl}[n+kN]$$

and

$$\phi_{Jl}^{per}[n] = \sum_{k=-\infty}^{\infty} \phi_{Jl}[n+kN].$$

Recall that for any $J \leq \log_2 N$, this set of periodic wavelets forms an orthogonal basis in $l_2([0, N-1])$ [77]. The same extension applies to the wavelet frame and, in this case, we get a frame in $l_2([0, N-1])$.

Finally, our interest is in the class of piecewise smooth and piecewise polynomial signals. The discrete-time signals we consider are sampled version of the signals given in (3.2),(3.4). That is, we define a discrete-time piecewise polynomial signal $p[n]$, $n \in [0, N-1]$ as:

$$p[n] = \sum_{i=0}^{K} p_i[n] \mathbf{1}_{[k_i, k_{i+1}[}[n], \tag{3.6}$$

where $k_0 = 0, k_{K+1} = N$ and $p_i[n]$ $i = 0, 1, .., K$ is a sampled polynomial of maximum degree D. In the same way, we say that a discrete-time piecewise smooth signal $f[n]$ $n \in [0, N-1]$ is given by:

$$f[n] = \sum_{i=0}^{K} f_i[n] \mathbf{1}_{[k_i, k_{i+1}[}[n], \tag{3.7}$$

where $k_0 = 0, k_{K+1} = N$ and $f_i[n]$ $i = 0, 1, .., K$ is a sampled version of a uniformly α-Lipschitz function.

Depending on the use of a wavelet frame or a wavelet basis, we can have different footprint dictionaries. These two situations are analyzed in the next sections.

[5] For simplicity we study only the orthogonal case. However, the notion of footprints easily generalizes to the case of biorthogonal wavelets.

3.3.2 Footprints built from a wavelet basis

In this section, we construct the footprint dictionary from a wavelet basis. First, we study the simple case of piecewise constant signal and Haar wavelets. In this particular setting, the footprint dictionary $\mathcal{D}$ is a biorthogonal basis. Then we consider the more general case of piecewise polynomial signals and show that in this case $\mathcal{D}$ is always overcomplete.

Piecewise constant signals

Consider a piecewise constant signal $x[n]$, $n \in [0, N-1]$ with only one discontinuity at position k and consider a J level wavelet decomposition of this signal with a Haar wavelet:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}[n] + \sum_{j=1}^{J} \sum_{l=0}^{N/2^j-1} y_{jl} \psi_{jl}[n], \tag{3.8}$$

where $y_{jl} = \langle x, \psi_{jl} \rangle$, and $c_l = \langle x, \phi_{Jl} \rangle$.[6] Since the Haar wavelet has one vanishing moment and finite support, the non-zero wavelet coefficients of this decomposition are only in the cone of influence of k. Thus, Eq. (3.8) becomes:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}[n] + \sum_{j=1}^{J} y_{jk_j} \psi_{jk_j}[n],$$

where $k_j = \lfloor k/2^j \rfloor$. Moreover, as in the continuous-time case, all these coefficients depend only on the amplitude of the discontinuity at k. Thus, if one defines a vector which contains all of them, one can specify any other step discontinuity at k by multiplying this vector by the right factor. This consideration leads to the following definition (see also Figure 3.10):

Definition 3.2 *Given a piecewise constant signal x with only one discontinuity at position k, we call footprint $f_k^{(0)}$ the norm one scale-space vector obtained by gathering together all the wavelet coefficients in the cone of influence of k and then imposing $\|f_k^{(0)}\| = 1$. Expressed in the wavelet basis, this footprint can be written as $f_k^{(0)}[n] = \sum_{j=1}^{J} d_{jk_j} \psi_{jk_j}[n]$, where $d_{jk_j} = y_{jk_j} / \sqrt{\sum_{j=1}^{J} y_{jk_j}^2}$.*

Now, any piecewise constant signal $x[n]$ with a step discontinuity at k can be represented in terms of the scaling functions $\phi_{Jl}[n]$ and of $f_k^{(0)}$. For instance, the signal $x[n]$ in Eq. (3.8) becomes:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}[n] + \alpha f_k^{(0)}[n], \tag{3.9}$$

where $\alpha = \langle x, f_k^{(0)} \rangle = \sum_{j=0}^{J} y_{jk_j} d_{jk_j}$. The above discussion can be repeated for any other step discontinuity at different locations and for each location l we

[6] Note that we are assuming N to be a power of 2, in this way, a wavelet decomposition with a Haar wavelet does not suffer from border effects.

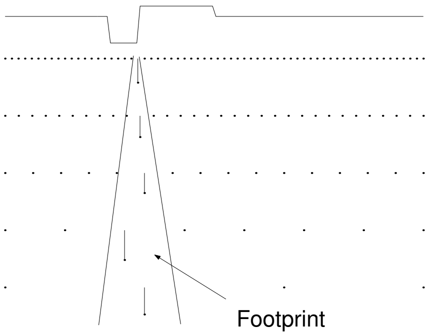

Figure 3.10: Time domain (top) and wavelet domain (bottom) representation of the footprint $f_k^{(0)}$ with $N = 128$, $J = 5$ and $k = 41$. Notice that, except for the case $J = \log_2 N$ (N being a power of 2), a footprint does not look like a pure step edge function.

have a different footprint $f_l^{(0)}$. Call $\mathcal{D} = \{f_k^{(0)}, k = 0, 1, .., N-1\}$ the complete dictionary of footprints. Some of the properties of this dictionary depend on the number J of wavelet decomposition levels. For instance, just like the wavelet basis, footprints are shift variant unless the shift is equal to $m \cdot 2^J$, $m \in \mathbb{Z}$. That is:

$$f_l^{(0)}[n] = f_k^{(0)}[n + l - k], \quad \text{if} \quad l - k = m \cdot 2^J, \quad m \in \mathbb{Z}. \tag{3.10}$$

In addition, footprints are orthogonal to the scaling functions, but the orthogonality condition between footprints depends on the number J of wavelet decomposition levels. Assume $k = k' + m \cdot 2^J$ and $l = l' + n \cdot 2^J$ and assume $l \geq k$. We have:

$$\begin{aligned} \langle f_k^{(0)}, f_l^{(0)} \rangle &= 0 && \text{if} \quad m \neq n, \\ \langle f_k^{(0)}, f_l^{(0)} \rangle &= \sqrt{\tfrac{k'(2^J - l')}{l'(2^J - k')}} && \text{otherwise.} \end{aligned} \tag{3.11}$$

So, footprints related to neighbouring discontinuities are biorthogonal (see examples in Figures 3.11 and 3.12).

Finally, consider again Eq. (3.10). Since $f_0^{(0)}[n] = 0$, it follows that $f_{m \cdot 2^J}^{(0)}[n] = 0$. Thus, $\mathcal{D}$ contains only $N - N/2^J$ elements. Moreover, we have that:

Proposition 3.1 *The elements of $\mathcal{D}$ together with the $N/2^J$ scaling functions $\phi_{Jl}[n]$, $l = 0, 1, ..N/2^J - 1$ form a biorthogonal basis for $l_2([0, N-1])$.*

Proof: See Appendix 3.A.3 □

So, any signal $x[n]$, $n \in [0, N-1]$ can be expressed in terms of footprints and scaling functions. In particular, if x is piecewise constant with K discontinuity, together with the scaling functions, K footprints are sufficient to represent

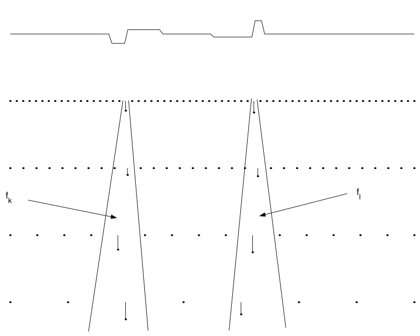

Figure 3.11: Orthogonal footprints. Time domain (top) and wavelet domain (bottom) representation of $f_k + f_l$. In this example $N = 128$, $J = 4$, $k = 37$ and $l = 77$. Two footprints are orthogonal if their cones of influence do not overlap.

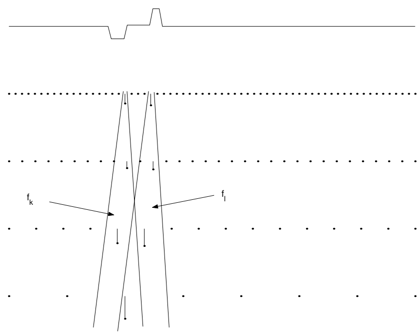

Figure 3.12: Biorthogonal footprints. Time domain (top) and wavelet domain (bottom) representation of $f_k + f_l$. In this example $N = 128$, $J = 4$, $k = 37$ and $l = 45$. Two footprints are biorthogonal if their cones of influence overlap.

it. This can be shown by noticing that a piecewise constant signal with only one discontinuity can be expressed in terms of one footprint (see Eq. (3.9)) and piecewise constant signals with K discontinuities are given by the superposition of K piecewise constant signals with only one discontinuity. Therefore, the footprint representation of a signal x with K discontinuities at positions $k_1, k_2, .., k_K$ is given by:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}[n] + \sum_{i=1}^{K} \alpha_i f_{k_i}^{(0)}[n]. \tag{3.12}$$

Note how this representation is sparser than the corresponding representation in a wavelet basis which requires J times more wavelets than footprints. The problem of finding the discontinuity locations and the correct values α_i in (3.12) will be treated in detail in Section 3.4.

Finally, one may wonder if any object generated with the superposition of K footprints is piecewise constant with a number of discontinuities equal to the number of footprints. That is, are $\mathcal{D}$ and the scaling functions an unconditional basis for the class of piecewise constant signals? It turns out that this property is satisfied when $J = \log_2 N$ (N being a power of 2).

Proposition 3.2 *For $J = \log_2 N$, the scaling function $\phi_{J0}[n]$ and the $N-1$ footprints $f_k^{(0)}$, $k = 1, 2, .., N-1$ represent a biorthogonal basis which is unconditional for the class of piecewise constant signals defined over* $[0, N-1]$.

Proof: The biorthogonality comes from Proposition 3.1. We only need to show that this basis is unconditional. That is, assume that x is a piecewise constant signal with discontinuities at $k_1, k_2, ...k_K$ and consider its representation in terms of footprints:

$$x[n] = c_0\phi_{J0}[n] + \sum_{i=1}^{K} \alpha_i f_{k_i}^{(0)}.$$

We need to show that for any set of coefficients $\hat{\alpha}_i$ satisfying $|\hat{\alpha}_i| \leq |\alpha_i|$, the signal

$$\hat{x}[n] = c_0\phi_{J0}[n] + \sum_{i=1}^{K} \hat{\alpha}_i f_{k_i}^{(0)}$$

is still piecewise constant with discontinuity locations $k_1, k_2, .., k_K$. This can be seen noticing that, for $J = \log_2 N$, $\phi_{J0}[n]$ is a constant function and $f_k^{(0)}[n]$ is piecewise constant with one discontinuity at k ($k = 1, 2, .., N-1$). Therefore, any linear combination of f_{k_i} $k_i \in \{k_1, k_2, ...k_K\}$ gives a piecewise constant signal with discontinuity locations $k_1, k_2, .., k_K$.

$\square$

Piecewise polynomial signals

We now generalize the above discussion to the case of discrete-time piecewise polynomial signals with polynomials of maximum degree D. We show that in this context each discontinuity is represented by $D+1$ footprints rather than one footprint.

Consider orthogonal wavelets with at least $D+1$ vanishing moments and compact support L and consider a piecewise polynomial signal $x[n]$ with only one discontinuity at k. Its J level wavelet decomposition with periodic wavelets is:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{j=1}^{J} \sum_{l=0}^{N/2^j-1} y_{jl} \psi_{jl}^{per}[n]. \tag{3.13}$$

First, notice that the periodic extension of the wavelet basis creates a second discontinuity at location zero and that this is a polynomial discontinuity. Thus, the non-zero wavelet coefficients of this expansion are only in the cone of influence of k and in the cone of influence of zero. Assume, for now, that $0 \ll k \ll N$ and $2^J \ll N$ so that there are no wavelet coefficients in common between these two cones of influence. We can write

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{j,l \in I_0} y_{jl} \psi_{jl}^{per}[n] + \sum_{j,l \in I_k} y_{jl} \psi_{jl}^{per}[n] \tag{3.14}$$

where I_k is the set of indices (j, l), which are in the cone of influence of k and I_0 is the set of indices (j, l), which are in the cone of influence of zero. It is easy to verify that there are no more than $J \times (L-1)$ wavelet coefficients in each cone of influence. From Eq. (3.5) we know that the wavelet coefficients in each of these cones of influence have only $D+1$ degrees of freedom. Thus, we want to find a set of $D+1$ footprints that can characterize these coefficients. To build this set of footprints, we resort to time-domain analysis.

The class of piecewise polynomial signals with one discontinuity at a fixed position $k \in [0, N-1]$ forms a linear space of dimension $2(D+1)$ and a possible basis for that space is represented by the following vectors:

$$P^{(d)}[n] = n^d, \qquad d = 0, 1, .., D, \qquad n \in [0, N-1];$$

$$T_k^{(d)}[n] = \mathbf{1}_{[k,N-1]}(n-k+1)^d, \qquad d = 0, 1, .., D, \qquad n \in [0, N-1].$$

We can express these signals in a wavelet basis and we have:

$$P^{(d)}[n] = \textstyle\sum_{l=0}^{N/2^J-1} b_l^{(d)} \phi_{Jl}^{per}[n] + \sum_{j,l \in I_0} p_{jl}^{(d)} \psi_{jl}^{per}[n]$$

$$T_k^{(d)}[n] = \textstyle\sum_{l=0}^{N/2^J-1} c_{kl}^{(d)} \phi_{Jl}^{per}[n] + \sum_{j,l \in I_0} t_{jl}^{(d)} \psi_{jl}^{per}[n] + \sum_{j,l \in I_k} t_{jl}^{(d)} \psi_{jl}^{per}[n], \tag{3.15}$$

where we have used the fact that the non-zero wavelet coefficients of $P^{(d)}[n]$ are only in I_0, while the non-zero wavelet coefficients of $T_k^{(d)}$ are in the cones of influence of k and zero. Now, any signal $x[n]$ in this class can be written as:

$$x[n] = \sum_{d=0}^{D} \alpha_0^{(d)} P^{(d)}[n] + \sum_{d=0}^{D} \alpha_k^{(d)} T_k^{(d)}[n]. \tag{3.16}$$

Therefore, combining Eqns. (3.14),(3.15),(3.16) and considering only the elements in I_k we have:

$$\sum_{j,l \in I_k} y_{jl} \psi_{jl}^{per}[n] = \sum_{d=0}^{D} \alpha_k^{(d)} \sum_{j,l \in I_k} t_{jl}^{(d)} \psi_{jl}^{per}[n]. \tag{3.17}$$

Call $f_k^{(d)}[n] = \sum_{j,l \in I_k} t_{jl}^{(d)} \psi_{jl}^{per}[n]$ the scale-space vector gathering the $J \times (L-1)$ wavelet coefficients generated by the discontinuity in $T_k^{(d)}$. Eq. (3.17) shows that the wavelet coefficients generated by any polynomial discontinuity at k are characterized by a linear combination of $f_k^{(d)}$. This indicates that the wavelet coefficients in the cone of influence of a polynomial discontinuity have only $D+1$ degrees of freedom and proves that these coefficients lie on a subspace of dimension $D+1$. The vectors $f_k^{(d)}$ $d = 0, 1, .., D$ span that subspace and can represent the set of footprints we are looking for. However, it is always better to have orthogonal bases, so the footprints that we will consider are obtained by applying a Gram-Schmidt orthogonalization process to $f_k^{(d)}$. Thus, from the above discussion it follows that:

Proposition 3.3 *Given a piecewise polynomial signal with polynomials of maximum degree D and with one discontinuity at position k, the $J \times (L-1)$ non-zero wavelet coefficients in the cone of influence of that discontinuity lie on a subspace of dimension $D+1$.*

Definition 3.3 *We call footprints $f_k^{(d)}$ $d = 0, 1, .., D$ the elements of an orthogonal vector basis which spans the subspace of dimension $D+1$ generated by a polynomial discontinuity at k. Footprints are obtained by gathering together all the non-zero wavelet coefficients generated by the discontinuity in $T_k^{(d)}$ $d = 0, 1, .., D$ and then imposing the two following conditions:*

$$\|f_k^{(d)}\| = 1 \quad d = 0, 1, .., D;$$

$$\langle f_k^{(i)}, f_k^{(j)} \rangle = \delta_{ij} \quad i = 0, 1, .., D; \quad j = 0, 1, .., D.$$

With this set of footprints, we can characterize any polynomial discontinuity at position k. In particular, Eq. (3.13) can be written as:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{j,l \in I_0} y_{jl} \psi_{jl}^{per}[n] + \sum_{d=0}^{D} \alpha^{(d)} f_k^{(d)}[n]$$

where $\alpha^{(d)} = \langle x, f_k^{(d)} \rangle$, $d = 0, 1, .., D$.[7] With a similar analysis, we can create a different set of $D+1$ footprints to characterize a polynomial discontinuity at a different location. To characterize any polynomial discontinuity (including the discontinuity in zero), we need a dictionary $\mathcal{D} = \{f_k^{(d)}, d = 0, 1, .., D; k = 0, 1, ..N-1\}$ of $(D+1)N$ footprints. With this dictionary of footprints and with the scaling functions we can represent any piecewise polynomial signal. In particular, a signal x with K discontinuities at locations $k_1, k_2, ..k_K$ is given by

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{K} \sum_{d=0}^{D} \alpha_i^{(d)} f_{k_i}^{(d)}[n], \tag{3.18}$$

where $k_0 = 0$ is the discontinuity due to the periodic extension. Note again how this representation is sparser than the corresponding representation in a wavelet basis.

As for the case of piecewise constant signals, footprints are orthogonal to the scaling functions, but footprints related to close discontinuities are biorthogonal. In particular, we have: $\langle f_l^{(d)}, f_k^{(c)} \rangle = 0$ for $|l-k| > (L-1) \cdot 2^J$. Moreover, footprints are periodically shift-invariant of period 2^J, hence:

$$f_l^{(d)}[n] = f_k^{(d)}[n+l-k], \quad \text{if} \quad l-k = m \cdot 2^J, \quad m \in \mathbb{Z}, \, d = 0, 1, .., D. \tag{3.19}$$

It is also of interest to note that, due to the periodic extension, the coefficients $\alpha_i^{(d)}$ in (3.18) are not independent. For instance, for $D = 0$ it follows that $\alpha_0^0 = -\sum_{i=1}^{K} w_{k_i}^0 \alpha_{k_i}^0$, where the weights $w_{k_i}^0$ depend on the normalization in Definition 3.3 (without normalization, it would be $w_{k_i}^0 = 1$). In general, we have that:

$$\alpha_0^0 = -\sum_{i=1}^{K} \sum_{d=0}^{D} w_{k_i}^d \alpha_{k_i}^d, \tag{3.20}$$

[7] In case of biorthogonal wavelets it would be: $\alpha^{(d)} = \langle x, \hat{f}_k^{(d)} \rangle$ with $\hat{f}_k^{(d)} = \sum_{j,l \in I_k} d_{jl} \hat{\psi}_{jl}^{per}[n]$ where $\hat{\psi}_{jl}^{per}$ is the dual of ψ_{jl}^{per}.

where the weights $w^d_{k_i}$ depend on the orthogonalization process in Definition 3.3.[8] For this reason, to extend Proposition 3.2 to the piecewise polynomial case we need to consider the constraint in (3.20). Thus:

Proposition 3.4 For $J = \log_2 N$, any linear combination of $\phi^{per}_{J0}[n]$ and of the $(D+1)N$ footprints $f^{(d)}_k$ which verifies (3.20) gives a piecewise polynomial signal.

Proof: We want to show that given a piecewise polynomial signal $x[n]$ represented as in Eq. (3.18); for any set of coefficients $\hat{\alpha}^{(d)}_i$ satisfying $|\hat{\alpha}^{(d)}_i| \leq |\alpha^{(d)}_i|$ and Eq. (3.20), the signal

$$\hat{x}[n] = c_0\phi^{per}_{J0}[n] + \sum_{i=0}^{K}\sum_{d=0}^{D}\hat{\alpha}^{(d)}_i f^{(d)}_{k_i}[n],$$

is still piecewise polynomial with discontinuity locations $k_1, k_2, .., k_K$. This can be proved using arguments similar to that of Proposition 3.2. The scaling function $\phi_{J0}[n]$ is constant. Moreover, any pair of footprints: $\hat{\alpha}^{(0)}_0 f^{(0)}_0[n] + f^{(d)}_{k_i}[n]$, with α^0_0 satisfying (3.20) represent a piecewise polynomial signal with one discontinuity at $k_i \in \{k_1, k_2, ..k_K\}$. Therefore, any linear combination of these pairs of footprints and of $\phi_{J0}[n]$ gives a piecewise polynomial signal with discontinuities at $k_1, k_2, ..k_K$.

□

Strictly speaking, Proposition 3.4 shows conditions under which any linear combination of footprints leads to piecewise polynomial signals, but it does not prove that footprints are an unconditional expansion for the class of piecewise polynomial signals. However, in the rest of the chapter, for simplicity, we will say that dictionaries of footprints satisfying the hypotheses of Proposition 3.4 are unconditional for the class of piecewise polynomial signals.

3.3.3 Footprints built from a wavelet frame

We have constructed a dictionary of $(D+1)N$ footprints that can efficiently represent piecewise polynomial signals. However, this representation, like the wavelet transform, is not shift-invariant. In some settings, it is useful to have a shift-invariant dictionary. Such a dictionary can be constructed by simply replacing the wavelet basis with the wavelet frame. In particular, let $x[n]$ be a piecewise constant signal with only one discontinuity at k. We have that

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l\tilde{\phi}^{per}_{Jl}[n] + \sum_{j,l\in\tilde{I}_0} y_{jl}\tilde{\psi}^{per}_{jl}[n] + \sum_{j,l\in\tilde{I}_k} y_{jl}\tilde{\psi}^{per}_{jl}[n]$$

[8] The easiest way to verify this property is by noticing that if we take the $(D+1)$-th order derivative of a periodic discrete-time piecewise polynomial signal, the sum of the resulting non-zero coefficients is always zero.

where we have again used the fact that the non-zero coefficients are only in the cones of influence of k and 0. In this case, the cone of influence of k contains $J \times (L_j - 1)$ coefficients, where L_j is the length of the equivalent filter at level j. Moreover, y_{jl} is given by $y_{jl} = \langle x, \hat{\psi}_{jl}^{per} \rangle$, where $\{\hat{\phi}, \hat{\psi}\}$ is the dual frame of $\{\tilde{\phi}, \tilde{\psi}\}$. Now, the shift invariant footprint related to location k is given by

$$\tilde{f}_k^{(0)}[n] = \sum_{j,l \in \tilde{I}_k} d_{jl} \tilde{\psi}_{jl}^{per}[n],$$

where $d_{jl} = y_{jl} / \sqrt{\sum_{j,l \in \tilde{I}_k} y_{jl}^2}$. The other footprints can be designed in the same way, and it follows that

$$\tilde{f}_k^{(0)}[n] = \tilde{f}_0^{(0)}[n+k].$$

That is, all footprints are shifted versions of one footprint. If J is chosen such that $\langle \tilde{f}_0^{(0)}, \tilde{f}_k^{(0)} \rangle = 0$, $x[n]$ can be expressed as:

$$x[n] = \sum_{l=0}^{N/2^J - 1} c_l \tilde{\phi}_{Jl}^{per}[n] + \alpha_0 \tilde{f}_0^{(0)} + \alpha_k \tilde{f}_k^{(0)}$$

where $\alpha_k = \langle x, \hat{f}_k^{(0)} \rangle$ and $\hat{f}_k^{(0)}[n] = \sum_{j,l \in \tilde{I}_k} d_{jl} \hat{\psi}_{jl}^{per}[n]$. In the same way, we can design the footprint dictionary related to higher order polynomials. In this case, one has to consider the signals $T_k^{(d)}$ and their transform with a wavelet frame. The footprints $\tilde{f}_k^{(d)}$ at location k are obtained following the same procedure given by Definition 3.3. Finally, given the dictionary $\mathcal{D} = \{\tilde{f}_k^{(d)}, d = 0, 1.., D; k = 0, 1, ..N - 1\}$, we have that:

$$\tilde{f}_k^{(d)}[n] = \tilde{f}_0^{(d)}[n+k], \qquad d = 0, 1, .., D.$$

As in the previous case, any piecewise polynomial signal can be expressed in terms of this dictionary and we have:

$$x[n] = \sum_{l=0}^{N/2^J - 1} c_l \tilde{\phi}_{Jl}^{per}[n] + \sum_{i=0}^{K} \sum_{d=0}^{D} \alpha_i^{(d)} \tilde{f}_{k_i}^{(d)}[n]. \tag{3.21}$$

3.4 Representation algorithms

In the previous sections, we have constructed different dictionaries of footprints according to the kind of wavelets involved (i.e. wavelet bases or wavelet frames) and to the class of signals considered (i.e. piecewise constant or piecewise polynomial signals). The main characteristics of these dictionaries are summarized in Table 3.1.

Before focusing on the representation algorithms, we want to mention that the space required to store these footprints dictionaries is not high, since it grows only linearly with the size N of the signal. In particular, in the case of shift-variant footprints the required storage space is of the order of $(L-1) \cdot J \cdot (D+1) \cdot 2^J$ coefficients, where $(L - 1) \cdot J$ are the wavelet coefficients contained in each

	piecewise constant signals and Haar wavelet	piecewise polynomial signals and wavelet basis	piecewise polynomial signals and wavelet frame
dictionary properties	complete, shift variant, unconditional if $J = \log_2 N$.	overcomplete, shift variant, unconditional if $J = \log_2 N$ and Eq. (3.20) is verified.	overcomplete, shift invariant, unconditional if $J = \log_2 N$ and Eq. (3.20) is verified.

Table 3.1: Footprint dictionaries.

footprint and $(D+1) \cdot 2^J$ are the number of footprints one has to store, since the others are shifted version of those (see Eq. (3.19)). Therefore, when $J = \log_2 N$ (worst case), we have that the required memory space grows like $N \log_2 N$. Similar results apply to the case of shift invariant footprints.

Now, we need to develop a fast and robust algorithm that can find the right representation of piecewise polynomial signals in terms of footprints. The algorithms that we present are valid for any of the families of footprints in Table 3.1. However, for the sake of simplicity, we study only the case of footprints built from a wavelet basis, the extension to the wavelet frame being straightforward.

Consider a piecewise polynomial signal x with polynomials of degree D and with K discontinuities at $k_1, k_2, .., k_K$. We have seen that this signal can be written as:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{K} \sum_{d=0}^{D} \alpha_i^{(d)} f_{k_i}^{(d)}[n]. \tag{3.22}$$

Thus, our target is to develop a fast and robust algorithm that can find this representation of x. In our analysis we do not consider the scaling functions, since coefficients c_l in (3.22) are always given by: $c_l = \langle x, \phi_{Jl}^{per} \rangle$.

We present two different approaches, the first one is a variation of the traditional matching pursuit algorithm. We show that in particular situations, this method can arrive at the correct representation of $x[n]$ in a finite number of iterations. The second approach is in spirit similar to matching pursuit, but it uses the property that the orthogonality condition between footprints depends on the number J of decomposition levels. We show that, with a slight increase in complexity, this second algorithm always attains the correct signal representation with $\lceil K/2 \rceil$ iterations, where K is the number of discontinuities in the signal.

3.4.1 Matching pursuit with footprints

Matching pursuit (MP) [79] is a greedy iterative algorithm which derives sparse approximated representations of a signal by successive refinements.

Let $\mathcal{D} = \{f_k\}_{k \in I}$ be a complete or overcomplete dictionary of unit norm vectors. In the first iteration, matching pursuit projects the signal of interest x on the vectors $f_k \in \mathcal{D}$ and chooses the vector f_{k_0} which maximizes $|\langle x, f_{k_0} \rangle|$.

The process is then iterated on the residual

$$R_x^1 = x - \langle x, f_{k_0}\rangle f_{k_0}.$$

Let $R_x^0 = x$. At the i^{th} iteration, the algorithm chooses $f_{k_{i-1}} \in \mathcal{D}$ such that

$$|\langle R_x^{i-1}, f_{k_{i-1}}\rangle| = \max_{k\in I} |\langle R_x^{i-1}, f_k\rangle| \tag{3.23}$$

and computes the residual

$$R_x^i = R_x^{i-1} - \langle R_x^{i-1}, f_{k_{i-1}}\rangle f_{k_{i-1}}.$$

Since R_x^i is orthogonal to $f_{k_{i-1}}$,we have that

$$\|R_x^{i-1}\|^2 = \|R_x^i\|^2 + |\langle R_x^{i-1}, f_{k_{i-1}}\rangle|^2.$$

Thus, by selecting $f_{k_{i-1}}$ such that $|\langle R_x^{i-1}, f_{k_{i-1}}\rangle|$ is maximized, we are sure to minimize the norm $\|R_x^i\|$ of the residual. Moreover, it is possible to prove that $\|R_x^i\|$ converges exponentially to 0 when i tends to infinity [79]. After M iterations, the approximated signal is given by:

$$\hat{x}[n] = \sum_{i=0}^{M-1} \langle R_x^i, f_{k_i}\rangle f_{k_i}.$$

The vector f_{k_i} selected by matching pursuit at the i^{th} step is, in general, not orthogonal to the previously selected vectors $\{f_{k_m}\}_{0\leq m<i}$. So, when subtracting the projection $\langle R^i x, f_{k_i}\rangle f_k$ from R_x^i, the algorithm reintroduces new components in the directions of $\{f_{k_m}\}_{0\leq m<i}$. For this reason, even in finite-dimensional space, matching pursuit is not guaranteed to converge in a finite number of iterations. One way to speed convergence is to use an orthogonalized version of matching pursuit which at each step orthogonalizes the direction of the chosen vector with respect to the previously selected vectors [28, 85]. In finite-dimensional space, this algorithm has the advantage of converging in a finite number of steps. However, it is computationally intensive (there is an orthogonalization process to compute at each iteration) so it will not be considered hereafter.

Assume that $\mathcal{D}$ is the footprint dictionary and that $x[n]$ is a piecewise polynomial signal. Matching pursuit can be used to approximate x with $\mathcal{D}$. We know that the wavelet coefficients generated by a single polynomial discontinuity at k lie on a subspace of size $D+1$ and that this subspace is spanned by the footprints $f_k^{(d)}$, $d = 0, 1, .., D$ (Proposition 3.3). Hence, instead of using the usual matching pursuit which projects the signal on single vectors, we employ a subspace pursuit, where the signal is projected on different sub-spaces.

In the first iteration, for each possible discontinuity location $k \in [0, N-1]$, the algorithm computes the $D+1$ inner products $\langle x, f_k^{(d)}\rangle$, $d = 0, 1, .., D$ and choose the location k_0 such that $\sum_{d=0}^{D} |\langle x, f_{k_0}^{(d)}\rangle|^2$ is maximum. Then, x can be written as its projection onto $f_{k_0}^{(d)}$, $d = 0, 1, .., D$ and a residual R_x^1:

$$x = \sum_{d=0}^{D} \langle x, f_{k_0}^{(d)}\rangle f_{k_0}^{(d)} + R_x^1.$$

Since footprints related to the same discontinuity location are orthogonal (Proposition 3.3), we can write:

$$\|x\|^2 = \|R_x^1\|^2 + \sum_{d=0}^{D} |\langle x, f_{k_0}^{(d)} \rangle|^2.$$

So, by choosing k_0 such that $\sum_{d=0}^{D} |\langle x, f_{k_0}^{(d)} \rangle|^2$ is maximum, we minimize the norm of the error R_x^1. The algorithm is then iterated on the residual.

Note that, for $D = 0$ (piecewise constant signals) the subspace pursuit reduces to the traditional matching pursuit. The subspace pursuit with the footprint dictionary has the same drawback as matching pursuit, that is, it is not guaranteed to converge in a finite number of steps. However, there exists situations in which it can obtain the exact representation of x in a finite number of iterations. The basic intuition is that, if the discontinuity location are sufficiently far apart, then the wavelet coefficients generated by the signal discontinuities do not overlap and so matching pursuit converges in a finite number of steps. In particular, for the case of piecewise constant signals and the Haar wavelet, we can state this exactly.

Theorem 3.2 *Given is a piecewise constant signal with K discontinuities at $k_1, k_2, ..k_K$. If the distance between the two closest discontinuities is larger than 2^J, matching pursuit with footprints obtains the exact footprint representation of x in K iterations.*

Proof: This result can be easily proved by verifying that matching pursuit chooses the correct footprint at each iteration. Assume, for instance, that x has two discontinuities at location k_1 and k_2, that is $x = \alpha_1 f_{k_1}^{(0)} + \alpha_2 f_{k_2}^{(0)}$ and assume that $\alpha_1 \geq \alpha_2$ and $k_2 - k_1 \geq 2^J$. Using Eq. (3.11), one can easily see that at the first iteration, matching pursuit chooses the vector $f_{k_1}^{(0)}$, since $|\langle x, f_{k_1}^{(0)} \rangle| = |\alpha_1|$ is maximum. The residual is $R_x^1 = x - \alpha_1 f_{k_1}^{(0)} = \alpha_2 f_{k_2}^{(0)}$, so in the second iteration matching pursuit will choose $f_{k_2}^{(0)}$. This result, trivially, extends to the case of more than two discontinuities.

$\square$

In the more general case of higher order polynomials, we have numerical evidence that matching pursuit converges in a finite number of steps if the minimum distance between the two closest discontinuity is larger than $(L-1) \cdot 2^J$.

3.4.2 Adaptive depth footprint pursuit

The basic intuition behind Theorem 3.2 is that the number of decomposition level J should be chosen according to the distance between discontinuities. If J is chosen properly, one can get the correct representation of x in a few iterations, with a very simple method like matching pursuit. The problem is that we do not know a priori the discontinuity locations. Therefore, we propose a new algorithm, where we first find the discontinuity locations and then estimate the footprint coefficients related to those discontinuities. For simplicity, we concentrate on the case of piecewise constant signals and Haar wavelets. Assume, for instance, that x has K discontinuities at positions $k_1, k_2, .., k_K$:

$$x[n] = \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}[n] + \sum_{i=1}^{K} \alpha_i f_{k_i}^{(0)}[n] \tag{3.24}$$

and that the footprint dictionary is chosen with $J = \log_2 N$: $\mathcal{D} = \{f_k^{(0)} = \sum_{j=1}^{J} d_{jk_j} \psi_{jk_j}; k = 0, 1, ..N-1\}$.[9] The discontinuity locations $k_1, k_2, ..k_K$ are found in the following way:

Algorithm 3.1 (locations estimation)

1. *Compute the dual basis of $\mathcal{D}$ and call $\tilde{f}_k^{(0)}$ $k = 1, 2, .., N-1$ the elements of this dual basis.*[10]
2. *Compute the inner products $\langle x, \tilde{f}_k^{(0)} \rangle$, $k = 1, 2, .., N-1$. The discontinuity locations correspond to the indexes of the basis' elements which have non-zero inner products with x.*

Now that $k_1, k_2, ..k_K$ are known, we need to evaluate the coefficients α_i. The footprint coefficients are evaluated with an iterative method which is, in spirit, similar to matching pursuit. At each iteration, we choose J such that the footprints related to the two closest discontinuities are orthogonal, we estimate the footprints coefficients of these two discontinuities and iterate the process on the residual. At each iteration, we do not project the signal directly on the two closest footprints, instead we compute the two dual footprints and project the signals on these two dual elements. The complete algorithm operates as follows:

Algorithm 3.2 (coefficient estimation)

1. *Call $\mathcal{K} = \{k_1, k_2, ..k_K\}$ the set of estimated discontinuity locations.*
2. *Assume that k_{m-1} and k_m are the two closest discontinuities in $\mathcal{K}$. Choose $J_1 = \lfloor \log_2(k_m - k_{m-1}) \rfloor$.*
3. *Call $\hat{f}_{k_m}^{(0)}$ the sub-footprint obtained by considering only the first J_1 elements of $f_{k_m}^{(0)}$. That is: $\hat{f}_{k_m}^{(0)} = \sum_{j=1}^{J_1} d_{jk_{mj}} \psi_{jk_{mj}}$. Define, in the same way, the sub-footprint $\hat{f}_{k_{m-1}}^{(0)}$.*

[9] It is worth pointing out that, in this case, $\mathcal{D}$ is a biorthogonal basis, so the exact representation of x can be found using the dual basis of $\mathcal{D}$. However, this solution is not robust to noise and does not generalize to piecewise polynomial signals. Therefore, it will not be considered here.

[10] It is of interest to emphasize that this dual basis turns out to be a first order derivative. See Appendix 3.A.4.

4. *The sub-footprint* $\hat{f}^{(0)}_{k_{m-1}}$ *is orthogonal to* $f^{(0)}_k$, $k \in \mathcal{K} - \{k_{m-1}\}$ *and verifies:* $\langle \hat{f}^{(0)}_{k_{m-1}}, f^{(0)}_{k_{m-1}} \rangle = \|\hat{f}^{(0)}_{k_{m-1}}\|^2$*. Likewise, the sub-footprint* $\hat{f}^{(0)}_{k_m}$ *is orthogonal to* $f^{(0)}_k$, $k \in \mathcal{K} - \{k_m\}$ *and* $\langle \hat{f}^{(0)}_{k_m}, f^{(0)}_{k_m} \rangle = \|\hat{f}^{(0)}_{k_m}\|^2$*. Thus, the contributions* α_{m-1}, α_m *are given by:*

$$\alpha_{k_{m-1}} = \frac{1}{\|\hat{f}^{(0)}_{k_{m-1}}\|} \langle x, \frac{\hat{f}^{(0)}_{k_{m-1}}}{\|\hat{f}^{(0)}_{k_{m-1}}\|} \rangle, \qquad \alpha_{k_m} = \frac{1}{\|\hat{f}^{(0)}_{k_m}\|} \langle x, \frac{\hat{f}^{(0)}_{k_m}}{\|\hat{f}^{(0)}_{k_m}\|} \rangle. \quad (3.25)$$

5. *Remove* k_{m-1}, k_m *from* $\mathcal{K}$ *and subtract the two estimated contributions from the original signal:* $R^1_x = x - \alpha_{k_{m-1}} f^{(0)}_{k_{m-1}} - \alpha_{k_m} f^{(0)}_{k_m}$*.*

6. *If* $\mathcal{K}$ *is not empty iterate the process on the residual, otherwise stop.*

Notice that, since at each iteration we estimate two footprint coefficients, the algorithm ends after $\lceil K/2 \rceil$ iterations. So, we are guaranteed that the algorithm converges after a finite number of steps. The interesting point of this algorithm is that, at each iteration, it is very easy to find the pair of dual footprints related to the footprints under consideration. There are two other advantages of this algorithm compared to matching pursuit. First, at each iteration, we chose the largest possible J_1 such that the footprints related to the two closest discontinuities are orthogonal. Since multiscale operators like footprints are robust to noise, by choosing J_1 as large as possible, we increase this robustness. Second, the signal is reconstructed in terms of the footprint dictionary with $J = \log_2 N$, this dictionary is unconditional for the class of piecewise constant signals (Proposition 3.2), thus, we are sure that the reconstructed signal is still piecewise constant. This is a useful property when the signal to estimate has been corrupted by noise.

The algorithm generalizes to the piecewise polynomial case. The discontinuities are estimated with a $D+1$ order derivative, while the coefficients $\alpha^{(d)}_i$ are evaluated with a procedure similar to that presented above. That is, at each iteration, we choose J such that the footprints related to the two closest discontinuities are orthogonal, we estimate the footprints coefficients of these two discontinuities and iterate the process on the residual. Finally, the coefficient α^0_0 is computed using equation (3.20). As for the previous case, since $J = \log_2 N$, Proposition 3.4 guarantees that the reconstructed signal is always piecewise polynomial.

3.5 Conclusions

In this chapter, we have presented a new way to model the dependency across scales of wavelet coefficients with atoms we called footprints. Footprints form an overcomplete dictionary and are efficient at representing the singular structures of a signal. Moreover, with footprints, it is possible to get a sparser representation of piecewise smooth signals than with wavelets. Together with the simplicity of the algorithms involved, this seems to indicate that footprints might be useful in several signal processing tasks, namely denoising, deconvolution and compression. Applications of footprints to signal processing problems will be the topic of the next chapter.

Appendix 3.A Proofs

3.A.1 Proof of Theorem 3.1

Consider, first, a piecewise smooth signal $f(t)$ $t \in [0,T]$ with only two pieces. That is, $f(t) = f_1(t)\mathbf{1}_{[0,t_1[} + f_2(t)\mathbf{1}_{[t_1,T]}$ and $f_1(t), f_2(t)$ are uniformly Lipschitz α over $[0,T]$. Recall that, if a function f is uniformly Lipschitz $\alpha > p$ in the neighbourhood of ν, then it is necessarily p times continuously differentiable in that neighbourhood.[11] Moreover, the polynomial $p_\nu(t)$ in Eq. (3.1) is the Taylor expansion of f at ν. Now, since $f_1(t), f_2(t)$ are uniformly Lipschitz α over $[0,T]$, then they are necessarily p times continuously differentiable on that interval with $p = \lfloor \alpha \rfloor$. Call $p(t) = p_{t_1^-}(t)\mathbf{1}_{[0,t_1[} + p_{t_1^+}(t)\mathbf{1}_{[t_1,T]}$, $t \in [0,T]$ the piecewise polynomial signal whose two pieces $p_{t_1^-}(t)$ and $p_{t_1^+}(t)$ are given by

$$p_{t_1^-}(t) = f_1(t_1) + f_1'(t_1)(t-t_1) + ... + \frac{f_1^{(p)}(t_1)}{p!}(t-t_1)^p$$

and

$$p_{t_1^+}(t) = f_2(t_1) + f_2'(t_1)(t-t_1) + ... + \frac{f_2^{(p)}(t_1)}{p!}(t-t_1)^p.$$

That is, $p_{t_1^-}(t)$ and $p_{t_1^+}(t)$ are the Taylor expansions of $f(t)$ about t_1 taken from the left and from the right of t_1. Now, the signal $r_\alpha(t) = f(t) - p(t)$ is p times continuously differentiable in $[0,T] - \{t_1\}$ and in t_1 it verifies:

$$\begin{aligned}
\lim_{t\to t_1^-} r_\alpha(t) &= \lim_{t\to t_1} f_1(t) - p_{t_1^-}(t) = \lim_{t\to t_1^+} r_\alpha(t) = \lim_{t\to t_1} f_2(t) - p_{t_1^+}(t) = 0, \\
\lim_{t\to t_1^-} r_\alpha^{(l)}(t) &= \lim_{t\to t_1} f_1^{(l)}(t) - p_{t_1^-}^{(l)}(t) \\
&= \lim_{t\to t_1^+} r_\alpha^{(l)}(t) = \lim_{t\to t_1} f_2^{(l)}(t) - p_{t_1^+}^{(l)}(t) = 0, \quad l = 1, 2, .., p.
\end{aligned}$$

Therefore, $r_\alpha(t)$ is p times continuously differentiable on the entire interval $[0,T]$. So, it is uniformly Lipschitz $\alpha' > p$ on that interval. The remaining step is to prove that $\alpha' = \alpha$. This is clearly true for all points away from t_1, we only need to prove that $r_\alpha(t)$ is α-Lipschitz in t_1. Using the definition of Lipschitz regularity, we have that for $t \leq t_1$

$$f(t) = p_{t_1^-}(t) + \epsilon_1(t) \quad \text{with} \quad |\epsilon_1(t)| \leq K_1|t-t_1|^\alpha;$$

and for $t \geq t_1$

$$f(t) = p_{t_1^+}(t) + \epsilon_2(t) \quad \text{with} \quad |\epsilon_2(t)| \leq K_2|t-t_1|^\alpha.$$

[11] The converse is also true. That is, a function which is p times continuously differentiable in the neighbourhood of ν is Lipschitz $\alpha' > p$ at ν.

Now, since $r_\alpha(t) = f(t) - p(t)$, we can write $|r_\alpha(t)| \leq K_1|t - t_1|^\alpha$, for $t \leq t_1$ and $|r_\alpha(t)| \leq K_2|t - t_1|^\alpha$, for $t \geq t_1$. Thus, if we call $K = \max\{K_1, K_2\}$, then in the neighbourhood of t_1 we have that $|r_\alpha(t)| \leq K|t - t_1|^\alpha$, which proves that $r_\alpha(t)$ is Lipschitz α in t_1. This completes the proof.

The generalization of this result to the case of a piecewise smooth signal $f(t)$ with K discontinuities at locations $t_1, t_2, .., t_K$ is straightforward. Assume that $f(t)$ is given by (3.2). Call $p_i(t) = p_{t_i^-}(t)\mathbf{1}_{[0,t_i[} + p_{t_i^+}(t)\mathbf{1}_{[t_i,T]}$, $i = 1, 2, .., K$; the piecewise polynomial signal whose two pieces $p_{t_i^-}(t)$ and $p_{t_i^+}(t)$ are given by

$$p_{t_i^-}(t) = f_{i-1}(t_i) + f'_{i-1}(t_i)(t - t_i) + ... + \frac{f_{i-1}^{(p)}(t_i)}{p!}(t - t_i)^p$$

and

$$p_{t_i^+}(t) = f_i(t_i) + f'_i(t_i)(t - t_i) + ... + \frac{f_i^{(p)}(t_i)}{p!}(t - t_i)^p.$$

That is, $p_{t_i^-}(t)$ and $p_{t_i^+}(t)$ are the Taylor expansions of $f(t)$ about t_i taken from the left and from the right of t_i. Then, as we proved before, the signal $f(t) - p_i(t)$ is p times continuously differentiable in t_i. So, the piecewise polynomial signal $p(t) = \sum_{i=1}^{K} p_i(t)$ is such that $r_\alpha(t) = f(t) - p(t)$ is p times continuously differentiable in $t_1, t_2, .., t_K$ and, thus, $r_\alpha(t)$ is p times continuously differentiable on the entire interval $[0, T]$ and uniformly Lipschitz $\alpha' > p$ on that interval. Finally, as in the previous example one can show that $\alpha' = \alpha$.

3.A.2 Proof of Corollary 1.1

Assume that $f(t)$ is a T-periodic function defined over the period $[0, T]$ as in Eq. (3.2) and assume that $f(t)$ has K discontinuities at locations $t_1, t_2, .., t_K$. Following the same procedure as in Theorem 3.1, one can construct a T-periodic piecewise polynomial signal $p(t)$ with pieces of degree $p = \lfloor \alpha \rfloor$ such that the signal $h(t) = f(t) - p(t)$ is uniformly Lipschitz α over $]0, T[$. Now, due to the periodicity, $h(t)$ has still a discontinuity in $t = m \cdot T$, $m \in \mathbb{Z}$. This function can be made p times continuously differentiable in $t = m \cdot T$ by constructing a T-periodic polynomial $\hat{p}(t)$ such that:

$$\hat{p}(T^-) - \hat{p}(0^+) = h(T^-) - h(0^+) \tag{3.26}$$

and

$$\hat{p}^{(l)}(T^-) - \hat{p}^{(l)}(0^+) = h^{(l)}(T^-) - h^{(l)}(0^+), \qquad l = 1, 2, .., p. \tag{3.27}$$

Assume, for instance, that we only want first order differentiability, that is

$$\hat{p}(T^-) - \hat{p}(0^+) = h(T^-) - h(0^+) \tag{3.28}$$

and

$$\hat{p}^{(1)}(T^-) - \hat{p}^{(1)}(0^+) = h^{(1)}(T^-) - h^{(1)}(0^+). \tag{3.29}$$

Then a T periodic polynomial $\hat{p}(t) = a_0 + a_1 t + a_2 t^2$, meets the two constrains in (3.28),(3.29), if a_0 is arbitrary and a_1 and a_2 satisfy

$$a_1 = \tfrac{h(T^-)-h(0^+)}{T} - \tfrac{h^{(1)}(T^-)-h^{(1)}(0^+)}{2},$$

$$a_2 = \tfrac{h^{(1)}(T^-)-h^{(1)}(0^+)}{2T}.$$

Moreover, it is easy to verify that the function $r_\alpha(t) = h(t) - \hat{p}(t)$ is T-periodic and in $C^1(\mathbb{R})$.

In general, equations (3.26),(3.27) impose $p+1$ constraints, and we need a polynomial of degree $p+1$ to meet them. This is because the constant factor a_0 in the polynomial is not subject to any of the constraints in (3.26),(3.27). Finally, notice that $p(t) + \hat{p}(t)$ is piecewise polynomial with polynomials of maximum degree $p+1$ and $r_\alpha(t) = f(t) - p(t) - \hat{p}(t)$ is periodic of period T and is in $C^p(\mathbb{R})$. Moreover, with the same argument as in the Theorem 3.1, one can show that $r_\alpha(t)$ is uniformly Lipschitz α over $\mathbb{R}$. This concludes the proof.

3.A.3 Proof of Proposition 3.1

We are considering signals in $\mathbb{R}^N$ and the union of footprints and scaling functions gives N elements. We need to show that this set of elements is complete. This is equivalent to showing that there exists no $x[n]$ with $\|x\| > 0$ such that it has a zero expansion, that is, such that:

$$\sum_k |\langle x, f_k^{(0)} \rangle|^2 + \sum_l |\langle x, \phi_{Jl} \rangle|^2 = 0. \tag{3.30}$$

We prove this for the case $J = \log_2 N$, noting that with the same method one can prove it for any J. Consider the representation of x in terms of the wavelet basis:

$$x[n] = c_0 \phi_{J0}[n] + \sum_{j=1}^{J} \sum_{l=0}^{N/2^j-1} y_{jl} \psi_{jl}[n]. \tag{3.31}$$

Eq. (3.30) already implies that the scaling coefficient $c_0 = \langle x, \phi_{J0} \rangle$ is zero. We will show that if Eq. (3.30) is true then also all the wavelet coefficients of x are zero and so it must be $\|x\| = 0$. Recall that, since $J = \log_2 N$, there is only one wavelet coefficient at level J, two wavelet coefficients at level $J-1$ and so on. First consider the footprint $f_k^{(0)}[n] = \sum_{j=1}^{J} d_{jk_j} \psi_{jk_j}[n]$ with $k = 2^J/2$ and the corresponding inner product $\langle x, f_k^{(0)} \rangle$. One can easily verify that the only non-zero coefficient d_{jk_j} of $f_k^{(0)}[n]$ is the one at scale J. That is: $f_k^{(0)}[n] = d_{Jk_J} \psi_{Jk_J}[n] = d_{J0} \psi_{J0}[n]$ where in the last equality we have used the fact that $k_J = 0$. Thus, we have that $\langle x, f_k^{(0)} \rangle = y_{J0} d_{J0}$ and this inner product is equal to zero only if $y_{J0} = 0$. Consider now the footprint $f_k^{(0)}$ with $k = 2^J/4$. In this case, $f_k^{(0)} = d_{J0}\psi_{J0}[n] + d_{(J-1)0}\psi_{(J-1)0}[n]$, that is, $f_k^{(0)}$ has only two non-zero coefficients d_{jk_J} at scales J and $J-1$. So, we have that

$\langle x, f_k^{(0)} \rangle = y_{J0} d_{J0} + y_{(J-1)0} d_{(J-1)0}$. Since we have seen that $y_{J0} = 0$, this second inner product is zero only if $y_{(J-1)0} = 0$. In the same way, but with the footprint related to position $2^J/4 + 2^J/2$ we can prove that $y_{(J-1)1} = 0$. So the wavelet coefficients at scales J and $J-1$ are zero. The same analysis can be repeated at each scale and in conclusion we have that condition (3.30) implies that all the wavelet coefficients of x are zero. Therefore, x must be the zero vector.

3.A.4 The footprints dual basis

We consider vector in $\mathbb{R}^{\mathbb{N}}$ with N being a power of 2. We have seen that, for $J = \log_2 N$, the scaling function ϕ_{J0} and the $N-1$ footprints $f_k^{(0)}$ form a biorthogonal basis. Now, we want to show that, except for a normalizing factor, the dual basis is given by:

$$\begin{aligned} &\tilde{f}_k^{(0)}[n] = \delta(n-k) - \delta(n-(k-1)), \qquad k = 1, 2, .., N; \\ &\tilde{\phi}_{J0}[n] = \phi_{J0}[n]. \end{aligned} \tag{3.32}$$

First notice that the vectors in (3.32) are linearly independent. We only need to show that they verify the biorthogonality conditions. That is:

$$\begin{aligned} \langle \tilde{f}_j^{(0)}, f_k^{(0)} \rangle &= \delta_{jk}, \\ \langle \tilde{f}_j^{(0)}, \phi_{J0} \rangle &= 0 \end{aligned}$$

and

$$\begin{aligned} \langle \tilde{\phi}_{J0}, f_k^{(0)} \rangle &= 0, \\ \langle \tilde{\phi}_{J0}, \phi_{J0} \rangle &= 1 \end{aligned}$$

We have that $\tilde{\phi}_{J0}[n]$ is orthogonal to $f_k^{(0)}$'s by construction and $\langle \tilde{\phi}_{J0}[n], \phi_{J0}[n] \rangle = 1$. Moreover, for $J = \log_2 N$, $f_k^{(0)}$ is piecewise constant with one discontinuity at k. Therefore, the inner product

$$\langle \tilde{f}_j^{(0)}[n], f_k^{(0)}[n] \rangle = f_k^{(0)}[j] - f_k^{(0)}[j-1]$$

is different from zero only for $j = k$. In fact, in all the other cases, we have $f_k^{(0)}[j] = f_k^{(0)}[j-1]$. Therefore, the vectors in (3.32) verify the biorthogonal conditions. The following example will clarify the result.

Example. We assume $N = 4$ and represent the footprint basis in matrix form. We have that (as usual, rows represent the basis vectors):

$$F = \begin{bmatrix} \cdots f_1^{(0)} \cdots \\ \cdots f_2^{(0)} \cdots \\ \cdots f_3^{(0)} \cdots \\ \cdots \phi_{J0} \cdots \end{bmatrix} = \begin{bmatrix} -\sqrt{3}/2 & \sqrt{3}/6 & \sqrt{3}/6 & \sqrt{3}/6 \\ -1/2 & -1/2 & 1/2 & 1/2 \\ -\sqrt{3}/6 & -\sqrt{3}/6 & -\sqrt{3}/6 & \sqrt{3}/2 \\ 1/2 & 1/2 & 1/2 & 1/2 \end{bmatrix}.$$

One can immediately verify that the inverse of F is given by (basis vectors are in columns):

$$F^{-1} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \tilde{f}_1^{(0)} & \tilde{f}_2^{(0)} & \tilde{f}_3^{(0)} & \phi_{J0} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} = \begin{bmatrix} -\sqrt{3}/2 & 0 & 0 & 1/2 \\ \sqrt{3}/2 & -1 & 0 & 1/2 \\ 0 & 1 & -\sqrt{3}/2 & 1/2 \\ 0 & 0 & \sqrt{3}/2 & 1/2 \end{bmatrix}.$$

Thus, the dual footprints follow the expression given in (3.32).

Chapter 4

Wavelet Footprint: Applications

4.1 Introduction

After the introduction of the notion of footprints in the previous chapter, we now examine practical applications. We focus on three main signal processing problems for which wavelets were successful, namely denoising, deconvolution and compression. We present algorithms based on the footprint expansion and show that these methods can further improve wavelet based algorithms. The main characteristic of the footprint methods is that they can deal efficiently with discontinuities. In Sections 4.2 and 4.3, we study denoising and deconvolution respectively. Compression is examined in Section 4.4, where we prove (Theorem 4.1) that at high rates footprints based compression algorithms perform as well as ideal methods. In Section 4.5, we assess the performance of the proposed algorithms numerically and we give concluding remarks in Section 4.6.

4.2 Denoising

The term denoising usually refers to the removal of noise from a corrupted signal. In the typical problem formulation, the original signal x has been corrupted by additive noise. One observes $z[n] = x[n] + e[n]$ where $e[n]$ are independent and identically distributed (i.i.d.) zero mean Gaussian variables with variance σ^2 and the original signal is deterministic and independent of the noise. The goal of the denoising algorithm is to obtain an estimate $\hat{x}$ of the original signal which minimizes a risk function, usually the mean square error $E[\| x - \hat{x} \|^2]$. The wavelet based denoising algorithm introduced by Donoho and Johnstone [32] simply shrinks the wavelet coefficients. That is, it sets all wavelet coefficients smaller than a threshold to zero and keeps the coefficients above the threshold (hard thresholding) or shrinks them by a fixed amount (soft thresholding). The threshold is usually set to $T = \sigma\sqrt{2 \ln N}$, where N is the size of the signal [32].

A limit of this approach is that it does not exploit the dependency across scales of the wavelet coefficients. Thus, to overcome this limit, we apply a threshold in the footprint domain rather than the wavelet domain. Doing so, we better exploit the dependency of the wavelet coefficients across scales. As a matter of fact, denoising in the footprint domain is equivalent to applying a vector threshold in the wavelet domain rather than a scalar threshold as in the usual methods.

Assume that the observed noisy signal is $z[n] = x[n] + e[n]$ and that $x[n]$ is piecewise polynomial. We can express piecewise polynomial signals in terms of footprints, thus our denoising system attempts to estimate this footprint representation from the observed noisy version $z[n]$. The estimation procedure follows the same steps as algorithms 3.1, 3.2. That is, one first estimates the discontinuity locations and then evaluates the footprint coefficients. Since we only observe a noisy version of the signal, we need to slightly modify these two steps to make them more robust to noise. Again, for simplicity, we focus on piecewise constant signals. The discontinuity locations are estimated in the following way:

Algorithm 4.1 (location estimation, noisy case)

1. Choose a dictionary $\mathcal{D} = \{f_k^{(0)} = \sum_{j=1}^{J} d_{jk_j}\psi_{jk_j}; k = 0, 1, ..N-1\}$ of footprints with $J = \log_2 N$. This dictionary represents a biorthogonal basis.
2. Compute the dual basis of $\mathcal{D}$ and call $\tilde{f}_k^{(0)}$ $k = 1, 2, .., N-1$ the elements of this dual basis.
3. Compute the $N-1$ inner products $\langle z, \tilde{f}_k^{(0)} \rangle$ $k = 1, 2, .., N-1$.
4. Consider as discontinuity locations the ones related to the inner products larger than the threshold $T_k = \|\tilde{f}_k^{(0)}\|T$. That is, if $|\langle z, \tilde{f}_k^{(0)} \rangle| \geq T_k$, then assume that there is a discontinuity at location k. T is the universal threshold equal to $\sigma\sqrt{2\ln N}$ [32].

We have a set of estimated discontinuity locations: $\hat{k}_1, \hat{k}_2, .., \hat{k}_{\hat{K}}$. The problem is that, due to the noise, this estimation can have errors. Thus, this possibility must be considered in the next step where the footprints coefficients are evaluated.

Algorithm 4.2 (coefficient estimation, noisy case)

1. Call $\hat{\mathcal{K}}$ the set of estimated discontinuity locations.
2. Choose $J_1 = \lfloor \log_2(\hat{k}_m - \hat{k}_{m-1}) \rfloor$, where $\hat{k}_{m-1}, \hat{k}_m$ are the two closest discontinuity locations in $\hat{\mathcal{K}}$.
3. For each possible location $k \in [\hat{k}_{m-1}, \hat{k}_m]$ compute the inner product $\langle z, \frac{\hat{f}_k^{(0)}}{\|\hat{f}_k^{(0)}\|} \rangle$, where $\hat{f}_k^{(0)}$ is the sub-footprint obtained by considering only the first J_1 wavelet coefficients of $f_k^{(0)}$.

4. Choose the location k_1 such that $|\langle z, \frac{\hat{f}^{(0)}_{k_1}}{\|\hat{f}^{(0)}_{k_1}\|}\rangle|$ is maximum.

5. If
$$|\langle z, \frac{\hat{f}^{(0)}_{k_1}}{\|\hat{f}^{(0)}_{k_1}\|}\rangle| \geq T, \tag{4.1}$$
then compute the residual:
$$R_z^1 = z - \frac{1}{\|\hat{f}^{(0)}_{k_1}\|}\langle z, \frac{\hat{f}^{(0)}_{k_1}}{\|\hat{f}^{(0)}_{k_1}\|}\rangle f^{(0)}_{k_1}.$$

6. Iterate step 4-5 on the residual until condition (4.1) is not verified anymore.

7. Once condition (4.1) is not verified anymore, remove the two discontinuity locations $\hat{k}_{j-1}$, $\hat{k}_j$ from $\hat{\mathcal{K}}$.

8. If $\hat{\mathcal{K}}$ is empty, stop. Otherwise go to step 2.

Finally, the estimated signal $\hat{x}$ is:
$$\hat{x} = \langle z, \phi_{J0}\rangle\phi_{J0}[n] + \sum_{m=0}^{M-1} \frac{1}{\|\hat{f}^{(0)}_{k_m}\|}\langle R_z^m, \frac{\hat{f}^{(0)}_{k_m}}{\|\hat{f}^{(0)}_{k_m}\|}\rangle f^{(0)}_{k_m}[n], \tag{4.2}$$
where M is the total number of iterations and $R_z^0 = z$.

First, notice that, since the footprints $f^{(0)}_{k_m}$ in Eq. (4.2) are obtained taking a wavelet transform with $J = \log_2 N$ decomposition levels, we are sure that the estimated signal $\hat{x}$ is piecewise constant like x (Proposition 3.2). This is an important property, because traditional denoising algorithms suffer from the presence of artifacts around discontinuities (pseudo-Gibbs effects). The advantage of denoising in the footprints basis is that these artifacts are automatically eliminated.

Notice that, at each iteration, given the two closest discontinuity locations $\hat{k}_{m-1}, \hat{k}_m$, we run a complete matching pursuit algorithm on the interval $[\hat{k}_{m-1}, \hat{k}_m]$ (step 3-6 of the algorithm). In this way, if there is a discontinuity that has not been detected in the discontinuity estimation step, it can be found in this step. This is the main difference between the noiseless and noisy version of the algorithm.

The proposed denoising algorithm generalizes to piecewise polynomial signals. In this case, given the interval $[\hat{k}_{m-1}, \hat{k}_m]$, instead of running matching pursuit on this interval, we run the subspace pursuit presented in Section 3.4.1. That is, for each $k \in [\hat{k}_{m-1}, \hat{k}_m]$, we project the set of the corresponding noisy wavelet coefficients on the right sub-space, we choose the largest projection and if this projection is larger than the threshold, we keep it. All the other previous considerations apply also to the piecewise polynomial case.

Denoising in the wavelet domain suffers from the lack of shift invariance of the wavelet basis. One way to overcome this limit is to use a denoising method called cycle-spinning [17]. For a range of shifts, cycle spinning shifts the noisy signal, denoises each shifted version and, finally, unshift and average the denoised signals. Since footprints suffer from the same lack of shift invariance as wavelets, one can use the idea of cycle spinning to reduce this shift dependency. The only difference between cycle spinning with wavelets and cycle spinning with footprints is that, in this second case, each shifted version of the signal is denoised with footprints (algorithms 4.1, 4.2) rather than wavelets. The only limit of this approach is that we cannot guarantee anymore that the denoised signal is piecewise polynomial. That is, Proposition 3.2 and 3.4 do not apply to this case. In Section 4.5, we consider both methods (denoising with footprints and cycle spinning with footprints) and compare them with the equivalent wavelet based algorithms.

4.3 Deconvolution

In its simplest form, the deconvolution problem can be stated as follows. The original unknown signal $x[n]$ is blurred by a convolution operator $h[n]$ and corrupted by additive white Gaussian noise. One searches for a good estimate of $x[n]$ from the observed signal

$$y[n] = h[n] * x[n] + e[n]. \tag{4.3}$$

Either $h[n]$ is known, or it has to be estimated (blind deconvolution). In most cases, $h[n]$ behaves as a low-pass filter and does not have a bounded inverse, for this reason such a deconvolution problem is usually called ill-posed.

There is a large number of methods that provide possible solutions to the deconvolution problem [3, 8, 18, 31, 68, 69, 82, 95]. Some linear approaches, for instance, use regularized inverse filters like Wiener filters to invert the effect of $h[n]$. The limit of these methods is that they are suited for regular signals, but they fail when signals are characterized by the presence of discontinuities. Indeed, discontinuities are the part of a signal which is most corrupted by the convolution filter $h[n]$. According to the amount of prior information about the signal, one can develop more sophisticated methods which force the estimated signal to meet some prior constraints. Those approaches include methods based on projection onto convex sets and methods based on iterative filters [95, 18, 69]. The main drawback of these techniques is that they are computationally intensive. Finally, wavelet based deconvolution methods have become popular recently [31, 68, 3, 82, 8], mostly because they deal well with singular structures in the signals. In our approach, we use the footprint expansion to further improve wavelet based techniques. We assume that $h[n]$ is known.

Consider the case where $x[n]$ is piecewise polynomial. We know that it can be written as a linear combination of footprints (see Eq. (3.18)). Thus, by replacing $x[n]$ with its footprints representation, Eq. (4.3) becomes

$$\begin{aligned} y[n] &= \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{K}\sum_{d=0}^{D} \alpha_i^{(d)} h[n] * f_{k_i}^{(d)}[n] + e[n] \\ &= \sum_{l=0}^{N/2^J-1} c_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{K}\sum_{d=0}^{D} \alpha_i^{(d)} \bar{f}_{k_i}^{(d)}[n] + e[n], \end{aligned}$$

where, in the last equality, we assumed $\bar{f}_{k_i}^{(d)}[n] = h[n] * f_{k_i}^{(d)}[n]$. That is, $y[n]$ is given by a linear combination of blurred footprints $\bar{f}_{k_i}^{(d)}[n]$ plus the additive white noise $e[n]$. Call $\bar{\mathcal{D}} = \{\bar{f}_k^{(d)}; k = 0,1,..,N-1; d = 0,1,..,D\}$ the dictionary of blurred footprints. In our deconvolution algorithm, we first attempt to remove the white noise and then the blurring effect. The noise is removed using the denoising algorithm 4.1, 4.2, but we use the blurred dictionary $\bar{\mathcal{D}}$ to perform denoising rather than $\mathcal{D}$. The deblurring process then simply consists in replacing the $\bar{f}_{k_i}^{(d)}$'s with the corresponding non-blurred footprints. The complete algorithm can be summarized as follows

Algorithm 4.3 (Deconvolution of piecewise polynomial signals)

1. Consider the dictionary of blurred footprints $\bar{\mathcal{D}} = \{\bar{f}_k^{(d)}; k = 0,1,..,N-1; d = 0,1,..,D\}$. Remove the noise in $y[n]$ using algorithms 4.1, 4.2 and assuming $\bar{\mathcal{D}}$ as the reference footprints dictionary.
2. Call the denoised signal $\hat{y}[n] = \sum_{l=0}^{N/2^J-1} \hat{c}_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{\hat{K}}\sum_{d=0}^{D} \hat{\alpha}_i^{(d)} \bar{f}_{\hat{k}_i}^{(d)}[n]$. The deconvolved signal $\hat{x}[n]$ is given by $\hat{x}[n] = \sum_{l=0}^{N/2^J-1} \hat{c}_l \phi_{Jl}^{per}[n] + \sum_{i=0}^{\hat{K}}\sum_{d=0}^{D} \hat{\alpha}_i^{(d)} f_{\hat{k}_i}^{(d)}[n]$, where we have simply replaced $\bar{f}_{\hat{k}_i}^{(d)}[n]$ with $f_{\hat{k}_i}^{(d)}[n]$.

If $x[n]$ is piecewise smooth, we use a *two step* deconvolution algorithm. The procedure of this algorithm is based on the result of Theorem 3.1, which says that $x[n]$ can be written as the sum of a piecewise polynomial signal $p[n]$ and a regular function $r[n]$. That is, $x[n] = p[n] + r[n]$. Therefore, the observed signal $y[n]$ can be written as: $y[n] = h[n] * p[n] + h[n] * r[n] + e[n]$. The aim of the algorithm is to estimate the two contributions $p[n]$ and $r[n]$ in two different phases. The complete algorithm operates in the following way (we assume that $h[n]$ is known)

Algorithm 4.4 (Two step deconvolution)

1. Estimate the piecewise polynomial behaviour underlying $y[n]$ with the deconvolution algorithm 4.3. Call the estimated signal $\hat{p}[n]$.
2. Compute the residual $\tilde{r}[n] = y[n] - h[n] * \hat{p}[n]$.
3. Deconvolve the residual with a Wiener filter $g[n]$: $\hat{r}[n] = g[n] * \tilde{r}[n]$
4. The estimated signal is: $\hat{x}[n] = \hat{p}[n] + \hat{r}[n]$.

4.4 Compression

Wavelets are widely used in compression. The reason is that wavelets have very good approximation properties for representing certain classes of signals like piecewise smooth signals. While good approximation properties are necessary for good compression, it might not be enough. In compression, one has to consider the costs corresponding to indexing and compressing the retained elements in the approximation and independent coding of these coefficients might be inefficient [107].

Consider a piecewise smooth signal defined as in Eq. (3.2), that is, a function with pieces that are α-Lipschitz regular and with a finite number of discontinuities. It was shown in [14] that standard wavelet based schemes such as zerotrees [93] can achieve the following distortion-rate performance:

$$D(R) \leq c_1 R_s^{-2\alpha} + c_2 \sqrt{R_e} 2^{-c_3 \sqrt{R_e}}, \tag{4.4}$$

where $R = R_s + R_e$ and R_e are the bits used to quantize the wavelet coefficients generated by the discontinuities, while R_s are the bits used to code the wavelet coefficients corresponding to the smooth parts of the signal. Now, suppose that the signal is piecewise polynomial. Then the wavelet coefficients related to the smooth parts of the signal are exactly zero, and so there is no need to use any bits to code them. The distortion of a wavelet based scheme becomes

$$D(R) \leq c_2 \sqrt{R} 2^{-c_3 \sqrt{R}}. \tag{4.5}$$

However, a direct approach to compression of piecewise polynomial signals, based on an oracle telling us where discontinuities are, will lead to $D(R) \leq c_4 2^{-c_5 R}$ [86] and such behaviour is achievable using dynamic programming [86]. This large gap between ideal performance given by the scheme based on dynamic programming and wavelet performance is mainly due to the independent coding of the wavelet coefficients across scales. Statistical modeling [20] of such dependencies can improve the constants in (4.5), but going from $\sqrt{R}$ to R in the exponent requires taking the deterministic behaviour of wavelet coefficients across scales at singularities into account. This is well done using footprints, which thus close the gap with the ideal performance:

Theorem 4.1 *Consider piecewise polynomial signals with polynomials of maximum degree D and no more than K discontinuities. A coder, which represents these signals in the footprints basis and which scalar quantizes the discontinuity locations and the footprint coefficients achieves:*

$$D(R) \leq c_6 2^{-c_7 R}. \tag{4.6}$$

Proof: See Appendix 4.A.

□

Thus, this theorem shows that, in case of piecewise polynomial signals, footprints significantly improve performance of wavelet coders. Footprints can be used for piecewise smooth signals too. Theorem 3.1 shows that a piecewise smooth signal can be separated into two contributions a piecewise polynomial part (call it $p[n]$) and a residual $r[n]$ which is regular (α-Lipschitz over $\mathbb{R}$). Now, $p[n]$ can be compressed with footprints and this coder achieves (4.6). The residual $r[n]$ can be compressed with any other coder which achieves [14]

$$D(R) \leq c_8 R^{-2\alpha}. \tag{4.7}$$

It is worth noticing that, because of the regularity of $r[n]$, the performance in (4.7) can be achieved with a simple coder based on linear approximation of $r[n]$ in a wavelet or Fourier basis [14]. Combining (4.6) and (4.7) shows that a two stage compression algorithm based on footprints and on linear approximation of the residual achieves

$$D(R) \leq c_8 R_s^{-2\alpha} + c_6 2^{-c_7 R_e}. \tag{4.8}$$

Comparing (4.4) and (4.8), we can seen that this coder does not change the asymptotics of the distortion-rate function of wavelet coders ($\sim c_8 R_s^{-2\alpha}$). But, by coding the discontinuities efficiently, this coder reaches the asymptotic behaviour more rapidly. Finally, notice that for this last performance, the underlying assumption is that the encoder knows in advance the signal to code, in this way it can separate the polynomial and the smooth parts of the signal. In the experimental results, we will show that a realistic encoder can obtain similar performance without knowing the signal characteristics in advance.

4.5 Numerical Experiments

In this section, we compare footprints with wavelet-based methods on several examples. Our purpose is to show that footprints are a versatile tool and that we can get good results in a variety of different applications.

4.5.1 Denoising

For denoising, we consider only piecewise polynomial signals. In Table 4.1, we compare the performance of our denoising systems against a classical hard thresholding algorithm [32] and against cycle-spinning [17]. In this experiment we consider piecewise linear signals with no more than three discontinuities. The performance is analyzed in function of the size N of the signal. The table clearly shows that denoising with footprints outperforms the hard thresholding system, while cycle-spinning with footprints outperforms traditional cycle-spinning. In Figures 4.1 and 4.2, we show examples of the denoising algorithms on piecewise linear and piecewise quadratic signals. We can see that signals denoised with footprints present better visual quality since they do not suffer from pseudo-Gibbs effects.

N	64	128	256	512
Footprints	17.4dB	19.9dB	21.9dB	24.5dB
Hard thresholding	15.1dB	17.1dB	18.7dB	21.4dB
Cycle spinning	17.9dB	20dB	22dB	24.5dB
Cycle-spin footprints	18.4dB	20.4dB	22.5dB	24.9dB

Table 4.1: Denoising with footprints. Piecewise linear signals with no more than three discontinuities.

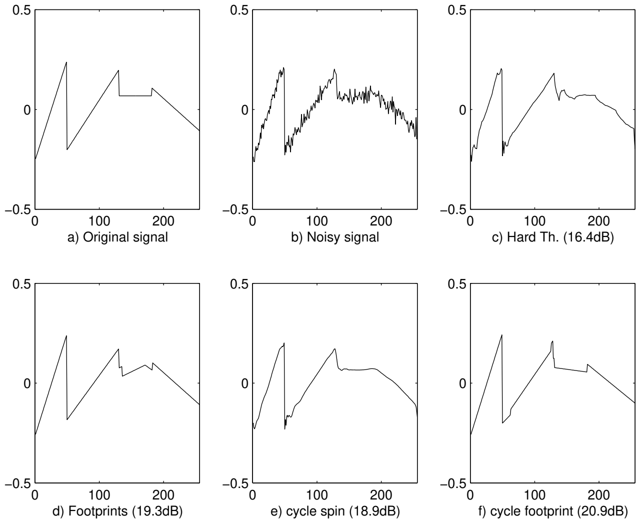

Figure 4.1: SNR results for denoising. a) Original signal. b) Noisy signal (13.8dB). c) Hard Thresholding (16.4dB). d) footprints (19.3dB). e) cycle spinning with wavelet transform (18.9dB). f) cycle spinning with footprints (20.9dB).

4.5.2 Deconvolution

In this case, we consider two different signals. One is a piecewise linear signal, the other one is a line of the image ‘Cameraman’, which represents a possible example of piecewise smooth signals. We first consider the case of a piecewise linear signal and compare the performance of our system with WaRD [82]. In this simulation, the original signal is first convolved with a box filter and then white noise is added. The noise variance is set to $\sigma^2 = 0.02$. Figure 4.3 shows that our system outperforms WaRD in both visual quality and SNR. It is of interest to note that the signal reconstructed with footprints does not present artifacts around discontinuities and that it manages to efficiently sharpen the discontinuities. Of course, one of the reasons why footprints perform so well is because the considered signal perfectly fits the model, since it is piecewise polynomial.

In Figure 4.4, we consider the case where the signal is piecewise smooth.

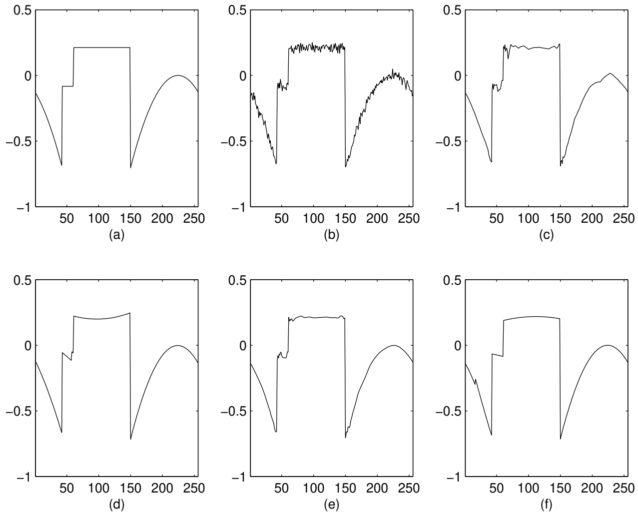

Figure 4.2: SNR results for denoising. a) Original signal. b) Noisy signal (22.5dB). c) Hard thresholding (25.3dB). d) Hard thresholding with footprints (28.5dB). e) Cycle spinning with wavelet transform (29.8dB). f) Cycle spinning with footprints (30.8dB).

Again, the original signal is convolved with a box filter and then white noise is added. In this case, we use the *two step deconvolution* algorithm. The estimated piecewise polynomial behaviour $\hat{p}[n]$ underlying the signal is shown in Figure 4.4(c). The estimated residual $\tilde{r}[n]$ and the deconvolved residual $\hat{r}[n]$ are shown in Figures 4.4(d) and 4.4(e) respectively. Finally, the reconstructed signal is shown in Figure 4.4(f).

4.5.3 Compression

In Theorem 4.1, we have shown that in case of piecewise polynomial signals, a footprint based coder can achieve the ideal rate-distortion performance. That is, it has the correct rate of decay of the R-D function. Now, we are interested in a numerical confirmation of this theorem. We consider piecewise constant signals with no more than five discontinuities. The signal has size $N = 2^{10}$ and the discontinuity locations are uniformly distributed over the interval $[0, N-1]$. The footprint coder operates as in Theorem 4.1, that is, it scalar quantizes the footprint coefficients and the discontinuity locations. Bits are allocated with a reverse waterfilling strategy as explained in Theorem 4.1. In Figure 4.5, we compare the rate-distortion performance of this footprint coder against the ideal bound and the ideal performance of a wavelet based coder. We can see that the behaviour of the footprint coder is consistent with the theory, since it has the same rate of decay as the ideal distortion function.

Finally, we consider a piecewise smooth signal. The compression operates in the following way. With a denoising-like algorithm, we estimate the piecewise

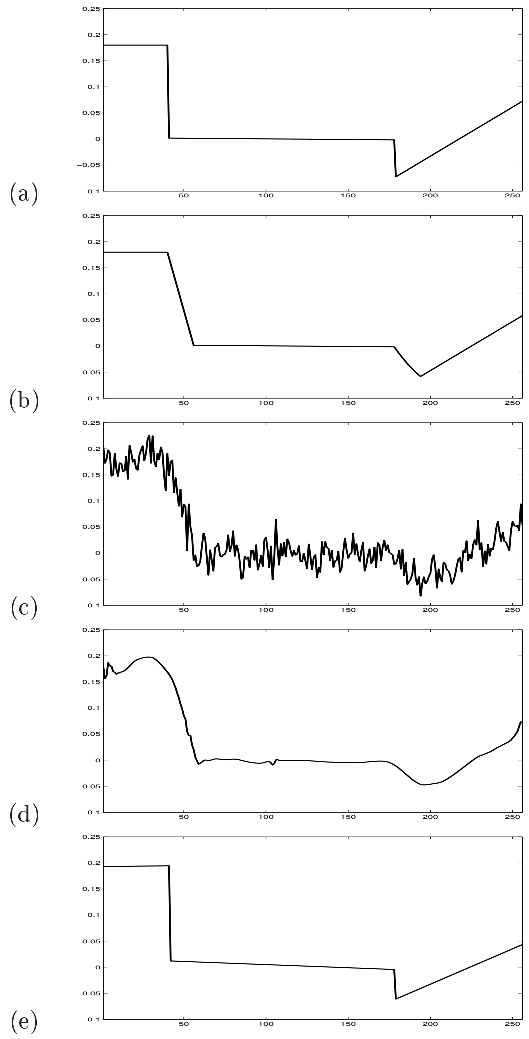

Figure 4.3: Deconvolution of a piecewise linear signal. (a) Test signal (N=256). (b) Signal convolved with a box filter. (c) Observed signal.(SNR=6.5dB) (d) Deconvolution with WaRD (SNR=8.8dB). (e) Deconvolution with Footprints (SNR=13.4dB).

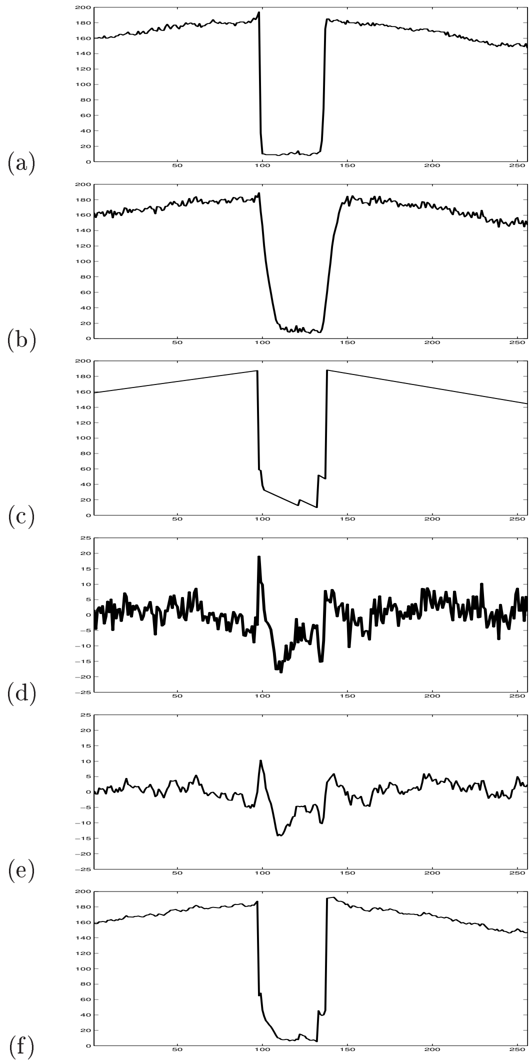

Figure 4.4: Deconvolution of a piecewise smooth signal. (a) Test signal (N=256). (b) Observed signal (SNR=16.7dB). (c) Piecewise polynomial estimation (SNR=21.1dB). (d) Residual: $\tilde{r} = y - h * \hat{p}$. (e) Deconvolution of the residual with a Wiener filter. (f) Complete deconvolved signal (SNR=21.8dB).

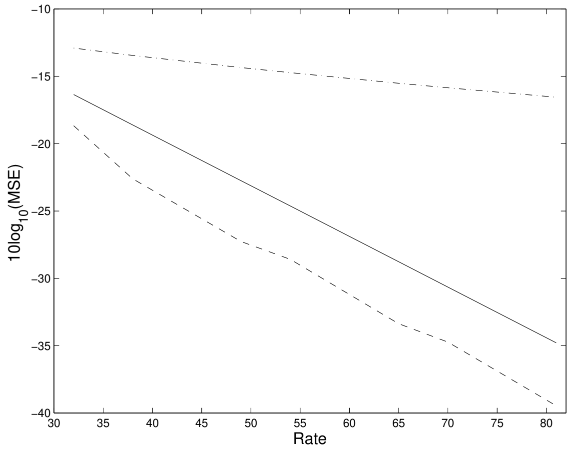

Figure 4.5: Theoretical and experimental D/R curves. Dashed-dotted: theoretical wavelet performance, dashed: empirical footprint performance, line: ideal performance.

polynomial behaviour underlying the signal and compress it with footprints. The residual is assumed regular and it is compressed in a wavelet basis. That is, the first k coefficients of the wavelet decomposition are quantized, while the other are set to zero (linear approximation). The allocation of the bits between the piecewise polynomial signal and the residual and the number k of wavelet coefficients that are quantized is chosen off-line, using some a-priori knowledge of the signal.

In Figure 4.6, we show an example of the performance of the proposed compression scheme and compare it with a 1-D version of SPIHT [90]. The signal to compress is given by the union of smooth pieces. In this example, our system outperforms SPIHT by about $3dB$. Since SPIHT is more suited to compress 2-D signals this comparison is only indicative. However, it shows that a compression system based on footprints can outperform traditional wavelet methods also in the case of piecewise smooth signals.

4.6 Conclusions

In this chapter, we have presented practical algorithms for denoising, deconvolution and compression. Numerical simulations have shown that footprints outperform wavelet methods. In short, wavelets have been very successful on signals with discontinuities, be it for denoising, deconvolution or compression. Wavelet footprints pursue this program further, by explicitly using the structure of discontinuities across scales. The results, both theoretical and experimental, confirm the usefulness of this approach. Together with the simplicity of the algorithms involved, this indicates the usefulness of this new data structure.

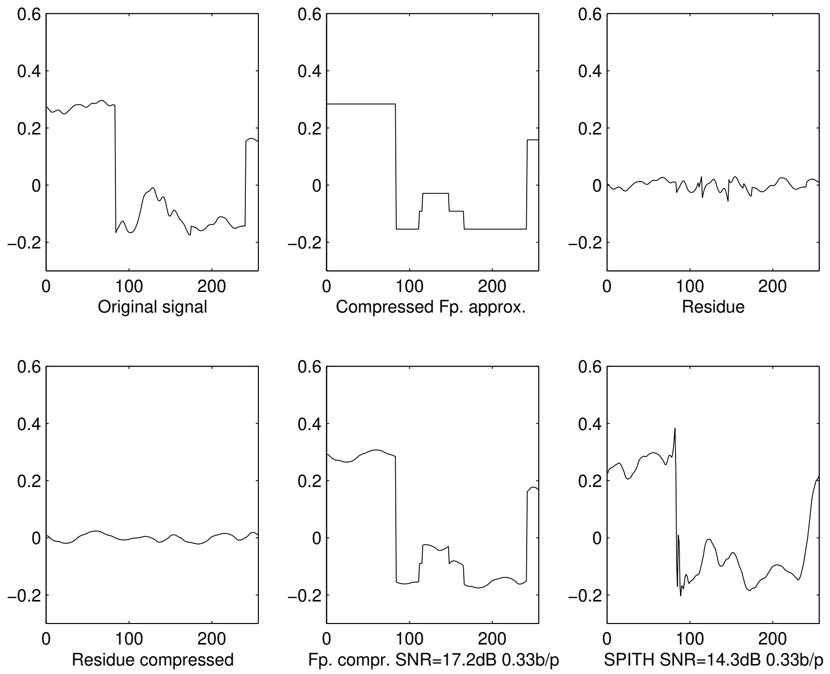

Figure 4.6: Compression of a piecewise smooth signal.

Appendix 4.A Proof of Theorem 4.1

Consider a piecewise polynomial signal $x[n] \in [0, N-1]$ of maximum degree D and with no more than K discontinuities. Assume that the signal is bounded in magnitude between $[-A, A]$. We want to prove that a compression scheme based on footprints can achieve:

$$D(R) \leq c_6 2^{-\frac{2R}{(D+3)K+(D+1)}} \sim c_6 2^{-c_7 R}.$$

Consider the representation of x in terms of footprints for the case $J = \log_2 N$:

$$x[n] = c_0 \phi_{J0}^{per}[n] + \sum_{i=0}^{K} \sum_{d=0}^{D} \alpha_i^{(d)} f_{k_i}^{(d)}[n], \tag{4.9}$$

The compression algorithm consists in uniform scalar quantizing the discontinuity locations k_i and the footprints coefficients $\alpha_i^{(d)}$. Since x is bounded, the square error relative to the quantization of a single discontinuity location can be upper bounded by

$$\|x - \hat{x}\|^2 \leq 4A^2 |k_i - \hat{k}_i|,$$

where $\hat{x}$ is the approximated signal. If R_{k_i} bits are used to quantize each discontinuity then $|k_i - \hat{k}_i| \leq (N/2)2^{-R_{k_i}}$ and the distortion related to a single discontinuity is:

$$D_l(R_{k_i}) \leq 2A^2 N 2^{-R_{k_i}}.$$

Consider now the quantization of the coefficients of the footprints expansion.[1] Since $\|f_k^{(d)}\| = 1$, the square error due to the quantization of a single coefficients $\alpha_i^{(d)}$ is

$$\|x - \hat{x}\|^2 = (\alpha_i^{(d)} - \hat{\alpha}_i^{(d)})^2.$$

Now, x is bounded and so each coefficient $\alpha_i^{(d)}$ is bounded too: $\alpha_i^{(d)} \in [-B_i^{(d)}, B_i^{(d)}]$. Thus, if $R_i^{(d)}$ bits are allocated to quantize $\alpha_i^{(d)}$, then the distortion due to this quantization can be upper-bounded by

$$D_\alpha(R_i^{(d)}) \leq B^2 2^{-2R_i^{(d)}},$$

where $B = \max_{i,d} B_i^{(d)}$. The global distortion bound is obtained by adding all the distortion contributions:

$$D(R) \leq \sum_{i=1}^{K} D_l(R_{k_i}) + \sum_{i=0}^{K} \sum_{d=0}^{D} D_\alpha(R_i^{(d)}),$$

where $R = \sum_{i=1}^{K} R_{k_i} + \sum_{i=0}^{K} \sum_{d=0}^{D} R_i^{(d)}$. Finally, by allocating bits over the different distortions with a reverse waterfilling scheme [19], the global distortion becomes:

$$D(R) \leq c_6 2^{-\frac{2R}{(D+3)K+(D+1)}} \sim c_6 2^{-c_7 R}.$$

[1] The scaling coefficient c_0 is included in this formulation.

Chapter 5

Multiple Description Coding: Theory

5.1 Introduction

Standard source coding techniques presupposes that the representation produced by the encoder will be available without errors to the decoder. This is in accordance with the famous separation principle due to Shannon [92]. However, there are errors in real communication systems. Consider, for instance, communications over heterogeneous packet switched networks. In this setting, packet losses can be due to transmission errors or congestion. If the network is able to provide preferential treatment to some packets, then the use of multiresolution or layered source coding is the obvious solution. But if the network cannot differentiate among packets, and if retransmissions are not allowed (e.g., due to real-time delay constraints or in multicast scenarios), then the source coding strategy should be different. Indeed, it would be attractive to be able to estimate the message despite packet losses and obtain a reproduction quality proportionate to the number of received packets. Multiple Description (MD) coding is a source coding method which offers a potentially attractive framework in which to develop coding algorithms for such scenarios. A MD coder represents an information source using multiple bit streams (descriptions). Each individual description provides an approximation to the original message, and multiple descriptions can refine each other to produce a better approximation than that attainable by any single one alone. In this chapter, after a comprehensive introduction in which the main information theoretic results are reviewed, we present a new MD rate-distortion region for stationary Gaussian sources which is asymptotically tight at high rates (Theorem 5.2 of Section 5.6). The review on MD coding is mostly based on [58, 59], while the new material comes from [37, 38].

[0]This chapter includes research conducted jointly with Sergio Servetto and Martin Vetterli [37, 38]

5.2 Problem Statement

In its simplest formulation, MD coding refers to the scenario illustrated in Figure 5.1 and involves only two descriptions. This is the so called case of two channels and three receivers. An encoder is given a sequence of source symbols $\{X_k\}$ to communicate to three receivers over two noiseless channels. One decoder receives information sent over both channels and can reconstruct the source at some small distortion value D_0 (the *central* distortion). The other two decoders receive information only over their respective channels and reconstruct the source at some higher distortion D_1 or D_2 (the *side* distortions). The central

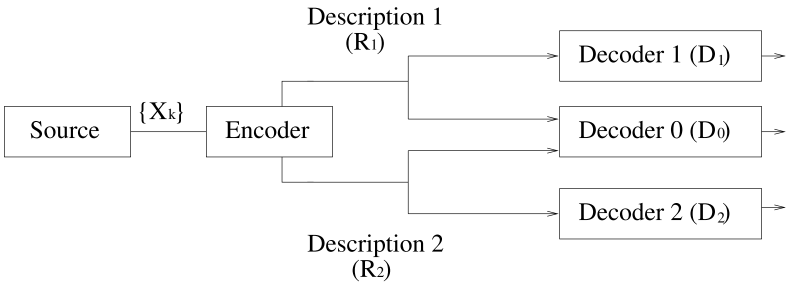

Figure 5.1: The multiple description problem.

theoretical problem is to determine the set of quintuples $(R_1, R_2, D_0, D_1, D_2)$ for which there exist codes of rates R_1 and R_2 achieving average distortions D_0, D_1 and D_2. Clearly, we can state that the rate R_1 necessary to achieve the distortion D_1 cannot be smaller than $R(D_1)$ ($R(\cdot)$ is the rate-distortion function for the source), similar arguments apply for the other two cases. So we can state that a first bound for the MD rate region is:

$$R_1 \geq R(D_1), \tag{5.1}$$

$$R_2 \geq R(D_2), \tag{5.2}$$

$$R_1 + R_2 \geq R(D_0). \tag{5.3}$$

In general, it is not possible to achieve equality simultaneously in the three equations. On the one hand, two individually good descriptions tend to be similar to each other. Thus, the second description will contribute very little to improve the quality of the first one. On the other hand, two descriptions which are complementary cannot be both individually good. Since the bounds (5.1)-(5.3) are usually loose, the gaps from equality in (5.1)-(5.3) are usually called *excess rate*. In particular, $R_i - R(D_i)$, $i = 1, 2$. are called *excess marginal rates* and $R_1 + R_2 - R(D_0)$ is called the *excess rate sum*[1].

The MD problem can be generalized to more than two channels and more than three receivers. The natural extension is to M channels and $2^M - 1$ receivers. The situation of three channel and seven receivers was studied by Zhang

[1]Sometimes the difference $R_1 + R_2 - R(D_0)$ is called redundancy.

and Berger [115]. while an achievable region for the M-channel case has been found recently [106]. Except for these works, there is no literature on all other cases.

5.3 Related problems

The successive refinement (SR) problem shown in Figure 5.2 is a special case of the MD problem. Unlike in MD coding, the channels in SR do not play symmetric roles. One channel (Channel 1) is received by both decoders, while the other channel (Channel 2) is received by one only decoder. For this reason, the information sent on Channel 2 need not be useful without Channel 1, that is, it only *refines* the information on Channel 1.

SR coding can be used for progressive transmission. In this case decoders in Figure 5.2 represent two different states of the same user. The information of Channel 1 is sent first and the receiver uses Decoder 1. If communication is not terminated, the information on Channel 2 is then sent and the receiver uses Decoder 0. SR coding also applies to layered broadcasting. In this case there are two classes of users. Both receive Channel 1 but only one class receive Channel 2. This situation may occur in wireless communication with multiresolution constellation [87] or when multicasting to groups with different available bandwidths in a packet network [81]. Since the SR problem can be obtained as

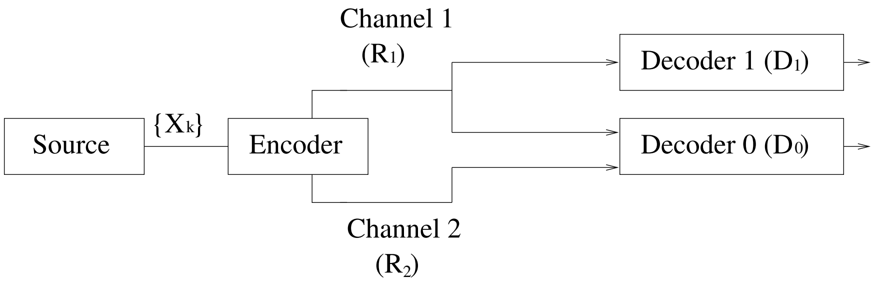

Figure 5.2: The successive refinement problem. It is obtained from MD problem when one of the side decoder is removed.

a reduction of the MD problem, bounds for MD coding apply to SR coding. In particular, since there is no constraint on D_2, the most favorable case is when equations (5.1) and (5.3) hold with equality. A source for which (5.1) and (5.3) hold with equality is called *successively refinable*.

The first results on successive refinability were published by Koshelev [71, 72, 73]. He found sufficient conditions for a source to be successively refinable. Unaware of the work of Koshelev, Equitz and Cover found necessary and sufficient conditions for a source to be successively refinable [47]. Examples of refinable sources include discrete sources with Hamming distortion and Gaussian sources with square error distortion. For related results on achievable bounds and on similar network problems, refer to [62, 89].

5.4 Domains of Application

The applicability of MD techniques depend on the type of data to be communicated and on the communication medium.

Clearly, MD coding can be used when lossless communication is not required. That is, when users are satisfied with a degraded form of the information. Examples of sources which allows lossy compression includes audio, speech, images and video. It is worth noticing that there might be exceptions. For instance, images in medicine and publishing should be not compressed lossily.

Moreover, MD coding applies to several communication media. As mentioned in the introduction, MD coding is appropriate for communication over packet networks. In fact, this possibility is mentioned in almost all works presenting MD techniques. In this setting, it is natural to relate packets in a communication network to descriptions in MD. Indeed, only certain types of communication scenarios benefit from MD coding. MD coding is appropriate for data transmission over packet networks when retransmission is not possible. Retransmission can be precluded for several reasons. The lack of a feedback channel is an example. In broadcast communications, for instance, acknowledgment messages create to much traffic and should be avoided. Moreover, broadcaster cannot afford independent retransmission to each receiver. In addition, retransmission implies delays and, in particularly congested situations, each repeated transmission might be lost as well, so delays might become unbounded. Long delays should be avoided in interactive communication or real time applications (i.e. streaming of audio or video). Thus, MD techniques are appropriate when retransmission is not possible and long delays are not acceptable.

Distributed storage matches the MD framework as well. Consider the basic interaction in multimedia browsing, where a client requests for some information (image, audio, video sequences) that is stored at a single server host. If the route leading to this server is congested, the user will experience long delays. In the Multiple Description scenario, different coarse representations of the source are stored at different databases (on different servers), so even if some of these servers are congested the user can satisfy himself with a lower resolution version of the signal using the description coming from one non-congested server.

Finally, there are situations in which a wireless channel is decomposed into more than one virtual channel. MD coding may be appropriate on these virtual channels. See [111, 58] for related results.

5.5 Fundamental results for memoryless sources

We now review a few fundamental results in MD coding which are also important to prove Theorem 5.2 in the next Section.

The main result in MD coding is the achievable rate region determined by El Gamal and Cover [45].

Theorem 5.1 (Achievable rates for multiple description coding [45])

Let $\{X_k\}$ be a sequence of i.i.d. random variables drawn according to a probability mass function $p(x)$ and let the distortion measures $d_i(\cdot,\cdot)$, $i = 0, 1, 2$, be bounded. Then an achievable rate region for MD coding with distortions (D_0, D_1, D_2) is given by the convex hull of all rates (R_1, R_2) such that

$$\begin{aligned} R_1 &\geq I(X; \hat{X}_1), & (5.4)\\ R_2 &\geq I(X; \hat{X}_2), & (5.5)\\ R_1 + R_2 &\geq I(X; \hat{X}_0, \hat{X}_1, \hat{X}_2) + I(\hat{X}_1; \hat{X}_2), & (5.6) \end{aligned}$$

for some probability mass function $p(x, \hat{x}_0, \hat{x}_1, \hat{x}_2)$ such that

$$E[d(X, \hat{X}_i) \leq D_i, \quad i = 0, 1, 2.$$

Note that with some simple manipulations Eq. (5.6) can be written as:

$$R_1 + R_2 \geq I(X; \hat{X}_1) + I(X; \hat{X}_2) + I(X; \hat{X}_0 | \hat{X}_1, \hat{X}_2).$$

This expression better highlights the fundamental trade-off in the MD problem. If the rates R_1, R_2 are only allocated to the side decoders, that is, $R_i = I(X; \hat{X}_i)$, $i = 1, 2$, then $I(X; \hat{X}_0 | \hat{X}_1, \hat{X}_2) = 0$. This means that the joint decoding of the two channels does not add any information about the original source or as we claimed in the beginning, the second description does not contribute to improve the quality of the first one. If we want to allocate part of the rate to the central decoder such that $I(X; \hat{X}_0 | \hat{X}_1, \hat{X}_2) > 0$, then we cannot have equality in (5.4),(5.5).

Except for the memoryless Gaussian case [84] that will be discussed later, there are no other cases where the converse to Theorem 5.1 has been proved. However, Ahlswede [1] studied the case of *no excess rate sum* and showed that, in this case, the El Gamal and Cover bounds are tight. Zhang and Berger [116] considered the *no excess marginal rate* case and they showed by counterexample that in the excess rate case the achievable region of El Gamal and Cover is not tight [115]. More recently, Linder et al. [76] found a rate region for memoryless sources and locally quadratic distortion measure which is tight in the limit of small distortions (high bit rate). Finally Zamir [112, 113] extended the Shannon bounds [7] to the MD case and showed that for a Gaussian source the outer bounds are asymptotically tight.

Thus, the case of memoryless Gaussian sources and square error distortion studied by Ozarow [84] remains the only case where the complete MD rate-region is known exactly. The explicit characterization of the set of achievable distortions (D_0, D_1, D_2) for a given pair of rates R_1, R_2 due to Ozarow is the following:

$$\begin{aligned} D_1 &\geq \sigma^2 \cdot 2^{-2R_1}, & (5.7)\\ D_2 &\geq \sigma^2 \cdot 2^{-2R_2}, & (5.8)\\ D_0 &\geq \frac{\sigma^2 \cdot 2^{-2(R_1+R_2)}}{1 - (\sqrt{\Pi} - \sqrt{\Delta})^2}, & (5.9) \end{aligned}$$

where σ^2 is the variance of the memoryless Gaussian source, $\Pi = (1-D_1/\sigma^2)(1-D_2/\sigma^2)$ and $\Delta = (D_1 D_2/\sigma^4) - 2^{-2(R_1+R_2)}$.

The inverse of these functions are the following [76, 84]

$$
\begin{aligned}
R_1 &\geq \frac{1}{2}\log\left(\frac{\sigma^2}{D_1}\right), & (5.10)\\
R_2 &\geq \frac{1}{2}\log\left(\frac{\sigma^2}{D_2}\right), & (5.11)\\
R_1 + R_2 &\geq \frac{1}{2}\log\left(\frac{\sigma^2}{D_1}\right) + \frac{1}{2}\log\left(\frac{\sigma^2}{D_2}\right) + \delta, & (5.12)
\end{aligned}
$$

δ is defined by:

$$
\delta = \begin{cases} \frac{1}{2}\log\left(\frac{1}{1-\rho^2}\right), & D_0 \leq D_0^{\max} \\ 0, & D_0 > D_0^{\max} \end{cases} \tag{5.13}
$$

where:

$$
D_0^{\max} = \frac{D_1 D_2}{D_1 + D_2 - (D_1 D_2/\sigma^2)} \tag{5.14}
$$

and

$$
\begin{aligned}
&\rho = -\frac{\sqrt{\Lambda\epsilon_0^2+\gamma}-\sqrt{\Lambda\epsilon_0^2}}{(1-\epsilon_0)\sqrt{\epsilon_1\epsilon_2}} && \gamma = (1-\epsilon_0)[(\epsilon_1-\epsilon_0)(\epsilon_2-\epsilon_0)+\epsilon_0\epsilon_1\epsilon_2-\epsilon_0^2]\\
&\Lambda = (1-\epsilon_1)(1-\epsilon_2) && \epsilon_i = D_i/\sigma^2 \quad (i = 0, 1, 2).
\end{aligned}
$$

The MD rate-region for memoryless Gaussian sources is sketched in Figure 5.3. Notice that δ depends on the three distortion (D_0, D_1, D_2) and on the variance σ^2 (Eq. (5.13)). However, by rearranging equation (5.12), one can see the relationship between δ and the rates R_1 and R_2 and interpret δ as the excess rate that is used to reduce the central distortion given the two side distortions or:

$$
\delta = R_1 - \frac{1}{2}\log\left(\frac{\sigma^2}{D_1}\right) + R_2 - \frac{1}{2}\log\left(\frac{\sigma^2}{D_2}\right).
$$

Now, if $\delta = 0$ then $R_1 = 1/2\log(\sigma^2/D_1)$ and $R_2 = 1/2\log(\sigma^2/D_2)$. This means that all the rate is used to minimize the side distortions and in this case D_0 equals its maximum value (D_0^{max}). If $\delta > 0$, it means that part of the rate is used to reduce the central distortion which becomes smaller than D_0^{max}. In particular, D_0 decreases from D_0^{max} to zero as δ increases from zero to infinity. This is why δ is also called the excess marginal rate [112, 116]. Finally, consider the high rate situation, namely the case where the three distortions (D_0, D_1, D_2) are very small compared to the variance σ^2 or, in other words, the case where the three ratios $D_0/\sigma^2, D_1/\sigma^2, D_2/\sigma^2$ go to zero. In this situation, the excess marginal rate δ (Eq. (5.13)) and the maximum central distortion $D_0^{\max}$ (Eq. (5.14)) do not depend on the variance of the source anymore. In particular, we have [76]:

$$
\begin{aligned}
\delta_{HR}(D_0, D_1, D_2) &\triangleq \lim_{\sigma\to\infty}\delta(\sigma^2, D_0, D_1, D_2)\\
&\triangleq \lim_{\lambda\to 0}\delta(\sigma^2, \lambda D_0, \lambda D_1, \lambda D_2)\\
&= \tfrac{1}{2}\log\left(\frac{1}{1-\rho_{HR}^2}\right),
\end{aligned} \tag{5.15}
$$

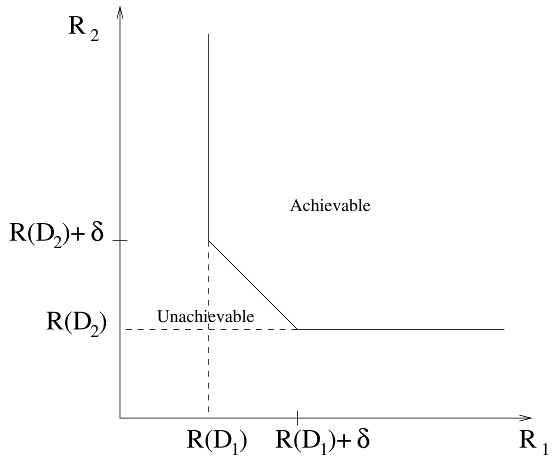

Figure 5.3: MD rate region of memoryless Gaussian sources.

where:

$$\rho_{HR} = -\frac{(\sqrt{D_1/D_0 - 1})(\sqrt{D_2/D_0 - 1}) - 1}{\sqrt{D_1 D_2/D_0^2}}$$

and

$$\lim_{\sigma\to\infty} D_0^{\max} = \frac{D_1 D_2}{D_1 + D_2}.$$

5.6 Multiple Description Rate Region for Stationary Gaussian Sources

As shown in the previous section, much of the multiple description literature is focused on memoryless sources. In this section, we present a new MD rate region for stationary Gaussian sources with memory.

Theorem 5.2 *In the limit of small distortions (i.e., $D_0, D_1, D_2 \to 0$), the asymptotic multiple description rate region for a stationary Gaussian source and MSE distortions is given by the following equations:*

$$\begin{aligned}
R_1 &\geq \tfrac{1}{4\pi}\textstyle\int_{-\pi}^{\pi} \log\left(\frac{S(\omega)}{D_1}\right) d\omega, \\
R_2 &\geq \tfrac{1}{4\pi}\textstyle\int_{-\pi}^{\pi} \log\left(\frac{S(\omega)}{D_2}\right) d\omega, \\
R_1 + R_2 &\geq \tfrac{1}{4\pi}\left(\textstyle\int_{-\pi}^{\pi} \log\left(\frac{S(\omega)}{D_1}\right) d\omega + \int_{-\pi}^{\pi} \log\left(\frac{S(\omega)}{D_2}\right) d\omega + 2\delta_{HR}\right),
\end{aligned}$$

where $S(\omega)$ is the power spectral density of the Gaussian source.

Proof: Let $\{X_t,\ t = 0, \pm 1, ..\}$ be a discrete-time stationary Gaussian source. We begin by considering N successive elements of this source and by calculating the asymptotic MD rate region of this N-sequence. Call Φ_N the $N \times N$ correlation matrix related to any N successive components of $\{X_t\}$. Since the source is stationary, Φ_N is a symmetric Toeplitz matrix. Apply a Karhunen-Loève Transform (KLT) to the N-sequence to get uncorrelated (and so independent) components. Because the KLT is unitary and invertible and we are considering MSE as our distortion measure, the problem of finding the MD rate region in the new coordinates is identical to that in the original ones, except that the new components are statistically independent. Call $Y = (y_1, y_2, .., y_N)$ the N-dimensional vector with independent components obtained after applying the KLT to the original N-sequence and call $\hat{Y_0}, \hat{Y_1}, \hat{Y_2}$ the appropriate reproducing vectors at the three receivers. Moreover, $\hat{y}_{ki}$ will represent the i-th component of the reproducing vector $\hat{Y_k}$, $k = 0, 1, 2$ and Y^i is a vector with the first i elements of Y. Extending the El-Gamal-Cover results [45] to the vector case, we have that the MD rate region is given by:

$$
\begin{aligned}
R_1 &\geq \frac{1}{N} \min I(Y; \hat{Y_1}), && (5.16)\\
R_2 &\geq \frac{1}{N} \min I(Y; \hat{Y_2}), && (5.17)\\
R_1 + R_2 &\geq \frac{1}{N} \min(I(Y; \hat{Y_0}, \hat{Y_1}, \hat{Y_2}) + I(\hat{Y_1}; \hat{Y_2})), && (5.18)
\end{aligned}
$$

where the minima are over all the probability density functions $p(Y, \hat{Y_0}, \hat{Y_1}, \hat{Y_2})$ satisfying:

$$E[\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_{ki})^2] \leq D_k, \quad k = 0, 1, 2.$$

First notice that the term $I(Y; \hat{Y_0}, \hat{Y_1}, \hat{Y_2}) + I(\hat{Y_1}; \hat{Y_2})$ in Eq. (5.18) can be equivalently expressed as:

$$
\begin{aligned}
& I(Y; \hat{Y_0}, \hat{Y_1}, \hat{Y_2}) + I(\hat{Y_1}; \hat{Y_2}) && (5.19)\\
&= I(Y; \hat{Y_0}, \hat{Y_1}, \hat{Y_2}) + H(\hat{Y_1}) + H(\hat{Y_2}) - H(\hat{Y_1}, \hat{Y_2}) && (5.20)\\
&= I(Y; \hat{Y_0}, \hat{Y_1}, \hat{Y_2}) + I(Y; \hat{Y_1}) + I(Y; \hat{Y_2}) - H(\hat{Y_1}, \hat{Y_2}), && (5.21)
\end{aligned}
$$

where in the last equality we have used the fact that $\hat{Y_1}, \hat{Y_2}$ are deterministic functions of Y and thus $H(\hat{Y_k}) = I(Y; \hat{Y_k})$, $k = 1, 2$. In the rest of the proof we will use Eq. (5.21) rather than Eq. (5.19). Now, the first thing we want to show is that the MD rate region of this N-vector reduces to the sum of the MD rate region of each component of Y and that the problem of minimizing (5.16-5.18) reduces to the problem of finding the right allocation strategy of the rates R_1, R_2 to the different components. Consider first Eq. (5.16), it follows that [19]:

$$\begin{aligned}
I(Y;\hat{Y}_1) &= h(Y) - h(Y|\hat{Y}_1) && (5.22)\\
&= \sum_{i=1}^{N} h(y_i) - \sum_{i=1}^{N} h(y_i|Y^{i-1},\hat{Y}_1) && (5.23)\\
&\geq \sum_{i=1}^{N} h(y_i) - \sum_{i=1}^{N} h(y_i|\hat{y_1}_i) && (5.24)\\
&= \sum_{i=1}^{N} I(y_i;\hat{y}_i) && (5.25)\\
&\geq \sum_{i=1}^{N} \frac{1}{2}\log\left(\frac{\lambda_i^2}{D_{1i}}\right), && (5.26)
\end{aligned}$$

where (5.23) follows from the independence of the components y_i and from the chain rule for entropy. The inequality in (5.24) follows from the fact that conditioning reduces entropy and we can achieve equality by choosing $p(Y|\hat{Y}_1) = \prod_{i=1}^{N} p(y_i|\hat{y}_{1i})$. The last inequality follows from the expression of the rate distortion function of a Gaussian source and equality can be achieved by choosing each $\hat{y}_{1i} \sim \mathcal{N}(0, \lambda_i^2 - D_{1i})$, where λ_i^2 is the variance of the i-th component and $D_{1i} = E[(y_i - \hat{y}_{1i})^2]$ is the distortion related to that component. Hence, from Eqns. (5.22-5.26), we obtain that the minimization in equation (5.16) reduces to [19]:

$$R_1 \geq \frac{1}{N}\min\sum_{i=1}^{N}\frac{1}{2}\log\left(\frac{\lambda_i^2}{D_{1i}}\right), \qquad (5.27)$$

where the minimum, now, is over all the possible distortions D_{1i} such that:

$$E[\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_{1i})^2] = \frac{1}{N}\sum_{i=1}^{N} D_{1i} \;\leq\; D_1.$$

Similar arguments apply to equation (5.17) and that minimization reduces to:

$$R_2 \geq \frac{1}{N}\min\sum_{i=1}^{N}\frac{1}{2}\log\left(\frac{\lambda_i^2}{D_{2i}}\right), \qquad (5.28)$$

where the minimum is over all the distortions D_{2i} such that $\frac{1}{N}\sum_{i=1}^{N} D_{2i} \leq D_2$. Consider, now, equation (5.18) and its alternative representation in (5.21). Consider, first, the term $I(Y;\hat{Y}_0,\hat{Y}_1,\hat{Y}_2)$; following the same procedure as in (5.22-

5.26) we have:

$$\begin{aligned} I(Y;\hat{Y_0},\hat{Y_1},\hat{Y_2}) &= h(Y) - h(Y|\hat{Y_0},\hat{Y_1},\hat{Y_2}) && (5.29)\\ &= \sum_{i=1}^{N} h(y_i) - \sum_{i=1}^{N} h(y_i|Y^{i-1},\hat{Y_0},\hat{Y_1},\hat{Y_2}) && (5.30)\\ &\geq \sum_{i=1}^{N} h(y_i) - \sum_{i=1}^{N} h(y_i|\hat{y}_{0i},\hat{y}_{1i},\hat{y}_{2i}) && (5.31)\\ &= \sum_{i=1}^{N} I(y_i;\hat{y}_{0i},\hat{y}_{1i},\hat{y}_{2i}), && (5.32) \end{aligned}$$

where inequality in (5.31) follows from the fact that conditioning reduces entropy and equality is achieved if $p(Y|\hat{Y_0},\hat{Y_1},\hat{Y_2}) = \prod_{i=1}^{N} p(y_i|\hat{y}_{0i},\hat{y}_{1i},\hat{y}_{2i})$. For the second term of (5.21), we obtain:

$$\begin{aligned} & I(Y;\hat{Y_1}) + I(Y;\hat{Y_2}) - H(\hat{Y_1},\hat{Y_2}) && (5.33)\\ \geq\ & \sum_{i=1}^{N} I(y_i;\hat{y}_{1i}) + \sum_{i=1}^{N} I(y_i;\hat{y}_{2i}) - H(\hat{Y_1},\hat{Y_2}) && (5.34)\\ \geq\ & \sum_{i=1}^{N} I(y_i;\hat{y}_{1i}) + \sum_{i=1}^{N} I(y_i;\hat{y}_{2i}) - \sum_{i=1}^{N} H(\hat{y}_{1i},\hat{y}_{2i}), && (5.35) \end{aligned}$$

where inequality in (5.34) follows from (5.22-5.25) and we can achieve equality by choosing $p(Y|\hat{Y_k}) = \prod_{i=1}^{N} p(y_i|\hat{y}_{ki})$ for $k = 1,2$. The last inequality follows from the fact that $H(\hat{Y_1},\hat{Y_2}) \leq \sum_{i=1}^{N} H(\hat{y}_{1i},\hat{y}_{2i})$ and equality is achieved if $p(\hat{Y_1},\hat{Y_2}) = \prod_{i=1}^{N} p(\hat{y}_{1i},\hat{y}_{2i})$. Thus, combining the results from (5.29-5.32) and (5.33-5.35), we have:

$$\begin{aligned} & I(Y;\hat{Y_0},\hat{Y_1},\hat{Y_2}) + I(Y;\hat{Y_1}) + I(Y;\hat{Y_2}) - H(\hat{Y_1},\hat{Y_2}) && (5.36)\\ \geq\ & \sum_{i=1}^{N} \left(I(y_i;\hat{y}_{0i},\hat{y}_{1i},\hat{y}_{2i}) + I(y_i;\hat{y}_{1i}) + I(y_i;\hat{y}_{2i}) - H(\hat{y}_{1i},\hat{y}_{2i})\right) && (5.37)\\ \geq\ & \sum_{i=1}^{N} \frac{1}{2}\log\left(\frac{\lambda_i^2}{D_{1i}}\right) + \frac{1}{2}\log\left(\frac{\lambda_i^2}{D_{2i}}\right) + \delta_i, && (5.38) \end{aligned}$$

where the last inequality comes from the Ozarow equations and equality can be achieved by a correct choice of each triple $(\hat{y_{0i}},\hat{y_{1i}},\hat{y_{2i}})$. This choice depends on the three distortions D_{0i}, D_{1i}, D_{2i} and for an explicit characterization refer to [84].

Eqns. (5.36-5.38) shows that minimization in (5.18) reduces to:

$$R_1 + R_2 \geq \min\left(\frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{1i}}\right) + \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{2i}}\right) + \frac{1}{N}\delta_i\right), \quad (5.39)$$

where the minimum, now, is over all the distortions D_{0i}, D_{1i}, D_{2i} such that $\frac{1}{N}\sum_{i=1}^{N} D_{ki} \leq D_k$, $k = 0,1,2$. Now, combining (5.27), (5.28) and (5.39), we can see that the MD rate region of Y reduces, indeed, to the sum of the MD rate region of each component and that the original minimization problem reduces to:

$$\begin{aligned}
R_1 &\geq \min_{\frac{1}{N}\sum_i D_{1i}=D_1} \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{1i}}\right), && (5.40)\\
R_2 &\geq \min_{\frac{1}{N}\sum_i D_{2i}=D_2} \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{2i}}\right), && (5.41)\\
R_1+R_2 &\geq \min_{\substack{\frac{1}{N}\sum_i D_{0i}=D_0\\ \frac{1}{N}\sum_i D_{1i}=D_1\, \frac{1}{N}\sum_i D_{2i}=D_2}} \left(\frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{1i}}\right)\right.\\
&\qquad \left. +\frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{2i}}\right)+\frac{1}{N}\delta_i\right). && (5.42)
\end{aligned}$$

Thus, the problem now is to understand how each component should contribute to the total distortion to minimize the quantities in (5.40-5.42) or, stated in a different way, the problem is to understand how the rates R_1, R_2 should be allocated to the various components to minimize (5.40-5.42). Using Lagrange multipliers we can construct the following three functionals:

$$\begin{aligned}
J_1 &= \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{1i}}\right)+\nu_1\sum_i D_{1i}, && (5.43)\\
J_2 &= \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{2i}}\right)+\nu_2\sum_i D_{2i}, && (5.44)\\
J_3 &= \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{1i}}\right)+\frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_{2i}}\right)\\
&\quad +\frac{1}{N}\delta_i+\nu_0\sum_i D_{0i}+\nu_1\sum_i D_{1i}+\nu_2\sum_i D_{2i}. && (5.45)
\end{aligned}$$

The problem of minimizing the first two functionals is equivalent to the problem of finding the optimal allocation strategy for the single description case. Differentiating with respect to D_{1i} and D_{2i} and setting equal to zero, we have:

$$\begin{aligned}
\frac{\partial J_1}{\partial D_{1i}} &= -\frac{1}{2N}\frac{1}{D_{1i}}+\nu_1=0, && (5.46)\\
\frac{\partial J_2}{\partial D_{2i}} &= -\frac{1}{2N}\frac{1}{D_{2i}}+\nu_2=0, && (5.47)
\end{aligned}$$

and

$$\begin{aligned}
D_{1i} &= c_1 = D_1, && (5.48)\\
D_{2i} &= c_2 = D_2, && (5.49)
\end{aligned}$$

where c_1 and c_2 are constants. Hence, the optimum allocation of the rates to the various components results in equal distortion for each component [7, 19]. This is due to the fact that the slopes of the curves (5.43)-(5.44) are independent of the variances. This argument is not valid for the third functional (5.45) since the slope of δ depends on the variance. However, in the limit of small distortions δ becomes independent of the variance $\delta \to \delta_{HR}$ (see Eq. (5.15)), and the minimization strategy for the third functional becomes the same as for the first two functionals (i.e., $D_{1i} = D_1, D_{2i} = D_2, D_{0i} = D_0 \quad i = 1, 2, .., N.$). Then the MD rate-region becomes:

$$
\begin{aligned}
R_1 &\geq \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_1}\right), \\
R_2 &\geq \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_2}\right), \\
R_1 + R_2 &\geq \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_1}\right) + \frac{1}{2N}\sum_{i=1}^{N}\log\left(\frac{\lambda_i^2}{D_2}\right) + \delta_{HR}.
\end{aligned}
$$

Notice that, since the Ozarow's MD rate region is achievable and tight [84], then also our MD rate region is (asymptotically) achievable and tight. Indeed, we have seen that the MD rate-distortion functions (5.16-5.18) of the vector Y are lower bounded by the sum of the MD rate-distortion functions of each component y_i and this lower bound is achieved by coding each component independently. Now, since the direct and converse part of the Ozarow theorem apply to each component, the minima in Eqns.(5.40-5.42), not only represent an achievable region, but they also represent a tight region.

Now, using the result of the Toeplitz Distribution Theorem [7, 63] (see Appendix 5.A), we can go to the limit of infinite N and find the MD rate region of the complete source $\{X_t\}$:

$$
\begin{aligned}
R_1 &\geq \frac{1}{4\pi}\int_{-\pi}^{\pi}\log\left(\frac{S(\omega)}{D_1}\right)d\omega, \\
R_2 &\geq \frac{1}{4\pi}\int_{-\pi}^{\pi}\log\left(\frac{S(\omega)}{D_2}\right)d\omega, \\
R_1 + R_2 &\geq \frac{1}{4\pi}\left(\int_{-\pi}^{\pi}\log\left(\frac{S(\omega)}{D_1}\right)d\omega + \int_{-\pi}^{\pi}\log\left(\frac{S(\omega)}{D_2}\right)d\omega + 2\delta_{hr}\right).
\end{aligned}
$$

□

A similar result in terms of the entropy rate power of the Gaussian source can be found in [112, 113]. In these papers Zamir extended the Shannon bounds [7] to the MD case and then showed that the outer bound is asymptotically tight. His results are valid both for sources with or without memory.

Theorem 5.2 shows that at high rates the single description allocation strategy is also optimal in the MD case. That is because the slopes of the three functionals (5.43-5.45) are independent of the source. At low rates, this last assumption is not valid. The functional (5.45) has a slope dependent on the input source variance and in general, it is not minimized with a single description allocation strategy. So we can state the following corollary:

Corollary 5.1 *Under a high rate assumption, and for stationary Gaussian sources, the single description rate allocation strategy is also optimal in the Multiple Description case. At low rates it is, in general, sub-optimal.*

In conclusion, the main contribution of this chapter is the extension of the Ozarow results to the case of stationary Gaussian sources. In the next chapter, we will concentrate on practical implementations of MD coders and propose a new MD system. The definition of the new performance bounds given by Theorem 5.2 will allow us to analyze the performance of MD systems which operates on sequences with memory and this was not possible before.

Appendix 5.A Toeplitz Distribution Theorem

For a proof of this theorem refer to [63] (see also [61, 60]):

Theorem 5.3 *Let Φ_∞ be an infinite Toeplitz matrix with entry ϕ_k on the kth diagonal. The eigenvalues of Φ_∞ are contained in the interval $\delta \leq \lambda \leq \Delta$ where δ and Δ are the essential infimum and supremum,*[2] *respectively, of the function:*

$$\Phi(\omega) = \sum_{k=-\infty}^{\infty} \phi_k e^{-jk\omega}.$$

Moreover, if both δ and Δ are finite and $G(\lambda)$ is any continuous function of $\lambda \in [\delta, \Delta]$, then:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} G(\lambda_k^{(n)}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} G[\Phi(\omega)] d\omega,$$

where the $\lambda_k^{(n)}$ are the eigenvalues of the nth-order matrix Φ_n centered about the main diagonal of Φ_∞.

[2]The essential supremum of a function f(x) is the $\inf_E \sup_{x \notin E} f(x)$ where E ranges over all sets of Lebesgue measure zero. Likewise, the essential infimum is $\sup_E \inf_{x \notin E} f(x)$.

Chapter 6

Multiple Description Coding: Practical Techniques

6.1 Introduction

Up to now, we have studied the performance bounds of MD coders, but we have not considered the problem of how to design practical systems that can achieve that performance. One can get the impression that MD coding arose as a pure theoretical problem and found applications later. In fact, to be more precise, MD coding was born as a practical scheme to deal with errors in a communication channel. To quote [58]: *More accurately, MD coding has come full circle from explicit practical motivation to theoretical novelty and back to engineering application.*

In this chapter we focus on practical implementations. The next section will briefly survey existing methods, while in Section 6.3, we present a new MD coder. In that section we show a way to design optimal two-channel filter banks for MD coding of stationary Gaussian sources (Theorem 6.1). Finally, in Section 6.4 we assess the performance of this system. In particular, we show that in some realistic settings (i.e., low complexity and low bit rate) the proposed system outperform existing methods.

6.2 Survey of existing MD techniques

Several efforts have been made to design practical MD coding systems. These systems can be divided into two main families which follow two different philosophies. One family focuses on the problem of designing particular quantizers that can meet the MD constraint, while the other family uses ordinary quantizers and get the MD property from the choice of a particular transform. The main methods based on these two different approaches are presented in the next two

[0]This chapter includes research conducted jointly with Martin Vetterli [40], and Sergio Servetto and Martin Vetterli [37, 38]

subsections. In Section 6.2.3, we present two extensions of these techniques to the case of stationary sequences.

6.2.1 Multiple description scalar quantizer

In [102], a design procedure for the construction of fixed-rate MD scalar quantizers was presented. A multiple description scalar quantizer (MDSQ) produces a pair of indices for each source sample. Denote the encoder mapping by α_0: $\mathbb{R} \to \mathcal{I}_1 \times \mathcal{I}_2$, where symbols in $\mathcal{I}_i$ are sent over channel i, $i = 1, 2$. The central decoder receives both indices and can reconstruct the source with a mapping from $\mathcal{I}_1 \times \mathcal{I}_2$ to $\mathbb{R}$. The side decoders receive only one of the two indices and have to reconstruct the source with a different mapping from $\mathcal{I}_i$ to $\mathbb{R}$, $i = 1, 2$. Therefore, as in all multiple description systems, the design challenge is to si-

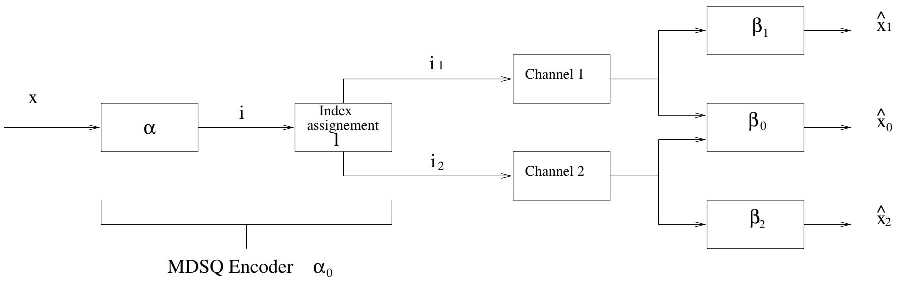

Figure 6.1: The multiple description scalar quantizer. The encoder α_0 produces two indices $i_1 \in \mathcal{I}_1, i_2 \in \mathcal{I}_2$ for each input sample. It can be seen as an ordinary encoder α and an index assignment ℓ. The decoder β_0 receives both indices, β_1 receives only i_1 and β_2 only i_2.

multaneously provide good individual descriptions and a good joint description. An MD scalar quantizer encoder can be seen as an ordinary quantizer encoder α which maps $\mathbb{R}$ into an index set $\mathcal{I}$ plus an *index assignment* $\ell : \mathcal{I} \to \mathcal{I}_1 \times \mathcal{I}_2$. In this interpretation, an MD scalar quantizer is an ordinary quantizer plus an index assignment and two extra decoder mappings for the side decoders. If we call $\beta_0, \beta_1, \beta_2$ the three different decoders, the design challenge reduces to the optimization of the five elements $\alpha, l, \beta_0, \beta_1, \beta_2$ (See Figure 6.1).

Vaishampayan [102] proposed a generalized Lloyd-Max-like algorithm to find the optimal $\alpha, \beta_0, \beta_1, \beta_2$ for a fixed ℓ. The algorithm uses Lagrange multipliers λ_1, λ_2 to create a scalar distortion criterion $D_0 + \lambda_1 D_1 + \lambda_2 D_2$. Now, just as in the iterative design of an ordinary fixed-rate quantizer, one can alternate between improving the encoder α and the decoders $\beta_0, \beta_1, \beta_2$, until a locally optimal MD scalar quantizer is obtained.[1]

The remaining problem is to optimize ℓ. Unfortunately, no efficient method for its optimization is known. Instead, Vaishampayan [102] gives several heuristic techniques that likely get close to the best possible performance. The easiest

[1]For more details on the Lloyd-Max algorithm refer to [50].

way to represent an index assignment is by writing ℓ as a two-dimensional array (See Figure 6.2). The basic idea is that this array should be populated from the main diagonal outward and that the numbering should run from upper-left corner to lower-right corner. The aim in doing this is to keep the reconstruction error at the side decoders as small as possible. MD scalar quantization is flexible in that it allows a designer to choose the relative importance of the central distortion and of each side distortion, by changing the number of filled diagonals. That is, if only the main diagonal is filled, we obtain a repetition code and we minimize the side distortions, but the central distortion is high; if the two-dimensional array is completely filled (maximum number of filled diagonals), we minimize the central distortion but we have poor side distortions. The following example will clarify the design method of ℓ and this trade-off.

Example. Consider a source uniformly distributed on $[0, 1]$. The encoder α is uniform in $[0, 1]$ and the number of cells in α depends on ℓ as it will result clear in a moment. The decoders β_i, $i = 0, 1, 2$ are optimal for a given α and ℓ. Assume that $R_1 = R_2 = 2$ (that is, $\mathcal{I}_i = \{0, 1, 2, 3\}$, $i = 1, 2$) and that the index assignment is represented by the array in Figure 6.2 (In this case only two diagonals have been filled). Note that the array contains 7 numbers, this means that the encoder α can only partition the interval $[0, 1]$ in 7 cells (see Figure 6.2). Now, assume that a realization x of the source is such that the

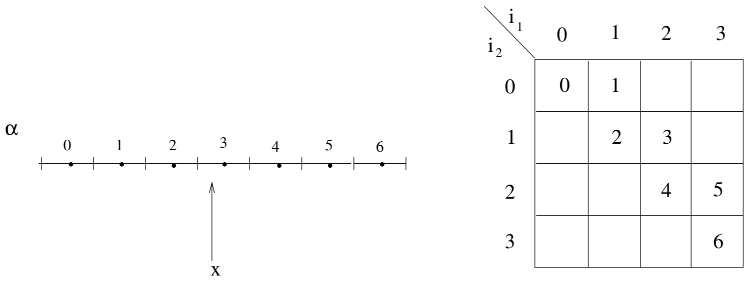

Figure 6.2: Example of a multiple description scalar quantizer. On the left, the encoder α; on the right, the index assignment array. In this case only two diagonals are filled.

encoder α produces the index 3. Then ℓ will produce the two indices $i_1 = 2$ and $i_2 = 1$ corresponding to the coordinates of 3 in the two-dimensional array. The three decoders $\beta_0, \beta_1, \beta_2$ have a copy of the same array. The central decoder gets both coordinates i_1, i_2 and can find the original index 3. Therefore the reconstructed value $\hat{x}_0$ will be the centroid of the 3^{th} cell. Decoder β_1 has only access to i_1 or, in other words, it has only access to the columns of the two-dimensional array but not to the rows. So it cannot recover the original index 3, instead it can only say that the index was either 3 or 4. Therefore, the reconstructed value $\hat{x}_1$ will be the centroid of the cells 3 and 4. Likewise, the reconstructed value at decoder β_2 will be the centroid of the cells $2, 3$. In Figure 6.3, different index assignment strategies for the case $R_1 = R_2 = 2$ are presented. The reader can easily convince himself that the upper-left strategy corresponds to a simple repetition code. That is, any input sample x produces the same indices i_1 and i_2 and the three decoders $\beta_0, \beta_1, \beta_2$ have the same performance. This is the most favorable case for the side decoders. The lower-right array represents the opposite strategy. That is, the central decoder has 16 different reconstruction levels which is the best case for a 4-bits quantizer, but in this case the side decoders show poor performance. The average central and side distortions for the four example in Figure 6.3 are shown in Table 6.1 [59].

$i_2 \backslash i_1$	0	1	2	3
0	0			
1		1		
2			2	
3				3

$i_2 \backslash i_1$	0	1	2	3
0	0	1	4	
1	2	3	5	8
2		6	7	9
3			10	11

$i_2 \backslash i_1$	0	1	2	3
0	0	2	3	
1	1	4	7	8
2	5	6	9	12
3		10	11	13

$i_2 \backslash i_1$	0	1	2	3
0	0	1	5	6
1	2	4	7	12
2	3	8	11	14
3	9	10	13	15

Figure 6.3: Example of different index assignment strategies. These examples exhibit the possibility of trading off central and side distortions. The upper-left strategy is the most favorable to the side decoders, while the lower-right one is the best for the central decoder.

Assignment	D_0	D_1	D_2
Upper-left above	0.0052	0.0052	0.0052
Upper-right above	0.0006	0.021	0.024
Lower-left above	0.0004	0.026	0.036
Lower-right above	0.0003	0.041	0.046

Table 6.1: Performance of different multiple description scalar quantizers according to the assignment strategy shown in Figure 6.3. The distortion is the MSE and the input source is uniform over $[0, 1]$ [59].

In [104], the procedure to design fixed rate MD scalar quantizers was extended to the entropy-constrained case. In [103], an asymptotic high rate analysis of MD scalar quantizers is performed and is based on high resolution quantization theory. In particular, it is shown that at high rates, for the case of balanced descriptions ($R_1 = R_2 = R$) and Gaussian sources, the distortion product $D_0 D_1$ of the entropy-constrained MD scalar quantizer takes the form $\frac{1}{4}(\frac{\pi e}{6})^2 2^{-4R}$. At the same time, the MD rate distortion bound [84] (when put in distortion product form) becomes $\frac{1}{4}2^{-4R}$. This is an important result because it shows that for the multiple description scalar quantizer both the side and the central distortion attain the optimal exponential rate of decay ($D_0 \sim 2^{-2R}, D_1 \sim 2^{-2R}$). The only sub-optimality of MDSQ at high rates is due to the use of a scalar quantizer which partitions the space into cubic regions instead of an ideal vector quantizer that would optimally partition the space into spheres. Various constructions of MD vector quantizers have been proposed [29, 48, 70, 105] and the MD lattice quantizers of [105] do effectively close the gap between the performance of the entropy constrained MD scalar quantizer and the MD rate-distortion bound.

6.2.2 Multiple description coding with correlating transforms

A rather different approach pioneered by Wang et al. [83, 110] and then extended by Goyal and Kovačević [53, 54] consists of applying a suitable blockwise transform to the input vector before coding to obtain the MD property. This approach is usually called MD Transform Coding or MD coding with correlating transforms. In this case, instead of using MDSQ to produce two indices that describe the same quantity, the MD character is achieved with a linear transform that introduces correlation between a pair of random variable. The basic idea is to decorrelate the vector components and then to introduce again correlation between coefficients, but in a known and controlled manner, so that erased coefficients can be statistically estimated from those received.

The MD transform coder is illustrated in Figure 6.4. The usual assumption is that the two input components x_1, x_2 are correlated Gaussian variables. M represents the decorrelating transform (KLT), while T is the transform that reintroduces correlation. The decorrelated components are first scalar quantized

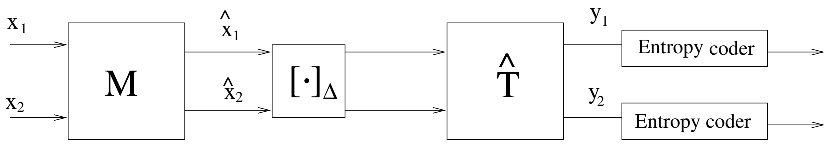

Figure 6.4: The encoding part of the multiple description transform coder with correlating transforms. The input components x_1, x_2 are correlated Gaussian variables. M is the decorrelating transform (KLT), $\hat{T}$ is the discrete version of the "recorrelating" transform T.

with a uniform scalar quantizer with stepsize Δ and then transformed with an invertible discrete transform $\hat{T}$. Recall that $\hat{T}$ is obtained by factoring the linear transform T into lifting steps and then by sequentially rounding each step of this decomposition [26]. The two components y_1, y_2 are entropy coded and sent over two independent erasure channels. When both components are received, the reconstruction process is just to invert $\hat{T}$ and the distortion is precisely the quantization error. If one component is lost, it is estimated from the received one and the distortion is dominated by the estimating error. This last error depends on the correlation introduced by $\hat{T}$. Call $R_{\hat{x}} = \mathrm{diag}(\sigma_1^2, \sigma_2^2)$ the correlation matrix of the decorrelated vector with the usual assumption that $\sigma_1^2 > \sigma_2^2$. Since in the high rate regime $\hat{T} \approx T$, the correlation matrix of y_1, y_2 is, with good approximation, given by $R_y = TR_xT^T$. Now, since quantization is fine, y_1, y_2 can be treated as Gaussian random variables quantized with a step size Δ. Therefore, if we call $R_{y_{11}}$ and $R_{y_{22}}$ the variances of y_1 and y_2, the rate necessary to code them is [53, 54]:

$$R_1 = \tfrac{1}{2}\log 2\pi e R_{y_{11}} - \log\Delta$$

$$R_2 = \tfrac{1}{2}\log 2\pi e R_{y_{22}} - \log\Delta,$$

whereas the rates necessary to code the decorrelated variables is:

$$R_1^* = \tfrac{1}{2}\log 2\pi e \sigma_1^2 - \log\Delta$$

$$R_2^* = \tfrac{1}{2}\log 2\pi e \sigma_2^2 - \log\Delta.$$

The difference $\rho = \frac{1}{2}(R_1 + R_2) - \frac{1}{2}(R_1^* + R_2^*)$ is called redundancy and represent the price we pay in rate in order to potentially reduce the distortion when there are erasures. So, the goal is to find a transform T that minimizes the side distortion (i.e., the distortion in case of erasure) for a given fixed ρ.

In the particular case where the two channels have the same erasure probability and the two components are coded at the same rate ($R_1 = R_2$), Goyal and Kovačević [53, 54] show that the optimal correlating transform is

$$T = \begin{bmatrix} a & \dfrac{1}{2a} \\ -a & \dfrac{1}{2a} \end{bmatrix}, \tag{6.1}$$

where the value of a depends on the redundancy ρ:

$$a = \sqrt{\frac{\sigma_2}{2\sigma_1(2^{2\rho} - \sqrt{2^{4\rho} - 1})}}. \tag{6.2}$$

The side distortion $D = D_1 = D_2$ is given by[2]

$$D = \frac{\sigma_1^2}{2} - \frac{1}{4\cdot 2^{2\rho}(2^{2\rho} - \sqrt{2^{4\rho} - 1})}(\sigma_1^2 - \sigma_2^2). \tag{6.3}$$

[2]It is interesting to notice that if the Gaussian source has a circularly symmetric probability density, i.e., $\sigma_1 = \sigma_2$, then the distortion is independent of ρ. In this case, the side distortion cannot be reduced with the addition of redundancy, so the approach based on correlating transforms is useless.

This result will be useful in the proof of Theorem 6.1. Notice that D starts at a maximum value of $(\sigma_1^2+\sigma_2^2)/2$ and asymptotically approaches a minimum value of σ_2^2.

Goyal and Kovačević give also analytical solutions to this problem in the most general setting where channels failures need not have equal probability or be independent, and R_1 is not necessarily equal to R_2. The interested reader can refer to [53, 54] for more details.

6.2.3 Multiple description coding of stationary sequences

MD scalar quantizers and MD transform coders with correlating transforms work in the case of memoryless sources or sources with finite memory. In particular, MD scalar quantizers treat each scalar sample independently and thus they are suited for memoryless sources. MD transform coders with correlating transforms operates on vectors of finite dimension and, thus, they are good for sources with finite memory.[3]

The problem of MD coding of stationary sources has been considered by Ingle and Vaishampayan in [67] and Batllo and Vaishampayan in [4]. In [67], Ingle et al. consider the problem of designing DPCM systems for MD coding of sources with memory. While Batllo et al. proposed a solution that combines the use of an orthogonal block transform and of MDSQ [4]. As for the MDSQ, this system has some good asymptotic properties. At low rates, however, except for some practical results obtained in the context of still image coding [91], much less is known. Note that Batllo and Vaishampayan use the term Multiple Description Transform Coder (MDTC) to refer to this system. From now on, we will also use that name to refer to their system. Since our main interest is in MD coding of stationary sources, we detail here the analysis of these two systems. We assume $R_1 = R_2 = R$ and $D_2 = D_1$.

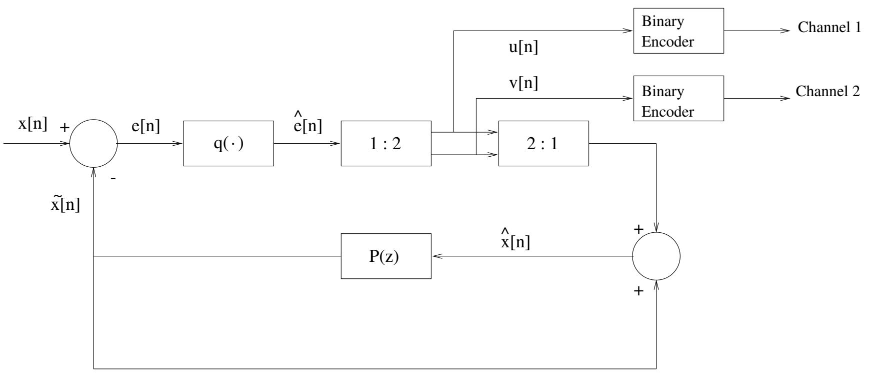

Figure 6.5: The MD-DPCM system.

The analysis part of the MD-DPCM system [67] is illustrated in Figure 6.5.

[3]To be precise, optimal MD transform coders are known only for the case of two channels and two-dimensional vectors. The case of larger vector is still open.

$P(z) = b_1 z^{-1} + b_2 z^{-2}$ is a second order predictor filter. The quantized predicted sequence $\hat{e}[n]$ is separated into two subsequences containing the even and the odd samples and these subsequences are sent over two different channels. If one subsequence is lost, it is linearly estimated using the received one. Now, if the input source is Gauss-Markov with regression coefficient α, it turns out that the estimating filters present in the synthesis part of the system are realizable filters. Moreover, in this case, the side distortion is given by [67]:

$$D_1 = \frac{1}{(1-\alpha^2)} \left(1 - \frac{(b_2^2 + 2b_2\alpha^2 + \alpha^2)}{2(\alpha^2 + b_2^2)}\right), \tag{6.4}$$

while the central distortion is [67]:

$$D_0 = (1 + b_2^2/\alpha^2)\epsilon^2 2^{-2R}, \tag{6.5}$$

where ϵ^2 depends on the kind of quantizer used and b_1, b_2 are related by the following equation: $b_1/\alpha + b_2/\alpha^2 = 1, \quad 0 \leq b_2 \leq \alpha^2$. An important point to note is that the side distortion of the MD-DPCM system does not go to zero even at infinite rate.

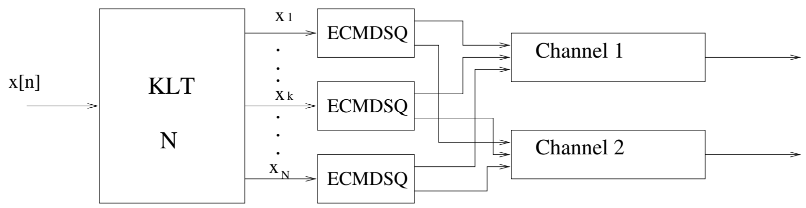

Figure 6.6: The Multiple Description Transform Coder.

The Multiple Description Transform Coder [4] is illustrated in Figure 6.6. It is represented by two main elements: a linear transform which turns out to be a KLT and a set of Entropy Constrained MDSQ. Recall that an MDSQ produces a pair of indices for each input scalar sample and that the behaviour of an MDSQ is characterized by the strategy in the assignment of the two output indices. This element defines the trade-off between side and central distortion. That is, it defines if the indices are assigned in a way to mainly minimize the central or the side distortion. The system works in the following way: it takes a block of N consecutive elements of the input sequence $x[n]$ and applies a KLT to them. Then each of the decorrelated component is encoded with a different Entropy Constrained MDSQ and the pair of indices produced by the MDSQ are transmitted over two separate channels. In case of Gaussian input sources and at high rates, optimal performance is achieved if the index assignment strategy is the same for each MDSQ and bits are allocated to each component according to a single description allocation strategy [4]. Finally letting N go to infinity and in the case of high rates, the performance of this system is given by [4]:

$$D_0 D_1 = \gamma 2^{-4R} \exp\left(\frac{1}{\pi}\int_{-\pi}^{\pi} \ln S(\omega) d\omega\right), \tag{6.6}$$

where D_0 is the central distortion, D_1 is the side distortion, R is the average rate per sample per channel, $S(\omega)$ is the input power spectral density and $\gamma = (\frac{\pi e}{12})^2$.

As for the case of MDSQ, in MDTC both side and central distortions decay exponentially with the rate. Since the side distortion of MD-DPCM does not go to zero even at infinite rate, it seems clear that, in the high rate regime, MDTC is superior to MD-DPCM. In Section 6.4, we analyze in detail the performance of these two systems and we compare them with the ideal bounds (Theorem 5.2 of the previous chapter) and with our MD system that we present in the next section.

6.3 Optimal two-channel filter banks for MD coding

We now present an algorithm for the design of optimal two-channel filter banks for MD coding of Gaussian sources. The filter banks are designed using an approach similar to the one proposed in the case of block transforms: we construct a first filter bank to decorrelate the two input sequences and then we use a second filter bank to efficiently recorrelate them. The main result of this section is Theorem 6.1 where we show that optimal filters are obtained by allocating the redundancy (excess rate) over frequency with a reverse "water-filling" strategy.

6.3.1 Problem formulation and notation

Consider the classical two-channel filter bank scheme shown in Figure 6.7. Here the input $x[n]$ is assumed to be a stationary Gaussian random process with known statistics and is fed through an analysis filter bank. The two output sequences are then separately quantized and sent over two different erasure channels. We suppose that the channels are independent, that they have the same erasure probability and that $R_1 = R_2$[4]. For convenience we will formulate our problem in the polyphase domain [101, 108]. In this case the analysis stage can be equivalently represented by the block scheme shown in Figure 6.8.

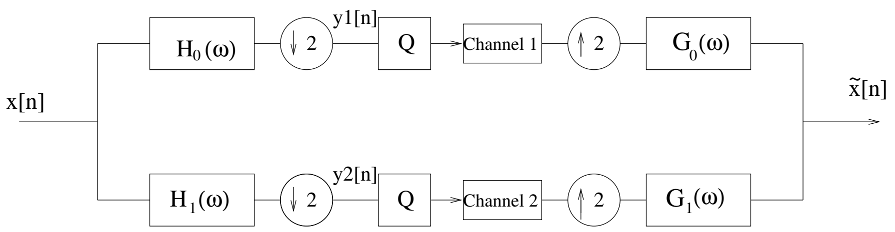

Figure 6.7: Two channel filter banks.

First we move the quantization step before the transform and approximate

[4]This last hypothesis, although reasonable, is not strictly necessary; but it simplifies the solution.

our continuous polyphase transform with a discrete one.[5] The discrete transform can be obtained by factoring the continuous one into a product of lifting steps and then sequentially rounding all these intermediate factors [10, 26]. It can be shown that the error due to this approximation can be bounded and that it goes to zero at high rates [52, 54]. The reason why we use this kind of structure is that if the quantization is performed before the transform, then the square partition cells are maintained. This enables the use of nonorthogonal transforms without increasing the quantization error. The importance of performing quantization before the transform in the MD case was pointed out for the first time in [83] (See also [109]). Since at high rates the difference between the discrete and the continuous transforms is small, our analysis will be based on properties of the continuous transform.

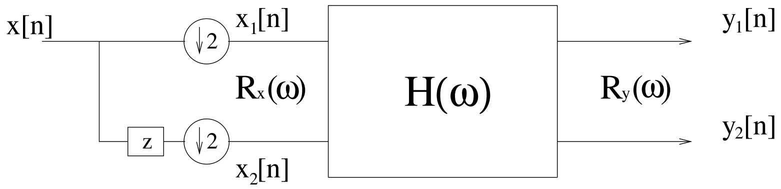

Figure 6.8: The polyphase representation of the analysis stage

Now consider again Figure 6.8. The input-output relation can be expressed in matrix notation introducing the analysis polyphase matrix $H(\omega)$:

$$\begin{pmatrix} Y_1(\omega) \\ Y_2(\omega) \end{pmatrix} = \begin{pmatrix} H_{11}(\omega) & H_{12}(\omega) \\ H_{21}(\omega) & H_{22}(\omega) \end{pmatrix} \begin{pmatrix} X_1(\omega) \\ X_2(\omega) \end{pmatrix}. \tag{6.7}$$

Call $R_x(\omega)$ the 2×2 polyphase power spectral density (p.s.d.) matrix of the input process. Likewise $R_y(\omega)$ is the p.s.d. matrix of the outputs. The system response has the following form:

$$R_y(\omega) = H(\omega) R_x(\omega) H^*(\omega), \tag{6.8}$$

where $H^*(\omega)$ denotes the Hermitian transpose of $H(\omega)$.
The synthesis part of the system can be analyzed in a similar fashion. Recall that, given the analysis matrix, the synthesis polyphase matrix $G(\omega)$ is uniquely defined (up to a phase factor). In fact $G(\omega)$ must be such that the condition $G(\omega)H(\omega) = I$ is satisfied [108].

Now, assume that the target central distortion is D_0 and that both channels are coded independently. Since $y_1[n], y_2[n]$ are stationary Gaussian sources and quantization is fine, the minimum bit rates necessary to scalar code the two

[5] By continuous transform we mean a generic linear operator in $l_2(\mathcal{Z})$. The discrete transform is a perfectly invertible operator that converts quantized sequences into quantized sequences [10, 66, 114].

sequences is [7]:

$$\begin{aligned} R_1(D_0) &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\frac{1}{2}\log\frac{R_{y_{11}}(\omega)}{D_0}d\omega + \frac{1}{2}\log(\frac{\pi e}{6}) \\ R_2(D_0) &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\frac{1}{2}\log\frac{R_{y_{22}}(\omega)}{D_0}d\omega + \frac{1}{2}\log(\frac{\pi e}{6}). \end{aligned} \tag{6.9}$$

In case we do not use any subband decomposition, the bit rate necessary to get the same central distortion D_0 is [7]:

$$R^*(D_0) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\frac{1}{2}\log\frac{S(\omega)}{D_0}d\omega + \frac{1}{2}\log(\frac{\pi e}{6}), \tag{6.10}$$

where $S(\omega)$ is the p.s.d. of the input process. We call redundancy the difference rate between these two cases:

$$\begin{aligned} \rho &= \frac{R_1(D_0)+R_2(D_0)}{2} - R^*(D_0) \\ &= \frac{1}{4\pi}\int_{-\pi}^{\pi}\frac{1}{2}\log\frac{R_{y_{11}}(\omega)R_{y_{22}}(\omega)}{S^2(\omega)}. \end{aligned} \tag{6.11}$$

Note that Eq. (6.9) holds because the transform is performed after the quantization. If the transform were performed before the quantization, the shape of the quantization cells would be affected and one should also consider this effect to compute the correct rates.

Now consider the case when one channel (e.g., channel 1) is cut off and $y_1[n]$ must be estimated from the received sequence $y_2[n]$. The optimal estimation is obtained by Wiener filtering:

$$\hat{Y}_1(\omega) = \frac{R_{y_{12}}(\omega)}{R_{y_{22}}(\omega)}Y_2(\omega). \tag{6.12}$$

Call $\eta(\omega)$ the error in predicting $Y_1(\omega)$ from $Y_2(\omega)$:

$$\eta(\omega) = Y_1(\omega) - \hat{Y}_1(\omega). \tag{6.13}$$

Since we have used a nonorthogonal transform, we must return to the original

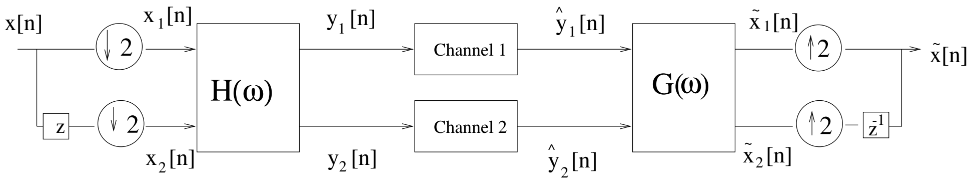

Figure 6.9: The complete MD system in the polyphase domain.

space in order to compute the distortion (mean square error distortion in our case); therefore (see also Figure 6.9):

$$\begin{aligned}\begin{pmatrix}\tilde{X}_1(\omega)\\ \tilde{X}_2(\omega)\end{pmatrix} &= \begin{pmatrix}G_{11}(\omega) & G_{12}(\omega)\\ G_{21}(\omega) & G_{22}(\omega)\end{pmatrix}\begin{pmatrix}Y_1(\omega)+\eta(\omega)\\ Y_2(\omega)\end{pmatrix}\\ &= \begin{pmatrix}X_1(\omega)\\ X_2(\omega)\end{pmatrix} + \begin{pmatrix}G_{11}(\omega)\eta(\omega)\\ G_{21}(\omega)\eta(\omega)\end{pmatrix}\end{aligned} \tag{6.14}$$

and

$$\|X(\omega) - \tilde{X}(\omega)\|^2 = \left\| \begin{pmatrix}G_{11}(\omega)\eta(\omega)\\ G_{21}(\omega)\eta(\omega)\end{pmatrix} \right\|^2. \tag{6.15}$$

Considering the fact that the error η is still a Gaussian process with p.s.d. $R_{y_{11}}(\omega) - \frac{|R_{y_{12}}(\omega)|^2}{R_{y_{22}}(\omega)}$ and using Parseval's relation we obtain:

$$\begin{aligned}D_2 &= E[\|x[n] - \tilde{x}[n]\|^2]\\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{2}(G^*_{21}(\omega)G_{21}(\omega) + G^*_{11}(\omega)G_{11}(\omega)) \cdot (R_{y_{11}}(\omega) - \frac{|R_{y_{12}}(\omega)|^2}{R_{y_{22}}(\omega)})d\omega\end{aligned} \tag{6.16}$$

and, finally, using the biorthogonal relations, we can express the distortion as a function of the analysis filters:

$$\begin{aligned}D_2 &= \frac{1}{4\pi}\int_{-\pi}^{\pi} \frac{1}{\det^2[H(\omega)]}(H^*_{21}(\omega)H_{21}(\omega) + H^*_{22}(\omega)H_{22}(\omega)) \cdot (R_{y_{11}}(\omega) - \frac{|R_{y_{12}}(\omega)|^2}{R_{y_{22}}(\omega)})d\omega\\ &= \frac{1}{4\pi}\int_{-\pi}^{\pi} (H^*_{21}(\omega)H_{21}(\omega) + H^*_{22}(\omega)H_{22}(\omega)) \cdot (\frac{\det[R_x(\omega)]}{R_{y_{22}}(\omega)})d\omega.\end{aligned} \tag{6.17}$$

Likewise, we can obtain an expression for the distortion D_1 associated with the loss of $y_2[n]$. Since the two channels have the same erasure probability, the expected distortion due to erasure (side distortion) is:

$$D = \frac{1}{2}(D_1 + D_2). \tag{6.18}$$

Note that in our formulation we have only considered the distortion due to erasure and have neglected the one due to quantization, since at high rates, it is much smaller.

Our target is to find a perfect reconstruction filter bank which minimizes the side distortion D. The perfect reconstruction condition is realized by the constraint: $\det[H(\omega)] = 1$.[6] The design of the filter bank is also constrained by the redundancy through equations (6.8) and (6.11). Thus, our optimization problem is to find a perfect reconstruction filter bank which minimizes the side distortion D for a given, fixed redundancy ρ.

[6] Strictly speaking the perfect reconstruction condition is satisfied if and only if $\det[H(\omega)] \neq 0$ on the unit circle. However, a factorization into lifting steps is possible only if $\det[H(\omega)]$ is a monomial [26] Since the side distortion (6.17) does not depend on the value of the determinant, we can assume, without loss of generality, $\det[H(\omega)] = 1$.

6.3.2 Optimal solution

As a first step we decompose the matrix $H(\omega)$ into the product of two matrices $M(\omega)$ and $T(\omega)$

$$H(\omega) = T(\omega)M(\omega). \tag{6.19}$$

$M(\omega)$ is a unitary decorrelating matrix that diagonalizes the input covariance matrix $R_x(\omega)$. Thus: $R_x(\omega) = M(\omega)\Lambda(\omega)M^*(\omega)$ where $\Lambda(\omega)$ is a diagonal matrix which contains the spectral eigenvalues of $R_x(\omega)$.

$$\Lambda(\omega) = \begin{bmatrix} \lambda_1^2(\omega) & 0 \\ 0 & \lambda_2^2(\omega) \end{bmatrix}. \tag{6.20}$$

For a stationary input process, the decorrelating matrix can be found analytically and has the following form [97] (see also [99, 98]):

$$M(\omega) = \frac{\sqrt{2}}{2} \begin{bmatrix} e^{j\omega/2} & 1 \\ -1 & e^{-j\omega/2} \end{bmatrix}; \tag{6.21}$$

the filter bank related to $M(\omega)$ is usually called the principal component filter bank [97]. Now, this factorization does not reduce the generality of the solution, since $M(\omega)$ is a unitary invertible matrix independent of ρ and we are considering square error distortions. So it is enough to solve the simpler problem of optimally designing the matrix $T(\omega)$ for the two input sequences with p.s.d. matrix $\Lambda(\omega)$. Then the final solution will be represented by the product between this matrix and the decorrelating matrix $M(\omega)$. From now on we will always assume that the two sequences $(x_1[n], x_2[n])$ have already been decorrelated and are represented by the diagonal p.s.d. matrix $\Lambda(\omega)$. Notice that these two sequences are still a realization of a stationary Gaussian process.

For the sake of clarity, we recall once more the results presented in [53],[54]. Here Goyal et al. study the problem of designing an optimal block transform to transmit two Gaussian variables over two erasure channels. As mentioned in Section 6.2.2, in the case the two channels have the same erasure probability and the two components are coded at the same rate ($R_1 = R_2$), they show that the optimal MD correlating transform, is:

$$T = \begin{bmatrix} a & \frac{1}{2a} \\ -a & \frac{1}{2a} \end{bmatrix}, \tag{6.22}$$

where the value of a depends on the redundancy ρ:

$$a = \sqrt{\frac{\sigma_2}{2\sigma_1(2^{2\rho} - \sqrt{2^{4\rho} - 1})}}; \tag{6.23}$$

σ_1^2 and σ_2^2 are the variances of the two Gaussian components, with the usual assumption that $\sigma_1^2 > \sigma_2^2$. Finally the side distortion is given by:

$$D = \frac{\sigma_1^2}{2} - \frac{1}{4 \cdot 2^{2\rho}(2^{2\rho} - \sqrt{2^{4\rho} - 1})}(\sigma_1^2 - \sigma_2^2). \quad (6.24)$$

We can now state the following theorem:

Theorem 6.1 *Assume that $\rho \gg 0$ and that the two p.s.d. $\lambda_1^2(\omega), \lambda_2^2(\omega)$ of the two decorrelated input sequences $x_1[n], x_2[n]$ are such that $\delta_1 > \Delta_2$ where δ_1 is the essential infimum of $\lambda_1^2(\omega)$ and Δ_2 is the essential supremum of $\lambda_2^2(\omega)$. Then the optimal analysis filters for MD Coding of $x_1[n]$ and $x_2[n]$ are represented by the following polyphase matrix:*

$$T(\omega) = \begin{bmatrix} a(\omega) & \frac{1}{2a(\omega)} \\ -a(\omega) & \frac{1}{2a(\omega)} \end{bmatrix},$$

where

$$a(\omega) = \sqrt{\frac{\lambda_2(\omega)}{2\lambda_1(\omega)(2^{2\rho(w)} - \sqrt{2^{4\rho(\omega)} - 1})}}$$

and

$$\rho(\omega) = \rho + \frac{1}{4}\log(\lambda_1^2(\omega) - \lambda_2^2(\omega)) - \frac{1}{8\pi}\int_{-\pi}^{\pi} \log(\lambda_1^2(\omega) - \lambda_2^2(\omega))d\omega.$$

Proof: Consider only N consecutive elements of the first channel sequence $x_1[n]$, and N consecutive elements of the second channel sequence $x_2[n]$ which are located at the same temporal interval. Call ϕ_{1N} and ϕ_{2N} the two $N \times N$ corresponding correlation matrices. Apply a KLT to each of the two N-sequences to get independent components and name Y_{1N} and Y_{2N} the two N-sequences after the transformation. Call $\lambda_{1i}^2, \quad i = 1, ..., N$, the variances related to the N-sequence Y_{1N} and $\lambda_{2i}^2, \quad i = 1, ..., N$, the variances related to the second N-sequence (Y_{2N}). Since ϕ_{1N} and ϕ_{2N} are Hermitian Toeplitz matrices it results (see Appendix 5.A) that:

$$\delta_1 \leq \lambda_{1i}^2 \leq \Delta_1 \quad \forall i \quad (6.25)$$

and that:

$$\delta_2 \leq \lambda_{2i}^2 \leq \Delta_2 \quad \forall i. \quad (6.26)$$

where Δ_j and $\delta_j, \quad j = 1, 2$, are the essential suprema and the essential infima of the power spectral densities $\lambda_1^2(\omega)$ and $\lambda_2^2(\omega)$. Equations (6.25) and (6.26) imply that $\lambda_{1i}^2 > \lambda_{2i}^2, \quad i = 1, ..., N$, since we assumed $\delta_1 > \Delta_2$.

Now, consider the generic i-th couple of elements (y_{1i}, y_{2i}). We can apply the results of [53] to this pair and say that if we are allowed to use a redundancy

ρ_i then the optimal correlating transform for that pair is:

$$T_i = \begin{bmatrix} a_i & \frac{1}{2a_i} \\ -a_i & \frac{1}{2a_i} \end{bmatrix}, \tag{6.27}$$

where a_i is given by:

$$a_i = \sqrt{\frac{\lambda_{2i}}{2\lambda_{1i}(2^{2\rho_i} - \sqrt{2^{4\rho_i} - 1})}}, \tag{6.28}$$

and that the side distortion is:

$$D_i = \frac{\lambda_{1i}^2}{2} - \frac{1}{4 \cdot 2^{2\rho_i}(2^{2\rho_i} - \sqrt{2^{4\rho_i} - 1})}(\lambda_{1i}^2 - \lambda_{2i}^2). \tag{6.29}$$

However, we want to minimize the global side distortion:

$$D = \frac{1}{N}\sum_i D_i, \tag{6.30}$$

given a global redundancy budget

$$\rho = \frac{1}{N}\sum_i \rho_i. \tag{6.31}$$

This is a typical problem of constrained minimization, so we define a new cost function L which combines the distortion and the redundancy through a positive Lagrange multiplier ν:

$$\begin{aligned} &L = D + \nu\rho, \\ &L_i = D_i + \nu\rho_i, \qquad\qquad i = 1, 2, .., N. \end{aligned} \tag{6.32}$$

Finding a minimum of L amounts to finding minima for each L_i (because the costs are additive). Writing distortion as a function of the redundancy, $D_i(\rho_i)$, and taking derivatives we get:

$$\frac{\partial L_i}{\partial \rho_i} = \frac{\partial D_i}{\partial \rho_i} + \nu = 0. \tag{6.33}$$

Thus, for a solution to be optimal, the set of chosen redundancy ρ_i has to correspond to constant-slope points on their respective distortion-redundancy curves. Uniqueness follows from the convexity of these curves and from the use of the Kuhn-Tucker conditions when necessary [13]. A constant-slope solution is obtained for any fixed value of ρ. To enforce the constraint (6.31) exactly, one has to search over all the values of ν until the budget is met. However, if we suppose that ρ_i is sufficiently large then it is possible to give a closed form for the allocation problem. In fact, it follows that:

$$\frac{\partial D_i}{\partial \rho_i} = -\frac{\ln 2(\lambda_{1i}^2 - \lambda_{2i}^2)}{4 \cdot 2^{2\rho_i}(\sqrt{2^{4\rho_i}-1})} \approx -\frac{\ln 2}{4}(\lambda_{1i}^2 - \lambda_{2i}^2)2^{-4\rho_i} = -\nu. \tag{6.34}$$

The constant-slope solution forces the redundancies to be of the following form

$$\rho_i = \alpha + \frac{1}{4}\log(\lambda_{1i}^2 - \lambda_{2i}^2). \tag{6.35}$$

Using the redundancy constraint (6.31)

$$\sum_i \rho_i = N\alpha + \frac{1}{4}\sum_i \log(\lambda_{1i}^2 - \lambda_{2i}^2) = N\rho, \tag{6.36}$$

we find

$$\alpha = \rho - \frac{1}{4N}\sum_i \log(\lambda_{1i}^2 - \lambda_{2i}^2) \tag{6.37}$$

and finally

$$\rho_i = \rho + \frac{1}{4}\log(\lambda_{1i}^2 - \lambda_{2i}^2) - \frac{1}{4N}\sum_i \log(\lambda_{1i}^2 - \lambda_{2i}^2). \tag{6.38}$$

The approximation in Eq. (6.34) holds if ρ_i is sufficiently large. Its value depends on the total redundancy budget ρ and on the difference $\lambda_{1i}^2 - \lambda_{2i}^2$. The difference $\lambda_{1i}^2 - \lambda_{2i}^2$ influences the slope of the distortion-redundancy curves (6.34). Now, the global distortion is minimized when the set of chosen redundancy ρ_i corresponds to constant slope points. If $\lambda_{1i}^2 - \lambda_{2i}^2 = 0$, the slope of the ith curve is zero and the optimal solution is always found imposing the Kuhn-Tucker condition: $\rho_i = 0$. For this reason the approximation in Eq. (6.34) holds only when both the conditions $\rho \gg 0$ and $\delta_1 > \Delta_2$ are verified ($\delta_1 > \Delta_2$ implies $\lambda_{1i}^2 > \lambda_{2i}^2, \quad \forall i$). In general we can say that the difference $\lambda_{1i}^2 - \lambda_{2i}^2$ influences the allocation strategy of the redundancy. The redundancy is mainly allocated in the region where this difference is higher.

Now we can let N go to infinity and find, in this way, the optimal spectral distribution of the redundancy:

$$\rho(\omega) = \rho + \frac{1}{4}\log(\lambda_1^2(\omega) - \lambda_2^2(\omega)) - \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{4}\log(\lambda_1^2(\omega) - \lambda_2^2(\omega))d\omega. \tag{6.39}$$

Once $\rho(\omega)$ is known, we can obtain the expression of the side distortion D:

$$D = \frac{1}{2\pi} \int_{-\pi}^{\pi} D(\omega) d\omega, \tag{6.40}$$

where:

$$D(\omega) = \frac{\lambda_1^2(\omega)}{2} - \frac{1}{4 \cdot 2^{2\rho(\omega)}(2^{2\rho(\omega)} - \sqrt{2^{4\rho(\omega)} - 1})} (\lambda_1^2(\omega) - \lambda_2^2(\omega)) \tag{6.41}$$

and the expression of the polyphase matrix $T(\omega)$:

$$\begin{bmatrix} a(\omega) & \frac{1}{2a(\omega)} \\ -a(\omega) & \frac{1}{2a(\omega)} \end{bmatrix}, \tag{6.42}$$

where:

$$a(\omega) = \sqrt{\frac{\lambda_2(\omega)}{2\lambda_1(\omega)(2^{2\rho(\omega)} - \sqrt{2^{4\rho(\omega)} - 1})}}. \tag{6.43}$$

□

When the approximation (6.34) is not verified, namely when at least one of the the two hypotheses $\rho \gg 0$ and $\delta_1 > \Delta_2$ is not satisfied, the optimal allocation of the redundancy over frequency can only be found numerically. This means that, for any fixed ν, one has to numerically solve Eq. (6.34) and then has to search over all the values of ν until the constraint (6.31) is met.

Consider, now, equations (6.40) and (6.41). They express the side distortion in function of the spectral distribution of the redundancy ρ. The side distortion is maximum when we are not allowed to allocate any redundancy over the frequency and its maximum value is:

$$D = \frac{1}{8\pi} \int_{-\pi}^{\pi} (\lambda_1^2(\omega) + \lambda_2^2(\omega)) d\omega. \tag{6.44}$$

Its mininum value occurs when we can allocate an infinite amount of redundancy over the frequency and it is equal to:

$$D = \frac{1}{4\pi} \int_{-\pi}^{\pi} \lambda_2^2(\omega) d\omega. \tag{6.45}$$

This value represents the *systematic error* due to the estimation of one subsequence with the other one and cannot be eliminated even at infinite redundancy. The systematic error typically occurs in MD systems based on correlating transforms [54, 109]. This is in contrast with the performance of other systems (i.e. MDTC), where at high rates both side and central distortions decrease with the rate. Thus, this result gives us a first insight about the performance of the filter bank system:

Corollary 6.1 *The filter bank system is not useful at high rates since, independently of the amount of redundancy allocated, the side distortion has a constant factor (the systematic error) that cannot be eliminated.*

6.3.3 Approximate FIR solutions

In general, the filters obtained with the optimization algorithm of the previous section are of infinite length. However, in some applications it is important to approximate them with FIR filters. Let us call $H_f(\omega)$ the polyphase matrix related to the FIR filter bank and D_f the corresponding side distortion obtained with this set of filters. Clearly $D_f \geq D$, where D is the ideal side distortion given by (6.18), since the best performance is usually achieved with infinite length filters. Now, the problem is to design a perfect reconstruction FIR filter bank that minimizes the performance gap $D_f - D$ for each fixed redundancy.

We solve this problem numerically by running a constrained minimization algorithm using a gradient descendent approach. The convex function to minimize is $\|D - D_f\|^2$, while the constraints are: the perfect reconstruction condition: $\det[H_f(\omega)] = 1$ and the maximum allowed redundancy ρ.

Recall that given an FIR analysis filter bank, perfect reconstruction with FIR filters is possible if and only if $\det[H(\omega)]$ is a monomial [108]. So, once we have designed FIR analysis filters with the constraint $\det[H_f(\omega)] = 1$, we know that it is possible to reconstruct the signal with FIR synthesis filters. These synthesis filters are obtained in the usual way:

$$G_0(\omega) = e^{j\omega} H_1(\omega + \pi),$$

$$G_1(\omega) = -e^{j\omega} H_0(\omega + \pi).$$

Finally, recall that once the FIR filter bank is obtained, it can always be factored into a finite number of lifting steps. These steps can be sequentially rounded and, in this way, one can obtain the discrete version of the continuous transform.

6.3.4 Application to a Gauss-Markov process

To conclude this section, we apply our filter design techniques to a Gaussian source and analyze the filter responses.

We consider a Gauss-Markov source $x[n] = \alpha x[n-1] + w[n]$, where the regression coefficient α has magnitude less than 1 and where $w[n]$ is a zero mean, unit variance, i.i.d. Gaussian source. The p.s.d. of this process is:

$$S_x(\omega) = \frac{1}{|1-\alpha e^{-j\omega}|^2}. \tag{6.46}$$

Now, the polyphase matrix $H(\omega)$ of the optimal filter bank is given by the product of the matrix $T(\omega)$ with the matrix $M(\omega)$. This second one is known and is given by (6.21). To design $T(\omega)$ we need to compute the spectral eigenvalues of the input p.s.d. matrix. First notice that the two subsequences obtained by downsampling $x[n]$ are still Gauss-Markov processes, but with the regression coefficient α replaced by α^2 and the i.i.d. original Gaussian source $w[n]$ replaced by a new i.i.d. Gaussian source with zero mean and variance $1+\alpha^2$. Hence the power spectral densities for these two processes are given by:

$$R_{x11}(\omega) = R_{x22}(\omega) = \frac{1+\alpha^2}{|1-\alpha^2 e^{-j\omega}|^2}. \tag{6.47}$$

The cross p.s.d. $R_{x12}(\omega)$ is given by:

$$R_{x12}(\omega) = \frac{\alpha(1+e^{(-j\omega)})}{1+\alpha^2} R_{x11}(\omega), \tag{6.48}$$

with $R_{x21}(\omega) = R^*_{x12}(\omega)$. Finally the p.s.d. matrix after decorrelation is:

$$\Lambda(\omega) = \begin{bmatrix} R_{x11}(\omega)\left(1+\frac{2\alpha\cos(\omega/2)}{(1+\alpha^2)}\right) & 0 \\ 0 & R_{x11}(\omega)\left(1-\frac{2\alpha\cos(\omega/2)}{(1+\alpha^2)}\right) \end{bmatrix}. \tag{6.49}$$

Observe that the two spectral eigenvalues are equal only at π (and of course at $-\pi$). As previously stated, at the points closest to the frequency values where $\lambda_1^2(\omega) = \lambda_2^2(\omega)$ it is not possible to use the closed-form (6.39) even in the high redundancy hypothesis. So, for the Gauss-Markov source, $a(\omega)$ (and consequently $T(\omega)$) can only be found numerically.

In Figure 6.10, we show the frequency responses of the two analysis filters as a function of the redundancy for the case $\alpha = 0.9$. It is interesting to notice that the amplitudes of the two frequency responses are exactly the same; the two filters differ only for the phase response. This is due to the presence of the principal component filter bank given by $M(\omega)$ and to the constraint $R_1 = R_2$ which forces the matrix $T(\omega)$ to have the shape given by Eq. (6.42). Moreover, notice that at high redundancies the two filters tend to be low-pass. In the case of $\alpha = 0.9$, the Gauss-Markov process is a low-pass process, thus the frequency responses of the two filters tend to preserve the frequency region where the p.s.d. of the input process is mostly concentrated. This is valid in general, that is at high redundancies the analysis filters better preserve the region where most of the p.s.d. of the input process is concentrated. It is also of interest to note that, at low redundancies, the two filter responses do not tend to be that of a principal component filter bank, that is an ideal low pass and an ideal high pass filter. This is because, if quantization is performed before the transform, the principal component filter bank does not represent the only solution that gives minimum coding rates. The same phenomenon happens in the block transform case, where the KLT does not represent the only transform that gives minimum rates if quantization is performed before the transform [52, 53]. Thus, this additional degree of freedom makes it possible to have a filter bank (or a block transform [53]) that achieves, at the same time, minimum coding rates and balanced rates. This is the solution that we have at low redundancies.

Finally, in Figure 6.11 we show the frequency response of the two FIR analysis filters obtained with the minimization algorithm presented in Section 3.3. The filters are all of length 6. It is interesting to compare these frequency responses with the ones in Figure 6.10. In the FIR case, the amplitude responses of the two filters are not equal, but they tend to be close to each others at high redundancies. Moreover, in the high redundancy region the two frequency responses tend to be low-pass as in the ideal case.

6.4 Performance analysis

In order to assess the performance of the filter bank proposed in the previous section, we compare it with the asymptotic ideal bounds found in Section 5.6 and with the MD Transform Coder [4] and the MD-DPCM system [67]. In the simulations, we consider two different Gaussian input sources: a classical Gauss-Markov source and a low-pass Gaussian source obtained as illustrated in Figure 6.12, where $x_1[n]$ and $x_2[n]$ are two i.i.d. Gaussian sources with variances σ_1^2, σ_2^2 and $G(\omega)$ is an ideal low-pass filter. Moreover, we consider two different scenarios: high rate, infinite delay/complexity and low rate, finite delay/complexity. In the first scenario, the analysis and the results presented in the first part of this chapter are valid. The second more realistic simulation is important, because we do not have clear theoretical answers on the behaviour of the considered systems in this particular context.

6.4.1 High rate, infinite complexity performance

We consider a first order Gauss-Markov source. In the high rate and infinite complexity hypothesis, the performance of the Multiple Description Transform Coder is given by (6.6) where $S(\omega)$ is given by (6.46). The side and central distortions of the MD-DPCM are given by (6.4) and (6.5). For the filter bank case, the filter responses are obtained numerically as shown in Section 6.3.4. Given the filter responses, the side distortion at high rates is given by Eqns. (6.17) and (6.18). The central distortion is obtained by numerically inverting the equations in (6.9).

In Figure 6.13, we compare the four performances: MD Transform Coder, MD-DPCM, MD filter bank and ideal bounds, for the case of $\alpha = 0.9$ and $R = 6$ bit/sample/channel. As we can see the MDTC outperforms the other two systems. This is not astonishing since in the MDTC both the central and

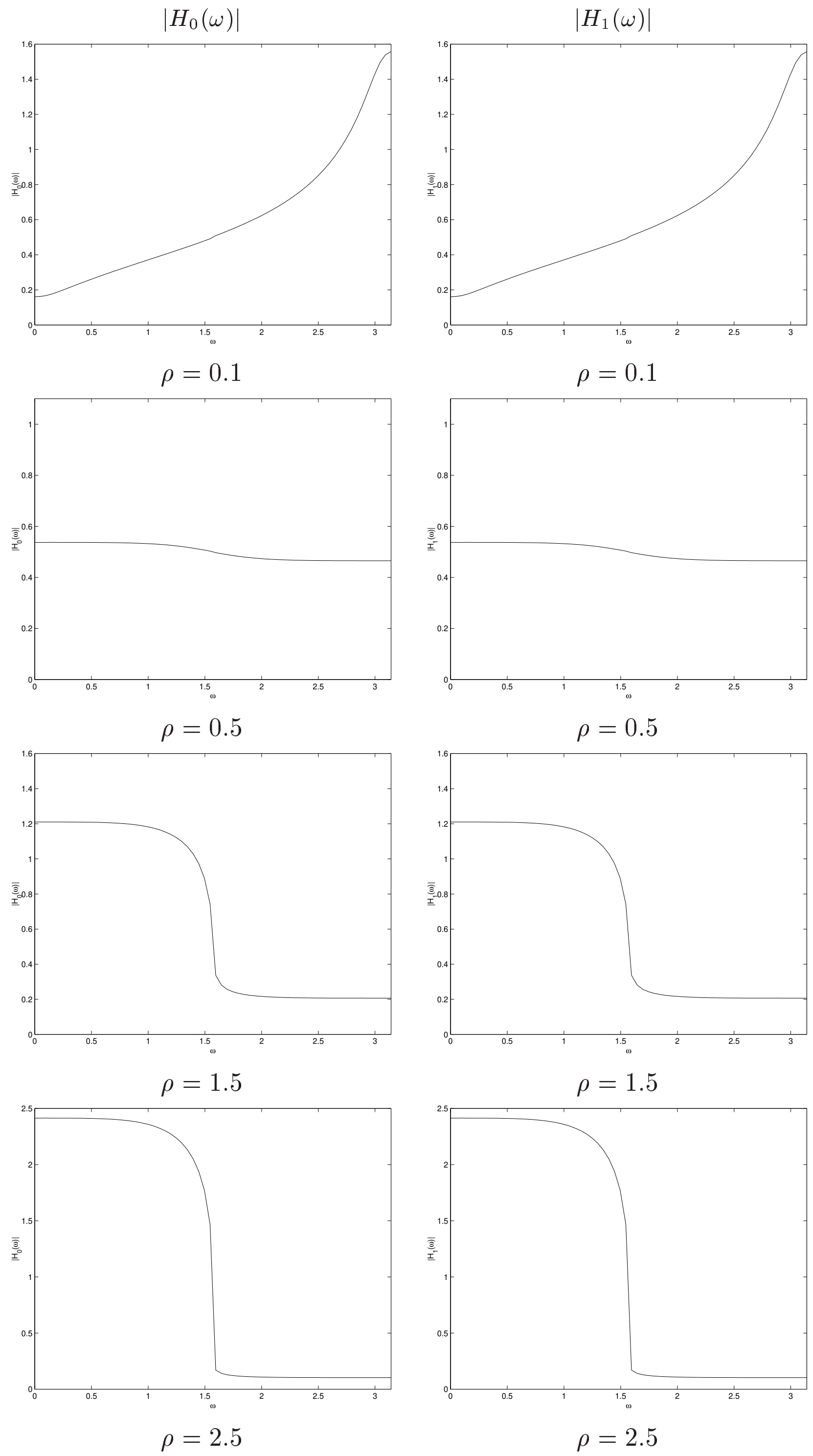

Figure 6.10: Frequency response of the analysis filters in function of the redundancy ρ.

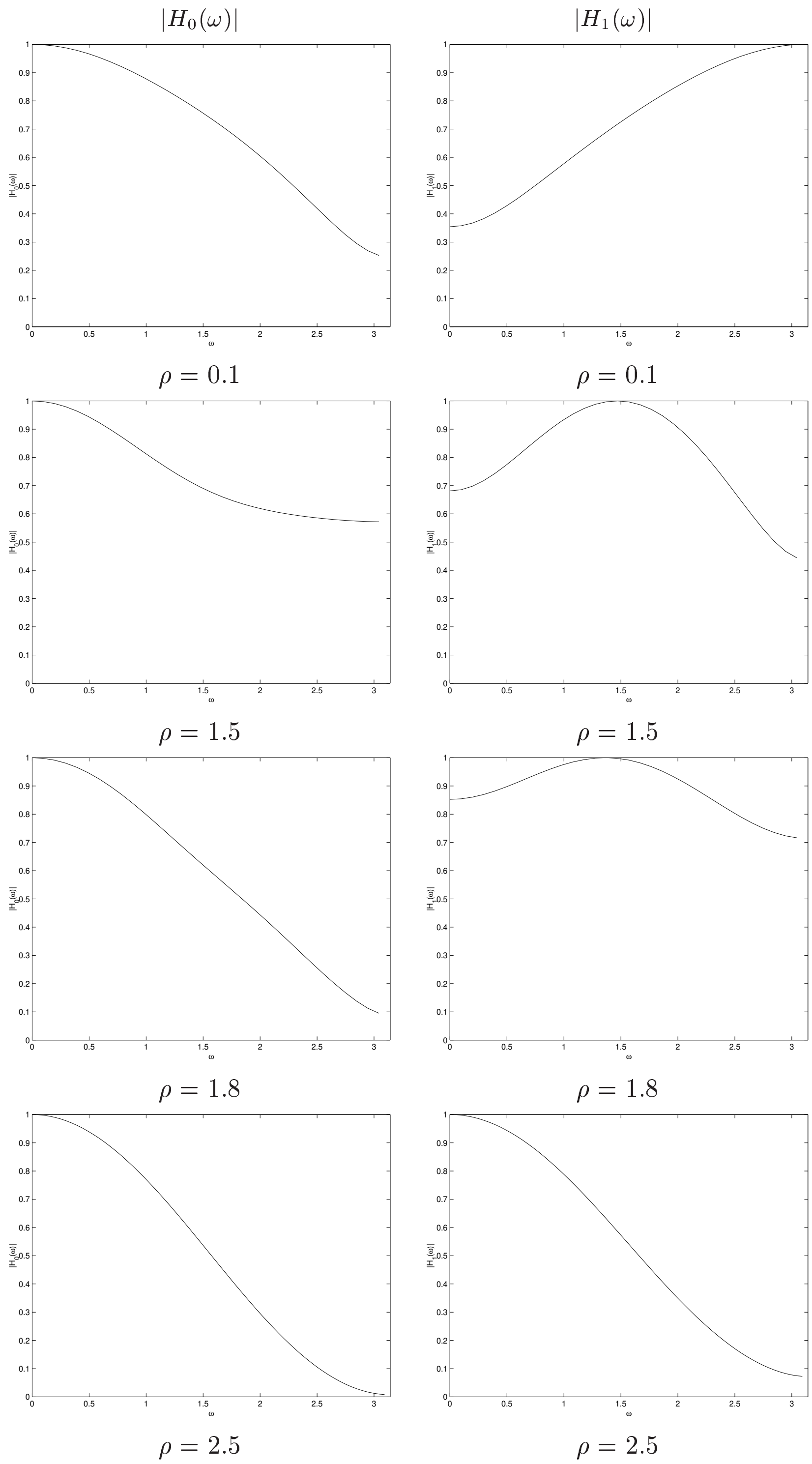

Figure 6.11: Frequency response of the analysis FIR filters in function of the redundancy ρ.

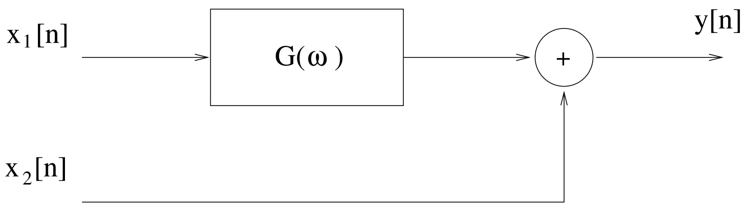

Figure 6.12: A low-pass Gaussian source. $G(\omega)$ is an ideal low-pass filter and $x_1[n]$, $x_2[n]$ are two i.i.d. Gaussian sources.

the side distortions decrease exponentially with the rate R. The side distortions of the MD-DPCM system and of our system suffer of the systematic estimation error that becomes dominant at high rates and that does not reduce with the rate. It is also interesting to note that the gap between the ideal bounds and the MDTC is constant and equal to 3.06dB. This confirms that this system attains asymptotically optimal performance.

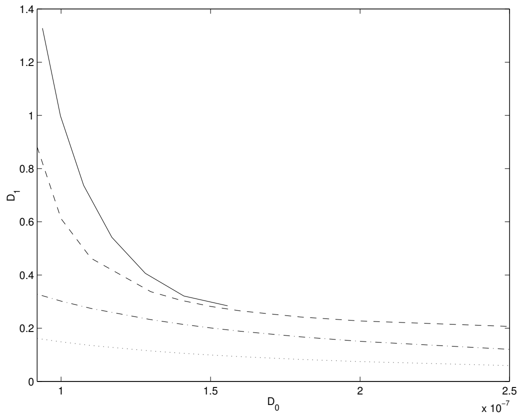

Figure 6.13: Asymptotic performance for a Gauss-Markov input source. Abscissa: central distortion, Ordinate: side distortion. Dotted: ideal bounds, dashed-dotted: Multiple Description Transform Coder, dashed: filter banks for MD coding, line: MD-DPCM.

6.4.2 Low rate, finite delay/complexity performance

In practical settings, we are more interested in low rate behaviours and we have to deal with finite delay/complexity constraints. That means that either the KLT or the filters in the filter bank have finite length N. The FIR filters are designed using the numerical optimization presented in Section 6.3.3.[7] The Multiple Description Transform Coder is the same shown previously except that

[7]For simplicity, we did not decompose the filters into lifting steps. Doing this decomposition would slightly improve the filter bank scheme.

the KLT operates on blocks of finite length N. Bits are still allocated according to a single description allocation strategy and the MD scalar quantizers are designed such that the index assignment strategy is the same for each of the N components. The MD-DPCM system is made of realizable filters and does not need to be approximated.

In the first simulation, we consider again a first order Gauss-Markov process with memory $\alpha = 0.9$. Numerical results are shown in Figure 6.14. Here, we consider two bit-rates $R = 2$ and $R = 3$ bit/sample/channel and two different length constraints: $N = 6$ and $N = 8$. The graphs show the trade-off between side and central distortion for the three systems. The first interesting thing to note is that, in the low rate regime, the Multiple Description Transform Coder, which is optimal at high rates, is generally outperformed by the other two systems. The MD-DPCM system is the best system in this context. Moreover, comparing the results of Figure 6.13 and Figure 6.14, one can conclude that, in this case, our system can attain the same performance as the MD-DPCM system only at the prize of infinite delay/complexity (i.e. with infinite length filters).

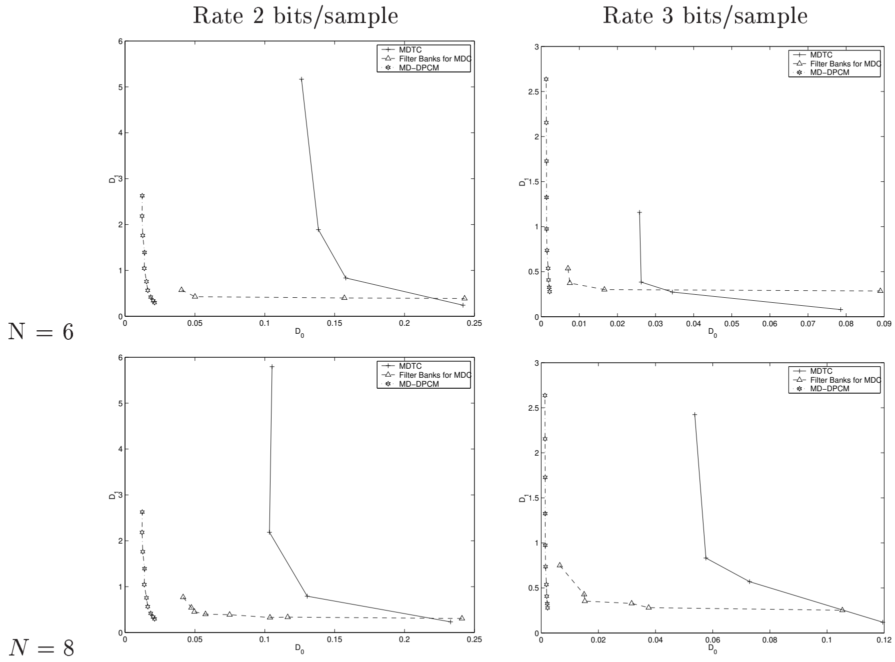

Figure 6.14: Comparison between Multiple Description Transform Coder, filter banks for MD coding and MD-DPCM system. Input source: Gauss-Markov. Line: Multiple Description Transform Coder, dashed: filter banks for MD coding, dashed-dotted: MD-DPCM.

It seems that one of the reasons why the MD-DPCM system is superior to the other two is because it has been designed assuming that the input source is

Gauss-Markov and, thus, it well exploits the particular structure of this source. The other two systems do not take particular advantage of the characteristics of the input source. For this reason, it is of interest to run similar simulations with a different Gaussian source. We consider a low-pass Gaussian source obtained as illustrated in Figure 6.12, where $x_1[n]$ and $x_2[n]$ are two i.i.d. Gaussian sources with variances $\sigma_2^2 = 1.5$ and $\sigma_2^2 = 0.5$ respectively. We consider two bit-rates $R = 1$ and $R = 2$ bit/sample/channel and two different length constraints: $N = 6$ and $N = 8$. Numerical results are shown in Figure 6.15. We can see that, in this context, our system is the best system in the medium-low redundancy region. It is also of interest to note that the performance gap between our system and the MDTC reduces with the rate, in particular we have noticed that for rates greater than $R = 3$ bit/sample/channel the MDTC performs better also at low redundancies (See Figure 6.16). Finally, the performance of these two systems slightly increases with the length N but, this length does not changes the performance gap between them.

In conclusion, this set of experiments indicate that, in the low bit rate regime and at low and medium redundancies, MD-DPCM and filter bank system perform better than the MDTC. Moreover, in this regime and for some classes of Gaussian sources, our system outperforms the other two.

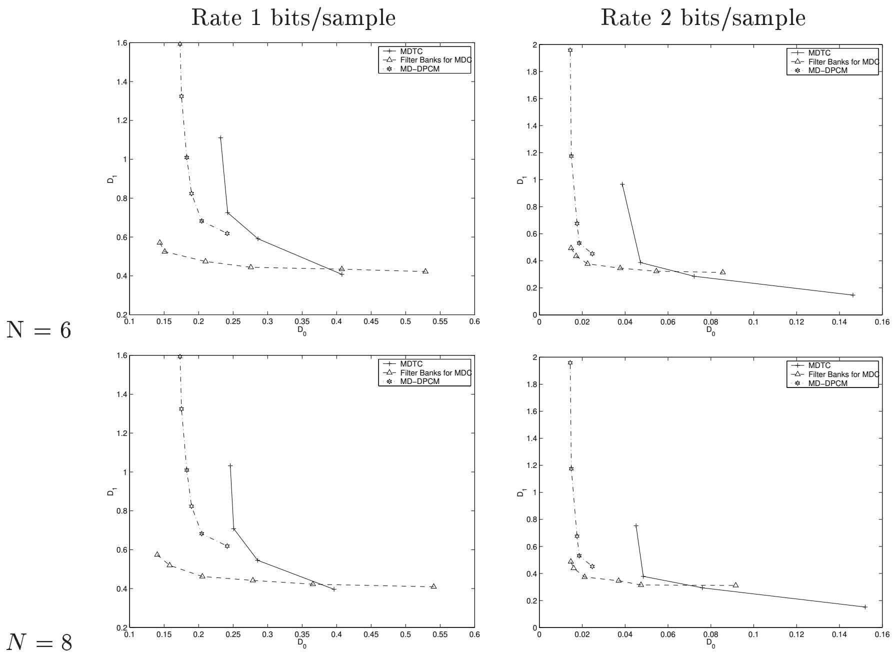

Figure 6.15: Comparison between Multiple Description Transform Coder, filter banks for MD coding and MD-DPCM system. Input source: low-pass Gaussian source. Line: Multiple Description Transform Coder, dashed: filter banks for MD coding, dashed-dotted: MD-DPCM.

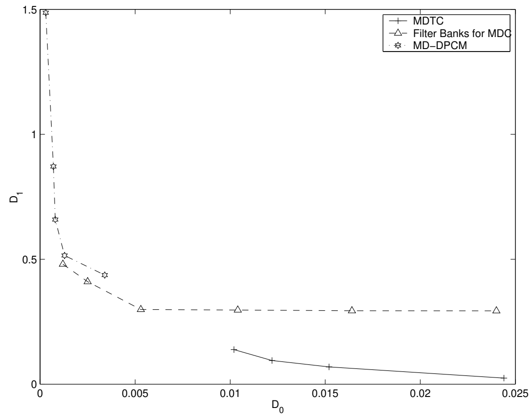

Figure 6.16: Comparison between Multiple Description Transform Coder,filter banks for MD coding and MD-DPCM system in the case of $R = 4$ bits/sample/channel. Input source: low-pass Gaussian source. Line: Multiple Description Transform Coder, dashed: filter banks for MD coding, dashed-dotted: MD-DPCM.

Chapter 7

Conclusions

In the introduction, we explained the importance of overcomplete expansions or frames. With a simple example, we showed that frames are more flexible than bases and that in many situations they are more useful than bases. In this thesis, we have studied redundant expansions with an interest on different possible applications. We summarize the key results of this work and discuss open issues below.

7.1 Summary

Oversampled filter banks for communications
In a recent work by Goyal and Kovačević [55], the redundancy present in a frame was used to reduce the effect of losses in a communication system. In that work, frame elements belonged to $\mathbb{R}^N$ (or $\mathbb{C}^N$). We investigated the more general case of frames with elements in $l_2(\mathbb{Z})$, which can be implemented with oversampled filter banks. We showed that filter banks implementing uniform and strongly uniform tight frames are the best in this context.

Wavelet footprints: theory and applications
We have seen that in many core signal processing problems, it is important to have a compact signal representation. We proposed a new way to represent piecewise polynomial signals in terms of elements which we called footprints. Footprints make up an overcomplete set of vectors and allow a compact representation of piecewise polynomial signals. Footprints are efficient at representing the singular structures of a signal. Numerical simulations showed that footprints outperform wavelet methods in several applications, namely denoising, deconvolution and compression. Together with the simplicity of algorithms involved this showed the usefulness of this new representation.

Multiple description coding
Multiple description coding is a source coding technique for data transmission over unreliable networks. We reviewed the fundamental theoretical results

on multiple description coding. Moreover, we found a multiple description rate distortion region for stationary Gaussian sources. This region generalizes the multiple description rate distortion region of El Gamal, Cover and Ozarow [45, 84] for memoryless Gaussian sources.

Practical multiple description coding techniques
Several systems which meet the multiple description constraints have been proposed in the last years. We reviewed the most popular among those systems. Then we presented an optimal way to design two channel filter banks for multiple description coding of Gaussian sources. We made a comprehensive analysis of the performance of this system and showed situations in which our system is superior to other multiple description systems.

7.2 Future Research

The development of new signal representations is an active research area in the signal processing community. In this thesis we have proposed new signal expansions which are promising and lead to a variety of possible expansions. We give an overview of future research directions.

Frames for communications
The purpose of our work was to come up with general methods to design oversampled filter banks for communication problems. We did not concentrate on the problem of designing application-specific filter banks. Now, the family of filters which we proved to be optimal in our communication settings, is still quite large. Therefore, it is of interest to see if it is possible to design filter banks which are optimal in our communication scenario and which show other desirable properties (which can be useful in other applications) such as good space-frequency localization.

Two-dimensional footprints
We have seen that footprints are a powerful tool for processing one-dimensional piecewise smooth signals. Thus, it is natural to look for an extension to the two dimensional case. In this case, the goal is to come up with efficient representations of images.

In images, edges and contours carry most of the information. These contours are usually regular, therefore the wavelet coefficients generated by them are spatially correlated. A good two dimensional footprint expansion should exploit this spatial correlation. A possible way to exploit this correlation is to perform the one dimensional footprint transform on the rows and the columns of the image and to chain the footprint coefficients generated by a contour to form a one-dimensional signal. Finally, this one-dimensional signal can be represented again with footprints. This idea of chaining of footprints is similar to the idea of bandelets introduced in [75]. However, by applying a footprint transform on the chain of footprints, one completely exploits the spatial correlation and the correlation across scales of the wavelet coefficients generated by a contours. Bandelets do not exploit the interscale dependency.

Multiple description coding for more than two descriptions
The main theoretical results on multiple description coding apply only to the case of two descriptions. The open problem is to find tight bounds for the case of more than two descriptions. This is also of interest for practical applications, since, in practice, one wants to create more than two descriptions of a source. Moreover, it is of interest to employ practical multiple description systems on concrete applications such as transmission of video over internet.

Bibliography

[1] R. Ahlswede. The rate distortion region for multiple descriptions without excess rate. *IEEE Trans. Information Theory*, 31(6):721–726, November 1985.

[2] N.I. Akhiezer and I. M. Glazman. *Theory of linear operators in Hilbert Spaces.* Volume I, Frederick Ungar Publisher, 1966.

[3] M.R. Banham and A.K. Katsaggelos. Spatially adaptive wavelet-based multiscale image restoration. *IEEE Trans. Image Processing*, 5:619–634, April 1996.

[4] J.-C. Batllo and V.A. Vaishampayan. Asymptotic performance of multiple description transform codes. *IEEE Trans. Information Theory*, 43(2):703–707, March 1997.

[5] J.J. Benedetto and D. Colella. Wavelet analysis od spectrogram seizure chirps. In *SPIE Wavelet Appl. in Signal and Image Proc.*, San Diego, CA, July 1995.

[6] J.J. Benedetto and D. Colella. Wavelet periodicity detection algorithms. In *SPIE Wavelet Appl. in Signal and Image Proc.*, San Diego, CA, July 1998.

[7] T. Berger. *Rate Distortion Theory.* Prentice-Hall, Englewood Cliffs, NJ, 1971.

[8] K. Berkner, M.J. Gormish, and E.L. Schwartz. Multiscape sharpening and smoothing in Besov spaces with applications to image enhancement. *Applied and Computational Harmonic Analysis*, 11:2–31, July 2001.

[9] H. Bolcskei, F. Hlawatsch, and Feichtiger H.G. Frame-theoretic analysis of oversampled filter banks. *IEEE Trans. Sig. Proc.*, 46(12):3256–3269, December 1998.

[10] R.C. Calderbank, I. Daubechies, W. Sweldens, and Yeo B.-L. Wavelet transforms that map integers to integers. *Applied and Computational Harmonic Analysis*, 5(3):332–369, 1998.

[11] E. J. Candès and D. L. Donoho. Curvelets – a suprisingly effective non-adaptive representation for objects with edges. In A. Cohen, C. Rabut, and L. L. Schumaker, editors, *Curve and Surface Fitting*, Saint-Malo, 1999. Vanderbilt University Press.

[12] S. Chen and D. Donoho. Atomic decomposition by basis pursuit. In *SPIE International Conference on Wavelets*, San Diego, CA, July 1995.

[13] E.K.P. Chong and S.H. Zak. *Introduction to Optimization*. John Wiley & Sons, 1996.

[14] A. Cohen, I. Daubechies, O. Guleryuz, and Orchard M.T. On the importance of combining wavelet-based non-linear approximation with coding strategies. *IEEE Trans. on Information Theory*, 48(7):1895–1921, July 2002.

[15] A. Cohen, I. Daubechies, and P. Vial. Wavelets on the interval and fast wavelet transforms. *Applied and Computational Harmonic Analysis*, 1:54–81, December 1993.

[16] R. Coifman and M. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. Information Theory*, 38:713–718, March 1992.

[17] R. R. Coifman and D.L. Donoho. Translation invariant denoising. *Technical Report 475, Dept. of Statistics, Stanford University*, May 1995.

[18] P.L. Combettes. Generalized convex set theoretic image recovery. In *IEEE Int. Conf. Image Processing*, Lausanne, Switzerland, September 1996.

[19] T. Cover and J.A. Thomas. *Elements of Information Theory*. John Wiley and Sons, NY, 1991.

[20] M. Crouse, R.D. Nowak, and R.G. Baraniuk. Wavelet-based signal processing using hidden Markov models. *IEEE Trans. Signal Processing*, 2:886–902, April 1998.

[21] Z. Cvetković. *Overcomplete Expansions for Digital signal processing*. PhD thesis, University of California at Berkeley, 1995.

[22] Z Cvetković and M. Vetterli. Oversampled filter banks. *IEEE Trans. Sig. Proc.*, 46(5):1245–1255, May 1998.

[23] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Information Theory*, 36(5):961–1005, September 1990.

[24] I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[25] I. Daubechies, A. Grossman, and Y. Meyer. Painless nonorthogonal expansions. *Journal Math. Phys.*, 27:1271–1283, November 1986.

[26] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 4(3):247–269, 1998.

[27] G. Davis. *Adaptive nonlinear approximations.* PhD, dissertation, New York University, 1994.

[28] G.M. Davis, S. Mallat, and Z. Zhang. Adaptive time-frequency decompositions. *SPIE J. of Opt. Engin.*, 33(7):2183–2191, July 1994.

[29] S.N. Diggavi, N.J.A. Sloane, and V.A. Vaishampayan. Design of asymmetric multiple description lattice vector quantizers. In *Data Compression Conference*, Snowbird, Utah, March 2000.

[30] Minh Do. *Directional Multiresolution Image Representations.* PhD, dissertation, Swiss Federal Institute of Technology, November 2001.

[31] D.L. Donoho. Nonlinear solution of linear inverse problems by wavelet-vaguelette decomposition. *Applied and Computational Harmonic Analysis*, 2:101–126, April 1995.

[32] D.L. Donoho and I.M. Johnstone. Ideal spatial adaptation via wavelet shrinkage,. *Biometrika*, 81:425–455, December 1994.

[33] D.L. Donoho and P.B. Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49/3:906–931, June 1989.

[34] D.L. Donoho, M. Vetterli, R.A. DeVore, and I. Daubechies. Data compression and harmonic analysis. *IEEE Trans. on Information Theory*, 44(6):2435–2476, October 1998.

[35] X. Donoho, D.L. Huo. Uncertainty principle and ideal atomic decomposition. *IEEE Trans. Information Theory*, 47:2845–2862, November 2001.

[36] P.L. Dragotti, J. Kovačević, and V. Goyal. Quantized oversampled filter banks with erasures. In *Data Compression Conference*, Snowbird, Utah (USA), March 2001.

[37] P.L. Dragotti, S. Servetto, and M. Vetterli. Analysis of optimal filter banks for multiple description coding. In *Data Compression Conference*, Snowbird, Utah (USA), March 2000.

[38] P.L. Dragotti, S. Servetto, and M. Vetterli. Optimal filter banks for multiple description coding: Analysis and synthesis. *IEEE Trans. Information Theory*, 48(7):2036–2052, July 2002.

[39] P.L. Dragotti and M. Vetterli. Wavelet footprints: Theory, algorithms and applications. *IEEE Trans. on Signal Processing*. Submitted.

[40] P.L. Dragotti and M. Vetterli. Optimal biorthogonal filter banks for multiple description coding. In *3th IEEE/ITG Source and Channel Coding Conference*, Munich, Germany, January 2000.

[41] P.L. Dragotti and M. Vetterli. Shift-invariant gibbs free denoising algorithm based on wavelet transform footprints. In *SPIE International Symposium on Optical Science and Technology*, San Diego, California (USA), July 2000.

[42] P.L. Dragotti and M. Vetterli. Wavelet transform footprints: Catching singularities for compression and denoising. In *IEEE Int. Conf. on Image Processing*, Vancouver, Canada, September 2000.

[43] P.L. Dragotti and M. Vetterli. Deconvolution with wavelet footprints for ill-posed inverse problems. In *IEEE Int. Conf. on acoustic, speech and signal processing*, Orlando, Florida (USA), May 2002. To appear.

[44] R. J. Duffin and A. C. Schaeffer. A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc.*, 72:341–366, December 1952.

[45] A. A. El Gamal and T.M. Cover. Achievable rates for multiple descriptions. *IEEE Trans. Information Theory*, 28(6):851–857, November 1982.

[46] M. Elad and A. M. Bruckstein. On sparse signal representations. In *IEEE Int. Conf. Image Processing*, Thessaloniki, Greece, October 2001.

[47] W. H.R. Equitz and T.M. Cover. Successive refinement of information. *IEEE Trans. Information Theory*, 37:269–275, March 1991.

[48] M. Fleming and M. Effros. Generalized multiple description vector quantization. In *Data Compression Conference*, Snowbird, Utah, March 1999.

[49] J.H.. Friedman and W. Stuetzle. Projection pursuit regression,. *Journal of Amer. Stat. Assoc.*, 76:817–823, 1981.

[50] A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression*. Kluwer Acad. Pub., Boston, MA, 1992.

[51] M. M. Goodwin and M. Vetterli. Matching pursuit and atomic signal models based on recursive filter banks. *IEEE Trans. Signal Processing*, 47:1890–1902, July 1999.

[52] V. Goyal. Transform coding with integer-to-integer transforms. *IEEE Trans. Information Theory*, 46(2):465–473, March 2000.

[53] V. Goyal and J. Kovačević. Optimal multiple description transform coding of Gaussian vectors. In *Data Compression Conference*, Snowbird, Utah, March 1998.

[54] V. Goyal and J. Kovačević. Generalized multiple description coding with correlating transforms. *IEEE Trans. Information Theory*, 47:2199–2224, September 2001.

[55] V. Goyal, J. Kovačević, and J.A. Kelner. Quantized frame expansions with erasures. *Applied and Computational Harmonic Analysis*, 10(3):203–233, May 2001.

[56] V. Goyal, J. Kovačević, and M. Vetterli. Quantized frame expansions as source-channel codes for erasure channels. In *Data Compression Conference*, Snowbird, Utah, March 1999.

[57] V. Goyal, M. Vetterli, and N. T. Thao. Quantized overcomplete expansions in $\mathbb{R}^n$: Analysis, synthesis and algorithms. *IEEE Trans. Information Theory*, 44(1):16–31, January 1998.

[58] V.K. Goyal. Multiple description coding: Compression meets the network. *IEEE Sig. Processing Mag.*, 18:74–93, September 1995.

[59] V.K. Goyal. *Single and Multiple Description Transform Coding with Bases and Frames.* SIAM, Philadelphia, PA, 2001.

[60] R.M. Gray. Toeplitz and circulant matrices: A review. Available on line at http://ee-www.stanford.edu/ gray/.

[61] R.M. Gray. On the asymptotic eigenvalue distribution of Toeplitz matrices. *IEEE Trans. on Information Theory*, 18 (6):725–730, November 1972.

[62] R.M. Gray and A. Wyner. Source coding for a simple network. *Bell Syst. Tech. J.*, 53(9):1681–1721, 1974.

[63] U. Grenander and G Szego. *Toeplitz Forms and Their Applications.* University of California Press, Berkeley and Los Angeles, Calif, 1958.

[64] D. Han and D. R. Larson. *Frames, bases and and group representations.* Number 697 in memoirs of AMS. American mathematical society, Providence, RI, 2000.

[65] B. Hochwald, T. Marzetta, T. Richardson, W. Sweldens, and R. Urbanke. Systematic design of unitary space-time constellations. *IEEE Trans. Information Theory*, 46:1962–1973, September 2000.

[66] J. Hong. *Discrete Fourier, Hartley, and Cosine Transforms in Signal Processing.* PhD thesis, Columbia University, 1993.

[67] A. Ingle and V.A. Vaishampayan. DPCM system design for diversity systems with applications to packetized speech. *IEEE Trans. on Speech and Audio Processing*, 1:48–58, January 1995.

[68] S. Kalifa, J. Mallat and B. Rouge. Image deconvolution in mirror wavelet bases. In *IEEE Int. Conf. Image Processing*, Chicago, Illinois, October 1998.

[69] A.K. Katsaggelos. Iterative image restoration algorithm. *Opt. Engin.*, 28:735–748, 1989.

[70] J.A. Kelner, V. Goyal, and J. Kovačević. Multiple description lattice vector quantization: Variations and extensions. In *Data Compression Conference*, Snowbird, Utah, March 2000.

[71] V. Koshelev. Multilevel source coding and data-transmission theorem. In *VII all-Union conf. Theory of coding and data transmission*, 1978.

[72] V. Koshelev. Hierarchical coding of discrete sources. *Probability Peredachi Inf.*, 16(3):31–49, 1980.

[73] V. Koshelev. An evaluation of the average distortion for discrete scheme of sequential approximation. *Probability Peredachi Inf.*, 17(3):20–33, 1981.

[74] J. Kovačević, P.L. Dragotti, and V. Goyal. Filter bank frame expansions with erasures. *IEEE Trans. Information Theory*, 48(6):1439–1450, June 2002. Special issue in Honour of Aaron D. Wyner.

[75] E. LePennec and S. Mallat. Image compression with geometrical wavelets. In *IEEE Int. Conf. Image Processing*, Vancouver, Canada, September 2000.

[76] T. Linder, R. Zamir, and K. Zeger. The multiple description rate region for high resolution source coding. In *Data Compression Conference*, Snowbird, Utah, March 1998.

[77] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[78] S. Mallat and W.L. Hwang. Singularity detection and processing with waveletes. *IEEE Trans. Information Theory*, 38:617–643, March 1992.

[79] S. Mallat and Z. Zhang. Matching pursuit with time -frequency dictionaries. *IEEE Trans. Signal Processing*, 41:3397–3415, December 1993.

[80] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(7):710–732, July 1992.

[81] S. McCanne, V. Jacobson, and M. Vetterli. Receiver-driven layered multicast. In *ACM SIGCOMM*, Stanford, CA, August 1996.

[82] H. Neelamani, R. Choi and R. Baraniuk. Wavelet-based deconvolution for ill-conditioned systems. In *IEEE Int. Conf. Acoustic, Speech and Signal Proc.*, Phoenix, Arizona, March 1999.

[83] M. Orchard, Y. Wang, V.A. Vaishampayan, and A.R. Reibman. Redundancy rate-distortion analysis of multiple description coding using pairwise correlation transforms. In *IEEE int. Conf. Image Proc.*, Santa Barbara, CA, October 1997.

[84] L. Ozarow. On a source-coding problem with two channels and three receivers. *Bell Syst. Tech. J.*, 59(10):1909–1921, 1980.

[85] Y.C. Pati and P.S. Rezaifar, R. Krishnaprasad. Orthogonal matching pursuit: recursive functions approximation with applications to wavelet decomposition. In 27^{th} *Asilomar Conf. on Signals, Systems and Comput.*, November 1993.

[86] P. Prandoni and M. Vetterli. Approximation and compression of piecewise smooth functions. *Phil. Trans. Royal Society London*, August 1999.

[87] K. Ramchandram, A. Ortega, M. Uz, and M. Vetterli. Multiresolution broadcast for digital HDTV using joint source/channel coding. *IEEE Journal Select. Areas Commun.*, 11:6–23, January 1993.

[88] K. Ramchandram and M. Vetterli. Best wavelet packet bases in a rate-distortion sense. *IEEE Trans. Image Processing*, 2:160–175, April 1993.

[89] B. Rimoldi. Successive refinement of information: Characterization of the achievable rates. *IEEE Trans. Information Theory*, 40:253–259, January 1994.

[90] A. Said and W.A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans. Circuits and Systems for Video Technology*, 6(3):243–249, June 1996.

[91] S.D. Servetto, K. Ramchandran, V.A. Vaishampayan, and K. Nahrstedt. Multiple description wavelet based image coding. *IEEE Trans. Image Processing*, 9(5):813–826, May 2000.

[92] C.E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423, July 1948. Continued 27:623-656, October 1948.

[93] J.M. Shapiro. Embedded image coding using zerotrees of wavelets coefficients. *IEEE Trans. on Signal Processing*, 41:3445–3462, December 1993.

[94] E. Soljanin. Frames and quantum information theory. preprint 2000.

[95] H. Stark. *Image Recovery: Theory and Application*. Academic Press, 1995.

[96] N.T. Thao and M. Vetterli. Reduction of the MSE in R-times oversampled A/D conversions from $\mathcal{O}(1/R)$ to $\mathcal{O}(1/R^2)$. *IEEE Trans. Sig. Proc.*, 42(1):200–203, January 1994.

[97] M.K. Tsatsanis and G. B. Giannakis. Principal component filter banks for optimal multiresolution analysis. *IEEE Trans. Sig. Proc.*, 43(8):1766–1777, August 1995.

[98] M. Unser. An extension of the Karhunen-Loève transform for wavelets and perfect reconstruction filterbanks. In *SPIE Wavelet Appl. in Signal and Image Proc.*, San Diego, (CA), July 1993.

[99] M. Unser. On the optimality of ideal filters for pyramid and wavelet signal approximation. *IEEE Trans. Sig. Proc.*, 41:3591–3596, December 1993.

[100] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Trans. Image Proc.*, 4(11):1549–1560, November 1995.

[101] P.P. Vaidyanathan. *Multirate Systems and Filter Banks.* Prentice-Hall, Englewood Cliffs, NJ, 1993.

[102] V.A. Vaishampayan. Design of multiple description scalar quantizers. *IEEE Trans. Information Theory*, 39(3):821–834, May 1993.

[103] V.A. Vaishampayan and J.-C. Batllo. Asymptotic analysis of multiple description quantizers. *IEEE Trans. Information Theory*, 44(1):278–284, January 1998.

[104] V.A. Vaishampayan and J. Domaszewicz. Design of entropy-constrained multiple-description scalar quantizers. *IEEE Trans. Information Theory*, 40(1):245–250, January 1994.

[105] V.A. Vaishampayan, N.J.A. Sloane, and S.D. Servetto. Multiple description vector quantization with lattice codebooks: design and analysis. *IEEE Trans. Information Theory*, 47(5):1718–1734, July 2001.

[106] R. Venkataramani, G. Kramer, and V.K. Goyal. Bounds on the achievable region for certain multiple description problems. In *International Symposium on Information Theory (ISIT)*, Washington, DC, June 2001.

[107] M. Vetterli. Wavelets approximation and compression. *IEEE Signal Processing Magazine*, 18:59–73, September 2001.

[108] M. Vetterli and J. Kovačević. *Wavelets and Subband Coding.* Prentice-Hall, 1995.

[109] Y. Wang, M. Orchard, V.A. Vaishampayan, and A.R. Reibman. Multiple description coding using pairwise correlation transforms. *IEEE Trans. Image Processing*, 10(3):351–366, March 2001.

[110] Y. Wang, T. Orchard, and A.R. Reibman. Multiple description image coding for noisy channels by pairing transform coefficients. In *IEEE 1997 First workshop on Multimedia Signal Processing*, Princeton, NJ, June 1997.

[111] S.M. Yang and V.A. Vaishampayan. Low-delay communication for Rayleigh fading channel: An application of the multiple description quantizers. *IEEE Trans. Commun.*, 43:2771–2783, November 1995.

[112] R. Zamir. Gaussian codes and Shannon bounds for multiple descriptions. *IEEE Trans. Information Theory*, 45(7):2629–2635, November 1999.

[113] R. Zamir. Shannon-type bounds for multiple descriptions of a stationary source. In *Journal of Combinatorics, Information and System Sciences*, December 2000.

[114] A. Zandi, J.D. Allen, E.L. Schwartz, and M. Boliek. CREW: Compression with reversible embedded wavelets. In *Data Compression Conference*, Snowbird, Utah, March 1995.

[115] Z. Zhang and T. Berger. New results in binary multiple descriptions. *IEEE Trans. Information Theory*, 33(4):502–521, July 1987.

[116] Z. Zhang and T. Berger. Multiple description source coding with no excess marginal rate. *IEEE Trans. Information Theory*, 41(2):349–357, March 1995.

Curriculum Vitae

Pier Luigi Dragotti
Audio-Visual Communications Laboratory
Swiss Federal Institute of Technology Lausanne (EPFL)
1015 Lausanne, Switzerland
Email: pierluigi.dragotti@epfl.ch
Web: http://lcavwww.epfl.ch/~dragotti

Education

1998 - 2002 Ph.D. candidate in Department of Communication Systems, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.

1997 - 1998 Graduate School in Communication Systems, EPFL.

1990 - 1997 "Laurea" degree in Electrical Engineering (summa cum laude), University of Naples, Federico II, Italy.

Professional Experience

1998 - 2001 **Research and Teaching Assistant**, Audio-Visual Communications Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.

Summer 2000 **Summer Internship**, Mathematics of Communications Research Division, Bell Laboratories, Lucent Technologies, Murray Hill (NJ), USA.

Summer 1997 **Summer Internship** Communication Systems Department, University of Naples, Federico II, Naples, Italy.

March-Sept. 1996 **Visiting Student**, Compression and Classification Group, Information Systems Laboratory, Stanford University, Stanford (CA), USA.

Personal

Born in Naples, Italy. Citinzeship: Italian.

Publications

Journals and Manuscripts

1. P.L. Dragotti and M. Vetterli. Wavelet Footprints: Theory, Algorithms and Applications. *IEEE Transaction on Signal Processing*, submitted.

2. P.L. Dragotti, S. Servetto and M. Vetterli. Optimal Filter Banks for Multiple Description Coding: Analysis and Synthesis. *IEEE Transaction on Information Theory*, vol. 48, No. 7, pp. 2036-2052, July 2002.

3. J. Kovačević, P.L. Dragotti and V.K. Goyal. Filter Bank Frame Expansions with Erasures. *IEEE Transaction on Information Theory*, vol.48 No.6, pp. 1439-1450, June 2002. Special issue in Honour of Aaron D. Wyner.

4. P.L. Dragotti. G. Poggi and A.R.P. Ragozini. Compression of Multispectral Images by Three-Dimensional SPIHT. *IEEE Transaction on Geoscience and Remote Sensing*, vol.38, No.1, pp. 416-428, January 2000.

Conferences

1. V. Velisavljević, P.L. Dragotti and M. Vetterli. Directional Wavelet Transforms and Frames. Submitted to *IEEE International Conference on Image Processing*, Rochester (NY), USA, September 2002.

2. R. Shukla, M. N. Do, P.L. Dragotti and M. Vetterli. Rate-Distortion Optimal Tree Algorithms for Piecewise Polynomials. To be presented at *IEEE International Symposium on Information Theory*, Lausanne, Switzerland, July, 2002.

3. P.L. Dragotti and M. Vetterli. Deconvolution with Wavelet Footprints for Ill-Posed Inverse Problems. To be presented at *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, May, 2002.

4. P.L. Dragotti and M. Vetterli. Footprints and Edgeprints for Image Denoising and Compression. *IEEE International Conference on Image Processing*, Thessaloniki, Greece, October 2001.

5. M. N. Do, P. L. Dragotti, R. Shukla and M. Vetterli. On compression of two-dimensional piecewise smooth functions. Invited paper, *IEEE International Conference on Image Processing, Special Session on Image*

Processing and Non-Linear Approximation, Thessaloniki, Greece, October 2001.

6. P.L. Dragotti, J. Kovačević and V.K. Goyal. Quantized Oversampled Filter Banks with Erasures. In *Proc. of Data Compression Conference*, Snowbird (UT), March 2001. 2001.

7. P.L. Dragotti and M. Vetterli. Wavelet Transform Footprints: Catching Singularities for Compression and Denoising. In *Proc. of IEEE International Conference on Image Processing*, Vancouver, Canada, September 2000.

8. P.L. Dragotti and M. Vetterli. Shift-Invariant Gibbs Free Denoising Algorithm based on Wavelet Transform Footprints. In *Proc. of SPIE Conference on Wavelet Applications in Signal and Image Processing*, San Diego, USA, July 2000.

9. P.L. Dragotti, S. Servetto and M.Vetterli. Analysis of Optimal Filter Banks for Multiple Description Coding. In *Proc. of Data Compression Conference*, Snowbird (UT), March 2000.

10. P.L. Dragotti and M.Vetterli. Optimal Biorthogonal Filter Banks for Multiple Description Coding. In *Proc. of IEEE/ITG Conference on Source Channel Coding*, Munich, Germany, January 2000.

11. P.L. Dragotti and A.R.P. Ragozini. Fast Color Transformation by means of Hierarchical Table Lookups. In *Proc. European Conference on Signal Processing*, Rhodes, Greece, September 1998.