Wavelets and Applications

Pier Luigi Dragotti
Communications and Signal Processing Research Group
Department of Electrical and Electronic Engineering
Imperial College London

September 20, 2018

Contents

Chapter 1

Mathematical Background

1.1 Hilbert Spaces

Finite-dimensional vector spaces are usually studied in linear algebra [14]. Such spaces involve vectors over $\mathbb{R}$ or $\mathbb{C}$ with finite dimension N and are denoted by $\mathbb{R}^N$ or $\mathbb{C}^N$. In many cases, however, one needs to generalise the notion of a vector space to infinite dimensions. Hilbert spaces are one of such examples. The good news is that the finite dimensional spaces $\mathbb{R}^N$ and $\mathbb{C}^N$ are particular examples of Hilbert spaces. Therefore, there is a very close link between the notion of bases, approximation, norms in infinite-dimensional and finite-dimensional spaces. This allows us to use the latter space to gain some understanding of infinite-dimensional spaces.

More formally [14, 37]:

Definition 1 (Vector Space) *By a vector space we mean a nonempty set E with two operations:*

- *a mapping $(x, y) \to x + y$ from $E \times E$ into E called addition,*
- *a mapping $(\lambda, x) \to \lambda x$ from $\mathbb{R} \times E$ into E.*

such that the following conditions are satisfied:

1. *Commutativity: $x + y = y + x$;*
2. *Associativity $(x + y) + z = x + (y + z)$;*
3. *Distributivity $(\alpha + \beta)x = \alpha x + \beta x$ and $\alpha(x + y) = \alpha x + \alpha y$;*
4. *For every $x, y \in E$ there exists $z \in E$ such that $x + z = y$;*
5. *$\alpha(\beta x) = (\alpha\beta)x$;*
6. *$1x = x$.*

Definition 2 (Inner Product Space) *Let E be a complex vector space. A mapping $\langle \cdot, \cdot \rangle$ is called an inner product in E if for any $x, y, z \in E$ and $\alpha, \beta \in \mathbb{C}$ the following conditions are satisfied:*

1. *$\langle x, y \rangle = \overline{\langle y, x \rangle}$;*
2. *$\langle\ \alpha x + \beta y, z \rangle = \alpha\langle x, z \rangle + \beta\langle y, z \rangle$;*
3. *$\langle x, x \rangle \geq 0$ and $\langle x, x \rangle = 0$ implies $x = 0$.*

A vector space E with an inner product is called Inner Product Space.

Given the definition of inner product, we define the norm of a vector x as:

$$\|x\| = \sqrt{\langle x, x \rangle},$$

which finally leads to the definition of Hilbert Spaces:

Definition 3 (Hilbert Space) *A complete inner product space is called Hilbert Space.*

By completeness of the inner product space E, we mean that every Cauchy sequence in E converges to an element of E. A Cauchy sequence is defined as a sequence of vectors $x_n \in E$ such that for every $\epsilon > 0$ there exists a number M such that $\|x_m - x_n\| < \epsilon$ for all $m, n > M$.

The relationship between the three different spaces is highlighted in Fig. 1.1.

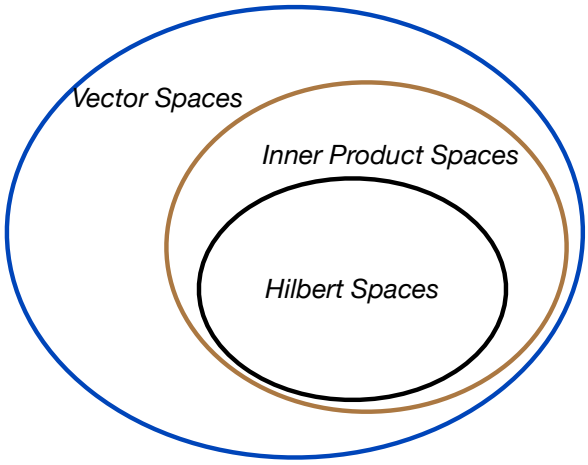


Figure 1.1: Relationship between different types of vector spaces.

Before presenting some examples of Hilbert spaces, we also introduce the notions of subspace, span, linear independence and dimension which will be used throughout the chapters.

Definition 4 (Subspace) *A subset S of a vector space E is a subspace when it is closed under the operations of vector addition and scalar multiplication:*

1. *For all x and y in S, $x + y$ is in S.*
2. *For all x in S and α in $\mathbb{C}$, αx is in S.*

Definition 5 (Span) *The span of a set of vectors S is the set of all finite linear combinations of vectors in S:*

$$span(S) = \left\{ \sum_{k=0}^{N-1} \alpha_k \varphi_k \mid \alpha_k \in \mathbb{C}, \varphi_k \in S \text{ and } N \in \mathbb{N}. \right\}$$

Note that a span is always a subspace.

Definition 6 (Linear Independence) *The set of vectors $\{\varphi_0, \varphi_1, ..., \varphi_{N-1}\}$ is called linearly independent when $\sum_{k=0}^{N-1} \alpha_k \varphi_k = 0$ is true only if $\alpha_k = 0$ for all k. Otherwise, the set is linearly dependent.*

Definition 7 (Dimension) *A vector space E is said to have dimension N when it contains a linearly independent set with N elements and every set with $N+1$ or more elements is linearly dependent. If no such finite N exists, the vector space is infinite dimensional.*

1.1.1 Examples of Hilbert Spaces

Example 1 (Square-integrable functions) The space of all complex-valued functions $f(t)$, $t \in \mathbb{R}$ such that

$$\int_{-\infty}^{\infty} |f(t)|^2 dt < \infty$$

and it is denoted by $L_2(\mathbb{R})$. The inner product in $L_2(\mathbb{R})$ is given by

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(t) g^*(t) dt,$$

where $g^*(t)$ is the complex conjugate of $g(t)$ and the norm is

$$\|f(t)\| = \sqrt{\langle f, f \rangle} = \sqrt{\int_{-\infty}^{\infty} |f(t)|^2 dt}.$$

Example 2 (Square-summable sequences) The space of all complex-valued sequences $x[n]$, $n \in \mathbb{Z}$ such that

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty$$

and it is denoted by $l_2(\mathbb{Z})$. The inner product in $l_2(\mathbb{Z})$ is given by

$$\langle x, y \rangle = \sum_{n=-\infty}^{\infty} x^*[n] y[n]$$

and the norm is

$$\|x\| = \sqrt{\langle x, x \rangle} = \sqrt{\sum_{n=-\infty}^{\infty} |x[n]|^2}.$$

Example 3 N-dimensional vector spaces $\mathbb{R}^N$ or $\mathbb{C}^N$.

1.1.2 Bases and Frames

Consider now a set of elements $\{\varphi_i(t)\}_{i \in \mathbb{Z}}$ that form a basis of $L_2(\mathbb{R})$. That is, the elements $\varphi_i(t)$ are linearly independent and any signal in $L_2(\mathbb{R})$ can be expressed as a linear combination of the φ_is or

$$f(t) = \sum_i \alpha_i \varphi_i(t). \tag{1.1}$$

This set may form an orthogonal or biorthogonal basis. What is the difference between these two types of bases?

We can resort to finite-dimensional vector spaces to understand this difference.[1] In Figure 1.2, we show an example of an orthogonal and of a biorthogonal basis of $\mathbb{R}^2$. The set $\{\varphi_i(t)\}$ is orthogonal (or

[1] For simplicity we focus on real functions only.

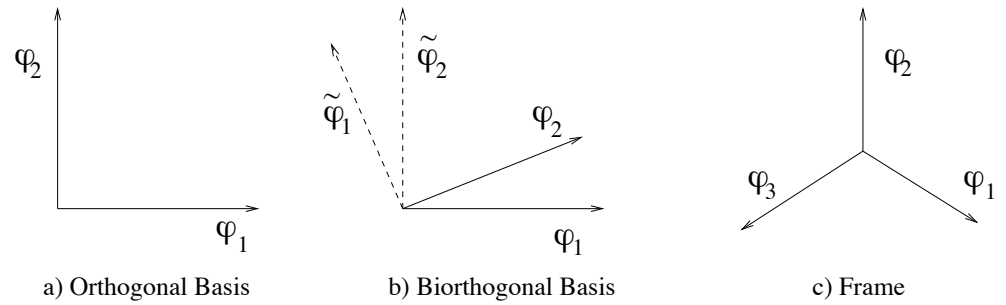


Figure 1.2: Example of different expansions. a) Orthogonal basis of $\mathbb{R}^2$, b) Biorthogonal basis of $\mathbb{R}^2$, c) Frames in R^2.

more precisely orthonormal) if $\langle \varphi_i(t), \varphi_j(t) \rangle = \delta_{ij}$. In the case of orthonormal bases the coefficients α_i of the expansion (1.1) are given by $\alpha_i = \langle f(t), \varphi_i(t) \rangle$ leading to the following expansion formula

$$f(t) = \sum_i \langle f(t), \varphi_i(t) \rangle \varphi_i(t).$$

An important element of orthonormal expansions is that they preserve the norm or

$$\|f\|^2 = \sum_i |\langle f, \varphi_i \rangle|^2$$

which is known as Parseval's equality.

In the case of biorthogonal bases, the decomposition of $f(t)$ is a bit more involved and one has to design the dual basis first. Assume that $\{\varphi_i(t)\}_{i \in \mathbb{Z}}$ is biorthogonal, the dual basis is given by the set of elements $\{\tilde{\varphi}_i(t)\}_{i \in \mathbb{Z}}$ satisfying

$$\langle \varphi_i(t), \tilde{\varphi}_j(t) \rangle = \delta_{ij}.$$

For instance, in Figure 1.2, the dual of φ_1 is the vector $\tilde{\varphi}_1$ that is orthogonal to φ_2 and the dual of φ_2 is the vector $\tilde{\varphi}_2$ orthogonal to φ_1.

Then the signal expansion formula becomes

$$f(t) = \sum_i \langle f, \tilde{\varphi}_i \rangle \varphi_i(t) = \sum_i \langle f, \varphi_i \rangle \tilde{\varphi}_i(t).$$

Notice that in this case the Parseval's identity does not hold anymore, that is, biorthogonal bases do **not** preserve the norm. Instead we have that there exist strictly positive constants $A, B, \tilde{A}, \tilde{B}$ such that

$$\begin{aligned} A\|f\|^2 \leq \textstyle\sum_i |\langle f, \varphi_i \rangle|^2 \leq B\|f\|^2, \\ \tilde{A}\|f\|^2 \leq \textstyle\sum_i |\langle f, \tilde{\varphi}_i \rangle|^2 \leq \tilde{B}\|f\|^2. \end{aligned} \tag{1.2}$$

The above analysis has a clear equivalent in linear algebra (this is not astonishing since $\mathbb{R}^N$ and $\mathbb{C}^N$ are Hilbert spaces). Consider an N-dimensional vector x and an $N \times N$ matrix M with linearly independent rows. That is, the rows of M form a basis of $\mathbb{R}^N$, call these elements $\tilde{\varphi}_i$ $i = 1, 2, ..., N$. Now, x can be decomposed as $y = Mx$ leading to the coefficients $y_i = \langle x, \tilde{\varphi}_i \rangle$. We can then reconstruct x by multiplying y with the inverse of M. Namely $x = M^{-1}y$. The columns of the synthesis matrix M^{-1} are linearly independent and form the dual basis of $\tilde{\varphi}_i$ $i = 1, 2, ..., N$. Call these vectors φ_i

$i = 1, 2, ..., N$. If the rows of M are orthonormal then $M^{-1} = M^T$. That is, the inverse of M is equal to the transpose of M. This means that the dual basis coincides with the original basis leading to the expansion formula for the orthogonal case. Moreover, since the eigenvalues of $M^T M$ are all ones, it follows that $\|y\| = \|x\|$.

If the rows of M are not orthonormal then $M^{-1} \neq M^T$. This means that the elements of the dual basis do not coincide with the elements of the original one. The condition $M^{-1}M = I$, where I is the identity matrix, ensures only that $\langle \varphi_i(t), \tilde{\varphi}_j(t) \rangle = \delta_{ij}$. We are indeed in the biorthogonal case. In this case, moreover, one can easily verify that in general $\|y\| \neq \|x\|$. In fact, the eigenvalues of $M^T M$ are not all equal to one as in the orthogonal case. Call λ_{min} and λ_{max} the minimum and maximum eigenvalue of $M^T M$, we then have that $\lambda_{min}\|x\|^2 \leq \|y\|^2 \leq \lambda_{max}\|x\|^2$, which is the finite-dimensional equivalent to (1.2).

So far, we have considered signal expansions onto bases. However, one can also expand signals using overcomplete sets of vectors. This overcomplete sets of vectors are called frames. For a more detailed treatment of frames we refer the reader to [11, 12, 15].

Exercise 1 *The notion of orthogonal and biorthogonal bases and the notion of frames are quite pervasive in applied mathematics and very important in this course. We thus recommend that you run some simple matlab simulations to digest these new notions better.*

Start by designing an orthogonal and a biorthogonal basis of $\mathbb{R}^2$. *For example, you may have the canonical basis for the orthogonal case leading to* $M = I$ *and* $M = [1, 0; \sqrt{2}/2, \sqrt{2}/2]$ *for the biorthogonal case. Verify the reconstruction formulas and the preservation or non presevation of the norms for the two cases. Make sure you understand what is happening.*

Try different bases. You may soon realize that there is no much freedom in the design of orthogonal bases. In fact, any other possible orthogonal basis is obtained by rotating the canonical one. However, you have more freedom with the biorthogonal case. For example, consider $M = [1, 0; \cos(\theta), \sin(\theta)]$ *with* $0 < \theta < \pi/2$. *Check how the eigenvalues of* $M^T M$ *change with* θ, *in particular, when* θ *tends to zero. See what happens to the reconstruction formula. Provide a geometrical interpretation of your findings.*

Finally, design a frame of $\mathbb{R}^2$. *For example, consider* $F = [1, 0; 0, 1; \sqrt{2}/2, \sqrt{2}/2]$. *Try to compute the inverse (known as pseudo-inverse) and see whether you understand the reconstruction formula.*

We conclude this section by introducing a class of subspaces of $L_2(\mathbb{R})$ called shift-invariant subspaces which will be frequently used throughout the course.

Definition 8 (Shift-Invariant Subspaces) *A subspace* $V \in L_2(\mathbb{R})$ *is a shift-invariant subspace with respect to shift* $T \in \mathbb{R}^+$ *when* $x(t) \in V$ *implies* $x(t - kT) \in V$ *for every integer* k. *In addition,* $\varphi(t) \in V$ *is called a generator of* V *when*

$$V = span(\{\varphi(t - nT)\}_{n \in \mathbb{Z}}).$$

1.1.3 Approximations and Projections

Often, a vector x from a Hilbert space H has to be approximated by a vector lying in a subspace V. Call this second vector $\hat{x}$. The best approximation in the least-squares sense is given by the orthogonal projection of x onto V. Namely $\min_{\hat{x} \in V} \|x - \hat{x}\|^2$ is achieved when $\hat{x}$ is the orthogonal projection of x in V. Call $V^\perp$ the orthogonal complement of V we have that [14]:

Definition 9 (Orthogonal Projection Operator) *Let* S *be a closed subspace of a Hilbert space* $\mathbb{H}$. *The operator* P *on* $\mathbb{H}$ *defined by*

$$Px = \hat{x} \qquad \text{if } x = \hat{x} + z, \quad \hat{x} \in V, \qquad \text{and } z \in V^\perp,$$

is called the orthogonal projection operator onto V. The vector $\hat{x}$ is called the projection of x onto V.

Theorem 1 (Projection Theorem) *Let V be a closed subspace of Hilbert space H and let x be a vector in H.*

1. *Existence: There exists $\hat{x} \in V$ such that $\|x - \hat{x}\| \leq \|x - v\|$ for all $v \in V$.*
2. *Orthogonality: $x - \hat{x} \perp V$ is necessary and sufficient for determining $\hat{x}$.*
3. *Uniqueness: The vector $\hat{x}$ is unique.*
4. *Linearity: $\hat{x} = Px$ where P is a linear operator that depends on V and not on x.*
5. *Idempotency: $P(Px) = Px$ for all $x \in H$.*
6. *Self-adjointness: $P = P^*$.*

We omit the proof of this theorem. However, recall that an operator P is self-adjoint if, for any pair of vectors $x_1, x_2 \in \mathbb{H}$, it satisfies

$$\langle Px_1, x_2 \rangle = \langle x_1, Px_2 \rangle.$$

Since the projection operator is linear, it can be expressed using matrices and therefore we can easily verify whether a matrix is related to an orthogonal projector or not. If we denote with P one such matrix, this is related to an orthogonal projector only when $P^2 = P$ and $P^* = P$ where * denotes in this case the conjugate transpose of the matrix.

Finally, based on the above analysis, we might be willing to separate linear idempotent operators which are self-adjont from those which are not, leading in this way to the notion of *oblique* projector:

Definition 10 (Orthogonal Projection Operator, Oblique Projection Operator) *Consider the idempotent operator P.*

1. *A projection operator is a bounded linear operator that is idempotent.*
2. *An orthogonal projection operator is a projection operator that is self-adjoint.*
3. *An oblique projection operator is a projection operator that is not self-adjoint.*

1.2 Fourier Theory

The signals to be expanded can be continuous or discrete in time. The same is true for the expanded or transformed signal. Thus, there are four possible combinations to be considered:

1. **Continuous-time Fourier transform** (from $L_2(\mathbb{R})$ to $L_2(\mathbb{R})$).

   Given a function $f(t) \in L_2(\mathbb{R})$ the Fourier transform is defined by

   $$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt.$$

   The inverse Fourier transform is given by

   $$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{j\omega t} d\omega.$$

The Fourier transform satisfies several properties. We just review here the moment property and the Poisson summation formula which will be used later on.

Call m_n the nth moment of $f(t)$:

$$m_n = \int_{-\infty}^{\infty} t^n f(t) dt, \quad n = 0, 1, 2, ...$$

the moment theorem states that

$$(-j)^n m_n = \left. \frac{\partial^n \hat{f}(\omega)}{\partial \omega^n} \right|_{\omega=0}.$$

Assume a function $f(t)$ with sufficient smoothness and decay, the Poisson summation formula states that

$$\sum_{n=-\infty}^{\infty} f(t - nT) = \frac{1}{T} \sum_{k=-\infty}^{\infty} \hat{f}\left(\frac{2\pi k}{T}\right) e^{j2\pi kt/T}.$$

In particular, for $T = 1$ and $t = 0$, we have that

$$\sum_{n=-\infty}^{\infty} f(n) = \sum_{k=-\infty}^{\infty} \hat{f}(2\pi k).$$

2. **Continuous-time Fourier series** (from $L_2([0, T])$ to $l_2(\mathbb{Z})$)

   Consider a periodic function $f(t)$ of period T or a function $f(t)$ defined in the interval $[0, T]$ only. This function can be expressed as a linear combination of complex exponentials at frequencies $n\omega_0 = 2\pi n/T$ or

   $$f(t) = \sum_k F[k] e^{jk\omega_0 t}$$

   with

   $$F[k] = \frac{1}{T} \int_T f(t) e^{-jk\omega_0 t} dt.$$

3. **Discrete-time Fourier transform** (from $l_2(\mathbb{Z})$ to $L_2([0, 2\pi])$)

   In the previous two cases we have considered continuous-time functions. Consider, now, a discrete-time signal $f[n]$ $n \in \mathbb{Z}$ and $f[n] \in l_2(\mathbb{Z})$. The discrete-time Fourier transform is defined as

   $$F(e^{j\omega}) = \sum_{n=-\infty}^{\infty} f[n] e^{-j\omega n}$$

   and its inverse is given by

   $$f[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{j\omega}) e^{j\omega n} d\omega.$$

   Notice that $F(e^{j\omega})$ is periodic of period 2π. Thus, the discrete-time Fourier transform is to some extent the dual of the Fourier series. The property of this transform will be discussed in the next section, when the z-transform is introduced.

4. **Discrete-time Fourier series (DFT)** (from $\mathbb{C}^N$ to $\mathbb{C}^N$)

   The DFT transform is of great importance in practice since it can be implemented with fast algorithms (the FFT algorithm). Given a finite-length sequence $f[n]$ $n = 0, 1, ..., N-1$, the DFT is defined as

   $$F[k] = \sum_{n=0}^{N-1} f[n] W_N^{nk}$$

   and its inverse is given by

   $$f[n] = \frac{1}{N} \sum_{k=0}^{N-1} F[k] W_N^{-nk}$$

   where $W_N^{kn} = e^{-j2\pi kn/N}$.

1.2.1 z-transform

Given a discrete-time signal $f[n]$ the z-transform is defined as

$$F(z) = \sum_{n \in \mathbb{Z}} f[n] z^{-n} \quad z \in \text{ ROC} .$$

Notice that the z-transform is equal to the discrete-time Fourier transform for $z = e^{j\omega}$. Some useful properties of the z-transform are listed below without proof.

- Delay
$$f[n - n_0] \longleftrightarrow z^{-n_0} F(z)$$

- Time-reversal
$$f^T[n] = f[-n] \longleftrightarrow F(z^{-1})$$

- Modulation
$$(-1)^n f[n] \longleftrightarrow F(-z)$$

- Up-sampling
$$[f]_{\uparrow 2}[n] = \begin{cases} 0 & n \text{ odd} \\ f[l] & n = 2l \text{ even} \end{cases} \longleftrightarrow F(z^2)$$

- Down-sampling
$$[f]_{\downarrow 2}[n] = f[2n] \longleftrightarrow Y(z) = \frac{1}{2} \left(F(z^{1/2}) + F(-z^{1/2}) \right)$$

1.3 Multirate Discrete-Time Signal Processing

Multirate signal processing deals with the problem of changing rates of discrete-time signals. In this section we review some basic operations performed in multirate signal processing such as upsampling and downsampling, and how these operations can be combined with filtering.

1.3.1 Upsampling by 2

Up-sampling a sequence $x[n]$ by 2 results in a new sequence $y[n]$ given by

$$y[n] = \begin{cases} 0 & n \text{ odd} \\ x[l] & n = 2l \text{ even} \end{cases}$$

In matrix notation upsampling is obtained by multiplying x by the 'upsampling matrix' U_2. This is the identity matrix where rows of zeros are added in the odd numbered locations:

$$y = U_2 x = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots \\ \dots & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & \dots \\ \dots & 0 & 1 & 0 & \dots \\ \dots & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & \dots \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} \cdots \\ x[0] \\ x[1] \\ x[2] \\ \cdots \end{bmatrix}$$

In the frequency domain we have that

$$Y(z) = X(z^2)$$

or

$$Y(e^{j\omega}) = X(e^{j2\omega})$$

Proof

$$Y(z) = \sum_n y[n] z^{-n} = \sum_k y[2k] z^{-2k} = \sum_k x[k] (z^2)^{-k} = X(z^2).$$

One can easily extend the above result to the case of upsampling by an integer N.

1.3.2 Sub-sampling by 2

Down-sampling or sub-sampling a sequence $x[n]$ by 2 results in a new sequence $y[n]$ given by

$$y[n] = x[2n].$$

This can be performed using the sub-sampling matrix $D_2 = U_2^T$. That is, D_2 is the identity matrix with odd-numbered rows removed:

$$y = D_2 x = \begin{bmatrix} \dots & . & . & . & . & . & . & \dots \\ \dots & 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & 0 & \dots \\ \dots & . & . & . & . & . & . & \dots \end{bmatrix} \begin{bmatrix} \cdots \\ x[0] \\ x[1] \\ x[2] \\ \cdots \end{bmatrix}.$$

In the frequency domain we have that

$$Y(z) = \frac{1}{2}[X(z^{1/2}) + X(-z^{1/2})]$$

or

$$Y(e^{j\omega}) = \frac{1}{2}[X(e^{j\omega/2}) + X(e^{j(\omega-2\pi)/2})]$$

Proof

$$\begin{aligned} Y(z) &= \textstyle\sum_k x[2k]z^{-k} \\ &= \textstyle\sum_k x[k]z^{-k/2}\frac{(1+(-1)^k)}{2} \\ &= \tfrac{1}{2}\textstyle\sum_k x[k]z^{-k/2} + \tfrac{1}{2}\sum_k (-1)^k x[k]z^{-k/2} \\ &= \tfrac{1}{2}[X(z^{1/2}) + X(-z^{1/2})]. \end{aligned}$$

Notice that $D_2U_2 = I$ while $U_2D_2 \neq I$. That is, there is no loss of information if we first perform up-sampling and then sub-sampling.

Finally, subsampling by an integer N leads to the sequence $y[n] = x[Nn]$. In the frequency domain we have that

$$Y(e^{j\omega}) = \frac{1}{N}\sum_{k=0}^{N-1} X(e^{j(\omega-k2\pi)/N}) = \frac{1}{N}\sum_{k=0}^{N-1} X(W_N^k z^{1/N}).$$

1.3.3 Filters Interchanged with Samplers

The sequence $x[n]$ may be filtered before sub-sampling. Call $h[n]$ the impulse response of the filter. The filter is linear and time-invariant. Expressed as an infinite matrix, H has the values $h[n]$ along its nth diagonal. That is,

$$\begin{bmatrix} \dots & . & . & . & . & . & . & \dots \\ \dots & h[2] & h[1] & h[0] & 0 & 0 & 0 & \dots \\ \dots & h[3] & h[2] & h[1] & h[0] & 0 & 0 & \dots \\ \dots & h[4] & h[3] & h[2] & h[1] & h[0] & 0 & \dots \\ \dots & . & . & . & . & . & . & \dots \end{bmatrix}.$$

Therefore, filtering followed by sub-sampling can be expressed in matrix notation as $y = D_2Hx$ or

$$y = \begin{bmatrix} \dots & . & . & . & . & . & . & . & \dots \\ \dots & h[2] & h[1] & h[0] & 0 & 0 & 0 & 0 & \dots \\ \dots & h[4] & h[3] & h[2] & h[1] & h[0] & 0 & 0 & \dots \\ \dots & h[6] & h[5] & h[4] & h[3] & h[2] & h[1] & h[0] & \dots \\ \dots & . & . & . & . & . & . & . & \dots \end{bmatrix} x.$$

That is,

$$y[n] = \sum_k h[2n-k]x[k] = \langle h[2n-k], x[k]\rangle_k = \langle h^T[k-2n], x[k]\rangle. \tag{1.3}$$

Normally we cannot reverse the ordering of the filtering and sub-sampling. However, sub-sampling by N followed by filtering with a filter $H(z)$ is equivalent to filtering with the upsampled filter $H(z^N)$ before sub-sampling.

Indeed, sub-sampling by N the signal $X(z)H(z^N)$ results in

$$\frac{1}{N}\sum_{k=0}^{N-1} X(W_N^k z^{1/N})H((W_N^k z^{1/N})^N) = H(z)\frac{1}{N}\sum_{k=0}^{N-1} X(W_N^k z^{1/N})$$

which is equivalent to filtering the sub-sampled version of $x[n]$ with $h[n]$. This is known as the first Noble identity. This identity is graphically illustrated in Figure 1.3.

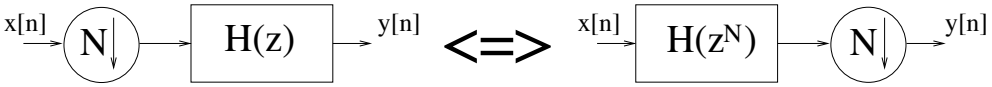


Figure 1.3: The first Noble identity.

Up-sampling is usually followed by filtering. Call $g[n]$ the linear time-invariant filter, it follows that $y = GU_2x$ or

$$y = \begin{bmatrix} \dots & . & . & . & \dots \\ \dots & g[0] & 0 & 0 & \dots \\ \dots & g[1] & 0 & 0 & \dots \\ \dots & g[2] & g[0] & 0 & \dots \\ \dots & g[3] & g[1] & 0 & \dots \\ \dots & g[4] & g[2] & g[0] & \dots \\ \dots & . & . & . & \dots \end{bmatrix} x.$$

That is,

$$y[n] = \sum_k x[k]g[n-2k] = \langle g[n-2k], x[k] \rangle. \tag{1.4}$$

The second Noble identity concerns upsampling and it shows that filtering with a filter $G(z)$ followed by upsampling by N is equivalent to upsampling followed by filtering with a filter $G(z^N)$. The proof of this identity is straight-forward and is omitted. The second Noble identity is shown in Figure 1.4. We are now in a position to combine the two operations and to try to analyze the effect.

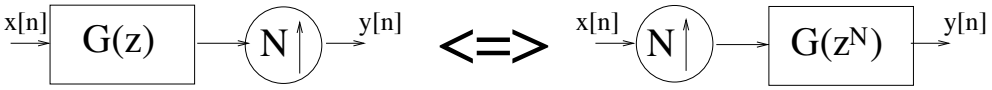


Figure 1.4: The second Noble identity.

Consider the discrete-time signals $x[n]$ and assume that the signal is filtered with a linear filter $h[n]$ and then down-sampled by two. The resulting signal is up-sampled and filtered with a synthesis filter $g[n]$. Call the output $y[n]$. Now, assume that $g[n]$ has been designed so that $\langle g[n], g[n-2k] \rangle = \delta_k$. That is, $g[n]$ is orthogonal to its shifted versions with shift equal to a multiple of two. Moreover assume that $h[n] = g[-n]$, that is, $h[n]$ is the time-reversed version of $g[n]$. Then the so designed operator P that maps $x[n]$ into $y[n]$ is an orthogonal projection from $l_2(\mathbb{Z})$ onto the sub-space spanned by $g[n]$ and its shifted versions (shifted by a multiple of two).

Proof

We need to show that P is idempotent (i.e., $P^2 = P$) and that it is self-adjoint (i.e., $P^T = P$).

Using matrix notation $P = GU_2D_2H$, but $H = G^T$ and $D_2G^TGU_2 = I$. Thus,

$$P^2 = (GU_2D_2H)(GU_2D_2H) = GU_2ID_2H = P.$$

Moreover

$$P^T = (GU_2D_2H)^T = (GU_2D_2G^T)^T = GD_2^TU_2^TG^T = GU_2D_2G^T = P.$$

which concludes the proof.

Finally, notice that the following holds

$$\langle g[n], g[n-2k] \rangle = \delta_k \longleftrightarrow G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2. \tag{1.5}$$

Call $p[k] = \langle g[n], g[n-k] \rangle = \sum_n g[n]g[n-k]$ the deterministic autocorrelation of $g[n]$. It clearly follows that $P(z) = G(z)G(z^{-1})$. The left size of equation (1.5) is simply the autocorrelation $p[k]$ sub-sampled by two. Thus if we call $\hat{p}[k] = p[2k]$ we have that

$$\hat{P}(z) = \frac{1}{2}(P(z^{1/2}) + P(-z^{1/2})).$$

By replacing $z^{1/2}$ with z and $P(z)$ with $G(z)G(z^{-1})$ we obtain the right side of (1.5). This equation will be used very often in future.

1.3.4 Polyphase Transforms

It is sometimes useful to decompose a signal $x[n]$ into its polyphase components. This is usually necessary when dealing with periodically shift-invariant systems such as up-samplers or down-samplers. The size N polyphase transform of a sequence $x[n]$ is defined as the vector of sequences $(x_0[n], x_1[n], ..., x_{N-1}[n])^T$ where

$$x_i[n] = x[nN + i].$$

Thus $X(z)$ can be written as the sum of shifted and upsampled polyphase components or

$$X(z) = \sum_{i=0}^{N-1} z^{-i} X_i(z^N)$$

where

$$X_i(z) = \sum_{i=-\infty}^{\infty} x[nN + i]z^{-n}.$$

The polyphase decomposition of a filter is instead defined as

$$H(z) = \sum_{i=0}^{N-1} z^{i} H_i(z^N),$$

with

$$H_i(z) = \sum_{n=-\infty}^{\infty} h[Nn - i]z^{-n}, \quad i = 0, 1, ..., N-1.$$

Thanks to the polyphase decomposition, the product $H(z)X(z)$ followed by sub-sampling by N can be written as

$$Y(z) = \sum_{i=0}^{N-1} H_i(z)X_i(z).$$

1.4 Exercises

1.1 Consider the system in Figure 1. Give the z-transform and Fourier transform of the signal at locations 1-4. Make the corresponding graphs of the Fourier transform assuming that $H(z)$ is an ideal half-band lowpass filter and that $X(z)$ has the following spectrum:

1.2 Consider the system shown in Figure 1.7

(a) What does the product filter $P(z) = H(z)G(z)$ have to satisfy in order to have perfect reconstruction such that $\hat{x}[n] = x[n]$?

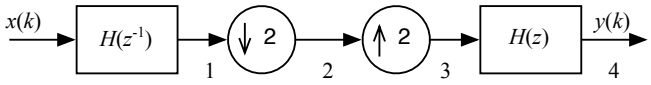


Figure 1.5: Multi-rate system.

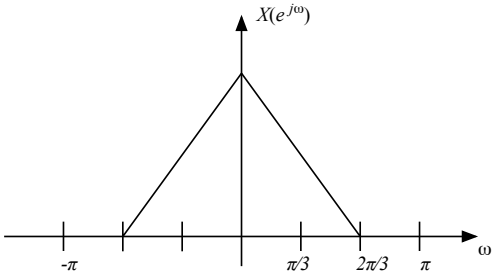


Figure 1.6: Spectrum of $x[k]$.

(b) Assume that $H(z) = (z^{-2} + z^{-1} + 1 + z + z^2)$. Find the shortest symmetric filter $G(z)$ such that perfect reconstruction is achieved.

(c) Now assume that $H(z) = (1 + z + z^2 + z^3)$. Design $G(z)$ so that the output $\hat{X}(z) = 0$.

1.3 Consider the multi-rate system shown in Figure 1.8, where $G_0(z)$ is an ideal low-pass filter with cutoff frequency $\pi/2$ and $G_1(z)$ is an ideal high-pass filter with cutoff frequency $\pi/2$. The two filters are shown in Figure 1.9.

Sketch and dimension the four spectra $Y_1(e^{j\omega})$, $Y_2(e^{j\omega})$, $Y_3(e^{j\omega})$, $Y_4(e^{j\omega})$ assuming that $x[n]$ has the spectrum shown in Figure 1.10.

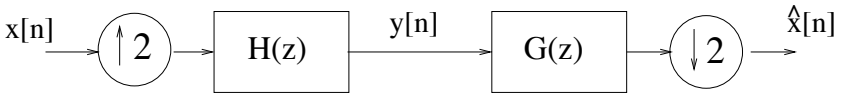


Figure 1.7: An interpolator.

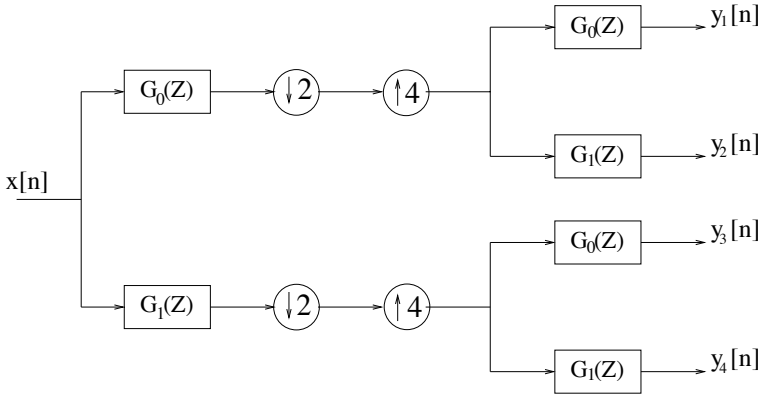


Figure 1.8: The multi-rate system.

1.4 Technically, one cannot talk of transfer function in the case of multi-rate systems since changes in the sampling rates are not time invariant. However, there are cases where by carefully designing the processing chain, the input/output relationship can indeed by modeled with an equivalent transfer function.

(a) Find the equivalent transfer function $H(z) = Y(z)/X(z)$ of the following system:

(b) Consider the system described by the block diagram of Figure 1.11 where $H(z)$ is an ideal low-pass filter with cutoff frequency $\pi/4$. Compute the transfer function of the system for $M = 2$ and for $M = 4$.

(c) Consider the system in Figure 1.12 where $H(z)$ is an ideal low-pass filter with cutoff frequency π/M. Show that this system implements a fractional delay (i.e., show that the equivalent transfer function of the system is that of a pure delay, where the delay is not necessarily an integer).

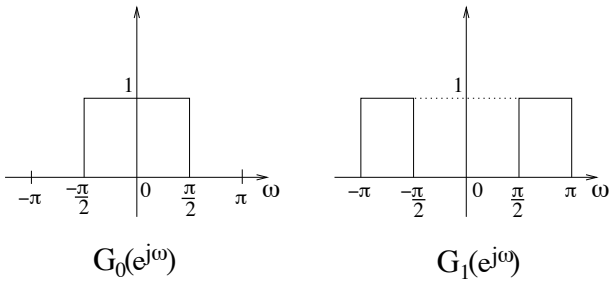


Figure 1.9: The ideal low-pass and high-pass filters $G_0(z)$, $G_1(z)$.

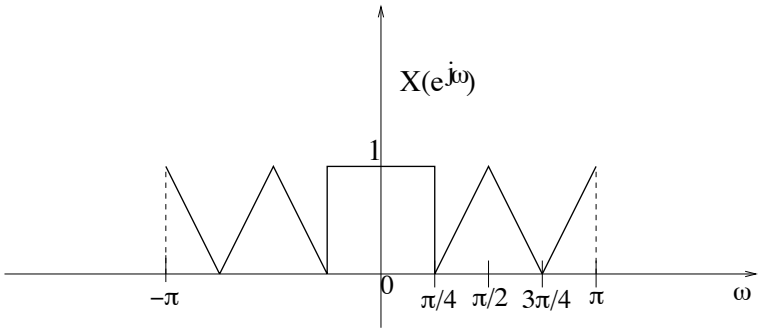


Figure 1.10: Spectrum of $x[n]$.

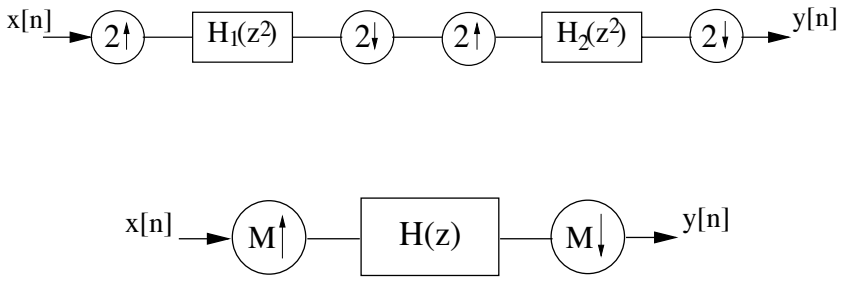


Figure 1.11: Multirate filtering.

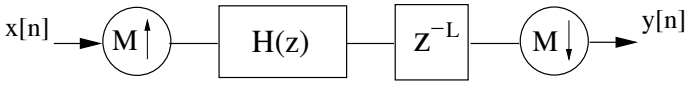


Figure 1.12: Multirate system implementing a fractional delay.

Chapter 2

Filter Banks and Discrete-Time Bases

In this chapter, we study how to design perfect reconstruction filter banks. We mostly concentrate on two-channel filter banks and tree-structured filter banks. We then show how these constructions are related to the design of bases of $l_2(\mathbb{Z})$.

For a more detailed treatment of the topic we refer to [28, 36].

2.1 Two-channel Filter Banks

Consider the two-channel filter bank shown in Figure 2.1. After downsampling, the signal might be coded for storage or transmission. Perfect reconstruction assumes no compression. Our goal is to discover the conditions for perfect reconstruction. That is, conditions that guarantee that $\hat{x}[n] = x[n]$. We will discover that perfect reconstruction is achieved when $g_0[n]$ and its shifted versions by two, together with $g_1[n]$ and its shifted versions by two form an orthogonal or biorthogonal basis of $l_2(\mathbb{Z})$.

Let us analyze this filter bank in the z-domain first. We have that

$$Y_0(z) = \frac{1}{2}(H_0(z^{1/2})X(z^{1/2}) + H_0(-z^{1/2})X(-z^{1/2}))$$

and

$$Y_1(z) = \frac{1}{2}(H_1(z^{1/2})X(z^{1/2}) + H_1(-z^{1/2})X(-z^{1/2})).$$

Thus,

$$\hat{X}(z) = \frac{1}{2}G_0(z)\,(H_0(z)X(z) + H_0(-z)X(-z)) + \frac{1}{2}G_1(z)\,(H_1(z)X(z) + H_1(-z)X(-z))$$

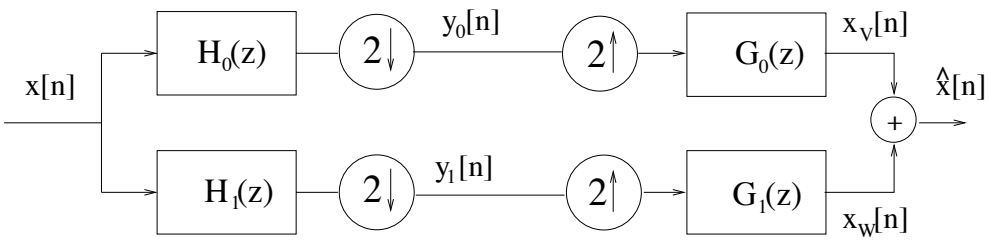


Figure 2.1: Two-channel filter bank.

So we have perfect reconstruction ($\hat{X}(z) = X(z)$) if and only if

$$\begin{aligned} &H_0(z)G_0(z) + H_1(z)G_1(z) = 2 \qquad &\text{(distortion-free)} \\ &H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0 \qquad &\text{(aliasing-free)} \end{aligned} \tag{2.1}$$

Our target is to find systematic ways to design the filters h_0, h_1, g_0, g_1 such that the perfect reconstruction conditions are satisfied. We will concentrate on FIR filters of length L, and in the orthogonal case, L will be an even number.

2.1.1 Orthogonal Filter Banks

Let us consider the orthogonal projector P we had designed previously. Recall that $g_0[n]$ was chosen such that $\langle g_0[n], g_0[n-2k]\rangle = \delta_k$ and that $h_0[n] = g_0[-n]$. We have seen that, in this case, the signal $x_V[n]$ obtained by filtering $x[n]$ with $h_0[n]$, sub-sampling by two, up-sampling by two and finally filtering with $g_0[n]$, is the orthogonal projection of $x[n]$ onto the subspace $V = \text{span}\{g_0[n-2k]\}_{k\in\mathbb{Z}}$. Now the error $e[n] = x[n] - x_V[n]$ due to this approximation is orthogonal to $x_V[n]$, since P is an orthogonal projection. That is, $e[n]$ lies on a subspace W that is the orthogonal complement to V. Namely, $V \perp W$ and $l_2(\mathbb{Z}) = V \oplus W$.

Our aim, now, is to design $h_1[n], g_1[n]$ such that the lower branch of the filter bank performs an orthogonal projection of $x[n]$ onto W. Since $V \perp W$, we first need to design $g_1[n]$ such that $\langle g_0[n], g_1[n-2k]\rangle = 0$. First notice that

$$\langle g_0[n], g_1[n-2k]\rangle = 0 \longleftrightarrow G_0(z)G_1(z^{-1}) + G_0(-z)G_1(-z^{-1}) = 0$$

Choose $g_1[n] = (-1)^n g_0[1-n]$ (shift and modulation) this is the *trick* that guarantees orthogonality. In fact, using the z-transform table , we have that $G_1(z) = -z^{-1}G_0(-z^{-1})$ which guarantees that $G_0(z)G_1(z^{-1}) + G_0(-z)G_1(-z^{-1}) = 0$. So orthogonality is satisfied. Moreover, $g_1[n]$ also satisfies $\langle g_1[n], g_1[n-2k]\rangle = \delta_k$. Thus, by choosing $h_1[n] = g_1[-n]$ we are sure that the lower branch of the filter bank is performing an orthogonal projection onto $\text{span}\{g_1[n-2k]\}_{k\in\mathbb{Z}}$.

To summarize, $x_V[n]$ is the orthogonal projection of $x[n]$ onto $V = \text{span}\{g_0[n-2k]\}_{k\in\mathbb{Z}}$, $x_W[n]$ is the orthogonal projection of $x[n]$ onto $W = \text{span}\{g_1[n-2k]\}_{k\in\mathbb{Z}}$. Moreover, by construction, $V \perp W$. The only question left to answer is: are we sure that $V \oplus W = l_2(\mathbb{Z})$? Or is it always true that $x_W[n] + x_V[n] = x[n]$? The answer is yes and we can verify this by noticing that the four filters satisfy the perfect reconstruction conditions. In fact, we have that:

$$\begin{gathered} \langle g_0[n], g_0[n-2k]\rangle = \delta_k \longleftrightarrow G_0(z)G_0(z^{-1}) + G_0(-z)G_0(-z^{-1}) = 2 \\ h_0[n] = g_0[-n] \longleftrightarrow H_0(z) = G_0(z^{-1}) \\ g_1[n] = (-1)^n g_0[1-n] \longleftrightarrow G_1(z) = -z^{-1}G_0(-z^{-1}) \\ h_1[n] = g_1[-n] \longleftrightarrow H_1(z) = G_1(z^{-1}). \end{gathered}$$

By replacing the above equalities in (2.1), we can easily see that the perfect reconstruction conditions are indeed satisfied. Therefore, it is always true that $x_W[n] + x_V[n] = x[n]$ and this means that $V \oplus W = l_2(\mathbb{Z})$.

We can rewrite the filter bank decomposition in time domain. Call $\varphi_{2k}[n] = g_0[n-2k]$ and $\varphi_{2k+1}[n] = g_1[n-2k]$, we have just shown that $x[n] = x_V[n] + x_W[n]$. For (1.3) and for (1.4) we have that

$$x_V[n] = \sum_k y_0[k]g_0[n-2k] = \sum_k \langle g_0[i-2k], x[i]\rangle g_0[n-2k] = \sum_k \langle \varphi_{2k}[i], x[i]\rangle \varphi_{2k}[n].$$

Likewise

$$x_W[n] = \sum_k \langle \varphi_{2k+1}[i], x[i] \rangle \varphi_{2k+1}[n].$$

This yields the orthonormal expansion

$$x[n] = \sum_k \langle \varphi_k[i], x[i] \rangle \varphi_k[n]$$

with $\langle \varphi_k[n], \varphi_j[n] \rangle = \delta_{kj}$.

Thus, putting all together we have that $\text{span}\{g_0[n-2k]\}_{k\in\mathbb{Z}}$ is an orthogonal basis for V, $\text{span}\{g_1[n-2k]\}_{k\in\mathbb{Z}}$ is an orthogonal basis for W and $l_2(\mathbb{Z}) = V \oplus W$ with $V \perp W$. We have developed perfect reconstruction filter banks as orthonormal expansions and projections into complementary sub-spaces.

As we shall see later, $g_0[n]$ is usually a low-pass filter, while $g_1[n]$ is high-pass. Therefore, V is a 'smooth' sub-space and provide a coarse or smooth approximation of $x[n]$, W represents the missing details of the smoothed approximation.

Orthogonal filter banks are quite constrained. The filters h_0, h_1 and g_1 are all determined by g_0. Thus the only freedom we have is in the choice of $g_0[n]$. However, $g_0[n]$ has to satisfy $\langle g_0[n], g_0[n-2k] \rangle = \delta_k$. In the next section we show how to construct $g_0[n]$. This will lead to the design of the famous 'Daubechies' filters.

2.1.2 Daubechies Filters

This section is about an important family of orthogonal filters which will lead to a famous family of wavelets.

We want to design an FIR filter $g_0[n]$ satisfying

$$\langle g_0[n], g_0[n-2k] \rangle = \delta_k \longleftrightarrow G_0(z)G_0(z^{-1}) + G_0(-z)G_0(-z^{-1}) = 2.$$

Call $P(z) = G_0(z)G_0(z^{-1})$, $P(z) = \sum_n p[n]z^{-n}$ has to satisfy the *halfband* property: $P(z)+P(-z) = 2$. In frequency, $P(e^{j\omega}) = |G_0(e^{j\omega})|^2$ achieves the orthogonality condition $|G_0(e^{j\omega})|^2 + |G_0(e^{j\omega+\pi})|^2 = 2$.

Since $P(z) = P(1/z)$, $P(z)$ has symmetric coefficients $p[n] = p[-n]$. If $g_0[n]$ is of length $N+1$ then $P(z) = \sum_{-N}^{N} p[n]z^{-n}$ is a trigonometric polynomial of degree N. Because of symmetry, there are only $N+1$ independent coefficients in P and the same in g_0.

From the above, it is obvious that if z_k is a zero of $P(z)$ so is $1/z_k$ (that also means that zeros on the unit circle are of even multiplicity). When $g_0[n]$ has real coefficients as in our case, then the complex conjugate z_k^* is a root when z_k is a root. Thus, complex roots come four at times $z_k, z_k^*, 1/z_k, 1/z_k^*$. Roots on the unit circle come two at a time instead. Thus the polynomial $z^N P(z)$ of degree $2N$ can be factored into its zeros as follows (recall that $P(z)$ must have $2N$ factors)

$$z^N P(z) = \alpha \prod_{i=1}^{M} (z - z_i)(z - \frac{1}{z_i}) \prod_{j=1}^{N-M} (z - z_j)^2,$$

where the first product contains pairs of zeros inside/outside the unit circle and the last product contains the zeros on the unit circle.

Suppose now that we are given the autocorrelation function $P(z)$ and we want to find $G_0(z)$. In this case, $G_0(z)$ is called a spectral factor of $P(z)$ and the technique of extracting $G_0(z)$ from $P(z)$ is called *spectral factorization*. The spectral factors are obtained by assigning one zero out of each

zero pair to $G_0(z)$. Different choices lead to different factors. Note that all these solutions have the same magnitude response but different phase. An important solution is the minimum phase solution in which one consistently chooses the zeros inside the unit circle.

Daubechies filters are obtained through spectral factorization of $P(z)$. These filters possess several key properties. They are the shortest orthogonal FIR filters with maximum flat frequency responses at $\omega = 0$ and $\omega = \pi$. The lowpass filters have p zeros at π. They have $2p$ coefficients. This last property is of fundamental importance in the construction of wavelets. It relates directly to the number of vanishing moments in the corresponding wavelets as we shall see in the next chapter.

If we want the low-pass filters to have p zeros at $\omega = \pi$, then the filters must be of length $2p$ which is the minimum possible length for orthogonal filters. This happens because the minimum number of requirements to satisfy the two conditions of orthogonality and flatness is $p + p = 2p$. Thus, if the filter has length $N = 2p$ the conditions we must impose are

- $G_0(e^{j\omega})$ has a zero of order p at π:

  $$G_0(e^{j\pi}) = G_0'(e^{j\pi}) = ... = G_0^{(p-1)}(e^{j\pi}) = 0.$$

  In the z-domain this means that $G(z)$ must have a factor of the form $(\frac{1+z^{-1}}{2})^p$.

- $P(z)$ is halfband, namely

  $$P(z) + P(-z) = 2.$$

  Since $P(z)$ is symmetric this means that

  $$p(0) = 1, \text{ and } p(2) = p(4) = ... = p(2p-2) = 0.$$

As we have already said, the p zeros at π mean that $G_0(e^{j\omega})$ has a factor $(1 + e^{-j\omega})^p$. Thus

$$G_0(e^{j\omega}) = \left(\frac{1 + e^{-j\omega}}{2}\right)^p R(e^{j\omega}).$$

$R(e^{j\omega})$ has degree $p-1$ and the p coefficients of $R(e^{j\omega})$ are chosen to satisfy the halfband condition. Therefore $P(z)$ can be written as follows (remember that $P(z) = P(1/z)$)

$$P(z) = \left(\frac{1 + z^{-1}}{2}\right)^p \left(\frac{1 + z}{2}\right)^p R(z)R(z^{-1}).$$

Ingrid Daubechies found an explicit formula for $P(z)$ for any choice of p:

$$P(\omega) = 2\left(\frac{1 + \cos\omega}{2}\right)^p \sum_{k=0}^{p-1} \left(\begin{array}{c} p+k-1 \\ k \end{array}\right) \left(\frac{1 - \cos\omega}{2}\right)^k.$$

Using Euler's formula one can rewrite the above expression in terms of z and then factor $P(z)$ to get $G_0(z)$. For $p = 1$ one obtains the Haar filters.

Example (p=2)

$$P(\omega) = 2\left(\frac{1 + \cos\omega}{2}\right)^2 \left[1 + 2\left(\frac{1 - \cos\omega}{2}\right)\right].$$

By replacing $2\cos\omega$ with $e^{j\omega} + e^{-j\omega} = z + z^{-1}$, we obtain

$$P(z) = \frac{1}{16}(1+z)^2(1+z^{-1})^2(-z+4-z^{-1}) = \frac{1}{16\alpha}(1+z)^2(1+z^{-1})^2(1-\alpha z)(1-\alpha z^{-1})$$

with $\alpha = 2 \pm \sqrt{3}$. Thus

$$G_0(z) = \frac{1}{4\sqrt{\alpha}}(1+z^{-1})^2(1-\alpha z^{-1}) = \frac{1}{4\sqrt{2}}\left[(1+\sqrt{3}) + (3+\sqrt{3})z^{-1} + (3-\sqrt{3})z^{-2} + (1-\sqrt{3})z^{-3}\right].$$

2.1.3 Biorthogonal Filter Banks

In the previous section, we have seen that the construction of orthogonal filter banks is quite constrained. The filter $G_0(z)$ is designed, for instance, through spectral factorization of $P(z)$. After that, the other three filters are derived directly from $G_0(z)$. Therefore there is no freedom in the design of these three filters. This has an important implication: With the exception of the Haar filter bank, it is not possible to design perfect-reconstruction real-valued linear-phase orthogonal FIR filter banks [28]. Essentially, if you need symmetric or anti-symmetric filters in your decomposition you have to relax orthogonality or may have to use complex-valued filters. The design of complex filters is beyond the scope of this notes and we refer to the original work by Lina and Mayrand for more details [19]. Instead, we will concentrate on the design of biorthogonal real-valued filter banks and present some symmetric or anti-symmetric constructions.

By relaxing the orthogonality condition, we gain some freedom. In particular, we do not need $H_0(z) = G_0(z^{-1})$. Assume that $H_0(z)$ and $G_0(z)$ are designed such that

$$H_0(z)G_0(z) + H_0(-z)G_0(-z) = 2.$$

Then design $g_1[n]$ orthogonal to $h_0^T[n]$ and $h_1^T[n]$ orthogonal to $g_0[n]$. That is, $G_1(z) = z^{-1}H_0(-z)$ and $H_1(z) = zG_0(-z)$ (modulation and shift, the usual *trick*). This new set of filter banks satisfy the perfect reconstruction conditions.

From a geometric point of view, we have generated a biorthogonal filter bank. In fact, the upper branch of the filter bank is **not** performing an orthogonal projection anymore. It is easy to see that the new operator $\hat{P}$ is idem-potent: $\hat{P}^2 = \hat{P}$ but it is not self-adjoint: $\hat{P}^* \neq \hat{P}$. This is an *oblique* projector. Perfect reconstruction is achieved if the biorthogonal conditions are satisfied. Namely, $h_0^T[n-2k]$ must be orthogonal to $g_1[n-2k]$ and $h_1^T[n-2k]$ must be orthogonal to $g_0[n-2k]$, which is achieved by imposing $G_1(z) = z^{-1}H_0(-z)$ and $H_1(z) = zG_0(-z)$.

So in the biorthogonal case, we have the freedom to choose two filters ($H_0(z)$ and $G_0(z)$). The design of them can be obtained using spectral factorization again. Call $P(z) = H_0(z)G_0(z)$. The condition we want the two filters to satisfy is $H_0(z)G_0(z)+H_0(-z)G_0(-z) = 2$ or $P(z)+P(-z) = 2$. This is exactly the half-band condition of the orthogonal case. The main difference, compared to the previous case, is that we can factorize $P(z)$ as we like since we do not have to satisfy the condition $H_0(z) = G_0(z^{-1})$ anymore.

Example Consider the half-band filter of the previous example:

$$P(z) = \frac{1}{16\alpha}(1+z)^2(1+z^{-1})^2(1-\alpha z^{-1})(1-\alpha z).$$

In the orthogonal case we were forced to assign one zero out of each pair to $G_0(z)$ to achieve $H_0(z) = G_0(z^{-1})$.

In the biorthogonal case, we can assign the zeros arbitrarily and this leads to a variety of factorizations. For instance we can have

$$G_0(z) = \frac{1}{2\sqrt{2}}(1+z^{-1})^2 \text{ and } H_0(z) = \frac{\sqrt{2}}{8\alpha}(1+z)^2(1-\alpha z^{-1})(1-\alpha z).$$

These filters form the shortest symmetrical biorthogonal pair of order 2 (i.e., they have two zeros at π). They are known as the 5/3 LeGall filters [18] and are quite important since they are used in the new image compression standard (JPEG2000). The other biorthogonal filters used in JPEG2000 are the so called 9/7 Daubechies filters, they were first introduced by Antonini et al. in [1]. For a very nice overview of the properties of these decompositions we refer to [33].

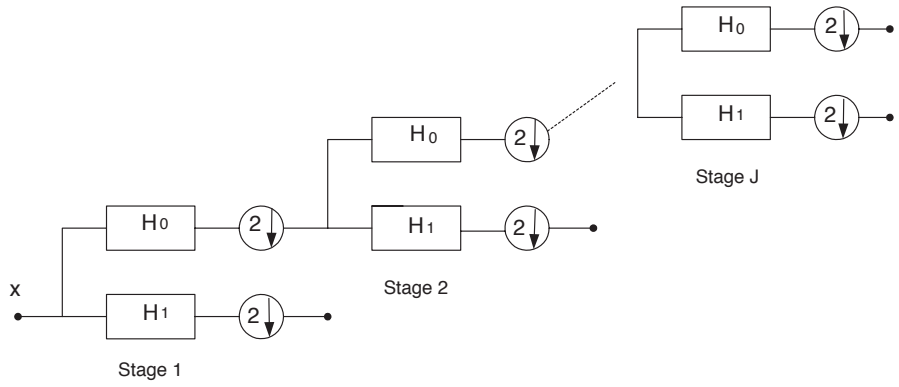


Figure 2.2: Analysis part of a J stages tree-structured filter bank.

An alternative solution is given by

$$G_0(z) = \frac{1}{4\sqrt{2}}(1+z^{-1})^3 \text{ and } H_0(z) = \frac{\sqrt{2}}{4\alpha}(1+z)(1-\alpha z^{-1})(1-\alpha z).$$

2.2 Tree-Structured Filter-Banks

A two-channel filter bank splits the input signal into two components: a low-pass or coarse version and a high-pass version. It is possible to iterate this process by splitting the two components again. Usually the process is iterated on the low-pass version leading to the decomposition shown in Figure 2.2. This is usually called an octave-band filter-bank.

Assume that the process is iterated J times. Using Noble identities we can see that the equivalent synthesis filters are given by

$$G_0^{(J)}(z) = G_0^{(J-1)}(z)G_0(z^{2^{J-1}}) = \prod_{k=0}^{J-1} G_0(z^{2^k})$$

$$G_1^{(j)}(z) = G_0^{(j-1)}(z)G_1(z^{2^{j-1}}) = G_1(z^{2^{j-1}}) \prod_{k=0}^{j-2} G_0(z^{2^k}), \quad j = 1, ..., J.$$

The tree-structured filter bank and the equivalent analysis and synthesis filters for the case $J = 2$ are shown in Figure 2.3 and 2.4 respectively. Assume, for simplicity, that the filter bank is orthogonal. In an octave-band filter banks, the coarse version of the original signals is recursively decomposed. This leads to a hierarchy of resolutions also called *multiresolution decomposition*. The multiresolution property of this decomposition is of fundamental importance in image processing and computer vision and has been exploited in many practical situations. We can formalize this multiresolution concept recalling that in the orthogonal case $\text{span}\{g_0[n-2k]\}_{k\in\mathbb{Z}} = V_1$, $\text{span}\{g_1[n-2k]\}_{k\in\mathbb{Z}} = W_1$ with $V_1 \perp W_1$ and $V_1 \oplus W_1 = l_2(\mathbb{Z})$. Now, when we iterate the filter bank decomposition on the coarse version

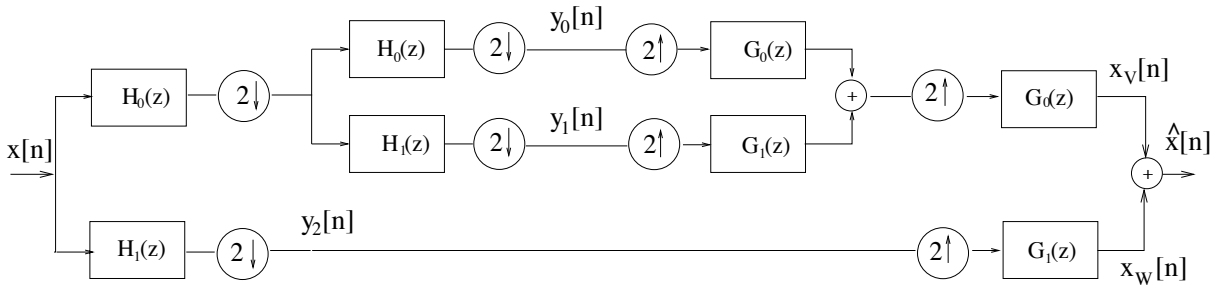


Figure 2.3: Two-stages of a tree-structured filter bank.

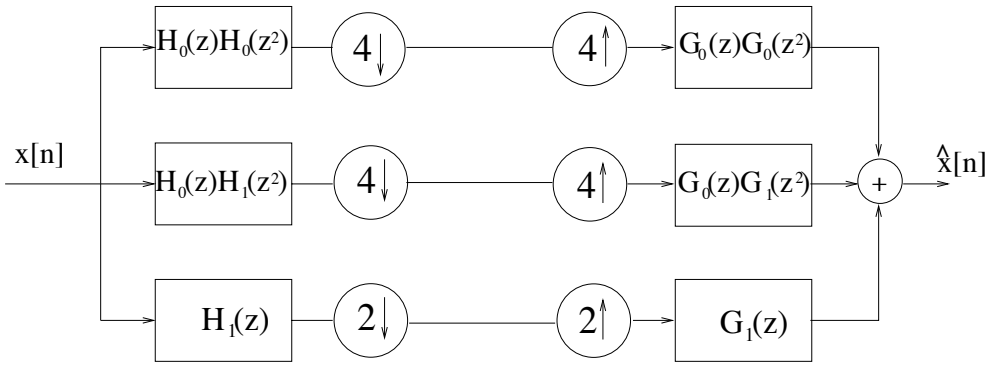


Figure 2.4: Equivalent filters for the tree-structured filter bank of Figure 2.3.

of $x[n]$, we are simply projecting the coarse version of $x[n]$ onto two new orthogonal sub-spaces. Call them V_2 and W_2. By construction we have that $V_1 = V_2 \oplus W_2$. and that $l_2(\mathbb{Z}) = V_2 \oplus W_2 \oplus W_1$. A J-stage octave-band filter bank leads to the creation of a sequence of embedded subspaces

$$V_j \subset \cdots \subset V_2 \subset V_1 \subset V_0$$

and to the decomposition of the space $l_2(\mathbb{Z})$ into

$$l_2(\mathbb{Z}) = W_1 \oplus W_2 \oplus \cdots \oplus W_J \oplus V_J.$$

It also follows that $V_j = \text{span}\{g_0^{(j)}[n - 2^j k]\}_{k\in\mathbb{Z}}$ and that $W_j = \text{span}\{g_1^{(j)}[n - 2^j k]\}_{k\in\mathbb{Z}}$ for $j = 1, ..., J$.

Example 4 *Consider the two-channel filter bank of Figure 2.1 and assume that* $G_0(z) = \frac{1+z^{-1}}{\sqrt{2}}$, $G_1(z) = \frac{1-z^{-1}}{\sqrt{2}}$, $H_0(z) = G_0(z^{-1})$ *and* $H_1(z) = G_1(z^{-1})$. *Namely, assume you have a Haar filter bank.*

Assume that the input signal $x[n]$ *is the one depicted in Figure 2.5. The corresponding* $x_{V_1}[n]$ *and* $x_{W_1}[n]$ *signals are shown in Figure 2.6(a) and 2.6(b) respectively. Recall that* $x_{V_1}[n]$ *is the orthogonal projection of* $x[n]$ *onto the sub-space* $V_1 = span\{g_0[n - 2k]\}_{k\in\mathbb{Z}}$ *and is therefore piecewise constant with pieces of size two.*

Assume now that the decomposition is iterated once on the low-pass branch leading to the tree-structured filter bank shown in Figure 2.3. The equivalent analysis and synthesis filters are shown in 2.4. Here $G_0^{(2)}(z) = G_0(z)G_0(z^2) = \frac{1+z^{-1}+z^{-2}+z^{-3}}{2}$ *and* $G_1^{(2)} = \frac{1+z^{-1}-z^{-2}-z^{-3}}{2}$. *The corresponding*

projections $x_{V_2}[n]$ *and* $x_{W_2}[n]$ *are depicted in Figure 2.6(c) and 2.6(d) respectively. The signal* $x_{V_2}[n]$ *is the orthogonal projection of* $x[n]$ *onto the sub-space* $V_2 = span\{g_0^{(2)}[n-4k]\}_{k\in\mathbb{Z}}$ *and, as expected, is piecewise constant with pieces of size four. You can also observe that* $x_{V_1} = x_{V_2} + x_{W_2}$*, this is also expected since* $V_1 = V_2 \oplus W_2$*. Notice that the regions were* $x[n]$ *is constant are normally represented exactly by* $x_{V_2}[n]$ *and, therefore, the corresponding* details *signals* x_{W_1}*,* x_{W_2} *are exactly zero in those regions.*

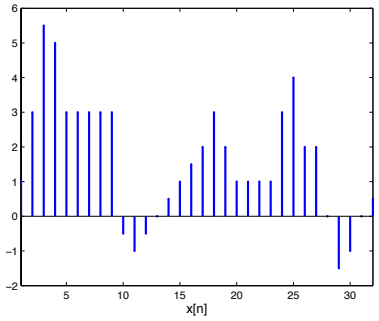
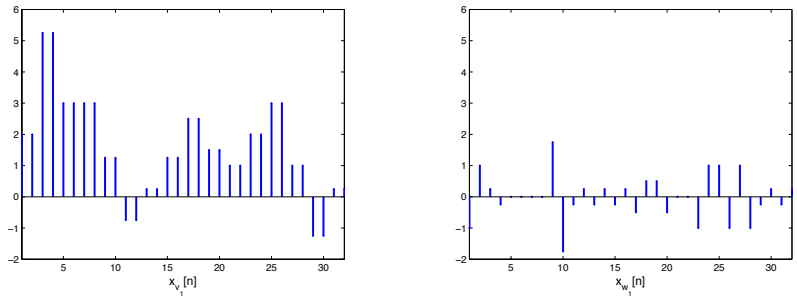
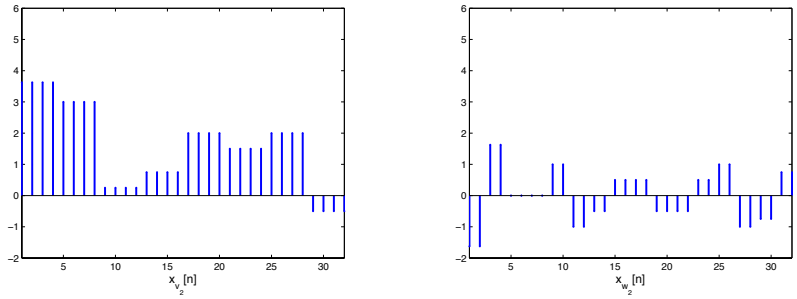


Figure 2.5: The discrete-time signal $x[n]$.



(a) $x_{V_1}[n]$ (b) $x_{W_1}[n]$



(c) $x_{V_2}[n]$ (d) $x_{W_2}[n]$

Figure 2.6: Decomposition of the signal $x[n]$ of Figure 2.5 with the tree-structured filter bank of Figure 2.3.

2.3 Exercises

2.1 *Spectral factorization method for two-channel filter banks.* Consider the factorization of $P(z)$ in order to obtain orthogonal or biorthogonal filter banks.

(a) Take

$$P(z) = (\frac{z^3}{2} + 1 + \frac{z^{-3}}{2}).$$

Compute a linear phase factorization of $P(z)$. In particular, assume that $H_0(z) = (z - 1 + z^{-1})$. Given this choice of $H_0(z)$, give the other filters of this biorthogonal filter bank.

(b) Now build an orthogonal filter bank based on this $P(z)$. (Hint: Remember that, if z_k is a root of $P(z)$ so is $1/z_k$, z_k^* and $1/z_k^*$).

2.2 Consider the three-channel filter bank shown in Figure 2.7

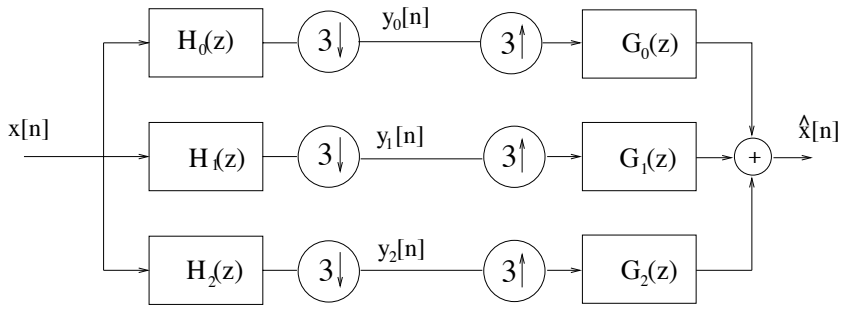


Figure 2.7: Three-channel filter bank.

(a) Express $\hat{X}(z)$ as a function of $X(z)$ and the filters. Then, derive the three perfect reconstruction conditions the filters have to satisfy.

(b) Assume that $G_0(z)$, $G_1(z)$ and $G_2(z)$ are $\frac{1}{3}$-band ideal filters as shown in Figure 2.8, and assume that $H_i(z) = G_i(z^{-1})$, for $i = 0, 1, 2$. Sketch and dimension the Fourier transform of $y_0[n], y_1[n], y_2[n]$ and $\hat{x}[n]$ assuming that $x[n]$ has the spectrum shown in Figure 2.9.

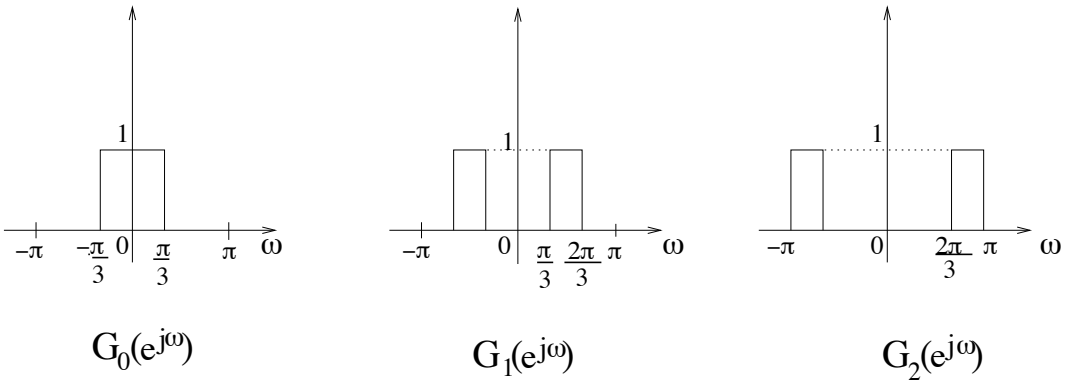


Figure 2.8: Fourier transforms of the synthesis filters of Figure 2.7.

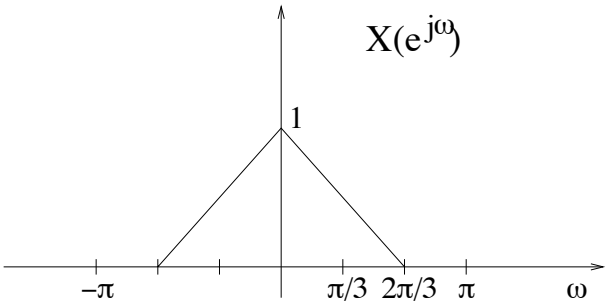


Figure 2.9: Spectrum of $x[n]$.

(c) Now, the filter bank is iterated on the H_0 branch to form a 2-level decomposition.

i. Draw either the synthesis or the analysis filter bank of the equivalent 5-channel filter bank clearly specifying the transfer functions and downsampling factors.

ii. If the filters are $\frac{1}{3}$-band and ideal as shown in Figure 2.8, draw the Fourier transform of the equivalent filters of each branch before downsampling.

2.3 Consider the two-channel filter bank of Figure 2.1

(a) Assume that $G_0(z) = (1+z^{-1})(a+bz^{-1}+az^{-2})$ with $a \neq 0$ and $b \neq 0$. Find the values of a and b such that the condition $\langle g_0[n], g_0[n-2k]\rangle = \delta[k]$ is satisfied, where $\langle g_0[n], g_0[n-2k]\rangle$ denotes the inner product between $g_0[n]$ and $g_0[n-2k]$.

(b) Assume $G_0(z)$ is the filter you obtain in part (a). Design the filters $H_0(z), H_1(z), G_1(z)$ in order to have a perfect reconstruction orthogonal filter bank.

(c) Consider the two-channel filter bank of Figure 2.1 without downsamplers and upsamplers. Such a filter bank is called a *Nonsubsampled* filter bank. Choose $\{H_0(z), H_1(z), G_0(z), G_1(z)\}$ as in an orthogonal two-channel filter bank. What is $\hat{x}[n]$ as a function of $x[n]$ and the filters?

(d) Assume $H_0(z)$ and $G_0(z)$ are given, show how to obtain $H_1(z)$ and $G_1(z)$ such that $\hat{x}[n] = x[n]$ in the nonsubsampled filter bank. Assume that $H_0(z) = 1$ and $G_0(z) = z^{-1} + 4 + z$, calculate $\{H_1(z), G_1(z)\}$. Is the solution $\{H_1(z), G_1(z)\}$ unique? If not, provide at least two more alternative solutions.

2.4 A transmultiplexer is the dual of a subband coder. Two signals are multiplexed and sent over a high bandwidth channel. A perfect reconstruction (PR) multiplexer cancels crosstalk and reconstruct the signals exactly.

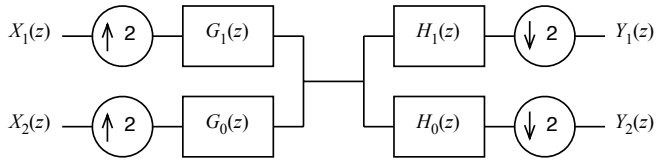


Figure 2.10: The Transmultiplexer.

(a) Give the input/output relations in the z-transform domain. What are the conditions on the filters that guarantee that the trans-multiplexer is PR?

(b) Suppose that you have a power complementary filter $G_0(z)$ (i.e., $g_0[n]$ is such that $\langle g_0[n], g_0[n-2k]\rangle = \delta_k$). How can you use it to get a PR trasmultiplexer? Specify all four filters in terms of this prototype.

2.5 *The Laplacian Pyramid (LP)* as shown in Figure 2.11 is frequently used in computer vision. The basic idea of the LP is the following: First, derive a coarse approximation of the original signal by lowpass filtering and downsampling. Based on this coarse version, predict the original (by upsampling and filtering) and then calculate the difference as the predictor error. Transmit $c[n]$ and $d[n]$. The corresponding synthesis system is shown in Figure 2.12.

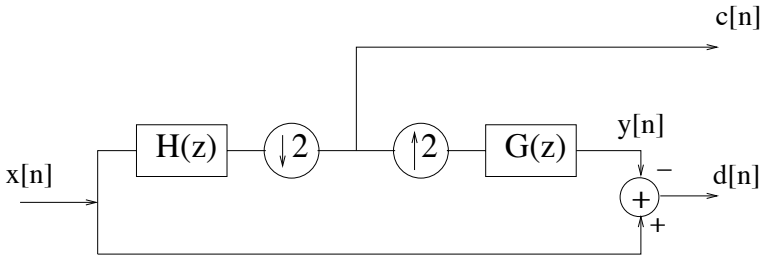


Figure 2.11: Decomposition of $x[n]$ using the Laplacian Pyramid.

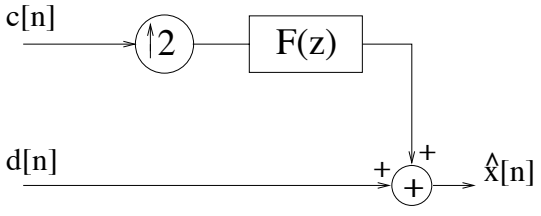


Figure 2.12: Reconstruction using the synthesis part of the LP.

(a) Express $\hat{X}(z)$ as a function of $X(z)$ and the filters. Then, derive the perfect reconstruction condition(s) the filters have to satisfy.

(b) Assume that $G(z)$ is half-band ideal low-pass filter as shown in Figure 2.13 with $A = \sqrt{2}$, also assume that $H(z) = G(z^{-1})$. Sketch and dimension the Fourier transform of c[n] and d[n] assuming that x[n] has the spectrum shown in Figure 2.14.

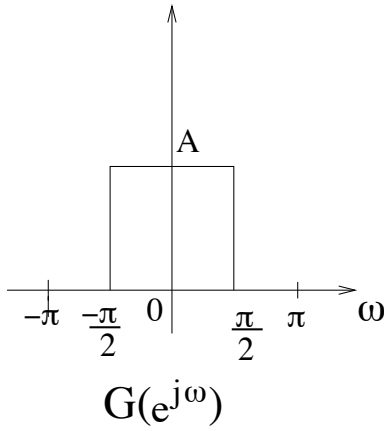


Figure 2.13: Lowpass filter.

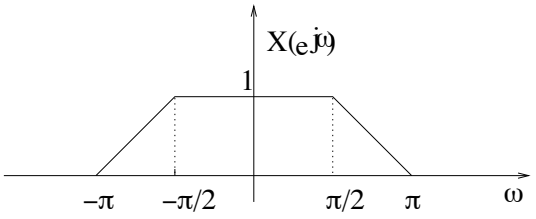


Figure 2.14: Fourier transform of $x[n]$

(c) Consider a filter $G(z) = z^{-1} + 2 + z$ and assume $H(z) = G(z)/2$. (This is very similar to the original Laplacian pyramid contruction). Show that the operator P that converts $x[n]$ into $y[n]$ is sub-optimal since it is not idempotent. That is $P^2 \neq P$.

(d) With $G(z) = z^{-1} + 2 + z$, design a 5-tap symmetric filter $H(z)$ with two zeros at $z = -1$ such that the idempotent contraint is met. That is, design $H(z)$ such that $P^2 = P$.

2.6 *Spectral Factorization methods for two-channel filter-banks.* First of all, recall that if a polynomial $P(z)$ is symmetric then if z_k is a root of $P(z)$, so is $1/z_k$. Moreover, when the coefficients of $P(z)$ are real then if z_k is a root of $P(z)$ so is z_k^* where * denotes the complex conjugate. Consider now the two-channel filter bank of Figure 2.1 and the 10th degree half-band polynomial $P(z) = (1+z)^3(1+z^{-1})^3 Q(z)$, where $Q(z)$ is a symmetric polynomial with real coefficients. Moreover $Q(z)$ has four complex roots in the right half complex plane.

(a) Denote with r one of the four complex roots of $Q(z)$ and assume $|r| < 1$. Draw a figure to show the ten roots on the complex plane. Notice that you do not need to compute the actual value of r.

(b) Without computing r, factorize $P(z)$ in order to have an orthogonal filter bank. Choose $G_0(z)$ to be minimum phase.

(c) Show that the high-pass branch of the orthogonal filter-bank you have just designed annihilates discrete-time polynomials of maximum degree 2. That is, show that $\sum_k x[n-k]h_1[k] = 0$, for $x[n] = n^l$ and $l = 0, 1, 2$.

(d) Now factorize $P(z)$ in order to have a biorthogonal filter bank with symmetric filters with real coefficients. There are many different possible factorizations, choose a factorization where both $G_0(z)$ and $H_0(z)$ have at least two zeros at $\omega = \pi$.

Chapter 3

Wavelet Bases of $L_2(\mathbb{R})$

In the previous chapter, we have studied perfect reconstruction filter banks and we have seen that the design of filter banks is related to the construction of orthogonal or biorthogonal bases of $l_2(\mathbb{Z})$. In the case of tree-structured filter banks we have also seen that the iteration of the decomposition of the low-pass component leads to a multiresolution decomposition of the original discrete-time signals.

We now move to the continuous-time scenario. Our target is to design good bases for $L_2(\mathbb{R})$. That is, we aim to find sets of elementary functions $\varphi_k(t)$ such that any continuous-time signals $f(t) \in L_2(\mathbb{R})$ can be expressed as a linear combination of the φ_ks.

We will introduce multiresolution analysis which is in spirit very similar to the one introduced in the previous chapter. The connection between the discrete-time world and the continuous-time one, however, does not reduce to the multiresolution definition. We will discover that good sets of bases of $L_2(\mathbb{R})$ can be obtained, under certain circumstances, by iterating the filter bank decomposition of the previous chapter. Moreover, most of the properties of the elementary functions $\varphi_k(t)$ (e.g., smoothness, vanishing moments, approximation power) can be inferred directly from the properties of the original filter bank. The main message here is that everything develops from the filter coefficients. Choose good coefficients and you get good continuous-time wavelets! If the filter bank is orthogonal you get orthogonal bases, you get biorthogonal bases otherwise.

This unifying theory that we will discover in this chapter remains one of the most fascinating element of the wavelet theory. So, enjoy it!

3.1 Some Examples of Series Expansions

We start by reviewing some known series expansions such as the Fourier series and the expansion of bandlimited signals. We then introduce the expansion in the Haar basis and discuss differences and analogies.

3.1.1 Periodic Signals and Bandlimited Signals

Our aim is to find sets of functions $\{\varphi_k(t)\}$ such that signals $f(t) \in L_2(\mathbb{R})$ can be expressed in terms of these elements. We will concentrate on the orthogonal case, therefore, our aim is to find orthonormal sets of functions (i.e., functions satisfying $\langle \varphi_k(t), \varphi_l(t) \rangle = \delta_{kl}$) that can express $f(t)$ as

$$f(t) = \sum_{k=-\infty}^{\infty} \langle \varphi_k(t), f(t) \rangle \varphi_k(t)$$

where

$$\langle \varphi_k(t), f(t) \rangle = \int_{-\infty}^{\infty} \varphi_k^*(t) f(t) dt.$$

A first example is the Fourier series. A periodic function of period T can be written in terms of sine and cosine waves or of complex exponentials:

$$f(t) = \sum_{k=-\infty}^{\infty} F[k] e^{j(2\pi kt)/T}.$$

The set of functions $\{e^{j(2\pi kt)/T}\}$ is orthogonal in the period T.

Another less-known example is that of bandlimited signals. We know that such signals can be sampled and perfectly reconstructed. Interesting enough such a reconstruction process can be interpreted in terms of expansions in orthogonal bases.

Assume $x(t)$ is bandlimited to $f_s = 1/(2T)$. Call $x_s(t)$ the sampled version of the signal with sampling period T or

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT).$$

The sampling theorem tells us that we can recover $x(t)$ by convolving $x_s(t)$ with the sinc function $\varphi(t) = \text{sinc}_T(t) = sin(\pi t/T)/(\pi t/T)$. That is

$$x(t) = x_s(t) * \varphi(t) = \sum_{n=-\infty}^{\infty} x(nT)\text{sinc}_T(t - nT) = \sum_{n=-\infty}^{\infty} x(nT)\text{sinc}(t/T - n).$$

Notice that the set $\{\text{sinc}_T(t-nT)\}_{n\in\mathbb{Z}}$ is orthogonal. Thus, any bandlimited signals can be expressed in terms of shifted versions of the sinc function.

3.1.2 The Haar Expansion

The limit of the two previous expansions is that they deal with a very limited class of signals, namely, bandlimited or periodic signals. Moreover, the previous two expansions have the advantage of being well localized in frequency, but they are not localized in time. For this reason any local perturbation in time is going to have an effect over the whole expansion. The short-time Fourier transform (STFT) was introduced precisely to reduce this problem. However, the STFT is redundant and the size of the time window is fixed. The advantage of the wavelet transform compared to the STFT is that the size of the window changes with the scale so that we can have a more flexible tiling of the time-frequency plane. In this section, we discuss a first example of the wavelet series given by the Haar expansion. The good news is that, though quite simple, this expansion contains all the elements and main properties of any wavelet expansion.

The Haar wavelet is defined as

$$\psi(t) = \begin{cases} 1 & 0 \le t < 1/2 \\ -1 & 1/2 \le t < 1 \\ 0 & \text{otherwise} \end{cases}$$

and the entire set of basis elements is obtained by dilation and translation of $\psi(t)$. That is

$$\psi_{m,n}(t) = 2^{-m/2}\psi(2^{-m}t - n) \qquad m, n \in \mathbb{Z}.$$

It is easy to see that this set of functions is orthonormal, that is $\langle \psi_{m,n}(t), \psi_{l,k}(t) \rangle = \delta_{m,l}\delta_{n,k}$. We now show, in an intuitive way, that this set is also complete, thus, it forms an orthonormal basis of $L_2(\mathbb{R})$. For a more rigorous proof we refer the reader to [36, 12].

Let us introduce the Haar scaling function first. It is defined as

$$\varphi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ \\ 0 & \text{otherwise} \end{cases}$$

Notice the analogy between the Haar functions and the Haar filter bank where the synthesis filters are given by $g_0[n] = 1/\sqrt(2)(\delta[n] + \delta[n-1])$ and $g_1[n] = 1/\sqrt(2)(\delta[n] - \delta[n-1])$. Notice also that $\varphi(t) = \sqrt{2}\sum_n g_0[n]\varphi(2t-n)$ and that $\psi(t) = \sqrt{2}\sum_n g_1[n]\varphi(2t-n)$. This is not a coincidence as we shall discover later on.

We want to show that any function $f(t) \in L_2(\mathbb{R})$ can be written in terms of the $\psi_{m,n}(t)$.

First, consider the space V_0 of all finite-energy functions $f(t)$ of the form

$$f(t) = \sum_{k\in\mathbb{Z}} \alpha_k \varphi(t-k),$$

where $\varphi(t)$ is the Haar scaling function. Clearly, by construction, V_0 is the space of piecewise constant functions with discontinuities at locations $k = ..., -2, 1, 0, 1, 2...$ and clearly $V_0 \subset Ł_2(\mathbb{R})$ since we can find finite-energy functions that are not in V_0. To represent more complicated functions we need to use thinner building blocks. For example we may consider $\varphi(2t)$ and we may define the new space V_{-1} of all finite-energy functions $f(t)$ of the form

$$f(t) = \sum_{k\in\mathbb{Z}} \alpha_k \varphi(2t-k).$$

Clearly, V_{-1} is the space of piecewise constant functions with discontinuities at location $k = ..., -1, -1/2, 0, 1/2, 1, ...$ and $V_0 \subset V_{-1}$. In general, for any $j \in \mathbb{Z}$ we can denote by V_{-j} the space spanned by $\varphi(2^j t - k)$ with $k \in \mathbb{Z}$. Clearly V_{-j} is the space of all piecewise constant functions with pieces of minimum size $1/2^j$ and discontinuities at locations $k = ..., -1/2^j, 0, 1/2^j, 2/2^j, ...$

Now, it is interesting to observe that the so-defined subspaces V_j are embedded:

$$... \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} ...$$

Notice that this happens because V_{-j} is defined using $\varphi(2^j t - k)$. That would have not been the case, had we used a different factor instead of the power of 2. For example, we can easily see that a definition of the spaces V_{-j} using $\varphi(jt-k)$ rather than $\varphi(2^j t-k)$ would not lead to embedded subspaces. Also notice that if $f(t) \in V_0$ then $f(t-n) \in V_0$ as well. Spaces satisfying the above property are called *shift-invariant*.

If we choose a sufficiently large negative j, we can approximate any function in $L_2(\mathbb{R})$ using $\varphi(2^j t - k)$. The open issue is: can we approximate any function in $L_2(\mathbb{R})$ using the $\psi_{m,n}(t)$ instead? Or using a combination of $\varphi(t)$ and $\psi_{m,n}(t)$? This is what we are going to show now.

Assume that the function $f(t)$ we want to represent has finite support on $[-2^{m_1}, 2^{m_1})$ and that it is constant on intervals $[n2^{-m_0}, (n+1)2^{-m_0})$. By choosing m_0 and m_1 large enough we can approximate any function in $L_2(\mathbb{R})$. This function can be expressed as a linear combination of shifted and rescaled versions of $\varphi(t)$. That is, call $\varphi_{m,n}(t) = 2^{-m/2}\varphi(2^{-m}t - n)$ then

$$f(t) = \sum_{n=-2^{(m_0+m_1)}}^{2^{(m_0+m_1)}-1} c_{-m_0,n}\varphi_{-m_0,n}(t).$$

Now consider two consecutive intervals $[2n2^{-m_0}, (2n+1)2^{-m_0})$ and $[(2n+1)2^{-m_0}, (2n+2)2^{-m_0})$. The function in these two intervals can be written as the average over the two intervals plus the difference. Call

$$c_{-m_0+1,n} = \frac{1}{\sqrt{2}}(c_{-m_0,2n} + c_{-m_0,2n+1})$$

and

$$d_{-m_0+1,n} = \frac{1}{\sqrt{2}}(c_{-m_0,2n} - c_{-m_0,2n+1}),$$

by applying the above procedure to the pairs of intervals of the whole function, we obtain

$$f(t) = \sum_n c_{-m_0+1,n}\varphi_{-m_0+1,n}(t) + d_{-m_0+1,n}\psi_{-m_0+1,n}(t).$$

That is, we have divided the function in a coarse component and in a detail component. This second component is represented by the wavelet function, the first one is constant on intervals of size $2^{-m_0+1}$and is represented by the scaling function. We can iterate the process on this coarse version and represent each pair of intervals $[2n2^{-m_0+1}, (2n+1)2^{-m_0+1})$, $[(2n+1)2^{-m_0+1}, (2n+2)2^{-m_0+1})$ in terms of $\varphi_{-m_0+2,n}(t)$ and $\psi_{-m_0+2,n}(t)$. The interesting point is that, since $f(t)$ has finite energy, the norm of the coarse version goes to zero as the scale goes to infinity. Thus the signal $f(t)$ can eventually be written in terms of its details only. That is,

$$f(t) = \sum_n \sum_m d_{n,m}\psi_{m,n}(t).$$

The above result tells us that any function with finite energy can be expressed as a succession of multiresolution details. Moreover, notice that in the above construction we have looked at spaces V_m of functions constant over intervals of length 2^m and decomposed those spaces into two orthogonal components $V_m = V_{m+1} \oplus W_{m+1}$. V_{m+1} is the space of piecewise constant function over intervals of length 2^{m+1}, the space W_{m+1} is spanned by the Haar wavelet at resolution 2^{m+1}, namely $\psi_{m+1,n}(t)$. Thus the Haar wavelet at a certain scale allows us to move from one coarse subspace V_m to the next finer one: V_{m-1}. We can thus notice the multiresolution structure of the Haar decomposition. This will be more formally discussed in the next section.

3.2 Multiresolution Analysis

Multiresolution analysis, developed by Mallat and Meyer [20, 21, 23], is a formal approach to constructing orthogonal wavelet bases using a definite set of rules and procedures.

Definition 11 *By a multiresolution analysis we mean a sequence of embedded closed subspaces*

$$...V_2 \subset V_1 \subset V_0 \subset V_{-1}...$$

such that

1. *Upward Completeness*

$$\lim_{m\to-\infty} V_m = \bar{\bigcup}_{m\in\mathbb{Z}} V_m = L_2(\mathbb{R})$$

2. *Downward Completeness*

$$\lim_{m\to\infty} V_m = \bigcap_{m\in\mathbb{Z}} V_m = \{0\}$$

3. *Scale Invariance*
$$f(t) \in V_m \leftrightarrow f(2^m t) \in V_0$$

4. *Shift Invariance*
$$f(t) \in V_0 \rightarrow f(t-n) \in V_0 \quad \textit{for all} \quad n \in \mathbb{Z}$$

5. *Existence of a Basis. There exists* $\varphi(t) \in V_0$*, such that*
$$\{\varphi(t-n)\}_{n\in\mathbb{Z}}$$
is an orthonormal basis for V_0.

The function $\varphi(t)$ is the scaling function. The orthogonality conditions $\langle \varphi(t-n), \varphi(t-m)\rangle = \delta_{m,n}$ is equivalent to the following in the Fourier domain (see [36, pag. 215]):
$$\sum_{k=-\infty}^{\infty} |\hat{\varphi}(\omega + 2k\pi)|^2 = 1.$$

The multiresolution analysis leads directly to the *two-scale equation*. Since V_1 is included in V_0, if $\varphi(t/2)$ belongs to V_1, it belongs to V_0 as well. Moreover, since $\{\varphi(t-n)\}_{n\in\mathbb{Z}}$ is a basis for V_0, we can express $\varphi(t/2)$ as a linear combination of $\{\varphi(t-n)\}_{n\in\mathbb{Z}}$, thus, we have that $\varphi(x/2) = \sqrt{2}\sum_{n=-\infty}^{\infty} g_0[n]\varphi(x-n)$. Replace x with $2t$ to obtain the two-scale relation
$$\varphi(t) = \sqrt{2}\sum_{n=-\infty}^{\infty} g_0[n]\varphi(2t-n).$$

Now, by taking the Fourier transform of both sides, we obtain
$$\hat{\varphi}(\omega) = \frac{1}{\sqrt{2}} G_0(e^{j\omega/2})\hat{\varphi}(\omega/2)$$

where
$$G_0(e^{j\omega}) = \sum_n g_0[n]e^{-j\omega n}.$$

Because of the orthogonality of $\varphi(t)$, $G(e^{j\omega})$ satisfies the following property
$$|G_0(e^{j\omega})|^2 + |G_0(e^{j(\omega+\pi)})|^2 = 2. \tag{3.1}$$

Proof: We know that
$$\sum_{k=-\infty}^{\infty} |\hat{\varphi}(2\omega + 2k\pi)|^2 = 1.$$

Thus we have that
$$\begin{aligned}
1 &= \tfrac{1}{2}\textstyle\sum_k |G_0(e^{j(\omega+k\pi)})|^2 |\hat{\varphi}(\omega+k\pi)|^2 \\
&= \tfrac{1}{2}\textstyle\sum_k |G_0(e^{j(\omega+2k\pi)})|^2 |\hat{\varphi}(\omega+2k\pi)|^2 \\
&+ \tfrac{1}{2}\textstyle\sum_k |G_0(e^{j(\omega+(2k+1)\pi)})|^2 |\hat{\varphi}(\omega+(2k+1)\pi)|^2 \\
&= \tfrac{1}{2}|G_0(e^{j\omega})|^2 \textstyle\sum_k |\hat{\varphi}(\omega+2k\pi)|^2 \\
&+ \tfrac{1}{2}|G_0(e^{j(\omega+\pi)})|^2 \textstyle\sum_k |\hat{\varphi}(\omega+(2k+1)\pi)|^2 \\
&= \tfrac{1}{2}(|G_0(e^{j\omega})|^2 + |G_0(e^{j(\omega+\pi)})|^2)
\end{aligned}$$

which completes the proof.

After these preliminary findings, we are in a position to outline the proof of the following fundamental theorem that indicates how the construction of wavelet bases is related to the multiresolution analysis.

Theorem 2 ([36, pp.218-219][22, pp.236-240]) *Let $\{V_n\}$, $n \in \mathbb{Z}$ be a multiresolution analysis with the scaling function $\varphi(t)$. There exists an orthonormal basis for $L_2(\mathbb{R})$:*

$$\psi_{m,n}(t) = 2^{-m/2}\psi(2^{-m}t - n) \qquad m, n \in \mathbb{Z}$$

and

$$\psi(t) = \sum_{n=-\infty}^{\infty} (-1)^n g_0[1-n]\varphi(2t-n)$$

such that $\{\psi_{m,n}\}$, $n \in \mathbb{Z}$ is an orthonormal basis for W_m, where W_m is the orthogonal complement of V_m in V_{m-1}.

Proof: We start the proof by constructing an orthogonal basis for the detail space W_m which is the orthogonal complement of V_m in V_{m-1} (i.e., $V_{m-1} = W_m \oplus V_m$) . We consider the case $m = 0$ which can be easily extended to $m \neq 0$ through proper scaling.

Recall that the scaling function satisfies the two-scale equation:

$$\varphi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} g_0[n]\varphi(2t-n).$$

We also know that $W_0 \in V_{-1}$ and that $\varphi(2t-n)$, $n \in \mathbb{Z}$ is a basis of V_{-1}. Therefore, any function in W_0 can be expressed as a linear combination of $\varphi(2t-n)$. We design our wavelet $\psi(t) \in W_0$ as follows

$$\psi(t) = \sqrt{2} \sum_n g_1[n]\varphi(2t-n)$$

where $g_1[n] = (-1)^n g_0[1-n]$ (do you remember the trick? Shift & modulation).

The so designed wavelet is an orthonormal basis of W_0. Indeed, as in the case of the scaling function, we know that $\{\psi(t-n)\}_{n\in\mathbb{Z}}$ is orthonormal if and only if

$$\sum_l |\hat{\psi}(\omega + 2k\pi)|^2 = 1$$

and this last condition is satisfied when

$$|G_1(e^{j\omega})|^2 + |G_1(e^{j\omega+\pi})|^2 = 2. \tag{3.2}$$

By construction we have that $G_1(e^{j\omega}) = -e^{-j\omega}G_0^*(e^{j(\omega+\pi)})$ and we know that $G_0(e^{j\omega})$ satisfies equation (3.1), thus by substituting $G_1(e^{j\omega})$ with $G_0(e^{j\omega})$ in (3.2), we obtain the desired condition

$$|G_1(e^{j\omega})|^2 + |G_1(e^{j\omega+\pi})|^2 = |G_0(e^{j\omega})|^2 + |G_0(e^{j\omega+\pi})|^2 = 2.$$

In a similar way, one can see that

$$\langle \varphi(t-k), \psi(t) \rangle = 0 \quad \text{for all } k.$$

In fact, by taking the Fourier transform of both sides and after some manipulations we obtain that

$$\langle \varphi(t-k), \psi(t) \rangle = 0 \leftrightarrow \sum_l \hat{\psi}(\omega + 2\pi l)\hat{\varphi}^*(\omega + 2\pi l) = 0$$

which in turn implies that

$$G_1(e^{j\omega})G_0^*(e^{j\omega}) + G_1(e^{j\omega+\pi})G_0^*(e^{j\omega+\pi}) = 0$$

and this last identity is clearly satisfied since $G_1(e^{j\omega}) = -e^{-j\omega}G_0^*(e^{j(\omega+\pi)})$.

Thus, by choosing $g_1[n] = (-1)^n g_0[1-n]$, we have been able to construct a family of orthogonal functions $\psi(t-n)$ that span W_0 and such that $W_0 \perp V_0$. We are now going to show that

$$V_{-1} = W_0 \oplus V_0. \tag{3.3}$$

We know that $\{\varphi(2t-n)\}_{n\in\mathbb{Z}}$ is a basis of V_{-1}, therefore, any signal $f(t) \in V_{-1}$ can be written as

$$f(t) = \sqrt{2}\sum_{n=-\infty}^{\infty} a[n]\varphi(2t-n)$$

for a proper choice of the coefficients $a[n]$. Condition (3.3) is satisfied when any $f(t) \in V_{-1}$ can be expressed in terms of $\{\varphi(t-n)\}_{n\in\mathbb{Z}}$ and $\{\psi(t-n)\}_{n\in\mathbb{Z}}$ or when

$$f(t) = \sqrt{2}\sum_{n=-\infty}^{\infty} a[n]\varphi(2t-n) = \sum_{n=-\infty}^{\infty} b[n]\varphi(t-n) + \sum_{n=-\infty}^{\infty} c[n]\psi(t-n)$$

for a proper choice of the coefficients $b[n]$ and $c[n]$. By writing the above equation in the Fourier domain we obtain:

$$\frac{1}{\sqrt{2}}A(e^{j\omega/2})\hat{\varphi}\left(\frac{\omega}{2}\right) = B(e^{j\omega})\hat{\varphi}(\omega) + C(e^{j\omega})\hat{\psi}(\omega). \tag{3.4}$$

We can also write the two-scale equations in the Fourier domain:

$$\begin{aligned}\hat{\varphi}(\omega) &= \tfrac{1}{\sqrt{2}}G_0(e^{j\omega/2})\hat{\varphi}(\omega/2),\\ \hat{\psi}(\omega) &= \tfrac{1}{\sqrt{2}}G_1(e^{j\omega/2})\hat{\varphi}(\omega/2).\end{aligned} \tag{3.5}$$

Thus by combining (3.4) and (3.5) we obtain

$$A(e^{j\omega/2}) = B(e^{j\omega})G_0(e^{j\omega/2}) + C(e^{j\omega})G_1(e^{j\omega/2})$$

and the above equality is satisfied when choosing:

$$B(e^{j2\omega}) = \frac{1}{2}\left[A(e^{j\omega})G_0^*(e^{j\omega}) + A(e^{j(\omega+\pi)})G_0^*(e^{j(\omega+\pi)})\right]$$

and

$$C(e^{j2\omega}) = \frac{1}{2}\left[A(e^{j\omega})G_1^*(e^{j\omega}) + A(e^{j(\omega+\pi)})G_1^*(e^{j(\omega+\pi)})\right].$$

This shows that $\{\psi(t-n)\}_{n\in\mathbb{Z}}$ is an orthonormal basis of W_0 and that W_0 is the orthogonal complement of V_0 in V_{-1} (i.e., $V_{-1} = W_0 \oplus V_0$). By using the same method but with proper scaling, we can also show that $\{\psi_{m,n}(t)\}_{n\in\mathbb{Z}}$ is an orthonormal basis for W_m and that $V_{m-1} = W_m \oplus V_m$[1].

We are now in a position to complete the proof by verifying that $\{\psi_{m,n}(t)\}_{m,n\in\mathbb{Z}}$ is an orthonormal basis of $L_2(\mathbb{R})$. First notice that the detail spaces $\{W_j\}_{j\in\mathbb{Z}}$ are orthogonal. In fact, by construction $W_j \perp V_j$, moreover $W_l \subset V_{l-1} \subset V_j$ for $j < l$. Therefore W_j and W_l are orthogonal.

[1] This is essentially a direct consequence of the scaling property of the V_m spaces.

We can also decompose $L_2(\mathbb{R})$ into the mutually orthogonal subspace W_j or

$$L_2(\mathbb{R}) = \bigoplus_{j=-\infty}^{\infty} W_j. \tag{3.6}$$

Indeed, since $V_{j-1} = V_j \oplus W_j$, for any $L < J$ we have that

$$V_L = \bigoplus_{j=L+1}^{J} W_j \oplus V_J.$$

Because of the upward/downward completeness properties, V_L and V_J tend respectively to $L_2(\mathbb{R})$ and $\{0\}$ when L and J go respectively to $-\infty$ and ∞ which leads to (3.6) and this completes the proof.

To summarize, by choosing $g_1[n] = (-1)^n g_0[1-n]$, we have been able to construct a family of orthogonal functions $\psi(t-n)$ that span W_0 and such that $W_0 \perp V_0$. We have then shown that W_0 is such that the condition $V_{-1} = W_0 \oplus V_0$ is satisfied. Then by proper scaling, we have seen that $\psi_{m,n}(t)$, $n \in \mathbb{Z}$ will be an orthonormal basis for W_m and, because of (3.6), the set $\{\psi_{m,n}\}_{m,n\in\mathbb{Z}}$ is complete, that is, it is an orthonormal basis of $L_2(\mathbb{R})$.

□

3.3 Scaling Functions and Splines

Central to multiresolution analysis is the design of a proper scaling function. It is therefore natural to wonder whether there exists a set of explicit mathematical requirements a scaling function has to satisfy that are equivalent to the axiomatic definition of multiresolution analysis. In other words, given a function $\varphi(t)$, how can we verify that such a function can be used to generate a multiresolution decomposition and a wavelet basis?

It is possible to show that $\varphi(t)$ is an admissible scaling function of $L_2(\mathbb{R})$ if and only if it satisfies the three following conditions [34, 30]:

1. Riesz basis criterion: There exists two constants $A > 0$ and $B < +\infty$ such that

$$A \leq \sum_{k\in\mathbb{Z}} |\hat{\varphi}(\omega + 2\pi k)|^2 \leq B \tag{3.7}$$

2. Two scale relation

$$\varphi(t) = \sqrt{2} \sum_{k\in\mathbb{Z}} g_0[k]\varphi(2t-k) \tag{3.8}$$

3. Partition of unity

$$\sum_{k\in\mathbb{Z}} \varphi(t-k) = 1. \tag{3.9}$$

Condition 1 ensures that $\varphi(t)$ generates a basis for the subspace

$$V_0 = \text{span}\{\varphi(t-k)\}_{k\in\mathbb{Z}}.$$

Notice that, in the case of orthonormal bases, $A = B = 1$.

The two scale relation guarantees that the subspaces $V_i = \text{span}\{\varphi_{i,k}(t)\}_{k\in\mathbb{Z}}$ generated by the scaled versions of $\varphi(t)$ with the usual notation $\varphi_{i,k}(t) = 2^{-i/2}\varphi(t/2^i - k)$, are embedded and form a multiresolution decomposition of $L_2(\mathbb{R})$. In other words, the two scale relation ensures

$$...V_2 \subset V_1 \subset V_0 \subset V_{-1}...$$

and is also equivalent to the scale invariance and shift invariance properties.

Finally, it is possible to show that the more technical partition of unity is equivalent to the upward and downward completeness. In other words, partition of unity ensures that the multiresolution decomposition $...V_2 \subset V_1 \subset V_0 \subset V_{-1}...$ is dense in $L_2(\mathbb{R})$ [34, 30].

There exists two main ways to design valid scaling functions: a direct way where one tries to design *paper and pencil* a function $\varphi(t)$ satisfying the above conditions or an indirect way where the scaling function is obtained by iterating the low-pass filter in a filter bank. This second approach has several advantages (e.g., more flexibility and simplicity), but also pitfalls (e.g., not all filters converge to a valid scaling function) and will be discussed in more detail in the next section.

A remarkable example of scaling functions obtained analytically is given by the family of B-splines [29]. A B-spline $\beta_N(t)$ of order N is obtained from the (N+1)-fold convolution of the box function $\beta_0(t)$ or

$$\beta_N(t) = \underbrace{\beta_0(t) * \beta_0(t)... * \beta_0(t)}_{N+1 \text{ times}} \qquad \text{with } \hat{\beta}_0(\omega) = \frac{1 - e^{-j\omega}}{j\omega}$$

where $\hat{\beta}(\omega)$ is the Fourier transform of $\beta(t)$. Notice that the so defined B-splines are not-centered in zero. In some cases, however, it is preferable to have symmetric functions and this is easily achieved by starting with a box function $\beta_0(t)$ centered in zero. The B-splines of order 0 to 3 are shown in Figure 3.1.

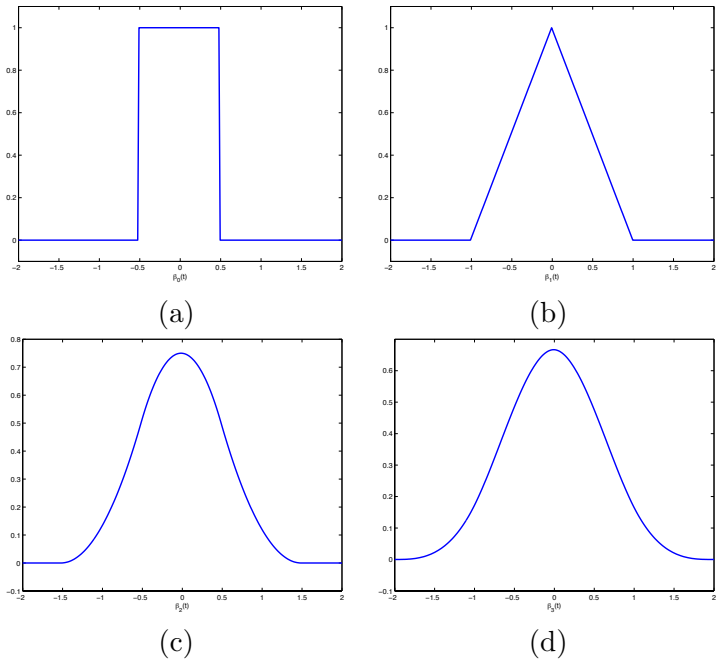


Figure 3.1: Centered B-spline of order 0 to 3.

It is easy to show that B-splines of any order satisfy the three conditions indicated before and, therefore, are valid scaling functions. Consider, for example, the zero order spline. One can see

graphically that $\beta_0(t)$ satisfies partition of unity and the two scale relation with $G_0(z) = (1+z^{-1})/\sqrt{2}$. Furthermore, the zero order spline is an orthogonal basis of V_0:

$$\sum_{n\in\mathbb{Z}} |\hat{\beta}_0(\omega + 2\pi n)|^2 = 1.$$

An easy way to prove the above result is by noticing that, if we denote with $a_\varphi[n]$ the sequence $a_\varphi[n] = \langle \varphi(t), \varphi(t-n)\rangle$, we have that

$$A_\varphi(e^{j\omega}) = \sum_{n\in\mathbb{Z}} |\hat{\varphi}(\omega + 2\pi n)|^2.$$

Therefore, since $a_{\beta_0}[n] = \delta_n$, it follows that $\sum_{n\in\mathbb{Z}} |\hat{\beta}_0(\omega + 2\pi n)|^2 = 1$. The zero order spline is, in fact, the Haar scaling function and, as we have seen before, generates the orthogonal Haar wavelet. Higher order splines are also valid scaling functions simply because convolution of two valid scaling functions leads to a new valid scaling function. Higher order splines, however, are not orthogonal. Because of biorthogonality, the design of the corresponding spline wavelets is not unique and is a bit more involved. An important family of such biorthogonal wavelets has been introduced by Cohen et al. [8]. It is not our aim to discuss in detail the multiresolution analysis in the biorthogonal case, however, let us state without proof the following result [8]:

Theorem 3 *Given two valid biorthogonal scaling functions $\varphi(t)$ and $\tilde{\varphi}(t)$ satisfing the following two scale relations*

$$\varphi(t) = \sqrt{2}\sum_{k\in\mathbb{Z}} g_0[k]\varphi(2t-k)$$

$$\tilde{\varphi}(t) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\tilde{\varphi}(2t-k).$$

There exist two biorthogonal wavelets ψ and $\tilde{\psi}$ such that

$$\psi(t) = \sqrt{2}\sum_{k\in\mathbb{Z}} (-1)^{k-1} h_0[1-k]\varphi(2t-k)$$

$$\tilde{\psi}(t) = \sqrt{2}\sum_{k\in\mathbb{Z}} (-1)^{k-1} g_0[1-k]\tilde{\varphi}(2t-k)$$

Let us highlight the basic ideas behind such a construction with a simple example.

Assume that $\varphi(t)$ is a linear spline. It follows that the two scale equation is satisfied when $G_0(z) = (\frac{1}{2}z^{-1} + 1 + \frac{1}{2}z)/\sqrt{2}$. Now, because of biorthogonality, we need to find the dual basis of $\varphi(t)$ first. Remember that the dual we are looking for is a valid scaling function as well. The biorthogonality relation says that

$$\langle \tilde{\varphi}(t), \varphi(t-n)\rangle = \delta_n.$$

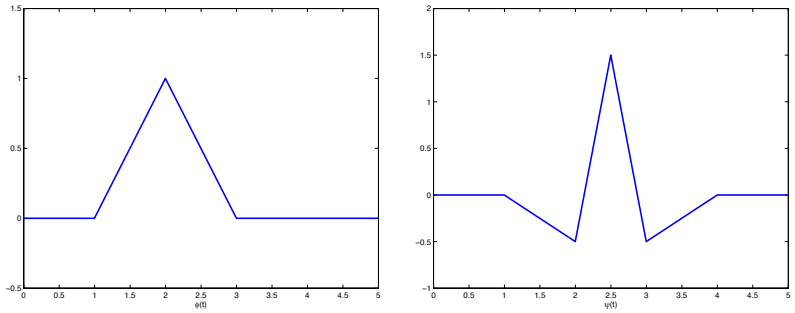
Since both $\varphi(t)$ and $\tilde{\varphi}(t)$ satisfy a two scale relation, it follows that

$$\langle \tilde{\varphi}(t), \varphi(t-n)\rangle = \langle h_0[k], g_0[k-2n]\rangle = \delta_n.$$

This is the biorthogonal relation for filter banks we saw in the previous chapter and this relation can be solved using spectral factorization. More precisely, the above relation is equivalent to the condition $P(z) + P(-z) = 2$ with $P(z) = H_0(z^{-1})G_0(z)$. Here $G_0(z)$ is known and has two zeros at $\omega = \pi$, the shortest $H_0(z)$ with the same number of zeros at π is then

$$H_0(z) = \frac{\sqrt{2}}{8}(1+z)(1+z^{-1})(-z+4-z^{-1}) = \frac{\sqrt{2}}{8}(z^{-1}+2+z)(-z+4-z^{-1}).$$

Given $H_0(z)$ the construction of the wavelet $\psi(t)$ is then straightforward. The scaling function $\varphi(t)$ and wavelet $\psi(t)$ for this example are shown in Fig 3.2.



(a) (b)

Figure 3.2: Spline scaling function and corresponding spline biorthogonal wavelet of compact support.

3.4 Wavelets from Iterated Filter Banks

We now show that filter banks can be used to obtain wavelet bases, assuming that the filters satisfy some regularity conditions.

Consider the tree-structured filter banks of the previous chapter. Let $g_0[n]$ and $g_1[n]$ denote low-pass and high-pass filters, respectively, and, for simplicity, assume that this is an orthogonal filter bank. The equivalent filters $g_0^{(i)}[n], g_1^{(i)}[n]$ after i steps of iteration are given by:

$$G_0^{(i)}(z) = \prod_{k=0}^{i-1} G_0(z^{2^k})$$

$$G_1^{(i)}(z) = G_1(z^{2^{i-1}}) \prod_{k=0}^{i-2} G_0(z^{2^k}).$$

Let us define a continuous-time function associated with $g_0^{(i)}[n]$ and $g_1^{(i)}[n]$ in the following way:

$$\begin{aligned} \varphi^{(i)}(t) &= 2^{i/2} g_0^{(i)}[n], \qquad n/2^i \le t < (n+1)/2^i \\ \psi^{(i)}(t) &= 2^{i/2} g_1^{(i)}[n], \qquad n/2^i \le t < (n+1)/2^i \end{aligned} \tag{3.10}$$

In the Fourier domain, using $M_0(\omega) = G_0(e^{j\omega})/\sqrt{2}$ and $M_1(\omega) = G_1(e^{j\omega})/\sqrt{2}$, we have that

$$\hat{\varphi}^{(i)}(\omega) = \Theta^{(i)}(\omega) \prod_{k=1}^{i} M_0\left(\frac{\omega}{2^k}\right)$$

where

$$\Theta^{(i)}(\omega) = e^{-j\omega/2^{i+1}} \frac{\sin(\omega/2^{i+1})}{\omega/2^{i+1}}$$

and

$$\hat{\psi}^{(i)}(\omega) = M_1\left(\frac{\omega}{2}\right) \Theta^{(i)}(\omega) \prod_{k=2}^{i} M_0\left(\frac{\omega}{2^k}\right).$$

Now assume that the limit for $i \to \infty$ of the two functions $\varphi^{(i)}(t), \psi^{(i)}(t)$ exists and is well defined. Let $\varphi(t), \psi(t)$ denote the two limit functions, that is

$$\begin{aligned} \varphi(t) &= \lim_{i\to\infty} \varphi^{(i)}(t), \\ \psi(t) &= \lim_{i\to\infty} \psi^{(i)}(t). \end{aligned} \tag{3.11}$$

In the Fourier domain

$$\hat{\varphi}(\omega) = \lim_{i\to\infty} \hat{\varphi}^{(i)}(\omega) = \prod_{k=1}^{\infty} M_0\left(\frac{\omega}{2^k}\right) \tag{3.12}$$

$$\hat{\psi}(\omega) = \lim_{i\to\infty} \hat{\psi}^{(i)}(\omega) = M_1\left(\frac{\omega}{2}\right) \prod_{k=2}^{\infty} M_0\left(\frac{\omega}{2^k}\right) \tag{3.13}$$

since $\Theta^{(i)}(\omega)$ tends to 1 as $i \to \infty$.

We now show that the so obtained functions are indeed a scaling function and a wavelet.

We first need to establish some necessary conditions for the limit to exist.

Theorem 4 *For the limit $\varphi(t) = \lim_{i\to\infty} \varphi^{(i)}(t)$ to exist, it is necessary that $G_0(e^{j\omega}) = \sqrt{2}$ for $\omega = 0$ and $G_0(e^{j\omega}) = 0$ for $\omega = \pi$.*

The conditions $G_0(1) = \sqrt{2}$ ensures that $M_0(\omega) = G_0(e^{j\omega})/\sqrt{2} = 1$ for $\omega = 0$. This normalization is necessary since otherwise the product $\prod_{k=1}^{i} M_0\left(\frac{\omega}{2^k}\right)$ would either blow up ($M_0(0) > 1$) or go to zero ($M_0(0) < 1$) which would mean that $\varphi(t)$ is not a lowpass function. We omit the proof of the necessity of a zero at π and this can be found in [36].

We can now demonstrate the following:

Theorem 5 *The functions $\varphi(t)$ and $\psi(t)$ given by (3.11) are respectively a valid scaling function and the corresponding wavelet. Moreover,*

1. *The scaling function is orthogonal to its translates:*

$$\langle \varphi(t-l), \varphi(t-k) \rangle = \delta_{k,l}.$$

2. *The scaling function is orthogonal to the wavelet and its translates:*

$$\langle \varphi(t), \psi(t-k) \rangle = 0.$$

3. *Wavelets are orthogonal across scales and with respect to shifts:*

$$\langle \psi(2^m t - n), \psi(2^{m'} t - n') \rangle = 2^{-m-m'} \delta_{m,m'} \delta_{n,n'}.$$

4. *The orthonormal set of functions $\{\psi_{m,n}(t)\}_{m,n\in\mathbb{Z}}$ is a basis of $L_2(\mathbb{R})$.*

Proof:

We start by showing that $\varphi(t)$ satisfies the three criteria of a valid scaling function. First, it satisfies the two-scale equation:

$$\varphi(t) = \sqrt{2} \sum_n g_0[n] \varphi(2t - n) \leftrightarrow \frac{1}{\sqrt{2}} G_0\left(e^{j\omega/2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right).$$

Indeed, by construction $\varphi(t)$ is given by (3.11) and the two-scale relation follows directly from the Fourier domain expression of the limit in (3.11):

$$\begin{aligned}\hat{\varphi}(\omega) &= \prod_{k=1}^{\infty} M_0\left(\tfrac{\omega}{2^k}\right) = M_0\left(\tfrac{\omega}{2}\right)\prod_{k=2}^{\infty} M_0\left(\tfrac{\omega}{2^k}\right)\\ &= M_0\left(\tfrac{\omega}{2}\right)\hat{\varphi}\left(\tfrac{\omega}{2}\right) = \tfrac{1}{\sqrt{2}}G_0\left(e^{j\omega/2}\right)\hat{\varphi}\left(\tfrac{\omega}{2}\right).\end{aligned}$$

In the same way, one can also show that $\psi(t)$ satisfies

$$\psi(t) = \sqrt{2}\sum_n g_1[n]\varphi(2t-n). \tag{3.14}$$

Moreover, $\varphi(t)$ satisfies the Riesz basis criterion. In fact, due to the two-scale relation, we know that the condition

$$\sum_l |\hat{\varphi}(\omega+2k\pi)|^2 = 1$$

is satisfied if and only if

$$|G_0(e^{j\omega})|^2 + |G_0(e^{j\omega+\pi})|^2 = 2. \tag{3.15}$$

Thus, since by construction $g_0[n]$ verifies (3.15), the scaling function satisfies the Riesz basis criterion and, moreover, since in this case the bounds are $A = B = 1$, $\varphi(t)$ is orthonormal (i.e., $\langle\varphi(t-l), \varphi(t-k)\rangle = \delta_{k,l}$).

Finally, by using Poisson summation formula, we can show that $\varphi(t)$ satisfies partition of unity. The Poisson summation formula says that:

$$\sum_{n=-\infty}^{\infty}\varphi(t-nT) = \frac{1}{T}\sum_{k=-\infty}^{\infty}\hat{\varphi}\left(\frac{2\pi k}{T}\right)e^{j2\pi kt/T}$$

and we want to verify that:

$$\sum_{n=-\infty}^{\infty}\varphi(t-n) = 1.$$

Thus by combining the two equations and for $T = 1$, we obtain the following:

$$\sum_{n=-\infty}^{\infty}\varphi(t-n) = \sum_{k=-\infty}^{\infty}\hat{\varphi}(2\pi k)e^{j2\pi kt} = 1.$$

The condition $\sum_{k=-\infty}^{\infty}\hat{\varphi}(2\pi k)e^{j2\pi kt} = 1$ is then clearly satisfied. Indeed, by using the infinite product formula in (3.12) and since $G(e^{j\omega}) = \sqrt{2}$ for $\omega = 0$ and $G(e^{j\omega}) = 0$ for $\omega = \pi$, we have that $\hat{\varphi}(2\pi k) = 1$ for $k = 0$ and $\hat{\varphi}(2\pi k) = 0$ otherwise.

Given a valid (orthonormal) scaling function the multi-resolution analysis (Theorem 2) showed that the orthonormal wavelet is obtained by choosing $\psi(t) = \sum_{n=-\infty}^{\infty}(-1)^n g_0[1-n]\varphi(2t-n)$. This is the case since we have just shown that (Eq. (3.14))

$$\psi(t) = \sqrt{2}\sum_n g_1[n]\varphi(2t-n)$$

and, by construction, $g_1[n] = (-1)^n g_0[1-n]$.

□

3.4.1 Regularity

We have seen that if the iterated filters converge to piecewise smooth functions then these function represent the scaling function and the corresponding wavelet.

We now want to understand what regularity conditions have these filters to satisfy in order to be guaranteed that they converge to a scaling function with some degrees of regularity and smoothness.

Given a filter $G_0(e^{j\omega})$, the limit function $\varphi(t)$ depends on the behaviour of the product

$$\prod_{k=1}^{i} M_0\left(\frac{\omega}{2^k}\right).$$

as $i \to \infty$.

We have already established some necessary conditions for the limit to exist. Namely, For the limit $\varphi(t) = \lim_{i\to\infty} \varphi^{(i)}(t)$ to exist, it is necessary that $G_0(e^{j\omega}) = \sqrt{2}$ for $\omega = 0$ and $G_0(e^{j\omega}) = 0$ for $\omega = \pi$.

Daubechies studied the regularity of iterated filter banks in detail and provided sufficient conditions for regularity.

Factor $M_0(\omega)$ as follows

$$M_0(\omega) = \left(\frac{1+e^{j\omega}}{2}\right)^N R(\omega).$$

Because of the necessary condition, we know that N must be at least equal to 1. Define B as

$$B = \sup_{\omega\in[0,2\pi]} |R(\omega)|$$

It then follows that

Theorem 6 ([36]) *If*

$$B < 2^{N-1}$$

then the limit $\lim_{i\to\infty} \varphi_i(t)$ *converges pointwise to a continuous function* $\varphi(t)$ *with Fourier transform*

$$\hat{\varphi}(\omega) = \prod_{k=1}^{\infty} M_0\left(\frac{\omega}{2^k}\right).$$

Moreover, if

$$B < 2^{N-1-n} \quad n = 1, 2, ...$$

then $\varphi(t)$ *is n-times continuously differentiable.*

We refer to [36, 10] for a proof of this theorem.

3.5 Properties of the wavelet series

In the previous sections, we have shown ways to construct orthonormal wavelet bases. We have seen that the construction of the mother wavelet $\psi(t)$ fundamentally depends on the scaling function $\varphi(t)$. We have also discussed sufficient and necessary conditions for $\varphi(t)$ to exist and to be regular in the case of iterated filters.

We will now analyze some general properties of the wavelet series and try to see their link with the design of the filters $g_0[n]$ and $g_1[n]$. As usual we consider orthonormal wavelets only. The same properties, however, applies with some minor modifications to the biorthogonal case.

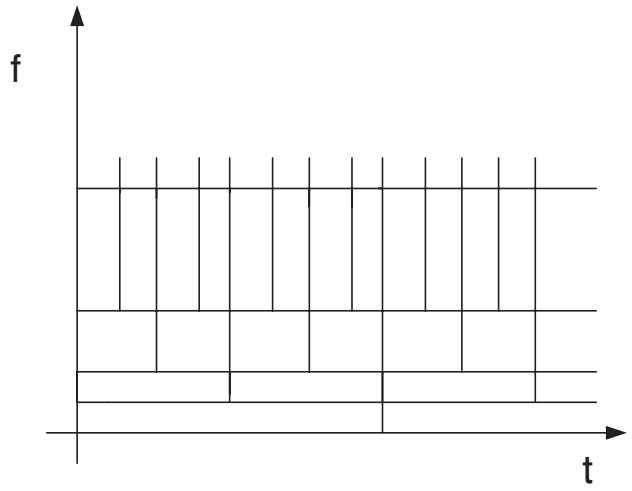


Figure 3.3: Tiling of the time-frequency plane.

We have shown in the previous section that any function $f(t) \in L_2(\mathbb{R})$ can be written as

$$f(t) = \sum_{m,n \in \mathbb{Z}} a_{m,n} \psi_{m,n}(t)$$

where

$$a_{m,n} = \langle f(t), \psi_{m,n}(t) \rangle.$$

This series satisfies several properties. For instance, notice that the sampling in time at scale m is done with a period 2^m leading to a dyadic sampling grid. In the frequency domain, we have the time-frequency tiling shown in Figure 3.3. The important element is that the wavelet is well localized in time **and** frequency. Therefore, it is able to characterize signals more efficiently than the Fourier transform or the Fourier series. For instance, assume we want to study the property of a signal around a point $t = t_0$. If the wavelet is of compact support, only a finite number of coefficients $a_{m,n}$ is influenced by the signal $f(t)$ at t_0. The region that includes all the wavelet coefficients influenced by $f(t_0)$ is called cone of influence. Notice that if the wavelet has been constructed from iterated FIR filter banks then we are sure that it is of compact support.

A key property of the wavelet transform is that of *vanishing moments*. We know that, in an iterated filter bank, $g_0[n]$ has at least one zero at $\omega = \pi$ and thus $g_1[n]$ has at least one zeros at $\omega = 0$. Since $\hat{\varphi}(0) = 1$ (from the normalization of $M_0(\omega)$), it follows that

$$\int_{-\infty}^{\infty} \psi(t) dt = \hat{\psi}(0) = \underbrace{\frac{G_1(1)}{\sqrt{2}}}_{=0} \hat{\varphi}(0) = 0.$$

In general, if $g_0[n]$ has a zero of order N at π then $\hat{\psi}(\omega)$ has N zeros at $\omega = 0$ and using the moment property of the Fourier transform we have that

$$\int_{-\infty}^{\infty} t^k \psi(t) dt = 0 \qquad k = 0, ..., 1, N-1. \tag{3.16}$$

This is a fundamental property of the wavelet function and it, basically, tells us that wavelets with N vanishing moments 'kills' polynomial of maximum degree $N-1$. However, Eq. (3.16) tells us a bit

more. Assume that a function $f(t)$ is well approximated by a polynomial $p_{t_0}(t)$ in the neighbourhood of $t = t_0$:

$$f(t) = p_{t_0}(t) + \epsilon(t) \quad \text{and} \quad |\epsilon(t)| \leq K|t - t_0|^\alpha$$

Then one can show that the wavelet coefficients $a_{m,n}$ around t_0 decay as $2^{m(\alpha+1/2)}$. This is what Mallat calls the wavelet zoom [22].

To be more precise, the regularity of a function is usually measured with the Lipschitz exponent [22].[2] We say that the restriction of $f(t)$ to $[a, b]$ is uniformly Lipschitz $\alpha \geq 0$ over $[a, b]$ if there exists $K > 0$ such that for all $\nu \in [a, b]$ there exists a polynomial $p_\nu(t)$ of degree $m = \lfloor \alpha \rfloor$ such that

$$\forall t \in (a, b), \quad |f(t) - p_\nu(t)| \leq K|t - \nu|^\alpha. \tag{3.17}$$

Now, assume that $f(t)$ is uniformly α-Lipshitz around t_0 and that $\psi(t)$ has at least $\lfloor \alpha \rfloor + 1$ vanishing moments. Moreover, assume for simplicity that $\psi(t)$ is of compact support C. The wavelet coefficients in the cone of influence of t_0 are then given by:

$$\begin{aligned}
\langle f, \psi_{m,n} \rangle &= \underbrace{\langle p_{t_0}(t), \psi_{m,n}(t) \rangle}_{=0} + \langle \epsilon(t), \psi_{m,n}(t) \rangle \\
&\overset{(a)}{\leq} K 2^{-m/2} \textstyle\int_{-\infty}^{\infty} |t - t_0|^\alpha \psi(2^{-m}t - n) dt \\
&= K 2^{m/2} \textstyle\int_{-\infty}^{\infty} |x 2^m + n 2^m - t_0|^\alpha \psi(x) dx \\
&\overset{(b)}{\leq} K C 2^{m(\alpha+1/2)} \underbrace{\textstyle\int_{-\infty}^{\infty} (|x| + |C|)^\alpha \psi(x) dx}_{=A} \\
&= C_1 2^{m(\alpha+1/2)}
\end{aligned}$$

where (a) follows from (3.17) and (b) from the fact that we are in the cone of influence of t_0 and therefore $|n2^m - t_0| \leq C2^m$.

In conclusion assume that a function $f(t)$ is piecewise smooth. That is, it is made of several smooth pieces and maybe some polynomial pieces. Then the wavelet coefficients $a_{m,n}$ are exactly zero around the polynomial regions, have a very fast decay around the smooth region and, finally, only the coefficients around discontinuities are large. This means that wavelets provide an extremely compact representation of piecewise smooth signals and this is the key reason behind the success of wavelets in compression.

3.6 Exercises

3.1 Consider the linear B-Spline given by

$$\varphi(x) = \begin{cases} 1 - |x|, & |x| < 1 \\ 0, & \text{otherwise.} \end{cases}$$

(a) Show that $\varphi(x)$ is a valid scaling function. That is, show that

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

i. it satisfies the two scale equation $\varphi(x/2) = \sqrt{(2)}\sum_{n\in\mathbb{Z}} g[n]\varphi(x-n)$,
ii. it satisfies the partition of unity $\sum_{n\in\mathbb{Z}} \varphi(x-n) = 1$,
iii. it satisfies the Riesz basis criterion $0 < A \leq \sum_{k\in\mathbb{Z}} |\Phi(\omega + 2\pi k)|^2 \leq B < \infty$.

(b) Now consider the derivative of $\varphi(x)$. Show that the derivative of $\varphi(x)$ is not a valid scaling function. (Hint: it is enough to show that at least one of the above criteria is not satisfied).

3.2 Consider the linear B-Spline given by

$$\varphi(t) = \begin{cases} 1-|t|, & |t|<1 \\ \\ 0, & \text{otherwise.} \end{cases}$$

We know that $\varphi(t)$ is a valid scaling function. However, the linear spline is not orthogonal. It is our aim to orthogonalize it.

(a) Compute the deterministic autocorrelation function

$$a[n] = \langle \varphi(t), \varphi(t-n) \rangle.$$

Denote $\hat{\varphi}(\omega)$ to be the Fourier transform of $\varphi(t)$ and $A(e^{j\omega})$ to be the discrete-time Fourier transform of $a[n]$. Show that the new function $\phi(t)$ with Fourier transform

$$\hat{\phi}(\omega) = \frac{\hat{\varphi}(\omega)}{\sqrt{A(e^{j\omega})}}$$

is an orthogonal basis of the subspace $V_0 =$ span $\{\phi(t-n)\}_{n\in\mathbb{Z}}$. (Hint: Show that the Riesz basis criterion $A \leq \sum_{n\in\mathbb{Z}} |\hat{\phi}(\omega + 2\pi n)|^2 \leq B$ is satisfied with $A = B = 1$).

(b) Using the Poisson sum formula:

$$\sum_{n=-\infty}^{\infty} f(t-n) = \sum_{k=-\infty}^{\infty} \hat{f}(2\pi k) e^{j2\pi kt},$$

show that $\phi(t)$ satisfies the partition of unity.

(c) Finally, find the z-domain expression of the filter $H_0(z)$ that leads to the two-scale equation:

$$\phi(t) = \sqrt{2}\sum_n h_0[n]\phi(2t-n).$$

(Hint: Use the fact that $\hat{\varphi}(\omega) = \frac{G_0(e^{j\omega/2})}{\sqrt{2}}\hat{\varphi}(\omega/2)$ and the fact that $\hat{\phi}(\omega) = \frac{\hat{\varphi}(\omega)}{\sqrt{A(e^{j\omega})}}$.)

(d) You now have a valid orthogonal scaling function, find the corresponding wavelet. That is, find the z-domain expression of the filter $H_1(z)$ such that: $\psi(t) = \sqrt{2}\sum_n h_1[n]\phi(2t-n)$.

3.3 Assume that two functions $\varphi_0(t)$ and $\varphi_1(t)$ are valid scaling functions. Show that the function $\varphi_2(t) = \varphi_0(t) * \varphi_1(t)$ given by the convolution of $\varphi_0(t)$ with $\varphi_1(t)$ satisfies:

(a) The partition of unity:

$$\sum_{n=-\infty}^{\infty} \varphi_2(t-n) = 1$$

(Hint: Use Poisson sum formula).

(b) The two-scale equation:

$$\varphi_2(t) = \sqrt{2}\sum_n g_2[n]\varphi_2(2t-n).$$

(c) Now assume that $\varphi_0(t) = \beta_0(t)$ and $\varphi_1(t) = \beta_1(t)$, where $\beta_0(t)$ is the box function with Fourier transform $\hat{\beta}_0(\omega) = \frac{1-e^{j\omega}}{j\omega}$ and $\beta_1(t) = \beta_0(t) * \beta_0(t)$. Thus, $\varphi_2(t) = \beta_0(t) * \beta_1(t)$. Find the exact expression of the filter $g_2[n]$ that leads to the two-scale equation:

$$\varphi_2(t) = \sqrt{2}\sum_n g_2[n]\varphi_2(2t-n).$$

3.4 Consider the wavelet series expansion of continuous-time signals with the Haar wavelet $\psi(t)$.

(a) Give the expansion coefficients

$$d_{m,n} = \langle \psi_{m,n}, f\rangle$$

for $f(t) = 1$, $t \in [0,1]$, and 0 otherwise (that is, $f(t)$ is the Haar scaling function).

(b) Verify that $\sum_m \sum_n |\langle \psi_{m,n}, f\rangle|^2 = \|f(t)\|^2$.

(c) Now consider $g(t) = f(t-2^{-i})$ where i is a positive integer. Give the range of scale over which expansion coefficients $d_{m,n} = \langle \psi_{m,n}, g\rangle$ are different from zero.

(d) Assume now that $f(t) = 1$ $t \in [0,2]$ and 0 otherwise. Can $f(t)$ be considered a valid scaling function?

3.5 Let $\varphi(t)$ and $\psi(t)$ be the Haar scaling and wavelet functions, respectively. Let V_j and W_j be the spaces generated by $\varphi_{j,n}(t) = \sqrt{2^{-j}}\varphi(2^{-j}t-n)$, $n \in \mathbb{Z}$ and $\psi_{j,n}(t) = \sqrt{2^{-j}}\psi(2^{-j}t-n)$, $n \in \mathbb{Z}$, respectively. Consider the function defined on $0 \le t < 1$ given by

$$f(t) = \begin{cases} -1 & 0 \le t < 1/4 \\ 4 & 1/4 \le t < 1/2 \\ 2 & 1/2 \le t < 3/4 \\ -3 & 3/4 \le t < 1. \end{cases}$$

(a) Express $f(t)$ in terms of the basis of V_{-2}. In other words, find the coefficients $c_{-2,n}$, $n \in \mathbb{Z}$ that leads to the decomposition $f(t) = \sum_{n\in\mathbb{Z}} c_{-2,n}\varphi_{-2,n}(t)$.

(b) Now, decompose $f(t)$ into its component parts W_{-1}, W_0, and V_0. In other words, find the coefficients $c_{0,n}$, $d_{-1,n}$ and $d_{0,n}$, $n \in \mathbb{Z}$ that leads to the following decomposition

$$f(t) = \sum_{n\in\mathbb{Z}} c_{0,n}\varphi_{0,n}(t) + \sum_{j=-1}^{0}\sum_{n\in\mathbb{Z}} d_{j,n}\psi_{j,n}(t).$$

(c) Sketch and dimension each of the decompositions of part (b).

(d) Verify the Parseval equality. That is, verify that:

$$\|f(t)\|^2 = \sum_n |c_{0,n}|^2 + \sum_{j=-1}^{0}\sum_n |d_{j,n}|^2.$$

3.6 Consider the two-channel filter bank of Figure 3.4. In 1984, Smith and Barnwell suggested that

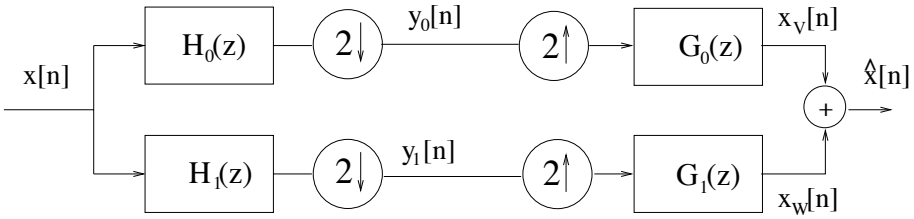


Figure 3.4: Two-channel filter bank.

the product filter $P(z) = H_0(z)G_0(z)$ should have the following form:

$$p[n] = \begin{cases} 1 & \text{for } n = 0 \\ \\ \frac{\sin(\pi n/2)}{\pi n} w[n] & \text{otherwise} \end{cases}$$

where $w[n]$ is a window function.

(a) Assume that $w[n]$ is the rectangular window:

$$w[n] = \begin{cases} 1 & \text{for } n = -M, -M+1, .., 0, 1, .., M \\ \\ 0 & \text{otherwise} \end{cases}$$

Show that the proposed $P(z)$ satisfies the half-band condition for any choice of M: $P(z) + P(-z) = 2$ for any M.

(b) Assume that $M = 1$. Factorize the resulting $P(z)$. Assign one zero of $P(z)$ to $G_0(z)$ and choose $G_0(z)$ to be minimum phase. Design the other filters in order to have perfect reconstruction biorthogonal filter banks.

(c) Now, consider the limit function

$$\hat{\varphi}(\omega) = \lim_{J\to\infty} \prod_{k=1}^{J} M_0\left(\frac{\omega}{2^k}\right),$$

where $M_0(\omega) = \frac{G_0(e^{j\omega})}{\sqrt{2}}$ and $G_0(z)$ is the filter you found in part (ii). What can you say about convergence of the above limit?

3.7 *Infinite products and Haar scaling function* Consider the following product:

$$p_i = \prod_{k=0}^{i} a^{b^k} \qquad |b| < 1.$$

(a) Show that $\lim_{i\to\infty} p_i = a^{1/(1-b)}$.

(b) Assume that $M_0(\omega) = G_0(e^{j\omega})/\sqrt{2}$ where $G_0(e^{j\omega}) = (1 + e^{-j\omega})/\sqrt{2}$ is the Haar low-pass filter. Show that

$$\lim_{i\to\infty} \prod_{k=1}^{i} M_0(\omega/2^k) = e^{-j\omega/2} \frac{\sin(\omega/2)}{\omega/2}.$$

Hint: Use the identity $\cos(\omega) = \sin(2\omega)/2\sin(\omega)$.

3.8 Consider the two-channel filter bank of Figure 3.4 .

(a) Take

$$P(z) = \frac{1}{16}(1+z)^2(1+z^{-1})^2(-z+4-z^{-1}),$$

where $P(z) = H_0(z)G_0(z)$. Compute a linear phase factorization of $P(z)$. That is, assume that $H_0(z) = \frac{1}{4\sqrt{2}}(1+z^{-1})^2(1+z)$. Given this choice of $H_0(z)$, define the other filters $H_1(z), G_0(z)$ and $G_1(z)$ in terms of their z-transforms.

(b) Now, consider the two limit functions

$$\hat{\varphi}(\omega) = \lim_{J\to\infty} \prod_{k=1}^{J} M_0\left(\frac{\omega}{2^k}\right),$$

$$\hat{\tilde{\varphi}}(\omega) = \lim_{J\to\infty} \prod_{k=1}^{J} \tilde{M}_0\left(\frac{\omega}{2^k}\right),$$

where $M_0(\omega) = \frac{G_0(e^{j\omega})}{\sqrt{2}}$, $\tilde{M}_0(\omega) = \frac{H_0(e^{j\omega})}{\sqrt{2}}$ and $\hat{\varphi}(\omega), \hat{\tilde{\varphi}}(\omega)$ are the Fourier transforms of $\varphi(t)$ and $\tilde{\varphi}(t)$ respectively. What can you say about convergence, continuity and differentiability of $\varphi(t)$ and $\tilde{\varphi}(t)$?

(c) Assume that the two limit functions $\varphi(t)$ and $\tilde{\varphi}(t)$ exist and that $\varphi(t)$ and $\tilde{\varphi}(t)$ are two valid scaling functions. Consider the two corresponding wavelets

$$\psi(t) = \sqrt{2}\sum_n h_1[n]\varphi(2t-n) \text{ and } \tilde{\psi}(t) = \sqrt{2}\sum_n g_1[n]\tilde{\varphi}(2t-n),$$

where $h_1[n]$ and $g_1[n]$ are the filters you found in (a).

i. State the number of vanishing moments of $\psi(t)$ and $\tilde{\psi}(t)$.

ii. Consider a function $f(t) \in L_2(\mathbb{R})$ and assume $f(t)$ is uniformly α-Lipschitz with $\alpha = 2.2$. You can write $f(t)$ either in terms of $\psi(t)$ or $\tilde{\psi}(t)$. That is:

$$f(t) = \sum_m \sum_n \langle f(t), \tilde{\psi}_{m,n}(t)\rangle \psi_{m,n}(t)$$

or

$$f(t) = \sum_m \sum_n \langle f(t), \psi_{m,n}(t)\rangle \tilde{\psi}_{m,n}(t),$$

with the usual assumption that $\psi_{m,n}(t) = 2^{-m/2}\psi(2^{-m}t - n)$. Which of these two representations leads to a faster decay of the wavelet coefficients across scales? Justify your answer numerically. In the light of these considerations, discuss whether or not it would be better to exchange the roles of the analysis and synthesis filters.

Chapter 4

Compression

Major applications of signal compression are with data storage and data transmission. In fact, compression deals with the problem of reducing the amount of memory necessary to store data or the problem of reducing the time necessary to transmit such data. Compression is normally measured in terms of bits per sample or in the case of images in terms of bits per pixel. For example, a grayscale image is typically represented using 8 bits per pixel. This means that there are 256 different gradation of gray that can be displayed. Using standard compression schemes we can store the same image using for example 0.5 bits per pixel.

There are two forms of compression: lossy and lossless compression. Lossless compression is reversible in that the original source can be reconstructed exactly from the compressed one, however, due to this constraint it is difficult to achieve high compression rates. Lossy compression is instead not reversible and only an approximated version of the original source can be obtained. The challenge then is to maximize the rate of compression for a fixed acceptable reconstruction error, or minimize the reconstruction error for a fixed target compression rate. Lossy compression is naturally more flexible than lossless compression and is therefore the form of compression used in practice. In fact, standard compression schemes combine lossy and lossless compression.

A typical compression system is shown in Figure 4.1. It has three main elements: a block implementing a (linear) transformation of the source, a quantizer and an entropy encoder. The basic principle behind the block implementing the linear transform is 'Divide et Impera'. More precisely, the aim of this transformation is to reduce the correlation or dependency of the samples in the source so that each transformed sample can be compressed independently. This reduces complexity while maintaining high compression efficiency. Typical transforms include wavelets, discrete cosine transform (DCT) or Karunhen-Loève transform (KLT). The quantizer performs the lossy compression by coarsening the values associated to each sample. Finally the entropy encoder performs lossless compression of the quantized samples in order to increase compression efficiency without incurring in further quality loss.

Lossless compression is discussed in the following section and quantization is discussed in Section 4.2. We then analyze the basic transform coding principles in Section 4.3. Finally, in Section 4.4, we put all the pieces together to discuss wavelet-based image compression algorithms.

4.1 Lossless Compression

As stated before, lossless compression is a form of reversible compression and as such can only be applied to discrete sources, namely, to sources that can take on only a finite number of different possible values.

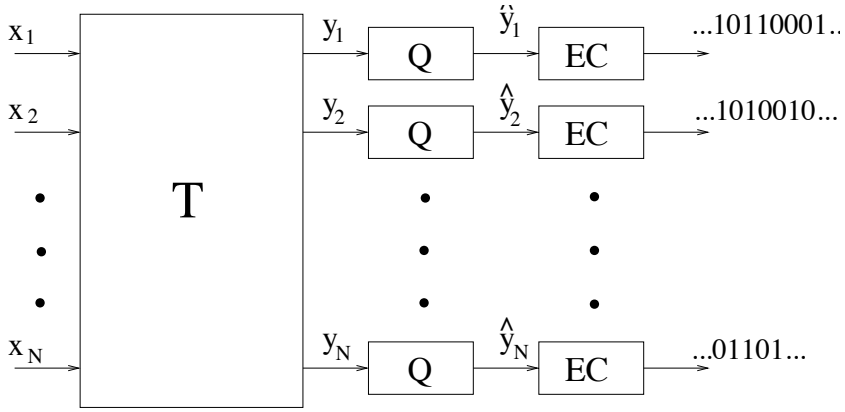


Figure 4.1: Transform Coding. A compression system based on linear transformations is made of three main elements: a transform (e.g., KLT, WT, DCT), a quantization and a lossless compression (entropy coding).

Consider a source X that can produce M different values a_i and denote with $p(a_i)$ the probability of occurrence of that particular symbol. The aim is to find a binary representation $b(a_i)$ of the the symbols a_i so that the expected length of the representation is minimized. That is, we want to minimize:

$$E(l(X)) = \sum_{i=0}^{M-1} p(a_i) l_i, \tag{4.1}$$

where l_i is the length of the binary codeword associated to a_i. The binary representation has to be invertible. However, we also need to impose a second constraint: we require the representation to be *uniquely decodable*. This means that no codeword is allowed to be the prefix of another codeword. In this way, we can attach different codewords one after the other and yet be able to retrieve the original sequence of symbols with no need of any 'punctuation' between codewords.

The lower bound on the expected length (4.1) of a prefix code is given by the entropy $H(X)$ of the source. The entropy is defined as follows:

$$H(X) = -\sum_{i=0}^{M-1} p(a_i) \log_2(p(a_i)) \quad \text{bits per symbol.}$$

The performance of prefix codes is bounded by the entropy because prefix codes have to satisfy the *Kraft inequality* that states the following:

Theorem 7 (Kraft Inequality [9]) *For any prefix code the codeword lengths l_i i=0,1,...,M-1 must satisfy the inequality*

$$\sum_{i=0}^{M-1} 2^{-l_i} \leq 1. \tag{4.2}$$

Conversely, given a set of codeword lengths that satisfy the inequality, there exists a prefix code with these word lengths.

For a proof of the above inequality we refer to [9]. A prefix code is optimal when equality in (4.2) is achieved. Now, consider choosing a code where the codeword length l_i satisfies

$$l_i = -\log_2 p(a_i), \tag{4.3}$$

then this code achieves equality in (4.2) and its expected length is precisely $H(X)$. In practice since the lengths l_i must be positive integers, the expected length of an optimal lossless or entropy code is normally bigger than the entropy of the source and equality is achieved only in special cases. Also notice that equation (4.3) indicates that an optimal code assigns short codewords to likely symbols and long codewords to less likely symbols.

Practical lossless coders that can come very close to the entropy of the source include Huffman codes and arithmetic codes. Modern lossless coders, like arithmetic coders, can also estimate *on line* the statistics of the source and deal efficiently also with sources with memory, that is, with sources that produce symbols that are correlated.

We conclude this overview with a simple canonical example:

Example 5 *Consider a source that produces four possible symbols:* $\{a, b, c, d\}$ *and assume that* $p(a) = 0.5$, $p(b) = 0.25$ *and* $p(c) = p(d) = 0.125$. *We may represent each symbol with two bits leading to a uniquely decodable code of expected length* 2. *A more efficient prefix code would represent 'a' with a short codeword and 'c' and 'd' with a long one. For example the code that represent 'a' with '0', 'b' with '*10*', 'c' with '*110*' and 'd' with '*111*', is uniquely decodable, has expected length equal to the entropy of the source and achieves the Kraft bound.*

4.2 Quantization

The amplitude values of a discrete-time signal are normally real numbers. The process of mapping these continuous values into a finite alphabet is called *quantization*. Usually, each sample is quantized individually and we call this form of quantization *scalar quantization*. A more sophisticated form of quantization, called *vector quantization*, operates on groups of samples. For a detailed treatment of vector quantization, we refer to [16].

An example of a scalar quantizer is depicted in Figure 4.2. In this case, the scalar quantizer is *uniform* since the input is divided into intervals of equal size. The interval I_i is given by $I_i = (x_{i-1}, x_i] = (i - 1/2, i + 1/2]$ and the output value in that interval is $y_i = i$. This means that if we have an input value $x = 1.736$ the output of the quantizer is $y_2 = 2$. Quantization is clearly a form of lossy compression and can also be applied to discrete-amplitude sources. In this second case, quantization is the mapping of a large alphabet of size M into a smaller alphabet of size N. If the quantizer outputs N different symbols y_i, we need $R = \lceil \log_2 N \rceil$ bits to represent each symbol using a fixed length binary code. We say that the quantizer has a rate R.

The performance of a quantizer $q(x)$ is normally measured in terms of the Mean Square Error (MSE):

$$D = E[(x - q(x))^2] = \sum_{i=0}^{N-1} \int_{I_i} (x - y_i)^2 f_X(x) dx,$$

where $f_X(x)$ is the probability density function of X.

For example, assume that X is uniformly distributed in $[0, A]$ and that this range is divided into N intervals of equal size $\Delta = \frac{A}{N}$, also assume that $y_i = (x_i + x_{i-i})/2$, the MSE is then equal to

$$D = \frac{\Delta^2}{12} = \frac{A^2}{12N^2}.$$

If we further assume for simplicity that N is a multiple of 2 so that $R = \log_2 N$, we obtain

$$D(R) = \frac{A^2}{12} \cdot 2^{-2R}. \tag{4.4}$$

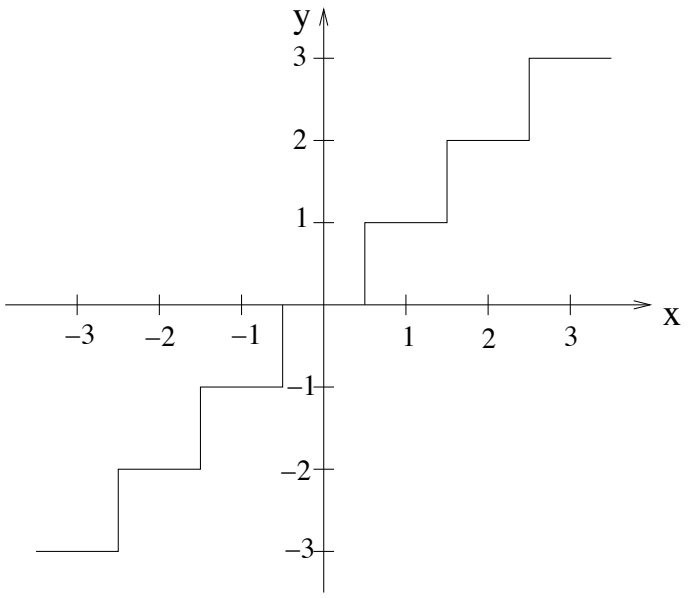


Figure 4.2: A uniform scalar quantizer.

Equation (4.4) gives a first example of the trade-off between the rate R of a quantizer and the distortion associated to that quantizer. What we have just computed is, in fact, an *operational* rate-distortion function. The issue then is to design a quantizer that minimizes D for a given R or minimize R for a target D. It is also important to notice that the quantization might be followed by an entropy coder to compress losslessly the N output symbols. In other words, rather than associate fixed-length codewords to each symbol, we may study the statistics of the output symbols and associate short codewords to the most probable symbols and longer codewords to the less probable ones. What is not clear at this stage is the interplay between quantization and lossless compression. Namely, should the design of the quantizer be influenced by the fact that the output symbols are entropy encoded? Moreover, there is a second issue that we would like to address: In the case of discrete sources and lossless compression we know from the previous section that the entropy provides a lower bound on the performance of a lossless coder. We now would like to have a similar bound for the lossy case. Namely, can we find a function that tell us what is the best possible achievable trade-off between rate and distortion?

While the optimal design of a quantizer is beyond the scope of this chapter and we refer to [16] for more details, we can still provide some answers to the above questions. First of all, rate distortion theory provides the desired lower bounds. The fundamental result in rate-distortion theory states the following:

Theorem 8 ([9]) *The information rate distortion function $R(D)$ for a discrete source X with distortion measure $d(x,\hat{x})$ is defined as*

$$R(D) = \min_{p(\hat{x}|x):\sum_{(x,\hat{x})} p(x)p(\hat{x}|x)d(x,\hat{x})\leq D} I(X,\hat{X})$$

where $I(X;\hat{X}) = H(X) - H(X/\hat{X})$ is the mutual information and the minimization is over all conditional distribution $p(\hat{x}|x)$ for which the joint distribution $p(x,\hat{x}) = p(x)p(\hat{x}|x)$ satisfies the expected distortion constraint.

A similar result applies to continuous sources. Unfortunately, the rate-distortion function is known only for a few cases and in most cases one must be satisfied by bounds. A remarkable exception is given by the Gaussian source and MSE. Assume that a source emits independent and identically distributed (i.i.d.) Gaussian random variables with variance σ^2. The distortion-rate of the source subject to MSE is:

$$D(R) = \sigma^2 2^{-2R}. \tag{4.5}$$

Given (4.5), we now want to understand how close a scalar quantizer can come to this bound and how the optimal quantizer should be designed. For high-rate quantizers (i.e., for large R) we actually have precise answers [17]. If we use a fixed-length scalar quantizer, that is, if the quantization is **not** followed by an entropy encoder, the optimal quantizer for Gaussian sources is non-uniform and achieves:

$$D(R) = \frac{1}{12} 6\pi\sqrt{3}\sigma^2 2^{-2R}. \tag{4.6}$$

Compared to the distortion-rate bound (4.5), the distortion of this non-uniform quantizer is higher by a factor ~ 4.35dB or the rate loss is ~ 0.72bits per symbol. In other words, this quantizer needs to use 0.72 extra bits per symbol to achieve the same distortion of the best possible lossy compression scheme. We can improve such performance by entropy encoding the output symbols. The interesting result is that in this case the optimal quantizer is **uniform**. This means that we do not have to worry too much about the design of a scalar quantizer, since uniform is optimal when followed by an entropy encoder. Such variable-length quantizer achieves [17]:

$$D(R) = \frac{\pi e}{6}\sigma^2 2^{-2R}. \tag{4.7}$$

Compared to the distortion-rate bound (4.5), the distortion of this variable-length uniform quantizer is higher by a factor ~ 1.53dB and the redundancy is ~ 0.255 bits per symbol. This is quite an improvement when compared to (4.6), yet we still do not hit the bound. Why? The answer is simple, even in the case of i.i.d. sources, quantizing each sample individually is sub-optimal. This is because scalar quantizers partition the space into (hyper)-cubic cells, whereas a vector quantizer would partition the space more efficiently. For example, consider two independent variables. If we scalar quantize them we are dividing the 2-D space into rectangular cells. However, it is well known that hexagonal cells achieves a better MSE for the same rate. A quantizer that compress the two variable jointly (i.e., a vector quantizer) would achieve this hexagonal tiling. This form of gain is know as *packing gain*. A vector quantizer, however, needs to operate on very long blocks of samples to achieve the rate-distortion bound. This is not feasible in practice, for this reason vector quantization is very rarely used in practical applications. If we stick with scalar quantization, Eq. (4.7) represents the best possible achievable performance at high rate for i.i.d. Gaussian sources. What is also interesting is that, at high rate, the above still applies even if the source is not Gaussian. More precisely, it is still optimal to use a uniform quantizer followed by an entropy encoder and the operational distortion-rate function is

$$D(R) = C\sigma^2 2^{-2R}, \tag{4.8}$$

where the constant C depends on the probability density function (p.d.f.) of the source.

In summary, at high-bit rates, the best thing to do is to use a uniform quantizer followed by an entropy encoder, this would give us a fairly simple lossy compression scheme that gets very close to the best possible achievable performance. Also at high rates any quantizer has a $D(R)$ function of the form $D(R) \sim C\sigma^2 2^{-2R}$. While this analysis is not valid at low-bit rates, this guiding principles are often used in practice where uniform or 'almost' uniform quantizers followed by entropy encoders are normally employed.

The last open issue, now, is to understand how we should operate on sources with memory, namely, on sources that are not i.i.d. Transform coding is the answer to this question as we shall discover in the next section.

4.3 Transform Coding

We start this section by reviewing some classical results on compression of correlated Gaussian sources and discuss the bit allocation problem. We then depart from the Gaussian assumption and show why wavelets are the right transform for non-Gaussian signals like images.

4.3.1 The Karhunen-Loève Transform and the bit allocation problem

We now study the problem of compressing sources with memory. Assume that a source produces blocks of statistically dependent samples and assume that each block has N elements. We denote this block as a column vector x. Clearly, it would be inefficient to scalar quantize each element of the block independently since we would not exploit the dependency of the samples. An alternative is to compress them jointly using a vector quantizer, but as stated before, this is not an option since the complexity of a vector quantizer increases exponentially with the size of the block. Instead we want to devise a simple but compression-effective strategy based on linear transformations. The transform coder operates as shown in 4.1. We denote with T the transform applied to the vector x leading to the transformed samples $y = Tx$. Each component $y[k]$ of the transformed vector y is then scalar quantized and entropy encoded. The reconstructed block is $\hat{x} = T^{-1}\hat{y}$, where $\hat{y}$ is the quantized version of y. The issue then is to devise the transform T that minimizes the MSE $D = E(\|x - \hat{x}\|^2)$ for a given rate R.

Recall that a transform is a way to represent x using a different basis. As discussed in Chapter 1, if T is orthogonal[1] and we denote the rows of T as φ_k^T, k=0,1,...,N-1 then we have that

$$x = \sum_{k=0}^{N-1} y[k]\varphi_k$$

and since T is orthogonal $D = E(\|x - \hat{x}\|^2) = E(\|y - \hat{y}\|^2)$.

We start our analysis by assuming that x is a jointly Gaussian zero-mean vector with covariance matrix $R_x = E(xx^T)$. The Karhunen-Loève transform (KLT) is the transform that diagonalizes R_x leading to $R_y = E(yy^T) = TR_xT^T = \Lambda$. Thus the coefficients $y = Tx$ are uncorrelated and in the case of Gaussian sources the $y[k]$'s are independent Gaussian variables with variance λ_k^2. Here λ_k^2 is the k-th diagonal element of R_y. Because of independence, it makes sense to compress each component $y[k]$ independently. Using Eq. (4.7), we have that at high-rate the k-th component of y contributes a distortion

$$D_k(R_k) = \frac{\pi e}{6}\lambda_k^2 2^{-2R_k}$$

where R_k is the rate allocated to this component. The overall distortion is therefore:

$$D(R) = E(\|x - \hat{x}\|^2) = E(\|y - \hat{y}\|^2) = \frac{1}{N}\sum_{k=0}^{N-1} D_k, \tag{4.9}$$

with $R = \sum_k R_k$.

[1] From now on we consider only orthonormal transforms.

We now want to minimize (4.9) subject to $R = \sum_k R_k$. Fundamentally, the issue is to decide how allocate the rate R amongst the different components. This is a typical constrained minimization problem that can be solved using Lagrange multipliers. In the high bit-rate regime, one can prove that optimal bit allocation leads to equal distortion for each component: $D_k = D$ for $k = 0, 1, .., N-1$. In other words more bits are given to the variables $y[k]$ with bigger variances. The expression for the rate R_k is:

$$R_k = \frac{R}{N} + \log_2 \lambda_k - \frac{1}{N} \sum_{i=0}^{N-1} \log_2 \lambda_i.$$

In practice R_k is positive integer, so the above equation is only approximately satisfied, in particular when R_k is negative, it is set to zero. This means that no rate is allocated to that component which is in fact discarded. The behaviour of the overall compression system is now clear: apply the KLT to x, discard the coefficient of y with smallest variance, scalar quantize the remaining coefficients and allocate bits in a way that each component contributes equal distortion.

One can again measure the performance of this transform coder and prove that it is optimal, namely, there is no better transform than the KLT and since the quantizers are followed by entropy encoders, uniform is good. This transform coder can only be outperformed by an ideal vector quantizer (VQ), however, the maximum performance gain that one can achieve with the VQ is a mere 1.53 dB. Therefore, for Gaussian sources, transform coding represent the right strategy.

If the input source is stationary but not Gaussian, we can still use the same scheme. In this new situation the KLT decorrelates the components of x, but does not provide independent components. The bit allocation analysis is still valid since we can use the operational $D(R)$ of (4.8) which is similar to (4.7). While this approach tends to give good results we are not guaranteed anymore that it is optimal.

When sources are not stationary, the problem becomes even more complicated and KLT is normally not optimal. This is where wavelets are going to rescue us.

4.3.2 Linear and non-linear approximation

We start this section by considering an apparently unrelated problem. Let f be a function in a Hilbert space $\mathbb{H}$. We have different bases of $\mathbb{H}$ and need to decide which of these bases is the best to represent f. One possible way to compare such bases is by analyzing their approximation capabilities. More precisely, given the function $f \in \mathbb{H}$ and an orthogonal basis $\{g_n\}_{n\in\mathbb{N}}$, we want to approximate this function using only M elements of $\{g_n\}_{n\in\mathbb{N}}$. We assume that the choice of these M elements is fixed a-priori and is never changed, namely, the choice of such elements is *independent* of the f we are trying to approximate. This form of approximation is therefore called *linear approximation* (LA). Consider choosing the first M elements of this basis. This leads to the following linear approximation of f:

$$f_M = \sum_{n=0}^{M-1} \langle f, g_n \rangle g_n.$$

Since the basis is orthogonal, the MSE of this approximation is:

$$\epsilon_M = \|f - f_M\|^2 = \sum_{n=M}^{\infty} |\langle f, g_n \rangle|^2. \tag{4.10}$$

Therefore, given a class of signals and a choice of possible bases, the best basis in this context is the one that leads to the smallest MSE as given in (4.10). Interestingly, one can show that if the signals

we are considering are realizations of a jointly Gaussian process which generates jointly Gaussian vectors, the best basis for linear approximation of these signals is again the KLT.

Signals can be better approximated by relaxing the hypothesis that the choice of the M elements of the basis is fixed. An alternative and, of course, more efficient approach is to observe the signal f first and then, given f, choose the M elements that would lead to the best approximation of that specific f. Namely, the M elements of the basis are chosen adaptively and, since the choice depend on f, this form of approximation is **non-linear**. If we denote with I_M the index of the elements of the basis that are used to represent f, we have that

$$\hat{f}_M = \sum_{n \in I_M} \langle f, g_n \rangle g_n$$

and

$$\hat{\epsilon}_M = \|f - \hat{f}_M\|^2 = \sum_{n \notin I_M} |\langle f, g_n \rangle|^2.$$

From the above equations, we realize that, in order to minimize $\hat{\epsilon}_M$, the indices in I_M must correspond to the M elements of $\{g_n\}_{n \in \mathbb{N}}$ with the largest $|\langle f, g_n \rangle|$. The strategy is therefore clear: observe f, compute the inner products $\langle f, g_n \rangle$, keep only those with the largest magnitude. It is also evident that $\hat{\epsilon}_M \leq \epsilon_M$.

Now assume we want to compare the wavelet representation against the Fourier representation and assume that the signals we want to approximate are defined over an interval and are globally smooth in that interval. More specifically, they are α-Lipschitz in the interval of interest. In this case, it turns out that a wavelet basis or a Fourier basis have the same approximation capabilities and that linear and non-linear approximations provide the same performance. The situation is more interesting when the signals are piecewise smooth. In that case non-linear approximation (NLA) using wavelets leads to the best performance. In particular, if the signals are made of α-smooth pieces then, for large M, the wavelet representation leads to an error decaying as $\hat{\epsilon}_M \sim M^{-2\alpha-1}$. A Fourier basis instead has an error that decays as $\hat{\epsilon}_M \sim M^{-1}$.

To conclude, wavelets have an advantage over other bases when the signals of interest are not stationary and non-linear approximation is involved. This is a consequence of the properties of the wavelet transform as discussed in the previous chapter. In particular, when a signal has a discontinuity, the effect of this discontinuity is limited in the wavelet case, since wavelets are of compact support. However, this discontinuity has an effect on all the elements of the Fourier series since the elements of this basis have infinite support. Real world signals are non-stationary therefore wavelets have an advantage with respect to other transforms.

Linear and non-linear approximation can be seen as forms of compression where one tries to represent a signal with a small number of basis elements rather than a large or infinite number. In fact, the transform coding strategy based on the KLT presented in the previous section include a form of linear approximation since the element of y with the smallest eigenvalues are always discarded. Compression, however, is a bit more involved since any form of information used in the representation has to be properly encoded. To be more precise, when performing non-linear approximation, we choose adaptively a subset of elements of the basis to represent the signal. Namely, the index set I_M is not fixed. When performing compression and therefore reconstruction, the set of indices contained in I_M needs to be stored or transmitted together with the quantized versions of the chosen inner products. Thus bits need to be used to describe I_M. In the case of linear approximation, this problem does not exist since the chosen elements of the basis are always the same. Therefore, when it comes to compression NLA is not necessarily the best solution.

A canonical example that highlights this problem is that of correlated Gaussian vectors. In this particular case, as we saw in the previous section, linear approximation using KLT is optimal. Here

we have a linear approximation since the elements of y which are kept in the transform coding process are always the same, this choice depends on the statistics of the source and does not depend on the particular realization x that we are observing. In other words, in the Gaussian case, the cost of indexing the chosen coefficients out-weights the advantage of an adaptive representation of x. When it comes to compression, a linear approximation of x is on average more efficient.

When signals are non stationary the advantage of the non-linear approximation can be vast and therefore NLA might be better than LA even when performing compression. NLA, however, needs to be coupled with an efficient indexing strategy. Wavelet-based compression algorithms had an impact in image compression only when an efficient method to encode the wavelet coefficients was devised. The first successful method based on zero-trees [25] is described in the next section.

4.4 Wavelet-based Image Compression

This section is mostly based on the paper [25] and the book [36].

Wavelet-based image compression algorithms are based on the transform coding principles highlighted in the previous sections, but also perform some forms of non-linear approximation. All that in an extremely efficient way so that no bits are wasted.

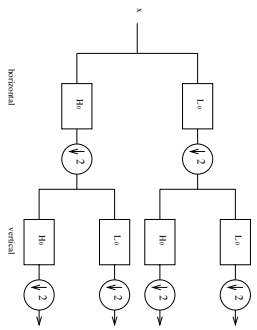


Figure 4.3: Separable wavelet transform: block diagram

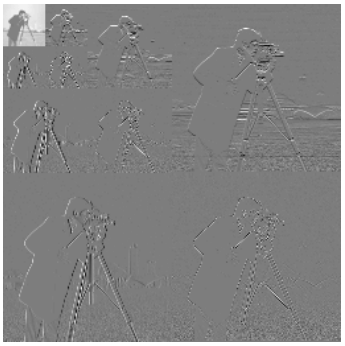


Figure 4.4: Three iterations of the 2-D wavelet transform

Images are two-dimensional (2-D) signals, therefore, the 1-D wavelet decomposition we have been considering so far needs to be extended to 2-D. The standard separable 2-D wavelet transform simply consists of two 1-D wavelet transforms along the horizontal and vertical directions. A block diagram of the standard 2-D wavelet transform is illustrated in Figure 4.3. The filter L_0 is usually a low pass filter while H_0 is high pass. Subsampling is first performed on columns and then on rows. The process is usually iterated on the 'low-low' pass version of the image. The resulting transformed image after three iterations is shown in Figure 4.4.

Now consider the wavelet transformed image shown in Figure 4.4. First of all, we can notice that most of the wavelet coefficients are small or close to zero, this is a consequence of the fact that images are, to some extent, piecewise smooth. By observing the image, however, it is also clear that there are dependencies across the scales. Namely, if a coefficients is large in the low-pass component, it is likely to be large in the bandpass and high pass components as well. Likewise, a small coefficients in the coarse scale is likely to lead to small coefficients in the same spatial location at finer scales. This observation lead Shapiro [25] to device an efficient algorithm for image compression. The algorithm is based on the idea that the dependency among coefficients follows a tree structure as the one depicted in Figure 4.5. According to this structure, a small coefficients in a subband is likely to

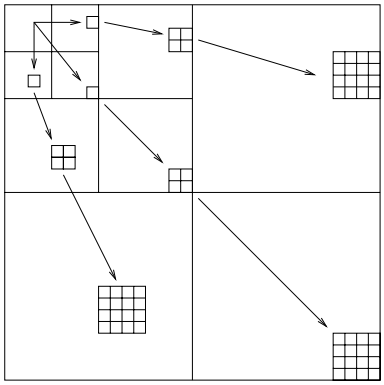


Figure 4.5: Zero-tree structure.

lead to four small coefficients in the same region at the next subband, while the following subband will have 16 small coefficients and so on. In the tree structure, the wavelet coefficient in the coarse subband is the parent, the coefficients in the next subband of the same tree are the children, the other coefficients in the tree are called descendants. A tree made of all *insignificant* symbols is encoded as a *zero tree*. This structure therefore allows to discard at once many small coefficients. Indeed, this zero-tree structure can be seen as an efficient way to index the basis elements that are discarded in a NLA. Namely, with zero tree, rather than indexing the large coefficients, one indexes those that are discarded. The zero-trees are combined with a bit plane coding leading to an efficient algorithm called *embedded zero-tree wavelet (EZW)* algorithm. The algorithm operates as follows:

After the wavelet transformation, the algorithm searches for the wavelet coefficient with the largest amplitude, denote this coefficient as y_{max}. A threshold T_0 is then defined as $T_0 = 2^n$ where $n = \lceil \log_2(|y_{max}|) \rceil - 1$. All wavelet coefficients with amplitude smaller than the threshold are called *insignificant*. When a coefficient in the coarse subband is insignificant, the algorithm looks for all the other coefficients in the corresponding tree and if they are all insignificant, the parent is encoded as a zero-tree root (ZTR). This allows the algorithm to discard all the coefficients in that tree at once. When the parent is insignificant, but some children or descendants are not, then the parent is encoded as an isolated zero (IZ). A significant parent is encoded as POS (positive significant) or NEG (negative significant). The coefficients are scanned in a zig-zag manner but also in a way that no children are encoded before the parents.

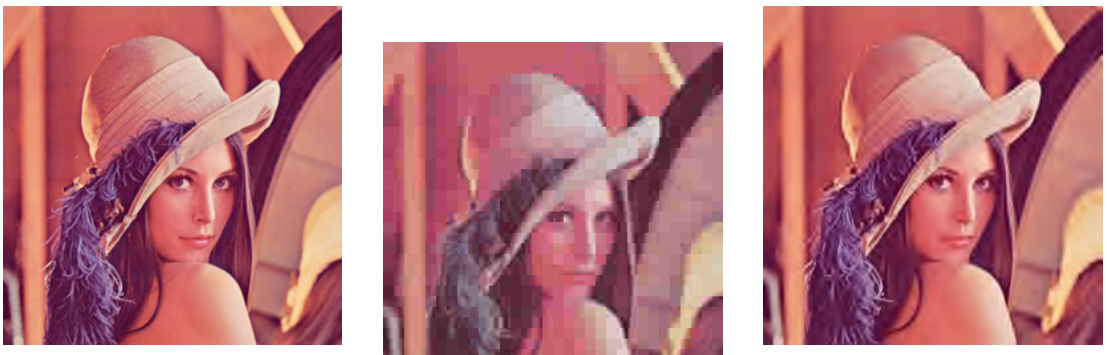
Once all the wavelet coefficients have been scanned and the appropriate symbol *ZTR, IZ, POS* or *NEG* has been assigned, the algorithm performs a successive quantization of the coefficients. This entails keeping two lists of coefficients: the *dominant list* that contains the coordinates of all the coefficients that have not yet been found to be significant and the *subordinate list* that contains the magnitudes of the coefficients that have been found to be significant. The threshold is then halved leading to $T_1 = T_0/2$ and the magnitude of the coefficients in the subordinate list is refined by one bit. More precisely, denote with y_i one of the significant coefficients, by construction $|y_i| \geq T_0$, then the symbol '0' is transmitted if $T_0 \leq |y_i| < T_0 + T_1$ and a '1' is transmitted if $|y_i| \geq T_0 + T_1$. The process is then iterated. The coefficient in the *dominant list* are scanned again to assign them the right symbol *ZTR, IZ, POS* or *NEG*. However, the threshold is now T_1 and the coefficients that had been found to be significant in the previous stage are set to zero so that they not preclude the possibility of finding a zero-tree. The threshold is than halved again and the process is iterated until a certain bit budget or target distortion is met. Finally, the symbols are losslessly compressed using an arithmetic coding.

63	–34	49	10	7	13	–12	7
–31	23	14	–13	3	4	6	–1
15	14	3	–12	5	–7	3	9
–9	–7	–14	8	4	–2	3	2
–5	9	–1	47	4	6	–2	2
3	0	–3	2	3	–2	0	4
2	–3	6	–4	3	6	3	6
5	11	5	6	0	3	–4	4

Figure 4.6: An example of a three-level wavelet decomposition of an 8×8 image.

To further clarify the algorithm, we consider a simple example given in [25] (see also [36]). We have an 8×8 image whose three-level wavelet transform is shown in Figure 4.6. The initial threshold is $T_0 = 32$ since the largest coefficient is 63. The algorithm scans the coefficients in a zig-zag order from the coarse to the finest scale. Thus, the first coefficient analyzed is 63 which is larger than T_0, so gets a *POS*, the following coefficient is -34 and gets a *NEG*. We then observe -31 whose absolute value is smaller than the threshold. However, by observing its tree, we realize that not all the coefficients are below the threshold due to the value 47, for this reason -31 gets an *IZ*. We continue with 23 and observe that it is a root of a zero-tree so gets *ZTR*. The process is then continued, but the coefficients that belong to a zero-tree are skipped. The final result is shown in Table 4.1.

We are now ready to quantize the selected coefficients. The new threshold is $T_1 = 16$. The first significant value 63 obtains a '1' since $63 > T_0 + T_1$ and is reconstructed to 56. The second one, -34, get a '0' and is reconstructed to -40, 49 gets a '1' and is reconstructed to 56, finally, 47 gets a '0' and is reconstructed to 40. The process can then be iterated by scanning again the coefficients in the dominant list with the new threshold T_1 and so on. The new image compression standard



Original Lena Image (256×256 pixels) JPEG (Compression Ratio 43:1) JPEG2000 (Compression Ratio 43:1)

Figure 4.7: JPEG2000 vs JPEG. Note: images courtesy of dspworx.com

JPEG2000 is based on the wavelet transform and employs a compression strategy similar in spirit

Coefficient	Symbol	Reconstruction
63	POS	48
-34	NEG	-48
-31	IZ	0
23	ZTR	0
49	POS	48
10	ZTR	0
14	ZTR	0
-13	ZTR	0
15	ZTR	0
14	IZ	0
-9	ZTR	0
-7	ZTR	0
7	Z	0
13	Z	0
3	Z	0
4	Z	0
-1	Z	0
47	POS	48
-3	Z	0
-2	Z	0

Table 4.1: The first step in the embedded zero-tree algorithm for the image shown in Figure 4.6.

to embedded zero-trees. The old compression standard JPEG was instead based on the DCT. The new standard has many nice features but most importantly outperforms in a rate-distortion sense the old standard. Following the analysis of this chapter we now understand that this is due to the fact that the wavelet basis provides a better NLA of images -which are piecewise smooth - and this coupled with an efficient strategy to index and quantize the largest coefficient has lead to a more efficient compression algorithm.

A comparison between the two standards is shown in Figure 4.7.

Chapter 5

Modern Sampling Theory

Sampling theory plays a central role in modern signal processing and communications, and has experienced a recent revival thanks, in part, to the recent advances in wavelet theory [30]. In the typical sampling setup depicted in Figure 5.1, the original continuous-time signal $x(t)$ is filtered before being (uniformly) sampled with sampling period T. The filtering may be a design choice or, as it is usually the case, may be due to the acquisition device. If we denote with $y(t) = h(t) * x(t)$ the filtered version of $x(t)$, the samples y_n are given by

$$y_n = \langle x(t), \varphi(t/T - n) \rangle = \int_{-\infty}^{\infty} x(t)\varphi(t/T - n)dt$$

where the sampling kernel $\varphi(t)$ is the scaled and time-reversed version of $h(t)$.

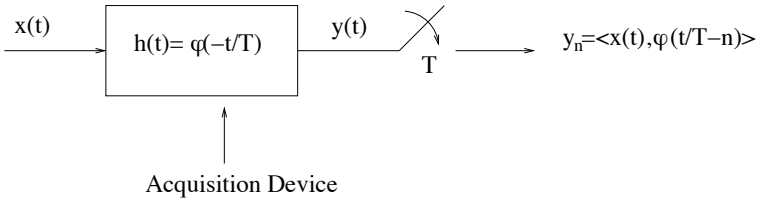


Figure 5.1: Sampling setup. Here, $x(t)$ is the continous-time signal, $h(t)$ the impulse response of the acquisition device and T the sampling period. The measured samples are $y_n = \langle x(t), \varphi(t/T - n) \rangle$.

The key problem then is to find the best way to reconstruct $x(t)$ from the given samples, and the key questions are: what classes of signals can be reconstructed? What classes of kernels allow such reconstructions? What kind of reconstruction algorithms are involved? Ideally, we would like to be able to reconstruct large classes of signals, using simple reconstruction algorithms and, most important, with general and physically realizable kernels.

The classical answer to the sampling problem is provided by the famous Shannon's sampling theorem which states the conditions to reconstruct bandlimited signals from their samples. The reconstruction process is, in this case, linear and the kernel is the sinc function. In fact, the whole sampling process can be interpreted as an approximation procedure in which the original signal is projected onto the shift-invariant subspace of bandlimited functions and only this projection can be reconstructed. This subspace interpretation has, later on, been used to extend the Shannon's theorem to classes of non-bandlimited signals that belong to shift-invariant subspaces, such as uniform splines [32, 30].

Recently, it was shown that it is possible to develop sampling schemes for classes of signals that are neither bandlimited nor belong to a fixed subspace [38] For instance, it was shown that it is possible to sample streams of Diracs or piecewise polynomial signals using a sinc or a Gaussian kernel. The common feature of such signals is that they have a parametric representation with a finite number of degrees of freedom and are, therefore, called signals with finite rate of innovation (FRI) [38]. The reconstruction process of these schemes is based on the use of a locator or annihilating filter, a tool widely used in spectral estimation [26] and error correction coding [2].

The fundamental limit of the above sampling methods, as well as of the classical Shannon reconstruction scheme, is that the choice of the sampling kernel is very limited and the kernels used are of infinite support. As a consequence, the reconstruction algorithm is usually physically non-realizable (e.g., realization of an ideal low-pass filter) or, in the case of FRI signals, becomes immediately complex and instable. The complexity is in fact influenced by the global rate of innovation of $x(t)$.

In this chapter we show that many signals with a local finite rate of innovation can be sampled and perfectly reconstructed using a wide range of sampling kernels and a local reconstruction algorithm. The reconstruction algorithm is also based on the annihilating filter method. The main property the kernel has to satisfy is to be able to reproduce polynomials or exponentials. Thus, functions satisfying Strang-Fix conditions (e.g., splines and scaling functions), exponential splines and functions with rational Fourier transforms can be used. This last family of kernels is of particular importance since most linear devices used in practice have a transfer function which is rational. Despite the fact that kernels with rational Fourier transform have infinite support, we show that the reconstruction algorithm remains local and, thus, its complexity still depends on the local, rather than global, rate of innovation of $x(t)$.

In the next section we review the notion of signals with finite rate of innovation and present the families of sampling kernels that are used in these sampling schemes. Section 5.2 presents the main sampling results for the case of kernels reproducing polynomials. In Section **??** and **??**, the previous sampling results are extended to the case in which the sampling kernel reproduces exponentials, moreover, as an example, we show how to estimate FRI signals at the output of electric circuit.

5.1 Signals and Kernels

In the introduction we have informally discussed about the signals and kernels that will be used in our sampling formulation. Let us now introduce more formally the notion of signals with finite rate of innovation [38] and present the families of sampling kernels that will be used in our sampling schemes.

5.1.1 Signals with Finite Rate of Innovation

Consider a signal of the form

$$x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R} \lambda_{k,r} g_r(t - t_k). \tag{5.1}$$

Clearly, if the set of functions $\{g_r(t)\}_{r=0,1,\dots,R}$ is known, the only free parameters in the signal $x(t)$ are the coefficients $\lambda_{k,r}$ and the time shifts t_k. It is therefore natural to introduce a counting function $C_x(t_a, t_b)$ that counts the number of free parameters in $x(t)$ over an interval $[t_a, t_b]$. The rate of innovation of $x(t)$ is then defined as [38]

$$\rho = \lim_{\tau\to\infty} \frac{1}{\tau} C_x\left(-\frac{\tau}{2}, \frac{\tau}{2}\right). \tag{5.2}$$

Definition 1 (Vetterli, Marziliano, Blu [38]) *A signal with a finite rate of innovation is a signal whose parametric representation is given in (5.1) and with a finite ρ as defined in (5.2).*

It is of interest to note that shift-invariant signals, including bandlimited signals, are included in Definition 5.1. For instance, if we call f_{max} the maximum non-zero frequency in a bandlimited real signal, then $\rho = 2f_{max}$. Therefore, one possible interpretation is that it is possible to sample bandlimited signals because they have finite rate of innovation and not because they are bandlimited.

In some cases it is more convenient to consider a local rate of innovation with respect to a moving window of size τ. The local rate of innovation at time t is thus given by [38]

$$\rho_\tau(t) = \frac{1}{\tau} C_x \left(t - \frac{\tau}{2}, t + \frac{\tau}{2} \right). \tag{5.3}$$

Clearly $\rho_\tau(t)$ tends to ρ as $\tau \to \infty$. In our context, as it will became evident later, the notion of local rate of innovation plays a more important role than the global rate of innovation. This is because our reconstruction schemes are local.

5.1.2 Sampling Kernels

As mentioned in the introduction, the signal $x(t)$ is usually filtered before being sampled. The samples y_n are given by $y_n = \langle x(t), \varphi(t/T - n) \rangle$, where the sampling kernel $\varphi(t)$ is the time reversed version of the filter's impulse response. The impulse response of the filter depends on the physical properties of the acquisition devise and, in most cases, is specified a-priori and cannot be modified. It is therefore important to develop sampling schemes that do not require the use of very particular or even physically non-realizable filters. In our formulation we can use a wide range of different kernels. For the sake of clarity, we divide them into three different families:

1. *Polynomial reproducing kernels*: Any kernel $\varphi(t)$ that together with its shifted versions can reproduce polynomials of maximum degree N. That is, any kernel that satisfies

$$\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t - n) = t^m \quad m = 0, 1, ..., N \tag{5.4}$$

   for a proper choice of the coefficients $c_{m,n}$.

2. *Exponential reproducing kernels*: Any kernel $\varphi(t)$ that together with its shifted versions can reproduce complex exponentials of the form $e^{\alpha_m t}$ with $\alpha_m = \alpha_0 + m\lambda$ and $m = 0, 1, ..., N$. That is, any kernel satisfying

$$\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t - n) = e^{\alpha_m t} \quad \text{with } \alpha_m = \alpha_0 + m\lambda \text{ and } m = 0, 1, ..., N \tag{5.5}$$

   for a proper choice of the coefficients $c_{m,n}$.

3. *Rational kernels*: Any stable kernel $\varphi(t)$ with rational Fourier transform of the form

$$\hat{\varphi}(\omega) = \frac{\prod_{i=0}^{I} (j\omega - b_i)}{\prod_{m=0}^{N} (j\omega - \alpha_m)} \quad \text{with } I < N,\ \alpha_m = \alpha_0 + m\lambda \text{ and } m = 0, 1, ..., N, \tag{5.6}$$

   where $\hat{\varphi}(\omega)$ is the Fourier transform of $\varphi(t)$.

In all cases, the choice of N depends on the local rate of innovation of the original signal $x(t)$ as will become clear later on. Since our reconstruction scheme is based on the use of a digital filter (i.e., the annihilating filter), the exponents in (5.5) and the poles in (5.6) must be restricted to $\alpha_m = \alpha_0 + m\lambda$, where α_0 and λ can be chosen arbitrarily but m is an integer. This fact will be more evident in Section **??**. Finally, the coefficients $c_{m,n}$ in (5.4) are given by $c_{m,n} = \frac{1}{T}\int_{-\infty}^{\infty} t^m \tilde{\varphi}(t/T - n)dt$, where $\tilde{\varphi}(t)$ is chosen to form with $\varphi(t)$ a quasi-biorthonormal set [5]. This includes the particular case where $\tilde{\varphi}(t)$ is the dual of $\varphi(t)$, that is, $\langle \tilde{\varphi}(t-n), \varphi(t-k)\rangle = \delta_{n,k}$. A similar expression applies to the coefficients $c_{m,n}$ in (5.5).

The first family of kernels includes any function satisfying the so called Strang-Fix conditions [27]. Namely, $\varphi(t)$ satisfies Eq. (5.4) if and only if

$$\hat{\varphi}(0) \neq 0 \text{ and } \hat{\varphi}^{(m)}(2n\pi) = 0 \text{ for } \begin{cases} n \neq 0 \\ \\ m = 0, 1, ..., N \end{cases}$$

where $\hat{\varphi}(\omega)$ is again the Fourier transform of $\varphi(t)$. These conditions were originally valid for functions with compact support only, more recently they have been extended to non-compactly supported functions [5, 13, 7].

One important example of functions satisfying Strang-Fix conditions is given by the family of B-splines [29]. The B-spline of order N can reproduce polynomials of maximum degree N and is the function with the smallest possible support that can achieve that order of approximation. More important, it is possible to show that any function $\varphi(t)$ that reproduces polynomials of degree N can be decomposed into a B-splines and a distribution $u(t)$ with $\int u(t)dt \neq 0$, that is, $\varphi(t) = u(t) * \beta_N(t)$ [3, 4, 24].

The theory related to the reproduction of exponentials is somewhat more recent and relies on the notion of Exponential splines (E-splines) [35]. A function $\beta_\alpha(t)$ with Fourier transform

$$\hat{\beta}_\alpha(\omega) = \frac{1 - e^{\alpha - j\omega}}{j\omega - \alpha}$$

is called E-spline of first order. Notice that α can be either real or complex. Moreover, notice that $\beta_\alpha(t)$ reduces to the classical zero-order B-spline when $\alpha = 0$. The function $\beta_\alpha(t)$ satisfies several interesting properties, in particular, it is of compact support and a linear combination of shifted versions of $\beta_\alpha(t)$ reproduces $e^{\alpha t}$. As in the classical case, higher order E-splines are obtained by successive convolutions of lower-order ones or

$$\hat{\beta}_{\vec{\alpha}}(\omega) = \prod_{n=0}^{N} \frac{1 - e^{\alpha_n - j\omega}}{j\omega - \alpha_n}$$

where $\vec{\alpha} = (\alpha_0, \alpha_1, ..., \alpha_N)$. The higher-order spline is again of compact support and it is possible to show that it can reproduce any exponential in the subspace spanned by $\{e^{\alpha_0 t}, e^{\alpha_1 t}, ..., e^{\alpha_N t}\}$ [35]. Moreover, since the exponential reproduction formula is preserved through convolution [35], any composite function of the form $\varphi(t) * \beta_{\vec{\alpha}}(t)$ is also able to reproduce exponentials. Therefore, the second group of kernels contains any composite function of the form $\varphi(t) * \beta_{\vec{\alpha}}(t)$ with $\beta_{\vec{\alpha}}(t) = \beta_{\alpha_0}(t) * \beta_{\alpha_1}(t) * ... * \beta_{\alpha_N}(t)$, $\alpha_m = \alpha_0 + m\lambda$ and $m = 0, 1, ..., N$.

Notice that the exponential case reduces to that of reproduction of polynomials when $\alpha_m = 0$ for $m = 0, 1, ..., N$. For this reason we could study our sampling schemes in the exponential case only and then particularize it to the polynomial case. However, we prefer to keep the two cases separated for the sake of simplicity.

The last group of kernels includes any *linear differential acquisition device*. That is, any linear device or system for which the input and output are related by a linear differential equation. This includes most of the commonly used electrical or mechanical systems.

The reason why we can sample signals with finite rate of innovation using such kernels is that we can *convert* a kernel $\varphi(t)$ with rational Fourier transform as in (5.6) into a kernel that reproduces exponentials. This is achieved by filtering the samples $y_n = \langle x(t), \varphi(t-n)\rangle$[1] with an FIR filter with z-transform $H(z) = \prod_{m=0}^{N}(1 - e^{\alpha_m} z)$.

For example, assume that $\hat{\varphi}(\omega) = \frac{1}{j\omega - \alpha}$ and $y_n = \langle x(t), \varphi(t-n)\rangle$. Then $H(z) = (1 - e^{\alpha} z)$ and we have that

$$\begin{aligned} z_n = h_n * y_n = y_n - e^{\alpha} y_{n+1} &\stackrel{(a)}{=} \langle x(t), \varphi(t-n) - e^{\alpha}\varphi(t-n-1)\rangle \\ &\stackrel{(b)}{=} \tfrac{1}{2\pi}\langle \hat{x}(\omega), e^{-j\omega n}\tfrac{(1-e^{\alpha - j\omega})}{(j\omega-\alpha)}\rangle \\ &\stackrel{(c)}{=} \langle x(t), \beta_{\alpha}(t-n)\rangle \end{aligned}$$

where (a) follows from the linearity of the inner product and (b) and (c) from Parseval's theorem. Here, $\hat{x}(\omega)$ is the Fourier transform of $x(t)$. Therefore, by filtering the samples y_n with the filter $H(z) = (1 - e^{\alpha} z)$ we obtain a new set of samples z_n that are equivalent to those that would have been obtained by sampling the original signal $x(t)$ with the E-spline $\beta_{\alpha}(t)$.

Likewise, when the original kernel has $N+1$ poles at locations $\vec{\alpha} = (\alpha_0, \alpha_1, ..., \alpha_N)$, by filtering the samples $y_n = \langle x(t), \varphi(t-n)\rangle$ with the filter $H(z) = \prod_{m=0}^{N}(1 - e^{\alpha_m} z)$ we have that

$$z_n = h_n * y_n = \langle x(t), \beta_{\vec{\alpha}}(t-n)\rangle$$

and the new kernel is of compact support and reproduces the exponentials $\{e^{\alpha_0 t}, e^{\alpha_1 t}, ..., e^{\alpha_N t}\}$.

In the most general case, the kernel has a frequency response as in (5.6) and by filtering the samples with the digital filter $H(z) = \prod_{m=0}^{N}(1 - e^{\alpha_m} z)$ we obtain a new kernel with Fourier transform $\prod_{i=0}^{I}(j\omega - b_i)\hat{\beta}_{\vec{\alpha}}(\omega)$. Functions with such Fourier transform are sometimes called generalized E-spline [31] and clearly are still able to reproduce the exponentials $\{e^{\alpha_0 t}, e^{\alpha_1 t}, ..., e^{\alpha_N t}\}$. Moreover, notice that since we are assuming $I < N$, these new kernels have compact support.

The use of E-splines and kernels with rational Fourier transforms will be investigated in Section **??** and **??**.

5.2 Reconstruction of FRI signals using kernels that reproduce polynomials

In this section, we assume that the sampling kernel $\varphi(t)$ satisfies the Strang-Fix conditions [27], that is, a linear combination of shifted versions of $\varphi(t)$ can reproduce polynomials of maximum degree N (see Equation (5.4)). We further assume that the sampling kernel is of compact support L, that is, $\varphi(t) \neq 0$ for $t \in [-L/2, L/2]$ where L is for simplicity an integer.[2]

We study the sampling of streams of Diracs, streams of differentiated Diracs and piecewise polynomial signals. Furthermore, possible extensions to any signal with finite rate of innovation are briefly discussed at the end of Section 5.2.2. We present the results for streams of Diracs in detail and derive the other sampling theorems directly from these results.

[1] We are assuming $T = 1$ for simplicity.

[2] Recall that functions satisfying Strang-Fix conditions can be of either compact or infinite support. However, the case of kernels with compact support is from a practical point of view more interesting. Thus, in this paper, we concentrate only on this case.

5.2.1 Streams of Diracs

Consider a stream of Diracs $x(t)$. Call y_n the observed samples, that is, $y_n = \langle x(t), \varphi(t-n)\rangle$ where, for simplicity, we have assumed $T = 1$. Assume for now that the signal contains only K Diracs, thus, $x(t) = \sum_{k=0}^{K-1} a_k \delta(t - t_k)$, $t \in \mathbb{R}$, and assume that the sampling kernel $\varphi(t)$ is able to reproduce polynomials of maximum degree $N \geq 2K-1$. We now show that under these hypotheses, it is possible to retrieve the locations t_k and the amplitudes a_k of $x(t)$ from its samples. The reconstruction algorithm operates in three steps. First, the first $N+1$ moments of the signal $x(t)$ are found. Second, the Diracs' locations are retrieved using an annihilating filter. Third, the amplitudes a_k are obtained solving a Vandermonde system. For a detailed description of the annihilating filter method we refer to [26, 38], the three steps of our scheme can be more precisely described as follows:

1. Retrieve the first $N+1$ moments of the signal $x(t)$.

   Call $\tau_m = \sum_n c_{m,n} y_n$, $m = 0, 1, ..., N$ the weighted sum of the observed samples, where the weights $c_{m,n}$ are those in Equation (5.4). We have that

$$\begin{aligned} \tau_m &= \textstyle\sum_n c_{m,n} y_n \\ &\overset{(a)}{=} \langle x(t), \textstyle\sum_n c_{m,n} \varphi(t-n)\rangle \\ &\overset{(b)}{=} \langle \textstyle\sum_{k=0}^{K-1} a_k \delta(t-t_k), \textstyle\sum_n c_{m,n}\varphi(t-n)\rangle \\ &\overset{(c)}{=} \textstyle\int_{-\infty}^{\infty} \sum_{k=0}^{K-1} a_k \delta(t-t_k) t^m dt \\ &= \textstyle\sum_{k=0}^{K-1} a_k t_k^m \quad m = 0, 1, ..., N \end{aligned} \tag{5.7}$$

   where (a) follows from the linearity of the inner product, (b) from the fact that $x(t) = \sum_{k=0}^{K-1} a_k \delta(t - t_k)$, and (c) from the polynomial reproduction formula in (5.4). The integral in (c) represents precisely the mth order moment of the original signal $x(t)$. Hence, proper linear combinations of the observed samples provides the first $N+1$ moments of the signal. This fact is graphically illustrated in Figure 5.2.

   Since the original signal is a stream of K Diracs, the moments of $x(t)$ have the form

$$\tau_m = \sum_{k=0}^{K-1} a_k t_k^m \quad m = 0, 1, ..., N$$

   which is very often encountered in spectral estimation as well as error-correction coding. It is therefore possible to estimate locations and amplitudes of the Diracs from the moments τ_m using the annihilating filter method which is normally used for harmonic retrieval.

2. Find the locations t_k of $x(t)$.

   Call h_m $m = 0, 1, ..., K$ the filter with z-transform

$$H(z) = \sum_{m=0}^{K} h_m z^{-m} = \prod_{k=0}^{K-1} (1 - t_k z^{-1}). \tag{5.8}$$

That is, the roots of $H(z)$ correspond to the locations t_k. It clearly follows that

$$h_m * \tau_m = \sum_{i=0}^{K} h_i \tau_{m-i} = \sum_{i=0}^{K} \sum_{k=0}^{K-1} a_k h_i t_k^{m-i} = \sum_{k=0}^{K-1} a_k t_k^m \underbrace{\sum_{i=0}^{K} h_i t_k^{-i}}_{0} = 0. \tag{5.9}$$

The filter h_m is thus called annihilating filter since it annihilates the observed signal τ_m. The zeros of this filter uniquely define the set of locations t_k since the locations are distinct. The filter coefficients h_m are found from the system of equations in (5.9). Since $h_0 = 1$, the identity in (5.9) leads to a Yule-Walker system of equations involving at least $2K$ consecutive values of τ_m and can be written in matrix form as follows

$$\begin{bmatrix} \tau_{K-1} & \tau_{K-2} & \cdots & \tau_0 \\ \tau_K & \tau_{K-1} & \cdots & \tau_1 \\ \vdots & \vdots & \ddots & \vdots \\ \tau_{N-1} & \tau_{N-2} & \cdots & \tau_{N-K} \end{bmatrix} \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_K \end{pmatrix} = - \begin{pmatrix} \tau_K \\ \tau_{K+1} \\ \vdots \\ \tau_N \end{pmatrix}. \tag{5.10}$$

This classic Yule-Walker system has, in this case, a unique solution since h_m is unique for the given signal. Given the filter coefficients h_m, the locations of the Diracs are the roots of the polynomial in (5.8). Notice that, since we need at least $2K$ consecutive values of τ_m to solve the Yule-Walker system, we need the sampling kernel to be able to reproduce polynomials of maximum degree $N \geq 2K - 1$.

3. Find the weight a_k.

   Given the locations $t_0, t_1, ..., t_k$, the weights a_k are obtained by solving, for instance, the first K consecutive equations in (5.7). These equations can be

   written in matrix form as follows:

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ t_0 & t_1 & \cdots & t_{K-1} \\ \vdots & \vdots & \ddots & \vdots \\ t_0^{K-1} & t_1^{K-1} & \cdots & t_{K-1}^{K-1} \end{bmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{K-1} \end{pmatrix} = \begin{pmatrix} \tau_0 \\ \tau_1 \\ \vdots \\ \tau_{K-1} \end{pmatrix}.$$

   This is a Vandermonde system which yields a unique solution for the weights a_k given that the t_ks are distinct.

The three steps above shows that it is indeed possible to retrieve amplitudes and locations of K Diracs from their samples. At the same time, the fact that the sampling kernels we consider have compact support can be used to reconstruct signals with more than K Diracs. To be more precise, the support of the kernel is L, thus, a single Dirac can influence at most L consecutive samples and K consecutive Diracs can generate a block of at most KL consecutive non-zero samples. Thus, if two groups of K consecutive Diracs are sufficiently distant, the two blocks of non-zero samples they

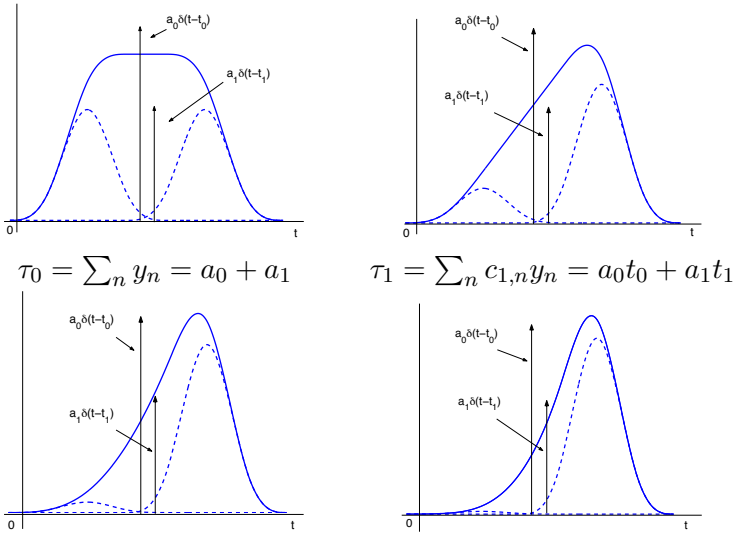


Figure 5.2: Illustration of the reproduction of polynomial of maximum degree three using cubic splines. In this example, only four translated versions of the splines overlap the two Diracs. The two dashed functions in each plot represent the properly weighted first and last spline overlapping the two Diracs. The four solid-line functions represent the weighted sums of these four splines. Because of the polynomial reproduction formula, the following is true: $c_{m,0}y_0+c_{m,1}y_1+c_{m,2}y_2+c_{m,3}y_3 = a_0t_0^m+a_1t_1^m$ for $m = 0, 1, 2, 3$.

generate do not influence each other and we can apply the above reconstruction method on each block independently.

In order to be able to separate two blocks of non-zero samples we need to have a sufficient number of zero samples in between. If we assume that there are at most K Diracs in an interval of size $KL+1$, we are assured that at least a zero sample will separate two groups of non-zeros. While in most cases the above condition is enough, there are situations in which it is not sufficient. This happens because in some very unlucky circumstances a zero sample may not indicate absence of Diracs, but may have been generated either by a particular combination of Diracs or by the fact that a Dirac may be located in a position where the sampling kernel is equal to zero. To avoid that these rare events prevent the algorithm from working properly, we need to make stronger assumptions. If the sampling kernel is non-zero in the whole support L as in the case of B-splines, then the assumption that there are at most K Diracs in an interval of size $K(L+1)-1$ is sufficient. On the other hand, if the sampling kernel can be zero in some locations, we need to assume that there are at most K Diracs in an interval of size $2KL$. The rational behind these conditions is that they ensure that for any sequence of consecutive 'fake' zeros, there is in the same window a longer sequence of consecutive 'true' zeros. Therefore, the only thing the algorithm has to do is to search for the longest sequence of zeros in a group of $2KL$ samples. More precisely, the reconstruction algorithm operates as follows (see also Figure 5.3): The algorithm starts by looking for the first non-zero sample in the sequence, call it y_{n_1}. The algorithm then checks the $2KL$ consecutive samples $y_{n_1}, y_{n_1+1}, ..., y_{n_1+2KL-1}$ and looks for the longest sequence of consecutive zeros inside this block. Denote this sequence as $y_{z_1}, y_{z_1+1}, ..., y_{z_n}$, it is easy to show that such a sequence must include 'true' zeros and as such can be used to separate two blocks of non-zero samples. This means that the Diracs that have generated the non-zero samples $y_{n_1}, y_{n_1+1}, ..., y_{z_1-1}$ are not influenced by any other Dirac and can therefore be reconstructed using

the reconstruction scheme presented before. After the reconstruction, the algorithm starts the whole process again from the sample y_{z_n+1} on.

While the above conditions can be made less stringent in many situations and, from a practical point of view, the event of having a 'fake' zero is very unlucky; for simplicity for the rest of the paper we will keep assuming the there are at most K Diracs in an interval of size $2KL$.

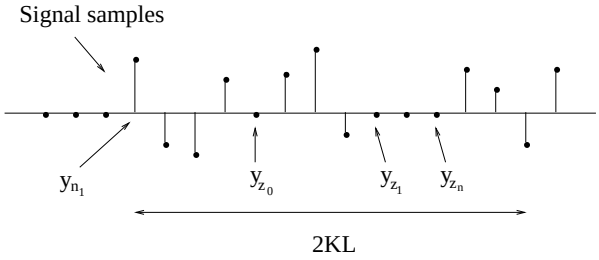


Figure 5.3: The sequential reconstruction algorithm starts by looking for the first non-zero sample (in this case the sample y_{n_1}), it then looks for the longest sequence of consecutive zeros in the block of $2KL$ samples $y_{n_1}, y_{n_1+1}, ..., y_{n_1+2KL-1}$. In this examples such a sequence start with the sample y_{z_1}. It is possible to show that such a sequence must contains 'true' zeros and can therefore be used to separate two blocks of non-zero samples. Notice that the algorithm disregards the isolated zero sample y_{z_0}. That is because we cannot guarantee that such a sample truly indicates absence of Diracs.

We can thus summarize the above discussion as follows:

Theorem 1 *Assume a sampling kernel $\varphi(t)$ that can reproduce polynomials of maximum degree $N \geq 2K-1$ and of compact support L. An infinite-length stream of Diracs $x(t) = \sum_{n\in\mathbb{Z}} a_n \delta(t-t_n)$ is uniquely determined from the samples defined by $y_n = \langle x(t), \varphi(t/T - n)\rangle$ if there are at most K Diracs in an interval of size $2KLT$.*

Using the notation introduced in Section 5.1, we can claim that the above theorem is fundamentally saying that it is possible to sample any stream of Diracs with local rate of innovation $\rho_{2KLT} \leq 1/LT$. This means that there is a fundamental connection between the local complexity of the signal and the complexity of the reconstruction process. For instance, if there is at most one Dirac in an interval of size $2TL$, only two moments need to be retrieved at each iteration and the estimation of the amplitude and location of the Dirac becomes straightforward. In contrast, the reconstruction process becomes more complex and instable, when the number K of Diracs to retrieve at each iteration becomes very large. This fact is of particular interest in the case of noisy measurements. In that context, in fact, stability of the reconstruction algorithm is of crucial importance.

To conclude this section we show in Figure 5.4 an example of our sampling scheme. In this example the signal is made of two groups of $K = 4$ Diracs and is shown in Figure 5.4(a). The signal is sampled with a B-spline that can reproduce polynomials of degree $2K - 1 = 7$ (Figure 5.4(b)) and the samples are shown in Figure 5.4(c). Since the non-zero samples generated by the two groups of Diracs are separated by a sequence of zero samples, the reconstruction algorithm can operate on the first group of non-zero samples to retrieve the first K Diracs, and then reiterate the process on the following group of non-zero samples to retrieve the remaining K Diracs. The reconstructed signal is shown in Figure 5.4(d) and reconstruction is exact to machine precision.

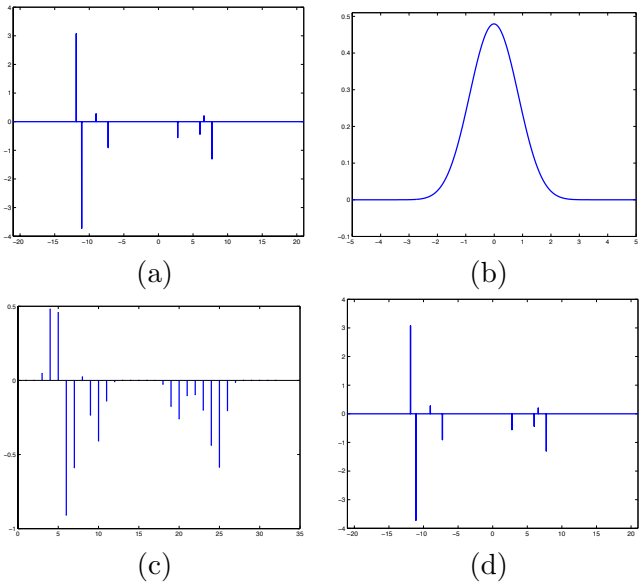


Figure 5.4: Sampling of streams of Diracs. In this example, the original signal, shown in Figure 5.4(a), is made of two groups of $K = 4$ Diracs. The sampling kernel is shown in Figure 5.4(b) and is a B-spline $\beta_7(t)$ that can reproduce polynomials of maximum degree $2K = 1 = 7$. The observed samples are shown in 5.4(c). Notice that the non-zero samples generated by the two sets of Diracs are separated by a sequence of zero samples. This allows the sampling algorithm to retrieve the two groups of K Diracs sequentially. The reconstructed signal is shown in Figure 5.4(d) and the reconstruction is exact to numerical precision.

5.2.2 Stream of Differentiated Diracs

Consider now a stream of differentiated Diracs:

$$x(t) = \sum_{k=0}^{K-1} \sum_{r=0}^{R_k-1} a_{k,r} \delta^{(r)}(t - t_k).$$

Note that this signal has K Diracs and $\hat{K} = \sum_{k=0}^{K-1} R_k$ weights. Moreover, recall that the rth derivative of a Dirac is a function that satisfies the property $\int f(t)\delta^{(r)}(t - t_0)dt = (-1)^r f^{(r)}(t_0)$.

Assume that $x(t)$ is sampled with a kernel that can reproduce polynomials of maximum degree $N \geq 2\hat{K} - 1$. As shown in the previous section, using the polynomial reproduction formula, we can compute the first $N + 1$ moments of $x(t)$ from its samples y_n:

$$\begin{aligned} \tau_m &= \textstyle\sum_n c_{m,n} y_n \\ &= \textstyle\int_{-\infty}^{\infty} x(t) t^m dt \\ &= \textstyle\sum_{k=0}^{K-1} \sum_{r=0}^{R_k-1} (-1)^l a_{k,l} m(m-1)\cdots(m-r+1) t_k^{m-r}, \qquad m = 0, 1, ..., N \end{aligned} \tag{5.11}$$

where we have used the fact that $\int t^m \delta^{(r)}(t - t_0)dt = (-1)^r m(m-1)\cdots(m-r+1)t_0^{m-r}$.

We can thus say that what we observe is

$$\tau_m = \sum_{k=0}^{K-1} \sum_{r=0}^{R_k-1} (-1)^l a_{k,r} m(m-1)\cdots(m-r+1) t_k^{m-r}.$$

It can be shown that the filter $(1-t_k z^{-1})^M$ annihilates the signal $m^r t_k^m$, with $r \leq M-1$. Therefore the filter h_m with z-transform

$$H(z) = \prod_{k=0}^{K-1} (1 - t_k z^{-1})^{R_k}$$

annihilates τ_m. The $\hat{K}$ unknown coefficients of h_m can be found solving a Yule-Walker system similar to the one in the previous section. We need at least $\hat{K}$ equations to find these coefficients, therefore, we need to know at least $2\hat{K}$ consecutive values of τ_m (this is why $N \geq 2\hat{K}-1$). From the annihilating filter we obtain the locations $t_0, t_1, ..., t_{K-1}$. We then need to solve the first $\hat{K}$ equations in (5.11) to obtain the weights $a_{k,r}$. This is a generalized Vandermonde system which has again a unique solution given that the t_ks are distinct.

The above analysis can be summarized in the following theorem:

Theorem 2 *Assume a sampling kernel $\varphi(t)$ that can reproduce polynomials of maximum degree $N \geq 2\hat{K}-1$ and of compact support L. An infinite-length stream of differentiated Diracs $x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R_k-1} a_{k,r}\delta^{(r)}(t-t_k)$ is uniquely determined by the samples $y_n = \langle x(t), \varphi(t/T-n)\rangle$ if there are at most K differentiated Diracs with $\hat{K}$ weights in an interval of size $2KLT$.*

Let us now return to the definition of signals with finite rate of innovation given in Section 5.1:

$$x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R} \lambda_{k,r} g_r(t-t_k). \tag{5.12}$$

The sampling schemes developed so far correspond to the case in which $g_0(t-t_k) = \delta(t-t_k)$ and $g_r(t-t_k) = \delta^{(r)}(t-t_k)$, $r = 1, ..., R$. However, further extensions are possible. Assume for instance that $g_0(t)$ is of compact support $\hat{L}$ and that $\hat{g}_0(\omega) \neq 0$ for $\omega = 0$. Moreover, assume that $g_r(t) = g_0^{(r)}(t)$, that is, $g_r(t)$ is the rth order derivative of $g_0(t)$. Then under these conditions the sampling of $x(t)$ is possible and can be reduced to the sampling of a stream of differentiated Diracs.

The observed samples $y_n = \langle x(t), \varphi(t-n)\rangle$ are in fact equivalent to those given by $y_n = \langle \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R} \lambda_{k,r}\delta^{(r)}(t-t_k), g_0(t-n) * \varphi(t-n)\rangle$. Now, assume the sampling kernel $\varphi(t)$ can reproduce polynomials of degree N and has compact support L, the new kernel $g_0(t-n) * \varphi(t-n)$ has compact support $L+\hat{L}$ and, since $\hat{g}(\omega) \neq 0$ for $\omega = 0$, can still reproduce polynomials of degree N (Strang-Fix conditions are still satisfied). Therefore, if there are no more than K Diracs in an interval of size $2K(L+\hat{L})$ and $N \geq 2KR-1$ the hypotheses of Theorem 2 are satisfied and the samples y_n are sufficient to retrieve the weights $\lambda_{k,r}$ and the locations t_k. We can formalize this discussion with the following corollary

Corollary 1 *Assume a sampling kernel $\varphi(t)$ of compact support L and that can reproduce polynomials of maximum degree N. An infinite-length signal $x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R} \lambda_{k,r} g_r(\frac{t-t_k}{T})$, where $g_0(t)$ is of compact support $\hat{L}$ and $\hat{g}_0(\omega) \neq 0$ for $\omega = 0$ and where $g_r(t) = g_0^{(r)}(t)$, is uniquely defined by the samples $y_n = \langle x(t), \varphi(t/T-n)\rangle$ if there are at most K time shifts t_k in an interval of size $2K(L+\hat{L})T$ and $N \geq 2KR-1$.*

Finally, another important example of signals with finite rate of innovation, namely, piecewise polynomial signals, will be discussed in the next section.

5.2.3 Piecewise Polynomial Signals

A signal $x(t)$ is piecewise polynomial with pieces of maximum degree R if and only if its $(R+1)$ derivative is a stream of differentiated Diracs or $x(t)^{(R+1)}(t) = \sum_{n\in\mathbb{Z}} \sum_{r=0}^{R} a_{n,r}\delta^{(r)}(t-t_n)$. This

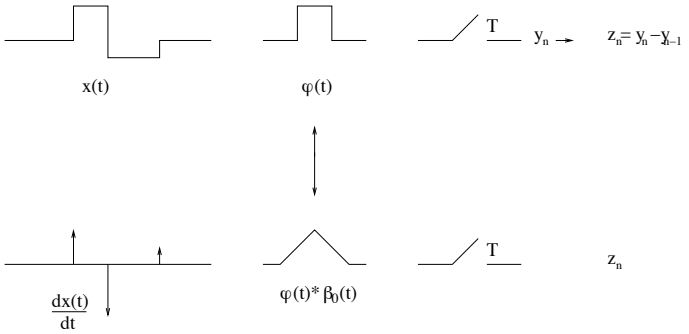


Figure 5.5: The observed samples y_n are given by $y_n = \langle x(t), \varphi(t/T - n)\rangle$, where the sampling kernel $\varphi(t)$ is, in this example, the box function and the original signal $x(t)$ is piecewise constant. The finite difference $y_n - y_{n-1}$ leads to the new samples z_n that are equivalent to those obtained by sampling $\frac{dx(t)}{dt}$ with the new kernel $\varphi(t) * \beta_0(t)$ which, in this case, is a linear spline.

means that if we are able to relate the samples of $x(t)$ to those of $x^{(R+1)}(t)$, we can use Theorem 2 to reconstruct $x(t)$. This is indeed possible by recalling the link existing between discrete differentiation and derivation in continuous domain.

Consider the samples $y_n = \langle x(t), \varphi(t - n)\rangle$ where $\varphi(t)$ is a generic sampling kernel. Let $z_n^{(1)}$ denote the finite difference $y_{n+1} - y_n$. It follows that

$$\begin{aligned} z_n^{(1)} = y_{n+1} - y_n &= \langle x(t), \varphi(t - n - 1) - \varphi(t - n)\rangle \\ &= \tfrac{1}{2\pi}\langle \hat{x}(\omega), \hat{\varphi}(\omega)(e^{-j\omega} - 1)e^{-j\omega n}\rangle \\ &= \tfrac{1}{2\pi}\langle \hat{x}(\omega), -j\omega\hat{\varphi}(\omega)\left(\tfrac{1-e^{-j\omega}}{j\omega}\right)e^{-j\omega n}\rangle \\ &= \tfrac{1}{2\pi}\langle j\omega\hat{x}(\omega), \hat{\varphi}(\omega)\hat{\beta}_0(\omega)e^{-j\omega n}\rangle \\ &= \langle \tfrac{dx(t)}{dt}, \varphi(t - n) * \beta_0(t - n)\rangle. \end{aligned}$$

This means that the samples $z_n^{(1)}$ are equivalent to those given by the inner products between the derivative of $x(t)$ and the new kernel $\varphi(t) * \beta_0(t)$. This equivalence is illustrated graphically in Figure 5.5. In the same way, it is straightforward to show that the $(R+1)$th finite difference $z_n^{(R+1)}$ represents the samples obtained by sampling $x^{(R+1)}(t)$ with the kernel $\varphi(t) * \beta_R(t)$, where $\beta_R(t)$ is the B-spline of degree R.

Now, assume that $\varphi(t)$ is of compact support L and that it can reproduce polynomials of maximum degree N. Then $\varphi(t) * \beta_R(t)$ has support $L+R+1$ and can reproduce polynomials of maximum degree $N + R + 1$. Thus, if the new kernel satisfies the hypotheses of Theorem 2, the samples $z_n^{(R+1)}$ are a sufficient representation of $x^{(R+1)}(t)$ and, therefore, of $x(t)$. This leads to the following theorem

Theorem 3 *Assume a sampling kernel $\varphi(t)$ of compact support L and that can reproduce polynomials of maximum degree N. An infinite-length piecewise polynomial signal $x(t)$ with pieces of maximum degree $R - 1$ $(R > 0)$ is uniquely defined by the samples $y_n = \langle x(t), \varphi(t/T - n)\rangle$ if there are at most K polynomial discontinuities in an interval of size $2K(L + R)T$ and $N + R \geq 2KR - 1$.*

Proof: Assume again $T = 1$. Given the samples y_n, compute the Rth finite difference $z_n^{(R)}$. As shown before, $z_n^{(R)} = \langle x^{(R)}(t), \varphi(t-n) * \beta_{R-1}(t-n) \rangle$ and $x(t)^{(R)}(t) = \sum_{n\in\mathbb{Z}} \sum_{r=0}^{R-1} a_{n,r} \delta^{(r)}(t-t_n)$. The new kernel $\varphi(t) * \beta_{R-1}(t)$ has support $L+R$ and can reproduce polynomials of maximum degree $N+R$. Since for hypothesis $x(t)$ has at most K polynomial discontinuities in an interval of size $2K(L+R)$, $x^{(R)}(t)$ has at most K Diracs in that interval with a total number of weights $\hat{K} = KR$. Since we are assuming $N + R \geq 2KR - 1$, the hypotheses of Theorem 2 are satisfied, thus, the samples $z_n^{(R)}$ are sufficient to reconstruct $x^{(R)}(t)$ and therefore $x(t)$.[3]

□

A numerical example is shown in Figure 5.6. In this case, the signal is piecewise constant and we assume that the signal can have at most two arbitrarily close discontinuities ($K = 2$). For this reason the sampling kernel must be able to reproduce polynomial of degree two and, in this example, is a quadratic spline $\beta_2(t)$. The observed samples y_n are shown in Figure 5.6(b) and the first order finite difference of y_n results in the samples z_n which are shown in Figure 5.6(c). These samples are equivalent to those obtained by sampling $\frac{dx(t)}{dt}$, which is a stream of Diracs, with the new kernel $\beta_3(t) = \beta_2(t) * \beta_0(t)$. Thus, the hypotheses of Theorem 1 are satisfied and the samples z_n are sufficient to reconstruct $\frac{dx(t)}{dt}$ and $x(t)$. The reconstructed piecewise constant signal is shown in Figure 5.6(d).

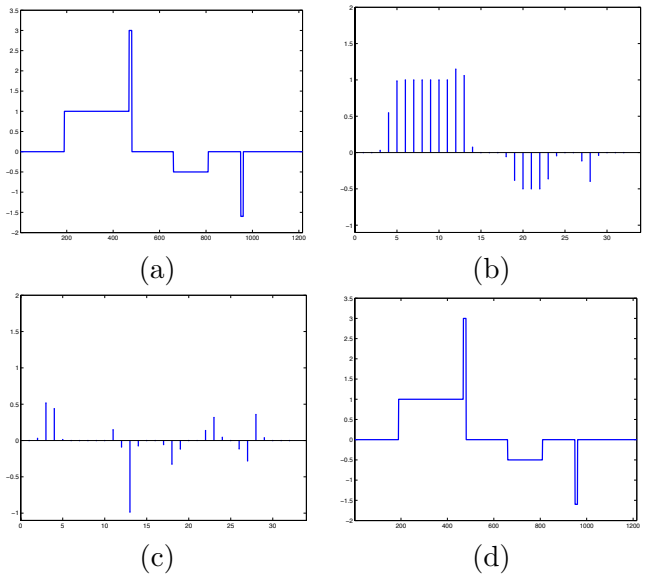


Figure 5.6: Sampling of piecewise polynomial signals. In this example the original signal is piecewise constant and is shown in Figure 5.6(a). The signal can have up to two arbitrarily close discontinuities ($K = 2$). The sampling kernel is in this case a quadratic spline and the observed samples y_n are shown in Figure 5.6(b). The first order finite difference of the samples y_n leads to the samples z_n shown in Figure 5.6(c). From this samples it is then possible to reconstruct the original signal exactly and the reconstructed signal is shown in Figure 5.6(d).

[3]Note that the mean of $x(t)$ is obtained directly.

5.3 FRI Signals with Noise

"Noise", or more generally model mismatch are unfortunately omnipresent in data acquisition, making the solution presented in the previous sections only ideal. For the sake of the argument we assume that the noise is additive Gaussian with variance σ^2 and assume that it is added to the moments τ_m. In order to combat noise we need a number N of moments higher than the minimum number $2K$ (i.e., $N > 2K$). In this way we can build a bigger Toeplitz matrix $\mathbf{S}$ given by:

$$S = \underbrace{\begin{bmatrix} \tau_K & \tau_{K-1} & \tau_{K-2} & \cdots & \tau_0 \\ \tau_{K+1} & \tau_K & \tau_{K-1} & \cdots & \tau_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tau_{N-1} & \tau_{N-2} & \tau_{N-3} & \cdots & \tau_{N-K-1} \end{bmatrix}}_{K+1 \text{ columns}}$$

Our denoising problem is similar to problems already encountered decades ago by researchers in spectral analysis. Thus we will not try to propose new approaches, instead our choice falls on to a simple but effective approach, the Total Least-Squares approximation (implemented using a *Singular Value Decomposition*), possibly enhanced by an initial "denoising" (more exactly: "model matching") step provided by what we call *Cadzow's iterated algorithm* [6].

5.3.1 Total least-squares approach

In the noiseless case, the annihilation equation essentially says that $\mathbf{S}H = 0$. In the presence of noise, the annihilation equation is not satisfied exactly, yet it is still reasonable to expect that the minimization of the Euclidian norm $\|\mathbf{S}H\|^2$ under the constraint that $\|H\|^2 = 1$ may yield an interesting estimate of H. It is known that this minimization can be solved by performing a *singular value decomposition* of $\mathbf{S}$ —more exactly: an eigenvalue decomposition of the matrix $\mathbf{S}^T\mathbf{S}$— and choosing for H the eigenvector corresponding to the smallest eigenvalue. More specifically, if $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$ where $\mathbf{U}$ is a $(N-K)\times(K+1)$ unitary matrix, $\mathbf{\Lambda}$ is a $(K+1)\times(K+1)$ diagonal matrix with decreasing positive elements, and $\mathbf{V}$ is a $(K+1)\times(K+1)$ unitary matrix, then H is the last column of $\mathbf{V}$. Once the t_k are retrieved, the α_k can be estimated using the noiseless approach.

5.3.2 Extra denoising: Cadzow

The previous algorithm works quite well for moderate values of the noise—a level that depends on the number of Diracs. However, for small SNR, the results may become unreliable and it is advisable to apply a robust procedure that 'denoise' the moments before applying total least-square (TLS) approach. This iterative procedure was already suggested by Tufts and Kumaresan and analyzed in [6]. It is possible to show that the noiseless matrix $\mathbf{S}$ is of rank K whenever $N > 2K$. The idea consists thus in performing the SVD of $\mathbf{S}$, say $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$, and keeping the K largest diagonal coefficients of Λ while forcing to zero the others to yield $\mathbf{\Lambda}'$. The resulting matrix $\mathbf{S}' = \mathbf{U}\mathbf{\Lambda}'\mathbf{V}^T$ is not Toeplitz anymore but its best Toeplitz approximation is obtained by averaging the diagonals of $\mathbf{S}'$. This leads to a new "denoised" sequence $\hat{\tau}'_m$ that matches the noiseless FRI model better than the original sequence. A few of these iterations lead to moments that are much closer to the noiseless one's, then on these moments one can apply TLS.

Bibliography

[1] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. Image Processing*, pages 205–205, April 1992.

[2] R.E. Blahut. *Theory and Practice of Error Control Codes.* Addison-Wesley, 1983.

[3] T. Blu, P. Thevenaz, and M. Unser. MOMS maximal-order interpolation of minimal support. *IEEE Trans. on Image Processing*, 10(7):1069–1080, July 2001.

[4] T. Blu, P. Thevenaz, and M. Unser. Complete parameterization of piecewise-polynomial interpolation kernels. *IEEE Trans. on Image Processing*, 12(11):1297–1309, November 2003.

[5] T. Blu and M. Unser. Approximation errors for quasiinterpolators and (multi-) wavelet expansions. *Applied and Computational Harmonic Analysis*, 6 (2):219–251, March 1999.

[6] J.A. Cadzow. Signal enhancement - a composite property mapping algorithm. *IEEE Trans. ASSP*, 36:49–62, January 1988.

[7] E.W. Cheney and W.A. Light. Quasiinterpolation with basis functions having noncompact support. *Constr. Approx.*, 8:35–48, 1992.

[8] A. Cohen, I. Daubechies, and J.-C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Commun. on Pure and Applied Math.*, 45:485–560, 1992.

[9] T. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley and Sons, NY, 1991.

[10] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Commun. on Pure and Appl. Math.*, 41:909–996, November 1988.

[11] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Information Theory*, 36(5):961–1005, September 1990.

[12] I. Daubechies. *Ten Lectures on Wavelets.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[13] C. de Boor, R. A. Devore, and A. Ron. Approximation from shift-invariant subspaces of $L_2(\mathbb{R}^d)$. *Trans. Amer. Math. Society*, 341(2):787–806, February 1994.

[14] L. Debnath and P Mikusinski. *Hilbert Spaces with Applications.* Academic Press, 1999.

[15] R. J. Duffin and A. C. Schaeffer. A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc.*, 72:341–366, December 1952.

[16] A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression.* Kluwer Acad. Pub., Boston, MA, 1992.

[17] R. M. Gray and D.L. Neuhoff. Quantization. *IEEE Trans. on Information Theory*, 44(6):2325–2383, October 1998.

[18] D. LeGall and A. Tabatabai. Subband coding of digital images using symmetric short kernel filters and arithmentic coding techniques. In *IEEE Int. Conf. Acoustic, Speech and Signal Proc.*, New York, USA, April 1988.

[19] J.M. Lina and M. Mayrand. Complex Daubechies wavelets. *Applied Computational Harmonic Analysis*, 2:219–229, 1995.

[20] S. Mallat. Multiresolution approximations and wavelet orthonormal bases of $l^2(\mathbb{R}$. *IEEE Trans. Amer. Math. Soc.*, 315:69–87, September 1989.

[21] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Patt. Recog. and Mach. Intell.*, 11:674–693, July 1989.

[22] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[23] Y. Meyer. *Wavelets and Operators*. Advanced Mathematics. Cambridge University Press, 1992.

[24] A. Ron. Factorization theorems fot univariate splines on regular grids. *Israel J. Math.*, 70(1):48–68, 1990.

[25] J.M. Shapiro. Embedded image coding using zerotrees of wavelets coefficients. *IEEE Trans. on Signal Processing*, 41:3445–3462, December 1993.

[26] P. Stoica and R Moses. *Introduction to Spectral Analysis*. Englewood Cliffs,NJ, Prentice-Hall, 2000.

[27] G. Strang and Fix. G. A Fourier analysis of the finite element variational method. In *Constructive Aspect of Functional Analysis*, pages 796–830, Rome, Italy, 1971.

[28] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Boston, 1996.

[29] M. Unser. Splines: a perfect fit for signal and image processing. *IEEE Signal Processing Magazine*, 16(6):22–38, November 1999.

[30] M. Unser. Sampling-50 years after Shannon. *Proc. IEEE*, 88:569–587, April 2000.

[31] M. Unser. Cardinal Exponential Splines: Part II-think analogue, act digital. *IEEE Trans. on Signal Processing*, 53(4):1439–1449, April 2005.

[32] M. Unser and A. Aldroubi. A general sampling theory for nonideal acquisition devices. *IEEE Trans. Signal Processing*, 42(11):2915–2925, November 1994.

[33] M. Unser and T. Blu. Mathematical properties of the JPEG2000 wavelet filters. *IEEE Trans. Image Processing*, 12(9):1080–1090, September 2003.

[34] M. Unser and T. Blu. Wavelet theory demystified. *IEEE Trans. Signal Processing*, 51(2):470–483, February 2003.

[35] M. Unser and T. Blu. Cardinal Exponential Splines: Part I-Theory and Filtering Algorithms. *IEEE Trans. on Signal Processing*, 53(4):1425–1438, April 2005.

[36] M. Vetterli and J. Kovačević. *Wavelets and Subband Coding*. Prentice-Hall, 1995.

[37] M. Vetterli, J. Kovačević, and V. K Goyal. *Foundations of Signal Processing*. Cambridge University Press, 2013.

[38] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Trans. Signal Processing*, 50(6):1417–1428, June 2002.