Available at
www.ElsevierMathematics.com
POWERED BY SCIENCE DIRECT®
Journal of Symbolic Computation 37 (2004) 227–259

Journal of Symbolic Computation

www.elsevier.com/locate/jsc

Gröbner bases and wavelet design

Jérôme Lebrun[a,*], Ivan Selesnick[b]

[a] *Laboratoire I3S, CNRS/UNSA, Sophia Antipolis, France*
[b] *Department of Electrical Engineering, Polytechnic University, Brooklyn, USA*

Received 5 September 2000; accepted 11 June 2002

Abstract

In this paper, we detail the use of symbolic methods in order to solve some advanced design problems arising in signal processing. Our interest lies especially in the construction of wavelet filters for which the usual spectral factorization approach (used for example to construct the well-known Daubechies filters) is not applicable. In these problems, we show how the design equations can be written as multivariate polynomial systems of equations and accordingly how Gröbner algorithms offer an effective way to obtain solutions in some of these cases.

Keywords: Discrete wavelet transform (DWT); Conjugate quadrature filters (CQFS)

0. Introduction

Wavelets and filter banks have become useful in digital signal processing in part because of their ability to represent piecewise smooth signals with relative efficiency. For such signals, the discrete wavelet transform (DWT) of an n-point vector is again an n-point vector, but one for which the energy is compacted into fewer values. In as far as this is true, the DWT is useful for signal compression (JPEG 2000), fast algorithms, and signal estimation and modeling (noise suppression and image segmentation, etc). The DWT is usually implemented as an iterated digital filter bank tree, so the design of a wavelet transform amounts to the design of a filter bank.

While the spectral factorization approach is the most convenient method to construct the classic wavelets (Daubechies, 1992) (and the corresponding digital filters), it is not

* Corresponding author.
E-mail addresses: lebrun@i3s.unice.fr (J. Lebrun), selesi@taco.poly.edu (I. Selesnick).
URLs: http://www.i3s.unice.fr/~lebrun, http://www.taco.poly.edu/selesi.

0747-7171/$ - see front matter
doi:10.1016/j.jsc.2002.06.002

applicable to many of the other wavelet design problems where additional constraints are imposed. However, for many of these design problems, the design equations can be written as a multivariate polynomial system of equations. Accordingly, Gröbner basis algorithms offer a way to obtain solutions in these cases. This paper describes the general wavelet design problem from the perspective of filter banks and explains the derivation of the core design equations. In addition, it is noted that the design of wavelet bases is intriguing in part for its limitations—specifically, in many cases it is not possible to obtain wavelets having all the properties one desires. This has motivated the development, for example, of multiwavelet bases, which are developed in Sections 5 and 6, and of wavelet frames (of *overcomplete* bases) which are described in Section 7. For both multiwavelet bases and some wavelet frames, the spectral factorization approach which is key in the construction of Daubechies wavelets, cannot be used anymore. However, as described in the following respective sections, it becomes possible to derive solutions to these new problems using Gröbner bases.

Recently, major advances have been achieved in the field of computational algebraic geometry that lead to new efficient ways to deal with one of the central applications of computer algebra: solving systems of multivariate polynomial equations. Using the new algorithms that have been developed, practical problems like (multi)wavelet design can now be solved exactly in a way that is very competitive with numerical methods. One of the most promising schemes to solve systems of polynomial equations has been by computing Gröbner bases. At the same time, even though the computation of a Gröbner basis is the crucial point in our approach, one should not forget that it is only the first step in the solving process. Methods to implement change of ordering of the Gröbner basis, and alternative approaches like triangular systems and rational univariate representation of the system are also key tools. We will discuss some of these methods in the following. For previous applications of Gröbner bases to the design of wavelets and digital filters, see for example the works of Park et al. (1997), Faugère et al. (1998), Lebrun (2000), Lebrun and Vetterli (2001), Selesnick and Burrus (1998), Selesnick (1999) and Selesnick (2000b).

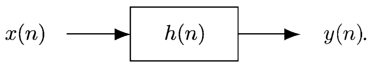
Following a filter bank perspective, we introduce filter banks based on conjugate quadrature filters (CQFs), and we give a simple introduction to wavelets. Iteration of the filter bank on the lowpass analysis generates discrete-time wavelet bases. In the limit, we end up with wavelet bases and the concept of multiresolution analysis. We also highlight the motivation for introducing multiwavelets as a way to overcome some limitations of CQFs. Readers interested in a more detailed presentation of filter bank and wavelet theory are referred to the classical books of Daubechies (1992), Vaidyanathan (1993), Vetterli and Kovačević (1995), Strang and Nguyen (1996), Burrus et al. (1998) and Mallat (1998).

1. Preliminaries

The Z-transform of a discrete-time signal, defined as

$$X(z) = \mathcal{Z}\{x(n)\} = \sum_n x(n) z^{-n}$$

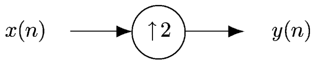
will be used often in the development that follows. A filter will be represented by its impulse response $h(n)$ or transfer function $H(z) = \mathcal{Z}\{h(n)\}$.



The output of the filter is the convolution of the input with the impulse response $h(n)$.

$$y(n) = h(n) * x(n) := \sum_k x(k)h(n-k),$$

or equivalently, $Y(z) = H(z)X(z)$. The *upsampler*, represented by the diagram



is defined by the relation

$$y(n) = \begin{cases} x(n/2) & \text{for } n \text{ even} \\ 0 & \text{for } n \text{ odd.} \end{cases} \tag{1}$$

The usual notation is $y(n) = [\uparrow 2]x(n)$. The upsampler simply inserts zeros between samples. For example, if $x(n)$ is the sequence

$$\{\ldots, 3, \underline{5}, 2, 9, 6, \ldots\}$$

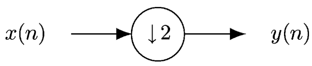
where the underlined number represents $x(0)$, then $y(n)$ is given by

$$\{\ldots, 0, 3, 0, \underline{5}, 0, 2, 0, 9, 0, 6, 0, \ldots\}.$$

In terms of the Z-transform, we have

$$y(n) = [\uparrow 2]x(n) \Longleftrightarrow Y(z) = X(z^2). \tag{2}$$

The discrete-time Fourier transform of $y(n)$ is given by $Y(e^{j\omega}) = X(e^{j2\omega})$. Or using the notation $\widehat{y}(\omega) = Y(e^{j\omega})$, $\widehat{x}(\omega) = x(e^{j\omega})$, we have $\widehat{y}(\omega) = \widehat{x}(2\omega)$. The *downsampler*, represented by the diagram,



is defined as $y(n) = x(2n)$. The usual notation is $y(n) = [\downarrow 2]x(n)$. The downsampler simply keeps every second sample, and discards the others. For example, if $x(n)$ is the sequence

$$\{\ldots, 7, 3, \underline{5}, 2, 9, 6, 4, \ldots\}$$

where the underlined number represents $x(0)$, then $y(n)$ is given by

$$\{\ldots, 7, \underline{5}, 9, 4, \ldots\}.$$

In terms of the Z-transform, we have

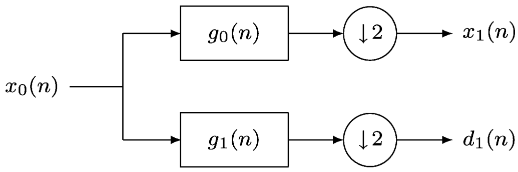
$$y(n) = [\downarrow 2]x(n) \Longleftrightarrow Y(z) = \tfrac{1}{2}(X(z^{1/2}) + X(-z^{1/2})) \tag{3}$$

and $\widehat{y}(\omega) = \frac{1}{2}(\widehat{x}(\frac{\omega}{2}) + \widehat{x}(\frac{\omega-2\pi}{2}))$. This operation induces aliasing in the frequency domain.

1.1. Filter banks

The basic principle of filter banks is to decompose signals into lowpass and highpass components at half the rate of the input signal (so as to keep the same amount of data) in such a way that it is possible to exactly reconstruct the input signal from these components. This subject of interest, *subband coding with multirate filter banks*, became an active topic when Croisier et al. (1976) showed it was possible to construct filter banks with aliasing cancellation using quadrature mirror filters (QMF) and simple downsampling and upsampling operations.

An analysis filter bank decomposes a signal x_0 into two subband signals x_1 and d_1 as shown in the diagram.



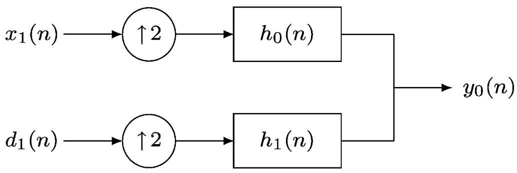
Consequently, a two-channel multirate filter bank first convolves the input signal x_0 with a lowpass filter g_0 and a highpass filter g_1 to minimize aliasing and then downsamples these two signals.

$$x_1 = [\downarrow 2](g_0 * x_0) \qquad \text{and} \qquad d_1 = [\downarrow 2](g_1 * x_0). \tag{4}$$

Explicitly, we have

$$x_1(n) = \sum_k x_0(k) g_0(2n-k) \qquad \text{and} \qquad d_1(n) = \sum_k x_0(k) g_1(2n-k). \tag{5}$$

A synthesis filter bank combines the subband signals into a single signal.



The output signal is then reconstructed by upsampling x_1 and d_1 and filtering again with a lowpass filter h_0 and a highpass filter h_1 to reject the out-of-band components in the spectrum. The synthesis is given by

$$y_0 = h_0 * ([\uparrow 2] x_1) + h_1 * ([\uparrow 2] d_1). \tag{6}$$

Explicitly, we have

$$y_0(n) = \sum_k x_1(k) h_0(n-2k) + \sum_k d_1(k) h_1(n-2k). \tag{7}$$

A perfect reconstruction (PR) filter bank is one where the synthesis filter bank perfectly reconstructs the input x_0 from the subband signals x_1 and d_1; that is, one where $y_0 = x_0$.

For a PR filter bank, the synthesis filters h_0, h_1 together with their translates by even integers form a basis for $l^2(\mathbb{Z})$ and (7) can be written as

$$x_0(n) = \sum_k \langle x_0, \tilde{h}_{0,k} \rangle h_{0,k}(n) + \sum_k \langle x_0, \tilde{h}_{1,k} \rangle h_{1,k}(n) \tag{8}$$

where

$$h_{i,k}(n) = h_i(n-2k) \qquad \text{and} \qquad \tilde{h}_{i,k}(n) = g_i(2k-n) \qquad \text{for } i = 0, 1. \tag{9}$$

The dual basis $\{\tilde{h}_{i,k}\}$ is comprised of the reversed versions of the analysis filter $g_i(n)$ and their translates by even integers.

Vetterli gave the necessary and sufficient conditions in the z-domain for PR (in fact generalized PR, since delays of the form z^{-l} are allowed):

$$G_0(z)H_0(z) + G_1(z)H_1(z) = 2z^{-l} \tag{10}$$

$$G_0(z)H_0(-z) + G_1(z)H_1(-z) = 0. \tag{11}$$

For filters satisfying these PR conditions, the signal x_0 can be recovered from x_1 and d_1. In this case the subband signals provide an alternate representation of the input signal x_0. The goal is to design the four filters such that the filter bank is PR and such that the new representation of x_0 is more efficient and thus facilitates signal processing tasks. Although the PR conditions do not demand it, applications of the subband decomposition (Crochiere et al., 1976) generally call for the filters h_0 and g_0 to be lowpass, and the filters h_1 and g_1 to be highpass so that x_1 and d_1 have more or less disjoint spectrums.

From the PR conditions, we get that g_0 and g_1 are uniquely determined from h_0 and h_1 by rewriting the previous equations

$$\begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{bmatrix} \begin{bmatrix} G_0(z) \\ G_1(z) \end{bmatrix} = \begin{bmatrix} 2z^{-l} \\ 0 \end{bmatrix}. \tag{12}$$

Introducing $D(z) := H_0(z)H_1(-z) - H_0(-z)H_1(z)$, supposed to be non-vanishing on $\mathbb{T}$, we get

$$G_0(z) = \frac{2z^{-l}}{D(z)} H_1(-z) \qquad \text{and} \qquad G_1(z) = -\frac{2z^{-l}}{D(z)} H_0(-z). \tag{13}$$

Now, if we require further that all filters have finite impulse response (FIR, i.e. a finite number of taps), then essentially only two choices are possible for $D(z)$ and z^{-l}. Namely,

Quadrature mirror (QMF). $D(z) = 2z^{-l}$.
This gives $G_0(z) = H_1(-z)$ and $G_1(z) = -H_0(-z)$. Now, Croisier et al. (1976) additionally imposed h_0 and h_1 to be mirror filters ($H_1(z) := H_0(-z)$), we then get

$$H_0^2(z) - H_0^2(-z) = 2z^{-l} \tag{14}$$

where l is necessarily odd. The solutions of this equation are naturally called QMF. Unfortunately, the only solutions being FIR QMF are variations of the Haar filter $H_0(z) = 1/\sqrt{2}(1 + z^{-1})$. The interest of these filters is rather limited.

Conjugated quadrature filters (CQF). $D(z) = 2z^{-l-1}$.
We get $G_0(z) = zH_1(-z)$ and $G_1(z) = -zH_0(-z)$. Smith and Barnwell (1984) and Mintzer (1985) were able to overcome the major limitation of QMF by imposing h_0 and h_1 to be CQFs: $H_1(z) := -z^{-1}H_0(-z^{-1})$. We get

$$H_0(z)H_0(z^{-1}) + H_0(-z)H_0(-z^{-1}) = 2z^{-l}, \tag{15}$$

where l is necessarily even. With this slight change, FIR solutions are now possible. And as we will see, these filters are closely linked to wavelets.

1.2. Zero moments

The filter bank provides an efficient representation of piece-wise smooth signals if the subband signal d_1 is close to zero for smooth signals x_0 and if the filter h_1 is short. That is, for signal compression, we want $d_1 \approx 0$ whenever x_0 is smooth. As a vehicle for achieving this, it is common to ask that $d_1 = 0$ whenever x_0 is a discrete-time polynomial of specified degree.

It can be shown that the filter $G_1(z)$ annihilates polynomial signals of degree $K - 1$ if and only if $(z-1)^K$ divides $G_1(z)$, or equivalently, if

$$\sum_n n^k(-1)^n g_1(n) = 0, \qquad \text{for } 0 \le k \le K-1.$$

That is, the filter g_1 has K *zero moments*.

1.3. Orthonormal filter banks

If the analysis filters g_0, g_1 of a PR filter bank are related to synthesis filter h_0, h_1 by a time-reversal,

$$g_0(n) = h_0(-n), \qquad g_1(n) = h_1(-n),$$

or equivalently

$$G_0(z) = H_0(z^{-1}), \qquad G_1(z) = H_1(z^{-1}),$$

then the filter bank is said to be an *orthonormal filter bank*. Orthonormal filter banks have desirable statistical properties. In this case, the PR conditions become

$$H_0(z)H_0(z^{-1}) + H_1(z)H_1(z^{-1}) = 2 \tag{16}$$

$$H_0(-z)H_0(z^{-1}) + H_1(-z)H_1(z^{-1}) = 0. \tag{17}$$

It is easily verified that setting h_1 to the CQF filter of h_0, i.e. $h_1(n) = (-1)^n h_0(1-n)$ or equivalently $H_1(z) = z^{-1}H_0(-z^{-1})$, it supplies a solution to the second of the two PR conditions. With this form for $H_1(z)$, the first PR condition becomes

$$H_0(z)H_0(z^{-1}) + H_0(-z)H_0(-z^{-1}) = 2, \tag{18}$$

or equivalently

$$\sum_n h_0(n)h_0(n-2k) = 2\delta(k) = \begin{cases} 2 & k = 0 \\ 0 & k \neq 0. \end{cases}$$

For an orthonormal filter bank, all the filters are determined by $H_0(z)$:

$$H_1(z) := -z^{-1}H_0(-z^{-1}), \qquad G_0(z) = H_0(z^{-1}), \qquad G_1(z) = zH_0(-z).$$

Moreover, the decomposition of a signal x_0 by an orthonormal filter bank may be interpreted as its expansion in an orthonormal basis of ℓ^2. Namely the expansion (8) is an orthonormal one. This result can be generalized to IIR (non-FIR) filter banks. Besides, if the filters are not imposed to be CQF then we get biorthogonal bases of ℓ^2 (Vetterli and Herley, 1992).

Note that if $(z-1)^K$ divides $G_1(z)$, then $G_1(1) = 0$, and so for an orthonormal filter bank we have $H_0(-1) = 0$. Substituting $z = 1$ in (18) gives

$$H_0^2(1) = 2. \tag{19}$$

Furthermore, if $(z-1)^K$ divides $G_1(z)$, then $(z+1)^K$ divides $H_0(z)$. Daubechies' problem is the following. Given K, find $H(z)$ of minimal degree such that

1. $\sum_n h(n)h(n-2k) = 2\delta(k)$
2. $(z+1)^K$ divides $H(z)$.

It turns out that the solutions $h(n)$ of minimal degree can be most conveniently found by defining a product filter $P(z)$. Letting $P(z) := H(z)H(z^{-1})$, we have the requirement that $P(z) + P(-z) = 2$. Hence, orthonormal filter banks can be obtained by designing $P(z)$ satisfying this *linear* condition. Because $P(e^{j\omega}) = |H(e^{j\omega})|^2$, $P(z)$ must be non-negative for all $z = e^{j\omega}$, otherwise it does not admit the factorization $P(z) = H(z)H(z^{-1})$. Also note that if $(z+1)^K$ divides $H(z)$, then $(z+1)^{2K}$ divides $P(z)$.

Gathering these conditions together gives an alternate form of Daubechies' design problem: given K, find $P(z)$ of minimal degree such that

1. $P(e^{j\omega}) \geq 0$ for all ω
2. $P(z) = P(z^{-1})$
3. $P(z) + P(-z) = 2$
4. $(z+1)^{2K}$ divides $P(z)$.

The solution is given by

$$P(z) = 2(1-y)^K \sum_{k=0}^{K-1} \binom{K+k-1}{k} y^k \tag{20}$$

where $z = e^{j\omega}$ and $y = 1/2(1-\cos\omega)$.

The key to this solution is (1) that all constraints on $H(z)$ can be converted into linear constraints on $P(z)$, and (2) that $H(z)$ can be obtained from $P(z)$ by the Féjer–Riesz theorem and spectral factorization. For other design problems where additional constraints are imposed, it is not possible to convert the constraints on $H(z)$ into linear constraints on $P(z)$. It is in those cases that Gröbner bases can be used to investigate the existence of solutions having various desired properties.

2. Daubechies solution

To illustrate the Gröbner basis-design of orthonormal filter banks we begin by showing an example of the design of Daubechies filters of length 8. Although they can be obtained through simpler means, it is a good example with which to begin.

Let $R(z)$ be the remainder obtained after dividing $H(z)$ by $(z+1)^K$. Then the requirement that $(z+1)^K$ divides $H(z)$ can be written as $R(z) = 0$. To simplify the notation, we denote $h(n)$ by h_n. When $K = 4$, the minimal length $h(n)$ that satisfies the orthonormality condition is of length 8 (the degree of $H(z)$ is 7). For $K = 4$, length 8, the design equations for $h(n)$ are:

$$
\begin{aligned}
&\text{// Orthonormality conditions}\\
&h_0^2+h_1^2+h_2^2+h_3^2+h_4^2+h_5^2+h_6^2+h_7^2-2=0\\
&h_7h_5+h_6h_4+h_2h_0+h_3h_1+h_4h_2+h_5h_3=0\\
&h_6h_2+h_7h_3+h_4h_0+h_5h_1=0\\
&h_6h_0+h_7h_1=0.\\
&\text{// Zero-moment conditions}\\
&h_0-h_4+20h_7-10h_6+4h_5=0\\
&h_1+15h_5+70h_7-36h_6-4h_4=0\\
&h_2-45h_6+84h_7+20h_5-6h_4=0\\
&h_3+35h_7-20h_6+10h_5-4h_4=0.
\end{aligned}
$$

Note that the first condition $\sum_n h_n^2 = 2$ is the only non-homogeneous equation. We can replace it by the equation $H^2(1) = 2$ without affecting the set of solutions. In addition, the negation of each solution vector is also a solution (if h_n is a solution, then so is $-h_n$). Therefore, we can reduce the number of solutions by replacing the constraint $H^2(1) = 2$ with the constraint $H(1) = +\sqrt{2}$. To simplify the Gröbner basis calculations, we can replace the equation $H(1) = +\sqrt{2}$ by $H(1) = 1$, then the equations are in terms of rationals only. This has the effect only of scaling all solution vectors by $1/\sqrt{2}$. The solution can be rescaled afterwards to obtain the correct normalization. This procedure reduces the degree of the set of equations by a factor of two.

If the first equation above is replaced by the equation

$$h_0+h_1+h_2+h_3+h_4+h_5+h_6+h_7-1=0$$

then lexicographic Gröbner basis is

$$
\begin{aligned}
&281474976710656\,h_7^8-17592186044416\,h_7^7-3848290697216\,h_7^6-150323855360\,h_7^5+3523215360\,h_7^4+183500800\,h_7^3-5734400\,h_7^2+32000\,h_7+625\\
&78400\,h_6-6597069766656\,h_7^7+481036337152\,h_7^6+93952409600\,h_7^5+1996488704\,h_7^4-233963520\,h_7^3-5877760\,h_7^2+229600\,h_7+125\\
&392000\,h_5+116548232544256\,h_7^7-6253472382976\,h_7^6-1537061421056\,h_7^5-85144371200\,h_7^4-811991040\,h_7^3+52326400\,h_7^2+1797600\,h_7+1875\\
&196000\,h_4+290271069732864\,h_7^7-17798344474624\,h_7^6-3949759299584\,h_7^5-162655109120\,h_7^4+2876375040\,h_7^3+181350400\,h_7^2-3869600\,h_7-11625\\
&196000\,h_3+248489627877376\,h_7^7-15874199126016\,h_7^6-3416109613056\,h_7^5-125074145280\,h_7^4+3867279360\,h_7^3+169881600\,h_7^2-6126400\,h_7-49625\\
&392000\,h_2-332052511588352\,h_7^7+19722489823232\,h_7^6+4483408986112\,h_7^5+200236072960\,h_7^4-1885470720\,h_7^3-192819200\,h_7^2+2200800\,h_7-148875\\
&392000\,h_1-613527488299008\,h_7^7+38001870635008\,h_7^6+8369280647168\,h_7^5+335292661760\,h_7^4-6922567680\,h_7^3-392089600\,h_7^2+10847200\,h_7-98625\\
&2000\,h_0-1099511627776\,h_7^7+68719476736\,h_7^6+15032385536\,h_7^5+587202560\,h_7^4-13762560\,h_7^3-716800\,h_7^2+22400\,h_7-125.
\end{aligned}
$$

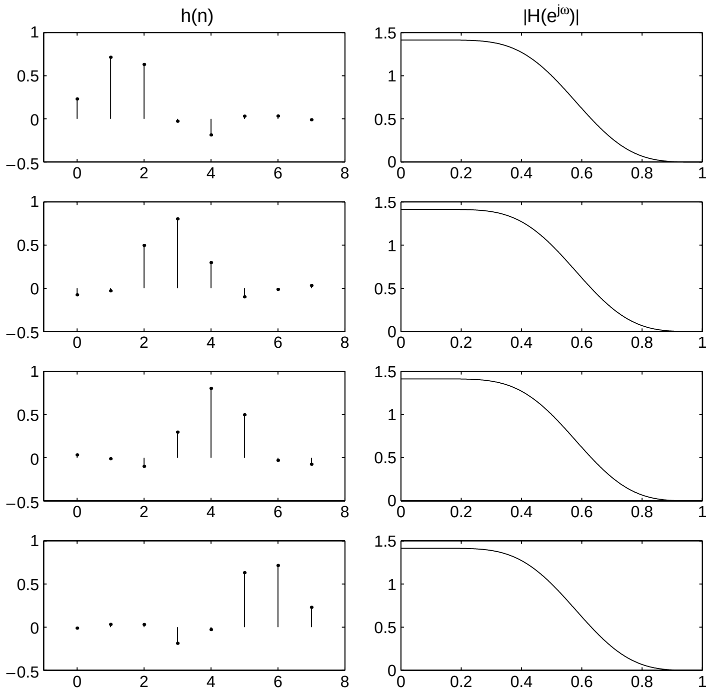


Fig. 1. The 4 orthonormal filters of length 8 with 4 zero moments.

Appending the following equation

$$h_1 + 2h_2 + 3h_3 + 4h_4 + 5h_5 + 6h_6 + 7h_7 - A \tag{21}$$

yields a more compact Gröbner basis:

$$\begin{array}{l}
4A^8 - 112A^7 + 1344A^6 - 9016A^5 + 36904A^4 - 94080A^3 + 145096A^2 - 122500A + 42385 \\
4480h_7 - 16A^7 + 392A^6 - 4004A^5 + 22050A^4 - 70476A^3 + 130074A^2 - 126910A + 48965 \\
4480h_6 - 16A^7 + 392A^6 - 4004A^5 + 22050A^4 - 70476A^3 + 129934A^2 - 126210A + 48265 \\
4480h_5 + 48A^7 - 1176A^6 + 12012A^5 - 66150A^4 + 211428A^3 - 390502A^2 + 381850A - 147735 \\
4480h_4 + 48A^7 - 1176A^6 + 12012A^5 - 66150A^4 + 211428A^3 - 390082A^2 + 379190A - 144795 \\
4480h_3 - 48A^7 + 1176A^6 - 12012A^5 + 66150A^4 - 211428A^3 + 390782A^2 - 384090A + 149695 \\
4480h_2 - 48A^7 + 1176A^6 - 12012A^5 + 66150A^4 - 211428A^3 + 390362A^2 - 380870A + 144795 \\
4480h_1 + 16A^7 - 392A^6 + 4004A^5 - 22050A^4 + 70476A^3 - 130354A^2 + 129150A - 53165 \\
4480h_0 + 16A^7 - 392A^6 + 4004A^5 - 22050A^4 + 70476A^3 - 130214A^2 + 127890A - 50505.
\end{array}$$

The lexicographic Gröbner basis can be obtained from the degree-Gröbner basis using the FGLM algorithm, as described below in the Appendix.

Of the eight solutions, four are real-valued, four are complex-valued. The four real-valued solutions are shown in Fig. 1. Notice that the reverse of each solution is also a solution. Not counting negation and reversal, there are two distinct solutions.

As noted above, the Daubechies filters can be obtained via the spectral factorization of a suitably designed (Laurent) polynomial $P(z)$, as described by Daubechies. In this procedure, Gröbner bases are not required, as the design of $P(z)$ is a linear problem and spectral factorization requires finding the roots of a univariate polynomial only. However, if it is desired that the filter $h(n)$ satisfy additional constraints, it is likely that the spectral factorization approach cannot be used. For example, if it is desired that $h(n)$ be nearly symmetric $h(n_o - n) \approx h(n)$ then the design problem becomes more complicated and Gröbner bases can be utilized. (For image processing, it is desirable that a filter bank consist of symmetric filters because the distortion introduced by filtering with non-symmetric filters is sometimes visible.) It is well-known that exactly symmetric finite-length solutions do not exist for the orthonormal two-channel filter bank design problem, for the exception of the Haar solution with $H(z) = \frac{1}{\sqrt{2}}(1 + z^{-1})$.

For this reason, it is common to use (1) orthonormal PR filter banks with nearly symmetric filters or (2) symmetric PR filter banks that are nearly orthonormal. While the design of these types of filter banks can be approximately carried out by different factorizations of $P(z)$ in (20), many algorithms have been suggested for these two classes that design $h(n)$ directly rather than $p(n)$.

The design of nearly symmetric orthonormal PR filter banks is described in the next section, where Gröbner bases are used to obtain the filters. An alternative is to use multiwavelets, for which orthonormality and symmetry are simultaneously possible. The design of multiwavelets is detailed in Sections 5 and 6.

3. Nearly symmetric orthonormal filter bank

While the classic Daubechies filters can be obtained without having to solve any multivariate nonlinear equations, many generalizations and specialized designs that satisfy additional constraints cannot be obtained so easily. As an example, consider the design of a length 8 filter $h(n)$ satisfying the orthonormality condition (18), with some zero moments and some degree of symmetry (Abdelnour and Selesnick, 2001). To enforce a degree of symmetry, we ask that

$$h(n_o + n) = h(n_o - n)$$

for some selected range of n. If there were no symmetry constraints, then the filter bank could have at most four zero moments. Because of the symmetry constraints, the filter bank will have fewer zero moments. Taking $K = 2$, $n_o = 2.5$, we can get the following design problem. Design $H(z)$ of minimal degree such that,

1. $h(2) = h(3)$, $h(1) = h(4)$.
2. $(z + 1)^2$ divides $H(z)$.
3. $\sum_n h_n h_{n-2k} = \delta(k)$.

This design problem gives rise to the following design equations.

// Orthonormality conditions

$$h_0 + h_1 + h_2 + h_3 + h_4 + h_5 + h_6 + h_7 - 1$$
$$h_2h_0 + h_3h_1 + h_4h_2 + h_5h_3 + h_6h_4 + h_7h_5$$
$$h_6h_2 + h_4h_0 + h_5h_1 + h_7h_3$$
$$h_6h_0 + h_7h_1.$$

// Zero-moment conditions

$h_0 - h_2 - 3h_4 - 5h_6 + 6h_7 + 4h_5 + 2h_3$

$h_1 + 3h_3 + 5h_5 + 7h_7 - 6h_6 - 4h_4 - 2h_2.$

// Partial symmetry conditions

$h_2 - h_3$

$h_1 - h_4.$

As above, appending Eq. (21) has the effect of simplifying the coefficients appearing in the Gröbner basis. It turns out that the lexicographic Gröbner basis then factors into two parts. We used the `facstd` command in *Singular* (Greuel et al., 2000) to perform the factorization. The first Gröbner basis is

$$
\begin{aligned}
&40A^6 - 984A^5 + 9796A^4 - 49888A^3 + 135314A^2 - 183246A + 95445,\\
&106688h_7 + 800A^5 - 18060A^4 + 156848A^3 - 646136A^2 + 1231488A - 828755,\\
&320064h_6 - 800A^5 + 18060A^4 - 156848A^3 + 646136A^2 - 1284832A + 962115,\\
&26672h_5 - 400A^5 + 9030A^4 - 78424A^3 + 321401A^2 - 607409A + 405209,\\
&32h_4 - 2A^2 + 18A - 35,\\
&106688h_3 + 800A^5 - 18060A^4 + 156848A^3 - 632800A^2 + 1138136A - 728735,\\
&106688h_2 + 800A^5 - 18060A^4 + 156848A^3 - 632800A^2 + 1138136A - 728735,\\
&32h_1 - 2A^2 + 18A - 35,\\
&80016h_0 - 400A^5 + 9030A^4 - 78424A^3 + 318067A^2 - 577403A + 353532.
\end{aligned}
$$

The second Gröbner basis is

$$
\begin{aligned}
&2A^2 - 18A + 33,\\
&16h_7 - 2A + 5,\\
&16h_6 - 2A + 5,\\
&16h_5 + 1,\\
&16h_4 - 1,\\
&16h_3 + 2A - 13,\\
&16h_2 + 2A - 13,\\
&16h_1 - 1,\\
&16h_0 + 1.
\end{aligned}
$$

The first part has four real-valued solutions and two complex-valued solutions. The second part has two real-valued solutions. The six real solutions are shown in Fig. 2. The frequency responses $|H(e^{j\omega})|$ are also shown in the figure. Only the last solution is a reasonable lowpass filter. The other five solutions can be considered parasitic solutions. They would not be favored in practice because they do not have acceptable lowpass frequency responses.

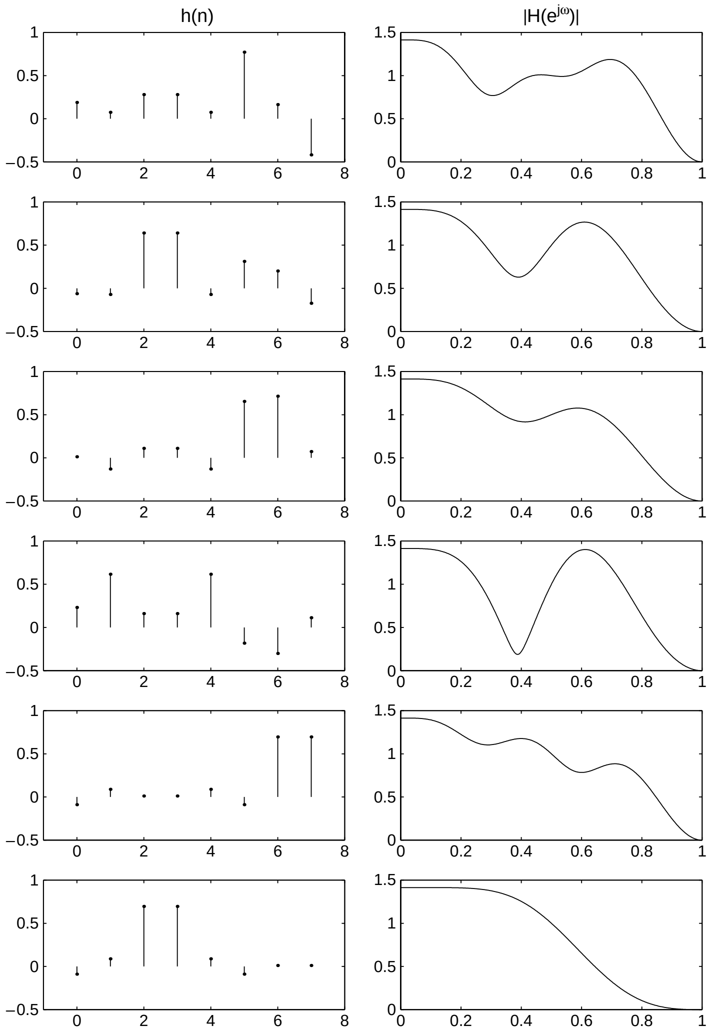


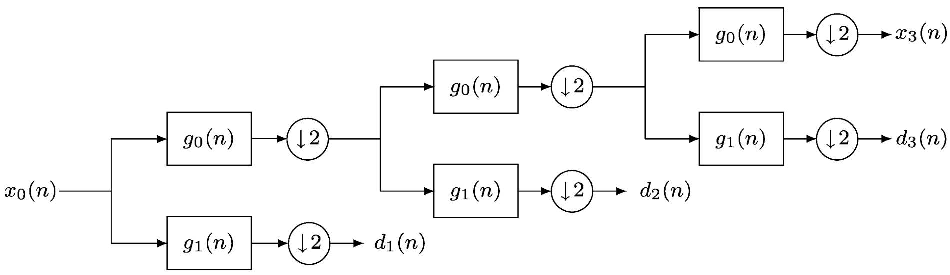
Fig. 2. The 6 orthonormal filters of length 8 with 2 zero moments and partial symmetry about $n_0 = 2.5$.

It is seen in the figure, that the sixth solution, while not exactly symmetric, is more symmetric than the solutions shown in Fig. 1. Furthermore, the solution has more symmetry than requested in the design problem; we have $h(0) = h(5)$ as well.

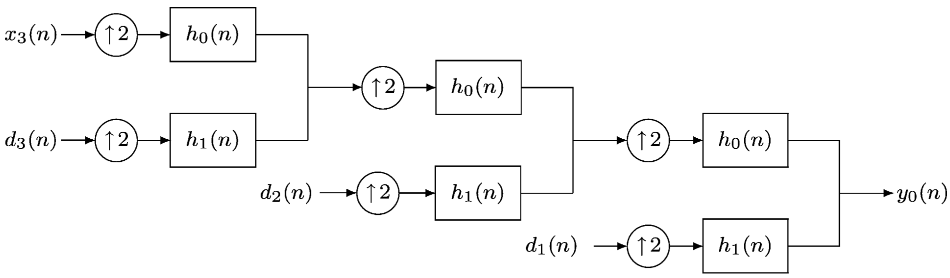
Other formulations of the nearly symmetric orthonormal filter bank design problem are based on moments of $h(n)$, and the corresponding wavelets are called *Coiflets* (Daubechies, 1992; Tian et al., 1997; Wei and Bovik, 1998). The design of Coiflets also requires the solution to nonlinear design equations and usually the solutions are found through iterative numerical optimization. As detailed in Section 6.2, Gröbner bases can also be used to obtain Coiflets (in fact multiCoiflets).

4. Iterated filter banks

The filter bank structure described above is often used in an iterated manner. Indeed, the analysis of a signal over several scales (multiresolution analysis) can be accomplished by iterating the filter bank on the first subband. The idea of filter bank trees is to cascade this iteration up to a certain level l. We then have $l + 1$ signals: the coarse signal x_l and the details signals $d_l, \ldots, d_1$.



The original signal x_0 can be reconstructed from these subband signals by the iterated synthesis filter bank.



If $(z-1)^K$ divides $G_1(z)$, then not only is $d_1 = 0$ whenever x_0 is a polynomial signal of degree less than K, but $d_2 = 0$ and $d_3 = 0$ also. This is clarified as follows. Let $\mathcal{P}_K$ denote the set of discrete-time polynomials of degree K and less; then we can write the following. If

1. $x_0(n) \in \mathcal{P}_{K-1}$, and
2. $(z - 1)^K$ divides $G_1(z)$

then

1. $x_1(n) \in \mathcal{P}_{K-1}$, and
2. $d_1(n) = 0$.

Note that polynomial signals are preserved; if x_0 is a polynomial signal, then so is x_1. Therefore, if $G_1(z)$ annihilates polynomials of a specified degree K, then *all* of the subbands d_i are zero whenever the input is a polynomial of the same degree.

Now, if we omit some detail signals, $d_i(n)$, in the reconstruction (this is the principle of compression), the "quality" of the signal reconstructed will depend largely on the "smoothness" (Mallat, 1989; Daubechies, 1992; Rioul, 1993) of the basis vectors with which the reconstruction is performed.

4.1. Wavelet bases

The transformation of a signal x_0 by an l-level iterated filter bank into subband signals $d_1, d_2, \ldots, d_l, x_l$ constitutes the DWT. A wavelet basis for $L_2(\mathbb{R})$ is closely related to the DWT. In particular, given an orthonormal DWT (fully determined by $h_0(n)$), an orthonormal wavelet basis for $L_2(\mathbb{R})$ is given by

$$\{\phi(t-k),\ 2^{j/2}\psi(2^j t-k) : j,k \in \mathbb{Z},\ j \geq 0\}$$

where the scaling function $\phi(t)$ is defined through the dilation equation (or two-scale relation):

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

and the wavelet is defined by

$$\psi(t) := \sqrt{2}\sum_n h_1(n)\phi(2t-n).$$

Furthermore, if $(z+1)^K$ divides $H_0(z)$ then

$$\int t^k \psi(t)\,\mathrm{d}t = 0 \qquad \text{for } 0 \leq k \leq K-1$$

and

$$\sum_n n^k \phi(t-k) \in \mathcal{P}_k.$$

Therefore, the design of an orthonormal wavelet basis for $L_2(\mathbb{R})$ is equivalent to the design of an orthonormal filter bank. Implementation is nearly always performed using filter banks, but the functions $\phi(t)$ and $\psi(t)$ are useful because they indicate how the filter bank behaves when the filter bank is iterated indefinitely. For example, if the filter bank is not designed so that $(z+1)^2$ divides $H_0(z)$, then $\phi(t)$ will not be continuous. The smoothness of $\phi(t)$ is important because it reflects what artifacts may appear in the synthesized signal $y(n)$.

The scaling functions $\phi(t)$ for the examples above are shown in Figs. 3 and 4, from which the comparative symmetry of the second problem is also visible.

It should be noted that a solution (in the L^2 sense) to the dilation equation (a scaling function) does not always exist. However, if $\inf_{|\omega|<\frac{\pi}{3}} |\widehat{h}_0(\omega)| > 0$, then the convergence is in L^2 norm (Cohen, 1992) to a bona-fide L^2 function. In that case, these two functions generate a multiresolution analysis of L^2 as defined by Mallat (1989). Defining $V_k := \mathrm{span}\{\phi(2^{-k}t-n) \mid n \in \mathbb{Z}\}$, we get by the two-scale equations, a nested sequence of subspaces of L^2 satisfying

- $V_n \subset V_{n-1}$.
- $\cap_n V_n = \{0\}$ and $\overline{\cup_n V_n} = L^2$.

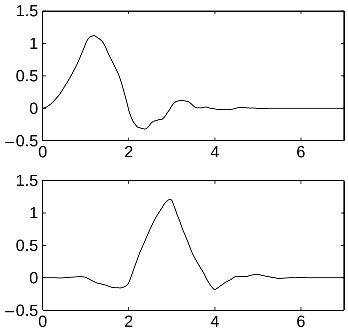


Fig. 3. The scaling functions generated by the first two filters shown in Fig. 1.

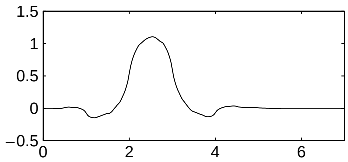


Fig. 4. The scaling function generated by the sixth filter shown in Fig. 2.

- $f \in V_n \Leftrightarrow f(2t) \in V_{n-1}$.
- $f \in V_0 \Leftrightarrow f(t-k) \in V_0, \forall k \in \mathbb{Z}$.
- $\{\phi(t-k) \mid k \in \mathbb{Z}\}$ is an orthonormal basis of V_0.

Introducing $W_k := \text{span}\{\psi(2^{-k}t - n) \mid n \in \mathbb{Z}\}$, we get $V_{n-1} = V_n \oplus W_n$ and so $\oplus_n W_n = L^2$. It is easily proven that $\{\psi(2^k t - n) \mid k, n \in \mathbb{Z}\}$ is an orthonormal basis of L^2. Starting from a CQF, we have constructed a basis of L^2 from dyadic dilations and translations of a single function. ψ is called an orthonormal wavelet, ϕ is called the associated scaling function. Again, the *quality* of the multiresolution analysis is measured by the number K of zeros at π of $H_0(e^{j\omega})$ since it implies that $1, t, \ldots, t^{K-1}$ can be exactly reconstructed from integer translates of the scaling function, thus giving approximation order K (Jia and Lei, 1993).

This way of constructing wavelets from iterated filter banks was pioneered by Daubechies (1988). It became since, a standard way to derive orthonormal and bi-orthogonal wavelet bases. The underlying CQF filter banks are now well-studied, the design procedure is well-understood. By the structure of the problem, certain solutions are however ruled out: since it is impossible to design FIR linear-phase CQF with real coefficients other than the Haar filter, this implies that the only orthonormal wavelet with compact support and symmetry is the Haar wavelet.

For multiwavelets, however, the relation between $\phi(t)$, $\psi(t)$ and the corresponding filter bank is more complicated. In the next section, the design of multiwavelets is considered in detail. It turns out that Gröbner bases are very useful in investigating the existence of multiwavelets having properties that are not possible in the scalar-wavelet framework.

5. Multiwavelets

Generalizing the wavelet case, one can allow a multiresolution analysis $\{V_n\}_{n\in\mathbb{Z}}$ of $L^2(\mathbb{R})$ to be generated by a finite orthonormal set of scaling functions $\phi_0(t)$, $\phi_1(t), \ldots, \phi_{r-1}(t)$ and their integer translates. In this framework, the so-called multiscaling function $\boldsymbol{\phi}(t) := [\phi_0(t), \ldots, \phi_{r-1}(t)]^\top$ satisfies now a matrix two-scale equation

$$\boldsymbol{\phi}(t) = \sum_k \mathbf{M}(k)\boldsymbol{\phi}(2t-k) \tag{22}$$

where now $\{\mathbf{M}(k)\}_k$ is a sequence of $r \times r$ matrices of real coefficients. The multiresolution analysis structure gives $V_{-1} = V_0 \oplus W_0$ where W_0 is the orthogonal complement of V_0 in V_{-1}. Again, starting from the orthonormal basis, $\phi_0(t), \phi_1(t), \ldots, \phi_{r-1}(t)$ and their integer translates, we can construct an orthonormal basis of W_0 generated by $\psi_0(t), \psi_1(t), \ldots, \psi_{r-1}(t)$ and their integer translates with the so-called multiwavelet $\boldsymbol{\psi}(t) := [\psi_0(t), \ldots, \psi_{r-1}(t)]^\top$ derived by

$$\boldsymbol{\psi}(t) := \sum_k \mathbf{N}(k)\boldsymbol{\phi}(2t-k) \tag{23}$$

where $\{\mathbf{N}(k)\}_k$ is a sequence of $r \times r$ matrices of real coefficients obtained by orthonormal *completion* (Lawton et al., 1996) of $\{\mathbf{M}(k)\}_k$. Introducing in the z-domain, the refinement masks $\mathbf{M}(z) := 1/2\sum_n \mathbf{M}(n)z^{-n}$ and $\mathbf{N}(z) := 1/2\sum_n \mathbf{N}(n)z^{-n}$, Eqs. (22) and (23) translate in Fourier domain into

$$\widehat{\boldsymbol{\phi}}(2\omega) = \mathbf{M}(\mathrm{e}^{j\omega})\widehat{\boldsymbol{\phi}}(\omega) \qquad \text{and} \qquad \widehat{\boldsymbol{\psi}}(2\omega) = \mathbf{N}(\mathrm{e}^{j\omega})\widehat{\boldsymbol{\phi}}(\omega). \tag{24}$$

Under some natural conditions of convergence (detailed in Cohen et al. (1997) and Lebrun (2000)), and similarly to the wavelet case, we can derive the behavior of the multiscaling function by iterating the first product above. We get at the limit

$$\widehat{\boldsymbol{\phi}}(\omega) = \mathbf{M}_\infty(\omega)\widehat{\boldsymbol{\phi}}(0) = \prod_{i=1}^{\infty} \mathbf{M}(\mathrm{e}^{j\omega/2^i})\widehat{\boldsymbol{\phi}}(0). \tag{25}$$

In the sequel, we will impose that the sequences $\{\mathbf{M}(k)\}_k$ and $\{\mathbf{N}(k)\}_k$ have finite support and thus that $\boldsymbol{\phi}(t)$ and $\boldsymbol{\psi}(t)$ have compact support (Massopust et al., 1996). The orthonormality of the multiscaling function translates also into a matrix orthonormality condition on $\mathbf{M}(z)$: for all z on the unit circle,

$$\mathbf{M}(z)\mathbf{M}^\top(z^{-1}) + \mathbf{M}(-z)\mathbf{M}^\top(-z^{-1}) = \mathbf{I}. \tag{26}$$

With this approach, one is finally able to overcome some of the limitations of CQF filter banks. It is now possible to get finitely generated multiresolution analysis with all the scaling functions and wavelets being orthogonal, compactly supported and (anti)symmetric.

The first multiwavelets were designed by Alpert (1993) using methods from numerical analysis (finite elements and splines methods). A construction using fractal interpolation of a multiresolution analysis having approximation order 2 (1 and t can be reconstructed from the translates of the scaling functions) using two symmetric, compactly supported, orthogonal scaling functions (that are furthermore Lipschitz) by Geronimo et al. (1994) (DGHM) triggered many other attempts to construct new multiwavelet bases (Vetterli and Strang, 1994; Strang and Strela, 1995; Donovan et al., 1996; Chui and Lian, 1996) and motivated a thorough study of the theory of multiwavelets (Heil et al., 1996; Cohen et al., 1997; Plonka, 1997; Plonka and Strela, 1998).

Considering a finitely generated multiresolution analysis with orthonormal multiscaling function $\boldsymbol{\phi}(t)$ and multiwavelet $\boldsymbol{\psi}(t)$, from $V_0 = V_1 \oplus W_1$, we get for $s(t) = \sum_n \mathbf{s}_0^\top(n)\boldsymbol{\phi}(t-n) \in V_0$,

$$s(t) = \sum_n \mathbf{s}_1^\top(n)\boldsymbol{\phi}\left(\frac{t}{2}-n\right) + \mathbf{d}_1^\top(n)\boldsymbol{\psi}\left(\frac{t}{2}-n\right). \tag{27}$$

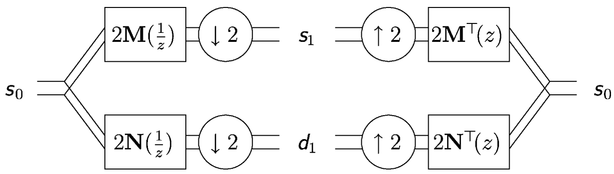
Hence, we derive the well-known Mallat (1989) algorithm for multiwavelets. At the analysis,

$$\mathbf{s}_1(n) = \sum_k \mathbf{M}(k-2n)\mathbf{s}_0(k) \qquad \text{and} \qquad \mathbf{d}_1(n) = \sum_k \mathbf{N}(k-2n)\mathbf{s}_0(k) \tag{28}$$

and at the synthesis, we get

$$\mathbf{s}_0(n) = \sum_k \mathbf{M}^\top(n-2k)\mathbf{s}_1(k) + \mathbf{N}^\top(n-2k)\mathbf{d}_1(k). \tag{29}$$

These relations enable us to construct a multi-input multi-output filter bank (multifilter bank) as shown below.



Because of their inherent vector nature, in order to process scalar signal, multifilter banks require a vectorization of the input signal to produce an r-dimensional input signal. A simple way to do this vectorization is to split scalar signals into their polyphase components. Introducing

$$\begin{bmatrix} m_0(z) \\ m_1(z) \\ \vdots \\ m_{r-1}(z) \end{bmatrix} := 2\mathbf{M}(z^r)\begin{bmatrix} 1 \\ z^{-1} \\ \vdots \\ z^{-(r-1)} \end{bmatrix} \tag{30}$$

and in the same way $n_0(z), n_1(z), \ldots, n_{r-1}(z)$, the system can then be rewritten as a $2r$ channel time-varying filter bank as shown in Fig. 5. Intuitively, this is a filter bank with relaxed requirements on the time invariance. In each filtering block, we periodically alternate between different filters. Lebrun and Vetterli (1998) and Selesnick (1998) pointed out that if the components $m_0(z), m_1(z), \ldots, m_{r-1}(z)$ of the lowpass branch have different spectral behavior, e.g. lowpass behavior for one, highpass for another, this will lead to unbalanced channels that will mix the coarse resolution signal and detail coefficients and

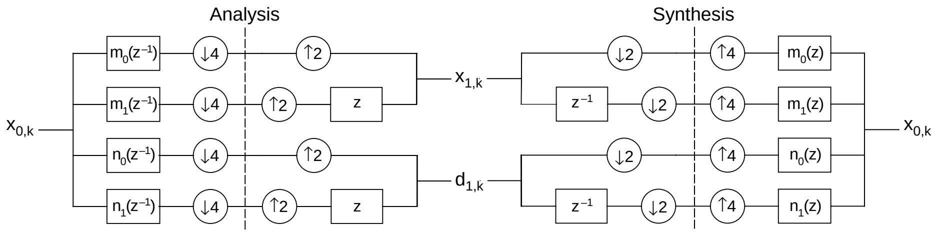


Fig. 5. Multifilter bank seen as a time-varying filter bank.

will create strong oscillations in the reconstructed signal. This leads us to introduce the concept of balanced multiwavelets.

5.1. Balancing

This problem relates also to the basic principle of filter banks: one expects a reasonable class of smooth signals (typically piecewise polynomial signals) to be preserved by the lowpass branch and annihilated by the highpass one. In the wavelet case, the two important issues of the reproduction of polynomials by the associated multiresolution analysis (approximation theory issue) and the preservation/cancellation of discrete-time polynomial signals by the associated filter bank (subband coding and compression issue) are tightly connected since they have been proved to be equivalent to the Strang-Fix conditions: the number of zeros at π in the factorization of the lowpass filter $H_0(\mathrm{e}^{j\omega})$ of the filter bank. This is however not the case anymore for multiwavelets (Lebrun and Vetterli, 1998). The preservation/annihilation of constant signals by the lowpass/highpass branches of the multifilters (called *balancing* of order 1) is proved to be equivalent (Lebrun and Vetterli, 1998; Selesnick, 1998) to any of the following conditions:

- $[1, 1, \ldots, 1]$ is a left eigenvector of $\mathbf{M}(1)$ with eigenvalue $\lambda_0(1) = 1$.
- $\widehat{\boldsymbol{\phi}}(0) = [1, 1, \ldots, 1]^\top$.
- $\sum_{i=0}^{r-1} m_i(z)$ has zeros on the unit circle at $z = \mathrm{e}^{jk\pi/r}$ for $k = 1, \ldots, 2r-1$.
- One can factorize $\mathbf{M}(z) = 1/2\boldsymbol{\Delta}(z^2)\mathbf{M}_0(z)\boldsymbol{\Delta}^{-1}(z)$ with

$$\boldsymbol{\Delta}(z) := \begin{bmatrix} 1 & -1 & 0 & \ldots & 0 \\ 0 & 1 & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 1 & -1 \\ -z^{-1} & 0 & \ldots & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{M}_0(1)\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

These conditions can be generalized to higher orders of balancing (preservation/annihilation of polynomials signals of higher degree). Introducing the polynomial interpolation vector filters (Selesnick, 1998; Lebrun, 2000),

$$\boldsymbol{\alpha}_n^\top(z) := \frac{1}{r}[\alpha_{0,r}^{(n)}(z), \alpha_{1,r}^{(n)}(z), \ldots, \alpha_{r-1,r}^{(n)}(z)]$$

where

$$\alpha_{i,r}^{(n)}(z) = 1 + \sum_{k=1}^{n} \frac{\Gamma(k+\frac{i}{r})}{\Gamma(k+1)\Gamma(\frac{i}{r})}(1-z^{-1})^k,$$

we get the following equivalent conditions:

- $\sum_{k=0}^{r-1} \alpha_{k,r}^{(p-1)}(z^{2r})m_k(z)$ has zeros of order p at $z = \mathrm{e}^{jk\pi/r}$ for $k = 1, \ldots, 2r-1$.
- $\mathbf{M}(z)$ can be factored for $n = 1, \ldots, p$ as

$$\mathbf{M}(z) = \frac{1}{2^n}\mathbf{\Delta}^n(z^2)\mathbf{M}_{n-1}(z)\mathbf{\Delta}^{-n}(z) \quad (31)$$

with $\mathbf{M}_{n-1}(1)\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix} = \begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}$ and $\mathbf{\Delta}(z)$ defined as before.

These two conditions are furthermore convenient to deal with in the practical design of multiwavelets.

Besides, one also proves that balancing of order p is equivalent to $\boldsymbol{\phi}(t)$ having an approximation order of p and for $i = 0, \ldots, r-1$, the shifted analysis scaling functions $\phi_i(t + i/r)$ having identical p first moments i.e. $\int \phi_i(t+i/r)t^n \mathrm{d}t = \int \phi_0(t)t^n \, \mathrm{d}t$ for $i = 0, \ldots, r-1$ and $n = 0, \ldots, p-1$. Intuitively, this says that the condition of balancing of order p imposes multiwavelets to behave like wavelets up to the order p of approximation. One can in fact show that the shortest length orthonormal multiwavelets for a given order of balancing are indeed the Daubechies orthonormal wavelets (Lebrun, 2000; Lebrun and Vetterli, 2001).

6. Algebraic design of multiwavelets

We are now able to deal with and solve the systems of polynomial equations that appear when designing high order balanced multifilters. Using the results obtained in the previous section (especially the factorization of the refinement mask) and inspired by the techniques used by Park et al. (1997) and Faugère et al. (1998) on similar problems of design, we are now ready to investigate the construction of orthonormal multifilters of arbitrary balancing order in a similar way to what Daubechies (1992) did for her well-known filters.

6.1. Symmetry oriented design: the Bat family

Given a balancing order p, we are looking for the shortest length orthonormal multifilters with real coefficients and symmetries. The symmetries on the filters allow easy and practical implementations on finite length signals. The scheme of construction is then the following.

First, we construct the refinement mask $\mathbf{M}(z)$, by putting degrees of freedom on a matrix $\mathbf{M}_{p-1}(z)$.

1. Impose the order of balancing to be p, i.e. for $n = 1, \ldots, p$,

$$\mathbf{M}(z) = \frac{1}{2^n}\mathbf{\Delta}^n(z^2)\mathbf{M}_{n-1}(z)\mathbf{\Delta}^{-n}(z)$$

with $\mathbf{M}_{n-1}(1)[1, \ldots, 1]^\top = [1, \ldots, 1]^\top$. This way we reduce the number of degrees of freedom in the design.
2. Impose the condition **O** (orthonormality) on $\mathbf{M}(z)$,

$$\mathbf{M}(z)\mathbf{M}^\top(z^{-1}) + \mathbf{M}(-z)\mathbf{M}^\top(-z^{-1}) = \mathbf{I}.$$

This gives quadratic equations on the free variables of $\mathbf{M}_{p-1}(z)$ (the idea is to introduce the Laurent polynomial matrix)

$$\mathbf{V}_{p-1}(z) := 2^{-p}(1 - z^{-r})^p\mathbf{M}_{p-1}(z)\mathbf{\Delta}^{-p}(z)$$

and translate the orthonormality condition on this matrix.
3. Impose conditions of symmetry. Here we look for flipping property: $m_1(z) = z^{-2L+1}m_0(z^{-1})$. The flipping property enables an easy lossless symmetrization (detailed in Lebrun (2000)) of finite length input signals.
4. We now have a system of polynomial equations. We compute the algebraic dimension of the system using a drl Gröbner basis approach and increase the degrees of freedom until we get solutions (and a drl Gröbner basis of dimension 0). We used here the programs `Singular` (Greuel et al., 2000) for order $p = 1, 2, 3$ and FGb (Faugère, 1999) for the order $p = 4$. We now have a zero-dimensional drl Gröbner basis $G_{<_{\mathrm{drl}}}$ that we can either transform into a lex Gröbner basis $G_{<_{\mathrm{lex}}}$ using FGLM in the case $p = 1, 2, 3$ or in the case $p = 4$, where FGLM showed its limits, we compute a rational univariate representation of $G_{<_{\mathrm{drl}}}$ by a modified version of the program RealSolving (Rouillier, 1999). We can then factorize the leading polynomial of the lex Gröbner basis $G_{<_{\mathrm{lex}}}$ in Maple and thus get rid of the multiplicities of the solutions. This means we factorize the Gröbner basis in local algebras that are much easier to solve exactly. In the case $p = 4$, we deal with a RUR and a similar idea is applied to the characteristic polynomial $\chi_u(t)$. We then have the set of solutions for the system.
5. Among this finite number of solutions, we can look for the one leading to the smoothest scaling functions using an estimate by invariant cycles (Lebrun and Vetterli, 2001).

Then, we easily derive the highpass filters $n_0(z), n_1(z)$ from the lowpass filters $m_0(z)$, $m_1(z)$ by imposing $n_0(z)$ to be symmetric and $n_1(z)$ to be antisymmetric. The orthonormality conditions give a unique solution up to a change of sign.

Using this approach, we have been able to construct all the minimal length orthonormal multiwavelets with compact support and flipped scaling functions, symmetric/antisymmetric wavelets for order 1, 2, 3 and 4 of balancing. Fig. 6 shows the smoothest scaling functions associated to these high order balanced multiwavelets. The coefficients are available from the webpage of J.L. For order 4 of balancing, because of the degree of the characteristic polynomial in the RUR, a real roots localization program (included in RealSolving) has been used and only numerical solutions (in fact exact intervals containing the solutions) have been obtained.

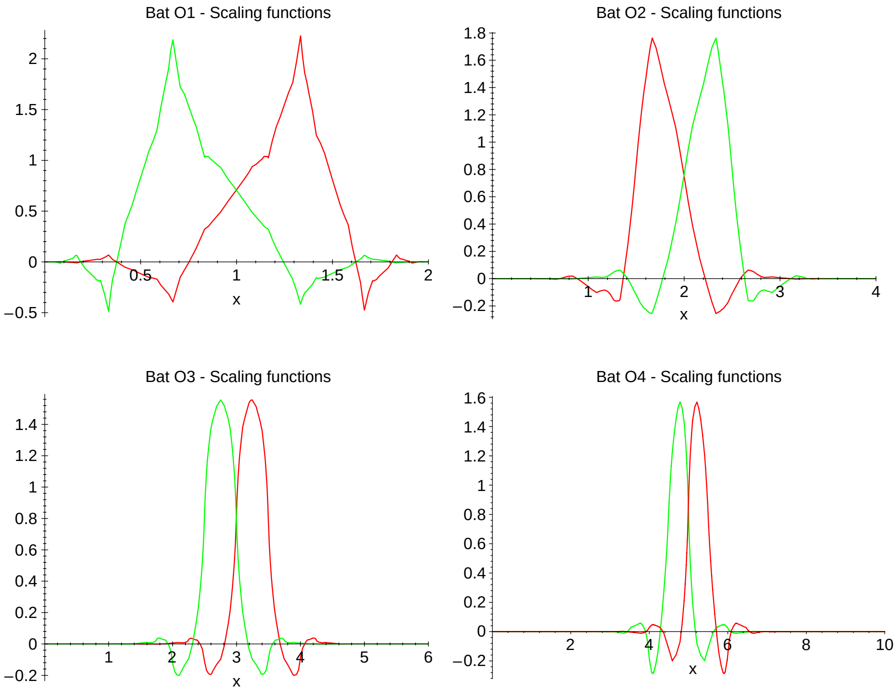


Fig. 6. Order 1 (resp. 2, 3 and 4) balanced orthogonal multiwavelet: the scaling functions are flipped around 1 (resp. 2, 4 and 8), the wavelets (not shown here) are symmetric/antisymmetric, the length is 3 (resp. 5, 7 and 11) taps (2×2).

6.2. *Interpolation oriented design: M-Coiflets*

Now, if the scaling function $\phi_0(t)$ has furthermore $p - 1$ vanishing moments, we get a multiwavelet generalization of Coiflets. *MultiCoiflets* are thus constructed as balanced multiwavelets with more stringent conditions on the moments of $\phi_0(t)$. For practical design, we will use the following extension of the balanced vanishing moments condition. For $n = 0, \ldots, p - 1$, we have

$$\frac{\mathrm{d}^n}{\mathrm{d}\omega^n}[\boldsymbol{\alpha}_{p-1}^\top(\mathrm{e}^{j2\omega})\mathbf{M}(\mathrm{e}^{j\omega})]_{|\omega=0} = \frac{\mathrm{d}^n}{\mathrm{d}\omega^n}[\boldsymbol{\alpha}_{p-1}^\top(\mathrm{e}^{j\omega})]_{|\omega=0}$$
$$\frac{\mathrm{d}^n}{\mathrm{d}\omega^n}[\boldsymbol{\alpha}_{p-1}^\top(\mathrm{e}^{j2\omega})\mathbf{M}(\mathrm{e}^{j\omega})]_{|\omega=\pi} = \mathbf{0}^\top.$$

The design procedure is then very similar to the one for the Bat family above. Two new conditions are added:

1. The filters $m_0(z)$ and $m_1(z)$ are supposed to be odd length and symmetric.
2. $\mathbf{M}(z)$ satisfies the multiCoiflet conditions above for $n = 0, \ldots, p - 1$.

Using this approach, we have been able to construct all the minimal length orthonormal multiCoiflets with compact support, symmetric scaling functions, symmetric/antisymmetric wavelets for order 1, 2 and 3 of balancing. Fig. 7 shows the smoothest scaling functions associated to multiCoiflets with these properties. More details are available in Lebrun (2000), Lebrun (2003) and also Selesnick (1999) where a family of cardinal multiwavelets is constructed (these can be seen as generalized multiCoiflets with the center of mass of the scaling functions not on the integer grid anymore).

7. Wavelet tight frames

This section describes a family of wavelet tight frames, or 'overcomplete' bases introduced in Selesnick (2000a) and Selesnick and Sendur (2000). With frames some fundamental properties can be better realized than is possible with bases. For example, better time-frequency localization can be achieved. In addition, wavelet frames can be shift-invariant, while wavelet bases cannot be. In general, frames provide more degrees of freedom to carry out design. Several applications have benefited from the use of frames, for example, denoising (Coifman and Donoho, 1995; Guo et al., 1995; Lang et al., 1995, 1996).

This section describes the design of frames that are analogous to the Daubechies (1992) orthonormal wavelets—that is, the design of minimal length filters with polynomial properties, but now in the case of oversampled filter banks. The wavelets presented below are much smoother than what can be achieved in the critically sampled case.

The nonlinear design equations that arise are then solved using Gröbner bases. As Gröbner bases are used here, we are able to obtain zero wavelet moments for wavelets of minimal length, in contrast to earlier work on wavelet tight frames of this type (Chui and He, 2000; Ron and Shen, 1997). Some later works (Chui et al., 2002; Daubechies et al., 2001) also describe other methods not based on Gröbner bases.

7.1. Preliminaries

The wavelet tight frames developed in this section are based on a single scaling function $\phi(t)$ and two distinct wavelets $\psi_1(t)$ and $\psi_2(t)$. Following the multiresolution framework, ϕ, ψ_1, ψ_2 satisfy the dilation and wavelet equations

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

$$\psi_i(t) = \sqrt{2}\sum_n h_i(n)\phi(2t-n), \qquad i = 1, 2.$$

Corresponding to ϕ, ψ_1, ψ_2, we have the scaling filter $h_0(n)$, the two wavelet filters $h_1(n)$ and $h_2(n)$, and the oversampled filter bank illustrated in Fig. 8.

Let $\phi_k(t) = \phi(t-k)$ and $\psi_{i,j,k}(t) = \psi_i(2^j t - k)$ for $i = 1, 2$. Then $\{\phi_k(t), \psi_{i,j,k}(t) : j, k \in \mathbb{Z}, j \geq 0, i \in \{0, 1\}\}$ forms a tight frame for $L_2(\mathbb{R})$ if any square integrable signal $f(t)$ can be expanded as

$$f(t) = \sum_{k=-\infty}^{\infty} c(k)\phi_k(t) + \sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} (d_1(j,k)\psi_{1,j,k}(t) + d_2(j,k)\psi_{2,j,k}(t)) \qquad (32)$$

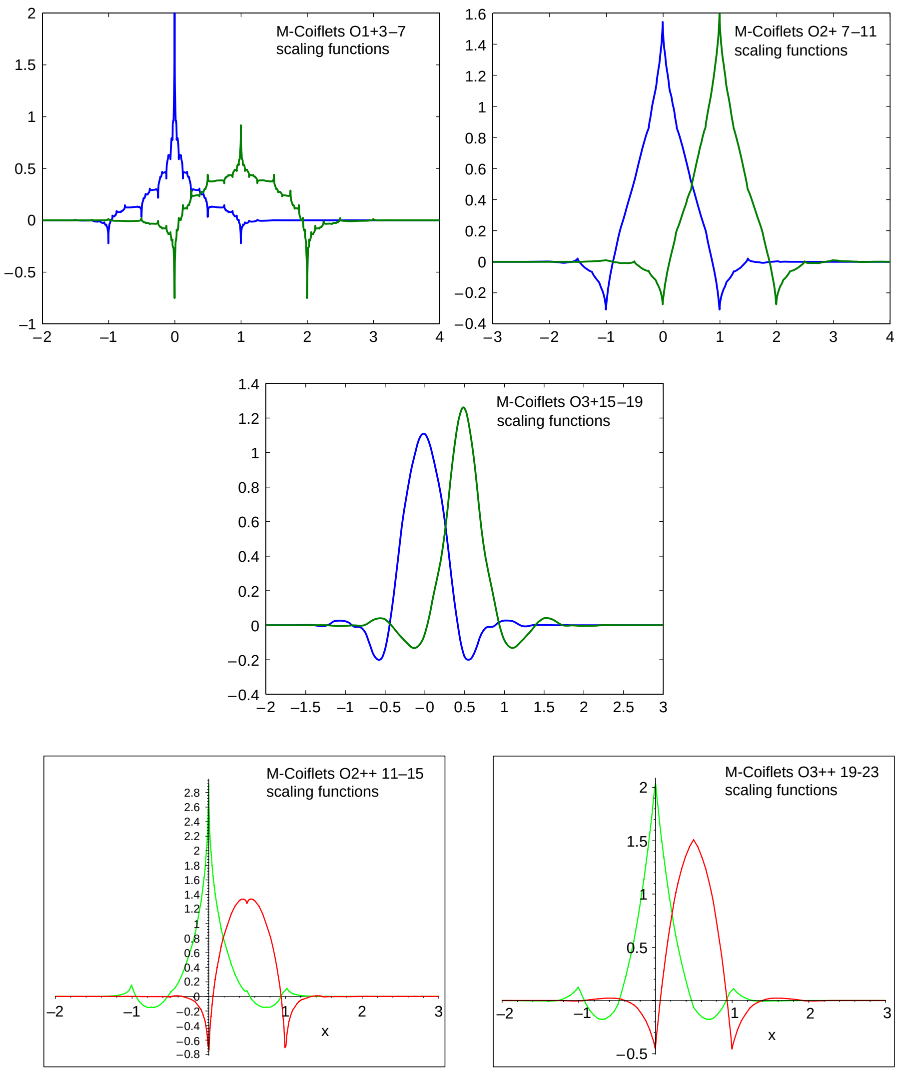


Fig. 7. Symmetric orthogonal balanced multiCoiflets of order 1, 2 and 3. On the third row, the multiCoiflets have one additional vanishing moment in 0 for the same balancing order.

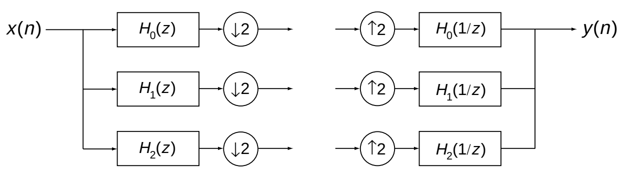


Fig. 8. An oversampled analysis and synthesis filter bank.

where

$$c(k) = \int f(t)\phi_k(t)\,\mathrm{d}t, \qquad d_i(j,k) = \int f(t)\psi_{i,j,k}(t)\,\mathrm{d}t, \qquad i = 1, 2.$$

That is, a function can be expanded in a tight frame in a way that resembles an expansion in an orthonormal basis. Like orthonormal bases, tight frames have a Parseval's relation:

$$\| f \|^2 = \sum_{k=-\infty}^{\infty} |c(k)|^2 + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} (|d_1(j,k)|^2 + |d_2(j,k)|^2).$$

By standard multirate identities, the PR conditions can be written as

$$H_0(z)H_0(z^{-1}) + H_1(z)H_1(z^{-1}) + H_2(z)H_2(z^{-1}) = 2 \tag{33}$$

$$H_0(-z)H_0(z^{-1}) + H_1(-z)H_1(z^{-1}) + H_2(-z)H_2(z^{-1}) = 0. \tag{34}$$

7.2. *Zeros at $\omega = 0$, $\omega = \pi$*

Let K_0 denote the number of zeros $H_0(\mathrm{e}^{j\omega})$ has at $\omega = \pi$. For $i = 1, 2$, let K_i denote the number of zeros $H_i(\mathrm{e}^{j\omega})$ has at $\omega = 0$.

$$(z+1)^{K_0} \mid H_0(z), \qquad (z-1)^{K_1} \mid H_1(z), \qquad (z-1)^{K_2} \mid H_2(z). \tag{35}$$

For orthonormal bases ($\psi_2(t) = 0$), it is necessary that $K_0 = K_1$, so no distinction need be made between K_0 and K_1. However, for tight wavelet frames of the form (32), it is not necessary that $K_0 = K_1 = K_2$. K_0 denotes the degree of polynomials representable by shifts of $\phi(t)$. K_1 and K_2 denote the number of zero moments of the wavelet filters $h_1(n)$ and $h_2(n)$, provided $K_0 \geq K_1$, and $K_0 \geq K_2$.

The value of K_0 influences the degree of smoothness of ϕ (and therefore of ψ_i). On the other hand, the values K_1 and K_2 indicate what polynomials are annihilated (compressed) by the given signal expansion. In contrast to orthonormal wavelet bases, with a tight frame one has the possibility to control these parameters more freely. If it is desired for a given class of signals that the wavelets have two zero moments (for example), then the remaining degrees of freedom can be used to achieve a higher degree of smoothness by making K_0 greater than K_1 and K_2.

7.3. *Example*

We seek to design FIR filters h_0, h_1, h_2 that generate tight frames of the form described in (32). We seek the shortest filters h_i having a prescribed number of zeros at $z = -1$ and $z = 1$ (specified by the values K_i) that satisfy the tight frame conditions (33) and (34). In the examples, we ask that $K_1 = K_2$. If they are unequal, then one wavelet annihilates more polynomials than the other, or one wavelet is doing 'more work' than the other.

Note that the conditions (33) and (34) are nonlinear equations in the filter coefficients $h_i(n)$. For the design problems considered below, these nonlinear design equations are handled using Gröbner bases.

We ask that $K_0 = 5$, $K_1 = K_2 = 2$. It was found that the shortest filters h_0, h_1, h_2 satisfying (33) and (34) are of length 7, 7, and 5, respectively. By utilizing Gröbner basis methods it is possible to obtain exact expressions for $h_i(n)$. (*Singular* (Greuel et al., 2000) was used for the Gröbner basis calculations.) The original design equations have

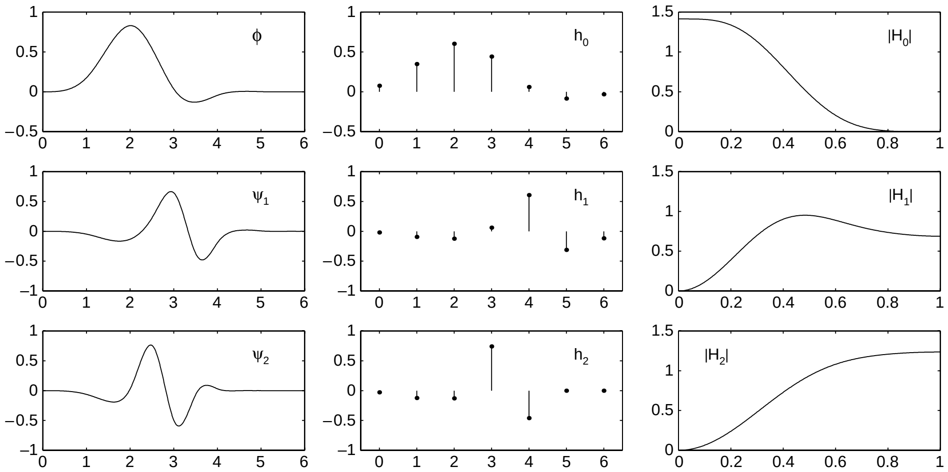


Fig. 9. The generators of a wavelet tight frame with parameters $K_0 = 5$, $K_1 = K_2 = 2$.

only rational coefficients, and we were able to obtain *explicit* expressions for $h_i(n)$ in terms of radicals. The expressions obtained for $h_i(n)$ are too long to include here, but are available from I.S.

The filters h_0, h_1, h_2 were found by converting the nonlinear design equations into a drl Gröbner basis, then converting that into a lex Gröbner basis, and factorizing that into two Gröbner bases. Then the ordering of the variables was changed, to obtain two lex Gröbner bases which are more compact. However, if this ordering is used from the beginning, then the original lex Gröbner basis does not factor. All minimal-length pairs of scaling filters can be found by solving these 2 Gröbner bases. The Gröbner bases indicate that there are 32 solutions to the nonlinear design equations (16 solutions from each part).

As in the orthonormal case, there are multiple solutions to this problem. However, in contrast to the orthonormal case, (i) the distinct solutions do not all share the same autocorrelation, and (ii) not all of the spectral factors of each autocorrelation are solutions. In this example, there are 4 distinct solutions, not counting their time-reversals ($h_i(-n)$) and negations ($-h_i(n)$). One of those 4 solutions is shown in Fig. 9. The other 3 solutions are tabulated on the webpage of I.S.

8. Conclusion

In conclusion, for many of the design problems arising in the construction of specialized wavelets and filter banks, Gröbner bases are a natural tool. And although the high computational and memory cost of Gröbner bases limits their utility, we are able to obtain solutions of practical interest, as illustrated in this paper. Indeed, we have introduced examples of multiwavelet bases and wavelet frames that we could not have obtained without them. As software for Gröbner bases, and the related theory, is advancing with time, we expect Gröbner bases will be no less useful for future problems arising in the design of filters and transforms for signal processing.

Acknowledgements

J.L. warmly thanks Martin Vetterli for having made Gröbner Bases *unavoidable* from the start of his Ph.D. thesis. A big credit has to be given also to Jean-Charles Faugère and Fabrice Rouillier for making their wonderful software available and a very special thank you goes to Gil Strang and Thierry Blu for their useful remarks and fruitful help. This work was partly supported by the Swiss National Science Foundation under grant 2100-043136.95/1. I.S. would like to thank Richard Baraniuk and Rob Nowak for bringing Gröbner bases to his attention in 1995. He also gratefully acknowledges support by the NSF under CAREER grant CCR-987452.

Appendix A. Computation algebra digest

A.1. Introducing Gröbner bases

In this paragraph, we will review the major ideas involved in the computation of Gröbner bases. We will not go much into the details, since many good books ranging from introductory (Fröberg, 1997) to advanced level (Cox et al., 1992, 1998) have been written on this now popular subject. We will rather develop an intuitive understanding of what a Gröbner basis is and describe some ways to compute them by using analogies to linear algebra.

We define a multivariate polynomial p to be a finite sum of terms $\sum_\alpha c_\alpha x^\alpha$, where a term $c_\alpha x^\alpha$ is the product of a coefficient c_α and a monomial x^α. One can draw an analogy between solving linear systems that can be seen as the study of the associated vector subspace and solving a polynomial system that can be seen as the study of the associated ideal. Namely, a polynomial system of equations is defined by a list $\{p_1, \dots, p_N\}$ of multivariate polynomials with rational coefficients ($p_1, \dots, p_N \in \mathbb{Q}[x_1, \dots, x_n]$). We associate to this system the generated ideal $I = \langle p_1, \dots, p_N \rangle$ i.e. the smallest ideal containing $p_1, \dots, p_N$ as well as $\sum_{k=1}^N h_k p_k$, for any $h_1, \dots, h_N \in \mathbb{Q}[x_1, \dots, x_n]$. Intuitively, the idea is that the polynomials $p_1, \dots, p_N$ have a common zero iff any polynomial of the ideal I vanishes also at that location. It is then equivalent to study a system of polynomial equations or the ideal generated by the polynomials.

For a set of linear equations, the Gauss elimination algorithm enables us to compute an equivalent triangular system by canceling the leading term of each equation. A similar algorithm (called the *Buchberger* algorithm) can be developed for the case of multivariate polynomials. An important aspect of the Gauss elimination algorithm is in the choice of the pivots that are used during the triangularization of the system. For the same reasons, the first thing we have to define is an ordering on the monomials (that needs to be compatible with polynomial multiplication). We introduce here two monomial orderings:

- The *lexicographic* ordering, abbreviated *lex*. This is the ordering used in dictionaries.
- The *degree reverse lexicographic* order, abbreviated *drl*. This is a modified reversed lexicographic ordering taking first into account the total degree of the polynomials.

We then introduce the leading term $\mathrm{lt}(p, <)$ of a polynomial p as its term with the highest order according to the ordering $<$, we also introduce the leading monomial

$\mathrm{lm}(p, <)$ as the leading term with a coefficient normalized to 1 and $\mathrm{lc}(p, <)$ as the leading coefficient. Notice that when no doubt remains, we will omit to mention the ordering. In a very similar way to what is done in the Gauss elimination algorithm, we introduce the *S-polynomial* as a monomial combination of two polynomials so as to cancel their leading terms.

$$\mathrm{Spol}(p_1, p_2) := \frac{\mathrm{lcm}(\mathrm{lt}(p_1), \mathrm{lt}(p_2))}{\mathrm{lt}(p_1)} p_1 - \frac{\mathrm{lcm}(\mathrm{lt}(p_1), \mathrm{lt}(p_2))}{\mathrm{lt}(p_2)} p_2 \tag{36}$$

where lcm stands for the *least common multiple* of a set of polynomials. For example, with the $<_{\mathrm{lex}}$ ordering $(x > y > z)$, $p_1 = 2x^3y + \cdots$ and $p_2 = x^2y^2 + \cdots$, we get $\mathrm{Spol}(p_1, p_2) = yp_1 - 2xp_2$. We have canceled the leading terms of p_1 and p_2. Of particular interest is when $\mathrm{Spol}(p_1, p_2) = p_1 - qp_2$ for some polynomial q (e.g. $p_1 = 3x^3y + \cdots$ and $p_2 = xy + \cdots$). In that case, we say that p_1 is *reducible* by p_2 and that q is the *reduction* of p_1 by p_2. This reduction can easily be extended to the reduction of a polynomial by an ordered list of polynomials, $L = [q_1, \ldots, q_N]$.

We shall emphasize the importance of the order in which the reductions are done: the same set of polynomials reordered in a different list will usually give rise to a different output of the reduction process. However, for any list of polynomials there exists an equivalent list such that the order has no influence anymore.

The famous Buchberger algorithm transforms by a progressive reduction process a general ordered list of polynomials generating the ideal I into an equivalent one that makes it much easier to deal with the ideal generated. The list of polynomials obtained by the Buchberger algorithm is called a Gröbner basis. One of the major properties of Gröbner bases is that it makes it algorithmically easy to verify if a given polynomial belongs or not to the ideal generated.

The major features of the Buchberger algorithm is that the list obtained $G := [g_1, \ldots, g_N]$ still generates I and satisfies the following Gröbner basis definition: $\mathrm{Spol}(g_i, g_j)$ reduces to 0 modulo G, for every $g_i, g_j \in G$. It is easily seen that Gröbner bases have the following equivalent characterizations:

- $f \in I$ iff f reduces to 0 modulo G ($\mathrm{Reduce}(f, G) = 0$).
- The leading term of any element of I is divisible by at least one leading term $\mathrm{lt}(g_i)$ of G.

For an ideal I, let $\langle \mathrm{LT}(I) \rangle$ denote the ideal of leading terms of I, i.e. the ideal generated by the set of leading terms $\mathrm{LT}(I) := \{cx^\alpha \mid \exists f \in I, \mathrm{lt}(f) = cx^\alpha\}$. We then get that $G := [g_1, \ldots, g_N]$ is a Gröbner basis of I iff the ideal of leading terms of I is generated by the leading terms of G i.e. $\langle \mathrm{LT}(I) \rangle = \langle \mathrm{lt}(g_1), \ldots, \mathrm{lt}(g_N) \rangle$.

Usually, one can compute infinitely many Gröbner bases. However, among all these, one satisfies some nicer properties: every element g_i of the basis G has its leading term normalized (coefficient equal to 1) and $\forall g_i \in G$, no term of g_i is divisible by a leading monomial $\mathrm{lm}(g_j)(j \neq i)$. This particular basis is called the *reduced* Gröbner basis: one verifies that for a given monomial ordering monomial $<$, a non-empty polynomial ideal always has a unique reduced Gröbner basis. With the reduced Gröbner basis, we get the very nice feature that the output of $\mathrm{Reduce}(p, G)$ does not depend anymore on the order of the polynomials in the list: $\mathrm{Reduce}(\cdot, G)$ becomes the *canonical reduction* modulo I.

In the case $<$ is lexicographic, the reduced Gröbner basis has a very nice structure. Namely, the reduced Buchberger algorithm gives a union of triangular arrays of polynomials of the following form:

$$\begin{cases} h_0^{(d+1)}(x_1,\ldots,x_d)\left(x_{d+1}^{k_{d+1}} + \frac{h_1^{(d+1)}(x_1,\ldots,x_d)}{h_0^{(d+1)}(x_1,\ldots,x_d)} x_{d+1}^{k_{d+1}-1} + \frac{h_2^{(d+1)}(x_1,\ldots,x_d)}{h_0^{(d+1)}(x_1,\ldots,x_d)} x_{d+1}^{k_{d+1}-2} + \cdots\right) \\ h_0^{(d+2)}(x_1,\ldots,x_{d+1})\left(x_{d+2}^{k_{d+2}} + \frac{h_1^{(d+2)}(x_1,\ldots,x_{d+1})}{h_0^{(d+2)}(x_1,\ldots,x_{d+1})} x_{d+2}^{k_{d+2}-1} + \frac{h_2^{(d+2)}(x_1,\ldots,x_{d+1})}{h_0^{(d+2)}(x_1,\ldots,x_{d+1})} x_{d+2}^{k_{d+2}-2} + \cdots\right) \\ \cdots \\ h_0^{(n)}(x_1,\ldots,x_{n-1})\left(x_n^{k_n} + \frac{h_1^{(n)}(x_1,\ldots,x_{n-1})}{h_0^{(n)}(x_1,\ldots,x_{n-1})} x_n^{k_n-1} + \frac{h_2^{(n)}(x_1,\ldots,x_{n-1})}{h_0^{(n)}(x_1,\ldots,x_{n-1})} x_n^{k_n-2} + \cdots\right) \end{cases} \tag{37}$$

where d gives the number of remaining degrees of freedom of the system when all of the equations are satisfied ($x_1, \ldots, x_d$ are now parameters). d is called the algebraic dimension of the ideal: the solutions of a system of polynomial equations can be seen as a geometric variety that can be classified by its algebraic dimension: $d = 0$: finite number of isolated points, $d = 1$: curves, $d = 2$: surfaces and so on. In case the system has different kinds of solutions (e.g. isolated points and curves), the global dimension is just the maximum dimension of each component.

When $d = 0$, i.e. when the system has a finite number of solutions, we get that the first equation becomes a univariate polynomial equation and we can then rewrite the reduced Gröbner basis as:

$$\begin{cases} x_1^{k_1} + g_1(x_1) & \deg_{x_1}(g_1) < k_1 \\ x_2^{k_2} + g_2(x_1, x_2) & \deg_{x_2}(g_2) < k_2 \\ \cdots & \cdots \\ x_n^{k_n} + g_n(x_1, x_2, \ldots, x_n) & \deg_{x_n}(g_n) < k_n. \end{cases} \tag{38}$$

On such a system, it is now easy to carry out many operations like counting exactly all complex/real solutions including the multiplicity, isolating the real roots with the desired precision or approximating the complex roots (Gonzalez-Vega et al., 1999). For example, to numerically solve the system: first solve the univariate equation $x_1^{k_1} + g_1(x_1) = 0$, then recursively substitute and solve the next equations. Moreover, in the case the variable x_1 is separating (intuitively two solutions cannot have the same first component; a rigorous definition is given in the next section), we get that $k_2 = k_3 = \cdots = k_n = 1$. The system is then of the form

$$\begin{cases} \tilde{g}_1(x_1) \\ x_2 + \tilde{g}_2(x_1) \\ \cdots \qquad \cdots \\ x_n + \tilde{g}_n(x_1). \end{cases} \tag{39}$$

and all we have to do is to solve $\tilde{g}_1(x_1) = 0$ and substitute in the other equations. This is called the *Shape lemma* case (Rouillier, 1999).

A.2. Linear algebra methods

The necessary time to compute a reduced Gröbner basis by the Buchberger algorithm depends strongly on the monomial ordering that is used. In general, computing a reduced Gröbner basis for the lexicographic ordering is much more time and memory consuming than computing the corresponding drl Gröbner basis. However, this additional computational cost may be worth it because, as seen before, the lexicographic ordering provides a triangular like structure (similar to the one obtained by Gauss elimination) that is really suitable for further processing. Fortunately, recent algorithms enable the efficient computation of lexicographic Gröbner bases by using an alternative approach:

- First, we compute a Gröbner basis for the drl ordering, using, for example, the standard Buchberger algorithm (note that the algorithm can be highly improved by using heuristics for the choice of the critical pairs and the reductors in the reducing process). An even better approach is to completely suppress the influence of these choices, by in fact *not choosing* anymore as introduced by Faugère (1999) in his F_4 algorithm: instead of choosing one critical pair, we take a subset of critical pairs and reduce this set. By using a linear algebra approach to deal with the pairs, the algorithm can be made extremely efficient for the computation of drl Gröbner bases. An implementation named FGb of this algorithm can be tested on the web at https://www-calfor.lip6.fr
- Finally, we compute the lexicographic Gröbner basis from the drl one by a change of ordering. For the case when the ideal is zero-dimensional, a very efficient algorithm called FGLM (Faugère et al., 1994) has been developed using again a linear algebra approach. Implementations of this algorithm are now available in most of the computer algebra programs.

We will now give some details on how the linear algebra approach works. Again, for more details, the reader can read the survey on the subject by Mourrain (1999). Starting from a list P of polynomials such that the generated ideal $I = \langle P \rangle$ is zero-dimensional, we show that the quotient space $\mathcal{A} := \mathbb{Q}[x_1, \ldots, x_n]/I$ inherits a structure of finite-dimensional algebra. Namely, assuming a reduced Gröbner basis $G := [g_1, \ldots, g_N]$ for some ordering $<$ (typically drl), any element of $\mathcal{A}$ has the form $\bar{p} = \text{Reduce}(p, G)$ for some $p \in \mathbb{Q}[x_1, \ldots, x_n]$. Since $\langle \text{LT}(I) \rangle = \langle \text{lt}(g_1), \ldots, \text{lt}(g_N) \rangle$, we easily construct a linear basis of $\mathcal{A}$ from the set of monomials $\{x^\alpha \mid x^\alpha \notin \langle \text{LT}(I) \rangle\}$, by taking in increasing order the monomials under the staircase, i.e. the x^α that are not a multiple of $\text{lt}(g_i)$ (since this implies that $\bar{x^\alpha} = \text{Reduce}(x^\alpha, G) = x^\alpha$). The linear basis $B := \{\omega_1, \ldots, \omega_d\}$ obtained this way is called the *monomial basis* of $\mathcal{A}$. Finally, constructing the multiplication table $[\omega_i \omega_j]_{i,j}$ of $\mathcal{A}$, we get a full description of the linear algebraic framework in which we will deal with the polynomials.

Now, any element $\bar{p} \in \mathcal{A}$ can be expressed as a vector $[p]$ since $\bar{p} = \sum_{k=1}^{d} [p]_i \omega_i$. The FGLM algorithm can then be described using linear algebra in $\mathcal{A}$. The lexicographic Gröbner basis is obtained by detecting linear combinations of monomials in $\mathcal{A}$. The idea is to construct in parallel the lex Gröbner basis $G_{<_{\text{lex}}}$ and a full rank $d \times d$ matrix $\mathbf{G}$, by scanning the monomials x^α in increasing lex ordering (starting from 1). There are two possibilities:

1. $[x^\alpha]$ is linearly dependent of the previous vectors put in $\mathbf{G}$ (i.e. we can write $[x^\alpha] = \sum_k c_k [x^{\beta_k}]$), then we add $g_\alpha := x^\alpha - \sum_k c_k x^{\beta_k}$ to $G_{<_{\text{lex}}}$ (namely, $g_\alpha \in I$ and $\text{lt}(g_\alpha) = x^\alpha$).
2. $[x^\alpha]$ is linearly independent of the previous vectors put in $\mathbf{G}$, then add $[x^\alpha]$ to $\mathbf{G}$.

Repeat the scan until $\text{rank}(\mathbf{G}) = d$; $G_{<_{\text{lex}}}$ is then a lex Gröbner basis of I.

A.3. The rational univariate representation

In many situations, the computation of a lex Gröbner basis of the ideal I is a bit of an overkill in the sense that in fact, all we are really interested in, is a good description of the set of solutions of the system, $\mathcal{Z}_{\mathbb{C}}(I) := \{\boldsymbol{\alpha} \in \mathbb{C}^n \mid \forall p \in P, p(\boldsymbol{\alpha}) = 0\}$ (we will denote by $\mu(\boldsymbol{\alpha})$ the multiplicity of a solution $\boldsymbol{\alpha}$). In the approach, alternative to FGLM, developed by Rouillier (1999) and Gonzalez-Vega et al. (1999), one constructs a list $\{\chi_u(t), g_u(1,t), g_u(x_1,t), \ldots, g_u(x_1,t)\}$ of polynomials of $\mathbb{Q}[x_1,\ldots,x_n]$ such that: if $\boldsymbol{\alpha}$ is a solution of the system, then $u(\boldsymbol{\alpha})$ is a root of $\chi_u(t)$ with the same multiplicity and conversely, if ζ is a root of $\chi_u(t)$, then

$$\left[\frac{g_u(x_1,\zeta)}{g_u(1,\zeta)}, \frac{g_u(x_2,\zeta)}{g_u(1,\zeta)}, \ldots, \frac{g_u(x_n,\zeta)}{g_u(1,\zeta)}\right] \tag{40}$$

is a solution of the system with the same multiplicity. Hence, $\mathcal{Z}_{\mathbb{C}}(I)$ is fully characterized. The basic tool of this method is the computation of linear operator $\mathbf{M}_u$ on $\mathcal{A}$ associated to polynomials u,

$$\begin{aligned} &\mathbf{M}_u : \mathcal{A} \to \mathcal{A} \\ &\bar{f} \mapsto \mathbf{M}_u \bar{f} := \overline{uf}. \end{aligned} \tag{41}$$

We identify $\mathbf{M}_u$ with its $\mathbb{C}^{d\times d}$ matrix representation in the monomial basis of $\mathcal{A}$. This matrix is easily computed by expressing $\bar{u}\omega_i$ in the monomial basis, which gives the ith column of $\mathbf{M}_u$.

The computations of $\mathbf{M}_u$ give much important information on $\mathcal{Z}_{\mathbb{C}}(I)$ and the system in general. From these matrices, we can construct a bijection between $\mathcal{Z}_{\mathbb{C}}(P)$, the set of solutions of the system, and the roots of the univariate polynomial $\chi_u(t)$. All we have to do now to get $\mathcal{Z}_{\mathbb{C}}(P)$ is to isolate the roots of $\chi_u(t)$. We then derive the solutions of the system using the RUR. The isolation of roots is usually a difficult problem. However, in the case we are only interested in the real solutions of the system, we can locate them very efficiently by computing the signature of trace matrices (Pedersen et al., 1993).

Alternatively, we can also factorize $\chi_u(t)$. We construct this way local algebras that enable us to simplify the problem of isolating the roots by lowering the degrees of characteristic polynomials in the RURs.

In short, the RUR approach appears to be a very efficient alternative to the computation of a lexicographic Gröbner basis: as detailed in Rouillier (1999), it is computationally easier to compute a RUR than to apply the FGLM algorithm and the characteristic polynomial $\chi_u(t)$ is usually easier to deal with than $\tilde{g_1}(x_1)$, the leading polynomial of the lexicographic Gröbner basis. Besides, when the polynomial system is overdetermined, similar linear algebra methods (Giusti and Schost, 1999) have been developed to completely avoid the computation of Gröbner bases.

References

Abdelnour, A.F., Selesnick, I.W., 2001. Nearly symmetric orthogonal wavelet bases. In: ICASSP, Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, May, 2001.

Alpert, B., 1993. A class of bases in ℓ^2 for the sparse representation of integral operators. SIAM J. Math. Anal. 24 (1), 246–262.

Burrus, C.S., Gopinath, R.A., Guo, H., 1998. Introduction to Wavelets and Wavelets Transforms—A Primer. Prentice-Hall.

Chui, C.K., He, W., 2000. Compactly supported tight frames associated with refinable functions. Appl. Comput. Harmon. Anal. 8 (3), 293–319.

Chui, C.K., He, W., Stöckler, J., 2002. Compactly supported tight and sibling frames with maximum vanishing moments. Appl. Comp. Harmonic Anal. 13, 224–262.

Chui, C.K., Lian, J.-A., 1996. A study of orthonormal multi-wavelets. Appl. Numer. Math. 20 (3), 273–298.

Cohen, A., 1992. Biorthogonal wavelets. In: Chui, C.K. (Ed.), Wavelets: A Tutorial in Theory and Applications. Academic Press, New York.

Cohen, A., Daubechies, I., Plonka, G., 1997. Regularity of refinable function vectors. J. Fourier Anal. Appl. 3, 295–324.

Coifman, R.R., Donoho, D.L., 1995. Translation-invariant de-noising. In: Antoniadis, A. (Ed.), Wavelets and Statistics. Springer-Verlag Lecture Notes.

Cox, D., Little, J., O'Shea, D., 1992. Ideals, Varieties and Algorithms. Springer-Verlag.

Cox, D., Little, J., O'Shea, D., 1998. Using Algebraic Geometry. Springer-Verlag.

Crochiere, R.E., Webber, S.A., Flanagan, J.L., 1976. Digital coding of speech in sub-bands. Bell Syst. Tech. J. 55 (8), 1069–1085.

Croisier, A., Esteban, D., Galand, C., 1976. Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. In: Int. Conf. on Inform. Sciences and Systems, Patras, Greece, 1976, pp. 443–446.

Daubechies, I., 1988. Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. 41, 909–996.

Daubechies, I., 1992. Ten Lectures on Wavelets. SIAM.

Daubechies, I., Han, B., Ron, A., Shen, Z., 2001. Framelets: MRA-based constructions of wavelet frames (preprint).

Donovan, G.C., Geronimo, J.S., Hardin, D.P., Massopust, P.R., 1996. Construction of orthogonal wavelets using fractal interpolation functions. SIAM J. Math. Anal. 27 (4), 1158–1192.

Faugère, J.-C., 1999. A new efficient algorithm for computing Gröbner bases (F_4). J. Pure Appl. Algebra 139, 61–88. Available from https://www-calfor.lip6.fr.

Faugère, J.-C., de Saint-Martin, F.M., Rouillier, F., 1998. Design of regular nonseparable bidimensional wavelets using Gröbner basis techniques. IEEE Trans. Signal Proc. 46 (4), 845–857.

Faugère, J.-C., Gianni, P., Lazard, D., Mora, T., 1994. Efficient computation of zero-dimensional Gröbner bases by change of ordering. J. Symbolic Comput. 16 (4), 329–344.

Fröberg, R., 1997. An Introduction to Gröbner Bases. John Wiley & Sons.

Geronimo, J.S., Hardin, D.P., Massopust, P.R., 1994. Fractal functions and wavelet expansions based on several scaling functions. J. Approx. Theory 78.

Gonzalez-Vega, L., Rouillier, F., Roy, M.-F., 1999. Symbolic recipes for polynomial system solving. In: Some Tapas of Computer Algebra. Springer-Verlag.

Greuel, G.-M., Pfister, G., Schönemann, H., 2000. Singular reference manual. University of Kaiserslautern. Available from http://www.singular.uni-kl.de.

Giusti, M., Schost, E., 1999. Solving some overdetermined polynomial systems. In: ISSAC, Proc. Int. Symp. on Symbolic and Algebraic Computation, July 1999.

Guo, H., Lang, M., Odegard, J.E., Burrus, C.S., 1995. Nonlinear shrinkage of undecimated DWT for noise reduction and data compression. In: Proc. Int. Conf. on Digital Signal Processing, Limassol, June 1995.

Heil, C., Strang, G., Strela, V., 1996. Approximations by translates of refinable functions. Numer. Math. 73, 75–94.

Jia, R.Q., Lei, J.J., 1993. Approximation by multi-integer translates of functions having global support. J. Approx. Theory 72, 2–23.
Lang, M., Guo, H., Odegard, J.E., Burrus, C.S., Wells Jr., R.O., 1995. Nonlinear processing of a shift-invariant DWT for noise reduction. SPIE Conference on Wavelet Applications, Orlando, FL, April, 1995, vol. 2491.
Lang, M., Guo, H., Odegard, J.E., Burrus, C.S., Wells Jr., R.O., 1996. Noise reduction using an undecimated discrete wavelet transform. IEEE Signal Process. Lett. 3 (1), 10–12.
Lawton, W., Lee, S.L., Shen, Z., 1996. An algorithm for matrix extension and wavelet construction. Math. Comp. 65, 723–737.
Lebrun, J., 2000. Balancing MultiWavelets. Ph.D. Thesis, Swiss Federal Institute of Technology (EPFL), Lausanne.
Lebrun, J., Vetterli, M., 1998. Balanced multiwavelets—Theory and design. IEEE Trans. Signal Proc. 46 (4), 119–125.
Lebrun, J., 2003. Theory and design of multiCoiflets, submitted.
Lebrun, J., Vetterli, M., 2001. High order balanced multiwavelets: theory, factorization and design. IEEE Trans. Signal Proc. 49 (9), 1918–1930.
Mallat, S., 1989. A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans. Pattern Anal. Mach. Int. 11 (7), 674–693.
Mallat, S., 1998. A Wavelet Tour of Signal Processing. Academic Press.
Massopust, P.R., Ruch, D.K., Van Fleet, P., 1996. On the support properties of scaling vectors. Appl. Comput. Harmon. Anal. 3, 229–238.
Mintzer, F., 1985. Filters for distortion-free two-band multirate filter banks. IEEE Trans. Acoust. Speech Signal Proc. 33 (3), 626–630.
Mourrain, B., 1999. An introduction to linear algebra methods for solving polynomial equations. In: Lipitakis, E.A. (Ed.), Proc. HERCMA' 9, pp. 179–200.
Park, H., Kalker, T., Vetterli, M., 1997. Groebner bases and multidimensional FIR multirate systems. J. Multidim. Syst. Signal Proc. 8, 11–30.
Pedersen, P., Roy, M.-F., Szpirglas, A., 1993. Computational Algebraic Geometry. Birkhauser, pp. 203–223.
Plonka, G., 1997. Approximation order provided by refinable function vectors. Const. Approx. 13, 221–244.
Plonka, G., Strela, V., 1998. Construction of multi-scaling functions with approximation and symmetry. SIAM J. Math. Anal. 29, 481–510.
Rioul, O., 1993. A discrete-time multiresolution theory. IEEE Trans. Signal Proc. 41 (8), 2591–2606.
Ron, A., Shen, Z., 1997. Affine systems in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator. J. Funct. Anal. 148, 408–447.
Rouillier, F., 1999. Solving zero-dimensional systems through the rational univariate representation. J. Appl. Alg. Eng., Comm. Comp. 9, 433–461.
Selesnick, I.W., 1998. Multiwavelet bases with extra approximation properties. IEEE Trans. Signal Proc. 46, 2898–2908.
Selesnick, I.W., 1999. Interpolating multiwavelet bases and the sampling theorem. IEEE Trans. Signal Proc. 47 (6), 1615–1621.
Selesnick, I.W., 2000a. Smooth wavelet tight frames with zero moments: design and properties. In: Proc. of the Ninth IEEE DSP Workshop.
Selesnick, I.W., 2000b. Balanced multiwavelet bases based on symmetric FIR filters. IEEE Trans. Signal Proc. 48 (1), 184–191.
Selesnick, I.W., Burrus, C.S., 1998. Maximally flat low-pass FIR filters with reduced delay. IEEE Trans. Circuits Syst. II 45 (1), 53–68.
Selesnick, I.W., Sendur, L., 2000. Iterated oversampled filter banks and wavelet frames. In: SPIE Conference on Wavelet Applications, San Diego, August, 2000.

Smith, M.J.T., Barnwell, T.P., 1984. A procedure for designing exact reconstruction filter banks for tree structured sub-band coders. In: Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc., San Diego, CA, March, 1984.

Strang, G., Nguyen, T., 1996. Wavelets and Filter Banks, Wellesley Cambridge Press, Boston.

Strang, G., Strela, V., 1995. Short wavelets and matrix dilation equations. IEEE Trans. Signal Proc. 43, 108–115.

Tian, J., Wells Jr., R.O., Odegard, J.E., Burrus, C.S., 1997. Coifman wavelet systems: approximation, smoothness, and computational algorithms. In: Bristeau, M-O. et al. (Eds.), Computational Science for the 21st Century. John Wiley & Sons.

Vaidyanathan, P.P., 1993. Multirate Systems and Filter Banks. Prentice-Hall, Englewood Cliffs, NJ.

Vetterli, M., Herley, C., 1992. Wavelets and filter banks: theory and design. IEEE Trans. Signal Proc. 40 (9), 2207–2232.

Vetterli, M., Kovačević, J., 1995. Wavelets and Subband Coding. Prentice-Hall, Englewood Cliffs, NJ.

Vetterli, M., Strang, G., 1994. Time-varying filter banks and multiwavelets. In: Proc. of the Sixth IEEE DSP Workshop, Yosemite.

Wei, D., Bovik, A.C., 1998. Generalized coiflets with nonzero-centered vanishing moments. IEEE Trans. Circuits Syst. II 45 (8), 988–1001.