Applications of Computational Commutative Algebra in Signal Processing

John B. Little

Department of Mathematics and Computer Science

College of the Holy Cross

Worcester, MA 01610

email: little@mathcs.holycross.edu

Grostat V

Tulane University

September 4, 2001

- In this talk, we will discuss some recent applications of computational commutative algebra (Gröbner bases, resultants, etc.) to problems in signal processing.

- Basic idea is that many design problems for signal processing systems can be phrased as solving systems of nonlinear polynomial equations.

- So algebraic/numerical techniques for solving polynomial systems can be used.

- However, the systems of practical interest are a challenge for algorithms and their implementations in current generation of software–both algebraic and numerical issues

Outline of Talk

1. Review of digital filters
2. A filter design problem
3. Examples
4. Filter banks and wavelets
5. Balanced multiwavelet systems
6. Examples
7. Conclusions

Digital Signals

A *digital* signal is a quantized function of a *discrete* variable.

- only have signal values at $n\Delta t$, n an integer, Δt the *sampling interval*

- signal values are elements of a discrete set.

For simplicity, we ignore quantization effects and study *discrete-time, complex-valued* signals.

Can write a signal as a complex-valued function of an integer variable, $x[n]$, or equivalently, a list of complex numbers indexed by the integers:

$$\cdots, x[-2], x[-1], x[0], x[1], x[2], \cdots$$

Filters

Signal processing operations can be described mathematically by means of functions Γ:

$$\Gamma : \{\text{signals}\} \to \{\text{signals}\}.$$

We mainly consider operations with the following properties

1. (Linearity) If $z[n] = cx[n] + y[n]$, then
$$\Gamma(z)[n] = c\Gamma(x)[n] + \Gamma(y)[n]$$

2. (Shift-invariance) If $y[n] = x[n+k]$ for all n, k fixed, and the outputs are $w = \Gamma(x)$, $z = \Gamma(y)$, then
$$z[n] = \Gamma(y)[n] = \Gamma(x)[n+k] = w[n+k]$$

A Γ satisfying both of these properties is said to be a (linear, shift-invariant) *filter*.

A "Moving Average" Filter

For any input signal $x[n]$, define

$$\begin{aligned} y[n] &= \Gamma(x)[n] \\ &= \left(\sum_{k=0}^{4} x[n-k] \right) /5 \end{aligned}$$

If $x[n] = \cos(n\pi/2)$,

$$\cdots, 1, 0, -1, 0, 1, 0, -1, 0, 1, \cdots$$

for instance, then $\Gamma(x)[n] =$

$$\cdots, 0, 1/5, 0, -1/5, 0, 1/5, 0, -1/5, 0, \cdots$$

It is easy to see that this satisfies the linearity and shift-invariance properties, so this defines an filter.

Impulse Response of a Filter

A nice result about filters: Let δ be

$$\cdots, 0, 0, 1, 0, 0, \cdots$$

(1 at time $n = 0$) – called a *unit impulse* input. $\Gamma(\delta)$ is called the *impulse response* of the filter.

If we know $h = \Gamma(\delta)$, then we also know the output of the filter on any other signal x. Reason: We can write

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

Hence by linearity and shift-invariance

$$y[n] = \Gamma(x)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k].$$

The Transfer Function

Package $x[n]$ by its "z-transform", the series

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

Do same for the impulse response $h[n]$ and the output $y[n]$ for input $x[n]$:

$$H(z) = \sum_{n=-\infty}^{\infty} h[n]z^{-n}, \ Y(z) = \sum_{n=-\infty}^{\infty} y[n]z^{-n}$$

Coefficient of z^{-n} in the product $H(z)X(z)$ is the discrete convolution:

$$\sum_{k=-\infty}^{\infty} h[k]x[n-k],$$

same as $y[n]$ (!)

$\Rightarrow Y(z) = H(z)X(z)$. $H(z)$ is called the *transfer function*.

Fourier Domain

Note:

$$H(e^{i\omega}) = \sum_{n=-\infty}^{\infty} h[n]e^{-in\omega}$$

is the (discrete-time) Fourier transform of h.

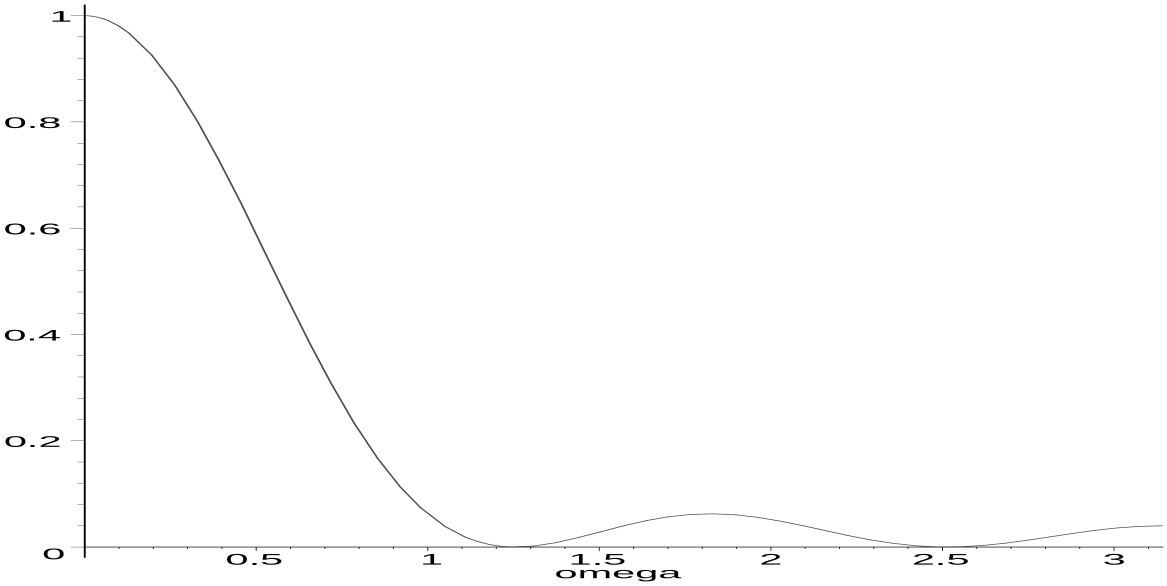
So $H(z)$ also determines the *frequency response* characteristics of the filter on input signals.

For instance, with input a pure sinusoid $x[n] = e^{in\omega_0}$ the output $y[n] = \Gamma(x)[n]$

$$\begin{aligned}
&= \sum_{k=-\infty}^{\infty} h[k]x[n-k] \\
&= \sum_{k=-\infty}^{\infty} h[k]e^{i(n-k)\omega_0} \\
&= \left(\sum_{k=-\infty}^{\infty} h[k]e^{-ik\omega_0}\right) e^{in\omega_0} \\
&= H(e^{i\omega_0})x[n]
\end{aligned}$$

Frequency Response

The magnitude of the frequency response for the 5-point moving average filter from before:



This is an example of a *"low-pass"* filter – it "passes" sinusoids of small normalized frequency and "stops" sinusoids of large normalized frequency.

Filter Design

Goal is to design filters to perform various operations on signals, by designing $H(z)$.

- Low-pass filters: remove high-frequency components of signals – typically *smooth out* or *blur* signals (can be used to remove high-frequency "noise")

- High-pass filters: remove low-frequency components – typically focus on fine details, rapid changes (can be used to detect features)

Typical problems: design a *low-pass* or *high pass* filter to meet certain specifications.

Selesnick-Burrus Design Paradigm

reference: Selesnick and Burrus, *IEEE Trans. Circuits and Systems II*, v. 45(1), January 1998.

Given the low-pass filter $H(z)$, consider

- *magnitude response:* $F(\omega) = |H(e^{i\omega})|^2$
- *group delay:* $G(\omega) = \frac{d}{d\omega}\Big(\arg(H(e^{i\omega}))\Big)$

We would like to, simultaneously,

1. make the passband *flat*,
2. make the stopband *flat*, and
3. make the group delay $G(\omega)$ *flat* across the passband.

Specifications

(The Selesnick-Burrus approach can also be extended to include extra parameters controlling the width of the transition band between the passband and the stopband, etc. For simplicity, we only consider the basic form.)

Provided $N \geq K + L + M + 1$, can find $H(z)$ satisfying:

- $h[0] = 1$ and $F^{(2i)}(0) = 0$, for $i = 1, \ldots M$. (*flatness in passband*).

- $z = -1$ a zero of $H(z)$ of order K (*flatness in stopband* $\omega = \pi$),

- $G^{(2j)}(0) = 0$, for $j = 1, \ldots, L$ (*flat group delay*)

(F and G are even functions, so odd-order derivatives at 0 automatically zero.)

The System of Equations

- Computing derivatives of F and G explicitly give *polynomial equations* in the filter coefficients $h[n]$, or in their *moments*

$$m[k] = \sum_{n=0}^{N-1} n^k h[n], k \geq 0 \qquad (1)$$

- $h[n]$ and $m[k]$ are related by the linear equations (1), so can use either set of variables.

- Using filter coefficients gives Toeplitz form equations – perhaps better for straight numerical approaches

- Using moments simplifies form of equations significantly – better for symbolic approaches

Form of The Equations in $m[i]$

- Flatness equations (quadratic in the $m[i]$):

$$\binom{2i}{i} m[i]^2 + 2\sum_{\ell=0}^{i-1}\binom{2i}{\ell}(-1)^{i+\ell} m[\ell]m[2i-\ell]$$

$$= 0,\ i = 1,\ldots,M.$$

$$\sum_{\ell=0}^{j}\left(1-\frac{2\ell}{2j+1}\right)\binom{2j+1}{\ell}(-1)^{\ell} m[\ell]m[2j+1-\ell]$$

$$= 0,\ j = 1,\ldots,L.$$

- Zeros at $z=-1$: remainder of $H(z)$ on division by $(1+z)^K$ is zero – K linear equations on $m[i]$

First Simplification

This looks like an underdetermined system.

But can express $m[k]$ for $k \geq N$ in terms of $m[0], \ldots, m[N-1]$ by solving linear equations, and substitute.

Normalization: $m[0] = 1$

Get $L + M + K = N - 1$ equations in $N - 1$ variables $m[1], \ldots, m[N-1]$.

Expect only finitely many solutions. (Mainly interested in real ones.)

Here is an example.

Some Notation

We will use the following notation for the ideals generated by the polynomials from the Selesnick-Burrus equations:

- $J_{L,M}$ is the ideal generated by the $L+M$ quadratic flatness equations (doesn't depend on K)

- $I_{K,L,M}$ is the ideal generated by these, plus the K linear equations from the zeroes at $z=-1$.

“Easy” and “Hard” Cases

- (”Region I”) The *easier* cases are
$$\left\lfloor \frac{M-1}{2} \right\rfloor \le L \le M$$
(In our example, $L = M = 6$, so in this case!) All variables except $m[1]$ can be eliminated using linear algebra techniques.

- (”Region II”) The *harder* cases are
$$0 \le L \le \left\lfloor \frac{M-1}{2} \right\rfloor - 1$$

Example: if $L = 0$, there are 2^M solutions of the design equations in these cases(!) (One way to see this: the generators for $I_{K,0,M}$ are already “almost” a Gröbner basis for $I_{K,0,M}$ with respect to a *grevlex* order with $m[N-1] > m[N-2] > \cdots > m[1]$.)

Problems

- Understand the Region II cases:

  1. for given L, M, how many solutions?

  2. how many are real?

  3. how many yield monotone decreasing $F(\omega)$?

  4. etc.

- Develop an efficient method to solve the S-B equations in this case

Some Results

$$\#\text{sols} = \dim \mathbf{C}[m[1], \ldots, m[N-1]]/I_{1,L,M}$$

in "Region II" ($K = 1$). $L = 0$ column omitted: 2^M.

M/L	1	2	3	4	5
5	16				
6	26				
7	48	24			
8	78	38			
9	152	66	32		
10		112	50		
11			86	40	
12			142	62	
13				106	48

Cases $M \leq 7$ check with Selesnick-Burrus. $M \geq 8$ new; blank space below Region II boundary means (?)

The "Best" Solution Method

Best general method I have found is a "hybrid" approach

- The quadratic flatness conditions are homogeneous if weight($m[i]$]) $= i$

- In fact, the variety defined by the quadratics, $V(J_{L,M})$, is a type of "warped" *rational scroll* – it has a parametrization via rational (monomial) curves with "directrix" a linear variety.

- It is always easy to find this parametrization from a *lex* GB of $J_{L,M}$ with $m[N-1] > m[N-2] > \cdots > m[1]$.

- Substituting from this parametrization reduces the number of variables considerably to $M - L$.

"Best" Solution Method, cont.

- The equations left have quite low degrees in all variables except $m[1]$, and elimination to a univariate polynomial in $m[1]$ can be done by either
  - Gröbner bases
  - sparse resultants,
  - or pairwise ordinary (Sylvester) resultants,

  can eliminate all the variables except $m[1]$ and get a univariate polynomial in the ideal $I_{K,L,M}$.

- By resultants, the actual computation is fast, but there are always extraneous factors in the result.

Here's the case $M = 11, L = 4$ (elimination via lex GB):

A Theoretical Result

In fact, using this approach, we can prove:

Theorem 1 *In the cases* $M = 2L + 3$, $L \geq 0$, $K = 1$ *("corners" in Region* **II** *boundary), the S-B equations always have* $8(L + 1)$ *distinct solutions, all of multiplicity 1. Of these* $2(L + 2)$ *are real* ($L + 2$ *different filters, up to time reversal). Two of these have monotonic square magnitude response.*

Solutions yield square magnitude responses that have properties similar to those of Sturm-Liouville eigenvalue problems(?)

Similar techniques should work on other "diagonals" $M = 2L + \ell$ with $\ell \geq 4$ also.

Things That Make These Systems Hard

- High degrees of univariate polynomials

- For large M, L, the univariate polynomial has *gigantic* coefficients.

- A *BIG* numerical analysis problem. Need to use very many decimal digits in floating-point computation to get accurate roots.

- In fact this is the current "bottle-neck" in the cases I have worked out.

Wavelets

In recent years, wavelets have become an attractive alternative or complement to more traditional Fourier-based methods in signal and image processing, PDE, etc. for applications such as

- signal analysis
- denoising
- data compression
- finite element methods

Wavelet expansions express signals as sums of components at different time/distance scales ("detail levels") rather than as sums of components at different frequencies.

Often better for strictly time-limited signals.

Multi-Resolution Analysis

Consider *continuous-time* signals in $L^2(\mathbf{R})$.

A *multiresolution analysis* of L^2 is a sequence of closed subspaces V_j, $j \in \mathbf{Z}$ satisfying:

1. $V_j \subset V_{j+1}$ for all j
2. $f(t) \in V_j \Leftrightarrow f(2t) \in V_{j+1}$
3. $\cap_{j\in\mathbf{Z}} V_j = \{0\}$
4. $\overline{\cup_{j\in\mathbf{Z}} V_j} = L^2(\mathbf{R})$
5. There exists $\phi \in V_0$ such that $\{\phi(t-k) : k \in \mathbf{Z}\}$ is an orthonormal basis of V_0.

ϕ is called the *scaling function*.

MRA to Wavelets

Let W_j be the orthogonal complement of V_j in V_{j+1}, so $V_{j+1} = V_j \oplus W_j$. Then

$$L^2(\mathbf{R}) = \overline{V_0 \oplus W_0 \oplus W_1 \oplus W_2 \oplus \cdots}$$

We assume that there is some $\psi \in W_0$, called the *mother wavelet*, such that $\{\psi(t-k) : k \in \mathbf{Z}\}$ is an orthonormal basis of W_0. From 2 in definition of MRA, W_j has an orthonormal basis $\{2^{j/2}\psi(2^j t - k) : k \in \mathbf{Z}\}$ formed from scaled, dilated, and translated versions of ψ.

General wavelet expansion of a signal f looks like this:

$$f(t) = \sum_k a_{0k}\phi(t-k) + \sum_{j=0}^{\infty} \sum_k b_{jk} 2^{j/2} \psi(2^j t - k)$$

Dilation and Wavelet Equations

Since V_0 and W_0 are contained in V_1, and 2 in the definition of MRA implies that $\{\phi(2t-k) : k \in \mathbf{Z}\}$ is a basis for V_1, we have

- the *dilation equation*:

  $$\phi(t) = \sqrt{2} \sum_{k \in \mathbf{Z}} h_0[k] \phi(2t-k)$$

  for some coefficients $h_0[k]$, and

- the *wavelet equation*:

  $$\psi(t) = \sqrt{2} \sum_{k \in \mathbf{Z}} h_1[k] \phi(2t-k)$$

  for some coefficients $h_1[k]$.

Haar Wavelets

The simplest example of this setup is the Haar wavelet system.

The Haar scaling function is the "box" function

$$\phi(t) = \begin{cases} 1 & \text{if } 0 \le t < 1 \\ 0 & \text{otherwise} \end{cases}$$

The dilation equation is

$$\phi(t) = \phi(2t) + \phi(2t - 1),$$

that is, $h_0[0] = h_0[1] = \frac{1}{\sqrt{2}}$ and all others 0.

The Haar wavelet is

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

The wavelet equation is

$$\psi(t) = \phi(2t) - \phi(2t - 1)$$

so $h_1[0] = \frac{1}{\sqrt{2}}, h_1[1] = -\frac{1}{\sqrt{2}}$, all others 0.

Analysis and Synthesis Equations

Let $f \in V_{j+1} = V_j \oplus W_j$. Then

$$f(t) = \sum_k a_{j+1,k} 2^{(j+1)/2} \phi(2^{j+1} t - k), \text{ or}$$

$$f(t) = \sum_k a_{j,k} 2^{j/2} \phi(2^j t - k) + \sum_k b_{j,k} 2^{j/2} \psi(2^j t - k)$$

From the dilation and wavelet equations, get

- the *analysis equations*

$$\begin{aligned} a_{j,k} &= \sum_n h_0[n-2k] a_{j+1,n} \\ b_{j,k} &= \sum_n h_1[n-2k] a_{j+1,n} \end{aligned}$$

- the *synthesis equation*

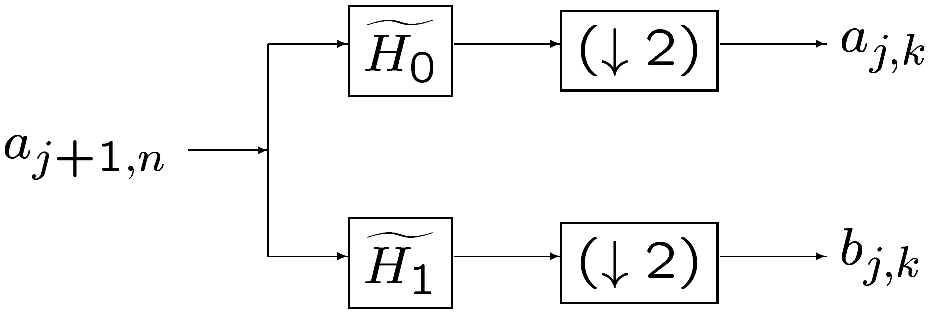
$$a_{j+1,n} = \sum_k (h_0[n-2k] a_{j,k} + h_1[n-2k] b_{j,k})$$

relating these expansion coefficients.

Wavelets and Filter Banks

Key observation here is that these have *same form* as the outputs of 2-channel *filter banks* used for multirate signal processing (sampling rate conversion, subband processing, etc.)

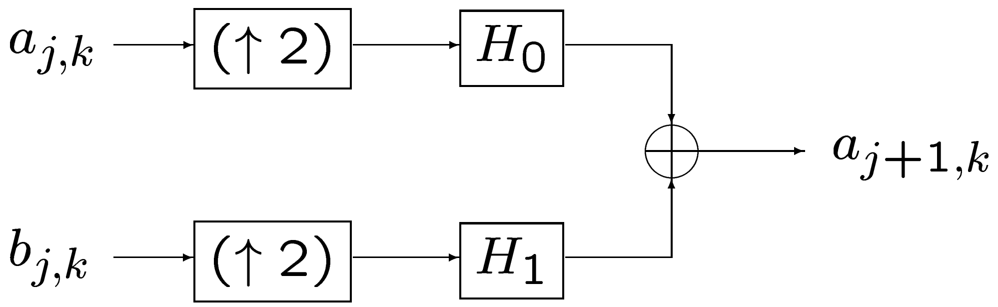
Analysis bank:



(where $\widetilde{H}_i(z) = H_i(1/z)$ and $(\downarrow 2)$ is *downsampling*, or *decimation* by 2)

Filter Banks, cont.

Synthesis bank:



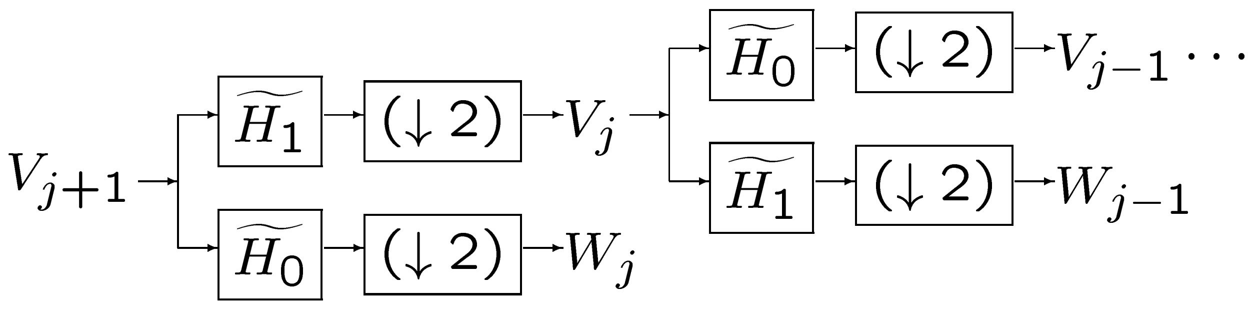
where $(\uparrow 2)$ is *upsampling*, or *"interpolation"* by 2 (expanding with zeroes).

H_0 is the *synthesis low-pass filter*; H_1 is the *synthesis high-pass filter*. For example, in Haar case, H_0 is two-point averaging filter, H_1 is two-point difference filter.

"Fast Wavelet Transform"

The filter bank approach to wavelet systems is fruitful for several reasons:

- *Iteration of the low-pass analysis branch*, starting from $a_{j+1,k}$ for some large $j+1$ (or sampled signal values) gives a very fast way to compute wavelet expansions:



Wavelet System Design

Can *design wavelet systems* by designing the analysis (or synthesis) bank filters.

- Often want H_i to be FIR filters, $\Rightarrow \phi, \psi$ have *compact support*.

- Often want $\{\phi(t-k), \psi(2^j t - k)\}$ *orthogonal* in $L^2(\mathbf{R})$ – Sufficient: *"double-shift orthogonality"*:

$$\sum_{n \in \mathbf{Z}} h_i[n] h_j[n-2k] = \delta[i-j]\delta[k] \qquad (2)$$

  (condition also implies "perfect reconstruction" – synthesis bank inverts analysis bank).

- Often want wavelet expansions to have good approximation properties – Sufficient: $H_0(z)$ has *zero order* $p \geq 1$ *at* $z = -1$ $(\omega = \pi)$. (Condition also related to *smoothness* of ϕ, ψ.)

To Multiwavelets

- There are limitations on what can be accomplished this way, though.

- For instance, a symmetric orthogonal FIR filter can have only two nonzero coefficients.

- To get around these limitations, people have introduced several generalizations of MRA, etc.

- We'll consider one – *multiwavelets*.

Multiwavelet Systems

Basic idea here is to allow *more than one* scaling function, mother wavelet:

- $V_j = \text{Span}\{\phi_0(2^j t - k), \ldots, \phi_{r-1}(2^j t - k) : k \in \mathbf{Z}\}$

- $W_j = \text{Span}\{\psi_0(2^j t - k), \ldots, \psi_{r-1}(2^j t - k) : k \in \mathbf{Z}\}$

- *dilation equation* becomes

$$\Phi(t) = \sqrt{2} \sum_{n \in \mathbf{Z}} C_0[n] \Phi(2t - n)$$

  where $\Phi = (\phi_0, \ldots, \phi_r)^T$, and $C_0[n]$ is an $r \times r$ matrix for each n. Similarly for *wavelet equation*.

Multiwavelets for Scalar Signals

Get multiwavelet analysis and synthesis filter banks similar to the ordinary case.

Although one natural way to use multiwavelets is for *vector-valued signals*, we will consider a different use.

For simplicity, take $r = 2$, and write

$$C_0[n] = \begin{pmatrix} h_0[2n] & h_0[2n+1] \\ h_1[2n] & h_1[2n+1] \end{pmatrix},$$

then the entries in the $C_0[n]$ are split into coefficients from *two scalar filters*, h_i, $i = 0, 1$. Similarly for matrix filter $C_1[n]$ from the wavelet equation.

Group consecutive sampled values from a scalar signal into 2-component vectors.

Multiwavelets for Scalar Signals, cont.

- The 2-channel multiwavelet filter banks can be regrouped into *4-channel scalar filter banks* (with interleaving of outputs).

- Condition for *orthogonality* (analogous to (2) above) becomes

$$\sum_{n \in \mathbf{Z}} h_i[n] h_j[n-4k] = \delta[i-j]\delta[k] \qquad (3)$$

- Natural analog of the "zeroes at $z = -1$ ($\omega = \pi$)" condition for approximation order is the *"balanced" property*: Selesnick, Lebrun and Vetterli, etc.

Balanced Multiwavelet Systems

references:

1. Selesnick, Multiwavelet Bases with Extra Approximation Properties,

2. Lebrun and Vetterli, Balanced Multiwavelets Theory and Design,

Both: *IEEE Trans. Signal Proc.*, v. 46, 1998.

Def. The $r = 2$ multiwavelet system with filters h_0, h_1 as above is *balanced* of order p if $(1+z^{-1}+z^{-2}+z^{-3})^p$ divides $H_0(z)+\alpha_p(z^4)H_1(z)$, where α_p is given by an explicit formula yielding:

$$\begin{aligned} \alpha_1 &= 1 \\ \alpha_2 &= \frac{3 - z^{-1}}{2} \\ \alpha_3 &= \frac{15 - 10z^{-1} + 3z^{-2}}{8}, etc. \end{aligned}$$

Balanced Multiwavelets, cont.

An orthogonal multiwavelet system which is balanced of order p has approximation order (at least) p:

The polynomial signals $1, t, t^2, \ldots, t^{p-1}$ can be *exactly* decomposed (on a finite interval) into linear combinations of the scaling functions $\phi_i(t)$ and their integer translates.

The balancing order is also related to the order of smoothness of the $\phi_i(t)$.

Conclusions

- Methods from computational commutative algebra are potentially very useful in signal processing, to solve the polynomial systems coming from filter design, wavelet system design problems, etc.

- Ad hoc approaches are often necessary, though, because of complexity of the problems of practical interest

- These applications pose a number of challenges to the "state of the art" in polynomial system solving.

Additional Bibliography

1. Faugère, J-C., Moreau de Saint-Martin, F., Rouillier, F. *Design of regular nonseparable bidimensional wavelets using Gröbner basis techniques*, IEEE Trans. Signal Process. **46** (1998), 845-856.

2. Kalker, T., Park, H. and Vetterli, M. *Gröbner bases techniques and multidimensional FIR multirate systems*, Multidim. Systems Signal Process. **8** (1997) 11-30.

3. Park, H. and Woodburn, C. *An algorithmic proof of Suslin's stability theorem*, J. Algebra **178** (1995), 277-298.

4. forthcoming special issue of Journal of Symbolic Computation, 2001