ECE 6123
Advanced Signal Processing: Multi-Rate Signal Processing, Multi-Resolution Decomposition and Wavelets

Peter Willett

Fall 2017

1 Decimation and Interpolation

1.1 Decimation

This is a review from basic Digital Signal Processing, but please bear with it. First, consider the *decimation* operation

$$y[n] \;=\; x[nD] \tag{1}$$

in which D is some integer – say, for $D = 2$ this amounts to forming $y[n]$ from the even-indexed samples of $x[n]$. The transform relationship comes from doing this in two steps:

$$y[n] = u[nD] \quad \text{where} \quad u[n] \;=\; \begin{cases} x[n] & n = mD \\ 0 & \text{else} \end{cases} \tag{2}$$

We begin with the *noble identity*

$$\sum_{p=0}^{N-1} e^{j2\pi pm/N} \;=\; N \sum_{q=-\infty}^{\infty} \delta[m - qN] \tag{3}$$

which is easy to show via the geometric series formula (special case of summing N 1's when $m = qN$).

Now we write

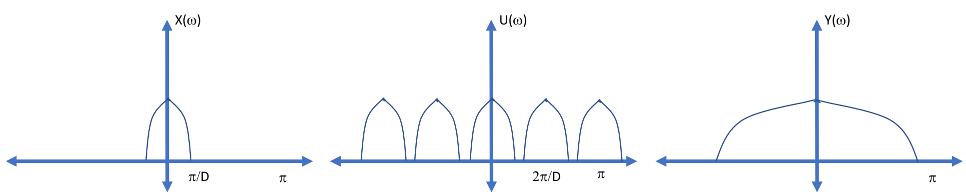
$$u[n] = \frac{1}{D} \sum_{k=0}^{N-1} e^{j2\pi kn/D} x[n] \tag{4}$$

so

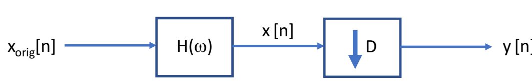
$$
\begin{aligned}
U(z) &= \sum_{n=-\infty}^{\infty} \frac{1}{D} \sum_{k=0}^{N-1} e^{j2\pi kn/D} x[n] z^{-n} && (5)\\
&= \frac{1}{D} \sum_{k=0}^{N-1} X(ze^{-j2\pi k/D}) && (6)\\
U(\omega) &= \frac{1}{D} \sum_{k=0}^{N-1} X\left(\omega - 2\pi k/D\right) && (7)
\end{aligned}
$$

It is easy to see that we have

$$
\begin{aligned}
Y(z) &= \sum_{n=-\infty}^{\infty} u[nD] z^{-n} && (8)\\
&= U(z^{1/D}) && (9)\\
&= \frac{1}{D} \sum_{k=0}^{N-1} X(z^{1/D} e^{-j2\pi k/D}) && (10)\\
Y(\omega) &= U(\omega/D) && (11)\\
&= \frac{1}{D} \sum_{k=0}^{N-1} X\left(\frac{\omega - 2\pi k}{D}\right) && (12)
\end{aligned}
$$



See above for an illustration. Note that in the above figure the bandwidth of $X(\omega)$ is constrained to be less than π/D; if this were not so we would have *aliasing*. We are not interested in aliasing in the current discussion. And in any case we could pre-filter (with an "anti-aliasing" filter) the signal $x[n]$ to make sure that no frequency components above π/D remain, as in the figure below.



It is useful to note that if $h[n]$ is *finite impulse-response* (FIR) of length L then while each $y[n]$ requires L operations, that means that only L/D operations are needed per sample of $x[n]$. It is also worth mentioning that the case $D = 2$

$$Y(z) = \frac{1}{2}\left(X(z^{1/2}) + X(-z^{1/2})\right) \tag{13}$$

$$Y(\omega) = \frac{1}{2}\left(X\left(\frac{\omega}{2}\right) + X\left(\frac{\omega}{2}+\pi\right)\right) \tag{14}$$

will especially interest us.

1.2 Interpolation

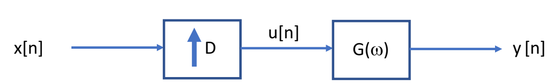
With reference to the previous discussion, the *interpolation* essentially refers to starting with $y[n]$ and re-formulating $u[n]$. Switching input to $x[n]$ we thence have

$$u[n] = \begin{cases} x[m] & n = mD \\ 0 & \text{else} \end{cases} \tag{15}$$

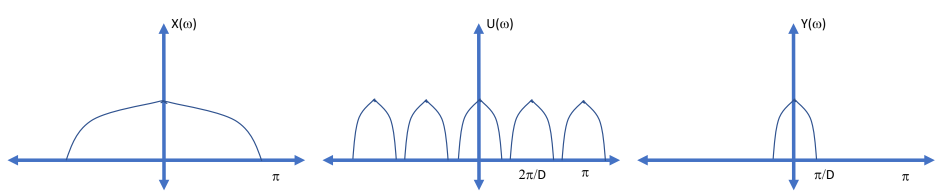
meaning that between each sample of $x[n]$ we simply insert $(D-1)$ 0's. It's obvious that we have

$$\begin{aligned} U(z) &= \sum_{n=-\infty}^{\infty} u[n]z^{-n} && (16)\\ &= \sum_{m=-\infty}^{\infty} u[m]z^{-nD} && (17)\\ &= X(z^D) && (18)\\ U(\omega) &= X(\omega D) && (19) \end{aligned}$$

The system is as shown below.



And the spectra are as shown here.

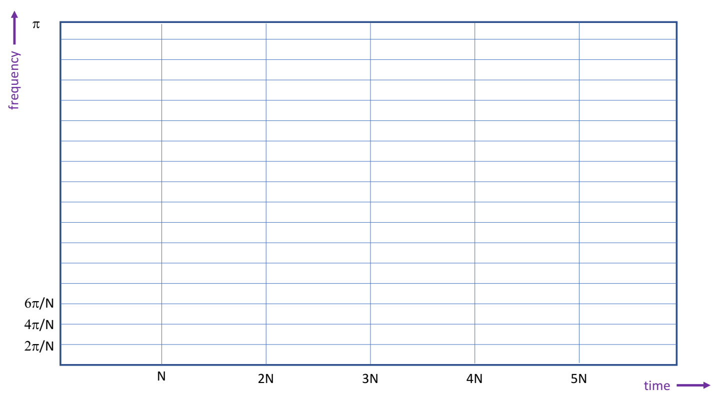


Note that replica spectra appear in $U(\omega)$; in many applications it is desirable to suppress these so we have above represented the final output $y[n]$ as being after another filter. It is commonly known as an "interpolation" filter since its function / effect is to insert smoothed values over the $(D-1)$ 0's that are in $u[n]$. Note, again, that since only every D^{th} sample of $u[n]$ is non-zero only L/D operations per output of $y[n]$ are needed for this interpolation operation.

2 Filter Banks

2.1 Transforming Data via the Block-DFT

A transformation of data that concentrates signal energy in a few samples makes for better signal understanding, representation, manipulation and coding. One such transformation is the block-DFT.



The illustration above is intended to illustrate the shape of the transformed components to the block-DFT. The block-DFT has several nice properties:

- It is invertible – no information is lost.
- It is orthogonal – if the input is white, the transformed components are white, too.
- It is efficient – via the FFT it requires only $\log_2(N)$ operations per output.

There is one disadvantage, however, and it is perhaps best illustrated in the notional plot just shown. It is this: high-frequency components correspond to features in the original signal that are of short duration. However, the time-swath of the DFT is the same for all frequencies: that is, a high-frequency component measures the amount of high-frequency energy over the entire block of N data. Since high-frequency components are by their nature fast-changing it would make more sense to have them measuring energy at those frequencies over shorter periods of time as compared to lower-frequency components that measure long-term trends and smooth features.

Now, a transformation that is invertible is really, in linear-algebraic terms, a change of basis. How is a DFT that? Consider

$$\mathbf{X} \ = \ \mathbf{W}\mathbf{x} \tag{20}$$

in which

$$\mathbf{W} \ \equiv \ \begin{pmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & W_N^3 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & W_N^6 & \dots & W_N^{2(N-1)} \\ 1 & W_N^3 & W_N^6 & W_N^9 & \dots & W_N^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & W_N^{3(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{pmatrix} \tag{21}$$

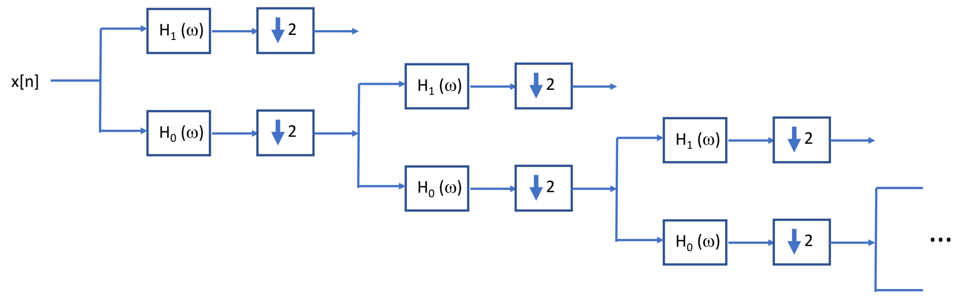
and of course

$$W_N \ \equiv \ e^{-j2\pi/N} \tag{22}$$

That is, the DFT operation is really a matrix/vector multiplication. We know that the FFT gives us an efficient way to implement it – better than the N^2 operations that a direct matrix/vector multiply would normally take – and we also know that $\mathbf{W}\mathbf{W}^H = N\mathbf{I}$, meaning that it is orthogonal. Are there other matrices that have the same properties?

2.2 Transforming Data via a Filter Bank

The question was just posed as to whether there any other transformation matrices that have the same three nice properties as the block-DFT. The answer is most certainly *yes* – see the figure below – and, even better, we have one such that does avoids the block-DFT's "disadvantage" in terms of compatibility of time-averaging to frequency. This an "octave" filter bank, and of course other decimation factors could be imagined.

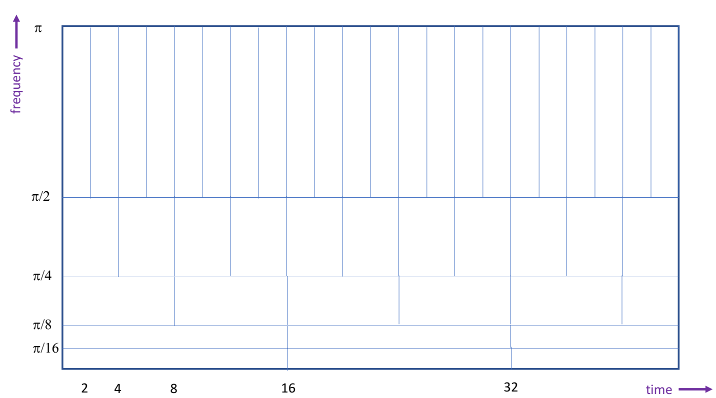


We will treat invertibility and orthogonality soon. But as for efficiency, let us assume that both filters $H_0(\omega)$ and $H_1(\omega)$ are of length L. Then the upper-most branch requires $L/2$ operations per input sample of $x[n]$ and so does the lower branch. The second level likewise $2L/4$. Overall, the computational load is

$$L \sum_{k=0}^{\infty} 2^{-k} \;=\; 2L \tag{23}$$

operations per input $x[n]$, assuming that the filter-back goes on "forever." In fact – and rather unusually for an FIR filter – we will not be interested in especially long filters. A value $L = 8$ is quite normal.

Let's just pretend that $H_0(\omega)$ is a perfect LPF with cutoff at $\pi/2$ and that $H_1(\omega)$ is a perfect HPF also at $H_0(\omega)$. Then the time-frequency plot would look like the below.



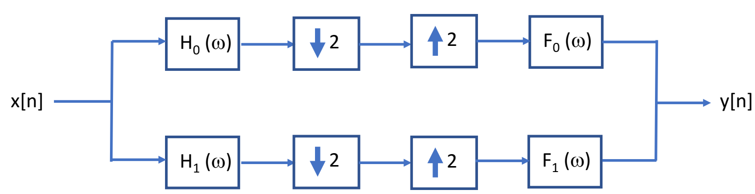
(Please note that only four levels of decimation have been represented here; in general this is arbitrary, and in principle it can go on . . . forever.) The point is that the representation may be more appropriate than the block DFT since components at higher frequencies use data over shorter time windows.

So we have efficiency and appropriateness. Now it is time to discuss invertibility and orthogonality. Before we begin, however, let us examine the notional idea that $H_0(\omega)$ and $H_1(\omega)$ are a perfect LPF and HPF. It seems relatively clear that such surgical splitting avoids aliasing and enables reconstruction. The cost, however, is that L would seem to need to be very large. But wait! The block DFT actually *allows* aliasing and uses filters of length N? It appears that *some* kinds of aliasing do not destroy information.

3 Perfect-Reconstruction Filter Banks

3.1 The Half-Band Condition

We wish to change the basis via a filter bank, but we demand that we lose no information as we do so – we could call this *invertibility* of *perfect reconstruction*. The basic building block for analysis is as below, and clearly we want $y[n] = x[n-l]$ for some l. Note that the two middle blocks (down-sample then up-sample) may look like they cancel; but they do not, since their back-to-back pair amounts to setting every other sample to zero.



Using what we have discovered about sample-rate conversion, we have after the up-sample operation on the upper branch

$$\frac{1}{2}\left(H_0(z)X(z) + H_0(-z)X(-z)\right) \tag{24}$$

which means

$$\begin{aligned} Y(z) &= \frac{1}{2}\left(H_0(z)F_0(z)X(z) + H_0(-z)F_0(z)X(-z)\right) \\ &\quad + \frac{1}{2}\left(H_1(z)F_1(z)X(z) + H_1(-z)F_1(z)X(-z)\right) \end{aligned} \tag{25}$$

which means that in order that we have $y[n] = x[n-l]$ we need

$$H_0(z)F_0(z) + H_1(z)F_1(z) = 2z^{-l} \tag{26}$$

$$H_0(-z)F_0(z) + H_1(-z)F_1(z) = 0 \tag{27}$$

This turns out to be way under-determined. So we adopt the common choice

$$F_0(z) \;=\; H_1(-z) \tag{28}$$

so that we require

$$F_1(z) \;=\; -H_0(-z) \tag{29}$$

in order to have (27) be satisfied.

It is interesting to substitute (28) and (29) into (26) and then to evaluate the result at both z and $-z$; we get

$$\begin{aligned} H_0(z)F_0(z) - H_0(-z)F_0(-z) &= 2z^{-l} && (30)\\ H_0(-z)F_0(-z) - H_0(z)F_0(z) &= 2(-1)^l z^{-l} && (31) \end{aligned}$$

which implies that l must be odd.

It is also convenient to define

$$\begin{aligned} P(z) &\equiv z^l H_0(z)F_0(z) && (32)\\ &= z^l H_0(z)H_1(-z) && (33) \end{aligned}$$

so since $(-z)^l = -z^l$ (as l is odd), we can write

$$P(z) + P(-z) = 2 \tag{34}$$

This is the *half-band* condition, and is perhaps familiar as the first Nyquist criterion for pulse-shaping from digital communications. The half-band condition is sufficient (and by no means necessary!) for perfect reconstruction. And in fact the half-band condition is itself underdetermined.

3.2 The Haar Example

Here we have

$$\begin{aligned} H_0(z) &= \frac{1}{\sqrt{2}}(1+z^{-1}) && (35)\\ F_0(z) &= \frac{1}{\sqrt{2}}(1+z^{-1}) && (36)\\ H_1(z) &= \frac{1}{\sqrt{2}}(1-z^{-1}) && (37)\\ F_1(z) &= -\frac{1}{\sqrt{2}}(1-z^{-1}) && (38) \end{aligned}$$

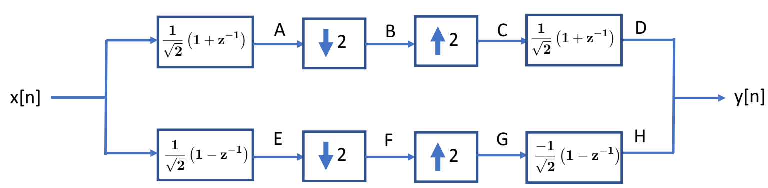
It is interesting that while (35) is the Haar filter[1] and that (37) & (38) follow from (28) & (29) applied to (35) & (36), the actual choice of (36) is really quite arbitrary. In fact, inserting (35) into (26) gives us

$$
\begin{aligned}
[F_0(z) - F_0(-z)] \; + \; z^{-1}[F_0(z) + F_0(-z)] &= 2\sqrt{2}z^{-l} \qquad (39) \\
(\text{odd-indexed terms}) + z^{-1}(\text{odd-indexed terms}) &= \sqrt{2}z^{-l} \qquad (40)
\end{aligned}
$$

This implies that there are only two adjacent non-zero terms in $F_0(z)$; it makes sense to choose a first-order $F_0(z)$, but we still have

$$f_0[1]z^{-1} + z^{-1}(f_0[0]) = 2\sqrt{2}z^{-l} \qquad (41)$$

from (39). For symmetry and linear phase we choose (36).



The figure above shows a scaled version of the Haar system. We have:

A: $\frac{1}{\sqrt{2}}\{\ldots, x[0]+x[-1],\ x[1]+x[0],\ x[2]+x[1],\ x[3]+x[2],\ x[4]+x[3],\ \ldots\}$

B: $\frac{1}{\sqrt{2}}\{\ldots, x[0]+x[-1],\ x[2]+x[1],\ x[4]+x[3],\ \ldots\}$

C: $\frac{1}{\sqrt{2}}\{\ldots, x[0]+x[-1],\ 0,\ x[2]+x[1],\ 0,\ x[4]+x[3],\ \ldots\}$

D: $\frac{1}{2}\{\ldots, x[0]+x[-1],\ x[0]+x[-1],\ x[2]+x[1],\ x[2]+x[1],\ x[4]+x[3],\ \ldots\}$

E: $\frac{1}{\sqrt{2}}\{\ldots, x[0]-x[-1],\ x[1]-x[0],\ x[2]-x[1],\ x[3]-x[2],\ x[4]-x[3],\ \ldots\}$

F: $\frac{1}{\sqrt{2}}\{\ldots, x[0]-x[-1],\ x[2]-x[1],\ x[4]-x[3],\ \ldots\}$

G: $\frac{1}{\sqrt{2}}\{\ldots, x[0]-x[-1],\ 0,\ x[2]-x[1],\ 0,\ x[4]-x[3],\ \ldots\}$

H: $\frac{1}{2}\{\ldots, x[-1]-x[0],\ x[0]-x[-1],\ x[1]-x[2],\ x[1]-x[2],\ x[3]-x[4],\ \ldots\}$

y: $\{\ldots, x[-1],\ x[0],\ x[1],\ x[2],\ x[3],\ \ldots\}$

[1] The Haar filter is just a running two-sampler average.

The last line (the final output $y[n]$) is obtained from adding the signals at D and H. Note that it is identical to the input $x[n]$ – perfect recovery! – except for a delay by a single time-step ($l = 1$).

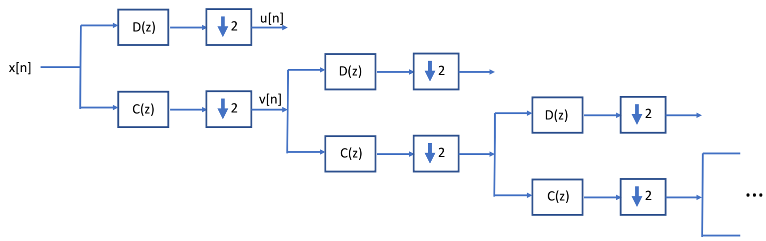
One more note on the Haar system is appropriate. Consider the octave filter bank structure, with the Haar filter and the change-of-basis interpretation (20). Stopping after three levels, the matrix $\mathbf{W}$ is

$$\mathbf{W} = \tag{42}$$
$$\begin{pmatrix}
a & -a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & a & -a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & a & -a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & a & -a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a & -a & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a & -a & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a & -a & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a & -a \\
b & b & -b & -b & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & b & b & -b & -b & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b & b & -b & -b & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b & b & -b & -b \\
c & c & c & c & -c & -c & -c & -c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c & c & c & c & -c & -c & -c & -c \\
d & d & d & d & d & d & d & d & -d & -d & -d & -d & -d & -d & -d & -d \\
d & d & d & d & d & d & d & d & d & d & d & d & d & d & d & d
\end{pmatrix}$$

where $a = 1/\sqrt{2}$, $b = 1/2$, $c = 1/\sqrt{8}$, $d = 1/4$. What is intended to be illustrated here is that the basis vector is the same at all levels, just translated within that level and dilated (by a factor of two) as the level is deepened. So if an artifact has a good projection onto (match with) some basis, the same artifact dilated by a factor of two would appear at a deeper level. This is why this is said to be a decomposition according to *scale*.

4 Orthogonal Filter Banks

To avoid too much subscripting, and to be in commonality with the literature, we'll switch from $H_0(z)$ & $H_1(z)$ to $C(z)$ & $D(z)$, as shown below.



It's worth expressing the output of the top two branches as a matrix-vector multiplication, shown in (43) for $L = 4$:

$$\begin{pmatrix} \vdots \\ u[n] \\ v[n] \\ u[n-1] \\ v[n-1] \\ \vdots \end{pmatrix} = \begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_0 & c_1 & c_2 & c_3 & 0 & 0 & \ldots \\ d_0 & d_1 & d_2 & d_3 & 0 & 0 & \ldots \\ 0 & 0 & c_0 & c_1 & c_2 & c_3 & \ldots \\ 0 & 0 & d_0 & d_1 & d_2 & d_3 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ x[n] \\ x[n-1] \\ x[n-2] \\ x[n-3] \\ x[n-4] \\ x[n-5] \\ \vdots \end{pmatrix} \tag{43}$$

For orthogonality we require

$$\sum_n c_n c_{n-2k} = \delta[k] \tag{44}$$

$$\sum_n c_n d_{n-2k} = 0 \tag{45}$$

$$\sum_n d_n d_{n-2k} = \delta[k] \tag{46}$$

As usual we have rather too much freedom to select the filters. For now assume that $\{c[n]\}$ is already picked. The Smith-Barnwell/Mintzer choice for $\{d[n]\}$ is

$$D(z) = -z^{-(L-1)} C(-z^{-1}) \tag{47}$$

$$= -z^{-(L-1)}(c_0 - c_1 z + c_2 z^2 - \ldots + (-1)^{(L-1)} c_{L-1} z^{L-1}) \tag{48}$$

$$= (-1)^L c_{L-1} + \ldots - c_2 z^{-(L-3)} + c_1 z^{-(L-2)} - c_0 z^{-(L-1)} \tag{49}$$

The Smith-Barnwell/Mintzer choice is not the only one, but it is fairly nice for the following reasons.

Smith Barnwell/Mintzer is Nice: Perfect Reconstruction

According to (33) we define, with $l = L - 1$,

$$
\begin{aligned}
P(z) &= z^{(L-1)} H_0(z) H_1(-z) && (50)\\
&= z^{(L-1)} C(z) D(-z) && (51)\\
&= z^{(L-1)} C(z)(-(-z)^{-(L-1)} C(z^{-1})) && (52)\\
&= C(z) C(z^{-1}) && (53)
\end{aligned}
$$

since $l = L - 1$ has to be odd. Now, notice that this refers to

$$p[n] \;=\; c[n] \star c[-n] \tag{54}$$

Looking at (44) and realizing that this is a constraint on the down-sampled $\{p[n]\}$, we have

$$P(z) + P(-z) \;=\; 2 \tag{55}$$

or

$$C(z)C(z^{-1}) + C(-z)C(-z^{-1}) \;=\; 2 \tag{56}$$

What this means is that (44) is the same as the *half-band condition* from (34). If we select $C(z)$ to satisfy (55) then we have both perfect reconstruction and one out of three conditions for orthogonality.

Smith Barnwell/Mintzer is Nice: Self-Orthogonality

We just found out that if (44) with the Smith-Barnwell-Mintzer condition (44) then we have perfect reconstruction (invertibility). We also have the same property for $\{d[n]\}$:

$$
\begin{aligned}
&D(z)D(z^{-1}) + D(-z)D(-z^{-1})\\
&\quad= (-z^{-(L-1)} C(-z^{-1}))(-z^{(L-1)} C(-z))\\
&\qquad +(z^{-(L-1)} C(z^{-1}))(z^{-(L-1)} C(z)) && (57)\\
&\quad= Q(z) + Q(-z) && (58)
\end{aligned}
$$

That is, if $C(z)$ is chosen to satisfy (56) then both (44) and (46) are satisfied.

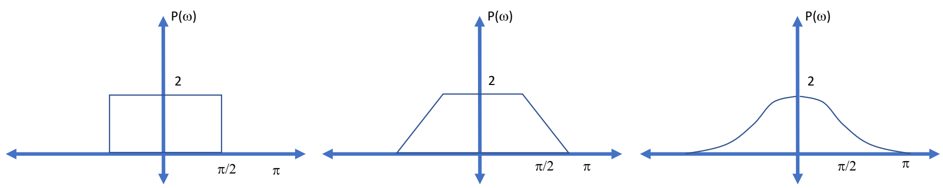
Smith Barnwell/Mintzer is Nice: Cross-Orthogonality

Just as good, we have

$$
\begin{aligned}
&C(z)D(z^{-1}) + C(-z)D(-z^{-1})\\
&\quad= C(z)(-z^{(L-1)} C(-z)) + C(-z)(z^{-(L-1)} C(z)) && (59)\\
&\quad= 0 && (60)
\end{aligned}
$$

so (45) is satisfied as well. That is, we have orthogonality!

This "half-band" condition – introduced as a sufficient condition for perfect reconstruction (invertibility) in (34) and rediscovered as a by-product of the Smith-Barnwell/Mintzer choice in (55) that also gives orthogonality – is also known as the first Nyquist condition in digital communications. It is perhaps worth mentioning that any filter satisfying the half-band condition gives rise to a structure commonly known as a *quadrature mirror filter* (QMF) bank. Below we see three possible configurations for a viable $P(z)$.



On the left is the rather obvious brick-wall filter. This is fine, but even to approximate it requires a very large L: no good. The middle is better, and it becomes clear how aliasing is not the deal-killer we thought it might be. On the right is the "raised-cosine" filter that uniquely satisfies both first and second Nyquist criteria. Here we have

$$\begin{aligned} P(\omega) &= 1+\cos(\omega) && (61)\\ &= \frac{1}{2}e^{j\omega} \,+\, 1 \,+\, \frac{1}{2}e^{-j\omega} && (62)\\ P(z) &= \frac{1}{2}z \,+\, 1 \,+\, \frac{1}{2}z^{-1} && (63) \end{aligned}$$

and since $P(z) = C(z)C(z^{-1})$ this means

$$C(z) = \frac{1}{\sqrt{2}}(1+z^{-1}) \tag{64}$$

That is, the raised-cosine filter and the Haar filter are the same thing.

As a side note, it is interesting to ask whether filters can be orthogonal *and* linear-phase. A linear-phase filter structure

$$\{c_0,\, c_1,\, c_2,\, ,\ldots,\, \pm c_2,\, \pm c_1,\, \pm c_0\,\} \tag{65}$$

meaning that the impulse response is either even or odd symmetric. Clearly the Haar filter works, it is even symmetric and hence linear-phase. For a filter of length ($L=4$) we interrogate (44) for $k=1$ and find it implies

$$\pm 2c_0c_1 \;=\; 0 \tag{66}$$

which means that any such filter has only two identical non-zero coefficients, and since $L = 4$ it means $c_1 = 0$. Similar analysis for $L = 6,\ 8, \ldots$ finds the same conclusion: c_0 is the only non-zero coefficient. While this is slightly different from the Haar filter it possesses no new richness, so we do not pursue it: aside from $L = 2$ (Haar) linear-phase is out of the question if orthogonality is desired.

5 Daubechies Filters

5.1 The Max-Flat Idea

The half-band conditions (resulting from the choice (28)) and even the Smith-Barnwell/Mintzer choice are decent but non-unique ways to get perfect reconstruction and orthogonality, respectively. But even the latter does not specify $C(z)$, only the half-band condition that $C(\omega)$ must satisfy. Daubechies came up with a set of conditions that are often thought to give the "best" QMF. Her idea is to look for a filter that is both short (small L) and decently frequency-selective.

5.2 The Really Technical Development

The development is rather indirect. Here goes. Consider the function

$$(1-y)^{-p} = \sum_{k=0}^{\infty} \binom{p+k-1}{k} y^k \tag{67}$$

We will truncate this to p terms

$$\begin{aligned} B(y) &= \sum_{k=0}^{p-1} \binom{p+k-1}{k} y^k & (68)\\ &= 1 + py + \binom{p+1}{2} y^2 + \ldots + \binom{2p-1}{p-1} y^{p-1} & (69) \end{aligned}$$

Now

$$\begin{aligned} \tilde{P}(y) &\equiv 2(1-y)^p B(y) & (70)\\ &= 2(1-y)^p((1-y)^{-p} + \mathcal{O}(y^p)) & (71)\\ &= 2 + \mathcal{O}(y^p) & (72) \end{aligned}$$

Now, notice from (72) we have

$$\tilde{P}'(y)|_{y=0} = \tilde{P}''(y)|_{y=0} = \ldots = \tilde{P}^{(p-1)}(y)|_{y=0} = 0 \tag{73}$$

and likewise we have

$$\tilde{P}'(y)|_{y=1} = \tilde{P}''(y)|_{y=1} = \ldots = \tilde{P}^{(p-1)}(y)|_{y=1} = 0 \tag{74}$$

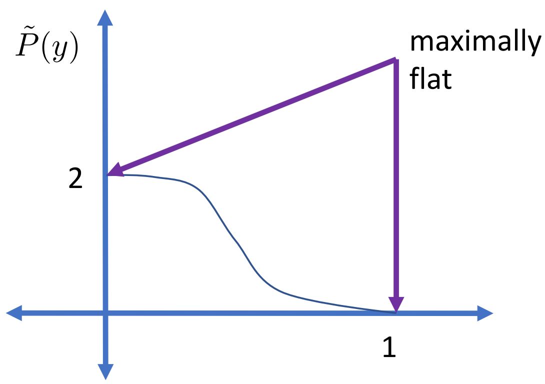
from (70). Similarly, from (72) we have

$$\tilde{P}(0) = 2 \tag{75}$$

and

$$\tilde{P}(1) = 0 \tag{76}$$

from (70). These are the *maximum-flatness* conditions: the function is flat and very smoothly so at both $y = 0$ and $y = 1$, and decreases from "passband" to "stopband" between. A notion is plotted below.



From (69) and (70) we have that $\tilde{P}(y)$ is a polynomial in y of degree $2p-1$. As such, $\tilde{P}'(y)$ is a polynomial in y of degree $2p-2$. And (73) and (74) tell us what it must be:

$$\tilde{P}'(y) \;=\; Cy^{p-1}(1-y)^{p-1} \tag{77}$$

Since $\tilde{P}'(y) = 0$ and hence $\tilde{P}'(1-y) = 0$ as well, and since we know

$$\begin{aligned}(\tilde{P}(y) + \tilde{P}(1-y))|_{y=0} &= \tilde{P}(y)|_{y=0} + \tilde{P}(y)|_{y=1} && (78)\\ &= 2 && (79)\end{aligned}$$

we can say

$$\tilde{P}(y) + \tilde{P}(1-y) \;=\; 2 \tag{80}$$

which is looking very close to our half-band condition, except in y as opposed to z.

Now substitute

$$y \longleftarrow \left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right) \tag{81}$$

Note

$$\begin{aligned} 1-y &= \frac{1}{4}\left(4-(-z+2-z^{-1})\right) && (82)\\ &= \frac{1}{4}\left(2+z+z^{-1}\right) && (83)\\ &= \left(\frac{1+z}{2}\right)\left(\frac{1+z^{-1}}{2}\right) && (84) \end{aligned}$$

So we substitute

$$P(z) = \tilde{P}(y)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} \tag{85}$$

Now we have

$$\begin{aligned} &P(z)+P(-z)\\ &\quad= \tilde{P}(y)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} + \tilde{P}(y)\big|_{y=\left(\frac{1+z}{2}\right)\left(\frac{1+z^{-1}}{2}\right)} && (86)\\ &\quad= \tilde{P}(y)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} + \tilde{P}(1-y)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} && (87)\\ &\quad= \left(\tilde{P}(y)+\tilde{P}(1-y)\right)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} && (88)\\ &\quad= 2 && (89) \end{aligned}$$

so the half-band condition is indeed satisfied by the Daubechies filters!

5.3 How to Make a Daubechies Filter

All we need to do now is to find find one. We need to write

$$\tilde{P}(y) = 2(1-y)^p\left(\sum_{k=0}^{p-1}\binom{p+k-1}{k}y^k\right) \tag{90}$$

from (70) and (68). Then we write

$$\begin{aligned} P(z) &= \tilde{P}(y)\big|_{y=\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)} && (91)\\ &= 2\left(\frac{1+z}{2}\right)^p\left(\frac{1+z^{-1}}{2}\right)^p\\ &\qquad\times\sum_{k=0}^{p-1}\binom{p+k-1}{k}\left(\frac{1-z}{2}\right)^k\left(\frac{1-z^{-1}}{2}\right)^k && (92) \end{aligned}$$

from (90), (85) and (84). Finally we must use

$$P(z) \;=\; C(z)C(z^{-1}) \tag{93}$$

to extract $C(z)$ from $P(z)$.

So let's try $p = 1$. From (92) we easily get

$$P(z) \;=\; \frac{1}{2}\left((1+z)(1+z^{-1})\right) \tag{94}$$

and it is easy to apply (94) to (93) to get

$$C(z) \;=\; \frac{1+z^{-1}}{\sqrt{2}} \tag{95}$$

which is the Haar filter!

To show something a little more interesting, let us try $p = 2$. We get

$$\begin{aligned} P(z) &= \frac{1}{8}(1+z)^2(1+z^{-1})^2\left(1+2\left(\frac{1-z}{2}\right)\left(\frac{1-z^{-1}}{2}\right)\right) && (96)\\ &= \frac{-1}{16}(1+z)^2(1+z^{-1})^2\left(z-4+z^{-1}\right) && (97)\\ &= \frac{-1}{16(2-\sqrt{3})}(1+z)^2(1+z^{-1})^2 \\ &\qquad \times\left((1-(2-\sqrt{3})z)(1-(2-\sqrt{3})z^{-1})\right) && (98)\end{aligned}$$

meaning that we have

$$C(z) \;=\; \frac{1}{\sqrt{32}}\left((1+\sqrt{3}) \;+\; (3+\sqrt{3})z^{-1} \;+\; (3-\sqrt{3})z^{-2} \;+\; (1-\sqrt{3})z^{-3}\right) \tag{99}$$

This is the D4 filter.

6 Wavelets

6.1 The Telescoping Subspaces

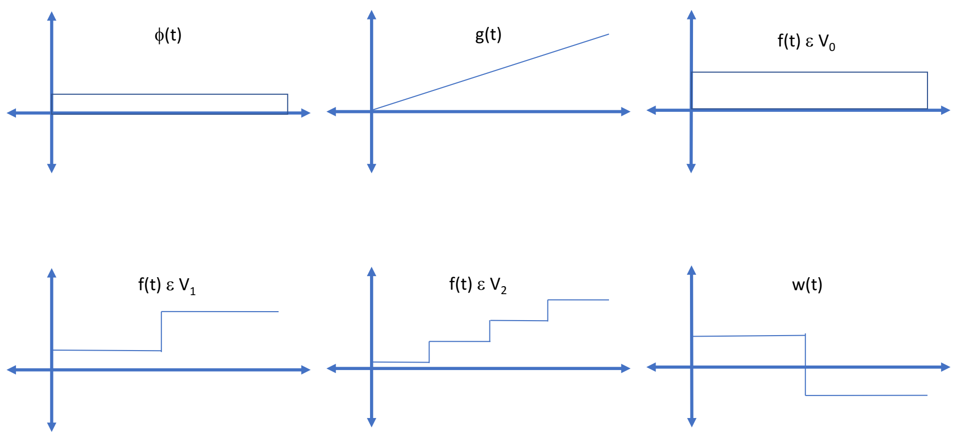
Wavelets are the continuous-time (or -space) version of multi-resolution decomposition. Begin with *telescoping* subspaces

$$\mathcal{V}_0 \subset \mathcal{V}_1 \subset \mathcal{V}_2 \subset \mathcal{V}_3 \subset \mathcal{V}_4 \subset \ldots \tag{100}$$

and require that if $f(t) \in V_j$ then

1. $f(t-k) \in \mathcal{V}_j, \quad \forall k \in \mathcal{I}$; and
2. $f(2t-k) \in \mathcal{V}_{j+1}, \quad \forall k \in \mathcal{I}$.

Also assume that there is *scaling function* $\phi(t)$ such that for $\{\phi(t-k)\}_{k\in\mathcal{I}}$ is a basis for $\mathcal{V}_0$.



An example is given above. We seek to approximate the ramp-function $g(t)$ (top middle) in a telescoping series of spaces that are formed by the scaling function $\phi(t)$, top left. The approximations in $\mathcal{V}_0$, $\mathcal{V}_1$ and $\mathcal{V}_2$ are shown in top right, bottom left and bottom middle, respectively. It is clear that the deeper one gets the better the approximation. We also define the *wavelet space* $\mathcal{W}_0$ with basis $w(t)$ (bottom right), such that

$$\mathcal{V}_j \bigcup \mathcal{W}_j = \mathcal{V}_{j+1} \tag{101}$$

and

$$\mathcal{V}_j \bigcap \mathcal{W}_j = \emptyset \tag{102}$$

The function $w(t)$ is called the *mother wavelet*.

6.2 Relationship to Multi-Resolution Decomposition

Now, $\mathcal{V}_0 \subset \mathcal{V}_1$ means that

$$\phi(t) = \sum_n c_n \phi(2t-n) \tag{103}$$

If we also have $\{\phi(t-k)\}$ orthogonal – and hence $\{\phi(2t-k)\}$ orthogonal – we can write

$$c_n = \int \phi(t)\phi(2t-n)dt \tag{104}$$

Expressing the orthogonality requirement using this, we have

$$\begin{aligned}
\delta[k] &= \int \phi(t)\phi(t-k)dt && (105)\\
&= \int \left(\sum_m c_m \phi(2t-m)\right)\left(\sum_n c_n \phi(2(t-k)-n)\right) dt && (106)\\
&= \sum_n c_n c_{n-2k} && (107)
\end{aligned}$$

It is very interesting that (107) is identical to (44) – that is, the condition for a telescoping basis based on orthogonal functions is the same as the condition for a multi-resolution decomposition filter to be orthogonal. Let us go a little further, and note that since $\mathcal{W}_j \subset \mathcal{V}_{j+1}$ we can write

$$w(t) = \sum_k d_k \phi(2t-k) \qquad (108)$$

where

$$d_k = \int w(t)\phi(2t-k)dt \qquad (109)$$

If we desire orthogonality of $\{w(t-m)\}$ we have

$$\begin{aligned}
\delta[k] &= \int \phi(t)\phi(t-k)dt && (110)\\
&= \int \left(\sum_m d_k \phi(2t-m)\right)\left(\sum_n d_n \phi(2(t-k)-n)\right) dt && (111)\\
&= \sum_n d_n d_{n-2k} && (112)
\end{aligned}$$

and similarly, if orthogonality of $\mathcal{W}_j$ to $\mathcal{V}_j$ is desired we have

$$\begin{aligned}
0 &= \int \phi(t)w(t-k)dt && (113)\\
&= \int \left(\sum_m c_k \phi(2t-m)\right)\left(\sum_n d_n \phi(2(t-k)-n)\right) dt && (114)\\
&= \sum_n c_n d_{n-2k} && (115)
\end{aligned}$$

That is, (107), (112) & (115) – demanded for orthogonality of the telescoping representation – are identical to (44), (46) & (45) for orthogonality of a multi-resolution decomposition.

6.3 How to Make the Mother Wavelet and Scaling Function

So what are $\phi(t)$ and $w(t)$? The relation

$$\phi(t) \;=\; \sum_k c_k \phi(2t-k) \tag{116}$$

provides the answer. Take the (continuous-time) Fourier transform

$$\begin{aligned}
\Phi(\Omega) \;&=\; \int_{-\infty}^{\infty} \phi(t) e^{-j\Omega t} dt && (117)\\
&=\; \int_{-\infty}^{\infty} \left(\sum_k c_k \phi(2t-k) \right) e^{-j\Omega t} dt && (118)\\
&=\; \sum_k c_k e^{-j\left(\frac{\Omega}{2}\right)k} \int_{-\infty}^{\infty} \phi(2t-k) e^{-j\left(\frac{\Omega}{2}\right)(2t-k)} dt && (119)\\
&=\; \frac{1}{2} C\left(\frac{\Omega}{2}\right) \Phi\left(\frac{\Omega}{2}\right) && (120)
\end{aligned}$$

where

$$C(\omega) \;\equiv\; \sum_k c_k e^{-j\omega k} \tag{121}$$

which of course repeats with period 2π. We are not interested in the factor of $\frac{1}{2}$ in (120) since we normalize to have unit energy; so let us drop it. We also have for the mother wavelet

$$W(\Omega) \;=\; \frac{1}{2} D\left(\frac{\Omega}{2}\right) \Phi\left(\frac{\Omega}{2}\right) \tag{122}$$

Note that as $k \to \infty$ we have $\frac{\Omega}{2^k} \to 0$. We arbitrarily set $\Phi(0) = 1$ – any non-zero constant will do – so we have

$$\begin{aligned}
\Phi(\omega) \;&=\; \prod_{k=1}^{\infty} C\left(\frac{\omega}{2^k}\right) && (123)\\
W(\omega) \;&=\; D\left(\frac{\omega}{2}\right) \prod_{k=2}^{\infty} C\left(\frac{\omega}{2^k}\right) && (124)
\end{aligned}$$

which explicitly define the (Fourier transforms of the) scaling function and mother wavelet in terms of chosen multi-resolution filter function.

6.4 Compactness of the Scaling Function and Mother Wavelet

First, let us observe that (123) and (124) require that $C(\omega) = 0$ else $\Phi(\omega)$ goes on forever. Let us also define $c(t)$ via

$$
\begin{aligned}
C(\omega) &= \mathcal{F}\left[c(t)\right] && (125)\\
&= \mathcal{F}\left[\sum_k c_k \delta(t-k)\right] && (126)\\
&= \int_{-\infty}^{\infty} \sum_k c_k \delta(t-k) e^{-j\omega t} dt && (127)\\
&= \sum_k c_k \delta(t-k) e^{-j\omega k} && (128)
\end{aligned}
$$

This is not especially useful except to tell is that $c(t)$ is time-limited if $\{c_n\}$ is FIR – $c(t)$ has support only on $[0, L)$ (actually $[0, L-1)$). Then (123) implies

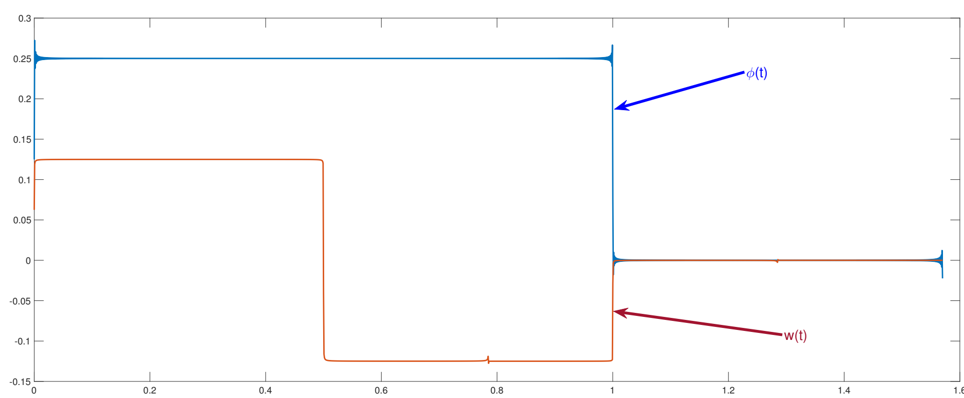
$$\phi(t) \;=\; c(2t) \star c(4t) \star c(8t) \star c(16t) \star c(32t) \star \ldots \qquad (129)$$

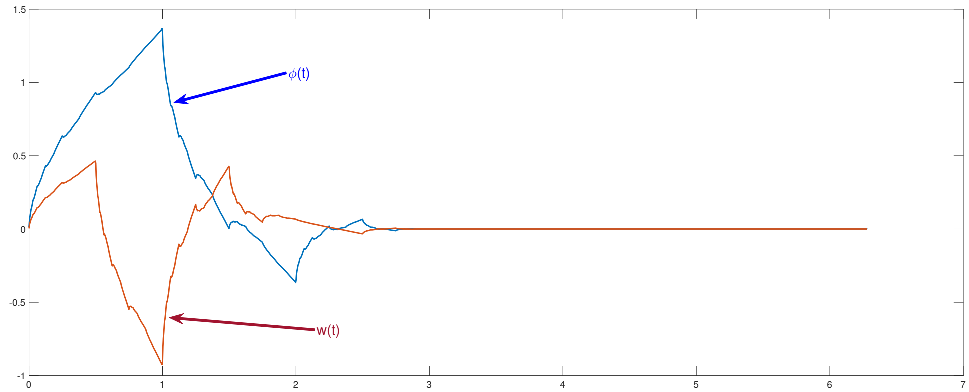
meaning that the support of $\phi(t)$ can be no greater than of length

$$\frac{L}{2} + \frac{L}{4} + \frac{L}{8} + \frac{L}{16} + \ldots = L \qquad (130)$$

That is, the scaling function $\phi(t)$ is supported only on $[0, L)$ – it is *compact*! The same can be said for the mother wavelet $w(t)$.

Examples of scaling functions and wavelets for the Daubechies-2 (i.e., Haar) and Daubechies-4 systems are given below





What is striking is that the Haar functions are exactly what one might think, and basically the same as in the earlier notional cartoon. The Daubechies-4 scaling function and mother wavelet are weird. But they are what they are.