Complex-valued Wavelets, the Dual Tree, and Hilbert Pairs: why these lead to shift invariance and directional m-D wavelets?

Nick Kingsbury

Signal Processing and Communications Laboratory
Department of Engineering, University of Cambridge, UK.

email: ngk@eng.cam.ac.uk web: www.eng.cam.ac.uk/~ngk

Part 3, UDRC Short Course, 21 March 2013

UNIVERSITY OF CAMBRIDGE

Important continuous-time complex-valued wavelets

The **Gabor** function

$$\psi(t) = k\ e^{-t^2/2\sigma^2}\ e^{j\omega_0 t} \qquad \hat{\psi}(\omega) = K\ e^{-\frac{1}{2}\sigma^2(\omega-\omega_0)^2}$$

The **Morlet** wavelet

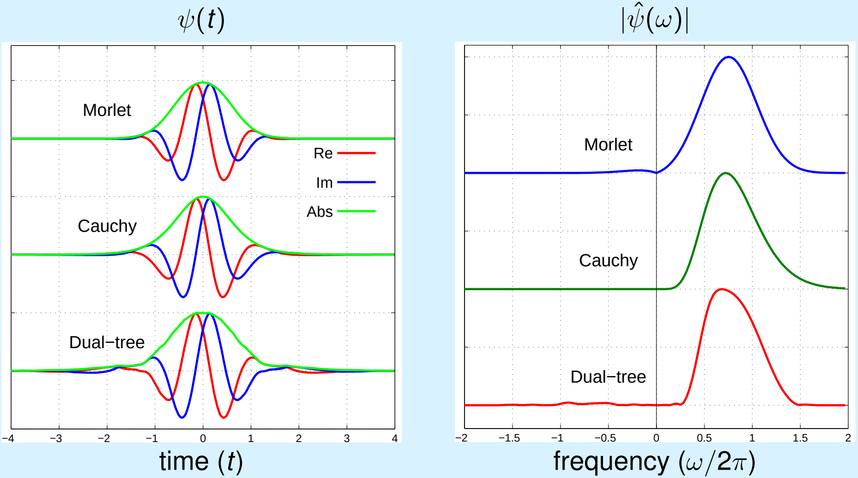
$$\psi(t) = k\ e^{-t^2/2\sigma^2}(e^{j\omega_0 t} - \kappa_0) \qquad \hat{\psi}(\omega) = K\left(e^{-\frac{1}{2}\sigma^2(\omega-\omega_0)^2} - \kappa_0\ e^{-\frac{1}{2}\sigma^2\omega^2}\right)$$

The **Cauchy** wavelet

$$\psi(t) = k\,(1 - j\beta t)^{-\alpha} \qquad \hat{\psi}(\omega) = \begin{cases} K\,\omega^{\alpha-1}e^{-\omega/\beta}, & \omega \geq 0 \\ 0\,, & \omega < 0 \end{cases}$$

Max gain is at $\omega_0 = \beta(\alpha - 1)$.
Typically $\alpha \approx 8$ for one octave half-power bandwidth.

Plots of $\psi(t)$ and $|\hat{\psi}(\omega)|$ for Morlet, Cauchy (continuous) and dual-tree (discrete) complex wavelets



Note: $k = (1 + j)/\sqrt{2}$; and the dual-tree filters are 18-tap Q-shift filters.

Complex-valued Wavelets, the Dual Tree, and Hilbert Pairs

How can we produce *good* discrete wavelet transforms?

- What are the problems with real-valued wavelet bases?
- Why do we need the Dual Tree?
- What is the Hilbert Pair delay condition?
- Why does this give shift invariance?
- Why do we use Q-shift filters?
- How do we extend the dual-tree to multi-dimensions?
- Why do we get good directional filters in m-D?
- What are some applications of the DT $\mathbb{C}$WT?

Real Discrete Wavelet Transform (DWT) in 1-D

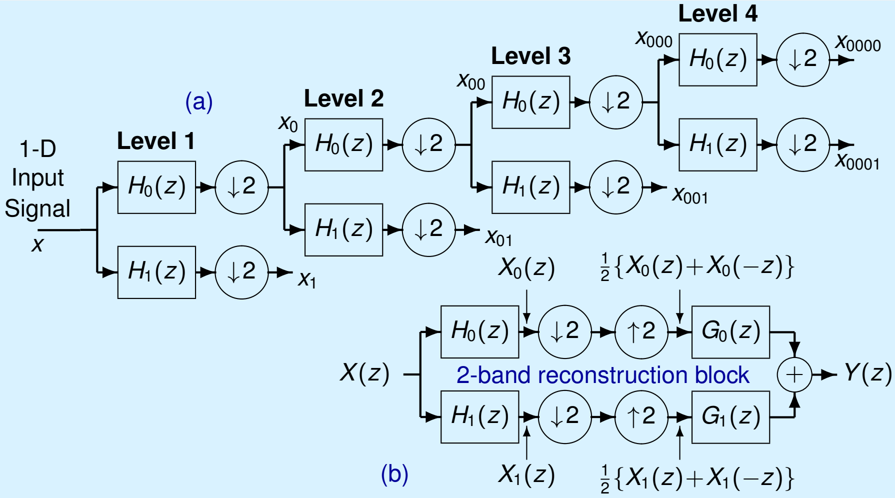


Figure: (a) Tree of real filters for the DWT. (b) Reconstruction filter block for 2 bands at a time, used in the inverse transform.

Features of the (Real) Discrete Wavelet Transform (DWT)

- **Good compression** of signal energy.
- **Perfect reconstruction** with short support filters.
- **No redundancy** – hence orthonormal or bi-orthogonal transforms are possible.
- **Very low computation** – order-N only.

But

- **Severe shift dependence**.
- **Poor directional selectivity** in 2-D, 3-D etc.

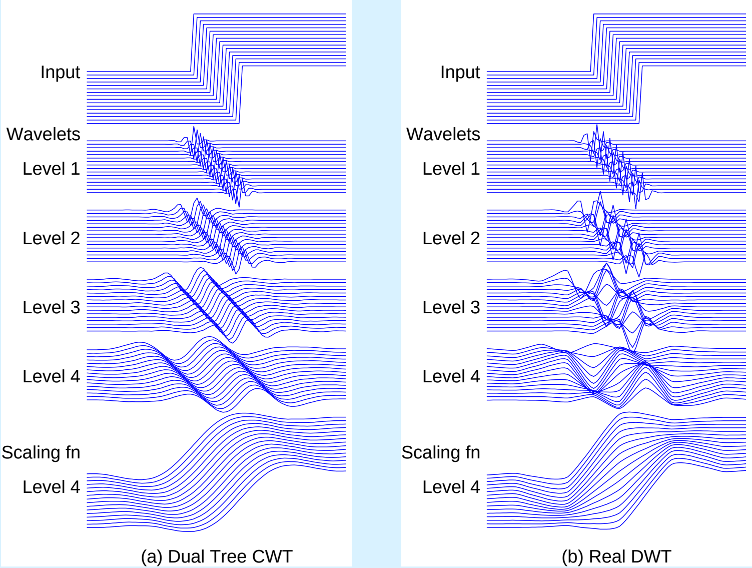
The DWT is normally implemented with a tree of highpass and lowpass filters, separated by $2:1$ decimators.

Visualising Shift Invariance / Dependence

- Apply a standard input (e.g. unit step) to the transform for a **range of shift positions**.
- Select the transform coefficients from **just one wavelet level** at a time.
- Inverse transform each set of selected coefficients.
- Plot the component of the reconstructed output for each shift position at each wavelet level.
- Check for **shift invariance** (similarity of waveforms).

See Matlab demonstration / next slide.

Shift Invariance of DT ℂWT / Dependence of DWT



Q-shift Dual Tree Complex Wavelet Transform (DT ℂWT) in 1-D

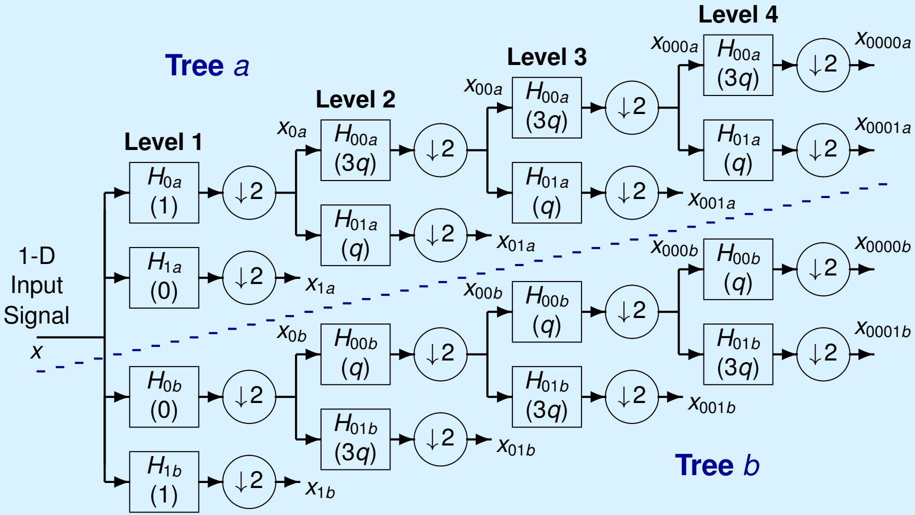


Figure: Dual tree of real filters for the Q-shift ℂWT, giving real and imaginary parts of complex coefficients from tree *a* and tree *b* respectively.

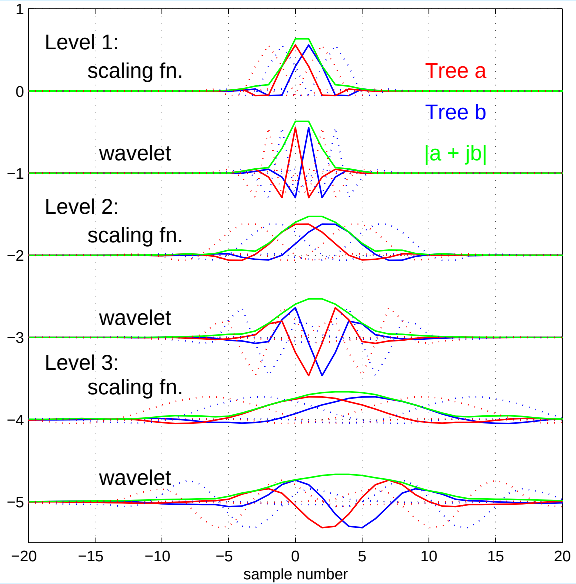
Features of the Dual Tree Complex Wavelet Transform

- Good **shift invariance** = **negligible aliasing**. Hence the transfer function through each subband is independent of shift **and** wavelet coefs can be interpolated within each subband, independent of all other subbands.
- Good **directional selectivity** in 2-D, 3-D etc. – derives from **analyticity** in 1-D (ability to separate positive from negative frequencies).
- **Perfect reconstruction** with short support filters.
- **Limited redundancy** – 2:1 in 1-D, 4:1 in 2-D etc.
- **Low computation** – much less than the undecimated (à trous) DWT.

Each tree contains purely real filters, but the two trees produce the **real and imaginary parts** respectively of each complex wavelet coefficient.

Q-shift DT ℂWT Basis Functions – Levels 1 to 3

Basis functions for adjacent sampling points are shown dotted.



Why do we need the Dual Tree?

Making the wavelet responses **analytic** is a good way to **halve their bandwidth** and hence minimise aliasing.
BUT we cannot use complex filters in the DWT to obtain analyticity and perfect reconstruction together, because requirements conflict in the 2-band reconstruction block – analytic filters must suppress negative frequencies, while PR requires a flat overall frequency response.

So we use the **Dual Tree**:

- to create the **real** and **imaginary** parts of the analytic wavelets separately, using 2 trees of **purely real** filters;
- to efficiently synthesise a multiscale **shift-invariant** filterbank, with perfect reconstruction and **only 2:1 redundancy**;
- to produce complex coefficients whose **amplitude varies slowly** and whose **phase shift** depends approximately **linearly** on displacement.

What is the Hilbert Pair Delay Condition ?

- *Given two parallel orthonormal discrete wavelet transforms, what is the constraint on the lowpass filters in each transform, such that the resulting continuous wavelets from each transform form a Hilbert Pair?*
  (This question and its answer are due to Ivan Selesnick in IEEE Signal Proc. Letters, June 2001.)
- A pair of wavelets, $\psi_g(t)$ and $\psi_h(t)$, are a **Hilbert Pair** if the complex function $\psi_g(t) + j\,\psi_h(t)$ is **analytic** (i.e. its Fourier transform is zero for $\omega < 0$).
- We shall show that this requires the lowpass filters, $g_0(n)$ and $h_0(n)$, of the two transforms to be related by the **half-sample delay condition**, expressed in the frequency domain as

$$H_0(\omega) = e^{-j\omega/2} G_0(\omega)$$

2-scale condition on Tree *a* filters of a dyadic wavelet transform

$$\textbf{Scaling function:} \quad \phi_g(t) = 2 \sum_n g_0(n)\, \phi_g(2t-n) \qquad (1)$$

$$\textbf{Mother wavelet:} \quad \psi_g(t) = 2 \sum_n g_1(n)\, \phi_g(2t-n) \qquad (2)$$

Taking the Fourier transform of (1) gives the frequency domain relationship

$$\begin{aligned}
\hat{\phi}_g(\omega) &= \int_{-\infty}^{\infty} 2 \sum_n g_0(n)\, \phi_g(2t-n)\, e^{-j\omega t}\, dt \\
&= \int_{-\infty}^{\infty} \sum_n g_0(n)\, \phi_g(u)\, e^{-j\omega u/2}\, e^{-j\omega n/2}\, du \quad \text{where } u = 2t-n \\
&= \sum_n g_0(n)\, e^{-jn\omega/2} \; . \int_{-\infty}^{\infty} \phi_g(u)\, e^{-ju\omega/2}\, du \\
&= G_0(\tfrac{\omega}{2}) \, . \, \hat{\phi}_g(\tfrac{\omega}{2}) \qquad (3)
\end{aligned}$$

Infinite Product formulae

Iterating on (3):

$$\hat{\phi}_g(\omega) = G_0(\tfrac{\omega}{2})\, G_0(\tfrac{\omega}{4})\, \hat{\phi}_g(\tfrac{\omega}{4}) = \cdots = \left[\prod_{k=1}^{\infty} G_0(2^{-k}\omega)\right] \hat{\phi}_g(0) \qquad (4)$$

Similarly, from (2) and (4):

$$\hat{\psi}_g(\omega) = G_1(\tfrac{\omega}{2})\, \hat{\phi}_g(\tfrac{\omega}{2}) = G_1(\tfrac{\omega}{2}) \left[\prod_{k=2}^{\infty} G_0(2^{-k}\omega)\right] \hat{\phi}_g(0) \qquad (5)$$

And similarly, for the Tree *b* filters:

$$\hat{\phi}_h(\omega) = \left[\prod_{k=1}^{\infty} H_0(2^{-k}\omega)\right] \hat{\phi}_h(0) \qquad (6)$$

$$\hat{\psi}_h(\omega) = H_1(\tfrac{\omega}{2}) \left[\prod_{k=2}^{\infty} H_0(2^{-k}\omega)\right] \hat{\phi}_h(0) \qquad (7)$$

The amplitude scaling of $\phi_g(t)$ and $\phi_h(t)$ is arbitrary, so we choose $\hat{\phi}_g(0) = \hat{\phi}_h(0) = 1$ to give them both unit area.

Conjugate Quadrature Filterbank (CQF)

In an **orthonormal** wavelet transform, **G_1 and G_0 form a CQF** (also known as a Quadrature Mirror Filterbank, QMF), such that

$$G_1(\omega) = e^{-jm\omega}\ G_0^*(\omega \pm \pi) \tag{8}$$

where we use $\pm\pi$ to emphasise the 2π-periodic nature of G_0 and G_1, and G_0^* means complex conjugate of G_0. The delay shift of m samples must be an odd integer and is usually chosen to approximately equalise the group delay or the support of G_0 and G_1.

Similarly

$$H_1(\omega) = e^{-jm\omega}\ H_0^*(\omega \pm \pi) \tag{9}$$

Hence we can now express the wavelet frequency responses, $\hat{\psi}_g(\omega)$ and $\hat{\psi}_h(\omega)$, purely in terms of the two lowpass filters G_0 and H_0.

The Hilbert Pair Condition

This condition is

$$\frac{\hat{\psi}_h(\omega)}{\hat{\psi}_g(\omega)} = \begin{cases} j & \text{if } \omega < 0 \\ -j & \text{if } \omega > 0 \end{cases} \tag{10}$$

Note that the behaviour of the RHS near $\omega = 0$ is immaterial, because, for wavelets to be admissible bandpass functions, $\hat{\psi}_g(0) = \hat{\psi}_h(0) = 0$.

Substituting (8) into (5) and (9) into (7), we get the following expression for this ratio

$$\begin{aligned} \frac{\hat{\psi}_h(\omega)}{\hat{\psi}_g(\omega)} &= \frac{e^{-jm\omega/2}\ H_0^*(\frac{\omega}{2} \pm \pi) \left[\prod_{k=2}^{\infty} H_0(2^{-k}\omega)\right] \hat{\phi}_h(0)}{e^{-jm\omega/2}\ G_0^*(\frac{\omega}{2} \pm \pi) \left[\prod_{k=2}^{\infty} G_0(2^{-k}\omega)\right] \hat{\phi}_g(0)} \\ &= R_0^*(\tfrac{\omega}{2} \pm \pi) \left[\prod_{k=2}^{\infty} R_0(2^{-k}\omega)\right] \end{aligned} \tag{11}$$

where $R_0(\omega) = H_0(\omega)/G_0(\omega)$ and is 2π-periodic. R_0 will give the desired relationship between H_0 and G_0 if (10) and (11) are both satisfied.

Solving for $R_0(\omega)$

Hence

$$R_0^*(\tfrac{\omega}{2} \pm \pi)\left[\prod_{k=2}^{\infty} R_0(2^{-k}\omega)\right] = \begin{cases} j & \text{if } \omega < 0 \\ -j & \text{if } \omega > 0 \end{cases} \tag{12}$$

Since the modulus of the RHS of (12) is unity everywhere, and the LHS contains an infinite product of terms in R_0, which will tend to grow or shrink if R_0 does not have unit magnitude, **we deduce that** $|R_0(\omega)| = 1$.

Now consider the phase $\theta(\omega)$ of R_0, by letting

$$R_0(\omega) = e^{j\theta(\omega)} \tag{13}$$

Equating the phases in (12), we require that

$$-\theta(\tfrac{\omega}{2} \pm \pi) + \sum_{k=2}^{\infty} \theta(2^{-k}\omega) = \begin{cases} \frac{\pi}{2} & \text{if } \omega < 0 \\ -\frac{\pi}{2} & \text{if } \omega > 0 \end{cases} \tag{14}$$

Deducing the form of $\theta(\omega)$

Because of the infinite sum in (14), we require $\theta(\omega) \to 0$ as $\omega \to 0$.
Hence $\theta(0) = 0$.

Since $g_0(n)$ and $h_0(n)$ are purely real and lowpass, $R_0(\omega)$ must be conjugate symmetric and so $\theta(\omega)$ must be a continuous odd function about $\omega = 0$.

It can be shown (by Fourier analysis on $\theta'(\omega)$) that **any non-linear terms in $\theta(\omega)$ would prevent (14) from being satisfied**, because in (14) the gradient of the first term must cancel out the gradient of the summation terms at all $\omega \neq 0$.

Therefore we let

$$\theta(\omega) = \alpha\omega \quad \text{for } -\pi < \omega < \pi \text{, where } \alpha \text{ is a constant.} \tag{15}$$

Hence

$$\theta(\tfrac{\omega}{2} \pm \pi) = \begin{cases} \alpha(\frac{\omega}{2} + \pi) & \text{if } -4\pi < \omega < 0 \\ \alpha(\frac{\omega}{2} - \pi) & \text{if } \quad 0 < \omega < 4\pi \end{cases} \tag{16}$$

Note that $\theta(\omega)$ must be 2π-periodic for $|\omega| \geq \pi$, and so it will have discontinuities at $\omega = \pm\pi$, if α is not an integer.
These become discontinuities at $\omega = 0$ in $\theta(\frac{\omega}{2} \pm \pi)$.

Typical plots of $\theta(\omega)$ and terms in equ.(14)

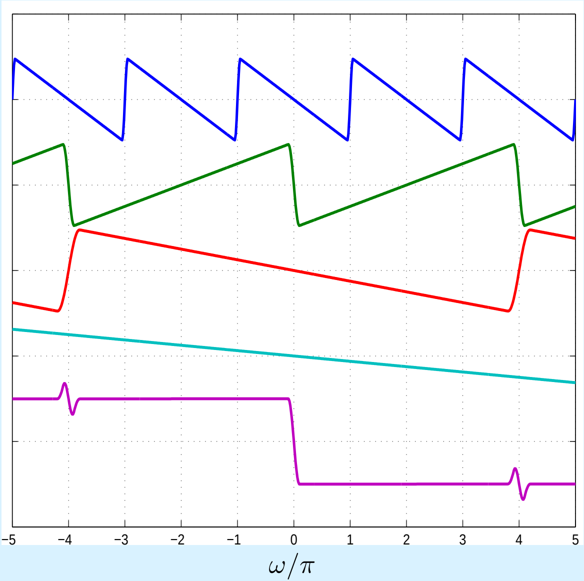
$\theta(\omega) \approx \alpha\omega$, $-\pi < \omega < \pi$
and $\theta(\omega)$ is 2π-periodic

$-\theta(\frac{\omega}{2} \pm \pi)$

$\theta(\frac{\omega}{4})$

$\theta(\frac{\omega}{8})$

$$-\theta(\tfrac{\omega}{2} \pm \pi) + \sum_{k=2}^{\infty} \theta(2^{-k}\omega)$$



Calculating α

Noting that $\sum_{k=2}^{\infty} \theta(2^{-k}\omega) = \alpha\omega[\frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \ldots] = \frac{\alpha\omega}{2}$ if $-4\pi < \omega < 4\pi$,

and substituting (16) into (14) gives

$$
\begin{aligned}
-\alpha(\tfrac{\omega}{2} + \pi) + \tfrac{\alpha\omega}{2} &= \tfrac{\pi}{2} \quad \text{if } -4\pi < \omega < 0 && (17) \\
\text{and} \quad -\alpha(\tfrac{\omega}{2} - \pi) + \tfrac{\alpha\omega}{2} &= -\tfrac{\pi}{2} \quad \text{if } 0 < \omega < 4\pi && (18)
\end{aligned}
$$

(17) and (18) are both satisfied if $\alpha = -\frac{1}{2}$, and therefore

$$
\frac{H_0(\omega)}{G_0(\omega)} = R_0(\omega) = e^{j\theta(\omega)} = e^{j\alpha\omega} = e^{-j\omega/2} \quad \text{for } -\pi < \omega < \pi \qquad (19)
$$

This is the **half-sample delay** solution that makes $\psi_h(t)$ the Hilbert transform of $\psi_g(t)$.

Ozkaramanli and Yu (Dec 2005 and June 2006) have shown this solution to be **unique** and applicable to **biorthogonal** as well as **orthonormal** wavelet transforms.

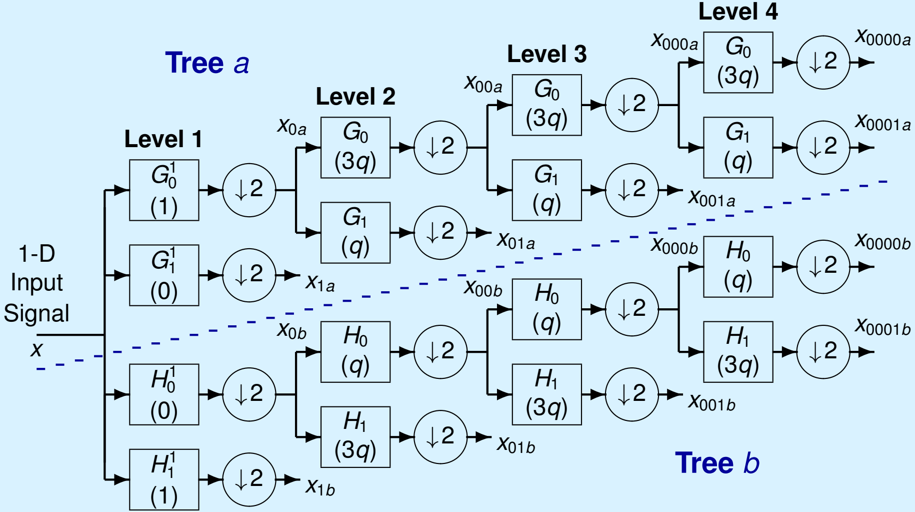


Fig. 1: Dual tree of real filters for the Q-shift $\mathbb{C}$WT, giving real and imaginary parts of complex coefficients from tree a and tree b respectively. Figures in brackets indicate the approximate delay for each filter, where $q = \frac{1}{4}$ sample period. Special level 1 filters, G^1 and H^1, allow for the finite number of levels.

Why does the delay condition give shift invariance?

- The **half-sample delay** between the G_0 and H_0 **lowpass** filters means that their output samples are **uniformly interleaved** at all scales, and hence the sample rate is effectively **doubled** everywhere.
- The doubled sampling rate is sufficient to virtually **eliminate aliasing** if filters of 12 or more taps are used.
- If aliasing is eliminated in the **lowpass** branch of each 2-band reconstruction block, then it must also be eliminated in the **highpass** branch (since perfect reconstruction is achieved).
- Elimination of aliasing means that each subband can be represented by a **unique z-transfer function**, and hence the filtering is **LTI**, linear time-invariant (i.e. shift-invariant).

At level 1 of a finite dual tree, the delay difference must **increase to one sample** to compensate for the absence of delay differences at finer levels.

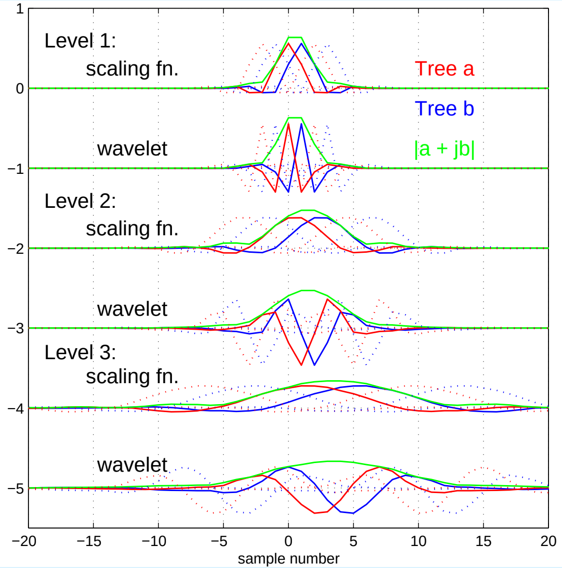
Why do we use Q-shift filters (below level 1)?

- Half-sample delay difference is obtained with filter delays of $\frac{1}{4}$ and $\frac{3}{4}$ of a sample period (instead of 0 and $\frac{1}{2}$ a sample for our original DT $\mathbb{C}$WT).
- This is achieved with an **asymmetric even-length** filter $G_0(z)$ and its time reverse $H_0(z) = z^{-1}\, G_0(z^{-1})$. $G_1(z)$ and $H_1(z)$ are the CQFs of these.
- Due to the asymmetry (like Daubechies filters), these may be designed to give an **orthonormal perfect reconstruction** wavelet transform in each tree.
- Tree ***b*** filters are the **reverse** of tree ***a*** filters, and reconstruction filters are the reverse of analysis filters, so **all filters** are from the **same orthonormal set**, yielding a **tight-frame** transform.
- Both trees have the **same frequency responses** (in magnitude).
- The combined **complex** impulse responses are **conjugate symmetric** about their mid points, even though the separate responses are asymmetric. Hence **symmetric extension** still works at image edges.

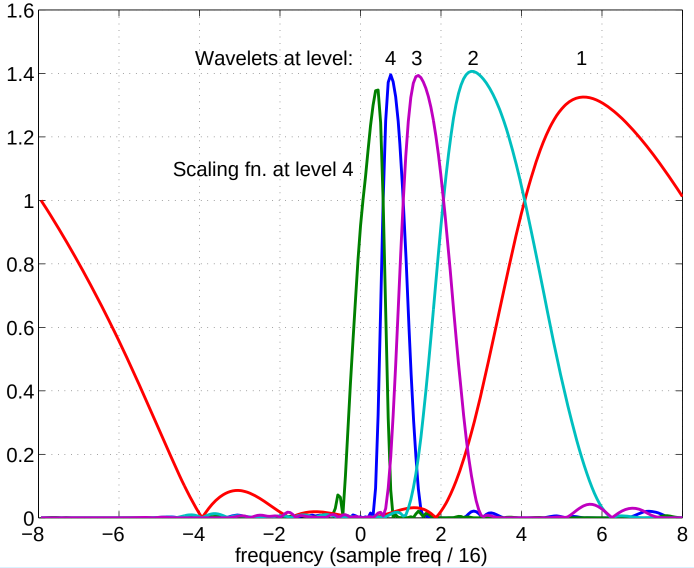
At level 1, **any DWT filters** can be used.

Q-shift DT ℂWT Basis Functions – Levels 1 to 3

Basis functions for adjacent sampling points are shown dotted.



Frequency Responses of 18-tap Q-shift filters

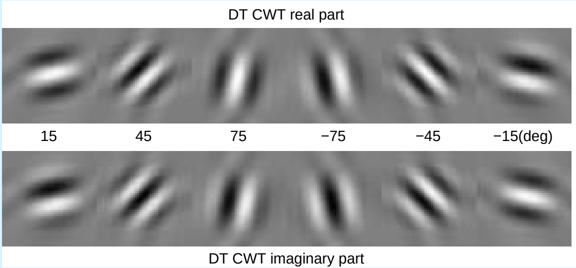


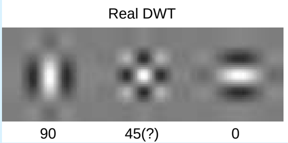
The DT $\mathbb{C}$WT in 2-D – why good directional selectivity?

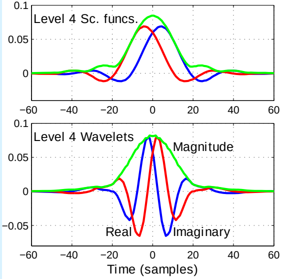
When the DT $\mathbb{C}$WT is applied to 2-D signals (images), it has the following features:

- It is performed separably, using 2 trees for the rows of the image and 2 trees for the columns – yielding a **Quad-Tree** structure (4:1 redundancy).
- The 4 quad-tree components of each coefficient are combined by simple sum and difference operations to yield a **pair of complex coefficients**. These are part of two separate subbands in adjacent quadrants of the 2-D spectrum.
- This produces **6 directionally selective subbands** at each level of the 2-D DT $\mathbb{C}$WT. The next slide shows basis functions of these subbands at level 4, and compares them with the 3 subbands of a 2-D DWT.
- The DT $\mathbb{C}$WT is directionally selective because the complex filters can **separate positive and negative frequency components** in 1-D, and hence **separate adjacent quadrants** of the 2-D spectrum. Real separable filters cannot do this!

2-D Basis Functions at level 4







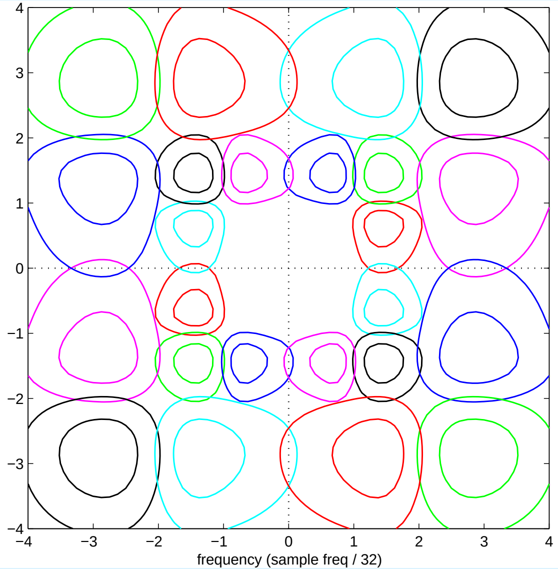
$$e^{j\omega_1 x}.\ e^{j\omega_2 y} = e^{j(\omega_1 x + \omega_2 y)}$$

only one wave!

Basis functions of 2-D Q-shift complex wavelets (top), and of 2-D real wavelet filters (bottom), all illustrated at level 4 of the transforms. The complex wavelets provide 6 directionally selective filters, while real wavelets provide 3 filters, only two of which have a dominant direction. The 1-D bases, from which the 2-D complex bases are derived, are shown to the right.

Frequency Responses of 2-D Q-shift filters at levels 3 and 4

Contours shown at
-1 dB and -3 dB.



Q-shift Filter Design Requirements

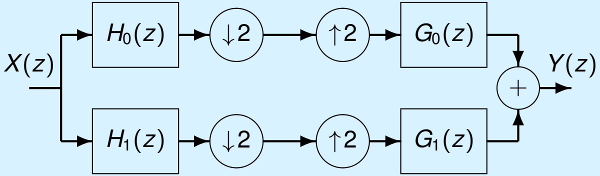


Fig 2: 2-band analysis and reconstruction filter bank.

- **No aliasing:** $G_1(z) = zH_0(-z)\ ; \quad H_1(z) = z^{-1}G_0(-z)$
- **Perfect reconstruction:** $H_0(z)G_0(z) + H_0(-z)G_0(-z) = 2$
- **Orthogonality:** $G_0(z) = z^{-k}\ H_0(z^{-1})$
- Group delay $\simeq \frac{1}{4}$ **sample period** for H_0.
- **Good smoothness** properties when iterated over scale.

Filter Design — Delay

To get $2n$-tap lowpass filters, $H_0(z)$ and $G_0(z)$, with $\frac{1}{4}$ and $\frac{3}{4}$ sample delays:

- Design a **4n-tap** symmetric lowpass filter $H_{L2}(z)$ with half the required bandwidth and a delay of $\frac{1}{2}$ sample;
- **Subsample** $H_{L2}(z)$ by 2:1 to get $H_0(z)$ and $G_0(z)$.

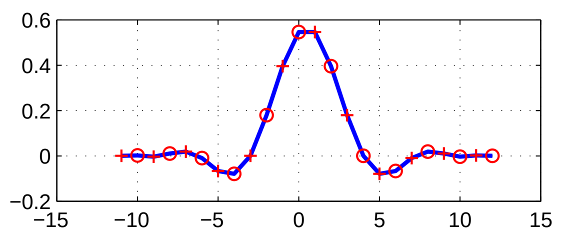


Fig 3: Impulse response of $H_{L2}(z)$ for $n = 6$. The H_0 and G_0 filter taps are shown as circles and crosses respectively.

Filter Design – Perfect Reconstruction (PR)

For PR and orthogonality, $H_0(z)\,G_0(z) = H_0(z)\,H_0(z^{-1})$ must have **no terms in z^{2k}** except the term in z^0.

Therefore $H_0(z^2)\,H_0(z^{-2})$ must have **no terms in z^{4k}** except the term in z^0.

But

$$H_{L2}(z) = H_0(z^2) + z^{-1} H_0(z^{-2})$$

and so

$$H_{L2}(z)\,H_{L2}(z^{-1}) = 2\,H_0(z^2)\,H_0(z^{-2}) + \underbrace{z^{-1} H_0^2(z^{-2}) + z\,H_0^2(z^2)}_{\textbf{odd powers of } z \textbf{ only}}$$

Therefore $H_{L2}(z)\,H_{L2}(z^{-1})$ must have **no terms in z^{4k}** except the term in z^0.

Hence we can include PR as a **direct design constraint on $H_{L2}(z)\,H_{L2}(z^{-1})$**.

Filter Design — Smoothness

To obtain smoothness when iterated over many scales:

- **Ensure that the stopband of $H_0(z)$ suppresses energy at frequencies where unwanted passbands appear from subsampled filters operating at coarser scales.**

Consider the combined frequency response of H_0 over just two scales:

$$H_0(z)\,H_0(z^2)|_{z=e^{j\omega}} = H_0(e^{j\omega})\,H_0(e^{2j\omega})$$

If the stopband of $H_0(e^{j\omega})$ covers $\omega_s \leq \omega \leq \pi$, then the unwanted transition band and passband of $H_0(e^{2j\omega})$ will extend from $\pi - \frac{\omega_s}{2}$ to π.

For $H_0(e^{j\omega})$ to suppress the unwanted bands of $H_0(e^{2j\omega})$:

$$\omega_s \leq \pi - \frac{\omega_s}{2} \qquad \text{and so} \quad \boldsymbol{\omega_s \leq \frac{2\pi}{3}} \qquad \text{(see fig 4)}$$

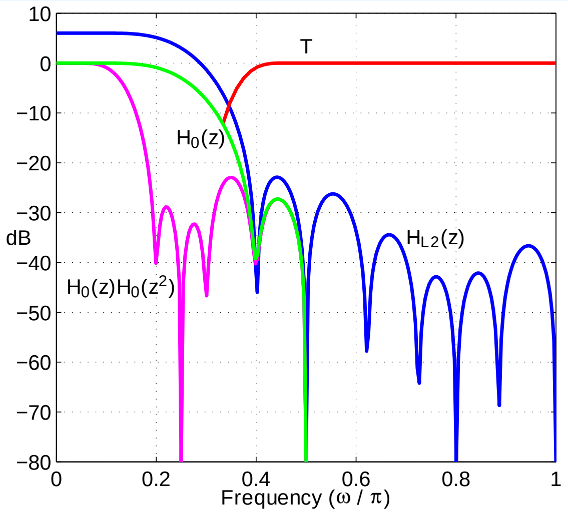


Fig 4: Frequency responses of $H_{L2}(z)$ (blue), $H_0(z)$ (green), $H_0(z)\ H_0(z^2)$ (magenta), and the gain correction matrix **T** (red) for $n = 6$ (12 taps for H_0).

Optimization for MSE in the frequency domain

We have now reduced the ideal design conditions for the length $4n$ symmetric lowpass filter H_{L2} to be:

- Zero amplitude for all the terms of $H_{L2}(z)\, H_{L2}(z^{-1})$ in z^{4k} except the term in z^0, which must be 1 (these are **quadratic constraints** on coef vector $\mathbf{h}_{L2}$);
- Zero (or near-zero) amplitude of $H_{L2}(e^{j\omega})$ for the stopband, $\frac{\pi}{3} \leq \omega \leq \pi$ (these are **linear constraints** on $\mathbf{h}_{L2}$).

If all constraints were linear, the LMS error solution for $\mathbf{h}_{L2}$ could be found using a matrix pseudo-inverse method. **Therefore we linearise the problem and iterate.**

If $\mathbf{h}_{L2}$ at iteration i is $\mathbf{h}_i = \mathbf{h}_{i-1} + \Delta\mathbf{h}_i$, then

$$\mathbf{h}_i * \mathbf{h}_i = (\mathbf{h}_{i-1} + \Delta\mathbf{h}_i) * (\mathbf{h}_{i-1} + \Delta\mathbf{h}_i) = \mathbf{h}_{i-1} * (\mathbf{h}_{i-1} + 2\Delta\mathbf{h}_i) + \Delta\mathbf{h}_i * \Delta\mathbf{h}_i$$

Since $\Delta\mathbf{h}_i$ **becomes small** as i increases, the final term can be neglected and the convolution ($*$) is expressed as a **linear function of $\Delta\mathbf{h}_i$**.

Hence we solve for $\Delta\mathbf{h}_i$ such that:

$$\begin{aligned} \mathbf{C}_{i-1}\,(\mathbf{h}_{i-1} + 2\Delta\mathbf{h}_i) &= [0\ldots 0\ 1]^T \\ \mathbf{F}\,(\mathbf{h}_{i-1} + \Delta\mathbf{h}_i) &\simeq [0\ldots 0]^T \end{aligned}$$

where $\mathbf{C}_{i-1}$ calculates every 4th term in the convolution with $\mathbf{h}_{i-1}$, and $\mathbf{F}$ evaluates the Fourier transform at M discrete frequencies ω from $\frac{\pi}{3}$ to π (typically $M \simeq 8n$).

Note that only one side of the symmetric convolution is needed in the rows of $\mathbf{C}_{i-1}$, and the columns of $\mathbf{C}_{i-1}$ and $\mathbf{F}$ can be combined in pairs so that only the first half of the symmetric $\Delta\mathbf{h}_i$ need be solved for.

To obtain **high accuracy solutions to the PR constraints**, we scale the equations in $\mathbf{C}_{i-1}$ up by $\beta_i = 2^i$ to get the following iterative LMS method for $\Delta\mathbf{h}_i$ and then $\mathbf{h}_i$:

$$\begin{bmatrix} 2\beta_i\mathbf{C}_{i-1} \\ \mathbf{F} \end{bmatrix} \Delta\mathbf{h}_i = \begin{bmatrix} \beta_i(\mathbf{c} - \mathbf{C}_{i-1}\mathbf{h}_{i-1}) \\ -\mathbf{F}\,\mathbf{h}_{i-1} \end{bmatrix} \quad \text{and} \quad \mathbf{h}_i = \mathbf{h}_{i-1} + \Delta\mathbf{h}_i$$

where $\mathbf{c} = [0\ldots 0\ 1]^T$.

Two final refinements

- To include **transition band** effects, we scale rows of $\mathbf{F}$ by diagonal matrix $\mathbf{T}_i$, the gain (at iteration i) of $H_0(z^2)/H_0(1)$ at frequencies corresponding to $\frac{\pi}{3} \leq \omega \leq \frac{\pi}{2}$ in the frequency domain of H_{L2} ($\mathbf{T}_i$ is the red curve in fig 4).
- To insert **predefined zeros** in $H_0(z)$ or $H_{L2}(z)$, we first note that a zero at $z = e^{j\pi}$ in H_0 will be produced by a pair of zeros at $z = e^{\pm j\pi/2}$ in H_{L2}. We can force zeros in H_{L2} by forming a convolution matrix $\mathbf{H}_f$ such that $\mathbf{H}_f\ \mathbf{h}'_i = \mathbf{h}_i$, where $\mathbf{h}'_i$ is the coef vector of the filter which represents all the zeros of H_{L2} that are **not** predefined, and $\mathbf{H}_f$ produces convolution with the predefined zeros.

Hence we now solve for $\Delta\mathbf{h}'_i$ and then $\mathbf{h}_i$ using

$$\begin{bmatrix} 2\beta_i \mathbf{C}_{i-1} \\ \mathbf{T}_{i-1}\mathbf{F} \end{bmatrix} \mathbf{H}_f\ \Delta\mathbf{h}'_i = \begin{bmatrix} \beta_i(\mathbf{c} - \mathbf{C}_{i-1}\mathbf{h}_{i-1}) \\ -\mathbf{T}_{i-1}\mathbf{F}\ \mathbf{h}_{i-1} \end{bmatrix} \quad \text{with} \quad \mathbf{h}_i = \mathbf{h}_{i-1} + \mathbf{H}_f\ \Delta\mathbf{h}'_i$$

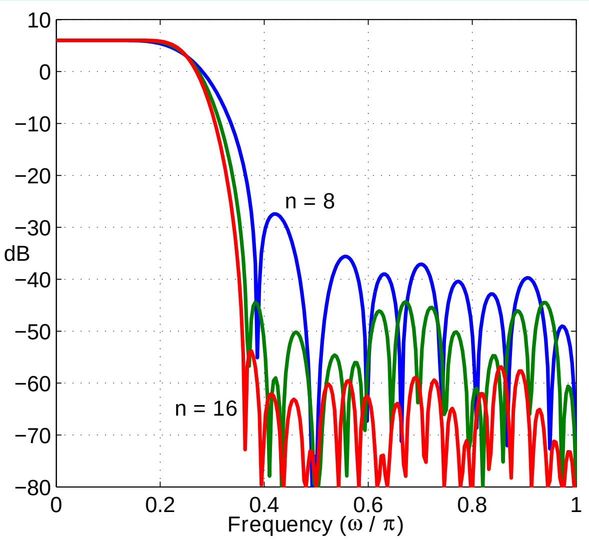


Fig 5: Frequency responses of $H_{L2}(z)$ for $n = 8$ (blue), $n = 12$ (green) and $n = 16$ (red). Each filter has one predefined zero at $\omega = \frac{\pi}{2}$ and one at $\omega = \pi$.

Initialisation

To initialise the iterative algorithm when $i = 1$, we must define $\mathbf{h}_0$ and hence $\mathbf{C}_0$ and $\mathbf{T}_0$.
This is not critical and can be achieved by a simple inverse FFT of an 'ideal' lowpass frequency response for $H_{L2}(e^{j\omega})$ with a root-raised-cosine transition band covering the range

$$\tfrac{\pi}{6} < \omega < \tfrac{\pi}{3}$$

The impulse response is truncated symmetrically to length $4n$ to obtain $\mathbf{h}_0$.
$\mathbf{C}_0$ and $\mathbf{T}_0$ may then be calculated from $\mathbf{h}_0$.

Convergence

For some larger values of n, convergence can be slow. We have found this can be improved by using

$$\mathbf{h}_i = \mathbf{h}_{i-1} + \alpha \mathbf{H}_f \, \Delta\mathbf{h}'_i \qquad \text{where} \quad 0 < \alpha < 1 \quad (\text{e.g. } \alpha \sim 0.8)$$

Filter Design Results

- Figs. 4 and 5 show the frequency responses of $H_{L2}(z)$ for the cases $n = 6$, 8, 12 and 16, when there is one predefined zero at $\omega = \frac{\pi}{2}$ and one at $\omega = \pi$.
- Figs. 6 to 15 show, for a range of values of n, the impulse response of $H_{L2}(z)$, the level-4 DT $\mathbb{C}$WT scaling functions and wavelets, the frequency responses of $H_0(z)$ and of $H_0(z)\,H_0(z^2)$, and the group delay of $H_0(z)$.
- Figs. 6 to 11 show these responses for the cases $n = 5$, 6 and 7, with either 0 or 1 predefined zero in $H_0(z)$ at $\omega = \pi$.
- Figs. 12 to 15 show these responses for the cases $n = 8$, 12 and 16, with 1 predefined zero in $H_0(z)$ at $\omega = \pi$.

Note how the responses improve with increasing n. The effect of predefining a zero in H_0 is in general quite small. More predefined zeros tend to degrade performance.
$n = 7$ gives a good tradeoff between complexity and performance.

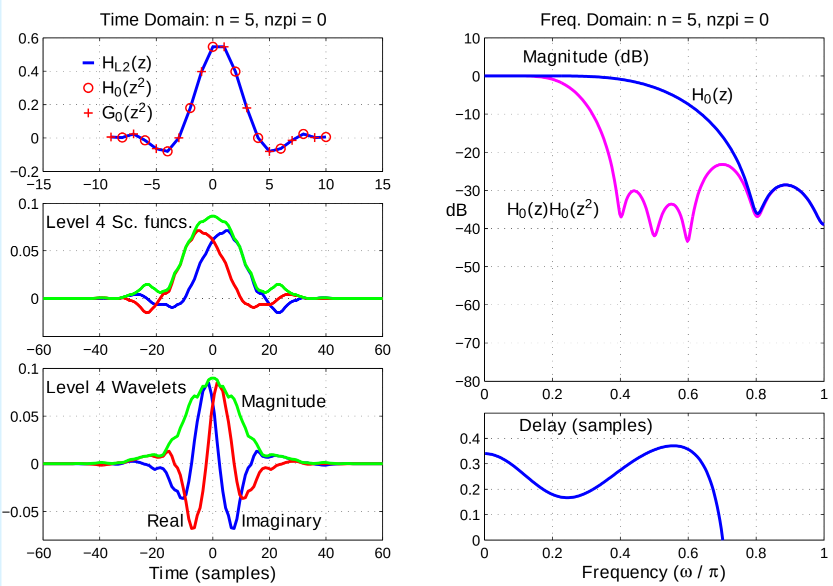


Fig 6: Q-shift filters for $n = 5$ (10 filter taps) and no predefined zeros.

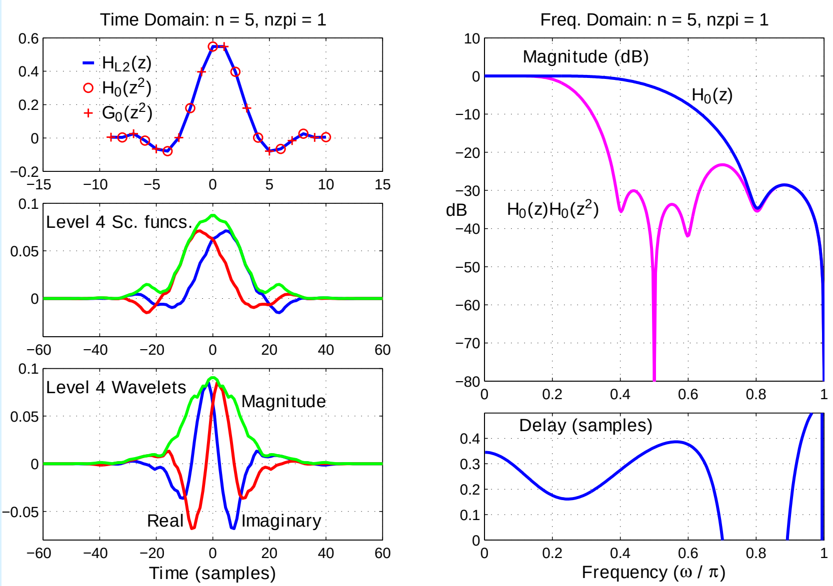


Fig 7: Q-shift filters for $n = 5$ (10 filter taps) and 1 predefined zero at $\omega = \pi$.

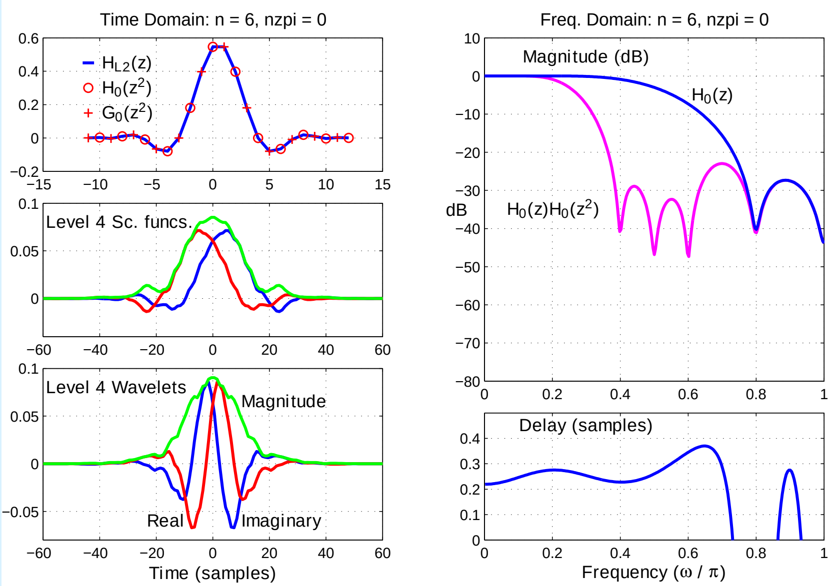


Fig 8: Q-shift filters for $n = 6$ (12 filter taps) and no predefined zeros.

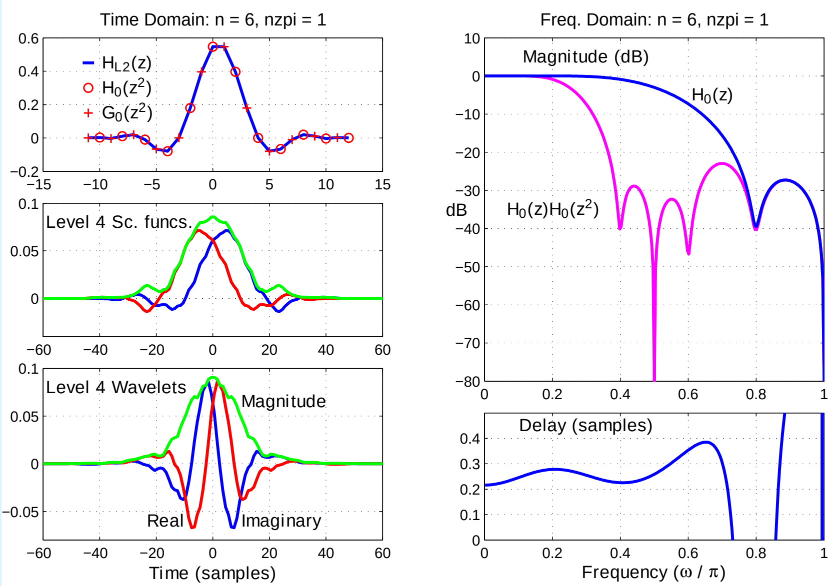


Fig 9: Q-shift filters for $n = 6$ (12 filter taps) and 1 predefined zero at $\omega = \pi$.

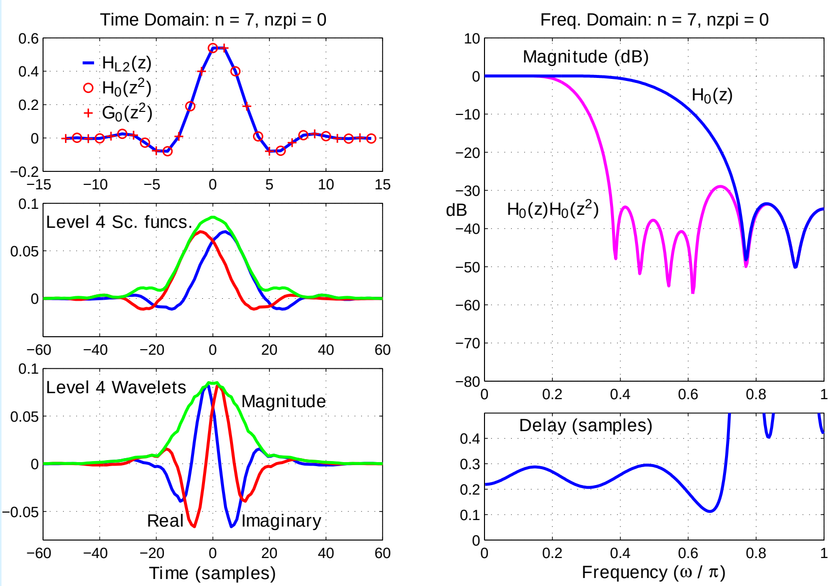


Fig 10: Q-shift filters for $n = 7$ (14 filter taps) and no predefined zeros.

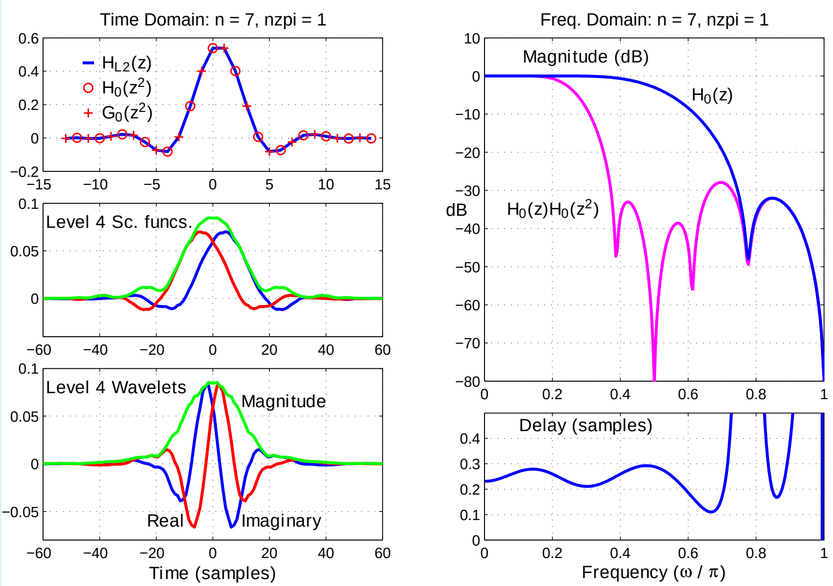


Fig 11: Q-shift filters for $n = 7$ (14 filter taps) and 1 predefined zero at $\omega = \pi$.

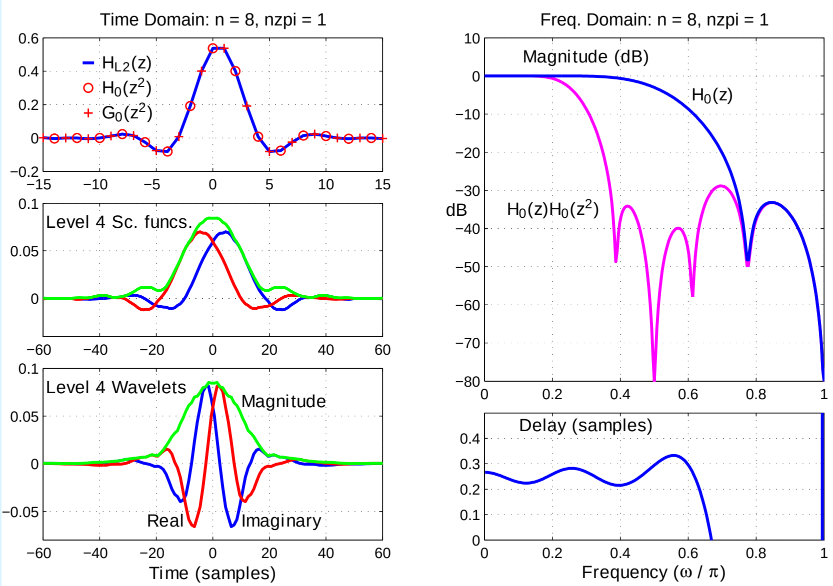


Fig 12: Q-shift filters for $n = 8$ (16 filter taps) and 1 predefined zero at $\omega = \pi$.

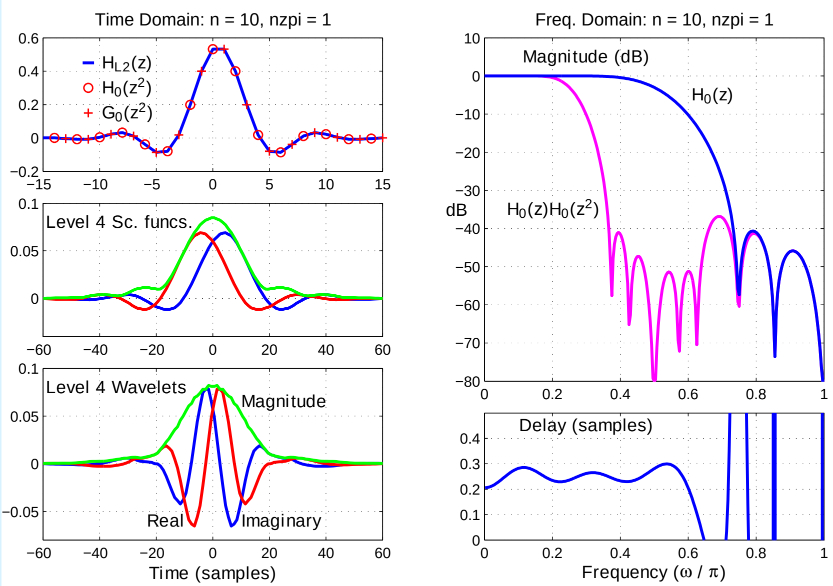


Fig 13: Q-shift filters for $n = 10$ (20 filter taps) and 1 predefined zero at $\omega = \pi$.

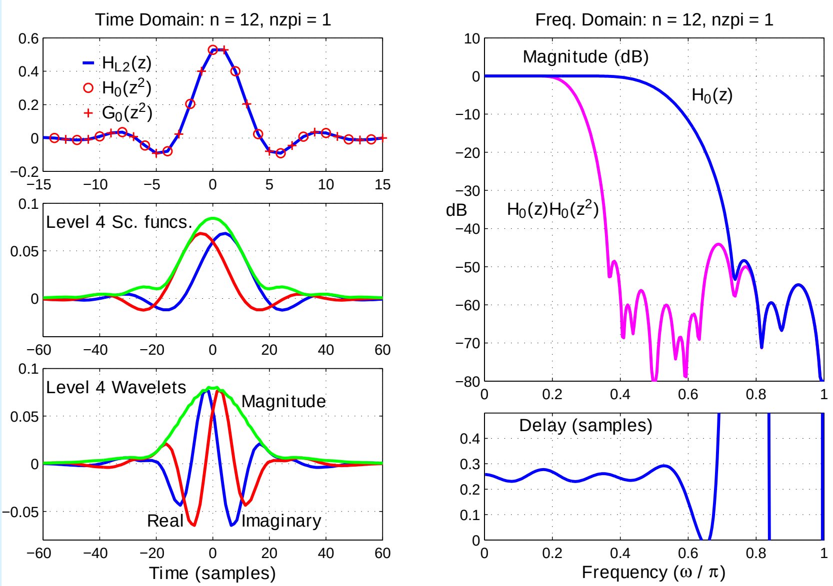


Fig 14: Q-shift filters for $n = 12$ (24 filter taps) and 1 predefined zero at $\omega = \pi$.

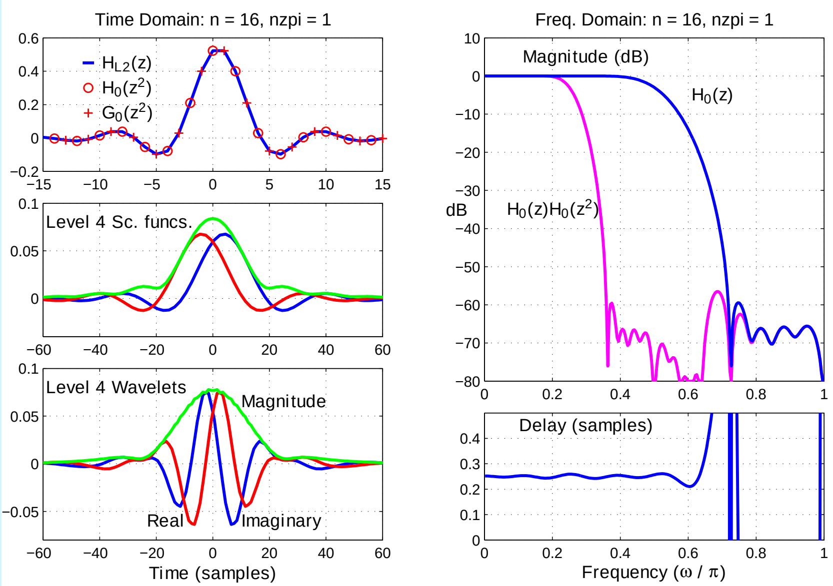


Fig 15: Q-shift filters for $n = 16$ (32 filter taps) and 1 predefined zero at $\omega = \pi$.

Filter Design – Conclusions

- The proposed algorithm gives a fast and effective way of designing Q-shift filters for the DT $\mathbb{C}$WT.
- All filters produce perfect reconstruction, tight frames and linear-phase complex wavelets.
- As the length of the filters ($2n$) increases, the design method gives improvements in stopband attenuation, constancy of group delay, and smoothness in the resulting wavelet bases. Hence we get increasing accuracy of shift-invariance.
- The algorithm works well for filter lengths from 10 to over 50 taps.
- Matlab code for the algorithm and papers on the DT $\mathbb{C}$WT can be downloaded from the author's website, **http://www-sigproc.eng.cam.ac.uk/~ngk/**.
- Matlab code to implement the DT $\mathbb{C}$WT is free for researchers and available by emailing the author at **ngk@eng.cam.ac.uk** .

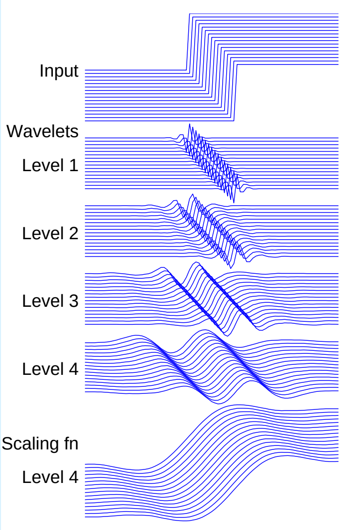
Shift Invariance

Visualising Shift Invariance / Dependence

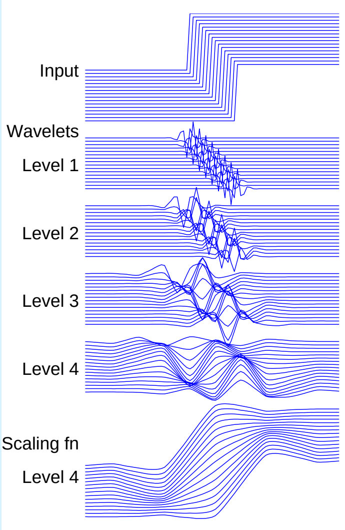
- Apply a standard input (e.g. unit step) to the transform for a **range of shift positions**.
- Select the transform coefficients from **just one wavelet level** at a time.
- Inverse transform each set of selected coefficients.
- Plot the component of the reconstructed output for each shift position at each wavelet level.
- Check for **shift invariance** (similarity of waveforms).

See Matlab demonstration / next slide.

Shift Invariance of DT ℂWT / Dependence of DWT

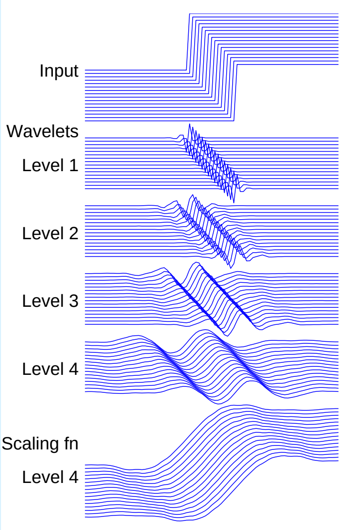


DT ℂWT, $n = 9$ (18-tap filters)

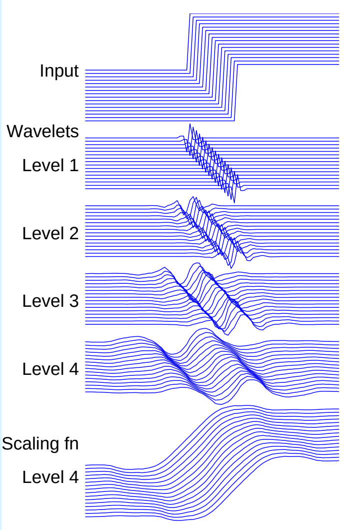


Real DWT, (13,19-tap filters)

Shift Invariance of simpler DT $\mathbb{C}$WTs



DT $\mathbb{C}$WT, $n = 7$ (14-tap filters)



DT $\mathbb{C}$WT, $n = 5$ (10-tap filters)

Shift Invariance – quantitative measurement

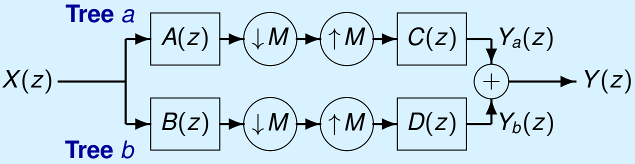


Fig 16: Basic configuration of the dual tree if either wavelet or scaling-function coefficients from just level m are retained ($M = 2^m$).

Letting $W = e^{j2\pi/M}$, **multi-rate analysis** gives:

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(W^k z)[A(W^k z)\, C(z) + B(W^k z)\, D(z)]$$

For shift invariance, **aliasing terms ($k \neq 0$) must be negligible.** So we design $B(W^k z)\, D(z)$ to cancel $A(W^k z)\, C(z)$ for all non-zero k that give overlap of the passbands of filters $C(z)$ or $D(z)$ with those of shifted filters $A(W^k z)$ or $B(W^k z)$.

A Measure of Shift Invariance – Aliasing Energy Ratio R_a

Since

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(W^k z)[A(W^k z)\, C(z) + B(W^k z)\, D(z)]$$

we quantify the shift dependence of a transform by calculating the ratio of the total energy of the **unwanted aliasing transfer functions** (the terms with $k \neq 0$) to the energy of the **wanted transfer function** (when $k = 0$):

$$R_a = \frac{\sum_{k=1}^{M-1} \mathcal{E}\{A(W^k z)\, C(z) + B(W^k z)\, D(z)\}}{\mathcal{E}\{A(z)\, C(z) + B(z)\, D(z)\}}$$

where $\mathcal{E}\{U(z)\}$ calculates the energy, $\sum_r |u_r|^2$, of the impulse response of a z-transfer function, $U(z) = \sum_r u_r z^{-r}$.

$\mathcal{E}\{U(z)\}$ may also be interpreted in the **frequency domain** as the integral of the squared magnitude of the frequency response, $\frac{1}{2\pi} \int_{-\pi}^{\pi} |U(e^{j\theta})|^2 \, d\theta$ from Parseval's theorem.

Types of DT $\mathbb{C}$WT filters

We show results for the following combinations of filters:

A: **(13,19)-tap** near-orthogonal filters at level 1,
18-tap Q-shift filters at levels ≥ 2. (Most complex)

B: **(13,19)-tap** near-orthogonal filters at level 1,
14-tap Q-shift filters at levels ≥ 2.

C: **(9,7)-tap** bi-orthogonal filters at level 1,
18-tap Q-shift filters at levels ≥ 2.

D: **(9,7)-tap** bi-orthogonal filters at level 1,
14-tap Q-shift filters at levels ≥ 2.

E: **(9,7)-tap** bi-orthogonal filters at level 1,
6-tap Q-shift filters at levels ≥ 2.

F: **(5,3)-tap** bi-orthogonal filters at level 1,
6-tap Q-shift filters at levels ≥ 2. (Least complex)

Aliasing Energy Ratios

Values of R_a **in dB**, for filter types A to F over levels 1 to 5:

Filters:	A	B	C	D	E	F	DWT
	13,19+18	13,19+14	9,7+18	9,7+14	9,7+6	5,3+6	13,19
Complexity:	2.3	2.0	1.9	1.6	1.0	0.7	1.0
Wavelet							
Level 1	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	-9.40
Level 2	-31.40	-29.06	-22.96	-21.81	-18.49	-14.11	-3.54
Level 3	-27.93	-25.10	-20.32	-18.96	-14.60	-11.00	-3.53
Level 4	-31.13	-24.67	-32.08	-24.85	-16.78	-15.80	-3.52
Level 5	-31.70	-24.15	-31.88	-24.15	-18.94	-18.77	-3.52
Scaling fn.							
Level 1	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	-9.40
Level 2	-32.50	-30.17	-24.32	-23.19	-19.88	-15.93	-9.38
Level 3	-35.88	-29.21	-36.94	-29.33	-21.75	-20.63	-9.37
Level 4	-37.14	-28.57	-37.37	-28.56	-24.37	-24.15	-9.37
Level 5	-36.00	-28.57	-36.01	-28.57	-24.67	-24.65	-9.37

Conclusions – DT $\mathbb{C}$WT

The **Dual-Tree Complex Wavelet Transform** provides:

- Approximate **shift invariance**
- **Directionally selective** filtering in 2 or more dimensions
- **Low redundancy** – only $2^m : 1$ for m-D signals
- **Perfect reconstruction**
- **Orthonormal filters** below level 1, but still giving **linear phase** (conjugate symmetric) complex wavelets
- **Low computation** – order-N; less than 2^m times that of the fully decimated DWT ($\sim$ 3.3 times in 2-D, $\sim$ 5.1 times in 3-D)
- A **general purpose multi-resolution front-end** for many image analysis and reconstruction tasks . . . (see next slide)

Conclusions – DT $\mathbb{C}$WT (cont.)

- A **general purpose multi-resolution front-end** for many image analysis and reconstruction tasks:
  - Tight-frame sparse representation of images and 3D data
  - Denoising
  - Enhancement (deconvolution)
  - Motion / displacement estimation and compensation
  - Registration in 2D or 3D.
  - Texture analysis / synthesis
  - Segmentation and classification
  - Feature-point detection and rotation-invariant description
  - Object recognition and tracking

Papers on complex wavelets are available at:

http://www.eng.cam.ac.uk/~ngk/

A Matlab DTCWT toolbox is available on request from:

ngk@eng.cam.ac.uk