Iterative Sparsity Methods for Coding and Deconvolution with Overcomplete Transforms

Nick Kingsbury, Tanya Reeves and Yingsong Zhang

Signal Processing & Communications Group, Dept. of Engineering
University of Cambridge, Cambridge CB2 1PZ, UK.
ngk@eng.cam.ac.uk
www.eng.cam.ac.uk/~ngk

Inspire Sparsity Workshop, Cambridge, 14 & 15 Dec 2008

UNIVERSITY OF CAMBRIDGE

Iterative Sparsity Methods for Coding / Compression with Overcomplete Transforms

Redundant representation with complex wavelets: How to achieve sparsity ?

- Brief overview of **dual-tree** complex wavelets:
  - Dual tree in 1-D – shift invariance
  - Dual tree in 2-D – directional selectivity

- **Iterative projection** method of coding with overcomplete transforms (frames):
  - How iterative projection can improve sparsity, and hence rate-distortion performance
  - Good convergence strategies
  - Results and comparisons with non-redundant real wavelet transforms (DWTs)

FEATURES OF THE DUAL TREE COMPLEX WAVELET TRANSFORM (DT CWT)

- Good **shift invariance**.
- Good **directional selectivity** in 2-D, 3-D etc.
- **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 and typically 3 times that of the maximally decimated DWT. (Lifting methods can still be used to improve efficiency.)

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

Q-shift Dual Tree Complex Wavelet Transform in 1-D

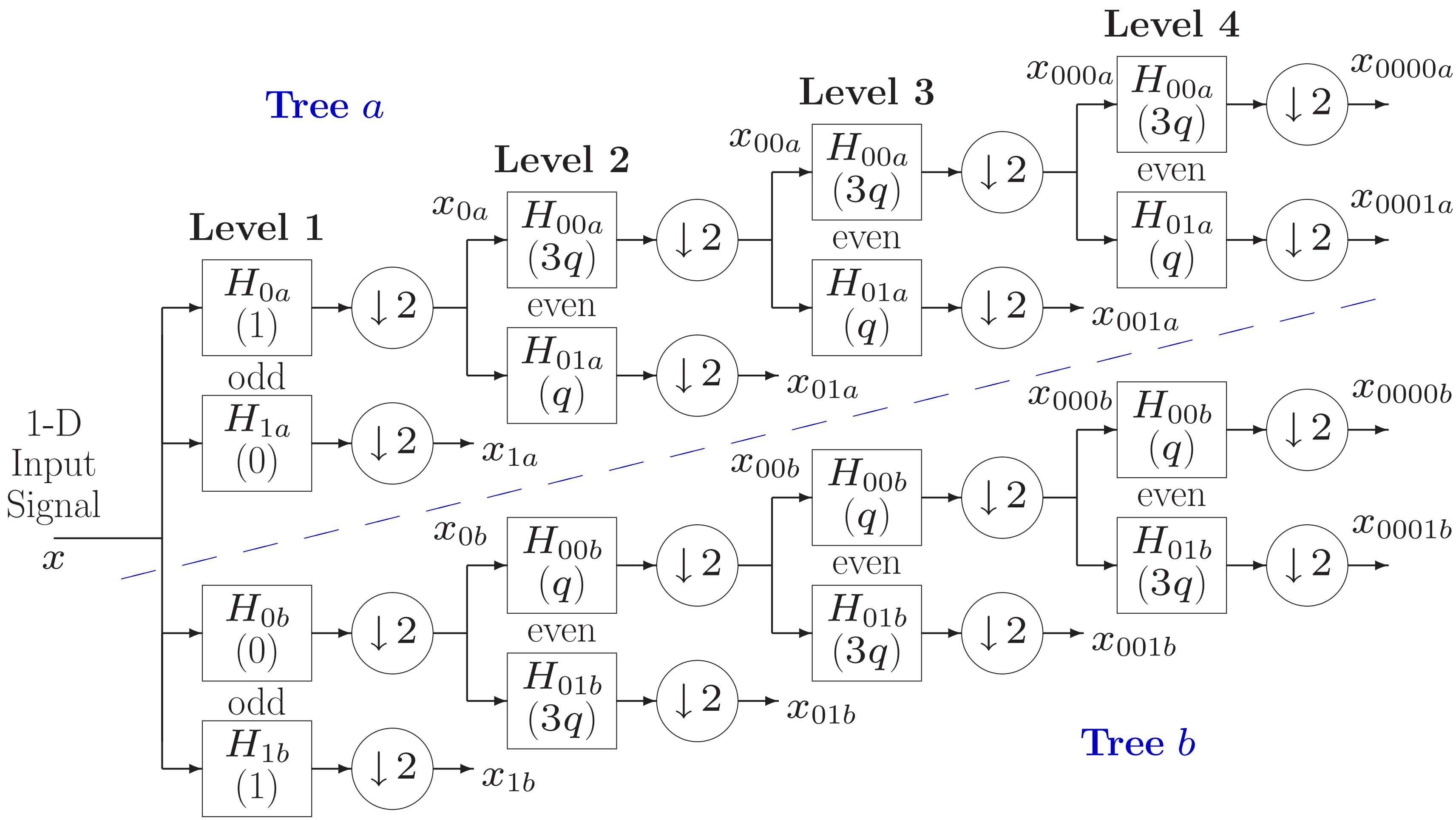


Figure 1: Dual tree of real filters for the Q-shift CWT, giving real and imaginary parts of complex coefficients from tree a and tree b respectively. Figures in brackets indicate the approximate delay for each filter, where $q = \frac{1}{4}$ sample period.

1-D Basis Functions at Level 4

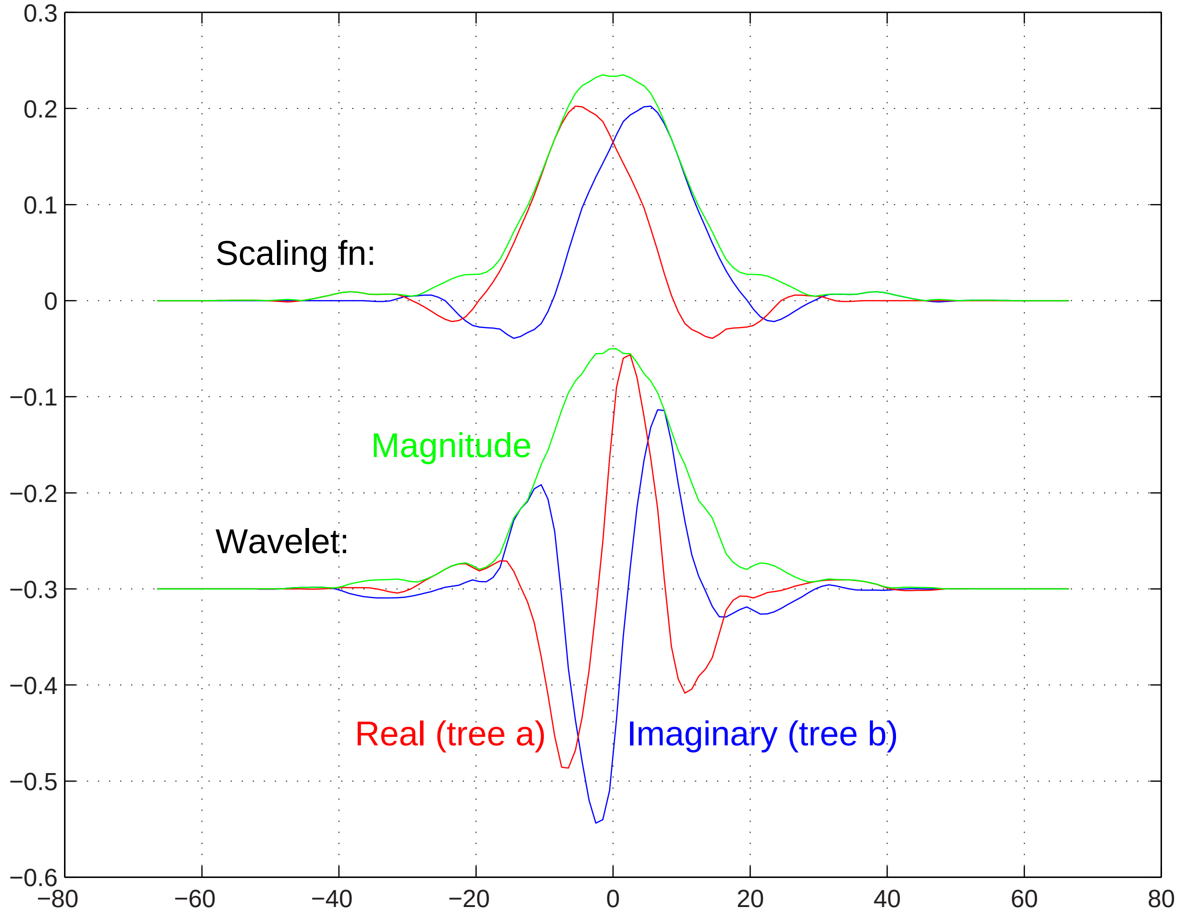


Figure 2: Scaling function and wavelet basis functions of the DT CWT at level 4, using the Daubechies 7-tap filter for level 1 (from 9,7 biorth. pair) and the 6-tap Q-shift wavelet filters for levels 2, 3 and 4.

The DT CWT in 2-D

When the DT CWT is applied to 2-D signals (images), it has the following features:

- It is performed **separably**, with 2 trees used 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 CWT. Fig 3 shows the basis functions of these subbands at level 4, and compares them with the 3 subbands of a 2-D DWT.

- The DT CWT is directionally selective (see fig 3) 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

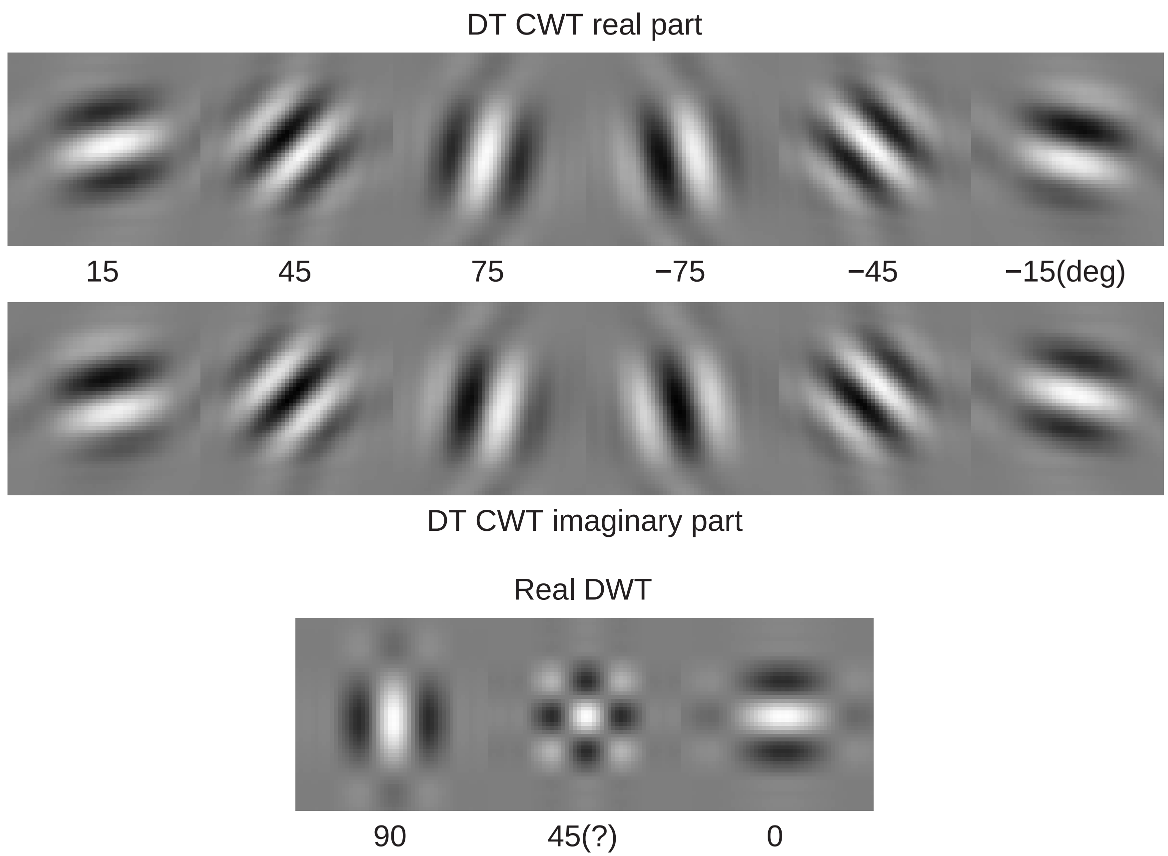


Figure 3: Basis functions of 2-D Q-shift complex wavelets (top), and of 2-D real wavelet filters (bottom), all illustrated at level 4 of the transforms. The complex wavelets provide 6 directionally selective filters, while real wavelets provide 3 filters, only two of which have a dominant direction.

2-D Shift Invariance of DT CWT vs DWT

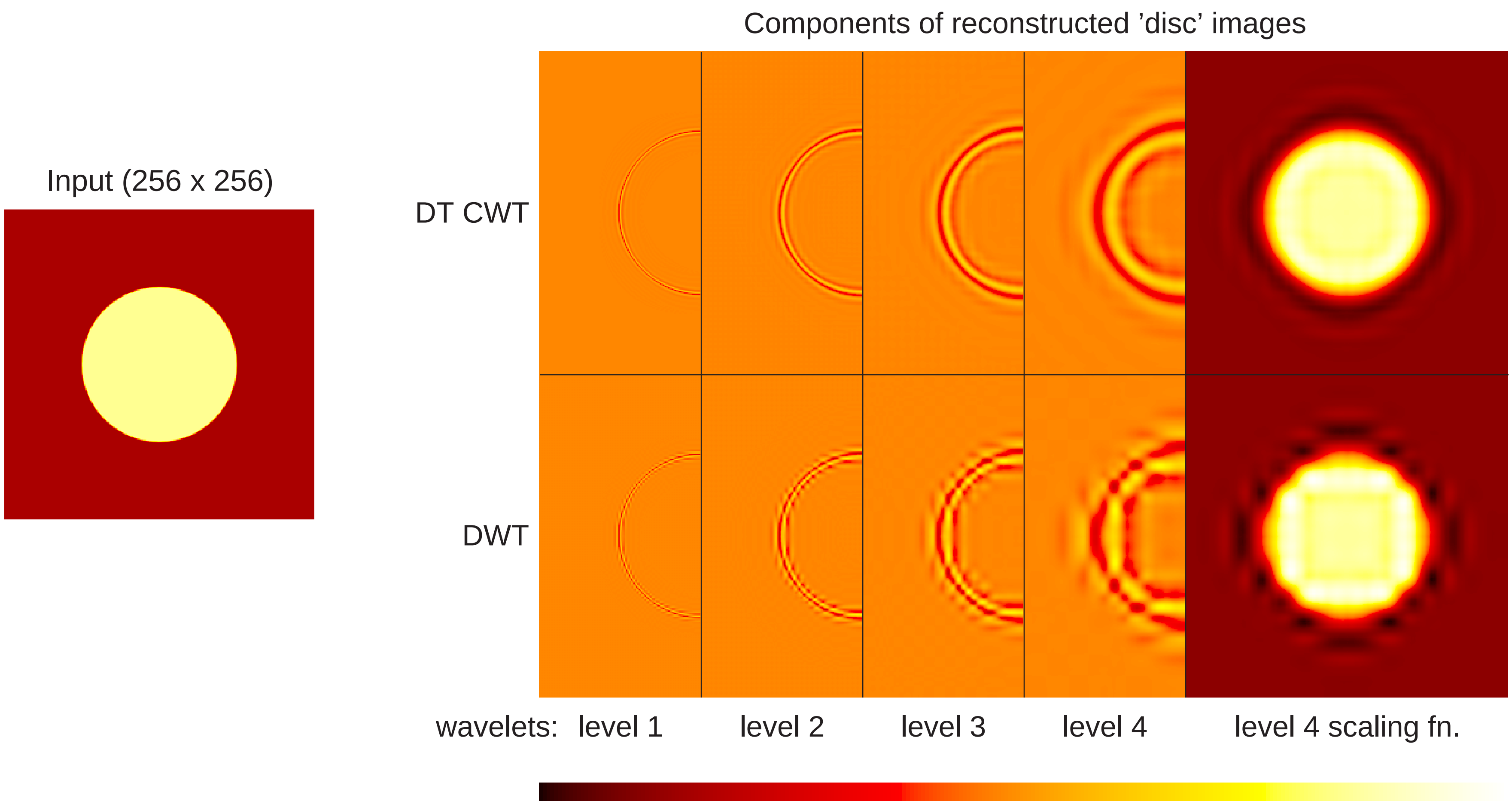


Figure 4: Wavelet and scaling function components at levels 1 to 4 of an image of a light circular disc on a dark background, using the 2-D DT CWT (upper row) and 2-D DWT (lower row). Only half of each wavelet image is shown in order to save space.

Coding with the DT CWT

- DT CWT is $4:1$ **redundant** – Why use it for compression?

Because:

- Overcomplete dictionaries of basis functions are known to provide the **potential for better coding** (e.g. Matching Pursuits).

- The 4 reconstruction trees **average** the quantisation noise.

- Reconstruction is a **projection** from $4N$-space to N-space. Noise components, which are not in the N-dimensional range space of the transform, are in the $3N$-dimensional null space and **do not affect the decoded image**.

- Complex wavelet coefficients can define edge locations more accurately than real coefficients.

How to achieve sparsity ?

Basic Algorithm – motivated by Matching Pursuits:

1. Set $i = 1$ and take the DT CWT of the input image.

2. Set to zero all wavelet coefs with magnitude smaller than a threshold θ_i.

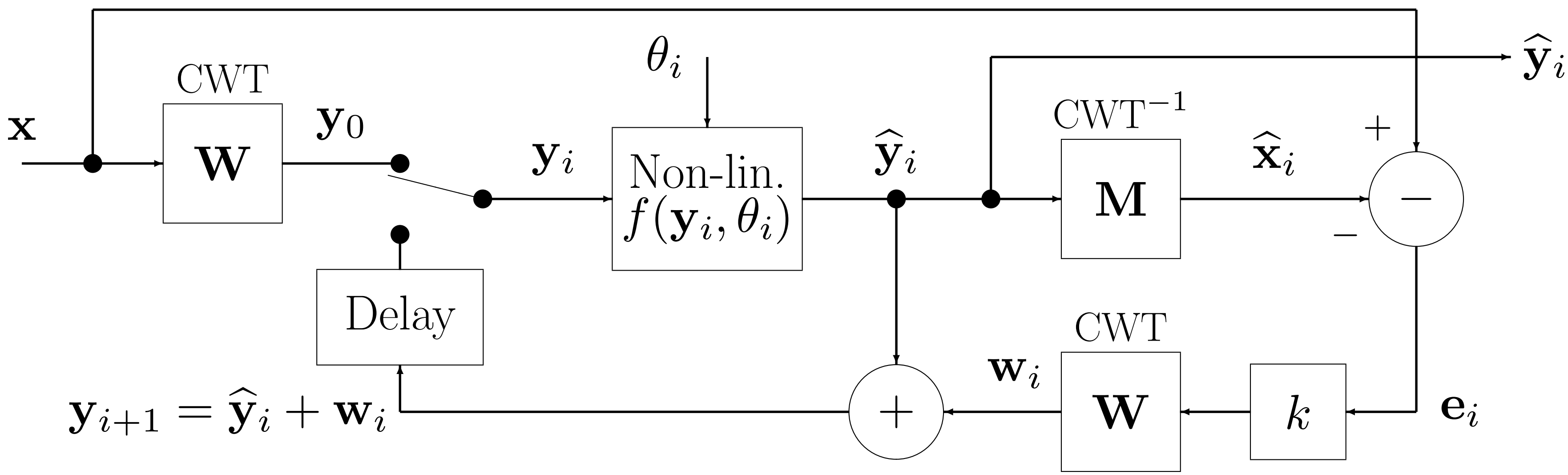
3. Take DT CWT^{-1} and measure the error due to loss of smaller coefs.

4. Take DT CWT of the error image and adjust the non-zero wavelet coefs from step **2** to reduce the error.

5. Increment i, reduce θ_i a little (to include a few more non-zero coefs) and repeat steps **2** to **4**.

6. When there are sufficient non-zero coefs to give the required rate-distortion tradeoff, keep θ_i constant and iterate a few more times until converged.

ITERATIVE PROJECTION



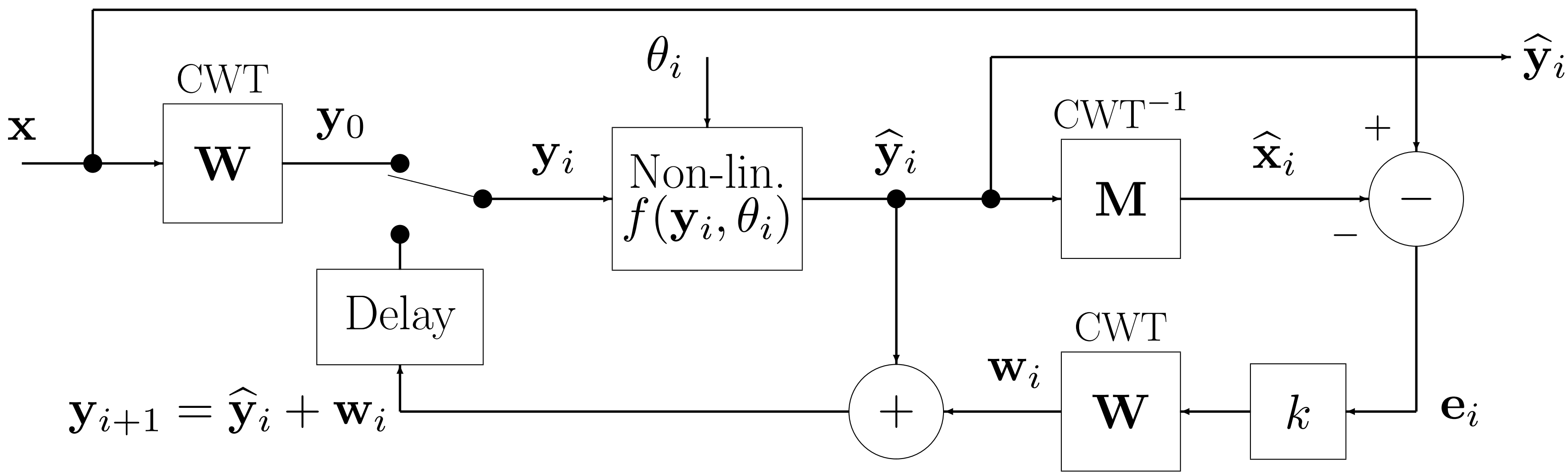
If $\mathcal{S}$ is the range space of the DT CWT, projection onto $\mathcal{S}$ is $\mathbf{P}^{\mathcal{S}} = \mathbf{WM}$, and onto the null space is $\mathbf{P}^{\perp} = \mathbf{I} - \mathbf{P}^{\mathcal{S}}$.

On iteration i: $$\mathbf{w}_i = k\mathbf{W}(\mathbf{x} - \mathbf{M}\widehat{\mathbf{y}}_i) = k\mathbf{y}_0 - k\mathbf{P}^{\mathcal{S}}\widehat{\mathbf{y}}_i$$

$$\therefore\ \mathbf{y}_{i+1} = \widehat{\mathbf{y}}_i + \mathbf{w}_i = k\mathbf{y}_0 + (\mathbf{I} - k\mathbf{P}^{\mathcal{S}})\widehat{\mathbf{y}}_i = \mathbf{y}_0 + \mathbf{P}^{\perp}\widehat{\mathbf{y}}_i \quad \text{if } k = 1$$

Thus on each iteration the range-space component of $\mathbf{y}_{i+1}$ remains at $\mathbf{y}_0$ (so its inverse transform is always $\mathbf{x}$) while its null-space component varies and attempts to minimise $||\mathbf{e}_i||$. Note that **$\mathbf{y}_{i+1}$ is a projection of $\widehat{\mathbf{y}}_i$**.

Convergence

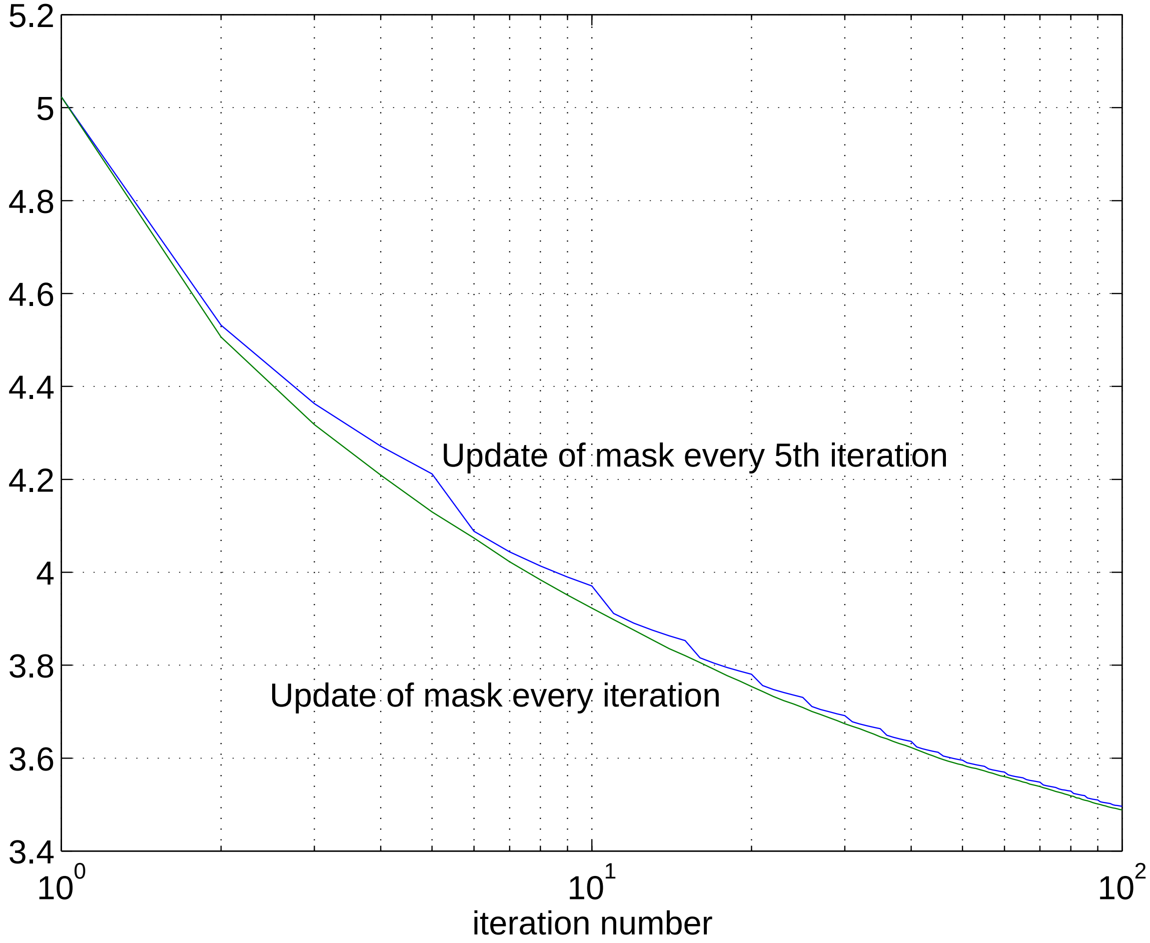


With a centre-clipping non-linearity and $k = 1$, convergence to a **local minimum** can be proved by **Projection onto Convex Sets** (POCS).

Substantial improvements in the converged result can be achieved by:

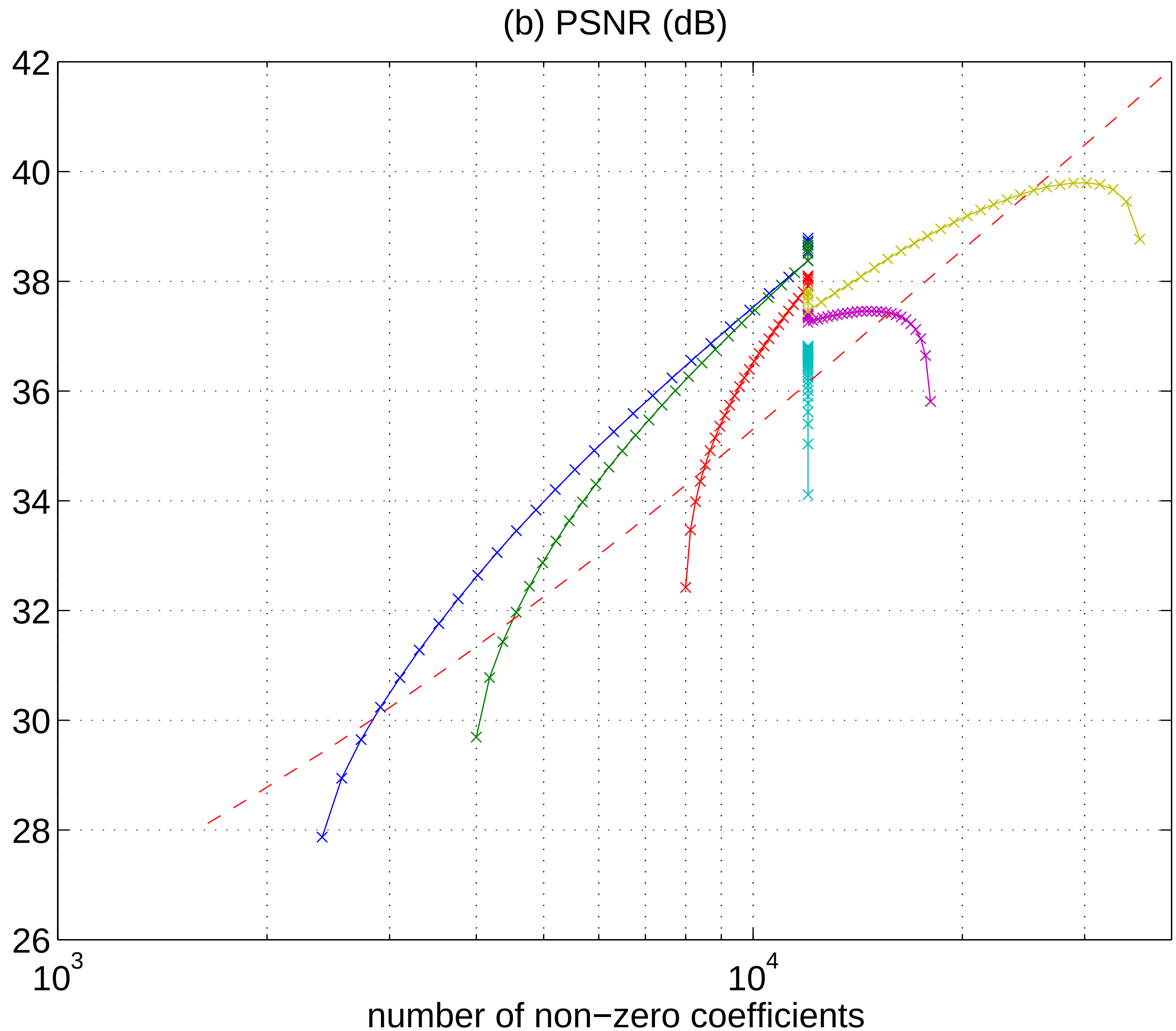
- Gradual **reductions in clip threshold** θ_i with i.
- Use of a **soft non-linearity**, such as a Wiener function $\widehat{\mathbf{y}}_i = \mathbf{y}_i \,.\, (|\mathbf{y}_i|^2 - \theta_i^2)_+ \;/\; |\mathbf{y}_i|^2$, for early iterations.
- **Increasing** k (must be kept < 2 for stability). $k \approx 1.8$ is good.

Convergence of loop RMS error for Centre-Clipper



The centre-clipper first selects a mask of coefs to clip, and then multiplies by the mask (a projection operation - hence can use POCS).

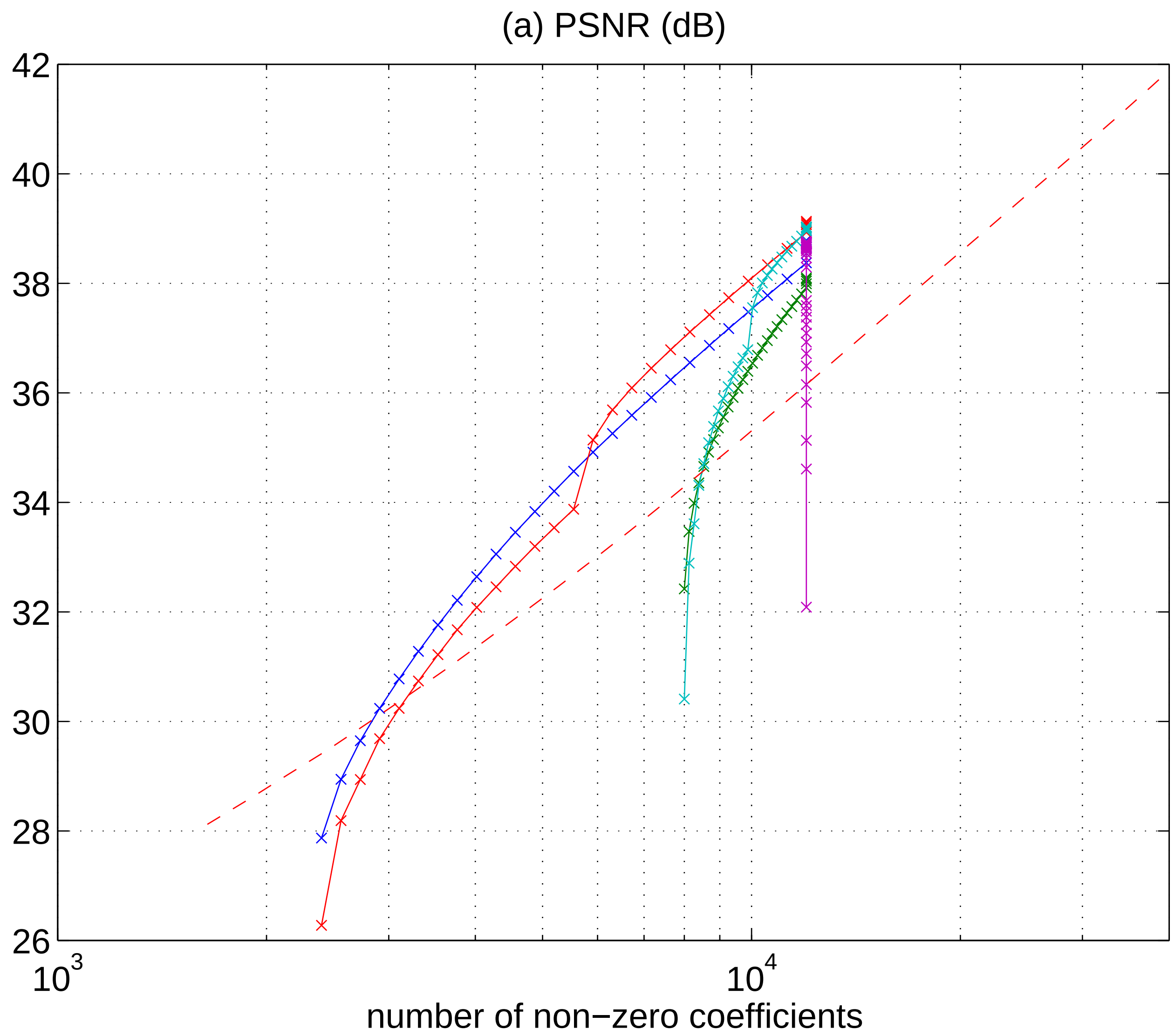
Threshold Modification Experiments for DT CWT ($k = 1$)



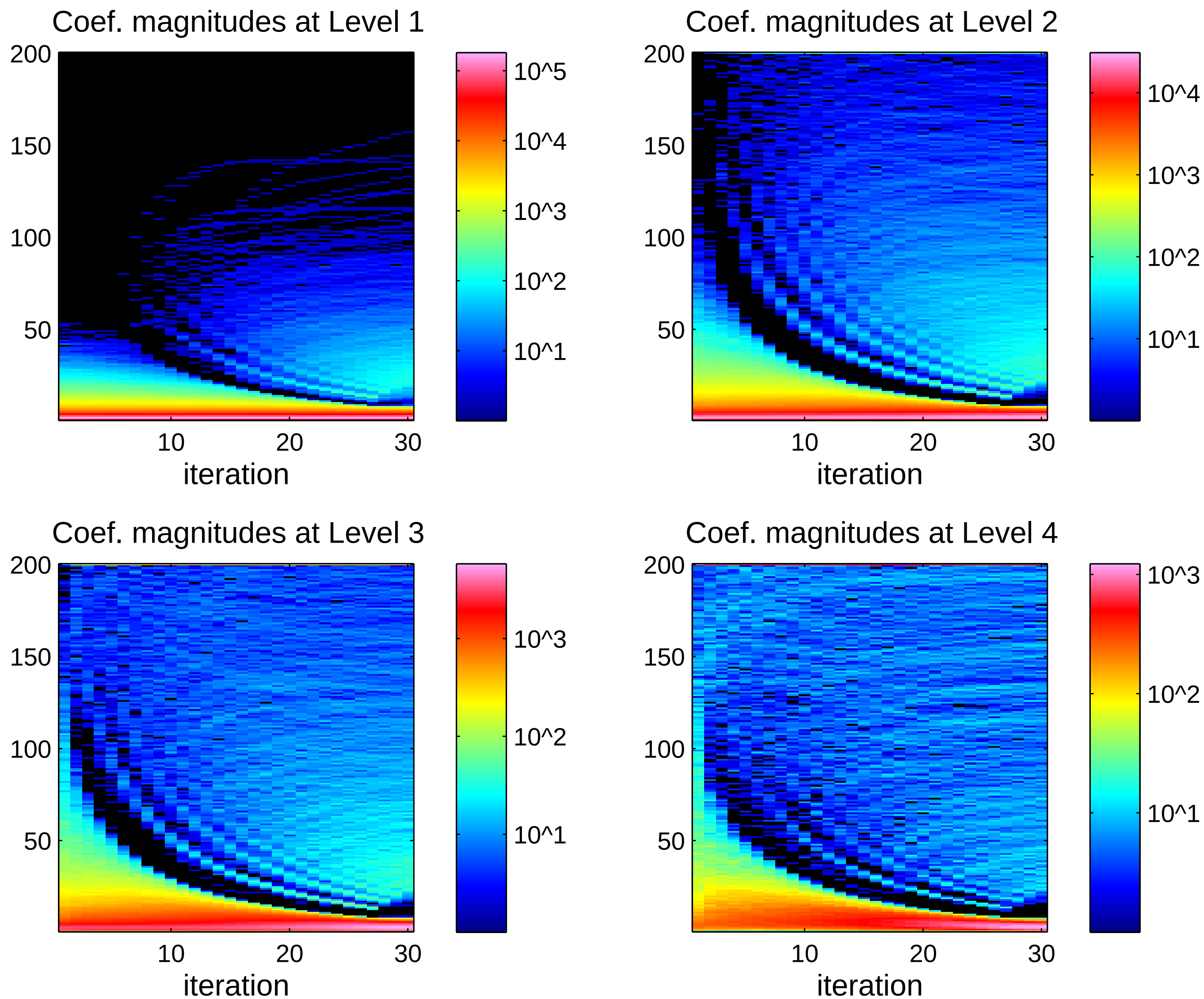
- - - shows non-redundant DWT for reference.

Threshold Modification Experiments:

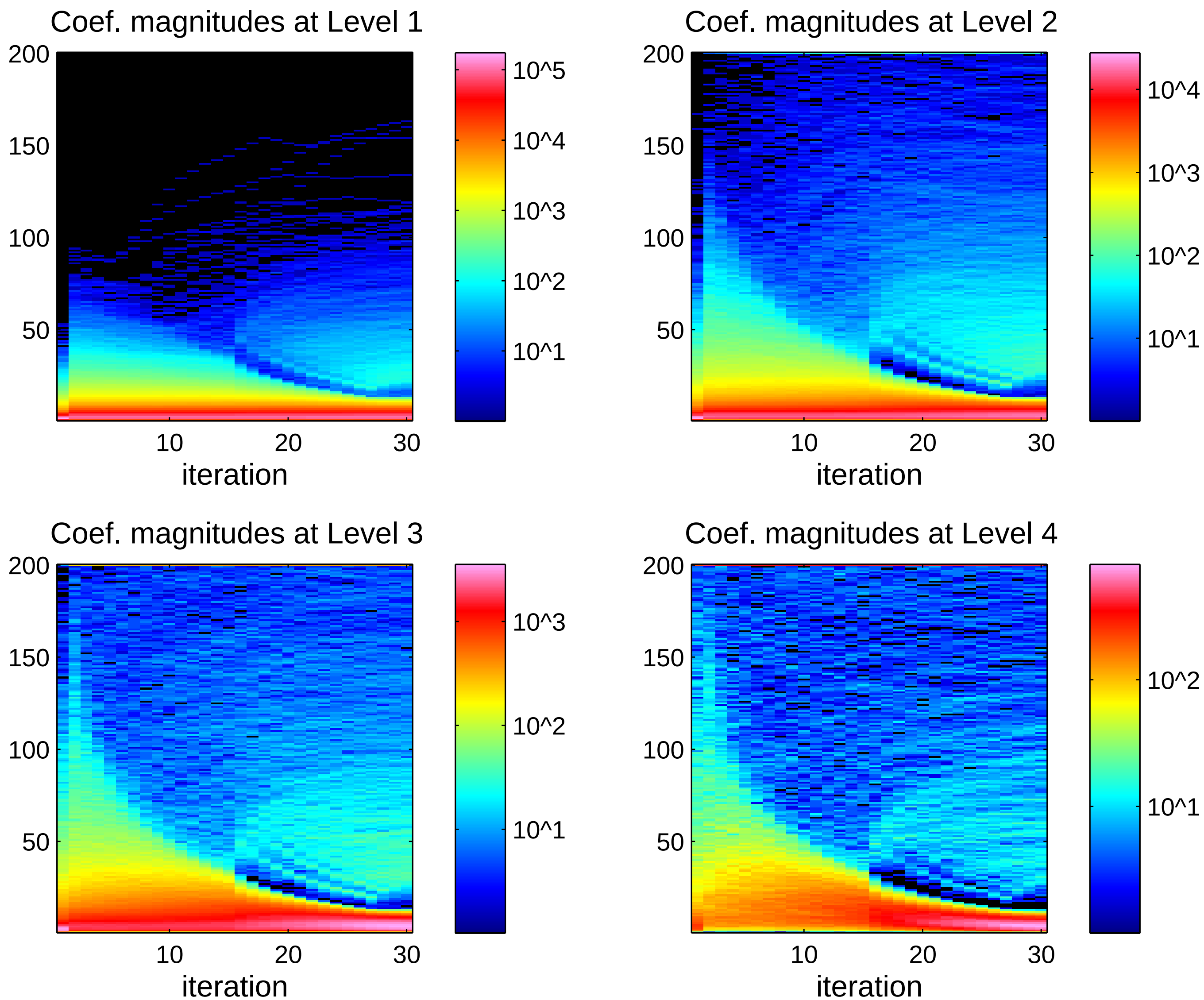
$k = 1.8$ and Wiener non-linearity for first 15 iterations (better by 0.34 dB).



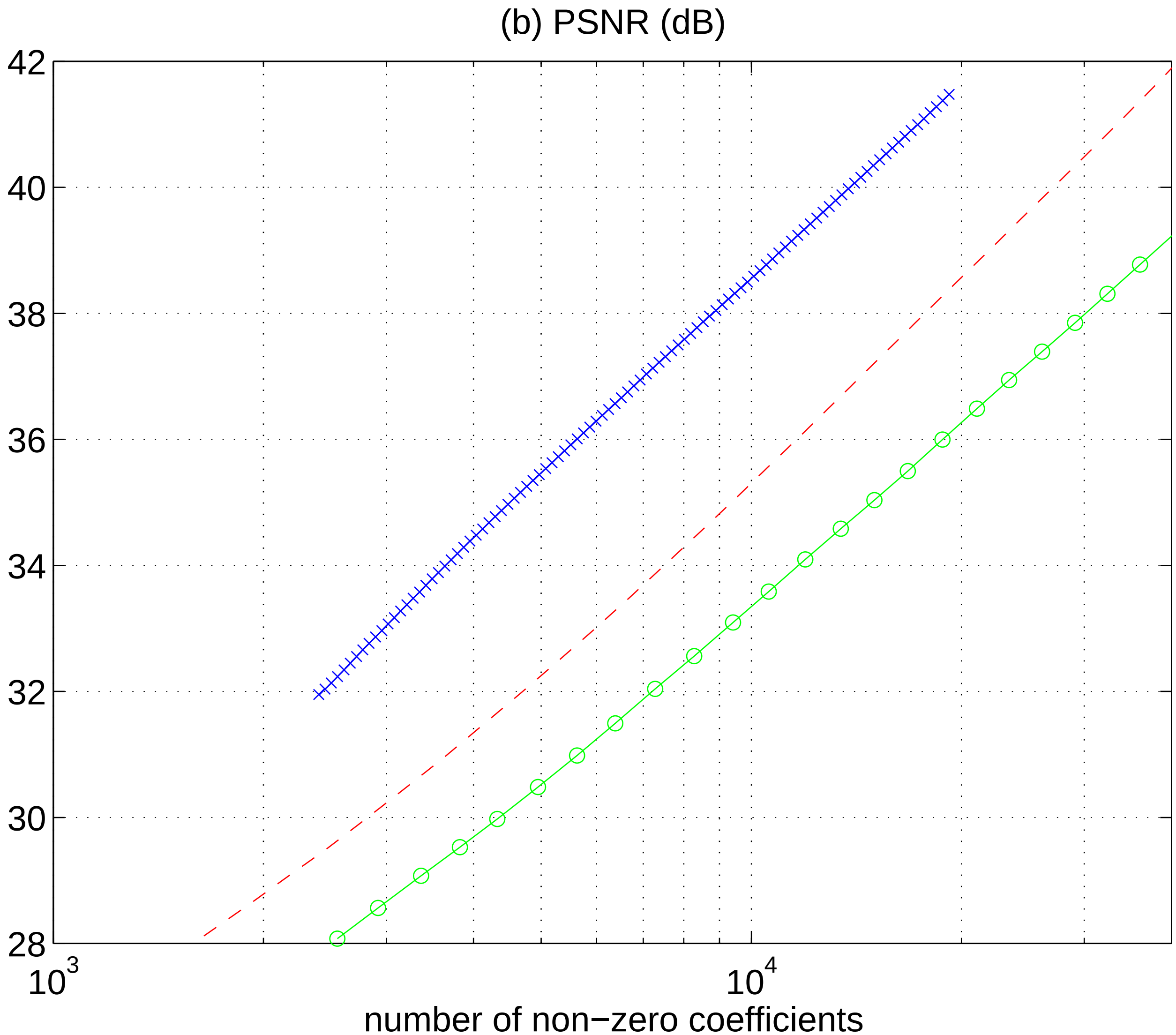
Histograms of DT CWT coefs $\mathbf{y}_i$: $k = 1$ and hard threshold.



HISTOGRAMS OF DT CWT COEFS $\mathbf{y}_i$: $k = 1.8$ and Wiener for 15 iters.

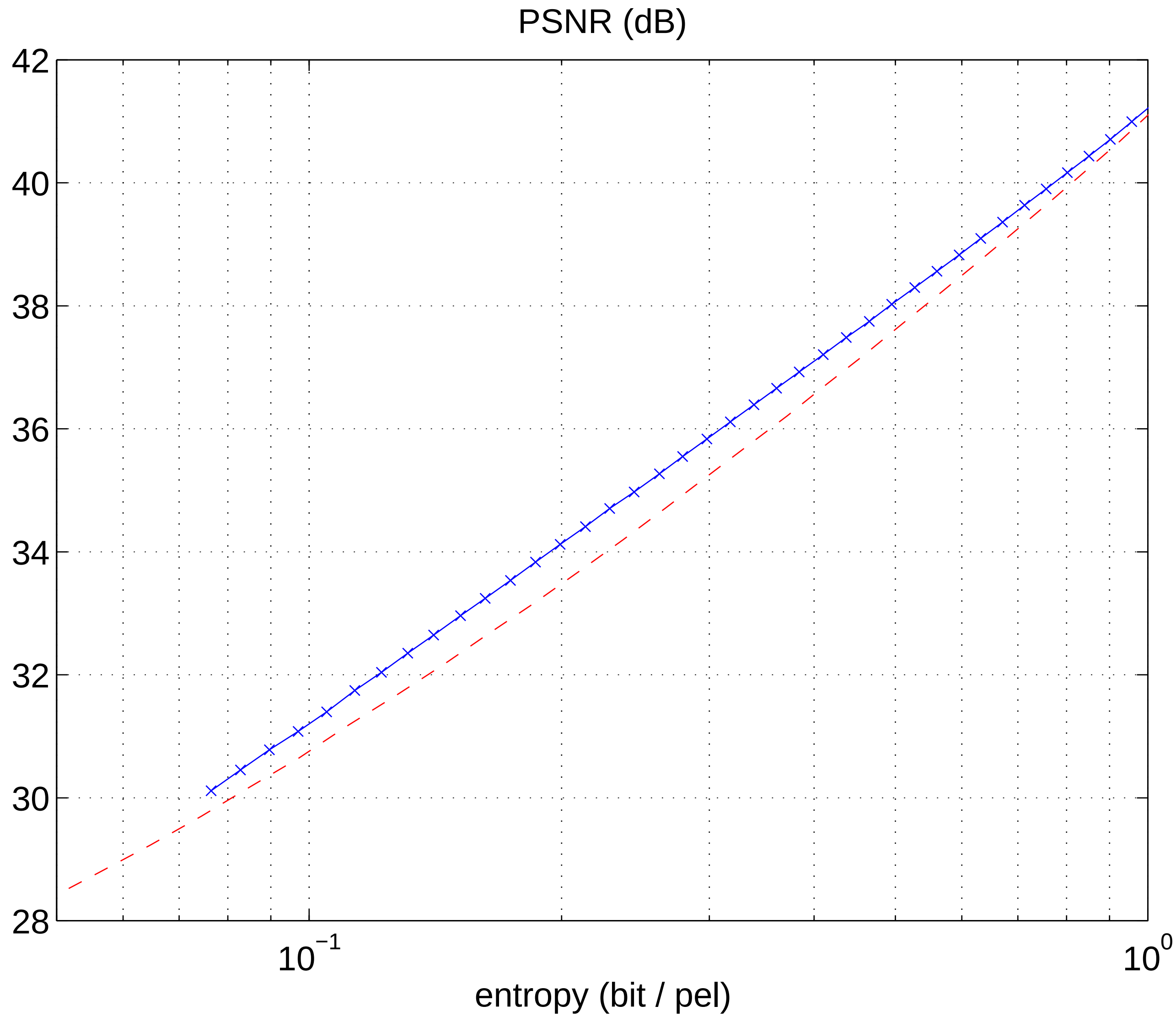


Comparison of DT CWT and DWT (centre-clipping only)



xxx Iterated DT CWT - - - DWT -o-o- non-iterated DT CWT

COMPRESSION RESULTS FOR 512×512 'LENA' IMAGE (FULLY QUANTISED)



xxx Iterated DT CWT - - - DWT

Non-redundant DWT
0.0975 bit/pel (30.66 dB PSNR)

4:1 Overcomplete DT CWT
0.0970 bit/pel (31.08 dB PSNR)

Non-redundant DWT
0.1994 bit/pel (33.47 dB)

4:1 Overcomplete DT CWT
0.1992 bit/pel (34.12 dB)

Iterative Projection – Conclusions

- Reducing the centre-clipping threshold θ_i from an initial value that is at least twice the final value, as iterations proceed, improves performance.

- Setting $k = 1.8$ and using a soft non-linearity for early iterations improves performance and convergence rate.

- Despite a redundancy of $4 : 1$, the DT CWT can achieve coding performance that is competitive with the non-redundant DWT (PSNR 0.65 dB better).

- Visibility of some coding artifacts can be reduced with the DT CWT.

- With a suitably optimised convergence strategy, computation rate should be significantly less than for matching pursuits.

Iterative Sparsity Methods for Deconvolution with Overcomplete Transforms

BAYESIAN WAVELET-BASED DECONVOLUTION

Assume an image measurement process with blur $\mathbf{H}$ and noise $\mathbf{n}$ of variance σ_n^2:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$$

Get **MAP estimate of x** by minimising

$$J(\mathbf{x}) = \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 - \sigma_n^2 \log(p(\mathbf{x}))$$

where $p(\mathbf{x})$ represents the prior expectation about the image structure.

It is often easiest to **model $p(\mathbf{x})$ in the wavelet domain**, with wavelet coefs $\mathbf{w} = \mathbf{W}\mathbf{x}$ and $\mathbf{x} = \mathbf{M}\mathbf{w}$. Then we find $\mathbf{w}$ to minimise

$$J(\mathbf{w}) = \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{M}\mathbf{w}||^2 + \frac{1}{2}\mathbf{w}^T\mathbf{A}\mathbf{w}$$

where $\mathbf{A}$ is diagonal and $A_{ii} = \sigma_n^2 / E(|w_i|^2)$, based on a **Gaussian Scale Mixture (GSM) model** for the wavelet coefs w_i, $\forall i$ in vector $\mathbf{w}$.

ADVANTAGES OF WORKING WITH WAVELET SUBBANDS

Simple steepest descent minimisation of $J(\mathbf{w})$ yields a gradient descent direction

$$\nabla_{\mathbf{w}} J(\mathbf{w}) = \mathbf{M}^T \mathbf{H}^T (\mathbf{y} - \mathbf{HMw}) - \mathbf{Aw}$$

but this blurs the differences between $\mathbf{y}$ and $\mathbf{HMw}$.

Subband emphasis can alleviate this and dramatically speed up convergence. We now minimise:

$$J(\mathbf{w}) = \tfrac{1}{2} || \mathbf{y} - \mathbf{H} \underbrace{\sum_{j \in S} \mathbf{M}_j \mathbf{w}_j}_{\mathbf{x} \,=\, \mathbf{Mw}} ||^2 + \tfrac{1}{2} \sum_{j \in S} \mathbf{w}_j^T \mathbf{A}_j \mathbf{w}_j$$

where $\mathbf{M}_j$, $\mathbf{A}_j$ and $\mathbf{w}_j$ are *subband versions* of $\mathbf{M}$, $\mathbf{A}$ and $\mathbf{w}$ in which all entries apart from those in subband j have been set to zero.

The term $||\mathbf{HMw}||^2$ makes it difficult to minimise $J(\mathbf{w})$ because of all the *cross terms* in $\mathbf{w}^T \mathbf{M}^T \mathbf{H}^T \mathbf{HMw}$; so we use the ideas of Daubechies, Defrise & De Mol (2004) **on each subband independently**, as suggested by Vonesch & Unser (2008), to minimise $\overline{J}(\mathbf{w})$, an upper bound on $J(\mathbf{w})$.

Let

$$\overline{J}_n(\mathbf{w}) = J(\mathbf{w}) + \frac{1}{2}\sum_{j\in S}\left(\alpha_j||\mathbf{W}_j\mathbf{x}^{(n)} - \mathbf{w}_j||^2 - ||\mathbf{H}\mathbf{M}_j(\mathbf{W}_j\mathbf{x}^{(n)} - \mathbf{w}_j)||^2\right)$$

where $\mathbf{x}^{(n)}$ is the estimate for $\mathbf{x}$ at iteration n. As long as **each α_j is chosen to be no less than $|\mathbf{H}(\underline{\omega})|^2$ for all frequencies $\underline{\omega}$ within the passband of subband j**, it can be shown that $\overline{J}_n(\mathbf{w}) \geq J(\mathbf{w})$, with approximate equality when $\mathbf{w}_j$ is near $\mathbf{W}_j\mathbf{x}^{(n)}$.

The proof of this requires that the transform defined by $\mathbf{W}$ and $\mathbf{M}$ is a **tight frame** and that it is **shift invariant** so that $\mathbf{M}_j\mathbf{W}_j\mathbf{H} = \mathbf{H}\mathbf{M}_j\mathbf{W}_j$ – i.e. the transfer function of each subband can commute with the blurring function.

The Q-shift DT CWT approximately satisfies these criteria. The Shannon wavelet also satisfies these, but it is not compactly supported.

By choosing α_j optimally for each subband, we can overcome the problems of slow convergence of wavelet coefficients in spectral regions where $\mathbf{H}$ has low gain.

The Resulting Algorithm:

$$
\begin{aligned}
\overline{J}_n(\mathbf{w}) &= \tfrac{1}{2}\Big(\,||\mathbf{y} - \mathbf{HMw}||^2 + \mathbf{w}^T \mathbf{A}\mathbf{w} \\
&\qquad + \sum_{j \in S} \alpha_j ||\mathbf{W}_j \mathbf{x}^{(n)} - \mathbf{w}_j||^2 - ||\mathbf{H}(\mathbf{x}^{(n)} - \mathbf{Mw})||^2 \,\Big) \\
&= C(\mathbf{x}^{(n)}, \mathbf{y}) + \sum_{j \in S} \Big((\mathbf{Hx}^{(n)} - \mathbf{y})^T \mathbf{HM}_j \mathbf{w}_j \\
&\qquad + \tfrac{1}{2}\alpha_j ||\mathbf{W}_j \mathbf{x}^{(n)} - \mathbf{w}_j||^2 + \tfrac{1}{2}\mathbf{w}_j^T \mathbf{A}_j \mathbf{w}_j \Big)
\end{aligned}
$$

where $C(\mathbf{x}^{(n)}, \mathbf{y})$ is independent of $\mathbf{w}$. This is a simple quadratic in $\mathbf{w}_j$, and its global minimum is achieved when $\partial \overline{J}_n(\mathbf{w}) / \partial \mathbf{w}_j = 0$. This gives

$$
(\alpha_j \mathbf{I} + \mathbf{A}_j)\mathbf{w}_j = \alpha_j \mathbf{W}_j \mathbf{x}^{(n)} + \mathbf{M}_j^T \mathbf{H}^T (\mathbf{y} - \mathbf{Hx}^{(n)}) \quad \forall j
$$

Hence, noting that $\mathbf{M}_j^T = \mathbf{W}_j$ for a tight frame, we get the new $\mathbf{w}_j$ and $\mathbf{x}$:

$$
\begin{aligned}
\mathbf{w}_j^{(n+1)} &= (\alpha_j \mathbf{I} + \mathbf{A}_j)^{-1} \Big(\alpha_j \mathbf{W}_j \mathbf{x}^{(n)} + \mathbf{W}_j \mathbf{H}^T (\mathbf{y} - \mathbf{Hx}^{(n)}) \Big) \quad \forall j \\
\mathbf{x}^{(n+1)} &= \mathbf{M} \sum_{j \in S} \mathbf{w}_j^{(n+1)}
\end{aligned}
$$

Updating the prior $\mathbf{A}$

Note: *In the preceding analysis, we have assumed that all coefs in* $\mathbf{w}$ *were purely real, and that complex transforms (like DT CWT) created coefs whose real and imaginary parts were separate real elements of* $\mathbf{w}$*.*
However in the following, we assume that these parts have been combined together into complex elements of $\mathbf{w}$*.*

Bayesian analysis with a Gaussian scale mixture (GSM) model gives a diagonal prior matrix $\mathbf{A}$ such that $A_{ii} = \sigma_n^2 / E(|w_i|^2)$.

In practise we use $A_{ii} = \dfrac{\sigma_n^2}{E(|w_i|^2) + \epsilon^2}$ so that

$$w_i^* \, A_{ii} \, w_i \;=\; \sigma_n^2 \, \frac{|w_i|^2}{E(|w_i|^2) + \epsilon^2} \;\approx\; \sigma_n^2 \, ||w_i||_0$$

In this way we **maximise sparsity**, where ϵ defines the approximate threshold for $|w_i|$ between being *counted* or *not counted* in $||w_i||_0$. $E(|w_i|^2)$ is updated from the squared magnitudes of the complex coefs of $\mathbf{W}\mathbf{x}^{(n)}$ at each iteration n.

We call this function the L_{02} penalty, because

- It is closer to the L_0-norm than to the L_1-norm;
- It is smooth and differentiable (like the L_2-norm) within each iteration of the algorithm.

But what are the expected wavelet variances, $E(|w_i|^2)\ \forall i$?

In practice, the estimated image is often contaminated by artifacts and noise, so the simple approach of calculating $E(|w_i|^2) = |w_i^{(n)}|^2$ direct from each complex coefficient in $\mathbf{Wx}^{(n)}$ does not work as well as we might hope.

We find we can obtain better estimates by calculating **denoised wavelet coefficients** $\widehat{w}_i^{(n)}$ and setting $E(|w_i|^2) = |\widehat{w}_i^{(n)}|^2$.

For denoising, we use the **Bayesian bi-variate shrinkage (Bay-bi-shrink)** algorithm of Sendur and Selesnick (2002), which models well the inter-scale (parent-child) dependencies of complex wavelet coefficients.

Initialisation and update strategies

- We initialise our algorithm with an under-regularised Wiener-like filter, implemented in the frequency domain:

$$\mathbf{x}^{(0)} = (\mathbf{H}^T\mathbf{H} + 10^{-3}\sigma_n^2\,\mathbf{I})^{-1}\,\mathbf{H}^T\,\mathbf{y}$$

- Diagonal regularisation matrix $\mathbf{A}$ is initialised using

$$A_{ii} = \frac{\sigma_n^2}{|\widehat{w}_i|^2 + \epsilon^2} \quad \text{where } \widehat{\mathbf{w}} = \text{denoise}(\mathbf{W}\mathbf{x}^{(0)}) \;\; \text{and } \epsilon = 0.01$$

- Optionally, $\mathbf{A}$ is updated using $\quad \widehat{\mathbf{w}} = \text{denoise}(\mathbf{W}\mathbf{x}^{(n)})$ at regular intervals in the iteration count n.

$\mathbf{y}$: Cameraman, 9×9 uniform blur + noise at 40 dB PSNR

$\mathbf{x}^{(0)}$: Initial image from under-regularised Wiener-like filter

$\mathbf{x}^{(10)}$: Iteration 10 of DT CWT with update of $\mathbf{A}$

$\mathbf{x}^{(0)}$: Initial image from under-regularised Wiener-like filter

$\mathbf{x}^{(10)}$: Iteration 10 of DT CWT with update of $\mathbf{A}$

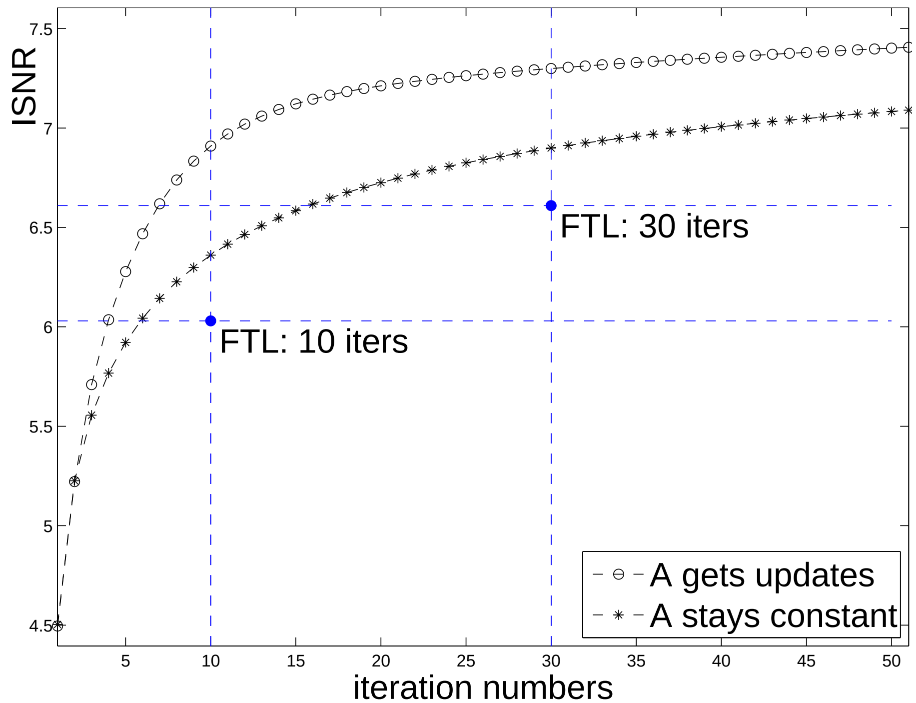
$\mathbf{x}^{(30)}$: Iteration 30 of DT CWT with update of $\mathbf{A}$

$\mathbf{x}$: Original of Cameraman

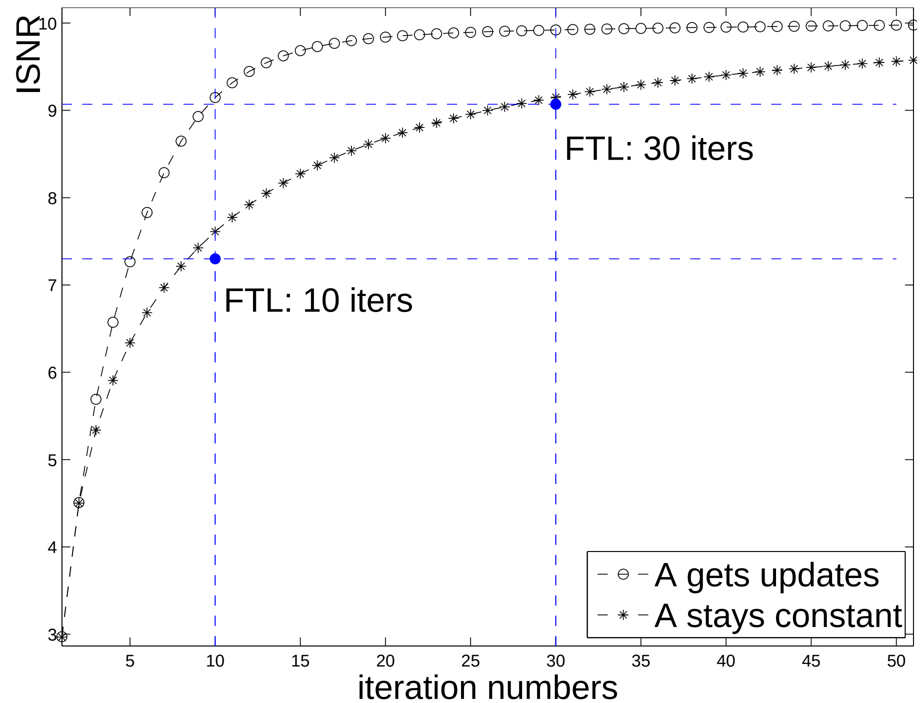
$\mathbf{x}^{(30)}$: Iteration 30 of DT CWT with update of $\mathbf{A}$

CONVERGENCE RATE COMPARISONS WITH FAST THRESHOLDED LANDWEBER ALGORITHM (Vonesch & Unser)

Improvement in SNR (dB) of Cameraman image

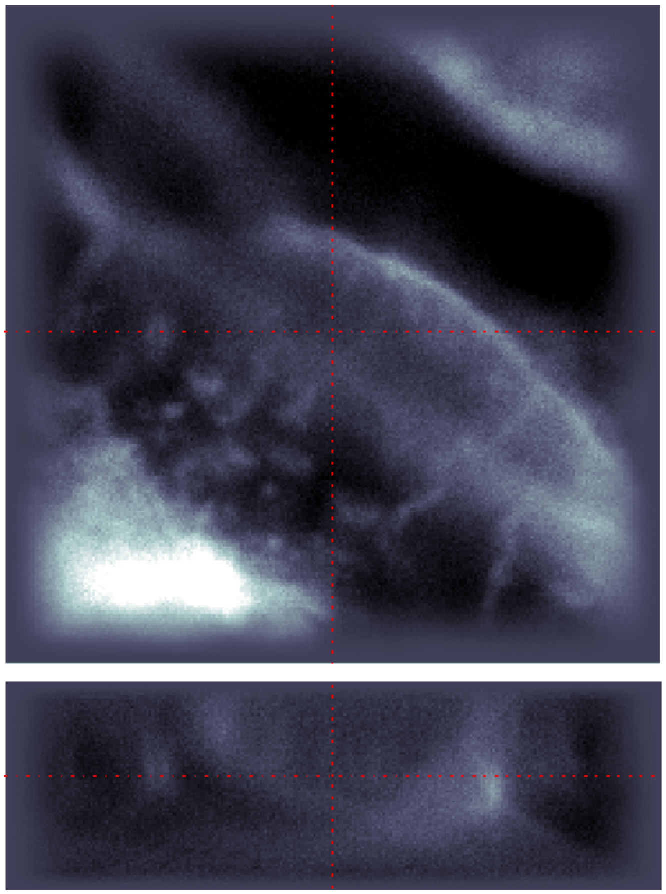


Improvement in SNR (dB) of House image

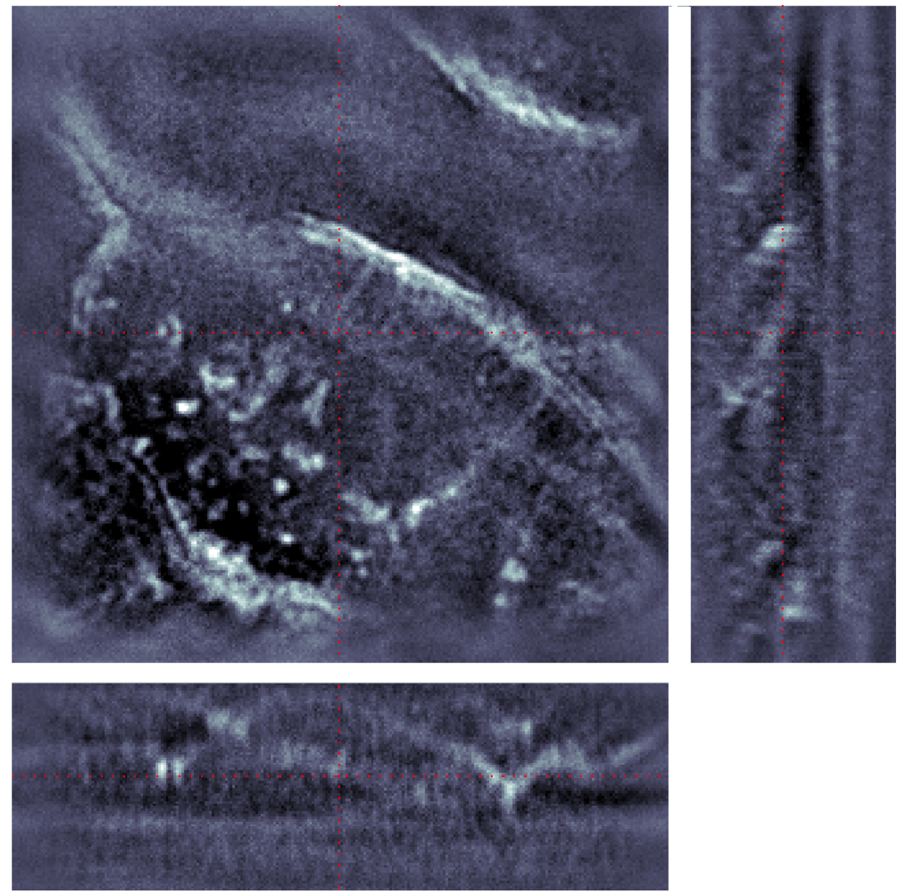


3D widefield fluorescence microscope data

$\mathbf{y}$: 3D fluorescence data with widefield imaging blur



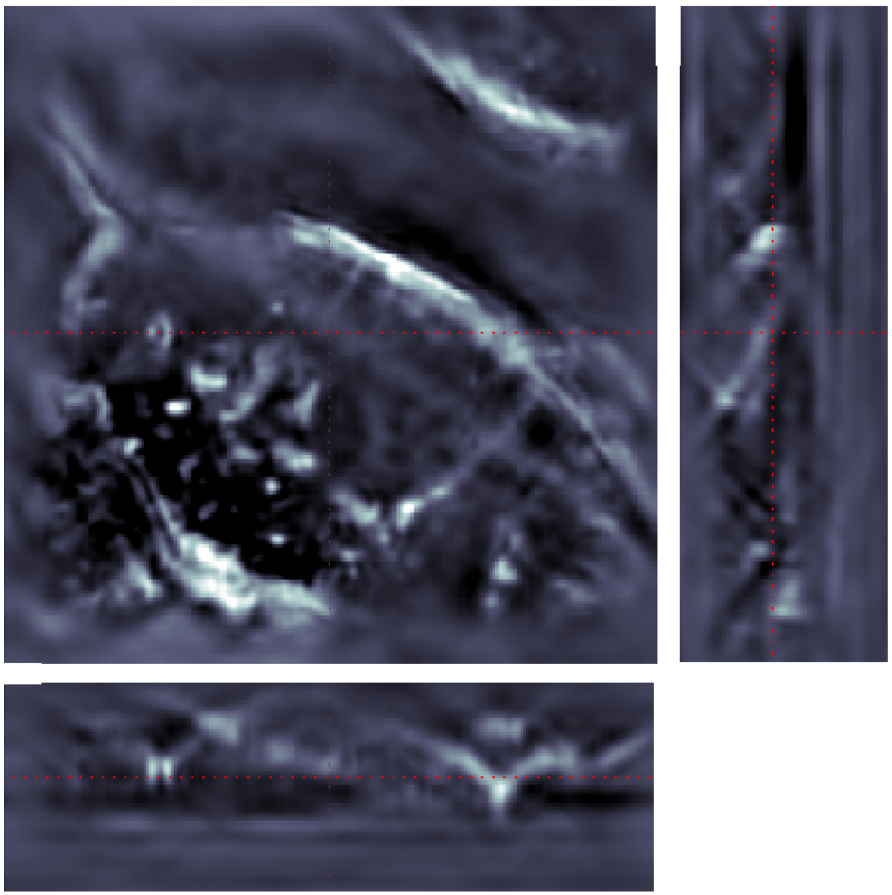
$\mathbf{x}^{(0)}$: Initial data from under-regularised Wiener-like filter



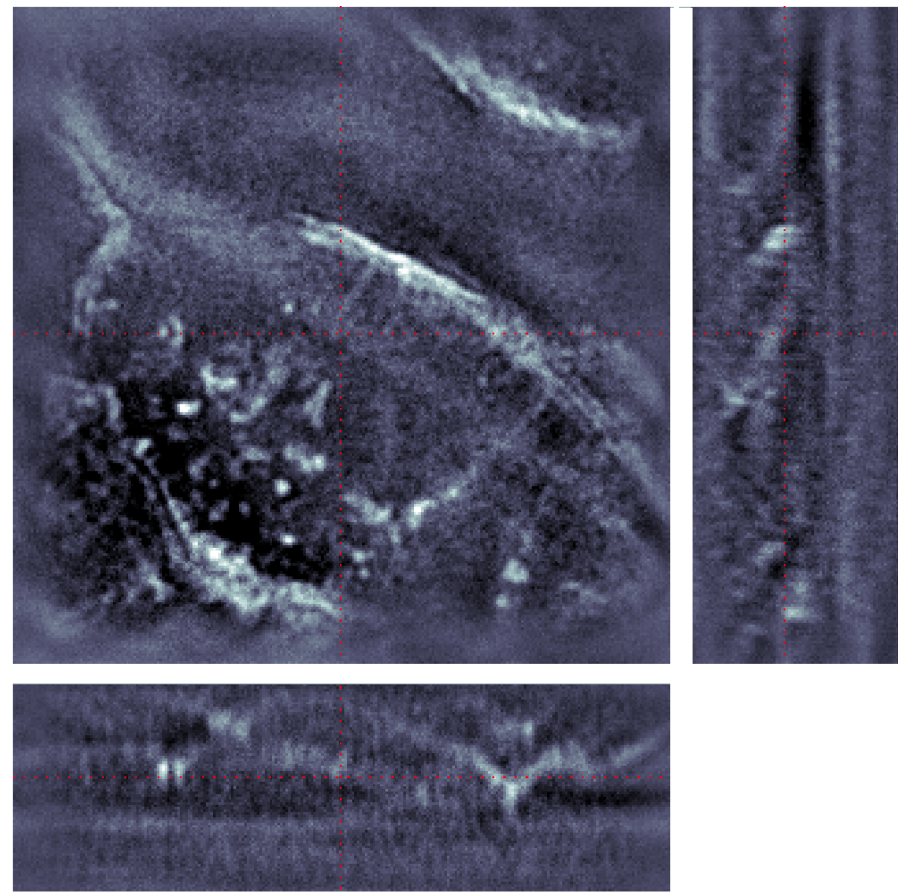
Size of 3D dataset$= 256 \times 256 \times 80 = 5.24 \,.\, 10^6$ voxels

3D widefield fluorescence microscope data

$\mathbf{x}^{(10)}$: Iteration 10 of DT CWT with update of $\mathbf{A}$



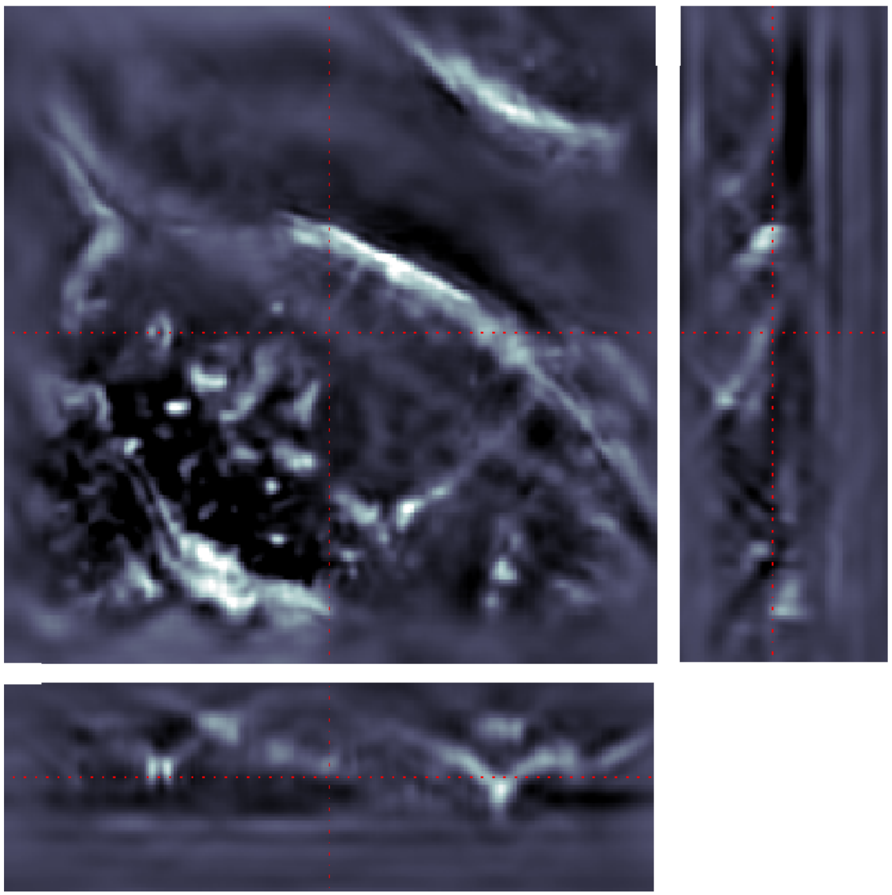
$\mathbf{x}^{(0)}$: Initial data from under-regularised Wiener-like filter



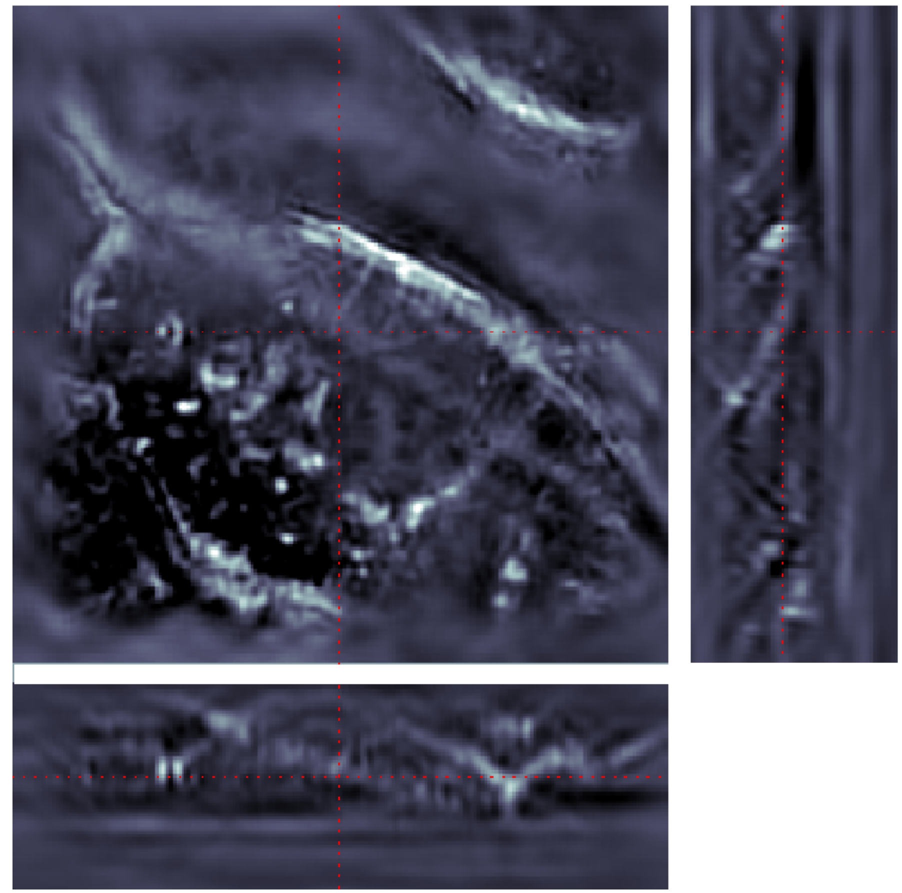
Size of 3D dataset$= 256 \times 256 \times 80 = 5.24\ .\ 10^6$ voxels

3D widefield fluorescence microscope data

$\mathbf{x}^{(10)}$: Iteration 10 of DT CWT with update of $\mathbf{A}$



$\mathbf{x}^{(30)}$: Iteration 30 of DT CWT with update of $\mathbf{A}$



Size of 3D dataset$= 256 \times 256 \times 80 = 5.24 \;.\; 10^6$ voxels

Conclusions

- We have discussed some techniques for performing both Compression and Deconvolution with overcomplete transforms.
- We have shown how sparsity helps with both of these types of large inverse problems.
- For Compression, we have demonstrated the effectiveness of iterative threshold-shrinkage methods and that there are some interesting outstanding questions regarding optimal use of soft thresholds.
- For Deconvolution, we have introduced the L_{02} penalty function and shown that Fast Thresholded Landweber (FTL) techniques may be used effectively with overcomplete transforms that possess tight-frame and shift-invariance properties, such as the DT CWT.

Papers on complex wavelets and related topics are available at:

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