ARTICLE IN PRESS

Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Signal Processing ▌(▌▌▌▌) ▌▌▌–▌▌▌

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Sparse audio representations using the MCLT

M.E. Davies[a,*], L. Daudet[b]

[a]DSP & Multimedia Group, Electronic Engineering Department, Queen Mary, University of London, Mile End Road,
London E1 4NS, UK
[b]Laboratoire d'Acoustique Musicale, Universite Paris 6, 11 rue de Lourmel 75015, Paris, France

Received 9 November 2004; received in revised form 9 May 2005

Abstract

We consider sparse representations of audio based around the modulated complex lapped transform (MCLT) and a generalized iteratively reweighted least squares algorithm which can be interpreted as a variation of expectation maximization. We compare this mildly overcomplete representation to the more traditional modified discrete cosine transform (MDCT) in terms of coding cost and explore the possibility of extending it to a dual-resolution analysis using a pair of MCLT transforms, illustrating its potential application for audio modification.

Keywords: Lapped transforms; Overcomplete dictionaries; Sparse coding

1. Introduction

Sparse signal representations are becoming increasingly popular in signal processing [1–4], independent component analysis (ICA) [5–7] and machine learning [8,9]. Typically the aim is to exploit the redundancy in an overcomplete dictionary to obtain a more compact representation of the signal. In contrast to traditional Frame theory [10], sparse decompositions are typically *generative*, concentrating on the synthesis equations.

A number of criteria for sparsity have been proposed and a variety of algorithms for solving the resulting optimization problem have been developed. In [4] we introduced an iterative reweighted least squares (IRLS) based algorithm to generate sparse modulated complex lapped transform (MCLT) synthesis coefficients. The MCLT is a 2× overcomplete subband decomposition composed of the union of 2 orthonormal bases recently introduced by Malvar [11]. The algorithm in [4] is equivalent to the regularized FOCUSS algorithm [3] and the adaptive sparseness model for regression of Figueiredo [9], tailored specifically to the structure of the MCLT.

In this paper we present a new sparsifying algorithm, initially proposed in [7], that avoids the

[*]Corresponding author. Tel.: +44 20 78827681; fax: +44 20 78827997.
E-mail addresses: michael.davies@elec.qmul.ac.uk (M.E. Davies), daudet@lam.jussieu.fr (L. Daudet).

0165-1684/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2005.05.024

expensive matrix inversion that dominates the computational cost of most previous sparsifying methods (e.g., [1,3,4]) which is replaced by a sequence of scalar *shrinkage* operations. The algorithm generalizes the IRLS framework and also has an interpretation as a generalized expectation-maximization (EM) algorithm. Furthermore the framework is applicable to a wide class of structured overcomplete dictionaries beyond the MCLT, where the dictionary can be described as the concatenation of orthonormal bases.

The rest of this paper is set out as follows. In the next section we discuss the concept of sparse approximation. We then introduce our sparse subband decomposition, based on the MCLT. Using the fact that the MCLT is the union of two orthonormal bases, we construct a new algorithm which we have coined *fast iterated re-weighted sparification* (FIRSP). In Section 4 we explore numerically the efficacy of our technique on a simple audio example where we also examine the coding costs for the sparse MCLT approximation. In Section 4.4 we extend the system, introducing a dual-resolution MCLT dictionary that can efficiently describe both transient and tonal components of an audio signal. We conclude by illustrating its potential for audio signal manipulation.

2. Sparse overcomplete approximations

Let $\Phi \in \mathbb{C}^{N\times M}$ define an overcomplete dictionary ($M>N$). Our aim is to determine an *approximate* overcomplete representation, $\Phi u = x + e$, of a signal x such that the coefficients, u, are sparse and approximation error, e, is small. Note that even when the approximation error is constrained to zero, overcompleteness provides us with the flexibility to search for a maximally sparse representation [1].

Unfortunately, there are currently a plethora of sparsity measures with little indication of their relative merits. One interesting class, that has been examined in the FOCUSS family of algorithms [3], aims to minimize the following cost function:

$$u = \arg\min_u \frac{1}{2v}||x-\Phi u||_2^2 + \lambda \sum_{k=1}^{M} |u_k|^p, \tag{1}$$

where v is the variance of e and λ is a scaling parameter for the sparsity measure $\sum_{k=1}^{M}|u_k|^p$. This measure is sometimes called the l_p pseudo-norm of u and has been shown to induce sparsity as long as $0<p\leqslant 1$. Here, by sparse we mean (following [2]) that the solution has no more than N non-zero coefficients. In practice we are looking for approximations that have $K \ll N$ non-zero coefficients.

This optimization problem also has various probabilistic interpretations [4,3,9], where the coefficient prior is modeled as a generalized Gaussian. The value of p can then be interpreted as the degree of sparsity of the prior placed on u_n. $p=1$ is equivalent to a Laplacian prior, while in the limit $p\to 0$ Eq. (1) becomes:

$$u = \arg\min_u \frac{1}{2v}||x-\Phi u||_2^2 + \sum_{k=1}^{M} \ln|u_k| \tag{2}$$

corresponding to an improper prior on u_n.[1]

The Laplacian prior ($p=1$) is also equivalent to the basis pursuit de-noising (BPDN) solution proposed by Chen et al. [1] and has the interesting property that it is unique in both guaranteeing a sparse solution (in the mild sense that $K\leqslant N$) while also guaranteeing that the cost function is convex and therefore has a unique minimum [1]. In contrast, for $p<1$, the cost function typically has a large number of local minima. While the algorithms presented in Section 3 below can equally be applied with any value of p we have so far found that the benefits of a guaranteed single minimum do not compensate for the mildness of the sparsity model. For this reason we will predominantly concentrate on the more severe model associated with $p\to 0$. In Section 3.3 we will discuss further the arguments for and against this choice of p.

2.1. The MCLT as an overcomplete dictionary

As we are interested in representing audio signals we must first select an appropriate dictionary within which to work. For example it is

[1] While the sparsity measure for $p\to 0$ does not correspond to an l_0 norm, in practice, it appears to provide a good approximation for it.

well known that audio signals can be well represented using modulated time-frequency transforms such as Gabor dictionaries [12]. In contrast, however, state-of-the-art codecs for audio signals typically use the modified discrete cosine transform (MDCT) which is a critically sampled orthonormal transform. Apart from not being overcomplete there is also a lack of explicit phase information in the MDCT transform. This introduces a problem when trying to perform a variety of linear and nonlinear signal processing tasks within the transform domain (e.g., filtering, thresholding, quantization).

Recently Malvar [11], introduced the MCLT to provide a transform with explicit phase information while being intimately linked to the MDCT. For our purposes we are most interested in the MCLT synthesis dictionary elements (‘atoms’) which for the pth frame are defined as

$$\phi_{p,k}[n] = \phi^{c}_{p,k}[n] + \mathrm{i}\phi^{s}_{p,k}[n], \tag{3}$$

where

$$\begin{aligned}\phi^{c}_{p,k}[n] &= h_p[n]\sqrt{\frac{1}{M}}\\ &\quad\times \cos\left[\left((n-a_p)+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right],\\ \phi^{s}_{p,k}[n] &= h_p[n]\sqrt{\frac{1}{M}}\\ &\quad\times \sin\left[\left((n-a_p)+\frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\frac{\pi}{M}\right],\\ h_p[n] &= -\sin\left[\left((n-a_p)+\frac{1}{2}\right)\frac{\pi}{2M}\right],\end{aligned}$$

a_p is the start of the pth frame, k is the frequency index which varies from 0 to $M-1$ and M is the frame length.

A signal $x[n]$ can be represented using this dictionary by a weighted sum of the dictionary elements, $x[n] = \sum_{p,k} u_{p,k}\phi_{p,k}[n]$, where $u_{n,k}$ are the complex synthesis coefficients. However, since we are only interested in real signals we can use the alternative reconstruction formula:

$$x[n] = 2\sum_{p,k} c_{p,k}\phi^{c}_{p,k}[n] + s_{p,k}\phi^{s}_{p,k}[n], \tag{4}$$

where $u_{p,k} = c_{p,k} + \mathrm{i}s_{p,k}$. Thus the MCLT takes the form of the union of the MDCT and the modified discrete sine transform (MDST).

It is important to emphasize these two complementary views of the MCLT.

(1) The MCLT can be viewed as a 2× overcomplete complex transform, similar to a short time Fourier transform (STFT) with 2× oversampling in the frequency domain.
(2) A second interpretation is, as an overcomplete transform that is the union of two real orthonormal bases (MDCT and MDST) where the imaginary coefficient values simply segregate those for the second orthonormal basis. This often makes algorithmic computation substantially simpler as in the reconstruction formula (4).

We will find subsequently that both viewpoints will be useful at different stages of our analysis.

2.2. Phase-invariance and sparsity

An important consideration in generating sparse MCLT approximations for audio is that the model should be approximately shift-invariant. That is: the probability of a signal should not change dramatically when the signal is translated in time. For a 2× overcomplete complex subband filterbank, shift-invariance can be approximated by imposing phase-invariance within each subband (e.g., [13,11]). This concept is similar to that of shiftability proposed by Simoncelli [14]. Since the MDCT is a real valued subband filter there is no such equivalent approximation.

A phase-invariant probability model can be imposed by selecting a prior that is only a function of the magnitude of the coefficient, $p_u(u_{p,k}) \propto f(|u_{p,k}|)$, for some function f. In contrast, if the real and imaginary components are treated independently we would be introducing a strong phase preference: i.e., preferring either sine or cosine components to arbitrarily phased signals.

3. IRLS-based schemes for sparse approximations

One approach to optimizing the cost function given in Eq. (1) is to use an IRLS based algorithm. These algorithms have attractive convergence properties when $p<1$ (see [15]). We begin by giving a brief description of the basic IRLS algorithm, along with its interpretation in terms of EM [16]. We then consider an efficient extension that allows us to exploit the orthogonal structure in the MCLT dictionary.

3.1. The basic IRLS scheme

Let $W \in \mathbb{R}^{M\times M}$ be a non-negative diagonal weighting matrix. A weighted least squares estimate for u can be obtained through matrix inversion as follows:

$$u = (vW^{-1} + \Phi^H\Phi)^{-1}\Phi^H x \tag{5}$$

solving the following problem:

$$u = \arg\min_u \frac{1}{2v}||\Phi u - x||_2^2 + \frac{1}{2}u^H W^{-1}u. \tag{6}$$

We can also use this to solve Eq. (1) by iteratively adapting the weighting matrix as a function of the previous estimate for u. For example, to minimize Eq. (2), we choose the weighting matrix at the ith iteration to be:

$$W^{(i)} = \mathrm{diag}(|u_n^{(i-1)}|^2). \tag{7}$$

As stated previously, Eq. (1) has a well defined probabilistic interpretation as an instance of the popular EM algorithm.

3.1.1. A probabilistic derivation

The interpretation of the IRLS algorithm as an EM for hierarchical Gaussian models dates back to the original paper on EM by Dempster et al. [16].

Recall our model: $\Phi u = x + e$. We will assume that the residual vector e is a set of independent zero mean Gaussian samples with variance v. The coefficients u_n will also be assumed to be independent and drawn from a sparse distribution that can be represented as a hierarchical Gaussian model: $p(u_n|w_n) = N_u\{0, w_n\}$. That is: each coefficient has its own variance w_n which in turn is drawn from a distribution $p(w_n)$. The log posterior for this model, *given* the values w_n becomes:

$$\log p(u, v|x, w) = -\frac{N}{2}\log v - \frac{||x-\Phi u||_2^2}{2v} - \frac{1}{2}u^H \mathrm{diag}(w)^{-1}u + \mathrm{const.} \tag{8}$$

Our aim is to obtain a MAP estimate for u. To do this we can apply the EM algorithm to marginalize out the coefficient variances, w_n. This requires taking the expectation of Eq. (8) with respect to $p(w|u)$. Denoting $E\{w_n|u_n\}$ by $\bar{w}_n$ the EM M-step becomes:

$$\hat{u}^{(i+1)} = (v\,\mathrm{diag}(\bar{w}^{(i)})^{-1} + \Phi^H\Phi)^{-1}\Phi^H x \tag{9}$$

which is clearly a weighted least squares update. The re-weighting procedure corresponds to the E-step and it is the choice of $p(w_n)$ that governs the nature of the re-weighting and the sparsity of the marginal distribution for u_n.

Recall that the l_p pseudo norms are equivalent to using a generalized Gaussian prior on the coefficients, u_n. That the generalized Gaussian can be constructed as a hierarchical Gaussian model was shown in [17]. Two values of p are of particular interest. Choosing $p=1$ is equivalent to using the exponential prior for w_n, $p(w_n) = \gamma/2\exp\{-w_n\gamma/2\}$, where γ is the hyperparameter that controls the scale. This results in the following E-step:

$$\bar{w}_n = E\{w_n|u_n\} = \frac{1}{\gamma}|u_n| \tag{10}$$

which is equivalent to assuming Laplacian priors on u_n giving the BPDN cost function.

Alternatively Figueiredo [9] has proposed the use of a non-informative prior: $p(w_n) \propto w_n^{-1}$ on the variance parameters, which is equivalent to letting $p \to 0$. This has two key advantages. First there is no additional hyper-parameter to estimate and second it results in a much more severe reweighting matrix:

$$\bar{w}_n = E\{w_n|u_n\} = |u_n|^2 \tag{11}$$

(or $\bar{w}_n = 2|u_n|^2$ if u_n is complex, as in the MCLT).

If desired the EM framework also provides a means of estimating the noise variance, v, within the maximization step:

$$\hat{v}^{(i)} = \frac{1}{N}||x - \Phi\hat{u}^{(i)}||_2^2. \tag{12}$$

Subsequently, however, we will fix the value of v to control the desired level of approximation.

Finally, we note that since the IRLS can be formulated as an EM algorithm we automatically know that it will exhibit the usual monotonic convergence property of EM [16].

3.2. A generalized IRLS scheme

The main drawback with the IRLS solution is that it requires an $M \times M$ matrix inversion to solve the weighted least squares equations at every iterate. This makes it prohibitively expensive for practical use. A similar problem occurs in the quadratic programming solutions to BPDN. Chen et al. [1] proposed using a conjugate gradient algorithm to estimate this inverse, however it turns out that full matrix inversion is both unnecessary and of no great advantage to an appropriate *partial* solution to the weighted least squares problem.

In EM theory it is well known that the maximization step can be replaced by any operation that guarantees to increase the likelihood function (decreases the cost function). One such generalization is the expectation conditional maximization (ECM) algorithm [18]. This replaces the maximization step by a sequence of conditional maximization (CM) steps that act on partitioned subsets of the parameter space. The nature of the EM theory means that there is a great deal of flexibility in the ordering of the various CM steps and the corresponding E step as discussed below.

For the MCLT dictionary the natural partition to consider is the splitting of the dictionary into the two orthonormal bases: $\Phi = (\Phi_c, \Phi_s)$. Φ_c and Φ_s thus correspond to the inverse MDCT and inverse MDST, respectively. We will see that the computational advantage of such a splitting is that it avoids the expensive matrix inverse calculation.

Consider the weighted conditional least squares problem where we freeze the values of s and optimize for c:

$$c = (vW^{-1} + \Phi_c^{\mathrm{T}}\Phi_c)^{-1}\Phi_c^{\mathrm{T}}(x - \Phi_s s). \tag{13}$$

Since Φ_c is orthonormal $\Phi_c^{\mathrm{T}}\Phi_c = I$ and therefore the matrix inversion reduces to a diagonal *shrinkage* operator [12]:

$$c_n = \left(\frac{w_n}{v + w_n}\right) \times [\Phi_c^{\mathrm{T}}(x - \Phi_s s)]_n \tag{14}$$

where $[\cdot]_n$ refers to the nth element of the vector and, with a slight abuse of notation, we are now using a one-dimensional indexing of the MDCT coefficients. An equivalent expression can be calculated for the CM of s given c.

The iteration is finally completed by determining the re-weighting calculation which has the same form as for the basic IRLS algorithm.

In the implementation used in the examples below the re-weighting step is performed after each CM. One full iteration is therefore composed of two CM steps and two re-weighting steps. Examining this iteration we see that the computational cost is dominated by the need to map from one transform domain to another (the cost of the shrinkage and weight calculations are trivial by comparison). Thus overall one iteration takes approximately 4× the computation for a single MDCT, which is orders of magnitude faster than the basic IRLS or alternative strategies such as BPDN.

Despite the fact that we are no longer solving the full weighted least squares, the convergence of the algorithm is not drastically reduced (see Section 4 below). Similar observations for the ECM algorithm have been made in other applications [18].

It is worth noting that there is a great deal of flexibility in the order in which we perform the E and CM steps. For example we could perform repeated M-steps until convergence followed by an E-step. This would have the advantage that the asymptotic mapping could be derived analytically and be similar to the work of Sardy et al. [19]. However, it does not reduce the computational complexity.

Finally, the FIRSP algorithm is in theory applicable to any dictionary that is the union of orthonormal bases (for example see Section 4.4).

Furthermore the generalized IRLS approach can also be extended to other classes of dictionary (for details see [15]).

3.3. Alternative strategies for sparse approximation

Before we illustrate the performance of the proposed algorithm we consider two alternative strategies for generating sparse approximations, namely: orthogonal matching pursuit (OMP) and BPDN. These two are of particular interest due to a number of recent theoretical results that link the OMP and BPDN solutions and the L_0 maximally sparse solution (the one with the fewest non-zero coefficients). That is: when the dictionary is sufficiently incoherent the unique OMP and BPDN solutions coincide with the maximally sparse solution (see for example [20,21]).

Unfortunately, there are two drawbacks. First the computation time for OMP and BPDN can be prohibitatively slow. Even when BPDN is implemented using the above scheme it proves to be too slow to produce competitive results.

The second, possibly more serious drawback is that such guarantees only hold when the dictionary being used is sufficiently incoherent:

$$\mu := \max_{i \neq j} |\langle \phi_i, \phi_j \rangle| \ll 1$$

and the signals being analyzed have a sufficiently sparse representation. While it has been shown that there exist large overcomplete dictionaries that are very incoherent there is no guarantee that such a dictionary exists in which the signals of interest are sparse. Indeed the dictionary choice must be data driven to ensure that we are likely to be able to obtain a sparse representation. For audio, Gabor-like dictionaries (such as the MCLT used here) seem well-suited to this task, at least for the tonal components within the signal. Unfortunately, the MCLT transform has a coherence $\mu \approx 0.5$. The most correlated atoms within the MCLT dictionary occur between neighboring frequency bins within the same synthesis frame, as illustrated in Fig. 1. This implies that here the OMP and BPDN approaches may well not find the maximally sparse solution.

Instead we choose a fast converging algorithm and accept that we may only find a local minimum. In practice the local minimum that is found appears to always be a good one (and in our experience better than the BPDN solution in terms of number of non-zero coefficients as a function of signal-to-noise ratio (SNR)).

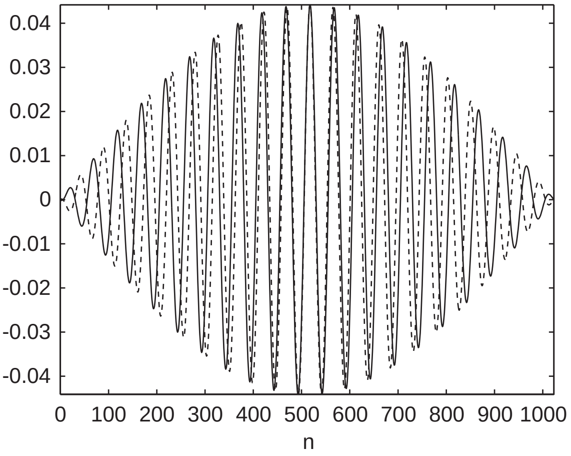

Fig. 1. The real part of the $k = 20$ atom (solid) is plotted along side the imaginary part of the $k = 21$ atom (dotted) within the same analysis window. There is strong coherence between the neighboring atoms as these functions become in phase at the center of the frame.

4. Numerical experiments

To illustrate the FIRSP algorithm and the power of MCLT based overcomplete representations of audio we now present some numerical experiments. We begin by showing the speed with which the algorithm can generate a sparse solution to a real world (44.1 kHz sampling rate) audio signal. We then explore the potential benefits that may be realized from using such a sparse representation in audio coding. Finally, we show how the basic MCLT dictionary can be extended to a dual-resolution time-frequency representation while still being amenable to processing with the FIRSP algorithm.

4.1. A simple audio example

We first apply the algorithm to a short extract (approx. 6 s) from a guitar solo. The audio was

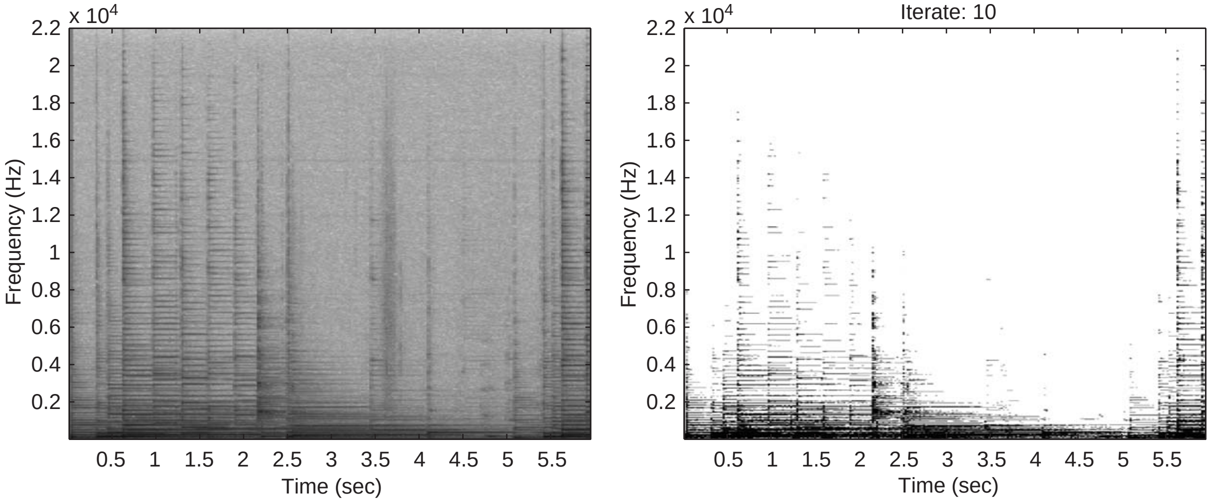

Fig. 2. MCLT "spectrogram" of the guitar data (left) and the generative sparse MCLT approximation.

sampled at 44.1 kHz and we used an MCLT with a frame size of 1024. Fig. 2 (left) shows the MCLT "spectrogram" for the audio signal. The signal was then processed using a fixed $v = 10^{-5}$ and 10 iterates of the FIRSP algorithm. In contrast to the initial redundant basis only 6% of the complex coefficients remained non-zero. The resulting approximation had a SNR of: 38 dB. The generative MCLT "spectrogram" for the sparse coefficients is also shown in Fig. 2 (right).

The evolution of the algorithm is best seen by plotting the size of the coefficients sorted in order of magnitude for each iterate, as in Fig. 3. Clearly most of the coefficients shrink to zero in a small number of iterats.

4.2. Coding costs for the sparse MCLT

We next make a preliminary examination of the coding costs associated with the sparse MCLT in comparison with the traditional MDCT transform. We will concentrate on relatively simple coding structures and emphasize that we are not arguing that the following coding strategies are competitive with state-of-the-art audio coding.

As with traditional transform coding, a high degree of sparsity in the coefficients means that a substantial part of the coding cost can be taken up coding the significance map (the map identifying which coefficients are non-zero) [12]. We therefore consider the coding rate, R in two parts:

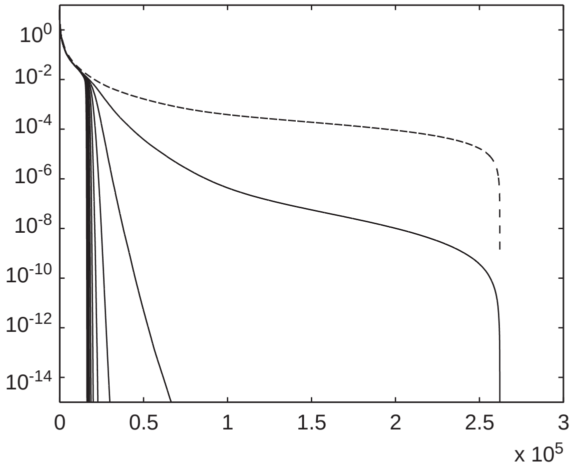

Fig. 3. A sorted plot of MCLT coefficient amplitude for each iteration of the FIRSP algorithm (solid lines—iterations increasing from right to left) and the magnitude of the basic MDCT coefficients are also shown (dashed).

$$R = R_{\text{sig\_map}} + R_{\text{coef}},$$

where $R_{\text{sig\_map}}$ measures the total bit budget required to code the significance map and R_{coef} measures the number of bits required to code the quantized non-zero coefficients. We begin by considering the cost of coding the significance map.

4.2.1. Coding the significance map

If we treat the significance of each coefficient as independent, we can estimate the rate, $R_{\text{sig\_map}}$, as the sample entropy, $H_{\text{sig\_map}}$, of the significance map:

$$R = H_{\text{sig\_map}} + R_{\text{coef}},$$

where $H_{\text{sig\_map}} = -(p_s \log p_s + (1 - p_s)\log(1 - p_s))$ and p_s is the probability of a coefficient being non-zero.

To measure $H_{\text{sig\_map}}$ for the sparse MCLT approximation we used the same audio sample examined in the last section. Again the frame size was set to 1024 but this time 50 iterations of the FIRSP operator were applied to guarantee absolute convergence. We calculated the SNR for a range of sparse approximations using different v and plotted these against the coding cost for the significance map. For comparison we also included the SNRs for the best K-coefficient MDCT approximation for the signal over the same sparsity range. The graphs in Fig. 4 show that there is approximately a 5 dB gain in using the sparse MCLT approximation over the MDCT for a wide range of bit rates. Note that both the MDCT and the MCLT have the same size significance maps and are thus directly comparable.

Further improvements in coding can be obtained by incorporating structure within the significance map into the coding strategy. Looking at Fig. 2 we see that the significance map exhibits strong persistence in time for each subband (other structure due to onsets and harmonicity is not considered here). The MDCT also exhibits this type of structure but to a lesser extent.

A relatively simple way to code this structure is to use run length encoding along each subband followed by entropy coding. This has a dramatic effect on the coding cost, as shown in Fig. 4. Here it can be seen that both the MDCT and the sparse MCLT gain substantially from run-length encoding with a slightly bigger improvement (in percentage terms) for the MCLT approximation.

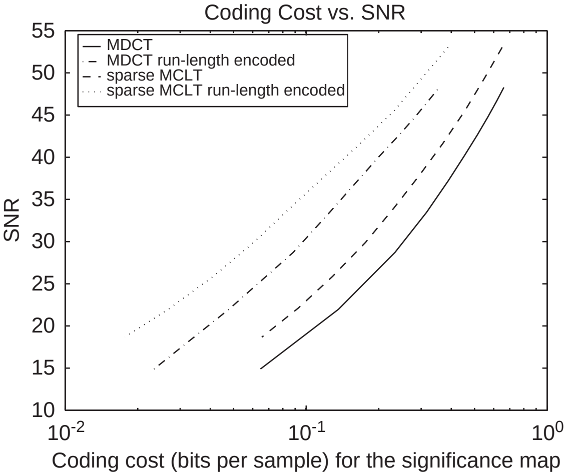

Fig. 4. A plot of signal-to-noise ratio against the significance map coding cost for: the MDCT with independent coding (solid); the sparse MCLT with independent coding (dashed); the MDCT with run-length encoding (dot-dashed); and the sparse MCLT with run-length encoding (dotted).

4.3. Coding the non-zero coefficients

So far, of course, we have ignored the coding cost of the quantized non-zero coefficients values, R_{coef}. As above, we consider a relatively simple coding strategy that treats each coefficient independently. We can then estimate the coding cost to be:

$$R = R_{\text{sig\_map}} + p_s H_{\text{coef}},$$

where H_{coef} is the sample entropy of the non-zero quantized coefficients values (measured per *significant* coefficient). To do this we now need to introduce quantization schemes for the two methods. For the MDCT we have used a uniform quantizer with a double-sized zero bin [12]. To construct a similar quantizer for the complex MCLT coefficients we chose to use an unconstrained polar quantizer (UPQ) [22]. The coefficient amplitude is the same as the MDCT quantizer. The phase components is then uniformly quantized but with the number of phase quantization bins, n_θ, being dependent on the amplitude value such that: $n_\theta(k) = 6(k + \frac{1}{2})$ for the kth amplitude region (with the exception of the zero bin, $k = 1$, where there is no phase). This UPQ is designed to space the regions approximately uniformly to enable subsequent efficient use of entropy coding. When calculating the sample entropy of the MCLT, to avoid the problem of limited data, we calculated the sample entropy for the coefficient amplitudes only and then assigned a cost of $\log_2 n_\theta(k)$ bits for the phase.

For the MCLT we also need to select an appropriate quantization resolution for a given approximation level v. For this we use the following informal argument. The signal x is fully represented by the coefficients, u, and the residual, e. A possible coding strategy is to code both separately to the same resolution. However, while u is expected to be sparse and therefore provide good energy compaction, the residual, e is assumed Gaussian and is therefore a less efficient representation. A natural choice of quantization resolution is to select a level such that with high probability the residual term is coded as zero (thereby requiring zero bits). Interestingly this is the same requirement that has been proposed for threshold selection in signal de-noising [12] where it can be shown that setting $T = \sqrt{2v\log_e N}$ should with high probability be just above the level of the noise. We adopt this value here, setting the amplitude bin size to T. We also note that experimentally this does indeed appear to be close to optimal choice.

The results for the total coding cost are presented in Fig. 5. While there is still a small amount of coding gain for the MCLT at very low bit rates (not of general interest for practical audio, due to poor sound quality), in general the MCLT performs slightly worse than the MDCT. To see why this is we look in detail at the coding cost for SNR $\approx 27\,\mathrm{dB}$ ($v = 10^{-4}$) where the rate-distortion performance is similar for both methods. The breakdown of the coefficient coding cost is displayed in Table 1. From the table we can see that the cost of coding the amplitude in either case is virtually identical. It is therefore the additional phase cost that negates the sparsity coding gain for the MCLT.

While our current results show no big coding gain for the sparse MCLT it still looks a competitive representation from these initial findings. Furthermore, for the MCLT, we should expect there to be more exploitable structure within the coefficient values themselves. For example when a subband is occupied by a single partial from a note, the temporal phase within that subband will be highly predictable (cf. the phase vocoder). Similarly we might expect that the amplitude will vary smoothly in time along the note. It is more difficult to see how this structure could be exploited within the MDCT transform where, due to the lack of explicit phase information, there will be a complicated fluctuation in coefficient values within a subband. The MCLT structure also makes the inclusion of perceptually weighted cost functions substantially simpler.

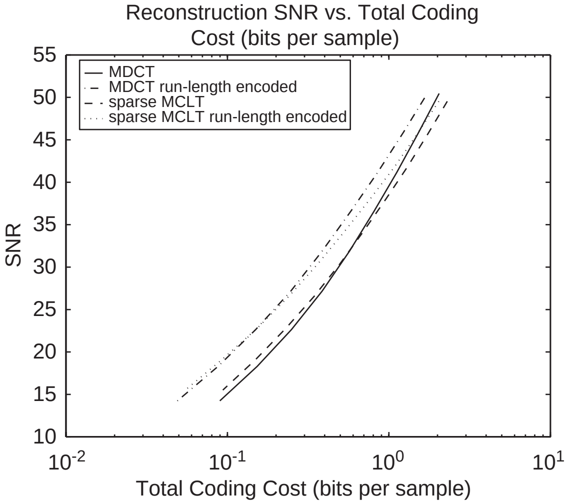

Fig. 5. A plot of signal-to-noise ratio against the total coding cost for: the MDCT (solid) and the sparse MCLT (dashed) with independent significance map coding; and the MDCT (dot-dashed) and the sparse MCLT (dotted) with run-length encoding for the significance map.

Table 1
Coding cost breakdown for non-zero coefficients (entropies in bits per significant coefficient)

	MDCT	MCLT
Number of non-zero coefficients	9353	5915
Total number of coefficients	262144	262144
Amplitude entropy	3.59	3.77
Sign/phase entropy	1.00	4.60
Coefficient entropy	4.59	8.37
Total cost per sample, R_{coef}	0.164	0.189

4.4. A dual-resolution time-frequency approximation

Sparse overcomplete time frequency representations also have a great potential in providing access to higher level information about an audio signal, such as distinguishing between steady state tones and note onsets. This in turn can be extremely useful for signal processing applications such as audio modification, source separation, note detection/recognition, and automatic music transcription. Here we show that the framework developed in this paper easily extends to allow an additive dual-resolution signal approximation.

The MCLT dictionary can easily be extended to include multi-resolution representations, at the cost of a larger dictionary, by taking the union of multiple MCLT dictionaries with differing frame sizes. Here we consider two distinct resolutions: an MCLT with a frame size of 2048 samples (approximately 46 ms) and a frame size of 256 samples (approximately 6 ms). Since we still have a union of orthonormal bases we are able to apply the same fast iterative sparsification algorithm presented in Section 3.2. Unfortunately a naive implementation of this can result in very slow convergence. One solution appears to be a judicious choice of initial condition. If we initialize the coefficients by sharing the signal energy equally between all 4 bases then convergence is painfully slow (several hundred iterates!). This is because the signal induces large coefficients in all bases. The transfer of energy is then achieved by small repeated shrinkage operations. We found that a better initialization was to begin with only the long frame MCLT representation and apply a couple of iterates of the shrinkage operator. The short frame MCLT bases were then included and used initially to model the residual of the long frame approximation. This has the effect of initializing the second set of coefficients to fit the signal that is most poorly represented by the sparse long frame MCLT (with a similar flavor to the hybrid coding scheme used in [23]). The resulting algorithm appeared to converge in 30–50 iterates. An alternative approach, proposed in [1], that is not explored here would be to initialize the coefficients with the best basis algorithm, the matching pursuit (see e.g., [10]) or any other (fast) approximate sparse representation.

To demonstrate our dual-resolution approximation we applied the algorithm to part of an MPEG standard test signal. This is a particularly good signal for our purposes since it contains strong ringing tones as well as sharp transients. Fig. 6 shows plots of: the original signal, the tonal component (formed from the long frame MCLT), the transient component (formed from the short frame MCLT) and the residual. The approximation parameter, v, was set at 10^{-5} and the algorithm was run for 50 iterates. We can see that the addition of the short frame MCLT has allowed us to not only approximate the transients of the signal much better, but more importantly, it has provided us with a separation of the transient components which are localized in time but not frequency and steady state components which are localized in frequency but not time, as illustrated in Fig. 7. This representation, although not perfect, simultaneously provides us with an excellent time and frequency resolution for the signal transient and tonal components. We now show how the decomposition can subsequently be used for audio modification.

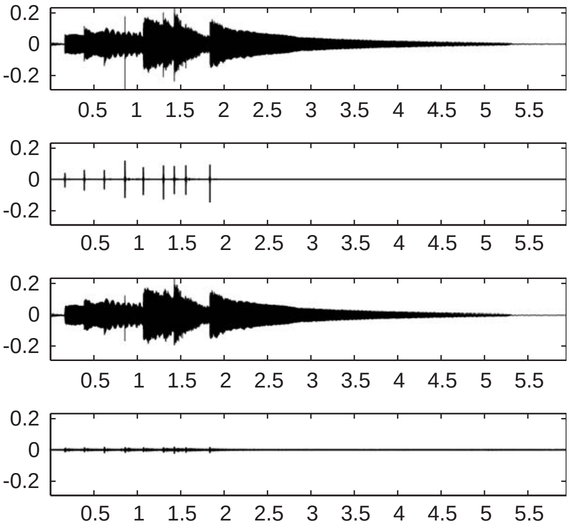

Fig. 6. Time domain plots of: (a) the original signal; (b) the approximated transient component; (c) the approximated tonal component; and (d) the residual error.

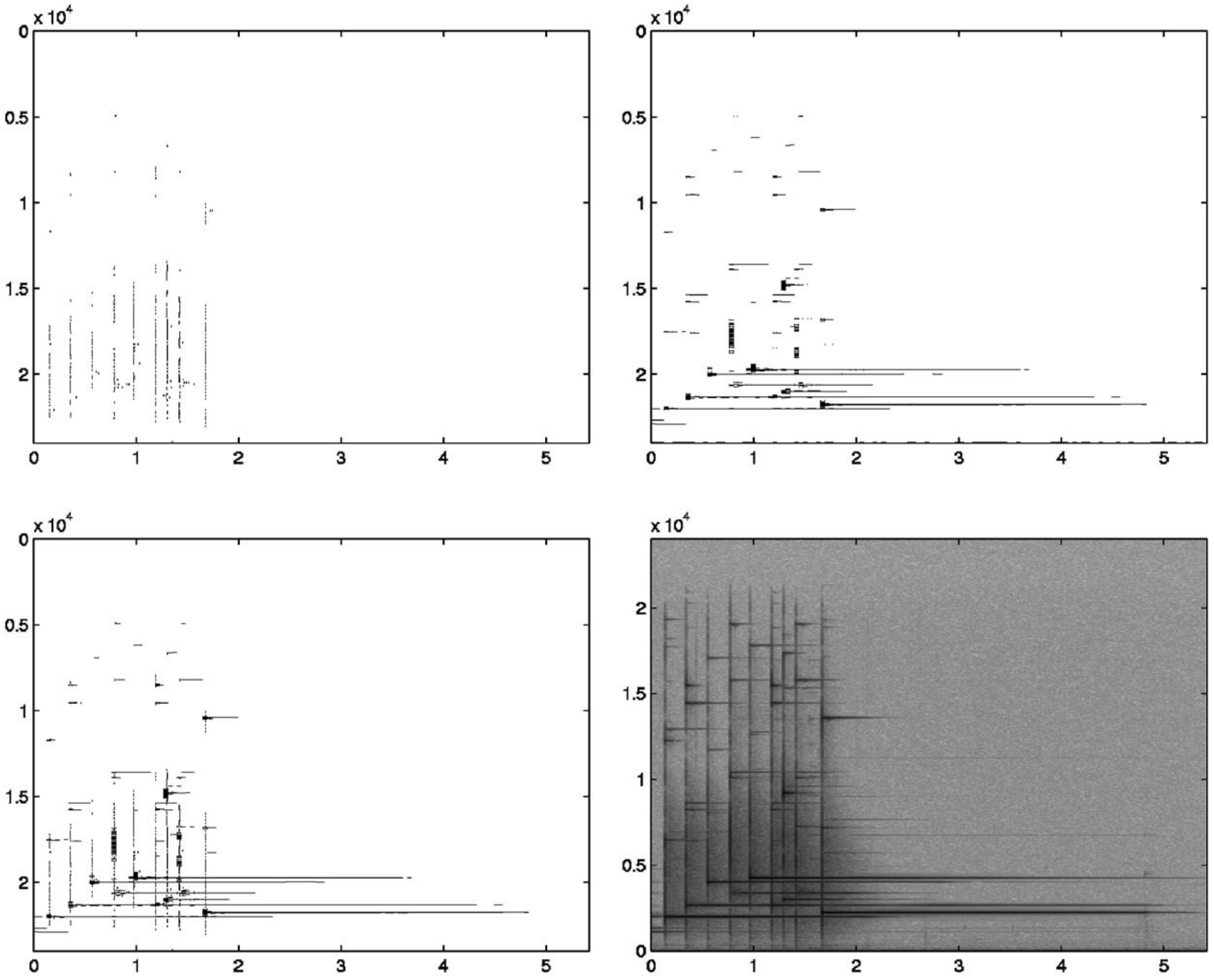

Fig. 7. The significance map for the transient components (top left); the significance map for the tonal components (top right); the combined significance map (bottom left); and, for comparison, the spectrogram for the original signal (bottom right).

4.4.1. Note extraction

A difficult task in such an audio signal is the extraction of a single note. If we were using a fixed resolution time frequency approximation (irrespective of sparsity) then without additional high level information the extraction of a single note would have to be done using some form of time-frequency mask. However, this will introduce distortions whenever notes overlap in the time frequency domain. For our signal this would be most acute where the tones intersect with the transients.

In contrast, because we have a fully additive representation that maps the transient and the tonal components to different spaces, we can extract notes even where transient and tonal components intersect.

To demonstrate this we manually grouped TF elements from both the transient and steady state dictionaries associated with the seventh note in the signal. To extract the note we simply zero these coefficients (applying two masks separately to the short and long frame coefficients. Fig. 8 shows plots of (a) the signal with the note removed and (b) the note itself. The resulting audio sounded as though the note had never been present. That we have not compromised overlapping notes can be seen in Fig. 9 where the spectrograms of the sparse approximation with and without the seventh note are shown. There is clearly huge potential for extending this to more complex modifications. For example we could easily move the note position (to possibly correct a mistimed note) or alter its sound before replacing it.

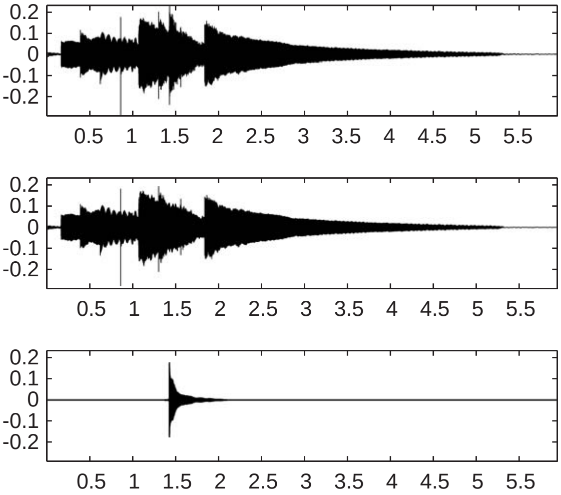

Fig. 8. Time domain plots of: (a) the sparse reconstruction; (b) the reconstruction with the seventh note removed; and (c) the removed note.

5. Conclusions

In this paper we have introduced an efficient means of generating a sparse approximation for audio signals in the form of an overcomplete subband representation. The method is sufficiently fast to enable the processing of full CD quality (44,100 samples per second) audio data, without downsampling. Indeed the proposed method is orders of magnitude faster than competitive techniques such as BPDN.

We further examined the potential of this representation as the basis of a sparse coding strategy. The high degree of sparsity means that there are substantial savings in encoding the significance map for the signal. However, the full coding cost is a function of both the significance map and the cost of coding the values of the significant coefficients. While we have made some preliminary observations in this direction, to make a full and fair comparison with state-of-the-art audio coders we would need to develop a complete coding scheme, which is beyond the scope of this paper. The approach advocated here tentatively provides a structure that lies in-between traditional transform/subband coding, such as MP3, and low rate parametric codes, such as the MPEG HILN coder, with the possibility of gaining benefits from both approaches.

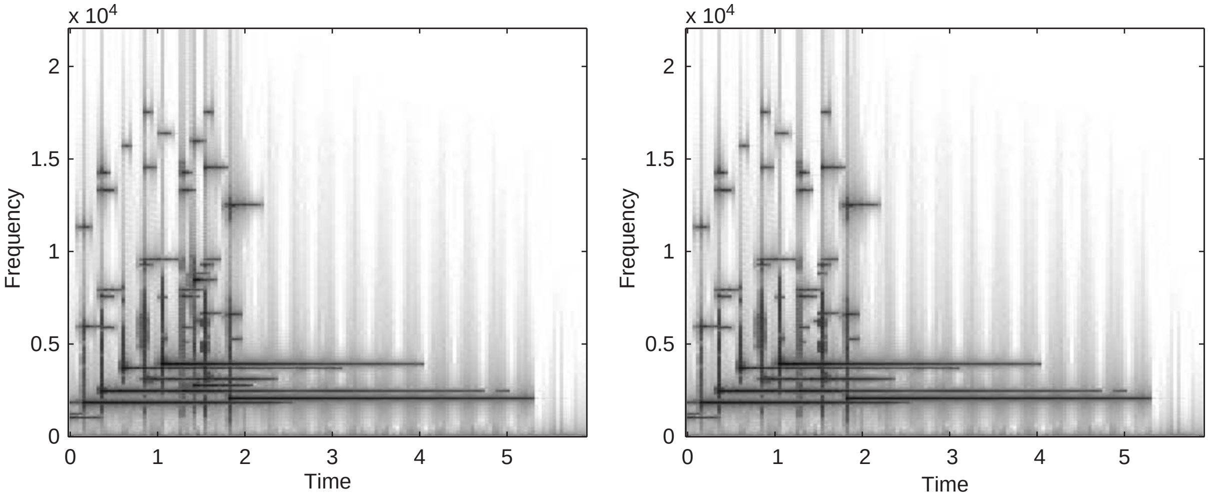

Fig. 9. The spectrograms for: the sparse approximation (top); and the same signal with the seventh note removed (bottom).

Finally we have shown that the generalized IRLS framework is very flexible and, in particular, can be extended to include multi-resolution time-frequency approximations. These could be used to further improve the coding of audio signals or, as demonstrated here, be used in signal separation and modification applications.

Our current research is focusing on a number of open questions and problems that have arisen from this work and we end by mentioning a number of these that we feel are of particular interest to the field.

How overcomplete should a dictionary be? Currently in the work on overcomplete representations there has been little said about how overcomplete the dictionary should be $(2\times ?\,10\times ?\,100\times ?\ldots)$.

What are good measures of quality for a dictionary? A fair amount of attention has been paid to the incoherence of a dictionary and how it effects the complexity of determining sparse representations. However, we believe that there is a need for a finer tool to determine the performance of a given dictionary. In particular, such a measure should also probably be signal dependent (see for example [24]).

Extensions to other signal types. We have recently also considered the use of generalized IRLS algorithms to images [15]. Here, the equivalent dictionary to the MCLT is Kingsbury's dual tree complex wavelet transform, however other dictionaries might prove more appropriate.

Extensions to structured priors. We saw in Section 4.2.1, that even more parimonious representations can be obtained by exploiting the structure of the significance map. So far this structure has been treated separately to the generation of the sparse approximation. However, it would be interesting to develop algorithms that simultaneously generated structured and sparse approximations.

Acknowledgment

The authors would like to thank the Centre for Digital Music at Queen Mary for many useful discussions and Pierre Vandegheynst for making us aware of a number of useful references. L. D. would also like to acknowledge the support of the French Ministry of Research and Technology, under contract ACI "Jeunes Chercheuses et Jeunes Chercheurs" number JC 9034".

References

[1] S. Chen, D.L. Donoho, Atomic decomposition by basis pursuit, SIAM J. Sci. Comput. 20 (1) (1999) 33–61.

[2] I.F. Gorodnitsky, B.D. Rao, Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm, IEEE Trans. Signal Process. 45 (3) (1997).

[3] B.D. Rao, K. Engan, S.F. Cotter, J. Palmer, K. Kreutz-Delgado, Subset Selection in Noise Based on Diversity Measure Minimization, IEEE Trans. Signal Process. 51 (3) (2003) 760–770.

[4] M.E. Davies, L. Daudet, Sparsifying subband decompositions, Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, October 2003.

[5] M. Zibulevsky, B.A. Pearlmutter, Blind separation of sources with sparse representations in a given signal dictionary, Neural Computation 13 (4) (2001) 863–882.

[6] M.S. Lewicki, T.J. Sejnowski, Learning overcomplete representations, Neural Computation 12 (2000) 337–365.

[7] M.E. Davies, L. Daudet, Fast sparse subband decomposition using FIRSP, Proceedings of the EUSIPCO 04, 2004.

[8] M.E. Tipping, Sparse Bayesian learning and the relevance vector machine, J. Mach. Learning 1 (2001) 211–244.

[9] M. Figueiredo, Adaptive sparseness using Jeffreys prior, Neural Information Processing Systems (NIPS), MIT Press, Cambridge, MA, 2001.

[10] S. Mallat, Z. Zhang, Matching pursuit with time-frequency dictionaries, IEEE Trans. Signal Process. 41 (1993) 3397–3415.

[11] H. Malvar, A modulated complex lapped transform and its applications to audio processing, ICASSP'99, 1999.

[12] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, New York, 1999.

[13] R.W. Young, N. Kingsbury, Frequency domain motion estimation using a complex lapped transform, IEEE Trans. Image Process. 2 (1993) 2–17.

[14] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, D.J. Heeger, Shiftable multi-scale transforms, IEEE Trans. Inform. Theory 38 (2) (1992) 587–607.

[15] M.E. Davies, T. Blumensath, L. Daudet, Generalized IRLS schemes for sparse signal approximation, 2004, in preparation.

[16] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Roy. Statist. Soc. Ser. B 39 (1) (1977) 1–38.

[17] M. West, On scale mixtures on normal distributions, Biometrika 74 (3) (1987) 646–648.

[18] G.J. McLachlin, T. Krishnan, The EM Algorithm and Extensions, Wiley Series in Probability and Statistics, 1997.

[19] S. Sardy, A.G. Bruce, P. Tseng, Block coordinate relaxation methods for nonparametric wavelet denoising, Comput. Graph. Statist. 9 (2) (2000).

[20] J.A. Tropp, Greed is good: algorithmic results for sparse approximation, IEEE Trans. Inform. Theory 50 (10) (2004) 2231–2242.

[21] J.A. Tropp, Just relax: convex programming methods for subset selection and sparse approximation, ICES report 04-04, 2004.

[22] S.G. Wilson, Magnitude/phase quantization of independent Gaussian variates, IEEE Trans. Commun. 28 (1980) 1924–1929.

[23] L. Daudet, B. Torrésani, Hybrid representations for audiophonic signal encoding, Signal Processing 82 (11) (2002) 1595–1617.

[24] S. Molla, B. Torrésani, Determining local transientness in audio signals, IEEE Signal Process. Lett. 11 (7) (2004) 625–628.