Simultaneous Noise Suppression and Signal Compression using a Library of Orthonormal Bases and the Minimum Description Length Criterion

Naoki Saito

Abstract. We describe an algorithm to estimate a discrete signal from its noisy observation, using a library of orthonormal bases (consisting of various wavelets, wavelet packets, and local trigonometric bases) and the information-theoretic criterion called minimum description length (MDL). The key to effective random noise suppression is that the signal component in the data may be represented efficiently by one or more of the bases in the library, whereas the noise component cannot be represented efficiently by any basis in the library. The MDL criterion gives the best compromise between the fidelity of the estimation result to the data (noise suppression) and the efficiency of the representation of the estimated signal (signal compression): it selects the "best" basis and the "best" number of terms to be retained out of various bases in the library in an objective manner. Because of the use of the MDL criterion, our algorithm is free from any parameter setting or subjective judgments.

This method has been applied usefully to various geophysical datasets containing many transient features.

§1. Introduction

Wavelet transforms and their relatives such as wavelet packet transforms and local trigonometric transforms are becoming increasingly popular in many fields of applied sciences. So far their most successful application area seems to be data compression; see e.g., [14], [6], [35], [30]. Meanwhile, several researchers claimed that wavelets and these transforms are also useful for reducing noise in (or denoising) signals/images [16], [7], [10], [21]. In this paper, we take advantage of both sides: we propose an algorithm for *simultaneously* suppressing random noise in data and compressing the

Wavelets in Geophysics
E. Foufoula-Georgiou and P. Kumar (eds.), pp. xxx-yyy.

ISBN 0-12-xxxxxx-x

signal, i.e., we try to "kill two birds with one stone."

Throughout this paper, we consider a simple degradation model: observed data consists of a signal component and additive white Gaussian noise. Our algorithm estimates the signal component from the data using a library of orthonormal bases (including various wavelets, wavelet packets, and local trigonometric bases) and the information-theoretic criterion called the Minimum Description Length (MDL) criterion for discriminating signal from noise.

The key motivation here is that the signal component in the data can often be efficiently represented by one or more of the bases in the library whereas the noise component cannot be represented efficiently by any basis in the library.

The use of the MDL criterion frees us from any subjective parameter setting such as threshold selection. This is particularly important for real field data where the noise level is difficult to obtain or estimate *a priori.*

The organization of this paper is as follows. In Section 2, we review some of the important properties of wavelets, wavelet packets, local trigonometric transforms which constitute the "library of orthonormal bases" which will be used for efficiently representing nonstationary signals. In Section 3, we formulate our problem. We view the problem of simultaneous noise suppression and signal compression as a model selection problem out of models generated by the library of orthonormal bases. In Section 4, we review the MDL principle which plays a critical role in this paper. We also give some simple examples to help understand its concept. In Section 5, we develop an actual algorithm of simultaneous noise suppression and signal compression. We also give the computational complexity of our algorithm. Then, we extend our algorithm for higher dimensional signals (images) in Section 6. In Section 7, we apply our algorithm to several geophysical datasets, both synthetic and real, and compare the results with other competing methods. We discuss the connection of our algorithm with other approaches in Section 8, and finally, we conclude in Section 9.

§2. A Library of Orthonormal Bases

For our purpose we need to represent signals containing many transient features and edges in an efficient manner. Wavelets and their relatives, i.e., wavelet packets and local trigonometric transforms, have been found very useful for this purpose; see e.g., [14], [6], [35], [30]. As shown below, each of these transforms (or basis functions) has different characteristics. In other words, the best transform to compress a particular signal may not be good for another signal. Therefore, instead of restricting our attention to a particular basis, we consider a *library* of bases. The most suitable basis for a particular signal is selected from this collection of bases. This approach leads to a vastly more efficient representation for the signal, compared with confining ourselves to a single basis.

In this section, we briefly describe the most important properties of these transforms. Throughout this paper, we only consider real-valued discrete signals (or vectors) with finite length N $(= 2^n)$. Also we limit our discussions to orthonormal transforms. Hence it suffices here to consider discrete orthonormal transforms, i.e., the orthonormal bases of $\ell^2(N)$, the N-dimensional space of vectors of finite energy.

More detailed properties of these bases can be found in the literature, most notably, in [2], [9], [13], [23], [22], [26], [33].

2.1. Wavelet Bases

The wavelet transform (e.g., [13], [23]) can be considered as a smooth partition of the frequency axis. The signal is first decomposed into low and high frequency components by the convolution-subsampling operations with the pair consisting of a "lowpass" filter $\{h_k\}$ and a "highpass" filter $\{g_k\}$ directly on the discrete time domain. Let H and G be the convolution-subsampling operators using these filters and H^* and G^* be their adjoint (i.e., upsampling-anticonvolution) operations. It turns out that we can choose finite-length (L) filters and satisfy the following orthogonality (or perfect reconstruction) conditions:

$$HG^* = GH^* = 0, \quad \text{and} \quad H^*H + G^*G = I,$$

where I is the identity operator of $\ell^2(N)$. Also we have the relation $g_k = (-1)^k h_{L-1-k}$. The pair of filters $\{h_k\}_{k=0}^{L-1}$ and $\{g_k\}_{k=0}^{L-1}$ satisfying these conditions are called *quadrature mirror filters* (QMFs).

This decomposition (or expansion, or analysis) process is iterated on the low frequency components and each time the high frequency coefficients are retained intact and at the last iteration, both low and high frequency coefficients are kept. In other words, let $\boldsymbol{f} = \{f_k\}_{k=0}^{N-1} \in \ell^2(N)$ be a vector to be expanded. Then, the convolution-subsampling operations transform the vector $\boldsymbol{f}$ into two subsequences $H\boldsymbol{f}$ and $G\boldsymbol{f}$ of lengths $N/2$. Next, the same operations are applied to the vector $H\boldsymbol{f}$ to obtain $H^2\boldsymbol{f}$ and $GH\boldsymbol{f}$ of lengths $N/4$. If the process is iterated J $(\leq n)$ times, we have the discrete wavelet coefficients $(G\boldsymbol{f}, GH\boldsymbol{f}, GH^2\boldsymbol{f}, \ldots, GH^J\boldsymbol{f}, H^{J+1}\boldsymbol{f})$ of length N. As a result, the wavelet transform analyzes the data by partitioning its frequency content dyadically finer and finer toward the low frequency region (i.e., coarser and coarser in the original time or space domains).

If we were to partition the frequency axis sharply using the characteristic functions (or box-car functions), then we would have ended up the so-called Shannon (or Littlewood-Paley) wavelets, i.e., the difference of two sinc functions. Clearly, however, we cannot have a finite-length filter in the time domain in this case. The other extreme is the Haar basis which partitions the frequency axis quite badly but gives the shortest filter length ($L = 2$) in the time domain.

The reconstruction (or synthesis) process is also very simple: starting from the lowest frequency components (or coarsest scale coefficients) $H^{J+1}\boldsymbol{f}$ and the second lowest frequency components $GH^J\boldsymbol{f}$, the adjoint operations are applied and added to obtain $H^J\boldsymbol{f} = H^*H^{J+1}\boldsymbol{f} + G^*GH^J\boldsymbol{f}$. This process is iterated to reconstruct the original vector $\boldsymbol{f}$. The computational complexity of the decomposition and reconstruction process is in both cases $O(N)$ as easily seen.

We can construct the basis vector $\boldsymbol{w}_{j,k}$ at scale j and position k simply by putting $(GH^j\boldsymbol{f})_l = \delta_{l,k}$, where $\delta_{l,k}$ denotes the Kronecker delta, and synthesizing $\boldsymbol{f} = \boldsymbol{w}_{j,k}$ by the reconstruction algorithm. Using these basis vectors, we can express the wavelet transform in a vector-matrix form as

$$\boldsymbol{\alpha} = \boldsymbol{W}^T\boldsymbol{f},$$

where $\boldsymbol{\alpha} \in \mathbf{R}^N$ contains the wavelet coefficients and $\boldsymbol{W} \in \mathbf{R}^{N\times N}$ is an orthogonal matrix consisting of column vectors $\boldsymbol{w}_{j,k}$. This basis vector has the following important properties:

- *vanishing moments:* $\sum_{l=0}^{N-1} l^m \boldsymbol{w}_{j,k}(l) = 0$ for $m = 0, 1, \ldots, M-1$.

The higher the degrees of vanishing moments the basis has, the better it compresses the smooth part of the signal. In the original construction of Daubechies [12], it turns out that $L = 2M$. There are several other possibilities. One of them is a family of the so-called "coiflets" with $L = 3M$ which are less asymmetric than the original wavelets of Daubechies [13].

- *regularity:* $|\boldsymbol{w}_{j,k}(l+1) - \boldsymbol{w}_{j,k}(l)| \le c\ 2^{-j\alpha}$,

where $c > 0$ is a constant and $\alpha > 0$ is called the *regularity* of the wavelets. The larger the value of α is, the smoother the basis vector becomes. This property may be important if one requires high compression rate since the shapes of the basis vectors become "visible" in those cases and one might want to avoid fractal-like shapes in the compressed signals/images [25].

- *compact support:* $\boldsymbol{w}_{j,k}(l) = 0 \quad \text{for } l \notin [2^j k, 2^j k + (2^j - 1)(L-1)]$.

The compact support property is important for efficient and exact numerical implementation.

2.2. Wavelet Packet Best-Bases

For oscillating signals such as acoustic signals, the analysis by the wavelet transform is sometimes inefficient because it only partitions the frequency axis finely toward the low frequency. The wavelet packet transform (e.g., [9], [22], [33]) decomposes even the high frequency bands which are kept intact in the wavelet transform. The first level decomposition is $H\boldsymbol{f}$ and $G\boldsymbol{f}$ just like in the wavelet transform. The second level is $H^2\boldsymbol{f}, GH\boldsymbol{f}, HG\boldsymbol{f}, G^2\boldsymbol{f}$. If we repeat this process for J times, we end up having JN expansion coefficients. Clearly, we have a redundant set of expansion coefficients, in fact, there are more than $2^{2^{(J-1)}}$ possible orthonormal bases. One way of selecting an efficient basis for representing the signal or vector is to use the entropy criterion [9], [33]. We can think of the wavelet packet bases as a set of different coordinate systems of $\mathbf{R}^N$. Then a signal of length N is a point in $\mathbf{R}^N$, and we try to select the most efficient coordinate system out of the given set of coordinate systems to represent this signal. The signal in an efficient coordinate system should have large magnitudes along a few axes and small magnitudes along most axes. In particular, the wavelet packet basis function becomes a unit vector along an axis of the coordinate systems. Then, it is very natural to use the entropy as a measure of efficiency of the coordinate system. The *best-basis* is the basis or coordinate system giving the minimum entropy for its coordinate distribution. The computational complexity of computing the best-basis is $O(N \log_2 N)$ as is the reconstruction of the original vector from the best-basis coefficients.

Remark. We would like to note that given a set of signals, the Karhunen-Loève basis gives the global minimum entropy. However, it is very expensive to compute; the cost is $O(N^3)$ since it involves solving an eigenvalue problem. On the other hand, the wavelet packet best-basis can be computed cheaply and is defined even for a single signal; see [34] for a comparison of these two bases using images of human faces.

2.3. Local Trigonometric Best-Bases

Local trigonometric transforms ([9], [22], [33], [2]) can be considered as conjugates of wavelet packet transforms: they partition the time (or space) axis smoothly. In fact, Coifman and Meyer [8] showed that it is possible to partition the real-line into any disjoint intervals smoothly and construct orthonormal bases on each interval. In the actual numerical implementation, the data is first partitioned into disjoint intervals by the smooth window function, and then on each interval the data is transformed by the discrete cosine or sine transforms (DCT/DST). Since it partitions the axis smoothly, these transforms, i.e., local cosine or sine transforms (LCT/LST), have less edge (or blocking) effects than the conventional DCT/DST. Wickerhauser [33] proposed the method of dyadically partitioning the time axis and computing the best-basis using the entropy criterion similarly to the wavelet packet best-basis construction. The computational complexity in this case is about $O(N[\log_2 N]^2)$. Local trigonometric transforms are clearly efficient for the signals with localized oscillating features such as musical notes.

§3. Problem Formulation

Let us consider a discrete degradation model

$$\boldsymbol{d} = \boldsymbol{f} + \boldsymbol{n},$$

where $\boldsymbol{d}, \boldsymbol{f}, \boldsymbol{n} \in \mathbf{R}^N$ and $N = 2^n$. The vector $\boldsymbol{d}$ represents the noisy observed data and $\boldsymbol{f}$ is the unknown true signal to be estimated. The vector $\boldsymbol{n}$ is white Gaussian noise (WGN), i.e., $\boldsymbol{n} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$. Let us assume that σ^2 is unknown.

We now consider an algorithm to estimate $\boldsymbol{f}$ from the noisy observation $\boldsymbol{d}$. First, we prepare the library of orthonormal bases mentioned in the previous section. This library consists of the standard Euclidean basis of $\mathbf{R}^N$, the Haar-Walsh bases, various wavelet bases and wavelet packet best-bases generated by Daubechies's QMFs, their less asymmetric versions (i.e., coiflets), and local trigonometric best-bases. This collection of bases is highly adaptable and versatile for representing various transient signals [7]. For example, if the signal consists of blocky functions such as acoustic impedance profiles of subsurface structure, the Haar-Walsh bases capture those discontinuous features both accurately and efficiently. If the signal consists of piecewise polynomial functions of order p, then the Daubechies wavelets/wavelet packets with filter length $L \geq 2(p+1)$ or the coiflets with filter length $L \geq 3(p+1)$ would be efficient because of the vanishing moment property. If the signal has a sinusoidal shape or highly oscillating characteristics, the local trigonometric bases would do the job. Moreover, computational efficiency of this library is also attractive; the most expensive expansion in this library, i.e., the local trigonometric expansion, costs about $O(N[\log_2 N]^2)$ as explained in the previous section.

Let us denote this library by $\mathcal{L} = \{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_M\}$, where $\mathcal{B}_m$ represents one of the orthonormal bases in the library, and M (typically 5 to 20) is the number of bases in this library. If we want, we can add other orthonormal bases in this library such as the Karhunen-Loève basis [1] or the prolate spheroidal wave functions [13], [36]. However, normally, the above-mentioned multiresolution bases are more than enough, considering their versatility and computational efficiency [7].

Since the bases in the library $\mathcal{L}$ compress signals/images very well, we make a strong assumption here: we suppose the unknown signal $\boldsymbol{f}$ can be *completely* represented by k $(< N)$ elements of a basis $\mathcal{B}_m$, i.e.,

$$\boldsymbol{f} = \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}, \tag{1}$$

where $\boldsymbol{W}_m \in \mathbf{R}^{N \times N}$ is an orthogonal matrix whose column vectors are the basis elements of $\mathcal{B}_m$, and $\boldsymbol{\alpha}_m^{(k)} \in \mathbf{R}^N$ is the vector of expansion coefficients of $\boldsymbol{f}$ with only k non-zero coefficients. At this point, we do not know the actual value of k and the basis $\mathcal{B}_m$. We would like to emphasize that in reality the signal $\boldsymbol{f}$ might not be strictly represented by (1). We regard (1) as a *model at hand* rather than a rigid physical model exactly *explaining* $\boldsymbol{f}$ and we will try our best under this assumption. (This is often the case if we want to fit polynomials to some data.) Now the problem of simultaneous noise suppression and signal compression can be stated as follows: *find the "best" k and m given the library $\mathcal{L}$.* In other words, we translate the estimation problem into a model selection problem where models are the bases $\mathcal{B}_m$ and the number of terms k under the additive WGN assumption.

For the purpose of data compression, we want to have k as small as possible. At the same time, we want to minimize the distortion between the estimate and the true signal by choosing the most suitable basis $\mathcal{B}_m$, keeping in mind that the larger k normally gives smaller value of error. How can we satisfy these seemingly conflicting demands?

§4. The Minimum Description Length Principle

To satisfy the above mentioned conflicting demands, we need a model selection criterion. One of the most suitable criteria for our purpose is the so-called *Minimum Description Length* (MDL) criterion proposed by Rissanen [27], [28], [29]. The MDL principle suggests that the "best" model among the given collection of models is the one giving the shortest description of the data *and* the model itself. For each model in the collection, the length of description of the data is counted as the codelength of encoding the data using that model in binary digits (bits). The length of description of a model is the codelength of specifying that model, e.g., the number of parameters and their values if it is a parametric model.

To help understand what "code" or "encoding" means, we give some simple examples. We assume that we want to transmit data by first encoding (mapping) them into a bitstream by an encoder, then receive the bitstream by a decoder, and finally try to reconstruct the data. Let $L(\boldsymbol{x})$ denote the codelength (in bits) of a vector $\boldsymbol{x}$ of deterministic or probabilistic parameters which are either real-valued, integer-valued, or taking values in a finite alphabet.

Example 4.1. *Codelength of symbols drawn from a finite alphabet.*
Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$ be a string of symbols drawn from a finite alphabet $\mathcal{X}$, which are independently and identically distributed (i.i.d.) with probability mass function $p(x)$, $x \in \mathcal{X}$. In this case, clearly the frequently occurring symbols should have shorter codelengths than rarely occurring symbols for efficient communication. This leads to the so-called Shannon code [11] whose codelength (if we ignore the integer requirement for the

codelength) can be written as

$$L(x) = -\log p(x) \qquad \text{for } x \in \mathcal{X}.$$

(From now on, we denote the logarithm of base 2 by "log", and the natural logarithm, i.e., base e by "ln".) The Shannon code has the shortest codelength *on the average*, and satisfies the so-called Kraft inequality [11]:

$$\sum_{x \in \mathcal{X}} 2^{-L(x)} \leq 1, \tag{2}$$

which is necessary and sufficient for the existence of an instantaneously decodable code, i.e., a code such that there is no codeword which is the prefix of any other codeword in the coding system. The shortest codelength on the average for the whole sequence $\boldsymbol{x}$ becomes

$$L(\boldsymbol{x}) = \sum_{i=1}^{N} L(x_i) = -\sum_{i=1}^{N} \log p(x_i).$$

Example 4.2. *Codelength of deterministic integers.*
For a deterministic parameter $j \in \mathbf{Z}_N = (0, 1, \ldots, N-1)$ (i.e., both the encoder and decoder know N), the codelength of describing j is written as $L(j) = \log N$ since $\log N$ bits are required to index N integers. This can also be interpreted as a codelength using Shannon code for a sample drawn from the uniform distribution over $(0, 1, \ldots, N-1)$.

Example 4.3. *Codelength of an integer (universal prior for an integer).*
Suppose we do not know how large a natural number j is. Rissanen [27] proposed that the code of such j should be the binary representation of j, preceded by the code describing its length $\log j$, preceded by the code describing the length of the code for $\log j$, and so forth. This recursive strategy leads to

$$L^*(j) = \log^* j + \log c_0 = \log j + \log\log j + \cdots + \log c_0,$$

where the sum involves only the non-negative terms and the constant $c_0 \approx 2.865064$ which was computed so that equality holds in (2), i.e., $\sum_{j=1}^{\infty} 2^{-L^*(j)} = 1$. This can be generalized for an integer j by defining

$$L^*(j) = \begin{cases} 1 & \text{if } j = 0, \\ \log^* |j| + \log 4c_0 & \text{otherwise.} \end{cases} \tag{3}$$

(We can easily see that (3) satisfies $\sum_{j=-\infty}^{\infty} 2^{-L^*(j)} = 1$.)

Example 4.4. *Codelength of a truncated real-valued parameter.*
For a deterministic real-valued parameter $v \in \mathbf{R}$, the exact code generally

requires infinite length of bits. Thus, in practice, some truncation must be done for transmission. Let δ be the precision and v_δ be the truncated value, i.e., $|v - v_\delta| < \delta$. Then, the number of bits required for v_δ is the sum of the codelength of its integer part $[v]$ and the number of fractional binary digits of the truncation precision δ, i.e.,

$$L(v_\delta) = L^*([v]) + \log(1/\delta). \tag{4}$$

Having gone through the above examples, now we can state the MDL principle more clearly. Let $\mathcal{M} = \{\boldsymbol{\theta}_m : m = 1, 2, \ldots\}$ be a class or collection of models at hand. The integer m is simply an index of a model in the list. Let $\boldsymbol{x}$ be a sequence of observed data. Assume that we do not know the true model $\boldsymbol{\theta}$ generating the data $\boldsymbol{x}$. As in [29], [24], given the index m, we can write the codelength for the whole process as

$$L(\boldsymbol{x}, \boldsymbol{\theta}_m, m) = L(m) + L(\boldsymbol{\theta}_m \mid m) + L(\boldsymbol{x} \mid \boldsymbol{\theta}_m, m). \tag{5}$$

This equation says that the codelength to rewrite the data is the sum of the codelengths to describe: (i) the index m, (ii) the model $\boldsymbol{\theta}_m$ given m, and (iii) the data $\boldsymbol{x}$ using the model $\boldsymbol{\theta}_m$. The MDL criterion suggests picking the model $\boldsymbol{\theta}_{m^*}$ which gives the minimum of the total description length (5).

The last term of the right-hand side (RHS) of (5) is the length of the Shannon code of the data assuming the model $\boldsymbol{\theta}_m$ is the true model, i.e.,

$$L(\boldsymbol{x} \mid \boldsymbol{\theta}_m, m) = -\log p(\boldsymbol{x} \mid \boldsymbol{\theta}_m, m), \tag{6}$$

and the maximum likelihood (ML) estimate $\widehat{\boldsymbol{\theta}}_m$ minimizes (6) by the definition:

$$L(\boldsymbol{x} \mid \widehat{\boldsymbol{\theta}}_m, m) = -\log p(\boldsymbol{x} \mid \widehat{\boldsymbol{\theta}}_m, m) \le -\log p(\boldsymbol{x} \mid \boldsymbol{\theta}_m, m). \tag{7}$$

However, we should consider a further truncation of $\widehat{\boldsymbol{\theta}}_m$ as shown in Example 4.4 above to check that additional savings in the description length is possible. The finer truncation precision we use, the smaller the term (7), but the larger the term $L(\widehat{\boldsymbol{\theta}}_m \mid m)$ becomes. Suppose that the model $\boldsymbol{\theta}_m$ has k_m real-valued parameters, i.e., $\boldsymbol{\theta}_m = (\theta_{m,1}, \ldots, \theta_{m,k_m})$. Rissanen showed in [27], [29] that the optimized truncation precision (δ^*) is of order $1/\sqrt{N}$ and

$$\begin{aligned}
&\min_\delta L(\boldsymbol{x}, \boldsymbol{\theta}_{m,\delta}, m, \delta) \\
&\quad= L(m) + L(\widehat{\boldsymbol{\theta}}_{m,\delta^*} \mid m) + L(\boldsymbol{x} \mid \widehat{\boldsymbol{\theta}}_{m,\delta^*}, m) + O(k_m) \\
&\quad\approx L(m) + \sum_{j=1}^{k_m} L^*([\widehat{\theta}_{m,j}]) + \frac{k_m}{2}\log N + L(\boldsymbol{x} \mid \widehat{\boldsymbol{\theta}}_m, m) + O(k_m),
\end{aligned} \tag{8}$$

where $\widehat{\boldsymbol{\theta}}_m$ is the optimal non-truncated value given m, $\widehat{\boldsymbol{\theta}}_{m,\delta*}$ is its optimally truncated version, and $L^*(\cdot)$ is defined in (4). We note that the last term $O(k_m)$ in the approximation in (8) includes the penalty codelength necessary to describe the data $\boldsymbol{x}$ using the truncated ML estimate $\widehat{\boldsymbol{\theta}}_{m,\delta*}$ instead of the true ML estimate $\widehat{\boldsymbol{\theta}}_m$. In practice, we rarely need to obtain the optimally truncated value $\widehat{\boldsymbol{\theta}}_{m,\delta*}$ and we should compute $\widehat{\boldsymbol{\theta}}_m$ up to the machine precision, say, 10^{-15}, and use that value as the true ML estimate in (8). For sufficiently large N, the last term may be omitted, and instead of minimizing the ideal codelength (5), Rissanen proposed to minimize

$$MDL(\boldsymbol{x}, \widehat{\boldsymbol{\theta}}_m, m) = L(m) + \sum_{j=1}^{k_m} L^*([\widehat{\theta}_{m,j}]) + \frac{k_m}{2}\log N + L(\boldsymbol{x} \mid \widehat{\boldsymbol{\theta}}_m, m). \quad (9)$$

The minimum of (9) gives the best compromise between the low complexity in the model and high likelihood on the data.

The first term of the RHS of (9) can be written as

$$L(m) = -\log p(m), \quad (10)$$

where $p(m)$ is the probability of selecting m. If there is prior information about m as to which m is more likely, we should reflect this in $p(m)$. Otherwise, we assume each m is equally likely, i.e., $p(m)$ is a uniform distribution.

Remark. Even though the list of models $\mathcal{M}$ does not include the true model, the MDL method achieves the best result among the available models. See Barron and Cover [4] for detailed information on the error between the MDL estimate and the true model.

We also would like to note that the MDL principle does not attempt to find the absolutely minimum description of the data. The MDL always requires an available collection of models and simply suggests picking the best model from that collection. In other words, the MDL can be considered as an "oracle" for model selection [24]. This contrasts with the algorithmic complexities such as the Kolmogorov complexity which gives the absolutely minimum description of the data, however, in general, is impossible to obtain [27].

Before deriving our simultaneous noise suppression and signal compression algorithm in the context of the MDL criterion, let us give a closely related example:

Example 4.5. *A curve fitting problem using polynomials.*
Given N points of data $(x_i, y_i) \in \mathbf{R}^2$, consider the problem of fitting a polynomial through these points. The model class we consider is a set of

polynomials of orders $0, 1, \ldots, N-1$. In this case, $\boldsymbol{\theta}_m = (a_0, a_1, \ldots, a_m)$ represents the $m+1$ coefficients of a polynomial of order m. We also assume that the data is contaminated by the additive WGN with known variance σ^2, i.e.,

$$y_i = f(x_i) + e_i,$$

where $f(\cdot)$ is an unknown function to be estimated by the polynomial models, and $e_i \sim \mathcal{N}(0, \sigma^2)$. To invoke the MDL formalism, we pose this question in the information transmission setting. First we prepare an encoder which computes the ML estimate of the coefficients of the polynomial, $(\widehat{a}_0, \ldots, \widehat{a}_m)$, of the given degree m from the data. (In the additive WGN assumption the ML estimate coincides with the least squares estimate.) This encoder transmits these m coefficients as well as the estimation errors. We also prepare a decoder which receives the coefficients of the polynomial and residual errors and reconstruct the data. (We assume that the abscissas $\{x_i\}_{i=1}^N$ and the noise variance σ^2 are known to both the encoder and the decoder.) Then we ask how many bits of information should be transmitted to reconstruct the data. If we used polynomials of degree $N-1$, we could find a polynomial passing through all N points. In this case, we could describe the data extremely well. In fact, there is no error between the observed data and those reconstructed by the decoder. However, we do not gain anything in terms of data compression/transmission since we also have to encode the model which requires N coefficients of the polynomial. In some sense, we did not "learn" anything in this case. If we used the polynomial of degree 0, i.e., a constant, then it would be an extremely efficient model, but we would need many bits to describe the deviations from that constant. (Of course, if the underlying data is really a constant, then the deviation would be 0.)

Let us assume there is no prior preference on the order m. Then we can easily see that the total codelength (9) in this case becomes

$$\begin{aligned} MDL(\boldsymbol{y}, \widehat{\boldsymbol{\theta}}_m, m) \quad &= \quad \log N + \sum_{j=0}^{m} L^*([\widehat{a}_j]) + \frac{m+1}{2} \log N \\ &\quad + \frac{N}{2} \log 2\pi\sigma^2 + \frac{\log \mathrm{e}}{2\sigma^2} \sum_{i=1}^{N} \left(y_i - \sum_{j=0}^{m} \widehat{a}_j {x_i}^j \right)^2 . \end{aligned}$$

The MDL criterion suggests to pick the "best" polynomial of order m^* by minimizing this approximate codelength.

The MDL criterion has been successfully used in various fields such as signal detection [32], image segmentation [19], and cluster analysis [31] where the optimal number of signals, regions, and clusters, respectively, should be determined. If one knows *a priori* the physical model to explain the observed data, that model should definitely be used, e.g., the complex sinusoids in [32]. However, in general, as a descriptor of real-life signals which are full of transients or edges, the library of wavelets, wavelet packets, and local trigonometric transforms is more flexible and efficient than the set of polynomials or sinusoids.

§5. A Simultaneous Noise Suppression and Signal Compression Algorithm

We carry on our development of the algorithm based on the information transmission setting as the polynomial curve fitting problem described in the previous section. We consider again an encoder and a decoder for our problem. Given (k,m) in (1), the encoder expands the data $\boldsymbol{d}$ in the basis $\mathcal{B}_m$, then transmits the number of terms k, the specification of the basis m, and k expansion coefficients, the variance of the WGN model σ^2, and finally the estimation errors. The decoder receives this information in bits and tries to reconstruct the data $\boldsymbol{d}$.

In this case, the total codelength to be minimized may be expressed as the sum of the codelengths of: (i) two natural numbers (k,m), (ii) $(k+1)$ real-valued parameters $(\boldsymbol{\alpha}_m^{(k)}, \sigma^2)$ given (k,m), and (iii) the deviations of the observed data $\boldsymbol{d}$ from the (estimated) signal $\boldsymbol{f} = \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}$ given $(k, m, \boldsymbol{\alpha}_m^{(k)}, \sigma^2)$. The approximate total description length (9) now becomes

$$\begin{aligned} & MDL(\boldsymbol{d}, \widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2, k, m) \\ & \quad = \quad L(k,m) + L(\widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2 \mid k, m) + L(\boldsymbol{d} \mid \widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2, k, m), \end{aligned} \tag{11}$$

where $\widehat{\boldsymbol{\alpha}}_m^{(k)}$ and $\widehat{\sigma}^2$ are the ML estimates of $\boldsymbol{\alpha}_m^{(k)}$ and σ^2, respectively.

Let us now derive these ML estimates. Since we assumed the noise component is additive WGN, the probability of observing the data given all model parameters is

$$P(\boldsymbol{d} \mid \boldsymbol{\alpha}_m^{(k)}, \sigma^2, k, m) = (2\pi\sigma^2)^{-N/2} \exp\left(-\frac{\|\boldsymbol{d} - \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}\|^2}{2\sigma^2} \right), \tag{12}$$

where $\|\cdot\|$ is the standard Euclidean norm on $\mathbf{R}^N$. For the ML estimate of σ^2, first consider the log-likelihood of (12)

$$\ln p(\boldsymbol{d} \mid \boldsymbol{\alpha}_m^{(k)}, \sigma^2, k, m) = -\frac{N}{2} \ln 2\pi\sigma^2 - \frac{\|\boldsymbol{d} - \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}\|^2}{2\sigma^2}. \tag{13}$$

Taking the derivative with respect to σ^2 and setting it to zero, we easily obtain

$$\widehat{\sigma}^2 = \frac{1}{N} \|\boldsymbol{d} - \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}\|^2. \tag{14}$$

Insert this equation back to (13) to get

$$\ln p(\boldsymbol{d} \mid \boldsymbol{\alpha}_m^{(k)}, \widehat{\sigma}^2, k, m) = -\frac{N}{2} \ln \left(\frac{2\pi}{N} \|\boldsymbol{d} - \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}\|^2 \right) - \frac{N}{2}. \qquad (15)$$

Let $\widetilde{\boldsymbol{d}}_m = \boldsymbol{W}_m^T \boldsymbol{d}$ denote the vector of the expansion coefficients of $\boldsymbol{d}$ in the basis $\mathcal{B}_m$. Since this basis is orthonormal, i.e., $\boldsymbol{W}_m$ is orthogonal, and we use the ℓ^2 norm, we have

$$\|\boldsymbol{d} - \boldsymbol{W}_m \boldsymbol{\alpha}_m^{(k)}\|^2 = \|\boldsymbol{W}_m(\boldsymbol{W}_m^T \boldsymbol{d} - \boldsymbol{\alpha}_m^{(k)})\|^2 = \|\widetilde{\boldsymbol{d}}_m - \boldsymbol{\alpha}_m^{(k)}\|^2. \qquad (16)$$

From (15), (16), and the monotonicity of the ln function, we find that maximizing (15) is equivalent to minimizing

$$\|\widetilde{\boldsymbol{d}}_m - \boldsymbol{\alpha}_m^{(k)}\|^2. \qquad (17)$$

Considering that the vector $\boldsymbol{\alpha}_m^{(k)}$ only contains k nonzero elements, we can easily conclude that the minimum of (17) is achieved by taking the largest k coefficients in magnitudes of $\widetilde{\boldsymbol{d}}_m$ as the ML estimate of $\boldsymbol{\alpha}_m^{(k)}$, i.e.,

$$\widehat{\boldsymbol{\alpha}}_m^{(k)} = \boldsymbol{\Theta}^{(k)} \widetilde{\boldsymbol{d}}_m = \boldsymbol{\Theta}^{(k)}(\boldsymbol{W}_m^T \boldsymbol{d}), \qquad (18)$$

where $\boldsymbol{\Theta}^{(k)}$ is a thresholding operation which keeps the k largest elements in absolute value intact and sets all other elements to zero. Finally, inserting (18) into (14), we obtain

$$\widehat{\sigma}^2 = \frac{1}{N} \|\boldsymbol{W}_m^T \boldsymbol{d} - \boldsymbol{\Theta}^{(k)} \boldsymbol{W}_m^T \boldsymbol{d}\|^2 = \frac{1}{N} \|(\boldsymbol{I} - \boldsymbol{\Theta}^{(k)}) \boldsymbol{W}_m^T \boldsymbol{d}\|^2, \qquad (19)$$

where $\boldsymbol{I}$ represents the N dimensional identity operator (matrix).

Let us further analyze (11) term by term. If we do not have any prior information on (k, m), then the cost $L(k, m)$ is the same for all cases, i.e., we can drop the first term of (11) for minimization purpose. However, if one has some prior preference about the choice of basis, knowing some prior information about the signal $\boldsymbol{f}$, $L(k, m)$ should reflect this information. For instance, if we happen to know that the original function $\boldsymbol{f}$ consists of a linear combination of dyadic blocks, then we clearly should use the Haar basis. In this case, we may use the Dirac distribution, i.e., $p(m) = \delta_{m,m_0}$, where m_0 is the index for the Haar basis in the library $\mathcal{L}$. By (10), this leads to

$$L(k, m) = \begin{cases} L(k) & \text{if } m = m_0, \\ +\infty & \text{otherwise.} \end{cases}$$

On the other hand, if we either happen to know *a priori* or want to force the number of terms retained (k) to satisfy $k_1 \le k \le k_2$, then we may want to assume the uniform distribution for this range of k, i.e.,

$$L(k, m) = \begin{cases} L(m) + \log(k_2 - k_1 + 1) & \text{if } k_1 \le k \le k_2, \\ +\infty & \text{otherwise.} \end{cases} \qquad (20)$$

As for the second term of (11), which is critical for our algorithm, we have to encode k expansion coefficients $\widehat{\boldsymbol{\alpha}}_m^{(k)}$ and $\widehat{\sigma}^2$, i.e., $(k+1)$ real-valued parameters. However, in this case, by normalizing the whole sequence by $\|\boldsymbol{d}\|$, we can safely assume that the magnitude of each coefficient in $\widehat{\boldsymbol{\alpha}}^{(k)}$ is strictly less than one; in other words, the integer part of each coefficient is simply zero. Hence we do not need to encode the integer part as in (9) if we transmit the real-valued parameter $\|\boldsymbol{d}\|$. Now the description length of $(\widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2)$ given (k, m) becomes approximately $\frac{k+2}{2}\log N + L^*([\widehat{\sigma}^2]) + L^*([\|\boldsymbol{d}\|])$ bits since there are $k+2$ real-valued parameters: k nonzero coefficients, $\widehat{\sigma}^2$, and $\|\boldsymbol{d}\|$. After normalizing by $\|\boldsymbol{d}\|$, we clearly have $\widehat{\sigma}^2 < 1$ (see (19)), so that $L^*([\widehat{\sigma}^2]) = 1$ (see (3)). For each expansion coefficient, however, we still need to specify the index of the coefficient, i.e., where the k non-zero elements are in the vector $\widehat{\boldsymbol{\alpha}}_m^{(k)}$. This requires $k\log N$ bits. As a result, we have

$$L(\widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2 \mid k, m) = \frac{3}{2}k\log N + c, \tag{21}$$

where c is a constant independent of (k, m).

Since the probability of observing $\boldsymbol{d}$ given all model parameters is given by (12), we have for the last term in (11)

$$L(\boldsymbol{d} \mid \widehat{\boldsymbol{\alpha}}_m^{(k)}, \widehat{\sigma}^2, k, m) = \frac{N}{2}\log\|(\boldsymbol{I} - \boldsymbol{\Theta}^{(k)})\boldsymbol{W}_m^T\boldsymbol{d}\|^2 + c', \tag{22}$$

where c' is a constant independent of (k, m).

Finally we can state our simultaneous noise suppression and signal compression algorithm. Let us assume that we do not have any prior information on (k, m) for now. Then, from (11), (21), and (22) with ignoring the constant terms c and c', our algorithm can be stated as:
Pick the index (k^*, m^*) *such that*

$$AMDL(k^*, m^*) = \min_{\substack{0\le k<N\\ 1\le m\le M}} \left(\frac{3}{2}k\log N + \frac{N}{2}\log\|(\boldsymbol{I} - \boldsymbol{\Theta}^{(k)})\boldsymbol{W}_m^T\boldsymbol{d}\|^2\right). \tag{23}$$

Then reconstruct the signal estimate

$$\widehat{\boldsymbol{f}} = \boldsymbol{W}_{m^*}\boldsymbol{\alpha}_{m^*}^{(k^*)}. \tag{24}$$

Let us call the objective function to be minimized in (23), the approximate MDL (AMDL) since we ignored the constant terms. Let us now show a typical behavior of the AMDL value as a function of the number of terms retained (k) in Figure 1. (In fact, this curve is generated using Example 7.1 below.) We see that the log(residual energy) always decreases as k increases. By adding the penalty term of retaining the expansion coefficients, i.e., $(3/2)k\log N$ (which is just a straight line), we have the AMDL

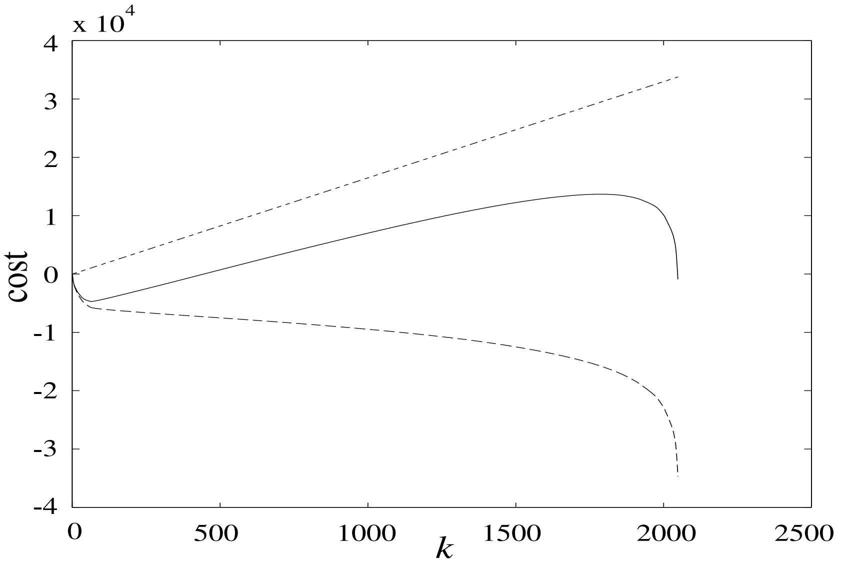


Figure 1. Graphs of AMDL versus k: AMDL [solid line] which is the sum of the $(3/2)k \log N$ term [dotted line] and the $(N/2)$ log(residual energy) term [dashed line].

curve which typically decreases for the small k, then starts increasing because of the penalty term, then finally decreases again at some large k near from $k = N$ because the residual error becomes very small. Now what we really want is the value of k achieving the minimum at the beginning of the k-axis, and we want to avoid searching for k beyond the maximum occurring for k near N. So, we can safely assume that $k_1 = 0$ and $k_2 = N/2$ in (20) to avoid searching more than necessary. (In fact, setting $k_2 > N/2$ does not make much sense in terms of data compression either.)

We briefly examine below the computational complexity of our algorithm. To obtain (k^*, m^*), we proceed as follows:

Step 1: Expand the data $\boldsymbol{d}$ into bases $\mathcal{B}_1, \ldots, \mathcal{B}_M$. Each expansion (including the best-basis selection procedure) costs $O(N)$ for wavelets, $O(N \log N)$ for wavelet packet best-bases, and $O(N[\log N]^2)$ for local trigonometric best-bases.

Step2: Let $K(= k_2 - k_1 + 1)$ denote the length of the search range for k. For $k_1 \le k \le k_2$, $1 \le m \le M$, compute the expression in the parenthesis of the RHS in (23). This costs approximately $O(N + 3MK)$ multiplications and MK calls to the log function.

Step 3: Search the minimum entry in this table, which costs MK compar-

isons.
Step 4: Reconstruct the signal estimate (24), which costs $O(N)$ for wavelets, $O(N \log N)$ for wavelet packet best-bases, and $O(N[\log N]^2)$ for local trigonometric best-bases.

§6. Extension to Images

For images or multidimensional signals, we can easily extend our algorithm by using the multidimensional version of the wavelets, wavelet packets, and local trigonometric transforms. In this section, we briefly summarize the two-dimensional (2D) versions of these transforms. For the 2D wavelets, there are several different approaches. The first one, which we call the sequential method, is the tensor product of the one-dimensional (1D) wavelets, i.e., applying the wavelet expansion algorithm separately along two axes t_1 and t_2 corresponding to column (vertical) and row (horizontal) directions respectively. Let $\boldsymbol{f} \in \mathbf{R}^{N_1 \times N_2}$ and H_i, G_i be the 1D convolution-subsampling operations along axis $t_i, i = 1, 2$. Then this version of the 2D wavelet transform first applies the convolution-subsampling operations along the t_1 axis to obtain $\boldsymbol{f}_1 = (G_1 \boldsymbol{f}, G_1 H_1 \boldsymbol{f}, \ldots, G_1 H_1^{J_1} \boldsymbol{f})$, then applies the convolution-subsampling operations along the t_2 axis to get the final 2D wavelet coefficients $(G_2 \boldsymbol{f}_1, G_2 H_2 \boldsymbol{f}_1, \ldots, G_2 H_2^{J_2} \boldsymbol{f}_1)$ of length $N_1 \times N_2$, where J_1 $(\leq \log N_1)$ and J_2 $(\leq \log N_2)$ are maximum levels of decomposition along t_1 and t_2 axes respectively. We note that one can choose different 1D wavelet bases for t_1 and t_2 axes independently. Given M different QMF pairs, there exist M^2 possible 2D wavelets using this approach.

The second approach is the basis generated from the tensor product of the multiresolution analysis. This decomposes an image $\boldsymbol{f}$ into four different sets of coefficients, $H_1 H_2 \boldsymbol{f}$, $G_1 H_2 \boldsymbol{f}$, $H_1 G_2 \boldsymbol{f}$, and $G_1 G_2 \boldsymbol{f}$, corresponding to "low-low", "high-low", "low-high", "high-high" frequency parts of the two variables, respectively. The decomposition is iterated on the "low-low" frequency part and this ends up in a "pyramid" structure of coefficients. Transforming the digital images by these wavelets to obtain the 2D wavelet coefficients are described in e.g., [20], [13].

There are also 2D wavelet bases which do not have a tensor-product structure, such as wavelets on the hexagonal grids and wavelets with matrix dilations. See e.g., [18], [17] for details.

There has been some argument as to which version of the 2D wavelet bases should be used for various applications [5], [13]. Our strategy toward this problem is this: we can put as many versions of these bases in the library as we can afford it in terms of computational time. Then minimizing the AMDL values automatically selects the most suitable one for our purpose.

As for the 2D version of the wavelet packet best-basis, the sequential method may be generalized, but it is not easily interpreted; the 1D best-bases may be different from column to column so that the resultant coefficients viewing along the row direction may not share the same frequency bands and scales unlike the 2D wavelet bases. This also makes the reconstruction algorithm complicated. Therefore, we should use the other tensor-product 2D wavelet approach for the construction of the 2D wavelet packet best-basis: we recursively decompose not only the "low-low" components but also the other three components. This process produces the "quad-tree" structure of wavelet packet coefficients instead of the "binary-tree" structure for 1D wavelet packets. Finally the 2D wavelet packet best-basis coefficients are selected using the entropy criterion [33].

The 2D version of the local trigonometric transforms can be constructed using the quad-tree structure again: the original image is smoothly folded and segmented into 4 subimages, 16 subimages, ..., and in each subimage the separable DCT/DST is applied, and then the quad-tree structure of the coefficients is constructed. Finally, the local trigonometric best-basis is selected using the entropy criterion [33].

For an image of $N = N_1 \times N_2$ pixels, the computational costs are approximately $O(N)$, $O(N \log_4 N)$, $O(N[\log_4 N]^2)$ for a 2D wavelet, a 2D wavelet packet best-basis, a 2D local trigonometric best-basis, respectively.

§7. Examples

In this section, we give several examples to show the usefulness of our algorithm.

Example 7.1. *The Synthetic Piecewise Constant Function of Donoho-Johnstone.*
We compared the performance of our algorithm in terms of the visual quality of the estimation and the relative ℓ^2 error with Donoho-Johnstone's method using the piecewise constant function used in their experiments [16]. The results are shown in Figure 2. The true signal is the piecewise constant function with $N = 2048$, and its noisy observation was created by adding the WGN sequence with $\|\boldsymbol{f}\|/\|\boldsymbol{n}\| = 7$. The library $\mathcal{L}$ for this example consisted of 18 different bases: the standard Euclidean basis of $\mathbf{R}^N$, the wavelet packet best-bases created with D02, D04, ..., D20, C06, C12, ..., C30, and the local cosine and sine best-bases (Dn represents the n-tap QMF of Daubechies and Cn represents the n-tap coiflet filter). In the Donoho-Johnstone method, we used the C06, i.e., 6-tap coiflet with 2 vanishing moments. We also specified the scale parameter $J = 7$, and supplied the *exact* value of σ^2. Next, we *forced* the Haar basis (D02) to

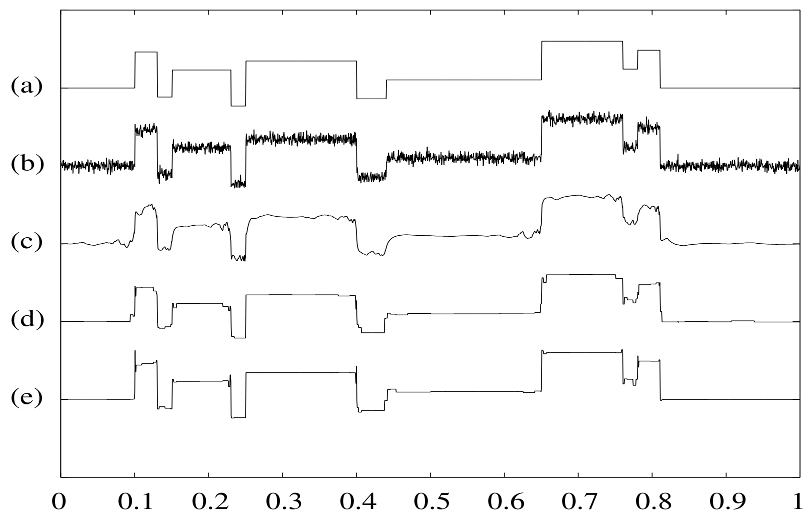


Figure 2. Results for the synthetic piecewise constant function: (a) Original piecewise constant function. (b) Noisy observation with (signal energy)/(noise energy) $= 7^2$. (c) Estimation by the Donoho-Johnstone method using coiflets C06. (d) Estimation by the Donoho-Johnstone method using Haar basis. (e) Estimation by the proposed method.

be used in their method. Finally, we applied our algorithm without specifying anything. In this case, the Haar-Walsh best-basis with $k^* = 63$ was automatically selected. The relative ℓ^2 errors are 0.116, 0.089, 0.051, respectively. Although the visual quality of our result is not too different from Donoho and Johnstone's (if we *choose* the appropriate basis for their method), our method generated the estimate with the smallest relative ℓ^2 error and slightly sharper edges. (See Section 8 for more about the Donoho-Johnstone method and its relation to our method.)

Example 7.2. *A Pure White Gaussian Noise.*
We generated a synthetic sequence of WGN with $\sigma^2 = 1.0$ and $N = 4096$. The same library as in Example 7.1 (with the best-bases adapted to this pure WGN sequence) was used. We also set the upper limit of search range $k_2 = N/2 = 2048$. Figure 3 shows the AMDL curves versus k for all bases in the library. As we can see, there is no single minimum in the graphs, and our algorithm satisfactorily decided $k^* = 0$, i.e., there is nothing to "learn" in this dataset.

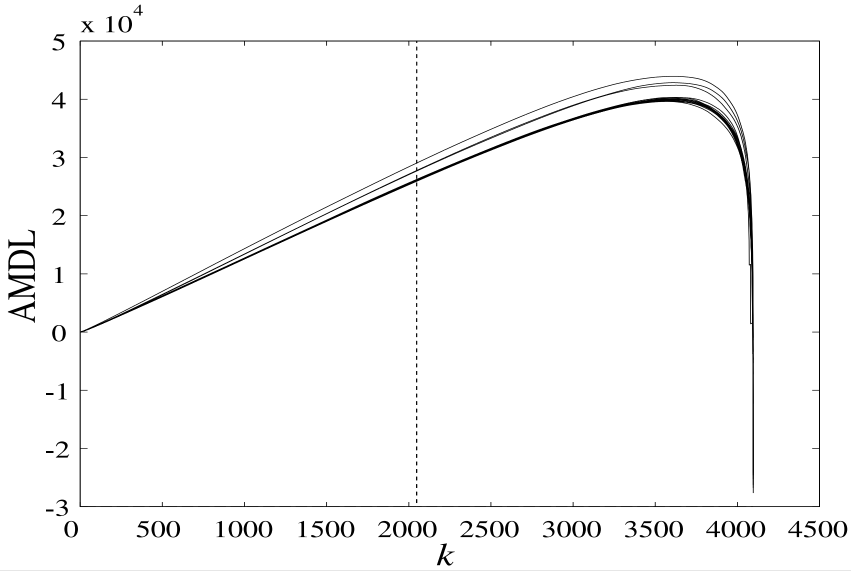


Figure 3. The AMDL curves of the White Gaussian Noise data for all bases. For each basis, $k = 0$ is the minimum value. The vertical dotted line indicates the upper limit of the search range for k.

Example 7.3. *A Natural Radioactivity Profile of Subsurface Formation.* We tested our algorithm on the actual field data which are measurement of natural radioactivity of subsurface formation obtained at an oil-producing well. The length of the data is $N = 1024$. Again, the same library was used as in the previous examples. The results are shown in Figure 4. In this case, our algorithm selected the D12 wavelet packet best-basis (Daubechies's 12-tap filter with 6 vanishing moments) with $k^* = 77$. The residual error is shown in Figure 4 (c) which consists mostly of a WGN-like high frequency component. The compression ratio is $1024/77 \approx 13.3$. However, to be able to reconstruct the signal from the surviving coefficients, we still need to record the indices of those coefficients.

Suppose we can store each index by b_i bytes of memory and the precision of the original data is b_f bytes per sample. Then the *storage reduction ratio* R_s can be computed by

$$R_s = \frac{N/r \times (b_f + b_i)}{N \times b_f} = \frac{1}{r}(1 + \frac{b_i}{b_f}), \tag{25}$$

where r is a compression ratio. The original data precision was $b_f = 8$ (bytes) in this case. Since it is enough to use $b_i = 2$ (bytes) for indices and

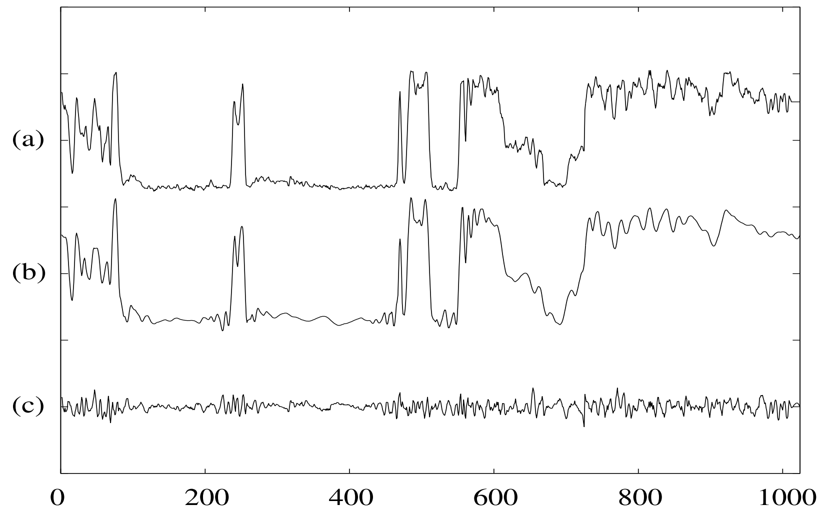
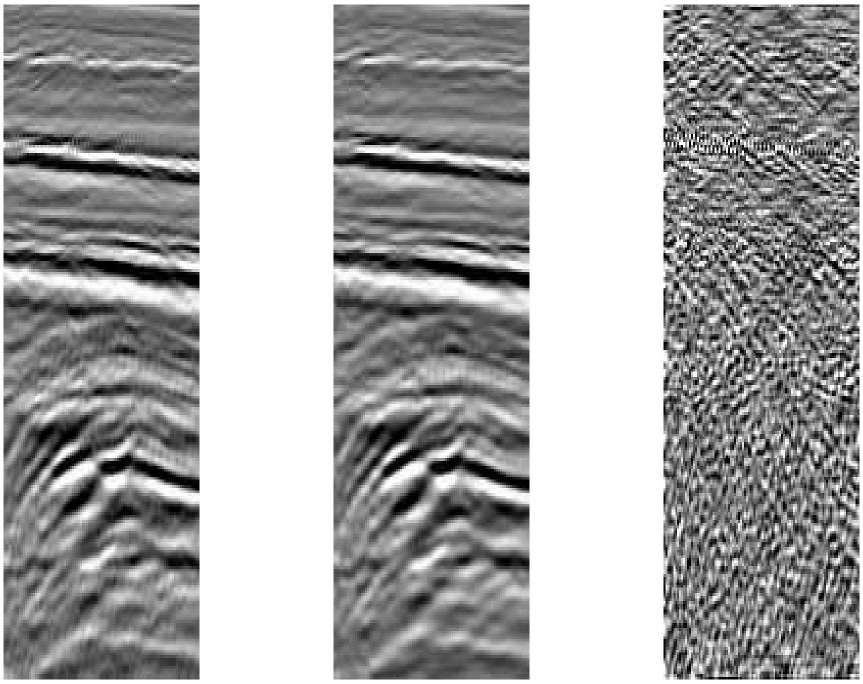


Figure 4. The estimate of the natural radioactivity profile of subsurface formation: (a) Original data which was measured in the borehole of an oil-producing well. (b) Estimation by the proposed method. (c) Residual error between (a) and (b).

$r = 13.3\%$, we have $R_s \approx 9.40\%$, i.e., 90.60% of the original data can be discarded.

Example 7.4. *A Migrated Seismic Section.*
In this example, the data is a migrated seismic section as shown in Figure 5 (a). The data consist of 128 traces of 256 time samples. We selected six 2D wavelet packet best-bases (D02, C06, C12, C18, C24, C30) as the library. Figure 5 (b) shows the estimate by our algorithm. It automatically selected the filter C30 and the number of terms retained as $k^* = 1611$. If we were to choose a good threshold in this example, it would be fairly difficult since we do not know the accurate estimate of σ^2. The compression rate, in this case, is $(128 \times 256)/1611 \approx 20.34$. The original data precision was $b_f = 8$ as in the previous example. In this case we have to use $b_i = 3$ (1 byte for row index, 1 byte for column index, and 1 byte for scale level). If we put these and $r = 20.34\%$ into (25), we have $R_s \approx 6.76\%$, i.e., 93.24% of the original data can be discarded. Figure 5 (c) shows the residual error between the original and the estimate. We can clearly see the random noise and some strange high frequency patterns (which are considered to



(a) (b) (c)

Figure 5. Results for the migrated seismic section: (a) Original seismic section with 128 traces and 256 time samples. (b) Estimation by the proposed method. (c) Residual error between (a) and (b). (Dynamic range of display (c) is different from those of (a) and (b).)

be numerical artifacts from the migration algorithm applied).

§8. Discussions

Our algorithm is intimately connected to the "denoising" algorithm of Coifman and Majid [7], [10]. Their algorithm first picks the best-basis from the collection of bases and sorts the best-basis coefficients in order of decreasing magnitude. Then they use the "theoretical compression rate" of the sorted best-basis coefficients $\{\alpha_i\}_{i=1}^{N}$ as a key criterion for separating a signal component from noise. The theoretical compression rate of a unit vector $\boldsymbol{u}$ is defined as $c(\boldsymbol{u}) = 2^{H(\boldsymbol{u})}/N(\boldsymbol{u})$, where $H(\boldsymbol{u})$ is the ℓ^2-entropy of $\boldsymbol{u}$, i.e., $H(\boldsymbol{u}) = -\sum_{i=1}^{N(\boldsymbol{u})} u_i^2 \log u_i^2$, and $N(\boldsymbol{u})$ is the length of $\boldsymbol{u}$. We note that $0 \le c(\boldsymbol{u}) \le 1$ for any real unit vector $\boldsymbol{u}$, and $c(\boldsymbol{u}) = 0$ implies $\boldsymbol{u} =$

$\{\delta_{i,i_0}\}$ for some i_0 (the best possible compression), and $c(\boldsymbol{u}) = 1$ implies $\boldsymbol{u} = (1, \ldots, 1)/\sqrt{N(\boldsymbol{u})}$ (the worst compression). Then to decide how many coefficients to keep as a signal component, they compare $c(\{\alpha_i\}_{i=k+1}^N)$, the theoretical compression rate of the noise component (defined as the smallest $(N-k)$ coefficients), to the predetermined threshold τ. They search $k = 0, 1, \ldots$ which gives an unacceptably bad compression rate: $c(\{\alpha_i\}_{i=k+1}^N) \geq \tau$. Their algorithm critically depends on the choice of the threshold τ whereas our algorithm needs no threshold selection. On the other hand, their algorithm does not assume the WGN model we used in this paper; rather, they *defined* the noise component as a vector reconstructed from the best-basis coefficients of small magnitude.

Our algorithm can also be viewed as a simple yet flexible and efficient realization of the "complexity regularization" method for estimation of functions proposed by Barron [3]. He considered a general regression function estimation problem: given the data $(x_i, y_i)_{i=1}^N$, where $\{x_i \in \mathbf{R}^p\}$ is a sequence of the (p-dimensional) sampling coordinates (or explanatory variables) and $\{y_i \in \mathbf{R}\}$ is the observed data (or response variables), select a "best" regression function $\widehat{f}_N$ out of a list (library) $\mathcal{L}_N$ of candidate functions (models). He did not impose any assumption on the noise distribution, but assumed that the number of models in the list $\mathcal{L}_N$ depends on the number of observations N. Now the complexity regularization method of Barron is to find $\widehat{f}_N$ such that

$$R(\widehat{f}_N) = \min_{f \in \mathcal{L}_N} \left(\frac{1}{N} \sum_{i=1}^{N} d(y_i, f(x_i)) + \frac{\lambda}{N} L(f) \right),$$

where $d(\cdot, \cdot)$ is a measure of distortion (such as the squared error), $\lambda > 0$ is a regularization constant, and $L(f)$ is a complexity of a function f (such as the $L(m) + L(\boldsymbol{\theta}_m \mid m)$ term in (5)). He showed that various asymptotic properties of the estimator $\widehat{f}_N$ as $N \to \infty$, such as bounds on the estimation error, the rate of convergence, etc. If we restrict our attention to the finite dimensional vector space, use the library of orthonormal bases described in Section 2, adopt the length of the Shannon code (6) as a distortion measure, assume the WGN model, and finally set $\lambda = 1$, then Barron's complexity regularization method reduces to our algorithm. Our approach, although restricted in the sense of Barron, provides a computationally efficient and yet flexible realization of the complexity regularization method, especially compared to the library consisting of polynomials, splines, trigonometric series discussed in [3].

Our algorithm also has a close relationship with the denoising algorithm via "wavelet shrinkage" developed by Donoho and Johnstone [16]. (A well-written summary on the wavelet shrinkage and its applications can be found in [15].) Their algorithm first transforms the observed discrete data into a wavelet basis (specified by the user), then applies a "soft threshold" $\tau = \sigma\sqrt{\ln N}$ to the coefficients, i.e., shrinks magnitudes of all the coefficients by the amount τ toward zero. Finally the denoised data is obtained by the inverse wavelet transform. Donoho claimed informally in [15] that the reason why their method works is the ability of wavelets to compress the signal energy into a few coefficients. The main differences between our algorithm and that of Donoho and Johnstone are:

- Our method automatically selects the most suitable basis from a collection of bases whereas their method uses only a *fixed* basis specified by the user.
- Our method includes adaptive expansion by means of wavelet packets and local trigonometric bases whereas their method only uses a wavelet transform.
- Their method requires the user to set the coarsest scale parameter $J \leq n$ and a good estimate of σ^2, and the resulting quality depends on these parameters. On the other hand, our method does not require any such parameter setting.
- Their approach is based on the minimax decision theory in statistics and addresses the risk of the estimation whereas our approach uses the information-theoretic idea and combines denoising and the data compression capability of wavelets explicitly.
- Their method thresholds the coefficients *softly* whereas our method can be said to threshold *sharply*. This might cause some Gibbs-like effects in the reconstruction using our method.

Future extensions of this research are to: incorporate noise models other than Gaussian noise, extend the algorithm for highly nonstationary signals by segmenting them smoothly and adaptively, investigate the effect of sharp thresholding, and study more about the relation with the complexity regularization method of Barron as well as the wavelet shrinkage of Donoho-Johnstone.

§9. Conclusions

We have described an algorithm for simultaneously suppressing the additive WGN component and compressing the signal component in a dataset. One or more of the bases in the library, consisting of wavelets, wavelet packets, and local trigonometric bases, compress the signal component quite well, whereas the WGN component cannot be compressed efficiently by any basis in the library. Based on this observation, we have tried to estimate the "best" basis and the "best" number of terms to retain for estimating the signal component in the data using the MDL criterion. Both synthetic and real field data examples have shown the wide applicability and usefulness of this algorithm.

Acknowledgements

The author would like to thank Prof. R. Coifman and Prof. A. Barron of Yale University for fruitful discussions.

References

1. Ahmed, N. and K. R. Rao, *Orthogonal Transforms for Digital Signal Processing*, Springer-Verlag, New York, 1975.
2. Auscher, P., G. Weiss, and M. V. Wickerhauser, Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets, in *Wavelets: A Tutorial in Theory and Applications*, C. K. Chui (ed.), Academic Press, San Diego, 1992, 237–256.
3. Barron, A. R., Complexity regularization with application to artificial neural networks, in *Proceeding NATO ASI on Nonparametric Functional Estimation*, Kluwer, 1991.
4. Barron, A. R. and T. M. Cover, Minimum complexity density estimation, *IEEE Trans. Inform. Theory*, **37** (4) (1991), 1034–1054.
5. Beylkin, G., R. Coifman, and V. Rokhlin, Fast wavelet transforms and numerical algorithms I, *Comm. Pure Appl. Math.*, **44** (1991), 141–183.
6. Bradley, J. N. and C. M. Brislawn, Image compression by vector quantization of multiresolution decompositions, *Physica D*, **60** (1992), 245–258.
7. Coifman, R. R. and F. Majid, Adapted waveform analysis and denoising, in *Progress in Wavelet Analysis and Applications*, Y. Meyer and S. Roques (eds.), Editions Frontieres, B.P.33, 91192 Gif-sur-Yvette Cedex, France, 1993, 63–76.
8. Coifman, R. R. and Y. Meyer. Remarques sur l'analyse de fourier à fenêtre, *Comptes Rendus Acad. Sci. Paris, Série I*, **312** (1991), 259–261.
9. Coifman, R. R. and M. V. Wickerhauser, Entropy-based algorithms for best basis selection, *IEEE Trans. Inform. Theory*, **38** (2) (1992), 713–719.
10. Coifman, R. R. and M. V. Wickerhauser, Wavelets and adapted waveform analysis, in *Wavelets: Mathematics and Applications*, J. Benedetto and M. Frazier (eds.), chapter 10, CRC Press, Boca Raton, Florida, 1993.
11. Cover, T. M. and J. A. Thomas, *Elements of Information Theory*, Wiley Interscience, New York, 1991.
12. Daubechies, I., Orthonormal bases of compactly supported wavelets, *Comm. Pure Appl. Math.*, **41** (1988), 909–996.
13. Daubechies, I., *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*, SIAM, Philadelphia, 1992.
14. DeVore, R. A., B. Jawerth, and B. J. Lucier, Image compression through wavelet transform coding, *IEEE Trans. Inform. Theory*, **38** (2) (1992), 719–746.
15. Donoho, D. L., Wavelet shrinkage and W.V.D.: a 10-minute tour, in *Progress in Wavelet Analysis and Applications*, Y. Meyer and S. Roques (eds.), Editions Frontieres, B.P.33, 91192 Gif-sur-Yvette Cedex, France, 1993, 109–128.
16. Donoho, D. L. and I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, preprint, Dept. of Statistics, Stanford University, Stanford, CA, Jun. 1992, revised Apr. 1993.

17. Gröchenig, K. and W. R. Madych, Multiresolution analysis, Haar bases, and self-similar tilings of R^n, *IEEE Trans. Inform. Theory*, **38** (2) (1992), 556–568.
18. Kovačević, J. and M. Vetterli, Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for R^n, *IEEE Trans. Inform. Theory*, **38** (2) (1992), 533–555.
19. Leclerc, Y. G., Constructing simple stable descriptions for image partitioning, *Intern. J. Computer Vision*, **3** (1989), 73–102.
20. Mallat, S., A theory for multiresolution signal decomposition, *IEEE Trans. Pattern Anal. Machine Intell.*, **11** (7) (1989), 674–693.
21. Mallat, S. and Z. Zhang, Matching pursuit with time-frequency dictionaries, *IEEE Trans. Signal Processing, the special issue on wavelets and signal processing*, **41** (12) (1993), to appear.
22. Meyer, Y., *Wavelets: Algorithms and Applications*, translated and revised by R. D. Ryan, SIAM, Philadelphia, PA, 1993.
23. Meyer, Y., *Wavelets and Operators*, volume 37 of *Cambridge Studies in Advanced Mathematics*, translated by D. H. Salinger, Cambridge University Press, New York, 1993.
24. Niblack, W., *MDL Methods in Image Analysis and Computer Vision*, IEEE Conf. Comput. Vision, Pattern Recognition, Tutorial note, New York, Jun. 1993.
25. Rioul, O., Regular wavelets: a discrete-time approach, *IEEE Trans. Signal Processing, the special issue on wavelets and signal processing*, **41** (12) (1993), to appear.
26. Rioul, O. and M. Vetterli, Wavelets and signal processing, *IEEE SP Magazine*, **8** (4) (1991), 14–38.
27. Rissanen, J., A universal prior for integers and estimation by minimum description length, *Ann. Statist.*, **11** (2) (1983), 416–431.
28. Rissanen, J., Universal coding, information, prediction, and estimation, *IEEE Trans. Inform. Theory*, **30** (4) (1984), 629–636.
29. Rissanen, J., *Stochastic Complexity in Statistical Inquiry*, World Scientific, Singapore, 1989.
30. Shapiro, J. M., Image coding using the embedded zerotree wavelet algorithm, in *Proc. SPIE Conf. on Mathematical Imaging: Wavelet Applications in Signal and Image Processing*, A. F. Laine (ed.), volume 2034, 1993, 180–193.
31. Wallace, R. S., *Finding natural clusters through entropy minimization*, PhD thesis, School of Comput. Science, Carnegie Mellon University, Pittsburgh, PA 15213, Jun. 1989.
32. Wax, M. and T. Kailath, Detection of signals by information theoretic criteria, *IEEE Trans. Acoust., Speech, Signal Processing*, **ASSP–33** (2) (1985), 387–392.
33. Wickerhauser, M. V., Lectures on wavelet packet algorithms, preprint, Dept. of Mathematics, Washington University, St. Louis, Missouri, Nov. 1991.
34. Wickerhauser, M. V., Fast approximate factor analysis, in *Proc. SPIE Conf. on Curves and Surfaces in Computer Vision and Graphics II*, volume 1610, 1991, 23–32.
35. Wickerhauser, M. V., High-resolution still picture compression, *Digital Signal Processing: A Review Journal*, **2** (4) (1992), 204–226.
36. Wilson, R., Finite prolate spheroidal sequences and their applications I: generations and properties, *IEEE Trans. Pattern Anal. Machine Intell.*, **PAMI-9** (6) (1987), 787–795.

Naoki Saito
Schlumberger-Doll Research
Old Quarry Road, Ridgefield, CT 06877
and
Department of Mathematics
Yale University
10 Hillhouse Avenue, New Haven, CT 06520
saito@ridgefield.sdr.slb.com