Genetic Algorithms for Component Analysis

Adelino R. Ferreira da Silva
Universidade Nova de Lisboa
2825 Monte de Caparica
Portugal
afs@mail.fct.unl.pt
Phone:351.212948338

Abstract

Adapted waveform analysis is used extensively in audio, speech, and video coding. In this context, the basic problem is to find out the best representation for a given signal. The strategy is to decompose the signal into time-frequency atoms. The main analysis tool is the expansion of the signal in orthonormal bases whose elements have good time-frequency localizations. We show that genetic algorithms can be exploited to search libraries of bases for atomic pattern recognition in noisy signals. The proposed algorithm generates best basis decomposition trees through the evolution of well-adapted genetic sequences. Two new types of constraint operators are introduced to guarantee that valid genetic sequences are generated. The approach behaves as an atomic analyzer for signal processing. A test environment is proposed to assess recognition of time-frequency atoms in signal spaces.

1 Introduction

One of the main goals of signal analysis in recent years has been the development of signal representations in terms of elementary waveforms well localized in time and frequency, called time-frequency atoms. Adapted waveform analysis is used extensively, in audio, speech, and video coding. The objective is to find out the best representation for a given signal, and the strategy is to decompose a signal into time-frequencies atoms. The time-frequency plane is a two-dimensional space useful for idealizing two important properties associated to transient signals: localization in time of transient phenomena, and presence of specific frequencies. A signal may be represented in this plane in a number of ways. Each of the elementary waveforms resides mostly in a well-defined area in the time-frequency plane. A waveform is represented by a rectangle in this plane with its sides parallel to the time and frequency axes. Such a rectangle is often referred to as an information cell. The amplitude of a waveform can be encoded by darkening the rectangle in proportion to its waveform's energy. The idealized time-frequency plane closely resembles a musical score, and the information cells play the role of notes. The main signal analysis tool is the expansion of the signal in orthonormal bases whose elements have good time-frequency localizations. Features in this context are just the basis elements which contribute large components to the expansion; they are detectable from their size. When we find a large component, we can mark the time-frequency location of its basis element to build a time-frequency picture of the analyzed signal. Currently, two main classes of time-frequency atoms are in use. The first, wavelet packets, splits the signal first in frequency and then in time, whereas the second, local cosine packet bases, does the opposite,i.e., it slices first in time and then in frequency.

In this paper, we show how genetic algorithms can be used for component analysis. By component analysis we mean a methodology for decomposing a signal into an "optimal" superposition of dictionary elements. Adapted waveform analysis uses libraries of orthonormal basis and an efficient functional to match a basis to a given signal or family of signals. The evolutionary computation approach permits to search larger libraries of bases for best basis selection than those considered in current signal processing approaches. A test environment was devised to assess recognition of well-adapted time-frequency atoms in signal spaces. Several classes of synthetic applications of increasing complexity are introduced, in which noisy signals are constructed from atoms randomly generated from collections of waveforms. The idealized time-frequency plane was selected to visualize atomic signal decomposition [Wickerhauser, 1994]. The proposed test environment substantiates the claim that time-frequency diagrams and residual error measures can be used as a new kind of fitness landscape to assess the comparative performance of genetic algorithms. The genetic algorithm uses variable length integer sequences as the basic genotype. In evolutionary terms, the objective is to select best basis decomposition trees through the evolution of well-adapted genetic sequences in terms of some suitable information criterion. A well-built decomposition tree for adaptive analysis purposes is generated by imposing appropriate constraints to the genetic sequences. Two new types of constraint operators are introduced to guarantee that valid genetic sequences are generated. In addition, these operators produce a more uniform sampling of the tree search space than other similar operators. A brief review of fundamental results in wavelet theory and time-frequency analysis is presented in section 2. The evolutionary formulation and model specification is presented in section 3. The application of the genetic program to signal decomposition and atomic recognition is presented in section 4. Finally, some conclusions are offered in section 5.

2 Time-Frequency Analysis

2.1 Wavelets and Wavelet Packets

A given function f can be represented in terms of orthonormal basis functions. The family of orthonormal basis functions is generated from a single function ψ, called mother wavelet. The orthogonal basis of compactly supported wavelets of $\mathbb{L}_2$ is formed by the dilation and translation of a function $\psi(x)$ [Wickerhauser, 1994],

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k),$$

where $j, k \in \mathbb{Z}$. The function has a companion, the scaling function $\phi(x)$, and these functions satisfy the following relations:

$$\phi(x) = \sqrt{(2)}\sum_{k=0}^{L-1} h_k \phi(2x - k), \tag{1}$$

$$\psi(x) = \sqrt{(2)}\sum_{k=0}^{L-1} g_k \phi(2x - k), \tag{2}$$

where $g_k = (-1)^k h_{L-k-1}$, $k = 0, 1, \ldots, L-1$, and $\int_{-\infty}^{+\infty} \phi(x)dx = 1$. The wavelet basis induces a multiresolution analysis on $\mathbb{L}_2(\mathbb{R})$, i.e., the decomposition of the Hilbert space into a chain of closed subspaces

$$\ldots \subset \mathcal{V}_2 \subset \mathcal{V}_1 \subset \mathcal{V}_0 \subset \mathcal{V}_{-1} \subset \mathcal{V}_{-2} \ldots .$$

On each fixed scale j, the wavelets $\{\psi_{j,k}(x)\}_{k\in\mathbb{Z}}$ form an orthonormal basis of $\mathcal{W}_j$ and the functions $\{\phi_{j,k}(x) = 2^{-j/2}\phi(2^{-j}x - k)\}_{k\in\mathbb{Z}}$ form an orthonormal basis of $\mathcal{V}_j$. The coefficient $H = \{h_k\}_{k=0}^{k=L-1}$ and $G = \{g_k\}_{k=0}^{k=L-1}$ in (1) and (2) are quadrature mirror filters. Once the filter H has been chosen, it completely determines the functions ψ and ϕ. Discrete wavelet transformations map data from the time domain (the original or input data vector) to the wavelet domain. The discrete wavelet transformation may be summarized by the well-known pyramid scheme,

$$\begin{tikzcd}
\{s_k^0\} \arrow[r] \arrow[dr] & \{s_k^1\} \arrow[r] \arrow[dr] & \{s_k^2\} \arrow[r] \arrow[dr] & \{s_k^3\} \ldots \\
 & \{d_k^1\} & \{d_k^2\} & \{d_k^3\} \ldots
\end{tikzcd} \tag{3}$$

where the coefficients s_k^0 for $k = 1, 2, \ldots, N$ are given by

$$s_k^j = \sum_{n=0}^{n=L-1} h_n s_{n+2k-1}^{j-1},$$

$$d_k^j = \sum_{n=0}^{n=L-1} g_n s_{n+2k-1}^{j-1}$$

and s_k^j and d_k^j are periodic sequences with the period $2^{n-j}, j = 0, \ldots, n$. In the pyramid scheme, on each scale j we compute one vector of differences $\{d_k^j\}_{k=1}^{k=2^{n-j}}$ and one vector of averages $\{s_k^j\}_{k=1}^{k=2^{n-j}}$. Wavelet packets are linear combinations of wavelet functions and represent a powerful generalization of standard orthonormal wavelet bases. A wavelet packet basis of $\mathbb{L}_2$ is an orthogonal basis selected from the library of packet functions. Each basis is indexed by a subset of indices: j - the scaling parameter, n - the sequence parameter, and k - the translation parameter. Two examples of mother wavelets of the Daubechies-8 and Antonini families respectively, are shown in Figure 1. Atoms of a given family are just rescaled and translated versions of the respective mother wavelet.

2.2 Extended Wavelet Packets and Local Cosine Packets

One well-known disadvantage of the discrete wavelet and wavelet packet transforms is the lack of shift invariance. Recently, several authors proposed independently to extend the library of bases in which the best representations are searched for, by introducing additional degrees of freedom that adjust the time-localization of the basis functions [Liang and Parks, 1996]. One such extension is the shift-invariant wavelet packet transform. The added dimension in the case of shift-invariant decompositions

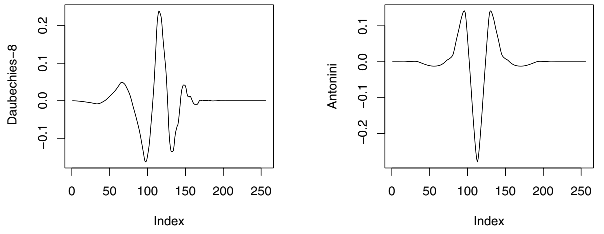


Figure 1: Daubechies-8 (left) and Antonini (right) wavelets

is a *relative shift*, between a given parent-node and its relative children nodes. Shifted versions of these transforms for a given input signal, represent new bases to be added to the library of bases, which may further improve our ability to find the ”best” adapted basis. These modifications of the wavelet transform and wavelet packet decompositions, lead to orthonormal best basis representations which are shift-invariant and characterized by lower information cost functionals.

Localized cosine functions can also be combined into a library of orthonormal bases [Wickerhauser, 1994]. The bases consist of cosines multiplied by smooth, compactly supported cutoff functions. Elements of the library are windowed bases and we adapt the window sizes by choosing an appropriate window-determining partition of the line. As in the case of wavelet packets, this library can be made into a tree. Two adjacent intervals I and J span the space of windowed functions over $I \cup J$. Wavelet packets as well as local cosine packets libraries are searchable for the ”best basis”. The importance of best basis lies mainly in the fact that they yield parsimonious representations of waveforms or $\mathbb{L}_2$ functions. An important question is how to select from a large number of bases in the dictionary a basis which performs ”best” for one's task. In order to measure the performance of each basis, we need a measure of efficiency or fitness of a basis for the task at hand. For this purpose, several so-called *information cost functionals* have been proposed [Wickerhauser, 1994]. The following *Minimum Description Length* (AMDL) information cost functional [Vidakovic, 1999] was used in the applications described in section 4,

$$AMDL(k,m) = \frac{3k}{2} log_2 n + \frac{n}{2} log_2 \parallel d_m - d_m^* \parallel^2,$$

where n is the sample size, d_m is the vector of coefficients in the basis B_m, and d_m^* is the hard-thresholded vector d_m (see section 2.3) with exactly k non-zero coefficients.

2.3 Denoising and Regression

In the following we will reference the standard wavelet regression procedure used in signal denoising [Donoho and Johnstone, 1995]. Similar considerations apply to signals from local cosine packets dictionaries. Suppose the function f is sampled at $n = 2^M$ equally spaced points, but is observed with additive white noise, $y_i = f(i/n) + \sigma\epsilon_i, i = 0, 1, \ldots, n-1$, where ϵ_i, $i = 0, 1, \ldots, n-1$, are iid standard normal random variables, and σ is unknown. For estimation of f, we use the usual ℓ_n^2 norm to evaluate performance. Let $f = \{f(x_i)\}_{i=1}^n$ and $\hat{f} = \{\hat{f}(x_i)\}_{i=1}^n$ be the vectors of true and estimated function values. Performance is measured by the mean squared error,

$$R(\hat{f}, f) = \frac{1}{n} \parallel \hat{f} - f \parallel_{2,n}^2 = \frac{1}{n} \sum_{i=1}^{n} [\hat{f}(x_i) - f(x_i)]^2.$$

Our goal is to find the estimate $\hat{f}$ that minimizes this risk. The procedure to calculate $\hat{f}$ involves three steps: wavelet decomposition, thresholding, and reconstruction. After the noisy signal y is decomposed, an appropriate thresholding value t is needed to reduce the noise, such that after reconstruction from the thresholded coefficients, the corresponding $\hat{f}$ makes $R(\hat{f}, f)$ as small as possible. Given a wavelet coefficient w and a thresholded value t, the hard-thresholded T_h is defined by, $T_h(w,t) = wI(\mid w \mid > t)$, where I is the usual indicator function. Thus ”hard” means ”keep or kill”.

3 Evolutionary Formulation

In an orthogonal adapted waveform analysis, the user is provided with a collection of standard libraries of waveforms or dictionaries (wavelets, wavelet packets and local cosine packets basis), which can be combined to fit specific classes of signals. Libraries of bases represent the population from which we want to select the best-fit individuals. The proposed approach generates a population of solutions based on basis expansions in two basic dictionaries: wavelet packets and local cosine packets. The genetic algorithm operates on the population to evolve the best solution according to an objective function specified in terms of the AMDL functional. A steady state algorithm using overlapping populations was used to drive the optimization process [Wall, 1996].

3.1 Representation

A variable length integer sequence was used as the basic genotype. The technique used to initialize the population is based on generating an initial random integer sequence, according to the values of the allele sets specified for the individual genes [Wall, 1996, Goldberg, 1989]. The initial genotype sequence which codifies the wavelet tree matches the complete breadth-first (BF) sequence required to generate a complete binary tree, up to a pre-specified maximum depth. We refer to these sequences as tree-mapped sequences. A well-built decomposition tree for wavelet analysis purposes, is generated by imposing appropriate constraints to the genotype sequence as specified in Sect. 3.2. The imposition of the constraints yields variable length code sequences after resizing. An alphabet $\mathbb{A} = \{0, 1, 2\}$ is used to codify the wavelet tree nodes according to their types as specified in Sect. 3.3, thus enabling us to map any tree structure into a code sequence. The mapping of a code sequence to a complete BF tree traversal yields an initial sequence with length $l = 2^L - 1$, for a tree of depth L. The length l is also the number of nodes in a complete binary tree of depth L. When coding a complete binary tree using a complete BF sequence, the last level of terminal nodes is redundant and may be omitted. In addition, after the application of the constraints complete BF sequences may be pruned, yielding shorter genetic sequences. The chromossomes are constructed as follows. The first gene assumes integer values $g_0 \in \mathbb{F}$, where $\mathbb{F}$ is the set of possible filter types used in the implementation. The remaining genes are used to codify the wavelet decomposition tree [da Silva, 1999].

3.2 Constraints

There are several methods for generating trees which can be used to initialize the population. The *full*, *grow* and *ramped half-and-half* methods of tree generation were introduced in the field of genetic programming [Koza, 1992]. These methods are based on tree depth. The ramped half-and-half method is the most commonly used method of generating random parse trees because of its relative higher probability of generating subtrees of varying depth and size. However, these methods do not produce a uniform sampling of the search space. We defined new constrained genetic operators to correct the application of standard genetic operators that initialize or modify genetic sequences. Two types of constraint operators have been used to guarantee that valid tree-mapped genetic sequences are generated: (1) top-down operator, and (2) bottom-up operator. In addition, by applying these operators we look for a uniform sampling of the tree search space. In terms of binary tree data structures, the top-down constraint guarantees that if a node t_i has null code $c_i = 0$ then its two sons $t_{i,0}$ and $t_{i,1}$ must have null code $c_{i,0} = 0$ and $c_{i,1} = 0$. The bottom-up constraint guarantees that if at least one of the sons $t_{i,0}$ and $t_{i,1}$ of a node t_i has non-null code, then the parent $t_{i,0}$ must have non-null code $c_i \neq 0$. The null code references a terminal node. These constraint operators are biased in opposite ways. Starting from a uniform random code sequence, the bottom-up constraint operator constructs valid genetic sequences which are biased towards complete, full-depth trees. By the same token, the top-down constraint operator constructs valid genetic sequence which is biased towards null, minimum depth trees. To get a more uniform sampling of the sequence space for sequences of maximum length $s = 2^L - 1$, we used an initialization procedure which randomly combines the operation of those two constraint operators [da Silva, 2000]. By resizing (pruning) constrained code sequences we allow for genetic sequences of variable length, hence tree representations of variable depth.

3.3 Specification

The collection of wavelet packets comprises a library of functions with a binary tree structure. To obtain the wavelet packet analysis of a function, or data set in the discrete case, we first find its coefficient sequence in the root subspace, then follow the branches of the wavelet packets tree to find the expansion in the descendent subspaces. Assigning to each tree node a wavelet split value $s_i \in \{0, 1, 2\}$ we may enumerate all possible binary tree structures. The value $s_w = 1$ references unshifted interior nodes, i.e., nodes with left and right children subtrees associated with unshifted decompositions. The value $s_w = 2$ references time shifted interior nodes, i.e, shifted transforms. The value $s_w = 0$ references terminal nodes. This tree codification is often called a joint tree, since it guides the process of generating not only wavelet packet transforms but also shifted wavelet packet transforms [da Silva, 1999]. Therefore, the approach of organizing libraries of bases as a tree has been extended to construct joint decomposition trees for enlarged basis collections. Similar considerations apply to the construction of local cosine packet trees. Libraries of bases represent the population from which we want to select the best-fit individuals. The genotype sequence $\mathcal{G}$ allows for three optimization parameters: best filter, best wavelet packet basis and best shifted basis. The genetic representation is used to create an initial population and evolve potential solutions to the optimization problem. The genotype is made up of the genes which guide the discrete wavelet or local cosine decomposition of each waveform, in accordance with the joint tree representation. A cost functional is then applied to the coefficients, and its value is used to derive the fitness of the individual. In terms of entropy, the optimization problem amounts to evolve a minimum-entropy genotype. Using entropy, the best individual would be the one with minimum evolved entropy in a given library space. The first gene g_0 in $\mathcal{G}$ is responsible for the optimization of the filter used in the decomposition. We have used in the implementation 16 possible types of different filters, thus $g_0 = \{0, \ldots, 15\}$. The filters considered in the implementation were the Haar filter, the Daubechies filters D4, D6 and D8, and several biorthogonal filters commonly used in image analysis as implemented in [Davis, 1997]. The analysis phase of the (discrete) shift wavelet packet transform is codified in the genetic sequence $\mathcal{G}$.

4 Atomic Recognition

We are interested in finding the best representation for a given signal by decomposing a signal into elementary waveforms well localized in time and frequency, called time-frequency atoms. A given signal is reproduced, up to a certain approximation, by summing up the individual atoms. We consider three transformed spaces from which the most important atomic coefficients for representing a given signal can be selected: wavelet packets, multi-filter shifted wavelet packets, and local cosine spaces. Once we have selected an atomic coefficient in a signal transformed space, the application of the appropriate inversion transform according to the coefficient dictionary generates an atom of the analyzing signal. We provide simulation examples of noisy synthetic signals to substantiate the estimation capabilities of the evolutionary computation for signal component analysis. A synthetic signal is generated by randomly selecting from a dictionary, or collection of dictionaries, a specified number of atoms and summing them up. By analogy with other science fields, we will call such a synthetic signal a *molecule* or molecular signal. This molecular signal is then corrupted by gaussian noise with a user-specified signal-to-noise (snr) ratio. The objective of the algorithm is to correctly identify the atoms presented in the randomly generated noisy molecule. The idealized time-frequency plane is a suitable tool to visualize atomic signal decomposition and atomic recognition. An atomic coefficient is represented by a single information cell in this plane with its sides parallel to the time and frequency axes. The amplitude of the atomic coefficient is encoded by darkening the information cell in proportion to the coefficient's energy. Different partitions of the plane produce information cells with different features, in terms of their shape and localization in the plane. By the same token, different information cell's features generate atomic waveforms with different signal adaptation capabilities.

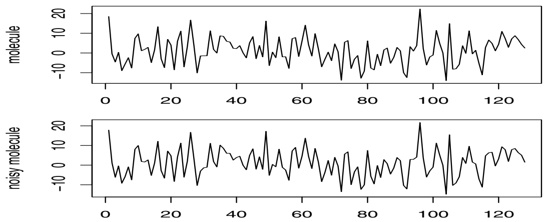


Figure 2: Molecular test signal (top), and its noisy counterpart (bottom)

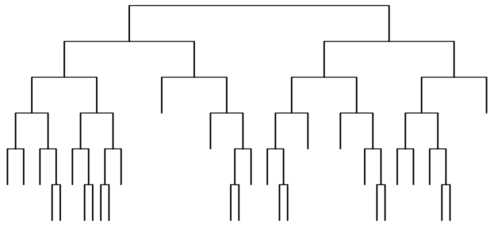
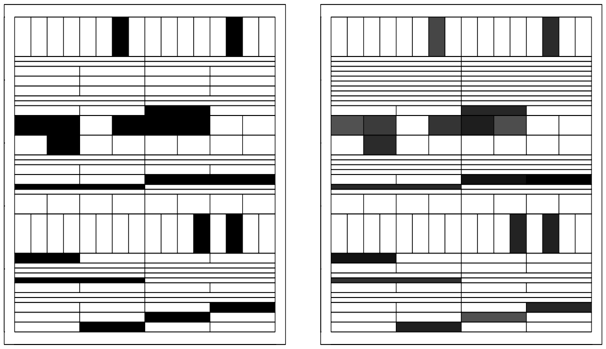


Figure 3: Test basis tree (class 1)

4.1 Class 1: recognition in wavelet packet dictionaries

Figure 2 depicts the molecular test signal generated by composition of 20 atomic waveforms randomly selected from the wavelet packets dictionary. A dis-



(a) Test

(b) Evolved

Figure 4: Time-frequency decompositions (class 1)

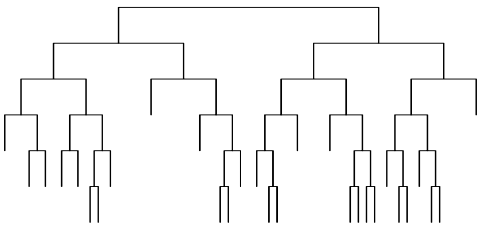


Figure 5: Evolved basis tree (class 1)

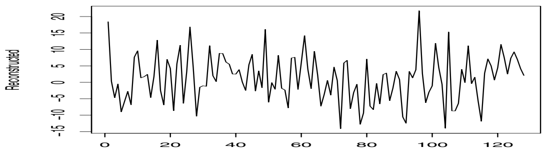


Figure 6: Reconstructed signal (class 1)

crete signal size $N = 128$, and the Antonini filter [Antonini et al., 1992] were used in this test signal. A $snr = 7$ was applied to the test signal to produce the noisy test signal counterpart, also depicted in Figure 2. Here, the signal-to-noise ratio is defined as the ratio of standard deviations of signal and noise. The time-frequency representation of the molecular test signal f (without noise) is represented in Figure 4(a). This representation is associated with the basis tree of Figure 3. The atomic coefficients' energy was normalized to 1. The evolutionary program was applied to the noisy test signal to select the best basis representation. The information cost measure AMDL was used in fitness evaluation. A hard threshold with universal cost measure $\sqrt{2logN}$ [Donoho and Johnstone, 1995] was applied to the wavelet packet coefficients produced by the best evolved basis. The thresholded coefficients were used to draw the time-frequency representation of the evolved signal in the transformed space, as shown in Figure 4(b). This figure shows that all atomic coefficients have been correctly identified, even though the evolved basis tree (Figure 5) is different from the original one. The 20 evolved atoms produced the reconstructed signal $\hat{f}$ represented in Figure 6. Notice that the cells in the diagram of Figure 4(b) have energy (color) close to the energy of the corresponding cells in the test diagram. The estimation process needs to filter the noise introduced in the original signal to get a closer representation of the original signal. The snr value affects the precision of the identication process itself.

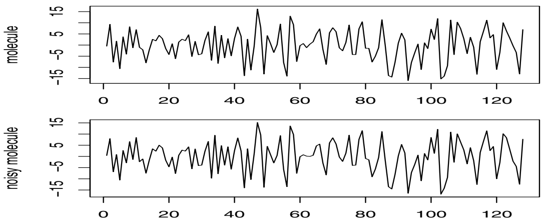
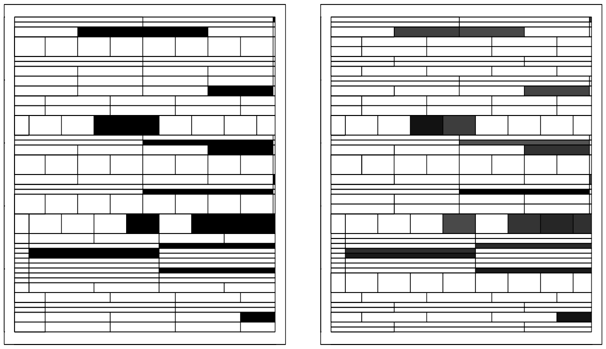


Figure 7: Molecular test signal (top), and its noisy counterpart (bottom) (class 2)



(a) Test

(b) Evolved

Figure 8: Time-frequency decompositions (class 2)

4.2 Class 2: recognition in multifilter, shifted wavelet packet dictionaries

A test procedure similar to the one described in section 4.1 was applied to signals randomly synthesized from a multifilter shifted wavelet dictionary. In this case, in addition to the best wavelet packet basis, the evolved genome has to correctly "guess" the best filter to use, and the best shift to apply at each wavelet packet tree node. The number of atoms used in the test signal, as well as the snr value and the information cost measure were the same as those referred to in section 4.1. Figures 7 through 10, have interpretations similar to the ones presented in section 4.1. In Figure 9, shifted wavelet packet transforms at a given transformation step are represented by thicker lines in the basis decomposition tree. Thinner lines represent unshifted wavelet packet transforms. By the same token, Figures 8(a) and 8(b) allow for the representation of time-shifted partitions of the time-frequency plane. As in section 4.1 the evolutionary program was able to correctly recognize the 20 atomic coefficients in the test signal.

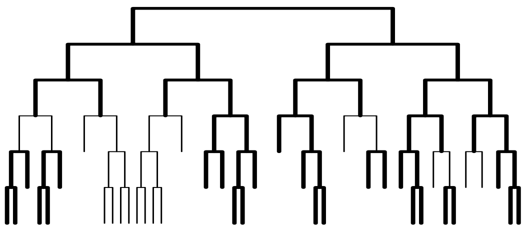


Figure 9: Evolved time-shifted basis tree (class 2)

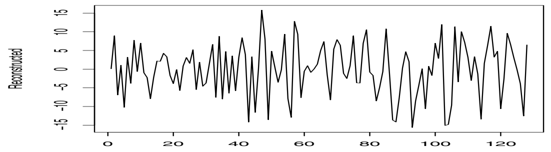


Figure 10: Reconstructed signal (class 2)

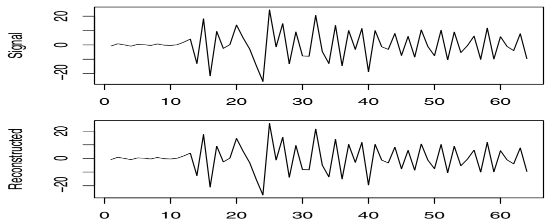
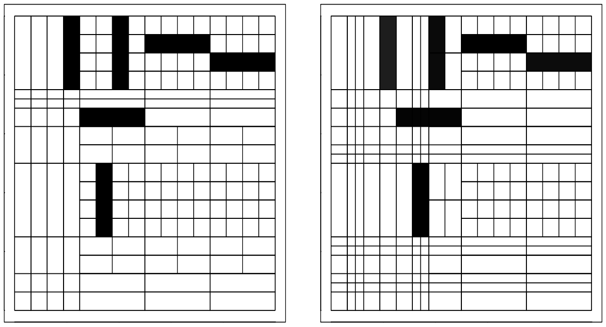


Figure 11: Composite test signal (top), and reconstructed signal (bottom) (class 3)



(a) Test

(b) Evolved

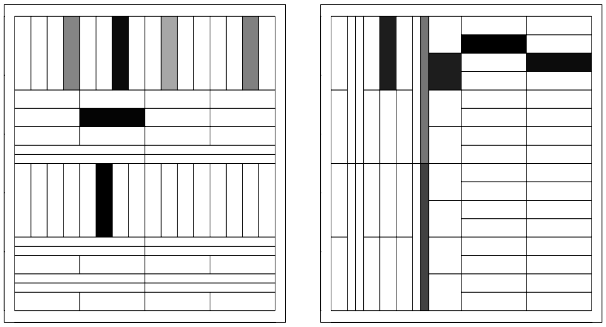
Figure 12: Composite time-frequency decompositions (class 3)

4.3 Class 3: recognition with multiple dictionaries

As a final example, we consider atoms taken from two dictionaries: wavelet packets and local cosine packets. To generate the test signal we took randomly six atoms, three from each of the above two dictionaries. The synthetic molecular signal with $N = 64$ and $snr = 7$ is represented in the top panel of Figure 11. The composite time-frequency diagram for the composite test signal is depicted in Figure 12(a), by graphical superposition of two time-frequency diagrams: wavelet packets and local cosine packet time-frequency diagrams. In this case, both the wavelet packets dictionary and the local cosine packets dictionary alone were unable to identify all the atoms in the test signal. The best evolved time-frequency diagrams for the above two dictionaries when taken in isolation, are represented in Figures 13(a) and 13(b). Clearly, these diagrams do not match the diagram in Figure 12(a). To estimate the atoms in the composite molecule we evolved a composite genome, built from the best evolved genomes from the above two dictionaries. The evolved composite genome was then used to generate the composite time-frequency diagram shown in Figure 12(b). Six atoms were correctly identified. We can represent the contribution of each dictionary for the composite signal. Figure 14 (top picture) shows the wavelet part of the reconstructed signal, formed by the three atoms of the wavelet packets dictionary which were used to build up the reconstructed signal. Similarly, the bottom picture in Figure 14 depicts the local cosine component (three atoms' molecule) present in the evolved composite signal. The composite reconstructed signal $\hat{f}$ is shown in Figure 11 (bottom). Table 1 presents the ℓ_2 reconstruction errors $\| f - \hat{f} \|_2$ for the test signals used in section 4. For the composite signal of class 3, the reconstruction errors when a single dictionary was used in the estimation process are referenced by $3 - WP$ and $3 - CP$, for the wavelet packets and cosine packets dictionary, respectively.

5 Conclusions

The combination of elementary waveforms from multiple dictionaries is important for the proper analysis of complex signals. Traditional signal processing algorithms have difficulty in finding well-adapted signal representations in increasingly larger and overcomplete dictionaries. The proposed genetic algorithm extends the scope and enhances the flexibility of standard best basis algorithms in order to find better adaptive signal representations. A statistical analysis of the



(a) Wavelet packets

(b) Cosine packets

Figure 13: Time-frequency decompositions by dictionary (class 3)

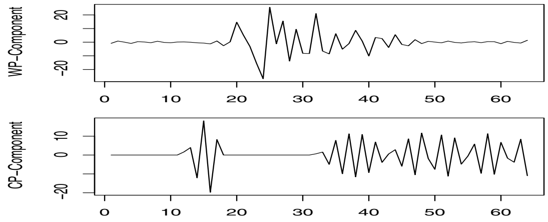


Figure 14: Reconstructed parts by dictionary (class 3)

component identification process may be conducted to define a statistical measure of the misidentification error rates for different signal classes. This is especially significant for analysis of complex signals derived from combinations of waveforms from multiple dictionaries. In our view, signal component analysis provides a suitable test environment for comparative performance of different algorithms. The following characteristics of this environment are worth noting. Synthetic test functions are easily generated. The complexity of the environment may be controlled by defining dictionaries and combination of dictionaries from which test signals may be drawn. The difficulty of the component analysis process may be controlled by defining appropriate signal-to-noise ratios. Visual tools and numeric error measures are available to assess performance. Statistical misidentification error criteria are readily available.

Table 1: Reconstruction errors

SIGNAL CLASS	ERROR
1	4.1
2	4.9
3	3.3
3-WP	55.5
3-CP	55.6

References

[Antonini et al., 1992] Antonini, M., Barlaud, M., Mathieu, P., and Daubechies, I. (1992). Image coding using wavelet transform. *IEEE Trans. on Image Process.*, 1(2).

[da Silva, 1999] da Silva, A. F. (1999). Evolving best-basis representations. In Angeline, P., editor, *Proceedings of the 1999 Congress on Evolutionary Computation, Washington D.C.*, pages 302–309. IEEE Press.

[da Silva, 2000] da Silva, A. F. (2000). Evolutionary wavelet bases in signal spaces. In *EvoIASP2000: Proceedings of the Second European Workshop on Evolutionary Computation in Image Analysis and Signal Processing, Edinburgh (to appear).*

[Davis, 1997] Davis, G. (1997). *Baseline Wavelet Transform Coder Constrution Kit.* Mathematics Department, Dartmouth College.

[Donoho and Johnstone, 1995] Donoho, D. L. and Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Statist. Assoc*, 90:1200–1224.

[Goldberg, 1989] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, Reading, Massachusetts.

[Koza, 1992] Koza, J. R. (1992). *Genetic Programming - On the Programming of Computers by Means of Natural Selection.* MIT Press, Cambridge, MA.

[Liang and Parks, 1996] Liang, J. and Parks, T. W. (1996). A translation-invariant wavelet representation algorithm with applications. *IEEE Trans. on Signal Process.*, 44(2):225–232.

[Vidakovic, 1999] Vidakovic, B. (1999). *Statistical Modeling by Wavelets.* John Wiley & Sons, New York.

[Wall, 1996] Wall, M. (1996). *GAlib: A C++ Library of Genetic Algorithm Components.* Mechanical Engineering Department, Massachusetts Institute of Technology.

[Wickerhauser, 1994] Wickerhauser, M. V. (1994). *Adapted wavelet analysis from theory to software.* A.K. Peters, Wellesley, Mass.