



Comparison of extrasystolic ECG signal classifiers using discrete wavelet transforms

Tom Froese^a, Sillas Hadjiloucas^{b,*}, Roberto K.H. Galvão^c,
Victor M. Becerra^b, Clarimar José Coelho^d

^a Department of Informatics, The University of Sussex, Brighton BN1 9QH, UK

^b Department of Cybernetics, The University of Reading, P.O. Box 225, Whiteknights Campus, Reading, Berkshire RG6 6AY, UK

^c Divisão de Engenharia Eletrônica, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP 12228-900, Brazil

^d Departamento de Ciência da Computação, Universidade Católica de Goiás, Goiânia, GO 74605-010, Brazil

Received 22 October 2004; received in revised form 4 August 2005

Communicated by Prof. H.H.S. Ip

Abstract

This work compares and contrasts results of classifying time-domain ECG signals with pathological conditions taken from the MIT-BIH arrhythmia database. Linear discriminant analysis and a multi-layer perceptron were used as classifiers. The neural network was trained by two different methods, namely back-propagation and a genetic algorithm. Converting the time-domain signal into the wavelet domain reduced the dimensionality of the problem at least 10-fold. This was achieved using wavelets from the db6 family as well as using adaptive wavelets generated using two different strategies. The wavelet transforms used in this study were limited to two decomposition levels. A neural network with evolved weights proved to be the best classifier with a maximum of 99.6% accuracy when optimised wavelet-transform ECG data was presented to its input and 95.9% accuracy when the signals presented to its input were decomposed using db6 wavelets. The linear discriminant analysis achieved a maximum classification accuracy of 95.7% when presented with optimised and 95.5% with db6 wavelet coefficients. It is shown that the much simpler signal representation of a few wavelet coefficients obtained through an optimised discrete wavelet transform facilitates the classification of non-stationary time-variant signals task considerably. In addition, the results indicate that wavelet optimisation may improve the classification ability of a neural network.

© 2005 Elsevier B.V. All rights reserved.

Keywords: ECG; Discrete wavelet transform; Neural networks; Genetic algorithms; Linear discriminant analysis

1. Introduction

We discuss the classification of extrasystolic heart beats, which are beats that have occurred prematurely within the cardiac cycle. When the source of the arrhythmias is localized in the upper part (atria) of the heart it is referred to as atrial, whereas when it is localized in the lower part (ventricles) of the heart they are referred to as of the ventricular type (Rangayyan, 2001). The classification is carried out

on the basis of the QRS complex, the electrical signature of ventricular contraction, which is the most prominent feature of the electrocardiographic (ECG) signal. Atrial beats usually have a normal QRS morphology and aberrated atrial beats are rare because the propagation of the electrical impulse proceeds normally after it reaches the ventricles. However, some alterations in the QRS morphology may be observed along an ECG record as a result of occasional abnormalities in the electrical conduction system of the heart. The ventricular beats, on the other hand, show a large morphological variability. Their shape depends upon the region in the ventricle where the beat was actually triggered (Galvão and Yoneyama, 2004).

* Corresponding author. Tel.: +44 1189316787; fax: +44 1189318220.
E-mail address: s.hadjiloucas@reading.ac.uk (S. Hadjiloucas).

Examples of both types of extrasystolic beats are shown in Fig. 1. Atrial extrasystoles may occur frequently, even in normal persons, without serious health consequences. In contrast, ventricular extrasystoles have severe implications in heart patients, because they may trigger life-threatening arrhythmias and ultimately ventricular fibrillation with consequent cardiac arrest. In intensive care units, alarms may be set to ring if the number of ventricular extrasystoles per minute exceeds a certain threshold (Jacobson and Webster, 1977).

This work compares and contrasts three different classifiers, (a) an artificial neural network (ANN) trained with back-propagation, (b) an ANN where a genetic algorithm (GA) was used to evolve the weights and (c) a linear discriminant analysis (LDA) classifier. These methods are representative of parametric (LDA) and non-parametric (ANN) approaches to the design of classification laws. It is worth noting that the use of GAs in this context is motivated by one of the main difficulties involved in ANN training, namely the existence of several local minima in the cost function.

Although it is possible to assign one ANN input for every point in the ECG time-domain sequence and perform classification using a standard multi-layer perceptron (MLP), the resulting architecture would be exceedingly large. Alternatively, one could consider using a recurrent MLP with only one input and present to it the entire time-domain sequence sequentially. The resulting feedback between the layers, however, would complicate the training process. Alternative training methods e.g., Angeline et al. (1994) should therefore, be used instead of the standard back-propagation procedure. Of further concern is the fact that the application of feedback to the neural network can cause a system that is originally stable to become unstable (Haykin, 1999).

Furthermore, a reduction of the input feature space to fewer dimensions is clearly desirable in order to simplify and improve the classification procedure (Duda et al., 2001; Kudo and Sklansky, 2000). The relevance of reducing the input dimensionality can be more easily illustrated in

the context of parametric classification strategies, in which the training of the classifier consists of estimating a parameter vector θ from a given set of M objects $D = \{x_m, m = 1, 2, \dots, M\}$ of known categories. For instance, in linear discriminant analysis, θ comprises the elements of the mean vectors of each class ($\mu_1, \mu_2, \dots, \mu_C$), as well as the elements of the common covariance matrix Σ . If the number M of training samples is not sufficiently large, as compared to the number of parameters to be estimated, the estimation may become ill-conditioned and overfitting problems in the determination of the decision surfaces are likely to occur (Tabachnick and Fidell, 2001).

If the object to be classified is described by a discrete-time signal, a reduction in dimensionality can usually be achieved by exploiting the autocorrelation properties of the signal. Such a reduction can be achieved by using the power spectral density (PSD) of the signal, for instance, because the autocorrelation between adjoining time samples causes the PSD to be concentrated in low frequencies. However, the PSD may not be the best dimensionality reduction technique when the signal has distinctive features localized in time (such as sharp transitions and peaks), because these features are associated to harmonic components with a broad frequency range. In such cases, joint time–frequency analysis tools may be more appropriate (Qian and Chen, 1996). In this context, the use of multi-resolution signal processing, such as the filter bank implementation of the discrete wavelet transform (Daubechies, 1992; Strang and Nguyen, 1996) shown in Fig. 2 has become widespread. In particular, electrocardiogram (ECG) signals, which are non-stationary in the time-domain, can be represented well in compressed form without significant loss of information using the wavelet transform (WT) (Addison, 2002).

This paper is organized as follows. In Section 2 we discuss different ways of transforming the time-domain ECG signals into the wavelet domain using the discrete WT. Two different wavelet optimisation strategies are considered. The differences between these adaptive wavelets and discrete wavelets from the db6 family are highlighted. All

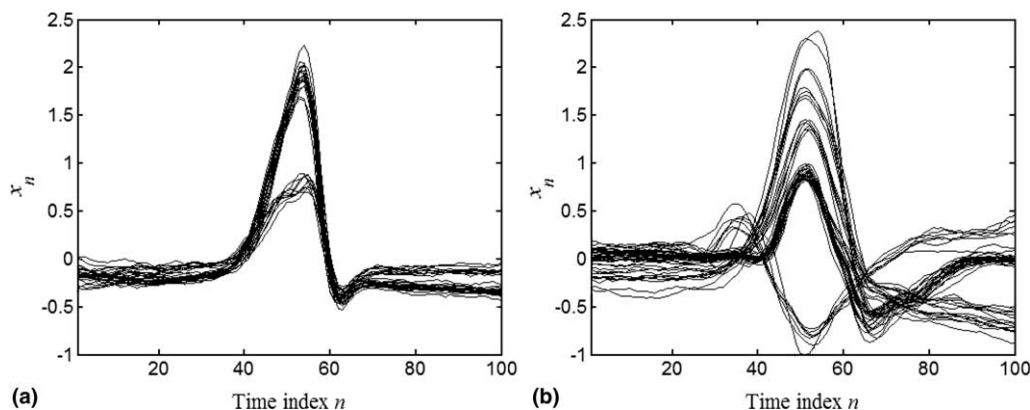


Fig. 1. Mean-centered ECG segments showing the QRS complex; the data has been extracted from MIT-BIH DB record 223: (a) atrial type; (b) ventricular type. The amplitude of the signals was divided by 200.

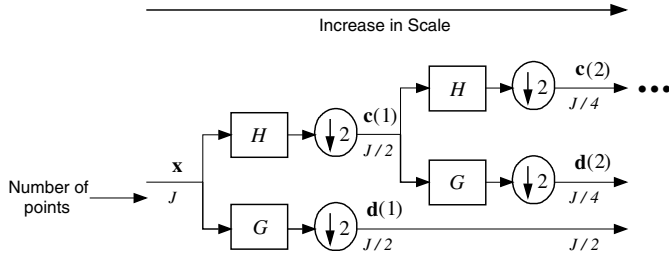


Fig. 2. Two-channel filter bank implementation of the wavelet transform applied to a data vector x . Blocks H and G represent a low-pass and a high-pass filter respectively and $\downarrow 2$ denotes the operation of dyadic downsampling. The decomposition can be carried out in more resolution levels by successively splitting the low-pass channel.

wavelet transforms used in this study have two decomposition levels. Section 3 provides information on the three classifiers used, namely an LDA classifier, and an ANN trained using either back-propagation or weight evolution using a GA. In Section 4, we firstly provide some details on the MIT-BIH arrhythmia database where the ECG signals were taken from. Then we examine the normalised variance of the wavelet transformed coefficients that result from the decomposition of the time-domain signals using db6 wavelets and optimal wavelets generated using the two different optimisation strategies. The optimal number of wavelet coefficients that had to be presented to the input of each classifier is determined and a comparison between the classification accuracy for testing and validation data sets is provided. Our results are discussed taking into account previous classification attempts using different methods. Finally, the main conclusions of this study are summarised in Section 5.

2. Optimisation of the discrete wavelet transform

The time-domain ECG signatures were firstly detrended by subtracting the mean and a transformation was performed using the discrete wavelet transform (Strang and Nguyen, 1996). Each time-domain signature was represented by a data vector x of length J , where the n th element of x , denoted by x_n , represents the measured signal at the n th sampling instant.

As described in the wavelet literature (Vaidyanathan, 1993; Sherlock and Monro, 1998; Tuqun and Vaidyanathan, 2000), the discrete wavelet transform can be calculated in a fast manner by using a finite-impulse-response (FIR) filter bank structure of the form depicted in Fig. 2.

It is worth noting that general multi-channel FIR filter bank decompositions could also be employed in this context (Vetterli and Kovacevic, 1995), but the scope of the present work will be restricted to two-channel filter banks. In this filter bank, the low-pass filtering result undergoes successive filtering iterations with the number of iterations N_{it} chosen by the analyst. The final result of the decomposition of data vector x is a vector resulting from the concatenation of row vectors $c(N_{it})$ (termed approximation coefficient at the largest scale level) and $d(s)$ (termed detail coefficients at the s th scale level, $s = 1, \dots, N_{it}$) in the following manner:

$$t = [c(N_{it}) | d(N_{it}) | d(N_{it} - 1) | \dots | d(1)] \quad (1)$$

with coefficients in larger scales (e.g. $d(N_{it})$, $d(N_{it} - 1)$, $d(N_{it} - 2)$, ...) associated with broad features in the data vector, and coefficients in smaller scales (e.g. $d(1)$, $d(2)$, $d(3)$, ...) associated with narrower features such as sharp peaks.

The filter bank transform can be regarded as a change in variables from \mathfrak{R}^J to \mathfrak{R}^J performed according to the following operation,

$$t_j = \sum_{n=0}^{J-1} x_n v_j(n), \quad j = 0, 1, \dots, J - 1 \quad (2)$$

where t_j is a transformed variable and $v_j(n) \in \mathfrak{R}$ is a transform weight. It proves convenient to write the transform in matrix form as

$$t_{1 \times J} = x_{1 \times J} V_{J \times J} \quad (3)$$

where $x = [x_0 \ x_1 \ \dots \ x_{J-1}]$ is the row vector of original variables, t is the row vector of new (transformed) variables and V is the matrix of weights. Choosing V to be unitary (that is, $V^T V = I$), the transform is said to be orthogonal and it, therefore, consists of a simple rotation in the coordinate axes (with the new axes directions determined by the columns of V).

Let $\{h_0, h_1, \dots, h_{2N-1}\}$ and $\{g_0, g_1, \dots, g_{2N-1}\}$ be the impulse responses of the low-pass and high-pass filters respectively. Assuming that filtering is carried out by circular convolution, the procedure for generating the approximation coefficients from the data vector x is illustrated in Table 1. The convolution consists of flipping the filtering sequence and moving it alongside the data vector. For each position of the filtering sequence with respect to the data vector, the scalar product of the two is calculated (with missing points in the filtering sequence replaced with zeros). For instance, if $N = 2$, the third row in Table 1

Table 1
Convolution procedure for low-pass filtering showing results before and after dyadic down-sampling

x_0	x_1	\dots	x_{2N-1}	x_{2N}	\dots	x_{J-1}	x_0	x_1	\dots	x_{2N-2}	Before	After
h_{2N-1}	h_{2N-2}	\dots	h_0				\dots				c'_0	
	h_{2N-1}	\dots	h_1	h_0				\dots			c'_1	c_0
		\dots		\vdots				\dots			\vdots	\vdots
						h_{2N-1}	h_{2N-2}	h_{2N-3}	\dots		c'_{J-2}	
							h_{2N-1}	h_{2N-2}	\dots	h_0	c'_{J-1}	$c_{J/2-1}$

shows that $c'_1 = x_1 h_3 + x_2 h_2 + x_3 h_1 + x_4 h_0$. Dyadic down-sampling is then performed to c'_{2i+1} to generate coefficients c_i . The detail coefficients d_i are obtained in a similar manner by using the high-pass filtering sequence.

If the approximation c and detail d coefficients are stacked in vector $t = [c|d]$, the wavelet transform can be expressed in the matrix form with the transformation matrix given by

$$V = \begin{bmatrix} 0 & 0 & \cdots & h_{2N-4} & h_{2N-2} & 0 & 0 & \cdots & g_{2N-4} & g_{2N-2} \\ h_{2N-1} & 0 & \cdots & h_{2N-5} & h_{2N-3} & g_{2N-1} & 0 & \cdots & g_{2N-5} & g_{2N-3} \\ h_{2N-2} & 0 & \cdots & h_{2N-6} & h_{2N-4} & g_{2N-2} & 0 & \cdots & g_{2N-6} & g_{2N-4} \\ h_{2N-3} & h_{2N-1} & \cdots & h_{2N-7} & h_{2N-5} & g_{2N-3} & g_{2N-1} & \cdots & g_{2N-7} & g_{2N-5} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_0 & h_2 & \cdots & 0 & 0 & g_0 & g_2 & \cdots & 0 & 0 \\ 0 & h_1 & \cdots & 0 & 0 & 0 & g_1 & \cdots & 0 & 0 \\ 0 & h_0 & \cdots & 0 & 0 & 0 & g_0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & h_{2N-2} & 0 & 0 & 0 & \cdots & g_{2N-2} & 0 \\ 0 & 0 & \cdots & h_{2N-3} & h_{2N-1} & 0 & 0 & \cdots & g_{2N-3} & g_{2N-1} \end{bmatrix} \quad (4)$$

A requirement for the transform to be orthogonal (i.e., $V^T V = I$) is that the sum of the squares of each column must be equal to one and the scalar product of different columns must be equal to zero. Therefore, for a filter bank that utilizes low-pass and high-pass filters, the following conditions ensure orthogonality of the transform so that no information is lost in the decomposition process (Strang and Nguyen, 1996):

$$\sum_{n=0}^{2N-1-2l} h_n h_{n+2l} = \begin{cases} 1, & l = 0 \\ 0, & 0 < l < N \end{cases} \quad (5a)$$

$$g_n = (-1)^{n+1} h_{2N-1-n}, \quad n = 0, 1, \dots, 2N - 1 \quad (5b)$$

Under these conditions, the filter bank is said to enjoy a perfect reconstruction (PR) property, because x can be reconstructed from t which means that there is no loss of information in the decomposition process. Although other non-orthogonal filter bank transforms can also enjoy a PR property, provided that they are associated to a non-singular matrix V , the analysis in the present work is restricted to orthogonal transforms. In fact, the orthogonality of the transform (with the consequent PR property) ensures that no information that may be potentially useful for classification purposes is lost in the decomposition process. Moreover, convenient parameterisation schemes may then be employed to cast the transform filters into forms amenable to optimisation, as discussed below.

One limitation of the procedure described for the transformation of the time-domain ECG signatures in the wavelet domain is that the low-pass and high-pass filters must be chosen a priori and are not adapted to optimally describe the experimental data set. Optimising the transform to maximize its compression ability and therefore its efficiency is normally achieved by optimising the orthonormal filter bank.

Condition (5b) shows that the high-pass filter is entirely defined once the low-pass filter is chosen. Condition Eq. (5a) states that the $2N$ weights $\{h_n\}$ of the low-pass filter are subject to N restrictions. Thus, there are N degrees of freedom that can be used to optimise the filter bank according to some performance criterion. However, since the restrictions are non-linear and may define a non-convex search space, the optimisation task is not trivial. To circumvent this problem, a convenient parameterisation can be employed to describe the orthonormal filter bank by a set of free parameters that can be adjusted to maximize the compression ability of the transform.

The parameterisation of PR FIR filter banks proposed by Vaidyanathan (1993) as adapted by Sherlock and Monro (1998) to parameterise orthonormal wavelets of arbitrary compact support may be used for this purpose. For a filter bank of the form shown in Fig. 2 where the conditions in Eqs. (5a) and (5b) are satisfied, the transfer function of the low-pass filter in the z -domain can be written as

$$H^{(N)}(z) = \sum_{n=0}^{2N-1} h_n^{(N)} z^{-n} = H_0^{(N)}(z^2) + z^{-1} H_1^{(N)}(z^2) \quad (6)$$

where superscript (N) denotes that the filtering sequences have length $2N$. The terms $H_0^{(N)}(z)$ and $H_1^{(N)}(z)$, which denote polyphasic components of $H^{(N)}(z)$, are given by

$$H_0^{(N)}(z) = \sum_{i=0}^{N-1} h_{2i}^{(N)} z^{-i} \quad (7a)$$

$$H_1^{(N)}(z) = \sum_{i=0}^{N-1} h_{2i+1}^{(N)} z^{-i} \quad (7b)$$

Defining the polyphasic components $G_0^{(N)}(z)$ and $G_1^{(N)}(z)$ of the high-pass filter $G^{(N)}(z)$ in a similar manner, a matrix $F^{(N)}(z)$ may be defined:

$$F^{(N)}(z) = \begin{bmatrix} H_0^{(N)}(z) & H_1^{(N)}(z) \\ G_0^{(N)}(z) & G_1^{(N)}(z) \end{bmatrix} \quad (8)$$

It can be shown (Sherlock and Monro, 1998; Tuqun and Vaidyanathan, 2000) that $F^{(N)}(z)$ can be factorised as

$$F^{(N)}(z) = \begin{bmatrix} C_0 & S_0 \\ -S_0 & C_0 \end{bmatrix} \prod_{k=1}^{N-1} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} C_k & S_k \\ -S_k & C_k \end{bmatrix} \quad (9)$$

where each pair of parameters (C_k, S_k) are related to a common angular parameter θ_k as $C_k = \cos(\theta_k)$ and $S_k = \sin(\theta_k)$, $k = 0, 1, \dots, N - 1$. It follows that the filters can be completely parameterised by N angles $\theta_0, \theta_1, \dots, \theta_{N-1}$, which can assume any value in the set of real numbers, as shown in Fig. 3a.

The weights of the low-pass filter can be easily recovered from a set of angles $\{\theta_k\}$ by using the following recursive formula (Vaidyanathan, 1993):

$$F^{(k+1)}(z) = F^{(k)}(z) \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} C_k & S_k \\ -S_k & C_k \end{bmatrix} \quad (10)$$

for $k = 1, 2, \dots, N - 1$ with

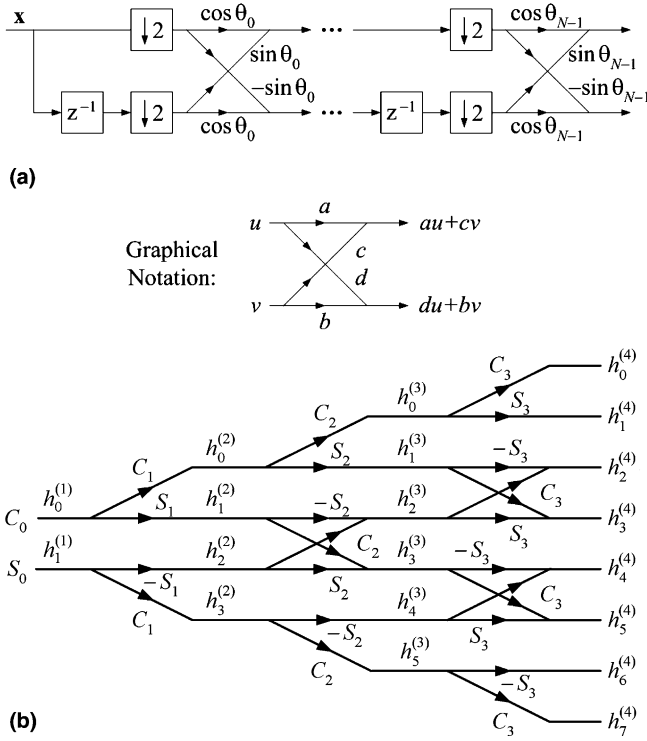


Fig. 3. (a) Procedure for parameterizing wavelet filter banks by N angles. (b) Recursive generation of low-pass filter weights $\{h_n^{(k+1)}\}$ in terms of $\{h_n^{(k)}\}$ by adding one additional angular parameter at a time. S_k and C_k represent the sine and cosine of angular parameter θ_k , respectively. By using this algorithm, any set of N angles $\{\theta_0, \theta_1, \dots, \theta_{N-1}\}$ leads to a sequence of low-pass filter weights that satisfies the orthogonality condition (5a).

$$F^{(1)}(z) = \begin{bmatrix} C_0 & S_0 \\ -S_0 & C_0 \end{bmatrix} \quad (11)$$

Eq. (10) with the initial condition of Eq. (11) provides a way to obtain the weights $\{h_i^{(k+1)}\}$ for a filter of length $2(k+1)$ from the weights $\{h_i^{(k)}\}$ for a filter of length $2k$. To do that, one starts by writing, from Eqs. (8) and (10),

$$H_0^{(k+1)}(z) = H_0^{(k)}(z)C_k - z^{-1}H_1^{(k)}(z)S_k, \quad k = 1, 2, \dots, N-1 \quad (12a)$$

$$H_1^{(k+1)}(z) = H_0^{(k)}(z)S_k + z^{-1}H_1^{(k)}(z)C_k, \quad k = 1, 2, \dots, N-1 \quad (12b)$$

with $H_0^{(1)}(z) = C_0$ and $H_1^{(1)}(z) = S_0$. Then, a recursive formula for the generation of low-pass filter weights with even indexes $\{h_{2i}\}$ can be stated by using the definitions in Eqs. (7a) and (7b) to expand Eq. (12a) as

$$\begin{aligned} \overbrace{\sum_{i=0}^k h_{2i}^{(k+1)} z^{-i}}^{H_0^{(k+1)}(z)} &= \overbrace{\left(\sum_{i=0}^{k-1} h_{2i}^{(k)} z^{-i} \right)}^{H_0^{(k)}(z)} C_k - z^{-1} \overbrace{\left(\sum_{i=0}^{k-1} h_{2i+1}^{(k)} z^{-i} \right)}^{H_1^{(k)}(z)} S_k \\ &\Rightarrow \sum_{i=0}^k h_{2i}^{(k+1)} z^{-i} = C_k h_0^{(k)} \\ &\quad + \sum_{i=1}^{k-1} (C_k h_{2i}^{(k)} - S_k h_{2i-1}^{(k)}) z^{-i} - S_k h_{2k-1}^{(k)} z^{-k} \end{aligned} \quad (13)$$

for $k = 1, 2, \dots, N-1$, with $h_0^{(1)} = C_0$ and $h_1^{(1)} = S_0$. From the identity of terms with the same power of z in the last line of Eq. (13), it follows that:

$$\begin{cases} h_0^{(k+1)} = C_k h_0^{(k)} \\ h_{2i}^{(k+1)} = C_k h_{2i}^{(k)} - S_k h_{2i-1}^{(k)}, \quad i = 1, 2, \dots, k-1 \\ h_{2k}^{(k+1)} = -S_k h_{2k-1}^{(k)} \end{cases} \quad (14a)$$

for $k = 1, 2, \dots, N-1$.

A similar formula can be stated for the low-pass filter weights with odd indexes, by expanding Eq. (12b) as

$$\begin{cases} h_1^{(k+1)} = S_k h_0^{(k)} \\ h_{2i+1}^{(k+1)} = S_k h_{2i}^{(k)} + C_k h_{2i-1}^{(k)}, \quad i = 1, 2, \dots, k-1 \\ h_{2k+1}^{(k+1)} = C_k h_{2k-1}^{(k)} \end{cases} \quad (14b)$$

for $k = 1, 2, \dots, N-1$. After obtaining the low-pass filtering sequence as explained above, the high-pass filtering sequence can be obtained by using Eq. (5b).

The procedure for obtaining filter weights $\{h_i^{(k+1)}\}$ in terms of $\{h_i^{(k)}\}$ by adding one additional angular parameter θ_k is depicted in Fig. 3b (Sherlock and Monroe, 1998). In this graphical notation, each arrow represents the multiplication of the element at the base of the arrow with the constant on top of the arrow. When two arrows arrive at the same point, the results of the multiplications are added together.

By adopting the parameterisation described above, the adjustment of the filter bank to the ensemble of signals under consideration can be formulated as a problem of unconstrained optimisation in R^N . The optimal filtering procedure employed in this work was aimed at maximizing the variance explained by the wavelet coefficients kept in the thresholding process. The optimisation consisted of maximizing an objective function $F(\theta) : R^N \rightarrow R$ defined as

$$F(\theta) = \sum_{j \in I} \sigma^2(j; \theta) \quad (15)$$

where θ is the vector of N angles that parameterise the filter bank as explained above, $\sigma(j; \theta)$ is the standard deviation of the j th wavelet coefficient calculated in the set of training signals, and I is the index set of the coefficients used. Since the overall variance of the data set is preserved by an orthogonal transform, maximizing (15) amounts to maximizing the relative explained variance of the wavelet coefficients employed. It is worth noting that I is defined on the basis of the variance of the wavelet coefficients before the optimisation.

The flexible polyhedron algorithm available in the Matlab Optimisation Toolbox (Nocedal and Wright, 1999) was employed to search for the optimum θ . As one of the purposes of this work was an inter-comparison between the performance gain provided through the optimisation procedure as opposed to using a non-optimised wavelet to decompose the ECG signal, the starting point for optimisation was taken as the set of parameters associated with a conventional wavelet of the Daubechies family (Misiti et al., 1996).

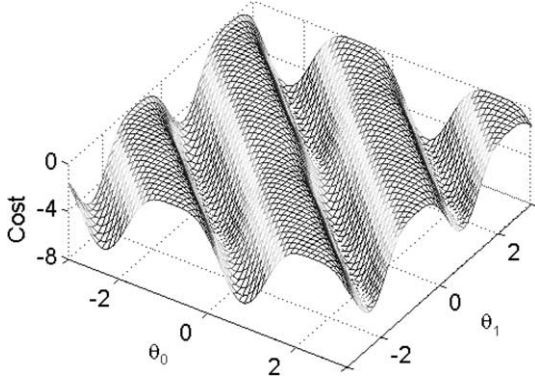


Fig. 4. Graphical representation of the cost function $-F(\theta)$.

For illustration, Fig. 4 depicts a graph of the cost function (objective function with minus sign) obtained by using a wavelet transform with two degrees of freedom as would be the case if a db2 wavelet was used. It is worth noting that the graph is repeated at modulo 2π , because of the periodicity of the sine and cosine functions involved in the parameterisation. As shown in Fig. 4, it is possible that the objective function (15) may have local maxima different from the global maximum. In that case, the flexible polyhedron algorithm will tend to converge to the closest local maximum. However, even if the global maximum is not attained, an improvement over the original wavelet transform may still be obtained.

An approach to circumvent the local maxima problem of the formulation described above consists of exploiting the product filter parameterisation proposed by Moulin et al. (1997). In order to describe the optimisation process, we use again a circular convolution process for the data vectors under the downsampling operation $\mathbf{c}' = \mathbf{h}\check{\mathbf{X}}$ and $\mathbf{d}' = \mathbf{g}\check{\mathbf{X}}$ where $\mathbf{h} = [h_{2N-1}, h_{2N-2}, \dots, h_0]$ and $\mathbf{g} = [g_{2N-1}, g_{2N-2}, \dots, g_0]$ are the corresponding impulse response sequences of the low- and high-pass filters respectively and $\check{\mathbf{X}}$ is a circulant matrix formed from the data vector \mathbf{x} as

$$\check{\mathbf{X}} = \begin{bmatrix} x_0 & x_1 & \cdots & x_{J-2} & x_{J-1} \\ x_1 & x_2 & \cdots & x_{J-1} & x_0 \\ x_2 & x_3 & \cdots & x_0 & x_1 \\ \vdots & \vdots & & \vdots & \vdots \\ x_{2N-1} & x_{2N} & \cdots & x_{2N-3} & x_{2N-2} \end{bmatrix}_{2N \times J} \quad (16)$$

and we consider the power (energy divided by the number of coefficients) of the low-pass and high-pass filter outputs. Since the number of coefficients is $J/2$, due to the downsampling operation, the power values of the approximation (\mathbf{c}^m) and of the detail (\mathbf{d}^m) coefficients for the m th training signal are given by

$$P_c^m = \frac{\mathbf{c}^m \mathbf{c}^{mT}}{J/2} \quad (17a)$$

$$P_d^m = \frac{\mathbf{d}^m \mathbf{d}^{mT}}{J/2} \quad (17b)$$

If M training signals are employed, the overall power of the approximations and details are

$$P_c = \sum_{i=1}^M P_c^m \quad (18a)$$

$$P_d = \sum_{i=1}^M P_d^m \quad (18b)$$

The relation between P_c and P_d can be expressed in an objective function $F: \mathfrak{R}^{2N} \times \mathfrak{R}^{2N} \rightarrow \mathfrak{R}$ given by

$$F(\mathbf{h}, \mathbf{g}) = \frac{0.5(P_c + P_d)}{\sqrt{P_c P_d}} \quad (19)$$

which is similar to the coding gain used to assess the compression performance of two-channel filter bank structures (Tuqun and Vaidyanathan, 2000).

If the conditions in Eq. (5) are satisfied, the WT is orthogonal, and thus it preserves power (Strang and Nguyen, 1996). As a result, the power of the input signal \mathbf{x}^m equals the sum of the power in both output channels, $P_c^m + P_d^m$. Hence, for a given training set, the sum $A = P_c + P_d$ is constant and thus maximizing F is equivalent to maximizing P_c . Writing:

$$F^2 = \frac{0.25A^2}{AP_c - P_c^2} \quad (20)$$

it follows that F reaches a minimum of 1 when $P_c = A/2$, that is, when the power is equally divided between the approximation and detail coefficients. As P_c increases from $A/2$ to A , F increases and tends to $+\infty$ when P_c tends to A , i.e., when all the power is contained in the approximation coefficients. The advantage of aiming at maximizing P_c lies in the fact that the signal-to-noise ratio is usually larger in the low-pass filter output. Thus, maximizing P_c further improves the filtering performance. It is worth noting that, if the data set is mean-centered (variable-wise) prior to the wavelet decomposition, the power is equal to the variance. In this manner, the optimisation amounts to maximizing the variance explained by the approximation coefficients.

Since P_c only depends on the low-pass filter weights \mathbf{h} , the problem can be restated as the maximization of an objective function $\varepsilon: \mathfrak{R}^{2N} \rightarrow \mathfrak{R}$ given by

$$\varepsilon(\mathbf{h}) = P_c = \frac{1}{J/2} \sum_{m=1}^M \mathbf{c}^m \mathbf{c}^{mT} \cong \frac{1}{J} \sum_{i=1}^M \mathbf{c}^m \mathbf{c}^{mT} \quad (21)$$

where $\mathbf{c}^m = \mathbf{h}\check{\mathbf{X}}^m$ is the vector of detail coefficients for the m th training signal, before the downsampling operation as shown in Table 1 and $\check{\mathbf{X}}^m$ is the circulant matrix formed from \mathbf{x}^m . This holds because we may assume that the power of the detail coefficients before and after downsampling is approximately the same. Using (21) one may write:

$$\varepsilon(\mathbf{h}) = \frac{1}{J} \sum_{m=1}^M \mathbf{h} \check{\mathbf{X}}^m \check{\mathbf{X}}^{mT} \mathbf{h}^T = \mathbf{h} \underbrace{\sum_{m=1}^M \left(\frac{1}{J} \check{\mathbf{X}}^m \check{\mathbf{X}}^{mT} \right)}_{\mathbf{R}} \mathbf{h}^T \quad (22)$$

Since $\check{\mathbf{X}}^m \check{\mathbf{X}}^{m^T}$ is Toeplitz for any \mathbf{x}^m , then $\mathbf{R}_{2N \times 2N}$ is also a Toeplitz matrix. Thus, the constraints in Eq. (5) allow the objective function to be rewritten, with a slight abuse of notation, in the following linear form (Moulin et al., 1997)

$$\varepsilon(\mathbf{a}) = \frac{r_0}{2} + \sum_{n=0}^{N-1} a_n r_{2n+1} \quad (23)$$

where $\{r_0, r_1, \dots, r_{2N-1}\}$ are the elements of the first row of \mathbf{R} and vector $\mathbf{a} = [a_0 \ a_1 \ \dots \ a_{N-1}]$ contains the coefficients of the product filter $P(z)$ defined as

$$P(z) = H(z)H(z^{-1}) = 1 + \sum_{n=0}^{N-1} a_n (z^{-2n-1} + z^{2n+1}) \quad (24)$$

Given \mathbf{a} , the transfer function $H(z)$ of the desired filter can be recovered from $P(z)$ by a spectral factorisation procedure (Strang and Nguyen, 1996). This factorisation is possible provided that the frequency response of the product filter given by $Q(f) = P(e^{j2\pi f})$ (where j is the imaginary unity), is non-negative at all frequencies f , that is, $Q(f) \geq 0$, $\forall f \in \mathfrak{R}$. It follows that the following restriction must be enforced:

$$Q(f) = 1 + 2 \sum_{n=0}^{N-1} a_n \cos[2\pi f(2n+1)] \geq 0 \quad (25)$$

Since $Q(f)$ is periodic with period 1 and $Q(f) + Q(f+0.5) = 2$, $\forall f \in \mathfrak{R}$, it is sufficient to consider the restriction $Q(f) \geq 0$ in the interval $0 \leq f \leq 0.5$, that is

$$1 + 2 \sum_{n=0}^{N-1} a_n \cos[2\pi f(2n+1)] \geq 0, \quad 0 \leq f \leq 0.5 \quad (26)$$

Maximizing $\varepsilon(\mathbf{a})$ defined in Eq. (23) with respect to \mathbf{a} subject to the inequality restrictions in Eq. (26) is a linear semi-infinite programming (LSIP) problem (Hettich and Kortanek, 1993), because there is a finite number of variables (a_0, a_1, \dots, a_{N-1}) and infinitely many restrictions. This problem can be solved by discretising the frequency interval $[0, 0.5]$ to generate a finite number of restrictions, and then applying standard linear programming techniques (Chvatal, 1983). The solution $\tilde{\mathbf{a}}$ to this approximated problem can then be used to generate a feasible solution \mathbf{a}_f to the original problem as discussed by Moulin et al. (1997):

$$\mathbf{a}_f = \frac{\tilde{\mathbf{a}}}{1 - \delta} \quad (27)$$

where $\delta \leq 0$ is the minimum of $Q(f)$ in the interval $0 \leq f \leq 0.5$ when $\tilde{\mathbf{a}}$ is used instead of \mathbf{a} in Eq. (25).

From the above description, it may be concluded that by adopting the coding gain as a measure of the compression performance of the low-pass/high-pass filter pair (Unser, 1993), the optimisation of coefficients $\{a_0, a_1, \dots, a_{N-1}\}$ can be cast into a LSIP problem. By using a convenient discretisation procedure (Moulin et al., 1997) such a problem can then be converted into a linear programming one, for which efficient solution algorithms exist (Chvatal, 1983).

In the present work, the filter bank was optimised by applying the above-mentioned approach to each low-pass/high-pass filter pair separately. For this purpose, the first-level pair was initially optimised. The low-pass filter output was then employed as the input signal for the second-level optimisation. Moreover, an overall coding gain was employed to measure the compression ability of the filter bank for the ensemble of signals in the training data set. In contrast to Sherlock and Monro's (1998) optimisation, in this case the resulting optimisation problem is convex and the solution found is guaranteed to be the global maximum of the objective function. However, it is worth noting that the objective function cannot target a specific set of wavelet coefficients, unlike the previous formulation. Instead, the optimisation is focused on maximizing the variance explained by the approximation coefficients. That may be a shortcoming if the model is to include detail coefficients in addition to the approximation ones.

In order to illustrate the effect of using different objective functions for the optimisation of the wavelet decomposition algorithms, in Fig. 5 we depict a contour plot of the cost surface shown in Fig. 4. The circle, the cross and the triangle indicate the points associated with three different wavelets (db2, Sherlock and Monro's and Moulin's), when the optimisations were performed using only two angles. A detail of the region around these three points is presented as an inset graph to Fig. 5 (note the different scaling used).

It can be seen that the db2 locus lies on a flat surface between two local minima. In this case, the flexible polyhedron algorithm employed in Sherlock and Monro's optimisation was not able to move away from that plateau towards neighbouring minima. The inset graph shows that Moulin's solution lies in approximately the same cost level

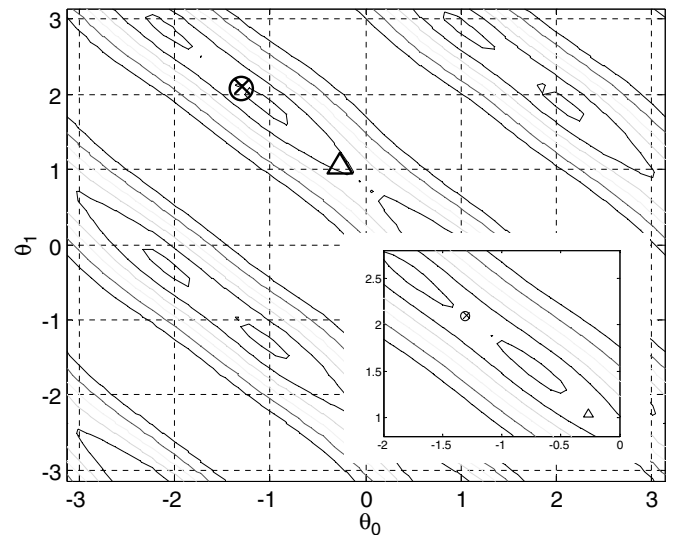


Fig. 5. Contour plot depicting the locus of the points associated with the db2 (circle), Sherlock and Monro's optimisation (cross) and Moulin's optimisation (triangle) in relation to the cost function landscape $-F(\theta)$. The inset shows further detail.

as Sherlock and Monro's, on the other side of a local minimum basin.

As described earlier, in Moulin's approach we optimised both decomposition levels (the first level solution is shown in Fig. 5). However, the optimisation is performed in a sequential manner rather than a batch manner as is in the Sherlock and Monro's case (we first optimise the first level and then we optimise the second level). Moreover, it is worth noting that, although both techniques are designed to optimise the compression performance of the filter bank, each technique employs a different criterion (either (15) or (19)) to evaluate the compression performance. In Fig. 5, the results are compared with regard to Sherlock and Monro's criterion (Eq. (15)). Furthermore, it is worth noting that neither of the techniques is designed to optimise the classification performance in a direct manner. However, we anticipated that an improvement in compression ability should be reflected in an improvement in the classification performance.

In contrast to the simplified example presented above, for the ECG classification study we used the db6 wavelet where six angles would have to be used to fully describe the low-pass filter. The calculations performed for Sherlock and Monro's optimisation as well as for Moulin's optimisation were also carried out by using six angles. The illustration in Fig. 5 using two angles avoids the problem of depicting possible differences occurring in a six-dimensional space. A comparison between the filters resulting from the two optimisation procedures described and the db6 filters is presented in Fig. 6.

Using Moulin's approach, it can be seen, that even though the filter configurations have been optimised by LSIP so that each decomposition level can have its own filter coefficients, the resulting filters are still relatively similar to each other. Such similarity between the filter coefficients of the first and second decomposition level has also been observed elsewhere (Coelho et al., 2003), where the LSIP strategy was also used to optimise the wavelet transform. For other approaches to the optimisation of the wavelet transform the reader is referred to other relevant literature (Vetterli and Kovacevic, 1995; Ramchandran et al., 1996).

3. ECG classifiers

3.1. Linear discriminant analysis

The most widely used discriminant analysis method is the one developed by Fisher in 1936, which attempts to maximize the ratio of between-groups to within-groups variance (Lachenbruch, 1975). For the classification of the ECG signals, the row vectors \mathbf{t} of the transformed variables according to Eq. (2) were used. Using vector-matrix notation, the following discriminant function can be derived for the case of binary classification in the wavelet domain:

$$Z(\mathbf{t}) = (\mu_1 - \mu_2)\Sigma^{-1}\mathbf{t} \quad (28)$$

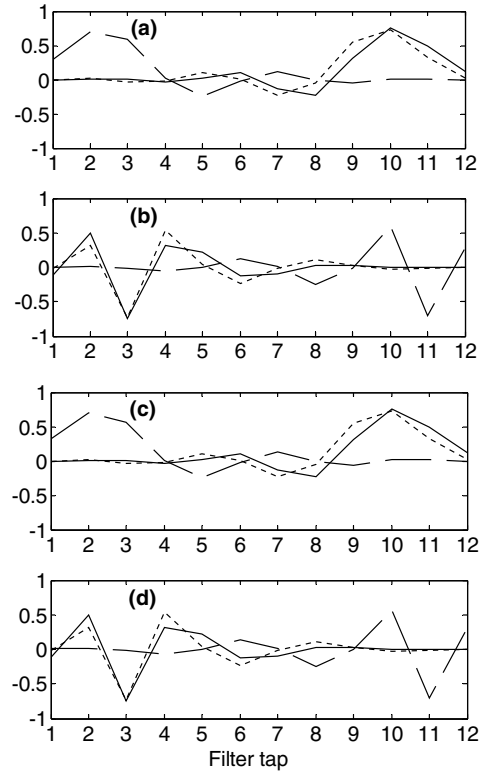


Fig. 6. (a) Low-pass and (b) high-pass filters: db6 (solid line), Moulin's optimisation (dashed line), Sherlock and Monro's optimisation (dotted line) at the first decomposition level. (c) and (d) depict the same filters at the second decomposition level.

where $Z: R^J \rightarrow R$ is a discriminant function, $\mathbf{t} = [t_0 \ t_1 \ \dots \ t_{J-1}]^T$ is a vector of J classification variables, $\mu_1 \in R^J$ and $\mu_2 \in R^J$ are the sample mean vectors of each group, and Σ is the common sample covariance matrix with dimensions $J \times J$. Eq. (28) is commonly written in expanded form as: $Z = w_0 t_0 + w_1 t_1 + \dots + w_{J-1} t_{J-1} = \mathbf{w}^T \mathbf{t}$, where $\mathbf{w} = [w_0 \ w_1 \ \dots \ w_{J-1}]^T$ is a vector of coefficients: $\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2)$. The cut-off value for classification is calculated as the midpoint of the mean scores of the two samples given by $z_c = 0.5(\mu_1 - \mu_2)\Sigma^{-1}(\mu_1 + \mu_2)$. A given vector \mathbf{t} is assigned to class 1 if $Z(\mathbf{t}) > Z_c$, and to class 2 otherwise.

3.2. The artificial neural network

The artificial neural network type used for this research was the standard feed-forward multi-layer perceptron. As a starting point, we define the net input to a processing element q for the m th ECG input pattern as

$$v_q(m) = \sum_{j=0}^{J-1} w_{qj} t_j(m) + b_q \quad (29)$$

where $t_j(m)$, $j = 0, 1, \dots, J-1$ are the values of the wavelet coefficients for the m th input pattern, w_{qj} is the neural network weight for the connection between coefficient j and processing element q and b_q is an ANN bias element. The output of the processing element is the result of passing the scalar value $v_q(m)$ through its activation function

$\varphi_q(\cdot) : y_q(m) = \varphi_q(v_q(m))$. In this work we have used the logistic sigmoid function, which has a range of values between zero and one:

$$\varphi_q(v_q(m)) = \frac{1}{1 + e^{-v_q(m)}} \quad (30)$$

The output of the feed-forward MLP is of the form $y = \varphi_2(\mathbf{W}_2\varphi_1(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1) + b_2)$, where φ_1 is the activation function of the hidden layer, which is assumed to apply element-wise to its vector argument, b_2 is the scalar bias in the output layer, \mathbf{W}_1 and \mathbf{W}_2 are weight matrices and \mathbf{b}_1 is a bias vector. We have used a back-propagation training algorithm that employs a gradient search to minimise the mean-square-error (MSE) cost function:

$$\text{MSE} = \frac{1}{M} \sum_{m=1}^M [y(m) - d(m)]^2 \quad (31)$$

where $d(m)$ is the desired output for the m th training pattern, $y(m)$ is the actual MLP output and M is the total number of training samples. The MLP weights were adjusted through a learning process with each training pattern being presented to it in a sequential manner. An ANN topology consisting of 2 hidden layers with 10 nodes each, was found sufficient to learn the training data. The output layer consisted of just 1 node and the number of nodes in the input layer was set equal to the number of wavelet coefficients used as inputs. All nodes except for the ones contained in the input layer had a bias associated with them.

Very often, the ANN architectures used are based on the principle that the neural network should be larger than required rather than not sufficiently complex. As a consequence of this design choice, there existed the possibility of data over-fitting, which is undesirable since such practice can potentially compromise the generalisation ability of the network. To minimise the risk of data over-fitting a cross-validation method (Stone, 1978) can be used, where the available data is divided into three distinct data sets: the training set, the validation set and the testing set. The ANN was then trained as usual with the training data set but every so often its generalisation ability was cross-validated with the validation set. As soon as the classification accuracy of the validation set deteriorated, the training stopped. This is commonly referred to in the NN literature as the “early stopping method of training” (Haykin, 1999).

Since the data set used in this work was limited to make the classification task more challenging, it was not feasible to divide the data into three different but equally representative sets. It was decided that the training of the neural network would be continued until the training set was fully recognised (accuracy of 100%) by the ANN within a maximum of 1000 epochs or 1500 generations depending on the approach used. The neural network performance was then assessed using the testing data set. This approach has the advantage that no independent cross-validation test is required and that the data can be used for determining the ideal number of wavelet coefficients instead. A draw-

back of the method is that it assumes that the data was 100% correctly classified by the experts. It was hoped that this measure would preserve the generalisation ability of the ANN by preventing over-fitting. The results presented later in this paper support this view.

3.3. The genetic algorithm

The third classifier was a feed-forward MLP where a steady-state genetic algorithm was used during the training process. Half of the fittest population was set to survive to the next generation. This provided an appropriate trade-off between speeding up the evolution by producing more solutions (offsprings) at every generation and preventing premature convergence by having a larger parental variability. The genetic representation of a neural network was implemented as a fixed two-dimensional array of floats. Each floating-point number directly corresponds to one weight of the ANN, which allows the crossover and mutation operators to have a one to one mapping. This is also known in the GA literature as direct encoding (Fogel, 1995). It is believed that this approach is superior to randomly flipping bits of a binary representation since in that case, an additional mapping function would also be required (Yao, 1999). The crossover operator used maintained the gene pool sufficiently diverse. Since a recent study found no significant difference between the effects of several popular crossover operators when used to evolve ANN's (Pendharkar and Rodger, 2004), it was decided that a standard one-point crossover was sufficient for the purpose of this study. In addition, the representational redundancy usually associated with direct encoding schemes gives the GA more possibilities of hitting on fit solutions (Hancock, 1992). Improved genotypes were fine-tuned by mutation. Whenever an individual was mutated, a ‘coin flip’ biased by the mutation rate was executed for each connection weight. For every successful mutation, a random number picked from a standard Gaussian distribution was then added to that weight.

The selection operator that was used is known as the roulette wheel selector, which picks an individual based on the magnitude of its fitness score compared to the rest of the population. The advantage of this selection operator is that it does not always guarantee that the best solution is passed on. This also helped reduce the chance of premature convergence. The GA terminated when either the training data was fully recognised or the given number of generations had run out.

The simplest implementation of a fitness or objective function for the classification task was to reward individuals proportionally to the percentage accuracy achieved on the entire training data set. A record was considered correctly classified when its absolute error was below a specified threshold, where the absolute error refers to the absolute difference between the target (binary: 0 or 1) and actual output (continuous: 0 to 1). The threshold was first set to 0.5, however, this turned out to be too high

because such evolved ANN's often had poor discriminatory abilities. Further testing with the threshold set to 0.2, improved the classification accuracy. This lower threshold provided a better trade-off between the ability to generalise and discriminate in the evolved ANNs.

A further concern of using the above-mentioned objective function was that thresholding may lead to valuable fitness information being lost. If fitness is only based on percentage accuracy, then the search landscape is not smooth but has many flat planes at different fitness heights (one plane for each possible number of correctly classified patterns). For example, if a neural network has a classification accuracy of 90% then it does not matter whether it achieved this score with a large or small absolute error sum even though a smaller error would indicate that it is better at capturing the regularities of the training data. If there is no fitness gradient information for the selection operator to work on, in that situation the GA would be reduced to random search (Langdon and Poli, 2002). By providing the fitness function with a method of evaluating the difference between the various classification percentages, the selection operator can work a lot more efficiently. This was accomplished by implementing an objective function that uses the inverse of the absolute error summed over all training patterns. In addition to rewarding the correct classification of patterns, this objective function also provides gradient information between the percentage accuracy fitness planes.

A concern in our work was that the GA would often start reducing the classification error of one class at the expense of the other class. The GA would then continue until the ANN population was sufficiently specialized in classifying one particular class, trapping itself in a local optimum. This could lead to a poor generalisation ability of the evolved solutions since it is easier to evolve the ANNs to fine-tune their existing classification strategy than finding new means of generalisation. An empirical solution to the problem is to further adjust the calculation of fitness scores. In order to balance the evolution of classification accuracy across both classes, the class with the least accuracy was assigned an error bias inversely proportional to its achieved percentage accuracy. Thus, the classification improvement of the least accurately classified class outweighed the fitness benefit of fine-tuning the other class. Therefore, if the GA started favouring a reduction in the error associated with one class, this soon become less attractive than improving the accuracy of the other class. This method of evaluating fitness resulted in an improved generalisation ability of the evolved networks and was used for generating the results presented in this paper.

4. Results and discussion

4.1. Database considerations

Data was supplied from the MIT–BIH Arrhythmia Database (MIT–BIH DB) (Moody and Mark, 2001). The data-

base contains 30 min long ECG records that have been digitized at 360 Hz. Each of those records has an “annotation file” associated with it containing the isolated QRS instants as well as the classification previously made by human experts. As required in all supervised learning methods, that classification was used as our target output. Following the recommendation of the AAMI (Association for the Advancement of Medical Instrumentation) the first 5 min of each record have been removed. Each sample consists of 100 time points and is taken around the central point of the QRS complex and detrended to have zero mean. The amplitude of the signals was divided by 200.

In the ECG data used in this work, there were a total of 148 records of type A (atrial) and 1164 records of type V (ventricular). The breakdown of the different records can be seen in Table 2. The patterns of type A also include aberrated atrial beats. Even though they are quite rare in the MIT–BIH records they were nevertheless included, in order to increase the difficulty of the classification problem. Out of the data available MIT–BIH record #213 was selected as the validation set because it included ECG data that displayed segmentation as well as baseline drift problems. The validation set was employed to ascertain the best number of wavelet coefficients to be included in the classification models, whereas the test set (records #200, #202, #210) was only used in the final assessment of classification performance. It should also be noted that records #201 and #202 are different records from the same patient. The reason why the classification task does not include any normal heartbeats is that it is assumed that an earlier processing stage has already detected the QRS complexes (the electrical signatures of ventricular contraction) and isolated the premature beats. This kind of detection can be performed by many well-known techniques (Okada, 1979; Kadambe and Boudreaux-Bartels, 1999; Li et al., 1995; Senhadji et al., 1995).

4.2. Determination of the number of wavelet coefficients presented to the classifiers

When using the DWT to obtain a compact representation of the signal, a choice needs to be made on how many

Table 2
Atrial (A) and ventricular (V) ECG data sets used from the MIT–BIH arrhythmia database

	MIT–BIH record	Beat type	
		A	V
Training	201	30	45
	223	30	45
	Total	60	90
Validation	213	24	195
Test	200	28	700
	202	36	15
	210	–	164
	Total	64	879

wavelet coefficients to retain. One way to determine the significance of a particular coefficient is to look at its variance. When ordering the coefficients from highest to lowest variance there comes a point when adding components of lower variance does not make a difference to the information content of the wavelet representation anymore. The relationship between variance and number of wavelet coefficients for the three wavelet networks is depicted in Fig. 7. It can be seen from the graph that after the 25th coefficient, the variance becomes negligible. Therefore, a preliminary reduction was performed by only preserving the first 25 wavelet coefficients. It is worth noting that these coefficients were all of the approximation type (output of the low-pass filter at the second resolution level). Such finding is in line with the assumption of Moulin's optimisation strategy, as mentioned above.

After this preliminary reduction there was still the possibility that even fewer wavelet coefficients were sufficient for the classification task. It was suspected that the information required to distinguish between the atrial and ventricular classes might be contained within a handful of coefficients.

4.2.1. Number of wavelet coefficients presented to the LDA classifier

The LDA approach does not have a problem of data overfitting so severe as the ANN does and, therefore, serves as a benchmark to the performance achieved using ANN classifiers. The reason why it is desirable to reduce the number of coefficients used for LDA is that the more inputs are used the more uncertain become the estimates of the covariance matrices. The number of training points increases linearly with J , the number of inputs, but the number of unknowns to be estimated increases with the square of J . This is because by presenting J inputs to the classifier, the covariance matrix has $[J(J+1)]/2$ different entries that need to be estimated. The best number of wavelet coefficients presented to the linear discriminant model was determined on the basis of the resulting error rate

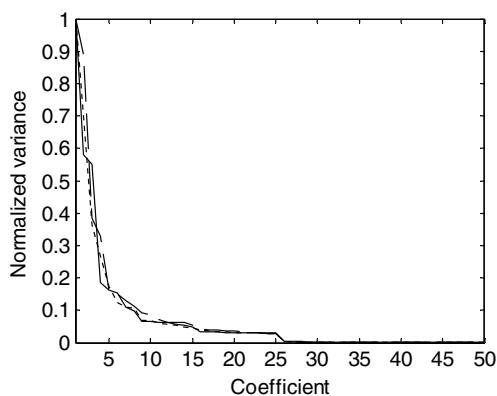


Fig. 7. Normalised variance of the first 50 wavelet coefficients (two-level decomposition): db6 (solid line); Moulin's optimisation (dashed line); Sherlock and Monro's optimisation (dotted line).

(average of the two-atrial and ventricular- classes of data) on the validation data set. The average was taken so as not to bias the results in favour of the class with more records. The coefficients were added in decreasing order of variance. The percentage error to the classification task for each of the possible number of wavelet coefficients ranging from 1 to 25 was then calculated.

As shown in Fig. 8, the smallest validation error in LDA was achieved using three wavelet coefficients irrespective of the method used to generate the wavelet coefficients.

4.2.2. Number of wavelet coefficients presented to the ANN trained by back-propagation

The best number of wavelet coefficients to be used as inputs for training the neural network with back-propagation was determined in an empirical manner. The learning rate was set to 0.4 throughout the experiment, which is close enough to the 0.35 suggested by Rich and Knight (1991) but should allow the weight configuration to converge slightly faster. Each training session was scheduled to run over a maximum of 1000 epochs unless the training data set was 100% correctly classified.

The ANN started off with 25 wavelet coefficient inputs and after five training sessions with random initial weights the coefficient with the lowest variance was removed and the procedure was repeated. After each training session the classification performance was tested on the validation data set and the resulting accuracy was recorded. From the five training sessions the best result was selected to represent the corresponding number of coefficients. The results for all three data sets were quite similar. For the db6 data, 11 wavelet coefficients were chosen as inputs because that configuration achieved a 97.6% classification accuracy on the validation data set. For the Moulin optimised data the best number of coefficients was 12 and a 97.4% accuracy was achieved. Finally, a classification accuracy of 97.6% was achieved when the network was presented with 11 wavelet coefficients generated using Sherlock and Monro's wavelet optimisation strategy. The percentage accu-

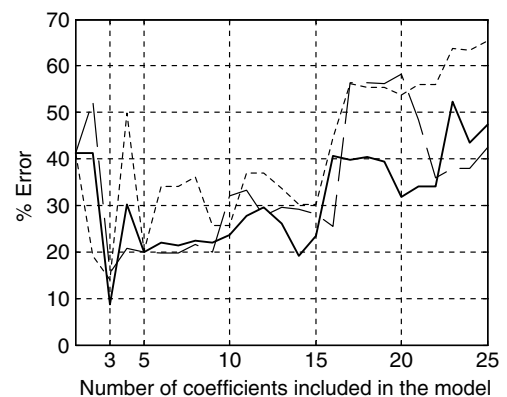


Fig. 8. Average validation error in LDA as a function of the number of wavelet coefficients included in the model (two-level transform): db6 (solid line), Moulin's optimisation (dashed line), Sherlock and Monro's optimisation (dotted line).

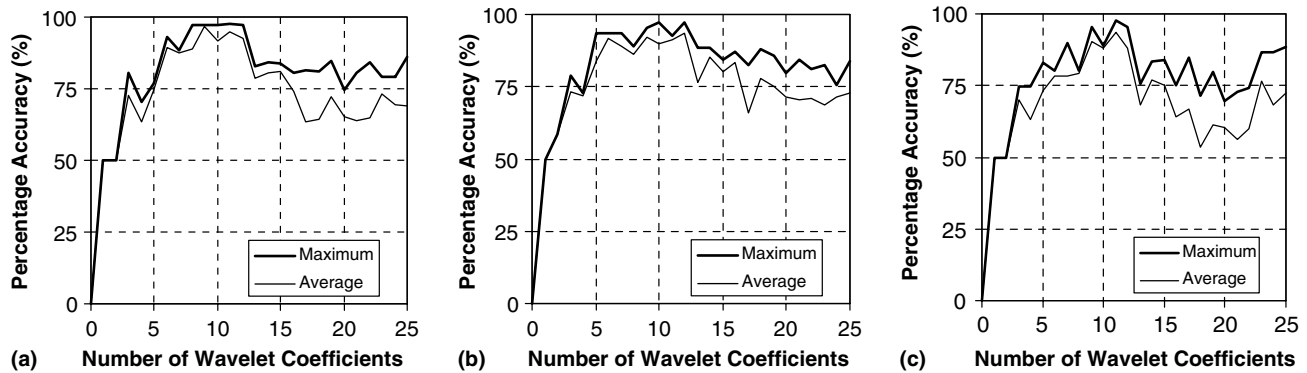


Fig. 9. Validation back-propagation classification results on (a) db6 two-level transform input data and optimised data; (b) using Moulin's optimisation; and (c) using Sherlock and Monro's optimisation method.

racy in the classification of the validation data set for the standard db6 and optimised wavelets are shown in Fig. 9.

It can be observed that further addition of wavelet coefficients result in a steep increase in the classification accuracy. Performing, however, a classification by presenting to the ANN more than 13 coefficients decreased the achieved accuracy since very little additional information was encoded in subsequent coefficients. We concluded that within the first 13 coefficients, the neural network had all the information and complexity needed to almost classify 100% of the validation data correctly. Any further addition of dimension in the ANN feature space would not provide any useful information that could be used to improve the classification accuracy but would only make it harder for the ANN to converge to the best solution. It is worth noting that although the variance of each additional coefficient gets smaller the neural network treats each of its inputs with the same importance.

4.2.3. Number of wavelet coefficients presented to the ANN trained by a GA

In order to evolve the connection weights the following procedure was adopted. Firstly, a steady-state GA was used where 50% of the population was replaced after every generation. Through a trial and error process, it was found that the evolved solutions were not very sensitive to the parameter variations and the achieved classification accu-

racies were difficult to improve upon by further changing the ANN. The population was kept to a constant of 100 individuals. Each new offspring had a 5% chance of being selected for genetic crossover. The mutation rate was set to 5% as well. Therefore, at least one in every twenty network weights of the offspring would be mutated by standard Gaussian mutation. The maximum number of generations was set to 1500. The GA was also terminated if an individual achieved 100% classification accuracy on the training data set.

Whenever an evolutionary run was finished, the best individual of the population was taken and its performance was tested on the validation data set. The rest of the procedure was the same as for the back-propagation approach. Five evolutionary runs were performed after presenting the same number of wavelet coefficients to the input of the network and the best classification result among the five runs was recorded as the 'score' for that amount of coefficients. The process was then repeated after increasing the number of coefficients presented to the input of the network. The percentage accuracy in the classification of the validation data set for the standard db6 and optimised wavelets are shown in Fig. 10.

One of the ANNs that was evolved with the db6 training data was actually able to classify the validation data 100% correctly with only 8 wavelet coefficients as inputs. The ANN architecture with 9 inputs also did very well with

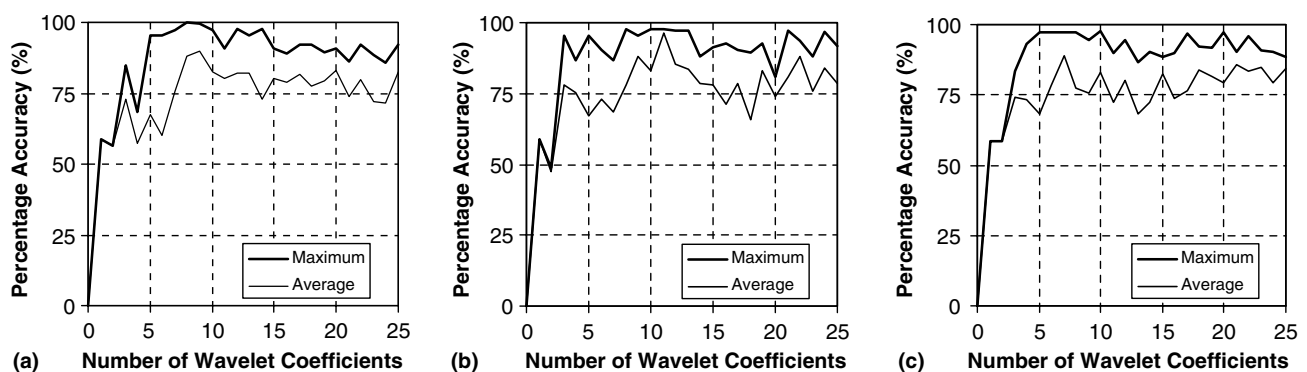


Fig. 10. Validation GA classification results on (a) db6 two-level transform input data and optimised data (b); using Moulin's optimisation; and (c) using Sherlock and Monro's optimisation method.

99.5% classification accuracy. The choice for the number of coefficients to be used with the optimally compressed data generated using Moulin's algorithm was slightly more difficult because there were quite a few configurations achieving around 97% classification accuracy on the validation data set. Included in this selection was the configuration with 12 inputs and since it had already proven to be the best choice for the back-propagation method it was chosen for the evolutionary weight approach as well. This ANN configuration achieved a 99.6% classification accuracy. Finally, a classification accuracy of 97.6% was achieved when the network was presented with 10 wavelet coefficients generated using Sherlock and Monro's wavelet optimisation strategy.

4.3. Comparison of classification accuracy for testing data sets

The classification results for the test set are summarised in Table 3 below. A metric based on the average classification accuracy makes sure that equal weighting is given to both atrial and ventricular data sets. All methods performed reasonably well. Overall the neural networks were slightly more successful than LDA at correctly classifying the ECG signals. In addition, the wavelet optimisation processes facilitated the classification task. The best classifier resulted from a combination of a neural network architecture with 12 inputs, training through weight evolution and the use of the optimised wavelet coefficients using Moulin's approach. It is worth noting that the wavelet transform reduced the dimensionality of the problem by a factor of about 10.

It was to be expected that the neural network classifiers would generally outperform an LDA classifier because the latter is best suited for linearly separable problems. The reason why the GA produced better weight configurations than the standard back-propagation algorithm can be attributed to the fact that the GA is better at avoiding local optima. It is worth noting that the neural network trained by GA also provided the best classification results for ventricular beats, which is an important aspect from a medical point of view, in light of the potential hazards associated to

such type of extrasystole. The best classification accuracy on both the testing and validation data sets was achieved when the ECG signals were transformed in the wavelet domain using a db6 decomposition as shown in Fig. 11a.

By choosing a difficult data set that included a lot of abnormal records the neural networks were forced to be very good at generalisation in order to succeed in their classification task. This generalisation ability then translated to good classification results on the relatively easier testing data set. Fig. 11a–c support this view.

It is worth noting that one of the usual requirements for standard wavelets is that they have zero mean to assure the invertibility of the wavelet transform. For such tasks, the optimisation procedures described is well suited to tackle the problem. If, however, the goal of the transformation is not to reconstruct the original signal—as is the case in our current application— but to reduce the number of inputs to the classifier so as to concentrate the discriminatory information as much as possible, a biased wavelet approach may also be suitable. The advantages using such approach are that (a) usually a large number of wavelets is required to make up for any zero-mean characteristics of a signal, (b) the non-biased wavelets lose time resolution when they are analyzing low-frequency features and (c) whereas conventional wavelets act as moving difference filters, bias allows the wavelet transform to calculate either differences or averages depending on which is more suited to extract a particular feature. Results presented elsewhere (Galvão and Yoneyama, 1999), which adopted such a biased wavelet approach, indicate that the introduction of a bias improves the discriminatory capabilities of the neural network classifier considerably. In that work, the ECG data to be classified into either 0 (atrial) or 1 (ventricular) came from the same MIT database as the data for this work. An important difference between the two studies is that in the previous work, the output included a range (0.25–0.75), where the classification was deferred to an expert cardiologist in order to reduce the risk of misdiagnosis. This contrasts with our GA approach where any prediction with an absolute error larger than 0.2 was already rejected. It was concluded that the biased wavelet neural network classified 89% of the records correctly while the

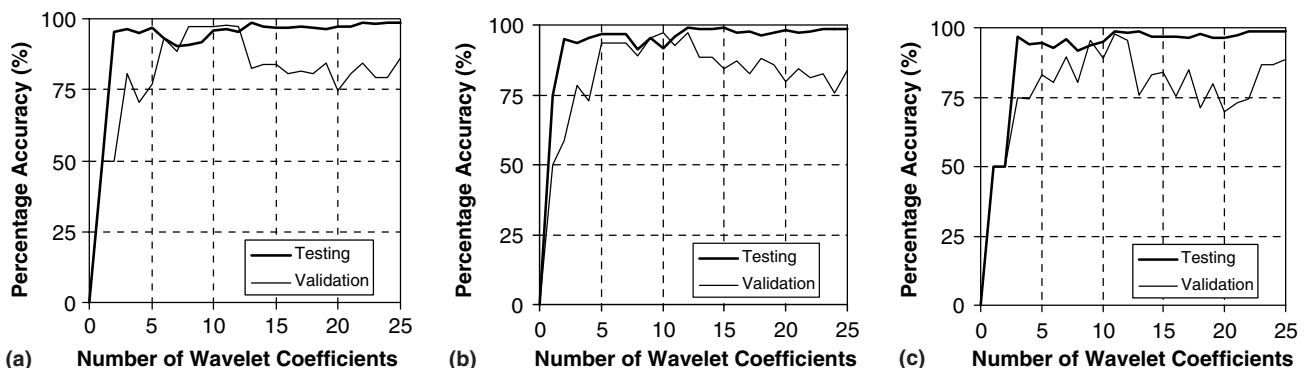


Fig. 11. Comparison between testing and validation classification accuracy of an ANN trained using back-propagation on (a) db6 data; (b) Moulin optimised data; and (c) Sherlock and Monro optimised data.

Table 3
Best classification results for the test set from the three classifiers using the output of the three wavelet transforms as the input to each classifier

Classifier type	Wavelet procedure	Atrial (%)	Ventricular (%)	Average (%)	Number of wavelet coefficients
LDA	Db6	95.3	95.6	95.5	3
	Moulin et al.	95.3	96.0	95.7	3
	Sherlock and Monro	89.1	94.7	91.9	3
Back-prop. ANN	Db6	93.8	99.1	96.5	11
	Moulin et al.	98.4	99.3	98.9	12
	Sherlock and Monro	98.4	99.0	98.7	11
GA trained ANN	Db6	96.9	94.8	95.9	8
	Moulin et al.	100	99.1	99.6	12
	Sherlock and Monro	98.4	97.7	98.1	10

unbiased counterpart only classified 41% correctly. Generally, the biased wavelet approach in the previously published work showed to be inferior to the results presented in Table 3.

Since the computing time required to train the neural networks was not significant on a standard PC for both back-propagation and artificial evolution, the methodology presented is also applicable to other databases which are likely to evolve as a by-product of the proliferation of wirelessly networkable wearable sensing schemes for ECG monitoring. Currently, in Europe there are significant incentives for the proliferation of web-based services, which have the potential of reducing the enormous workload of the public health services.

5. Conclusions

Very good classification results can be obtained by transforming time-variant ECG into the wavelet domain. All three methods resulted in very promising classifiers but in the end a neural network classifier trained by a genetic algorithm was shown to be best suited for the problem at hand. Although the filter banks resulting from the two wavelet optimisation strategies differed from each other at each decomposition level, both approaches could be efficiently used to generate a data vector that would be presented at the input of the classifier. It was also revealed that wavelet optimisation actually improved the results by allowing the neural networks to generalise better. This was particularly important because in this study we have chosen to work with a limited amount of ECG data. Although it can be argued that a GA could evolve a feedback structure as well as the connection weights (Schaffer et al., 1992), the results presented here suggest that such a complex feedback system may not be actually needed for this particular application.

References

Addison, P.S., 2002. *The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science, Engineering, Medicine and Finance*. The Institute of Physics Publishing, London.

- Angeline, P.J., Saunders, G.M., Pollack, J.B., 1994. An evolutionary algorithm that constructs recurrent neural networks. *IEEE Trans. Neural Networks* 5, 54–65.
- Chvatal, V., 1983. *Linear Programming*. Freeman, New York.
- Coelho, C.J., Galvão, R.K.H., Araújo, M.C.U., Pimentel, M.F., Silva, E.C., 2003. A linear semi-infinite programming strategy for constructing optimal wavelet transforms in multivariate calibration problems. *J. Chem. Inf. Comput. Sci.* 43 (3), 928–933.
- Daubechies, I., 1992. *Ten lectures on wavelets* CBMS-NSF Series in Applied Maths, vol. 61. SIAM, Philadelphia.
- Duda, R.O., Hart, P.E., Stork, D.G., 2001. *Pattern Classification*, second ed. John Wiley, New York.
- Fogel, D.B., 1995. Phenotypes, genotypes, and operators in evolutionary computation. In: *Proc. IEEE Conf. on Evolutionary Computation*, pp. 193–198.
- Galvão, R.K.H., Yoneyama, T., 1999. Improving the discriminatory capabilities of a neural classifier by using a biased-wavelet layer. *Internat. J. Neural Syst.* 9 (3), 167–174.
- Galvão, R.K.H., Yoneyama, T., 2004. A competitive wavelet network for signal clustering. *IEEE Trans. Systems Man Cybernet. Part B—Cybernetics* 34 (2), 1282–1288.
- Hancock, P.J.B., 1992. Genetic algorithms and permutation problems: A comparison of recombination operators for neural net structure specification. In: *Proc. Internat. Workshop on Combinations of GAs and ANNs*, pp. 108–122.
- Haykin, S., 1999. *Neural Networks: A Comprehensive Foundation*, second ed. Prentice Hall, New Jersey.
- Hettich, R., Kortanek, K.O., 1993. Semi-infinite programming—theory, methods, and applications. *SIAM Rev.* 35, 380–429.
- Jacobson, B., Webster, J.G., 1977. *Medicine and Clinical Engineering*. Prentice Hall, New Jersey.
- Kadambe, S., Boudreaux-Bartels, G.F., 1999. Wavelet transform-based QRS complex detector. *IEEE Trans. Biomed. Eng.* 46 (7), 838–848.
- Kudo, M., Sklansky, J., 2000. Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition* 33, 25–41.
- Lachenbruch, P.A., 1975. *Discriminant Analysis*. Hafner Press, New York.
- Langdon, W.B., Poli, R., 2002. *Foundations of Genetic Programming*. Springer-Verlag, London.
- Li, C., Zheng, C., Tai, C., 1995. Detection of ECG characteristic points using wavelet transforms. *IEEE Trans. Biomed. Eng.* 42 (1), 21–28.
- Misiti, M., Misiti, Y., Oppenheim, G., Poggi, J.M., 1996. *Wavelet Toolbox User's Guide*. Mathworks, Natick.
- MIT/BIH arrhythmia database directory, 1992. Technical Report BMEC TR010, MIT, Cambridge.
- Moody, G.B., Mark, R.G., 2001. The impact of the MIT/BIH arrhythmia database, history, lessons learned and its influence on current and future databases. *IEEE Eng. Med. Biol.* 20 (3), 45–50.

- Moulin, P., Anitescu, M., Kortanek, K.O., Potra, F.A., 1997. The role of linear semi-infinite programming in signal-adapted QMF bank design. *IEEE Trans. Signal Process.* 45, 2160–2174.
- Nocedal, J., Wright, S.J., 1999. *Numerical Optimization*. Springer Series in Operations Research. Springer Verlag, New York.
- Okada, M., 1979. A digital filter for the QRS complex detection. *IEEE Trans. Biomed. Eng.* 26 (12), 700–703.
- Pendharkar, P.C., Rodger, J.A., 2004. An empirical study of impact of crossover operators on the performance of non-binary genetic algorithm based neural approaches for classification. *Comput. Operat. Res.* 31, 481–498.
- Qian, S., Chen, D., 1996. *Joint Time–Frequency Analysis—Methods and Applications*. Prentice Hall PTR, New Jersey.
- Ramchandran, K., Vetterli, M., Herley, C., 1996. Wavelets, subband coding, and best bases. *Proc. IEEE* 84 (4), 541–560.
- Rangayyan, R.M., 2001. *Biomedical Signal Analysis—A Case-Study Approach*. IEEE-Wiley.
- Rich, E., Knight, K., 1991. *Artificial Intelligence*, second ed. McGraw-Hill, New York.
- Schaffer, J.D., Whitley, D., Eshelman, L.J., 1992. Combinations of genetic algorithms and neural networks: A survey of the state of the art. In: *Proc. Internat. Workshop on Combinations of GAs and ANNs*, pp. 1–37.
- Senhadji, L., Carrault, G., Bellanger, J.J., Passariello, G., 1995. Comparing wavelet transforms for recognizing cardiac patterns. *IEEE Trans. Med. Biol.* 13 (2), 167–173.
- Sherlock, B.G., Monro, D.M., 1998. On the space of orthonormal wavelets. *IEEE Trans. Signal Process.* 46, 1716–1720.
- Stone, M., 1978. Cross-validation: A review. *Math. Operation. Statist., Serie Statistics* 9, 127–138.
- Strang, G., Nguyen, T., 1996. *Wavelets and Filter Banks*. Wellesley-Cambridge, Massachusetts.
- Tabachnick, B.G., Fidell, L.S., 2001. *Using Multivariate Statistics*, fourth ed. Allyn and Bacon, Boston.
- Tuqun, J., Vaidyanathan, P.P., 2000. A state-space approach to the design of globally optimal FIR energy compaction filters. *IEEE Trans. Signal Process.* 48 (10), 2822–2838.
- Unser, M., 1993. On the optimality of ideal filters for pyramid and wavelet signal approximation. *IEEE Trans. Signal Process.* 41, 3591–3596.
- Vaidyanathan, P.P., 1993. *Multirate Systems and Filter Banks*. Prentice Hall, New Jersey.
- Vetterli, M., Kovacevic, J., 1995. *Wavelets and Subband Coding*. Prentice-Hall PTR, New Jersey.
- Yao, X., 1999. Evolving artificial neural networks. *Proc. IEEE* 87 (9), 1423–1447.