JrnlID 11265_ArtID 358_Proof# 1 - 14/03/09

J Sign Process Syst
DOI 10.1007/s11265-009-0358-z

Characterisation of Signal Modality: Exploiting Signal Nonlinearity in Machine Learning and Signal Processing

Q1

Beth Jelfs · Soroush Javidi · Phebe Vayanos · Danilo Mandic

Received: 3 March 2008 / Revised: 14 November 2008 / Accepted: 10 March 2009

Abstract A novel method for online tracking of the changes in the nonlinearity within both real-domain and complex–valued signals is introduced. This is achieved by a collaborative adaptive signal processing approach based on a hybrid filter. By tracking the dynamics of the adaptive mixing parameter within the employed hybrid filtering architecture, we show that it is possible to quantify the degree of nonlinearity within both real- and complex-valued data. Implementations for tracking nonlinearity in general and then more specifically sparsity are illustrated on both benchmark and real world data. It is also shown that by combining the information obtained from hybrid filters of different natures it is possible to use this method to gain a more complete understanding of the nature of the nonlinearity within a signal. This also paves the way for building multidimensional feature spaces and their application in data/information fusion.

Keywords Adaptive signal processing · Distributed signal processing · Collaborative signal processing · Convex optimisation · Machine learning · Wind modelling · EEG modelling · Hybrid filtering

B. Jelfs (✉) · S. Javidi · P. Vayanos · D. Mandic
Department of Electrical and Electronic Engineering, Imperial College London, London, UK
e-mail: beth.jelfs@ic.ac.uk

S. Javidi
e-mail: soroush.javidi@ic.ac.uk

P. Vayanos
e-mail: foivi.vayanos@ic.ac.uk

D. Mandic
e-mail: d.mandic@ic.ac.uk

1 Introduction

Signal modality characterisation reveals the changes in the nature of real world data (degree of sparsity, nonlinearity, stochasticity etc.) and as such is a key topic of multidisciplinary research. The applications of signal modality characterisation are only recently becoming apparent in signal processing and machine learning and are very important in online applications. When considering characterisation of signal modality we adhere to the definition of a linear signal from [5] as one which is a linear time-invariant system driven by white Gaussian noise measured by a static (possibly nonlinear) observation function. As such we define a nonlinear system as one which cannot be generated in this way. The range of signals spanned by just the characteristics of nonlinearity and stochasticity are shown in Fig. 1 (modified from [18]) and whilst there are some small areas which are well understood these tend to be extremes in nature, such as purely nonlinear deterministic signals (chaos), or linear and stochastic signals represented by autoregressive moving average (ARMA) models. These extremes however do not cover the majority of real world signals, and the presence of factors such as noise or uncertainty leads to most real world signals being represented in the areas (a), (b), (c) or '?'.

Knowing more about the nature of the signal being processed can provide valuable information in many areas such as health or weather conditions, and some aspects of this problem for the analysis of EEG data have been addressed in [12]. This knowledge can also be used to provide prior knowledge for the selection of appropriate models as the use of inappropriate models can lead to problems in their use or training and in some situations (such as the use of nonlinear

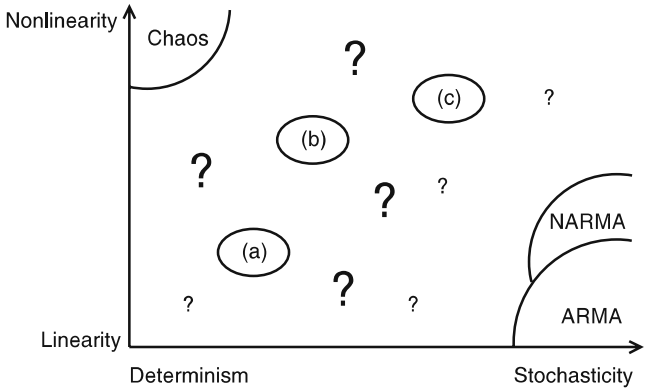


Figure 1 Deterministic vs. stochastic nature and linear vs. nonlinear nature of real world signals.

model in absence of nonlinearity) can add unnecessary computational complexity. Another benefit of tracking the degree of linearity in a signal in real-time, is for example, to provide prior knowledge to a blind algorithm. Combining the degree of nonlinearity for various different types of nonlinearity can be used as a signal "fingerprint", and when used in conjunction with other signal modality trackers [21] this can be a powerful tool for machine learning, detailing changes within the signal dynamics along time.

Many existing approaches to signal characterisation are based upon hypothesis testing, describing the signal in a statistical manner [19]. There is however a need for online approaches to signal characterisation which not only identify the nature of a signal but also track any changes in signal modality. Some disadvantages of existing online approaches are due to their tendency to rely on underlying models [16], making their application somewhat limited. To overcome these limitations we propose a much more flexible method based on collaborative adaptive filtering; by means of hybrid filters [2]. Whilst previous implementations of hybrid filters based on convex combinations of adaptive subfilters have focused mainly on the quantitative improvement in performance, our approach relies on observing the evolution of the mixing parameter. As such the hybrid filter is designed to have two constituent subfilters with sufficiently different characteristics so that when the mixing parameter is observed we can gain an insight into the nature of the signals. In addition through the evolution of the mixing parameter, we can also track any changes in the modality of the signals.

It is, however, much more complicated to achieve the tracking of modality change in the complex domain $\mathbb{C}$. The extensions of hybrid filters from $\mathbb{R}$ to $\mathbb{C}$ are non-trivial; this is due to the fact that the nature of nonlinearity in $\mathbb{C}$ is fundamentally different from that in $\mathbb{R}$. For instance, in the design of learning algorithms, it should be taken into account that the only continuously differentiable function in $\mathbb{C}$ is a constant (Liouville's theorem).

Complex-valued signals are either complex by design or are made complex by convenience of representation. An example of a real-valued signal which is best analysed in $\mathbb{C}$ is wind, where the fusion of the speed and direction creates a single complex-valued wind signal, seen in Fig. 2.

A first attempt to track the modality change of *complex* signals using hybrid filters was introduced in [21], where the changes between the split- and fully-complex natures of the data were tracked. The nonlinearity of the input signal was implicitly assumed, this however, may not necessarily be the case. Thus, before checking for the split- or fully-complex nonlinear nature, we first need to assess whether the input is linear or nonlinear.

Our underlying idea is to use a technique similar to that in [13], whereby each subfilter is designed so as to perform best, on either linear or nonlinear input. By making these subfilters collaborate and by tracking the values of the mixing parameter, we can then assess the degree of nonlinearity. We shall first describe the hybrid filter configuration and derive the update for the mixing parameter. Next, we shall introduce specific implementations of this structure, and the relevant algorithms used. Firstly to assess the degree of nonlinearity then extending to include the more specific case of sparseness of a signal as well as general nonlinearity before moving on to discuss tracking signal modality in the complex domain. The performance of the method is assessed using benchmark linear, nonlinear and sparse signals, as well as EEG data and a complex–valued wind signal.

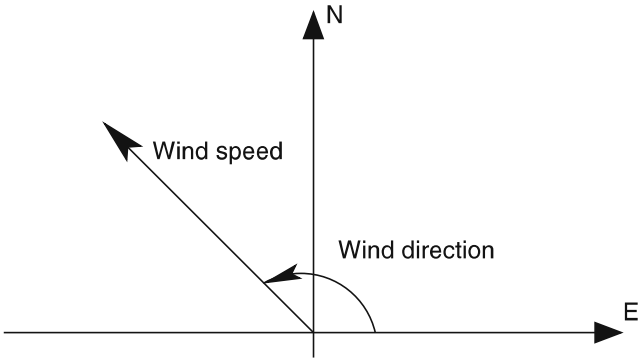


Figure 2 Wind as a complex vector.

2 Hybrid Filter Configuration

Hybrid filters have been previously introduced to improve the performance of adaptive filters, illustrating that by collaborative learning using a combination of subfilters of different characteristics it is possible to achieve better overall performance than that obtained from any of the individual subfilters [8]. One of the keys to designing the hybrid filter is the method in which the subfilters are combined, one simple but effective method to combine two subfilters is to combine the outputs of the subfilters in a convex manner. Convexity can be described as [4]

$$\lambda x + (1-\lambda)y \quad \text{where } \lambda \in [0,1] \tag{1}$$

For x and y being two points on a line, as shown in Fig. 3, their convex mixture Eq. 1 will lie on the same line between x and y. Hybrid filters using convex combinations of two subfilters both being trained by the same algorithm have been shown to perform well in stationary environments and always perform at least as well as the better of the two subfilters [1]. An alternative to this using a combination of two subfilters trained by two different algorithms has shown that by careful selection of the training algorithms it is possible to take the desired properties of both subfilters to give a better overall performance [13]. These hybrid filters have also been shown to improve the overall stability of the filter, as should one subfilter fail to converge the hybrid filter tracks the output of the second subfilter.

A hybrid filter, shown in Fig. 4, consists of two subfilters, each being adapted independently, with a convex combination of the two filters then taken as the output of the hybrid filter. The two subfilters within the hybrid filtering architecture operate in the prediction setting, sharing the common input vector $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)]^T$ for filters of length N. The outputs of the two subfilters are dependent on the algorithms used to train the subfilters and are given by $y_1(k) = \mathbf{x}^T(k)\mathbf{w}_1(k)$ and $y_2(k) = \mathbf{x}^T(k)\mathbf{w}_2(k)$. The corresponding weight vectors $\mathbf{w}_1(k) = [w_{1,1}(k), \ldots, w_{1,N}(k)]^T$ and $\mathbf{w}_2(k) = [w_{2,1}(k), \ldots, w_{2,N}(k)]^T$, where to preserve its inherent characteristics each subfilter is updated by its own error $e_1(k)$ and $e_2(k)$, using a common desired signal $d(k)$. The convex combination of the subfilter

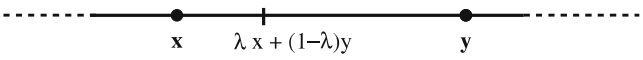


Figure 3 Convex combination of two points x and y.

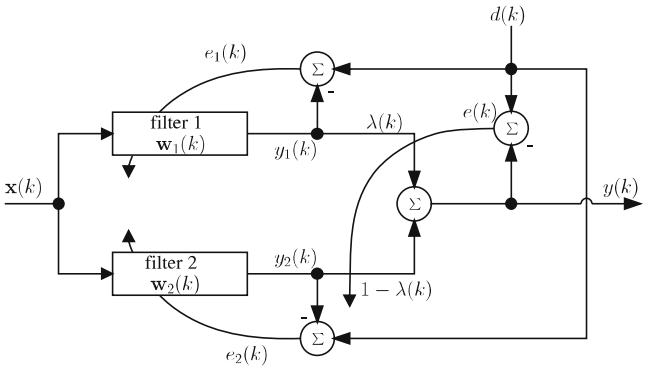


Figure 4 Hybrid filter structure.

outputs $y_1(k)$ and $y_2(k)$ forms the overall system output $y(k)$, given by

$$y(k) = \lambda(k)y_1(k) + (1-\lambda(k))\,y_2(k) \tag{2}$$

where $\lambda(k)$ is the mixing parameter, which is made adaptive, and is updated by minimising the cost function

$$E(k) = \frac{1}{2}|e(k)|^2 = \frac{1}{2}|d(k) - y(k)|^2 \tag{3}$$

We can obtain the update for $\lambda(k)$ using a stochastic gradient based adaptation, such as the LMS, whereby

$$\lambda(k+1) = \lambda(k) - \mu_\lambda \nabla_\lambda E(k)|_{\lambda=\lambda(k)} \tag{4}$$

and μ_λ is the step size. From Eqs. 2 and 4, the λ update can be shown to be

$$\begin{aligned}\lambda(k+1) &= \lambda(k) - \frac{\mu_\lambda}{2}\frac{\partial e^2(k)}{\partial \lambda(k)} \\ &= \lambda(k) + \mu_\lambda e(k)\,(y_1(k) - y_2(k))\end{aligned} \tag{5}$$

By understanding that the output of the hybrid filter will always be dominated by the better performing of the two subfilters, it is natural to assume that this information can be obtained by observing the adaptive mixing parameter λ. With this in mind we show that it is possible to design a hybrid filter which uses subfilters trained by algorithms with known different properties and that the behaviour of λ within such a combination will reveal not only which filter is currently giving the best response to the input signal, but also that with appropriately chosen subfilters the response of λ can then be used to reveal knowledge about the nature of the input signal.

Due to the convex nature of the hybrid filter, providing at least one of the subfilters converges the hybrid filter is guaranteed to converge [1, 2]. The key therefore is ensuring that the mixing parameter remains within

the range [0, 1]. To achieve this different approaches have been proposed, including the introduction of a sigmoid nonlinearity into the update. However, as the purpose of the proposed approach is to track the response of the mixing parameter we do not interfere with the evolution of λ, but a hard bound on the values of λ is used if $\lambda(k) > 1$ and $\lambda(k) < 0$.

2.1 Learning Algorithms

For the purposes of tracking nonlinearity in a signal the constituent finite impulse response (FIR) subfilters of the hybrid filter were trained one by a linear algorithm and the other by a saturation type nonlinear algorithm. The algorithms selected were the normalised least mean square (NLMS) algorithm [23] and the normalised nonlinear gradient descent (NNGD) algorithm [9]. These two algorithms were chosen to train the subfilters as the NLMS is widely used and known for its robustness and excellent steady state properties whereas the NNGD has faster convergence and better tracking capabilities making it more suited to nonlinear inputs than the NLMS.

The output of the NLMS trained subfilter y_{NLMS} is generated from

$$\begin{aligned} y_{NLMS}(k) &= \mathbf{x}^T(k)\mathbf{w}_{NLMS}(k) \\ e_{NLMS}(k) &= d(k) - y_{NLMS}(k) \\ \mathbf{w}_{NLMS}(k+1) &= \mathbf{w}_{NLMS}(k) \\ &\quad + \frac{\mu_{NLMS}}{\|\mathbf{x}(k)\|_2^2 + \varepsilon(k)} e_{NLMS}(k)\mathbf{x}(k) \end{aligned} \tag{6}$$

and y_{NNGD} is the corresponding output of the NNGD trained subfilter given by

$$\begin{aligned} y_{NNGD}(k) &= \Phi\left(net(k)\right) \\ net(k) &= \mathbf{x}^T(k)\mathbf{w}_{NNGD}(k) \\ e_{NNGD}(k) &= d(k) - y_{NNGD}(k) \\ \mathbf{w}_{NNGD}(k+1) &= \mathbf{w}_{NNGD}(k) \\ &\quad + \eta(k)\Phi'\left(net(k)\right) e_{NNGD}(k)\mathbf{x}(k) \\ \eta(k) &= \frac{1}{C + \left[\Phi'\left(net(k)\right)\right] \|\mathbf{x}(k)\|_2^2} \end{aligned} \tag{7}$$

where the step-size parameter of the NLMS filters is μ_{NLMS} and ε is the regularisation term. In the case of the NNGD $\Phi(\cdot)$ represents a nonlinear activation function and C a constant representing the ignored higher terms, for simulation purposes these were $\tanh(\cdot)$ and unity respectively.

As "nonlinearity" covers a wide range of signals, to highlight the ability to track more specific changes in signal modality the changes in the sparseness of a signal (a subset of nonlinearity) are also investigated. The subfilters of this hybrid filter were trained by the signed sparse LMS (SSLMS) [15] and the NLMS. The NLMS was selected for the nonsparse filter as it was found to be a better choice than the LMS due to its faster convergence speeds allowing it to adapt quickly to changes in the input signal (preventing the sparse filter from dominating). The output of the NLMS trained subfilter is given as above Eq. 6 and the output of the corresponding SSLMS trained subfilter $y_{SSLMS}(k)$ is given by

$$\begin{aligned} y_{SSLMS}(k) &= \mathbf{x}^T(k)\mathbf{w}_{SSLMS}(k) \\ e_{SSLMS}(k) &= d(k) - y_{SSLMS}(k) \\ \mathbf{w}_{SSLMS}(k+1) &= \mathbf{w}_{SSLMS}(k) \\ &\quad + \mu\left(|\mathbf{w}_{SSLMS}(k)| + \varepsilon\right) e_{SSLMS}(k)\mathbf{x}(k) \end{aligned} \tag{8}$$

3 Tracking Changes in Signal Nonlinearity

We shall now investigate the behaviour of λ for benchmark synthetic linear, nonlinear and sparse inputs. Values of λ were averaged over a set of 100 independent simulation runs, for the inputs described by a stable linear AR(4) process:

$$\begin{aligned} x(k) =& 1.79x(k-1) - 1.85x(k-2) + 1.27x(k-3) \\ & - 0.41x(k-4) + n(k) \end{aligned} \tag{9}$$

an AR(1) process

$$x(k) = 0.9x(k-1) + n(k) \tag{10}$$

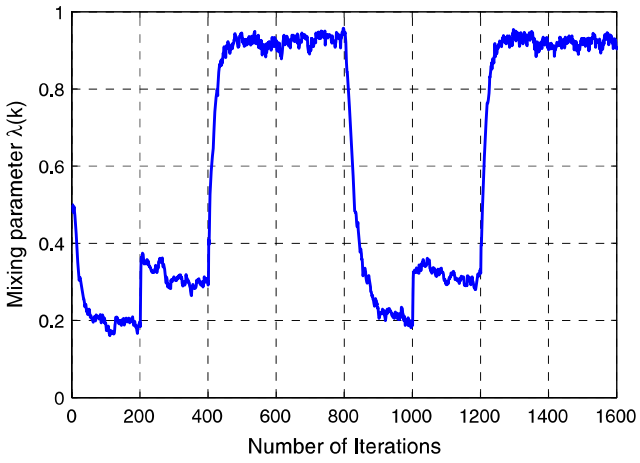


Figure 5 Evolution of the mixing parameters λ for hybrid filter combining NNGD and NLMS subfilters for an input signal alternating from linear AR(4) to AR(1) to nonlinear every 200 samples.

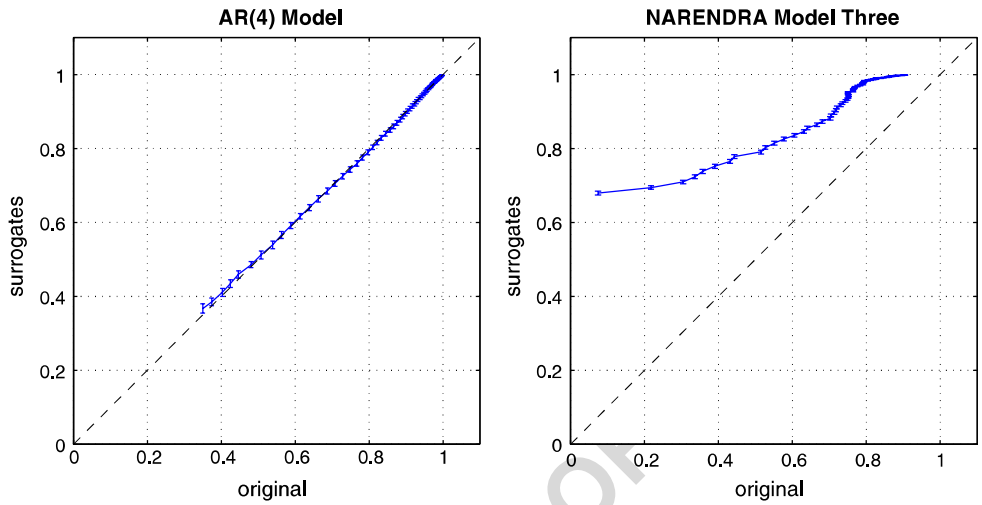


Figure 6 DVV scatter diagrams, obtained by plotting the target variance of the original data against the mean of the target variances of the surrogate data where *error bars* denote the standard deviation of the target variance of surrogate data.

(a) DVV scatter diagram for linear AR(4) (9)

(b) DVV scatter diagram for nonlinear signal (11)

and benchmark nonlinear signals [17]:

$$x(k+1) = \frac{x(k)}{1+x^2(k)} + n^3(k) \tag{11}$$

$$x(k) = \frac{x^2(k-1)(x(k-1)+2.5)}{1+x^2(k-1)+x^2(k-2)} + n(k-1) \tag{12}$$

where $n(k)$ is a zero mean, unit variance white Gaussian process. The convex combination of the NLMS and NNGD was presented with an input signal which alternated every 200 samples. For the first 200 samples the input signal was represented by the linear AR(4) Eq. 9, this was then followed at samples 200–400 by the linear AR(1) Eq. 10. After 400 samples the signal changed to the first of the nonlinear signals Eq. 11 and at 600 samples to the second nonlinear signal Eq. 12, after 800 samples the sequence was then repeated. The corresponding dynamics of the mixing parameter $\lambda(k)$ is shown in Fig. 5, where a value of $\lambda = 1$ corresponds to the output of the NNGD trained subfilter and a value of $\lambda = 0$ corresponds to the output of the NLMS trained subfilter. It is clear from Fig. 5 that the value of λ adapts to be dominated by the filter most suited to the current dynamics of the input signal. These results are supported by the offline statistical hypothesis testing results obtained through the delay vector variance technique. Figure 6 shows DVV results for both the AR(4) Eq. 9 and the nonlinear signal Eq. 11 [3], the scatter diagram for AR(4) signal lies on the bisector line, indicating its linear nature whereas that for the nonlinear signal deviates from the bisector line, indicating its nonlinear nature.[1]

[1]The DVV method is a test for signal nonlinearity. For more detail, see [5, 10].

Figure 7 shows the behaviour for both hybrid filters where the input signal alternated from linear AR(4) Eq. 9 to nonlinear Eq. 11 then back to linear then to sparse. The sparse signal was defined by a benchmark sparse distribution [15], with a 100 taps and four nonzero taps located located in positions [1, 30, 35, 85]. As expected (as sparsity can be considered a subset of nonlinearity) the nonlinear hybrid filter obtains similar results for both the nonlinear and sparse inputs, whereas the sparse hybrid filter shows a marked difference in levels of sparsity for the same inputs. These algorithms have also been shown to give good results not only on synthetically generated data but also real world data in the form of EEG data from epileptic seizures [7] and speech data [20]. It is natural therefore to combine these results to track not only changes in nonlinearity

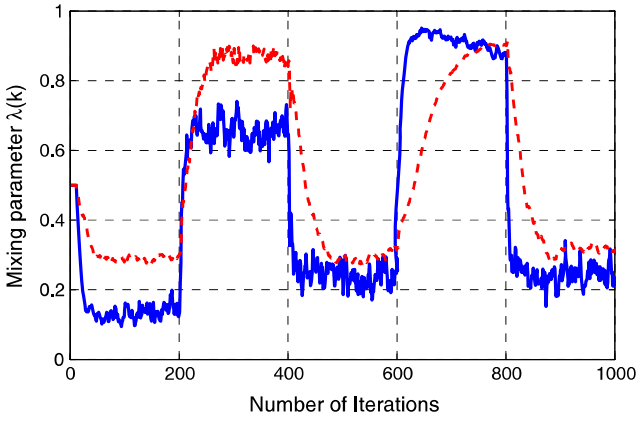


Figure 7 Evolution of the mixing parameters λ for hybrid filters combining NNGD and NLMS subfilters (*broken line*) and SSLMS and NLMS (*solid line*) for an input signal alternating from linear to nonlinear and linear to sparse every 200 samples.

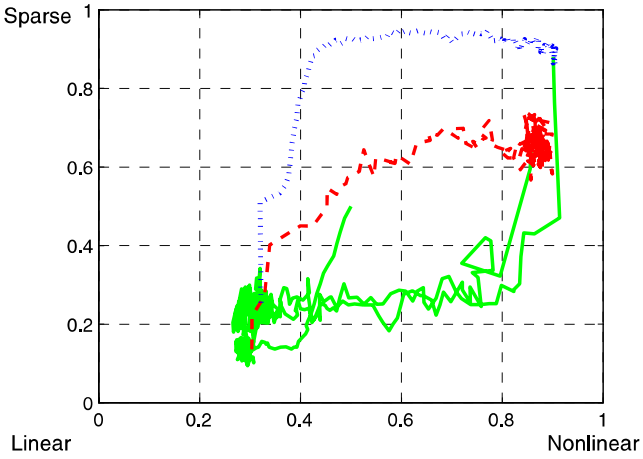
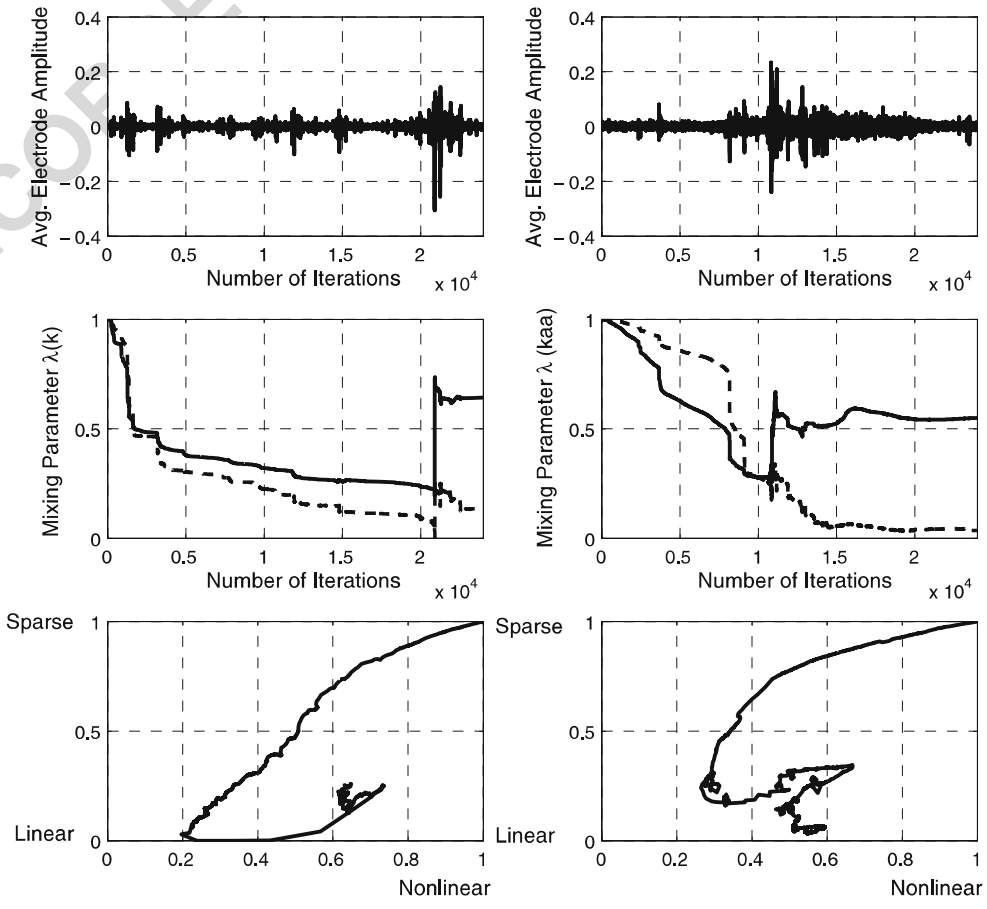


Figure 8 Comparison of the evolution of the mixing parameters for linear/nonlinear and sparse/nonsparse for an input signal alternating every 200 samples. *Solid line*: linear sections, *broken line*: nonlinear sections, *dotted line*: sparse sections.

but also at the same time changes in sparsity. This allows us to distinguish not only between linear and nonlinear signals but to distinguish also particular types of nonlinearity of interest allowing us to build up a more complete interpretation of the nature of the signal in question [6].

Figure 8 shows the response of λ for both the nonlinear hybrid filter and the sparse hybrid filter for the alternating input signal previously described, with the solid lines representing the linear sections, the broken lines the nonlinear sections and the dotted lines the sparse sections. For the linear sections although the evolution of the two λs do not follow the same path, there is an obvious correlation between them and the difference in responses can be attributed to the different learning rates of the subfilters of each hybrid filter. For the nonlinear and the sparse signals, however, the sparsity and saturation type nonlinearity are different phenomena and the sparse and nonlinear filter behaved differently. This representation is similar to the phase space representation in chaos theory, and allows for the signal modality characterisation to be considered within the framework of nonlinear dynamics. The results obtained from this method give a clearer picture of the nature of the nonlinearity of the signal, than any results obtained using a hybrid filter with nonlinear and sparse subfilters. This is due to the fact any signals which are sparse are also nonlinear so there is little

Figure 9 *Top panels*: EEG epileptic seizure data. *Middle panels*: corresponding evolution of λ *solid line*: nonlinear hybrid filter, *broken line*: sparse hybrid filter. *Bottom panels* comparison of nonlinearity and sparsity, evolution over time starting from coordinates (1,1).



advantage gained from the sparse subfilter over the nonlinear subfilter and therefore a hybrid filter using this combination would result in less clearly defined values of λ.

To demonstrate the application of this method to real-world data, two sets of EEG data showing the onset of epileptic seizures were analysed. Figure 9 shows the EEG data, along with the corresponding evolution of the mixing parameters λ for both hybrid filters and the resultant changes in nonlinearity against sparsity. These results show that the proposed approach can effectively detect changes in the nature of the EEG signals which can be very difficult to achieve otherwise, but it can also differentiate between changes in nonlinearity and sparsity.

4 Complex Hybrid Filter Update Algorithm

Consider the update of the mixing parameter $\lambda(k)$ Eq. 4—note that since the input to the filters is complex, the error $e(k)$ is also complex, and therefore [14]

$$\nabla_\lambda E(k)|_{\lambda=\lambda(k)} = \left\{ e(k)\frac{\partial e^*(k)}{\partial \lambda(k)} + e^*(k)\frac{\partial e(k)}{\partial \lambda(k)} \right\} \tag{13}$$

The two gradient terms from Eq. 13 can be evaluated as

$$\begin{aligned} \frac{\partial e(k)}{\partial \lambda(k)} &= \frac{\partial e_r(k)}{\partial \lambda(k)} + j\frac{\partial e_i(k)}{\partial \lambda(k)} \\ \frac{\partial e^*(k)}{\partial \lambda(k)} &= \frac{\partial e_r(k)}{\partial \lambda(k)} - j\frac{\partial e_i(k)}{\partial \lambda(k)} \end{aligned} \tag{14}$$

where $(\cdot)_r$ and $(\cdot)_i$ denote respectively the real and imaginary part of a complex number. Rewriting Eq. 2 in terms of its real and imaginary part and substituting into Eq. 3 yields

$$\begin{aligned} \frac{\partial e(k)}{\partial \lambda(k)} &= y_1(k) - y_2(k) \\ \frac{\partial e^*(k)}{\partial \lambda(k)} &= \Big(y_1(k) - y_2(k)\Big)^* \end{aligned} \tag{15}$$

Finally, the gradient Eq. 13 becomes

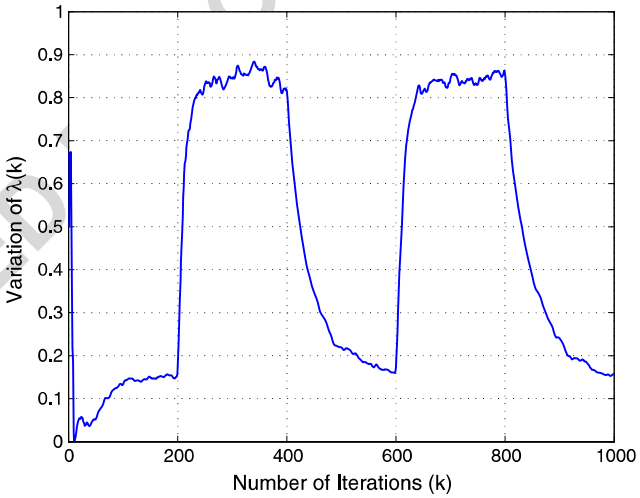
$$\nabla_\lambda E(k)|_{\lambda=\lambda(k)} = \Re\left\{ e(k)\Big(y_1(k) - y_2(k)\Big)^* \right\} \tag{16}$$

where $\Re(\cdot)$ denotes the real part of a complex number, which yields the mixing parameter update as

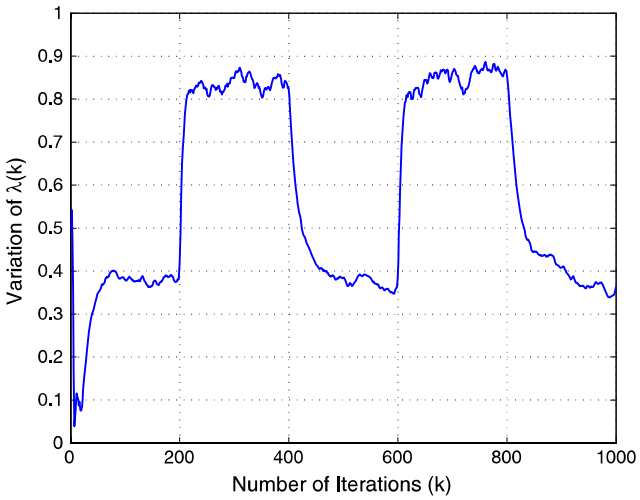
$$\lambda(k+1) = \lambda(k) + \mu_\lambda \Re\left\{ e(k)\Big(y_2(k) - y_1(k)\Big)^* \right\} \tag{17}$$

4.1 Complex Learning Algorithms

For the purposes of tracking nonlinearity in the complex domain, the subfilters of Fig. 4 are adapted using the CNGD [11] and CLMS [22], respectively. Their

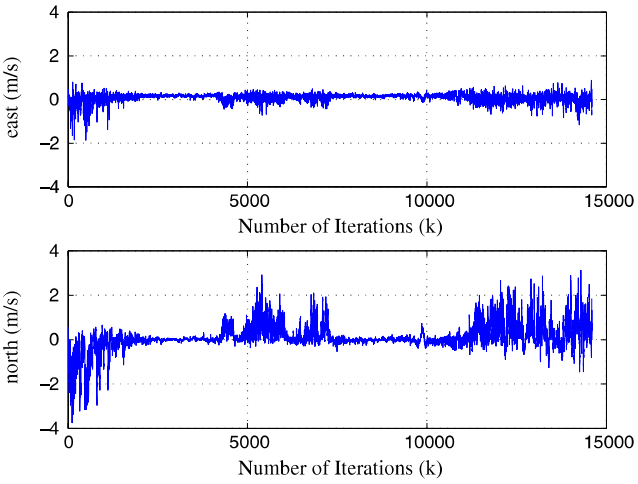


(a) Variation of the mixing parameter λ for the alternating inputs (10) and (11) ($\mu_{CNGD} = 0.08$, $\mu_{CLMS} = 0.08$, $\mu_\lambda = 50$)

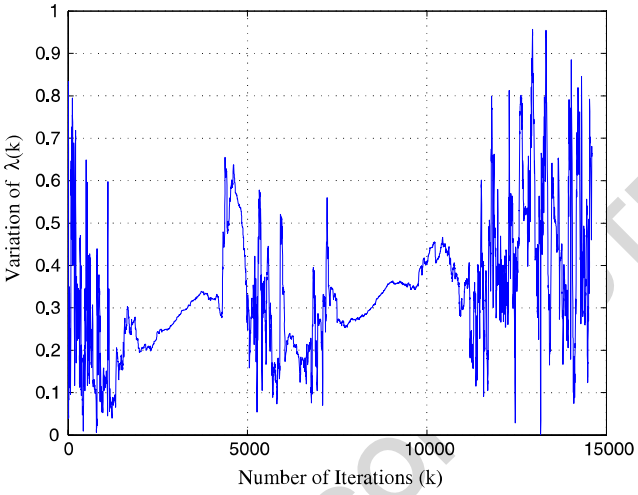


(b) Variation of the mixing parameter λ for the alternating inputs (9) and (12) ($\mu_{CNGD} = 0.2$, $\mu_{CLMS} = 0.2$, $\mu_\lambda = 60$)

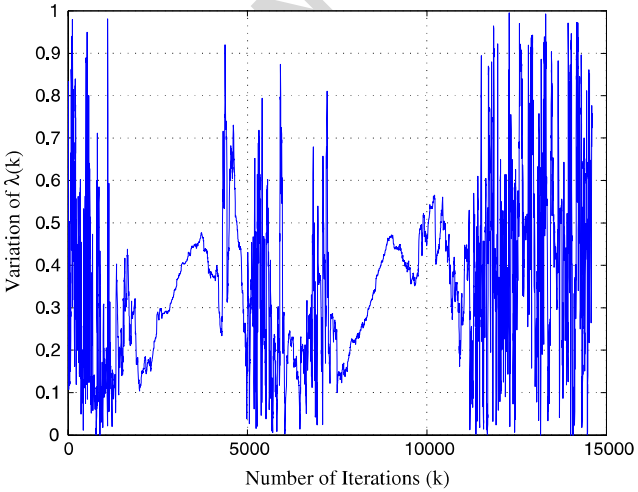
Figure 10 Hybrid combination of CNGD and CLMS, for input natures alternating between linear and nonlinear every 200 samples (**a**, **b**).



(a) The real and imaginary parts of complex–valued wind



(b) Variation of the mixing parameter λ for complex wind ($\mu_{CNGD} = 0.2, \mu_{CLMS} = 0.2, \mu_\lambda = 20$)



(c) Variation of the mixing parameter λ for complex wind ($\mu_{CNGD} = 0.2, \mu_{CLMS} = 0.2, \mu_\lambda = 60$)

Figure 11 Hybrid combination of CNGD and CLMS, for complex-valued wind data (**a–c**).

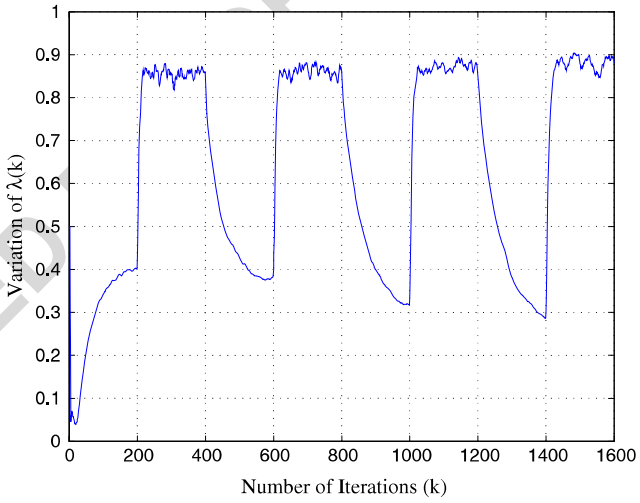
normalised variants CNNGD and CNLMS will also be used.

The linear CLMS update is given by

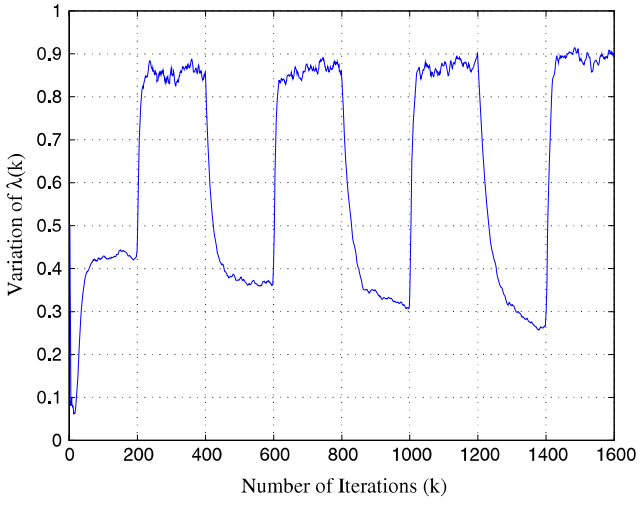
$$y_{CLMS}(k) = \mathbf{x}^T(k)\mathbf{w}_{CLMS}(k)$$

$$e_{CLMS}(k) = d(k) - y_{CLMS}(k)$$

$$\mathbf{w}_{CLMS}(k+1) = \mathbf{w}_{CLMS}(k) + \mu e_{CLMS}(k)\mathbf{x}^*(k) \qquad (18)$$

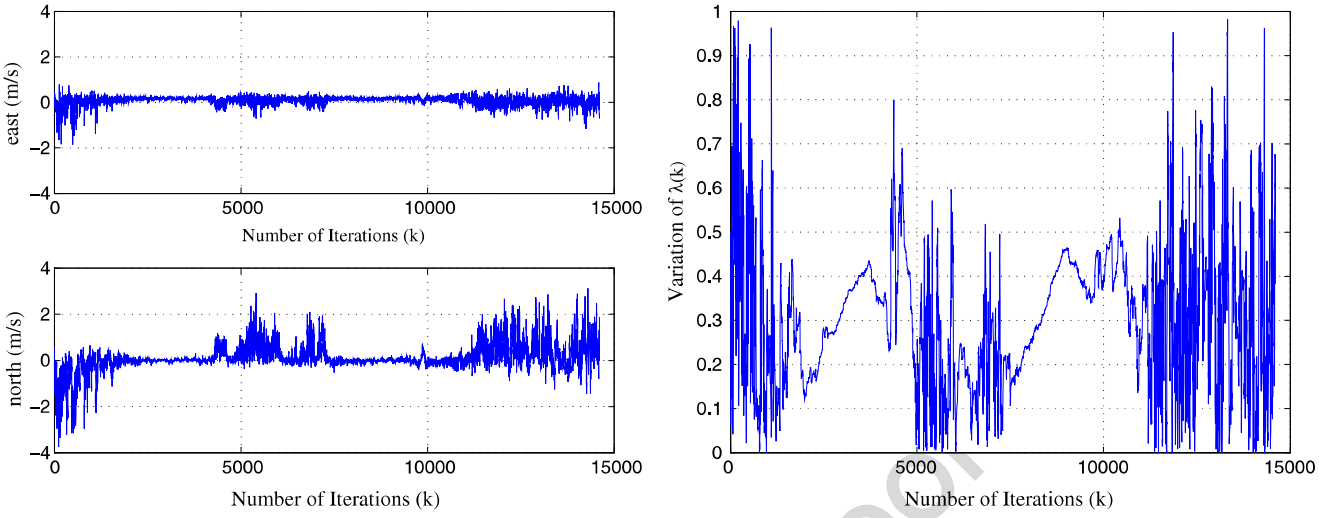


(a) Input nature alternating between linear (10) and nonlinear (11) ($\mu_{CNNGD} = 0.8, \mu_{CNLMS} = 0.8, \mu_\lambda = 50$)



(b) Input nature alternating between linear (9) and nonlinear (12) ($\mu_{CNNGD} = 0.8, \mu_{CNLMS} = 0.8, \mu_\lambda = 50$)

Figure 12 Mixing parameter λ at the output of the hybrid combination of CNNGD and CNLMS, for input nature alternating between linear and nonlinear every 200 samples.



(a) The real and imaginary parts of complex–valued wind

(b) Evolution of λ for complex wind

Figure 13 Evolution of λ in the hybrid combination of CNNGD and CNLMS for the wind data set, for $\mu_{CNNGD} = 0.8$, $\mu_{CNLMS} = 0.8$ and $\mu_\lambda = 60$ (**a**, **b**).

whereas the update for the adaptation of the nonlinear subfilter CNGD is given by

$$y_{CNGD}(k) = \Phi\left(\underbrace{\mathbf{x}^T(k)\mathbf{w}_{CNGD}(k)}_{net(k)}\right)$$

$$e_{CNGD}(k) = d(k) - y_{CNGD}(k)$$

$$\mathbf{w}_{CNGD}(k+1) = \mathbf{w}_{CNGD}(k) + \eta e_{CNGD}(k)\left[\Phi'(net(k))\right]^* \mathbf{x}^*(k) \quad (19)$$

The normalised variants, CNLMS and CNNGD, are specified by

$$\mu_{CNLMS} = \mu/\left(\|\mathbf{x}(k)\|_2^2 + \epsilon\right) \quad (20)$$

$$\eta_{CNNGD} = \eta/\left(\left[\Phi'(net(k))\right]^2 \|\mathbf{x}\|_2^2 + \epsilon\right) \quad (21)$$

5 Tracking of Nonlinearity within Complex Signals

For generality, two sets of synthesized benchmark signals and a real-world complex wind dataset were used in simulations. The linear and nonlinear processes considered were a stable complex autoregressive $AR(4)$ Eq. 9, and $AR(1)$ Eq. 10 and the benchmark nonlinear signals Eqs. 11 and 12 where $n(k) = n_r(k) + \jmath n_i(k)$ is a complex white Gaussian noise (CWGN), for which the real and imaginary parts are independent real WGN sequences $\sim \mathcal{N}(0, 1)$ and $\sigma_n^2 = \sigma_{n_r}^2 + \sigma_{n_i}^2$.

To illustrate the ability of the hybrid filter to track the modality changes within a signal, experiments were performed on alternating sequences of linear (Eq. 10 or Eq. 9) and nonlinear (Eq. 11 or Eq. 12) data. An additional set of experiments was conducted on a set of real-world wind data.[2] For all simulations, the initial weight vectors for both filters were set to zero and the filter order was $N = 10$. When a nonlinear CNGD or CNNGD was used, the nonlinearity at the output of the filter was the complex logistic function

$$\Phi(z) = 1/(1 + e^{-z}) \quad (22)$$

5.1 Combination of CNGD and CLMS

In the first set of experiments, the nature of the input alternated every 200 samples between linear and nonlinear. The evolution of λ is shown in Fig. 10 for two different settings. In both cases, μ_{CNGD} was set equal to μ_{CLMS}, and it was always possible to detect both the direction of the change from linear to nonlinear and vice versa, and the degree of such change, as illustrated by the values of λ approaching 0.85 for nonlinear data and 0.1 and 0.4 for linear data. Also, this approach was robust to changes in the relative values of μ_{CNGD} and μ_{CLMS}.

In another set, complex-valued wind input was used as the input to the hybrid filter. The wind is made

[2]The wind data with speed v and direction φ were made complex as $\mathbf{v} = ve^{i\varphi}$ [11].

complex through the heterogeneous fusion of the wind speed v and direction φ to form a complex signal $\mathbf{v} = ve^{i\varphi}$. Figure 11 shows the variation in λ for two different values. In both experiments, it was seen that for different values of μ_{CNGD} and μ_{CLMS} the changes in the wind nonlinearity were detected with similar results.

5.2 Combination of CNNGD and CNLMS

Next in order to overcome some problems with signal conditioning, CNNGD Eq. 21 and CNLMS Eq. 20 were combined in a hybrid fashion. The results shown in Fig. 12 demonstrate the robustness of this combination compared to that from Fig. 10. Indeed, by setting $\mu_{CNNGD} = \mu_{CNLMS} = 0.8$, the hybrid filter performed well on a range of synthetically generated signals and accurately detected both the direction of the change from linear to nonlinear and vice versa, and the degree of the change. Furthermore, it can be seen that as the two transversal filters converge, the tracking of the degree of nonlinearity in the input is improved in terms of the range swept by λ (due to learning).

Next, the experiments were performed on complex-valued wind data, and the simulation results are shown in Fig. 13. The combination of CNNGD and CNLMS was clearly capable of tracking changes in the linear/nonlinear nature of the intermittent and nonstationary wind. The wind was changing its nature in the region between (1–2500), (5000–7000) and (10000–15000) samples, which was correctly reflected in the values of λ. For steady wind, the nature of wind exhibited a medium degrees of nonlinearity. In conclusion, the combination of CNNGD and CNLMS provided excellent results in the identification of signal nonlinearity on a broad range of inputs, provided the individual filters converged.

6 Conclusions

A method for the tracking of signal modality in an online manner has been introduced. It was shown that by using the a hybrid filter configuration and monitoring the variation of the mixing parameter, it is possible to distinguish changes in the signal nature in terms of sparsity and nonlinearity. This was shown for signals in the real domain, as well as those in the complex domain. We have also presented a method by which it is possible to build on this information by combining the responses of several mixing parameters to obtain a more complete understanding of the nature of signals. The simulation results have shown that the proposed approach is capable of tracking the nonlinearity and sparsity within both synthesized and real-world EEG and complex-valued wind data.

Acknowledgements We would like to thank Prof. Kazuyuki Aihara, Institute of Industrial Science, University of Tokyo, Japan for providing the wind data.

References

1. Arenas-Garcia, J., Figueiras-Vidal, A., & Sayed, A. (2005). Steady state performance of convex combinations of adaptive filters. In *Proceedings IEEE international conference on acoustics, speech and signal processing (ICASSP '05)* (Vol. 4, pp. 33–36).
2. Arenas-Garcia, J., Figueiras-Vidal, A., & Sayed, A. (2006). Mean-square performance of a convex combination of two adaptive filters. *IEEE Transactions on Signal Processing, 54*(3), 1078–1090.
3. Chen, M., Rutkowski, T., Jelfs, B., Souretis, G., Cao, J., & Mandic, D. (2007). Assessment of nonlinearity in brain electrical activity: A DVV approach. In *Proceedings RISP international workshop on nonlinear circuits and signal processing*.
4. Cichocki, A., & Unbehauen, R. (1993). *Neural networks for optimisation and signal processing*. New York: Wiley.
5. Gautama, T., Mandic, D., & Van Hulle, M. (2004). The delay vector variance method for detecting determinism and nonlinearity in time series. *Physica D, 190*(3–4), 167–176.
6. Jelfs, B., & Mandic, D. (2008). Signal modality characterisation using collaborative adaptive filters. In *Proceedings IAPR workshop on cognitive information processing*.
7. Jelfs, B., Vayanos, P., Chen, M., Goh, S., Boukis, C., Gautama, T., et al. (2006). An online method for detecting nonlinearity within a signal. In *Knowledge-based intelligent information and engineering systems* (pp. 1216–1223).
8. Kozat, S., & Singer, A. (2000). Multi-stage adaptive signal processing algorithms. In *Proceedings of the IEEE sensor array and multichannel signal processing workshop* (pp. 380–384).
9. Mandic, D. (2000). NNGD algorithm for neural adaptive filters. *Electronics Letters, 36*(9), 845–846.
10. Mandic, D., Chen, M., Gautama, T., Van Hulle, M., & Constantinides, A. (2008). On the characterisation of the deterministic/stochastic and linear/nonlinear nature of time series. *Proceedings of the Royal Society A, 464*(2093), 1141–1160.
11. Mandic, D., & Goh, S. (2008). *Complex valued nonlinear adaptive filters: A neural network approach*. New York: Wiley. Q2
12. Mandic, D., Golz, M., Kuh, A., Obradovic, D., & Tanaka, T. (Eds.) (2007). *Signal processing techniques for knowledge extraction and information fusion*. New York: Springer.
13. Mandic, D., Vayanos, P., Boukis, C., Jelfs, B., Goh, S., Gautama, T., et al. (2007). Collaborative adaptive learning using hybrid filters. In *ICASSP 2007* (Vol. 3, pp. 921–924).
14. Mandic, D., Vayanos, P., Javidi, S., Jelfs, B., & Aihara, K. (2008). Online tracking of the degree of nonlinearity within complex signals. In *ICASSP 2008*.
15. Martin, R., Sethares, W., Williamson, R., & Johnson, C., Jr. (2002). Exploiting sparsity in adaptive filters. *IEEE Transactions on Signal Processing, 50*(8), 1883–1894.

16. Mizuta, H., Jibu, M., & Yana, K. (2000). Adaptive estimation of the degree of system nonlinearity. In *Proceedings IEEE adaptive systems for signal processing and control symposium (AS-SPCC)* (pp. 352–356).
17. Narendra, K., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks, 1*(1), 4–27.
18. Schreiber, T. (1999). Interdisciplinary application of nonlinear time series methods. *Physics Reports, 308*(1), 1–64.
19. Schreiber, T., & Schmitz, A. (1997). Discrimination power of measures for nonlinearity in a time series. *Physical Review E, 55*(5), 5443–5447.
20. Vayanos, P., Chen, M., Jelfs, B., & Mandic, D. (2007). Exploiting nonlinearity in adaptive signal processing. In *Proceedings ISCA workshop on non linear speech processing*. New York: Springer.
21. Vayanos, P., Goh, S., & Mandic, D. (2006). Online detection of the nature of complex-valued signals. In *Proceedings of the 16th IEEE signal processing society workshop on machine learning for signal processing* (pp. 173–178).
22. Widrow, B., McCool, J., & Ball, M. (1975). The complex LMS algorithm. *Proceedings of the IEEE, 63*(4), 719–720.
23. Widrow, B., & Stearns, S. (1985). *Adaptive signal processing*. Englewood Cliffs: Prentice-Hall.