It is important to mention that the design of nonlinear maximally decimated decomposition schemes is still an open problem. However, we believe that, although we have presented just a heuristic approach to such a design, its strength lies in the fact that it opens new research possibilities towards finding more effective design methods.

 7 January 2000
Electronics Letters Online No: 20000592
DOI: 10.1049/el:20000592

E.S. Cardoso and E.A.B. da Silva (*PEE/COPPE/DEL/EE/Universidade Federal do Rio de Janeiro, Cx. P. 68504, Rio de Janeiro, RJ, 21945-970, Brazil*)

E-mail: xaua@lps.ufrj.br

References

1 EGGER, O., LI, W., and KUNT, M.: 'High compression image coding using an adaptive morphological subband decomposition', *Proc. IEEE*, 1995, **83**, pp. 272–287
2 FLORÊNCIO, D.A.F., and SCHAFER, R.W.: 'Perfect resconstructing nonlinear filter banks'. Proc. 1996 ICASSP Conf., 1996, Vol. III, pp. 1814–1817
3 HAMPSON, F.J., and PESQUET, J.-C.: 'M-band nonlinear subband decompositions with perfect reconstruction', *IEEE Trans.*, 1998, **IP-7**, pp. 1547–1560
4 DAUBECHIES, I., and SWELDENS, W.: 'Factoring wavelet transforms into lifting steps'. Technical Report, Bell Laboratories, Lucent Technologies, 1996
5 SERRA, J.: 'Image analysis and mathematical morphology' (Academic Press, 1982)
6 SHAPIRO, J.M.: 'Embedded image coding using zerotrees of wavelet coefficients', *IEEE Trans.*, 1993, **ASSP-41**, pp. 3445–3462
7 ANTONINI, M., BARLAUD, M., MATHIEU, P., and DAUBECHIES, I.: 'Image coding using wavelet transform', *IEEE Trans.*, 1992, **IP-1**, pp. 205–220

NNGD algorithm for neural adaptive filters

D.P. Mandic

A novel normalised nonlinear gradient descent (NNGD) algorithm for training neural adaptive feedforward filters is presented. The algorithm is based on minimisation of the instantaneous prediction error for contractive activation functions of a neuron, and provides an adaptive learning rate. Normalisation is performed *via* calculation of the product of the tap input power to the filter and the squared first derivative of the activation function of a neuron. The NNGD algorithm outperforms a gradient based algorithm for use in a neural adaptive filter, as well as the standard least mean squares (LMS) and normalised LMS algorithms. To support the analysis, simulation results on real speech are provided.

Introduction: The least mean squares (LMS) algorithm is the most popular algorithm for linear adaptive filtering. However, its inherent limitations, especially when dealing with nonlinear and nonstationary signals, have motivated researchers to try to improve its performance. Some LMS based algorithms which have improved convergence properties are the *a posteriori* LMS [1] and normalised LMS (NLMS) algorithms [2]. However, these linear algorithms suffer from slow convergence and coupling effects when dealing with nonlinear and nonstationary signals such as speech. Attempts have also been made to devise nonlinear neural adaptive filters [3]. However, they too can suffer from slow convergence speed, due to the gradient descent algorithm employed, and the fact that the learning rate η and the slope of the nonlinearity within the filter are interrelated [4]. In this Letter, following the approach from [5, 6], a normalised nonlinear gradient descent (NNGD) algorithm is presented for use in a feedforward neural adaptive filter and it is shown that the algorithm exhibits a higher performance than both the LMS and NLMS algorithms, as well as that of a common gradient descent trained neural adaptive filter.

Derivation of optimal adaptive learning rate: The equations that define the adaptation for a neural adaptive filter with one neuron are

$$e(k) = d(k) - \Phi\big(\mathbf{x}^T(k)\mathbf{w}(k)\big) \quad (1)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \eta\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)e(k)\mathbf{x}(k) \quad (2)$$

where $e(k)$ is the instantaneous error at the output neuron, $d(k)$ is some training (desired) signal, $\mathbf{x}(k) = [x_1(k), ..., x_N(k)]^T$ is the input vector, $\mathbf{w}(k) = [w_1(k), ..., w_N(k)]^T$ is the weight vector, $\Phi(\cdot)$ represents a nonlinear activation function of a neuron, and $(\cdot)^T$ denotes the vector transpose. The learning rate η is supposed to be a small positive real number.

When the error term (eqn. 1) is expanded with a Taylor series [5], we have

$$\begin{aligned} e(k+1) = e(k) &+ \sum_{i=1}^{N} \frac{\partial e(k)}{\partial w_i(k)}\Delta w_i(k) \\ &+ \frac{1}{2!}\sum_{i=1}^{N}\sum_{j=1}^{N} \frac{\partial^2 e(k)}{\partial w_i(k)\partial w_j(k)}\Delta w_i(k)\Delta w_j(k) \\ &+ \cdots \end{aligned} \quad (3)$$

From eqn. 1, the first partial derivatives can be obtained as follows:

$$\frac{\partial e(k)}{\partial w_i(k)} = -\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)x_i(k) \quad i = 1, 2, ..., N \quad (4)$$

whereas the weight correction is given by

$$\Delta w_i(k) = w_i(k+1) - w_i(k) = \eta\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)e(k)x_i(k) \qquad i = 1, 2, ..., N \quad (5)$$

The second partial derivatives are calculated as follows:

$$\frac{\partial^2 e(k)}{\partial w_i(k)\partial w_j(k)} = -\Phi''\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)x_i(k)x_j(k) \qquad i, j = 1, 2, ..., N \quad (6)$$

Combining the above equations, we have

$$\begin{aligned} e(k+1) = e(k) &- \eta\left[\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)\right]^2 e(k)\sum_{i=1}^{N} x_i^2(k) \\ &- \frac{1}{2!}\eta^2 e^2(k)\left[\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)\right]^2 \\ &\times \Phi''\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)\sum_{i=1}^{N}\sum_{j=1}^{N} x_i^2(k)x_j^2(k) + \cdots \end{aligned} \quad (7)$$

If we use the logistic activation function of a neuron $\Phi(v) = 1/(1 + e^{-\beta v}$, with slope β, and positive and normalised input data, to match the range of the function, then the second derivative Φ'' is positive for the range of interest. If, for the time being, we neglect the second order derivatives of Φ, then the error term becomes

$$e(k+1) = e(k)\left[1 - \eta\left[\Phi'\big(\mathbf{x}^T(k)\mathbf{w}(k)\big)\right]^2 \|\mathbf{x}(k)\|_2^2\right] \quad (8)$$

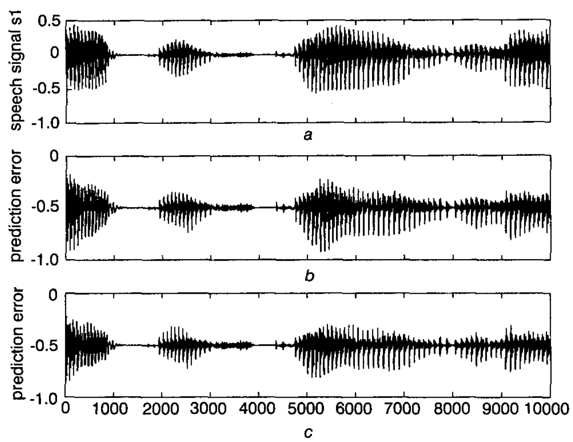
which is equal to zero for

$$\eta_{OPT} = \frac{1}{[\Phi'(\mathbf{x}^T(k)\mathbf{w}(k))]^2\|\mathbf{x}(k)\|_2^2} \quad (9)$$

However, bearing in mind the bounds on the values of higher derivatives of Φ, for a contractive activation function ($\beta < 4$) the following learning rate for a normalised GD based algorithm for a neural adaptive filter can be adopted:

$$\eta_{OPT}(k) = \frac{1}{C + [\Phi'(net(k))]^2\|\mathbf{x}(k)\|_2^2} \quad (10)$$

where $net(k) = \mathbf{x}^T(k)\mathbf{w}(k)$, and C is a constant comprising the contributions from higher order terms from eqn. 3. The physical meaning of such an adaptive learning rate is self-normalisation of the algorithm, since the magnitude of the learning rate varies in time with the tap input power and the gradient in the state space of the filter.



563/1

Fig. 1 *Comparison between GD and NNGD algorithms*

a Speech signal $s1$
b Prediction error for nonlinear GD
Gain = 4.923 dB (18.8908 dB), $N = 10$, $\beta = 1$, $\eta = 0.3$
c Prediction error for normalised nonlinear GD
Gain = 6.9528 dB (20.9205 dB), $N = 10$, $\beta = 1$

Simulations: Simulations were carried out on speech, a nonlinear and nonstationary input signal. In the first experiment, the speech signal was denoted by $s1$, the tap length was $N = 10$, the learning rate was $\eta = 0.3$, and the slope of the activation function Φ was $\beta = 1$. Constant C was chosen to be equal to unity. The prediction errors for a chosen speech signal $s1$ and the NGD and NNGD algorithms are shown in Fig. 1. Clearly, in this case, the NNGD algorithm outperforms the NGD algorithm. The quantitative performance measure was the standard prediction gain, a logarithmic ratio between the expected signal and error variances $R_p = 10\log(\hat{\sigma}_s^2/\hat{\sigma}_e^2)$. Another measure, which is the logarithm of the ratio of the signal to noise power, is given in parentheses. In another experiment, the performance of neural adaptive algorithms was compared with the performances of the LMS and NLMS algorithms for a speech signal $s2$. The setting was the same as for $s1$, except that β was chosen to be $\beta = 4$, since this value makes Φ close to the linear function in the vicinity of the origin. For this setting, the prediction gain for the LMS algorithm was 7.24 and 8.26dB for the NLMS algorithm, 7.67dB for a nonlinear GD algorithm, and 9.28dB for the NNGD algorithm, confirming the analysis from the preceding Section.

Conclusions: A normalised nonlinear gradient descent (NNGD) algorithm for training neural adaptive feedforward filters has been presented. The algorithm was derived based on a Taylor series expansion of the instantaneous output error concerning the current point in the state space of the filter. This leads to the optimal learning rate which minimises the instantaneous output error of the filter. The NNGD has been shown to outperform the LMS, NLMS, and nonlinear GD algorithms when applied to the prediction of some nonlinear and nonstationary signals.

1 March 2000
Electronics Letters Online No: 20000631
DOI: 10.1049/el:20000631

D.P. Mandic (*School of Information Systems, University of East Anglia, Norwich, NR4 7TJ, United Kingdom*)

E-mail: d.mandic@uea.ac.uk

References

1 TREICHLER, J.R., JOHNSON, C.R. Jr., and LARIMORE, M.G.: 'Theory and design of adaptive filters' (John Wiley & Sons, 1987)

2 HAYKIN, S.: 'Adaptive filter theory' (Prentice-Hall, Third Edn., 1996)

3 MANDIC, D.P., and CHAMBERS, J.A.: 'Relations between the a priori and a posteriori errors in nonlinear adaptive neural filters', *to be published in Neural Comput.*, 1999,

4 MANDIC, D.P., and CHAMBERS, J.A.: 'Relating the slope of the activation function and the learning rate within a recurrent neural network', *Neural Comput.*, 1999, **11**, (5), pp. 1069–1077

5 MANDIC, D.P., and CHAMBERS, J.A.: 'Towards an optimal learning rate for backpropagation', *Neural Process. Lett.*, 2000, **11**, (1), pp. 1–5

6 SORIA-OLIVAS, E., CALPE-MARAVILLA, J., GUERRERO-MARTINEZ, J.F., MARTINEZ-SOBER, M., and ESPI-LOPEZ, J.: 'An easy demonstration of the optimum value of the adaptation constant in the LMS algorithm', *IEEE Trans. Education*, 1998, **41**, (1), p. 81

Robust word boundary detection using fuzzy logic

F. Beritelli

A pattern recognition approach to robust word boundary detection in adverse acoustic noise conditions is proposed. The algorithm uses four simple differential parameters calculated in the time domain and pattern matching based on a set of six fuzzy rules extracted by a hybrid learning tool. The experimental results demonstrate that the new endpoint detector outperforms traditional methods, in particular in the presence of high levels of background noise.

Introduction: In recent years the growth of multimedia applications has increased the demand for new and more efficient speech command and control systems for man-machine interaction. In this context, the new application scenarios will require systems and methodologies able to guarantee good performance levels, even under adverse acoustic noise conditions, with as low a computational load as possible [1]. As the recognition rate of an isolated word recogniser strongly depends on the accuracy of the end point detector (EPD) [1, 2], for the effective automatic recognition of speech a robust word boundary detection algorithm is essential [3].

In this Letter we propose a new fuzzy logic-based boundary detection algorithm that meets the requirements of both computational simplicity and robustness to background noise. The fuzzy end point detector (FEPD) uses a set of four simple differential parameters and a matching phase based on a set of fuzzy rules. Experimental evaluation shows that the FEPD outperforms traditional word detection methods.

Description of algorithm: The architecture of the endpoint detector proposed is based on a pattern recognition approach. More specifically, it consists of pre-processing the speech signal, extracting its significant features, a matching phase, a post-processing module and finally a decision block. Pre-processing of the speech signal consists of 140Hz highpass filtering in order to eliminate the undesired low-frequency components.

To guarantee robust word boundary detection in the presence of high noise levels, rather than using absolute parameters such as energy, correlation and zero crossing, in this Letter we propose a different approach based on a set of parameters differentiated with respect to a local average. More specifically, the four differential parameters used for speech/noise classification are calculated in a window of 80 speech samples (a 10ms frame, sampling at 8kHz) and are: the full-band energy difference ΔE_f, the low-band energy difference ΔE_l, the zero-crossing difference ΔZC and the spectral distortion ΔS [4]. For the matching phase a new methodology is used which has the advantage of exploiting all the information in the input pattern by means of a set of six fuzzy rules automatically extracted by a hybrid learning tool [5]. The fuzzy system has the task of mapping the pattern of input parameters onto a scalar value, ranging between 0 and 1, which indicates the degree of membership in the voice inactivity/activity classes [6].

To reduce sharp variations in the fuzzy system output, it is post-processed by a seventh order median filter. Finally, the decision module, by means of a threshold comparison, returns the start and the ending point. Experimentally we chose a threshold value of $T_h = 0.9$ in that, observing the output of the post-processing module for several types of background noise and different signal-to-noise ratios, it was rarely observed to exceed 0.9 in seg-