Signal Processing 22 (1991) 77-93
Elsevier

Reconstruction of binary signals using an adaptive radial-basis-function equalizer

S. Chen, G.J. Gibson[1], C.F.N. Cowan and P.M. Grant
Department of Electrical Engineering, University of Edinburgh, Mayfield Road, Edinburgh EH9 3JL, United Kingdom

Received 6 February 1990
Revised 25 May 1990

Abstract. This paper investigates the application of a non-linear structure to the adaptive channel equalization of a bipolar signal passed through a dispersive channel in the presence of additive noise. It is shown that the optimal equalization solution under the classical architecture of making decisions symbol-by-symbol is an inherently non-linear problem and therefore some degree of non-linear decision making ability is desirable in the equalizer structure. The radial-basis-function structure is considered as an adaptive equalizer and simulation examples are included to compare its performance with the optimal equalization solution. A brief comparison with the maximum likelihood sequence estimation is given and it is demonstrated that the non-linear filter approach can also be applied to the decision feedback equalizer.

Zusammenfassung. Dieser Beitrag untersucht die Anwendung einer nichtlinearen Struktur auf die adaptive Entzerrung eines bipolaren Signals, das über einen dispersiven und additiv durch Rauschen gestörten Kanal übertragen wurde. Es wird gezeigt, daß die Lösung mit optimaler Entzerrung unter Verwendung der klassischen Architektur mit einer Entscheidung Symbol für Symbol von Haus aus ein nichtlineares Problem ist und daß deswegen in gewisssem Umfang eine Fähigkeit zu nichtlinearer Entscheidungsfindung in der Entzerrerstruktur wünschenswert ist. Die Radial-Basisfunktions-Struktur wird als adaptiver Entzerrer betrachtet, und es werden Simulationsbeispiele mit aufgenommen, in denen ihre Leistungsfähigkeit mit derjenigen des optimalen Entzerrers verglichen wird. In aller Kürze wird auch mit dem Maximum-Likelihood-Schätzer der Signalfolge verglichen, und es wird nachgewiesen, daß der Ansatz mit der nichtlinearen Filterung auch auf den Entzerrer mit Entscheidungsrückkopplung anzuwenden ist.

Résumé. Cet article étudie l'application d'une structure non-linéaire à l'égalisation adaptative d'un signal bipolaire transmis par un canal dispersif en présence de bruit blanc additif. Il est montré que l'égalisation optimale avec l'architecture classique de prise de décision symbole par symbole est de manière inhérente un problème non-linéaire et de ce fait un certain degré de non-linéarité pour la prise de décision de l'égalisateur est souhaitable. Nous considérons pour l'égalisateur adaptatif une structure fondée sur les fonctions au base radiale (en anglais radial-basis-functions, RBF) et des exemples de simulation sont inclus à des fins de comparaison avec la solution d'égalisation optimale. Nous présentons une brève comparaison avec l'estimation séquentielle à maximum de vraisemblance et nous démontrons que l'approche par filtrage non-linéaire peut être également appliquée à l'égalisateur à décision par rétroaction.

Keywords. Channel equalization, radial basis functions, adaptive algorithms.

1. Introduction

This paper considers the problem of reconstructing digital signals which have been passed through a dispersive channel and then corrupted with additive noise. This problem is known as the channel equalization in communications. The present study is mainly concerned with the classical equalizer architecture which has two typical features. The operation of the equalizer at each sample instant is based on a finite number of channel observations and decisions are made symbol-by-symbol. Traditional techniques for communications channel

[1] Current address: Plessey Electronic Systems Research Ltd., Roke Manor, Romsey, Hants SO5 0ZN, United Kingdom.

equalization are based on linear finite filters. Adaptive linear equalizers are robust and can easily be implemented. Channel equalization is however an inherently non-linear problem [8] and it is desired to incorporate some non-linearity in the equalizer structure.

Gibson et al. [8] introduced a non-linear equalizer structure based on the multi-layer perceptron and demonstrated its superior performance over the linear transversal equalizer. The multi-layer perceptron has very general capabilities of non-linear decision making. The training of multi-layer perceptron equalizers are usually based on a gradient algorithm known as the back propagation algorithm [17] and learning times are typically very long because of the highly non-linear-in-the-parameters structure. Furthermore convergence to which local minimum of the mean square error surface depends upon the initial choice of the equalizer parameters and a theoretical analysis of the training algorithm is very difficult.

The present study investigates an alternative non-linear equalizer structure based on radial basis function (RBF) networks [2]. RBF networks possess non-linear decision making ability and yet have a linear-in-the-parameters structure. The former property is essential to realize the optimal equalization solution and the latter is beneficial in practical implementation. The well-understood least mean square (LMS) algorithm [21] or its momentum version [20] can readily be used to train an RBF equalizer and the training is guaranteed to converge to a single global minimum of the mean square error surface. Computer simulation is employed to evaluate the performance of adaptive RBF equalizers. Further justifications for introducing non-linear decision making ability into the adaptive equalizer structure are provided by examining the performance of the Wiener filter, which can be considered as the performance bound for any linear equalizer structure.

The differences between the present optimal equalization solution and the maximum likelihood sequence estimation (MLSE) [7] are briefly discussed. The decision feedback equalizer (DFE) [1, 16] is another equalization architecture that makes decisions on a symbol-by-symbol base. It is shown that the non-linear filter approach to the decision feedback equalization offers advantages over the conventional DFE.

2. Channel equalization

The digital communications system considered is depicted in Fig. 1, where a random binary sequence $s(t)$ is transmitted through a dispersive channel and then corrupted by additive noise. The transmitted symbol $s(t)$ is assumed to be an independent sequence taking values of either 1 or -1 with an equal probability. The channel is modelled as a finite impulse response filter whose transfer function is defined by

$$H_n(z) = \sum_{i=0}^{n} a_i z^{-i}. \tag{1}$$

The additive noise $e(t)$ is assumed to be a Gaussian white sequence. These assumptions can however be relaxed and the approach discussed in the present study can directly be applied to the case of non-linear channel model and additive non-Gaussian correlated noise [4]. The task of the equalizer is to reconstruct input signals using the information contained in the channel observations $x(t), \ldots, x(t-m+1)$, where the integer m is known as the order of the equalizer. Often a delay d is introduced to the equalizer so that at the sample instant t the equalizer estimates the input symbol $s(t-d)$.

2.1. Optimal equalization solution

The information constraint on the general equalizer architecture depicted in Fig. 1 is characterized by the equalizer order m and delay d. Decisons are made symbol-by-symbol and the decision at each sample instant is based on the m most recent channel observations. For a given channel transfer function $H_n(z)$ and a noise distribution, we now examine what is the best possible

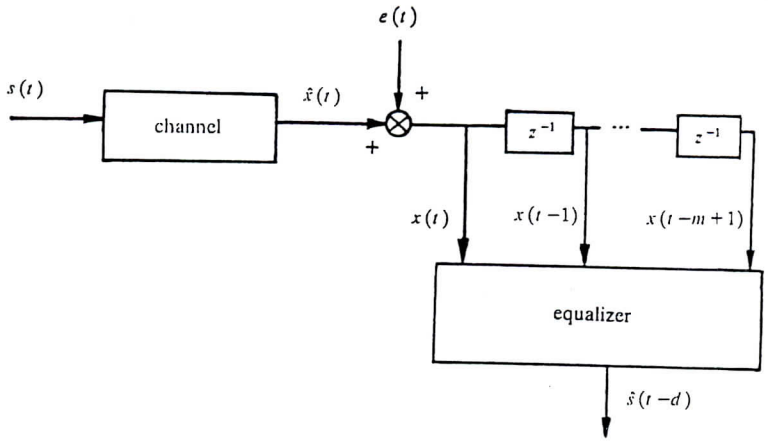

Fig. 1. Schematic of data transmission system.

performance, in terms of bit error rate, that an equalizer of order m and delay d can achieve. This question can be answered using the two-state Bayes decision rule [19] to yield the following minimum bit-error-rate equalizer:

$$\hat{s}(t-d) = \operatorname{sgn}(f_{de}(\boldsymbol{x}(t))) = \operatorname{sgn}(f_1(\boldsymbol{x}(t)) - \rho f_{-1}(\boldsymbol{x}(t))), \quad (2)$$

where

$$\operatorname{sgn}(y) = \begin{cases} 1, & y \geq 0, \\ -1, & y < 0 \end{cases} \quad (3)$$

is a slicer and the scalar ρ is defined as

$$\rho = \frac{l_{-1} p_{-1}}{l_1 p_1}, \quad (4)$$

where p_1 and p_{-1} are the a priori probabilities of $s(t-d)=1$ and $s(t-d)=-1$, respectively, l_1 is the loss associated with the decision $\hat{s}(t-d)=-1$ when actually $s(t-d)=1$ and, similarly, l_{-1} is the loss associated with $\hat{s}(t-d)=1$ when $s(t-d)=-1$. $\boldsymbol{x}(t)$ is the channel observation vector given by

$$\boldsymbol{x}(t) = [x(t), \ldots, x(t-m+1)]^{\mathrm{T}}, \quad (5)$$

$f_1(\boldsymbol{x}(t))$ and $f_{-1}(\boldsymbol{x}(t))$ are the conditional density functions of $\boldsymbol{x}(t)$ given $s(t-d)=1$ and $s(t-d)=-1$, respectively. Under the conditions introduced previously for the transmitted symbol $s(t)$, $p_1 = p_{-1} = 0.5$. If we further assign l_1 and l_{-1} with a same value then $\rho = 1$ and the optimal equalizer (2) becomes identical to that given by Gibson et al. [8].

The function $f_{de}(\,)$ is known as the optimal decision or classification function and the set of points $\boldsymbol{x}$ that satisfy

$$f_{de}(\boldsymbol{x}) = 0 \quad (6)$$

is often referred to as the decision boundary, which partitions the m-dimensional Euclidean space $\mathbb{R}^m$ into two disjoint sets D_1 and D_{-1}. D_1 plus the decision boundary, denoted as $\bar{D}_1$, will be called the decision region. The optimal equalization solution can alternatively be stated as

$$\hat{s}(t-d) = \begin{cases} 1, & \text{if } \boldsymbol{x}(t) \in \bar{D}_1, \\ -1, & \text{if } \boldsymbol{x}(t) \in D_{-1}. \end{cases} \quad (7)$$

It is easy to show that the channel noise-free output vector $\hat{\boldsymbol{x}}(t) = [\hat{x}(t), \ldots, \hat{x}(t-m+1)]^{\mathrm{T}}$ consists of finite state which can be partitioned into two classes as follows:

$$P_{m,d}(1) = \{\hat{\boldsymbol{x}}(t) \in \mathbb{R}^m \mid s(t-d) = 1\}, \quad P_{m,d}(-1) = \{\hat{\boldsymbol{x}}(t) \in \mathbb{R}^m \mid s(t-d) = -1\}. \quad (8)$$

That is $P_{m,d}(1)$ and $P_{m,d}(-1)$ represent the two sets of possible values for $\hat{\boldsymbol{x}}(t)$ which can be produced from sequences of channel inputs $s(t), \ldots, s(t-m+1-n)$ containing $s(t-d)=1$ and $s(t-d)=-1$, respectively. The task of the equalizer is to decide whether an observation $\boldsymbol{x}(t)$ represents the

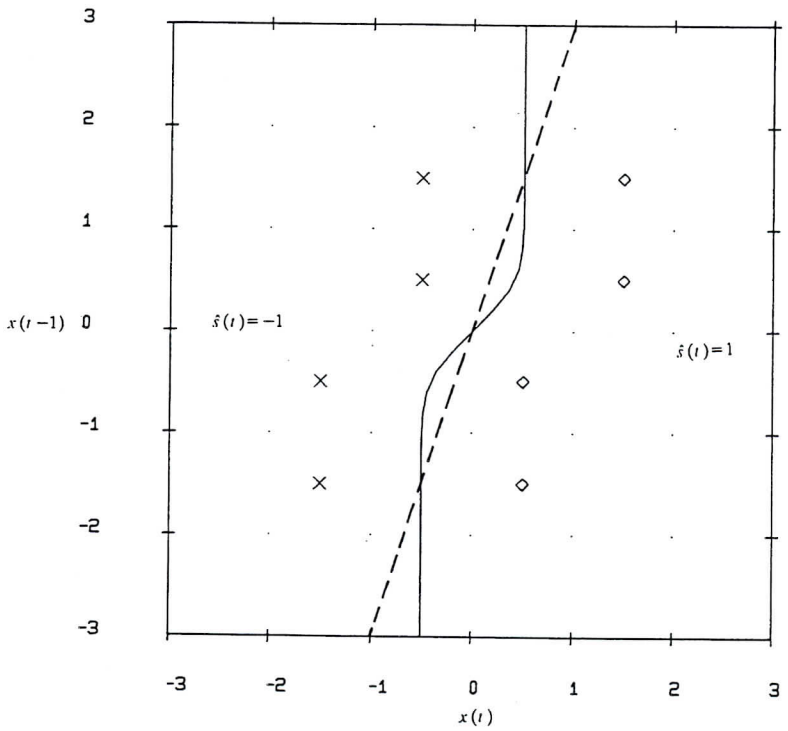

Fig. 2. Channel output points and decision boundaries. Channel $1.0+0.5z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=0$; solid line: optimal boundary; dashed line: linear boundary.

noise corruption of an element of $P_{m,d}(1)$ or $P_{m,d}(-1)$ and thus to determine the input sample $s(t-d)$. Some illustrations are given next.

2.2. Some illustrations

The case of equalizer order $m=2$ is used for easy graphic display. Let the channel response be

$$H_1(z)=1.0+0.5z^{-1} \tag{9}$$

and assume that the equalizer has a delay $d=0$. The elements of the sets $P_{2,0}(1)$ and $P_{2,0}(-1)$ are plotted in Fig. 2 using the symbols 'diamond' $\diamond$ and 'cross" $\times$, respectively. This channel is minimum phase and, therefore, $P_{2,0}(1)$ and $P_{2,0}(-1)$ are linearly separable. The four 'diamonds' and four 'crosses' represent all the possible noise-free output states corresponding to $s(t)=1$ and $s(t)=-1$, respectively. Each of these states is associated with a same probability of 0.125. When $\hat{x}(t)$ takes a particular value $\hat{x}$, due to the additive Gaussian white noise $e(t)$, the observed output vector $\boldsymbol{x}(t)$ is a stochastic process having a Gaussian probability distribution centred at $\hat{x}$ with a variance equal to that of the noise. Thus the optimal decision function $f_{de}(\ )$ is completely specified by $P_{2,0}(1)$, $P_{2,0}(-1)$ and the noise variance. The optimal boundary for the case of noise variance 0.2 is also plotted in Fig. 2. Notice that the optimal boundary is non-linear and, because the decision boundary generated by a linear equalizer is a straight line as shown in Fig. 2, any linear equalizer will produce a bit error rate considerably larger than that of the optimal equalizer in a noisy condition.

The channel transfer function for the second example is

$$H_1(z)=0.5+1.0z^{-1}. \tag{10}$$

The elements of $P_{2,0}(1)$ and $P_{2,0}(-1)$ are illustrated in Fig. 3 where the shaded region is the corresponding optimal decision region $\bar{D}_1$ for noise variance 0.2. Notice that $P_{2,0}(1)$ and $P_{2,0}(-1)$ are not linearly

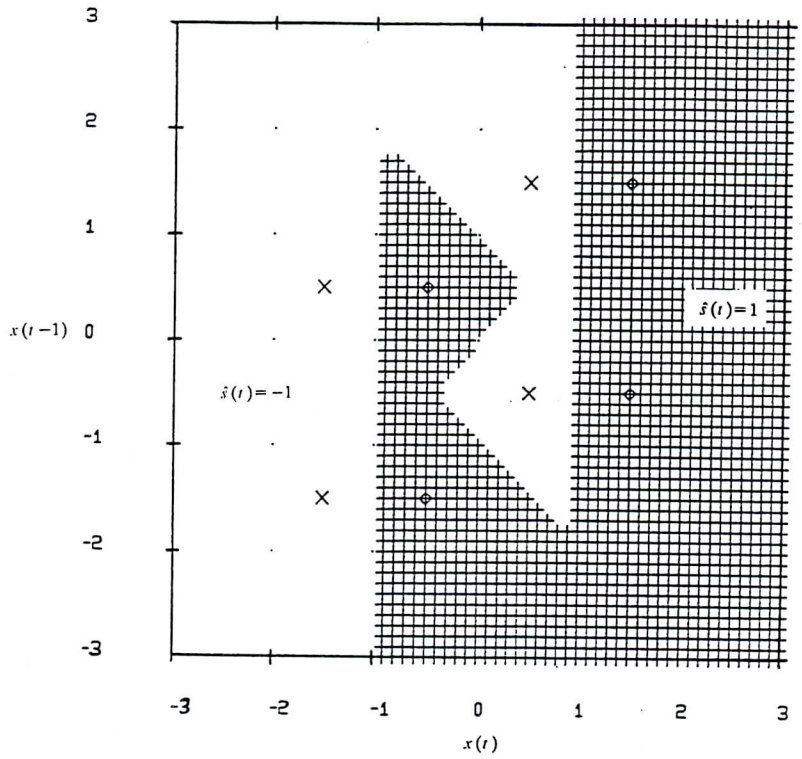

Fig. 3. Channel output points and optimal decision region. Channel $0.5+1.0z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=0$.

separable because the channel is non-minimum phase. Even in a noise-free situation, a linear equalizer of zero-delay is incapable of reconstructing input signals. The case of non-zero delay is also investigated. $P_{2,1}(1)$ and $P_{2,1}(-1)$ are shown in Fig. 4. Although $P_{2,1}(1)$ and $P_{2,1}(-1)$ are linearly separable, the optimal classification boundary is non-linear and a linear equalizer is incapable of realizing such a boundary as can be clearly seen from Fig. 4.

In general the optimal boundary is a hypersurface in the m-dimensional space and can be highly non-linear so that any linear equalizer structure is inherently suboptimal. This motivates the investigation of non-linear structures capable of realizing highly non-linear boundaries.

2.3. Extension to non-linear channel models

A variety of communications systems such as the HF communications channel can be modelled by (1) with additive Gaussian white noise [15] and most of the literature concerning adaptive channel equalization uses this linear channel representation. In certain applications however it may be more appropriate to employ the following finite non-linear channel model:

$$x(t)=g(s(t),\ldots,s(t-n);\boldsymbol{a})+e(t), \qquad (11)$$

where $g(\,)$ is some non-linear function, $\boldsymbol{a}$ is a channel parameter vector and additive noise $e(t)$ can be non-Gaussian and correlated.

For this kind of non-linear channel, the optimal symbol-by-symbol equalizer solution is still defined by (2). Some illustrations of optimal equalization boundary in the presence of channel non-linearities are given in [4]. It is straightforward to show that channel equalization is a classification problem and the optimal decision boundary is a hypersurface in the equalizer input space just as in the case of linear channel model. Therefore the crucial point is that the equalizer should possess

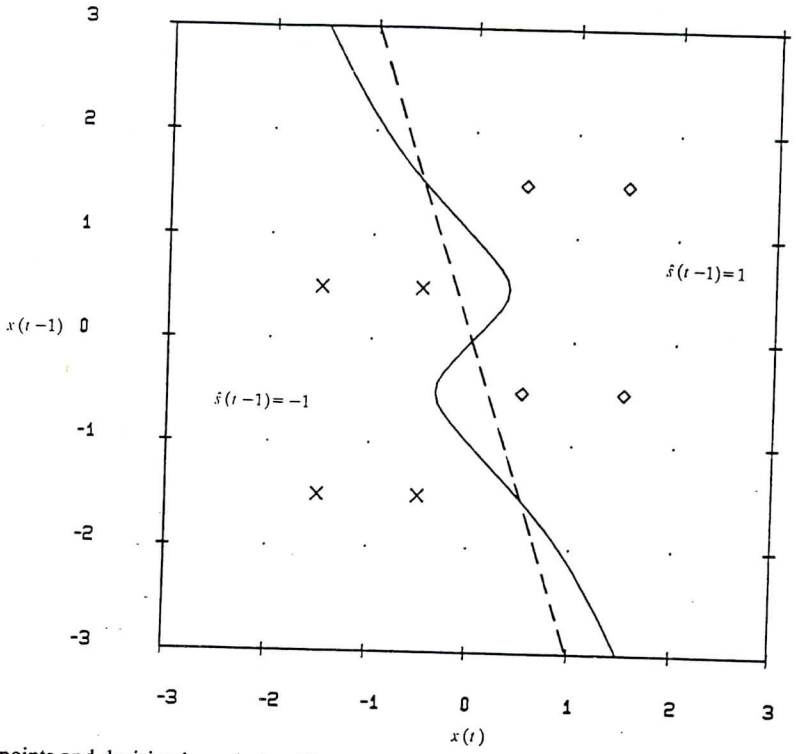

Fig. 4. Channel output points and decision boundaries. Channel $0.5+1.0z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=1$; solid line: optimal boundary; dashed line: linear boundary.

non-linear decison making ability in order to realize or to approximate the optimal performance regardless of whether a channel is non-linear or not. Previous research [4, 8] has investigated the multi-layer perceptron structure. Present study discusses an alternative structure called the RBF network. Although the linear channel model is used in the present discussion, this does not lead to any loss of generality. The application of the adaptive RBF equalizer in the presence of channel non-linearities is straightforward.

3. An adaptive equalizer based on a radial-basis-function expansion

An RBF expansion with m inputs and a scalar output implements a mapping $f_r: \mathbb{R}^m \to \mathbb{R}$ according to

$$f_r(\boldsymbol{x}) = \lambda_0 + \sum_{i=1}^{n_r} \lambda_i \phi(\|\boldsymbol{x} - \boldsymbol{c}_i\|), \tag{12}$$

here $\boldsymbol{x} \in \mathbb{R}^m$, $\phi(\ )$ is a function from $\mathbb{R}^+$ to $\mathbb{R}$, $\|\ \|$ denotes the Euclidean norm, λ_i, $0 \leq i \leq n_r$ are the tap weights, $\boldsymbol{c}_i \in \mathbb{R}^m$, $1 \leq i \leq n_r$, are the RBF centres and n_r is the number of centres. The functional form $\phi(\ )$ is assumed to have been given and the centres $\boldsymbol{c}_i$ are some fixed points in the m-dimensional space. The centres $\boldsymbol{c}_i$ must appropriately sample the input domain and they are usually chosen either to be a subset of the data or distributed uniformly in the input domain. Typical choices for $\phi(\ )$ are the thin-plate-spline function

$$\phi(y) = y^2 \log(y), \tag{13}$$

the multi-quadric function

$$\phi(y) = \sqrt{y^2 + \sigma^2}, \tag{14}$$

the inverse multi-quadric function

$$\phi(y) = \frac{1}{\sqrt{y^2 + \sigma^2}} \tag{15}$$

and the Gaussian function

$$\phi(y) = \exp(-y^2/\sigma^2), \tag{16}$$

where σ is a real constant. The RBF expansion is a traditional technique for interpolating in multi-dimensional space [12–14]. The generalization to the form of (12) was introduced by Broomhead and Lowe [2]. Because our aim is to use the RBF expansion $f_r(\,)$ to approximate the optimal decision function $f_{de}(\,)$, the approximation capabilities of RBF are first summarized.

3.1. Capabilities of radial basis functions

Current interests on RBF within the engineering community are largely motivated by the realization that there are strong connections between RBF and neural networks. In fact (12) can be implemented in the two-layered network structure depicted in Fig. 5. Given fixed centres the first layer performs a fixed non-linear transformation which maps the input space onto a new space, and the output layer implements a linear combiner on this new space. If however the centres are not predetermined and are regarded as adjustable parameters, the structure represented in Fig. 5 then becomes a two-layered feedforward neural network, that is a two-layer perceptron [6, 10]. Each term $\phi(\|\boldsymbol{x}-\boldsymbol{c}_i\|)$ forms a basic computing unit of the network called a hidden neuron. A detailed relationship between RBF and two-layered neural networks was given by Broomhead and Lowe [2]. Although the approximation capabilities of RBF have been well established by researchers working in multi-dimensional interpolation techniques (e.g. [14]), the strong connections between RBF and neural networks offer a heuristic explanation.

Assume for the time being that the centres are adjustable parameters. Then Fig. 5 represents a two-layer perceptron. Consider the basic component of the network, a hidden neuron, shown in Fig. 6 in the two-dimensional case. The surface generated by the Gaussian neuron (16) looks like a 'bump' as illustrated in Fig. 6. If the hidden layer consists of enough neurons, sufficient 'bumps' can be generated in appropriate positions by adjusting the parameters of the first layer. A variety of continuous surfaces or functions from $\mathbb{R}^m \to \mathbb{R}$ can thus be well approximated by adding up a series of such 'bumps'. Lapedes and Farber

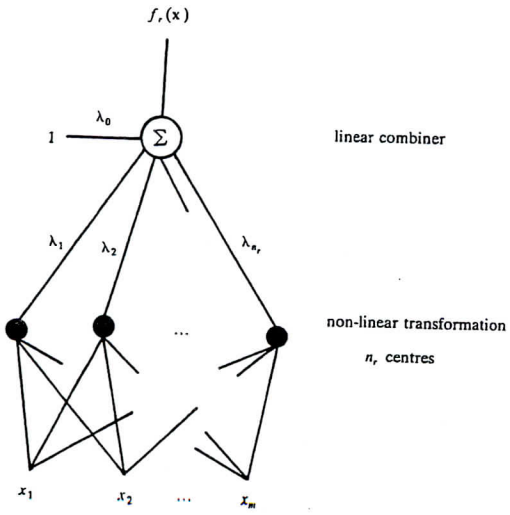

Fig. 5. Schematic of RBF expansion.

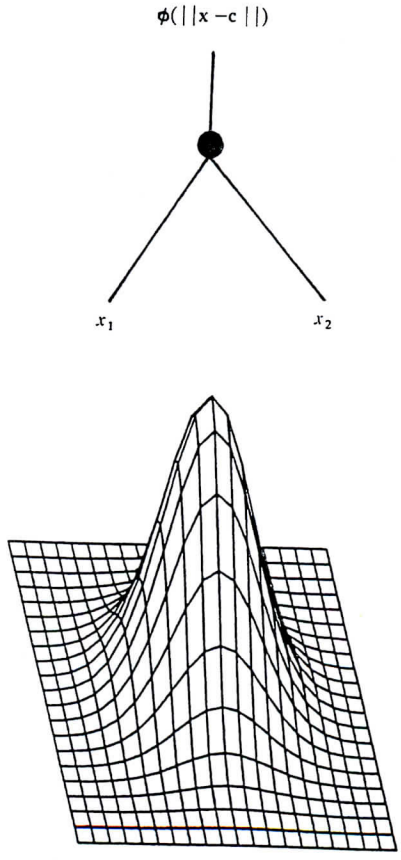

Fig. 6. Surface generated by a Gaussian neuron.

[9] used these ideas to explain how feedforward neural networks are able to approximate a large class of functions. In an RBF network, the centres are all fixed. If however a sufficient number of centres are used and they are suitably distributed in the input domain, it is possible that the much simpler RBF model will also provide a reasonable representation for various functions. The surface generated by the inverse multi-quadric non-linearity (15) also looks like a 'bump'. A similar conclusion can therefore be reached for this choice of non-linearity.

When the RBF expansion (12) is used to approximate some unknown and complicated non-linear mapping $f:\mathbb{R}^m \to \mathbb{R}$, its approximation capabilities are highly related to its localization properties [14]. The weights λ_i are generally computed based on observed data $f(\boldsymbol{x})$. Let $X \subset \mathbb{R}^m$ be the domain of the approximation and X_0 be any part of X. $f_r(\boldsymbol{x})$ is said to have good localization properties if the contribution to $f_r(\boldsymbol{x})$ from $\{f(\tilde{\boldsymbol{x}}) | \tilde{\boldsymbol{x}} \in X_0\}$ is small for $\boldsymbol{x} \in X$ that are far away from X_0. For the choices of non-linearities (15) and (16), $\phi(y) \to 0$ as $y \to \infty$ and the RBF expansion (12) has good localization properties. It is apparent that, for the non-linearities (13) and (14), $\phi(y) \to \infty$ as $y \to \infty$. The surprising results of Powell [14] indicate that these kinds of RBF approximations can have good localization properties and, in fact, the success of approximation is easier to achieve if $\phi(y) \to \infty$ as $y \to \infty$. These results suggest that the choice of the non-linearly $\phi(\,)$ is not crucial for performance.

Cybenko [6] has rigorously proved in the case of sigmoid non-linearity that the two-layer perceptron can uniformly approximate any continuous function. The proofs can easily be extended to other choices of non-linearities. By choosing centres appropriately, it is possible that the simpler RBF expansion (12) can offer similar approximation capabilities. It is appropriate to end this discussion with a brief comparison between the multi-layer perceptron and the RBF expansion. Neural networks have very general approximation apabilities and have proved to be highly successful in dealing with complex data. The structure of neural networks however is highly non-linear-in-the-parameters. Training times are usually very long and a theoretical analysis of the training algorithm is difficult when models are highly non-linear-in-the-parameters. By carefully pre-fixing centres, RBF networks can closely match the performance of two-layer perceptrons and yet have a linear-in-the-parameters formulation. The well-known analysis results for the linear combiner [21] can readily be applied to the RBF expansion and learning based on the mean-square-error criterion guarantees to converge to the single minimum.

In the present application, the non-linearity $\phi(\,)$ is chosen to be the Gaussian function (16). The parameter σ is selected to be the maximum distance between the chosen centres and this merely ensures that the Gaussian functions are not too peaked or too flat. The selection of RBF centres will be discussed later.

3.2. Training algorithms

The training of the RBF equalizer can be carried out either using the LMS algorithm [21] or its momentum version [20]. By defining $\phi_0(t) = 1$ and $\phi_i(t) = \phi(\|\boldsymbol{x}(t) - \boldsymbol{c}_i\|)$ for $1 \leq i \leq n_r$, the RBF expansion (12) can be rewritten as

$$f_r(\boldsymbol{x}(t)) = \sum_{i=0}^{n_r} \lambda_i \phi_i(t). \tag{17}$$

The LMS algorithm is then given by

$$\lambda_i(t+1) = \lambda_i(t) + \alpha \varepsilon(t) \phi_i(t), \quad 0 \leq i \leq n_r, \tag{18}$$

and its momentum version is defined as

$$\begin{aligned} \Delta_i(t+1) &= \beta \Delta_i(t) + \alpha \varepsilon(t) \phi_i(t), \\ \lambda_i(t+1) &= \lambda_i(t) + \Delta_i(t+1), \end{aligned} \quad 0 \leq i \leq n_r, \tag{19}$$

where

$$\varepsilon(t) = s(t-d) - \sum_{i=0}^{n_r} \lambda_i(t) \phi_i(t), \tag{20}$$

and α and β are the adaptive gain and momentum constant, respectively. During data transmission,

$s(t-d)$ is substituted by its estimate $\hat{s}(t-d)$ and the algorithm (18) or (19) can continuously be employed to track a time-varying environment. $\varepsilon(t)\phi_i(t)$, which is the negative gradient of $\varepsilon^2(t)/2$ with respect to $\lambda_i(t)$, is often referred to as a stochastic gradient. Similarly $\Delta_i(t)$ is known as a smoothed stochastic gradient. Using a smoothed stochastic gradient usually improves the performance at the cost of more computation in each recursion.

The calculation of the RBF mapping f_r requires n_r inner products, n_r divisions and n_r multiplications which is more than that required by a linear combiner of n_r tap weights. The computational complexity of the algorithm (18) or (19) is well-known to be of order n_r+1 [5]. The properties of the LMS algorithm and its momentum version are well understood and practical guidelines for choosing α and β can be found in many existing literature.

3.3. Simulation study

The capability of the RBF equalizer to form non-linear classification boundaries is illustrated using the channel given in (10). In all the cases, the algorithm (19) was used to train RBF equalizers and the adaptive gain and momentum constant were set to $\alpha = 0.05$ and $\beta = 0.5$, respectively. The length of the training sequence was 350 and the additive Gaussian white noise had a variance 0.2. For the case of $m=2$ and $d=0$, 40 centres were generated by randomly selecting points from the region $[-2.5, 2.5]\times[-2.5, 2.5]$. The trained RBF equalizer produced the decision region shown as the shaded region in Fig. 7. For the case of $m=2$ and $d=1$, we first used the 13 data points, $x(t)$, $t=1,\ldots,13$, as the centres. The decision region produced by this RBF equalizer after training is given as the shaded region in Fig. 8. Next, 50 centres were randomly selected from the region

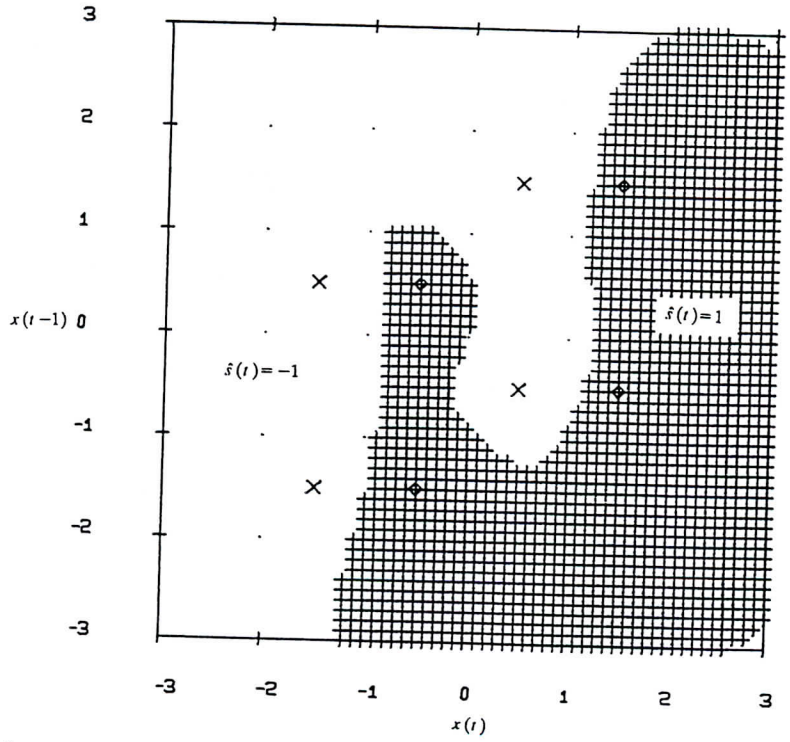

Fig. 7. Decision region formed by an RBF equalizer. Channel $0.5+1.0z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=0$, 40 randomly generated centres.

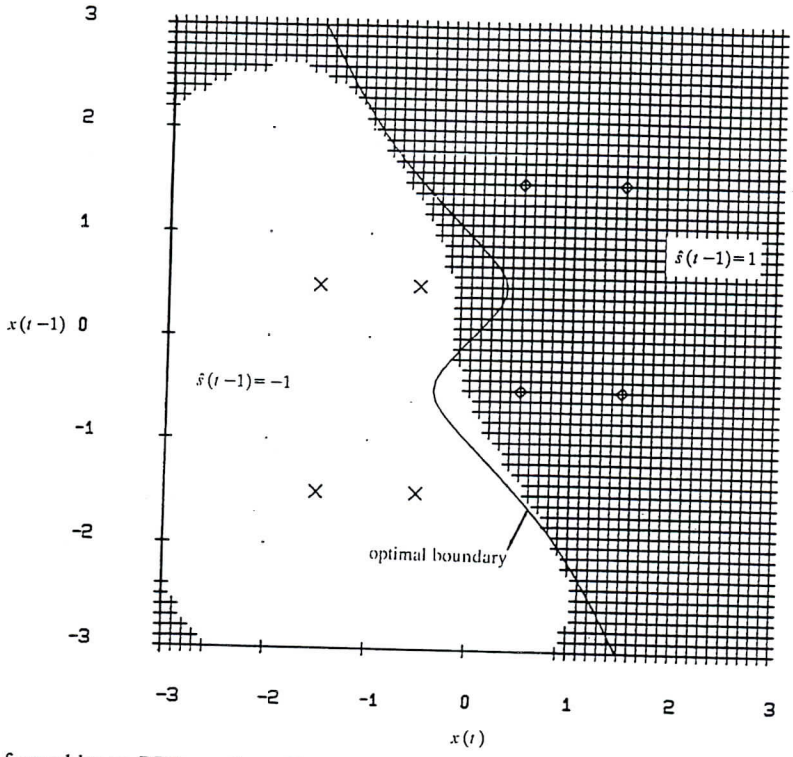

Fig. 8. Decision region formed by an RBF equalizer. Channel $0.5+1.0z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=1$, 13 centres from first 13 data points.

$[-2.75, 2.75]\times[-2.75, 2.75]$. After training, this RBF equalizer produced the decision region shown in Fig. 9.

Another example was used to compare the bit error rates achieved by the optimal and RBF equalizers for different signal-to-noise ratios. The channel was defined by

$$H_2(z)=0.3482+0.8704z^{-1}+0.3482z^{-2}. \quad (21)$$

The equalizer had the structure of $m=4$ and $d=1$. For each signal-to-noise ratio tested, the first 65 data points, $x(t)$, $t=1,\ldots,65$, were used as the centres. The results obtained using algorithm (19) are illustrated in Fig. 10, where the bit error rate was computed over 500 000 points of different realizations of stochastic processes $s(t)$ and $e(t)$. As mentioned before, channel obervation vectors $\boldsymbol{x}(t)$ are the noise corruption of the points of $P_{m,d}(1)$ and $P_{m,d}(-1)$. Choosing a few consecutive observation vectors as RBF centres is a simple and practical method. It however may not be an ideal solution to the selection of centres because not all the outcomes of $P_{m,d}(1)$ and $P_{m,d}(-1)$ will always be represented and some of the selected centres may come from the same element of $P_{m,d}(1)$ or $P_{m,d}(-1)$ corrupted by different realizations of noise. This may explain the performance gap between the RBF equalizer and the optimal one as can be seen from Fig. 10.

For the above given channel and equalizer structure, if the points of $P_{4,1}(1)$ and $P_{4,1}(-1)$, a total of 64, are used as centres, it is obvious that the optimal performance can be obtained. To test the robustness of this optimal performance, random noise of uniform distribution between -0.1 and 0.1 was used to corrupt the points of $P_{4,1}(1)$ and $P_{4,1}(-1)$ and the resulting 64 points were employed as centres. The performance of this RBF equalizer after training is also plotted in Fig. 10, where it is seen that the bit-error-rate curve of this equalizer is closely matched to the optimal one.

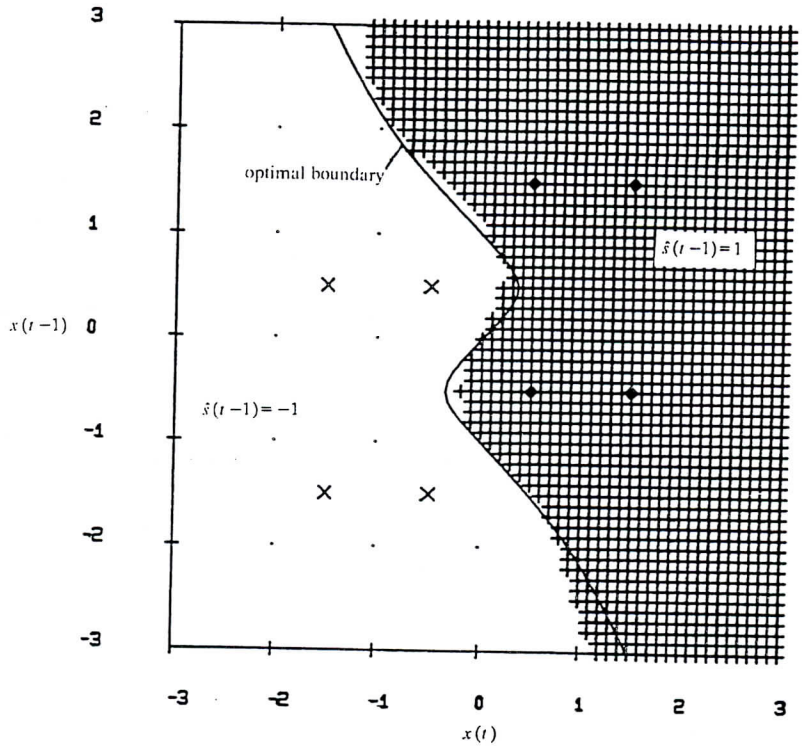

Fig. 9. Decision region formed by an RBF equalizer. Channel $0.5+1.0z^{-1}$ and Gaussian white noise variance 0.2, equalizer order $m=2$ and delay $d=1$, 50 randomly generated centres.

3.4. Selection of radial-basis-function centres

The simulation results of Fig. 10 suggest two points. Firstly the selection of centres has an important influence on the performance of RBF equalizers and an ideal choice would be the points of $P_{m,d}(1)$ and $P_{m,d}(-1)$, which in general are not available. Secondly slight changes in the positions of centres do not alter performance significantly. In practice the RBF centres are chosen either to be a subset of the data or distributed randomly in the input space.

When the centres are determined by choosing points randomly from the input domain, enough centres must be used in order to generate a sufficiently complex classification mapping. On the other hand, if too many centres are employed, the equalizer will become unnecessarily complex and the convergence of the training process can be slow. Presently the number of centres required is typically determined by experience or through experimentation. This is clearly unsatisfactory, and there is thus a considerable scope for further research on the mechanism of selecting RBF centres.

In channel equalization applications, it is often convenient to choose RBF centres directly from observation vectors. The simplest approach is to select some consecutive observation vectors as centres. Currently an adequate number of centres required is usually determined through experimentation. In certain situations, however, an off-line approach to the selection of centres may be feasible by employing the orthogonal forward regression (OFR) routine [3]. The OFR routine is a very efficient algorithm for selecting a significant subset model based on the orthogonal least squares method. This algorithm has been successfully used to select an adequate set of RBF centres from the data set in the modelling of non-linear systems. The application to the channel equalization is relatively simple. After a few hundreds of observations

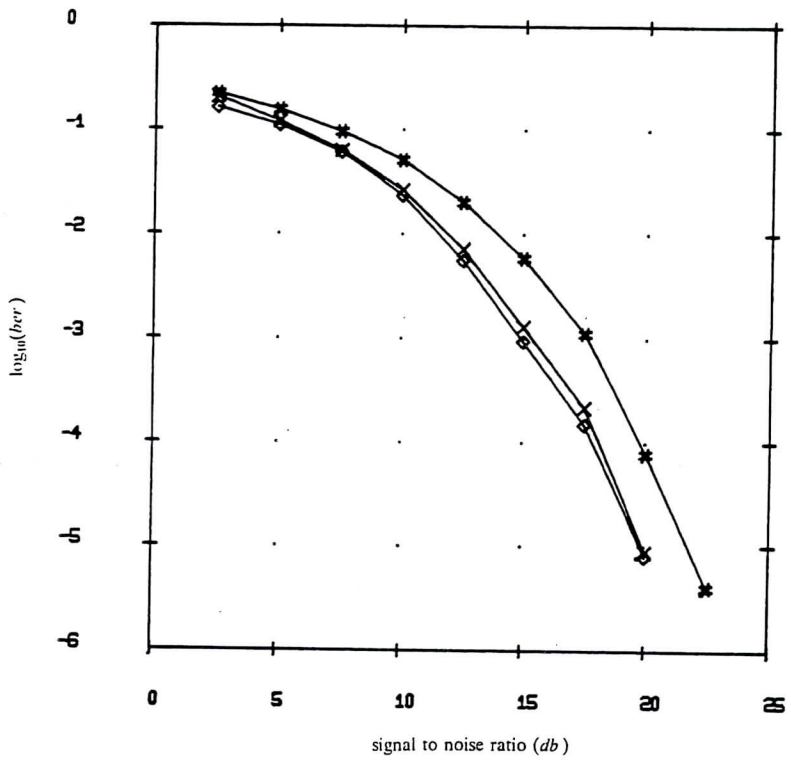

Fig. 10. Performance comparison. Channel $0.3482+0.8704z^{-1}+0.3482z^{-2}$, equalizer order $m=4$ and delay $d=1$; —◇—◇— optimal equalizer, —#—#— RBF equalizer with 65 centres from first 65 data points, —×—×— RBF equalizer with 64 centres from noise corrupted $P_{4,1}(1)$ and $P_{4,1}(-1)$.

have been collected, the OFR algorithm is used to choose a subset of centres from the data set. When an adequate set of centres has been selected based on a version of the least-squares criterion, the procedure is automatically terminated. Thus the number of centres required is automatically determined. These selected centres can then be employed in real-time adaptation. The detailed description of the OFR is given in [3].

4. Performance bound of the linear equalizer

The RBF equalizer used in the previous simulation study of Fig. 10 had a relatively low equalizer order ($m=4$) and a quite large number (66) of tap weights. A linear equalizer of order 4 only has 4 tap weights. It therefore may be argued that comparing non-linear and linear structures based on a same equalizer order m is unfair and a more reasonable comparison would be based on similar numbers of tap weights. The real question here is whether a higher-order linear equalizer can offer a similar performance to that of a lower-order non-linear equalizer, and this is investigated next.

Because the weight vector of a linear equalizer after convergence should approximate the Wiener optimal filter of same order, the bit error rate achieved by the linear equalizer can therefore be predicted from that of the Wiener filter. Denote the weight vector of the Wiener filter of order m as

$$\hat{w}=[\hat{w}_1,\ldots,\hat{w}_m]^{\mathrm{T}}. \tag{22}$$

Under the conditions given in Section 2, $\hat{w}$ can easily be calculated [21]. The bit error rate achieved by the Wiener filter is defined as

$$\text{Prob}\{\hat{w}^{\mathrm{T}}x(t)<0\,|\,\hat{x}(t)\in P_{m,d}(1)\}\quad\text{or}$$

$$\text{Prob}\{\hat{w}^{\mathrm{T}}x(t)>0\,|\,\hat{x}(t)\in P_{m,d}(-1)\}. \tag{23}$$

Under the assumption that the channel noise $e(t)$ is an additive Gaussian white sequence with variance σ_ε^2, $\hat{w}^T x(t)$ is also Gaussian distributed with mean $\hat{w}^T \hat{x}(t)$ and variance $\sigma_\varepsilon^2 \hat{w}^T \hat{w}$ and, therefore, it is not difficult to compute the probability (23).

For the channel (21) and equalizer delay $d = 1$, Fig. 11 shows the relationships between the theoretical bit error rate (23) and the Wiener filter order in a variety of noise conditions. It is obvious from Fig. 11 that the performance curve quickly becomes flat. Therefore the performance achievable by simply increasing the order of the linear equalizer cannot match that of a lower-order non-linear equalizer. Furthermore little advantage can be gained in a noise environment by employing a linear equalizer which has an order greater than 4 for this example.

The phenomenon shown in Fig. 11 is known as the noise enhancement. In a noisy environment, as the order of the equalizer increases so does the total power of the noise on the equalizer input and this tends to diminish any advantage gained by increasing the equalizer order. The problem of noise enhancement provides further justification for considering non-linear equalizers of low order, such as the present RBF equalizer, in high noise conditions.

5. Comparison with maximum likelihood sequence estimation

The performance of the optimal equalizer derived in Section 2.1 is the best possible performance that can be achieved for given m and d and under the constraint of symbol-by-symbol decisons. It has long been recognized that even better performance can be obtained if symbol decisons are based on the entire received sequence. Forney [7] derived an MLSE receiver for known

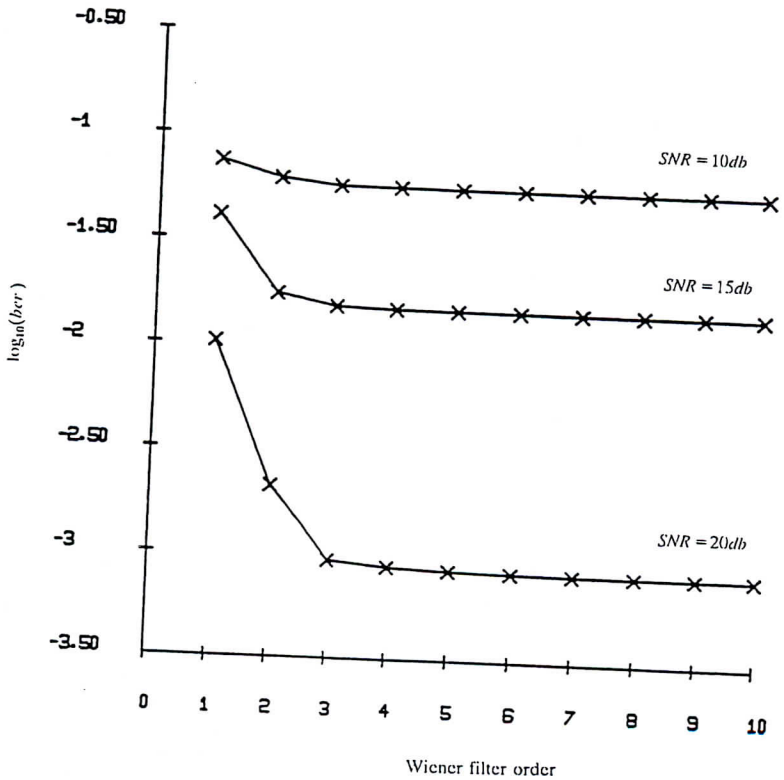

Fig. 11. Bit error rate versus Wiener filter order. Channel $0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$ and equalizer delay $d = 1$.

channel response. The main processing unit of the structure is an algorithm called Viterbi algorithm and the structure is optimal in the sense that it is the maximum likelihood estimator for the entire received sequence. For a known channel, the MLSE structure will generally produce a better bit error rate compared with the present optimal equalizer (2). However these two structures have very different information requirements and therefore have quite different practical implications.

When the channel response is unknown, a channel estimator can be employed to provide a channel estimate [11] and this gives rise to an adaptive MLSE architecture depicted in Fig. 12 (the usual whitened matched filter is not shown here). Deferring decisions are essential in such an architecture. In practice a final decision is made after some fixed delay and this delay must be chosen large enough so that the degradation due to premature decisions is negligible. Such a long delay is not welcome in many practical applications. Because the operation is based on a long segment of the received sequence, the computation and memory requirements of the adaptive MLSE are far more than those of the simple architecture based on symbol-by-symbol decisions. The adaptive MLSE structure is sensitive to channel estimate. If the channel estimator does not provide a reasonably good estimate of the true channel model, significant performance degradation is expected. The Viterbi algorithm is also sensitive to the assumption of additive Gaussian white noise. If the noise is correlated, performance loss will occur.

It is seen that, although the MLSE architecture generally produces a better performance compared with the simple architecture of Fig. 1, it is much more difficult to implement. This may explain why most of the practical applications to adaptive channel equalization are based on the architecture of symbol-by-symbol decisions. The present study has demonstrated that the optimal solution to this simple architecture can easily be realized using the non-linear finite filter approach which is a natural extension to the linear finite filter. Another advantage of the current approach is that it can be applied to the case of non-linear channel model and additive correlated noise without any modification [4].

6. Extension to decison feedback equalizer

Decision feedback equalization is another widely used technique which makes decisions on a symbol-by-symbol base. The architecture of the DFE is shown in Fig. 13. The advantage of the DFE is that intersymbol interference is eliminated without enhancement of noise by using past decisions, and a disadvantage is the error propagation caused by decision errors. The operation of the DFE at each sample instant is based on m most recent channel observations and k past decisions. The conventional DFE employs two linear finite

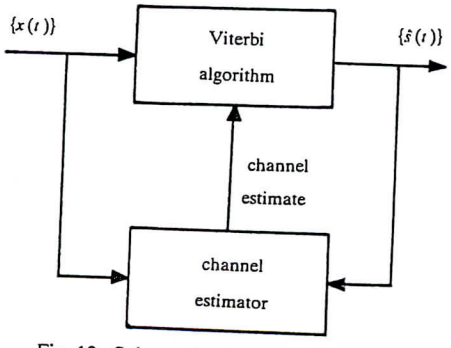

Fig. 12. Schematic of adaptive MLSE.

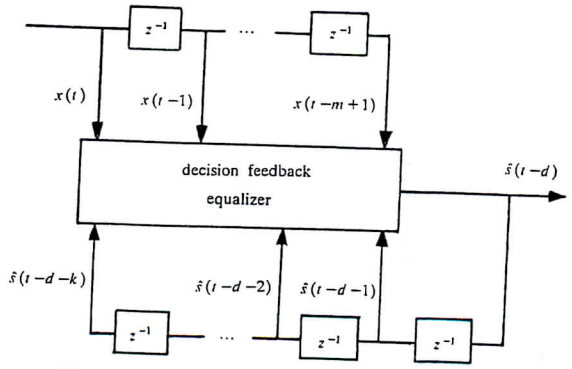

Fig. 13. Schematic of decision feedback equalizer.

filters called feedforward and feedback filters and it makes decisions according to

$$\hat{s}(t-d)=\operatorname{sgn}\left(\sum_{j=1}^{m} b_j x(t-j+1) + \sum_{j=1}^{k} c_j \hat{s}(t-d-j)\right), \qquad (24)$$

where the b_j and c_j are the coefficients of the feedforward and feedback filters, respectively, the integers m and k are the orders of the feedforward and feedback filters, respectively.

Siu et al. [18] applied the multi-layer perceptron structure to the decision feedback equalization and demonstrated its superior performance over the conventional DFE. The better performance of the non-linear filter approach to the DFE can similarly be explained by examining the optimal solution of the DFE architecture depicted in Fig. 13. It can easily be shown that the optimal solution of the DFE is a non-linear problem and therefore the non-linear filter approach offers advantages over the conventional DFE. We will use a simple example to illustrate this aspect.

The DFE has the structure of $m=2$, $k=1$ and $d=0$. The channel response is given by

$$H_2(z)=1.0+2.0z^{-1}+1.0z^{-2} \qquad (25)$$

and the additive Gaussian white noise has a variance 0.1. The input to the DFE is $[x(t)x(t-1)\hat{s}(t-1)]^{\mathrm{T}}$. The optimal decision boundary is a surface in the three-dimensional space and is difficult to visualize. Under the assumption that $\hat{s}(t-1)=s(t-1)$, the two cross-sections of the optimal boundary with the planes $s(t-1)=1$ and $s(t-1)=-1$ are plotted in Fig. 14. It is obvious from Fig. 14 that the conventional DFE cannot realize such a non-linear boundary and, therefore, it is desirable to include non-linear decision making capabilities into the DFE structure. The application of the RBF and other non-linear structures to the DFE is currently under investigation.

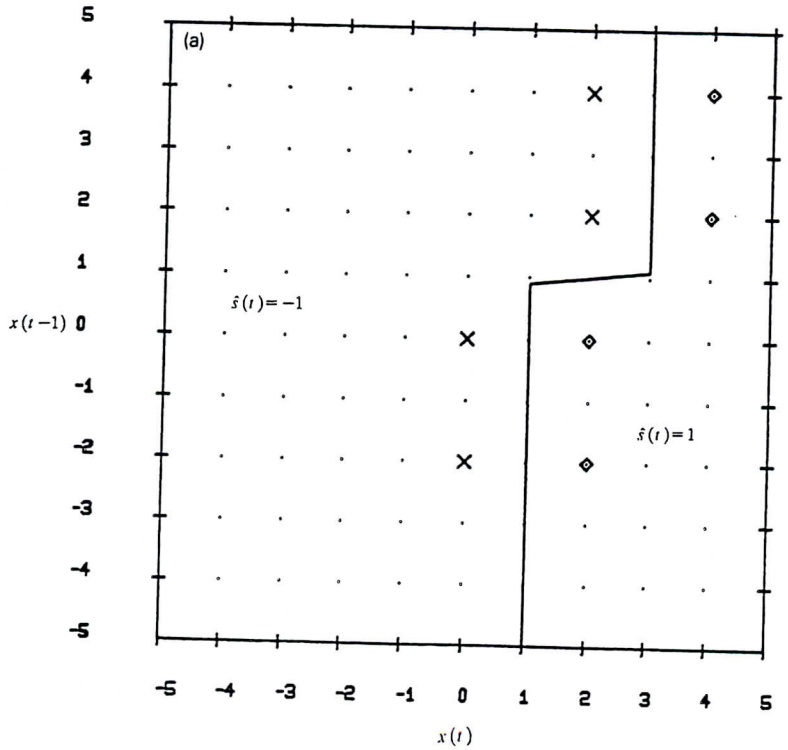

Fig. 14. Optimal decision boundary of DFE. Channel $1.0+2.0z^{-1}+1.0z^{-2}$ and Gaussian white noise variance 0.1, $m=2$, $k=1$ and $d=0$. (a) Cross-section of optimal boundary with plane $s(t-1)=1$. (b) Cross-section of optimal boundary with plane $s(t-1)=-1$.

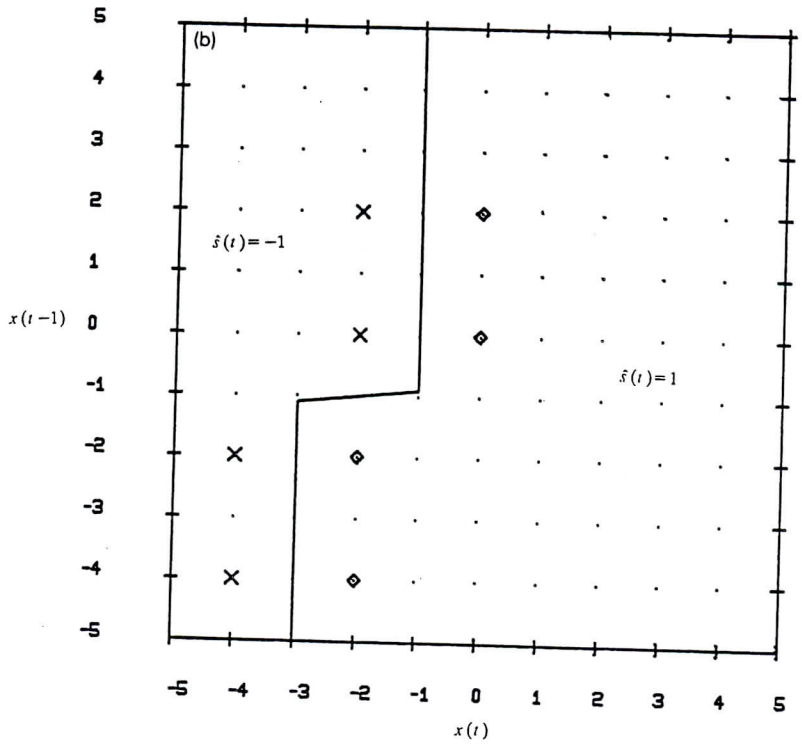

Fig. 14—continued

7. Conclusions

The adaptive equalization of bipolar signals passed through dispersive channels can be considered as a classification problem. It has been shown that an equalizer which incorporates some degree of non-linearity can achieve a bit error rate superior to that offered by linear structures. The RBF network has been cited as one architecture capable of realizing this improvement. The adaptive RBF equalizer has a further advantage of being linear-in-the-parameters compared with other non-linear structures. The differences between the MLSE and the optimal symbol-by-symbol equalization solution has been discussed and it has been demonstrated that the present approach can be extended to the DFE.

It has been shown that employing a low equalizer order is justified in poor signal-to-noise ratio conditions. The performance of RBF equalizers depends upon the selection of their centres. Simple and practical methods for choosing RBF centres have been discussed. As yet the whole question of centre selection has not been on a theoretical basis and would be a subject worthy of further investigation. The OFR algorithm offers an effective solution to this problem in certain applications.

Acknowledgments

This work was supported by the U.K. Science and Engineering Research Council (Grant Ref. GR/E/10357). The authors wish to thank the referees for their valuable comments on the manuscript.

References

[1] C.A. Belfiore and J.H. Park, "Decision feedback equalization", *Proc. IEEE*, Vol. 67, No. 8, 1979, pp. 1143-1156.

[2] D.S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks", *Complex Systems*, Vol. 2, 1988, pp. 321-355.

[3] S. Chen, S.A. Billings, C.F.N. Cowan and P.M. Grant, "Non-linear systems identification using radial basis functions", *Internat. J. Systems Sci.*, 1990, to appear.

[4] S. Chen, G.J. Gibson, C.F.N. Cowan and P.M. Grant, "Adaptive equalization of finite non-linear channels using multilayer perceptrons", *Signal Processing*, Vol. 20, No. 2, 1990, pp. 107-119.

[5] C.F.N. Cowan and P.M. Grant, *Adaptive Filters*, Prentice-Hall, Englewood Cliffs, NJ, 1985.

[6] G. Cybenko, "Approximations by superpositions of a sigmoidal function", *Math. Control Signals Systems*, Vol. 2, No. 4, 1989, pp. 303-314.

[7] G.D. Forney, "Maximum-likelihood sequence estimation of digitial sequences in the presence of intersymbol interference", *IEEE Trans. Inform. Theory*, Vol. IT-18, No. 3, 1972, pp. 363-378.

[8] G.J. Gibson, S. Siu and C.F.N. Cowan, "Application of multilayer perceptrons as adaptive channel equalisers", *Proc. IEEE Internat. Conf. Acoust. Speech Signal Process.*, Glasgow, Scotland, 23-26 May 1989, pp. 1183-1186.

[9] A. Lapedes and R. Farber, "How neural nets work", in: D.Z. Anderson, ed., *Neural Information Processing Systems*, American Institute of Physics, New York, 1988, pp. 442-456.

[10] R. P. Lippmann, "An introduction to computing with neural nets", *IEEE ASSP Mag.*, Vol. 4, 1987.

[11] F.R. Magee and J.G. Proakis, "Adaptive maximum-likelihood sequence estimation for digital signalling in the presence of intersymbol interferences", *IEEE Trans. Inform. Theory*, Vol. IT-19, No. 1, 1973, pp. 120-124.

[12] C.A. Micchelli, "Interpolation of scattered data: distance matrices and conditionally positive definite functions", *Constructive Approximation*, Vol. 2, 1986, pp. 11-22.

[13] M.J.D. Powell, "Radial basis functions for multivariable interpolation: A review", *IMA Conf. Algorithms for the Approximation of Functions and Data*, RMCS Shrivenham, 1985.

[14] M.J.D. Powell, "Radial basis function approximations to polynomials", *12th Biennial Numerical Analysis Conference*, Dundee, 1987, pp. 223-241.

[15] J.G. Proakis, *Digital Communications*, McGraw-Hill, New York, 1983.

[16] S.U.H. Qureshi, "Adaptive equalization", *Proc. IEEE*, Vol. 73, No. 9, 1985, pp. 1349-1387.

[17] D.E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning internal representations by error propagation", in: D.E. Rumelhart and J.L. McClelland, eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, Cambridge, MA, 1986, pp. 318-362.

[18] S. Siu, G.J. Gibson and C.F.N. Cowan, "Decision feedback equalization using neural network structures", *Proc. 1st IEE Internat. Conf. Artificial Neural Networks*, London, 16-18 October 1989, pp. 125-128.

[19] D.F. Specht, "Generation of polynomial discriminant functions for pattern recognition", *IEEE Trans. Electron. Comput.*, Vol. EC-16, No. 3, 1967, pp. 308-319.

[20] M.A. Tugay and Y. Tanik, "Properties of the momentum LMS algorithm", *Signal Processing*, Vol. 18, No. 2, 1989, pp. 117-127.

[21] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.