ELSEVIER

Signal Processing 43 (1995) 181–195

SIGNAL PROCESSING

A functional link artificial neural network for adaptive channel equalization

Jagdish C. Patra[1], Ranendra N. Pal*

Department of Electronics & Electrical Communication Engineering, Indian Institute of Technology, Kharagpur 721 302, India

Received 21 February 1994; revised 4 August 1994

Abstract

Application of artificial neural network (ANN) structures to the problem of channel equalization in a digital communication system has been considered in this paper. The difficulties associated with channel nonlinearities can be overcome by equalizers employing ANN. Because of nonlinear processing of signals in an ANN, it is capable of producing arbitrarily complex decision regions. For this reason, the ANN has been utilized for the channel equalization problem. A scheme based on a functional link ANN (FLANN) has been proposed for this task. The performance of the proposed network along with two other ANN structures has been compared with the conventional LMS based channel equalizer. Effect of eigenvalue ratio of the input correlation matrix on the performance of the equalizers has been studied. From the simulation results, it is observed that the performance of the proposed FLANN based equalizer outperforms the other two in terms of bit-error rate (BER) and attainable MSE level over a wide range of eigenvalue spread, signal to noise ratio and channel nonlinearities.

Zusammenfassung

In dieser Arbeit wird die Anwendung von artifiziellen neuronalen Netzwerken (ANN) auf das Problem der Kanalentzerrung in digitalen Kommunikationssystemen betrachtet. Die mit Nichtlinearitäten des Kanals verbundenen Probleme können durch die Verwendung von ANN-Entzerrern überwunden werden, Wegen der nichtlinearen Signalverarbeitung ist es möglich, beliebige komplexe Entscheidungsbereiche zu realisieren. Aus diesem Grunde wurde das ANN auf die Kanalentzerrung angewendet. Ein auf einer Funktional-ANN-Verbindung (functional link ANN, FLANN) basierendes Schema wird für diese Aufgabe vorgeschlagen. Das Verhalten des vorgeschlagenen Netzwerks zusammen mit zwei anderen ANN-Strukturen wird mit dem konventionellen LMS-Entzerrer verglichen. Die Auswirkungen der Eigenwertverhältnisse der Autokorrelationsmatrix des Eingangssignals wurden untersucht. Simulationsresultate zeigen, daß das Verhalten des vorgeschlagenen FLANN-basierten Entzerrers den anderen beiden bezüglich Bitfehlerrate (BFR) und erreichbarem MSE-Pegel in einem weiten Bereich von Eigenwert-Verhältnissen, Signal-Stör-Verhältnis und Kanal-Nichtlinearitäten überlegen ist.

Résumé

L'application de structures de réseaux de neurones artificiels (ANN) au problème de l'égalisation d'un système de communication digital est abordée dans cet article. Les difficultés associées aux non linéarités du canal peuvent être

* Corresponding author.

[1] On leave from the Department of Applied Electronics & Instrumentation Engineering, Regional Engineering College, Rourkela, Orissa 769 008, India.

SSDI 0165-1684(94)00152-9

surmontées par des égaliseurs employant des ANN. Du fait du traitement non linéaire des signaux par un ANN, un tel égaliseur est capable de produire des régions de décision arbitrairement complexes. Pour cette raison, l'ANN a été utilisé pour le problème de l'égalisation de canal. Un système basé sur un ANN à lien fonctionnel (FLANN) est proposé pour cette tâche. Les performances du réseau proposé ainsi que celles de deux autres structures d'ANN sont comparées à celles de l'égaliseur de canal LMS conventionnel. L'effet du rapport de valeurs propres de la matrice de corrélation de l'entrée sur les performances de l'égaliseur sont étudiées. D'après les résultats des simulations, il est observé que les performances de l'égaliseur basé sur un FLANN proposé sont meilleures que celles des deux autres en termes de taux d'erreur de bit (BER) et de niveau de MSE atteignable sur une plage étendue de dispersion des valeurs propres, de rapport signal sur bruit et de non linéarités du canal.

Keywords: Neural networks; Multilayer perceptron; Polynomial perceptron; Functional link neural network; Adaptive channel equalization

1. Introduction

Adaptive channel equalization has been found to be very important for effective digital data transmission over linear dispersive channels. In high speed data transmission, the amplitude and phase distortion due to variation of channel characteristics to which the data signal will be subjected is to be suitably compensated. This compensation is usually accomplished by passing samples of the received signal through a linear adaptive equalizer consisting of a tapped delay line (TDL) having adjustable coefficients. In this form of equalizer structure, the current and past values of the received signal are linearly weighted by equalizer coefficients and summed to produce the output. Most of the known methods used to adjust the tap coefficients of the equalizer are iterative in which some error criterion is minimized [7, 13]. In such techniques, a known sequence of a white spectrum is transmitted; based on the difference between this known sequence and the output sequence of the equalizer its coefficients are determined.

However, the distortion caused by the dispersive channel is nonlinear in nature in most of the practical situations. The received signal at each sample instant may be considered as a nonlinear function of the past values of the transmitted symbols. Further, since the nonlinear distortion varies with time and from place to place, effectively the overall channel response becomes a nonlinear dynamic mapping. Because of this, the performance of the linear TDL equalizer is limited.

Recently, for classification and optimization problems, artificial neural networks (ANNs) have been found to be a powerful technique [8, 9, 14]. There are extensive applications of ANN in the field of pattern recognition, functional approximation, nonlinear control, system identification, forecasting, etc. Because of their large parallelism and nonlinear processing characteristics, ANNs are capable of performing complex nonlinear mapping between their input space and output space. They are capable of forming arbitrarily nonlinear decision boundaries to take up complex classification tasks. A channel equalizer using a multilayer perceptron (MLP) has been reported recently [2]. It has been shown that the ANN based equalizers are capable of performing quite well in compensating the nonlinear distortion introduced by the channel. An MLP based decision feedback equalizer has been reported [10]. It is reported that in case of nonlinear channel models the MLP structures outperforms the LMS equalizer. Further, channel equalization based on a different structure, i.e. polynomial perceptron network (PPN) has been reported in [3] which is capable of forming highly nonlinear decision boundaries between the two classes of equalizer output. A radial basis function (RBF) equalizer to overcome the cochannel interference [4] and an adaptive Bayesian equalizer with decision feedback has been reported [5]. It has been shown that the RBF structures outperform the LMS based equalizer in both linear and nonlinear channels. Recently, a Pao network applied to digital communication channel equalization has been reported [1]. With preliminary simulation results it has been pointed out that this equalizer structure performs much better than an LMS based equalizer. Performance comparison between a functional link ANN (FLANN) based equalizer and an LMS equalizer has been reported [12]. Here, the superior performance of FLANN based equalizer has been highlighted in terms of the attainable MSE level over a wide variation of eigenvalue ratio (EVR) of the input correlation matrix. However, detailed performance comparison between equalizers of different ANN structure has not been attempted so far.

In this paper, a structure based on a FLANN for adaptive channel equalization has been proposed. Performance comparison between the three ANN (i.e. MLP, PPN and FLANN) based equalizer structures along with a simple least mean square (LMS) based equalizer has also been carried out. Further, the performance of the four structures under different noise conditions and over a wide range of EVR of the input correlation matrix has been evaluated through extensive simulation studies. Mean square error (MSE) as well as the bit-error rate (BER) have been chosen as performance criteria in all these simulation studies. It is seen that in case of LMS equalizer, increasing the EVR has the effect of slowing down the rate of convergence and also increasing the MSE floor [7]. In case of the ANN based equalizer, the effect of increase of the EVR is the same. However, because of the nonlinear processing of signals, the MSE floor is much less than the additive noise in the channel for a wide variation of EVR. It has been shown that all the three ANN based equalizer structures perform much better than the LMS equalizer. Between the three ANN structures the FLANN based equalizer found to perform best in terms of BER and attainable MSE floor level.

This paper is organized as follows. Section 2 introduces a general nonlinear channel model used in the equalization problem. The concept of pattern classification to the problem of channel equalization has been introduced in Section 3. Introduction to MLP and the backpropagation (BP) algorithm used in all the ANN based equalizers is briefly stated in Section 4. The PPN has been described in Section 5. The proposed FLANN structure for the equalization problem is presented in Section 6. Channel equalizer utilizing three different types of ANN structures has been explained in Section 7. Detailed simulation studies in Section 8 provide performance evaluation of each type of equalizer structure and performance comparison between the four structures. The effect of EVR and signal to noise ratio (SNR) on the performance of the equalizer has also been described in this section. Finally, conclusion is drawn in Section 9, where we summarize the relative performance of the three ANN based equalizers and the LMS based equalizer.

2. The nonlinear channel model

Fig. 1 depicts a typical adaptive channel equalizer. The combined effect of the transmitted filter and the transmission medium is included in the 'channel'. Let the transmitted sequence $t(n)$ be real and multivalued as used in a K-ary pulse amplitude modulation (PAM) with a symbol constellation defined by

$$t_s = 2s - K - 1, \quad 1 \leqslant s \leqslant K. \tag{1}$$

Further, the transmitted data sequence $t(n)$ is assumed to be independent and equiprobable for all t_s. The channel output $a(n)$ is corrupted by an additive Gaussian white (assumed) noise with variance of σ^2. The task of the equalizer at the sampling time n is to set its coefficients in such a way that its output $z(n)$ is a close estimate of the desired output $d(n)$.

A widely used model for a linear dispersive channel is the finite impulse response (FIR) model. The

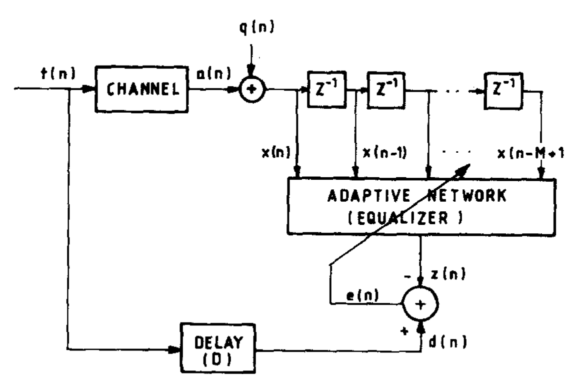

Fig. 1. Schematic of adaptive channel equalizer.

output of the FIR channel may be written as

$$a(n) = \sum_{i=0}^{N_h-1} h(i)t(n-i) + q(n), \quad (2)$$

where $h(i)$ are the channel taps and N_h is the length of the channel impulse response. If nonlinear distortion of the channel is to be considered, then the channel model should be treated as nonlinear and its output may be expressed as

$$a(n) = f(t(n), t(n-1), \dots, t(n-N_h+1); \\ h(0), h(1), \dots, h(N_h-1)) + q(n), \quad (3)$$

where $f(\cdot)$ is some nonlinear function. Due to physical constraints the dynamics of model (3) is always stable and thus produces a bounded output for a bounded input. In the case of linear tapped delay equalizer, the coefficients are formed to force the combined channel and equalizer impulse response to approximate to a unit pulse, whereas, in the case of an ANN based equalizer, channel equalization may be carried out considering it to be a nonlinear pattern classifier as described below.

3. Channel equalizer as pattern classifier

The channel output vector is given by

$$X(n) = [x(n) \;\; x(n-1) \;\; \dots \;\; x(n-M+1)]^{\mathrm{T}}, \quad (4)$$

where M is the equalizer order and $[\cdot]^{\mathrm{T}}$ denotes transpose. The ANN utilizes this information to produce an output which is an estimate of the desired output $d(n) = t(n-D)$. The decision delay of the equalizer is denoted by D. The channel equalization problem may be viewed as a pattern classification task as described below.

Depending on the value of the channel output vector $X(n)$, the equalizer tries to estimate an output which is close to one of the transmitted values, $t_s, s = 1, 2, \dots, K$. In other words, the equalizer seeks to classify the vector $X(n)$ into any one of the K classes. According to the values of M and N_h, the current and past $J-1$ number of the transmitted symbols affect the classification decision of the equalizer. The J-dimensional transmitted symbol vector at the time instant n is given by

$$\boldsymbol{t}(n) = [t(n) \;\; t(n-1) \;\; \dots \;\; t(n-J+1)]^{\mathrm{T}}, \quad (5)$$

where $J = M + N_h - 1$. The total number of possible combinations of $\boldsymbol{t}(n)$ is given by $N_t = K^J$.

When the noise $q(n)$ is zero, let the M-dimensional channel output vector be given by

$$B(n) = [b(n) \;\; b(n-1) \;\; \dots \;\; b(n-M+1)]^{\mathrm{T}}. \quad (6)$$

Corresponding to each one of the $\boldsymbol{t}(n)$ vector there will be a $B(n)$ vector. Thus, $B(n)$ will also have N_t number of possible combinations, called desired channel states. These N_t states are to be partitioned into K classes $C_i, i = 1, 2, \dots, K$, depending on the value of $d(n)$. The states belonging to class C_i are given by

$$B(n) \in C_i \quad \text{if } d(n) = t_i. \quad (7)$$

When the white Gaussian noise is added to the channel, $B(n)$ becomes $X(n)$, which is a stochastic vector. Now the channel observation vector is a random process which possesses a conditional Gaussian density function centered at each of the desired channel states. Since each of t_i is assumed to be equiprobable, the number of channel states in each class is given by N_t/K. The observation vectors form clusters around the desired channel states and thus the means of these data clusters are the desired states. Decision boundaries can be formed in the observed pattern space to classify the observed vectors between K classes. Therefore, determining the value of the transmitted symbol $d(n)$ with the knowledge of the observation vector $X(n)$ is basically a classification problem. The ANNs can be employed for this purpose because of their capability of forming complex decision regions in the pattern space. Thus, the channel equalization is equivalent to partitioning of the M-dimensional channel output vectors into K decision regions.

For this purpose, a decision function may be formed as follows [16]:

$$\mathrm{DF}(X(n)) = w_0 + w_1 x(n) + w_2 x(n-1) \\ + \dots + w_M x(n-M+1). \quad (8)$$

Here $w_i, i = 0, 1, \dots, M$, are the weight parameters. Ignoring the time index n, the decision function may be written as

$$\mathrm{DF}(X) = W^{\mathrm{T}} X, \quad (9)$$

where X is the current channel observation vector augmented by 1 and $W = [w_0 \;\; w_1 \;\; \dots \;\; w_M]^{\mathrm{T}}$ is the weight parameter vector. For the K classes, K number of decision functions are formed with the property

$$\mathrm{DF}_i(X) = W_i^{\mathrm{T}} X \begin{cases} \geqslant 0 & \text{if } X \in C_i, \\ < 0 & \text{otherwise,} \end{cases} \tag{10}$$

for $i = 1, 2, \ldots, K$. Here W_i is the weight vector associated with the ith decision function. In case of a two class problem $K = 2$, and a single decision function is formed. The decision is taken as follows:

$$\begin{aligned} &X \in C_1 \quad \text{if } \mathrm{DF}(X) \geqslant 0, \\ &X \in C_2 \quad \text{otherwise}. \end{aligned} \tag{11}$$

Linear decision functions may be utilized when the pattern classes are linearly separable. However, in many practical situations, these are not linearly separable. Thus, there is a need of nonlinear discriminant functions. A generalized nonlinear decision function may be formed as

$$\begin{aligned} \mathrm{DF}(X) &= w_0 + w_1 f_1(X) + w_2 f_2(X) + \cdots + w_{N_f}(X) \\ &= \sum_{i=0}^{N_f} w_i f_i(X), \end{aligned} \tag{12}$$

where the $\{f_i(X)\}$, $i = 1, 2, \ldots, N_f$, are real, single-valued functions of the input pattern X, $f_0(X) = 1$ and $N_f + 1$ is the number of terms used in the expansion of the decision function DF(X). Depending on the choice of the functions $\{f_i(X)\}$ and the value of N_f, varieties of complex decision functions may be generated.

Let us define a vector X^* where the components are the functions $f_i(X)$ and is given by

$$X^* = [1 \; f_1(X) \; f_2(X) \; \ldots \; f_{N_f}(X)]^{\mathrm{T}}. \tag{13}$$

Using (13) the decision function may be expressed as

$$\mathrm{DF}(X) = W^{\mathrm{T}} X^*. \tag{14}$$

Thus, by using $\{f_i(X)\}$ the $(M + 1)$-dimensional augmented channel observation vector X has been transformed into an $(N_f + 1)$-dimensional vector X^*. Using this decision function, complex decision boundaries can be formed. This has been achieved by the use of ANN structures for the channel equalization problem and is described in the subsequent sections.

4. The multilayer perceptron network

Artificial neural networks based on MLP are feedforward nets with one or more layer of nodes between its input and output layers. Because of the nonlinearity used within the nodes, the MLP is capable of forming arbitrarily complex decision regions in the pattern space. The popular BP algorithm, which is a generalization of the LMS algorithm, is used to train the MLP. The BP algorithm is an iterative gradient search algorithm designed to minimize the MSE between the output of the MLP and its corresponding desired output.

Consider an L-layer MLP [11] as shown in Fig. 2. This network has N_I inputs, N_l hidden layer nodes in layer l, $l = 1, 2, \ldots, L - 1$, and N_L output nodes. At training sample n, the relationship between input and output of a node in a layer $l + 1$ is characterized by a nonlinear recursive difference equation

$$x_i^{(l+1)} = g(S_i^{(l+1)}), \tag{15}$$

where

$$S_i^{(l+1)} = \sum_{j=1}^{N_l} w_{ij}^{(l)} x_j^{(l)} + \theta_i^{(l)}, \tag{16}$$

in which $x_i^{(l+1)}$ is the ith node output of $(l + 1)$th layer, $w_{ij}^{(l)}$ is the connecting weight between the ith node of $(l + 1)$th layer and jth node of lth layer, and $\theta_i^{(l)}$ is the weight between a fixed bias of $+1$ and the ith node of $(l + 1)$th layer. The nonlinear function $g(\cdot)$ is generally chosen as a sigmoid function as

$$g(\phi) = \frac{1}{1 + \exp(-\phi)} \tag{17}$$

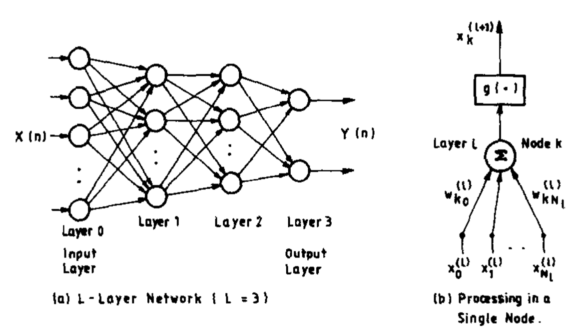

Fig. 2. Multilayer perceptron.

or as a hyperbolic tangent function given by

$$g(\phi) = \tanh(\phi) = \frac{1 - \exp(-2\phi)}{1 + \exp(-2\phi)}. \qquad (18)$$

The error signal $\varepsilon_j(n)$ at the nth training time required for weight adaptation is the difference between the desired response and the output of the perceptron

$$\varepsilon_j(n) = d_j(n) - y_j(n), \quad j = 1, 2, \ldots, N_L, \qquad (19)$$

where $d_j(n)$ and $y_j(n)$ are the desired response and output at the jth node of the output layer, respectively, and N_L is the number of nodes in the output layer. Thus, the sum of error squares produced by the network is given by

$$E(n) = \sum_{j=1}^{N_L} [\varepsilon_j(n)]^2. \qquad (20)$$

The BP algorithm minimizes the cost functional $E(n)$ recursively altering the coefficients $\{w_{ij}^{(l)}, \theta_i^{(l)}\}$ based on a gradient search technique. The MSE is defined as $E(n)/N_L$. The BP algorithm for weight adaptation of the entire network may be given by [11]

$$w_{ij}^{(l)}(n+1) = w_{ij}^{(l)}(n) + \alpha \Delta_{ij}(n) + \beta \Delta_{ij}(n-1), \qquad (21)$$

where

$$\Delta_{ij}(n) = \delta_i^{(l+1)} x_j^{(l)}, \quad l = L-1, L-2, \ldots, 0,$$

$$\delta_i^{(p)} = \begin{cases} \varepsilon_i(n) g'(S_i^{(p)}) & \text{for output layer } (p = L), \\ \left[\sum_{k=1}^{N_{p+1}} \delta_k^{(p+1)} w_{ki}^{(p)}(n)\right] g'(S_i^{(p)}) & \text{for other layers } (p = L-1, L-2, \ldots, 1). \end{cases}$$

The partial derivative of the nonlinear sigmoid function (17) or hyperbolic tangent function (18) with respect to S is given by $g'(s)$ and α and β are the learning rate and momentum rate, respectively, whose values should lie between 0 and 1.

5. The polynomial perceptron network

Weierstrass approximation theorem states that any function which is continuous in a closed interval can be uniformly approximated within any prescribed tolerance over that interval by some polynomial. Considering a binary PAM system (i.e. $K = 2$ in (1)) the channel equalization becomes a two class classification problem, and a decision boundary can be established between the pattern classes. Fig. 3 depicts a PPN network where X is the input pattern given by

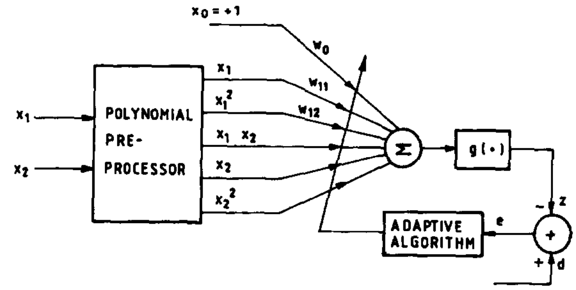

Fig. 3. Polynomial perceptron network.

$$X = [x_1 \; x_2 \; \ldots \; x_M]^{\mathrm{T}}. \qquad (22)$$

Considering a two-dimensional pattern $X = [x_1 \; x_2]^{\mathrm{T}}$ and polynomial order 2, the decision function (14) may be written as

$$\mathrm{DF}(X) = W^{\mathrm{T}} X^*, \qquad (23)$$

where

$$W = [w_0 \; w_{11} \; w_{12} \; w_{22} \; w_1 \; w_2]^{\mathrm{T}} \qquad (24)$$

and

$$X^* = [1 \; x_1^2 \; x_1 x_2 \; x_2^2 \; x_1 \; x_2]^{\mathrm{T}}. \qquad (25)$$

The general quadratic case can be formed by considering all combination of components of X which forms terms of degree two or less. Thus, for an M-dimensional pattern,

$$\begin{aligned} \mathrm{DF}(X) &= \sum_{j=1}^{M} w_{jj} x_j^2 + \sum_{j=1}^{M-1} \sum_{k=j+1}^{M} w_{jk} x_j x_k \\ &\quad + \sum_{j=1}^{M} w_j x_j + w_0 \\ &= W^{\mathrm{T}} X^*. \end{aligned} \qquad (26)$$

The number of terms needed to describe a polynomial decision function grows rapidly as the polynomial degree r and the dimension M of the pattern increases. For the M-dimensional case, the number of coefficients in a function of rth degree is given by

$$N_{M_r} = {}^{M+r}C_r = \frac{(M+r)!}{M!r!}. \quad (27)$$

For the equalization problem, we have seen that polynomial degree of 2 or 3 and pattern dimension of 2 to 4 are enough to describe the nonlinear decision function. The PPN channel equalizer is basically a single-layer ANN. The input pattern X to the PPN at time n is the channel output vector $X(n)$. This is then converted into $X^*(n)$ by passing it into a polynomial preprocessor. The weighted sum of the components of $X^*(n)$ is passed through a nonlinear function (17) or (18) to produce the output $z(n)$. The output of the PPN is compared with the desired response to generate an error $\varepsilon(n)$ which is then used to update its weights by the BP algorithm of (21).

6. The functional link ANN

In the case of FLANN based channel equalizer also, the problem of equalization is treated as a classification problem. FLANN is a single-layer nonlinear network. In contrast to the linear weighting of the input pattern produced by the linear links of MLP, the functional link acts on an element of a pattern or on the entire pattern itself by generating a set of linearly independent functions, then evaluating these functions with the pattern as the argument. Thus, separability is possible in the enhanced space [6, 11]. Further, the FLANN structure offers less computational complexity compared to that of MLP because of its single-layer structure.

A FLANN structure is shown in Fig. 4. Let us consider a two-dimensional input pattern $X = [x_1 \;\; x_2]^T$. This pattern has been enhanced by functional expansion using trigonometric functions such as

$$[x_1 \cos(\pi x_1) \sin(\pi x_1) \ldots \cos(2\pi x_1) \sin(2\pi x_1) \ldots$$
$$x_2 \cos(\pi x_2) \sin(\pi x_2) \ldots \cos(2\pi x_2) \sin(2\pi x_2) \ldots$$
$$x_1 x_2]^T.$$

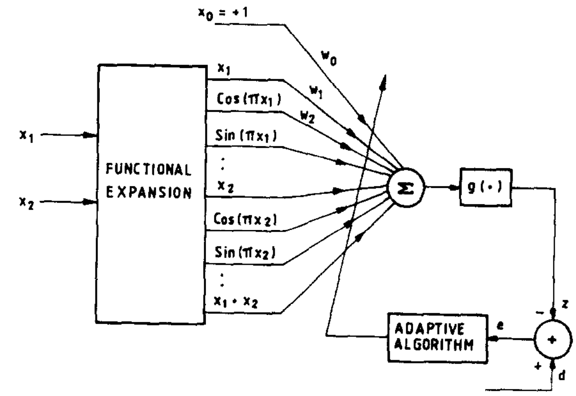

Fig. 4. Functional link ANN.

The weighted sum of the components of the enhanced input is passed through a hyperbolic tangent nonlinear function to produce an output z. The error resulted out of comparison between output of the FLANN and desired response is used to update the weights of the FLANN by the BP algorithm.

6.1. Justification for trigonometric polynomials

The FLANN structure with functional expansion using trigonometric functions has the following advantages with respect to the MLP and PPN based structures.

(i) Since MLP structure is multilayered and the BP algorithm involves high computational complexity, this structure requires excessive training time for learning. Further, the number of weights and in turn the training time increases as the number of layers and the nodes in a layer increases.

(ii) The PPN and the FLANN structures are single layered and thus the training time is must less than that of the MLP structure. However, in the case of the PPN, the number of weights grow rapidly as the polynomial order and the dimension of the input pattern increases.

(iii) It has been pointed out that if appropriate trigonometric polynomials are used in the preprocessor, the weight solution will approximate the terms in the multidimensional Fourier series decomposition version of the desired response function [17]. In case of a FLANN, the link acts on an element of a pattern or on the entire pattern itself by generating a set of independent functions. Then, these functions are evaluated with the pattern as the argument. The functions are chosen as a subset of a complete set of orthonormal basis functions spanning an n-dimensional representation space, such as $\sin \pi x$, $\cos \pi x$, $\sin 2\pi x$, $\cos 2\pi x$, and so on. The net effect is to map the input pattern into a larger pattern space. However, when the outer product terms were used in combination with the functional expansion good results were obtained in the case of learning a two-variable function [11].

(iv) Besides trigonometric functions, other basis functions such as Gaussian or other orthogonal polynomials (e.g. Legendre, Chebyshev, etc.) may also be used. However, the Fourier basis functions were chosen here because under certain conditions this basis forms a more compact representation than a local Gaussian basis, and also the *sin* and *cos* functions can be computed quickly. Other bases are possible and may be chosen to be optimally useful for a particular class of problems or perhaps to maximize a performance for a special hardware [15].

6.2. Learning with FLANN

Let us consider the possibility of classification with a FLANN. Let there be p patterns each with n elements. The FLANN has n inputs and a single output node. All the inputs are connected to the output node by weights defined by $W = [w_0 \; w_1 \; \dots \; w_{n-1}]^{\mathrm{T}}$. For pattern j, the input pattern is defined by $X^{(j)} = [x_0^{(j)} \; x_1^{(j)} \; \dots \; x_{n-1}^{(j)}]$. The output for the jth pattern is given by

$$z^{(j)} = g(Y^{(j)}), \tag{28}$$

where

$$Y^{(j)} = \sum_{k=0}^{n-1} x_k^{(j)} w_k = X^{(j)} W. \tag{29}$$

Here $g(\cdot)$ is a nonlinear function given by (18), $x_0^{(j)}$ is a fixed bias unit set to $+1$ and its connection weight is w_0. Then, (29) may be written as

$$X^{(j)} W = Y^{(j)}, \tag{30}$$

where $Y^{(j)}$ may be found from (28) as

$$Y^{(j)} = \frac{1}{2} \log_e \left(\frac{1 + z^{(j)}}{1 - z^{(j)}} \right). \tag{31}$$

Considering all p number of patterns, (30) may be written as

$$\underset{(p \times n)}{X} \; \underset{(n \times 1)}{W} = \underset{(p \times 1)}{Y}, \tag{32}$$

where

$$X = [X^{(1)} \; X^{(2)} \; \dots \; X^{(p)}]^{\mathrm{T}},$$

$$Y = [Y^{(1)} \; Y^{(2)} \; \dots \; Y^{(p)}]^{\mathrm{T}}.$$

Thus, solving the weights of the FLANN requires solution of p number of simultaneous equations given by (32). Depending on the value of p and n three cases may be visualized.

Case I: $p = n$ and the determinant of X, $\operatorname{Det} X \neq 0$; then W may be found as

$$W = X^{-1} Y. \tag{33}$$

Case II: $p < n$, X is partitioned to obtain X_F of dimension $p \times p$ and by setting the weights $w_k = 0$ for $k \geqslant p$, W is modified to W_{F}. If $\operatorname{Det} X_{\mathrm{F}} \neq 0$, the solution is given by

$$W_{\mathrm{F}} = X_{\mathrm{F}}^{-1} Y. \tag{34}$$

Case III: $p > n$, the columns of X are increased by functional expansion so that n is increased to n_{FL}, X becomes X_{FL} and W is modified to W_{FL}. By suitably choosing the expansion terms, n_{FL} can always be made equal to or greater than p. Now we have

$$\underset{(p \times n_{\mathrm{FL}})}{X_{\mathrm{FL}}} \; \underset{(n_{\mathrm{FL}} \times 1)}{W_{\mathrm{FL}}} = \underset{(p \times 1)}{Y}. \tag{35}$$

If $p = n_{\mathrm{FL}}$ and $\operatorname{Det} X_{\mathrm{FL}} \neq 0$, then

$$W_{\mathrm{FL}} = X_{\mathrm{FL}}^{-1} Y \tag{36}$$

and when $p < n_{\mathrm{FL}}$ one can obtain the solution for W_{FL} as given in Case II.

This analysis indicates that the functional expansion model always yields a flat net solution. The FLANN obtains the solution for W iteratively by using the BP algorithm (21).

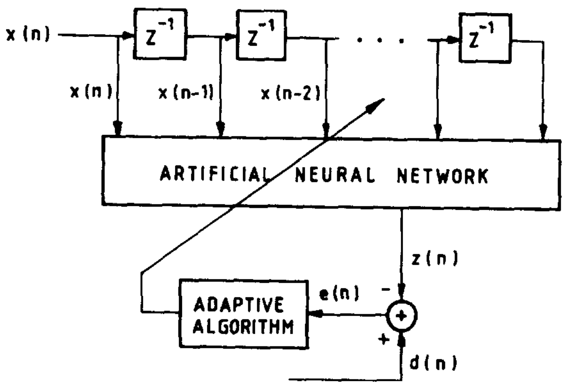

Fig. 5. ANN based channel equalizer.

7. Channel equalization with ANN

An equalizer structure based on ANN is shown in Fig. 5. The received signal $x(n)$ passes through a TDL to produce the observation vector $X(n)$, and this in turn is fed to the ANN. The output of the ANN, $z(n)$, is compared with the desired response to produce an error $\varepsilon(n)$ which is used to update the weights of the ANN.

7.1. The MLP channel equalizer

We have found that the performance of a two-layer perceptron (an input, one hidden and one output layer) based channel equalizer is quite satisfactory. By introducing more hidden layers, it will increase the complexity of the network and the training time. Since a three-layer perceptron with enough nodes in each layer can generate arbitrarily complex decision regions, there is no need to increase the number of hidden layers. Let us consider a two-layer perceptron structure ($L = 2$ in Fig. 2, i.e. an input layer, one hidden layer and an output layer) to be used in the equalization problem as shown in Fig. 5. This structure has M, N_1, and 1 number of nodes in the input, hidden, and output layer, respectively. At the time instant n, the M-dimensional channel observation vector

$$X(n) = [x(n)\ x(n-1)\ \dots\ x(n-M+1)]^{\mathrm{T}} \tag{37}$$

enters into the ANN. Thus, signals at the input layer of the MLP can be given by an $X(n)$. The weighted sum of the input passes through a sigmoid-type nonlinearity to produce an output of the hidden unit. Corresponding to N_1 nodes of the hidden layer, the N_1 dimensional output vector of the hidden layer is given by

$$X^{(1)} = [x_1^{(1)}\ x_2^{(1)}\ \dots\ x_{N_1}^{(1)}]^{\mathrm{T}}, \tag{38}$$

where

$$x_i^{(1)} = g\left(\sum_{j=1}^{M} w_{ij}^{(0)} x(n-j+1) + \theta_i\right), \tag{39}$$

θ_i is the weight between a fixed bias of $+1$ and the ith node of first layer.

The weighted sum of the outputs of the hidden layer produces the final output of the MLP and is given by

$$z(n) = g\left(\sum_{k=1}^{N_1} w_k^{(1)} x_k^{(1)} + \theta_1\right). \tag{40}$$

Here $z(n)$ is an estimate of the desired output of the equalizer at time n. The nonlinear function $g(\cdot)$ is given by either (17) or (18). It has been seen that the MLP approach of channel equalization offers equal effectiveness in case of linear or nonlinear channels with white or colored noise. This provides a general solution to the problem of channel distortion in digital communication systems.

7.2. The PPN channel equalizer

In the case of PPN, the input vector $X(n)$, at the time instant n, is given by (37). Using the structure shown in Fig. 3, satisfactory results have been reported for this problem [3]. With a polynomial degree of 3 ($r = 3$) and TDL order 4 ($M = 4$), we have shown that the MSE and BER performance of a PPN based equalizer is substantially improved over an LMS based equalizer. Further, it has been shown that the PPN based equalizer performance is comparable to that of the MLP equalizer. However, in the former case, the training time is much less and convergence speed is faster because of its single-layer structure.

7.3. The FLANN channel equalizer

The proposed FLANN based channel equalizer utilizes the structure given in Fig. 4. At the time instant n, the input vector $X(n)$ of (37) is used as the input pattern in this structure. The functional expansion is carried out by using cosine, sine and cross products between the components of the input vector as given in Section 6. In the simulation study we have taken a four-dimensional input pattern and it has been enhanced to 18 elements by using the functional expansion. The performance of this equalizer is found to be the best between the three ANN based and the LMS based equalizers. The training time and the complexity of FLANN structure are similar to that of PPN; however, it outperforms the PPN based equalizer in terms of BER and MSE level attained.

8. Simulation studies

Extensive simulation studies have been carried out for the channel equalization problem using the two discussed ANN based structures and the conventional LMS algorithm. For the channels with different EVR, under various SNR conditions, simulations were carried out using Fig. 5. The MSE as well as the BER were chosen as criteria for the performance evaluation of different equalizer structures. The MSE and BER obtained from the most popularly used LMS based equalizer were taken as the reference level. A channel impulse response given in [7] was used throughout for comparison purpose. Two types of nonlinearity were introduced into the channel and the performance comparison is carried out. Selection of the detailed structure of the ANNs used in the simulation is mainly by experiment. The parameter values such as learning rate, momentum rate, polynomial order, number of functions used in the PPN and FLANN, number of nodes in each layer of the MLP based equalizer, etc., are determined by numerous experiments to give best result in the respective architectures. The channel impulse response used in this study is given by

$$h(i) = \begin{cases} \frac{1}{2}\{1 + \cos[\frac{2\pi}{W}(i-2)]\}, & i = 1,2 \text{ and } 3, \\ 0, & \text{otherwise.} \end{cases} \tag{41}$$

The parameter W determines the EVR of the input correlation matrix, $R = E[X(n)X(n)^{\mathrm{T}}]$. The EVR is defined as $\lambda_{\max}/\lambda_{\min}$ where $\lambda_{\max}$ and $\lambda_{\min}$ are the largest and the smallest eigenvalues of R, respectively. The digital message applied to the channel was in random bipolar form $\{-1, 1\}$ corresponding to a binary PAM system. This was obtained from a uniform distribution. To the channel output a zero mean white Gaussian noise was added. The received signal power is normalized to unity so as to make the SNR equal to reciprocal to noise variance at the input of the equalizer. To study the performance of the equalizer under different EVR conditions of the channel, the parameter W of (41) was varied from 2.9 to 3.5 in steps of 0.2. The value of the EVR is given by 6.08, 11.12, 21.71 and 46.82 for the W value of 2.9, 3.1, 3.3 and 3.5, respectively.

In the MLP based equalizer, the number of nodes used including bias unit in the input and hidden layer was 5 and 9, respectively, with a single output node. The learning rate α and momentum rate β were chosen to be 0.1 and 0.2, respectively. The delay parameter D was set at 2 for all the equalizer structures under study. To have a fair comparison between the PPN and the FLANN based equalizers, in both cases, a four-dimensional input pattern was taken and then, this was enhanced to a 18-dimensional pattern. In both cases the α and the β values were set to 0.2. In case of PPN structure the enhanced pattern was obtained by crossmultiplying the individual terms of the input vector as shown in Section 5, whereas in the FLANN structure, the enhancement is done by the use of trigonometric polynomials and the cross-product terms as explained in Section 6. Finally, to have a reference performance level the above channel was simulated for equalization using the LMS algorithm. The equalizer order was chosen to be 8. The value of α and β was set at 0.02 and 0.5, respectively. It may be questioned how the performance of an LMS equalizer of order 8 can be compared with that of the ANN based structures having more number of weight parameters? However, it has been observed that by increasing the number of weights (i.e. number of order) of the LMS based equalizer does not improve its performance. As the order of this equalizer increases, the total noise power on the equalizer input also increases, and this tends to diminish any advantage gained by increasing the equalizer order [2].

The MSE floor corresponds to the steady-state value of the MSE obtained for a particular equalizer structure with a fixed value of additive noise level. This was obtained after averaging over 500 independent runs each consisting of 3000 iterations. In each run a different random sequence and random starting weights were used. While plotting, the MSE values were averaged over past ten iterations to get a smoothed curve.

Two channels having different impluse responses have been considered for this study. The z-transform of the normalized impluse response of Channel-1 and Channel-2 corresponds to W value of 2.9 and 3.5, respectively, and is given by

$$\begin{aligned} &\text{Channel 1: } 0.209 + 0.955z^{-1} + 0.209z^{-2}, \\ &\text{Channel 2: } 0.340 + 0.876z^{-1} + 0.340z^{-2}. \end{aligned} \tag{42}$$

Figs. 6(a) and (b) depict the learning characteristics of the equalizer based on the three ANN structures (i.e. MLP, PPN and FLANN) and the conventional LMS equalizer with 15 dB SNR for Channel-1 and Channel-2, respectively. From these plots, it can be seen that the FLANN equalizer outperforms the rest three equalizer structures. The MSE floor level of the FLANN equalizer is -37 dB (-20 dB), whereas it is only -12.5 dB (-8 dB) in the case of LMS equalizer for Channel-1 (Channel-2). The MSE floor of the MLP and PPN equalizer settles at -35 dB (-17 dB) and -33 dB (-16 dB), respectively, for Channel-1 (Channel-2). It may be noted that for all the three ANN structures the steady-state MSE converges to a value which is less than the additive noise of -15 dB. This superior performance is due to the nonlinear nature of the equalizer transfer function obtained by the use of the ANN structure. Even in the case of Channel-2 which is more difficult to equalize because of high EVR, the MSE floor of the ANN based equalizer structures is less than the additive noise level of -15 dB. It may be observed that the MSE performance of the PPN and the MLP equalizers is similar.

The MSE performance of the equalizer with the variation of additive noise level for all the four types

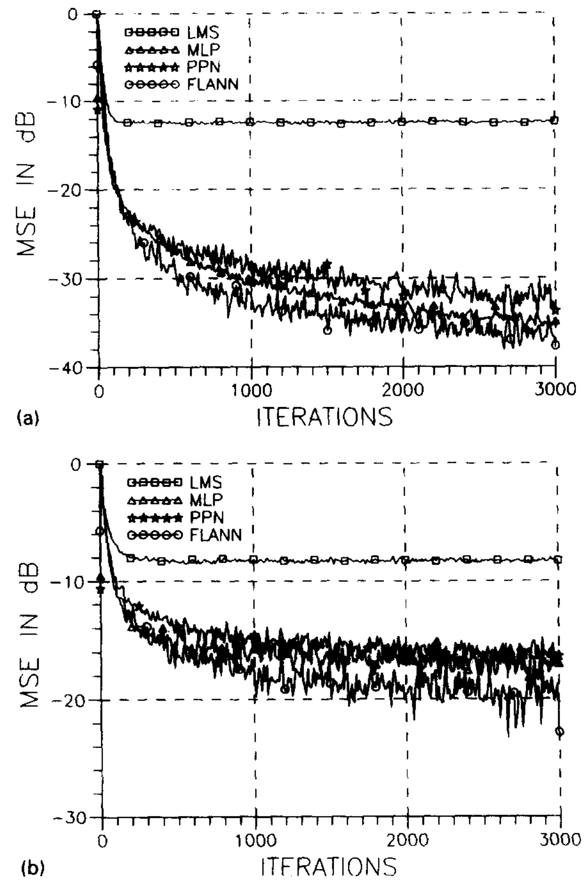

Fig. 6. Convergence characteristics of four equalizers (additive noise = -15 dB): (a) Channel-1; (b) Channel-2.

of structures is shown in Figs. 7(a) and (b) for Channel-1 and Channel-2, respectively. For different values of SNR (ranging from 10 to 30 dB) the MSE floor for each equalizer structure was obtained and plotted here. From these plots it may be observed that the FLANN based equalizer performs the best over a wide range of SNR conditions. It may be seen that in the case of ANN based equalizers there is substantial improvement of MSE performance over the linear LMS based equalizers.

The BER performance of the equalizer structures has been found out from the simulation studies. Each of the equalizer structures had been trained for 3000 iterations for obtaining optimal weight solution. After the learning phase is over, the weights were freezed and the testing of the equalizer

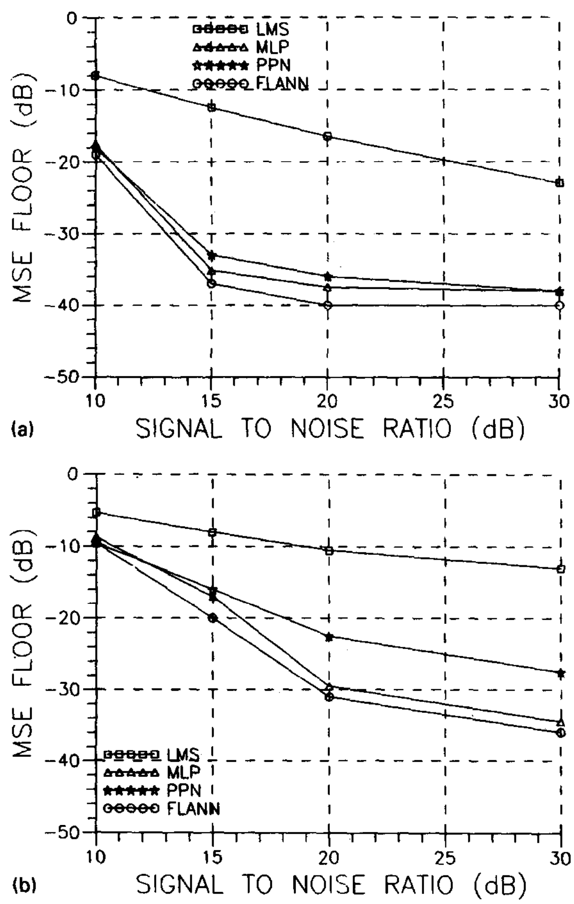

Fig. 7. MSE performance of four equalizers with variation of SNR: (a) Channel-1; (b) Channel 2.

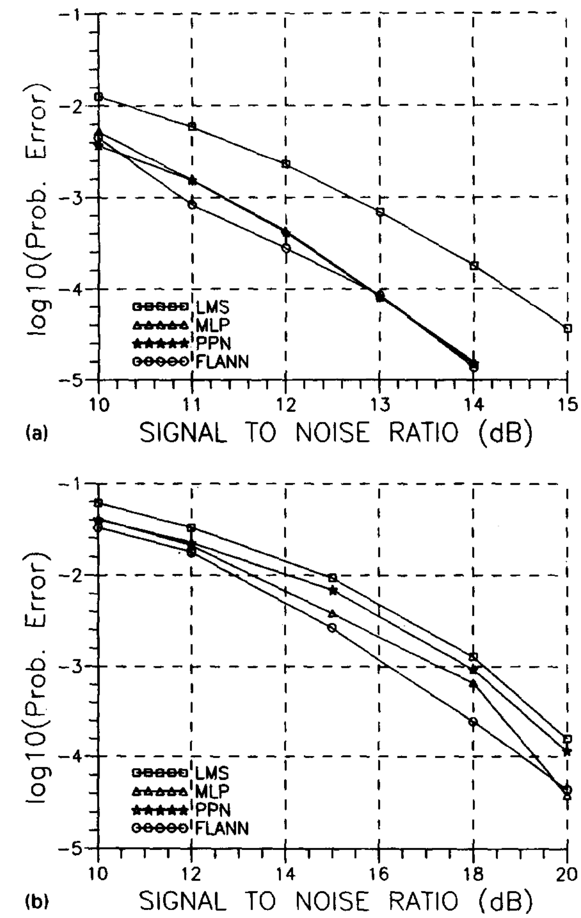

Fig. 8. BER performance of four equalizers with variation of SNR: (a) Channel-1; (b) Channel-2.

was carried out to verify the effectiveness of equalization. The BER was calculated over 100 independent runs each consisting of 10^4 data samples. For plotting convenience, log 10 of the probability of error is calculated and plotted.

Figs. 8(a) and (b) depict the BER performance of the four equalizers for Channel-1 and Channel-2, respectively. From these plots the superior performance of the ANN based equalizer structures is quite evident. For Channel-1 between the ANN structures, the FLANN equalizer performs better than the other two structures for low SNR conditions. However, above 13 dB of SNR the performance of the three ANN structures is similar. In the case of Channel-2, the FLANN equalizer performs better than other two ANN structures upto SNR value of 20 dB. After this SNR value, however, the MLP and FLANN equalizer perform similarly.

The effect of variation of EVR on the MSE performance of the four channel equalizer structures with -15 dB additive noise is shown in Fig. 9(a). From this plot it may be seen that as the EVR increases the MSE floor of all the four equalizers increases. In case of the LMS equalizer, when the EVR increases from 6.08 to 46.82 the MSE floor increases from -12.5 to -8 dB, whereas the MSE floor in case of all the three ANN based equalizers is less than the additive noise in the channel for the same variation of EVR. Between the three ANN structures, the FLANN based equalizer performance is found to be superior. The BER

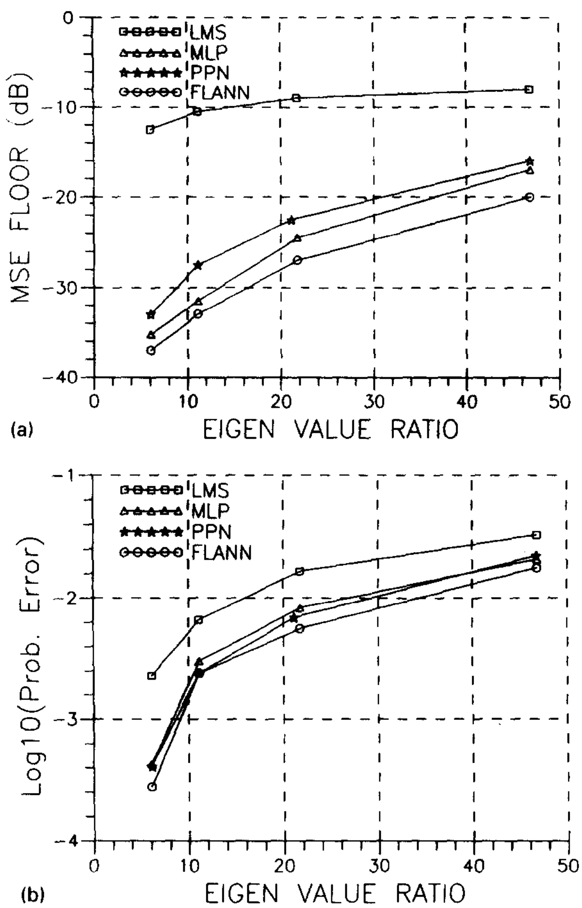

Fig. 9. Performance of four equalizers with variation of EVR: (a) MSE performance with additive noise of -15 dB; (b) BER performance with additive noise of -12 dB.

performance under variation of EVR of the four equalizers with SNR of 12 dB is depicted in Fig. 9(b). From this figure the superior performance of the ANN based equalizers is evident. For the above range of variation of EVR, the BER of the three equalizers is much less than the linear LMS based equalizer. Out of the three ANN equalizers the FLANN structure offers least BER for a wide variation of EVR. It may be observed that the performance degradation due to increase of EVR is less severe in the case of ANN based equalizers than the LMS based equalizer.

To evaluate the performance of the ANN based channel equalizers under nonlinear channel models, two types of nonlinearities are introduced into the channel as given below:

$$\text{Type 1: } x(n) = a(n) + 0.2a(n)^2 - 0.1a(n)^3 + q(n), \quad (43)$$

$$\text{Type 2: } x(n) = a(n) + 0.3[\tanh(a(n))]^2 + q(n). \quad (44)$$

The MSE performance of a worse channel, i.e. the Channel-2 with Type-1 and Type-2 nonlinearity is plotted in Figs. 10(a) and (b), respectively. Type-1 nonlinearity is worse than Type 2. In the case of Type-1 nonlinearity, for LMS equalizer, the MSE floor remains within -10 dB when the SNR varies from 10 to 30 dB. In the case of the ANN based

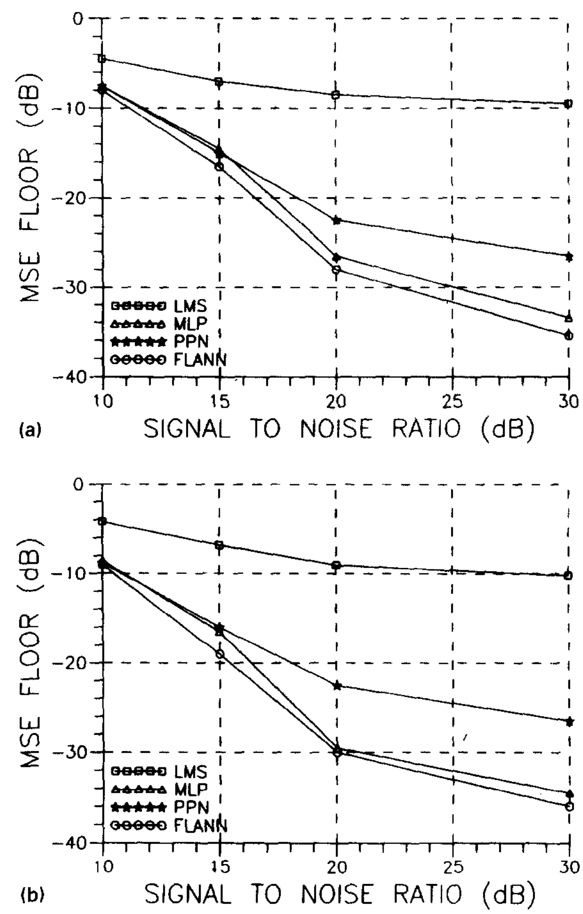

Fig. 10. MSE performance of four equalizers for Channel-2 with nonlinearity: (a) Type-1 nonlinearity; (b) Type-2 nonlinearity.

equalizers, the MSE floor remains below the additive noise level for SNR values above 15 dB. Similar behaviour is observed for the Type-2 nonlinearity. In both types of nonlinearities the performance of FLANN based equalizer is superior than that of other two ANN based equalizers.

The BER performance of the four equalizers for Channel-2 with Type-1 and Type-2 nonlinearity is plotted in Figs. 11(a) and (b), respectively. In both types of nonlinearity, the three ANN equalizer structures provide superior BER performance compared to that of an LMS equalizer. For Type-I nonlinearity, the FLANN structure offers better BER than the MLP and PPN structures for a wide variation of SNR ranging from 10 to 30 dB. In case of Type-2 nonlinearity, the FLANN equalizer maintains its superior performance upto a SNR value of 20 dB after which its performance is similar to that of the MLP structure.

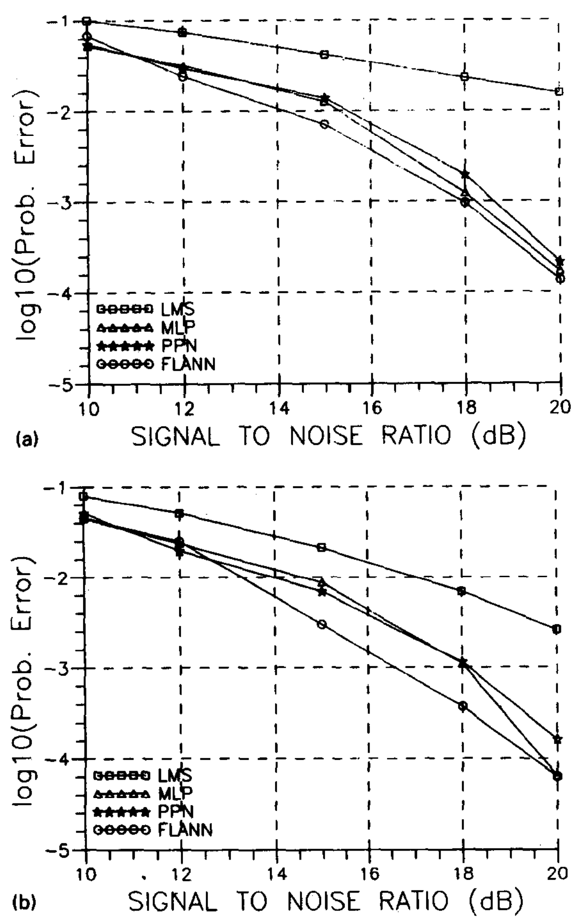

Fig. 11. BER performance of four equalizers for Channel-2 with nonlinearity: (a) Type-1 nonlinearity; (b) Type-2 nonlinearity.

9. Conclusions

The linear signal processing in a conventional channel equalizer limits the performance of the system. To improve the system performance, nonlinear ANN structures have been proposed for the channel equalization task. Considering the channel equalization as a two class classification problem, it has been shown that the performance of the ANN based equalizer improves substantially in terms of MSE floor level and BER. In the case of linear equalization, when the EVR increases the equalizer performance degrades severely. It has been shown that for the ANN based equalizers this performance degradation is less severe than that of the LMS equalizer. Out of the three ANN structures, the performance of MLP and PPN based equalizers is similar. However, the single-layer PPN structure is preferable because of its lower training time requirement and computational complexity. The best performance results from the proposed single-layer FLANN structure. Over a wide range of SNR conditions and large variation of EVR, the FLANN channel equalizer offers advantage in terms of least training time requirement, minimum attainable MSE level and lowest BER. In this structure, the MSE level is found to remain much less than the additive noise in the channel. This superior performance of the ANN structure is due to its capability of forming complex decision regions in the input pattern space. In the case of nonlinear channels also the ANN structures show superior performance with respect to the linear LMS equalizer. In both the types of nonlinear channels the BER and MSE performance of the FLANN based equalizer was found to be better than other two ANN structures. The ANN structures and specially the FLANN structure can be conveniently applied to other applications of communication systems such as echo cancellation, system identification, etc.

Acknowledgements

The authors wish to thank the anonymous referees for their valuable comments and suggestions that improved the quality of the paper.

References

[1] S. Arcens, J.C. Sueiro and A.R.F. Vidal, "Pao networks for data transmission equalization", *Proc. Internat. Jt. Conf. on Neural Networks*, Baltimore, Maryland, June 1992, pp.II. 963–II.968.

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

[3] S. Chen, G.J. Gibson and C.F.N. Cowan, "Adaptive channel equalization using a polynomial perceptron structure", *Proc. IEE*, Vol. 137, Pt. I, October 1990, pp. 257–264.

[4] S. Chen and B. Mulgrew, "Overcoming co-channel interference using an adaptive radial basis function equalizer", *Signal Processing*, Vol. 28, No. 1, 1992, pp. 91–107.

[5] S. Chen, B. Mulgrew and S. McLaughlin, "Adaptive Bayesian equalizer with decision feedback", *IEEE Trans. Signal Processing*, Vol. 41, No. 9, September 1993, pp. 2918–2927.

[6] C.L. Giles and T. Maxwell, "Learning, invariance and generalization in higher order neural networks", *Appl. Optics*, Vol. 26, 1987, pp. 4972–4978.

[7] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1986, Chapter 5, pp. 194–268.

[8] D.R. Hush and B.G. Horne, "Progress in supervised neural networks", *IEEE Signal Process. Mag.*, January 1993, pp. 8–39.

[9] R.P. Lippmann, "An introduction to computing with neural networks", *IEEE ASSP Mag.*, April 1987 pp. 4–25.

[10] M. Meyer and G. Pfeiffer, "Multilayer perceptron based decision feedback equalizers for channels with intersymbol interference", *Proc. IEE*, Vol. 140, Pt. I, No. 6, December 1993, pp. 420–424.

[11] Y.H. Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley, Reading, MA, 1989, Chapter 8, pp. 197–222.

[12] J.C. Patra and R.N. Pal, "Functional link artificial neural network model for channel equalization", *Proc. 5th Internat. Conf. on Signal Processing Applications and Technology, ICSPAT'94*, Dallas, USA, 18–21 October 1994, pp. 1251–1256.

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

[14] D.E. Rumelhart and J.L. McClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, Foundations MIT Press, Cambridge, MA, 1986.

[15] T.D. Sanger, "A tree structured adaptive network for function approximation in high-dimensional spaces", *IEEE Trans. Neural Networks*, Vol. 2, March 1991, pp. 285–293.

[16] J.T. Tou and R.C. Gonzalez, *Pattern Recognition Principles*, Addison-Wesley, Reading, MA, 1981.

[17] B. Widrow and M.A. Lehr, "30 years of adaptive neural networks: Perceptron, madaline and backpropagation", *Proc. IEEE*, Vol. 78, No. 9, September 1990, pp. 1415–1442.