



Adaptive near minimum error rate training for neural networks with application to multiuser detection in CDMA communication systems

S. Chen*, A.K. Samingan, L. Hanzo

School of Electronics and Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, UK

Received 2 October 2003; received in revised form 24 February 2005

Abstract

Adaptive training of neural networks is typically done using some stochastic gradient algorithm that aims to minimize the mean square error (MSE). For many classification applications, such as channel equalization and code-division multiple-access (CDMA) multiuser detection, the goal is to minimize the error probability. For these applications, adopting the MSE criterion may lead to a poor performance. A nonlinear adaptive near minimum error rate algorithm called the nonlinear least bit error rate (NLBER) is developed for training neural networks for these kinds of applications. The proposed method is applied to downlink multiuser detection in CDMA communication systems. Simulation results show that the NLBER algorithm has a good convergence speed and a small-size radial basis function network trained by this adaptive algorithm can closely match the performance of the optimal Bayesian multiuser detector. The results also confirm that training the neural network multiuser detector using the least mean square algorithm, although generally converging well in the MSE, can produce a poor error rate performance.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Neural networks; Adaptive algorithms; Mean square error; Error probability; CDMA; Multiuser detectors; Bayesian detector

1. Introduction

We consider a class of neural network classifiers where pattern vectors are drawn from a finite set and corrupted by an additive noise. Examples include neural network equalizers and multiuser

detectors in communication systems [1–15]. Typically, sample-by-sample adaptation is needed for practical applications to meet real-time computational constraints, and the training of neural network classifiers is usually done using some stochastic gradient algorithm based on the mean square error (MSE) criterion. It is often reported that a nonlinear classifier (e.g. a neural network equalizer or multiuser detector) can provide a

*Corresponding author. Tel./fax: +44 23 8059 6660/4508.
E-mail address: sqc@ecs.soton.ac.uk (S. Chen).

considerable performance improvement over a linear one. However, a close examination of the literature shows that the reported results are often inconsistent, namely many of these reported works do not compare the classification performance of their neural networks with the potentially achievable optimal performance for the given classifier structure. As pointed out by Pados and Papantoni-Kazakos [16], a strange situation exists that, on one hand, the performance of a classifier is evaluated using probability of error while, on the other hand, a different MSE criterion is used at the learning stage.

For linear classifiers, such as linear equalizers and code-division multiple-access (CDMA) multiuser detectors, there is a partial relationship between the MSE and the error probability. A small MSE is usually associated with a small error rate. However, even in the linear case, the minimum MSE (MMSE) solution in general is not the minimum error rate (MER) solution. For the linear equalizer or multiuser detector with binary signalling, it is now well-known that the bit error rate (BER) difference between the MMSE solution and the minimum BER (MBER) one can be large in certain situations [17–29]. Recent research has aimed to develop adaptive linear equalizer and multiuser detector based on the MBER criterion [20,22,25–29]. For nonlinear classifiers, the relationship between the MSE and the error rate is more dubious, and the MMSE solution does not necessarily correspond to a small error rate.¹ In effect, standard adaptive algorithms for training nonlinear classifiers, such as the least mean square (LMS) algorithm, is based on a criterion that may not be relevant to the true performance indicator. Notice that this scenario exists only for the adaptive learning case.

For off-line or block-data based learning, the need for adopting a relevant criterion to train classifiers has always been recognized. Given the underlying pattern space, that is, the information available to a classifier, the maximum a posteriori

probability or Bayesian classifier provides the true optimal performance. The definition of the MER used in this paper is referred to the achievable error rate for a classifier with an additional constraint of a given structure (e.g. a radial basis function (RBF) classifier with a given number of hidden nodes). The basic question is then whether it is possible to achieve this MER and how close it is to the true performance of the Bayesian classifier. It is not surprising that the Bayesian learning approach is the most general block-data based method for training a nonlinear classifier with given structure. Typical Bayesian learning algorithms include the so-called type-II maximum likelihood or evidence procedure [30,31], and the Markov chain Monte Carlo sampling method [32]. If the classifier has a special structure of a kernel representation, the support vector machine [33] and the relevance vector machine [34] have become popular. All these training algorithms suffer from high computational costs and cannot be implemented in a true adaptive sample-by-sample training manner.

The main contribution of this paper is to develop an adaptive near MER training algorithm for a class of neural network classifiers that includes nonlinear equalizers and multiuser detectors. It should be pointed out that adaptive near MER training can in theory be achieved by only adjusting the classifier parameters when a classification error occurs. However, for applications considered in this paper, error rate is typically very small. This strategy is impractical, since it would require an extremely long training period. The approach adopted in this paper is based on a Parzen window or kernel density estimation [35–37] to approximate the error rate from training data and to derive a stochastic gradient adaptive algorithm. The resulting algorithm will be called the nonlinear least error rate (NLER) algorithm. In channel equalization and multiuser detection applications with binary modulation schemes, this NLER algorithm will be referred to as the nonlinear least bit error rate (NLBER). The algorithm is used to train downlink RBF multiuser detectors in CDMA communication systems [38,39]. Convergence rate of the NLBER algorithm is investigated in simulation, and BERs of

¹Multiplying the two-class Bayesian classifier by a positive constant, its MSE value may increase but it remains the optimal nonlinear classifier, see Section 3.2. Basically, the MSE is irrelevant here.

the RBF multiuser detector trained by the LMS and NLBER algorithms are compared with those of the optimal Bayesian multiuser detector. The results obtained show that the NLBER algorithm achieves consistent performance and has a reasonable convergence speed. A small-size RBF network trained by the NLBER algorithm can closely approximate the optimal Bayesian detector. The simulation study also demonstrates that the RBF network trained by the LMS algorithm, although converging consistently in the MSE, can produce poor BER performance.

2. Adaptive near minimum error rate training

Consider a class of nonlinear classifiers that can be represented by

$$\hat{c}(k) = \text{sgn}(y(k)) \text{ with } y(k) = f(\mathbf{r}(k); \mathbf{w}), \quad (1)$$

where k indicates the sample number, $\mathbf{r}(k)$ is an M -dimensional pattern vector with its associated class label $c(k) \in \{\pm 1\}$, $f(\bullet; \bullet)$ denotes the classifier map, the vector \mathbf{w} consists of all the (adjustable) parameters of the classifier, and $\hat{c}(k)$ is the estimated class label for $\mathbf{r}(k)$. The pattern vector $\mathbf{r}(k)$ is assumed to take the form

$$\mathbf{r}(k) = \bar{\mathbf{r}}(k) + \mathbf{n}(k), \quad (2)$$

where the “clean” or noise-free part $\bar{\mathbf{r}}(k)$ takes values from a finite set with equal probability

$$\bar{\mathbf{r}}(k) \in \{\bar{\mathbf{r}}_j, 1 \leq j \leq N_b\} \quad (3)$$

and the noise vector $\mathbf{n}(k)$ is white Gaussian with covariance matrix $E[\mathbf{n}(k)\mathbf{n}^T(k)] = \sigma_n^2 \mathbf{I}$, \mathbf{I} being an identity matrix of appropriate dimension. Each $\bar{\mathbf{r}}_j$ has an associated class label $c^{(j)} \in \{\pm 1\}$.

A usual way of training such a nonlinear classifier is to adjust the classifier’s parameters \mathbf{w} so that the MSE

$$E[(c(k) - y(k))^2] \quad (4)$$

is minimized. Typically, a stochastic gradient algorithm called the LMS can be used in adaptive implementation, and the algorithm has a simple form

$$y(k) = f(\mathbf{r}(k); \mathbf{w}(k - 1)),$$

$$\mathbf{w}(k) = \mathbf{w}(k - 1) + \mu(c(k) - y(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w}(k - 1))}{\partial \mathbf{w}}, \quad (5)$$

where μ is an adaptive gain. However, the true performance criterion is the error rate and it is desirable to develop an adaptive training algorithm based on the MER criterion. We will first consider the theoretical error rate of the classifier (1) and a block-data based training. This will provide insight into the development of our adaptive near MER training algorithm.

2.1. An approximate error rate expression

As an error only occurs when the sign of $y(k)$ is different from $\text{sgn}(c(k))$, the error probability of the classifier (1) is

$$P_E(\mathbf{w}) = \text{Prob}\{\text{sgn}(c(k))y(k) < 0\}. \quad (6)$$

Define the signed variable

$$y_s(k) = \text{sgn}(c(k))y(k) \quad (7)$$

and let the probability density function (p.d.f.) of $y_s(k)$ be $p_y(y_s)$. Then

$$P_E(\mathbf{w}) = \int_{-\infty}^0 p_y(y_s) dy_s. \quad (8)$$

By linearizing the classifier around $\bar{\mathbf{r}}(k)$, it can be approximated as²

$$\begin{aligned} y(k) &= f(\bar{\mathbf{r}}(k) + \mathbf{n}(k); \mathbf{w}) \approx f(\bar{\mathbf{r}}(k); \mathbf{w}) \\ &\quad + \left[\frac{\partial f(\bar{\mathbf{r}}(k); \mathbf{w})}{\partial \mathbf{r}} \right]^T \mathbf{n}(k) \\ &= f(\bar{\mathbf{r}}(k); \mathbf{w}) + e(k), \end{aligned} \quad (9)$$

where $e(k)$ is Gaussian with zero mean and variance

$$\begin{aligned} \rho^2(\mathbf{w}) &= E \left[\sigma_n^2 \left[\frac{\partial f(\bar{\mathbf{r}}(k); \mathbf{w})}{\partial \mathbf{r}} \right]^T \frac{\partial f(\bar{\mathbf{r}}(k); \mathbf{w})}{\partial \mathbf{r}} \right] \\ &= \frac{\sigma_n^2}{N_b} \sum_{j=1}^{N_b} \left[\frac{\partial f(\bar{\mathbf{r}}_j; \mathbf{w})}{\partial \mathbf{r}} \right]^T \frac{\partial f(\bar{\mathbf{r}}_j; \mathbf{w})}{\partial \mathbf{r}}. \end{aligned} \quad (10)$$

²This linearization is only used to derive an approximate expression of error rate.

Essentially, the classifier is approximated as an additive Gaussian noise model

$$y(k) \approx \bar{y}(k) + e(k) \quad (11)$$

when deriving its error rate expression, with $\bar{y}(k)$ taking values from the finite set

$$\bar{y}(k) \in \{\bar{y}_j = f(\bar{\mathbf{r}}_j; \mathbf{w}), 1 \leq j \leq N_b\}. \quad (12)$$

The p.d.f. of $y_s(k)$ can thus be approximated by

$$p_y(y_s) \approx \frac{1}{N_b \sqrt{2\pi} \rho(\mathbf{w})} \sum_{j=1}^{N_b} \exp\left(-\frac{(y_s - \text{sgn}(c^{(j)})\bar{y}_j)^2}{2\rho^2(\mathbf{w})}\right) \quad (13)$$

and the error probability of the classifier is approximately

$$\begin{aligned} P_E(\mathbf{w}) &\approx \frac{1}{N_b \sqrt{2\pi}} \sum_{j=1}^{N_b} \int_{g_j(\mathbf{w})}^{\infty} \exp\left(-\frac{x_j^2}{2}\right) dx_j \\ &= \frac{1}{N_b} \sum_{j=1}^{N_b} Q(g_j(\mathbf{w})), \end{aligned} \quad (14)$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp\left(-\frac{y^2}{2}\right) dy \quad (15)$$

and

$$g_j(\mathbf{w}) = \frac{\text{sgn}(c^{(j)})\bar{y}_j}{\rho(\mathbf{w})} = \frac{\text{sgn}(c^{(j)})f(\bar{\mathbf{r}}_j; \mathbf{w})}{\rho(\mathbf{w})}. \quad (16)$$

The linearization (9) is valid only for small $\mathbf{n}(k)$ (in some statistical sense), but the assumption of small $\mathbf{n}(k)$ usually holds in practice. In general, however, the error rate expression (14) is a good approximation of the true error probability, and minimizing this approximate error rate expression will lead to a near MER solution.

2.2. Approximate minimum error rate solution

If the set described by (3) is known (for example, in equalization application if the channel impulse response (CIR) is known), an approximate MER solution can be obtained by minimizing the

approximate error rate expression (14) numerically. The gradient of $P_E(\mathbf{w})$ is approximately

$$\begin{aligned} \nabla P_E(\mathbf{w}) &\approx -\frac{1}{N_b \sqrt{2\pi}} \sum_{j=1}^{N_b} \exp\left(-\frac{\bar{y}_j^2}{2\rho^2}\right) \frac{\partial g_j(\mathbf{w})}{\partial \mathbf{w}} \\ &\approx -\frac{1}{N_b \sqrt{2\pi} \rho} \sum_{j=1}^{N_b} \exp\left(-\frac{\bar{y}_j^2}{2\rho^2}\right) \\ &\quad \times \text{sgn}(c^{(j)}) \frac{\partial f(\bar{\mathbf{r}}_j; \mathbf{w})}{\partial \mathbf{w}}. \end{aligned} \quad (17)$$

In the above second approximation, we have dropped the term containing $\partial \rho / \partial \mathbf{w}$. The following iterative steepest-descent gradient algorithm can be used to arrive at an approximate MER solution. Given an initial $\mathbf{w}(0)$, at l th iteration, the algorithm computes

$$\bar{y}_j(l) = f(\bar{\mathbf{r}}_j; \mathbf{w}(l-1)), \quad 1 \leq j \leq N_b,$$

$$\begin{aligned} \nabla P_E(\mathbf{w}(l)) &= -\frac{1}{N_b \sqrt{2\pi} \rho} \sum_{j=1}^{N_b} \exp\left(-\frac{\bar{y}_j^2(l)}{2\rho^2}\right) \\ &\quad \times \text{sgn}(c^{(j)}) \frac{\partial f(\bar{\mathbf{r}}_j; \mathbf{w}(l-1))}{\partial \mathbf{w}}, \end{aligned}$$

$$\mathbf{w}(l) = \mathbf{w}(l-1) - \mu \nabla P_E(\mathbf{w}(l)), \quad (18)$$

where μ is a step size. Since $P_E(\mathbf{w})$ is a highly complex nonlinear function of \mathbf{w} , a steepest-descent gradient algorithm may converge slowly. A simplified conjugate gradient algorithm [28,40] with a periodical resetting of the search direction to the negative gradient can alternatively be used to speed up convergence.

Assuming $\partial \rho / \partial \mathbf{w} = 0$ is to assume that the equivalent noise standard deviation ρ is independent of \mathbf{w} . Clearly, ρ depends on the value of \mathbf{w} , unless the algorithm has already converged to the (near) optimal solution \mathbf{w}_{MER} and ρ has been fixed to its optimal value. Therefore, in the algorithm (18), ρ^2 needs to be set appropriately. It is seen that in theory at least an approximate MER solution can be obtained. More importantly, the derivation of this algorithm points out a way of deriving the adaptive algorithms of the next two subsections.

2.3. Block-data based gradient adaptation

In practice, the set of $\bar{\mathbf{r}}_j$ is unknown. The key to developing an effective adaptive algorithm is the p.d.f. $p_y(y_s)$ of the decision variable $y_s(k)$. Parzen window or kernel density estimation [35–37] is a well-known method for estimating a probability distribution. Parzen window method estimates a p.d.f. using a window or block of $y_s(k)$ by placing a symmetric unimodal kernel function (such as the Gaussian function) on each $y_s(k)$. This kernel density estimation is capable of producing reliable p.d.f. estimates with short data records and in particular is extremely natural when dealing with Gaussian mixtures. Given a block of K training samples $\{\mathbf{r}(k), c(k)\}_{k=1}^K$, a kernel density estimate of the true p.d.f. $p_y(y_s)$ is readily given by

$$\hat{p}_y(y_s) = \frac{1}{K\sqrt{2\pi}\bar{\rho}} \sum_{k=1}^K \exp\left(-\frac{(y_s - \text{sgn}(c(k))y(k))^2}{2\bar{\rho}^2}\right), \quad (19)$$

where the kernel width $\bar{\rho}$ is an appropriately chosen positive constant. From the estimated p.d.f. (19), an estimated error probability

$$\hat{P}_E(\mathbf{w}) = \int_{-\infty}^0 \hat{p}_y(y_s) dy_s \quad (20)$$

is obtained, and its gradient $\nabla \hat{P}_E(\mathbf{w})$ can be calculated exactly according to

$$\nabla \hat{P}_E(\mathbf{w}) = -\frac{1}{K\sqrt{2\pi}\bar{\rho}} \sum_{k=1}^K \exp\left(-\frac{y^2(k)}{2\bar{\rho}^2}\right) \times \text{sgn}(c(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w})}{\partial \mathbf{w}}. \quad (21)$$

Thus a block-data based adaptive steepest-descent gradient algorithm can be derived. At l th iteration, the algorithm computes

$$y(k) = f(\mathbf{r}(k); \mathbf{w}(l-1)), \quad 1 \leq k \leq K,$$

$$\nabla \hat{P}_E(\mathbf{w}(l)) = -\frac{1}{K\sqrt{2\pi}\bar{\rho}} \sum_{k=1}^K \exp\left(-\frac{y^2(k)}{2\bar{\rho}^2}\right) \times \text{sgn}(c(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w}(l-1))}{\partial \mathbf{w}},$$

$$\mathbf{w}(l) = \mathbf{w}(l-1) - \mu \nabla \hat{P}_E(\mathbf{w}(l)), \quad (22)$$

where the adaptive gain μ and the kernel width $\bar{\rho}$ are the two algorithm parameters that require tuning. Specifically, μ and $\bar{\rho}$ control the rate of convergence, and $\bar{\rho}$ also helps to determine the accuracy of the p.d.f. and hence error rate estimate. Alternatively, conjugate gradient based adaptation can be adopted.

Several critical points need to be emphasized. Provided that the kernel width $\bar{\rho}$ is chosen appropriately, the Parzen window estimate (19) is an accurate estimate of the true density $p_y(y_s)$ regardless whether the approximation (13) is valid or not. Accuracy analysis of Parzen window density estimate is well documented in the literature. The p.d.f. estimate (19) is known to possess a mean integrated square error convergence rate at order of K^{-1} [35] and it can achieve an accurate estimate with a remarkably short data record. It is also worth re-iterating that the gradient (21) is exact and involves no approximation.

2.4. Stochastic gradient adaptation

Our aim is to develop a stochastic gradient adaptive algorithm with sample-by-sample updating, in a similar manner to the LMS (5). The LMS algorithm is derived from its related ensemble gradient algorithm by replacing the ensemble average of the gradient with a single data point estimate of the gradient. Adopting a similar strategy, at sample k , a single-data-point estimate of the p.d.f. is

$$\hat{p}_y(y_s, k) = \frac{1}{\sqrt{2\pi}\bar{\rho}} \exp\left(-\frac{(y_s - \text{sgn}(c(k))y(k))^2}{2\bar{\rho}^2}\right). \quad (23)$$

Using the instantaneous or stochastic gradient

$$\nabla \hat{P}_E(k; \mathbf{w}) = -\frac{1}{\sqrt{2\pi}\bar{\rho}} \exp\left(-\frac{y^2(k)}{2\bar{\rho}^2}\right) \times \text{sgn}(c(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w})}{\partial \mathbf{w}} \quad (24)$$

a stochastic gradient algorithm is readily given by

$$y(k) = f(\mathbf{r}(k); \mathbf{w}(k-1)),$$

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \frac{\mu}{\sqrt{2\pi\bar{\rho}}} \exp\left(-\frac{y^2(k)}{2\bar{\rho}^2}\right) \times \text{sgn}(c(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w}(k-1))}{\partial \mathbf{w}}, \quad (25)$$

where the adaptive gain μ and the kernel width $\bar{\rho}$ are the two algorithmic parameters that have to be set appropriately. Specifically, they are chosen to ensure adequate performance in terms of convergence rate and steady-state error rate misadjustment.

Following a similar reasoning to the LMS for the MMSE criterion, the algorithm (25) will be called the NLER for the near MER criterion. Specially, in equalization and multiuser detection applications involving binary signalling, this stochastic gradient algorithm will be called the NLBER for the near MBER criterion.

3. Adaptive training of neural network multiuser detectors

The performance of the NLBER algorithm (25) is investigated in an application to multiuser detection in the CDMA downlink (base station to mobile), in which a detector estimates the transmitted information bits of a desired user in the presence of interfering users.

3.1. Synchronous CDMA downlink system model

Using notations from the multi-rate filtering literature [41], the discrete-time baseband model of the synchronous CDMA downlink system supporting N users with M chips per symbol is depicted in Fig. 1, where $b_i(k) \in \{\pm 1\}$ denotes the

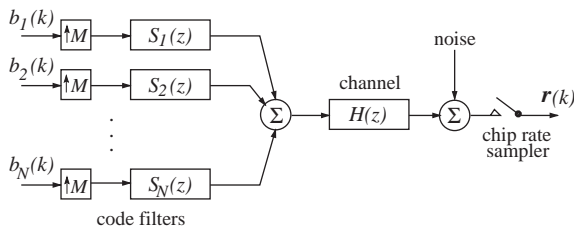


Fig. 1. Discrete-time model of synchronous CDMA downlink.

k th symbol of user i , the unit-length signature sequence for user i is

$$\tilde{\mathbf{s}}_i = [\tilde{s}_{i,1} \ \cdots \ \tilde{s}_{i,M}]^T \quad (26)$$

and the transfer function of the CIR at the chip rate is

$$H(z) = \sum_{i=0}^{n_h-1} h_i z^{-i}. \quad (27)$$

The baseband model for received signal sampled at chip rate is given by [42,43]

$$\mathbf{r}(k) = \mathbf{P} \begin{bmatrix} \mathbf{b}(k) \\ \mathbf{b}(k-1) \\ \vdots \\ \mathbf{b}(k-L+1) \end{bmatrix} + \mathbf{n}(k) = \bar{\mathbf{r}}(k) + \mathbf{n}(k), \quad (28)$$

where the user symbol vector $\mathbf{b}(k) = [b_1(k) \ \cdots \ b_N(k)]^T$, the white Gaussian noise vector $\mathbf{n}(k) = [n_1(k) \ \cdots \ n_M(k)]^T$ with $E[\mathbf{n}(k)\mathbf{n}^T(k)] = \sigma_n^2 \mathbf{I}$, $\bar{\mathbf{r}}(k)$ denotes the noise-free received signal, and the $M \times LN$ system matrix \mathbf{P} has the form

$$\mathbf{P} = \mathbf{H} \begin{bmatrix} \tilde{\mathbf{S}}\mathbf{A} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{S}}\mathbf{A} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \tilde{\mathbf{S}}\mathbf{A} \end{bmatrix} \quad (29)$$

with the $M \times LM$ CIR matrix \mathbf{H} given by

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \cdots & h_{n_h-1} & & & \\ & h_0 & h_1 & \cdots & h_{n_h-1} & & \\ & & \ddots & \ddots & \cdots & \ddots & \\ & & & h_0 & h_1 & \cdots & h_{n_h-1} \end{bmatrix} \quad (30)$$

the normalized user code matrix given by $\tilde{\mathbf{S}} = [\tilde{\mathbf{s}}_1 \ \cdots \ \tilde{\mathbf{s}}_N]$, and the diagonal user signal amplitude matrix given by $\mathbf{A} = \text{diag}\{A_1 \ \cdots \ A_N\}$. The channel intersymbol interference span L depends on the CIR length n_h and the chip sequence length M : $L = 1$ for $n_h = 1$, $L = 2$ for $1 < n_h \leq M$, $L = 3$ for $M < n_h \leq 2M$, and so on.

The detector at the receiver for user i estimates the transmitted bit $b_i(k)$ based on the received

signal $\mathbf{r}(k)$, and has a general form of

$$\hat{b}_i(k) = \text{sgn}(y(k)) \text{ with } y(k) = f(\mathbf{r}(k); \mathbf{w}), \quad (31)$$

where \mathbf{w} is the detector parameter vector for user i . This is obviously an example of the classifier discussed in the previous section with $b_i(k)$ serving as the class label for $\mathbf{r}(k)$. Let the $N_b = 2^{LN}$ possible combinations or sequences of $[\mathbf{b}^T(k) \mathbf{b}^T(k-1) \cdots \mathbf{b}^T(k-L+1)]^T$ be

$$\mathbf{b}^{(j)} = \begin{bmatrix} \mathbf{b}^{(j)}(k) \\ \mathbf{b}^{(j)}(k-1) \\ \vdots \\ \mathbf{b}^{(j)}(k-L+1) \end{bmatrix}, \quad 1 \leq j \leq N_b \quad (32)$$

and $b_i^{(j)}$ the i th element of $\mathbf{b}^{(j)}(k)$. Define the set of N_b noise-free received signal states

$$\mathcal{R} = \{\bar{\mathbf{r}}_j = \mathbf{P}\mathbf{b}^{(j)}, 1 \leq j \leq N_b\} \quad (33)$$

and the set of N_b scalars

$$\{\bar{y}_j = f(\bar{\mathbf{r}}_j; \mathbf{w}), 1 \leq j \leq N_b\}. \quad (34)$$

Notice that $\bar{\mathbf{r}}(k)$ can only take the values from the set \mathcal{R} , the class label for $\bar{\mathbf{r}}_j$ is $b_i^{(j)} \in \{\pm 1\}$ for user i , and \mathcal{R} can be divided into two subsets

$$\mathcal{R}_{\pm} = \{\bar{\mathbf{r}}_j \in \mathcal{R} : b_i^{(j)} = \pm 1\}. \quad (35)$$

3.2. Linear and optimal detectors

A linear detector for user i has the decision variable given by

$$y_L(k) = f_L(\mathbf{r}(k); \mathbf{w}) = \mathbf{w}^T \mathbf{r}(k). \quad (36)$$

The most popular solution for this linear detector is the MMSE one given [42,44–47]

$$\mathbf{w}_{\text{MMSE}} = (\sigma_n^2 \mathbf{I} + \mathbf{P}\mathbf{P}^T)^{-1} \mathbf{p}_i, \quad (37)$$

where \mathbf{p}_i is the i th column of \mathbf{P} . More recently, the linear MBER solution for the linear detector (36) has been derived [28]. However, a linear detector only performs adequately if \mathcal{R}_+ and \mathcal{R}_- are linearly separable. If \mathcal{R}_+ and \mathcal{R}_- are not linearly separable, a linear detector will have a high BER floor even without noise and a nonlinear detector will be required [48]. Even in the case that \mathcal{R}_+ and \mathcal{R}_- are linearly separable, a nonlinear detector can

often outperform a linear one considerably at a cost of increased complexity.

Applying the maximum a posteriori probability principle, it can be shown that the optimal one-shot detector is the following Bayesian one [48]:

$$y_B(k) = f_B(\mathbf{r}(k); \mathbf{w}) = \sum_{\mathbf{r}_j \in \mathcal{R}} \frac{\xi_j b_i^{(j)}}{(2\pi\sigma_n^2)^{M/2}} \times \exp\left(-\frac{\|\mathbf{r}(k) - \bar{\mathbf{r}}_j\|^2}{2\sigma_n^2}\right), \quad (38)$$

where ξ_j are a priori probabilities of $\bar{\mathbf{r}}_j$ and, since all the $\bar{\mathbf{r}}_j$ are equiprobable, $\xi_j = 1/N_b$. The Bayesian decision variable can also be written as

$$y_B(k) = f_B(\mathbf{r}(k); \mathbf{w}) = \sum_{j=1}^{N_b} \beta_j \exp\left(-\frac{\|\mathbf{r}(k) - \bar{\mathbf{r}}_j\|^2}{2\sigma_n^2}\right) \quad (39)$$

with

$$\beta_j = \frac{b_i^{(j)}}{N_b(2\pi\sigma_n^2)^{M/2}}. \quad (40)$$

Notice that, for binary data symbols $\{\pm 1\}$, multiplying all β_j by any positive constant still gives the same optimal Bayesian solution, and the performance of the Bayesian solution is insensitive to whether a precise noise variance or an estimate is used [49,50]. For example, substituting σ_n^2 in (39) by, say, $0.5\sigma_n^2$ or $2\sigma_n^2$, the BER performance are indistinguishable from the exact Bayesian solution. Implementation of the optimal Bayesian detector (39) is computationally very expensive with the associated difficulty of adaptively estimating the set of noise-free signal states (33).

3.3. Adaptive radial basis function network detector

To test the NLBER algorithm for adaptive training of neural network multiuser detectors, we choose the RBF network detector of the form

$$y_{\text{RBF}}(k) = f_{\text{RBF}}(\mathbf{r}(k); \mathbf{w}) = \sum_{j=1}^{n_c} \alpha_j \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right). \quad (41)$$

The parameter vector \mathbf{w} contains all the RBF weights α_j , widths $\tilde{\sigma}_j$ and centers \mathbf{c}_j . The dimension

of \mathbf{w} is therefore $N_p = n_c \times (M + 2)$. It should be emphasized that other neural networks, such as the multilayer perceptron or the polynomial kernel function network of the form

$$y_{\text{Pol}}(k) = f_{\text{Pol}}(\mathbf{r}(k); \mathbf{w}) = \sum_{j=1}^{n_p} \alpha_j (\mathbf{c}_j^T \mathbf{r}(k) + 1)^d \quad (42)$$

can similarly be used as multiuser detectors. A reason for using the RBF network (41) in this study is that we would like to investigate whether the NLBER algorithm can achieve the optimal Bayesian performance when the form and size of the detector is similar to that of the Bayesian detector.

To implement an adaptive algorithm, such as the LMS or NLBER, the derivatives of the detector with respect to the detector parameters are required. For the RBF network (41), these derivatives can readily be calculated:

$$\left. \begin{aligned} \frac{\partial f_{\text{RBF}}}{\partial \alpha_j} &= \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \\ \frac{\partial f_{\text{RBF}}}{\partial \tilde{\sigma}_j} &= \alpha_j \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j^2} \\ \frac{\partial f_{\text{RBF}}}{\partial \mathbf{c}_j} &= 2\alpha_j \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \frac{\mathbf{r}(k) - \mathbf{c}_j}{\tilde{\sigma}_j} \end{aligned} \right\} \quad (43)$$

$1 \leq j \leq n_c.$

Both the LMS and NLBER algorithms are used to train the RBF network detector with the adaptive gain μ given in the form $\mu_k = \mu_0 k^{-1/4}$, where μ_0 is an appropriately chosen constant. For the NLBER algorithm, the value of $\tilde{\rho}^2$ also needs to be determined. In the simulation study, μ_0 and $\tilde{\rho}^2$ are chosen empirically. Specifically, μ_0 is chosen for the LMS algorithm to ensure fast convergence speed and a small steady-state MSE, while μ_0 and $\tilde{\rho}^2$ are chosen for the NLBER algorithm to achieve fast convergence rate and a small steady-state BER.

3.4. Simulation study

In all simulations, the first $n_c/2$ data points that belong to the class +1 and the first $n_c/2$ data points that belong to the class -1 are used as

initial centers. The initial weights are set to $\pm\eta$ accordingly, where η is a small positive number. To take into account the influence of initial centers, the algorithm are run many times with different random initializations for different runs. All the RBF widths are initially set to $8\sigma_n^2$, assuming an estimated noise variance of $4\sigma_n^2$. Two kinds of BER are mentioned in the results, the true BER that is computed using Monte Carlo simulation with a sufficiently long test sequence and the estimated BER calculated using the approximate BER expression (14) with $g_j(\mathbf{w}) = \text{sgn}(b_i^{(j)})\tilde{y}_j/\tilde{\rho}$. The value of $\tilde{\rho}^2$ is fixed such that, for $k = 0$ and k equal to the final training sample, the estimated BERs agree with the true BERs. This $\tilde{\rho}$ should not be confused with the NLBER algorithm parameter $\tilde{\rho}$ used in adaptation. The estimated BER is used to illustrate the learning rate of an adaptive algorithm, providing an “estimate” for $P_E(\mathbf{w}(k))$, the true BER of the detector with the weight vector $\mathbf{w}(k)$, at each training sample k . An alternative would be to provide the true learning rate of the algorithm. This would require to calculate the true BER using Monte Carlo simulation at each training sample k , which is computationally too demanding, if not impossible.

Example 1. This was a very simple two-user system with 2 chips per symbol. The code sequences of the two users were $(-1, -1)$ and $(-1, +1)$, respectively, and the transfer function of the CIR at chip rate was $H(z) = 1.0 + 0.4z^{-1}$. The two users had equal signal power, that is, the user 1 signal to noise ratio SNR_1 was equal to SNR_2 of user 2. The set \mathcal{R} had 16 points, but only 12 were distinct. This example was chosen to demonstrate that multiuser detection can be considered as a classification problem, as a two-dimensional space can graphically be illustrated. The system was so set up to ensure that \mathcal{R}_+ and \mathcal{R}_- were linearly separable and hence a linear detector could work adequately. Fig. 2 displays \mathcal{R}_+ and \mathcal{R}_- for user 2 together with the two decision boundaries of the linear MBER and optimal Bayesian detectors for a $\text{SNR}_2 = 17$ dB (corresponding to a user 2 signal to interference plus noise ratio of $\text{SINR}_2 = -0.09$ dB). The BERs of the linear MBER and

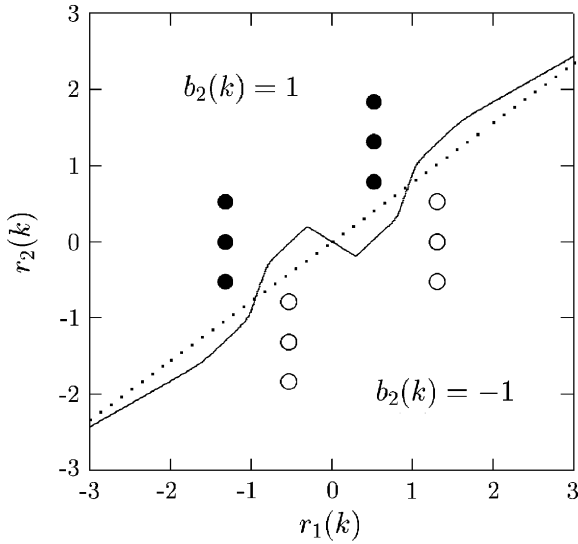


Fig. 2. The set of noise-free signal points and the two decision boundaries (dotted: linear MBER, solid: optimal) for user 2 of Example 1. $SNR_1 = SNR_2 = 17$ dB.

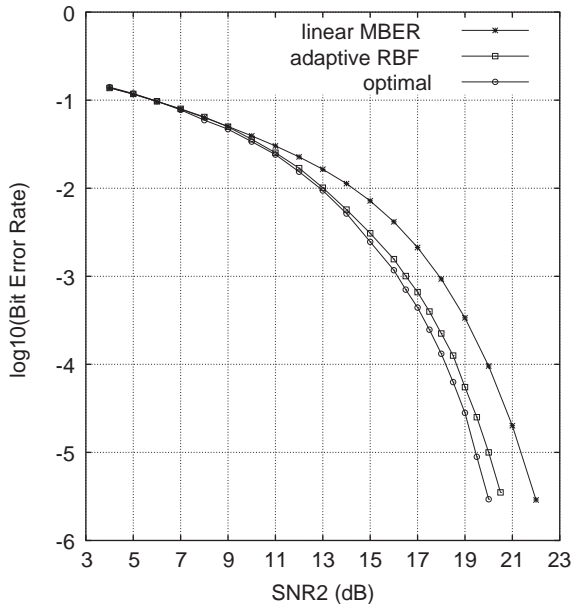


Fig. 3. Performance comparison of three detectors for user 2 of Example 1. $SNR_1 = SNR_2$. The adaptive RBF detector has 4 centers and is trained by the NLBER algorithm.

optimal Bayesian detectors are depicted in Fig. 3 for user 2 given the range of SNR_2 from 3 to 23 dB ($SINR_2$ from -1.76 to -0.02 dB).

Given $SNR_2 = 17$ dB, RBF detectors with 4 and 12 centers were trained by the LMS and NLBER algorithms, respectively. The NLBER had $\bar{\rho}^2 = 4\sigma_n^2$ for both detectors, and $\mu_0 = 0.2$ for the 4-center RBF and $\mu_0 = 0.25$ for the 12-center RBF; while the LMS had $\mu_0 = 0.1$ for the 4-center RBF and $\mu_0 = 0.3$ for the 12-center RBF. These values were found empirically to be appropriate. At each sample k , the estimated BER was calculated for a detector with $\mathbf{w}(k)$, and this resulted in the learning rates, plotted in Fig. 4, for the respective detectors, where the results were averaged over 100 runs. For the LMS training, the MSE for a detector with $\mathbf{w}(k)$ was also calculated using a block of 100 test samples, and this produced the learning rates in terms of the MSE given in Fig. 5, whereagain the results were averaged over 100 runs. The decision boundary of a typical 4-center RBF detector trained by the NLBER algorithm is compared with the optimal Bayesian boundary in Fig. 6.

The learning rates for the estimated BER given in Fig. 4 need some explanations. It was found that the estimated BERs of the 4-center RBF detector with the LMS training varied greatly for different runs. For some runs the estimated BERs were close to that obtained by the average NLBER training (0.001), but for other runs the estimated

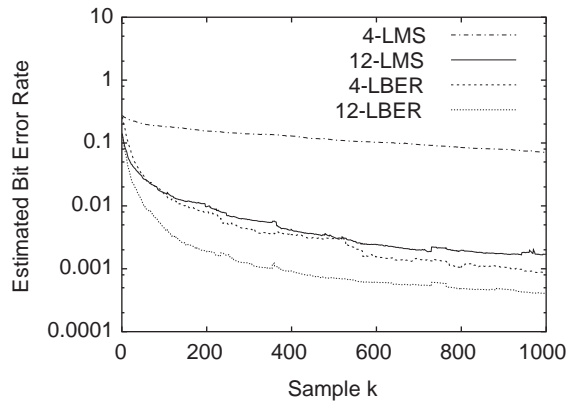


Fig. 4. Learning curves in terms of the estimated BER for user 2 of Example 1. $SNR_1 = SNR_2 = 17$ dB. The results are averaged over 100 runs. 4-LMS: the 4-center RBF trained by the LMS with $\mu_0 = 0.1$, 12-LMS: the 12-center RBF trained by the LMS with $\mu_0 = 0.3$, 4-LBER: the 4-center RBF trained by the NLBER with $\mu_0 = 0.2$ and $\bar{\rho}^2 = 4\sigma_n^2$, and 12-LBER: the 12-center RBF trained by the NLBER with $\mu_0 = 0.25$ and $\bar{\rho}^2 = 4\sigma_n^2$.

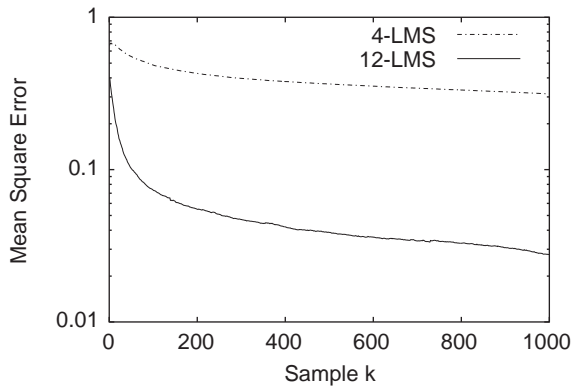


Fig. 5. Learning curves in terms of the MSE for user 2 of Example 1. $SNR_1 = SNR_2 = 17$ dB. The results are averaged over 100 runs. 4-LMS: the 4-center RBF trained by the LMS with $\mu_0 = 0.1$, and 12-LMS: the 12-center RBF trained by the LMS with $\mu_0 = 0.3$.

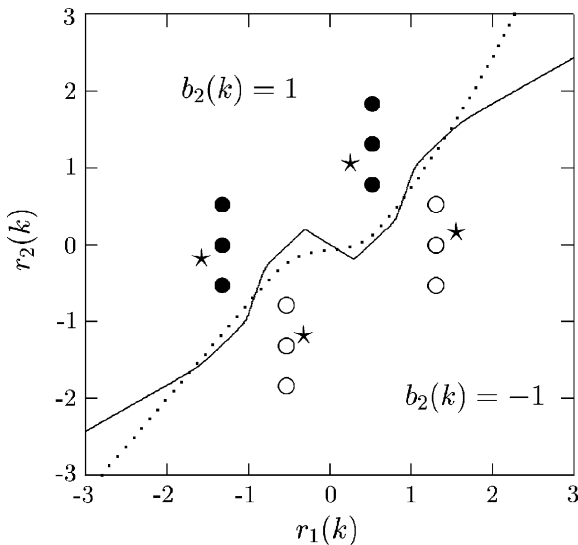


Fig. 6. Comparison of two decision boundaries (dotted: adaptive RBF detector, solid: optimal) for user 2 of Example 1. $SNR_1 = SNR_2 = 17$ dB. The adaptive RBF detector has 4 centers and is trained by the NLBER algorithm. The stars indicate the final center positions.

BERs converged to 0.5. Examining the resulting RBF detectors for the latter case, it was seen that the 4 centers all converged to near the origin with a symmetric configuration. This is not surprising, since this configuration can correspond to a small

MSE and is consistent with the LMS criterion. In fact, there was on average about 4 dB reduction in the MSE for the 4-center RBF detector trained by the LMS algorithm. Similar situations occurred for the 12-center RBF detector with the LMS training, and the averaged BER performance of the 12-center RBF trained by the LMS is poorer than that obtained for the 4-center detector trained by the NLBER. Note that this was more fundamental than “local minima problem”. In fact, examining the MSE learning rate for the 12-center RBF trained by the LMS, it was seen that different runs produced consistent performance and on average it had 11 dB reduction in the MSE. However, there was no direct link between the MSE value and the BER. In comparison, the NLBER training was found to produce consistent BER results in different runs, and the 12-center RBF detector with the NLBER training converged consistently to the optimal Bayesian performance, in terms of BER.

The influence of the algorithm parameter $\bar{\rho}^2$ on the performance of the NLBER algorithm was also investigated. Fig. 7 shows the (true) BERs of the 4-center RBF detector after the NLBER training with a range of $\bar{\rho}^2$, where it can be seen that the algorithm performance is not overly sensitive to $\bar{\rho}^2$ over a large range of values. The (true) BERs of the 4-center RBF detector after the NLBER training are depicted in Fig. 3. The (true)

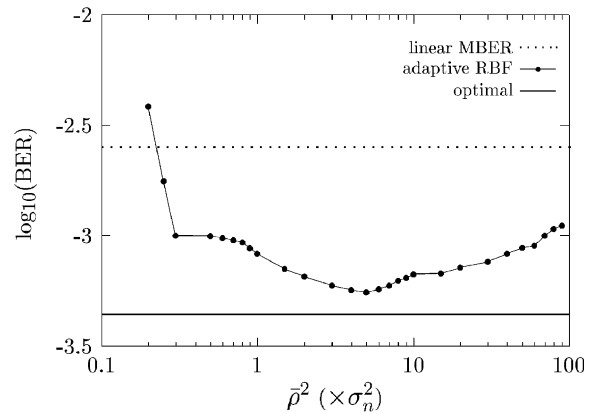


Fig. 7. Influence of $\bar{\rho}^2$ to the performance of the NLBER algorithm. User 2 of Example 1 with $SNR_1 = SNR_2 = 17$ dB. The adaptive RBF detector has 4 centers with $\mu_0 = 0.3$.

BERs of the 12-center RBF detector after the NLBER training are not shown here, as they are indistinguishable from the optimal performance. The 4-center RBF detector trained by the LMS did not work as the (true) BERs produced were often 50%, even though the algorithm converged well in the MSE. The (true) BERs of the 12-center RBF detector trained by the LMS algorithm, not shown here, were not much better than those of the linear MBER detector.

Example 2. This was a 3-user system with 8 chips per symbol. The code sequences for the three users were $(+1, +1, +1, +1, -1, -1, -1, -1)$, $(+1, -1, +1, -1, -1, +1, -1, +1)$ and $(+1, -1, -1, +1, -1, +1, +1, -1)$, respectively, and the transfer function of the CIR at chip rate was $H(z) = 0.8 + 0.6z^{-1} + 0.5z^{-2}$. The three users had equal signal power. Again a linear separable situation was simulated. The detector for user 3 was considered, and the BERs of the linear MBER and optimal detectors are displayed in Fig. 8 for the range of SNR_3 from 0 to 25 dB (SINR_3 from -4.77 to -3.02 dB). The noise-free state set \mathcal{R} had 64 points. Given $\text{SNR}_3 = 15$ dB ($\text{SINR}_3 = -3.08$ dB), RBF detec-

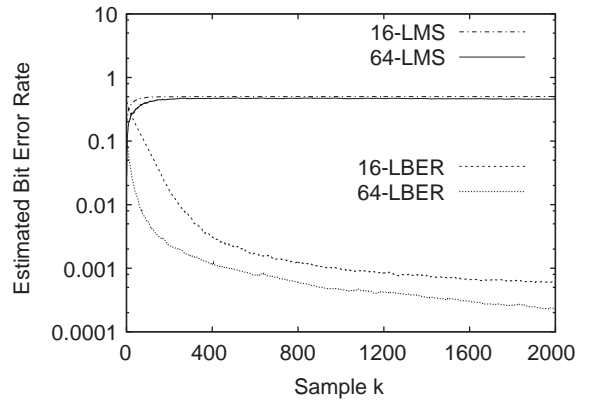


Fig. 9. Learning curves in terms of the estimated BER for user 3 of Example 2. $\text{SNR}_i = 15$ dB, $1 \leq i \leq 3$. The results are averaged over 100 runs. 16-LMS: the 16-center RBF trained by the LMS with $\mu_0 = 0.2$, 64-LMS: the 64-center RBF trained by the LMS with $\mu_0 = 0.2$, 16-LBER: the 16-center RBF trained by the NLBER with $\mu_0 = 0.6$ and $\bar{\rho}^2 = 1000\sigma_n^2$, and 64-LBER: the 64-center RBF trained by the NLBER with $\mu_0 = 0.1$ and $\bar{\rho}^2 = 50\sigma_n^2$.

tors with 16 and 64 centers were trained by the LMS and NLBER algorithms, respectively. The NLBER had $\bar{\rho}^2 = 1000\sigma_n^2$ and $\mu_0 = 0.6$ for the 16-center RBF, and $\bar{\rho}^2 = 50\sigma_n^2$ and $\mu_0 = 0.1$ for the 64-center RBF; while the LMS had $\mu_0 = 0.2$ for the both RBF detectors. These values were found empirically to be appropriate. The learning rates in terms of the estimated BER are plotted in Fig. 9 for the respective detectors, where the results were averaged over 100 runs. For the LMS training, the MSE convergence performance, averaged over 100 runs, are given in Fig. 10.

The NLBER algorithm produced consistent results and, in particular, the 64-center detector was able to achieve the optimal performance in terms of BER. For the LMS training, the algorithm converged very well in the MSE and there was an almost 30 dB reduction in the MSE, as can be seen in Fig. 10. However, the BERs of the two detectors trained by the LMS algorithm both approached to 0.5! In fact, averagely, the initial 16-center RBF detector had a BER=0.2 and the initial 64-center RBF detector had a BER=0.008. Yet, after training using the LMS, both yielded almost 1 in 2 errors (not much better than a random guess). This clearly illustrates the fact that

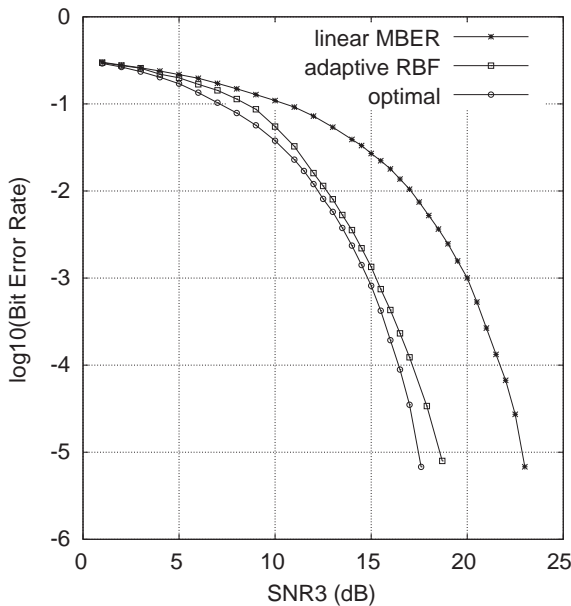


Fig. 8. Performance comparison of three detectors for user 3 of Example 2. SNR_i , $1 \leq i \leq 3$, are identical. The adaptive RBF detector has 16 centers and is trained by the NLBER algorithm.

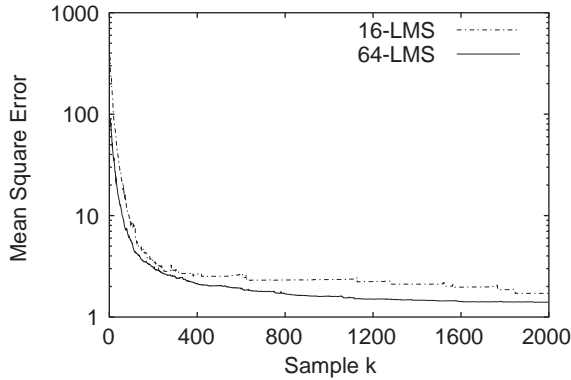


Fig. 10. Learning curves in terms of the MSE for user 3 of Example 2. $\text{SNR}_i = 15 \text{ dB}$, $1 \leq i \leq 3$. The results are averaged over 100 runs. 16-LMS: the 16-center RBF trained by the LMS with $\mu_0 = 0.2$, and 64-LMS: the 64-center RBF trained by the LMS with $\mu_0 = 0.2$.

a small MSE is not related to a small BER. As far as the LMS algorithm is concerned, it does a good job in what it supposes to do: getting the MSE down. The true BERs of the 16-center RBF detector after the NLBER training are compared with the optimal performance in Fig. 8, where it can be seen that its performance is very close to the optimal Bayesian detector of 64 states. The true BERs of the 64-center RBF detector trained by the NLBER, not depicted here, are indistinguishable from the optimal performance.

Example 3. The system had 4 equal power users with 8 chips per symbol. The code sequences for the four users were $(+1, +1, +1, +1, -1, -1, -1, -1)$, $(+1, -1, +1, -1, -1, +1, -1, +1)$, $(+1, +1, -1, -1, -1, -1, +1, +1)$ and $(+1, -1, -1, +1, -1, +1, +1, -1)$, respectively, and the transfer function of the CIR at chip rate was $H(z) = 0.4 + 0.7z^{-1} + 0.4z^{-2}$. The detector for user 2 was considered. For user 2, $\mathcal{R}^{(+)}$ and $\mathcal{R}^{(-)}$ are almost linearly inseparable, and a linear detector has a relatively poor BER performance at low SNRs, as is shown in Fig. 11. The BERs of the optimal Bayesian detector is also shown in Fig. 11, where the range of SNR_2 from 10 to 30 dB corresponds to the SINR_2 from -4.91 to -4.77 dB. Note that in this example the number of channel states $N_b = 256$, and the Bayesian detector is computationally very

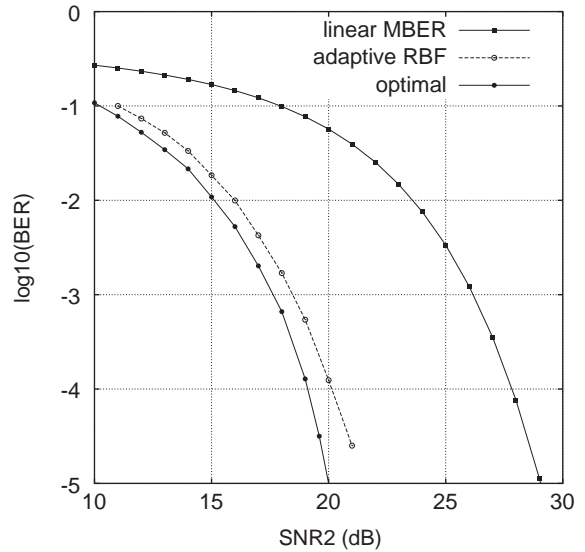


Fig. 11. Performance comparison of three detectors for user 2 of Example 3. SNR_i , $1 \leq i \leq 4$, are identical. The adaptive RBF detector has 64 centers and is trained by the NLBER algorithm.

expensive. The performance of the 64-center RBF detector trained by the NLBER algorithm is depicted in Fig. 11. It can be seen that the performance of this NLBER RBF detector is very close to the full optimal Bayesian performance. In the simulation it was again observed that the same 64-center RBF detector under the identical conditions but trained by the LMS algorithm, although converged well in the MSE, often resulted in BERs not much better than those of the linear MBER detector.

4. Conclusions

Adaptive training based on the MER criterion has been considered for a class of neural network classifiers that includes nonlinear equalizers and multiuser detectors. A main contribution of this research has been the derivation of an adaptive near MER algorithm called the NLER for this kind of applications. In the context of channel equalization and multiuser detection with binary modulation schemes, this adaptive algorithm has been referred to as the NLBER. Our approach has

been motivated from a kernel density estimation of the error rate as a smooth function of the training data and an adoption of stochastic gradient of the estimated error probability. This adaptive algorithm has been applied to downlink multiuser detection in CDMA communication systems using a RBF network. Simulation results have demonstrated that the NLBER algorithm performs consistently and the algorithm has a good convergence speed. A small-size RBF detector trained by the NLBER algorithm can closely approximate the optimal Bayesian detector. When the size of the RBF detector is similar to the Bayesian detector, the optimal performance can be achieved. The results also demonstrate that the standard adaptive algorithm, the LMS, may not be very relevant for training neural network classifiers, as the underlying criterion of the LMS is the MSE not the error probability.

References

- [1] S. Chen, G.J. Gibson, C.F.N. Cowan, P.M. Grant, Recursive prediction error algorithm for training multi-layer perceptrons, in: Proceedings of the IEEE Colloquium Adaptive Algorithms for Signal Estimation and Control, Edinburgh, Scotland, 1989, pp. 10/1–10/7.
- [2] S. Chen, G.J. Gibson, C.F.N. Cowan, Adaptive channel equalization using a polynomial-perceptron structure, IEE Proc. Part I 137 (5) (1990) 257–264.
- [3] S. Chen, G.J. Gibson, C.F.N. Cowan, P.M. Grant, Adaptive equalization of finite non-linear channels using multilayer perceptrons, Signal Processing 20 (2) (1990) 107–119.
- [4] G.J. Gibson, S. Siu, S. Chen, C.F.N. Cowan, P.M. Grant, The application of nonlinear architectures to adaptive channel equalisation, in: Proceedings of ICC'90, Atlanta, USA, 1990, pp. 312.8.1–312.8.5.
- [5] S. Siu, G.J. Gibson, C.F.N. Cowan, Decision feedback equalisation using neural network structures and performance comparison with the standard architecture, IEE Proc. Part I 137 (4) (1990) 221–225.
- [6] S. Chen, G.J. Gibson, C.F.N. Cowan, P.M. Grant, Reconstruction of binary signals using an adaptive radial-basis-function equalizer, Signal Processing 22 (1) (1991) 77–93.
- [7] G.J. Gibson, S. Siu, C.F.N. Cowan, The application of nonlinear structures to the reconstruction of binary signals, IEEE Trans. Signal Processing 39 (8) (1991) 1877–1884.
- [8] B. Aazhang, B.P. Paris, G.C. Orsak, Neural networks for multiuser detection in code-division multiple-access communications, IEEE Trans. Comm. 40 (7) (1992) 1212–1222.
- [9] S. Chen, S. McLaughlin, B. Mulgrew, Complex-valued radial basis function network, Part II: application to digital communications channel equalisation, Signal Processing 36 (1994) 175–188.
- [10] U. Mitra, H.V. Poor, Neural network techniques for adaptive multiuser demodulation, IEEE J. Selected Areas Comm. 12 (9) (1994) 1460–1470.
- [11] Z.J. Xiang, G.G. Bi, T. Le-Ngoc, Polynomial perceptrons and their applications to fading channel equalization and co-channel interference suppression, IEEE Trans. Signal Process. 42 (1994) 2470–2480.
- [12] I. Cha, S.A. Kassam, Channel equalization using adaptive complex radial basis function networks, IEEE J. Selected Areas Comm. 13 (1) (1995) 122–131.
- [13] C.H. Chang, S. Siu, C.H. Wei, A polynomial-perceptron based decision feedback equalizer with a robust learning algorithm, Signal Processing 47 (1995) 145–158.
- [14] D.G.M. Cruickshank, Radial basis function receivers for DS-CDMA, Electron. Lett. 32 (3) (1996) 188–190.
- [15] R. Tanner, D.G.M. Cruickshank, Volterra based receivers for DS-CDMA, in: Proceedings of the 8th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 3, September, 1997, pp. 1166–1170.
- [16] D.A. Pados, P. Papantoni-Kazakos, New nonleast-squares neural network learning algorithms for hypothesis testing, IEEE Trans. Neural Networks 6 (3) (1995) 596–609.
- [17] E. Shamash, K. Yao, On the structure and performance of a linear decision feedback equalizer based on the minimum error probability criterion, in: Proceedings of ICC'74, 1974, pp. 25F1–25F5.
- [18] S. Chen, E.S. Chng, B. Mulgrew, G.J. Gibson, Minimum-BER linear-combiner DFE, in: Proceedings of ICC'96, vol. 2, Dallas, Texas, 1996, pp. 1173–1177.
- [19] N.B. Mandayam, B. Aazhang, Gradient estimation for sensitivity analysis and adaptive multiuser interference rejection in code-division multi-access systems, IEEE Trans. Comm. 45 (7) (1997) 848–858.
- [20] C.C. Yeh, J.R. Barry, Approximate minimum bit-error rate equalization for binary signaling, in: Proceedings of ICC'97, vol. 2, Montreal, Canada, 1997, pp. 1095–1099.
- [21] S. Chen, B. Mulgrew, E.S. Chng, G.J. Gibson, Space translation properties and the minimum-BER linear-combiner DFE, IEE Proc. Comm. 145 (5) (1998) 316–322.
- [22] C.C. Yeh, R.R. Lopes, J.R. Barry, Approximate minimum bit-error rate multiuser detection, in: Proceedings of Globecom'98, Sydney, Australia, November 1998, pp. 3590–3595.
- [23] S. Chen, B. Mulgrew, The minimum-SER linear-combiner decision feedback equalizer, IEE Proc. Comm. 146 (6) (1999) 347–353.
- [24] X.F. Wang, W.S. Lu, A. Antoniou, Constrained minimum-BER multiuser detection, in: Proceedings of ICASSP'99, vol. 5, Phoenix, USA, May 14–18, 1999, pp. 2603–2606.

- [25] I.N. Psaromiligkos, S.N. Batalama, D.A. Pados, On adaptive minimum probability of error linear filter receivers for DS-CDMA channels, *IEEE Trans. Comm.* 47 (7) (1999) 1092–1102.
- [26] B. Mulgrew, S. Chen, Stochastic gradient minimum-BER decision feedback equalisers, in: *Proceedings of IEEE Symposium on Adaptive Systems for Signal Processing, Communication and Control*, Lake Louise, Alberta, Canada, October 1–4, 2000, pp. 93–98.
- [27] C.C. Yeh, J.R. Barry, Adaptive minimum bit-error rate equalization for binary signaling, *IEEE Trans. Comm.* 48 (7) (2000) 1226–1235.
- [28] S. Chen, A.K. Samingan, B. Mulgrew, L. Hanzo, Adaptive minimum-BER linear multiuser detection for DS-CDMA signals in multipath channels, *IEEE Trans. Signal Process.* 49 (6) (2001) 1240–1247.
- [29] B. Mulgrew, S. Chen, Adaptive minimum-BER decision feedback equalisers for binary signalling, *Signal Processing* 81 (7) (2001) 1478–1489.
- [30] D.J.C. MacKay, Bayesian interpolation, *Neural Computation* 4 (3) (1992) 415–447.
- [31] D.J.C. MacKay, The evidence framework applied to classification networks, *Neural Computation* 4 (1992) 720–736.
- [32] X. Wang, R. Chen, Adaptive Bayesian multiuser detection for synchronous CDMA with Gaussian and impulsive noise, *IEEE Trans. Signal Process.* 47 (7) (2000) 2013–2028.
- [33] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.
- [34] M.E. Tipping, Sparse Bayesian learning and the relevance vector machine, *J. Mac. Learning Res.* 1 (2001) 211–244.
- [35] E. Parzen, On estimation of a probability density function and mode, *Ann. Math. Statist.* 33 (1962) 1066–1076.
- [36] B.W. Silverman, *Density Estimation*, Chapman & Hall, London, 1996.
- [37] A.W. Bowman, A. Azzalini, *Applied Smoothing Techniques for Data Analysis*, Oxford University Press, Oxford, UK, 1997.
- [38] R. Prasad, *CDMA for Wireless Personal Communications*, Artech House, Inc., 1996.
- [39] S. Verdú, *Multiuser Detection*, Cambridge University Press, Cambridge, UK, 1998.
- [40] M.S. Bazaraa, H.D. Sherali, C.M. Shetty, *Nonlinear Programming: Theory and Algorithms*, Wiley, New York, 1993.
- [41] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, NJ, 1993.
- [42] H.V. Poor, S. Verdú, Probability of error in MMSE multiuser detection, *IEEE Trans. Inform. Theory* 43 (3) (1997) 858–871.
- [43] B. Mulgrew, Nonlinear signal processing for adaptive equalisation and multi-user detection, in: *Proceedings of EUSIPCO 98*, Rhodes, Greece, September 1998, pp. 537–544.
- [44] Z. Xie, R.T. Short, C.K. Rushforth, A family of suboptimum detectors for coherent multiuser communications, *IEEE J. Selected Areas Comm.* 8 (4) (1990) 683–690.
- [45] U. Madhow, M.L. Honig, MMSE interference suppression for direct-sequence spread-spectrum CDMA, *IEEE Trans. Comm.* 42 (12) (1994) 3178–3188.
- [46] S.L. Miller, An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection, *IEEE Trans. Comm.* 43 (2/3/4) (1995) 1746–1755.
- [47] G. Woodward, B.S. Vucetic, Adaptive detection for DS-CDMA, *Proc. IEEE* 86 (7) (1998) 1413–1434.
- [48] S. Chen, A.K. Samingan, L. Hanzo, Support vector machine multiuser receiver for DS-CDMA signals in multipath channels, *IEEE Trans. Neural Networks* 12 (3) (2001) 604–611.
- [49] S. Chen, B. Mulgrew, S. McLaughlin, Adaptive Bayesian equaliser with decision feedback, *IEEE Trans. Signal Process.* 41 (9) (1993) 2918–2927.
- [50] S. Chen, S. McLaughlin, B. Mulgrew, P.M. Grant, Adaptive Bayesian decision feedback equaliser for dispersive mobile radio channels, *IEEE Trans. Comm.* 43 (5) (1995) 1937–1946.