Least Bit Error Rate Adaptive Nonlinear Equalizers for Binary Signalling

S. Chen[†0], *B. Mulgrew*[‡] *and L. Hanzo*[†]

[†] Department of Electronics and Computer Science
University of Southampton, Highfield
Southampton SO17 1BJ, U.K.

[‡] Department of Electronics and Electrical Engineering
University of Edinburgh, King's Buildings
Edinburgh EH9 3JL, U.K.

Abstract

The paper considers the problem of constructing adaptive minimum bit error rate (MBER) neural network equalizers for binary signalling. Motivated from a kernel density estimation of the bit error rate (BER) as a smooth function of training data, a stochastic gradient algorithm called the least bit error rate (LBER) is developed for adaptive nonlinear equalizers. This LBER algorithm is applied to adaptive training of a radial basis function (RBF) equalizer in a channel intersymbol interference (ISI) plus co-channel interference setting. Simulation study shows that the proposed algorithm has a good convergence speed and a small-size RBF equalizer trained by the LBER can closely approximate the performance of the optimal Bayesian equalizer. The results also demonstrates that the standard adaptive algorithm, the least mean square (LMS), performs poorly for neural network equalizers, due to the reason that the minimum mean square error (MMSE) is irrelevant to the equalization goal.

Index Terms: Mean square error, bit error rate, least mean square, least bit error rate, adaptive equalizers, neural networks

[0]Contact author. Tel./Fax: +44 (0)23 8095 6660/4508; E-mail: sqc@ecs.soton.ac.uk.

1 Introduction

Equalization technique plays a crucial role in combating distortion and interference in modern communication systems. The topic of equalization is well researched and a variety of solutions are available [1]–[7]. Recently, neural networks have been used as adaptive nonlinear equalizers [8]–[17]. Typically, adjusting an equalizer's parameters in a sample-by-sample fashion is required in practical applications to meet real-time computational constraints, and adaptive training of neural network equalizers is usually carried out using some stochastic gradient algorithm that tries to minimize the mean square error (MSE). This kind of stochastic gradient adaptive algorithm is classically referred to as the LMS. It is a well-known fact that adaptive training of neural network equalizers often encounters difficulties. Such difficulties are typically attributed to "local minima" in the literature. In this study, we ask a more fundamental question: whether the underlying criterion adopted in these adaptive nonlinear equalizers, the MMSE, is a relevant one.

For linear equalizers, there is at least a partial relationship between the MSE and the BER. A small MSE is usually associated with a small BER. However, even in the linear case, the MMSE solution in general is not the MBER solution, and it is well-known that the BER gap between the linear MMSE solution and linear MBER one can be large in certain situations [18]-[25]. Since the BER is the true performance indicator, recent research has derived some adaptive linear MBER equalizers [20],[24]. For nonlinear equalizers, it can be argued that there exists no relationship between the MSE and the BER, and the MMSE solution may not correspond to a small BER. Consider, for example, the maximum *a posterior* probability or optimal Bayesian equalizer [7] for which the MMSE criterion is completely irrelevant[1]. Since the MMSE is a "wrong" criterion, training a neural network equalizer by the LMS algorithm may lead to a poor BER performance, and this may actually be a cause of difficulties associated with adaptive nonlinear equalizers.

In this paper, we develop a stochastic gradient adaptive MBER algorithm for nonlinear equalizers with binary signalling. It should be emphasized that adaptive MBER training can in theory be achieved by adjusting the equalizer's parameters only when an error

[1]Multiplying the Bayesian equalizer by a positive constant, it remains the optimal Bayesian equalizer while the MSE value can become very large

occurs. Since the BER is typically very small, such a strategy would require an extremely long training sequence and is therefore impractical. We adopt an approach similar to the one used in developing an adaptive MBER linear equalizer [24],[25], namely using a kernel density estimation to approximate the BER from training data and to derive a stochastic gradient adaptive algorithm. The resulting algorithm is called the LBER for its link to the MBER criterion, in a manner analogous to the LMS algorithm for the MMSE criterion. This LBER algorithm is tested with an equalization application in the presence of channel ISI, additive Gaussian white noise and co-channel interference, where a RBF network is trained as an adaptive equalizer. Simulation results obtained show that the LBER algorithm achieves consistent performance and has a reasonable convergence speed. A small-size RBF network trained by the LBER algorithm can closely approximate the optimal Bayesian performance. The simulation study also confirms that a neural network equalizer trained by the LMS algorithm, although converging well in the MSE, can produce a poor BER performance.

2 Equalization problem

It is assumed that the channel is modelled as a finite impulse response filter with an additive noise source [26]. For notational simplification, it is further assumed that there exists only one relatively strong co-channel interference. Thus, the received signal at sample k is given by

$$r(k) = \bar{r}(k) + n(k) = \bar{r}_0(k) + \bar{r}_1(k) + n(k) = \sum_{i=0}^{n_0-1} a_{0,i} b_0(k-i) + \sum_{i=0}^{n_1-1} a_{1,i} b_1(k-i) + n(k) \, , \quad (1)$$

where $n(k)$ is a Gaussian white noise with variance $E[n^2(k)] = \sigma_n^2$; $\bar{r}(k)$, $\bar{r}_0(k)$ and $\bar{r}_1(k)$ are the noise-free received signal, desired signal and interfering signal, respectively; $a_{0,i}$, $0 \le i \le n_0 - 1$, are the channel taps and $a_{1,i}$, $0 \le i \le n_1 - 1$, are the co-channel taps; the desired and interfering data $b_0(k)$ and $b_1(k)$ are binary taking value from the set $\{\pm 1\}$, and they are uncorrelated. Let $E[b_0^2(k)] = \sigma_0^2$ and $E[b_1^2(k)] = \sigma_1^2$. The signal to interference ratio of the system is defined by

$$\text{SIR} = \frac{\sigma_0^2 \sum_{i=0}^{n_0-1} a_{0,i}^2}{\sigma_1^2 \sum_{i=0}^{n_1-1} a_{1,i}^2} \, , \quad (2)$$

the signal to noise ratio is defined by

$$\mathrm{SNR} = \frac{\sigma_0^2 \sum_{i=0}^{n_0-1} a_{0,i}^2}{\sigma_n^2}, \tag{3}$$

and the signal to interference plus noise ratio is defined by

$$\mathrm{SINR} = \frac{\sigma_0^2 \sum_{i=0}^{n_0-1} a_{0,i}^2}{\sigma_n^2 + \sigma_1^2 \sum_{i=0}^{n_1-1} a_{1,i}^2}. \tag{4}$$

The equalizer considered in this study has a sample-decision finite-memory structure, and it uses the information contained in the received signal vector

$$\mathbf{r}(k) = [r(k)\ r(k-1)\ \cdots\ r(k-m+1)]^T \tag{5}$$

to produce an estimate of $b_0(k-d)$, where m and d are the equalizer order and delay, respectively. Specifically, the equalizer is defined by:

$$\hat{b}_0(k-d) = \mathrm{sgn}(y(k)) \quad \text{with} \quad y(k) = f(\mathbf{r}(k); \mathbf{w}), \tag{6}$$

where $f(\cdot;\cdot)$ denotes the equalizer map, and $\mathbf{w}$ consists of all the (adjustable) parameters of the equalizer. Such an equalizer has finite "states", since

$$\mathbf{b}_0(k) = [b_0(k)\ b_0(k-1)\ \cdots\ b_0(k-n_0-m+2)]^T \tag{7}$$

has $N_0 = 2^{m+n_0-1}$ combinations, denoted as $\mathbf{b}_{0,j}$, $1 \le j \le N_0$, and

$$\mathbf{b}_1(k) = [b_1(k)\ b_1(k-1)\ \cdots\ b_1(k-n_1-m+2)]^T \tag{8}$$

has $N_1 = 2^{m+n_1-1}$ combinations, denoted as $\mathbf{b}_{1,l}$, $1 \le l \le N_1$. From the system model (1), the received signal vector can be written as:

$$\mathbf{r}(k) = \bar{\mathbf{r}}(k) + \mathbf{n}(k) = \bar{\mathbf{r}}_0(k) + \bar{\mathbf{r}}_1(k) + \mathbf{n}(k). \tag{9}$$

It is obvious that $\bar{\mathbf{r}}(k)$, $\bar{\mathbf{r}}_0(k)$ and $\bar{\mathbf{r}}_1(k)$ all have finite states, that is,

$$\bar{\mathbf{r}}_0(k) \in \mathcal{R}_0 \triangleq \{\mathbf{r}_{0,j},\ 1 \le j \le N_0\}, \tag{10}$$

$$\bar{\mathbf{r}}_1(k) \in \mathcal{R}_1 \triangleq \{\mathbf{r}_{1,l},\ 1 \le l \le N_1\}, \tag{11}$$

and

$$\bar{\mathbf{r}}(k) \in \mathcal{R} \triangleq \{\mathbf{r}_{0,j} + \mathbf{r}_{1,l},\ 1 \le j \le N_0 \text{ and } 1 \le l \le N_1\}. \tag{12}$$

Notice that the set $\mathcal{R}$ contains $N_r = N_0 \times N_1$ states, that is,

$$\bar{\mathbf{r}}(k) \in \mathcal{R} \triangleq \{\mathbf{r}_i,\ 1 \le i \le N_r\}\,. \tag{13}$$

Denote the dth element of $\mathbf{b}_{0,j}$ as $b_{0,j}^{(d)}$. Since $b_{0,j}^{(d)} \in \{\pm 1\}$, the sets $\mathcal{R}_0$ and $\mathcal{R}$ can each be divided into two subsets $\mathcal{R}_0^{(\pm)}$ and $\mathcal{R}^{(\pm)}$, depending on the value of $b_{0,j}^{(d)}$. Thus $b_{0,j}^{(d)}$ is the "class" label for $\mathbf{r}_{0,j} \in \mathcal{R}_0$ or $\mathbf{r}_{0,j} + \mathbf{r}_{1,l} \in \mathcal{R}$. For notational convenience, define

$$b_i^{(d)} = b_{0,j}^{(d)}, \quad \text{for} \quad 1 \le i \le N_r \quad \text{with} \quad j = 1 + \left\lfloor \frac{i-1}{N_1} \right\rfloor\,, \tag{14}$$

where $\lfloor \frac{\cdot}{\cdot} \rfloor$ denotes the integer part of division. It can be seen that $b_i^{(d)}$ serves as the class label for $\mathbf{r}_i \in \mathcal{R}$.

If the channel and co-channel are known, applying the maximum *a posterior* probability principle leads to the following optimal Bayesian equalizer [27]:

$$\begin{aligned} y_B(k) = f_B(\mathbf{r}(k);\mathbf{w}) &= \sum_{\mathbf{r}_{0,j} \in \mathcal{R}_0} \sum_{\mathbf{r}_{1,l} \in \mathcal{R}_1} \frac{b_{0,j}^{(d)}}{(2\pi\sigma_n^2)^{m/2} N_0 N_1} \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{r}_{0,j} - \mathbf{r}_{1,l}\|^2}{2\sigma_n^2}\right) \\ &= \sum_{\mathbf{r}_i \in \mathcal{R}} \frac{b_i^{(d)}}{(2\pi\sigma_n^2)^{m/2} N_r} \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{r}_i\|^2}{2\sigma_n^2}\right)\,, \end{aligned} \tag{15}$$

where equiprobable states in $\mathcal{R}_0$ and $\mathcal{R}_1$ have been assumed. It should be pointed out that, multiplying $y_B(k)$ by a positive constant, it remains the optimal Bayesian solution, and it is irrelevant to consider the MSE value of this optimal equalizer. It is also clear that this Bayesian solution has a very high computational complexity. An often used equalizer is the linear one:

$$y_L(k) = f_L(\mathbf{r}(k);\mathbf{w}) = \mathbf{w}^T \mathbf{r}(k)\,, \tag{16}$$

The most popular solution for the linear equalizer is the MMSE solution, which can readily be implemented adaptively using the LMS algorithm.

3 Derivation of the LBER algorithm

Consider the generic equalizer (6) where the equalizer map f is realized, for example, by a neural network. Classically, such a nonlinear equalizer is trained by adjusting $\mathbf{w}$ so that the MSE, $E[(b_0(k-d)-y(k))^2]$, is minimized. Typically, this is implemented adaptively using the LMS algorithm, which has a very simple form:

$$\left.\begin{aligned} y(k) &= f(\mathbf{r}(k);\mathbf{w}(k-1))\,, \\ \mathbf{w}(k) &= \mathbf{w}(k-1)+\mu(b_0(k-d)-y(k))\tfrac{\partial f(\mathbf{r}(k);\mathbf{w}(k-1))}{\partial \mathbf{w}}\,, \end{aligned}\right\} \tag{17}$$

where μ is an adaptive gain. However, minimizing the MSE does not necessarily produce a small BER. A main objective of this study is to derive an adaptive algorithm for the nonlinear equalizer (6) based on the MBER criterion.

3.1 Approximate bit error rate expression

The error probability of the equalizer (6) is

$$P_E(\mathbf{w}) = \text{Prob}\{\text{sgn}(b_0(k-d))y(k) < 0\}\,. \tag{18}$$

Define the signed decision variable

$$y_s(k) = \text{sgn}(b_0(k-d))y(k)\,, \tag{19}$$

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

$$P_E(\mathbf{w}) = \int_{-\infty}^{0} p_y(y_s)\,dy_s\,. \tag{20}$$

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

$$y(k) = f(\bar{\mathbf{r}}(k)+\mathbf{n}(k);\mathbf{w}) \approx f(\bar{\mathbf{r}}(k);\mathbf{w}) + \left[\frac{\partial f(\bar{\mathbf{r}}(k);\mathbf{w})}{\partial \mathbf{r}}\right]^T \mathbf{n}(k) \tag{21}$$

or

$$y(k) \approx f(\bar{\mathbf{r}}(k);\mathbf{w}) + e(k) = \bar{y}(k) + e(k)\,, \tag{22}$$

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

$$\rho^2 = \mathrm{E}\left[\left[\frac{\partial f(\bar{\mathbf{r}}(k);\mathbf{w})}{\partial \mathbf{r}}\right]^T \mathbf{n}(k)\mathbf{n}^T(k)\frac{\partial f(\bar{\mathbf{r}}(k);\mathbf{w})}{\partial \mathbf{r}}\right] = \frac{\sigma_n^2}{N_r}\sum_{i=1}^{N_r}\left[\frac{\partial f(\mathbf{r}_i;\mathbf{w})}{\partial \mathbf{r}}\right]^T \frac{\partial f(\mathbf{r}_i;\mathbf{w})}{\partial \mathbf{r}} \tag{23}$$

with $\mathbf{r}_i \in \mathcal{R}$. Basically, the equalizer is approximated as an additive Gaussian noise model, with the "clean" signal $\bar{y}(k)$ taking values from the finite set

$$\bar{y}(k) \in \{y_i = f(\mathbf{r}_i;\mathbf{w}),\;\; 1 \le i \le N_r\}\,. \tag{24}$$

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

$$p_y(y_s) \approx \frac{1}{N_r\sqrt{2\pi}\rho}\sum_{i=1}^{N_r}\exp\left(-\frac{(y_s-\mathrm{sgn}(b_i^{(d)})y_i)^2}{2\rho^2}\right) \tag{25}$$

and the error probability of the equalizer is approximately

$$P_E(\mathbf{w}) \approx \frac{1}{N_r\sqrt{2\pi}}\sum_{i=1}^{N_r}\int_{g_i(\mathbf{w})}^{\infty}\exp\left(-\frac{x_i^2}{2}\right)\,dx_i = \frac{1}{N_r}\sum_{i=1}^{N_r}Q(g_i(\mathbf{w}))\,, \tag{26}$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty}\exp\left(-\frac{y^2}{2}\right)\,dy \tag{27}$$

and

$$g_i(\mathbf{w}) = \frac{\mathrm{sgn}(b_i^{(d)})y_i}{\rho} = \frac{\mathrm{sgn}(b_i^{(d)})f(\mathbf{r}_i;\mathbf{w})}{\rho}\,. \tag{28}$$

3.2 Approximate minimum bit error rate solution

If the channel and co-channel are known, an approximate MBER solution for the equalizer (6) can be obtained by minimizing the approximate BER expression (26). Noting the gradient of $P_E(\mathbf{w})$:

$$\nabla P_E(\mathbf{w}) \approx -\frac{1}{N_r\sqrt{2\pi}\rho}\sum_{i=1}^{N_r}\exp\left(-\frac{y_i^2}{2\rho^2}\right)\mathrm{sgn}(b_i^{(d)})\frac{\partial f(\mathbf{r}_i;\mathbf{w})}{\partial \mathbf{w}}\,, \tag{29}$$

the following iterative gradient algorithm can be used to arrive at an approximate MBER solution. Given an initial $\mathbf{w}(0)$, at lth iteration, the algorithm computes:

$$\left.\begin{aligned}
&y_i(l) = f(\mathbf{r}_i;\mathbf{w}(l-1)),\quad 1\le i\le N_r\,,\\
&\nabla P_E(\mathbf{w}(l)) = -\tfrac{1}{N_r\sqrt{2\pi}\rho}\textstyle\sum_{i=1}^{N_r}\exp\left(-\tfrac{y_i^2(l)}{2\rho^2}\right)\mathrm{sgn}(b_i^{(d)})\tfrac{\partial f(\mathbf{r}_i;\mathbf{w}(l-1))}{\partial \mathbf{w}}\,,\\
&\mathbf{w}(l) = \mathbf{w}(l-1) - \mu\nabla P_E(\mathbf{w}(l))\,.
\end{aligned}\right\} \tag{30}$$

Although the "variance" ρ^2 could iteratively be calculated by

$$\rho^2(l) = \frac{\sigma_n^2}{N_r}\sum_{i=1}^{N_r}\left[\frac{\partial f(\mathbf{r}_i;\mathbf{w}(l-1))}{\partial \mathbf{r}}\right]^T\frac{\partial f(\mathbf{r}_i;\mathbf{w}(l-1))}{\partial \mathbf{r}}\,, \tag{31}$$

for numerical and convergence considerations, it is preferred to fix ρ^2 to an appropriately chosen constant. Thus, ρ^2 is considered as an algorithm parameter that requires tuning.

Notice that if the equalizer (6) is linear, all the approximations done so far become exact, and the variance of $e(k)$ is $\rho^2 = \sigma_n^2\mathbf{w}^T\mathbf{w}$. In this case, we arrive at the exact MBER solution for the linear equalizer (16) (see [24],[25]).

3.3 Block-data gradient algorithm

In practice, the set $\mathcal{R}$ is unknown. The key to developing an effective adaptive algorithm is the p.d.f. $p_y(y_s)$ of the signed decision variable $y_s(k)$. Kernel density estimation is known to produced reliable p.d.f. estimates with short data records and in particular extremely natural in dealing with Gaussian mixtures [28],[29]. Given a block of K training samples $\{\mathbf{r}(k), b_0(k-d)\}$, a kernel density estimate of the p.d.f. is

$$\hat{p}_y(y_s) = \frac{1}{K\sqrt{2\pi}\rho}\sum_{k=1}^{K}\exp\left(-\frac{(y_s - \mathrm{sgn}(b_0(k-d))y(k))^2}{2\rho^2}\right) . \tag{32}$$

From the estimated error probability

$$\hat{P}_E(\mathbf{w}) = \int_{-\infty}^{0} \hat{p}_y(y_s)\, dy_s \, , \tag{33}$$

$\nabla\hat{P}_E(\mathbf{w})$ can be calculated

$$\nabla\hat{P}_E(\mathbf{w}) = -\frac{1}{K\sqrt{2\pi}\rho}\sum_{k=1}^{K}\exp\left(-\frac{y^2(k)}{2\rho^2}\right)\mathrm{sgn}(b_0(k-d))\frac{\partial f(\mathbf{r}(k);\mathbf{w})}{\partial \mathbf{w}} . \tag{34}$$

Thus a block adaptive gradient algorithm can be derived. At lth iteration, the algorithm computes:

$$\left.\begin{array}{l} y(k) = f(\mathbf{r}(k);\mathbf{w}(l-1)), \quad 1 \le k \le K \, , \\ \nabla\hat{P}_E(\mathbf{w}(l)) = -\frac{1}{K\sqrt{2\pi}\rho}\sum_{k=1}^{K}\exp\left(-\frac{y^2(k)}{2\rho^2}\right)\mathrm{sgn}(b_0(k-d))\frac{\partial f(\mathbf{r}(k);\mathbf{w}(l-1))}{\partial \mathbf{w}} \, , \\ \mathbf{w}(l) = \mathbf{w}(l-1) - \mu\nabla\hat{P}_E(\mathbf{w}(l)) \, . \end{array}\right\} \tag{35}$$

3.4 Stochastic gradient algorithm

Our real aim is to develop a stochastic gradient adaptive algorithm with sample-by-sample updating, in a similar manner to the LMS (17). 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:

$$\hat{p}_y(y_s, k) = \frac{1}{\sqrt{2\pi}\rho}\exp\left(-\frac{(y_s - \mathrm{sgn}(b_0(k-d))y(k))^2}{2\rho^2}\right) . \tag{36}$$

Using the instantaneous or stochastic gradient

$$\nabla\hat{P}_E(k;\mathbf{w}) = -\frac{1}{\sqrt{2\pi}\rho}\exp\left(-\frac{y^2(k)}{2\rho^2}\right)\mathrm{sgn}(b_0(k-d))\frac{\partial f(\mathbf{r}(k);\mathbf{w})}{\partial \mathbf{w}} \, , \tag{37}$$

a stochastic gradient algorithm is given by

$$\left.\begin{array}{l} y(k) = f(\mathbf{r}(k); \mathbf{w}(k-1))\,, \\ \mathbf{w}(k) = \mathbf{w}(k-1) + \frac{\mu}{\sqrt{2\pi}\rho} \exp\left(-\frac{y^2(k)}{2\rho^2}\right) \mathrm{sgn}(b_0(k-d)) \frac{\partial f(\mathbf{r}(k);\mathbf{w}(k-1))}{\partial \mathbf{w}}\,, \end{array}\right\} \tag{38}$$

assuming that an appropriate ρ has been chosen. The influence of ρ to the algorithm performance will be investigated in a simulation study. Following a similar reasoning to the LMS for the MMSE criterion, the algorithm (38) is called the LBER.

4 Simulation study

To test the above LBER algorithm for adaptive training of neural network equalizers, the RBF network equalizer of the form

$$y_{RBF}(k) = f_{RBF}(\mathbf{r}(k); \mathbf{w}) = \sum_{j=1}^{n_c} \alpha_j \exp\left(-\frac{\|\mathbf{r}(k) - \mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \tag{39}$$

is used. The equalizer parameter vector $\mathbf{w}$ thus contains all the RBF weights α_j, widths $\tilde{\sigma}_j$ and centers $\mathbf{c}_j$. The dimension of $\mathbf{w}$ is therefore $n_c \times (m+2)$. The derivatives of the equalizer output with respect to the equalizer's parameters in this case are given by:

$$\left.\begin{array}{l} \frac{\partial f_{RBF}}{\partial \alpha_j} = \exp\left(-\frac{\|\mathbf{r}(k)-\mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right)\,, \\ \frac{\partial f_{RBF}}{\partial \tilde{\sigma}_j} = \alpha_j \exp\left(-\frac{\|\mathbf{r}(k)-\mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \frac{\|\mathbf{r}(k)-\mathbf{c}_j\|^2}{\tilde{\sigma}_j^2}\,, \\ \frac{\partial f_{RBF}}{\partial \mathbf{c}_j} = 2\alpha_j \exp\left(-\frac{\|\mathbf{r}(k)-\mathbf{c}_j\|^2}{\tilde{\sigma}_j}\right) \frac{\mathbf{r}(k)-\mathbf{c}_j}{\tilde{\sigma}_j}\,, \end{array}\right\} \; 1 \le j \le n_c\,. \tag{40}$$

For a comparison purpose, both the LMS and LBER algorithms are used to train the RBF equalizer (39) with the adaptive gain μ given in the form of $\mu_k = \mu_0 k^{-1/4}$, where μ_0 is an appropriately chosen constant. For the LBER algorithm, the value of ρ^2 also needs to be determined.

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 $\pm\frac{1}{n_c(2\pi\sigma_n^2)^{m/2}}$ accordingly. All the widths are initially set to $8\sigma_n^2$. Two kinds of BER are used in the simulation investigation, 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 (26) with $g_i(\mathbf{w}) = \mathrm{sgn}(b_i^{(d)}) y_i/\tilde{\rho}$. The value of

$\tilde{\rho}$ is so chosen that the estimated BER agrees with the true one. This $\tilde{\rho}$ should not be confused with the algorithm parameter ρ used in the LBER. The estimated BER is used to illustrate the learning rate of an adaptive algorithm, since computing the true BER of the adaptive RBF equalizer at each sample k would be computationally too demanding.

Example 1. The transfer functions of the channel and co-channel were respectively $A_0(z) = 0.5 + 1.0z^{-1}$ and $A_1(z) = \lambda(1.0 + 0.5z^{-1})$. The value of λ was set to give an SIR = 12 dB. The equalizer order was chosen to be $m = 2$ and the decision delay $d = 1$. The sets $\mathcal{R}_0$ and $\mathcal{R}_1$ each had 8 points, and the set $\mathcal{R}$ had 64 states. Fig. 1 shows the sets $\mathcal{R}_0$ and $\mathcal{R}$, together with the decision boundaries of the linear MMSE and optimal Bayesian equalizers given SNR = 20 dB (SINR = 11.36 dB). Given this noise and interference condition, RBF equalizers with 4 and 6 centers were trained by the LMS and LBER algorithms, respectively. At each sample k, the estimated BER was calculated for an equalizer with $\mathbf{w}(k)$, and this resulted in the learning rates, plotted in Fig. 2, for the corresponding adaptive equalizers, where the results were averaged over 100 runs. The LBER algorithm had $\rho^2 = 20\sigma_n^2$ and $\mu_0 = 0.15$ for the 4-center RBF equalizer, and $\rho^2 = 2\sigma_n^2$ and $\mu_0 = 0.1$ for the 6-center RBF equalizer; while the LMS algorithm had $\mu_0 = 0.5$ for the 4-center RBF and $\mu_0 = 0.4$ for the 6-center RBF. For the LMS training, the MSE for an equalizer 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. 3, where again the results were averaged over 100 runs.

It was found that the estimated BERs of an RBF equalizer with the LMS training varied greatly for different runs. For the 4-center RBF with the LMS training, on average, the estimated BER was hardly reduced, even though there was around 7 dB reduction in the MSE, as can be seen in Figs. 2 and 3. For the 6-center RBF with the LMS training, in some runs the BERs were close to that obtained by the LBER training and for other runs the BERs were very poor, resulting in on average a BER larger than that of the linear MMSE equalizer. On the contrast, examining the MSE learning rate in Fig. 3 closely shows that the corresponding MSE reductions were consistent in different runs. This is not surprising, since the LMS is designed for minimizing the MSE not the BER. This also confirms the fact that there is no relationship between the MSE and BER for a nonlinear equalizer. In comparison, the LBER training was found to produce consistent results in different runs, and the 6-center RBF equalizer with the LBER training converged consistently very close to the optimal performance of the 64-state Bayesian equalizer.

Typical decision boundaries of the 4-center and 6-center RBF equalizers trained by the LBER algorithm are compared with the optimal Bayesian boundary in Figs. 4 and 5, respectively. The (true) BERs of the 4-center RBF equalizer trained by the LBER together with those of the linear MMSE and optimal Bayesian equalizers are depicted in Fig. 6 as functions of SINRs. The BERs of the 6-center RBF equalizer trained by the LBER are not shown here, as they are almost indistinguishable from the optimal performance. Fig. 7 depicts the same BERs of the three equalizers as functions of SNRs. The influence of the algorithm parameter ρ^2 to the performance of the LBER algorithm was also investigated. Fig. 8 shows the BERs of the 4-center RBF equalizer trained by the LBER algorithm with a range of ρ^2 in a given noise and interference condition, where it can be seen that the algorithm performance is not overly sensitive to ρ^2 over a large range of values.

Example 2. The channel transfer function was $A_0(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}$, and the co-channel transfer function was $A_1(z) = \lambda(0.6 + 0.8z^{-1})$ with the value of λ chosen to give an SIR = 20 dB. The equalizer order was set to $m = 4$ and decision delay $d = 1$. For this example, the channel state set $\mathcal{R}_0$ had 64 points and the co-channel state set $\mathcal{R}_1$ had 32 points. Thus, the number of states in $\mathcal{R}$ was 2048, and it was computationally too demanding to implement the optimal Bayesian equalizer. Given SNR = 18 dB (SINR = 15.88 dB), RBF equalizers with 16 and 32 centers were trained by the LMS and LBER algorithms, respectively. The learning rates in terms of the estimated BER are plotted in Fig. 9 for the respective adaptive equalizers, where the results were averaged over 100 runs. The LBER algorithm had $\rho^2 = 10\sigma_n^2$ and $\mu_0 = 0.3$ for the 16-center RBF equalizer, and $\rho^2 = 8\sigma_n^2$ and $\mu_0 = 0.3$ for the 32-center RBF equalizer; while the LMS algorithm had $\mu_0 = 0.3$ for both the 16-center and 32-center RBF equalizers. For the LMS training, the MSE convergence performance, averaged over 100 runs, are also given in Fig. 10.

For the 16-center RBF equalizer with the LMS training, the estimated BER on average was extremely poor, even though the corresponding MSE reduction was reasonably large. For the 32-center RBF equalizer with the LMS training, the estimated BER on average was no better than the performance of the linear MMSE equalizer. Again, it was found that the estimated BERs varied greatly for different runs. In some runs, the 32-center RBF equalizer with the LMS training were as good as the LBER training, but for other runs the results were very poor. The LBER algorithm was seen to produce consistent results. The BERs of the linear MMSE equalizer as functions of SINR and SNR are compared with those of the adaptive LBER 16-center and 32-center RBF equalizers in Figs. 11 and 12, respectively.

5 Conclusions

Adaptive training of neural network equalizers using standard stochastic gradient algorithms, such as the LMS, often encounters difficulties in practice. It has been argued that the underlying cause of these difficulties is due to adopting a "wrong" criterion, namely the MMSE. As there is no relationship between the MSE and BER for a nonlinear equalizer, it is not clear at all why the LMS algorithm is chosen for this kind of adaptive application except for its computational simplicity. A novel adaptive stochastic gradient algorithm called the LBER has been developed for nonlinear equalizers. This algorithm has a similar simplicity as the LMS but is directly linked to the MBER criterion, the real goal of equalization. This LBER algorithm has been applied to train an adaptive RBF equalizer in the presence of channel ISI, co-channel interference and additive Gaussian noise. Simulation results have shown that the LBER algorithm achieves consistent performance and has a good convergence speed. In particular, a small adaptive RBF equalizer trained by the LBER can closely approximate the optimal Bayesian performance. The results also confirm that an adaptive nonlinear equalizer trained by the LMS algorithm can produce a poor BER performance, even though it converges consistently in the MSE. Further research is warranted to investigate theoretical convergence analysis of the LBER algorithm.

References

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

[2] R.W. Chang and J.C. Hancock, "On receiver structures for channels having memory," *IEEE Trans. Information Theory*, Vol.IT-12, No.4, pp.463–468, 1966.

[3] K. Abend, T.J. Harley, Jr., B.D. Frichman and C. Gumacos, "On optimum receivers for channels having memory," *IEEE Trans. Information Theory*, Vol.IT-14, pp.818–819, 1968.

[4] K. Abend and B.D. Fritchman, "Statistical detection for communication channels with intersymbol interference," *Proc. IEEE*, Vol.58, No.5, pp.779–785, 1970.

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

[6] D. Williamson, R.A. Kennedy and G.W. Pulford, "Block decision feedback equalization," *IEEE Trans. Communications*, Vol.40, No.2, pp.255-264, 1992.

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

[8] G.J. Gibson, S. Siu, S. Chen, C.F.N. Cowan and P.M. Grant, "The application of nonlinear architectures to adaptive channel equalisation," in *Proc. ICC'90* (Atlanta, USA), 1990, pp.312.8.1.–312.8.5.

[9] S. Siu, G.J. Gibson and C.F.N. Cowan, "Decision feedback equalisation using neural network structures and performance comparison with the standard architecture," *IEE Proc. Pt.I*, Vol.137, No.4, pp.221–225, 1990.

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

[11] S. Chen, G.J. Gibson and C.F.N. Cowan, "Adaptive channel equalization using a polynomial-perceptron structure," *IEE Proc. Part I*, Vol.137, No.5, pp.257–264, 1990.

[12] G.J. Gibson, S. Siu and C.F.N. Cowan, "The application of nonlinear structures to the reconstruction of binary signals," *IEEE Trans. Signal Processing*, Vol.39, No.8, pp.1877–1884, 1991.

[13] S. Chen, G.J. Gibson, C.F.N. Cowan and P.M. Grant, "Reconstruction of binary signals using an adaptive radial-basis-function equalizer," *Signal Processing*, Vol.22, No.1, pp.77–93, 1991.

[14] S. Chen, S. McLaughlin and B. Mulgrew, "Complex-valued radial basis function network, Part II: application to digital communications channel equalisation," *Signal Processing*, Vol.36, pp.175–188, 1994.

[15] Z.J. Xiang, G.G. Bi and T. Le-Ngoc, "Polynomial perceptrons and their applications to fading channel equalization and co-channel interference suppression," *IEE Trans. Signal Processing*, Vol.42, pp.2470–2480, 1994.

[16] I. Cha and S.A. Kassam, "Channel equalization using adaptive complex radial basis function networks," *IEEE J. Selected Areas in Communications*, Vol.13, No.1, pp.122–131, 1995.

[17] C.H. Chang, S. Siu and C.H. Wei, "A polynomial-perceptron based decision feedback equalizer with a robust learning algorithm," *Signal Processing*, Vol.47, pp.145–158, 1995.

[18] E. Shamash and K. Yao, "On the structure and performance of a linear decision feedback equalizer based on the minimum error probability criterion," in *Proc. ICC'74*, 1974, pp.25F1–25F5.

[19] S. Chen, E.S. Chng, B. Mulgrew and G. Gibson, "Minimum-BER linear-combiner DFE," in *Proc. ICC'96* (Dallas, Texas), 1996, Vol.2, pp.1173–1177.

[20] C.C. Yeh and J.R. Barry, "Approximate minimum bit-error rate equalization for binary signaling," in *Proc. ICC'97* (Montreal, Canada), 1997, Vol.2, pp.1095-1099.

[21] S. Chen, B. Mulgrew, E.S. Chng and G. Gibson, "Space translation properties and the minimum-BER linear-combiner DFE," *IEE Proc. Communications*, Vol.145, No.5, 1998, pp.316–322.

[22] S. Chen and B. Mulgrew, "The minimum-SER linear-combiner decision feedback equalizer," *IEE Proc. Communications*, Vol.146, No.6, pp.347–353, 1999.

[23] C.C. Yeh and J.R. Barry, "Adaptive minimum bit-error rate equalization for binary signaling," *IEEE Trans. Communications*, Vol.48, No.7, pp.1226–1235, 2000.

[24] B. Mulgrew and S. Chen, "Adaptive minimum-BER decision feedback equalisers for binary signalling," submitted to *Signal Processing*, 1999.

[25] B. Mulgrew and S. Chen, "Stochastic gradient minimum-BER decision feedback equalisers," in *Proc. IEEE Symposium on Adaptive Systems for Signal Processing, Communication and Control* (Lake Louise, Alberta, Canada), Oct.1-4, 2000, pp.93–98.

[26] J.G. Proakis, *Digital Communications.* 3rd edition, New York: McGraw-Hill, 1995.

[27] S. Chen, S. McLaughlin, B. Mulgrew and P.M. Grant, "Bayesian decision feedback equaliser for overcoming co-channel interference," *IEE Proc. Communications*, Vol.143, pp.219–225, 1996.

[28] B.W. Silverman, *Density Estimation.* London: Chapman Hall, 1996.

[29] A.W. Bowman and A. Azzalini, *Applied Smoothing Techniques for Data Analysis.* Oxford University Press, 1997.

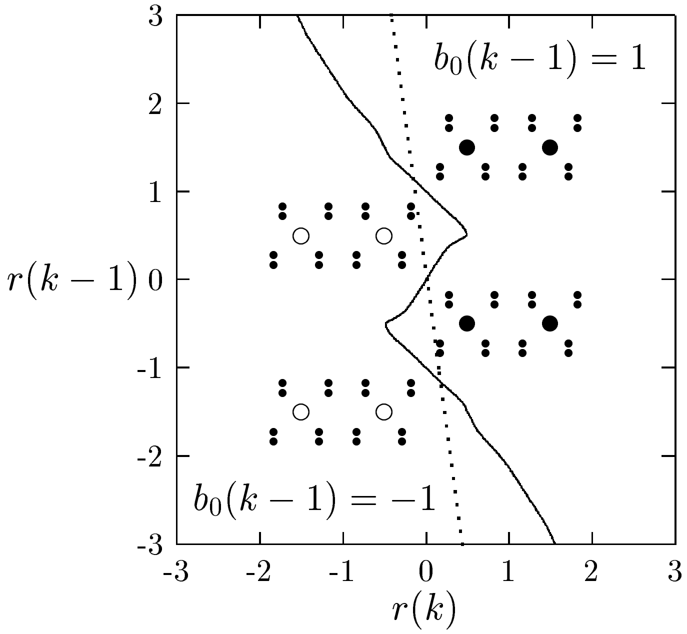

Figure 1: The sets of $\mathcal{R}_0^{(+)}$ (discuses), $\mathcal{R}_0^{(-)}$ (circles) and $\mathcal{R}$ (dots) together with the two decision boundaries (dotted: linear MMSE, solid: optimal) for Example 1. SIR = 12 dB and SNR = 20 dB (SINR = 11.36 dB).

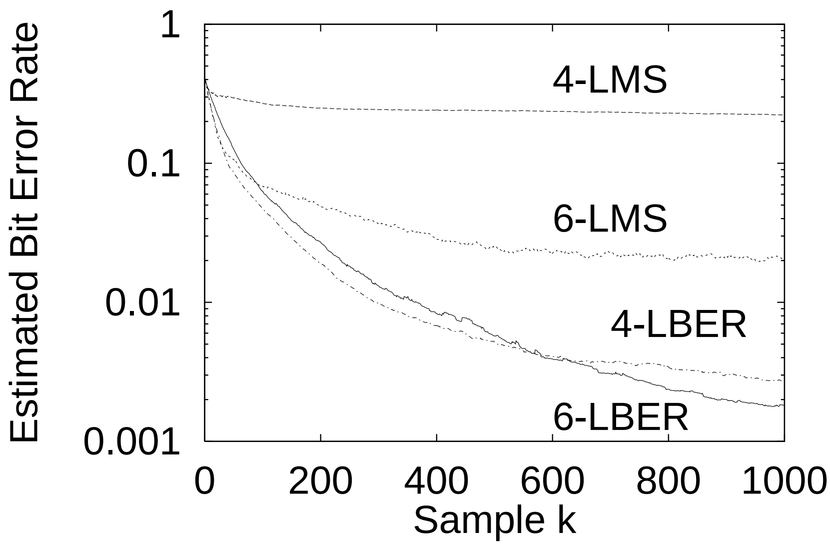

Figure 2: Convergence rates in terms of the estimated BER, averaged over 100 runs, for Example 1. SIR = 12 dB and SNR = 20 dB. 4-LMS: the 4-center RBF trained by the LMS with $\mu_0 = 0.5$, 6-LMS: the 6-center RBF trained by the LMS with $\mu_0 = 0.4$, 4-LBER: the 4-center RBF trained by the LBER with $\mu_0 = 0.15$ and $\rho^2 = 20\sigma_n^2$, and 6-LBER: the 6-center RBF trained by the LBER with $\mu_0 = 0.1$ and $\rho^2 = 2\sigma_n^2$.

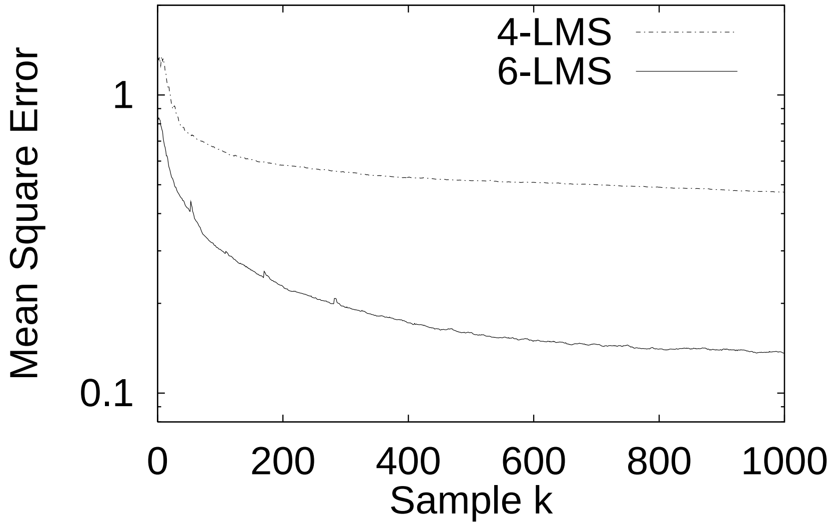

Figure 3: Convergence rates in terms of the MSE, averaged over 100 runs, for Example 1. SIR = 12 dB and SNR = 20 dB. 4-LMS: the 4-center RBF trained by the LMS with $\mu_0 = 0.5$, and 6-LMS: the 6-center RBF trained by the LMS with $\mu_0 = 0.4$.

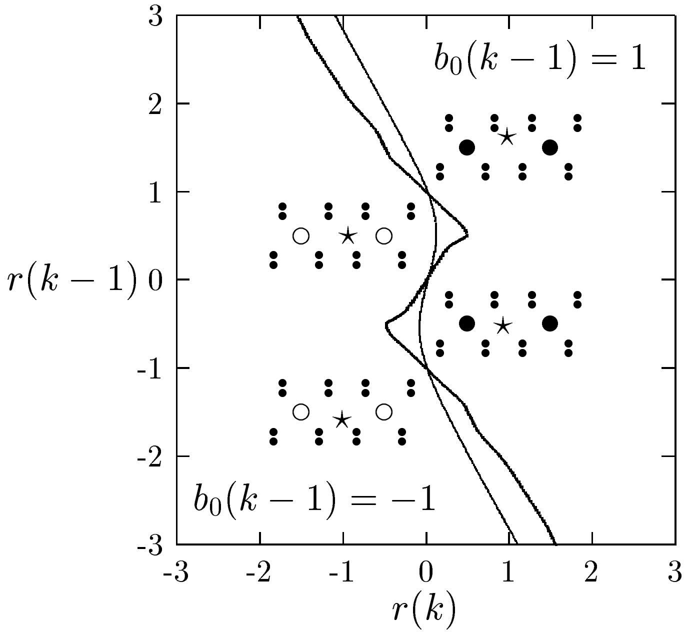

Figure 4: Comparison of two decision boundaries (thin solid: adaptive RBF equalizer, thick solid: optimal) for Example 1. SIR = 12 dB and SNR = 20 dB. The adaptive RBF equalizer has 4 centers and is trained by the LBER algorithm. The stars indicate the final center positions.

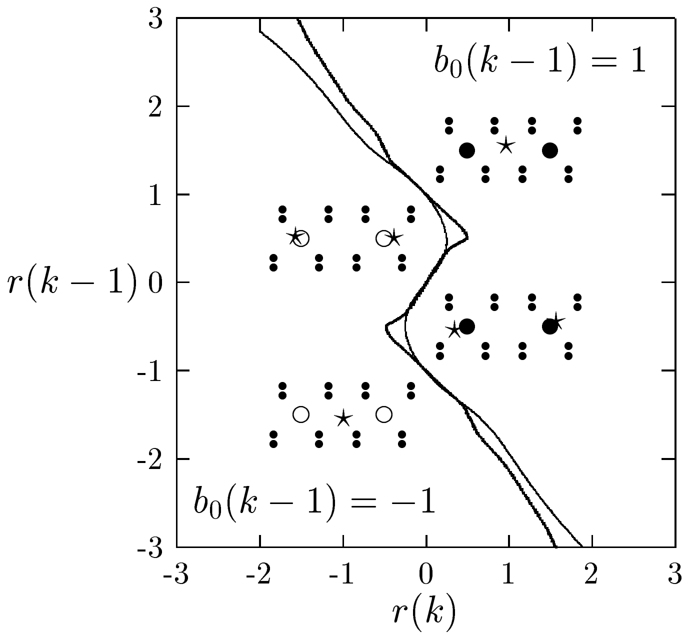

Figure 5: Comparison of two decision boundaries (thin solid: adaptive RBF equalizer, thick solid: optimal) for Example 1. SIR = 12 dB and SNR = 20 dB. The adaptive RBF equalizer has 6 centers and is trained by the LBER algorithm. The stars indicate the final center positions.

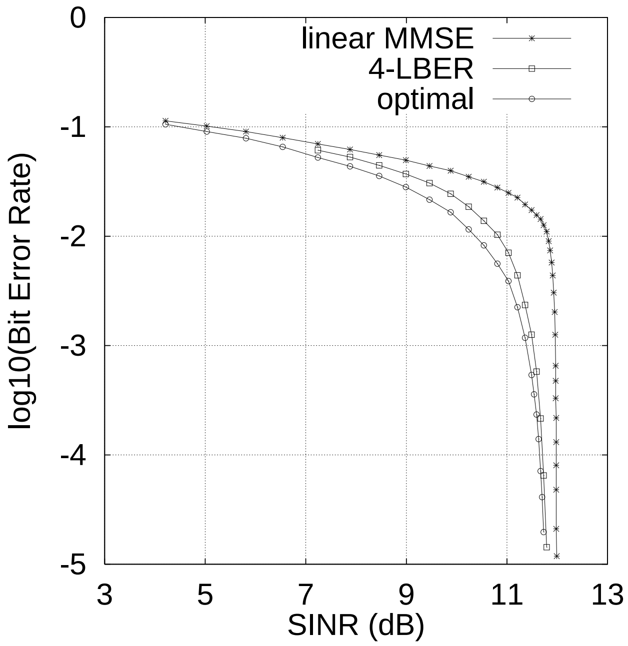

Figure 6: Performance comparison of three equalizers in terms of BER versus SINR for Example 1. SIR = 12 dB. The adaptive RBF equalizer has 4 centers and is trained by the LBER algorithm.

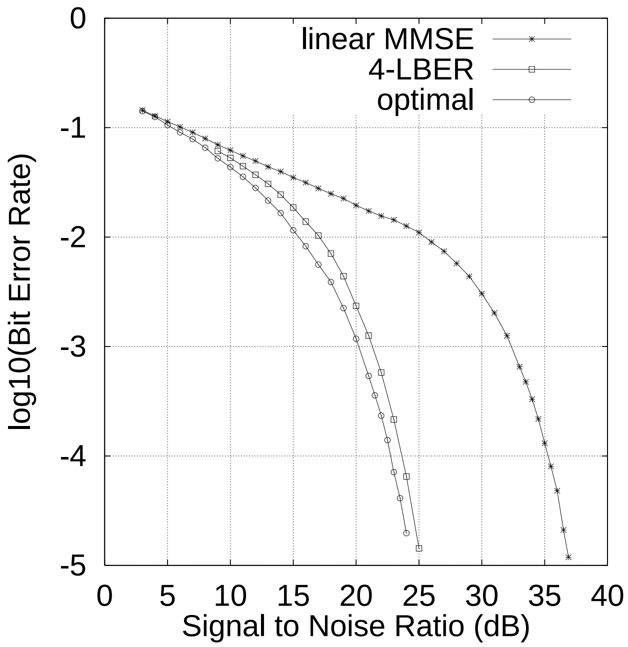

Figure 7: Performance comparison of three equalizers in terms of BER versus SNR for Example 1. SIR = 12 dB. The adaptive RBF equalizer has 4 centers and is trained by the LBER algorithm.

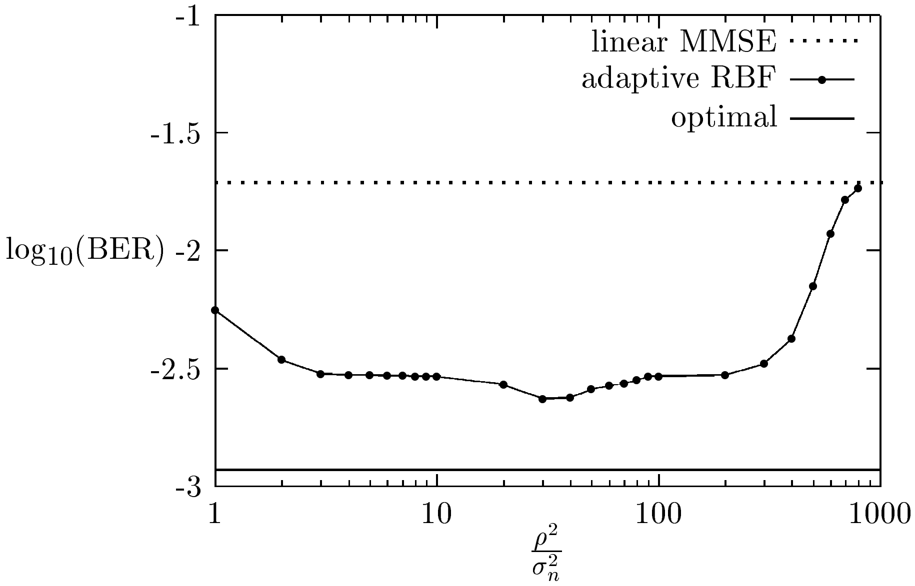

Figure 8: Influence of ρ^2 to the performance of the LBER algorithm. Example 1 with SIR = 12 dB and SNR = 20 dB. The adaptive RBF equalizer has 4 centers and the algorithm has a fixed μ_0.

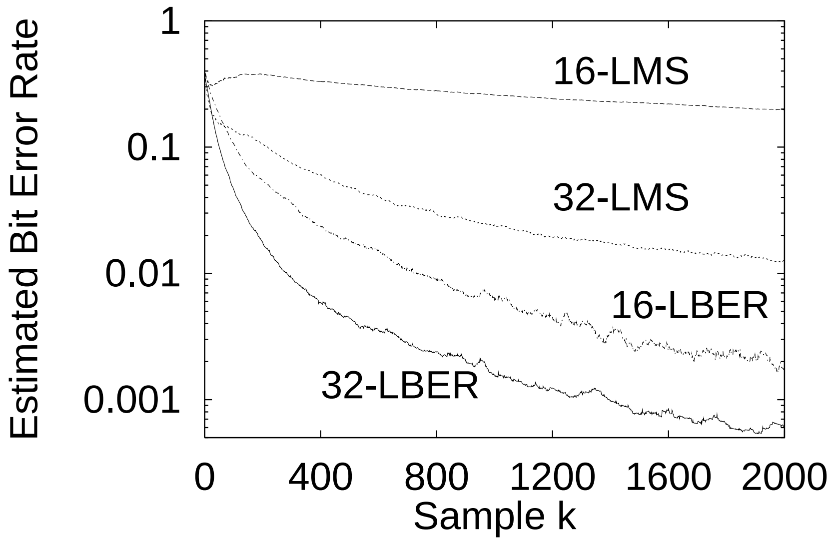

Figure 9: Convergence rates in terms of the estimated BER, averaged over 100 runs, for Example 2. SIR = 20 dB and SNR = 18 dB. 16-LMS: the 16-center RBF trained by the LMS with $\mu_0 = 0.3$, 32-LMS: the 32-center RBF trained by the LMS with $\mu_0 = 0.3$, 16-LBER: the 16-center RBF trained by the LBER with $\mu_0 = 0.3$ and $\rho^2 = 10\sigma_n^2$, and 32-LBER: the 32-center RBF trained by the LBER with $\mu_0 = 0.3$ and $\rho^2 = 8\sigma_n^2$.

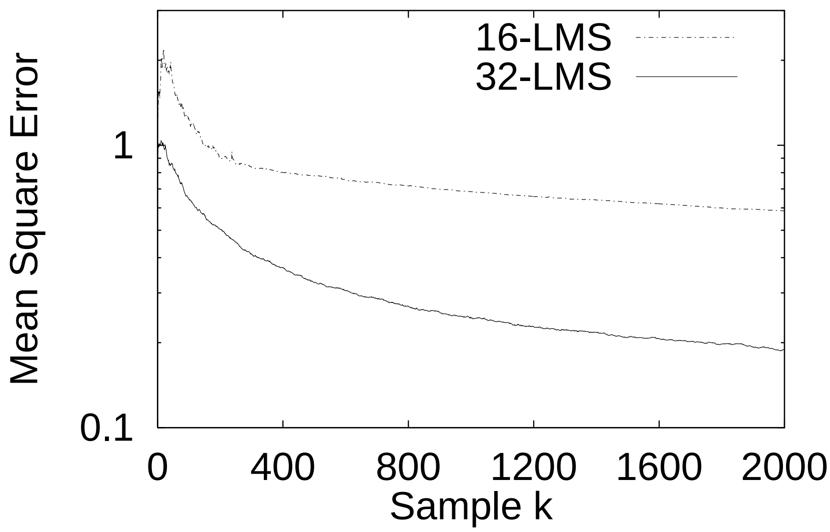

Figure 10: Convergence rates in terms of the MSE, averaged over 100 runs, for Example 2. SIR = 20 dB and SNR = 18 dB. 16-LMS: the 16-center RBF trained by the LMS with $\mu_0 = 0.3$, and 32-LMS: the 32-center RBF trained by the LMS with $\mu_0 = 0.3$.

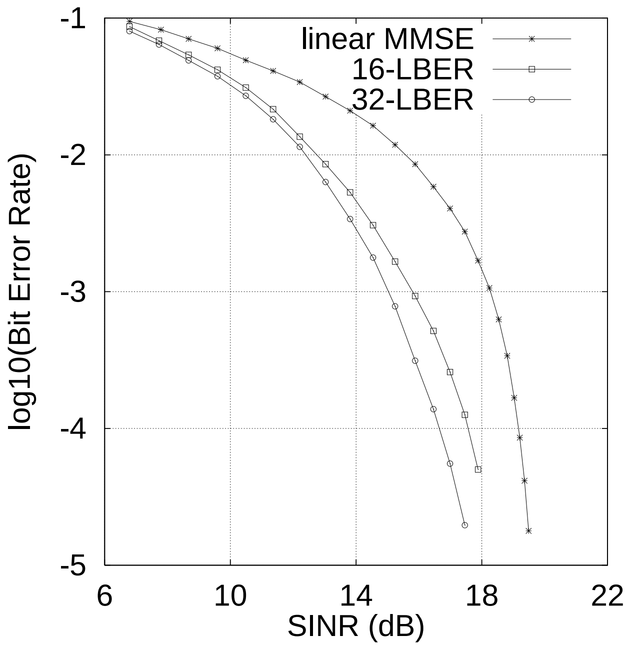

Figure 11: Performance comparison of three equalizers in terms of BER versus SINR for Example 2. SIR = 20 dB. The adaptive RBF equalizers have 16 and 32 centers, respectively, and are trained by the LBER algorithm.

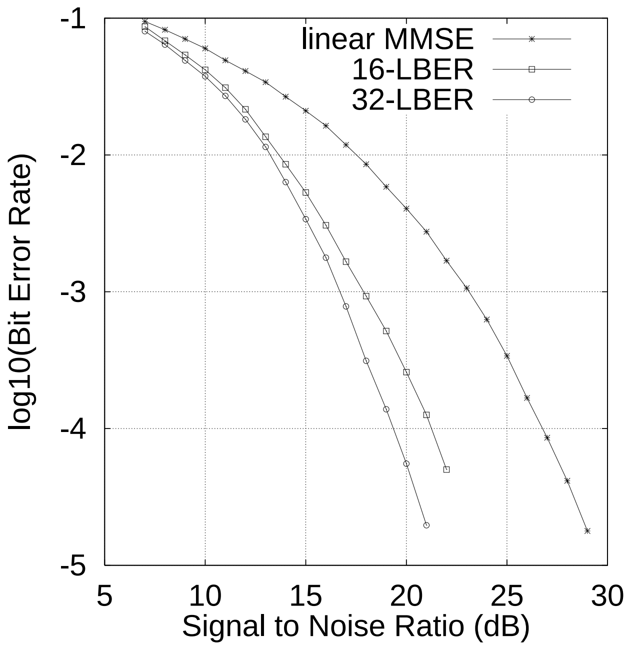

Figure 12: Performance comparison of three equalizers in terms of BER versus SNR for Example 2. SIR = 20 dB. The adaptive RBF equalizers have 16 and 32 centers, respectively, and are trained by the LBER algorithm.