MINIMIZING BER IN DFE's WITH THE ADATRON ALGORITHM

Ignacio Santamaría, Carlos Pantaleón
DICOM, ETSII y Telecom, Univ. of Cantabria, Spain
E-mail: nacho,carlos@gtas.dicom.unican.es

Jose C. Principe
Computational NeuroEngineering Laboratory, Univ. of Florida
E-mail: principe@cnel.ufl.edu

Abstract. In this paper we apply the Structural Risk Minimization (SRM) principle to minimize the Bit Error Rate in Decision Feedback Equalizers (DFE). We consider both linear discriminant (Optimal Hyperplane) and nonlinear discriminant (Support Vector Machine) classifiers as an alternative to the linear MMSE-DFE and Radial Basis Function (RBF) networks, respectively. A fast and simple adaptive algorithm called the Adatron is applied to obtain the linear or nonlinear classifier. In this way we avoid the high computational cost of quadratic programming. We also study the performance of soft margin classifiers: it is shown that to consider a regularized problem improves the BER, mainly at low SNR's. Furthermore, an adaptive implementation is discussed. Some simulation examples show the advantages of the proposed linear (OH) and nonlinear (SVM) DFE's: a better performance in comparison to the linear MMSE-DFE and a simpler structure in comparison to the RBF-DFE.

INTRODUCTION

In many digital communication receivers, a DFE is used to compensate the intersymbol interference (ISI) caused by the channel. In the equalizer literature, both a linear filter and the RBF network are being proposed for the feedforward filter in DFE's [1, 2, 3]. The RBF network can approach the optimal Bayesian detector [3], while the conventional FIR can be thought as a linear regressor in the input/ desired response space. Therefore, they implement respectively a nonlinear and linear discriminant function, and they both utilize the Minimum Mean Square Error (MMSE) criterion in choosing the optimal weights.

However, it is well known that, in general, the MMSE design does not

yield the minimum BER solution. A more appropriate criterion would be to formulate the problem as a classification task and design a maximum margin classifier by applying the SRM principle [4]. The maximum margin classifier can be applied in the input space, providing a linear discriminator that we call the Optimal Hyperplane (OH), or in feature space implementing a Support Vector Machine (SVM). The architecture of the equalizer is now given in terms of a reduced set of critical training samples known as Support Vectors.

Recently, some research has been conducted to study the application of SV machines in equalization problems. For instance, in [5, 6], the OH was obtained for a DFE showing its potential improvement over the linear MMSE-DFE solution. In [7] an SVM using a polynomial kernel was applied for nonlinear equalization achieving a similar performance than neural network-based equalizers.

In this paper we extend these previous works in the following directions: firstly, we systematize the comparisons between the MMSE and the SRM criteria as it is shown in Table 1. Secondly, we consider the application of soft margin SV machines as a way to include *a priori* information about the noise variance. Finally, to avoid the high computational cost of training SV machines by solving a quadratic optimization problem, we use a fast and simple procedure known as the Adatron algorithm [8, 9]. It obtains exactly the SVM solution, but with an exponential rate of convergence in the number of iterations.

By means of some simulation examples we show that the soft margin OH outperforms its MMSE counterpart. On the other hand, if the kernel function to map the input space into the feature space is chosen as the Gaussian kernel, we have a topology similar to the RBF network, that differs only in the criterion for optimization. The advantage of SVM over RBF equalizers is that a SVM provides a pruned structure since only those training samples or channel states that are important for classification become SVs. In this way, a SVM equalizer allows a tradeoff between complexity and performance.

	MMSE	**SRM**
Linear	FIR-MMSE	OH
Non-linear	RBF	SVM

Table 1: Characterization of equalizers in terms of the optimization criterion and the discriminant function.

LINEAR AND NONLINEAR MMSE-DFE'S

The received signal at the input of the equalizer can be expressed as

$$y(k) = \sum_{i=0}^{n_c} h_i s(k-i) + e(k) \tag{1}$$

where the transmitted symbol sequence $s(k)$ is assumed to be an equiprobable binary sequence $\{+1, -1\}$, h_i are the channel coefficients, and the measurement noise $e(k)$ can be modeled as a zero-mean Gaussian with variance σ_n^2.

The linear DFE estimates the value of a transmitted symbol as a linear combination of the channel observations and the past decisions, i.e.,

$$\hat{s}(k-d) = \text{sgn}(\mathbf{w}^T\mathbf{y}(k) + \mathbf{b}^T\hat{\mathbf{s}}_b(k)) \tag{2}$$

where $\mathbf{w} = [w_0, \cdots, w_{m-1}]^T$ are the forward coefficients, $\mathbf{b} = [b_0, \cdots, w_{n-1}]^T$ are the backward coefficients, $\mathbf{y}(k) = [y(k), \cdots, y(k-m+1)]^T$ is the vector of observations and $\hat{\mathbf{s}}_b(k) = [\hat{s}_b(k-d-1), \cdots, \hat{s}_b(k-d-n)]^T$ is the vector of past decisions. Without loss of generality we take the equalizer delay as $d = m-1$.

The noiseless vector of channel observations can be expressed using matrix notation as

$$\mathbf{y}(k) = \mathbf{H}\mathbf{s}(k) \tag{3}$$

where $\mathbf{s}(k) = [s(k), \cdots, s(k-m-n_c+1)]^T$ is the vector of transmitted symbols and $\mathbf{H}$ is an $m \times (m+n_c)$ Toeplitz channel matrix given by

$$\mathbf{H} = \begin{pmatrix} h_0 & \cdots & h_{nc} & 0 & \cdots & 0 \\ 0 & h_0 & \cdots & h_{nc} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_0 & \cdots & h_{nc} \end{pmatrix}. \tag{4}$$

For the DFE, (3) can be partitioned as

$$\mathbf{y}(k) = \mathbf{H}_1\mathbf{s}_f(k) + \mathbf{H}_2\mathbf{s}_b(k) \tag{5}$$

where, $\mathbf{H}_1$ and $\mathbf{H}_2$ are composed of the first m columns and the last n_c columns of the channel matrix $\mathbf{H}$. Now, assuming that the past decisions are correct, the vector of channel states for the DFE can be expressed as

$$\mathbf{r}(k) = \mathbf{y}(k) - \mathbf{H}_2\mathbf{s}_b \tag{6}$$

and then, as it was shown in [10], the conventional linear DFE is equivalent to an FIR filter over the translated channel states: $\hat{s}(k-d) = \text{sgn}(\mathbf{w}^T\mathbf{r}(k))$.

There are $N = 2^m$ different channel states $\mathbf{r}_i$, which can be obtained as $R = \{\mathbf{r}_i = \mathbf{H}_1\mathbf{s}_{fi};\ i = 1, \cdots, N\}$, where $\mathbf{s}_{fi}$ represent the 2^m possible feedforward sequences. For a binary digital signal, R can be partitioned into the following two subsets

$$R^{(\pm 1)} = \{\mathbf{r}_i, \quad s(k-d) = \pm 1\}. \tag{7}$$

These two classes are always linearly separable for the DFE [10]; specifically, the linear MMSE-DFE obtains the separating hyperplane that minimizes

$$J(\mathbf{w}) = E\left[\left(s(k-d) - \mathbf{w}^T\mathbf{r}(k)\right)^2\right]. \tag{8}$$

Nevertheless, this linear solution is not optimal in terms of BER: the optimal symbol-by-symbol Bayesian equalizer defines a nonlinear boundary $g(\mathbf{r}(k)) = 0$, which is given by

$$g(\mathbf{r}(k)) = \sum_{\mathbf{r}_i \in R^{(+1)}} \lambda_i \exp\left(-\frac{||\mathbf{r}(k) - \mathbf{r}_i||_2^2}{2\sigma_n^2}\right) - \sum_{\mathbf{r}_i \in R^{(-1)}} \lambda_i \exp\left(-\frac{||\mathbf{r}(k) - \mathbf{r}_i||_2^2}{2\sigma_n^2}\right). \tag{9}$$

As it is shown in [3, 11] the Bayesian detector (9) can be implemented using an RBF network by placing a Gaussian RBF unit at each channel state. The number of RBF units grows exponentially with the feedforward filter length and then, even for moderate channel lengths, this optimal approach becomes unfeasible. To mitigate this shortcoming, some techniques to reduce the number of relevant channel states via clustering have been recently proposed [12, 13].

SUPPORT VECTOR MACHINES AS DFE'S

OH and SVM DFEs

Instead of optimizing the MMSE before the slicer, the goal should be to minimize the BER. From this point of view, we can formulate our equalization problem as follows: given the training set $I = \{(\mathbf{r}_i, s_i), i = 1, \cdots, N\}$, where $\mathbf{r}_i$ are the channel states and $s_i \in \{+1, -1\}$ are the desired symbols; obtain the classifier that minimizes the BER.

Taking into account that for the DFE the channel states are linearly separable, a reasonable approximation to the minimum BER solution can be obtained by constructing an Optimal Hyperplane that maximizes the distance between the closest vectors to the hyperplane (i.e., the margin). In [4], it is shown that the maximum margin hyperplane can be obtained by minimizing

$$J(\mathbf{w}, b) = \frac{1}{2}||\mathbf{w}||_2^2, \tag{10}$$

subject to $s_i(\mathbf{r}_i\mathbf{w} + b) \leq 1$, $i = 1, \cdots, N$. This problem is equivalent to maximize the following quadratic form

$$W(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j s_i s_j < \mathbf{r}_i, \mathbf{r}_j >, \tag{11}$$

subject to the constraints: $\alpha_i \geq 0$, and $\sum_{i=1}^{N} \alpha_i s_i = 0$.

Using matrix notation we can rewrite (11) as

$$W(\mathbf{\Lambda}) = \mathbf{\Lambda}^T\mathbf{1} - \frac{1}{2}\mathbf{\Lambda}^T\mathbf{D}\mathbf{\Lambda}, \tag{12}$$

where $\mathbf{1}$ is an N-dimensional unit vector, $\mathbf{\Lambda}$ is a vector with elements α_i, and $\mathbf{D}$ is an $N \times N$ matrix with elements given by $D_{ij} = s_i s_j < \mathbf{r}_i, \mathbf{r}_j >$.

The solution of this optimization problem can be expanded in terms of the input patterns and the coefficients α_i as $\mathbf{w} = \sum_i \alpha_i s_i \mathbf{r}_i$, [4, 14]. Only the training patterns which lie closest to the hyperplane have $\alpha_i > 0$ and are called Support Vectors; all others have $\alpha_i = 0$.

Finally, the decision function for the OH equalizer is given by

$$f(\mathbf{r}) = \text{sgn}\left(\sum_i \alpha_i s_i < \mathbf{r}_i, \mathbf{r} > +b\right). \tag{13}$$

For this particular problem, due to the symmetry of the classes, we know that the border passes through the origin, so that $b = 0$.

The linear classifier can be easily extended to implement a nonlinear decision boundary by replacing the inner product in (11) and (13) by a nonlinear kernel function $k(\mathbf{r}_i, \mathbf{r}_j)$ that satisfies the Mercer condition [4]. For an equalization problem, a suitable nonlinear kernel is a Gaussian RBF

$$k(\mathbf{r}_i, \mathbf{r}_j) = \exp\left(\frac{-||\mathbf{r}_i - \mathbf{r}_j||_2^2}{\sigma^2}\right) \tag{14}$$

where σ^2 is a given parameter. In this case, the SVM takes the form of an RBF equalizer

$$f(\mathbf{r}) = \text{sgn}\left(\sum_i \alpha_i s_i k\left(\mathbf{r}_i, \mathbf{r}\right)\right). \tag{15}$$

Nevertheless, in a conventional RBF equalizer the units are located at each channel state (or a subset of them [12]), while the SVM selects only those channel states that maximize the margin in the feature space and therefore are relevant for classification purposes. This approach usually leads to a simpler structure.

Soft margin DFE's

The channel states, which are used as Support Vectors, do not take into account the noise information. Actually, the pdf of the channel observations is a set of Gaussians centered at each of the channel states. In this way, the solution provided by solving (12) can be considered optimal only for the asymptotic case of $SNR \to \infty$.

The MMSE solution, on the other hand, takes into account the noise variance through the autocorrelation matrix of the channel observations. When the noise variance increases, the MMSE hyperplane tends to rotate. This difference explains the observation that, at low SNRs, the MMSE solution can achieve a better BER that the OH solution (at least for some channels).

In the following we discuss an alternative to handle these high-noise situations. A first possibility to incorporate the noise into our classification problem could be to train the SV machine directly using the channel observations, $\mathbf{y}_i$, instead of the channel states, $\mathbf{r}_i$. However, this approach would yield a SV machine with a larger number of support vectors.

An alternative solution consists in constructing a soft margin hyperplane by minimizing the following regularized functional

$$J(\mathbf{w}, b) = \frac{1}{2}||\mathbf{w}||_2^2 + C\sum_{i=1}^{N}\xi_i^k \tag{16}$$

subject to the constraints $s_i(\mathbf{r}_i\mathbf{w} + b) \leq 1 - \xi_i$ and $\xi_i > 0$.

Typically, this soft margin alternative is used to handle situations where all the patterns cannot be correctly classified. Here we use it to improve the performance of the classifier in high noise situations.

If we consider $k = 2$ in (16), the second term minimizes the Least Square Error (LSE). In this case the optimization problem is still quadratic [14]

$$W(\mathbf{\Lambda}) = \mathbf{\Lambda}^T\mathbf{1} - \frac{1}{2}\left(\mathbf{\Lambda}^T\mathbf{D}\mathbf{\Lambda} + \frac{1}{C}\mathbf{\Lambda}^T\mathbf{\Lambda}\right) \tag{17}$$

subject to the constraints: $\mathbf{\Lambda} > 0$, and $\sum_i \alpha_i s_i = 0$.

From (17) we see that the LSE soft margin reduces to regularize the kernel matrix $\mathbf{D}$ by adding $1/C$ to the elements of the main diagonal. The similarity with the regularization performed in the MMSE solution suggests to select the regularization parameter as $1/C \propto \sigma_n^2$. After an extensive number of simulations, we have found that the optimal value is $1/C_{opt} = 2^m\sigma_n^2$ (m being the length of the feedforward filter).

THE ADATRON ALGORITHM

The computational cost associated to the quadratic programming problems (12) or (17) is one the main drawbacks in applying either the OH or the SVM to practical equalization problems. In this paper, we propose to use a simple and fast adaptive algorithm called the Adatron [8, 9] to obtain the SV machine. At each iteration the Adatron chooses a pattern from the training set and updates the corresponding Lagrange multiplier according to

$$\alpha_i = \alpha_i + \max\{-\alpha_i, \eta(1 - s_i f(\mathbf{r}_i))\} \tag{18}$$

where $f(\mathbf{r}_i)$ is the output of the SV machine and η is the learning rate.

In [8], it was proved that the Adatron converges to a maximum margin solution; that is, the minimum of (12) is a fixed point of the adaptive algorithm. Moreover, its convergence rate is exponential with the number of iterations.

The main advantages of the Adatron algorithm are its conceptual and implementation simplicity. However, it is a memory intensive algorithm, since all the kernel products, $k(\mathbf{r}_i, \mathbf{r}_j)$, must be precomputed and stored (in $\mathbf{D}$). Finally, using the Adatron algorithm we can propose the following adaptive implementation for the OH or SVM DFE:

1. Use a training sequence to estimate the channel impulse response $\hat{\mathbf{h}}$, and the noise variance $\hat{\sigma}_n^2$.

2. Using $\hat{\mathbf{h}}$ estimate the channel states as $\mathbf{r}_i = \hat{\mathbf{H}}_1 \mathbf{s}_{fi}$, $i = 1, \cdots, 2^m$.

3. Initialize $\alpha_i = 0$, $i = 1, \cdots, N$; and the learning rate η

4. While convergence criterion not true

   4.1. Choose pattern $\mathbf{r}_i$, $i \in \{1, \cdots, N\}$
   4.2. Calculate update as $\delta_i = \eta\,(1 - s_i f(\mathbf{r}_i))$
   4.3. If $(\alpha_i + \delta_i) > 0$ then $\alpha_i = \alpha_i + \delta_i$, else $\alpha_i = 0$

5. End while

Note that if the kernel correlation matrix $\mathbf{D}$ is precomputed and stored, $f(\mathbf{r}_j)$ can be efficiently obtained as

$$f(\mathbf{r}_j) =< \alpha \otimes \mathbf{s}, \mathbf{D}_j > \tag{19}$$

where $\otimes$ denotes elementwise multiplication, and $\mathbf{D}_j$ represents the jth row of matrix $\mathbf{D}$. Furthermore, to consider a soft margin SV machine we just have to regularize the kernel matrix as $\mathbf{D} + (1/C)\mathbf{I}$.

Finally, let us point out that the channel states can also be estimated using a clustering algorithm as in [3]. Using this alternative the proposed procedure can also be applied to nonlinear as well as time-varying channels (using a decision-directed mode during data transmission).

RESULTS

The aim of the first example is to compare the performance of the linear MMSE-DFE with the OH using different values of the regularization parameter $1/C$. We send binary symbols through the channel $H(z) = 0.2052 - 0.5131z^{-1} + 0.7183z^{-2} + 0.3695z^{-3} + 0.2052z^{-4}$. The structure of the DFE is $m = 4$, $n = 4$ and $d = 3$; then, the total number of channel states is 16. To train the OH we use the Adatron algorithm with a learning rate $\eta = 0.05$. Fig. 1 shows the BER curve for the MMSE-DFE and the soft margin OH. At low SNRs the non-regularized solution provides worse results than the MMSE. On the other hand, the soft margin solution with an optimal regularization parameter, $1/C_{opt} = 2^m \sigma_n^2$, is able to rotate the separating hyperplane according to the SNR, thus providing the best results. This kind of behavior has been found in a number of channels.

In our second example we compare the performance of the adaptive versions of the linear MMSE-DFE and the OH. We use a different channel: $H(z) = 0.35 + 0.8z^{-1} + z^{-2} + 0.8z^{-3}$, and the structure of the DFE is $m = 4$, $n = 3$ and $d = 3$. The channel states and the noise variance are estimated using a training sequence of 150 symbols. For this particular channel, due

to the location of the channel states, the performance of the OH-DFE is not critical with respect to $1/C$. Fig. 2 shows the values obtained for the OH-DFE with C_{opt}: its performance is clearly better than the MMSE-DFE.

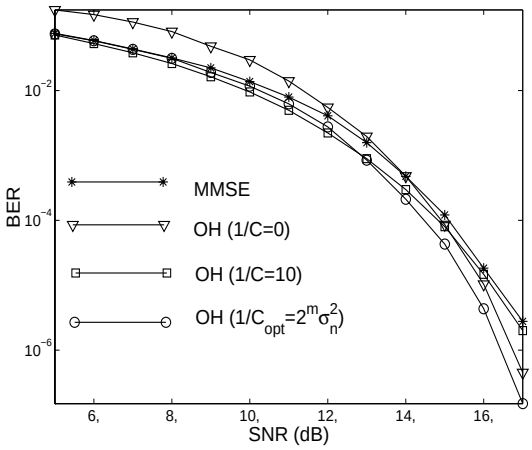

Figure 1: MMSE-DFE versus OH for channel $H(z) = 0.2052 - 0.5131z^{-1} + 0.7183z^{-2} + 0.3695z^{-3} + 0.2052z^{-4}$.

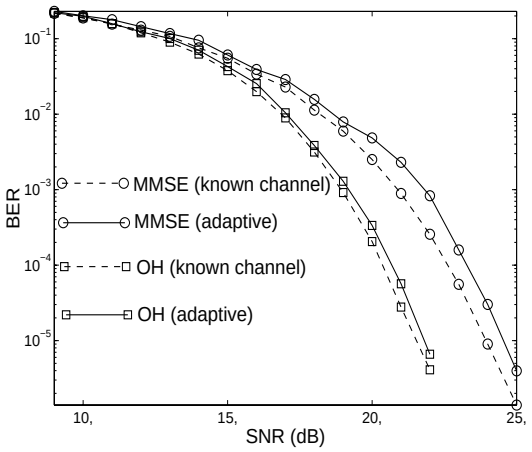

Figure 2: MMSE-DFE versus OH for channel $H(z) = 0.35 + 0.8z^{-1} + z^{-2} + 0.8z^{-3}$. Dashed lines: estimated channel states using 150 training symbols. Solid lines: noiseless channel states.

In our third example we use the following multipath channel: $H(z) = 0.4 + 0.7z^{-3} + z^{-6} + 0.6z^{-11}$, and a DFE with $m = 8$, $d = 7$ and $n = 11$. In this case, we have $2^8 = 256$ channel states. Fig. 3 shows the results obtained with the linear MMSE-DFE, the OH with $1/C = 0$ (in this case the Adatron algorithm yields 92 SV's), the soft margin OH with $1/C_{opt} = 2^m\sigma_n^2$ (using 108 SV's), the SVM using a Gaussian kernel with $\sigma^2 = 2$ (expanded in terms of 152 SV's) and the optimal Bayesian equalizer, which is implemented through

an RBF using all the 256 channel states as SV's. With approximately half the complexity of the Bayesian equalizer, the SVM only needs 0.5 dB more in SNR to achieve the same BER. By using SVM's with different values of σ^2, we can tradeoff complexity for performance.

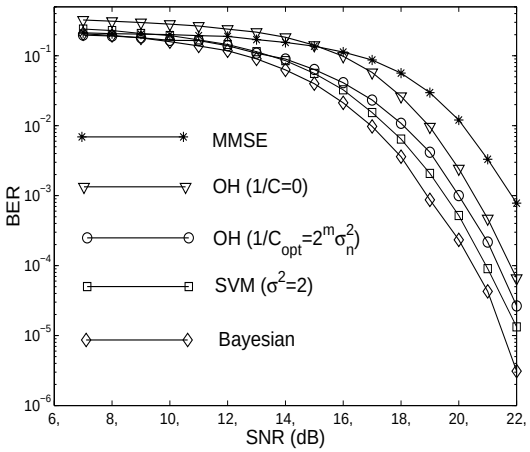

Figure 3: MMSE-DFE, OH, SVM and Bayesian for channel $H(z) = 0.4 + 0.7z^{-3} + z^{-6} + 0.6z^{-11}$.

CONCLUSIONS

Taking into account that equalization is in fact a classification problem, we have applied the SRM principle to obtain maximum margin classifiers for DFE's. Instead of quadratic programming, we train the OH and SVM equalizers using an iterative algorithm called the Adatron. We have shown that, for some channels the OH solution can achieve a large improvement over the FIR MMSE-DFE. On the other hand, in comparison with the optimal Bayesian DFE (implemented through an RBF network), the SVM uses only those channel states that are closest to the maximum margin hyperplane in the feature space, and then it yields a reduced structure with only a slight degradation in performance.

The extension of the ideas considered in this paper to multilevel modulations (M-PAM) would require solving a multiclass classification problem. Recently, several multiclass SVM's have been proposed which could be applied to this problem [15].

ACKNOWLEDGEMENTS

This work was partially supported by the European Community and the Spanish MCYT under the projects 1FD97-1863-C02-01 and 1FD97-1066-

C02-01. The first author performed the work while at CNEL, UF, supported by the Spanish Education Ministry under grant PR2000-0183.

REFERENCES

[1] S. U. H. Qureshi, "Adaptive equalization", *Proc. IEEE*, vol. 73, pp. 1349-1387, 1985.

[2] B. Mulgrew, C. F. N. Cowan, *Adaptive Filters and Equalizers*, Boston: Kluwer, 1988.

[3] S. Chen, B. Mulgrew, S. McLaughlin, "Adaptive Bayesian equalizer with decision feedback", *IEEE Trans. on Signal Processing*, vol. 41, pp. 2918-2926, 1993.

[4] V. Vapnik, *The Nature of Statistical Learning Theory*, New York: Springer Verlag, 1995.

[5] S. Chen, S. Gunn, C. J. Harris, "Decision feedback equaliser design using Support Vector Machines", *IEE Proc. Vis. Image Signal Process.*, vol. 147, no.3, pp. 213-219, June 2000.

[6] S. Chen, C. J. Harris, "Design of the optimal separating hyperplane for the decision feedback equalizer using Support Vector Machines", *Proc. of ICASSP 2000*, pp. Istanbul, Turkey, June, 2000.

[7] D. J. Sebald, J. A. Bucklew, "Support Vector Machine techniques for nonlinear equalization", *IEEE Trans. on Signal Processing*, vol. 48, no. 11, pp. 3217-3226, Nov. 2000.

[8] J. K. Anlauf, M. Biehl, "The AdaTron: an adaptive perceptron algorithm", *Europhys. Lett.*, vol. 10, pp. 687-692, 1989.

[9] T. Frieβ, N. Cristianini, C. Campbell, "The kernel-Adatron algorithm: a fast and simple learning procedure for support vector machines", in *Machine Learning: Proc. of the 15th Int. Conf.*, Shavlik, J. Ed., San Francisco: Morgan Kauffman Publishers, 1998.

[10] S. Chen, B. Mulgrew, E. S. Chng, G. J. Gibson, "Space-translation properties and the minimum-BER linear-combiner DFE", *IEE Proc. Commun.*, vol. 145, no.5, pp. 316-322, Oct. 1998.

[11] J. Cid-Sueiro, A. Artés-Rodríguez, A. R. Figueiras-Vidal, "Recurrent radial basis function networks for optimal symbol-by-symbol equalization", *Signal Processing*, vol. 40, pp.53-63, 1994.

[12] J. Lee, C. Beach, N. Tependelenlioglu, "A practical Radial Basis Function equalizer", *IEEE Trans. on Neural Networks*, vol. 10, no. 2, pp. 450-455, March 1999.

[13] A. Lyhyaoui, M. Martinez, I. Mora, M. Vazquez, J. L. Sancho, A.R. Figueiras-Vidal, "Sample selection via clustering to construct support vector-like classifiers", *IEEE Trans. on Neural Networks*, vol. 10, no. 6, pp. 1474-1481, Nov. 1999.

[14] C. Cortes, V. Vapnik, "Support-Vector networks", *Machine Learning*, vol. 20, pp- 273-297, 1995.

[15] C.W. Hsu, C. J. Lin, "A comparison on methods for multi-class Support Vector Machines", Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2001.