THE APPLICATION OF SUPPORT VECTOR MACHINES WITH GAUSSIAN KERNELS FOR OVERCOMING CO-CHANNEL INTERFERENCE

Felix Albu (1) and Dominique Martinez*(2)

(1) Faculty of Electronics & Telecommunications,
1-3 Bd. Iuliu Maniu, Bucharest, Romania
Phone: +40 1 410 54 00
Fax: +40 1 411 30 88
E-mail: felix@pns.comm.pub.ro

(2) LORIA-CNRS, Campus Scientifique - BP 239,
54506 Vandoeuvre-Les-Nancy, France
Phone: +33 383 59 30 72
Fax: +33 383 41 30 79
E-mail: dmartine@loria.fr

Abstract. This paper investigates the application of Support Vector machines (SVMs) for the equalization of communication systems corrupted with additive white Gaussian noise, intersymbol and co-channel interference. Performance obtained with SVMs for this task is compared to the one obtained with linear and Radial Basis Function (RBF) equalizers. The centers and the weights of the RBF networks are determined by the k-means and LMS algorithms, respectively. Experimental results shown that the SVM equalizer outperforms both linear and RBF equalizers, particularly for small training set. In case of time-varying channels, it is envisaged that the length of the training sequence which needs to be periodically transmitted would be reduced by SVM equalizers.

INTRODUCTION

Many communication systems are corrupted not only by channel intersymbol interference, but also by co-channel interference. An equalizer is then required to obtain a reliable data transmission. The most efficient equalizer structure is the maximum likelihood sequence estimator [8]. However, it is of limited use owing to its large computational complexity. Therefore, the most commonly used equalizers employ a symbol-decision structure based on linear filtering. This is referred to as the linear transversal equalizer (LTE).

*also with LAAS-CNRS, 7 av. Col. Roche, 31077 Toulouse, France

Parameters of the LTE are fully specified by the Wiener filter solution when a priori knowledge of the channel is available. In practice, the channel statistics are unknown and the parameters are estimated by the least-mean-square (LMS) algorithm. The LTE is easy to train but suffers from poor performance in severe channel conditions compared to nonlinear equalizers based on neural networks, such as multilayer perceptrons [10] or radial basis functions (RBFs)[4]. Unfortunately, neural networks typically require a prohibitive amount of training data before exhibiting good generalization performance. This is a serious drawback in case of time-varying channels because the training sequence needs to be periodically transmitted. In order to limit the side information required for training, there is a need for constructing nonlinear equalizers that generalize well using a restricted amount of training data. A similar problem arises in machine learning [12], [13]. A classifier that generalizes well in a high-dimensional space is the so-called *optimal hyperplane*. It provides the largest distance or *margin* from the separating hyperplane to the closest training vector. The basic idea behind the Support Vector Machines (SVMs) is first to transform the original data onto a high-dimensional space by some fixed a priori mapping, and then find the optimal hyperplane in this feature space [2], [6]. This paper investigates the application of Support Vector machines (SVMs) with Gaussian kernels for the equalization of communication systems corrupted with additive white Gaussian noise, intersymbol and co-channel interference.

SYSTEM DESCRIPTION

The discrete-time model for the digital communication system considered here is depicted in Fig.1. In this model [3], $H_0(z)$ represents the transfer function of the channel of interest. In addition, there exists p interfering cochannels with transfer functions $H_i(z)$, $i = 1 \cdots p$. Channels are traditionally modeled by finite impulse response filters

$$H_i(z) = \sum_{j=0}^{n_i} h_{ij} z^{-j}, \quad i = 0 \cdots p$$

where, n_i and h_{ij} is the filter length and the jth impulse response component for the ith channel, respectively. It is further assumed that the transmitted binary symbols $s_i = \pm 1$ are independently, identically distributed random variables and e is an additive white Gaussian noise with zero mean and variance σ_e^2. At time k, the channel observation $x(k)$ is the sum of the signal of interest $\hat{y}(k)$, the interfering signal $u(k)$ and the noise $e(k)$. The signal to noise ratio (SNR), signal to interference ratio (SIR) and signal to interference noise ratio are usually defined as

$$\mathrm{SNR} = \sigma_s^2/\sigma_e^2 \qquad \mathrm{SIR} = \sigma_s^2/\sigma_{co}^2 \qquad \mathrm{SINR} = \sigma_s^2/(\sigma_e^2 + \sigma_{co}^2)$$

where σ_s^2 and σ_{co}^2 are the signal power and the co-channel signal power, respectively.
The task of the equalizer is then to estimate the transmitted symbol $s_0(k)$ from the observation vector $\mathbf{x} = (x(k), x(k-1) \cdots x(k-m+1))$. For training the equalizer, a desired signal $s_0(k-d)$ including the equalizer delay d is available at each time step k of the training sequence. As an example, a channel corrupted with a single interfering co-channel is considered in Fig. 2.

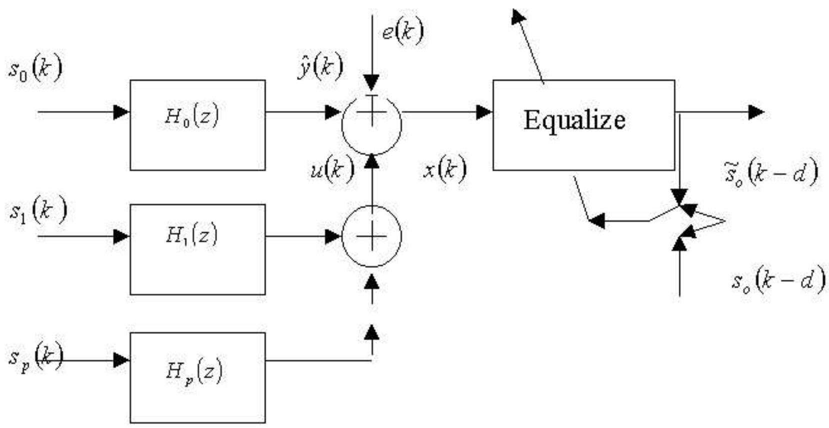

Figure 1: Discrete-time model of data transmission system corrupted with co-channel interference. At time k, the channel observation $x(k)$ is the sum of the signal $\hat{y}(k)$, the interfering signal $u(k)$ and the noise $e(k)$.

RBF EQUALIZERS

A radial basis function (RBF) network is a two-layer network with a single hidden layer. Each neuron i within the hidden layer computes the Euclidean distance between a center vector $\mathbf{c_i}$ and the input vector $\mathbf{x} = (x(k), x(k-1) \cdots x(k-m+1))$. The result is then passed through a Gaussian function $\phi_i = \exp(-||\mathbf{x}-\mathbf{c_i}||^2/\sigma_i^2)$ of width σ_i and the output is obtained by a weighted linear combination $f(\mathbf{x}) = \sum_i w_i \phi_i$. Provided a priori knowledge of the channel, the parameters $\mathbf{c_i}$, σ_i and w_i can be chosen so that the RBF network implements the optimal Bayesian equalizer [4]. Unfortunately, this requires a correct estimate of the channel order. In addition, the resulting RBF equalizer can have a prohibitive number of neurons for large m, as shown in [3]. The performance of the RBF equalizer depends to a great extent on locating the centers at the desired channel states. If this is achieved, the full effects of the co-channel interference can be taken into account and the

RBF equalizer can discriminate the signal of interest from the interfering signals and the noise [3]. Center locations are usually estimated by using supervised or unsupervised clustering algorithms [4] which typically requires a prohibitive amount of training data before exhibiting good generalization performance.

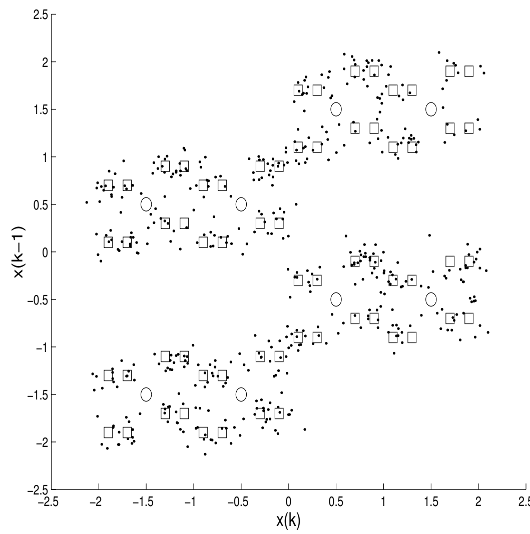

Figure 2: Example of channel output vectors and signal states for a channel $H_0(z) = 0.5 + z^{-1}$ corrupted by a single interfering co-channel $H_1(z) = 0.1 + 0.3z^{-1}$. The signal to noise ratio is 20 dB. Data points represent 500 observation vectors $(x(k), x(k-1))$, $k = 1 \cdots 500$. The locations of the signal states for channel H_0 are indicated by circles. The interfering co-channel H_1 causes an increase in the number of signal states, represented by squares. For lower signal to interference ratios, the distribution of the observation vectors spreads more widely around the signal states [3].

SVM EQUALIZERS

Let us consider a training set of n labeled observations $(\mathbf{x}_i, y_i)$, $i = 1 \cdots n$, where $\mathbf{x}_i \in R^d$ and $y_i \in \{-1, +1\}$. If the training set is linearly separable, then there exist a weight vector $\mathbf{w}$ and a bias b such that $y_i = \mathrm{sgn}(\mathbf{w}.\mathbf{x}_i + b)$ for all i, where $\mathrm{sgn}(a) = +1$ if $a \geq 0$ and -1 otherwise. The optimal hyperplane is the one that maximizes the margin or, equivalently minimizes the norm of $\mathbf{w}$ subject to $y_i(\mathbf{w}.\mathbf{x}_i + b) \geq 1$ for all i, see e.g. [12]. The solution of this optimization problem is given by the saddle point of the Lagrangian

$$\mathcal{L}_P = \frac{1}{2}||\mathbf{w}||^2 - \sum_i \alpha_i y_i (\mathbf{w}.\mathbf{x}_i + b) + \sum_i \alpha_i \tag{1}$$

where the $\alpha_i \geq 0$ are the Lagrange multipliers taking into account the inequality constraints. At the saddle point, the solution should satisfy the Karush-Kuhn-Tucker (KKT) conditions

$$\begin{aligned} \frac{\partial \mathcal{L}_P}{\partial \mathbf{w}} &= 0 \quad \Rightarrow \quad \mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i \\ \frac{\partial \mathcal{L}_P}{\partial b} &= 0 \quad \Rightarrow \quad \sum_i \alpha_i y_i = 0 \end{aligned} \tag{2}$$

The weight vector $\mathbf{w}$ given by (2) is a linear combination of the vectors of the training set. Those vectors $\mathbf{x}_i$, for which α_i are nonzero, are called support vectors. Putting expressions of $\mathbf{w}$ into (1), and taking into account the other KKT condition, one obtains

$$\mathcal{L}_D = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (\mathbf{x}_i . \mathbf{x}_j) \tag{3}$$

which now has to be maximized with respect to the α_i under the constraints $\alpha_i \geq 0$. Maximizing (3) is a quadratic programming problem with linear constraints.

The optimal hyperplane is only suitable for the linearly separable case. Possible classification errors are taking into account in the optimization problem by introducing slack variables $\xi_i \geq 0$. The optimization problem now consists of finding the minimum of $\frac{1}{2}||\mathbf{w}||^2 + C \sum_i \xi_i$ subject to

$$y_i(\mathbf{w}.\mathbf{x}_i + b) \geq 1 - \xi_i$$

where C is a parameter to be chosen by the user, a larger value corresponding to assigning a higher penalty to errors. It can be shown that the quadratic programming problem (3) remains unchanged except for the constraints which now become $0 \leq \alpha_i \leq C$ (see [6] for details). The basic idea behind the Support Vector Machines (SVMs) is first to transform the original data onto a high-dimensional space by some fixed a priori mapping, and then find the optimal hyperplane in this feature space [2], [6]. The mapping Φ onto the feature space is obtained by a given kernel representation $K(\mathbf{x}_i, \mathbf{x}_j)$ of the inner product $\Phi(\mathbf{x}_i).\Phi(\mathbf{x}_j)$ [1]. In practice, we used a Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-||\mathbf{x_i} - \mathbf{x_j}||^2/\sigma^2)$. For the optimization problem, we used an active set strategy (also known as a projection method) [11].

SIMULATION STUDY

The channel used for our simulations has the form [5]

$$H_0(z) = -0.2052 - 0.5131z^{-1} + 0.7183z^{-2} + 0.3695z^{-3} + 0.2052z^{-4}$$

and the co-channel transfer function considered was [3]

$$H_1(z) = \lambda(0.6 + 0.8z^{-1})$$

for $\lambda > 0$. This way, the SIR and SINR can be modified by varying the factor λ. The bit error rate (BER) performance of the different equalizers was estimated using 10^5 to 10^7 samples, depending on the SINR value. The delay of the equalizers was set to $d = 2$. Two experimental conditions were used. 1) The SIR was set to 24 dB and the noise power was changed to produce different SINR ratios. 2) The SNR was set to 24 dB by fixing the noise power and the interfering signal power was changed by choosing different values for λ.

The performance obtained using the Wiener filter is the best that a linear transversal equalizer can attain. The BER performance of the Wiener filter is shown in Fig. 3 as a function of the order of the filter for different SINR values and for the two experimental conditions described above (SIR=24dB and SNR=24dB). These results indicate that the BER performance does not improve when the filter order is higher than 5. This is known as the problem of noise enhancement [3]. In the following, the order of the Wiener filter was set to $m = 5$.

We now compare the performance of the Wiener filter with the one of RBF and SVM equalizers under a variety of SINR conditions. The order of the equalizer was set to $m = 5$ for the Wiener filter and $m = 4$ for the nonlinear equalizers. The amount of training data (640 samples) is small compared to the number of desired signal states, equal to 256. It should be noted that the optimal Bayesian solution would be very costly (an RBF network with 8192 centers), and was not investigated in our study. Instead, the number of centers of the RBF equalizer was set to the number of desired signal states. Training the RBF equalizer was done in the following way. The center locations were first determined by a k-means clustering procedure [7] and the Gaussian widths were set as in [3]. Then, the weights were trained with the LMS algorithm. Parameters for the SVM equalizer were determined empirically. We experimented with various C and σ^2 but we only report here the results for $C = \infty$ and $\sigma^2 = 0.5$ since they performed best overall on the data tested. The BER performance for linear and nonlinear equalizers is reported in Fig. 4 for the two experimental conditions. These results show that both non-linear equalizers outperform the Wiener filter, particularly for high SINRs. In addition, the BER value attained with an SVM equalizer is lower to that of an RBF equalizer, particularily for high SINRs. For lower values of SINR, the larger C value was penalizing too much the errors. The

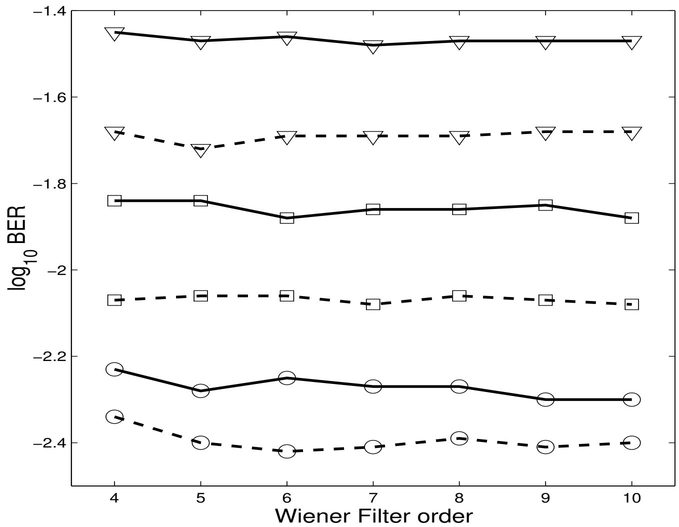

Figure 3: Performance in terms of Bit Error Rate (BER) for the Wiener filter as a function of the filter order for different values of SINR = 18 dB (circles), 14 dB (squares) and 10 dB (triangles). These performance have been estimated on the two experimental conditions : constant SIR=24dB (full lines) and constant SNR=24dB (dashed lines).

value for C could be chosen optimally by estimating both the empirical risk and the VC dimension on the training set [13].

The BER performance for the RBF and SVM equalizers is reported in Fig. 5 as a function of the length of the training sequence for the two experimental conditions. It shows that the SVM equalizer trained with 100 samples yields better performance than the RBF equalizer trained with 400 samples. Therefore, in case of time-varying channels, it is envisaged that the length of the training sequence which needs to be periodically transmitted would be reduced by SVM equalizers.

CONCLUSION

A comparative study of BER performance among three techniques for channel equalization in hostile environments has been given. Our simulations showed the superiority in overcoming co-channel interference of SVM equalizers over RBF and linear equalizers. SVMs provide a better way of choosing the number and locations of RBF centers using a limited amount of training data as was the case in our equalization problem. Our future work will focus on extending the current study to more general channel models and to online training algorithms such as, for instance, the recently proposed kernel adatron algorithm [9].

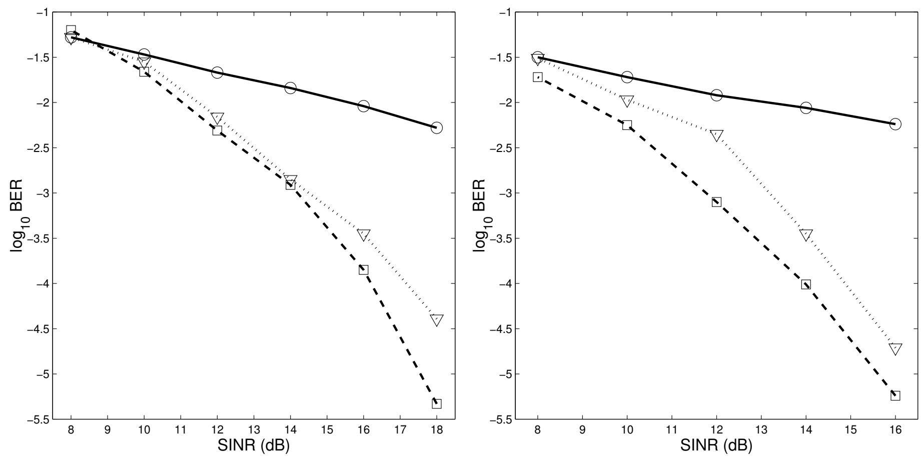

Figure 4: BER performance for the Wiener filter (full lines), the RBF equalizer (dotted lines) and the SVM equalizer (dashed lines) for different values of SINR. Figures on the left and right are for the two experimental conditions SIR=24dB and SNR=24dB, respectively.

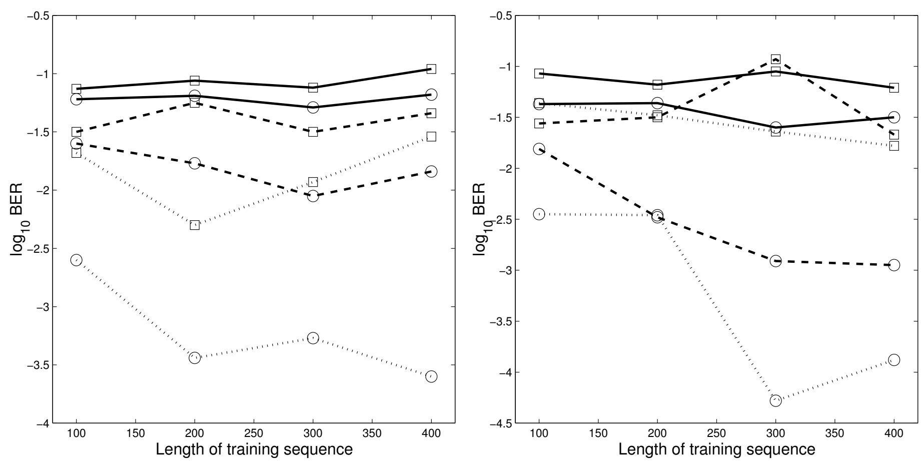

Figure 5: BER performance for the RBF equalizer (squares) and the SVM equalizer (circles) as a function of the length of the training sequence for different values of SINR equal to 8 dB (full lines), 12 dB (dashed lines) and 16 dB (dotted lines). Figures on the left and right are for the two experimental conditions SIR=24dB and SNR=24dB, respectively.

Acknowledgement

The authors wish to thank the reviewers for their valuable comments on the manuscript.

REFERENCES

[1] M. Aizerman, B. E.M. and L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," **Automation and remote control**, vol. 25, pp. 821–837, 1964.

[2] B. Boser, I. Guyon and V. Vapnik, "A training algorithm for optimal margin classifiers," in **Fith Annual Workshop on Computational Learning Theory, Pittsburgh ACM,**, 1992, pp. 144–152.

[3] S. Chen and B. Mulgrew, "Overcoming co-channel interference using an adaptive radial basis function equaliser," in **EURASIP Signal Processing**, 1992, vol. 28, pp. 91–107.

[4] S. Chen, B. Mulgrew and P. M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," **IEEE Transactions on Neural Networks**, vol. 4, no. 4, pp. 570–578, 1993.

[5] S. Chen, B. Mulgrew and S. McLaughlin, "Adaptive bayesian equalizer with decision feedback," **IEEE Transactions on Signal Processing**, vol. 41, no. 9, pp. 2918–2927, 1993.

[6] C. Cortes and V. Vapnik, "Support vector networks," **Machine Learning**, vol. 20, pp. 1–25, 1995.

[7] R. O. Duda and P. E. Hart, **Pattern Classification and Scene Analysis**, John Wiley & Sons, 1973.

[8] G. Forney, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," **IEEE Trans. Inform. Theory**, vol. IT-18, no. 3, pp. 363–378, 1972.

[9] T.-T. Friess, N. Cristianini and C. Campbell, "The kernel adatron algorithm: a fast and simple learning procedure for support vector machines," in **15th Intl. Conf. Machine Learning, Morgan Kaufman Publishers**, 1998.

[10] G. Gibson, S. Siu and C. Cowan, "The application of nonlinear structures to the reconstruction of binary signals," **IEEE Transactions on Signal processing**, vol. 39, no. 8, pp. 1877–1884, 1991.

[11] P. E. Gill, W. Murray and M. H. Wright, **Practical Optimization**, Academic Press, 1981.

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

[13] V. Vapnik, **Statistical learning theory**, John Wiley & Sons, Inc, 1998.