# NON-LINEAR CHANNEL EQUALISATION USING MINIMAL RADIAL BASIS FUNCTION NEURAL NETWORKS

*P Chandra Kumar, P Saratchandran,Senior Member,IEEE and N Sundararajan,Fellow,IEEE*

School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798
Phone: +65-799-5029 Fax: +65-792-0415

## ABSTRACT

This paper presents the study results of non-linear channel equalisation problems in data communications using a recently developed minimal radial basis function neural network structure, referred to as MRAN(Minimal Resource Allocation Network). MRAN algorithm uses on-line learning and has the capability to grow and prune the RBF network's hidden neurons ensuring a parsimonious network structure. Compared to earlier methods, the proposed scheme does not have to estimate the channel order first, and fix the model parameters. Results showing the superior performance of the MRAN algorithm for two different non-linear channel equalisation problems, along with a linear non-minimum phase problem, are presented.

## 1. INTRODUCTION

In high speed data communication, channel equalisation plays a major role in extracting true data from the noisy transmitted data corrupted with intersymbol interference (ISI) and other channel distortions. Conventional methods[5] use linear channel equalisation schemes which employ a linear filter with a finite impulse response(FIR) or lattice structure, and non-linear methods like Decision Feedback Equalisation (DFE), and Maximum Likelihood (ML) sequence detection schemes.

Previous studies had shown that non-linear equalisation methods perform better than linear methods[5], as they exploit some non-linearity in the equalisation process. Multi-layer feed-forward neural networks and Radial Basis Function (RBF) networks have been proposed recently to exploit the non-linearity in channel equalisation[6][7]. This is because Artificial Neural Networks (ANN) can easily perform non-linear classifications and function associations. Recently, a new minimal RBF neural network called MRAN (Minimal Resource Allocation Network) was developed by Yingwei et al [2], which uses a scheme for adding and pruning RBF centres, so as to achieve a minimal network. This paper presents the use of MRAN for non-linear channel equalisation problems, as well as a linear non-minimum phase channel equalisation problem. The superiority of this method over existing methods is that, a separate channel order estimation is not necessary. The algorithm uses an Extended Kalman Filter(EKF) to determine the weight and width of each of the nodes. This is different from previous studies, where the width values have to be set to an estimate of the noise variance of the received data. The weights are also not fixed as 1 or -1, as suggested by Chen[7]. This means that the binary nature of the data is not exploited. However, the advantage in MRAN is that the RBF nodes could adjust their weight and width values, so as to accommodate more data around their location. This would mean than the resultant network may have even fewer nodes than that required, if a node is to be placed in each of the desired channel states.

## 2. EQUALISATION PROBLEM

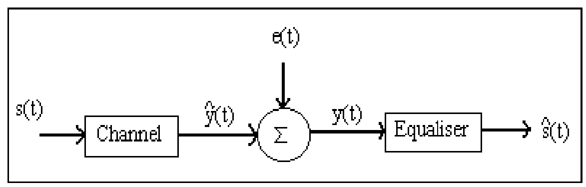


**Fig 1** Discrete time model of data transmission system

A discrete time model of a digital communication system is shown in figure 1, where the input digital sequence s(t) is transmitted through a dispersive channel. Often, this is a non-linear channel, like the one shown by equation (1)

$$y(t) = x(t) + k_1x^2(t) + e(t)$$
$$H(z) = X(z)/S(z) = k_2 + k_3z^{-1} + k_4z^{-2} \qquad (1)$$

where $k_1$,$k_2$,$k_3$ and $k_4$ are constants. This 'channel' includes the effects of the transmitter filter, the transmission medium, the receiver matched filter, and other components. The transmitted symbol s(t) is assumed to be an equiprobable and independent binary sequence taking values of either +1 or -1. The noise-free output of the channel, $\hat{y}(t)$ is added with a zero mean Gaussian white noise, e(t), to obtain a noisy channel output, y(t). The equaliser performs the task of recovering an estimate $\hat{s}(t)$ of the transmitted symbols, s(t), based on the noisy channel observation y(t).

Using estimation theory [5], it is known that the Maximum Likelihood Sequence Estimator (MLSE) for the entire transmitted sequence would yield the best performance for symbol detection. This method however, is highly complex with high computational requirements, and the delay in decision output is often unacceptable in many practical communication systems. Thus, most practical equalisation systems employ some form of symbol-by-symbol, decision making architecture.

In such architecture, the past *m* channel observations are used to make an estimate, $\hat{s}(t-\tau)$ of the input symbol, s(t), with a delay

$\tau$. For the channel given in (1), there will be $n_s$ combinations of the input sequence where $n_s=2^{n_h+m}$, and $n_h$ is the order of the linear component of the channel. The $n_s$ input sequence

$$\mathbf{s(t)} = [s(t)\ldots s(t-m+1-n_h)]^T \qquad (2)$$

would result in $n_s$ points of noise free channel output vector

$$\hat{\mathbf{y}}(\mathbf{t}) = [\hat{y}(t)\ldots \hat{y}(t-m+1)]^T \qquad (3)$$

These output vectors are also referred to as desired channel states, and are partitioned into different classes, $Y^+_{m,\tau}$ & $Y^-_{m,\tau}$, for $s(t-\tau) = 1$ & $s(t-\tau) = -1$ respectively. Due to the additive white gaussian noise (AWGN), the channel outputs will form clusters around each of these desired channel states. The noisy observation vector

$$\mathbf{y(t)} = [y(t)\ldots y(t-m+1)]^T \qquad (4)$$

is used to determine the transmitted symbol $s(t-\tau)$, according to the Bayes decision theory[5][7]. For equiprobable symbols, the Bayesian decision function is defined by

$$f_B(\mathbf{y}(t)) = \sum_{i=1}^{n_s^+} \exp(-\|\mathbf{y}(t)-\mathbf{y}_i^+\|^2/2\sigma_e^2) - \sum_{j=1}^{n_s^-} \exp(-\|\mathbf{y}(t)-\mathbf{y}_j^-\|^2/2\sigma_e^2) \qquad (5)$$

where $n_s^+$ and $n_s^-$ are the number of $\mathbf{y}^+$ and $\mathbf{y}^-$ states in $Y^+_{m,\tau}$ and $Y^-_{m,\tau}$ respectively, and $\sigma_e^2$ is the noise variance. As $\hat{s}(t-\tau)=1$, when $f_B(\mathbf{y})\geq 0$, and $\hat{s}(t-\tau)=-1$ otherwise, the optimal decision boundary is defined by

$$\{\mathbf{y} \mid f_B(\mathbf{y}) = 0\} \qquad (6)$$

which is a hypersurface in the observation space. As RBF networks are well suited for realizing this Bayesian function[7] as well as performing such non-linear mapping we investigate an algorithm that could build such a equaliser network.

## 3. MINIMAL RESOURCE ALLOCATION NETWORK (MRAN)

The MRAN is a sequential learning algorithm for minimum RBF neural network, recently developed by Yingwei et al[2],[3]. Only a brief description of the algorithm is given here. For details please refer to [2],[3].

The output of the network used by this algorithm has the following form :

$$f(x) = \alpha_0 + \sum_{k=1}^{K} \alpha_k \phi_k(x) \qquad (7)$$

where $\phi_k(x)$ is the response of the $k^{th}$ hidden neuron to the input x, and $\alpha_k$ is the weight connecting the $k^{th}$ hidden unit to the output unit. $\alpha_0$ is the bias term. Here, K represents the number of hidden neurons in the network. $\phi_k(x)$ is a Gaussian function given by,

$$\phi_k(b)=\exp(-\| b-\mu_k \|^2/\sigma_k^2) \qquad (8)$$

where $\mu_k$ is the centre and $\sigma_k$ is the width of the Gaussian function. $\|\ \|$ denotes the Euclidean norm.

In the algorithm, the network begins with no hidden units. As each input-output training data is received, and processed, the network is built up based on certain growth criteria. The algorithm adds hidden units, as well as adjusts the existing network, according to the data received. The criteria that must be met before a new hidden unit is added are :

$$\|x_n - \mu_{nr}\| > \epsilon_n \qquad (9)$$

$$e_n=y_n - f(x_n) > e_{min} \qquad (10)$$

$$e_{rmsn} = \sqrt{\frac{\sum_{i=n-(M-1)}^{n} e_i^2}{M}} > e_{\min 1} \qquad (11)$$

where $\mu_{nr}$ is the centre (of the hidden unit) which is closest to $x_n$. $\epsilon_n$, $e_{min}$ and $e_{min1}$ are thresholds to be selected appropriately. Equation (9) ensures that the new node to be added is sufficiently far from all the existing nodes. Equation (10) decides if the existing nodes are insufficient to obtain a network output that meets the error specification. Equation (11) checks that the network has not met the required sum squared error specification for the past *M* outputs of the network. Only when all these criteria are met, is a new node added to the network.

When an input to the network does not meet the criteria for a new hidden unit to be added, the network parameters are adapted using the EKF. The algorithm also incorporates a pruning strategy, which is used to prune hidden nodes that do not contribute significantly to the outcome of the network, or are too close to each other. The former is done by observing the output of each of the hidden nodes for a period of time, and then removing the node that has not been contributing a significant output for that period. Also, if two hidden units are found to be closer than a threshold value, and the values of the parameters $(\alpha,\sigma)$ associated with the nodes are close, the two nodes are combined to form a single node, and the dimensions of the EKF are reduced.

A number of successful applications of MRAN in different areas such as function approximation, time series prediction, time-varying non-linear system identification have been reported in [2],[3] and [4]. This is the first time MRAN is being applied to equalisation problems.

## 4. MRAN FOR NON-LINEAR CHANNEL EQUALISATION

The performance of MRAN for equalisation is shown here on the following examples.

Example 1 (Non-Linear Channel 1)

To test the algorithm for non-linear channels, the following non-linear channel [1] was chosen :

$$y(t) = x(t) + 0.2x^2(t) + e(t)$$

$$H(z) = X(z)/S(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2} \qquad (12)$$

In their paper, Kechriotis et al [1] had used a recurrent neural network with multiple iterations to realise an equaliser network. For the purpose of graphical display, the equalizer order is chosen as m=2. Thus, a two dimensional plot can be made, to show the two most recent inputs to the equaliser for each input data passed through the channel. In the example, $n_h$=2. Thus, there will be 16 desired states for the channel output, ( $2^{n_h+m}$ =16). The decision delay was set to one ($\tau$=1). By using the MRAN algorithm with 1000 data samples at 25dB snr, we were able to obtain the classification boundary shown in figure 2.

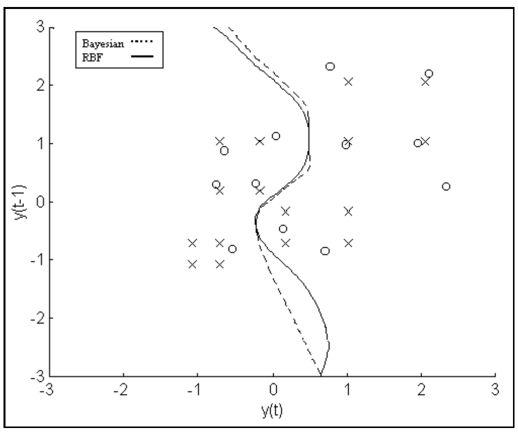


**Fig 2** Bayesian and MRAN boundary and location of RBF centres for non-linear channel

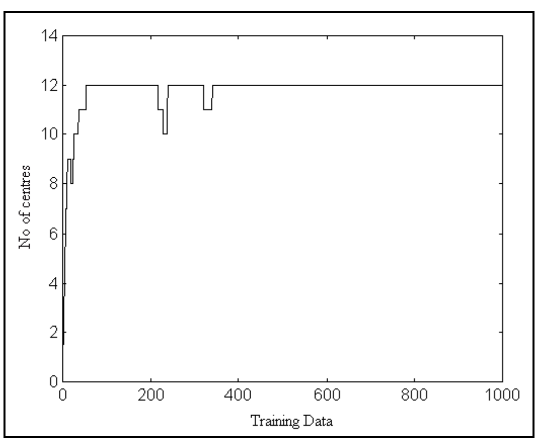


**Fig 3** Number of centres obtained as training progresses

The value of the thresholds $e_{min}$ and $e_{min1}$ were both set to 0.1. The other parameters were set as $\in$=0.5, and $M$=10. The Bayesian boundary is shown by the dotted line, while the boundary obtained by the algorithm is shown by the continuous line. The RBF centres created by the algorithm are indicated by a ‘o’, while the actual desired states are indicated by the ‘x’.

The network has built up 12 hidden nodes. Figure 3 shows the growth of the RBF network, as training progresses. This is much less than the 16 desired channel states. However, it can be seen that the Bayesian boundary is still well approximated, at the critical region, which is at the centre of the figure. The network boundary deviates from the Bayesian boundary, at the bottom region in the figure, but this can be seen to be less critical in the equalisation task, from the BER curves shown in figure 4. The Bit-Error Rate (BER) is one method of testing to see if an equaliser is performing as required. The network was tested with a million test data of various SNR to obtain the BER curves. As it can be seen, the performance of the network is comparable to that of the ideal case.

Example 2 (Non-Linear Channel 2)

A second non-linear channel with the following model [1] was used :

$$y(t) = x(t) + x^2(t) + 0.7x^3(t) + 0.5x^4(t) + e(t)$$

$$H(z) = X(z)/S(z) = 1 + 0.7z^{-1} \qquad (13)$$

Such channel models are frequently encountered in data transmission over digital satellite links, especially when the signal amplifiers operate in their high-gain limits. The equaliser order and delay were chosen to be m=2 and $\tau$=1 respectively. 1000 training data bits were used. They were mixed with low noise to get 30dB SNR. The parameters $e_{min}$ and $e_{min1}$ were both set to 0.9. The other parameter values were, $\in$=0.5, and M=10. The resulting network had 6 units, as compared with the 8 desired channel states. A comparison of their BER in figure 4, shows that the network performs only slightly poorer than the Bayesian equaliser.

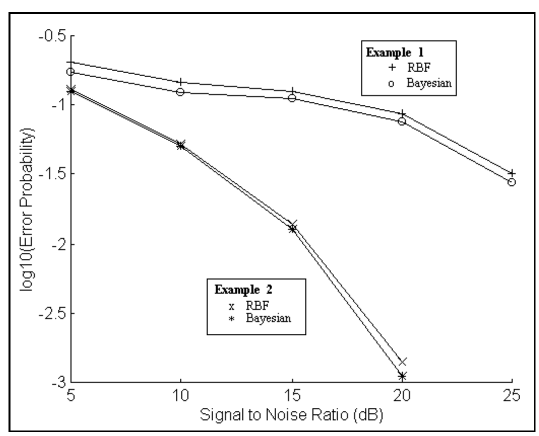


**Fig 4** Error curves for ideal and RBF networks for the non-linear channel examples

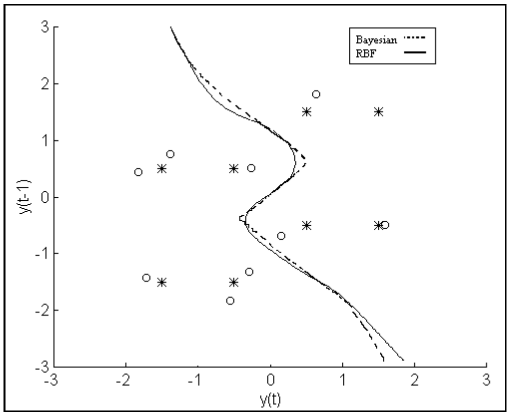


**Fig 5** RBF boundary obtained for 160 samples at 15dB

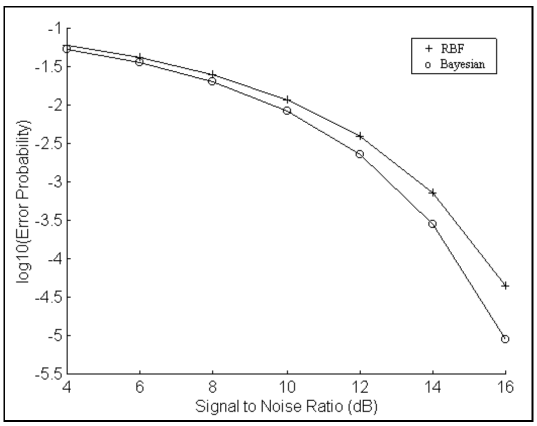


**Fig 6** Error curves for ideal and RBF networks

Linear non-minimum phase Channel Example

To compare the algorithm with earlier methods of RBF equalisation, a linear non-minimum phase channel was chosen with the following transfer function :

$$H(z) = 0.5 + 1.0z^{-1} \qquad (14)$$

This is the channel used by Chen et al[7]. In his paper a clustering algorithm for 160 data samples of 10dB snr to obtain an RBF equaliser network was used. The method, however, had to have knowledge of which state was being transmitted, and the total number of states, so that the data belonging to each state could be clustered together. In our method, 160 samples of the data at 15dB snr were used to build-up/train the network using the MRAN algorithm. There was no need for estimation of the channel order, to estimate the number of states. Neither was there a need to know which state the transmitted data belonged to.

Figure 5 shows the boundary obtained after using MRAN to train the network. The parameter values were $e_{min}=0.1$, $e_{min1}=0.3$, $\in_n=0.5$, $M=10$. The desired states are shown by a '*' while the RBF centres are shown by a 'o'. Though the number of RBF centres is 9, as compared with the ideal case of 8, the boundary obtained is indeed comparable with that of the actual Bayesian boundary. This shows that the MRAN algorithm is able to build up a network that can perform equalisation comparable to that of a Bayesian equaliser, without having a need to estimate the channel order.

A plot of the probability of error for the RBF network and the Bayesian equaliser is shown in Figure 6, for test data of varying SNR. One million data bits were used to test the MRAN network, and the ideal Bayesian equaliser, for each SNR. From the probability of error curves, it can be seen that the network's performance is indeed comparable to that of the ideal equaliser.

# 6. CONCLUSION

The MRAN algorithm using Radial Basis Function Neural Networks was seen to be well suited for non-linear channel equalisation problems. Its ability to build up a network, based on certain parameters was seen to have an advantage over other methods, as it could be used for on-line training of the data for equalisation. The algorithm's performance was evaluated by using it to build up an equalisation network for two non-linear channels, along with one non-minimum phase channel. The resulting networks were then tested by comparing their bit error rate (BER) performance to that of the ideal Bayesian equaliser. The results show that the networks obtained, are comparable in performance to ideal equalisers when suitable training parameters are selected.

# 7. REFERENCES


[1] G.Kechriotis, E.Zervas, and E.S.Manolakos, "Using Recurrent Neural Networks for Adaptive Communication Channel Equalization", *IEEE Transactions on Neural Networks*, Vol 5,No.2, March 1994.

[2] Lu Yingwei, N Sundararajan, P Saratchandran, "Adaptive nonlinear system identification using minimal radial basis function neural networks", *IEEE ICASSP*, Vol 6, pp 3521-3524, 1996.

[3] Lu Yingwei, N Sundararajan, P Saratchandran, "A sequential learning scheme for function approximation using minimal radial basis function neural networks", *Neural Computation*, Vol 9, No. 2, pp 461-478, Feb 1997.

[4] Lu Yingwei, N Sundararajan, P Saratchandran, "Identification of time-varying nonlinear systems using minimal radial basis function neural networks", *IEE Proceedings -Control Theory Applications*, Vol 144, No. 2, pp 202-208, March 1997.

[5] Proakis, J. G., Digital Communications. New York: McGraw-Hill, 1983.

[6] 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, pp 107-119, 1990.

[7] S. Chen, Mulgrew B, Grant P M, "A Clustering Technique for Digital Communications Channel Equalization Using Radial Basis Function Networks", *IEEE Transactions on Neural Networks*, Vol 4, No 4, July 1993.