



**HAL**  
open science

# Neural Networks-Based Turbo Equalization of a Satellite Communication Channel

Hasan Abdulkader, Bouchra Benammar, Charly Poulliat, Marie-Laure Boucheret, Nathalie Thomas

► **To cite this version:**

Hasan Abdulkader, Bouchra Benammar, Charly Poulliat, Marie-Laure Boucheret, Nathalie Thomas. Neural Networks-Based Turbo Equalization of a Satellite Communication Channel. 15th International Workshop on Signal Processing Advances in Wireless Communications - SPAWC 2014, Jun 2014, Toronto, Canada. pp. 494-498. hal-01147225

**HAL Id: hal-01147225**

**<https://hal.science/hal-01147225>**

Submitted on 30 Apr 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : <http://oatao.univ-toulouse.fr/>  
Eprints ID : 13123

**To link to this article** : DOI :10.1109/SPAWC.2014.6941914  
URL : <http://dx.doi.org/10.1109/SPAWC.2014.6941914>

**To cite this version** : Abdulkader, Hasan and Benammar, Bouchra and Poulliat, Charly and Boucheret, Marie-Laure and Thomas, Nathalie *Neural Networks-Based Turbo Equalization of a Satellite Communication Channel*. (2014) In: 15th International Workshop on Signal Processing Advances in Wireless Communications - SPAWC 2014, 22 June 2014 - 25 June 2014 (Toronto, Canada).

Any correspondance concerning this service should be sent to the repository administrator: [staff-oatao@listes-diff.inp-toulouse.fr](mailto:staff-oatao@listes-diff.inp-toulouse.fr)

# Neural Networks-Based Turbo Equalization Of a Satellite Communication Channel

Hasan Abdulkader\*, Bouchra Benammar\*, Charly Poulliat\*, Marie-Laure Boucheret\* and Nathalie Thomas\*

\* University of Toulouse, INPT-ENSEEIH/IRIT

Email: hasan.abdulkader, bouchra.benammar, charly.poulliat, marie-laure.boucheret, nathalie.thomas}@enseeiht.fr

**Abstract**—This paper proposes neural networks-based turbo equalization (TEQ) applied to a non linear channel. Based on a Volterra model of the satellite non linear communication channel, we derive a soft input soft output (SISO) radial basis function (RBF) equalizer that can be used in an iterative equalization in order to improve the system performance. In particular, it is shown that the RBF-based TEQ is able to achieve its matched filter bound (MFB) within few iterations. The paper also proposes a blind implementation of the TEQ using a multilayer perceptron (MLP) as an adaptive model of the nonlinear channel. Asymptotic analysis as well as reduced complexity implementations are also presented and discussed.

## I. INTRODUCTION

Essential satellite communication requirements are spectral and power resources. High efficient modulations are often used to enhance the capacity of communication systems with limited bandwidth and power resources. On board a satellite, a HPA is used to amplify the incoming signal to be sent back to earth. It is desirable to operate near the HPA saturation in order to provide sufficient power to the transmitted signal. However, the HPA performs non linearly near its saturation point and thus becomes a bottleneck to the system capacity. Constant modulus modulations are frequently used in such a channel to reduce the impact of non linear amplification.

Turbo equalization is a scheme attempting to iteratively mitigate and overcome the ISI incurred in the channel [1]–[3]. It is based on the turbo-principle widely used in modern digital communication receivers. In the literature, many schemes apply this principle to jointly and iteratively decode, equalize or demodulate received signals. Optimal MAP and suboptimal MMSE algorithms have been widely studied for linear ISI channels equalization [1]–[3].

Neural networks (NN) are efficiently used to carry out complex and non linear problems [4]. NN equalization of communication channels has been intensively investigated [5]–[7] using different architectures such as MLP, RBF, recurrent neural network (RNN), etc. In [8] RBF neural networks has been applied as SISO device to equalize linear channels. Based on the optimal decision theory, it showed good performance allowing a reduced complexity implementation. Later, it has been used for turbo equalization of linear channels and has shown good performance for different modulation techniques [9], [10].

In this paper, we show how RBF-Based equalization can be successfully applied to non linear channels equalization when Volterra-based non linear channel models are considered. The RBF-Based TEQ can be used to implement the optimal decision rule given in [8]–[10]. We use the reduced complexity algorithm for its interesting performance and simplicity.

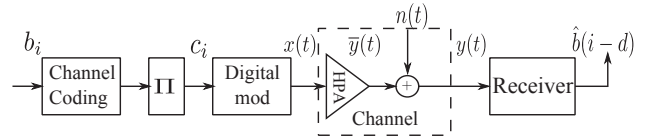


Fig. 1. Model of the communication system.

Besides, it can be easily implemented using detected symbols at the equalizer output. These symbols are fed back to the equalizer input allowing to reduce the size of the RBF hidden layer. We also propose a blind equalization scheme. In [10] the transmission channel has been estimated using linear filter. In our case, non linear tools are needed to model the non linear channel. We propose to estimate the HPA response using a MLP neural network.

The paper is organized as follows: first we describe the satellite non linear channel, then we introduce the turbo equalizer in section II. In section III, we derive the RBF-Based TEQ in the case of a Volterra non linear channel and its low complexity implementation. The blind equalization structure is briefly described in section IV. In section V, simulation results are presented and analysed using EXIT chart. Finally conclusions are drawn in section VI.

## II. SYSTEM MODEL

In this paper, we consider a non linear communication channel given by a Volterra filter [11], [12]. Figure 1 illustrates the proposed communication system model. Independent and identically distributed bits  $b_i$  are first encoded by a channel encoder, interleaved and as coded bits  $c_i$  fed to a digital modulator that maps blocks of  $\log_2(M)$  coded bits into a complex symbol  $x$ . Let  $M$  be the cardinality of the digital modulation constellation. After pulse shaping, the symbol sequence is sent to a satellite transponder consisting of an HPA and input/output filters. The amplified signal is transmitted downlink to the receiver with an additive Gaussian noise and is matched-filtered before being sampled at the baud rate. Many models of the non linear satellite channel have been proposed in literature; some as a complex gain [4], [5], [13] and others as a non linear Volterra series expansion [14]. This study assumes a Volterra series-based channel model. As such, the received symbols can be written as follows:

$$y(k) = h_0 x(k) + \sum_{i=1}^{L-1} h_i x(k-i) + \sum_{i,j,l=0}^{L-1} h_{i,j,l} x(k-i)x(k-j)x^*(k-l) + n(k) \quad (1)$$

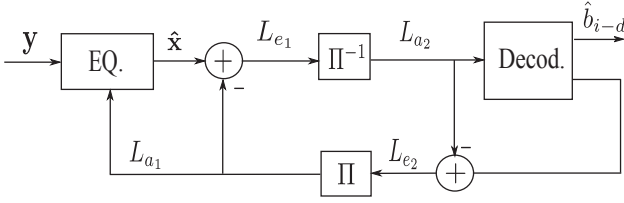


Fig. 2. Structure of the turbo equalizer.

where  $n$  is an additive white Gaussian noise (AWGN) with variance  $\sigma_n^2$  and  $h_i, h_{ijl}$  kernel coefficients of first and third order taking into account all chain filters.  $x_k$  is the transmitted symbol at time  $k$ ,  $L-1$  is the channel memory and  $(\cdot)^*$  stands for the conjugate of complex signal. The second and third RHS terms of equation 1 represent respectively the linear ISI and the non linear ISI generated by the nonlinear amplification of the HPA combined with chain filters memory. SISO equalizers/decoders proved efficiency by accepting a priori probability of received bits and generating new probabilities, a posteriori, depending on knowledge of the channel response/encoder structure. A turbo equalizer iteratively exchanges a priori information with a decoder (see figure 2). Both equalizer and decoder enhance the bit probability as the number of iterations increases until a convergence is reached. The probability of an encoded bit  $c_i$  is measured by the log likelihood ratio (LLR) defined by equation (2), [1]–[3] as:

$$L(c_i) = \ln \frac{p(c_i = 0)}{p(c_i = 1)} \quad (2)$$

Hence probabilities of  $c_i$  can be computed by:

$$p(c_i = 0) = \frac{e^{L(c_i)}}{1 + e^{L(c_i)}}, p(c_i = 1) = \frac{1}{1 + e^{L(c_i)}}$$

The equalizer receives a priori LLRs  $L_{a1}$ , uses them together with the received symbols to generate extrinsic LLRs  $L_{e1}$ . On the other hand, the decoder uses  $L_{a2}$  together with the channel code to generate extrinsic coded bits LLRs  $L_{e2}$  and information bits estimates. In this paper, we will consider trellis based codes using BCJR algorithm [15].

### III. RBF-BASED TURBO EQUALIZER

Radial basis function neural networks have gained much interest in engineering applications. They are used in system modelling and identification since they have a universal approximation property [16]. They are also used in classification and equalization where they show optimum decision property [10]. The RBF is structured in three layers: an input layer, a hidden layer and an output layer. The hidden layer of  $n_h$  neurons has radial activation functions centred around centres  $\mathbf{c}$ .

Let  $m$  be the equalizer memory. The expression of the equalizer input vector is  $\mathbf{y} = [y(k), y(k-1), \dots, y(k-m-1)]^T$  where  $y(k)$  are the noisy received symbols. From equation (1), we can write the Bayes rule as [8]:

$$p(\bar{\mathbf{y}}|\mathbf{y}) \propto p(\mathbf{y}|\bar{\mathbf{y}}) p(\bar{\mathbf{y}})$$

where  $p(\mathbf{y}|\bar{\mathbf{y}})$  is the probability of the vector  $\mathbf{y}$  given  $\bar{\mathbf{y}} = [\bar{y}(k), \bar{y}(k-1), \dots, \bar{y}(k-m+1)]^T$  which is the vector of noise-free channel output corresponding to a transmitted vector

of symbols  $\mathbf{x} = [x(k), \dots, x(k-L-m+2)]^T$ . The above equation can be rewritten in the form  $p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x}) p(\mathbf{x})$  since  $\bar{\mathbf{y}}$  corresponds to the vector  $\mathbf{x}$ . Henceforth, we use equality instead of  $\propto$  since the computation of LLR eliminates other constant multiplicative factors. We are interested in estimating  $x(k-d)$  given by the rule:

$$p(x(k-d)|\mathbf{y}) = \sum_{\tilde{\mathbf{x}}} p(\mathbf{y}|\tilde{\mathbf{x}}) p(\tilde{\mathbf{x}}) \quad (3)$$

with  $\tilde{\mathbf{x}} = \mathbf{x} \setminus x(k-1)$  is the vector  $\mathbf{x}$  without the entry  $x(k-1)$ . Probability  $p(y(k)|\bar{y}(k))$  follows normal distribution with zero mean and variance  $\sigma_n^2$ .

$$p(y(k)|\bar{y}(k)) = \frac{1}{\sqrt{2\pi\sigma_n^2}} e^{-|y(k)-\bar{y}(k)|^2/\sigma_n^2} \quad (4)$$

In the following we create the link to the RBF structure. The expression of the RBF output is:

$$y_{rbf} = \sum_{i=1}^{n_h} \phi(\mathbf{y}, \mathbf{c}_i) w_i \quad (5)$$

where  $w_i$  a weight of the output layer connected to hidden neuron  $i$  and  $\phi(\cdot)$  a radial function similar to the Gaussian function,

$$\phi(\mathbf{y}, \mathbf{c}_i) = e^{-|\mathbf{y}-\mathbf{c}_i|^2/\sigma} \quad (6)$$

with  $\sigma$  a parameter characterizing the width of  $\phi$ . Identifying equations (3)-(4) with equations (5)-(6), it becomes apparent that choosing  $\mathbf{c} = \bar{\mathbf{y}}$  and  $w_i = p(\tilde{\mathbf{x}}_i)$  allows the RBF to model the optimal decision rule.

Henceforth the RBF has a priori computed centres given the satellite model and the output weights are computed iteratively given the a priori LLRs (extrinsic of the decoder). In the first iteration, the a priori LLRs are set to zero or equivalently the a priori probabilities set to  $p(c_i = \mp 1) = 0.5$ . The computation of weights follows the rule given by:

$$w_i = p(\bar{y}_i(k), \bar{y}_i(k-1), \dots, \bar{y}_i(k-m+1)) \quad (7)$$

$$w_i = \prod_{j=0}^{m-1} p(\bar{y}_i(k-j)) = \prod_{j=0}^{L+m-2} p(x_i(k-j)) \quad (8)$$

$$w_i = \prod_{j=0}^{L+m-2} \left( \prod_{u=0}^{n_m-1} p(c_i(k-n_m j - u)) \right) \quad (9)$$

with  $n_m = \log_2(M)$  the number of bits per constellation symbol. In [10], the implementation of the RBF-Based equalizer is realized by  $M$  independent RBFs. The RBF  $i$  computes  $p(x(k-d) = s_i|\mathbf{y})$ ,  $i = 1, \dots, M$  where:

$$p(x(k-d) = s_i|\mathbf{y}) = p(s_i) \sum_{\tilde{\mathbf{x}}} p(\mathbf{y}|\tilde{\mathbf{x}}) p(\tilde{\mathbf{x}})$$

The number of hidden neurons in each RBF is given by  $n_h = M^{L+m-2}$ . Figure 3 illustrates the structure of one branch of the RBF-Based equalizer. A decision of the maximum  $p(x(k-d) = s_i|\mathbf{y})$  defines the best estimation of transmitted symbol  $x(k-d)$  when a Hard-Output equalizer is used.

Finally the extrinsic LLRs, noted by  $L_{e1}$ , will be computed by the following rule:

$$L_{e1}(c_i) = \ln \frac{p(c_i = 0|\mathbf{y})}{p(c_i = 1|\mathbf{y})} - L_{a1}(c_i) \quad (10)$$

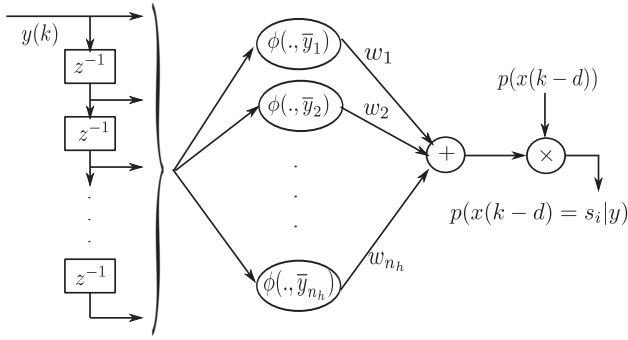


Fig. 3. The structure of a branch of the RBF equalizer.

The computation of  $\ln(p(c_i = 0|\mathbf{y})) - \ln(p(c_i = 1|\mathbf{y}))$ , will be detailed as follows:

$$p(c_i = 0|\mathbf{y}) = \sum_{j=1}^{n_m} p(x(n-d) = s_j|\mathbf{y}) \delta_i^j$$

where  $\delta_i^j = 1$ , iff  $c_i$  of  $s_j$  is equal to 0. It can be also rewritten as:

$$p(c_i = 0|\mathbf{y}) = \sum_{j=1}^{n_m} \left( p(x(n-d) = s_j) \sum_{l=1}^{n_h} p(\mathbf{y}|\bar{\mathbf{y}}_l) p(\tilde{\mathbf{x}}_l) \right) \delta_i^j \quad (11)$$

The computational complexity of the extrinsic LLR calculation can be reduced by using the generalized Jacobian algorithm [10]. Computation of  $\ln p(c_i = 0|\mathbf{y})$ , for instance, becomes:

$$\ln(p(c_i = 0|\mathbf{y})) = \ln \left( \sum_{j=1}^{n_m} \left( \sum_{l=1}^{n_h} e^{-\frac{\|\mathbf{y}-\bar{\mathbf{y}}_l\|^2}{2\sigma^2}} e^{\ln(p(\mathbf{x}_l))} \right) \delta_i^j \right) \quad (12)$$

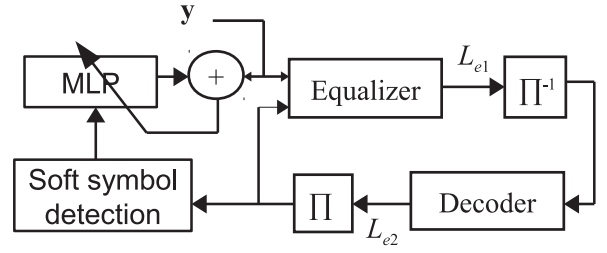
where  $\ln(p(\mathbf{x}_l)) = \ln(p(\tilde{\mathbf{x}}_l)) + \ln(p(x(n-d) = s_j))$ . Similarly, we can write:

$$\ln(p(c_i = 1|\mathbf{y})) = \ln \left( \sum_{j=1}^{n_m} \left( \sum_{l=1}^{n_h} e^{-\frac{\|\mathbf{y}-\bar{\mathbf{y}}_l\|^2}{2\sigma^2}} e^{\ln(p(\mathbf{x}_l))} \right) \delta_i^j \right) \quad (13)$$

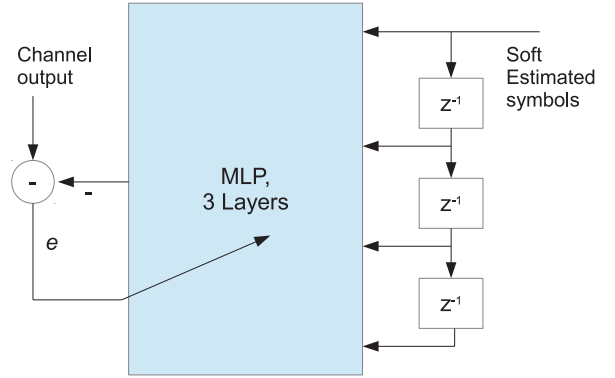
where  $\delta_i^j = 1$ , iff bit  $c_i$  of symbol  $s_j$  is equal to 1. Then the Jacobian algorithm  $J(\dots, J(\theta_3, J(\theta_2, \theta_1)))$  will be applied to compute equations (12)-(13) with:

$$J(\theta_2, \theta_1) = \ln(e^{\theta_2} + e^{\theta_1}) \\ = \max(\theta_2, \theta_1) + \ln(1 + e^{-|\theta_2 - \theta_1|}) \quad (14)$$

In [10], authors proposed to reduce the complexity of the RBF equalizer using feedback from the output of the RBF into the input layer. The feedback is used to select a subset of hidden neurons involved into the output computation. The reduced complexity algorithm considers the first two symbols of the vector  $\mathbf{x}$  known and detected on the RBF output. Finally, the number of hidden neurons used to compute each branch equals  $M^{L+m-FB-2}$  instead of  $M^{L+m-2}$ , with  $FB$  being the number of symbols fed back. For example, given simulation parameters of section V;  $L = 4$ ,  $m = 3$ ,  $FB = 2$  and  $M = 8$ , the feedback allows a reduction of the hidden neurons to  $8^3$  neurons for each RBF instead of  $8^5$ .



(a) Blind receiver



(b) Blind MLP

Fig. 4. Estimation of the satellite channel in the receiver.

#### IV. BLIND TURBO EQUALIZER

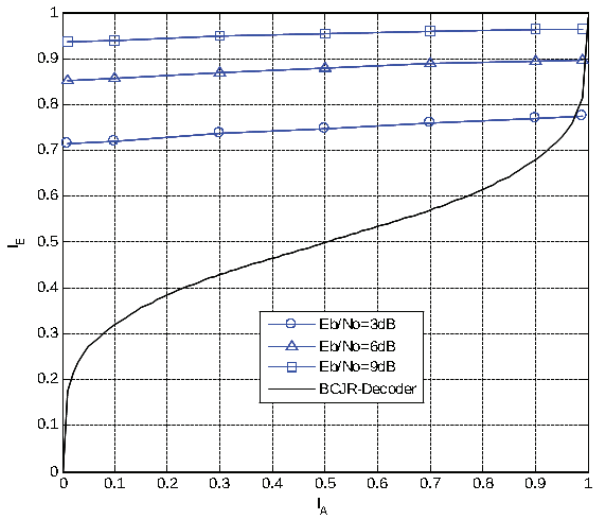
In blind equalization the centres of the RBF are iteratively updated based on an estimation of the channel model. Since the satellite channel is non linear, it should be modelled by a non linear tool like a NN. We choose to model the channel by a multilayer perceptron MLP with one input, one hidden and one output layer. A tapped-delay line is used at the MLP input to add a memory to the MLP (cf. figure 4(b)). Centres of the RBF-Based TEQ are computed using the MLP model. On line, MLP is implemented into the TEQ and trained using soft symbols  $\hat{x}$  estimated by the decoder output, after interleaving, as input of the MLP (Figure 4(a)). The output of the MLP  $\hat{y}$  is compared to the channel output  $y$  to produce the error signal  $e = y - \hat{y}$ , which is used by the backpropagation (BP) algorithm to adjust the MLP weights. Learning rules use a small learning rate  $\mu$ . After each iteration the centres of the RBF-TEQ are updated using the new MLP coefficients. The output of the MLP is computed by: The following equations present simplified learning rules of the hidden layer parameter  $w_h$  and output layer parameters  $w_o$  respectively:

$$w_o = w_o + \mu e \cdot y_h \\ w_h = w_h + \mu \delta \cdot \hat{x}$$

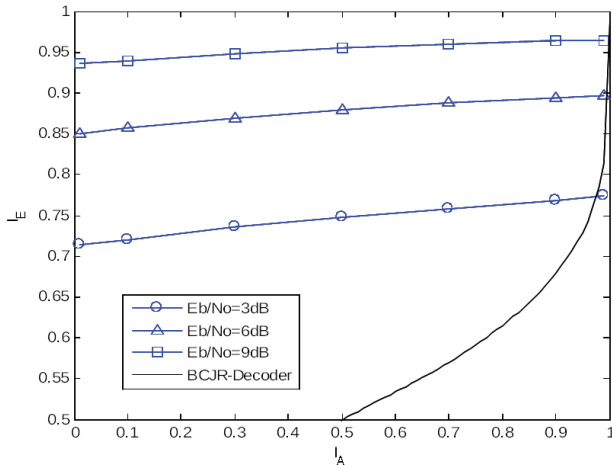
with  $\delta = e \cdot w_o \cdot f'$  the error of BP algorithm computed at the hidden layer, and  $f'$  is the derivative of the activation function.

#### V. SIMULATION RESULTS

We consider the Volterra filter as a model of the satellite nonlinear channel, with kernel coefficients given in table 1



(a) EXIT charts for RBF equalizers



(b) EXIT chart details

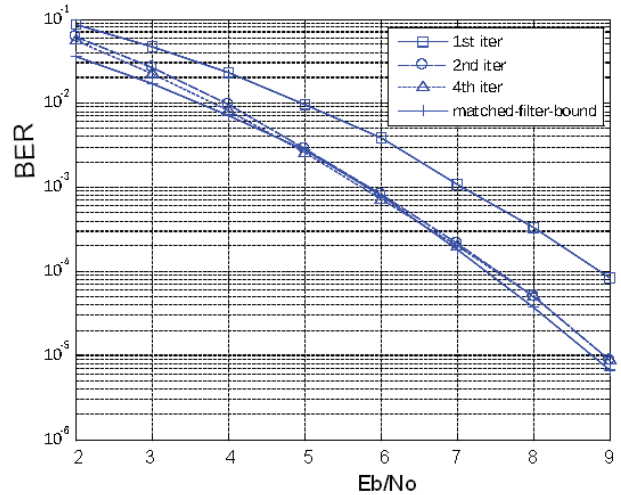
Fig. 5. EXIT Chart curves of the RBF-Based TEQ.

TABLE I. COEFFICIENTS OF VOLTERRA KERNELS

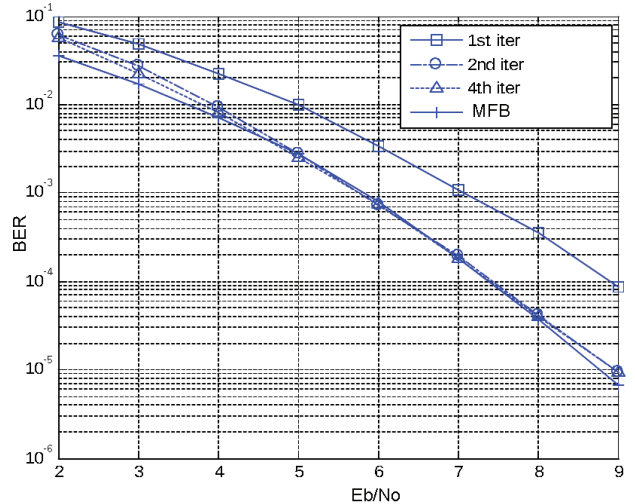
1st order	3rd order
$h_0 = 0.8529 + 0.4502j$	$h_{001} = 0.0979 - 0.0979j$
$h_1 = 0.0881 - 0.0014j$	$h_{002} = 0.1091 - 0.0615j$
$h_2 = -0.0336 - 0.0196j$	$h_{003} = -0.1119 - 0.0252j$
$h_3 = 0.0503 + 0.0433j$	$h_{110} = -0.0280 - 0.0475j$
	$h_{330} = 0.0503 - 0.0503j$

[11], [12]. The memory depth of the channel is  $L - 1 = 3$ , and we consider the equalizer of memory  $m = 3$  and a delay  $d = 2$ . Thus, the number of hidden neurons in each RBF is equal to  $M^{L+m-2}$  neurons, with  $M = 8$  the number of symbols in the modulation 8-PSK. The system uses a convolutional encoder with polynomials given in octal by (5, 7) with rate = 0.5 and a BCJR decoder with trellis termination [15].

Extrinsic information transfer (EXIT) chart [1] and [2] are used to analyse the performance of the equalizer. An EXIT Chart [17] plots the mutual information of the equalizer versus the mutual information of the decoder which is considered as a priori. The decoder output determines the value of the horizontal axis of the EXIT chart and the output of the equalizer



(a) Performance of blind equalizers



(b) Performance of non blind equalizer

Fig. 6. Performance of RBF-Based TEQ.

determines the value on the vertical axis. The computation of the mutual information takes into account the extrinsic of the equalizer and the decoder only. EXIT chart allows to analyse the performance of the iterative system. Figure 5 shows the EXIT chart within different conditions of noise ( $\frac{E_b}{N_0} = 3, 6$  and  $9\text{dB}$ ). Curves show the equalizer mitigation of both the linear and the nonlinear ISI since the equalizer output  $I_E$  increases monotonically when the a priori  $I_A$  increases.

Figure 6(a) shows the performance of the blind implementation of the reduced RBF-Based turbo equalizer. Centres of the RBF are deduced using the MLP. At the beginning, the MLP is sufficiently trained to approximate the channel model. It though continues to be trained inside the receiver blindly using soft estimation of transmitted symbols together with the channel output. A small learning rate, of order  $1e^{-4}$ , is used in learning rules to prevent the divergence of MLP. At the end of each iteration, the RBF centres are updated according to the new weights of MLP. The BER, in fig. 6(a), is close to the matched filter bound after few iterations. Although 2 iterations are sufficient to get close to the MFB, a higher iteration number

still remains useful to enhance the BER for  $\frac{E_b}{N_0}$  values smaller than 5dB. By comparing the two parts of figure 6, It appears the blind equalizer has almost as good performance as the non blind equalizer.

## VI. CONCLUSION

This paper presents an RBF-Based turbo equalizer for satellite non linear communication channel applications. We described the turbo equalisation and the blind turbo equalization for the non linear channel using neural networks. The non linear channel is modelled by a multilayer perceptron trained on line using soft symbols and the channel output. The derivation of soft input soft output representing extrinsic versus a priori has been done. The reduced complexity implementation has been presented and results outlined in the simulation. We plotted EXIT chart curves of the RBF-TEQ. They showed a monotonic increase of the extrinsic when a priori increases. This proves the ability of the RBF-equalizer to reduce ISI and to achieve the matched filter bound performance if iterative detection-decoding is used. The blind scheme of the reduced-complexity of RBF-Based TEQ, applied to the Volterra non linear channel, has good performance and BER is close to the MFB of the full RBF TEQ. BER converges to the MFB reached after almost 2 iterations.

## REFERENCES

- [1] M. Tuchler and A. Singer, "Turbo equalization: An overview," *Information Theory, IEEE Transactions on*, vol. 57, no. 2, pp. 920–952, 2011.
- [2] M. Tuchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *Communications, IEEE Transactions on*, vol. 50, no. 5, pp. 754–767, 2002.
- [3] R. Koetter, A. Singer, and M. Tuchler, "Turbo equalization," *Signal Processing Magazine, IEEE*, vol. 21, no. 1, pp. 67–80, 2004.
- [4] H. Abdulkader, F. Langlet, D. Roviras, and F. Castanie, "Natural gradient algorithm for neural networks applied to non-linear high power amplifiers," *International Journal on Adaptive Control and Signal Processing*, vol. 16, pp. 557–576, 2002.
- [5] S. Bouchired, M. Ibnkahla, D. Roviras, and F. Castanie, "Equalization of satellite umts channels using rbf networks," in *Personal, Indoor and Mobile Radio Communications, 1998. The Ninth IEEE International Symposium on*, vol. 3, 1998, pp. 1250–1254 vol.3.
- [6] B. Lu and B. Evans, "Channel equalization by feedforward neural networks," in *Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999 IEEE International Symposium on*, vol. 5, 1999, pp. 587–590 vol.5.
- [7] G. Kechriotis, E. Zervas, and E. Manolakos, "Using recurrent neural networks for adaptive communication channel equalization," *Neural Networks, IEEE Transactions on*, vol. 5, no. 2, pp. 267–278, 1994.
- [8] S. Chen, B. Mulgrew, and S. McLaughlin, "Adaptive bayesian equalizer with decision feedback," *Signal Processing, IEEE Transactions on*, vol. 41, no. 9, pp. 2918–2927, 1993.
- [9] M. Yee, T. Liew, and L. Hanzo, "Burst-by-burst adaptive turbo-coded radial basis function-assisted decision feedback equalization," *Communications, IEEE Transactions on*, vol. 49, no. 11, pp. 1935–1945, 2001.
- [10] M.-S. Yee, B. Yeap, and L. Hanzo, "Radial basis function-assisted turbo equalization," *Communications, IEEE Transactions on*, vol. 51, no. 4, pp. 664–675, 2003.
- [11] B. Benammar, N. Thomas, C. Poulliat, M.-L. Boucheret, and M. Dervin, "On linear mmse based turbo-equalization of nonlinear volterra channels," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 4703–4707.
- [12] D. Ampeliotis, A. Rontogiannis, K. Berberidis, M. Papaleo, and G. Corazza, "Turbo equalization of non-linear satellite channels using soft interference cancellation," in *Advanced Satellite Mobile Systems, 2008. ASMS 2008. 4th*, 2008, pp. 289–292.
- [13] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of twt amplifiers," *Communications, IEEE Transactions on*, vol. 29, no. 11, pp. 1715–1720, November 1981.
- [14] S. Benedetto, E. Biglieri, and R. Daffara, "Modeling and performance evaluation of nonlinear satellite links-a volterra series approach," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 4, pp. 494–507, July 1979.
- [15] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *Information Theory, IEEE Transactions on*, vol. 20, no. 2, pp. 284–287, Mar 1974.
- [16] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Computation*, vol. 3, no. 2, pp. 246–257, 1991.
- [17] J. Hagenauer, "The EXIT chart-Introduction to extrinsic information transfer in iterative processing," in *Proceeding 12th EUSIPCO*, 2004, pp. 1541–1548.