Chapter 12
A Novel Transform Domain Based Hybrid Recurrent Neural Equaliser for Digital Communication Channel

Susmita Das

Abstract Efficient neural network based adaptive equalisations for digital communication channels have been suggested in recent past. Recurrent neural network (RNN) exhibits better performance in nonlinear channel equalization problem. In this present work a hybrid model of recurrent neural equaliser configuration has been proposed where a Discrete Cosine Transform (DCT) block is embedded within the framework of a conventional RNN structure. The heterogeneous configuration on the RNN framework needs training and involves updation of the connection weights using the standard RTRL algorithm, which necessitates the determination of errors at the nodes of the RNN module. To circumvent this difficulty, an adhoc solution has been suggested to back propagate the output error through this heterogeneous configuration. Simulation study and bit-error-rate performance analysis of the proposed Recurrent Transform Cascaded (RTCS) equaliser for standard communication channel models show encouraging results.

Keywords Recurrent neural network · equaliser · bit error rate · discrete cosine transform · normalization

12.1 Introduction

Channel equalization is a powerful technique for compensating intersymbol interference in a dispersive communication channel, the nonlinearities introduced by the modulation/demodulation processes and the noise generated in the system. However, linear equalisers do not perform well on channels with deep spectral nulls or with nonlinear distortions. Researchers have shown that nonlinear equalisers based nonlinear theory exhibit better performance than linear equalisers in applications

S. Das (✉)
Department of Electrical Engineering, National Institute of Technology,
Rourkela – 769008 Orissa, India
e-mail: sdas@nitrkl.ac.in

S.-I. Ao and L. Gelman (eds.), *Electronic Engineering and Computing Technology*,
Lecture Notes in Electrical Engineering 60, DOI 10.1007/978-90-481-8776-8_12,

where the channel nonlinear distortions exist [1, 2]. When the channel itself has nonlinear characteristics or nonlinear channel distortions are too severe to ignore, even the Decision Feedback Equaliser cannot recover the corrupted signals effectively. Since neural networks (NN) [3] can perform complex mapping between its input and output space, and are capable of forming complex decision regions with nonlinear decision boundaries, many types of NNs have successfully applied in channel nonlinear equalization problem [2].The use of NN's is justified by noting that in most cases, the boundaries of the optimal decision regions are highly nonlinear, thus requiring the use of nonlinear classifiers, even with linear channels. Efficient neural network based adaptive equalisations for digital communication channels have been suggested in recent past. Different ANN architectures such as multilayer perceptron (MLP), radial basis function (RBF) etc. and many novel architectures and efficient training algorithms have been proposed in the literature [4]. Moreover structure selection for an ANN equaliser has always been a point of concern because a less complex structure is much easier to implement in real-time using VLSI, DSP chips etc. and also more suitable for typical applications like time varying channels in mobile communication system, optical recording media [5] etc.

Among the techniques based NN, Recurrent Neural Network (RNN) [6, 7] equalisers are proposed to solve the nonlinear channel equalization problem. RNN has shown better performance than feed forward neural network, because it approximates infinite impulse response (IIR) filter while feed forward neural network approximates FIR filter, which makes it attractive in the presence of channels with deep spectral nulls. In addition, RNN is more attractive for their small size [8]. Results from the simulations show that the RNE with simple size can yield a significant improvement in performance relative to the equalisers with linear filter, and outperform MLP equalisers of larger computational complexity in no minimum phase, partial response, and nonlinear channel equalizations cases. Complex versions of the RNE based on a real time current learning (RTRL) algorithm are developed to process complex signals [9]. Although various algorithms and hybrid structures [10, 11] have improved the performance of RNE, the computational burdens would become greater. In summary, the heavy computational load and low convergence speed have limited the practical applications of RNE.

In this paper, a hybrid configuration has been proposed where a Discrete Cosine Transform (DCT) block is embedded within the framework of a conventional RNE structure. A signal vector is mapped from a given domain to another when fed to a transform block and basically the transform block performs a fixed filtering operation. The basic difference between the transform block and the neural block is that while adaptive weights are associated with the later, fixed weights are inherent in the former. Hence, this cascaded network representing a heterogeneous configuration has been proposed to solve the conventional RNE problem keeping the complexity of the weight adaptation less. It is obvious that the transform block does not require any weight adaptation, but the RNN module needs updation of the connection weights using the standard RTRL algorithm, which necessitates the determination of errors at the nodes of the RNN module. To circumvent this difficulty, an adhoc solution has been suggested. The primary objective of the proposed work is to design cascaded RNE on reduced structural framework with faster convergence keeping in mind real-time implementation issue.

The organization of this paper is as follows. In Section 12.2, cascaded RNE equaliser based on the hybrid technique utilizing the modified version of the RTRL algorithm used to train it are described in detail. In Section 12.3, the performances of the proposed equaliser through various simulations for linear and nonlinear channels are illustrated. Finally, Section 12.4 summarizes the research work.

12.2 Proposed Hybrid Recurrent Neural Equaliser

A real-valued discrete cosine transform block followed by power normalization block is cascaded with an RNN module at the output end as given in Fig. 12.1. Power normalisation technique [9] is applied to the transformed signals and the final output of the proposed structure is evaluated as a weighted sum of all normalised signals. In order to update the connection weights of this cascaded framework, a novel idea has been developed based on propagation of the output error through the network in the light of the conventional BP algorithm. The transform block does not require any weight adaptation as it consists of fixed weights, but the RNN module needs updation of the connection weights using the standard RTRL algorithm, which necessitates the determination of errors at the nodes of the RNN module. But this estimate cannot be accomplished directly by using BP algorithm due to positioning of the transform block close to the output end, so problem is encountered here in propagating the final output error back into the network. To circumvent this difficulty, an adhoc solution has been evolved and error estimation at the input end of the transform block is done from the knowledge of the error at its output by considering its inverse transform. The mathematical expressions governing this concept are described in subsequent section.

12.2.1 Training Algorithm of Hybrid Neural Structure

The proposed structure shown in Fig. 12.1 consists of nr processing units in the RNN module with nx external inputs and a transform block. A step by step procedure has been adopted to update the weights of the neural network as mentioned below. Sensitivity parameters $\left\{p\,{}^{j}_{kl}\right\}$ of all RNN nodes are intialised to zero. The input signal to the proposed equaliser structure is represented by a *m x 1* vector $\boldsymbol{x}(n) = [r(n), r(n-1), \ldots \ldots, r(n-m+1)]^T$.

Input signal vector to the RNN module is defined as $u(n)$, lth element of which is

$$u_l(n) = \begin{cases} y_j(n), & 1 \le j \le nr \\ x_i(n), & 1 \le i \le nx \end{cases} \quad \text{for} \quad 1 \le l \le (nr + nx) \tag{12.1}$$

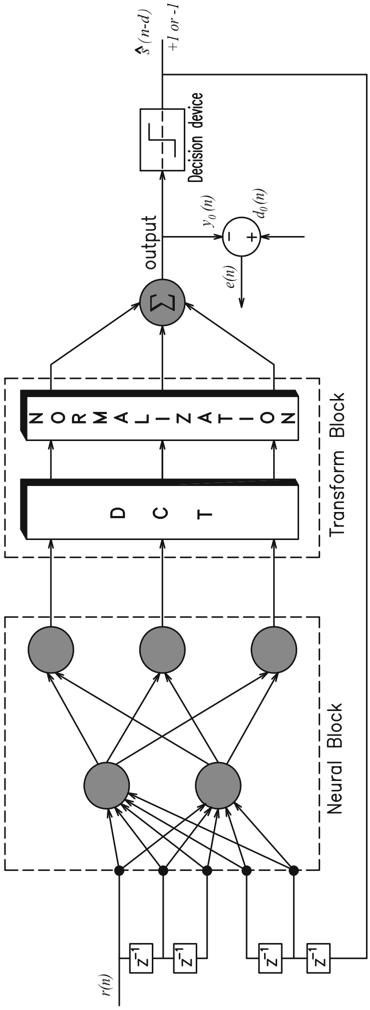

Fig. 12.1 Hybrid neural – transform equaliser structure

The output of jth neuron of the RNN module at time index n is given by

$$y_j(n) = \frac{1 - e^{-\phi . c_j(n)}}{1 + e^{-\phi . c_j(n)}} \tag{12.2}$$

Where the net internal activity is described by

$$c_j(n) = \sum_{l=1}^{nx+nr} w_{kl}(n) \cdot u_l(n), \quad 1 \le k \le nr \tag{12.3}$$

where W denotes *nr* by *(nx + nr)* weight matrix of the RNN module. Sigmoid activation functions ($\mathcal{F}$) with slope parameter ϕ for neurons of the RNN module have been considered. Input signal vector to the transform block can be expressed as *z(n)*, whose jth element is denoted as,

$$z_j(n) = y_j(n), j = nr \tag{12.4}$$

Here all the processing units of the RNN module act as visible units giving externally reachable outputs. The jth element of the output from the transform block (DCT) is defined as

$$z_{\mathcal{T}j}(n) = DCT\left\{z_j(n)\right\} = \mathcal{T}z_j(n) \tag{12.5}$$

The $\mathcal{T}_{\mathrm{pq}}$th element of the N X N transforms matrix $\mathcal{T}$ is defined as

$$\mathcal{T}_{pq} = \begin{cases} \dfrac{1}{\sqrt{N}}, & p = 0;\ q = 0, 1, \ldots\ldots, N-1 \\ \left(\sqrt{\dfrac{2}{N}}\right)\cos\dfrac{\pi(2q+1)p}{2N}, & p = 1, 2, \ldots. N-1;\ q = 0, 1, \ldots., N-1 \end{cases} \tag{12.6}$$

Transformed signal $y_{Tk}(n)$ is then normalised by the square root of their power $\mathcal{B}_j(n)$ which can be estimated by filtering the signal an exponentially decaying window of scaling parameter $\gamma \in [0, 1]$ as derived in the literature [12, 13] and shown below.

The jth element of the normalized signal becomes

$$z_{\mathcal{N}j}(n) = \frac{z_j(n)}{\sqrt{\mathcal{B}_j(n) + \varepsilon}} \tag{12.7}$$

and

$$B_j(n) = \gamma\, B_j(n-1) + (1-\gamma)z^2_{\mathcal{T}j}(n) \tag{12.8}$$

The small constant ε is introduced to avoid numerical instabilities when signal power $B_j(n)$ is close to zero.

The final output of the hybrid structure at time index n, y_o(n) is expressed as the weighted sum of all normalized signals from the transform blocks.

$$y_o(n) = \sum_{j=1}^{nr} g_j(n) \cdot z_{\mathcal{N}j}(n) \tag{12.9}$$

Where g denotes the weight matrix at the output end of the proposed network.

The error at the equaliser output at time index n is en by,

$$e(n) = d_o(n) - y_o(n) \tag{12.10}$$

With the knowledge of the output error, the errors at all the nodes of RNN module can be evaluated in order to facilitate the updation of weights using RTRL algorithm. But this is not possible directly as already explained before and hence a technique has been employed to tackle the situation.

At first the error e(n) is back propagated through various connection paths. Then the error at the jth output of normalization block is computed as given by

$$e_{\mathcal{N}_j}(n) = e(n) \cdot g_j(n),\ 1, \quad 1 \le j \le nr \tag{12.11}$$

The error terms at the output of the transform block $\delta_{\mathcal{T}j}(n)$ can be calculated using the following approach. The power normalisation can be considered as a process, whose operation is quite similar to the nonlinear transformation produced by sigmoid activation function of a neuron. This concept helps to calculate the error terms (i.e., local gradients) at the output of the transform block using the following equation

$$\begin{aligned} \delta_{\mathcal{T}j}(n) &= e_{\mathcal{N}j}(n)\ \frac{\partial y_{\mathcal{N}k}(n)}{\partial y_{\mathcal{T}k}(n)} \\ &= e_{\mathcal{N}j}(n)(y_{\mathcal{N}k}(n)/y_{\mathcal{T}k}(n)) \left\{1-(1-\gamma)y^2_{\mathcal{T}k}(n)\right\} \end{aligned} \tag{12.12}$$

Further, to propagate the error back through the transform block and to estimate the error magnitudes at the input side of the transform block, Inverse Discrete Cosine Transform (IDCT) is applied. This provides an estimate of the error at the input end of the transform block.

The error at the jth processing unit of the RNN module at time index n is given by

$$err_{rnn-node_j}(n) = IDCT\left\{\delta_{\mathcal{T}j}(n)\right\} \tag{12.13}$$

Application of RTRL algorithm involves primarily the evaluation of sensitivity parameter, a triply indexed set of variables $\left\{p^{\,j}_{kl}\right\}$ defined in literature [06].

$$p^j_{kl}(n) = \frac{\partial y_j(n)}{\partial w_{kl}(n)}, \quad k \in \mathbf{A} \text{ and } l \in \mathbf{A} \cup \mathbf{B}$$

where, A $= \{1, 2, \ldots, nr\}$ and B $= \{1, 2, \ldots, nf\}$.

The sensitivity parameters $\left\{p^{\,j}_{kl}\right\}$ are updated as follows

$$p^j_{kl}(n+1) = \mathsf{F}'\left\{c_j(n)\right\}\left[\sum_{i=1}^{nr} w_{ji}(n) \cdot p^i_{kl}(n) + \partial_{kj} u_l(n)\right] \tag{12.14}$$

where, $1 \leq j \leq nr$, *and* $1 \leq l \leq (nr + nx)$

$$\mathsf{F}'\{c_j(n)\} = \{1 - y_j(n+1)^2\}(\phi/2) \text{ and}$$

∂_{kj} is termed as *Kronecker delta* as given by,

$$\partial_{kj} = 1 \text{ for } j = k \text{ and zero otherwise.}$$

While the incremental weight change $\Delta g_j(n)$ is calculated using BP algorithm, RTRL algorithm computes the incremental weight change $\Delta w_{kl}(n)$.

$$\Delta g_j(n) = \theta \cdot e(n) \cdot z_{Nj}(n), \quad 1 \leq j \leq nr \tag{12.15}$$

$$\Delta w_{kl}(n) = \lambda \cdot \sum_{j=1}^{nr} err_{rnn-node_j}(n) \cdot p_{kl}^{j}(n), \quad 1 \leq k \leq nr \text{ and } 1 \leq l \leq (nr + nx) \tag{12.16}$$

where, λ and θ are learning-rate parameters of the RNN module and the output layer respectively.

The connection weights are updated as given below.

$$g_j(n+1) = g_j(n) + \Delta g_j(n) \tag{12.17}$$

$$w_{kl}(n+1) = w_{kl}(n) + \Delta w_{kl}(n) \tag{12.18}$$

The objective here is to minimise the cost function i.e. to change the weights in the direction that minimizes $J(n)$. The recursion process of updating weights of the cascaded network continues till a this predefined condition is achieved.

12.3 Simulation Study and Discussions

An exhaustive computer simulation study has been undertaken for evaluating the performance of all the proposed neural equaliser structures based on FNN topologies for a variety of linear and non-linear real communication channels models. The simulation model of an adaptive equaliser considered is illustrated in Fig. 12.2. In the simulation study the channel under investigation is excited with a 2-PAM signal, where the symbols are extracted from uniformly distributed bipolar random numbers $\{-1, 1\}$. The channel output is then contaminated by an AWGN (Additive White Gaussian Noise). The pseudo-random input and noise sequences are generated with different seeds for the random number generators. For mathematical convenience, the received signal power is normalised to unity. Thus the received

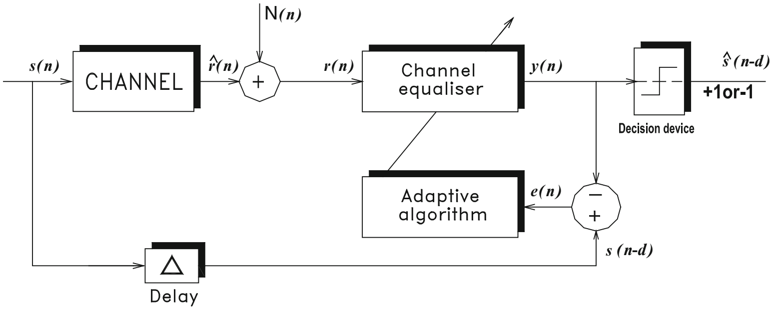

Fig. 12.2 Simulation model of channel equaliser in training phase

signal to noise ratio (SNR) is simply the reciprocal of the noise variance at the input of the equaliser. The power of additive noise has been taken as 0.01, representing a SNR of 20 dB.

Equalisation of different types of channel models (both linear and non-linear type) are attempted in order to establish the efficacy of the proposed equaliser structures based on RNN topology and to prove their robustness. It has been already reported in the literatures [6, 7], that a two-unit, one input, one output RNN is a non-linear IIR model which is sufficient to model many communication channels. Considering this aspect, all the proposed cascaded equalisers in RNN framework are compared with a conventional RNN equaliser (CRNN) with two recurrent units and one external input sample from the channel output. Further the TDRNN structure has two nodes in RNN module followed by a 2×2 DCT block with power normalisation and a summing unit at the output end.

For a comparative study and analysis purpose the number of training samples presented to the proposed equaliser considered here are restricted to 200 samples only as it is observed that their performances are quite satisfactory. The BER performance comparison of the proposed equaliser structures based on RNN topology has been carried out after all the structures has undergone a training phase (200 samples) The weight vectors of the equalisers are frozen after the training stage is over and then the performance test is continued. The BER performances for each SNR are evaluated, based on 10^7 more received symbols (test samples) and averaged over 20 independent realizations. All the proposed equalisers in RNN domain require fewer samples in training phase for satisfactory BER performance. Simulation results demonstrate this advantages offered by these structures. For the RTCS structure, the number of processing units remains the same as the CRNN equaliser. After the input signal is preprocessed in the RNN module, it is fed to the DCT transform block for further processing. As expected, such a proposed structure performs better than a CRNN due to the further signal de-correlation in the transform block followed by power normalization.

An example of a three tap channel characterized by

$$H_1(z) = 0.407 - 0.815z^{-1} - 0.407z^{-2} \tag{12.19}$$

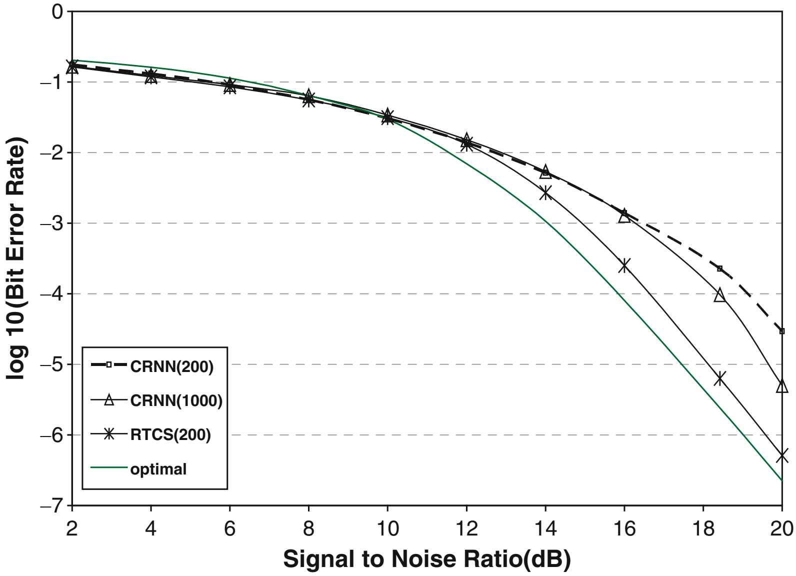

Fig. 12.3 BER performance of the proposed hybrid equaliser for channel $H_1(z)$

RTCS equaliser show distinct SNR gains of about 4.4 dB at a prefixed BER level of 10^{-4} over a conventional RNN equaliser which is quite encouraging (Fig. 12.3).

In order to prove the robustness and consistency in performance of all the proposed neural structures, equalisation of nonlinear channels is simulated. Such nonlinear channels are frequently encountered in several places like the telephone channel, in data transmission over digital satellite links, especially when the signal amplifiers operate in their high gain limits and in mobile communication where the signal may become non-linear because of atmospheric nonlinearities. These typical channels encountered in real scenario and commonly referred to in technical literatures [4, 6] are described by the following transfer functions.

$$H_2(z) = (1 + 0.5\,z^{-1}) - 0.9(1 + 0.5\,z^{-1})^3 \tag{12.20}$$

$$\begin{aligned} H_3(z) = {} & (0.3482 + 0.8704z^{-1} + 0.3482\,z^{-2}) \\ & +.2(0.3482 + 0.8704z^{-1} + 0.3482\,z^{-2})^2 \end{aligned} \tag{12.21}$$

For the nonlinear channel $H_2(z)$, the proposed RTCS equaliser results a significant 2 dB gain in SNR level at a prefixed BER of 10^{-4} over the CRNN equaliser in Fig. 12.4 which clearly justifies their application for such type of channel. RTCS equaliser in Fig. 12.5 shows distinct SNR gains of about 4 dB at a prefixed BER level of 10^{-4} over a conventional RNN equaliser in channel $H_3(z)$. For all the examples proposed structure performance is approaching the optimal Bayesian equaliser. Further it is noticed that increasing the number of training samples of the conventional RNN equaliser to 1,000 samples does not yield comparable performance.

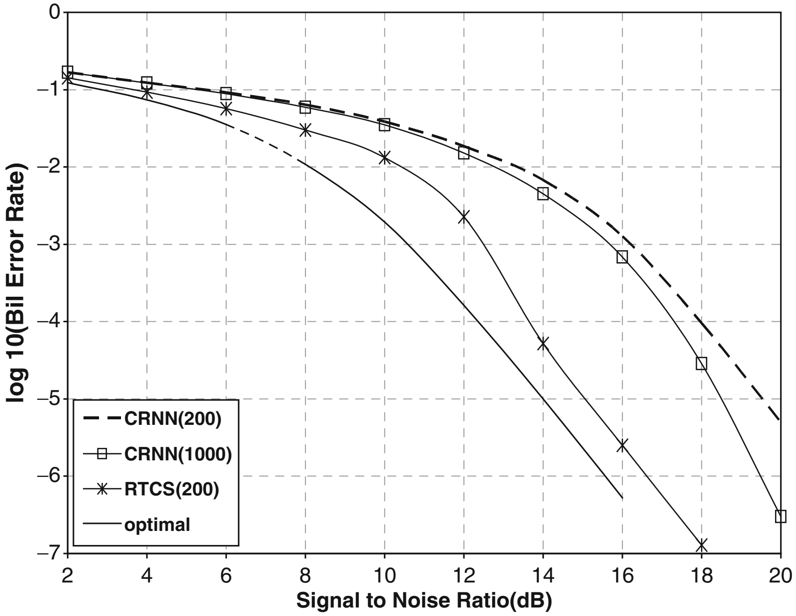

Fig. 12.4 BER performance of the proposed hybrid equaliser for channel $H_2(z)$

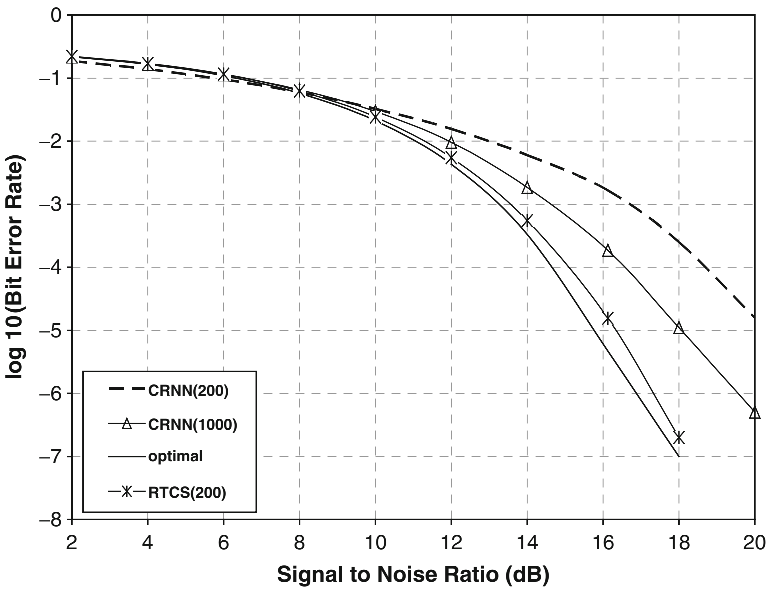

Fig. 12.5 BER performance of the proposed hybrid equaliser for channel $H_3(z)$

12.4 Conclusion

A real-valued transform is a powerful signal decorrelator which performs whitening of the signal by causing the eigen value spread of an auto-correlation matrix to reduce. The proposed neural equalisers with hybrid structures have outperformed their conventional counterparts to a large limit and require less number of samples in training phase simultaneously. The basic objective of this research of developing reduced network configurations remains and hence, while cascading is employed, it is ensured that under no circumstances this main purpose be defeated. It is interesting to note that recurrent neural structure with two nodes cascaded with a 2×2 DCT block with power normalization can outperform the conventional equaliser. As Bit Error Rate performance is a significant measure of channel equalization and proposed hybrid neural structure has an edge over conventional ones and even it is observed that it is close to the theoretically optimal Bayesian equalisers. Further a reduced structure has low computational complexity. Hence this hybrid ANN architecture has opened up new directions in designing efficient adaptive nonlinear equalisers and can be implemented in DSP processors for real – time applications.

References

1. Gibson, G.J., Siu, S., Chen, S., Cowan, C.F.N., Grant, P.M.: The application of nonlinear architectures to adaptive channel equalisation. Proceeding ICC'90 (Atlanta, GA), pp. 312.8.1–312.8.5, April 1990
2. Theodoridis, S., Cowan, C.F.N., Callender, C.P., See, C.M.S.: Schemes for equalisation of communication channels with non-linear impairments. IEEE Proc. Commun. **142**(3), 165–171 (June 1995)
3. Haykin, S.: Neural Networks – A Comprehensive Foundation. Macmillan, New York (1994)
4. Chen, S., Gibson, G.J., Cowan, C.F.N.: Adaptive channel equalisation using a polynomial perceptron structure. IEEE Proc. I Commun. Speech Vision **137**(5), 257–264 (October 1990)
5. Chen, C.N., Chen, K.H., Chiueh, T.D.: Algorithm and architecture design for a low complexity adaptive equaliser. IEEE Int. Symp. Circ. Syst. ISCAS'03 **2**, 304–307 (May 2003)
6. Kechriotis, G., Zervas, E., Manolakos, E.S.: Using recurrent neural networks for adaptive communication channel equalisation. IEEE Trans. Neural Networks **5**(2), 267–278 (1994)
7. Choi, J., Bouchard, M., Yeap, T.H.: Decision feedback recurrent neural equalization with fast convergence rate. IEEE Trans. Neural Network **16**(3), 699–708 (2005)
8. Parisi, R., Di Claudio, E.D., Orlandi, G., Rao, B.D.: Fast adaptive digital equalisation by recurrent neural networks. IEEE Trans. Signal Process **45**(11), 2731–2739 (November 1997)
9. Ortiz-Fuentes, J.D., Forcada, M.L.: A comparison between recurrent neural network architectures for digital equalisation. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-97) **4**, 3281–3284, 21–24, April 1997
10. Jiang, H.R., Kwak, K.S.: On modified complex recurrent neural network adaptive equaliser. J. Circ. Syst. Comput. **11**(1), 93–101 (2002)
11. Wang, X., Lin, H., Lu, J., Yahagi, T.: Combining recurrent neural networks with self-organizing map for channel equalisation. IEICE Trans. Commun. **E85-B**, 2227–2235 (October 2002)
12. Beaufays, F.: Transform-domain adaptive filters: an analytical approach. IEEE Trans. Signal Process. **43**(2), 422–431 (February 1995)
13. Marshall, D.F., Jenkins, W.K., Murphy, J.J.: The use of orthogonal transforms for improving performance of adaptive filters. IEEE Trans. Circ. Syst. **36**(4), 474–484 (April 1989)