Comparative Study on Non-Linear Artificial Neural Network Equalization Techniques

Vaishnavi Patel[1], Prof. Khyati Zalawadia[2]
[1, 2] Department of EC
[1, 2] BITs Edu Campus, Varnama, Vadodara

Abstract**-** *The field of digital data communications has experienced an explosive growth in the last three decades with the growth of internet technologies, high speed and efficient data transmission over communication channel has gained significant importance. The rate of data transmissions over a communication system is limited due to the effects of nonlinear distortion. Linear distortions occur in from of inter-symbol interference (ISI), co-channel interference (CCI) and adjacent channel interference (ACI) in the presence of additive white Gaussian noise. Different equalization techniques used to mitigate these effects. Adaptive channel equalizers used in digital communication systems. The equalizer located at the receiver removes the effects of ISI, CCI, burst noise interference and attempts to recover the transmitted symbols. It has seen that linear equalizers show poor performance, whereas nonlinear equalizer provide superior performance. Artificial neural network based multi-layer perceptron (MLP) based equalizers have used for equalization in the last two decades.*

The equalizer is a feed-forward network consists of one or more hidden nodes between its input and output layers and is trained by popular error based back propagation (BP) algorithm. However, this algorithm suffers from slow convergence rate, depending on the size of network. It has seen that an optimal equalizer based on maximum a-posterior probability (MAP) criterion can be implemented using Radial basis function (RBF) network. In a RBF equalizer, centers fixed using K-mean clustering and weights are trained using LMS algorithm. RBF equalizer can mitigate ISI interference effectively providing minimum BER plot. However, when the input order is increased the number of center of the network increases and makes the network more complicated. A RBF network, to mitigate the effects of CCI is very complex with large number of centers.

To overcome computational complexity issues, a single neuron based Chebyshev neural network (ChNN) and functional link ANN (FLANN) have proposed. These neural networks are single layer network in which the original input pattern expanded to a higher dimensional space using nonlinear functions and have capability to provide arbitrarily complex decision regions.

Keywords- Artificial neural network (ANN), MLP, FLANN, ChNN, RBF, Fussy systems, Wilcoxon neural network, channel equalization

I. INTRODUCTION

Artificial neural networks (ANNs) takes their name from the network of nerve cells in the brain. Recently, ANNs has been found to be an important technique for classification and optimization problem. Neural networks (NNs) have been extensively used in many signal processing applications. Linear & Nonlinear adaptive filters based on a variety of neural network models have been used successfully for system identification in a wide class of application. NNs have been most popularly applied to channel equalization of digital communication channels, in particular, due to their capacity to form complex decision regions. The mean square error(MSE) criterion, which is usually adopted in neural learning is of interest in the channel equalization.

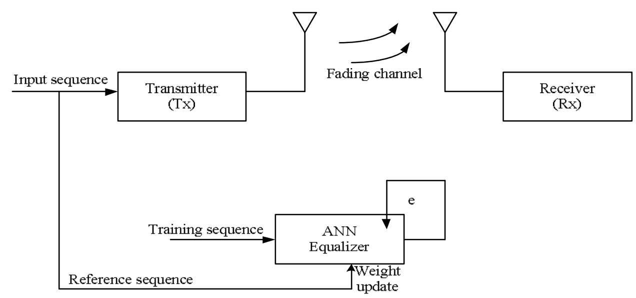

Fig. 1 Block diagram of ANN

ANNs are non–linear data driven & follows self-adaptive approach as opposed to traditional model based methods. They are powerful tools for modelling, especially when the underlying data relationship is unknown. ANNs can identify and learn correlated patterns between input data sets and corresponding target values. ANNs can be used to predict the outcome of new independent input data after training ANNs imitate the learning process of human brain & can process problems involving non-linear & complex data even if the data are imprecise and noisy.A very important feature of these networks is their adaptive nature, where “learning by example “replaces “programming” in solving problems this feature makes such computational model very appealing in application domain where one has little or incomplete understanding of the problem to be solved but where training data readily available. ANN is capable of performing nonlinear mapping between the input and output space due to its large parallel interconnection between different layers andthe nonlinear processing characteristics. Thus, they are ideally suited for the equalization of AWGN communication channel, which are noisy as well as non-linear.

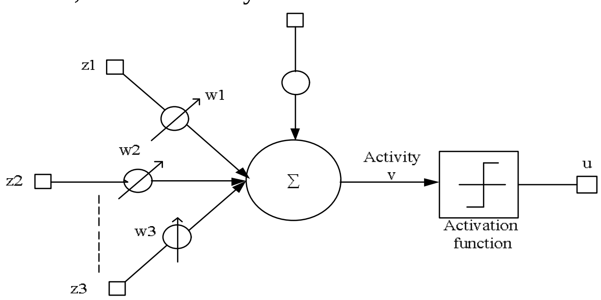

Fig. 2 Structure of a single Neuron

The terminology of ANNs has been developed from a biological model of the brain. A neural network consists of a set of connected cells: The neurons. The neurons receive input from either input cells or other neurons and perform some kind of transformation of the input & transmit the outcome to other neurons or to output cells. The neural networks are built from layers of neurons connected so that one layer receives input from the preceding layer of neurons and passes the output on the subsequent layers. Each neuron is associated with three parameters whose learning can be adjusted; these are the connecting weights, the bias and the slope of nonlinear function. A feed forward structure with input, output, hidden layers and nonlinear sigmoid functions are used in this type of network. The basic structure of an artificial neuron is presented in Fig.2.The operation in a neuron involves the computation of the weighted sum of the inputs and threshold. The resultant signal is then passed through a nonlinear activation function. This is also called a perceptron, which is built around a non-linear neuron.

II. ANN EQUALIZATION TECHNIQUE

A. Multilayer Perceptron Network:

Email address is compulsory for the corresponding author. In 1958, Rosenblatt demonstrated some practical applications using the perceptron. The perceptron is a single level connection of McCulloch-Pitts neurons is called as Single-layer feed forward network*s*. The network is capable of linearly separating the input vectors into pattern of classes by a hyper plane. Similarly, many perceptron can be connected in layers to provide a MLP network, the input signal propagates through the network in a forward direction, on a layer-by-layer basis. This network has been applied successfully to solve diverse problems.

Generally, MLP is trained using popular error back-propagation algorithm. The scheme of MLP using four layers is shown in Fig.3 represent the inputs s1, s2… s0 to the network, and represents the output of the final layer of the neural network. The connecting weights between the input to the first hidden layer, first to second hidden layer and the second hidden layer to the output layers are represented by respectively.

$$y_k = \psi_k \left[\sum_{k=1}^{P_2} w_{kj}\, \psi_j \left(\sum_{j=1}^{P_1} w_{ji}\, \psi_i \left\{ \sum_{i=1}^{n} w_i\, s_i + b_i \right\} + b_{jl} \right) + b_k \right]$$

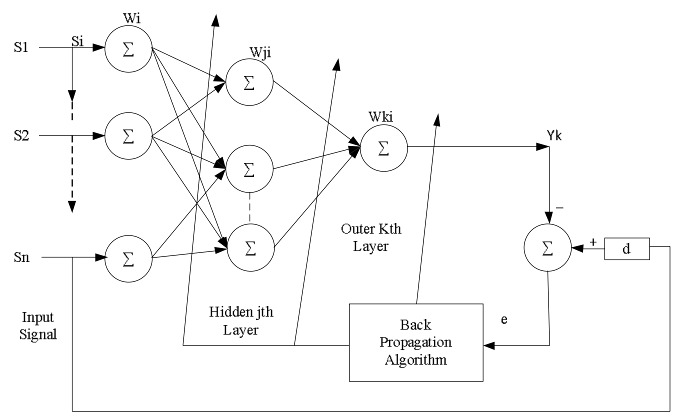

Fig. 3 MLP network block diagram

Where, P1, P2 and P3 are the number of neurons in the layer. *bi,* and $b_k$is the threshold to the neurons of the layer, *n* is the number of inputs and is the nonlinear activation function respectively. Most popular from of activation functions for signal processing application are sigmoid and the hyperbolic tangent since there are differentiable.
The time index *t* has been dropped to make the equations simpler.

B. Function Link Artificial neural network:

FLANN is a novel single layer ANN network in which the original input pattern is expanded to a higher dimensional space using nonlinear functions, which provides arbitrarily complex decision regions by generating nonlinear decision boundaries. The main purpose of enhanced the functional expansion block to use for the channel equalization process.

$$s_i = \begin{cases} x_k, & i = 1 \\ x_k^l, & i = 2,3,4,\ldots\ldots\ldots, M \end{cases}$$

Where, l= 1, 2….M for trigonometric expansion,

$$s_i = \begin{cases} x_k, i = 1 \\ \sin(l\pi x_k), \quad i = 2,4,\ldots, M \\ \cos(l\pi x_k), \quad i = 3,5,\ldots, M+1 \end{cases}$$

Where, l= 1, 2….M/2. In matrix notation the expanded elements of the input vector E, is denoted by **S** of size $N \times (M+1)$.

The bias input is unity. So, an extra unity value is padded with the **S** matrix and the dimension of the **S** matrix becomes $N \times Q$, where Q = (M+2).

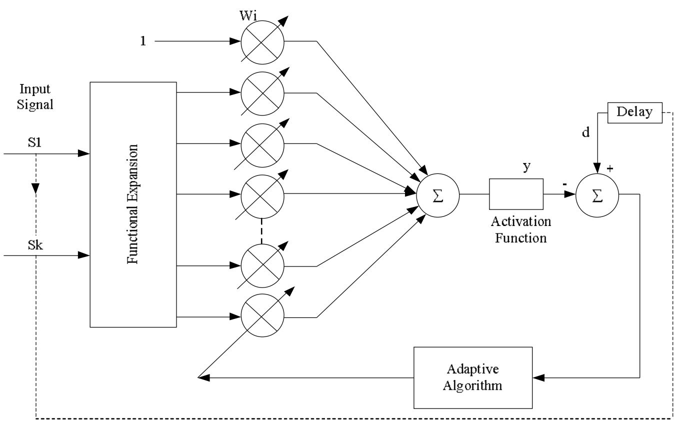

Fig. 4 FLANN block diagram

C. Chebyshev Artificial Neural Network:

Chebyshev artificial neural network (ChNN), it is similar to FLANN. The difference being that in a FLANN the input signal is expanded to higher dimension using functional expansion. In Chebyshev the input is expanded using Chebyshev polynomial. The Chebyshev polynomials generated using the recursive formula given as

$$s_{n+1} = 2_x s_n(x) - s_{n-1}(x)$$

The first few Chebyshev polynomials are given as

$$s_0(x) = 1$$
$$s_1(x) = x$$
$$s_2(x) = 2x^2 - 1$$
$$s_3(x) = 4x^3 - 3x$$

The weight vector represented as, here i = 0, 1, 2 . . . n. The weighted sum of the components of the enhanced input is passed through a hyperbolic tangent nonlinear function to produce an output.

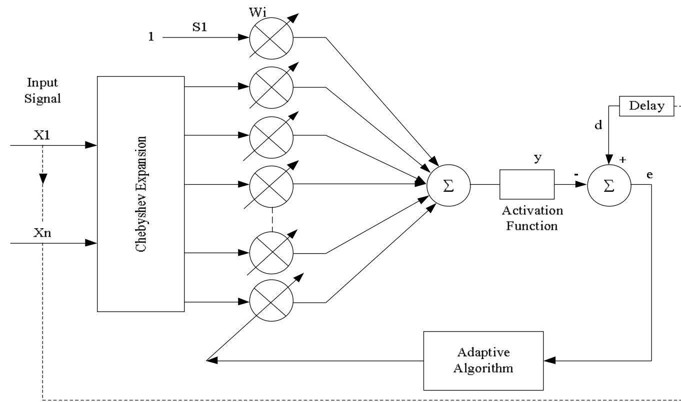

Fig. 5 ChNN block diagram

The advantage of ChNN over FLANN is that the Chebyshev polynomials are computationally more efficient than using trigonometric polynomials to expand the input space.

D. Radial Basis Function Network:

The RBF network was originally developed for interpolation in multidimensional space. The schematic of this RBF network with *m* inputs and a scalar output is presented in Figure. This network can implement a mapping F_{rbf}: R^m -> R by the function.

$$y = F_{rbf}\{s\} = \sum_{i=1}^{n} w_i \psi(\|s_i - c_i\|)$$

Where S R^m is the input vector $s_i = [s_1, s_2 \ldots \ldots s_n]^T | \epsilon R^n$, $\psi(.)$is the given function from R+ to R, are weights and, $c_i \epsilon R^m$are known as RBF centres. The centres of the RBF networks are updated using k-means clustering algorithm. This RBF structure can be extended for multidimensional output as well. Gaussian kernel is the most popular form of kernel function for equalization application, it can be represented as

$$\psi(y) = \exp(y / -\sigma_r^2)$$

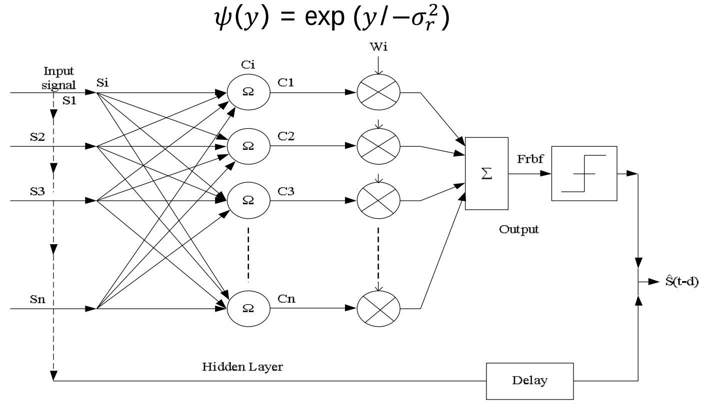

Fig.6 RBF block diagram

Here, the parameter σ_r^2controls the radius of influence of each basis functions and determines how rapidly the function approaches 0 with. In equalization applications the RBF inputs are presented through a TDL. Training of the RBF networks involves setting the parameters for the centers Ci, spread $\sigma_r$and the linear weights ω_iRBF spread parameter σ_r^2, is set to channel noise variance σ_η^2this provides the optimum RBF network as an equalizer. The RBF networks are easy to train since the training of centers, spread parameter and the weights can be done sequentially and the network offers a nonlinear mapping, maintaining its linearity in parameter structure at the output layer.

One of the most popular schemes employed for training the RBF in a supervised manner is to estimate the centers using a clustering algorithm like the k-means clustering and settingσ_r^2 to an estimate of input noise variance calculated from the center estimation error. The output layer weights can be trained using popular stochastic gradient LMS algorithm.

The RBF equaliser can provide optimal performance with small training sequences but they suffer from computational complexity. The number of RBF centres required in the equaliser increases exponentially with equaliser order and the channel delay dispersion order. This increases all the computations exponentially.

E. Fuzzy Systems:

The first step in the application of fuzzy logic to any problem is the formulation of input membership functions which transform the system inputs into fuzzy variables. The input fuzzy variable is then mapped into a fuzzy output variable by means of rules which describe the relation between these variables. Both membership functions and rules are usually determined on a heuristic basis and are problem dependent. For example, for the equalizer problem considered here, we note that even in the absence of interchannel interference orchannel noise, the channel usually introduces a delay. Thus, if the channel delay is k samples, the received signal at any time n will be a (n — k). Since the channel delay is usually unknown, if we assume that the delay is I samples, we may formulate a rule of the form, "if the channel output (received signal) delayed by I samples, u (n — l), is positive, then the corresponding channel input, — k) was positive," where I and k are nonnegative numbers. In the presence of either intersymbol interference or observation noise, the output may be either positive or negative irrespective of whether a positive or a negative signal was transmitted, so that a rule may also be of the form, "if channel output delayed by I samples is positive, then the corresponding input was negative." Thus a set of rules can be associated with each assumed value for the delay at the output. For each decision, a minimum of N sets of such rules must be used, where each set corresponds to an output with different delay and N is the maximum delay associated with the channel. If no information about the channel is available, then N is chosen arbitrarily. The input to each of the rules is a fuzzy value determined by the membership function.

The membership functions for the equalizer input can be derived from the training data set as explained below. Suppose that a test sequence n) consisting of positive and negative values corresponding to a one or a zero (here assumed to be 0.9 and —0.9) is transmitted over a channel, with an assumed maximum delay of N. The desired output at time n from the equalizer will be set as a(n — k), where k is an integer greater than N. We obtain plots of the delayed received signal u(n — l) versus a(n — k), the desired equalizer output, for each value of I in the range [O, L], where L is an integer that lies between N and k, N < L < k. Fig. 3 shows an example of such a plot of the values of the desired output versus the values of the received signal for the simulation example considered in Section Ill, for a particular training sequence and with the channel delay assumed to be a specific value, namely I = 1. Note that the desired output is either +0.9 or —0.9, while the received signal may take any value because of intersymbol interference.

If the channel delay is known, then — L = k. typically, however, the maximum delay is unknown, and the values have to be chosen experimentally based on the information contained in u (•n — I) about — k). In Section Ill we suggest a method to choose these values.

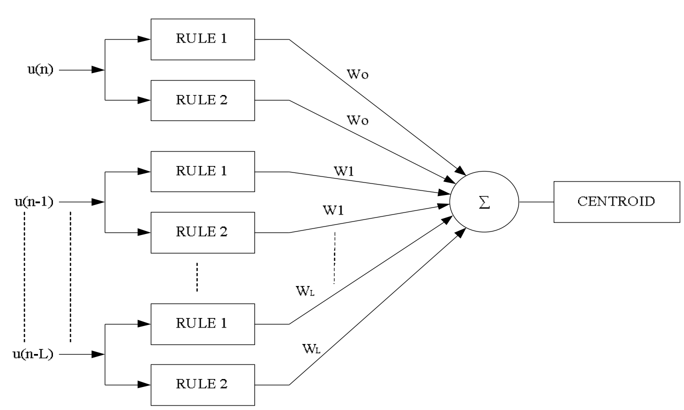

Fig. 7 Fuzzy system block diagram

The characteristic function XA of a set A by,

$$X_A(x) = f(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

Which is an indicator of members and non-members of the crisp set A. In the case that an element has only partial membership of the set, we need to generalize this characteristic function to describe the membership grade of this element in the set: larger values denote higher degrees of the membership.

F. Wilcoxon Neural Network:

The Wilcoxon neural network consists of Multilayer Perceptron Neural Network trained with Wilcoxon learning method, is named as Wilcoxon Multilayer Perceptron Neural Network (WMLPNN). For equalization WMLPNN, has one input layer with n+1 nodes, one hidden layer with m+1 nodes, and one output layer with one nodes,

$$x = [x_1, x_2 \ldots \ldots x_n]^T \in R^n$$
$$S = [S_1, S_2 \ldots \ldots S_n, S_{n+1}]^T$$
$$= [X_1, X_2 \ldots \ldots X_n 1]^T \in R^{n+1}$$

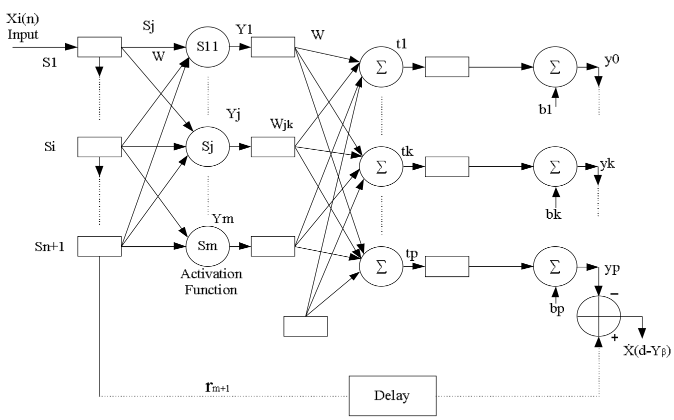

Fig. 8 Wilcoxon neural network

III. CONCLUSION

The main aim of the paper is to develop novel artificial neural network equalizer (trained with linear, nonlinear and evolutionary algorithms) to mitigate the linear and nonlinear distortion like ISI, CCI and burst noise interferences occurs in the communication channel and can provide minimum mean square error and bit-error-rate plot for wide variety of channel condition.

This paper analyses in details the performance of different types of nonlinear ANN based equalizer like MLP, RBF, FLANN, ChNN and fuzzy system algorithm for channel equalization in digital communication system. Their performance compared with proposed Wilcoxon neural network equalizer. Through extensive simulation study, observed that the proposed Wilcoxon learning algorithm trained neural network equalizer performed similar as RBF equalizer in ISI and high intensity burst noise interference condition, but its perform better in CCI environment than RBF equalizer. In addition, WNN equalizer performs better than MLP, in all ISI, CCI and burst noise environment. RBF equalizer provides MAP decision performance.

ACKNOWLEDGMENT

It is honour and pleasure to express my heartfelt gratitude to those who helped me and contributed towards the preparation of this review paper. I am indebted to my guide Prof. KHYATI ZALAWADIA whose invaluable guidance and timely suggestion and constructive encouragement inspired me to complete the project in the present form. I express my thanks to the Library of BITS Edu Campus, which is a source of such invaluable information and of course the Internet Facility of the same. I would like to thank to the entire team of M.E. Staff whose direct and indirect suggestion helped me creating this project. I would like to pay a special thanks to my parents for the sparing their invaluable time and inspiring me. Although there remain some names but none are remain un-thanked.

REFERENCES

[1] S. U. H. Qureshi, “Adaptive Equalization,” IEEE, vol. 73, pp. 1349–1387, September 1985

[2] S. Haykin, Neural Networks - A Comprehensive Foundation. New York: Macmillan, 2006.

[3] William J. Ebel, and William H. Tranter,” The Performance of Reed-Solomon Codes on a Bursty-Noise Channel”, IEEE Transactions on communications, Vol. 43, no. 2/3/4, February/March/April 1995.

[4] S. Haykin, Adaptive Filter Theory. Englewood Cliff, NJ, USA: Prentice Hall, 1991.

[5] R.W. Lucky, “Techniques for adaptive equalization of digital communication systems," Bell System Tech. J., pp. 255-286, Feb 1966.

[6] Gibson, G.J., S.Siu and C.F.N.Cowan,”Application of multilayer perceptrons as adaptive channel equalizer,” IEEE Int. Conf. Acoust., Speech, Signal Processing, Glasgow, Scotland, pp.1183-1186, 1989

[7] M. Meyer and G. Pfeiffer, "Multilayer perceptron based decision feedback equalizers for channels with intersymbol interference,”. IEE, Pt IVol 140, No 6, , pp 420-424 Dec 1993.

[8] B. Mulgrew, “Applying Radial Basis Functions,” IEEE Signal Processing Magazine, vol. 13, pp. 50–65, March 1996.

[9] S. Chen, S. McLaughlin, and B. Mulgrew, “Complex-valued Radial Basis Function Network, Part I: Network Architecture and Learning Algorithms,” Signal Processing (Eurasip), vol. 35, pp. 19–31, January 1994.

[10] S. Chen, B. Mulgrew and S. Mc Laughlin, “Adaptive Bayesian Equaliser with Decision Feedback”, IEEE Transactions on Signal Processing, vol.41, pp. 2918-2927, September 1993.

[11] S. Haykin, Digital Communications, John Wiley and Sons, 2006.

[12] M.J. Bradley and P. Mars, “Application of Recurrent Neural Networks to Communication Channel Equalisation”, International Conference on Acoustics, Speech and Signal Processing, ICASSP-95, vol. 5, pp. 3399-3402, 9-12 May 1995.

[13] J.D. Ortiz-Fuentes and M.L. Forcada, “A Comparison between Recurrent Neural Network Architectures for Digital Equalisation”, IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP-97, vol. 4, pp. 3281-3284, 21-24 April 1997.

[14] M. T. O zeden, A. H. Kayran, and E. Panayirci, “Adaptive Volterra Channel Equalisation with Lattice Orthogonalisation,” IEE Proceedings - Communication, vol. 145, pp. 109–115, April 1998.

[15] J. Patra, R. Pal, B.N.Chatterji, and G. Panda, “Identification of nonlin-ear dynamic systems using functional link artificial neural networks," IEEETransactions on Systems,Man and cybernetics, vol. 29, no. 2, pp. 254-262,April 1999.

[16] J. C. Patra and A. C. Kot, "Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks" IEEE Tran. SMC, vol.32, pp. 505-51 1, Aug 2002.

[17] R. V. Hogg, J. W. McKean and A. T. Craig, Introduction to Mathematical Statistic, 6th edition, Englewood Cliffs, NJ: Prentice-Hall, 2005. [18]. S. Wan and L. E. Banta, “Parameter incremental learning algorithm for neural networks”, IEEE transaction, Neural Network, vol.17, no. 6, pp. 1424-1438, November 2006.

[18] S. Mishra “A Hybrid Least Square-Fuzzy Bacterial Foraging Strategy For Harmonic Estimation”, IEEE Transactions On Evolutionary Computation, Vol. 9, No. 1, February 2005, pp 61-73.

[19] B. W. Lee and B. J. Sheu, “Parallel Hardware Annealing for Optimal Solutions on Electronic Neural Netw’orks,” IEEE Transactions on Neural Networks, vol. 4, pp. 588–598, July 1993