Performance Enhancement Of OFDM Using Intelligent System

Jinan N. Shehab
Asst. lect. Communication Department , College of Engineering, University of Diyala
Eng_jinan83@yahoo

الخلاصة: أن ارتفاع معدل نقل البيانات في بيئة المحمول يجعل القناة ذات تأثير سلبي للغاية . ولتقليل هذه المشكلة ، تم اقتراح وتطوير عدة أنظمة. نظام التقسيم الترددي المضاعف المتعامد (OFDM) هو احد التقنيات لمعالجة هذه القناة. في هذا العمل، تم افتراض طريقه لتحسين اداء تخمين القناة في نظام التقسيم الترددي المضاعف المتعامد وذلك باستخدام عدة انواع من خوارزمية الانتشار العكسي لتعليم الشبكة العصبية الاصطناعية لتقليل الخطأ الذي يحصل اثناء نقل البيانات ما بين المرسل والمستلم في نظام التقسيم الترددي المضاعف المتعامد. تتضمن الطريقة المقترحة، تدريب الشبكة العصبية الاصطناعيه ذات التغذية الامامية (FNN) و الشبكة العصبية الاصطناعيه ذات الاسترجاع المتكرر (RNN) باستخدام خوارزمية(Conjucate Gradient algorithm)، Quasi-Newton algorithm (BFGS algorithm) Baysiyan regularization and لتعليم الشبكة العصبية الاصطناعية . المقارنة بين هذه الخوارزميات يعتمد على (MSE) لهذه الخوارزميات اثناء تدريب الشبكة العصبية الاصطناعية و BER المحسوبه. تم استخدام برنامج الماتلاب للحصول على النتائج.

الكلمات المفتاحية: OFDM، الشبكات العصبية الاصطناعية (ANN)، الانتشار الرجعي (BP).

Abstract: Transmission of high data rate in a mobile environment makes the channel highly hostile. To combat with this problem, many techniques were proposed and developed. Orthogonal frequency division multiplexing (OFDM) system is a technique to combat this adverse channel. In this work, a method to enhance the performance of channel estimation in Orthogonal Frequency Division Multiplexing (OFDM) is proposed by using different types of Back-Propagation (BP) for learning the Artificial Neural Network (ANN) to minimize Bit Error Rate (BER) when transmitting data. The proposed method includes learning Feed Forward Neural Network (FNN) and Recurrent Neural Network (RNN) by Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization. The comparison among Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization depends on the Mean-Square Error (MSE) convergence and precision generated in the BER calculation. This work is software implemented with MATLAB (R2013a) technical programming language.

Keywords: OFDM, Artificial Neural Network (ANN), Back-Propagation (BP).

1. Introduction

The next generation communication systems require transmission data at high rate and high equality of service. To realize these requirements, Orthogonal frequency division multiplexing has drawn explosive attention as a new type of high data rate transmission scheme for wireless communication system. OFDM allow high data rates to transmit over broadband channel due to the spectral bandwidth efficiency, robustness to the multipath delay [1].

In a mobile radio channel the transmitted signal is distorted and attenuated during transmission through the frequency and time selective. In order to reduce BER in OFDM systems, the estimation of channel is necessary before the demodulation fading channel [2]. Channel Estimation is defined as finding out how much data can be transmitted over a channel without any interference. It is considered as one of the challenging task where the Channel State Information (CSI) is required for coherent detection of the data at the receiver [3]. There are several techniques for channel estimation in OFDM system. Among these techniques, both pilot-based channel estimation and blind channel estimation techniques are most popular. In pilot based channel estimation algorithms, training symbols or pilot tones that are known a priori to the receiver, are multiplexed along with the data stream for channel estimation. The training-based method channel estimation can be performed by either block type pilots where pilot tones are inserted into all frequency bins within periodic intervals of OFDM blocks or by comb pilots where pilot tones are inserted into each OFDM symbol symbols with a specific period of frequency bins [4].

This paper proposes a technique, which based on artificial neural networks, that is one type of intelligent system to enhance the performance of channel estimation in OFDM. Back propagation training algorithms have been analyzed for training Feed Forward Neural Network (FNN) and Recurrent Neural Network (RNN).

2. Related Works

There are several works research on OFDM channel prediction or estimation with ANN and most of these work are either frequency synchronization or on combating Inter Symbol Interference (ISI) due to channel impairment. Further, because of characteristics of the ANN's are not linear, these networks of different architecture have found successful application in channel estimation problem. One of the earliest applications of the ANN in digital communication channel estimation is reported by E. Chen, et. al [5], proposed a novel equalizer based on the complex-domain back propagation (BP) algorithm to improve the performance of the equalizer in OFDM system. Gowrishankar and P.S.Satyanarayana investigated the prediction of Bit Error Rate (BER) in OFDMA Channel [6]. M. M. A. Moustafa and S. H. A. El-Ramly, proposed using Back Propagation Neural Networks (BPNNs) for channel estimation and equalization for OFDM systems. They used a steepest descent algorithm as a standard Back Propagation (BP) algorithm in flat fading [7]. Chia Hsin Cheng, et. al, combined a BPNN for channel estimation and compensation of signals with a genetic algorithm [8].

3. Modeling of OFDM System

The baseband OFDM system based on pilot channel estimation is shown in figure (1).On the transmitter side, binary information is mapped depending on chosen modulation. After serial/parallel (S/P) conversion pilots are inserted either to all subcarriers with a specific period or uniformly between the information data. The modulated data X(k) is converted into a time domain signal by taking the N point IFFT. After IFFT, the time domain signal is given by following equation [9]:

$$x(n) = IFFT\big(X(k)\big), n = 0, 1, 2 \ldots, N-1$$

$$= \sum_{k=0}^{N-1} X(k)e^{j2\pi kn/N} \quad \cdots (1)$$

where N is the length of FFT, X(k)is baseband data sequence. After IFFT, cyclic prefix is inserted to prevent ISI. This interval should be chosen to be more than the expected delay spread of the multipath channel. The guard time includes the cyclically extended part of the OFDM symbol in order to delete the inter-carrier interference (ICI). The symbol extended

$$x_f(n) = \begin{cases} x(N+n), n = -N_g, -N_g+1, \ldots, -1 \\ \quad x(n) \quad , \quad n = 0,1,\ldots,N-1 \end{cases} \quad \cdots (2)$$

where $x(n)$ is data bit, N number of subcarriers and N_g is the length of the guard interval. The transmitted signal $x_f(n)$ will pass through the frequency selective time varying fadingchannel with Additive White Gaussian Noise (AWGN). The received signal is given by following equation:

$$y_f(n) = x_f(n) \otimes h(n)w(n) \quad \cdots (3)$$

where $h(n)$ is the impulse response of the frequency selective channel and $w(n)$is AWGN.

The channel response *h(n)* can be represented by:

$$h(n) = \sum_{k=1}^{K} a_k \, sinc\left[\frac{\tau_k}{T_{samp}} - n\right] \quad \cdots (4)$$

Where,

- T_{samp} is the input sample period to the channel.
- τ_k, where 1 ≤ k ≤ K, is the set of path delays. K is the total number of paths in the multipath fading channel.
- $\{a_k\}$,where 1 ≤ k ≤ K , is the set of complex path gains of the multipath fading channel. These path gains are uncorrelated with each other.
- N_1 and $N_2$are chosen so that $h(n)$is small when *n* is less thanN_1or greater thanN_2.

At the receiver, the guard time is deleted:

$$y(n) = \begin{cases} y_f(n) \quad for -N_g \le n \le N-1 \\ y_f(n+N_g) \quad n = 0, 1, \ldots, N-1 \end{cases} \cdots (5)$$

Then, *y(n)* is sent to FFT block for the following operation

$$Y(k) = FFT\{y(n)\} \quad k = 0, 1, 2, \ldots, N-1$$

$$Y(k) = \frac{1}{N}\sum_{n=0}^{N-1} y(n)e^{-j\left(\frac{2\pi kn}{N}\right)} \cdots (6)$$

After FFT block, the pilot signals are extracted and the estimated Channel $\hat{H}$ for the data sub-channels is obtained in channel estimation block using LS estimator. Then, the transmitted data is estimated by [9]

$$\hat{X} = \frac{Y(k)}{\hat{H}(k)} \quad , k = 0, 1, \ldots, N-1 \quad \cdots (7)$$

At the end, the binary information data is obtained back in demodulator and signal demapper block.

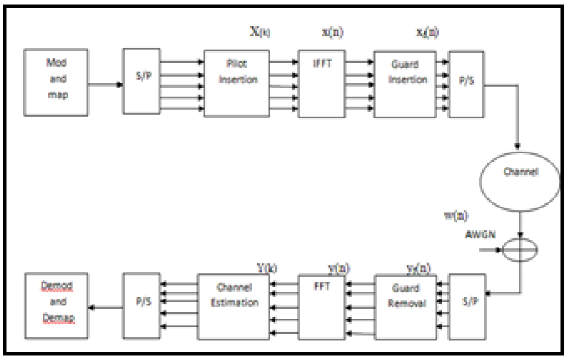

Figure 1. OFDM system based pilot channel estimation

4. Channel Estimation Based On Block-Type Pilot Arrangement

In block-type pilot-based channel estimation, symbols of OFDM channel estimation are transmitted periodically, and all subcarriers are used as pilots. The target here is to estimate the channel conditions (specified by *H* or *h*) given the pilot signals (specified by *X*) and received signals (specified by *Y*), with or without using limited knowledge of the channel statistics. The receiver uses the estimated channel conditions to decode the received data inside the block until the next pilot symbol arrives. The estimation can be depend on Least Square (LS) and minimum mean-square error (MMSE) estimators. The estimation that used in this paper is LS estimator.

LS channel estimation method finds the channel estimate in such a manner that weighted errors between the measurements and the model are minimized. The LS estimates H, given the received data Y and the transmitted symbols X is given by [10]:

$$\widehat{H}_{LS} = X^{-1}\mathrm{Y} \cdots (8)$$

5. Artificial Neural Networks (ANNs)

Artificial neural networks (ANNs) are one of the most popular branches of artificial intelligence (AI). Artificial Neural Network (ANN) isn't a parametric statistical tool which can be used for a host of pattern classification and prediction problems. ANNs can be used in a difficult mapping between their input and output space and are capable of forming difficult decision regions with nonlinear decision boundaries. Further, and because of nonlinear properties of the ANNs, these networks of different architecture have found successful applications in different problems [11]. The most important characteristic of an artificial neural network is its ability to learn. Learning is a process in which the network adjusts its parameters the (synaptic weight) in response to input stimuli, so that the actual output response converges to the response of desired output. The most popular supervised learning technique in ANN is BP algorithm. The work in this paper comes under supervised learning method, which incorporates an external reference signal and generates an error signal by comparing the reference with the detected result. Based on this error signal, neural network modifies its synaptic connections (weights) to improve the system performance [12].

5.1 Architecture of Neural Networks

According to the architecture, neural networks can be classified into

1. FNN (Feedforward Neural Network): In an FNN, the connections between neurons are in a feedforward manner. The network is usually arranged in the form of layers. In layer of FNNs, there is no connection between the neurons within each layer, and feedback isn't found between layers.
2. RNN (Recurrent Neural Network): In an RNN, there is at least one feedback connection that corresponds to an integration operation or unit delay.
3. A lattice network: A lattice network consists of one, two or more dimensional array of neurons.
4. A layered FNN with lateral connections: A layered FNN with lateral connections is a neural network that has lateral connections between the units at the same layer of its layered FNN architecture

5. CNN (Cellular Neural Network): A CNN consists of regularly spaced neurons, called cells, which connect only with the neurons in its immediate neighborhood [12]. In this work , two types of neural network are used (FNN and RNN)

5.2 Back-Propagation Algorithm (BP)

The back propagation (BP) algorithm propagates backward the error between the desired signal and the network output through the network. After providing an input pattern, the output of the network is compared with a given desired pattern and the error of each output unit calculated. This error signal is propagated backward, and a closed-loop control system is established. In order to implement the BP algorithm, a continuous, nonlinear, monotonically increasing, differentiable activation function is needed. The two most-used activation functions are sigmoid function the as in equation 9 and the linear transfer function as in equation 10. Figure (2) shows the activation functions [12-13]. Backpropagation can be used to learn FNN and RNN.

$$tansig(n) = \frac{e^{n}-e^{-n}}{e^{n}+e^{-n}} \quad \dots \ (9)$$

$$pureline(n) = n \quad \dots \ (10)$$

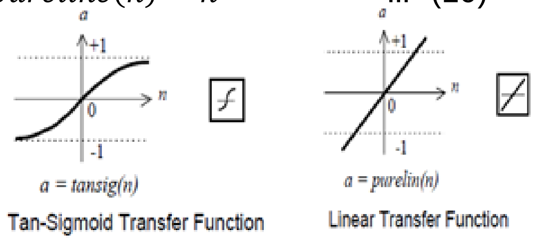

Figure 2. Activation functions

6. Implementation Of Research Methodology

The research methodology can be classified in two parts:

6.1 proposed FNN with OFDM

Figure (3) shows the block diagram of proposed FNN with OFDM for enhance channel estimation, contains FNN, which works on the received signals to recover signals transmitted from transmitter. The symbols after parallel to serial converter block are taken as desired symbols. The data after passing through the channel are taken as the training data. The OFDM system uses complex signals; whereas, the neural network uses real signals. To adapt the neural network to the OFDM system, each complex signal has been separated into two real signals, the real and imaginary parts. With these training data and target data, the network will be trained for varying the SNR. Figure (3) shows the training data equals to $y_f(n)$ as indicated in equation 3, then the input to ANN can be expressed as:

$$G = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \qquad \cdots (11)$$

let$y_f(n)$ = y(received signal), then g_1 is the real part of y and g_2 is the imaginary part of y. The input layer plays the role of distributing the input to all neurons in the first processing layer, where, first hidden layer (layer (1)) contains four neurons and every input in the input layer is connected to every neuron in layer (1). The output of layer (1) is computed as

$$O = tansig\left\{(W\{1\}.G) + B\{1\}\right\} \quad \cdots (12)$$

O represents the output of layer (1), W{1} represents the weights that connect the input layer with layer (1). B{1} represents the bias values of layer (1) and $tansig$ represents the activation function as mentioned in equation (9) for layer (1). $W\{1\}$, $B\{1\}$and O can be expressed as

$$W\{1\} = \begin{bmatrix} w\{1\}_{1,1} w\{1\}_{1,2} \\ w\{1\}_{2,1} w\{1\}_{2,2} \\ \vdots \quad\quad \vdots \\ w\{1\}_{4,1} w\{1\}_{4,2} \end{bmatrix}, B\{1\} = \begin{bmatrix} b\{1\}_1 \\ b\{1\}_2 \\ \vdots \\ b\{1\}_4 \end{bmatrix} \text{ and } O = \begin{bmatrix} o_1 \\ o_2 \\ \vdots \\ o_4 \end{bmatrix}$$

After layer (1), there is output layer. The output of this layer is computed as

$$R = purelin\left\{(W\{2\}.O) + B\{2\}\right\} \quad \cdots (13)$$

R represents the output of output layer, W{2} represents the weights that connect the layer (1) with output layer. B{2} represents the bias values of output layer and $purelin$ represents the activation function as mentioned in equation (10) for output layer. $W\{2\}$, $B\{2\}$and R can be expressed as

$$W\{2\} = \begin{bmatrix} w\{2\}_{1,1} w\{2\}_{1,2} \cdots w\{2\}_{1,4} \\ w\{2\}_{2,1} w\{2\}_{2,2} \cdots w\{2\}_{2,4} \end{bmatrix},$$

$$B\{2\} = \begin{bmatrix} b\{2\}_1 \\ b\{2\}_2 \end{bmatrix} \text{ and } R = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}$$

r_1 is the output received from first neuron of output layer. r_1 represents the real part of transmitted signal (target data) and r_2 is the output received from second neuron of output layer. r_2 represents the imaginary part of transmitted signal (target data). The architecture of FNN can be seen in Figure (4). After training r_1 and r_2 are merged again then, the steps from removing of guard time as indicated in equation (5) to (7) will be repeated. In table (1), the parameters of ANNs are given and in table (2), the parameters of OFDM are given.

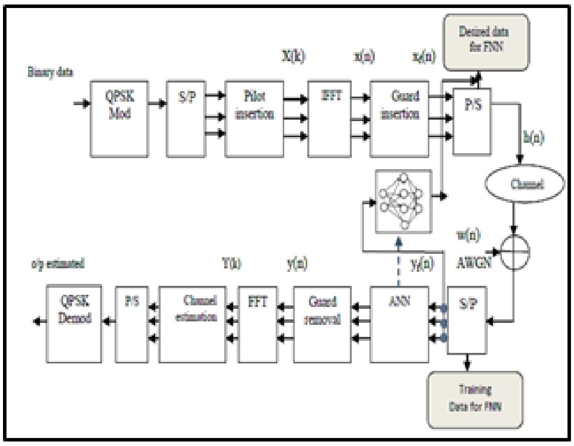

Figure 3. block diagram of proposed FNN with OFDM

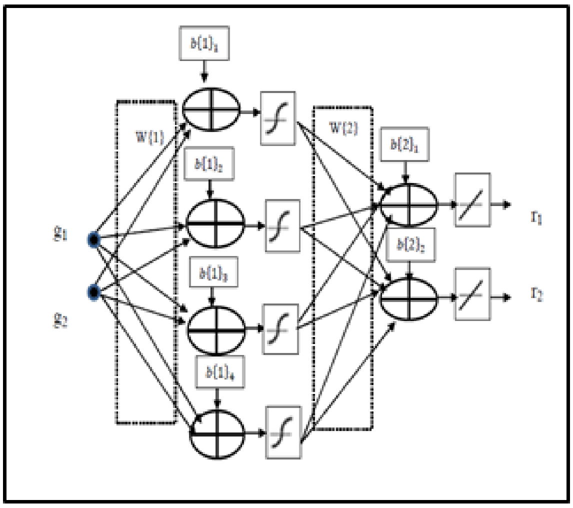

Figure 4. Proposed Architecture of FNN

Table 1. Parameters of ANN

Parameter	Value
Number of inputs	2
Number of hidden layers	1
Number of neurons	4
Epoch number	1000
Training functions	Conjugate Gradient algorithm,Quasi Newton algorithm and Bayesian regularization

Table 2. OFDM system

Parameter	Value
Modulation type	QPSK
FFT size	128
Number of carrier	128
Guard length	32 symbols
Type of guard interval	Cyclic prefix

6.2 Proposed RNN with OFDM

There are several types of RNN, in this work, Elman networks-RNN is used. Elman Neural Network (ENN) is a type of partial recurrent neural network, which consists of two-layer back propagation networks with an additional feedback connection from the output of the hidden layer to its input layer. Architecture of The Elman network commonly is a two-layer network with feedback from the first-layer output to the first layer input as shown in Figure (5). This recurrent connection allows the Elman network to both detect and generate time-varying patterns. The Elman network has tansig neurons in its hidden (recurrent) layer, and purelin neurons in its output layer. This combination is special in that two-layer networks with these transfer functions can approximate any function (with a finite number of discontinuities) with arbitrary accuracy [14].

Figure (6) shows the block diagram of proposed Elman-RNN with OFDM for enhance channel estimation contains RNN, which works on the received signals to recover signals transmitted from transmitter. Figure (6) shows the training data equals to $y_f(n)$. Then, the input to ANN can be expressed as:

$$G = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad \dots \quad (14)$$

g_1 and g_2 are defined in a last section(6.1). The output of layer (1) is computed as

$$O_n = tansig\left\{(W\{1\}.G) + O_{(n-1)}W\{d\} + B\{1\}\right\}$$

O_n represents the outputs of layer (1), W{1} and B{1} are defined in a last section(6.1) , $W\{d\}$ represents the weight that connect from the hidden layer (1) to input layer, $O_{(n-1)}$ represents the outputs of layer(1) after delay time and considered as input to input layer. For layer (1). $W\{1\}$, $B\{1\}, W\{d\}$ and O_n can be expressed as

$$W\{1\} = \begin{bmatrix} w\{1\}_{1,1} w\{1\}_{1,2} \\ w\{1\}_{2,1} w\{1\}_{2,2} \\ \vdots \quad \vdots \\ w\{1\}_{4,1} w\{1\}_{4,2} \end{bmatrix}, B\{1\} = \begin{bmatrix} b\{1\}_1 \\ b\{1\}_2 \\ \vdots \\ b\{1\}_4 \end{bmatrix} \text{ and } O = \begin{bmatrix} o_1 \\ o_2 \\ \vdots \\ o_4 \end{bmatrix}, W\{d\} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_4 \end{bmatrix}$$

After layer (1), there is output layer. The output of this layer is computed as

$$R = purelin\left\{(W\{2\}.O) + B\{2\}\right\} \quad \cdots (15)$$

R represents the output of output layer, W{2} represents the weights that connect the layer (1) with output layer. B{2} represents the bias values of output layer. For output layer. $W\{2\}$, $B\{2\}$and R can be expressed as:

$$W\{2\} = \begin{bmatrix} w\{2\}_{1,1} w\{2\}_{1,2} & \cdots & w\{2\}_{1,4} \\ w\{2\}_{2,1} w\{2\}_{2,2} & \cdots & w\{2\}_{2,4} \end{bmatrix},$$

$$B\{2\} = \begin{bmatrix} b\{2\}_1 \\ b\{2\}_2 \end{bmatrix} \text{ and} R = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}$$

The architecture of Elman-RNN can be seen in figure (7).

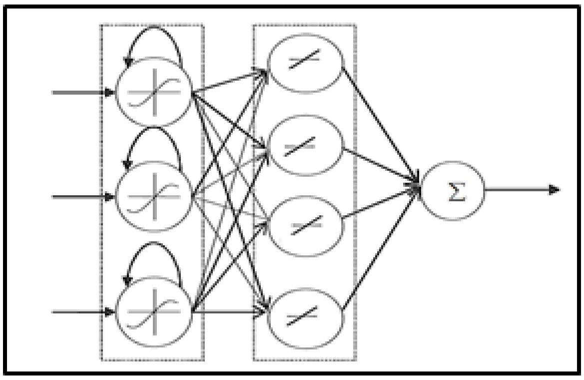

Figure 5. Common Architecture of Elman RNN.

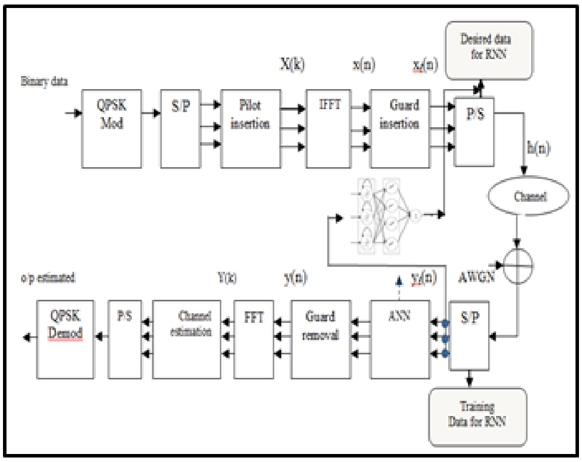

Figure 6. block diagram of proposed RNN with OFDM

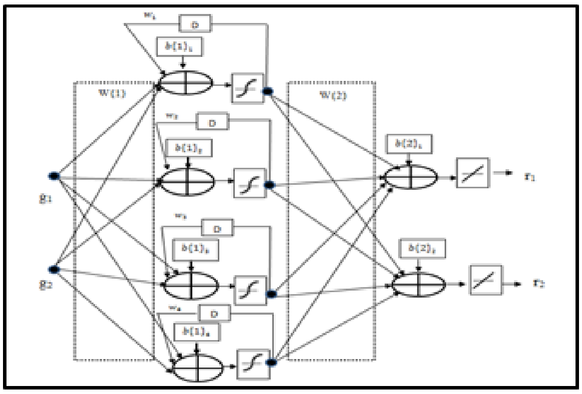

Figure 7. Proposed Architecture of Elman-RNN

7. Simulation Results

The BER for different values of SNR is calculated in one path and two paths of Rayleigh fading by using LS channel estimation and LS with ANN(FWW and Elman-RNN). Comparison of BER performance between OFDM system based on LS and LS with ANN(FWW and RNN) estimators in one path and two paths of Rayleigh fading are shown in Figures (8-11). Figure (8) and Figure (9) show the comparisons between LS and LS with ANN(in this case FNN neural network is trained by Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization) in one path and two paths respectively. Figure (10) and Figure (11) show the comparisons between LS and LS with ANN(in this case Elman-RNN neural network is trained by Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization) in one path and two paths respectively. Table (3) and Table (4) lists the SNR values obtained at BER=0.01 for one and two paths delay and for three different algorithms that used for training neural networks (in FNN and Elman-RNN respectively). Table (5) and Table (6) show the mean square error for each algorithm used to train the proposed artificial neural network in OFDM for one and two paths in FNN and Elman RNN respectively.

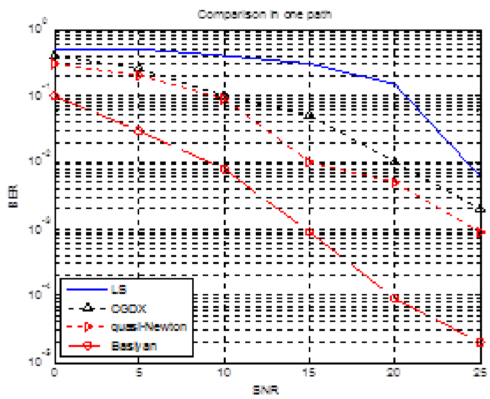

Figure 8. BER performance of OFDM system for one path in FNN

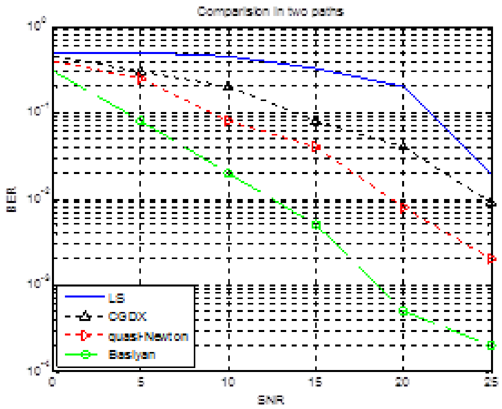

Figure 9. BER performance of OFDM system for two paths in FNN

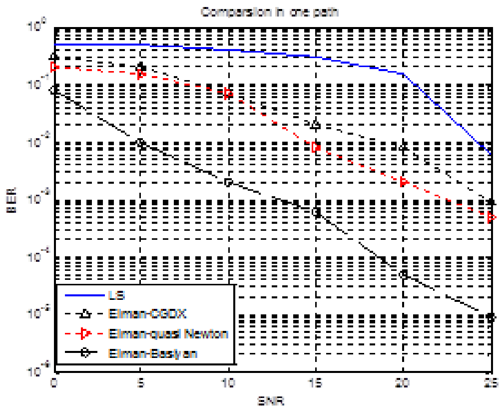

Figure 10. BER performance of OFDM system for one path paths in RNN

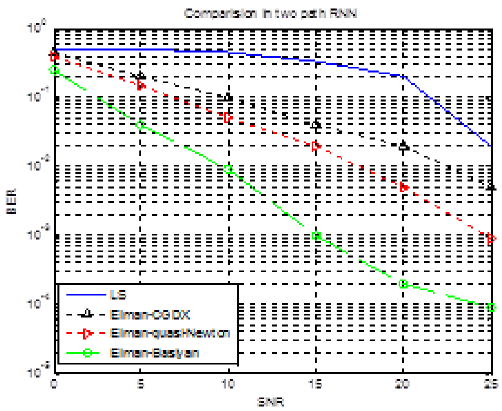

Figure 11. BER performance of OFDM system for two paths in RNN

Table (3)SNR values obtained for different paths using LS estimator and LS with ANN(FNN) at BER = 0.01

SNR (dB) for one-path				SNR (dB) for two-path			
LS	LS with ANN (Conjugate Gradient algorithm)	LS with ANN (Quasi Newton)	LS with ANN (Bayesian regularization)	LS	LS with ANN (Conjugate Gradient algorithm)	LS with ANN (Quasi Newton)	LS with ANN (Bayesian regularization)
24	19	15	5	26	20	17	10

Table (4) SNR values obtained for different paths using LS estimator and LS with ANN(Elman-RNN) at BER = 0.001

SNR (dB) for one-path				SNR (dB) for two-path			
LS	LS with ANN (Conjugate Gradient algorithm)	LS with ANN (Quasi Newton)	LS with ANN (Bayesian regularization)	LS	LS with ANN (Conjugate Gradient algorithm)	LS with ANN (Quasi Newton)	LS with ANN (Bayesian regularization)
24	20	15	10	26	25	20	15

Table(5) The Performance (MSE) for the training algorithms in FNN

Training Algorithm	Bayesian regularization		Quasi Newton		Conjugate Gradient Descent (CGDX)	
	SNR=15	SNR=25	SNR=15	SNR=25	SNR=15	SNR=25
Performance (MSE) in one path	0.0002	0.00017	0.0004	0.00025	0.00045	0.0004
Performance (MSE) in two paths	0.0004	0.00034	0.00057	0.0003	0.0007	0.00065

Table(6) The Performance (MSE) for the training algorithms in RNN

Training Algorithm	Bayesian regularization		Quasi Newton		Conjugate Gradient Descent (CGDX)	
	SNR=15	SNR=25	SNR=15	SNR=25	SNR=15	SNR=25
Performance (MSE) in one path	0.00015	0.0001	0.0004	0.00025	0.00045	0.0004
Performance (MSE) in two paths	0.0003	0.00025	0.00057	0.0004	0.0007	0.00065

8. Conclusions

This paper presents a performance enhancement of channel estimation in OFDM communication systems. In this paper, three different algorithms (i.e. Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization) have been tested to train neural networks, From the comparison in BER performances versus SNR values between LS estimator and LS with ANN estimator in one and two-paths Rayleigh fading channel with all types of BP for training NN, it can be concluded that, the LS with ANN estimator achieves better BER performance, as compared with LS estimator for all types of BP for training FNN and RNN . it has been established that Bayesian regularization can effectively train neural networks better than Conjugate Gradient algorithm and Quasi-Newton algorithm. Table (3) show that the gain in (dB), obtained at BER=0.01 for one and two paths in FNN and for different algorithms are (4, 9,14) (dB)between using the traditional method(LS) and the proposed method using (Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization) and (1,6,11) (dB) respectively. Table (4) shows that the gain in (dB), obtained at BER=0.01 for one and two paths in RNN and for different algorithms are (6, 9,19) (dB)between using the traditional method(LS) and the proposed method using (Conjugate Gradient algorithm, Quasi-Newton algorithm and Bayesian regularization) and (1,6,11) (dB) respectively. From comparison between gains in FNN and RNN in one and two path, we conclude that Elman-RNN is better than FNN. Table (5) and Table (6) show that the relation between MSE performance of ANN and number of paths in OFDM system, when number of path increase, MSE is increased and when SNR increased, MSE is decreased.

References:

[1] Chhavi Choudhary, Rachna Khanduri and Vishal Gupta,“ Performance Enhancement Of Ofdm Signals Using Selected Mapping Technique And Overview Of Different Papr Reduction Schemes”, International Journal of Advances in Engineering & Technology(IJAET), May ,2013.

[2] Han Wang, Wencai Du and Yong Bai, “Compressed Sensing Based Channel Estimation for OFDM Transmission under 3GPP Channels”, Vol. 9, No. 4 , International Journal of Future Generation Communication and Networking, 2016.

Estimation in MIMO-OFDM System using Neural Network as a Classifier”, International journal of Science Technology & Management (IJSTM), 2015.

[4] A. Filippi, “Non-Conventional Multi-Carrier Carrier Air Interface for Mobile Radio Systems”, Ph.D. Thesis, Technical University of Kaiserslautern, Germany, 2005.

[5] E. Chen, R. Tao and X. Zhao,“ Channel Equalization for OFDM System Based on the BP Neural Network”, IEEE, 8th International Conference Signal Processing (ICSP), Vol. 3, 2006 [IVSL].

[6] Gowrishankar and P.S.Satyanarayana, “Recurrent Neural Network Based BER Prediction for OFDMA Channel”, IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.12, December 2007.

[7] M. A. Moustafa and Salwa H. A. El-Ramly,“Channel Estimation and Equalization Using Backpropagation Neural Networks in OFDM Systems”, IEEE, Wireless and Optical Communication Networks (WOCN), International Conference, pp. 1-4, April, 2009 [IVSL].

[8] C. H. Cheng, Y. H. Huang and H.C. Cheng ,“ Channel Estimation in OFDM systems using neural network technology combined with a genetic algorithm”, springer , 2015.

[9] Begüm Korunur Engiz, ÇetinKurnaz and GökhanKayhan, “ Performance Evaluation of ANN Based Channel Interpolation for OFDM System”, Innovations in Intelligent Systems and Applications (INISTA), July, 2012.

[10] Y. Soo, J.Kim, W. Young and C. G. Kang, “MIMO-OFDM Wireless Communications with Matlab”, John Wiley&Sons (Asia) Pte Ltd ,2010.

[11] K. K. Sarma and A. Mitra, “ Estimation of MIMO Wireless Channels Using Artificial Neural Networks”, Book Chapter in Cross-disciplinary applications of artificial intelligence and pattern recognition: advancing technologies, 2012.

[12] K. Du and M. N. S. Swamy, “Neural Networks in a Soft computing Framework”, Springer-Verlag London Limited, 2006.

[13] J. Zurada, “Introduction to Artificial Neural Network System”, West Publishing Company, ISBN: 0-3 14-93391 -3, October, 1992.

[14] ZhiQiang Zhang, Zheng Tang and Catherine Vairappan, “A Novel Learning Method for Elman Neural Network Using Local Search”, Neural Information Processing – Letters and Reviews Vol. 11, No. 8, August 2007.