CHAPTER 4

LINK ADAPTATION USING NEURAL NETWORK

4.1 INTRODUCTION

For accurate system level simulator performance, link level modeling and prediction [103] must be reliable and fast so as to improve the throughput of the system. EESM and MMIBM based link level prediction has been elaborated in earlier chapters to yield better link level prediction for OFDM based systems. WiMax technology does not rely totally on power adaptation, instead depends on adaptation of other channel parameters and advanced signal processing techniques to meet the target. OFDM systems operate within the coherence time of the wireless channel for proper synchronization and detection hence requires the LA process to work fast with the channel [108].

Due to the aforementioned complexities the LA in OFDM based systems is not straight forward. EESM algorithm requires a large dimension look up table for each channel realization under every MCS possible defined in the system. MMIBM based system uses Bit Interleaved coded modulation which uses soft bit decoder that calibrates the LLR for every bit and in turn accumulated as the metric. For both methods to be effective either the reduction in memory size or a fast response is a necessary criterion. This paved the way to implement the EESM and MMIBM in a neural network. From the literature the LA for OFDM systems has been performed in supervised learning approach [75] or straight forward approach. Both uses k-nearest neighbor (k-NN) algorithm and Multilayer Perceptron (MLP) as a machine learning process. K-NN suffers from increases memory storage when the size of training set grows [77] and the disadvantage of MLP is that it needs computationally
intensive training process to be effective. The proposed Neural Network (NN) uses Feed forward back propagation algorithm using supervised learning approach.

4.2 FEEDFORWARD BACK PROPAGATION ALGORITHM

4.2.1 Principle

This research focuses on the Back propagation algorithm learning method for LA process. The Fig.4.1 depicts a single artificial neuron, which learns using the Back propagation learning [77] algorithm.

Fig. 4.1 Input data processing by an artificial neuron

where

\[ N_i \] represents the output of the previous node

\[ W \] represents the weights

\[ D_j \] represents the desired output of node j

\[ E \] represents the overall error for a single pass
The Back propagation algorithm seeks to minimize the error term between the output of the NN and the actual desired output value. The error term is calculated by comparing the net output to the desired output and is then fed back through the network causing the synaptic weights to be changed in an effort to minimize error. The process is repeated until the error reaches a minimum value. The learning parameter, or learning rate, controls the rate at which the weight is changed as learning takes place.

The total input to a node is represented as

\[
S_j = \sum_i N_i W_{ij}
\]  

(4.1)

where \(S_j\) is the sum of all inputs to a node, \(N_i\) is the output of the previous node, and \(W_{ij}\) is the weight connection between the \(i^{th}\) node of the previous layer. The total output of node \(j\) is represented by \(N_j\)

\[
N_j = \frac{1}{1 + e^{-S_j}}
\]  

(4.2)

The overall error for a single pass of the neural network is represented by (4.4), where \(D_j\) is the desired output of the output node \(j\).

\[
E = \frac{1}{2} \sum_{\text{output}} (N_j - D_j)^2
\]  

(4.3)

Now that the error term for the entire network has been calculated, it is fed back through the network to reduce error.

### 4.2.2 AMC on FFBP Algorithm

According to physical layer specifications of 802.16e for SISO OFDM [76] model, the neural network has been trained for study. The feed forward back propagation algorithm with multilayer perceptron network has been applied to the
OFDM system under consideration. The network exploits supervised learning approach to select the appropriate class index or the MCS for the current channel conditions. The basic element set consists of post processing SNRs defined as set \( W = \{1,2,\ldots,N_s\} \) where the elements are the SNRs, corresponding to the subcarriers set \( \{1,2,\ldots,N\} \). Each class label \( C \) corresponds to the MCS \( c \) index so that the target BER with maximum throughput can be achieved. Hence the neural network has been trained to achieve the condition defined as

\[
\arg \max \{R_c : BER_c \leq BER_{\text{target}}\} \tag{4.4}
\]

where the index \( c \) corresponds to different classes in this case it is MCS. The equation states that the class is selected if the corresponding BER is less than or equal to the target BER.

4.3 PROPOSED NEURAL NETWORK

4.3.1 EESM Implementation

This section discusses the techniques used to develop the neural network and training it to perform EESM based link adaptation. A MLP network performs nonlinear functional mapping between the feature sets and class indices. In our proposed network model we consider 3-layer perceptron model with 3 hidden layers. The network is trained using Levenberg-Marquardt algorithm to minimize the error between the output and the target value. After training the weights converge to a particular solution so that the network can select the perfect class to achieve the target. The network structure for EESM link adaptation consist of two stages in which the stage 1 computes the Effective SNR value and the stage 2 decides the best MCS based on effective SNR from stage 1.
The Fig. 4.2 elaborates the neural network structure that was developed and trained to perform Link adaptation using EESM method. The learning rate and the momentum is set to 0.005 and 0.5. The pattern selection to train the criteria was set to random. Within the training module, the network was set to automatically save the weights for the best test set. The trained network was tested with 50 sample SNR.
random patterns initially and then simulated with the full input set that was used for training. The same network has been used to select 2 different MCS at the output for two set of subcarriers.

The network specifications to perform AMC using EESM are listed in the Table 4.1 and Table 4.2

**Table 4.1 Stage 1 Network: Effective SNR Calibration**

<table>
<thead>
<tr>
<th>Slab No</th>
<th>Slab type</th>
<th>Transfer Function</th>
<th>No. of Neurons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slab 1</td>
<td>Input</td>
<td>Radial Basis (N)</td>
<td>(N) No of subcarriers</td>
</tr>
<tr>
<td>Slab 2</td>
<td>Hidden</td>
<td>Radial Basis</td>
<td>300</td>
</tr>
<tr>
<td>Slab 3</td>
<td>Hidden</td>
<td>Radial Basis</td>
<td>300</td>
</tr>
<tr>
<td>Slab 4</td>
<td>Hidden</td>
<td>Radial Basis</td>
<td>300</td>
</tr>
<tr>
<td>Slab 5</td>
<td>Output</td>
<td>Radial Basis</td>
<td>1</td>
</tr>
</tbody>
</table>

**Table 4.2 Stage 2 Network: EESM MCS selection**

<table>
<thead>
<tr>
<th>Slab No</th>
<th>Slab type</th>
<th>Transfer Function</th>
<th>No. of Neurons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slab 1</td>
<td>Input</td>
<td>Tansig</td>
<td>1</td>
</tr>
<tr>
<td>Slab 2</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 3</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 4</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 5</td>
<td>Output</td>
<td>Tansig</td>
<td>1</td>
</tr>
</tbody>
</table>

### 4.3.2 Simulation Set Up and Results

The system under consideration was simulated in MATLAB which consists of 52 data subcarriers in an OFDM symbol with FFT size of 64 for 24 MHz channels. Perfect frequency and timing synchronization is assumed. The system was
simulated using 4 QAM (1/2 coded), 4 QAM uncoded, 16 QAM, 64 QAM to determine the BER. The simulated SNR vectors are used as data set for training phase with the number of channel realizations set to 50. Hence there will be 50 different BER values for each MCS. At any instant the stage 2 of the network will choose one of the four MCS schemes to achieve the target BER of $10^{-2}$. Regression testing has been performed by rerunning the test cases to verify the error for all the output stages.

Fig. 4.3 shows the SNR$_{eff}$ calibrated by the Neural Network and its comparison with simulated results for 50 time instants. 52 set of SNR vectors are generated by setting the Doppler shift from 10Hz to 100Hz and delay spread to 200 ns. Fig.4.4 shows the regression plot achieved by the Neural Network in calibrating the SNR$_{eff}$. It is observed that 0.92695 regression is achieved by the stage 1 of the network. Fig.4.5 shows the EESM link adaptation simulation result achieved by simulation and Neural Network for 50 time instants. The adaptation is carried out between 4 MCS schemes. At any time instant the MCS which gives the best throughput at the target bit error rate of $(10^{-2})$ is selected. Fig.4.6 shows that a regression of 0.99892 is achieved which infers a very accurate performance by the Neural Network is achieved. On the whole any error performance should be from the effective SNR calibration.

Subcarriers are grouped in an OFDM symbol size of 128 subcarriers with three groups A, B, C with threshold levels fixed at 10, 20, 30db. Group A and Group B FFT size is fixed at 52 subcarriers with Group C is considered as no transmission mode as the subcarriers are weak that they cannot achieve the desired target. If necessity arises group C shall be also utilized to make the FFT size in the group. The signaling information i.e. the group in which all the subcarriers falls is fed back via control channel to the transmitter. Fig.4.7 and Fig.4.8 depicts the signaling information simulated by NN and the regression achieved. It infers that regression of 0.82523 is achieved. Fig.4.9 shows the MCS selection performed on the two groups A,B within the OFDM symbol. The regression achieved is 0.99992 as shown in the Fig.4.10.
Fig. 4.3  Effective SNR calibration by neural network

Fig. 4.4  Regression plot for effective SNR calibration
**Fig. 4.5** EESM MCS selection by proposed Neural Network

**Fig. 4.6** Regression plot for MCS selection
Fig. 4.7  Signaling information for EESM unequal modulation

Fig. 4.8  Regression plot for signaling by neural network
Fig. 4.9 AMC for unequal grouping using EESM technique

Fig. 4.10 Regression plot for AMC prediction
4.4 MMIBM USING NEURAL NETWORK

4.4.1 Principle

To perform LA using MMIBM procedure the input vectors to the neural network is MIB, i.e., the mutual information per bit are considered as the set $W=\{1,2,\ldots,N\}$ where $N$ represents the number of bits in a codeword after the interleaver. The MMIB is calibrated for a codeword based on which the class index $c_i$ is selected from the class labels $C=\{c_1,c_2,\ldots,c_N\}$, where the class labels to the different MCS level to achieve the desired target value. Hence the constraint to achieve the target can be defined as

$$\arg \max \{ R_c : BER_c \leq BER_{\text{target}} \}$$

(4.5)

The equation selects the particular code rate under the condition that the target error rate is achieved to maximize the throughput.

4.4.2 Proposed Neural Network for MMIB Implementation

This section discusses the techniques used to develop the neural network and training it to perform MMIB based link adaptation. The basic principle of MMIBM based LA is explained in chapter 2. The network structure for MMIB link adaptation consist of two stages in which the stage 1 computes the MMIB value and the stage 2 maps the MMIB metric to the BLER and decides the best MCS. The input vectors to stage 1 are MIB’s corresponding to a single codeword and the output is the mean of the MIBs referred as MMIB. This network is already trained for 40 different set of input vectors which corresponds to channel realizations and the MIB values are set between 0.3 to 0.9. For each set of input there exists a set of 3 BER values and the stage 2 network decides the best coding order. The network specifications for the implementation of the stages are listed in the following Table 4.3 and Table 4.4
### Table 4.3 Stage 1 Network: MMIB Calibration

<table>
<thead>
<tr>
<th>Slab No</th>
<th>Slab type</th>
<th>Transfer Function</th>
<th>No. of Neurons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slab 1</td>
<td>Input</td>
<td>Tansig</td>
<td>(N) No of subcarriers</td>
</tr>
<tr>
<td>Slab 2</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 3</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 4</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 5</td>
<td>Output</td>
<td>Tansig</td>
<td>1</td>
</tr>
</tbody>
</table>

### Table 4.4 Stage 2 Network: MMIB MCS selection

<table>
<thead>
<tr>
<th>Slab No</th>
<th>Slab type</th>
<th>Transfer Function</th>
<th>No. of Neurons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slab 1</td>
<td>Input</td>
<td>Tansig</td>
<td>1</td>
</tr>
<tr>
<td>Slab 2</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 3</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 4</td>
<td>Hidden</td>
<td>Tansig</td>
<td>30</td>
</tr>
<tr>
<td>Slab 5</td>
<td>Output</td>
<td>Tansig</td>
<td>1</td>
</tr>
</tbody>
</table>
Fig. 4.11 shows the neural network structure that was developed and trained to perform Link adaptation using MMIB method. The learning rate and the momentum are set to 0.005 and 0.5. The pattern selection to train the criteria was set to random. Within the training module, the network was set to automatically save the weights for the best test set. The trained network was tested with 40 sample random patterns initially and then simulated with the full input set that was used for training.
4.4.3 Simulated Results and Discussion

The MMIB based link adaptation has been performed for Rayleigh channel using convolution coder at the rate of 1/2, 3/4, 5/6. Since the bit interleaver is used in the transmitter no other complicated coded system is used as the convolution coder with interleaver gives a better performance. From the simulation result the threshold level of MMIB is set to 0.58 for QAM(1/2) and 0.85 for 16QAM(3/4) and 0.92 for 64 QAM(5/6) to achieve the desired target BER. The system adapts for any one of the three MCS scheme based upon the channel instant inferred by the MMIB achieved at the receiver. Fig.4.12 shows the mapping performed by the network for 40 time instants between the listed MCS schemes. The Fig.4.13 depicts regression achieved for MMIB based MCS selection by neural network. A regression of 0.99959 was achieved which proves the accuracy of the neural network.

Using the same architecture the mapping has been performed for two MMIBs corresponding to the two groups of bits in a single codeword. The mapping has been performed for the same mentioned MCS schemes for 40 different time instants. The group of 52 bits has been divided into 2 groups of 26 bits. The Groups 1 is mapped always with 4QAM(1/2) as it involves the transmission of sensitive information. Group2 selects the high order MCS between 16QAM(3/4) and 64 QAM(5/6) as it carriers the approximate details of the information and considerable tolerance can be given on the target BER. If the MI value received is too low to make a transmission the groups are considered as No transmission mode. Selection of AMC is shown in Fig.4.14 and Fig.4.15 depicts the regression achieved which is 0.9994 for grouping and the desired result is accurate compared to simulation.
Fig. 4.12  MCS selection by the proposed Neural Network

Fig. 4.13  Regression plot for MCS selection
Fig. 4.14 AMC mapping for unequal loading of bits using MMIBM procedure

Fig. 4.15 Regression plot for unequal mapping using MMIBM
4.5 SUMMARY

A Multilayer Perceptron neural network model trained with Levenberg-Marquardt back propagation algorithm to perform Adaptive Modulation and Coding using EESM and MMIB procedure for frequency selective channels has been proposed. The contribution of this thesis is to model the NN for grouping of subcarriers based on the channel conditions and perform AMC for multiple groups so as to improve the throughput performance. Neural network has been trained to perform the AMC for unequal grouping of EESM and MMIB based Link Adaptation procedures. It is proved that the proposed Neural Networks accurately predicts the MCS group for at least 90% accuracy compared to simulated results.