A New Class of ANFIS based Channel Equalizers for Mobile Communication Systems

K.C.Raveendranathan
Department of Electronics and Communication Engineering,
Government Engineering College Barton Hill,
Thiruvananthapuram-695035.
Kerala State, India.
E-mail: ravi@gecbh.ac.in

M.Harisankar
Department of Information Technology and Systems,
Indian Institute of Management Kozhikode
IIMK Campus P. O., Kozhikode-673570, Kerala State.
E-mail: mhari@iimk.ac.in

M.R.Kaimal
Department of Computer Science and Engineering,
Amrita Vishwa Vidyapeetham,
Amritapuri Campus, Amritapuri, Clappana P. O. Kollam – 690525, Kerala State
E-mail: mrkaimal@amritapuri.amrita.edu

***Abstract*— System modelling based on conventional mathematical tools like differential equations is not well suited for dealing with ill-defined and uncertain systems. By contrast, a fuzzy inference system, employing fuzzy if–then rules can model the qualitative aspects of human knowledge and reasoning processes without employing precise quantitative analyses. This fuzzy modeling or fuzzy identification explored by Takagi and Sugeno, has found numerous practical applications in control, prediction and inference. However there are some basic aspects of this approach which are in need of better understanding. More specifically, no standard methods exist for optimally transforming human knowledge or experience into the rule base and database of a fuzzy inference system. There is a need for effective methods for tuning the membership functions (MFs) so as to minimize the output error measure or maximize performance index. In this perspective, a novel architecture called Adaptive Network Based Fuzzy Inference System (ANFIS) which can serve as a basis for constructing a set of fuzzy if–then rules with appropriate membership functions to generate the stipulated input–output pairs, is taken up by Jang. We in this paper present a class of novel channel equalizers based on the ANFIS architecture.**

Keywords- System Modelling, Channel Equalization, ANFIS, System Identification, Radial Basis Function.

I. INTRODUCTION

Adaptive filtering has achieved widespread application and success such as control, image processing, and communication [1]. Among the various adaptive filters, the adaptive linear filter is the most widely used one mainly due to its low hardware implementation cost and its other properties, like the convergence, global minimum, mal-adjustment error, and training algorithms. It can be analyzed and derived easily. The adaptive linear filtering has achieved a large amount of success in many situations. The maximum likelihood sequence estimators (MLSE) [2] were implemented using the Viterbi algorithm. The large computational complexity associated with the Viterbi algorithm and the poor performance of the linear equalizers have led to the development of symbol-by-symbol equalizers using the maximum a posteriori probability (MAP) principle– Bayesian equalizers [3]. These Bayesian equalizers have been approximated using non-linear signal processing techniques like artificial neural networks (ANN) [4, 5], radial basis functions (RBF) [6, 7], recurrent neural networks [8], and fuzzy filters [9, 10, 11, 12]. The study of these new techniques can provide adaptive equalizers which have the advantages of both good performance and low computational cost [10]. Fuzzy filters are nonlinear filters that incorporate linguistic information in the form of fuzzy IF–THEN rules. Fuzzy filters have been used for equalization due to their success in the related area of pattern classification [9, 11, 12]. Wang and Mendel [9] presented fuzzy basis functions (FBF) for channel equalization. The block diagram of the fuzzy inference systems is given in Figure 1.

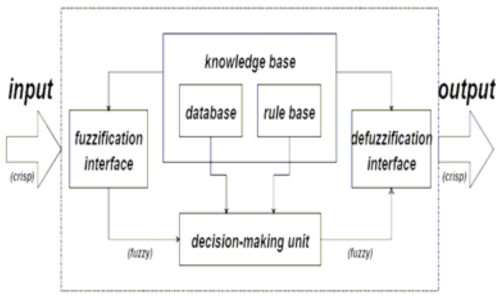

Figure 1: Block Diagram of Fuzzy Inference System (FIS). Note that the *Fuzzy Knowledge Base* comprise of *Data Base* and *Fuzzy Rule Base*.

Lin and Juang [10] developed the ANFFs and used it for equalization and noise reduction. This ANFF constructs its rule base in a dynamic way with the training samples. Patra and Mulgrew [11] derived the close relationship between the fuzzy equalizers and the equalizer based on maximum a posteriori probability principle (MAP). Liang and Mendel [12] developed type-2 fuzzy adaptive filters (FAF) and demonstrated that it can implement the Bayesian equalizer. The structures and learning algorithms of these models are both complicated and not suitable for practical implementation. The Compensation-Based Neuro-Fuzzy Filter (CNFF) can be constructed by learning from training examples. It can be contrasted with the traditional fuzzy logic control systems in their network structure and learning ability. The Equalizer based on ANFIS architecture is a relatively new entrant in this field. We consider a number of ANFIS equalizers with varying parameters. We organize the rest of the paper as follows: In Section 2, we consider the ANFIS Architecture. In Section 3, we discuss the main issues on ANFIS based Channel Equalizers. The simulation results are given in Section 4, which is followed by the conclusion in Section 5.

II. The ANFIS Architecture

Fuzzy if–then rules or fuzzy conditional statements are expressions of the form *IF A THEN B*, where A and B are labels of fuzzy sets, characterized by appropriate membership functions [13]. Due to their concise form, fuzzy if–then rules are often employed to capture the values or labels that are characterized by membership functions [14]. Qilian Liang and Jerry Mendel in [15], suggested the use of type-2 fuzzy adaptive filters (FAF) for co-channel interference mitigation. Raveendranathan et.al. successfully employed neuro-fuzzy controllers in channel equalization [16,17,18]. Another form of fuzzy if–then rule, proposed by Takagi and Sugeno has fuzzy sets involved only in the premise part. By using Takagi and Sugeno's fuzzy if–then rule, we can describe the resistant force on a moving object as follows:

If velocity is high then force = k × velocity^2, where, again, *high* in the premise part is a linguistic label characterized by an appropriate membership function. However, the consequent part is described by a non-fuzzy equation of the input variable, velocity. Both types of fuzzy if–then rules have been used extensively in modeling and control.

A. The ANFIS Model

Functionally, there are almost no constraints on the node functions of an adaptive network except piecewise differentiability. Structurally, the only limitation of network configuration is that it should be of feed-forward type. Due to these minimal restrictions, the adaptive network's applications are immediate and immense in various areas. We now consider a class of adaptive networks which are functionally equivalent to fuzzy inference systems, are referred to as ANFIS (abbreviation for Adaptive-Network-based Fuzzy Inference System).

For simplicity, we assume the fuzzy inference system under consideration has two inputs *x* and *y* and one output *z* [13]. Suppose that the rule base contains two fuzzy if–then rules of Takagi and Sugeno's type:

Rule1 : ***If x is*** A_1 ***and y is*** B_1***, then*** $f_1 = p_1x+q_1y + r_1$.
Rule2 : ***If x is*** A_2 ***and y is*** B_2***, then*** $f_2 = p_2x+q_2y + r_2$.

then the type–3 fuzzy reasoning is illustrated in Fig. 2(a) and the corresponding equivalent ANFIS architecture (type–3 ANFIS) is shown in Fig. 2(b).

B. Node Functions

The node functions in the same layer are of the same function family as described below [13]:

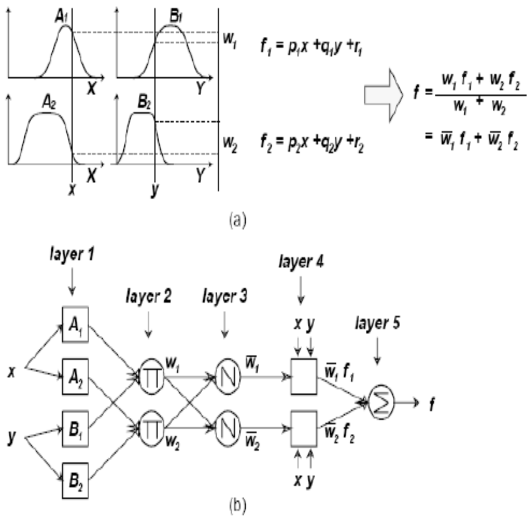

Figure 2: (a) Type–3 Fuzzy Reasoning; (b) Equivalent ANFIS (Type–3 ANFIS).

1. Layer 1: Every node *i* in this layer is a square node with a node function:

$$\mathbf{O^1_i = \mu_{A_i}(x),} \qquad \textbf{(1)}$$

where x is the input to node i, and A_i is the linguistic label (small, large, etc.) associated with this node function. In other words, O^1_i is the membership function of A_i and it specifies the degree to which the given x satisfies the quantifier A_i. Usually we choose $\mu_{A_i}(x)$ to be bell-shaped with maximum equal to 1 and minimum equal to 0, such as:

$$\mu_{A_i}(x)=\exp\{-[(x-c_i/a_i)^2]^{b_i}\}, \qquad \textbf{(2)}$$

where $\{a_i, b_i, c_i\}$ is the parameter set. As the values of these parameters change, the bell-shaped functions vary accordingly, thus exhibiting various forms of membership functions on linguistic label A_i. In fact, any continuous and piecewise differentiable functions, such as commonly used trapezoidal or triangular-shaped membership functions, are also qualified candidates for node functions in this layer [13]. Parameters in this layer are referred to as premise (or, antecedent) parameters.

2. Layer 2: Every node in this layer is a circle node labeled Π which multiplies the incoming signals and sends the product out [13]. For example,

$$\mathbf{w_i = \mu_{Ai}(x)\times\mu_{Bi}(y),\ i = 1,2.} \quad (3)$$

Each node output represents the firing strength of a rule. In fact, other t-norm operators that perform generalized AND can also be used as the node function in this layer.

3. Layer 3: Every node in this layer is a circle node labeled N. The i^{th} node calculates the ratio of the i^{th} rule's firing strength to the sum of all rules' firing strengths:

$$\mathbf{w^{i'}=w_i/(w_1+w_2), i=1,2.} \quad \mathbf{(4)}$$

For convenience, outputs of this layer will be called "normalized firing strengths".

4. Layer 4: Every node i in this layer is a square node with a node function:

$$\mathbf{O^4_i=w_i f_i=w_i(p_i x+q_i y+r_i),} \quad \mathbf{(5)}$$

where w_i is the output of layer 3, and $\{p_i, q_i, r_i\}$ is the parameter set. Parameters in this layer will be referred to as consequent parameters [13].

5. Layer 5: The single node in this layer is a circle node labeled Σ that computes the overall output as the summation of all incoming signals, i.e.:

$$\mathbf{O^5_i=\text{overall output} = \Sigma w_i f_i,} \quad \mathbf{(6)}$$

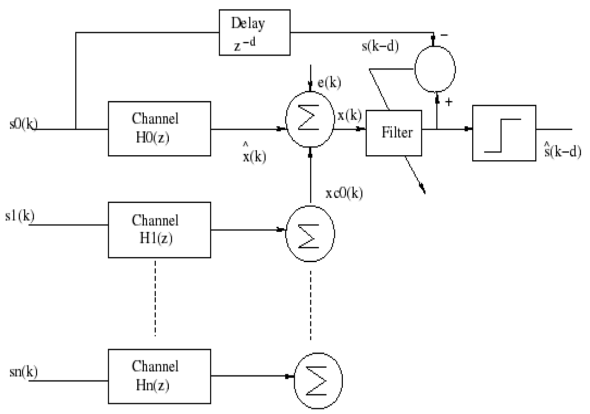

Figure 3: The Digital Communication System with CCI and ACI

II. ANFIS BASED CHANNEL EQUALIZER

Since the equalization of mobile cellular channel, which is basically a non-linear time-variant system, is a non-linear problem; a solution using the ANFIS is most suitable for it. Here, we have to choose a channel model. Fuzzy Logic Toolbox in MATLAB is extensively used for system simulations. For the ANFIS based equalizer, we use a type–3 TSK FIS with Gaussian membership functions. For a practical case, we choose three/five rules for the input variables, each with a Gaussian membership function given by:

$$\mathbf{\mu_{Ai}(x)=\exp\{-[(x-c_i/a_i)^2]^{2b_i}\},} \quad \mathbf{(7)}$$

where $\{a_i, b_i, c_i\}$ is the parameter set. For a channel with 6 co-channels, (i.e., N=7), we can consider the ANFIS equalizer as having 7 components in its input (plus the AWGN in the channel) and one output, which is connected to the ANFIS equalizer and detector, as shown in Figure 3. The output of the channel (received signal), which is a non-linear combination of the channel, the co-channels, and the AWG noise, is a random waveform taking values around +1 and -1, as seen from the simulated waveform. In the simulations, we assume that the external input to the ANFIS equalizer is the output of the channel, which is the sum of the desired channel output plus the weighted sum of the co-channel outputs and the Gaussian noise, which is assumed to be AWGN, with zero mean and standard deviation up to 0.64. In the ensuing sections, we use the following definitions for SNR, signal-to- interference ratio (SIR) and signal-to-interference noise ratio (SINR):

$$\mathbf{SNR = 20\log_{10}\{\sigma_r/\sigma_e\},} \quad \mathbf{(8)}$$
$$\mathbf{SIR = 20\log_{10}\{\sigma_r/\sigma_u\},} \quad \mathbf{(9)}$$
$$\mathbf{SINR = 10\log_{10}\{\sigma^2_r/(\sigma^2_e+ \sigma^2_u)\},} \quad \mathbf{(10)}$$

where σ_r, σ_e, and σ_u are the standard deviations of the signal, AWG noise and the co-channel interference noise respectively.

The output of the equalizer is given to a limiter to clip the output levels to limiting values of +1 or -1. The different parameters of the various simulation setups are as tabulated in Table 1. Structure of ANFIS–27 is given in Figure 4.

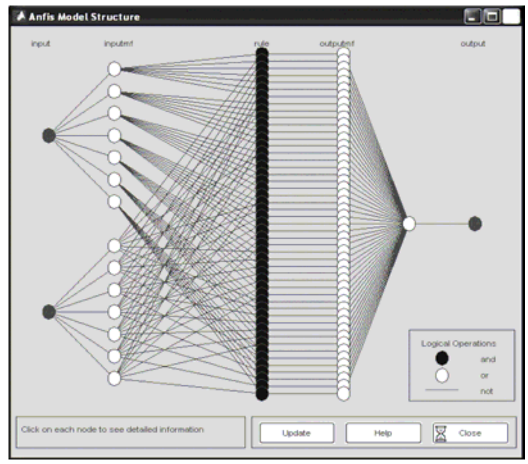

Figure 4: Structure of ANFIS–27 Generated Using MATLAB Fuzzy Logic Toolbox: Number of Inputs = 2, Number of Outputs = 1, Total Number of Fuzzy Rules = 49, Type of Membership Function: Gaussian, and Number of Nodes = 131.

The equivalent ANFIS Architecture for the channel Equalizer is illustrated in Figure 5. The Figure 5 shows the architecture of the proposed ANFIS based channel equalizer, for 7 fuzzy rules. We follow a first-order ANFIS with the antecedent parameters being the variances of the signal, CCI and ACI interference, and the AWGN (σ_s, σ_i, and σ_n, respectively), collectively represented as A_i. The only consequent parameter is the scaling factor of the signal (ρ_i) at the output. The membership functions of A_i, $i = 1, 2, 3, \ldots, 7$ are chosen to be Gaussian functions. The rule base can be written as:

$$\textit{If } x \textit{ is } A_i, \textit{ then } y = \rho_i x, \textit{ for } i = 1, 2, \ldots, 7 \quad \textbf{(11)}$$

The overall output of y is given by:

$$y = \sum_{i=1}^{7} [\mu_{A_i}(x).\rho_i x] \,/\, \sum_{i=1}^{7} \mu_{A_i}(x) \quad \textbf{(12)}$$

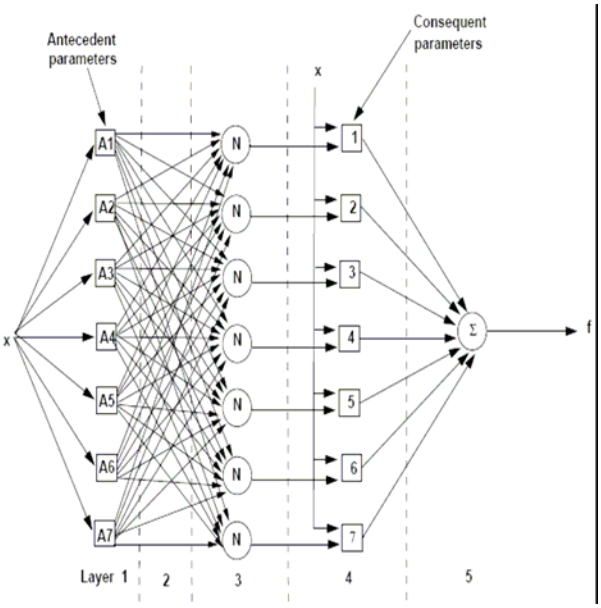

Figure 5: The Equivalent ANFIS Architecture for the Channel Equalizer.

The steps in the algorithm for simulation of the ANFIS–27 based equalizer are as given below:
1. The standard deviations of CCI and AWGN are logarithmically varied from -0.0969 (= $log_{10}0.8$) to -2 (= $log_{10}0.01$). This information is derived from literature.
2. The random binary input data (which represents the input to the channel from the transmitter) is generated and the corrupted data available at the outputs of the two multipaths due to CCI and AWGN is obtained.
3. Set the number of membership functions as 7, membership function type as "Gaussian" and the number of epochs to 20.
4. Simulate the ANFIS (which implements the equalizer) and plot the results.
The error plot of the ANFIS–27 training is illustrated in Figure 6. We have set the number of epochs as 80 in this case. The ANFIS-27 consists of 2 inputs, and one output, 7 fuzzy rules for each membership functions. The fuzzy membership functions are chosen to be Gaussian.

In Table 1 on parameters for various ANFIS, the first digit in the ANFIS type (column 1) indicates the number of inputs to the ANFIS structure (as the 1 in ANFIS–115), and the following digit(s) indicate the number of fuzzy rules for each input(s). The last column indicates the total number of fuzzy rules for the entire ANFIS. The number of outputs is one in all cases.

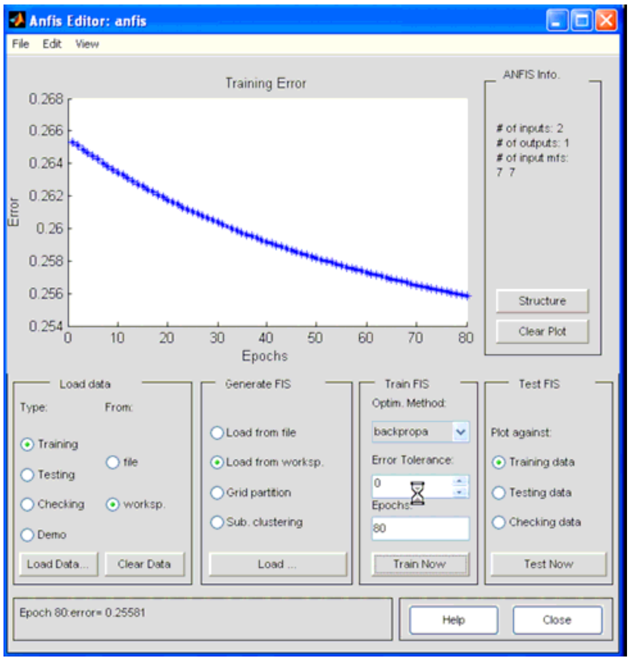

Fig.6: The Error Plot of Training of ANFIS–27; Generated Using MATLAB Fuzzy Logic Toolbox: Number of Inputs = 2, Number of Outputs = 1, Total Number of Fuzzy Rules = 49, Type of Membership Function: Gaussian, and Number of Epochs = 80.

The simulation results for ANFIS–23 (with two input and 3 membership functions) for 4096 training data pairs are shown in Figure 7. Results for other combinations of number of inputs and membership functions, as listed in Table 1, were found to be similar.

In one of the simulations, the standard deviation of CCI and AWGN are logarithmically varied between −2 and $\log_{10}(0.8)$ using the MATLAB command ([*logspace*(*log10*(0.01), *log*10(0.8), 16)]) and simulation is run on a total of 2048/4096 training data pairs. The results are shown in Fig. 8, as a plot of log(*BER*) at output of the equalizer *versus* SINR in dBs.

Then, in another simulation, the log(*BER*) at output of the equalizer is calculated for standard deviation of noise varying from log(0.02) to log(0.8) for two versions of ANFIS Equalizers (ANFIS–115 and ANFIS–125) for 1024/4096 training data pairs and standard deviation of AWGN fixed at 0.42, and the results are plotted in Figure 9. The performance for the above ANFIS pairs, as regards log(*BER*) at output of the equalizer *versus* SNR in dBs for standard deviation of co-channel interference signal fixed at 0.08, is given in Figure 10. The performance for the above ANFIS pairs, as regards BER versus SIR for σ^2_e (variance of AWGN) fixed at 0.2 is given Figure 10.

Table 1: Simulation Parameters for Various ANFIS Equalizers.

Type	*Nodes*	*Linear/Non-linear Parameters*	*Fuzzy Rules*
ANFIS–15	24	10/10	5
ANFIS–17	32	14/14	7
ANFIS–115	64	30/30	15
ANFIS–125	104	50/50	25
ANFIS–25	75	75/20	25
ANFIS–27	131	147/28	49
ANFIS–35	286	500/30	125
ANFIS–37	734	1372/42	343

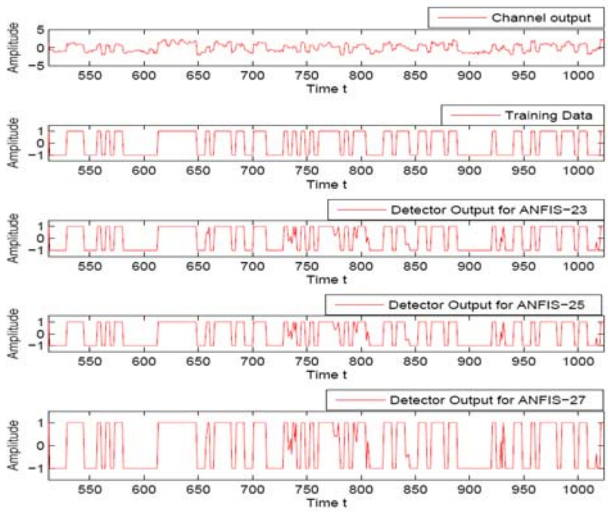

Figure 7: Simulation Results for ANFIS–23, ANFIS–25 and ANFIS–27 Equalizers for 4096 training data pairs, showing the *time domain response*: The plots at the bottom represent the output of ANFIS–23, ANFIS–25 and ANFIS–27 Equalizers with an attached hard thresholding detector.

A plot of performance of ANFIS (average BER versus SNR and standard deviation of BER versus standard deviation of AWGN) based on 100 Monte-Carlo (MC) simulations is given in Figure 11 for different ANFIS structures. 1024 training data pairs are used in the simulation.

The simulation times in various ANFIS for 20 epochs and for 1024 training data pairs are tabulated in Table 2.

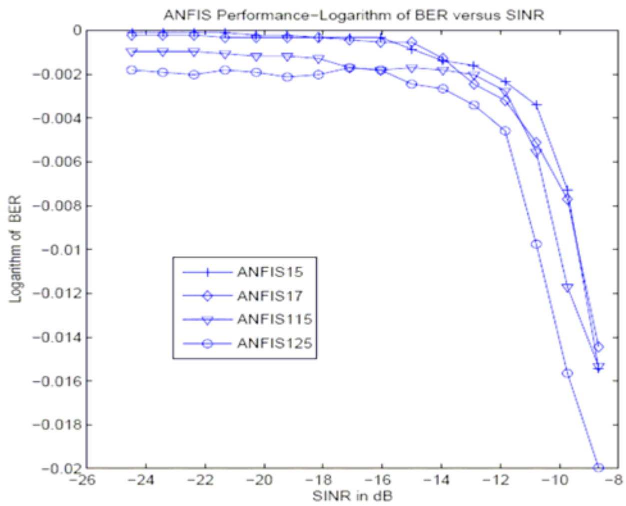

Figure 8: Performance of ANFIS Equalizers–Logarithm of BER at output of the equalizer *versus* SINR in dBs (varies from -26 to -8).

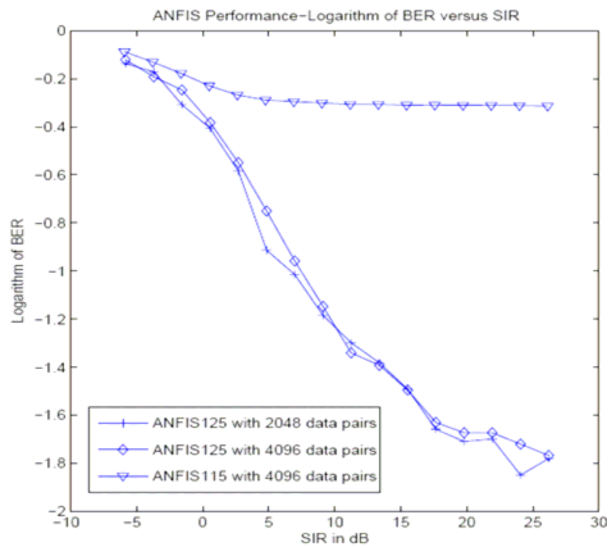

Figure 9: Performance of ANFIS Equalizers–logarithm of BER at output of the equalizer *versus* SIR in dBs (varies from -10 to 30), with standard deviation of AWGN fixed at 0.42.

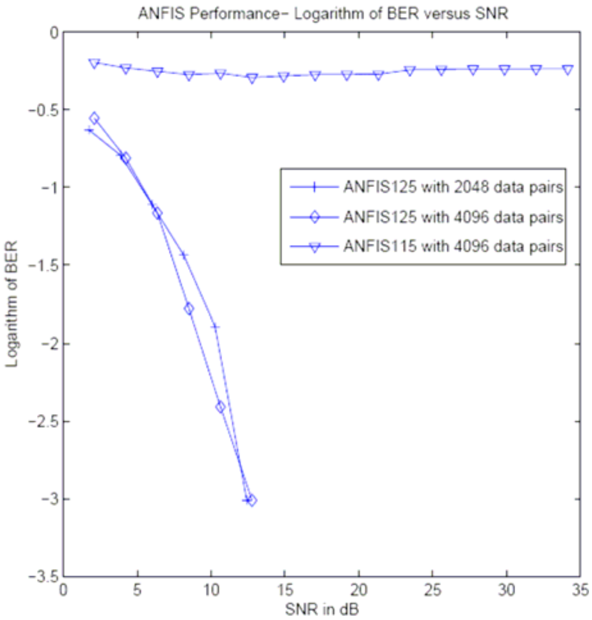

Figure 10: Simulation Results: BER at output of the equalizer (varies from 0 to 0.35) *versus* SNR in dBs (varies from 0 to 35) for 2048/4096 training data pairs with Standard Deviation of CCI fixed at 0.08.

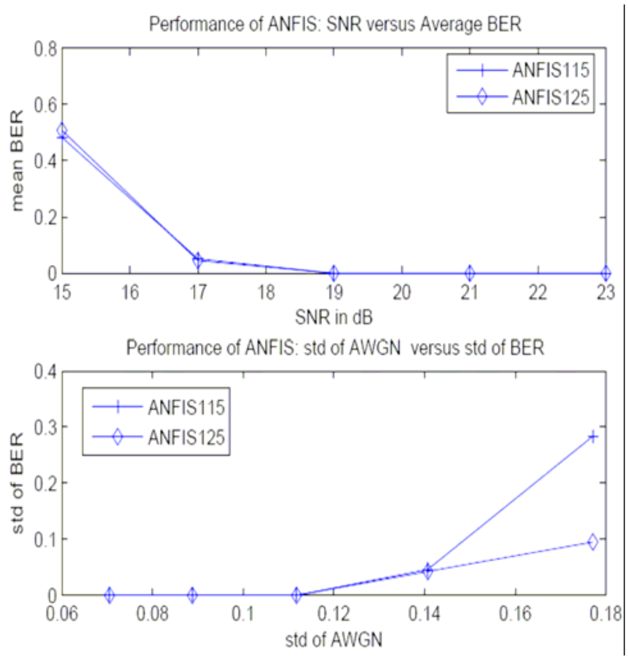

Figure 11: Simulation Results-Performance of ANFIS Equalizers: (a) Mean BER at Output of Equalizer (varies from 0 to 0.8) *versus* SNR in dBs. (varies from 15 to 23dBs) (b) Standard Deviation of BER at output of Equalizer (varies from 0 to 0.4) *versus* Standard Deviation of AWGN. (varies from 0.06 to 0.18)

Table 2: Simulation Time for ANFIS with 1024 training data pairs

ANFIS Type	*Number of Epochs*	*Time in Seconds*
ANFIS–15	20	0.84
ANFIS–17	20	1.04
ANFIS–115	20	2.34
ANFIS–125	20	5.50
ANFIS–25	20	7.85
ANFIS–27	20	26.80
ANFIS–35	20	558.80
ANFIS–37	20	4255.64

V. Conclusion

The following observations are made based on Figures 8, 9, 10 and 11 and Tables 1 and 2 as well as results of simulations with less number of data pairs:

1. With more number of training data pairs, BER at the output of the equalizer is reduced. This is due to the fact that the ANFIS gets optimally tuned with more training data pairs.
2. As the number of rules is increased, the BER at the output of the equalizer is reduced. A finer control is effected by increasing the number of rules, thereby reducing the BER. But this is attained at the cost of more time for ANFIS training.
3. As shown in Figure 8, performance of all ANFIS Equalizers w.r.t. $\log(BER)$ at the output of the equalizer *versus* SINR, is nearly identical. When the SINR is above $-10dB$, practically the $\log(BER)$ becomes close to zero. However, ANFIS–125 performs slightly better than other structures.
4. The performance of ANFIS–125 w.r.t. $\log(BER)$ at the output of the equalizer *versus* SNR is almost identical with 2048 or 4096 data pairs. However for ANFIS–115, performance is poor, even at SIR of $25dB$.
5. As we increase the number of rules or the number of inputs applied in parallel to the ANFIS2X structure, the number of internal nodes and the ANFIS training time increase. This is because, with more number of rules or more internal inputs to the ANFIS, the system can be modeled more accurately.
6. For MISO or MIMO systems, increasing the number of membership functions is the option for accurate system modeling, since in these cases number of inputs applied to the ANFIS is two or more, and hence it will not be optimal to increase the number of internal inputs in ANFIS.
7. An optimal ANFIS structure can be obtained based on the training time and the maximum error that can be tolerated. As indicated in Figure 10, at higher values of standard deviation of AWGN, and that of standard deviation of BER will be less with more number of

membership functions. Hence standard deviation of BER can be yet another criterion in selecting a particular ANFIS structure.

8. The optimal number of fuzzy membership rules can be arrived at by simulation. We can see that the optimum number of fuzzy membership rules for ANFIS–2X is very close to 5.

9. There is an indication that there is indeed an *optimum number of training data pairs*.

10. If one can afford to go for a large number of membership rules, optimal tuning can be attained only if the number of training data pairs is increased to a correspondingly larger extent.

ACKNOWLEDGMENT

The first author wishes to place on record his immense thanks to his wife and daughters for the moral support rendered, during the entire project.

REFERENCES

[1] Widrow B. and Stearns S.D., Adaptive Signal Processing, Englewood Cliffs, New Jersey, Prentice Hall.

[2] Forney G.D., “Maximum-likelihood Sequence Estimation of Digital Sequences in the Presence of Inter-Symbol-Interference”, *IEEE Transactions on Information Theory*, 1972, No.18, pp.363–378.

[3] Cheng J. L. and Wen H. H., “Blind Equalization Using Pseudo-Gaussian-Based Compensatory Neuro-Fuzzy Filters”, *International Journal of Applied Science and Engineering*, Vol.2, No.1, pp.72–89, January 2004.

[4] Gibson G.J., Siu S., and Cowan C.F.N., “The Application of Non-linear Structures to Reconstruction of Binary Signals”, *IEEE Transactions on Signal Processing*,1991, No.39, pp.1877–1884.

[5] Al Mashouq K.A., and Reed I.S., “The Use of Neural Nets to Combine Equalization with Decoding for Severe Inter-Symbol-Interference Channels”, *IEEE Transactions on Neural Networks*, 1994, No.5, pp.982–988.

[6] Chen S., Mulgrew B., and McLaughlin S., “Adaptive Bayesian Equalizer with Decision Feedback”, *IEEE Transactions on Signal Processing*, Vol.41, pp.2918–2927, September 1993.

[7] Chen S., Gibson G.J., Cowan C.F.N., and Grant P.M., “Reconstruction of Binary Signals Using Radial Basis Function Equalizer”, *Signal Processing,* 1991, No.22, pp.77–93.

[8] Kechriotis G., and Manolakos E.S., “Using Recurrent Neural Networks for Adaptive Communication Channel Equalization”, *IEEE Transactions on Neural Networks*, 1994, Vol.5, No.2, pp.267–278.

[9] Wang L.X., and Mendel J.M., “An RLS Fuzzy Adaptive Filter, with Application to Non-linear Channel Equalization”, *IEEE Transactions on Fuzzy Systems*, Vol.1, pp.895–900, August 1993.

[10] Lin C.T., and Juang C.F., “Adaptive Neural Fuzzy Filter and its Applications”, *IEEE Transactions on Systems, Man and Cybernetics* (B),Vol.27, No.4, pp.640–656, April 1994.

[11] Patra S.K., and Mulgrew B., “Fuzzy Techniques for Adaptive Nonlinear Equalization”, 2000, *Signal Processing*, No.80, pp.985–1000.

[12] Liang Q., and Mendel J.M., “Equalization of Nonlinear Time-Varying Channels Using Type–2 Fuzzy Adaptive Filters”, *IEEE Transactions on Fuzzy Systems*, Vol.8, No.5, pp.551–563, October 2000.

[13] Jang J.S.R., “ANFIS: Adaptive Network-Based Fuzzy Inference System”, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.23, No.3, pp.665–685, May/June 1993.

[14] Zadeh L.A., “Outline of a New Approach to the Analysis of Complex Systems and Decision Processes”, *IEEE Transactions on Systems, Man and Cybernetics*, Vol.SMC-3, No.2, pp.28–44, January 1973.

[15] Liang Q., and Mendel J.M., “Overcoming Time-Varying Co-Channel Interference Using Type–2 Fuzzy Adaptive Filters”, *IEEE Transactions on Circuits and Systems–II Analog and Digital Signal Processing*, Vol.47, No.12, pp.1419–1428, December 2000.

[16] Raveendranathan K.C., “Neuro-Fuzzy Controller for Mobile Cellular Channel Equalizers”, Ph.D. Thesis, University of Kerala, 2006.

[17] Raveendranathan K.C., “A Radial Basis Function Framework for Various Adaptive Fuzzy Equalizers Used in Mobile Cellular Channels”, *Proceedings of IEEE R10 International Conference (TENCON2008*), Nov. 18–21, 2008, pp.1–6.

[18] Raveendranathan K.C., “A Class of ANFIS Based Channel Equalizers for Mobile Communication Systems”, *Proceedings of First International Conference of Computational Intelligence, Communication Systems and Networks (CICSyN2009),* .Jul. 23–25, 2009, pp.486-491.