# Chapter 16
# ANFIS Modeling of Dynamic Load Balancing in LTE

**Matthew K. Luka**
*Modibbo Adama University of Technology, Nigeria*

**Aderemi A. Atayero**
*Covenant University, Nigeria*

## ABSTRACT

*Modelling of ill-defined or unpredictable systems can be very challenging. Most models have relied on conventional mathematical models which does not adequately track some of the multifaceted challenges of such a system. Load balancing, which is a self-optimization operation of Self-Organizing Networks (SON), aims at ensuring an equitable distribution of users in the network. This translates into better user satisfaction and a more efficient use of network resources. Several methods for load balancing have been proposed. While some of them have a very buoyant theoretical basis, they are not practical. Furthermore, most of the techniques proposed the use of an iterative algorithm, which in itself is not computationally efficient as it does not take the unpredictable fluctuation of network load into consideration. This chapter proposes the use of soft computing, precisely Adaptive Neuro-Fuzzy Inference System (ANFIS) model, for dynamic QoS aware load balancing in 3GPP LTE. The use of ANFIS offers learning capability of neural network and knowledge representation of fuzzy logic for a load balancing solution that is cost effective and closer to human intuition. Three key load parameters (number of satisfied user in the network, virtual load of the serving eNodeB, and the overall state of the target eNodeB) are used to adjust the hysteresis value for load balancing.*

## INTRODUCTION

Mobile communication systems are unpredictable and stochastic in nature due to a number of factors such as constantly changing propagation channels, random mobility of users and sudden changes in network load. This renders conventional mathematical tools less effective for system modelling of communication systems. Thus communication systems can be best modelled by adopting soft computing which exploits the tolerance for imprecision, partial truth and uncertainty to achieve robustness, low solution cost and tractability. One of such soft computing platforms is the Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS is an architecture which can serve as a basis for constructing a set of fuzzy if-then rules with appropriate membership functions to give the

DOI: 10.4018/978-1-4666-2208-1.ch016

specified input/output pairs model (Jang, 1993). ANFIS modelling have been utilized in a number of applications such modelling of Microarray Cancer Gene Expression Data (Wang, 2005), Speed Control of Induction Motor (Kusagur, Kodad, & Ram, 2010), and for Optimization of Multiple Response Systems (Cheng, Cheng, & Lee, 2002). This chapter proposes the use of ANFIS modelling for dynamic load balancing for the Third Generation Partnership Project (3GPP) Long Term Evolution (LTE).

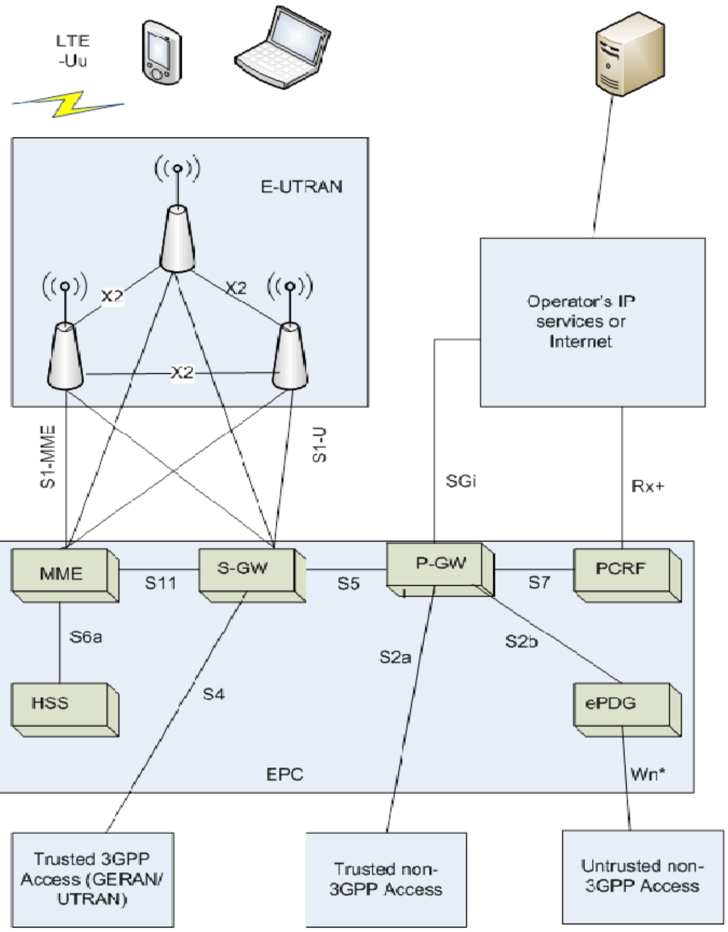
The 3GPP LTE is Self-Organizing Network (SON). Self-Organizing Network operation was introduced to enhance system performance by improving network operations and maintenance. SON operations are also promising in reducing both CAPital EXpenditure (CAPEX) and OPerational EXpenditure (OPEX). Load balancing is a SON operation which aims at ensuring an equitable distribution of cell load among eNodeBs in order to improve the overall system capacity of the network (ETSI TS 136 300, 2011), (M. of WINNER, 2005). To this end, several algorithms have been proposed. In (Lobinger, Stefanski, Jansen, & Balan, 2010), a load balancing algorithm aimed at finding the Optimum Handover (OH) offset value between the overloaded cell and a possible target cell was proposed. Another approach, which is based on a network formulation of heterogeneous services with different quality of service requirements was proposed in (Wang et al, 2010). A utility-based load-balancing framework was used to develop an algorithm called Heaviest-First Load Balancing (HFLB) in (Wang et al, 2010). However, these methods and algorithms are not computationally efficient because they involve the use of iterative processes. Moreover, the need to minimize load overhead due to excessive handover and Ping-Pong effect needs to be taken into consideration. Also, to make a more informed and informed load balancing decision, there is a need to consider not only the load of the serving cell, but other indicators such as the overall state of the serving cell and the number of satisfied users in the entire network must be taken into account. These challenges points to the need for a robust and cost effective approach.

## OVERVIEW OF 3GPP LTE

The Long Term Evolution (LTE) started in 3GPP (Third Generation Partnership Project) release 8 and continued in release 10 with the objective of meeting the increasing performance requirements of mobile broadband (Dahlman, Parkvall, & Skold, 2011). LTE is a new radio-access technology geared towards higher data rates, high spectral efficiency, very low latency, support of variable bandwidth, simple protocol architecture, and support for Self-Organizing Networks (SON) operation. Release 10, otherwise known as LTE advanced is a fourth generation (4G) specification that provides enhanced peak data rates to support advanced services and applications (100 Mb/s for high mobility and 1 Gb/s for low mobility). LTE is the radio access network for Evolved Packet System (EPS), which has a core network known as Evolved Packet Core (EPC). The overall architecture of the EPS is shown in Figure 1.

The LTE radio access network consists of evolved Node Bs (eNodeBs) and no centralized controller (for normal user traffic). Due to the absence of a network controller, it is said to have a flat architecture. This structure reduces system complexity and cost and allows better performance over the radio interface. The eNBs are interconnected by the X2 interface. The S1-MME interface connects the eNBs to the key control plane of the core network-the MME, while the S1-U interface connects the eNBs and the S-GW. Intra-LTE load balancing is usually accomplished over the X2 interface.

*Figure 1. EPS network elements*




## REVIEW OF ANFIS MODELLING

Adaptive Neuro-Fuzzy Inference System (ANFIS) otherwise referred to as Adaptive Network-based Fuzzy inference System was originally proposed in (Jang, 1993). ANFIS is a blend of Fuzzy Logic (FL) and Artificial Neural Network (ANN) that captures the strengths and offsets the limitations of both techniques for building Inference Systems with improved results and enhanced intelligence. Fuzzy logic is associated with the theory of fuzzy set, which relates to classes of objects with rough boundaries in which membership is a matter of degree. It is an extensive of multivalued logical system that departs in concept and substance from the traditional multivalued logical systems. Much of fuzzy logic may be viewed as a platform for computing with words rather than numbers. The use of words for computing is closer to human intuition and exploits the tolerance for imprecision, thereby lowering the cost of the solution (Mathwork Inc., 2011). However, there are no known appropriate or well-established methods of defining rules and membership functions based on human knowledge and experience for fuzzy inference systems. ANFIS uses ANN for adapting these membership functions by adjusting the adaptive parameters associated with the membership functions. Artificial Neural Networks are made up of simple processing elements operating concurrently. These elements model the biological nervous system, with the network functions

*Box 2.*

$$SINR_{c,\ u} = \frac{\alpha_i L_{sf,\ 0,U} L_{pl,\ 0,U} P_1}{\beta_i P_1 + \gamma_i \sigma^2 + \sum_1^{N_{int}} \theta_{i,1} L_{sf,\ T_i,U_j} L_{pl,\ T_i,U_j} P_1} \tag{9}$$

Where

$\alpha_i$ and $\beta_i$ models the channel estimation errors, $P_1 = P_{tx} / v$ represents the homogenously distributed transmit power, $\gamma_i$ models a simple Zero Forcing (ZF) receiver noise enhancement, $\sigma^2$ is the uncorrelated receiver noise and $\theta$ models the interference. $L_{sf,\ T_i,U_j}$ and $L_{pl,\ T_i,U_j} P_1$ stand for the shadow fading and pathloss between the UE, $u$ and its attached eNodeB $c$ (for $T_i = 0)$ and its interferers (for $T_i = 1,\ldots,N_t)$ respectively.

MCS with a higher throughput needs a higher SINR to operate [7]. We assume that the best modulation coding scheme (MCS) is used for a given SINR and the highest data rate $R(SINR)$ is achievable, this can be represented by Shannon formula as shown below:

$$R\left(SINR_u\right) = \log_2(1 + SINR_u) \tag{10}$$

For better approximation to realistic MCS, the mapping function is scaled by attenuation factor (say 0.75) and is bounded by the minimum required SINR (-6.5 dB) and a maximum bitrate (4.8 bps/Hz).

## LOAD METRIC

The specific number of subcarriers allocated to users for a predetermined amount of time is referred to as the Physical Resource Blocks (PRBs) (Zyren & McCoy, 2007). PRBs possess both frequency and time dimension. The eNodeB is responsible for the allocation of PRBs using a scheduling function. The amount of Physical Resource Blocks (PRBs) required by user $u$ can be expressed as:

$$N_u = \frac{D_u}{R\left(SINR\right)_u \cdot BW} \tag{11}$$

where $D_u =$ required data rate and BW is the transmission bandwidth of one resource blocks (180 kHz for LTE). The load of cell $c$ can be expressed as the sum of required resources of all users connected to cell $c$ to the total number of resources $N_t$:

$$\rho_c = \min\left(\frac{\sum_{u:X(u)=c} N_u}{N_t}, 1\right) \tag{12}$$

The total number of available resources (subcarriers) depends on the chosen transmission bandwidth of the system as shown in Table 1 (Holma & Toskala, 2009).

If we chose the number of unsatisfied users as an assessment and simulation metric, then we can focus on the CBR traffic rather than the network throughput. In this case, the UEs either get exactly the CBR or they totally unsatisfied. Equation (12) implies that the cell load parameter should not exceed 1 for all users to be satisfied. This can be extended to give a general indication of how overloaded (or otherwise) a cell is, by defining a virtual load given by:

*Box 1.*

$$z = \overline{w}_1 z_1 + \overline{w}_2 z_2 = \overline{w}_1 (p_1 x + q_1 y + r_1) + \overline{w}_2 (p_2 x + q_2 y + r_2) \quad (6)$$

$$z = \overline{w}_1 z_1 + \overline{w}_2 z_2 = (\overline{w}_1 x) p_1 + (\overline{w}_1 y) q_1 + \left(\overline{w}_1\right) r_1 + \left(\overline{w}_2 x\right) p_2 + \left(\overline{w}_2 y\right) q_2 + r_2 \quad (7)$$

be noted as $L_{mp,\ T_i,U_j}$ where $T_i$ is the i-th transmitter (denoted as 0 for the attached eNodeB and $1,\ldots,N$ for the interfering eNodeBs. $U_j$ is the j-th UE which is located at an $(x,y)$ position. The pathloss was generated using a distance dependent pathloss of $128.1 + 37.6 log_{10}(R[Km])$ (ETSI TR 136 942, 2009) and a $\theta_{3dB} = 65^{\circ}$ / 15dBi antenna (3GPP TR 25.814, 2006). Shadow fading occurs due to obstacles in the propagation path between the eNodeB and UE. Shadow fading can be seen as the changes in the geographical properties of the terrain associated with the mean pathloss derived from the macroscopic pathloss model. It is often approximated by a log-normal distribution of standard deviation 10 dB and mean 0 dB. A UE moving in the Region of Interest (ROI) will experience a slowly changing pathloss due to the shadow fading of the attached eNodeB being correlated with the shadow fading of the interfering eNodeBs. Shadow fading can be denoted by $L_{sf,\ T_i,U_j}$. The large scales fading (shadow fading and pathloss) are position dependent and time-invariant. Small scale fading results primarily due to the presence of reflectors and scatterers that cause multiple versions of the transmitted signal to arrive at receiver. The small scale fading is modelled as a time dependent process for different transmission modes.

LTE supports both Single-Input Single-Output (SISO) and Multiple-Input Multiple-Output (MIMO) transmission techniques. The MIMO transmission modes supported are Transmit Diversity and Spatial Multiplexing. Transmit diversity provides a source of diversity for averaging out the channel variation either for delay sensitive services (Such as voice over internet protocol) at both low and high User Equipment (UE) speeds or for operation at higher UE speeds (Khan, 2009). Transmission diversity is useful for delay sensitive services, but does not help in improving the peak data rates because only a single data stream is always maintained. Spatial multiplexing facilitates achieving higher peak data rates by utilizing the multiple transmission antennas at the eNodeB in combination with multiple receive antennas at the UE. The MIMO OSLM channel can be modelled to obtain the per-layer SINR. This transmission mode consists of a precoding for Spatial Multiplexing (SM) with large-delay Cyclic Delay Diversity (CDD) (ETSI TS 136 211 (2011)). The OLSM MIMO precoding is defined by:

$$\begin{bmatrix} y_{(0)}(i) \\ \vdots \\ y_{(N_t-1)}(i) \end{bmatrix} = W\left(i\right) D\left(i\right) U \begin{bmatrix} x_{(0)}\left(i\right) \\ \vdots \\ x_{(v-1)}\left(i\right) \end{bmatrix} \quad (8)$$

where:

$N_t =$ Number of transmit antennas

$v =$ Number of layers (a layer is a mapping of symbols to the transmit antenna)

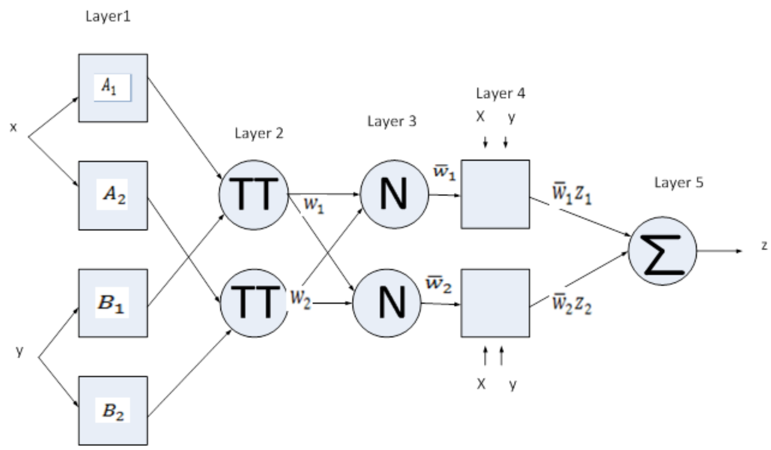
$W\left(i\right) = N_t \times v$ Is the precoding matrix

D and U are $v \times v$ diagonal matrixes introducing the CDD.

For the MIMO OLSM, the SINR for the UE can be expressed as Equation (9) in Box 2.

A given MCS (Modulation Coding Scheme) requires a certain SINR (measured at the receiver of the UE) to operate with an acceptably low BER (Bit Error Rate) in the output data. An

*Figure 2. Type 3 ANFIS architecture*




This layer essentially computes the contribution of each rule to the overall output. It is the defuzzification layer and provides output values resulting from the inference of rules. The parameters in this layer $\left\{p_i, q_i, r_i\right\}$ are known as consequent parameters.

**Layer 5:** There is only one fixed node in this layer. It computes the overall output as the summation of contribution from each rule:

$$\sum_i \overline{w}_i z_i = O_i^5 = \sum_i \frac{w_i z_i}{\sum_i z_i} \tag{5}$$

## HYBRID LEARNING ALGORITHM

The objective of learning is to tune all the adjustable parameters to make the ANFIS output match the desired data. In order to improve the training efficiency, a combination of learning algorithms is adopted to adjust the parameters of the input and output membership functions. The consequent parameters are optimized using the least square method with the antecedent parameters fixed. After updating the consequent parameters, the gradient descent method using back-propagation training algorithm is used to fine-tune the premise parameters (Jang, 1993). Assuming the premise parameters are held fixed, then the overall output of the ANFIS will be a linear combination of the consequent outputs given by that show in Box 1.

## LOAD BALANCING SYSTEM MODEL AND METRIC

The system model is based on a 3GPP downlink multi-cell network serving multiple users with a homogenous QoS requirement. Specifically, constant bit error rate (CBR) users are taken into account. The Signal to Interference Noise Ratio (SINR) is used as a metric measuring the link quality of the link model (M. of WINNER, 2005). Performance analysis is hinged on two factors, namely: fairness distribution of load and the number of unsatisfied users in the network.

### Link Model

The post-equalization symbol SINR was determined from three parts of the link measurement model: (1) shadow fading, (2) macroscopic pathloss and (3) small scale fading (for Multiple-Input-Multiple Output). The propagation pathloss due to distance and antenna gain can be modelled by the macroscopic pathloss between an eNodeB sector and a User Equipment. The pathloss can predominantly determined by the connections between the elements. Neural Networks have the ability to learn from data by adjusting the values of the connections (weights) between the elements. Merging these two artificial intelligence paradigms together offers the learning power of neural networks and the knowledge representation of fuzzy logic for making inferences from observations (input/output data sets).

## BASIC ANFIS ARCHITECTURE

The ANFIS architecture described here is based on type 3 fuzzy inference system (other popular types are the type 1 and type 2). In the type 3 inference system, the Takagi and Sugeno's (TKS) if-then rules are used (Takagi & Sugeno, 1985). The output of each rule is obtained by adding a constant term to the linear combination of the input variables. Final output is then computed by taking the weighted average of each rule's output. The type 3 ANFIS architecture with two inputs (x and y) and one output, z, is shown in Figure 2.

Assuming the rule base contains two first order TKS if-then rules as follows:

$$Rule1 : if\, x\, is\, A_1\, and\, y\, is\, B_1, then\, z_1 = p_1 x + q_1 y + r_1$$

$$Rule2 : if\, x\, is\, A_2\, and\, y\, is\, B_2, then\, z_2 = p_2 x + q_2 y + r_2$$

The ANFIS structure is functionally equivalent to a supervised, feed-forward neural network with one (1) input layer, three (3) hidden layers and one output layer, whose functionality are:

**Layer 1:** Every node in this layer is an adaptive layer that generates the membership grades of the input vectors. A bell-shaped (Gaussian) function with maximum equal to 1 and minimum equal to 0 is often used for implementing the node function:

$$O_i^1 = \mu_{A_i}\left(x\right) = \frac{1}{1 + \left|(x - c_i) / a_i\right|^{2b_i}} \quad (1)$$

where $O_i^1 =$ output of the $ith$ node in the first layer, $\mu_{A_i}\left(x\right)$ is the membership function of input $x$ in the linguistic variable $A_i$. The parameter set $\{a_i, b_i, c_i\}$ are responsible for are responsible for defining the shapes of the membership functions. These parameters are called premise parameters.

**Layer 2:** Each node in this layer determines the firing strength of a rule by multiplying the membership functions associated with the rules. The nodes in this layer are fixed in nature. The firing strength of a particular rule (the output of a node) is given by:

$$w = O_i^2 = \mu_{A_i}\left(x\right) \cdot \mu_{B_i}\left(y\right), i = 1, 2, \ldots \quad (2)$$

Any other T-norm operator that performs fuzzy AND operation can be used in this layer.

**Layer 3:** This layer consists of fixed nodes that are used to compute the ratio of the $ith$ rule's firing strength to the total of all firing strengths:

$$\overline{w} = O_i^3 = \frac{w_i}{w_1 + w_2}, i = 1, 2, \ldots \quad (3)$$

The outputs of this layer are otherwise known as *normalized firing strength* for convenience.

**Layer 4:** This is an adaptive layer with node function given by:

$$\overline{w}_i z_i = O_i^4 = \overline{w}_i (p_i x + q_i y + r_i) \quad (4)$$

*Table 1. PRBs of different downlink bandwidths*

| Bandwidth (MHz) | Physical Resource Blocks $(N_t)$ |
|---|---|
| 1.4 | 6 |
| 3.0 | 15 |
| 5.0 | 25 |
| 10 | 50 |
| 20 | 100 |

$$\hat{\rho}_C = \frac{\sum_{u:X(u)=c} N_u}{N_t} \quad (13)$$

where $\hat{\rho}_C \leq 1$ means all users in the cell are satisfied, $\hat{\rho}_C = U$ means $1/U$ of the users are satisfied

The total number of unsatisfied users in the whole network (With a total number of $M_c$ users in cell $c$) is given by:

$$z = \sum_c \max(0, M_c \cdot (1 - 1/\hat{\rho}_C)) \quad (14)$$

For performance analysis, the use of a fairness distribution index proposed in (Jain, Chiu, & Hawe, 1984) is employed. Thus, the load distribution index measuring the degree of load balancing of the entire network is given as:

$$\mu(t) = \frac{\left(\sum_c \rho_c(t)\right)^2}{|N| \sum_c (\rho_c(t))^2} \quad (15)$$

where $|N|$ is the number of cells in the network (used for simulation) and t is the simulation time. The load balance index $\mu(t)$ takes the value in the interval $\left[\frac{1}{|N|}, 1\right]$. A larger $\mu$ indicates a more balanced load distribution among the cells. Thus, the load distribution index is 1 when the load is completely balanced. The aim of load balancing (for CBR users) is to maximize is to maximize $\mu(t)$ at each time $t$.

In order to improve the load balancing performance among adjacent cells, it is necessary to find the optimum target cell. This can be achieved by adopting a two-layer inquiry scheme proposed in (Zhang et al, 2011). The source eNodeB (the cell requiring load balancing) requests both the load state and environment state from all neighbouring eNBs (first layer cells). The load state is the load of the first layer cell, while the environment state is the average load of the first layer cell's adjacent cells excluding the one to be adjusted (denoted as the second layer cells). The Overall State of the first layer cell $i$ is obtained by a weighted combination of the load state $(LS_i)$ and environment state $(ES_i)$ in one figure as follows:

$$OS_i = \alpha LS_i + (1 - \alpha) ES_i \quad (16)$$

where the environmental state is given by:

$$ES_i = (\rho_{1i} + \rho_{2i} + \ldots + \rho_{ni}) / n = \frac{\sum_{j=1}^{n} \rho_j}{n} \quad (17)$$

$LS_i = \rho_i$, the load of first layer cell $i$, and $\alpha$ is a parameter that indicates the relative contribution of $LS_i$ and $ES_i$ to $OS_i$.

$OS_i$ gives a comprehensive load information of the first layer cell, thereby indicating whether the eNodeB can be a target cell. Taking the value of $\alpha = 0.2$ Equation (17) can be expressed as:

$$OS_i = \left(0.2 \times \rho_i\right) + 0.8 \times \left( \frac{\sum_{j=1}^{n} \rho_j}{n} \right) \qquad (18)$$

## DESIGN OF LOAD BALANCING INFERENCE SCHEME

In the first stage, that is, the fuzzification process, the crisp variables (the virtual load of the source cell, the overall state of the target cell and number of unsatisfied users) are converted into fuzzy (linguistic) variables. The fuzzification maps the three (3) input variables to fuzzy labels of the fuzzy sets. Each linguistic variable has a corresponding membership function. A triangular-shaped membership function was determined to be the most suitable for this scenario. There are three 3 inputs and 3 fuzzified variables; thus the inference system has a set of 27 rules (Figure 5). The 27 rules included in the inference system are given in Figure 3 in the form of a flowchart:

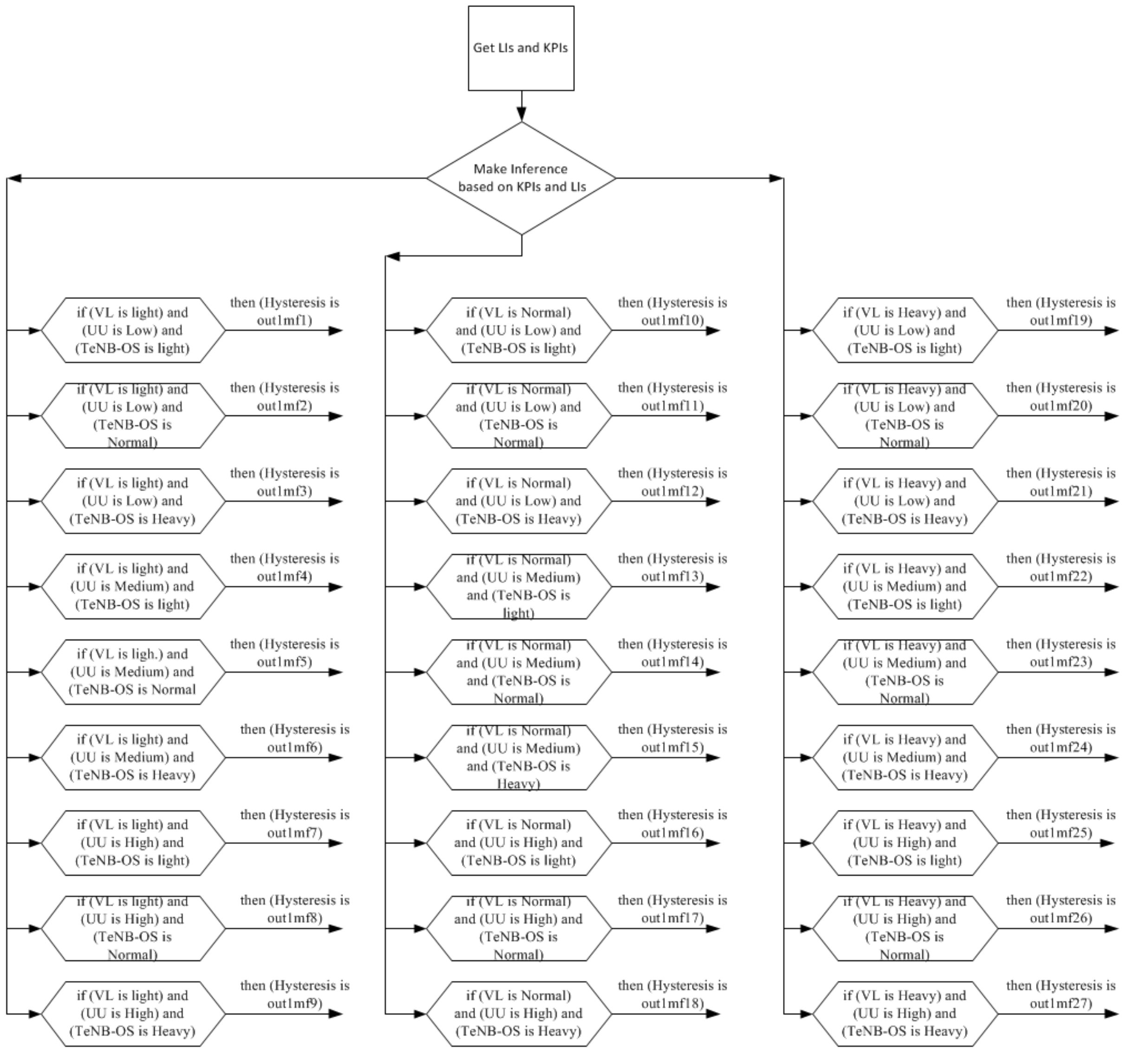
The neural network training helps select the appropriate rule to be fired. Next, the rules are de-fuzzified to produce quantifiable results. De-fuzzification can be achieved using several techniques such as maximum methods, centre of gravity method, centre of singleton method etc. The centre of gravity method is adopted for this work. The de-fuzzified output is then used for making dynamic load balancing decisions. The structure of the ANFIS Model used is depicted in Figure 4. The model consists of 78 nodes, 27 fuzzy rules, 27 linear parameters and 27 nonlinear parameters. The total number of parameter is very important in deciding the number of training data pairs required. In order to realize a good generalization capability, it is recommended to have the number of training data points to be many times larger than the number of parameters being evaluated (Mathwork Inc., 2011). 1500 input/output pairs of training data was used for training. Thus, the ratio between the data points and parameters is about twenty seven times (1500/54).

For parameter optimization, hybrid training (which combines least square errors and back-propagation) was used. To ascertain how well the training data models the load balancing system, model validation using checking and testing data sets was adopted. Model validation involves presenting input/output data sets on which the inference system was not trained to check the degree to which the inference system model predicts the corresponding data set outputs values. This is achieved using the *testing data set*. The second type of data set for model validation is the *checking data set.* The checking data helps prevent the potential of model overfitting of the data, by selecting model parameters that corresponds to the minimum checking data model error. The training, testing and checking data sets used for modelling were obtained from simulation result using a tweaked version of an open source LTE system level simulator (Ikuno, Wrulich, & Rupp, 2010). A sample of training, testing and is given in Table 2.

## SIMULATION RESULTS AND DISCUSSION

The Neuro-fuzzy model was developed using the ANFIS Editor GUI (Graphical User Interface). In the initial stage of the simulation, the UEs in the network are scheduled using the best channel quality indicators (CQIs) scheduling function of the eNodeB. The 27 rules written using the rule editor of the ANFIS Editor GUI is saved as a .fis file. The .fis file is then imported into the simulator using Matlab command line function *readfis*. After running the simulator, the three (3) inputs and one (1) output training parameters are stored in a variable in the command window. The ANFIS is properly trained using the *anfis* Matlab command function which takes the fuzzy rule base, training

*Figure 3. Fuzzy inference system flowchart*

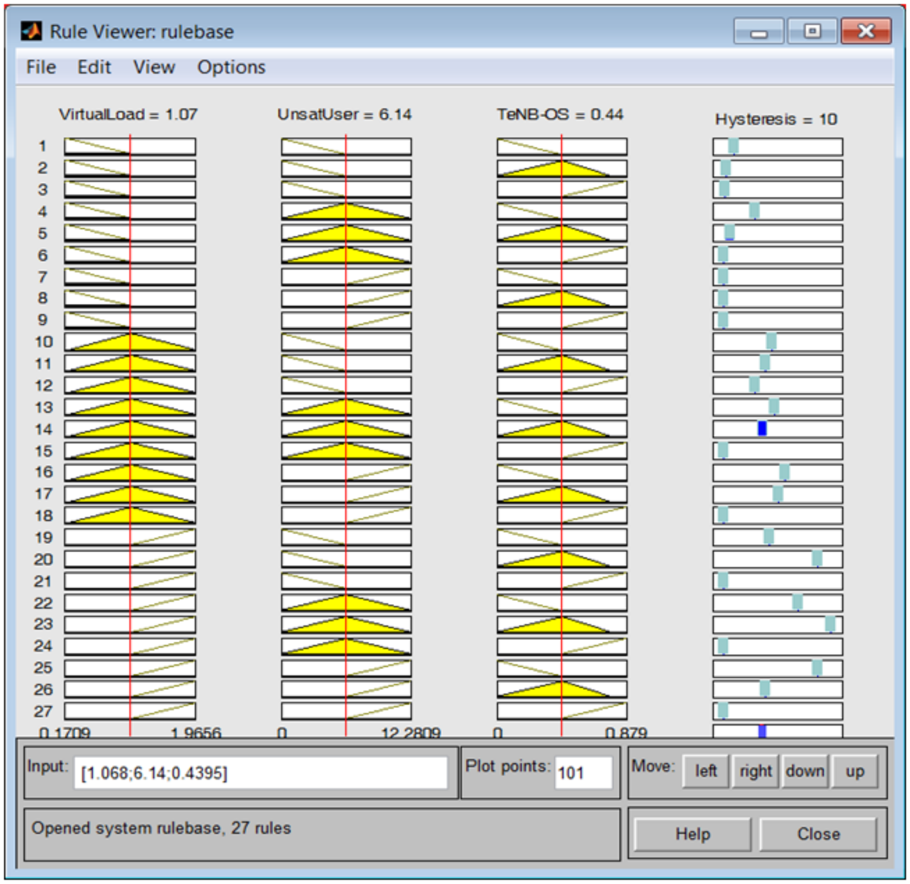



VL=Virtual Load, UU = Unsatisfied Users and TeNB-OS = Overall state of target eNodeB

data and the optional checking data as input arguments. The *checking data set* was used to control the potential of overfitting the data. The checking data and the training data are presented to the ANFIS so that the fuzzy inference model selects parameters associated with the minimum checking data model error. An average checking error of 0.087521 was realized using 1500 input/output checking data set (Figure 6). The ANFIS model was validated using *testing data sets.* The testing data sets were presented to the trained ANFIS to see how well the ANFIS model predicts output values. An average testing error of 0.086525 was achieved for an average training error of 0.0067812 using 20 training epochs.

Having trained and validated the model, it is now ready use in making inferences for load balancing. The simulation is now run with the

*Figure 4. Rule viewer for the inference system*




functionality for load balancing activated and the ANFIS model incorporated. As the simulation runs, the ANFIS model evaluates the hysteresis using the Matlab function *evalfis*. The output is used to decide the number of UEs to be transferred from the overloaded cell in order to improve users' satisfaction and load distribution fairness index which indicates equitable distribution of UEs in the network. The steps needed for the simulation purposes can be summed up in the following steps:

1. Write the rules using the ANFIS rule editor and save the rules as a .fis file in the same directory with the LTE system level simulator.
2. Import the .fis file into the Matlab Work Space (WS).
3. Run the simulator to schedule users and get training parameters for the ANFIS, omitting the evaluation (usage) stage of the ANFIS model.
4. Train the ANFIS using hybrid training algorithm and a suitable number of epochs.
5. Run the simulator again; this time skip the training stage and include the evaluation stage of the ANFIS Model.

The ANFIS model uses the hysteresis value for load balancing. The hysteresis increases as the virtual load of the serving eNodeB increases. This

*Figure 9. Relative contribution of target eNodeB overall state users to hysteresis value*

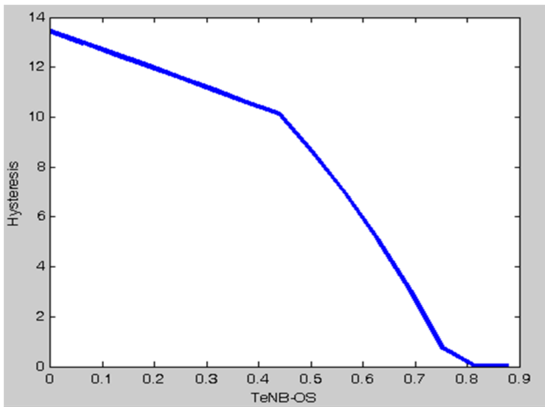
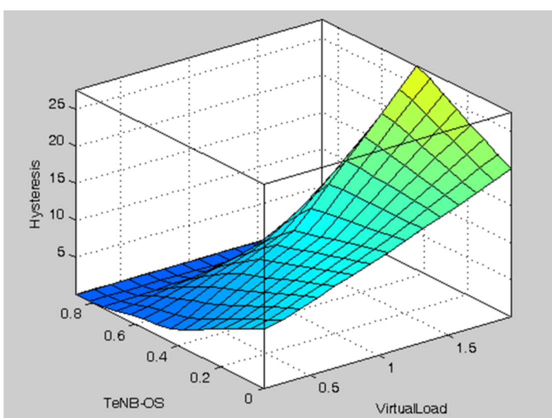



*Figure 10. Combined effect of virtual load and overall state on load balancing hysteresis*




Adaptive Neuro-Fuzzy Inference System has been investigated in this chapter. The ANFIS Model was developed using Matlab and incorporated into an LTE system level simulator. The Inference system of the ANFIS is hinged on 27 fuzzy rules. The main advantage of using the ANFIS model in load balancing is to exploit the tolerance for imprecision and uncertainty associated with

*Figure 11. Combined effect of unsatisfied user and overall state on load balancing hysteresis*

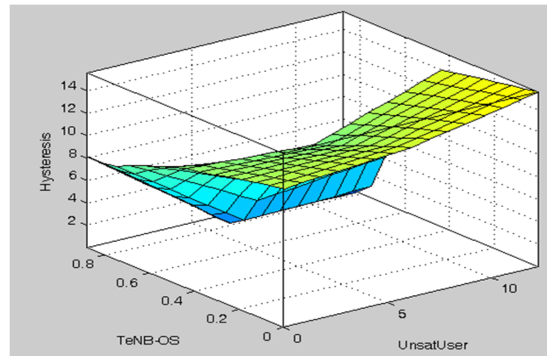
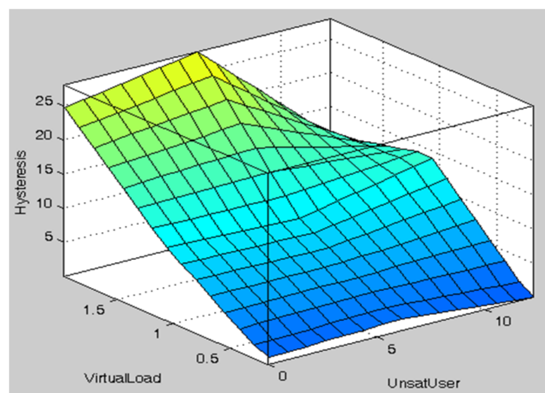



*Figure 12. Combined effect of unsatisfied user and virtual load on load balancing hysteresis*




wireless network to achieve a cost effective load balancing strategy.

The virtual load of the source eNodeB plays a more vital load in determining the output value of the ANFIS model associated with load balancing. The overall load state of the target eNodeB ensures that the target eNodeB is not overload by the source eNodeB by forcing the hysteresis to zero when it is getting overloaded. The number of unsatisfied users in the entire tends to decrease the load balancing hysteresis when the source eNodeB is not

*Figure 6. Model validation using checking data sets*

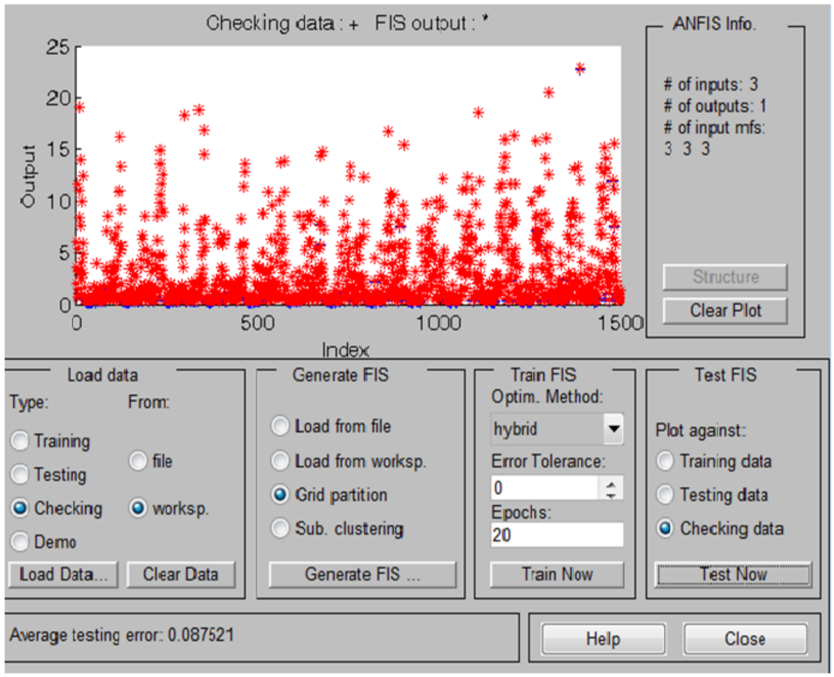



*Figure 7. Relative contribution of virtual load to hysteresis value*

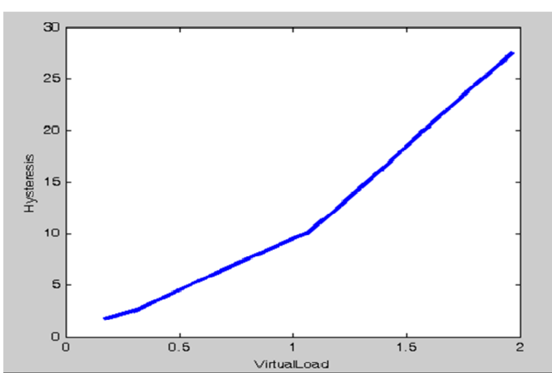
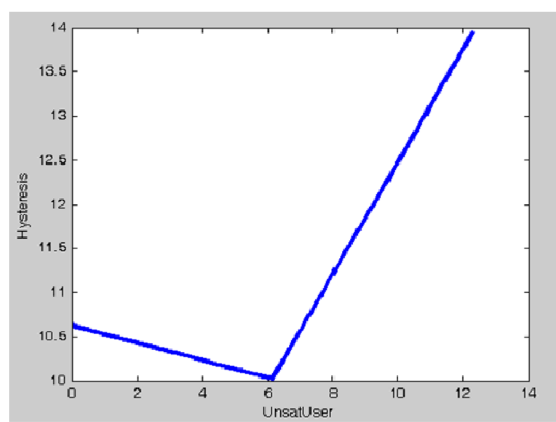



*Figure 8. Relative contribution of unsatisfied users to hysteresis value*




of unsatisfied users. This is because the virtual load is more specific to the source eNodeB, which is directly involved in the load balancing, whereas the number of unsatisfied users is a network wide performance indicator.

## CONCLUSION

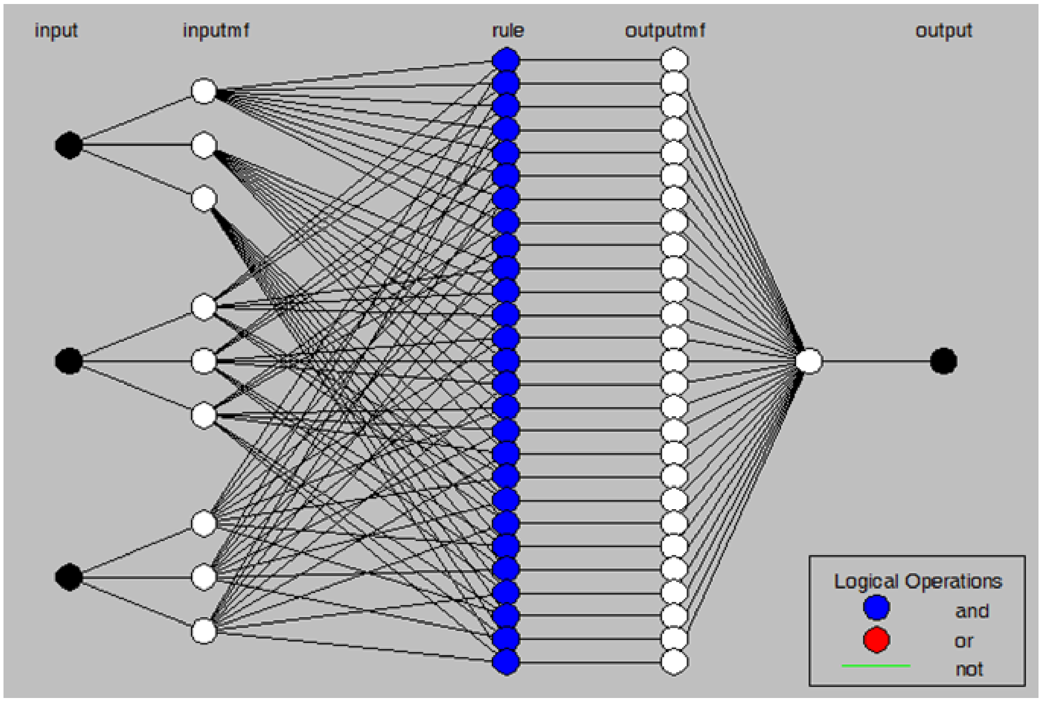
A systematic method of equitably distributing the loads among cells in an LTE network by means of

*Table 2. Training and testing data sets for the ANFIS modelling*

| S/N | Training Data | | | | Testing Data | | | |
|---|---|---|---|---|---|---|---|---|
| | Inputs | | | Output | Inputs | | | Output |
| | $\hat{\rho}_C$ | $z$ | $OS_i$ | Hysteresis | $\hat{\rho}_C$ | $z$ | $OS_i$ | Hysteresis |
| 1 | 1.361042 | 3.390306 | 0.180923 | 12.72999 | 0.353646 | 0 | 0.749619 | 1.766065 |
| 2 | 0.986637 | 0.187466 | 0.286651 | 10.57362 | 0.330942 | 0 | 0.765693 | 1.517912 |
| 3 | 1.156888 | 1.838407 | 0.235804 | 11.60829 | 0.313339 | 0 | 0.778156 | 1.332797 |
| 4 | 1.007555 | 0 | 0.394951 | 8.38957 | 0.303509 | 0 | 0.785115 | 1.232539 |
| 5 | 1.079373 | 0 | 0.48721 | 6.560096 | 0.290151 | 0 | 0.794573 | 1.100192 |
| 6 | 0.854589 | 0 | 0.52705 | 5.783147 | 0.282019 | 0 | 0.80033 | 1.022009 |
| 7 | 0.72428 | 1.68249 | 0.240914 | 11.5041 | 0.288542 | 0 | 0.795712 | 1.084577 |
| 8 | 0.668008 | 0 | 0.427388 | 7.74233 | 0.296069 | 0 | 0.790383 | 1.158243 |
| 9 | 1.072156 | 0 | 0.470869 | 6.881359 | 0.285589 | 0 | 0.797803 | 1.056098 |
| 10 | 0.808774 | 0 | 0.481104 | 6.679979 | 0.769553 | 0 | 0.455156 | 7.191513 |
| 11 | 0.74736 | 0 | 0.422182 | 7.845935 | 0.296266 | 0 | 0.790243 | 1.1602 |
| 12 | 0.732904 | 0 | 0.482266 | 6.657147 | 0.274937 | 0 | 0.805344 | 0.95548 |
| 13 | 0.816126 | 0 | 0.430747 | 7.675532 | 0.330087 | 0 | 0.766298 | 1.508767 |
| 14 | 0.731263 | 0 | 0.51925 | 5.9345 | 0.308539 | 0 | 0.781554 | 1.283552 |
| 15 | 0.80403 | 8.511785 | 0 | 17.97495 | 0.365803 | 0 | 0.741012 | 1.902821 |
| 16 | 0.679026 | 7.971445 | 0 | 17.26954 | 0.350186 | 0 | 0.752068 | 1.727625 |
| 17 | 1.516234 | 3.992071 | 0.157461 | 13.21086 | 0.473118 | 0 | 0.665032 | 3.197888 |
| 18 | 1.468122 | 8.975134 | 0 | 18.61837 | 0.711819 | 0 | 0.496032 | 6.387242 |
| 19 | 1.190027 | 3.390306 | 0.180923 | 12.72999 | 0.446675 | 0 | 0.683754 | 2.866685 |
| 20 | 1.560075 | 0.187466 | 0.286651 | 10.57362 | 0.438088 | 0 | 0.689834 | 2.760585 |

overloading the target eNodeB. Wedging the number of unsatisfied users against TeNB-OS yields a similar result. As the number of unsatisfied users increase from 0 to 10, the hysteresis value increases correspondingly. However, a corresponding increase in the TeNB-OS tends to reduce the hysteresis value in order to force the serving eNodeB to choose another target cell as the current target cell's overall load state approaches (0.8 see Figure 11). The interplay between virtual load and the number of unsatisfied users is illustrated in Figure 12. The impact virtual load on determining load balancing (hysteresis) is more pronounced than that of the number

*Figure 5. ANFIS model structure*




increase is gradual before the cell is overloaded $(\hat{\rho}_C \leq 1)$. However, when the serving eNodeB gets overloaded $(\hat{\rho}_C > 1)$, the hysteresis value increases rapidly Figure 7. Before the number of unsatisfied users reaches a certain threshold (in this particular case, 6), the hysteresis tend to decrease. This is due to overriding influence of other parameters (especially the overall state of the target eNodeB). However, the trend changes spontaneously when the number of unsatisfied users becomes significant (or reaches a certain threshold). The slope of increase in hysteresis when the threshold is attained is much higher than the rate of decrease experienced earlier: see Figure 8. The change in hysteresis due to the overall state of the target eNodeB (TeNB-OS) depicts a completely different trend from that of the other two indicators. Generally, the TeNB-OS sets a check on the value of the hysteresis due virtual load and number of unsatisfied users. Between TeNB-OS values of 0.0 and 0.45, the decrease in hysteresis due to TeNB-OS is gradual. The rate of decline in hysteresis becomes more pronounced between TeNB-OS values of 0.45 and 0.75. When the TeNB-OS approaches the value of 1.0, it forces the hysteresis to zero, indicating that the target eNodeB cannot accept more loads even if the source eNodeB is still overloaded, see Figure 9. When this happens, the serving eNodeB is forced to choose another target eNodeB.

When the virtual load of the serving eNodeB is benchmarked against the TeNB-OS of the selected target eNodeB, their respective effects is depicted in Figure 10. If the decrease in hysteresis value due TeNB-OS does not offset the increase in hysteresis value due to virtual load, another target eNodeB will be selected (when TeNB-OS for the currently selected target eNodeB reaches 0.8). This is necessary in order to avoid overloaded and increase the hysteresis when the source cell is overloaded because it is a network wide parameter. The results obtained from model validation using testing and checking data sets shows that the ANFIS model is robust tool for a dynamic load-balancing scheme in 3GPP LTE.

## REFERENCES

Alamouti. (1998). A simple transmit diversity technique for wireless communications. *IEEE Journal on Selected Areas in Communications.*

Cheng, Cheng, & Lee. (2002). Neuro-fuzzy and genetic algorithm in multiple response optimization. *International Journal of Computers & Mathematics with Applications*, *44*, 1503–1514. doi:10.1016/S0898-1221(02)00274-2.

Dahlman, Parkvall, & Skold. (2011). *4G LTE/LTE-advanced for Mobile Broadband.* London: Elsevier.

ETSI TR 136 942. (2009). LTE; evolved universal terrestrial radio access (E-UTRA); radio frequency (RF) system scenarios. *Technical Report Version 8.2.0.* Retrieved from http://www.3gpp.org

ETSI TS 136 201. (2011). LTE; evolved universal terrestrial radio access (E-UTRA); LTE physical layer; general description. *Technical Specification Version 10.0.0.* Retrieved from http://www.3gpp.org

ETSI TS 136 211. (2011). LTE; evolved universal terrestrial radio access (E-UTRA); physical channels and modulation. *Technical Specification Version 10.2.0.* Retrieved from http://www.3gpp.org

ETSI TS 136 300. (2011). LTE; evolved universal terrestrial radio access (E-UTRA) and evolved universal terrestrial radio access network (E-UTRAN); overall description; stage 2. *Technical Specification Version 10.4.0.* Retrieved from http://www.3gpp.org

3. GPP TR 25.814. (2006). Physical layer aspects for E-UTRA. *Technical Specification Version 7.1.0.* Retrieved from http://www.3gpp.org

Holma & Toskala. (2009). *LTE for UMTS: OFDMA and SC-FDMA based radio access*. New York: Wiley & Sons.

Ikuno, Wrulich, & Rupp. (2010). System level simulation of LTE networks. In *Proceedings of Vehicular Technology Conference (VTC '10).* Taipei, Taiwan: IEEE Press.

Jain, Chiu, & Hawe. (1984). A quantitative measure of fairness and discrimination for resource allocation in shared systems. *Technical Report: Digital Equipment Corporation, DEC-TR-301.*

Jang. (1993). ANFIS: Adaptive network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics, 23*(3), 665-685.

Khan. (2009). *LTE for 4G mobile broadband: Air interface technologies and performance.* Cambridge CB2 8RU. Cambridge, UK: Cambridge University Press.

Kusagur, K., & Ram, S. (2010). Modeling, design, & simulation of an adaptive neuro-fuzzy inference system (ANFIS) for speed control of induction motor. *International Journal of Computers and Applications*, *6*(12), 29–44. doi:10.5120/1123-1472.

Lobinger, S. Jansen, & Balan. (2010). Load balancing in downlink LTE self-optimizing networks. In *Proceedings of IEEE 71st Vehicular Technology Conference 2010.* Taipei, Taiwan: IEEE Press.

M. of WINNER. (2005). Assessment of advanced beamforming and MIMO technologies. *WINNER, IST-2003-507581.*

Mathwork Inc. (2011). *Fuzzy logic toolbox user guide version 2.2.14.* Retrieved from www.mathworks.com

Sesia, Toufik, & Baker. (2009). *LTE-The UMTS long term evolution: From theory to practice.* New York: Wiley & Sons, Ltd.

Takagi & Sugeno. (1985). Fuzzy identification of systems and its application to modelling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, *15*, 116–132.

Wang. (2005). Neuro-fuzzy modeling for microarray cancer gene expression data. *First Year Transfer Report*. Oxford, UK: Linacre College, Oxford University, Computing Laboratory.

Wang. (2010a). Dynamic load balancing in 3GPP LTE multi-cell networks with heterogeneous services. In *Proceedings of ICST Conference*. Beijing, China: LNCIST.

Wang. (2010b). *Dynamic load balancing and throughput optimization in 3GPP LTE networks (IWCMC 2010)*. Caen, France: IWCMC.

Zhang, Liu, Zhang, Jia, & Duan. (2011). A two-layer mobility load balancing in LTE self-organization networks. In *Proceedings of IEEE Internal Conference on Communication Technology*. Beijing: IEEE Press.

Zyren & McCoy. (2007). Overview of the 3GPP long term evolution physical layer. (White Paper) *Document number:3GPPEVOLUTIONWP Rev. 0: Freescale semiconductor.*

## ADDITIONAL READING

Ali-Yahiya. (2011). *Understanding LTE and its performance*. Berlin: Springer.

Atayero, A. A., & Luka, M. K. (2012). Adaptive neuro-fuzzy Inference System for load balancing in 3GPP LTE. *International Journal of Advanced Research in Artificial Intelligence*, *1*(1), 11–16.

Atayero, A. A., & Luka, M. K. (2012). A soft computing approach to dynamic load balancing in 3GPP LTE. *International Journal of Computers and Applications*, *43*(19), 35–41. doi:10.5120/6213-8895.

Atayero, A. A., Luka, M. K., Orya, M. K., & Iruemi, J. O. (2001). 3GPP long term evolution: Architecture, protocols, and interfaces. *IJICT Journal*, *1*(7), 307–310.

Corporation, N. E. C. (2009). *Self-organizing network: NEC's proposal for next-generation radio network management* (White Paper). NEC Corporation.

Dahlman, P. Sköld, & Beming. (2007). 3G evolution HSPA and LTE for mobile broadband. New York: Elsevier.

Ergen. (2009). *Mobile broadband including WiMAX and LTE*. Berlin: Springer.

Ericsson. (2009). Radio network measurement guidelines. Ericsson WCDMA Radio Access Network: Ericsson Internal Document, 1(64).

Fakhreddine, O. Karray, & de Silva. (2004). Soft computing and intelligent systems design. Upper Saddle River, NJ: Pearson, Education, Ltd.

Jang, Sun, & Mizutani. (1997). *Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence*. Englewood Cliffs, NJ: Prentice Hall.

Konar, A. (1999). *Artificial intelligence and soft computing: Behavioural and cognitive modelling of the human brain*. Boca Raton, FL: CRC Press. doi:10.1201/9781420049138.

Luka & Atayero. (2012). *Adaptive neuro-fuzzy inference system models for dynamic load balancing in 3GPP LTE*. Berlin: Lambert Academic Publishing. Retrieved from http://qr.net/anfis2

Mehlführer, C. (2011). The Vienna LTE simulators – Enabling reproducibility in wireless communications research. *EURASIP Journal on Advances in Signal Processing*. doi:10.1186/1687-6180-2011-29.

Ofcom Consultants. (2011). *LTE technical modelling: Revised methodology. 800 MHz & 2.6 GHz Combined Award–Technical Modelling for Competition Assessment*. Canada: Annex Business Media.

Piro. (2010). Simulating LTE cellular systems: An open source framework. *IEEE Transaction on Vehicular Technology.*

Proakis & Salehi. (1998). *Contemporary communication systems using Mathlab*. Boston: PWS Publishers.