Artificial Neural Network based Parameter Estimation and Design Optimization of Loop Antenna

Kumaresh Sarmah and Kandarpa Kumar Sarma

Abstract—Artificial Neural Network (ANN)s are best suited for prediction and optimization problems. Trained ANNs have found wide spread acceptance in several antenna design systems. Four parameters namely antenna radiation resistance, loss resistance, efficiency, and inductance can be used to design an antenna layout though there are several other parameters available. An ANN can be trained to provide the best and worst case precisions of an antenna design problem defined by these four parameters. This work describes the use of an ANN to generate the four mentioned parameters for a loop antenna for the specified frequency range. It also provides insights to the prediction of best and worst-case design problems observed in applications and thereby formulate a model for physical layout design of a loop antenna.

Keywords—MLP, ANN, parameter, prediction, optimization.

I. INTRODUCTION

Artificial Neural Network (ANN)s as non-parametric tools are used for a host of pattern recognition and related applications. These can also be used for prediction and optimization problems. ANNs like the Multi Layer Perceptrons (MLPs) trained with (Error) Back Propagation (BP) [1] in particular have found wide spread acceptance in several antenna design systems. In such cases design parameters have been optimized to suit the requirements. One notable aspect in these cases has been the fact that design requirements can be estimated with controlled precision - a characteristic feature for which ANNs are widely adopted for such applications [2], [3], [4], [5]. A loop antenna can be used throughout the HF to VHF band. Out of several characteristic parameters, four can provide a sketch of the antenna-behaviour [6], [7]. These four parameters are antenna radiation resistance, loss resistance, efficiency, and inductance. The basic guidelines for the antenna design can be fixed in many ways. The most practiced method is tedious theoretical calculations and the other includes software approaches. Custom built software tools provide ready made solutions. These have the advantage of offering the readily derivable solutions considering a host of pre-defined constraints to match the perceived scenarios. But several limitations exist in such design tools. One is the inability of these tools to control precision and provide prediction. They also cannot deal with unforeseen situations and events controlled by random behaviour of transmit-receive conditions of a communication set-up. The ability of custom designed softwares to provide ready- made solutions can go haywire if input parameters suffer fluctuations and deviate from the required values. Moreover the custom made softwares cannot provide optimized forms of the output. Also, such software solutions cannot be modified to suit extensive transmit-receive conditions observed in applications. In situations like imprecise tuning due to faulty or impure components during reception, a frequency dependant parameter will deviate from the desired value. This can lead to design alterations triggered by variations in the parameters leading to faulty antenna layout.

An ANN offers a solution for such situations. MLPs can be trained to provide the best and worst case precisions of an antenna design problem. A properly configured ANN can acquire knowledge about antenna parameter distributions applied to it as learning patterns during training and act as expert system during testing. Such abilities make ANN an effective design tool with the provision of providing controlled prediction. Due to their ability to provide controlled prediction, ANNs can deal with optimization problems- an advantage which can simplify physical design issues. This work is an attempt in that direction. Here, a loop antenna designed for frequency range between 3 to 300 MHz is considered. The following sections describe the use of an ANN to generate the four mentioned parameters for a loop antenna for the specified frequency range. It also provides insights to the prediction of best and worst-case parameter estimation as inputs to antenna design problems observed in applications.

The model includes an ANN which is configured to accept frequency as the input parameter for the mentioned range and provide the four parameters as the output. Several ANN configurations are used to ascertain the best set-up for the testing. The ANNs trained with (Error) Back Propagation (BP) show different results for different training methods. Also, the outcome varies depending upon the number of training sessions and the data used. Mean square error (MSE) convergence and prediction precision are used to ascertain the performance of the ANNs during training. The trained ANNs are used for testing. Data considered includes samples with variance upto 50%. The results show that the ANNs are robust enough in prediction of the parameters. Success rates generally observed are in the 90-95 % range but in certain cases the results are around the 99 %.

Kumaresh Sarmah is student of the Department of Electronics and Communication Engineering, Tezpur University, Assam, India, and Kandarpa Kumar Sarma is with Department of Electronics and Communication Technology, Gauhati University, Guwahati-781014, Assam, India. e-mail: ( kumareshr10@gmail.com and kandarpaks@gmail.com ).
II. Four Basic Parameters of a Loop Antenna and Proposed Model

Loop antennas can have various shapes: circular, triangular, square, elliptical etc. They are widely used in applications up to the microwave bands. The loop antenna is perfect for portable instruments which are not too demanding in terms of range. The main advantage of the loop antenna is its cost - it can be readily fabricated for less demanding applications. The loop antenna is a conductor bent into the shape of a closed curve such as a circle or a square with a gap in the conductor to form the terminals as shown in Figures 1 and 2. There are two types of loop antennas - electrically small loop antennas and electrically large loop antennas. If the total loop circumference is very small as compared to the wavelength, then the loop antenna is said to be electrically small. An electrically large loop antenna typically has its circumference close to a wavelength.

Four parameters namely antenna radiation resistance, loss resistance, efficiency, and inductance - can be used to obtain a sketch of an antenna behaviour. These parameters are essential because these are directly related to the physical dimension of the antenna and frequency of application [6], [7]. Radiation resistance of a loop antenna is defined as the resistance in series with the antenna that will consume the same amount of power as it actually radiated. It is the ratio between power radiated by the antenna and the square of the root mean square value of the maximum current flowing in the antenna at the best matched condition. The radiation and loss resistances of an antenna determine the radiation efficiency. It is the ratio between the radiation resistances to the total resistance of the loop. Loss resistance of an antenna depends on several factors. A few of them are as below:

- Resistance due to dielectric losses,
- Resistance due to dielectric losses,
- Brushing losses,
- Loss by leakage over insulation,
- Resistance due to conductor losses in antenna and earth and
- Eddy current losses.

The proposed work is related to the formulation of an ANN based system which predicts the four mention antenna parameters such that the system maybe defined as:

$$P_{ij} = F(f_i)$$

(1)

where,

$$P_j = [P_1, P_2, P_3, P_4]$$

(2)

such that $P_1$ is Radiation resistance, $P_2$ is Loss resistance, $P_3$ is Efficiency and $P_4$ is Inductance and $F(\cdot)$ is a trained MLP.

These four parameters are directly related to the physical parameters and frequency of application [6] [7]. The parameters determine the physical layout of an antenna design. Their values are governed by the frequency of application. Hence, frequency of operation becomes an important factor in the layout design of the antenna. This is reflected by the equations 3 to 6.

Radiation Resistance of a loop antenna with a being the loop radius and b being the wire radius is given below:

$$R_r = 31.171 \left\{ \frac{S^2}{\lambda^2} \right\}$$

(3)

where, $S = \text{Loop area} = \pi a^2, \lambda = \text{Wavelength},$ and loss resistance is given by:

$$R_l = \left( \frac{\pi a}{b} \right) \sqrt{ \frac{f \mu_0}{\pi \delta}}$$

(4)

where $N$ stand for number of turns. The constant parameters are, $\sigma = 5.8 \times 10^7 \text{ mho/m}$ and $\mu_0 = 4\pi \times 10^{-7}$.

The radiation and loss resistance of an antenna determine the radiation efficiency. It is the ratio between the radiation resistance to the total resistance of the loop.

$$\epsilon_{loop} = \frac{R_r}{R_r + R_l}$$

(5)

where $R_r =$ Radiation resistance and $R_l =$ Loss resistance. Approximate formula for the reactance is given below.

$$X_A_{loop} = 2\pi f \mu_0 a [\ln \left( \frac{8a}{b} \right) - 2]$$

(6)
Antenna resistance shows dependence on frequency. This is due to the reactive nature generated by the components constituting the antenna.

Let \( P_{ij} \) be the output of \( F(.) \) corresponding to an input vector \( f_i \).

For each \( f_i \)

\[
\tilde{P}_{ij} \leftarrow F(f_i)
\]

\[
e = P_{ij} - \tilde{P}_{ij}
\]

\[-\text{do}
\]

\{ training of \( F(.) \) with sample of \( P_{ij} \)

while \( e \rightarrow 0 \)

\} end

The mathematical consideration of \( F(.) \) and its training is provided below.

A. Design Consideration of MLP

A simple perceptron is a single McCulloch-Pitts neuron trained by the perceptron algorithm is given as:

\[
O_x = g([w],[x]) + b \tag{7}
\]

where \([x]\) is the input vector, \([w]\) is the associated weight vector, \(b\) is a bias value and \(g(x)\) is the activation function. Such a setup, namely the perceptron will be able to classify only linearly separable data. A MLP, in contrast, consists of several layers of neurons. The equation for output in a MLP with one hidden layer is given as:

\[
O_x = \beta_i g([w], [x]) + b_i \tag{8}
\]

where \(\beta_i\) is the weight value between the \(i^{th}\) hidden neuron, \([w]\) is the vector of weights between the input and the hidden layer, \([x]\) is the vector of inputs and \([b]\) is the input bias of the hidden neuron layer. Such a setup maybe depicted as in Figure 3. The process of adjusting the weights and biases of a perceptron or MLP is known as training. The perceptron algorithm (for training simple perceptrons) consists of comparing the output of the perceptron with an associated target value. The most common training algorithm for MLPs is error back propagation.

B. Application of Error Back Propagation for MLP training

The MLP is trained using (error) Back Propagation (BP) depending upon which the connecting weights between the layers are updated. This adaptive updating of the MLP is continued till the performance goal is met. Training the MLP is done in two broad passes - one a forward pass and the other a backward calculation with error determination and connecting weight updating in between. Batch training method is adopted as it accelerates the speed of training and the rate of convergence of the MSE to the desired value. The steps are as below:

- **Initialization**: Initialize weight matrix \( W \) with random values between \([0, 1]\). \( W \) is a matrix of \(1 \times 100\) where 100 frequencies between 3 MHz to 300 MHz are used.

- **Presentation of training samples**: Input is \( p_m = [p_{m1}, p_{m2}, \ldots, p_{mL}] \). The desired output is \( d_m = [d_{m1}, d_{m2}, \ldots, d_{mL}] \).

  - Compute the values of the hidden nodes as:

    \[
    net_{mj}^h = \sum_{i=1}^{L} w_{ji}^h p_{mi}^t + b_{j}^h \tag{9}
    \]

  - Calculate the output from the hidden layer as

    \[
    o_{mj}^h = f_{j}^h (net_{mj}^h) \tag{10}
    \]

  where \( f(x)= \frac{1}{1+e^{-x}} \) or \( f(x)= e^{-x} \) depending upon the choice of the activation function.

  - Calculate the values of the output node as:

    \[
    o_{mk}^o = f_{k}^o (net_{mk}^o) \tag{11}
    \]

- **Forward Computation**: Compute the errors:

  \[
  e_{jn} = d_{jn} - o_{jn} \tag{12}
  \]

  Calculate the mean square error (MSE) as:

  \[
  MSE = \frac{\sum_{j=1}^{M} \sum_{n=1}^{L} e_{jn}^2}{2M} \tag{13}
  \]

  Error terms for the output layer is:

  \[
  \delta_{mk}^o = \phi_{mk}^o (1-o_{mk}^o) e_{mn} \tag{14}
  \]

  Error terms for the hidden layer:

  \[
  \delta_{mk}^h = \phi_{mk}^h (1-o_{mk}^h) \sum_j \delta_{mj}^o w_{jk} \tag{15}
  \]

- **Weight Update**:

  - Between the output and hidden layers

    \[
    w_{kj}^o (t+1) = w_{kj}^o (t) + \eta \delta_{mk}^o o_{mj} \tag{16}
    \]
where $\eta$ is the learning rate ($0 < \eta < 1$). For faster convergence a momentum term $\alpha$ maybe added as:

$$w_{kj}^{o}(t+1) = w_{kj}^{o}(t) + \eta \delta_{m}^{o} s_{m} + \alpha (w_{kj}^{o}(t+1) - w_{kj}^{o}(t))$$

(17)

Between the hidden layer and input layer:

$$w_{ji}^{h}(t + 1) = w_{ji}^{h}(t) + \eta \delta_{m}^{h} p_{i} + \alpha (w_{ji}^{h}(t + 1) - w_{ji}^{h}(t))$$

(18)

One cycle through the complete training set forms one epoch. The above is repeated till MSE meets the performance criteria while the number of epochs elapsed is counted.

The data generated to carry out the training of the ANNs show variations of the order of $10^{-3}$ to $10^{3}$. This sort of data is not suitable for training the ANNs as it generates inconsistency in the learning curve convergence. Hence, each of the data sets of the four parameters are normalized and applied for training the ANNs. Though the randomness disappears, minor oscillations remain and the training continues without much difficulty. The learning curves show no indication of getting stuck to some local minima in the error surface. The ANNs trained using this normalized training set also generates a similar data set during testing which is de-normalized and converted back to the required. The ANNs trained with (Error) Back Propagation (BP) show different results for different training methods. Also, the outcome varies depending upon the number of training sessions and the data used. Mean Square Error (MSE) convergence and prediction precision are used to ascertain the performance of the ANNs during training. The training continues till the MSE convergence attains the desired goal and the accuracy of the ANN reaches the required precision level. The performance of training of the ANN also is dependent on the training method used. Hence four different training methods are used to ascertain the performance of the ANN and determine the best configuration. The four methods used for training are Gradient Descent (GDBP), Gradient Descent with Momentum BP (GDBMP), Gradient Descent with Adaptive Learning Rate BP (GDLRBP) and Gradient Descent with Adaptive Learning Rate and Momentum BP (GDALRMBP).

- **Gradient Descent (GDBP)**: In this back-propagation method the training will continue as long as the network has its weight, net input, and transfer functions generate derivative functions. Back-propagation is used to calculate derivatives of performance with respect to the weight and bias variables. Each variable is adjusted according to gradient descent with momentum a specific value and it depends on the previously changed weight or bias passing with every epochs with a given learning rate.

- **Gradient Descent with Adaptive Learning Rate BP (GDALBP)**: The ANN is trained using a learning rate that changes adaptively to adjust as per the requirements of the learning.

- **Gradient Descent with Momentum and Adaptive Learning Rate BP (GDMALRBP)**: This method is used to train any network as long as its weight, net input, and transfer functions have derivative functions. Each variable is adjusted according to gradient descent with momentum constant, and also depends upon previously changed weights. For each epoch, the performance decreases toward the goal, then the learning rate is increased by a specific factor. If performance increases by more than the specified factor, the learning rate is adjusted and the change, which increased the performance, is not made.

The training method GDMALRBP is found experimentally to be better suited for the present work (Table I). It generates better precision at least training time. Table I shows some of the derived results for an electrically small circular loop antenna for training epochs between 1000 and 7000. All learning methods are batch learning methods where, weights are updated only after the entire set of training has been presented to the network. Thus the weight update is only performed after every epoch.

<table>
<thead>
<tr>
<th>Sessions</th>
<th>GDBP</th>
<th>GDMBP</th>
<th>GDALBP</th>
<th>GDMALRBP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>84%</td>
<td>86.6%</td>
<td>88.4%</td>
<td>90.2%</td>
</tr>
<tr>
<td>2000</td>
<td>84.3%</td>
<td>87.3%</td>
<td>89.1%</td>
<td>92.3%</td>
</tr>
<tr>
<td>3000</td>
<td>85.8%</td>
<td>88.5%</td>
<td>91.3%</td>
<td>93.8%</td>
</tr>
<tr>
<td>4000</td>
<td>86.7%</td>
<td>89.9%</td>
<td>92.8%</td>
<td>95.6%</td>
</tr>
<tr>
<td>5000</td>
<td>87.8%</td>
<td>91.5%</td>
<td>93.6%</td>
<td>97.1%</td>
</tr>
<tr>
<td>6000</td>
<td>89%</td>
<td>92.3%</td>
<td>94.8%</td>
<td>98.1%</td>
</tr>
<tr>
<td>7000</td>
<td>90.5%</td>
<td>93.6%</td>
<td>96.1%</td>
<td>99.4%</td>
</tr>
</tbody>
</table>

III. ALGORITHMIC STEPS OF PRESENT WORK

The entire work can be represented by Figure 4 and described by the following steps:

**Training**: The first set of data used for training the MLP represents the ideal conditions around a loop antenna where the design parameters follow text-book definitions. The training phase of the MLP should be robust enough to deal with the variations of transmit-receive considerations. After the configuration of the ANN has been set, the subsequent stages involve the following:

- Generation of training data- It contains the four parameters. The input is a row vector consisting of a set of frequencies between 3 to 300 MHz. These are used for training the network.
- Training of the neural networks- Several training methods of (Error) Back Propagation (BP) are used to ascertain the
Table II

<table>
<thead>
<tr>
<th>Case</th>
<th>ANN trained by data</th>
<th>ANN tested by Data</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Noise free</td>
<td>Noise free and mixed</td>
<td>Ideal conditions</td>
</tr>
<tr>
<td>2</td>
<td>Noise mixed, variance 10%</td>
<td>Noise free and mixed</td>
<td>Practical condition</td>
</tr>
<tr>
<td>3</td>
<td>Noise mixed, variance 30%</td>
<td>Noise free and mixed</td>
<td>Practical condition</td>
</tr>
<tr>
<td>4</td>
<td>Noise mixed, variance 50%</td>
<td>Noise free and mixed</td>
<td>Worst case condition</td>
</tr>
</tbody>
</table>

Table III shows the performance obtained during training by varying the size of the hidden layer.

Table III

<table>
<thead>
<tr>
<th>Case</th>
<th>Size of hidden layer (x input layer)</th>
<th>MSE Attained</th>
<th>Precision attained in %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.75 - 1.5</td>
<td>1.2 x 10^{-2}</td>
<td>87.1</td>
</tr>
<tr>
<td>2</td>
<td>1.0</td>
<td>0.6 x 10^{-3}</td>
<td>87.8</td>
</tr>
<tr>
<td>3</td>
<td>1.25</td>
<td>0.8 x 10^{-4}</td>
<td>87.1</td>
</tr>
<tr>
<td>4</td>
<td>1.3</td>
<td>0.3 x 10^{-4}</td>
<td>90.1</td>
</tr>
<tr>
<td>5</td>
<td>1.75</td>
<td>0.6 x 10^{-4}</td>
<td>89.2</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0.7 x 10^{-4}</td>
<td>89.8</td>
</tr>
</tbody>
</table>

The case where the size of the hidden layer taken to be 1.5 times to that of the input layer is found to be computationally efficient. Its MSE convergence rate and learning ability is found to be superior to the rest of the cases. Hence, the size of the hidden layer of the ANNs considered is 1.5 times to that of the input layer. The size of the input layer depends upon the length of the input vector and the output layer represents the number of parameters.

The selection of the activation functions of the input, hidden and output layers plays another important part in the performance of the system. A common practice can be to use a similar type of activation function in all layers. But certain combinations and alterations of activation function types carried out during training provide certain directions and show a way to attain better performance. Two types of MLP configurations are considered - the first type constituted by a set of similar activation functions in all layers of the ANNs and the other with a varied combination of activation functions in different layers. Both these two configurations are trained with GDMALBP as a measure of training performance standardization.

Table IV shows the results derived. The setup of ANNs should be modified often during training to ascertain the best configuration that can be selected to perform the subsequent tasks which in this case is prediction of the parameters with controlled precision. The results provided in Table IV show that the configuration formed by tan-sigmoid functions at the input and output layers and log-sigmoid at the hidden layer provides superior MSE convergence taking a time which is marginally higher (9.2 sec.s) than the log-sigmoid, tan-sigmoid - log-sigmoid combination (9.1 sec.s) case. Since MSE con-
vergence is more critical, it is taken to be the decisive factor and the combination with tansigmoid-logsigmoid-tansigmoid activation functions is taken for carrying out the entire training regime. Moreover, an ANN with a faster learning ability is not always effective as it restricts the capacity to generalize and provides more memorization. It is undesirable and does not serve the purpose of an appropriately configured predication mechanism. An ANN with moderate learning rate is superior as it has better generalization and finer discerning capability [1].

<table>
<thead>
<tr>
<th>Case</th>
<th>Input layer</th>
<th>Hidden layer</th>
<th>Output layer</th>
<th>MSE x 10^4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>log-sigmoid</td>
<td>log-sigmoid</td>
<td>log-sigmoid</td>
<td>13.76095</td>
</tr>
<tr>
<td>2</td>
<td>tansigmoid</td>
<td>tansigmoid</td>
<td>tansigmoid</td>
<td>35.38631</td>
</tr>
<tr>
<td>3</td>
<td>tansigmoid</td>
<td>log-sigmoid</td>
<td>tansigmoid</td>
<td>70.77592</td>
</tr>
<tr>
<td>4</td>
<td>log-sigmoid</td>
<td>tansigmoid</td>
<td>log-sigmoid</td>
<td>104.19008</td>
</tr>
<tr>
<td>5</td>
<td>log-sigmoid</td>
<td>log-sigmoid</td>
<td>tansigmoid</td>
<td>145.47293</td>
</tr>
<tr>
<td>6</td>
<td>log-sigmoid</td>
<td>tansigmoid</td>
<td>tansigmoid</td>
<td>196.58505</td>
</tr>
</tbody>
</table>

V. RESULTS AND DISCUSSION

After the training is over the testing of the system is carried out by following the considerations summarized in Table II. The objective is to extend the conditions so as to cover the maximum possible events that occur in practical transmit receive condition. The unpredictability associated with the transmit-receive condition is represented the variance incorporated in the testing data. A similar set of data is generated for validation as it has better generalization and finer discerning capability [1].

Table VII shows a reduced form of the sets of data used for estimation and optimization of the design. Table VIII shows a curtailed form in Table VIII. The success rate is obvious. The data provided in Table VIII give a glimpse of the best case result where the conditions are customized for worst case performance yet the output obtained are for best case situations. The worst case condition of training is determined to be the case when the ANN has been trained with noise-free data and the test input has 50% variance compared to the input during training. Loop antenna dimension variation is also considered for ascertaining the performance of the model. Table V gives the results derived from an ANN configured to perform under ideal conditions with variations of loop radius and wire thickness of the antenna.

A similar set of results are also derived for the worst case condition. Table VI provides the summarized results for a circular loop antenna tested under worst case condition with variations of dimension. The results prove the robustness of the training of the ANN and the configuration adopted. Table X provide another curtailed set of data representing the worst case results.

Figure 5 shows the MSE convergence rate for four parameters generated by an ANN trained upto 7000 sessions. These convergence curves for the four parameters represent the learning ability of the ANN applied for optimization of the antenna design parameters showing controlled precision of prediction. The results obtained validate the objective of the work. Table I shows the success rates of the ANN during training time in generating the precision of prediction of the design parameter. The precision levels increase with
Table IX
A TRUNCATED SET OF DATA GENERATED BY THE ANN FOR WORST CASE CONDITION

<table>
<thead>
<tr>
<th>Frequency (MHz)</th>
<th>Radiation Resistance (Ω)</th>
<th>Loss Resistance (mΩ)</th>
<th>Efficiency (%)</th>
<th>Inductance (μH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>0.0276</td>
<td>0.0348</td>
<td>0.0671</td>
<td>21.338</td>
</tr>
<tr>
<td>54</td>
<td>0.0933</td>
<td>0.056</td>
<td>0.0181</td>
<td>55.8067</td>
</tr>
<tr>
<td>108</td>
<td>0.0291</td>
<td>0.0336</td>
<td>0.0210</td>
<td>39.5141</td>
</tr>
<tr>
<td>159</td>
<td>0.0447</td>
<td>0.0732</td>
<td>0.1884</td>
<td>41.4311</td>
</tr>
<tr>
<td>222</td>
<td>0.1777</td>
<td>0.0809</td>
<td>0.5844</td>
<td>135.1055</td>
</tr>
<tr>
<td>300</td>
<td>0.2587</td>
<td>0.1075</td>
<td>0.6166</td>
<td>172.8901</td>
</tr>
</tbody>
</table>

Fig. 5. Convergence of MSE with epochs for the four parameters

The results thus obtained provide useful insights into loop antenna design. That way the proposed model provides a means of providing an optimized design format for obtaining the physical layout of a loop antenna. The parameters predicted for a given frequency is made available as the best combination known to the ANN in terms of physical dimensions. The usefulness of the model for antenna design is thus obvious.

VI. CONCLUSION
The work shows the effectiveness of utilization of an ANN for parameter estimation of an antenna design. Such estimation can be the basis of determining the practical requirements of an antenna design. The present work can be extended further to fix the physical dimensions of an antenna design as required by different communication requirements. Moreover, the controlled nature of prediction precision as generated by the ANN can be also used for designing electronically alterable models of antenna suitable for smart designs.

REFERENCES