Chapter 3

Design and Analysis of Microstrip Patch Antennas Using Artificial Neural Network

Vivek Singh Kushwah and Geetam Singh Tomar

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.69522

Abstract

The microstrip patch antenna can also be designed using an artificial neural network (ANN) modeling technique where size of the antenna is major limitation especially in mobile and wireless applications. In this chapter, analysis and synthesis problems for designing of microstrip patch antennas were discussed using the artificial neural network technique. An analysis problem refers to calculation of resonant frequency of microstrip patch antenna whereas a synthesis problem refers to calculation of dimensions of patch antenna. Both problems are reciprocal of each other. Results are implemented using graphical user interface (GUI) tools of MATLAB programming language. Back-propagation training algorithm of artificial neural network is used to train the network for minimization of error and computation time. Therefore, the geometric dimensions of patch are obtained with high accuracy in less computation time as compared to simulation software.

Keywords: microstrip antennas, artificial neural networks, ANN modeling, IE3D electromagnetic simulator, resonant frequency, return-loss, GUI

1. Introduction

Nowadays, many antennas are used for various wireless applications but microstrip antenna is the most preferable antenna for microwave communication. The microstrip antennas are low cost and very easily used fabrication techniques [1, 2]. Since a photoetching process is applied for the fabrication of microstrip patch antennas hence it is also known as "patch antennas." The shape of this patch may be square, hexagonal, rectangular, trapezoidal, or any other pattern. In this chapter, rectangular microstrip patch antennas are designed for synthesis and analysis by an artificial neural network (ANN) modeling procedure.

There has been a tremendous development in the field of patch antenna during the last two decade [3, 4]. It is the result of the overwhelming research and this technology has been quickly absorbed for consumption. A few previous research works done in this field are discussed in this section. The length of an antenna is inversely proportional to the resonant frequency of patch, that is, an antenna operating at higher frequency is smaller in size than compared to the antenna operating at lower frequency. So, to decrease the antenna dimensions required for operating the antenna at lower frequency, the electrical length of the patch should be enlarged at lower frequency. This technique has been successfully used for minimizing the antenna dimensions. A technique of improving the electrical length is by meandering the surface current path of the patch. One such research work was done by Dey and Mittra [5]. In the mentioned article, 70% size reduction of the proposed microstrip patch design is reported. In Ref. [6], authors used three meandering slots and a shorting pin on a rectangular patch to obtain the results. In Ref. [7], an innovative technique is proposed for calculating the resonant frequency of circular microstrip patch antennas using artificial neural for computer-aided design (CAD) applications. In Ref. [8], the back-propagation multilayered-perceptron network is used for calculating the resonant frequency of electrically thin and thick rectangular microstrip antennas. This technique is applicable for a wide range of substrate thicknesses and permittivity for the computer-aided design (CAD) applications of microstrip antennas. One such research work was conducted by Kuo et al. [9]. In this work, the researchers have chosen the ground plane for modification in designed antenna. A total number of three slots were loaded in the ground plane. The authors have shown that the variation seen by changing the length of the slots. For the best case, the size reduction of 56% was obtained by the authors. Some researchers also reported size reduction using inductive loading. One such research work was reported by Reed et al. [10]. In the design, the researchers have loaded a printed inductor to minimize the frequency of resonant antenna. Using this, a size reduction of 50% was achieved. A bow-tie patch antenna loaded with Sierpinski fractal was also proposed by Anguera et al. [11] which has good directivity along with compactness of 42%. One of the methods of increasing the electrical path length is by meandering the surface current path of the patch. In [13], a new method for developing computer-aided design (CAD) models for microstrip antenna is proposed using spectral domain (SD) formulation and artificial neural network (ANN) technique. In Ref. [14], design of rectangular microstrip antennas is proposed using the artificial neural network technique for the most efficient dielectric materials. In Ref. [15], the authors have suggested the combination of inductive loading and shorting pin. The authors have suggested that using the novel design, a size reduction of 75% may be achieved. A bow-tie patch antenna loaded with Sierpinski fractal was also presented by Garima et al. [16] along with compactness of 42% and good directivity. In Ref. [17], single-fed broadband square patch antenna was constructed for ultra high frequency (UHF) radio frequency identification (RFID) applications using meandered probe feeding techniques for good impedance matching. The purpose of the antenna was to optimize and modify antenna. In Ref. [18], two L-shaped patch antenna was constructed using a probe or coaxial feeding techniques. Two parallel L-shaped antennas were used to obtain better bandwidth. The bandwidth of the antenna was 6.41 GHz ranging from 3.51

to 9.65 GHz. Return-Loss of the proposed antenna is −28 dB at 3.51 GHz and −25 dB at 9.65 GHz. In [19], E-shaped antenna was constructed for wireless communications, and its applications using U-slot patch was introduced under patch antenna to increase and optimize bandwidth of the proposed antenna. The systematic study of antenna was enhanced gain, return loss, bandwidth, and radiation pattern. In Ref. [20], four mini fractal antennas were constructed to improve bandwidth and size of the antenna using coaxial feeding techniques. The proposed design was used for broadband applications. Four spring resonators form an ultra-wideband frequency. Low percentage error and high precision in less time are obtained by using a knowledge-based hybrid neural network (KBHNN) model for designing of microstrip antennas (proximity coupled), which is used for various WiMAX, WiFi, and WLAN applications. **Figure 1** illustrates a simple rectangular microstrip patch antenna of substrate height h, dielectric constants ε_r, ε_y, width W, and length L.

In this chapter, synthesis is defined as to obtain dimensions (W, L) of microstrip antenna while providing the resonant frequency (f_r), height of the dielectric substrate (h), and dielectric constants at the input of the ANN model (**Figure 2**). Dimensions of patch are computed using the designed equations of the microstrip patch antennas. For the analysis problem of patch antenna, resonant frequency (f_r) or both upper and lower cutoff frequencies are obtained at the output of ANN model while providing the dimensions of patch and other parameters as the inputs of ANN model, as shown in **Figure 3**. This model is very significant for antenna researchers to determine the dimensions and other parameters of microstrip antenna.

The range of dielectric constants should be taken between 2.2 and 12 but the dielectric substrate should be thicker and the dielectric constant should be less for obtaining high efficiency and

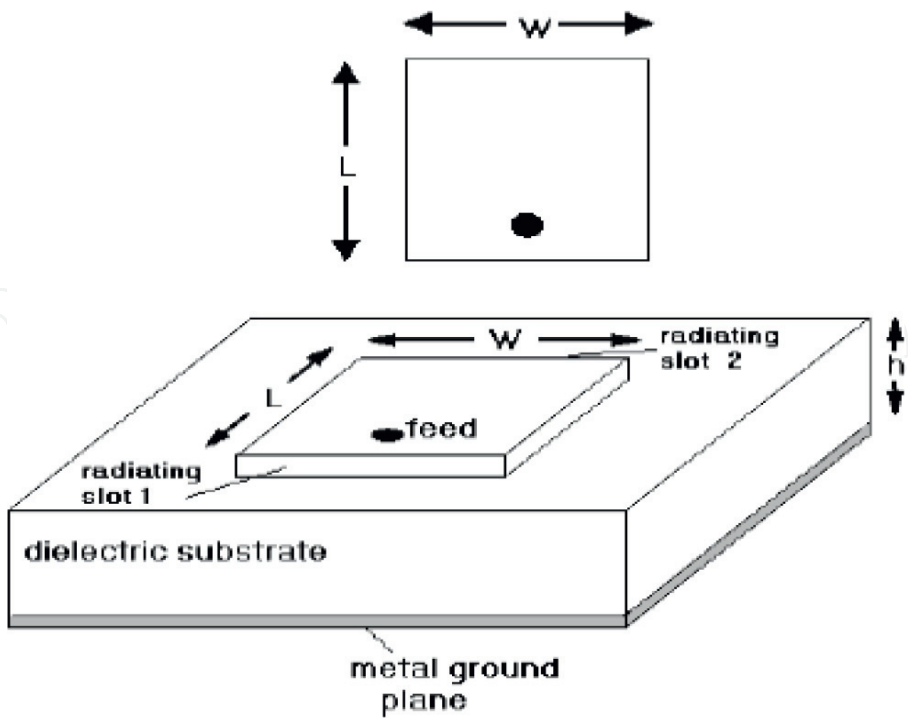

Figure 1. Basic layout of rectangular patch antenna.

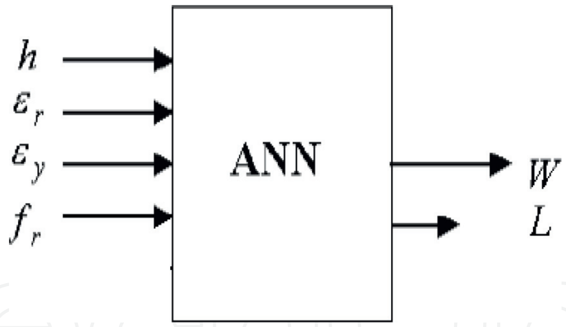

Figure 2. The synthesis of microstrip patch using ANN model.

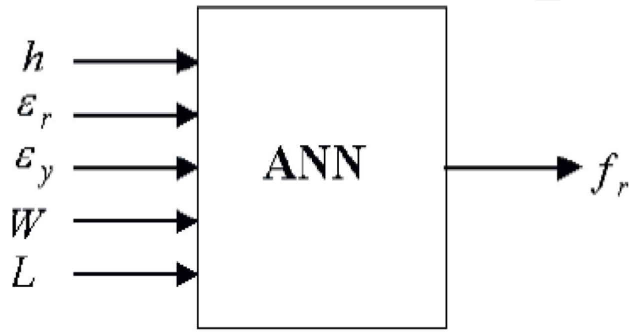

Figure 3. The analysis of microstrip patch using ANN technique.

wide bandwidth. The effective thickness (h_e) and effective dielectric constant ($\varepsilon_{\text{reff}}$) are calculated using Eqs. (1) and (2), where h is thickness and ε_g is the geometric mean of dielectric constant.

$$h_e = \sqrt{\frac{\varepsilon_r}{\varepsilon_g} h} \tag{1}$$

$$\varepsilon_{eff} = \frac{\varepsilon_g + 1}{2} + \frac{\varepsilon_g - 1}{2}\left[1 + 12\frac{h_e}{W}\right]^{-\frac{1}{2}} \tag{2}$$

If the velocity of light is denoted by C, the width of antenna is calculated using Eq. (3).

$$W = \frac{C}{2f_o}\sqrt{\frac{2}{\varepsilon_g + 1}} \tag{3}$$

Eqs. (4) and (5) represents the real length of the patch (L).

$$L = L_{eff} - 2\Delta L \tag{4}$$

$$L = \frac{1}{2f_r\sqrt{\varepsilon_{eff}}\sqrt{\mu_o\varepsilon_r}} - 2\Delta L \tag{5}$$

The fringing property of patch gives rise to the extended electric length, which is illustrated by Eq. (6)

$$\Delta L = 0.412h \frac{(\varepsilon_{reff} + 0.3)\left(\frac{w}{h} + 0.264\right)}{(\varepsilon_{reff} - 0.258)\left(\frac{w}{h} + 0.813\right)} \tag{6}$$

2. Synthesis and analysis problem for microstrip patch antenna

Synthesis is defined as to obtain the dimensions of microstrip patch while providing the resonant frequency, thickness, and dielectric constants of the dielectric material as the input parameters of ANN model (**Figure 2**). Dimensions of patch are computed using the designed equations of the microstrip patch antennas. For the analysis problem of patch antenna, resonant frequency (f_r) or both upper and lower cutoff frequencies are obtained at the output side of ANN model while providing the dimensions of patch (W, L) and other parameters (ε_r, ε_y, h) at the input side of ANN model (**Figure 3**). These ANN models are easily applicable and very useful for researchers to judge the accurate upper and lower cutoff frequency, bandwidth, resonant frequency, and accurate dimensions of patch antenna. Here ε_r and ε_y represent the electrical properties of the dielectric material.

2.1. ANN architecture for microstrip antenna

There are many algorithms of ANN which is used to train the neural network [12]. In this work, back-propagation training algorithm was applied to train the neural network and to build the ANN models for patch antenna. This algorithm is explained in Section 2.1.1.

2.1.1. Back-propagation algorithm

In this algorithm, connections are done from input layer to hidden layer and to output layer. The network is trained using the gradient descent method and the error calculated is sent back to the hidden layer as well as the output layer for weight adjustments [13]. In multilayer perceptron network there are two sets of weights, namely from input layer to hidden layer and from hidden to output layer. The error due to the second set of weights is calculated using the delta rule and the error is required to propagate from output layer to input layer so that the error can be assigned proportionately to the weights which caused it. This problem is known as credit-assignment problem and helps in deciding which weights should be altered and by how much. This problem can be solved by using the error back-propagation algorithm, which works in two passes:

1. **Forward pass**: it takes input vector, computes function, and evaluates derivative of error function with respect to weights.
2. **Backward pass**: it propagates the error derivatives backward and computes the weight adjustments.

This algorithm uses the supervised learning mechanism.

In multilayer feed-forward networks, normally, there is an input layer, one or two hidden layers and one output layer. The raw values are given to the input layer, computations are done at the hidden layer and their activations are given to the output layer where each pattern is given a specific classification category. In this algorithm, we are giving a solution to credit assignment problem using partial derivatives to find out how weight w_{ij} effects the error which is represented by Eq. (7).

$$\frac{\partial E(t)}{\partial W_{ij}(t)} \tag{7}$$

Here, partial derivatives are used because the direction of the errors can be determined, which may be positive or negative. If this derivative to be negative, this will add in weight and finally decreases the error and thus local minima can be reached (see step 6 of algorithm). This shows that if derivative of error is positive, the weight increases by adding a negative value to the weight and conversely, if negative. Procedure of taking derivative in this algorithm starts from weights of output layer to input layer, that is why, it is known as "back-propagation algorithm." **Figure 4** represents the three layers fully connected feed-forward network using the back propagation algorithm [14]. It consists of three layers: one input layer, one hidden layer, and one output layer. Here, $x_1, x_2, \ldots, x_n$ are the input neurons. x_b is the bias input, with associated weight w_{1b}. w_{1b} can increase or decrease in the net output of activation function depending on its positive or negative value.

The value of w_{1b} is kept constant throughout the process. It can be 0 or $\pm$ 1. $y_1, y_2, \ldots, y_j$ are the hidden neurons and $O_1, O_2, \ldots, O_m$ are the output neurons. v_{jn} represents the input hidden layer weights, that is, these are the weights associated with input layer neurons connected to hidden layer neurons. w_{mj} represents the output-hidden layer weights, that is, these are the weights associated with hidden layer neurons connected to output layer. Input, hidden, and output layers can be represented with the help of vectors as illustrated in Eqs. (8-10).

$$x = [x_1, x_2, \ldots, x_n]^t \leftarrow \text{input training vector} \tag{8}$$

$$y = \left[y_1, y_2, \ldots, y_j\right]^t \leftarrow \text{hidden layer vector} \tag{9}$$

$$O = [O_1, O_2, \ldots, O_m]^t \leftarrow \text{output vector} \tag{10}$$

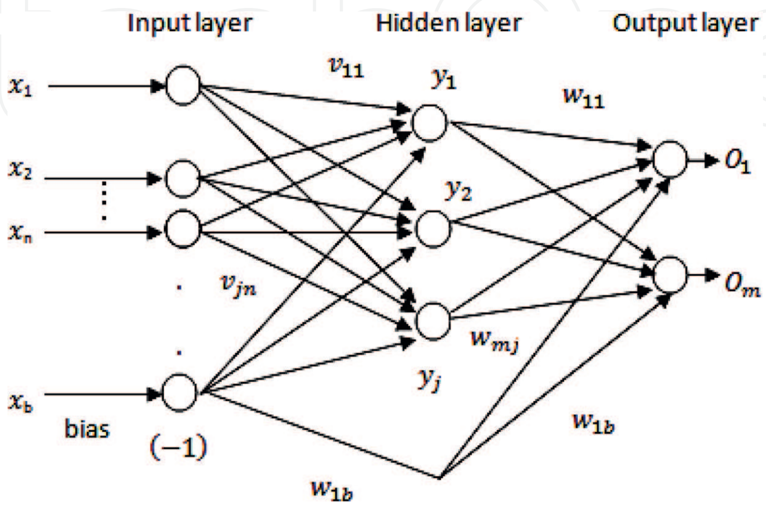

Figure 4. Three-layer fully connected feed-forward network.

where 't' indicates transposition of matrix.

Given are P training pairs

$\{x_1, a_1, x_2, a_2, \ldots, x_P, a_P\}$ and there is a bias $= -1$, η is the learning constant. Here a_1, a_2, and a_P are the desired output vector response.

Step 1: $\eta > 0$, Maximum error $E_{\max}$ is chosen.

Step 2: Weights are initialized at small random values, $E \leftarrow 0, p \leftarrow 1, k \leftarrow 1$

where E = present error; p = counter with in the training cycle; k = training step counter.

Step 3: Now begin the training process. Input is provided to input layer and output is Calculated as given in Eqs. (11) and (12).

$$x \leftarrow x_P, a \leftarrow a_P$$

$$y_j = f\left(v_j^t x\right) \text{ where } j = 1, 2, 3, 4, \ldots, J \tag{11}$$

$$O_k = f\left(w_j^t y\right) \text{ where } k = 1, 2, 3, 4, \ldots, K \tag{12}$$

Here f (.) is the activation function.

Step 4: Calculate the new value of error by using previous value of error, desired output a_k and actual output O_k using Eq. (13).

$$E' = \frac{1}{2}(a_k - O_k)^2 + E \text{ where } k = 1, 2, 3, 4, \ldots, K \tag{13}$$

Step 5: Now value of error for both the layer is computed.

Error signal for output layer is calculated using Eq. (14).

$$\delta_{ok} = \frac{1}{2}(a_k - O_k)(1 - {O_k}^2) \text{ for } k = 1, 2, 3, 4, \ldots, K \tag{14}$$

Eq. (15) represents the Value of error for middle (hidden) layer

$$\delta_{yj} = \frac{1}{2}\left(1 - y_j\right)^2 \sum_{k=1}^{k} \delta_{ok}\, w_{kj} \text{ for } j = 1, 2, 3, 4, \ldots, J \tag{15}$$

Step 6: Now weights of output layer are adjusted using Eq. (16):

$$w_{kj}^{'} = w_{kj} + \eta \delta_{ok} y_j \text{ where } k = 1, 2, 3, 4, \ldots, K, \text{ where } j = 1, 2, 3, 4, \ldots, J \tag{16}$$

Step 7: Now weights of Hidden layer are modified using Eq. (17).

$$v_{ji}^{'} = v_{ji} + \eta \delta_{yj} x_i \text{ for } i = 1, 2, 3, \ldots, I \tag{17}$$

Step 8: If p is less than P then $p = p + 1$ and $k = k + 1$ and go to step 3, otherwise return back to step 9.

Step 9: At last, the sequence of full training process is finished.

If the value of present error (E) < maximum error ($E_{\max}$), then stop the training process and now check out weights of output layer w and weights of hidden layer v.

If the present value of error (E) > maximum error ($E_{\max}$), then $E = 0$, $p = 1$ and start a new training sequence by return back to step 3.

In first step of training, η and $E_{\max}$ are chosen. Step 2 is the initialization step where values of p, k, E, and weights are initialized. Step 3 is feed-forward step where both hidden layer and output layer outputs are calculated. In step 4, errors are computed. In step 5, back-propagation training takes place and error signal is computed for both hidden layer and output layer. In step 7, output layer and hidden layer weights are adjusted, respectively. In step 8, it is checked if $p < P$, then increment p and k and go to step 3, otherwise go to step 9. In step 9, if $E < E_{\max}$, the training is complete and if $E > E_{\max}$, a new training cycle is started.

2.1.2. Learning factors of back-propagation network

The training of back-propagation networks is dependent on various parameters such as initial weights, number of layers, number of neurons per layer, and updation rule.

1. **Initial weight**: the choice of initial weights determines how fast the networks will converge. Typical value for choosing initial weights is between −1.00 and 1.00 or −0.5 and 0.5. The final solution may be affected by the initial weights of multilayer feed-forward networks. One method of choosing the weight w_{ij} choosing is in the range $\left[\frac{-3}{\sqrt{O_i}}, \frac{3}{\sqrt{O_i}}\right]$, where O_i is the number of processing elements.

   If all the weights are given equal values, the network may not be trained properly. If large values are given to weights at the initial stage, then the system may be stuck at local minima very near to starting point itself. So, it is necessary to initialize with small weights uniformly distributed in a small range

2. **Learning rate (η)**: the learning rate affects the convergence of back-propagation. A large value of η simply increases the speed of convergence but this large value introduces instability into the learning rule causing oscillations in the learned weights and on the other hand, if η has small value, the speed of training will be less. For successful operation, the range of η lies between10^{-3} and 10.

3. **Number of training data**: the numbers of input nodes are determined by the dimension, or size or the input vector to be classified or generalized with a certain output quality. The training data should be sufficient and proper for training.

4. **Momentum:** the gradient decent is very slow using small η and oscillates using large η. The problem of the oscillations can be sorted out by adding a momentum factor to the weight adjustments as illustrated in Eq. (18)

http://dx.doi.org/10.5772/intechopen.69522

$$\Delta w(t) = -\eta \nabla E(t) + \alpha \Delta w(t-1) \tag{18}$$

where α is the selected momentum constant having values between 0.1 and 0.8. $\alpha \Delta w (t-1)$ is called the momentum term. t and $(t{-}1)$ indicates the current and most recent training step, respectively.

5. **Number of hidden layers nodes**: in the case of all multilayer feed-forward networks, the size of hidden layer is determined experimentally. For a network of a reasonable size, the size of a hidden layer has to be only a small fraction of the input layer. For example, if the network does not converge to a solution, it may need more hidden layers. On the other hand, the user may use a small number of hidden layers when the network converges.

6. **Stopping criteria**: stopping criteria are used to decide when the network problem has been solved. It is possible to stop when:

   i. The mean squared error is sufficiently small.

   ii. The rate of change of the mean squared error is sufficiently small.

   iii. Combination of above two criteria.

2.1.3. Realization of patch antenna with numerical results

IE3D electromagnetic simulation software is used for designing of the rectangular patch antenna using the following design parameters.

Resonant frequency = 4.99 GHz, dielectric constant = 4.4, height of dielectric material = 1.6 mm

Now, the dimensions of patch can be calculated using Eqs. (1)–(6) for 4.99 GHz resonant frequency and given by

$L = 13.34$ mm, $W = 17.7$ mm

The basic design of microstrip patch antenna is shown in **Figure 5**. Here, 50 ohm port (Z_o) is connected to the patch by using the microstrip feed line technique. Feed is the point where patch radiates maximum. Eq. (19) is used to calculate the width W_o of microstrip line, where ε_{reff} is the effective dielectric constant.

$$Z_0 = 120\Pi / [\sqrt{}(\varepsilon_{reff})\{W_0/h + 1.393 + .667 \ln(W_0/h + 1.444)\}] \tag{19}$$

which gives $W_0 = 2.94$ mm.

When IE3D electromagnetic simulation is performed, then we obtain the different resultant graph between return loss and frequency. While changing the dimension of patch, the different resultant graphs are obtained between return loss and frequency using simulation software, as shown in **Figures 6–8**. Here all the dimensions are taken in mm.

(1) When $W = 17.7$, $L = 13.34$, $\varepsilon_r = 4.4$, $h = 1.6$

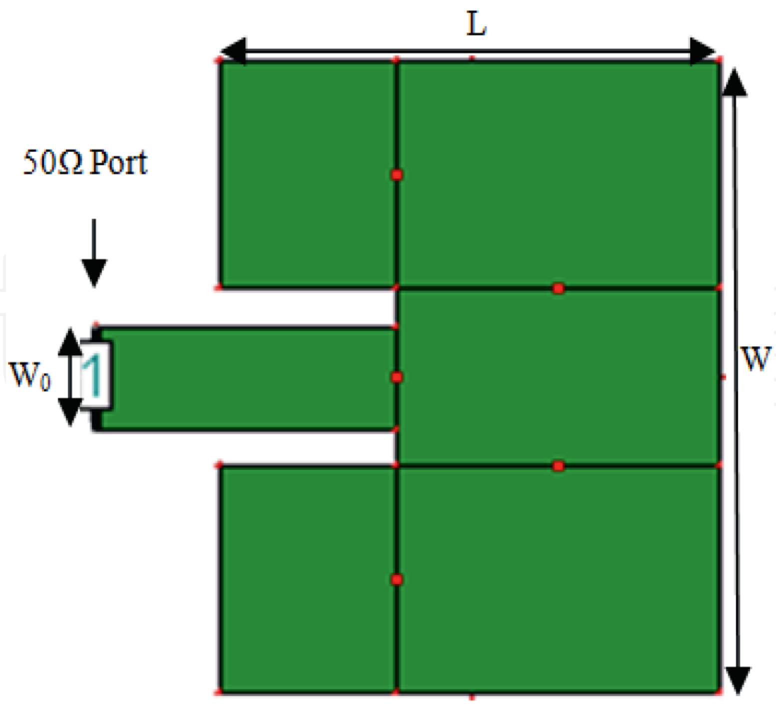

Figure 5. Design of microstrip patch antenna using simulation software.

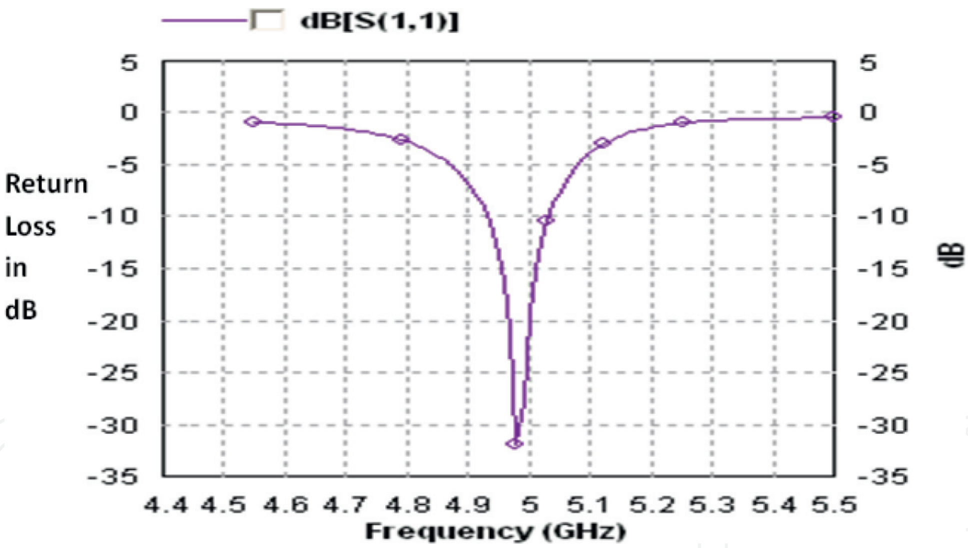

Figure 6. Simulated response of patch at 4.99 GHz resonant frequency.

Here, 4.94 GHz is the lower cutoff frequency (f_1) and 5.04 GHz is the upper cutoff frequency (f_2). Furthermore, only one dimension of the patch will be varied while maintaining all other parameters constant and vice versa. Frequency response between return loss and frequency are obtained, as shown in **Figure 7**, which gives the upper and lower cutoff frequencies (f_2, f_1) when the width and length of patch is 17.7 and 13.55 mm, respectively.

(2) When $W = 17.7$ mm, $\varepsilon_r = 4.4$, $L = 13.55$ mm, $h = 1.6$ mm

(3) When $W = 17.7$ mm, $L = 14.15$ mm, $h = 1.6$, $\varepsilon_r = 4.4$ mm

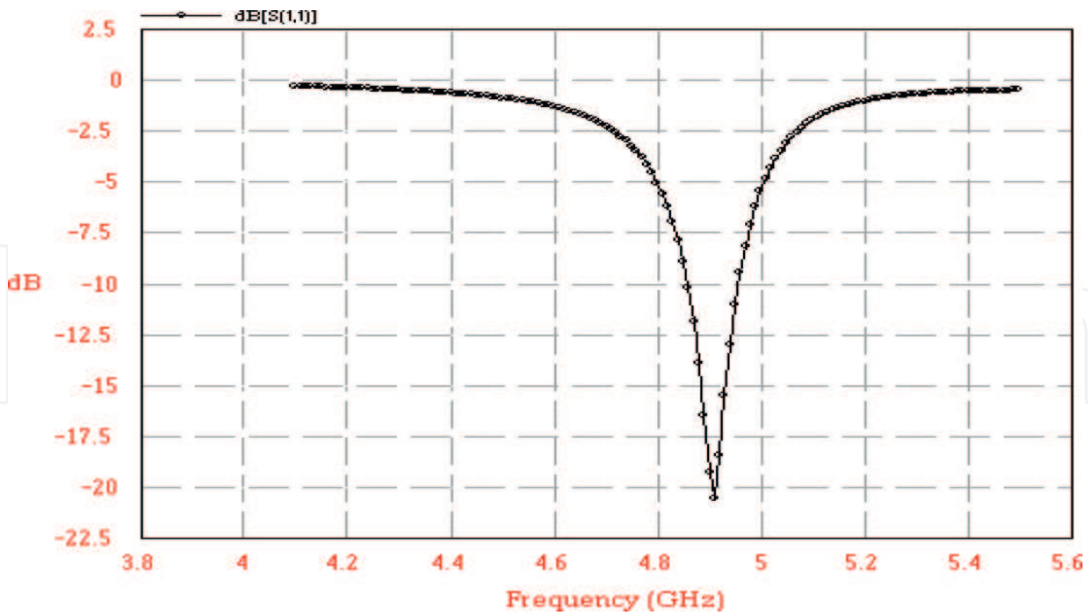

Figure 7. Simulated response of patch at 4.89 GHz resonant frequency.

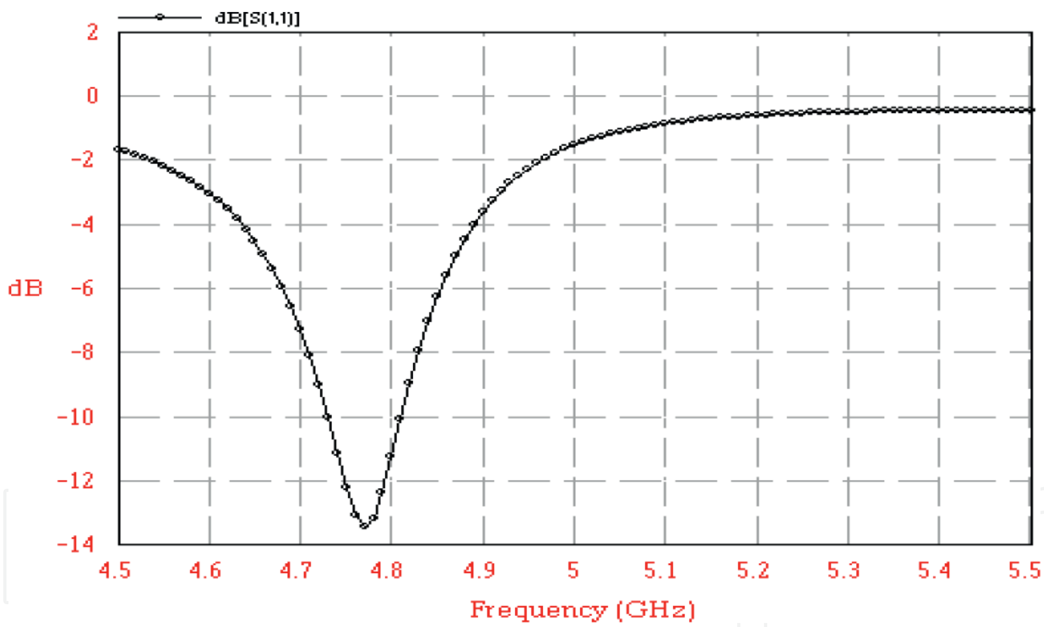

Figure 8. Simulated response of patch at 4.77 GHz resonant frequency.

Figures 6–8 represent the variation of frequency range of microstrip patch antenna with respect to variation in dimensions. Around 25–30 samples are collected using the EM simulation tool for ANN training for different dimensions, which are illustrated in **Table 1**. It is used for the analysis of patch antenna.

Now, the neural network is constructed for rectangular microstrip patch antenna. For the analysis problem of neural network, the dimensions (*W*, *L*) are considered as the input parameters of the ANN model whereas output parameters are upper and lower cutoff frequencies,

Inputs (dimensions of patch in mm)		Targets (lower and upper cutoff frequency in GHz)	
W	*L*	f_1	f_2
17.7	13.34	4.94	5.04
17.7	13.55	4.86	4.95
17.7	13.65	4.82	4.91
17.7	13.85	4.8	4.89
17.7	14.05	4.77	4.85
17.7	14.15	4.73	4.81
17.7	14.25	4.71	4.79
17.7	14.35	4.69	4.76
17.7	14.45	4.66	4.73
18.3	13.85	4.78	4.87
18.3	14.35	4.65	4.73
18.8	14.35	4.61	4.7
18.8	14.85	4.49	4.55
19.3	14.85	4.47	4.55
19.3	15.35	4.37	4.41
19.8	15.35	4.35	4.41
20.3	15.85	4.31	4.37
20.3	16.35	4.29	4.35
20.8	16.35	4.27	4.33
20.8	16.85	4.26	4.33
21.3	16.85	4.21	4.27
21.3	17.35	4.16	4.21
21.8	17.35	4.14	4.19
21.8	17.85	4.02	4.04
22.3	17.85	3.99	4.01
22.3	18.35	3.92	3.93

Table 1. Simulated results for ANN analysis of patch.

which act as targets. The artificial neural network is trained by using the ANN feed-forward back-propagation training algorithm and transfer function so that error can be minimized with less computation time. Matlab neural network graphical user interface (GUI) toolbox is used as a plateform for ANN training. The "***nntool***" command is used for ANN training in the GUI, which will display the network or data manager window, as given in **Figure 9**.

After displaying this window, input and outputs are provided in the matrix form by clicking on "**new data**.". Now, a neural network is created by clicking on "**New network**." Here, the feed-forward back-propagation algorithm and trainlm (Levenberg-Marquardt back-propagation

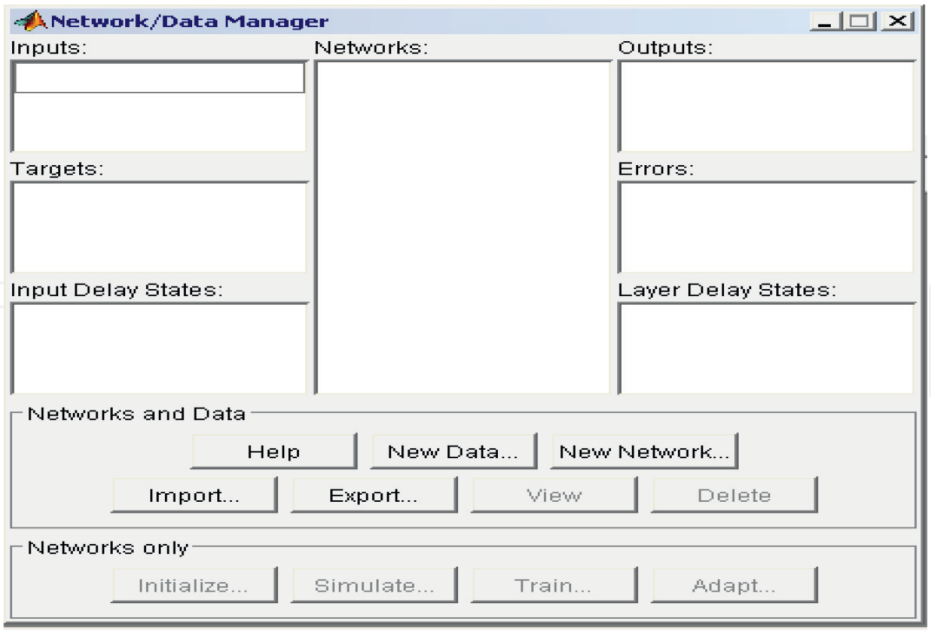

Figure 9. GUI of ANN.

training function) are applying where supported transfer functions are **purelin** (linear transfer function) and **tansig** (hyperbolic tangent sigmoid transfer function).

2.1.4. Analysis of microstrip patch antenna using artificial neural network

The network architecture is one of the basic building blocks for ANN to be deployed. It determines how the number of layers and number of neurons are decided in the different layers and how these layers are interconnected with each other through corresponding weights. The numbers of inputs and output layers are always constant, that is, a single layer is required at the input and output sides but it is very tough task to determine the number of hidden layers and the number of neurons in it. The rule-of-thumb method is used for deciding the number of neurons in the hidden nodes. According to this rule, the total numbers of hidden layer neurons are summation of the size of output layer plus 2/3 of the input layer size. The number of hidden layer neurons should be less than twice of the number of neurons in input layer. The size of the hidden layer neurons is between the input layer size and the output layer size. But the above three methods for determining the number of hidden nodes are not always true because not only the input layer and the output layer determines the size of the hidden layer neurons but also the dataset of training samples, types of architecture, the training algorithm, and complexity of the activation function applied on the neurons decides it. Multiple hidden layers are used in the applications where accuracy is the criteria and training time is not important. Usually, one or two hidden layers are adequate for resolving any nonlinear complex task. The third hidden layer can be added for improving accuracy, but this will increase overall complexity of the neural network and the total training time will be increased. Four hidden layers are not used in the neural network architecture because it cannot follow the rule of thumb. If there are three hidden layers then total neurons in hidden layers $= 2\ (3 \times 2/3)$ (twice of input). Minimum hidden layer must be used in ANN to reduce the complexity and training time. As shown in **Figure 10**, neural network architecture consists of

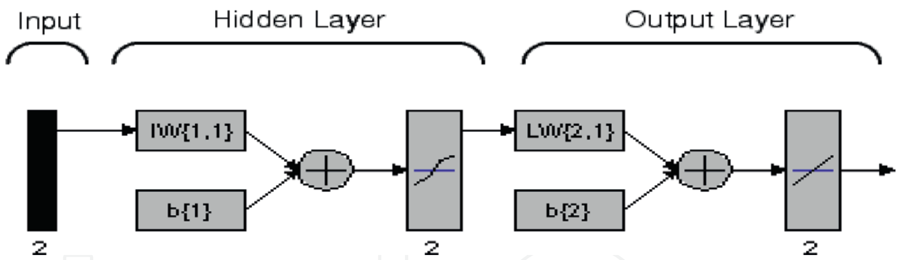

Figure 10. Neural network architecture for microstrip patch antenna.

three layers. It has one input, one hidden, and one output layer. Input, hidden, and output layer consists of two neurons. Inputs *W* and *L* are applied at the input neurons while outputs f_1 and f_2 are obtained from the output neurons. The ANN training graph results for the analysis of microstrip patch antenna are shown in **Figure 11**, which shows that 21 epochs are required for ANN training. The full sets of input samples are passed through the artificial neural network for minimization of the error in the back-propagation algorithms, which is known as an *epoch*. Training graph reveals that error reduced from 10^1 to nearly 10^{-4}, as shown in **Figure 11**.

Trained results (f_1 and f_2) for ANN analysis of patch are given in **Table 2**.

2.1.5. Synthesis of microstrip patch antenna using artificial neural network

Synthesis is defined as to obtain the dimensions of microstrip patch (*W*, *L*) as targets while providing the cutoff frequencies (f_1, f_2), thickness and dielectric constants of the dielectric material as the input parameters of ANN model. Around 25–30 samples are collected for ANN training which is obtained by varying the dimensions of microstrip patch antenna using electromagnetic simulation software as illustrated in **Table 3**. It is used for the synthesis of patch antenna. Trained Results (*W*, *L*) for ANN Synthesis of patch is illustrated in **Table 4**. The neural network is trained by using back-propagation algorithm. The ANN training graph

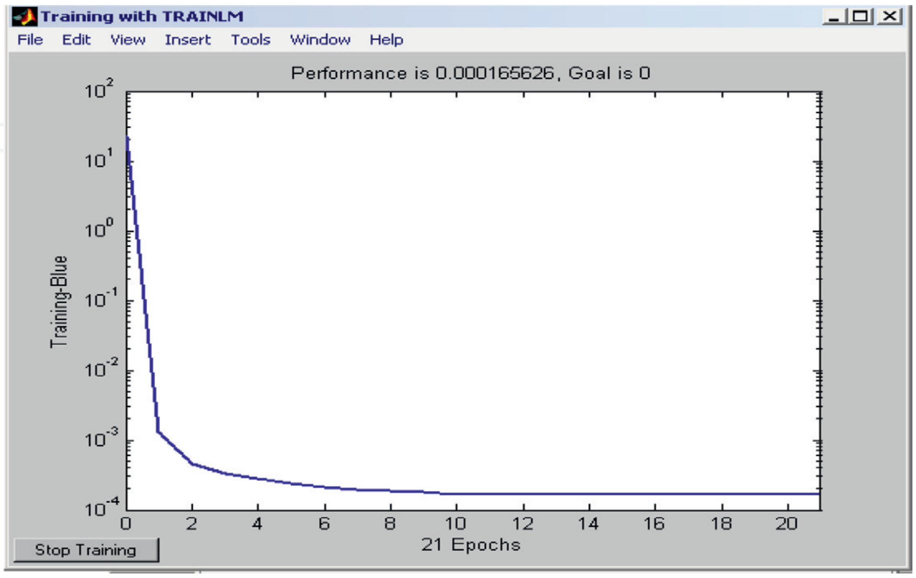

Figure 11. ANN training results for analysis of microstrip patch antenna.

Inputs (dimensions of patch in mm)		Targets (lower and upper cutoff frequency in GHz)	
W	*L*	f_1	f_2
17.7	13.34	4.9308	5.038
17.7	13.55	4.8705	4.9605
17.7	13.65	4.8512	4.9402
17.7	13.85	4.808	4.8948
17.7	14.05	4.7594	4.8438
17.7	14.15	4.7335	4.8165
17.7	14.25	4.7067	4.7884
17.7	14.35	4.6793	4.7596
17.7	14.45	4.6517	4.7305
18.3	13.85	4.7806	4.866
18.3	14.35	4.6474	4.726
18.8	14.35	4.6154	4.6924
18.8	14.85	4.4872	4.5577
19.3	14.85	4.4617	4.5309
19.3	15.35	4.3742	4.439
19.8	15.35	4.3592	4.4232
20.3	15.85	4.3042	4.3653
20.3	16.35	4.279	4.3385
20.8	16.35	4.2735	4.3324
20.8	16.85	4.2466	4.3027
21.3	16.85	4.2351	4.2897
21.3	17.35	4.1668	4.2123
21.8	17.35	4.1367	4.1781
21.8	17.85	4.0197	4.045
22.3	17.85	3.9887	4.0098
22.3	18.35	3.919	3.930

Table 2. Trained results for ANN analysis of patch.

Inputs (lower and upper cutoff frequency in GHz)		Targets (dimensions of patch in mm)	
f_1	f_2	*W*	*L*
4.94	5.04	17.7	13.34
4.86	4.95	17.7	13.55
4.82	4.91	17.7	13.65
4.8	4.89	17.7	13.85
4.77	4.85	17.7	14.05

Inputs (lower and upper cutoff frequency in GHz)		**Targets (dimensions of patch in mm)**	
f_1	f_2	W	L
4.73	4.81	17.7	14.15
4.71	4.79	17.7	14.25
4.69	4.76	17.7	14.35
4.66	4.73	17.7	14.45
4.78	4.87	18.3	13.85
4.65	4.73	18.3	14.35
4.61	4.7	18.8	14.35
4.49	4.55	18.8	14.85
4.47	4.55	19.3	14.85
4.37	4.41	19.3	15.35
4.35	4.41	19.8	15.35
4.31	4.37	20.3	15.85
4.29	4.35	20.3	16.35
4.27	4.33	20.8	16.35
4.26	4.33	20.8	16.85
4.21	4.27	21.3	16.85
4.16	4.21	21.3	17.35
4.14	4.19	21.8	17.35
4.02	4.04	21.8	17.85
3.99	4.01	22.3	17.85
3.92	3.93	22.3	18.35

Table 3. Simulated results for ANN synthesis of patch.

Inputs (lower and upper cutoff frequency in GHz)		**Targets (dimensions of patch in mm)**	
f_1	f_2	W	L
4.94	5.04	17.6823	13.2641
4.86	4.95	17.6905	13.751
4.82	4.91	17.7813	13.8311
4.8	4.89	17.8317	13.8756
4.77	4.85	17.8594	13.9008
4.73	4.81	17.9753	14.0034
4.71	4.79	18.0396	14.0602
4.69	4.76	18.0393	14.0613
4.66	4.73	18.1438	14.154

http://dx.doi.org/10.5772/intechopen.69522

Inputs (lower and upper cutoff frequency in GHz)		Targets (dimensions of patch in mm)	
f_1	f_2	W	L
4.78	4.87	17.8858	13.9234
4.65	4.73	18.2604	14.2558
4.61	4.7	18.5302	14.4931
4.49	4.55	18.8415	14.7789
4.47	4.55	19.2449	15.1307
4.37	4.41	19.3158	15.2211
4.35	4.41	19.8384	15.6756
4.31	4.37	20.1946	15.9979
4.29	4.35	20.3849	16.1706
4.27	4.33	20.5834	16.3509
4.26	4.33	20.9421	16.6629
4.21	4.27	21.2251	16.9371
4.16	4.21	21.4873	17.2026
4.14	4.19	21.7157	17.4157
4.02	4.04	21.885	17.7323
3.99	4.01	22.1746	18.0291
3.92	3.93	22.3253	18.3111

Table 4. Trained results for ANN synthesis of patch.

results for the synthesis of microstrip patch antenna is shown in **Figure 12** which shows that 100 epochs are required for ANN training and error get reduced from 10^2 to nearly 10^{-2}.

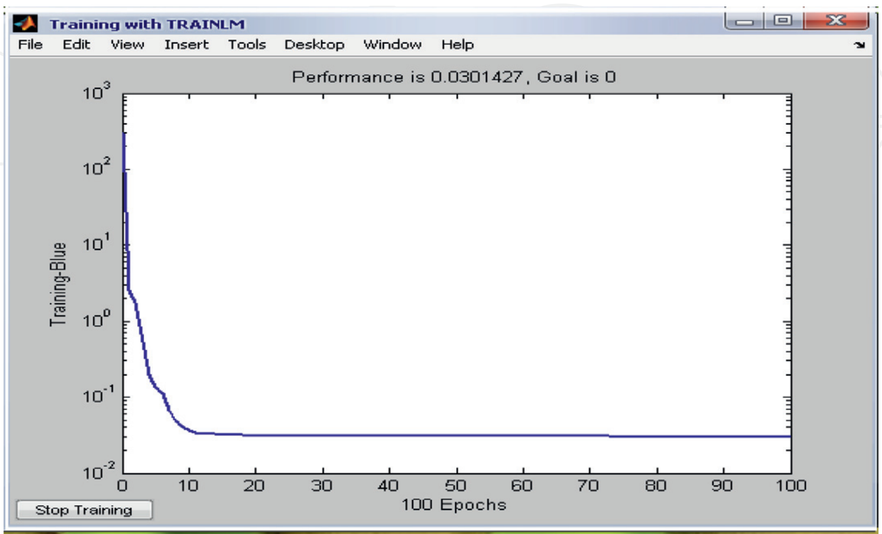

Figure 12. ANN training results for synthesis of microstrip patch antenna.

Network errors and weights are obtained in matrix form during ANN training as given below.

Network errors:

[0.20766 0.10947 0.018694 −0.031744 −0.05942 −0.17535 −0.23957 −0.2393 −0.34378 0.41415 0.039568 0.26975 −0.041452 0.055095 −0.015806 −0.038441 0.1054 −0.084949 0.21663 −0.14213 0.074867 −0.18731 0.084323 −0.085043 0.12542 −0.025298;

−0.31411 −0.20102 −0.1811 −0.025609 0.14918 0.14663 0.1898 0.28873 0.296 −0.073367 0.094171 −0.14313 0.071068 −0.28069 0.12893 −0.32561 −0.14788 0.17943 −0.00094985 0.18709 −0.087086 0.14741 −0.065711 0.11766 −0.17911 0.038876]

Weights

[1.0349 3.4722;

3.0377 2.2758]

3. Conclusion

In this work, the rectangular microstrip patch antenna is designed using the artificial neural network modeling procedure. Here synthesis refers to forward side and analysis refers to reverse side of the problem. Therefore in synthesis problem, the geometric dimensions such as length and the width of antenna are obtained with more accuracy in less time as compared to simulation software while providing resonant frequency, thickness, and dielectric constant at the input side of the ANN model. In the analysis problem of patch antenna, resonant frequency or both upper and lower cutoff frequencies of patch antenna are obtained at the output side of the ANN model while providing the dimensions of patch (W, L) and other parameters at the input side of the ANN model with much accuracy in less time.

Now the future work in this work includes trying more topologies to obtain more compact patch antennas, filters, and many other microwave/RF modeling design using artificial neural network for different band of applications such as ultra-wideband (UWB), Global System for Mobile communications (GSM), and WiMAX applications.

Appendix

MATLAB program for finding out the dimensions of microstrip patch antenna using the ANN

NNTWARN OFF

disp_freq = 100;

max_epochs = 8000;

%1r = 0.013;

err_goal = 0.009;

p = [4.94 4.86 4.82 4.8 4.77 4.73 4.71 4.69 4.66 4.78 4.65 4.61 4.49 4.47 4.37 4.35 4.31 4.29 4.27 4.26 4.21 4.16 4.14 4.02 3.99 3.92;

5.04 4.95 4.91 4.89 4.85 4.81 4.79 4.76 4.73 4.87 4.73 4.7 4.55 4.55 4.41 4.41 4.37 4.35 4.33 4.33 4.27 4.21 4.19 4.04 4.01 3.93];

tx = [17.7 17.7 17.7 17.7 17.7 17.7 17.7 17.7 17.7 18.3 18.3 18.8 18.8 19.3 19.3 19.8 20.3 20.3 20.8 20.8 21.3 21.3 21.8 21.8 22.3 22.3;

13.35 13.55 13.65 13.85 14.05 14.15 14.25 14.35 14.45 13.85 14.35 14.35 14.85 14.85 15.35 15.35 15.85 16.35 16.35 16.85 16.85 17.35 17.35 17.85 17.85 18.35];

%bus6

[w1,b1,w2,b2] = initff(p,2,'tansig',tx,'purelin');

Tp = [disp_freq,max_epochs,err_goal];

[w1,b1,w2,b2,epochs,tr] = trainbpx(w1,b1,'tansig',w2,b2,'purelin',p,tx,tp);

gil = simuff(p,w1,b1,'tansig',w2,b2,'purelin')

save patch1.mat;

Find the output:

P = [4.94 ; 5.04];

Result = simuff(p,w1,b1,'tansig',w2,b2,'purelin')

outputs:

[17.683 ; 13.265]

Author details

Vivek Singh Kushwah[1]* and Geetam Singh Tomar[2]

*Address all correspondence to: vivek_kushwah@rediffmail.com

1 Amity University, Madhya Pradesh, India

2 Govt. THDC Engg. College, Tihri, Uttarakhand, India

References

[1] Balanis CA. Antenna Theory. United States: John Wiley & Sons, Inc; 1997

[2] Bahl IJ, Bhartia P. Microstrip Antennas. Dedham, MA: Artech House; 1980

[3] Zhang QJ, Gupta KC. Neural Networks for RF and Microwave Design. USA: Artech House Publishers; 2000

[4] Wood C. Improved bandwidth of microstrip antennas using parasitic elements. IEEE Proceedings on Microwaves, Optics and Antennas. 1980;**127**(4):231–234

[5] Dey S, Mittra R. Compact microstrip patch antenna. Microwave and Optical Technology Letters. 1996;**13**(1):12–14

[6] Wong KL, Tang CL, Chen HT. A compact meandered circular microstrip antenna with a shorting pin. Microwave and Optical Technology Letters. 1997;**15**(3):147–149

[7] Sağıroğlu S, Güney K, Erler M. Resonant frequency calculation for circular microstrip antennas using artificial neural networks. International Journal of RF and Microwave Computer-Aided Engineering. 1998;**8**(3):270–277

[8] Karaboğa D, Güney K, Sağıroğlu S, Erler M. Neural computation of resonant frequency of electrically thin and thick rectangular microstrip antennas. Microwaves, Antennas and Propagation, IEEE Proceedings. 1999;**146**(2):155–159

[9] Kuo YL, Chiou TW, Wong LK A novel dual-band printed inverted-F antenna. Microwave and Optical Technology Letters. 2001;**31**(5):353–355

[10] Reed S, Desclos L, Terret C, Toutain S. Patch antenna size reduction by means of inductive slots. Microwave and Optical Technology Letters. 2001;**29**(2):79–81

[11] Anguera J, Puente C, Borja C, Montero R, Soler J. Small and high-directivity bow-tie patch antenna based on the Sierpinski fractal. Microwave and Optical Technology Letters. 2001;**31**(3):239–241

[12] Güney K, Sağıroğlu S, Erler M. Generalized neural method to determine resonant frequencies of various microstrip antennas. International Journal of RF and Microwave Computer-Aided Engineering. 2002;**12**(1):131–139

[13] Mishra RK, Patnaik A. Designing rectangular patch antenna using the neurospectral method. Antennas and Propagation, IEEE Transactions. 2003;**51**(8):1914–1921

[14] Tqürker N, Günes F, Yildirim T. Artificial neural design of microstrip antennas. Turkish Journal of Electrical Engineering & Computer Sciences. 2006;**14**(3):445–453

[15] Ansari JA, Yadav NP, Mishra A, Kamakshi, Singh A. Broadband rectangular microstrip antenna loaded with a pair of u-shaped slot. International Journal of Microwave and Optical Technology. 2011;**6**(4):185–190

[16] Garima, Bhatnagar D, Saini JS, Saxena VK, Joshi LM. Design of broadband circular patch microstrip antenna with diamond shapeslot. Indian Journal of Radio & Space Physics. 2011;**40**:275–281

[17] Wang Z, Fang S, Fu S, Jia S. Single-fed broadband circularly polarized stacked patch antenna with horizontally meandered strip for universal UHF RFID applications. IEEE Transactions on Microwave Theory and Techniques. 2011;**59**(4):1066–1073

[18] Yadav D. L-slotted rectangular microstrip patch antenna. In: Proceedings of the IEEE International Conference on Communication Systems and Network Technologies; 3–5 June 2011; Jammu, India. US: IEEE; 2011. pp. 220–223

[19] Ali MT, Aizat, Pasya I, Mazlan Zaharuddin MH, Yaacob N. E-shape microstrip patch antenna for wideband applications. In: Proceedings of the IEEE International RF and Microwave Conference; 12–14 December 2011; Negeri Sembilan, Malaysia. US: IEEE; 2011. pp. 439–443

[20] Varma R, Ghosh J. Design of slot loaded proximity coupled microstrip antennas using knowledge based neural networks. In: Proceedings of the IEEE International Conference on Recent Advances in Information Technology; 3–5 March 2016; Dhanbad, India. US: IEEE; 2016. pp. 209–214