Deep Learning Based Antenna Design and Beam-Steering Capabilities for Millimeter-Wave Applications

AHMED M. MONTASER AND KORANY R. MAHMOUD (Senior Member, IEEE)

1Electrical Engineering Department, Faculty of Technology and Education, Sohag University, Sohag 82524, Egypt
2Department of Electronics and Communications, Faculty of Engineering, Helwan University, Cairo 11795, Egypt
3National Telecommunications Regulatory Authority, Ministry of Communication and Information Technology, Giza 71515, Egypt

Corresponding author: Korany R. Mahmoud (kurany_hameda@h-eng.helwan.edu.eg)

ABSTRACT In this study, a deep neural network (DNN) is implemented to soft computation of the dual-band circularly polarized bone-shaped patch antenna (BSPA) at 28 GHz and 38 GHz for 5G applications. Via a simulated database of 150 BSPAs, a DNN model is constructed on a 5-layer system using an adaptive learning rate algorithm. The framework and hyper-parameters of the DNN model are optimized in the training phase of a hybrid algorithm combining strengths of both particle swarm optimization (PSO) and a modified version of the gravitational search algorithm (MGSA-PSO). To generate the database for training and testing the model, 150 BSPAs with different geometrical are simulated in terms of the resonant frequency using a precise electromagnetic analysis platform. A fabricated BSPA operating at 28 GHz and 38 GHz is used to test and verify the DNN model. Then, the application of DNN with back-propagation algorithm and weighted MGSA-PSO algorithm is used for beam-steering the main beam pattern of the designed uniform circular antenna array with side-lobe level \( \leq -30 \) dB by estimating the appropriate feeding phases of the 16 elements. Several illustrative examples are placed to beam-steer the pattern in the desired direction to check the validity of the technique.

INDEX TERMS Deep learning, artificial neural network, optimization algorithm, mm-wave antennas, circular polarization, beamforming technique.

I. INTRODUCTION

Recently, the demand for increased capacity in mobile and personal communications systems in addition to other modern applications such as satellite and, multi-input multi-output (MIMO) networks, biomedical imaging, remote sensing, radio astronomy, and radar, have motivated researchers towards the development of algorithms and standards that exploit space selectivity [1]. In this regard, one pertinent problem is finding antenna rotation for desired beam direction. Many techniques have been used to steer an antenna’s radiation pattern over the years [2]. The mechanical phased arrays rotated with motors started in military applications several decades ago. But nowadays it became undesirable, especially when factors such as weight, antenna size, and weather conditions have been considered. In addition to its limited use in static or very slow-changing environments due to the limitation in steering speed [3]. Rotating mechanisms are also prone to mechanical failure due to fatigue and the wearing of moving parts. The solutions for these problems led to electronic ways of steering beams. As a result, there are many efforts on the design of phased antenna array systems that play an important role in shaping and scanning the radiation pattern and constraining the adaptive algorithm used by the digital signal processor. These methods of beam steering based on controlling the phase values, the excitation amplitudes only, and both amplitudes and phases have been extensively considered in the literature [4]–[8]. The most important method is based on controlling the complex weights since the technique utilizes fully the degrees of freedom for the solution space. On the other hand, it is also considered the most expensive approach taking into consideration the cost of both phase shifters and variable attenuators for all elements. Therefore, beam pattern scanning based on controlling phase values was the only valid method in this work.

In literature, many papers studied the synthesis of antenna array using different optimization techniques, such as genetic algorithm [9], particle swarm optimization algorithm [10],
central force optimization [11], gravitational search algorithm [12], in addition to hybrid techniques those have been successfully used [5]–[8], [13]. However, the computational time to find the optimum weights will increase by considering more antenna array elements. Therefore, the deep neural network (DNN) is an essential computational tool with an unprecedented computational efficiency for these time-critical applications.

Recently there is a great effort by researchers to find easier and faster analysis techniques such as developed neural networks and optimization algorithms inspired by nature. Therefore, any computational system can find a relationship between inputs and outputs of an engineering system through the association of multiple layers of nodes and each node has its connection weight. This system is called the neural network [14]. The neural network has a high performance in accuracy and rapid if designed correctly. Even though in cases with large and computationally complex problems, their network structure may not be interconnected and deep enough to duly train the model. Hence, an urgent need for the emergence of DNNs, or as they are called deep learning, as a deep analytical method for difficult and complex simulated problems.

In the beginning, DNN was used in the field of image and speech recognition, as a modern tool for NN [15]. Recently, DNN has been used in many applications and its effectiveness has been proven with great success, for example, in antennas designs and beamforming capabilities. As for traditional neural networks, which can be described as shoal networks compared to the features provided by DNN, the DNN is more in-deep, complexity, number of layers, and neurons. In addition to an essential feature that distinguishes DNN from traditional NN, which is its ability to discover useful and new features of the input data. This, in turn, makes DNN have great computational depth and is more compatible with major, large-scale, and complex systems. Therefore, when DNN deals with difficult and complex problems that have a large amount of data, they need further computing engines to accelerate the process, such as a graphics processing unit. Moreover, a smart network such as DNN can perform many features such as simultaneous multi-layer processing, feature selection, and monitoring of certain excessive parameters. In the past few years, machine-learning scientists turned their attention to the astonishing results of DNN that have achieved on the ground, particularly in audiovisual media research and prediction [16], so they decided to apply it to various engineering problems [17]–[19]. Recently, DNN is provided to different electromagnetic applications included antenna design, direction-of-arrival estimation, beamforming, multi-input multi-output systems, forward/inverse scattering, radar, and remote sensing due to its superior capabilities [20]–[24].

DNN can supply a hurried beamforming synthesis process while preserving high accuracy levels, minimizing error and time saving, and a possible prediction of the antenna behavior, a better computational efficiency, and a reduced number of necessary simulations. Therefore, the proposed DNN model is an accurate and robust computing approach as an alternative to expensive measurement and simulation. [23] provides a detailed review of various research papers that address the design and optimization of antennas using machine learning, including the various techniques and algorithms used to produce antenna parameters based on desired radiation characteristics and other antenna specifications. Also, [24] presents a novel modified efficient K-Nearest Neighbors (KNN) method, the advantage of this method, which is considered a type of neural network, is the reduction in the number of training and testing data samples. When applying this method to a model (the parameters of this model are less than ten parameters), this model requires only a small number of samples (only from the 10 to 100 samples), for the dataset and some prior information at the beginning to constrain the target domain. Then comes the self-learn stage, and by using some types of rapid simulation, the optimum value can be predicted quickly and accurately. One of the important advantages of this method is its ability to generate more valuable data samples during the training process, so the efficiency of this method is very high.

The Multilayer Perceptron (MLP) with a single hidden layer is a big implication that has been emerged in recent years as interest in NNs has grown. Whenever a large number of hidden layer neurons are used, this network can predict any smooth nonlinear input-output mapping to an arbitrary degree of accuracy [25]. Radial basis function (RBF) neural networks are used in [26] to refine the radiation pattern of non-uniform linear arrays of high superconducting rectangular microstrip antennas. In [27], a phased array in a coordinated scheme based on Taguchi-neural networks is presented. The authors of [28] presented a typical use of back-propagation neural networks for antenna array synthesis and optimization. In [29], the authors used an array $4 \times 1$ of the patch antenna with an inter-element space of 0.28 $\lambda$ for synthesizing the radiation patterns. A DNN was constructed with the input being the radiation pattern and the output being the amplitude and phase of the antenna elements. The proposed DNN has been trained with a large number of samples of radiation models that show reasonable performance in synthesizing the radiation patterns. The radiation pattern produced by DNN was quite similar to that of the input radiation. This proved that deep learning can be used surely for radiation pattern synthesis. In general, antennas are found to be ideal candidates for DNNs because of the intrinsic nonlinearities involved with their radiation patterns.

In this paper, a DNN-based model is optimized in the training phase of a hybrid MGSA-PSO algorithm for the dual resonant frequency computation of the bone-shaped patch antenna (BSPA) with an axial ratio (AR) $< 3 \, \text{dB}$. The accuracy of the model is further validated on a measured BSPA resonating at 28 GHz and 38 GHz simultaneously. Also, the DNNs are used to simplify the antenna array modeling by assessing phases. The key challenge is to find optimal antenna array element weights that result in
a beam-steering radiation pattern with a minimum side-lobe level (SLL) of less than $-30$ dB, thereby improving antenna array efficiency. To verify the validity of the technique, several illustrative examples are placed to beam-steering the pattern toward the desired direction. The paper is organized as follows. The antenna design and array configuration are presented in Section II. Deep neural networks are described in Section III. Section IV presents the results and discussions. Finally, Section VI makes conclusions.

II. DESIGNS AND CONFIGURATIONS
This section presents firstly the designed structure of the proposed dual-band and circularly polarized antenna element. Then, the geometrical arrangement of the uniform circular array consists of 16 antenna elements is introduced.

A. CONFIGURATION OF ANTENNA ELEMENT DESIGN

Fig. 1 shows the proposed antenna structure which consists of a bone-shaped linear array with sinusoidal length distribution and connected through periodic rhombus structures as shown in Fig. 1(a). The antenna ground plane consists of a perfect electric conductor (PEC) layer etched with an elliptical shape as illustrated in Fig. 1(b). The radii of bone ends are R and C. The Rhombus shape with a concave rib and a solid circle connects the groups to each other. As depicted in Fig. 1(a), the antenna mainly consists of four bone shapes, which are $A_{11}$, $A_{12}$, $A_{13}$, $A_{14}$. They are arranged from largest to smallest, as shown in group index 1, and this group of bones is repeated twice in group index 2. Then the group was repeated in the same order and placed group index 1, which are the bones $A_{25}$, $A_{26}$, $A_{27}$, $A_{28}$. Finally, the antenna is fed by a coaxial cable 50 ohm at a point in the antenna structure’s middle.

The antenna is designed then fabricated on the Rogers Duroid™ RT5880 with a 0.508 mm substrate thickness, relative permittivity $\varepsilon_r = 2.2$, and loss tangent $\tan\delta = 0.0009$ substrate. The antenna initial dimensions were illustrated in Table 1.

B. ANTENNA ARRAY DESIGN

In this work, a uniform circular array (UCA) antenna geometry is introduced as shown in Fig. 3. A strong justification for this selection is the symmetry possessed which provides UCAs with a major advantage: the ability to scan a beam azimuthally through $360^\circ$ with little change in either the beam width or the SLL. The array consists of 16 CP elements of the optimized BSPA antenna operating at 28 and 38 GHz simultaneously. The elements are uniformly distributed in a circle configuration with inner ring radius $R_1 = 1.273\lambda$, with equal spacing of $r = 0.5\lambda$ between any two consecutive elements has been considered, where $\lambda$ is the wavelength of 28 GHz.

![FIGURE 1. Proposed antenna structure (a) Top view, (b) Back view.](image1)

![FIGURE 2. (a) Shape of bone structure, (b) Rhombus shape with a concave rib and a solid circle connects the groups to each other.](image2)

| Variable | $W_{11}$ | $W_{12}$ | $W_{13}$ | $W_{14}$ | $W_{21}$ | $W_{22}$ | $W_{23}$ | $W_{24}$ | $W_{25}$ | $W_{31}$ | $W_{32}$ | $W_{33}$ | $W_{34}$ |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Initial value | 3.5 | 3.1 | 2.35 | 1.4 | 1.4 | 2.35 | 3.1 | 3.5 | 3.5 | 3.1 |
| Optimum value | 3.5 | 2.94 | 2.2 | 1.25 | 1.15 | 1.9 | 2.38 | 2.5 | 2.38 | 1.95 |

| Variable | $W_{71}$ | $W_{72}$ | $W_{73}$ | $W_{74}$ | $W_{12}$ | $W_{13}$ | $W_{14}$ | $W_{24}$ | $W_{32}$ | $W_{33}$ | $W_{34}$ |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Initial value | 2.35 | 1.4 | 1.4 | 2.35 | 3.1 | 3.5 | 10 | 0.4 | 2.1 | 7 |
| Optimum value | 2 | 0.8 | 0.7 | 1.1 | 1.4 | 1.59 | 9.9 | 0.38 | 1.9 | 6.57 |

| Variable | $L_1$ | $B_{11}$ | $C_1$ | $S_2$ | $G_1$ | $R_1$ | $S_1$ | $G_1$ | $S_1$ |
|----------|---------|-----------|---------|--------|-------|--------|--------|-------|--------|
| Initial value | 2.55 | 0.13 | 2.1 | 1.7 | 6.1 | 2.25 | 2.5 | 0.6 | 9.3 | 3.5 |
| Optimum value | 2.5 | 0.12 | 2 | 1.68 | 5.9 | 2.3 | 2.3 | 0.58 | 9.21 | 3.39 |
The proposed 5G mm-wave antenna array is designed to synthesize the beam patterns in different directions. The radiation pattern can be steered in the desired direction with a high gain and a side-lobe level less than $-30\text{ dB}$ by adjusting the phase of the input signal allocated to each antenna element. Therefore, the MGSA-PSO algorithm is considered to optimize the feeding phases for DNN learning.

### III. DEEP NEURAL NETWORK (DNN)

Deep learning distinguishes itself from machine learning by combining feature collection and regression/classification, having a larger number of neurons, processing simultaneously on several layers, inherently extracting features, and evaluating optimum network hyperparameters. The data in the DNN system is evaluated by moving it through the neurons in the multi-layered hierarchy, and the evaluated information is then passed on to the next layers, allowing a more convenient learning model to be constructed.

In the beginning, we briefly introduce the DNN description that was applied to the BSPA design. Multilayer perceptions (MLPs) [31]–[33], which were successfully and frequently employed in many engineering applications, are favored in this investigation. Many algorithms, such as Levenberg-Marquardt (LM), backpropagation, and delta-bar-delta, can be used to train the MLP. MLPs are trained in this study using the GSA-PSO algorithm [6], [7], which has quick learning and high convergence capabilities. As shown in Fig. 4, the MLP has five layers: an input layer, an output layer, and three hidden layers. The neurons of the layer merely act as buffers for distributing the input signals $x_i$ to the neurons of the hidden layer. Each hidden layer neuron $j$ sums its input signals $x_i$ after weighting them with the strengths of the respective input layer connections $w_{ji}$, and computes its output $y_j$ as a function $f$ of the sum, namely

$$y_j = f \left( \sum \omega_{ji} x_i \right) \quad (1)$$

where $f(\cdot)$ can be a simple threshold function, a sigmoid, hyperbolic tangent, a radial basis function, a purelin function, etc. [32], [33]. Similarly, the output of neurons in the output layer is computed. When training a network, one of the available learning algorithms is used to adjust the network’s weights. At time $t$, the learning algorithm returns the change $\Delta \omega_{ji}(t)$ in the weight of a connection between neurons $i$ and $j$. The weights for the LM learning algorithm are updated using the formula below.

$$\omega_{ji} (t + 1) = \omega_{ji} (t) - \Delta \omega_{ji} (t) \quad (2)$$

$$\Delta \omega_{ji} = [J^T (\omega) J (\omega) + \mu I]^{-1} J^T (\omega) E (\omega) \quad (3)$$

where, $\mu$, $I$, and $E (\omega)$ are the Jacobian matrix, a constant, identity matrix, and error function, respectively. The Jacobian matrix contains the first derivatives of the errors with respect to the weights and biases. After each successful step, the value of $\mu$ is dropped, and it is only increased when a step would increase the sum of squares of errors. A DNN model with five layers was applied in this work, including the input layer, three hidden layers, and the output layer. The number of epochs in the training procedure is 150. In addition, the input layer, hidden layer, and output layer all used the tangent sigmoid, tangent sigmoid, and purlin functions, respectively.

Secondly, we applied DNN on a circular disk antenna array for beam-steering. As shown in Fig. 5, multi-layer networks have an input layer whose neurons code the information supplied to the network, a configurable number of “hidden” internal layers, and an output layer. In the same layer, neurons do not communicate with one another. These networks’ learning process is supervised. The input nodes make up the first layer. A feed-forward neural network with one hidden layer and a Multilayer Perceptron MLP node function at each hidden node is known as an MLP network. The dimension of the input vector is equal to the number of nodes, $L$ [34]–[36]. Where $j$ is the input layer’s index ($j = 1, 2, \ldots, L$) and $i$ is the hidden layer’s index ($i = 1, 2, \ldots, N$). With $k = 1, 2, \ldots, M$, $k$ is the index of the output layer.

The interconnection weights are calculated using the minimal error between the neural model output $y_k$ and the training data $d_k$. The goal of the training procedure is to fine-tune the network interconnection weights $\omega_{ij}$ and $\omega_{ki}$ in order to reduce the error function $E(p)$, which is defined as:

$$E(p) = \frac{1}{2} \sum_{k=1}^{M} \sum_{i=1}^{N} \left[ y_k (x_j, \omega_{ij}, \omega_{ki}) - d_k \right]^2 \quad (4)$$

where $p = 1, 2, \ldots, P$ denotes the training set’s index. The back-propagation technique described in [27] is used in this iterative procedure. The weights $\omega_{ij}$ and $\omega_{ki}$ are updated for

![FIGURE 3. 2D view of UCA antenna structure, (a) front view, (b) back view.](image)

![FIGURE 4. The proposed 5-layer DNN framework.](image)
cepts described in the preceding part, divide the space into
is the neuron employed in this network. To study the con-
neuron, whose activation function is a tan sigmoid function,
converged quickly and that the neural model we created was
enced by the essence of the nonlinearity to be modeled.
works. Finally, we get the output of the network.
\[ x_1, x_2, \ldots, x_n \]
samples. Then present input vectors \( x \):
for the testing input
\[ \Delta \omega_m = -\eta \frac{\partial E}{\partial \omega_m} \quad (5) \]

Sector-width intervals of 15° and SLL intervals of −30 dB
were used in the training set. Fig. 6 depicts the mean square
error performance of the MLP Network. The ability of neural
networks to generalize is one of their main advantages. This
means that even if a trained network has never seen data
from the same class as the learning data, it will classify it.
Developers of real-world applications typically only have
access to a small portion of all possible patterns of neural
network generation. The dataset should be divided into three
sections to achieve the best generalization: The training set is
used to train a neural network; during training, the dataset’s
error is minimized, the validation set is used to assess a neural
network’s success on patterns that have not been trained
during the learning process and a test set for determining a
neural network’s overall efficiency.

In such a case, we have two main steps called network
designing and network testing (generalization). In network
designing, the input vectors \( \{x_p, p = 1, 2, \ldots, 10\} \) is firstly
formed, then generating input/output pairs \( \{x_p, \phi_q\} \), where
\( q = 1, 2, \ldots, 18 \), then design the neural networks. In the
network testing, we form the vectors \( x'p \) for the testing input
samples. Then present input vectors \( x'p \) to the neural
networks. Finally, we get the output of the network.

The number of hidden neurons chosen is heavily influ-
enced by the essence of the nonlinearity to be modeled.
In our situation, 30 hidden neurons ensured that the algorithm
converged quickly and that the neural model we created was
accurate as depicted in Table 2. The continuous nonlinear
neuron, whose activation function is a tan sigmoid function,
is the neuron employed in this network. To study the con-
cepts described in the preceding part, divide the space into
24 sectors and repeat every 15 degrees between 0° and 360°
degrees inclusively. More exact space division sectors can
be achieved by increasing the number of element arrays.
A 24-bit binary code is used as the input vector for the
neural network (one bit for each sector). A bin input of (+1)
indicates a source in the sector that is exactly on (main lobe).
Convergence could then be completed faster.

| Table 2. Typical values of parameters use in Back-Propagation algorithm. |
|--------------------------|-----------------|-------|
| Parameters               | Symbol | Value |
| Neuron in the input layer| n     | 18    |
| Neuron in the output layer| m    | 16    |
| Neuron in the hidden layer| h    | 30    |
| Coefficient of training  | η     | 0.02  |

IV. RESULTS AND DISCUSSIONS
In this section, the simulation results of the optimized dual-
band circular polarized BSPA antenna will be firstly pre-
sent, analyzed, and compared as the building block of the
antenna array. Then, the UCA array simulation results will
be offered to show their capabilities for steering the beam
patterns in different directions.

A. ANTENNA DESIGN
The authors in this work present for the first time a design for
the antenna called Bone-Shaped Patch Antenna (BSPA) that
consists of several parts each one is considered for a specific
function. The main objective is to resonate the antenna at 28
and 38 GHz simultaneously with an AR < 3 dB. Firstly,
the shape of the sinusoidal bones has been studied as shown
in Fig. 7a. It is found that changing the sinusoidal envelop
shape effect directly on the antenna reflection coefficient and
correspondingly on the antenna matching. Then the effect
of the oval shape defected in the ground plane (DGS) is
analyzed. The results revealed its effect on the antenna real-
ized gain as presented in Fig. 7b, whereas, the larger DGS
oval shape area leads to a higher realized gain value at the
operating frequencies. Regarding the rhombus shape located
in the middle of the antenna, it plays a major role in improving
the antenna radiation efficiency as shown in Fig. 7c. Finally,
the effect of radii bone ends has been studied as presented in
Fig. 7d. Whereas, the circular polarity and the axial ratio of
the antenna are greatly affected just by changing the direction
of the bone shape in each group.

In order to generate a database for modeling the DNN, sim-
ulations of 150 BSPAs with various geometrical parameters
are performed using CST-MWS [30]. The parameters of the
simulated BSPAs are topologically illustrated in Fig. 8. The
antenna parameters are considered in three ground plane slot
dimensions \( G_1 \times S_2 \) groups of \( 10 \times 3.6 \text{ mm}^2 \), \( 9.3 \times 3.5 \text{ mm}^2 \),
and \( 8.5 \times 3 \text{ mm}^2 \). Each group has 50 BSPAs that comprise a
parameter combination of \( G_1 \times S_2 \), e.g. for the first group of
\( 10 \times 3.6 \text{ mm}^2 \), there are 50 ESPAs including the parameter
combination of \( (W_{11}: 3.5, 3.36, 3.22, 3.08, 2.94 \text{ mm}) \times (W_{34}: 3.5, 3.36, 3.22, 3.08, 2.94 \text{ mm}) \times (W_{24}: 3.5, 3 \text{ mm}) \).
The simulated resonant frequency $f_r$ of each BSPA with a particular antenna parameter is determined by CST.

Fig. 9 shows the simulated resonant frequency variation versus antenna number. Whereas the resonant frequencies decrease with the antenna ground plane slot dimensions, and hence there is a high nonlinearity between the antenna parameters and the resonant frequencies. Therefore, computing the resonant frequency of BSPA is a complex and high nonlinear problem. The simulations are performed between the frequency range of 28–38 GHz at 300 points.

According to the relationship between the input and the target, the DNN model with three layers was trained to produce the resonant frequency for each parameter set of the antenna. As many as 150 BSPAs were employed for training while 15 BSPAs were used for testing the DNN model. In order to visually recognize the relationships between the results, the scatter diagrams of simulated and computed resonant frequency results are shown in Fig. 10 for the training and testing datasets. The value of the average percentage errors (APE) for the resonant frequencies has been computed by the DNN model as depicted in Fig. 11 [37]. It is clear that, the APE value is affected by the assigned number of training points for any deep learning application. Whereas, the system accuracy improves by increasing the number of training points and vice versa. Based on Fig. 11, which illustrates the topology of calculating the APE for CST models, an appropriate APE of 0.236 % was obtained as for the 150 BSPAs’ training data which increased to 1.587 by decreasing the training data to 72. It is clearly seen that the points will follow a linear pattern; means there is a high linear correlation between the results.

To further investigate the validity of the present approach, a BSPA operating at 28 and 38 GHz with the dimensions illustrated in Table 1 was designed via CST and then fabricated. These parameters not used in the training process, whereas, in the training data there is no antenna resonated simultaneously at 28 and 38 GHz as shown in the inset Fig. 10a. Fig. 12 shows a good agreement between the measured and simulated results. As shown in Fig. 12(a), the antenna can achieve good matching at both frequency bands to be $-26.61$ and $-24.54$ at 28 and 38 GHz, respectively, with realized gain of 8.97 and 8.65 dBi as depicted in Fig. 12b. Furthermore, the antenna had an axial ratio less than 3dB for $\theta = 90^\circ$ and $\phi = 0^\circ$ at the resonance frequencies, in addition, as shown in Fig. 12c, the radiation efficiency is found to be 90.5 % and 87 % at 28 GHz and 38 GHz, respectively. Fig. 13 presents...
the simulated 3D radiation pattern of the optimized BSPA antenna at 28 and 38 GHz.

**B. ANTENNA ARRAY**

The proposed approach has been thoroughly tested as shown in Fig. 6, as seen by the examples below. For synthesizing the 16-element antenna array, the feeding voltages were set with constant amplitudes and variable phases [38], [39]. For the reference antenna, the predicted simulation results must demonstrate radiation patterns with low SLL (at −30 dB) and major lobes pointing in the direction of valuable signal. The desired radiation pattern is given from 0° to 360° in our application, and the database contains all the data (input/output) produced through simulation with the MGSA-PSO algorithm.

![Figure 10](image1.png)

**FIGURE 10.** Scatter diagrams of the simulated and computed resonant frequency values by the DNN model for (a) Training data, (b) Test data.

![Figure 11](image2.png)

**FIGURE 11.** The topology of the calculating APE for CST models.

The proposed antenna array is analyzed using CST-MWS and linked with MGSA-PSO algorithm, MATLAB-coded, to optimize the antenna array phases. Accordingly, the following objective function is applied to achieve the goal. 24 desired directions of the UCA from 0° to 360° by step 15° were optimized by this objective function as a training data for DL mechanism.

$$Obj = \max |E_t(\theta, \phi)| + \min(\text{SLL}_{dB} - 30)$$

The graphical output of the regression is shown in Fig. 14. The network outputs are placed against the targets as open circles. The best linear fit is indicated by a dotted line. The solid line indicates a perfect match (output equal to the targets). Because the fit is so superb in this case, it is difficult to tell the difference between the best linear fit line and the ideal fit line.

To demonstrate the effectiveness of the method identified in the previous section for steering single beams in the desired direction by controlling the phase excitation of each array element, 24 desired directions of the disc array with N = 16 elements were performed. In various settings, the numerical
findings in Fig. 15 show that NNs with the MGSA-PSO algorithm have outstanding phase control capabilities for beam pattern synthesis. It is vital to test the neural network once it has completed the training phase on a different database than the one used for learning. This test allows you to evaluate the neural system’s performance as well as identify the problematic data types. It will either update the network design or adjust the learning base if the performance is not adequate (each data class’s distinguishing traits or representativeness).

Many examples of simulations are explored at $\varphi = 40^\circ$, $142^\circ$, $205^\circ$, and $320^\circ$, in order to test the proposed methodology for the synthesis of a circular disc array. It is evident that the side-lobe level requirement of $-30$ dB has been met. Fig. 16 shows the simulated results for 3D antenna radiation pattern synthesis using the DNN approach at $28$ GHz for the built antenna array system with 16 elements. The results reveal that the desired and synthesized specifications are extremely closely aligned. This proves the recommended procedure’s efficacy. This shows that the proposed procedures are effective. The DNN has superior learning, generalization, parallel processing, and error endurance properties, resulting in ideal solutions in applications where nonlinear mapping of complex data should be modelled. This method employs a DNN, which can be trained to handle any amount of elements, spacing, and excitation. The parameters with respect to the input can be found once the network has been trained.

V. CONCLUSION

In this study, DNN is implemented to the computation of the resonant frequency of the BSPAs, and hence a DNN-based soft computing framework is modeled using a full-wave 3D EM analysis platform. The network is trained with a set of input-output data pairs based on MGSA-PSO algorithm. A database enclosing the resonant frequency of 150 BSPAs is defined by the simulations with different geometry and electrical parameters. For training and testing the model, the database is split into datasets #135 and #15, respectively. As a result, the proposed DNN model was used to estimate the resonant frequencies with the greatest precision, making it an efficient and potentially viable alternative to costly measurement and simulations. Then, the DNN model has applied to beam-steer the radiation pattern of the designed antenna array. Results show that there is an agreement between the desired specifications and the synthesized ones.

ACKNOWLEDGMENT

The authors would like to express their gratitude to the National Telecommunication Regulatory Authority (NTRA), Ministry of Communication and Information Technology in Egypt for their support.

REFERENCES

[1] R. C. Hansen, Phased Array Antennas, vol. 213. Hoboken, NJ, USA: Wiley, 2009.
[2] I. Uchendu and J. R. Kelly, “Survey of beam steering techniques available for millimeter wave applications,” *Prog. Electromagn. Res. B*, vol. 68, pp. 35–54, 2016.
[3] M. U. Afzal and K. P. Esselle, “Steering the beam of medium-to-high gain antennas using near-field phase transformation,” *IEEE Trans. Antennas Propag.*, vol. 65, no. 4, pp. 1680–1690, Apr. 2017.
[4] S. S. K. Puranam, A. Gopinath, and R. Sainati, “Amplitude based beam steering,” in *Proc. IEEE Int. Symp. Phased Array Syst. Technol. (PAST)*, Oct. 2019, pp. 1–4.
[5] K. R. Mahmoud and A. M. Montaser, “Design of dual-band circularly polarised array antenna package for 5G mobile terminals with beam-steering capabilities,” *IET Microw., Antennas Propag.*, vol. 12, no. 1, pp. 29–39, 2018.
[6] K. R. Mahmoud and A. M. Montaser, “Synthesis of multi-polarised upside conical frustum array antenna for 5G mm-wave base station at 28/38 GHz,” *IET Microw., Antennas Propag.*, vol. 12, no. 9, pp. 1559–1569, Jul. 2018.
[7] K. R. Mahmoud and A. M. Montaser, “Performance of tri-band multi-polarized array antenna for 5G mobile base station adopting polarization and directivity control,” IEEE Access, vol. 6, pp. 8862–8869, 2018.

[8] K. R. Mahmoud, A. Baz, W. Alhakami, H. Alhakami, and A. M. Montaser, “The performance of circularly polarized phased sub-array antennas for 5G laptop devices investigating the radiation effects,” Prog. Electromagn. Res. C, vol. 110, pp. 267–283, 2021.

[9] K.-K. Yan and Y. Lu, “Sidelobe reduction in array-pattern synthesis using genetic algorithm,” IEEE Trans. Antennas Propag., vol. 45, no. 7, pp. 1117–1122, Jul. 1997.

[10] K. R. Mahmoud, M. I. Eladawy, R. Bansal, S. H. Zainud-Deen, and S. M. Ibrahim, “Analysis of uniform circular arrays for adaptive beam-forming applications using particle swarm optimization algorithm,” Int. J. RF Microw. Comput.-Aided Eng., vol. 18, no. 1, pp. 42–52, Jan. 2008.

[11] K. R. Mahmoud, M. I. Eladawy, R. Bansal, S. H. Zainud-Deen, and S. M. Ibrahim, “Analysis of uniform circular arrays for adaptive beam-forming applications using particle swarm optimization algorithm,” Int. J. RF Microw. Comput.-Aided Eng., vol. 18, no. 1, pp. 42–52, Jan. 2008.

[12] K. R. Mahmoud and S. Hamad, “Parallel implementation of hybrid GSA-NM algorithm for adaptive beam-forming applications,” Prog. Electromagn. Res. B, vol. 58, pp. 47–57, 2014.

[13] K. R. Mahmoud, “Synthesis of unequally-spaced linear array using modified central force optimisation algorithm,” IET Microw., Antennas Propag., vol. 10, no. 10, pp. 1011–1021, 2016.

[14] I. A. Basheer and M. Hajmeir, “Artificial neural networks: Fundamentals, computing, design, and application,” J. Microbiol. Methods, vol. 43, no. 3, pp. 1–31, 2000.

[15] M. Sarevska, “Signal detection for neural network-based antenna array,” in Proc. PIERS, Mar. 2013, pp. 25–28.

[16] K. R. Mahmoud and S. Hamad, “Parallel implementation of hybrid GSA-NM algorithm for adaptive beam-forming applications,” Prog. Electromagn. Res. B, vol. 58, pp. 47–57, 2014.

[17] K. R. Mahmoud, “Synthesis of unequally-spaced linear array using modified central force optimisation algorithm,” IET Microw., Antennas Propag., vol. 10, no. 10, pp. 1011–1021, 2016.

[18] A. M. Montasser, K. R. Mahmoud: Deep Learning Based Antenna Design and Beam-Steering Capabilities