International Journal of Scientific & Engineering Research Volume 12, Issue 7, July-2021 1300
ISSN 2229-5518

Direction of Arrival (DOA) Estimation Based on Multi-layer Neural Network

Kabiru N. Aliyu, M. H. Ali, A. Musa* Physics/Electronics Department,
Bayero University, Kano. Nigeria
Kbrnsr30@gmail.com, *amusa.phy@buk.edu.ng

Abstract— The Direction of arrival (DOA) estimation in wireless communication enhances the communication quality through beam steering. Different algorithms of DOA estimation like beamforming, maximum likelihood, and subspace-based techniques failed due to sparsity and complexity of the system. To solve this problem, in this work, Deep Neural Network (DNN) is proposed for the DOA estimation of received signal. The estimation is performed by assuming one source is transmitting signal at a time, the training data is generated with 1-degree resolution at different Signal-to-noise ratio (SNR) scenario. Simulation result shows high resolution in localizing the DOA of the received signal with real time performance appropriate for many applications such as fault detection, radar and seismic exploration.

Index Terms— Direction of Arrival (DOA), Deep Neural Network (DNN), Beam Steering, Multi-layer Neural Network, SNR.

1 Introduction

Many researchers in wireless communication system experience many challenges in designing wireless system that can achieved high network capacity, high data rates, better quality of service with less dropped calls, fading and multipath distortion. These results in requirements of promising technology that can offer significant improvement in data throughput and spectral capacity. Multiple input multiple output (MIMO) system is essential for these features [1]. MIMO system has multiple antennas at the transmitter and receiver for communication. Finding DOA of received signal in wireless communication is very important in communication improvement through beam steering for signal level enhancement and interference avoidance. The most widely used localization system is the global positioning system (GPS) but it is not applicable in obstructed environments like foliage, and it is usually associated with higher power consumption and hardware cost. The basic localization methods are; time of arrival (TOA), time difference of arrival (TDOA) and angle of arrival (AOA) or direction of arrival (DOA) estimation [2].

DOA estimation is a technique usually used for finding the direction of the received signals at a reference array sensor transmitted from a far-field localized source, this technique is applicable to the field of localization and tracking system such as radar system, sonar, seismic exploration, medicine, and mobile communications. It is also applicable to the determination of multipath channel structure. Different algorithm is applied for DOA estimation which includes beamforming, spectral estimation, Bartlett, Capon, ESPRIT, Min-norm, multiple signal classification (MUSIC) and recent algorithm that shows high resolution and real time application i.e., machine learning algorithm.

Multi-layer or Deep learning is a neural network with multiple hidden layers that has an inputs, analogous weights and bias with activation function and output. Machine learning have wide range of applications and high efficiency in executing tasks due to non-linearity in the system.

In this work, The multi-layer neural network is proposed for the DOA estimation. The estimation accuracy of the proposed network is evaluated under different SNR scenarios and other parameters of the network. The transmitted signal propagates through additive white Gaussian noise (AWGN) channel and impinges multiple uniform linear arrays at the receiving ends (far-field).

The remaining part of the paper constitute different sections as follow. Section 2 presents DOA estimation formulation. Also, general formulation of DNN and its application to DOA estimation is presented in Section 3. Section 4 presents the performance evaluation of DOA estimation using DNN and conclusion followed in Section 5.

2 Direction of Arrival (DOA) Estimation

Different algorithm is applied for DOA estimation. Capon algorithm was initially applied to MIMO system by [3] based on time reversal for MIMO radar system. Also, [4] compare Beamforming, Capon, MUSIC, and first-norm singular value decomposition ($l_1 - SVD$) in estimating DOA of two sources with practical application using uniform linear array antennas. Moreover, recently, Deep neural network is applied to the DOA estimation [1],[5],[6].

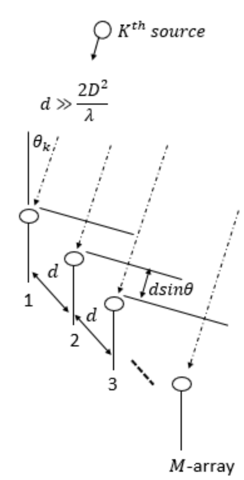

Fig. 1. M uniform linear array.

2.1 Correletion Matrix of Receive Signal

Let consider Fig. 1. a MIMO system with K number of sources transmitting a narrow band signal $g_k(t)$, $k = 1,2,3,\dots,K$ from a far field localized source, with speed of light c and wavelength λ. The transmitted signal arrives at the M-array antenna with an equally interspaced distance between the two neighboring antenna be $\mathrm{d} = {}^{\lambda}/_{2}$, The transmitted signal arrive at reference antenna element with incidence angle θ_k, And the received signal $x_m(t)$ at time t is expressed as

$$x_m(t) = \sum_{k=1}^{K} U(\theta) g_k(t) + n_m(t) \qquad (3)$$

$$t \in \{1,2,\dots,T\} \qquad (4)$$

where T is the number of snapshot, $\boldsymbol{U}(\theta)$ is a steering matrix which is a function of steering vector $u(\theta_k)$ defined as

$$\boldsymbol{U}(\theta) = [u(\theta_1), u(\theta_2), \dots, u(\theta_k)] \qquad (5)$$

$$\boldsymbol{u}(\theta_k) = [1, e^{\frac{j2\pi dM \sin(\theta_k)}{\lambda}}, \dots, e^{\frac{j2\pi d(M-1)\sin(\theta_k)}{\lambda}}]^T \qquad (6)$$

$$\boldsymbol{g}(t) = [g_1(t), g_2(t), \dots, g_k(t)]^T \qquad (7)$$

$$\boldsymbol{n}(t) = [n_1(t), n_2(t), \dots, n_m(t)]^T \qquad (8)$$

where $\boldsymbol{g}(t)$ is transmitted signal, $\boldsymbol{n}$(t) is AWGN, and $[.]^T$ is transpose. the received signal in vector form can be written as

$$\boldsymbol{x}(t) = \boldsymbol{U}(\theta)\boldsymbol{g}(t) + \boldsymbol{n}(t) \qquad (9)$$

The correlation matrix of the MxM of the received signal vector can be written as:

$$\boldsymbol{R}_{xx} = E[\boldsymbol{x}(t)\boldsymbol{x}^H(t)] = \boldsymbol{U}\boldsymbol{R}_{gg}\boldsymbol{U}^H + \sigma_N^2 \boldsymbol{I}_M \qquad (10)$$

where $\boldsymbol{R}_{gg} = E\{\boldsymbol{g}(t)\boldsymbol{g}^H(t)\}$ is the signal covariance matrix, σ_N^2 is the common variance of the noises and E{.}is the statistical expectation, and $[.]^H$ is a Hermitian matrix.
The correlation matrix $\boldsymbol{R}_{xx}$ is use as input to the DNN.

3 Formulation of Multi-layer Neural Network

Generally, a multilayer neural network with inputs $q_1, q_2, q_3 \dots q_L$ multiple hidden layers, and output v is shown in Fig. 2, where each input has analogous weight $w_{1,1}, \dots w_{I,L}$ and summed up with a bias. Then activated with non-linear activation function f. Some of the activation functions are sigmoid, tanh, Relu and leaky Relu.

The output of the lth NN can be written as

$$y_i = \sum_{l=1}^{L} w_{i,l}\, q_l + b_i \qquad (11)$$

where b_i is a bias.
when $w_{i,0} = b_i$ and $q_0 = 1$, equation (11) can be written as

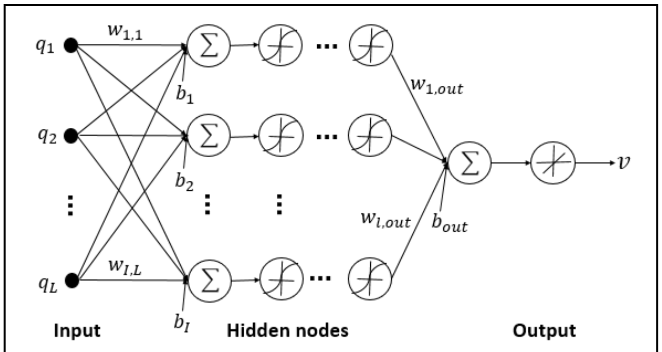

Fig. 2. Schematic diagram of multi-layer neural network.

$$v_i = f\left(\sum_{l=0}^{L} w_{i,l} q_l\right) \qquad (12)$$

the vector-form of equation (13) can be written as

$$\begin{vmatrix} y_2 \\ \vdots \end{vmatrix} = \begin{vmatrix} w_{2,0} & w_{2,1} & \cdots & \cdots w_{I,L} \\ \vdots & \vdots & . & . \end{vmatrix} \begin{vmatrix} q_1 \\ \vdots \end{vmatrix} \qquad (13)$$

These can be simplified as

$$\boldsymbol{v} = f(\boldsymbol{y}) \qquad (14)$$

where

$$\boldsymbol{y} = \boldsymbol{W} \begin{vmatrix} 1 \\ \cdot \end{vmatrix} \qquad (15)$$

The output of h^{th} multilayer structured NN with can be formulated as

$$\boldsymbol{v}^{(h)} = f(\boldsymbol{y}^{(h)}) \qquad (16)$$

where

$$\boldsymbol{y}^{(h)} = \boldsymbol{W}^{(h)} \begin{vmatrix} 1 \\ ..(h-1) \end{vmatrix} \qquad (17)$$

and $\boldsymbol{v}^{(0)} = \boldsymbol{y}$.

All tables and figures will be processed as images. You need to embed the images in the paper itself. Please don't send the images as separate files.

3.1 DOA Estimation in Multi-layer NN

The input to DNN is MxM correlation matrix R_{xx} of the received signal, the element of the upper and lower triangular part of R_{xx} are the same and complex-valued. The diagonal and lower triangular part is used, where the complex element is decomposed into real and imaginary values excluding diagonal which gives one-row vector of length M^2, the input to the DNN is expressed as

$$q = [r_{1,1} r_{2,2} \dots r_{M,M} \, \Re(r_{2,1}), \Im(r_{2,1}), \dots \dots \Im_{M,M-1})]^T \tag{18}$$

where $\Re(\cdot)$ and $\Im(\cdot)$ is real and imaginary element.

The total number of angles at the output of the network depends on the search range from θ_{min} *to* θ_{max} and its resolution $\Delta\theta$, this can be determined as

$$\frac{\theta_{max} - \theta_{min}}{\Delta\theta} + 1 \tag{19}$$

where θ_{max} and θ_{min} is the maximum angle, minimum angle of the search range.

The output of DNN is determined based on the probability of the angle at which the received signal arrive at the array element.

4 Accuracy of DOA Estimation Using DNN

The performance of the proposed DNN for DOA estimation is evaluated in this section,

4.1 Simulation Parameters and conditions

In our study, Matlab software is used as a simulation toolbox and other parameters used include, narrow band signal which is randomly generated. a $2.4GHz$ carrier frequency was used, and each transmitted signal is arriving at a five-array element within $[-60\ 60]$ degree in a step of 1-degree resolution. The interspacing between the array element is d=λ/2. The number of snap-shot used is T=300 samples is taken at a time to calculate $\boldsymbol{R}_{xx}$. A column vector of 25 rows is used as input to the proposed network. The search range of the DOA at the output of network is $[-60\ 60]$ in step of 1-degree, this gives a total of 121-degree at the outputs. Different number of hidden layers with certain number of neurons is used. The DNN is trained with Levenberg Marquart back-propagation algorithm using $96{,}800$ training data. The data used for training was divided into 75% train sets, 10% validation sets and 15% testing sets. $16{,}940$ randomly generated test data with different DOA were analyzed and one DOA is evaluated at a time.

The performance of the estimated output is assessed using the probability of correct DOA estimation and mean absolute error (MAE). The MAE is written as

$$\text{MAE} = \frac{1}{N} \sum_{n=1}^{N} \left| \hat{\theta}^n - \theta^n \right| \tag{17}$$

Where $[.]^{(n)}$ and the nth number of test is N.

4.3 Train Data Generation

Different SNR scenarios was used for the generation of train data,

1. Constant $0dB$ SNR.
2. Constant $5dB$ SNR
3. Constant $30dB$ SNR.
4. Five step increase SNR from 0 to $30dB$.
5. One step increase SNR from 0 to $30dB$.

For training the DNN, each generated train data (input data) has it is corresponding target (output), the target is the actual angle of arrival used for the generation of covariance matrix.

In the training process, 2 different cases of DNN were considered,

1. Network with 2 hidden layers, each hidden layer has 10 neurons.
2. Network with 2 hidden layers, each hidden layer has 20 neurons.

Fig. 3. shows the Matlab model of DNN with 25 input, 2 hidden layers, 10 neurons in each layer and one output.

The probability of correct DOA estimation is evaluated in Fig. 4 and Fig. 5 and MAE of DOA estimation is evaluated inFig. 6, and Fig. 7 for different number neurons in each hidden layer. The abscissa of each figure represents the SNR use for the generation of testing data while the legend represents the SNR for training the DNN.

Fig. 4 and Fig. 5 shows the probability of correct estimation using DNN with 10 and 20 neurons in each layer. The results show that, the DNN trained with SNR in step of $1dB$ and $5dB$ have similar performance with slight difference around $3dB$ SNR and the performance of estimation DOA reach almost 100% correct estimation at anything above $4dB$, this means DNN trained with $1dB$ and $5dB$ step SNR can estimate all DOA at this stage with almost no error or minimum error. However, the performance of the network trained with constant $30dB$ SNR is increasing as the SNR of the testing data increase, then it reaches almost 100% accuracy at $15dB$ testing SNR. This is because, the level of the noise in the testing data decreases as its SNR is approaching the SNR used for the training. Hence, the network recognizes the features in the testing data. Also, the DNN trained with constant $0dB$ or $5dB$ have good performance at these SNR, then the performance deteriorates at other testing SNR. This implies that, DNN trained with constant SNR get confuse at estimation stages due to lack of information of the other SNR. Furthermore, it indicates that the DNNs trained with high SNR training data provide better performance compared to the DNNs trained with low SNR training data, this means that correlation matrice

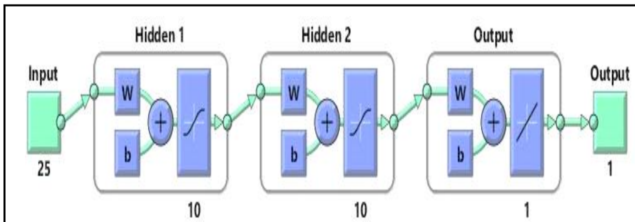

Fig. 3. Matlab model of DNN

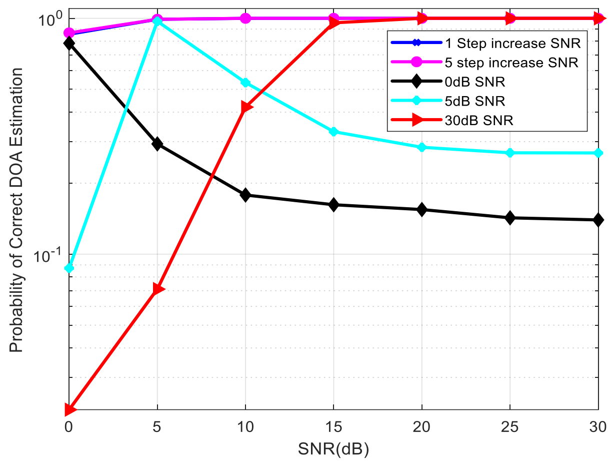

Fig. 4. Probability of correct DOA estimation using 2 hidden layers, each layer has 10 neurons

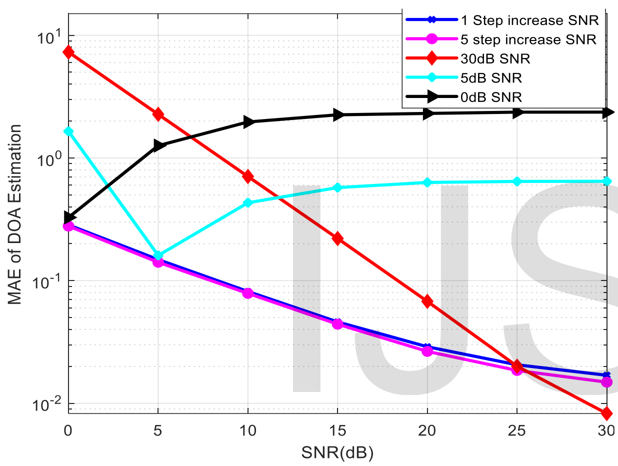

Fig. 5. MAE using 2 hidden layers, each layer has 10 neurons.

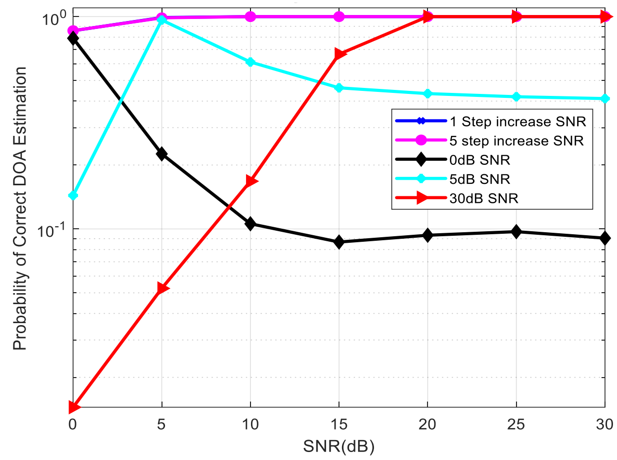

Fig. 6. Probability of correct DOA estimation using 2 hidden layers, each layer has 20 neurons.

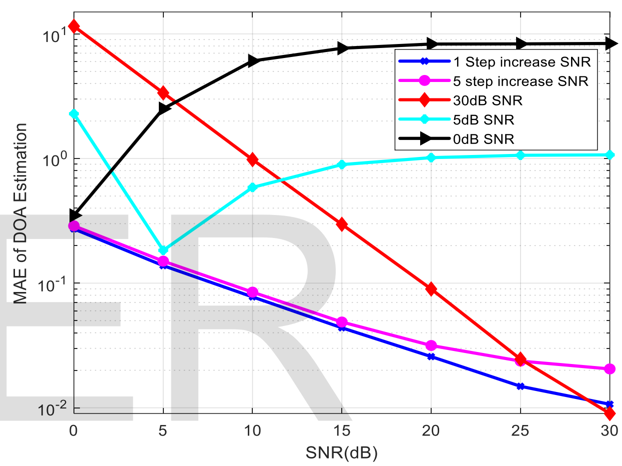

Fig. 7. MAE using 2 hidden layers, each layer has 20 neurons

generated with low SNR data is featureless matrices and the DNN get confuse in the learning process due high level of noise in the received signal.

However, to describe the error occur in the network while estimating the DOA of the received signal MAE were used.

Fig. 6, and Fig. 7 depict the MAE for DOA estimation using 2 hidden layers with 10 and 20 neurons in each layer. it displays that, DNN with 2 hidden layers, 20 neurons outperform DNN with 2 hidden layers, 10 neurons. because, the DNN with 2 hidden layers, 10 neurons get over-fitted with the training data and it can learn from the training data but it cannot generalize to the new testing data due to large number of training data with inadequate number of layers and neurons.

5 CONCLUSIONS

In this paper, we propose DOA estimation of received signal using DNN. The estimation accuracy of the proposed network is evaluated under different SNR scenarios and other parameters of the network. Additive white Gaussian noise (AWGN) channel and uniform linear arrays antenna were used at the receiving ends (far-field). The simulation result shows that, the propose DOA estimation using DNN with 2 hidden layers, 20 neurons have better performance in the probability of correct DOA estimation and MAE. In application to the field of wireless sensor network, one cell is divided into 3 sectors with one sensor served as wireless gateway node placed at a center of the cell, 3 trained DNN with 2 hidden layers, 20 neurons is implemented into WGN, each sector has one trained DNN to estimate the DOA of received signal in that sector. This can be used for localization of sensor or fault position and communication improvement through beam steering.

References

[1] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive MIMO system," IEEE Trans. Veh. Technol., vol. 67, no. 9, pp. 8549–8560, 2018.

[2] W. dargie and C. Poellabauer, Fundamentals of Wireless Sensor Networks Theory and Practice. 2008.

[3] F. Foroozan, A. Asif, Y. Jin, and J. M. F. Moura, "Direction finding algorithms for time reversal MIMO radars," IEEE Work. Stat. Signal Process. Proc., pp. 433–436, 2011.

[4] S. A. Alawsh, A. H. Muqaibel, and M. S. Sharawi, “Sparse Direction-of-Arrival Estimation for Two Sources with Constrained Antenna Arrays,” in 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), 2017, pp. 666–670.
[5] Y. Kase, T. Nishimura, T. Ohgane, Y. Ogawa, D. Kitayama, and Y. Kishiyama, “DOA Estimation of two targets with deep learning,” 2018 15th Work. Positioning, Navig. Commun. WPNC 2018, no. 2, pp. 1–5, 2018.
[6] M. F. Unlersen and E. Yaldiz, “Direction of Arrival Estimation by Using Artificial Neural Networks,” Proc. - UKSim-AMSS 2016 10th Eur. Model. Symp. Comput. Model. Simul., pp. 242–245, 2017.