Deep Learning with Structure - Project Report "Deep Learning-Based Calibration for Millimeter-Wave Phased-Array Antennas"

Idan Roth
idanroth@ece.ubc.ca
44149631

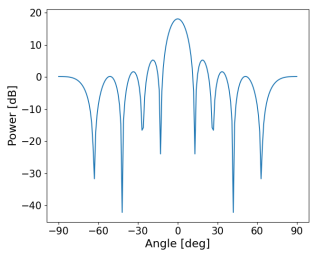
Abstract

In the following project, a Transformer-based neural network model for phased array antenna calibration, which uses sub-patterns of the radiation power pattern as input, is suggested. The proposed trained model estimates the antenna array system's excitation phase errors, which are crucial for the successful operation of the antenna prior to its deployment. Only a single radiation pattern measurement is required for the proposed method which indicates how efficient and time-saving the method is compared to the common conventional methods. Moreover, an attempt to reproduce the results of a close-related work are made, its generalization performance in terms of the antenna characteristics is investigated, and a comparison between the two models is presented.

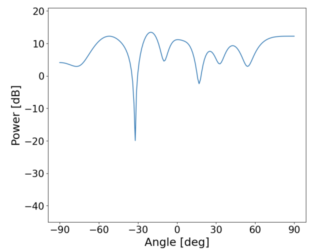
1 Introduction

The rapid growth in wireless networks traffic rates and user demands, drawn increased attention towards millimeter-wave (mmWave) communication in fifth-generation (5G) and beyond wireless systems, as a life line for the bandwidth limitations. There are some challenges associated with switching to mmWave communications, especially the severe path loss caused by its short wavelength. In order to cope with it, high-gain directional beams are required which can be achieved through the use of large scale antenna arrays with beamforming technology. Hence, beamforming is one of the key enabling techniques for 5G and beyond for reliable high data-rate transmission [4], [8]. In phased array antennas, the desired beam radiation pattern is produced by controlling the excitation phase and amplitude of each antenna element [7], [2]. To focus the signal in a specific direction, multiple close proximity antennas broadcast the same signal at slightly different phases. In a linear antenna array, where d is the spacing between two neighboring antenna elements and λ denotes the signal wavelength, in order to steer the beam along θ_0, the corresponded phase shift $\Delta\phi = \frac{2\pi}{\lambda}d(sin\theta - sin\theta_0)$ need to be applied, as the constructive and destructive interference of the overlapping waves create the directional beam. The beam pattern is extremely sensitive to deviations in the elements' phase and amplitude [6]. Generally, a phase error can have a significant impact on the beam direction, whereas an amplitude error affects the beam side lobe levels and peak gain but not its direction. Fig. 1 shows the effect of random phase and amplitude error on to the radiation pattern. As a result of manufacturing failures, temperature shifts, and hardware degradation, the referred errors are common in nowadays systems and may have a critic effect on the radiation pattern and the overall system performance. Therefore, it is essential to calibrate the excitation errors of all the antenna elements so that the desired beamform is achieved.

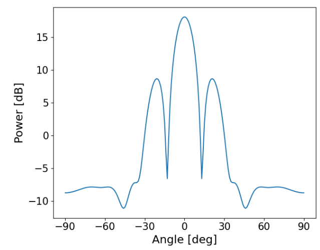
Preprint. Under review.



(a)



(b)



(c)

Figure 1: Effect of phase error and amplitude error on the antenna radiation pattern. (a) The desired radiation pattern. (b) The desired radiation pattern with randomly added phase error to each element. (c) The desired radiation pattern with randomly added amplitude error to each element.

2 Related Work

The calibration of antenna arrays has been researched for over two decades. The conventional calibration methods can be divided into two main categories, namely, the amplitude-based approach and the complex-signal based approach [10]. In the former, the calibration is based on the received power measurements (i.e., amplitude only), whereas in the latter, the received complex-signal measurements are used in the calibration process (i.e., both amplitude and phase). Two of the widely used conventional methods for phased array calibration are the rotating element electric field vector (REV) [6] and the multielement phase toggle (MEP) [1]. The REV is an amplitude-based method where the amplitude variation of the array is measured, while the phase of one of its elements changes from 0 to 2π, and then numerically processed for determining the corresponding element's relative amplitude and phase. The MEP is a complex-signal based method that calibrates the phases deviation using the inverse fast Fourier transform (IFFT). It can calibrate the phase error more effectively but less accurately compared to REV. The main drawbacks of these conventional calibration methods are the fact that they require a large amount of measurements (at least twice the number of antenna elements), and repetitive numerical calculations which make the entire calibration process long.

Recently, machine learning (ML) techniques, such as the use of neural networks (NNs), has been attempted to perform antenna calibration. They are designed to overcome the previously mentioned drawbacks and to enhance the calibration performance. For instance, in [3] the authors proposed a fully connected NN model to estimate the excitation phase error in response to the amplitude (power) of a radiation pattern input. They trained a NN model as an inverse function to the radiation field pattern of a linear array.

Considering the fact that the following method estimates the phase error using a simple NN model, a new method which calibrates the phase based on a more advanced architecture is suggested to try and increase the calibration performance. Furthermore, the above work lack proof of concept regarding the generalization performance of their model. All the results are simulated and tested on a specific kind of antenna array, so an evaluation of the model's sensitivity in term of changes in the antenna characteristics is required.

3 Method

First of all, a brief overview of the synthesis process for the creation of the dataset is described. Then, a thorough explanation of the NN model of [3] is introduced, because in this project, first I based a model on this work and treat it as the "vanilla" method, where attempts to reproduce their results, analyze their model generalization capabilities, and improve their model architecture are made. Finally, the new transformer-based method is presented.

3.1 The Synthesis of Antenna Array Radiation Pattern

A radiation field pattern produced from a linear antenna array can be described as follows [5]:

$$E_\delta(\theta) = \sum_{n=1}^{N} g_n(\theta) A_n exp \left\{ j \left[\frac{2\pi}{\lambda} (n-1)\, d\, sin\,(\theta) - \delta_n \right] \right\} \tag{1}$$

where $\delta = [\delta_1, ..., \delta_N]$ is the initial phase vector error, θ is the azimuth observation angle, N is the number of antenna elements, λ is the wavelength, d is the spacing between antenna elements, $g_n(\theta)$, A_n, and δ_n are the radiation gain pattern (directivity function), normalized amplitude, and the initial phase error of the nth antenna element. Based on (1), a large number of radiation patterns could be assembled to create the dataset. We can say that $g_n(\theta)$ represents how each antenna element radiation pattern respond to the observation angle θ in term of gain. It is common to assume that $g_n(\theta)$ is the same for all antenna elements, i.e., $g_n(\theta) \approx g(\theta), \forall n \in [N]$. Moreover, for isotropic array elements which have a radiation gain pattern that is independent of θ, $g(\theta) = 1$. In a more general case where the elements are not isotropic, most elements' radiation gain pattern favor broadside scan ($\theta_0 = 0°$) and depend on θ such that they taper over the scan angle. Therefore, a typically antenna radiation gain pattern is of the form $g(\theta) \propto cos^k(\theta)$.

3.2 "Vanilla" Method

In [3], 1,250,000 radiation patterns were created for a range of $-90° \leq \theta \leq 90°$ in $1°$ step with constant amplitude and uniformly distributed excitation phase coefficients by setting $-180° \leq \delta_n \leq 180°$ for $n = 2, ..., N$ where $\delta_1 = 0°$. $A_n = 1$ is assumed for all antenna elements, the linear antenna array consists of $N = 8$ elements which operates in 28 GHz and are evenly spaced $d = 6.0 \times 10^{-3}\,[m]$. The radiation patterns were transformed to their power form $P(\delta, \theta) = |E(\delta, \theta)|^2$ and then were normalized to range from 0 to 1. 96% of the dataset were considered for the training phase, and the remaining 4%, consisting of $50,000$ patterns, were used for validation. Hence, a real valued input matrix with the size of $1,200,000 \times 181$, and a training label matrix of the randomly selected phase error values of size $1,200,000 \times 7$ were used to train the NN.

The NN contains eight layers with six hidden layers. The input layer consist of 181 nodes, every hidden layer has 900 nodes, and the output layer contain 7 nodes. The activation function used in the output layer is the "sigmoid" function, whereas the rest of the layers use the rectified linear unit (ReLU) function. As a loss function, the mean square error (MSE) was used, the optimizer was "Adam", and a batch size of $16,384$ was generated per step for a total of 1000 epochs in training.

First, the NN would be trained using the above dataset, with the following assumption $g(\theta) = 1$, to reproduce [3] results. Afterwords, the model would be evaluated on a test set which represent a different antenna array, e.g., $g(\theta) = cos(\theta)$, for analyzing the model sensitivity in terms of the elements' radiation gain pattern.

3.3 Transformer-based Method

A Transformer-based model was proposed due to the great successes that this model achieved in many different tasks. Inspired by the use of patches as tokens on image datasets, I decided to divide the radiation pattern input into several sub-patterns and fed them as tokens to a Transformer encoder. In that way, I adapted my problem's input to the Transformer scenario. Moreover, since the 'vanilla' model uses a simple MLP architecture, one can assume that a complex Transformer-based architecture should outperform it.

The proposed method dataset creation is based on the one presented in the 'vanilla' method with three differences, 1) 92% of the dataset were used for training, and the rest 8% were divided equally between validation and test sets. 2) a batch size of 2048 was used. 3) each radiation power input vector $P(\delta, \theta)$ would be treated as one which consists of $m \in \mathbb{N}$ tokens (sub-patterns), to create an input which is represented by a sequence of tokens: $P(\delta, \theta) \to \{p_1, ..., p_m\}$ where $P(\delta, \theta) \in \mathbb{R}^{180}, p_i \in \mathbb{R}^{\lfloor \frac{180}{m} \rfloor}$, $\forall i \in [m-1]$, and the last element of the original radiation pattern was omitted for an easy division. Then, the new sequential dataset could be used as an input to a Transformer encoder [9], where for each sequence of input tokens, all tokens' outputs would be aggregated (mean function) to a combined output vector which is similar in size to the input tokens.

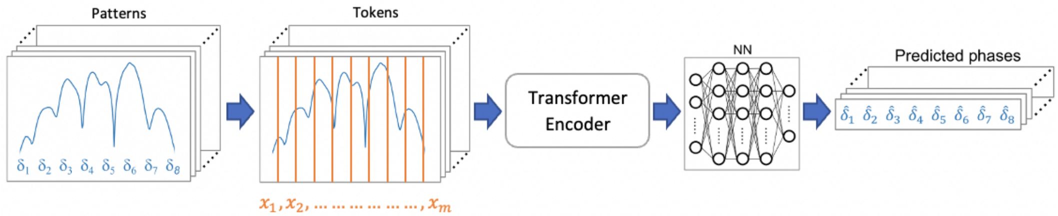


Figure 2: Schematic architecture diagram of the suggested method.

Finally, the phase error estimation vector would be achieved using a fully connected layer to the desired output size. Fig. 2 depict the suggested method architecture. Let us denote $l = \lfloor 180/m \rfloor$, then overall, a real valued input tensor of size $1,200,000 \times m \times l$, and a training label matrix of the randomly selected phase error values of size $1,200,000 \times 7$ will be used to train the model.

4 Experiments

4.1 'Vanilla' model's results reproduction

The topic of ML bsaed methods for antenna calibration, is quiet new, hence, there are not many published papers. And from the ones who are published, they usually do not offer their code. Therefore, my efforts to try to reproduce the reported results of [3] are challenged. First of all, there are some unknowns variable of their NN hyperparameters such as the learning rate, and it is unclear if each hidden layer consists of 900 nodes or if all of them together. Moreover, they reported a learning duration of 3.5h, while my learning duration, which occurred on a more powerful GPU, took more then 13h. I must state, that I verified my implementation using the MNIST dataset, with similar eight layers NN, and batch size. I concluded that my code works well and that it is impossible to train this NN in 1000 epochs for a dataset of size 1,200,000 in 3.5h. Due to those factors, and to the main fact that I wasn't able to reproduce their results, I am not certain that their reported resulted are valid, because they might be biased. The authors reported they loss records for the training and validation are 1.76×10^{-4} and 8.041×10^{-3} respectively. Compared to their reported results, I was able to reach at most, loss values of 3.5×10^{3} and 4.4×10^{-2} respectively.

My efforts to try and reproduce their results begin with trying several learning rates from the range: $10^{-3} - 10^{-5}$. My first observations was that the model learn well over the training data but have a severe overfitting for the validation data as illustrated in figure 3. To try and overcome this, I used two techniques: weight decaying and dropout layers. For each one of them alone and combined, I tried several values but none of them helped to improve the overfitting. I concluded that there must be an inner factor which prevent the model to generalize the good training performance to data outside of the training data. My first thought was that maybe the dataset creation process made it to be sampled from different distributions, such that there isn't a probabilistic connection between two unkike data points. In order to contradict this assumptions, I preformed a training phase over both the training set and validation set, and kept for the validation phase the same validation data points. Since both the training phase and validation phase achieved the same learning curve and loss results as for the original training phase, I concluded that this is not the case. Next, I thought that it might be caused from the chosen loss function, and I began looking into it.

4.2 Loss Function

The authors in [3] used the MSE loss function for training evaluation. From my perspective, this choice of loss function is not optimal and even a major source for model overfitting. The reason for that is the ambiguity in the angle system, where $180° = -180°$. Since $-180° \leq \delta_n \leq 180°$, MSE loss function worst case scenario, which would be penalized for the most, is when $\delta_n = 180°$ and $\hat{\delta}_n = -180°$ where $\hat{\delta}_n$ is the estimated phase error of the nth antenna. But this is actually the best case scenario, a perfect estimation.

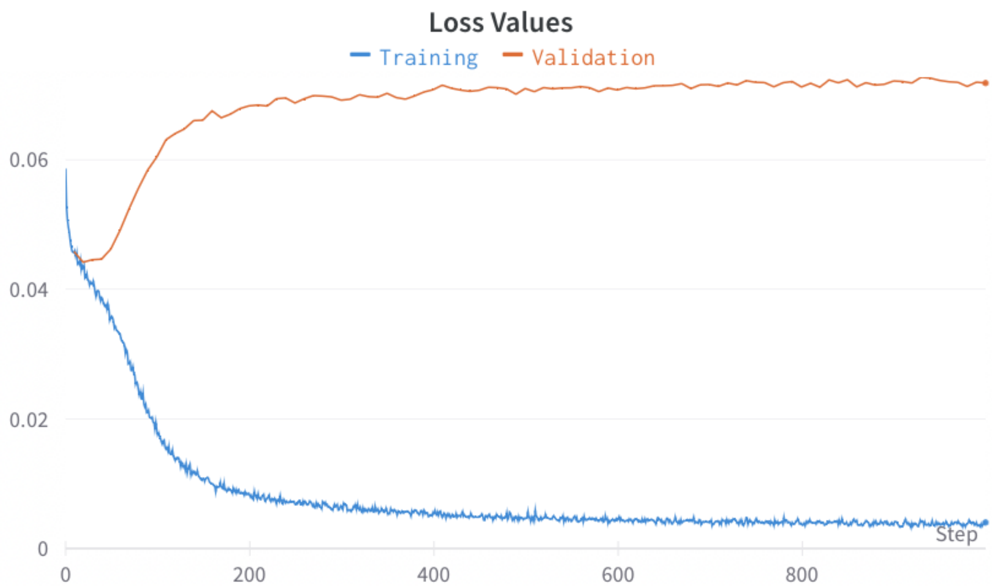


Figure 3: Loss values recorded for the 'vanilla' model using a learning rate of 10^{-3}.

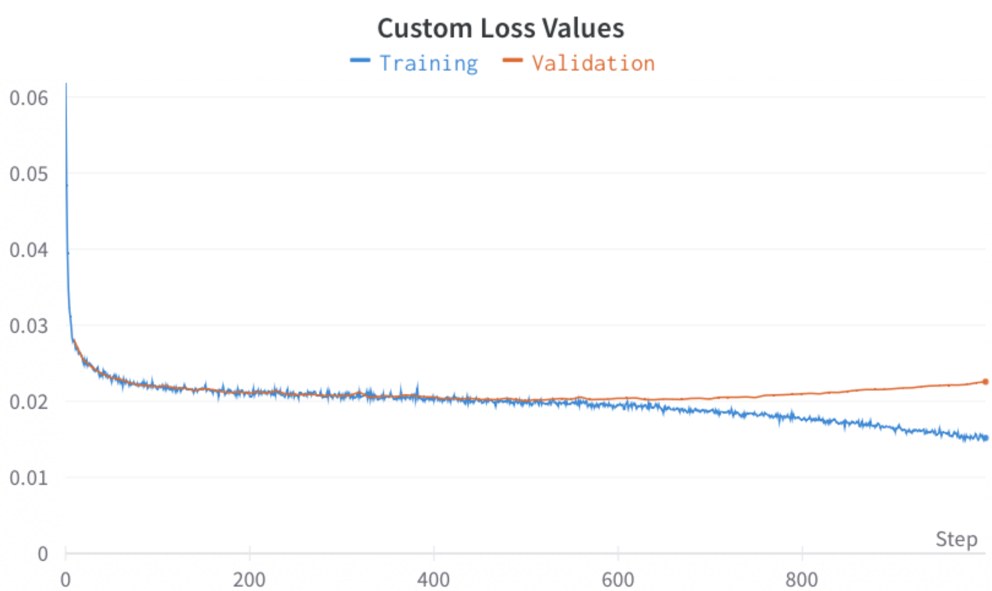


Figure 4: Loss values recorded for the 'vanilla' model using a learning rate of 10^{-4} and the custom loss function.

Therefore, I believe that a better choice for a loss function is a custom one that treats this angles ambiguity:

$$l_{mirrorMSE}\left(\delta,\hat{\delta}\right)=\frac{1}{7}\sum_{n=2}^{N}min\left(|\delta_n-\hat{\delta}_n|,\ 360^{\circ}-|\delta_n-\hat{\delta}_n|\right)^2.$$

Using this custom loss function, and after experimenting several learning rates, I was able to obtain a promising learning curves which have less overfitting, as depict in Figure 4.

Next, I tried to enhance the learning phase by adding Batch Normalization layers to the model's architecture. After experimenting the improved architecture for several learning rates, I observed

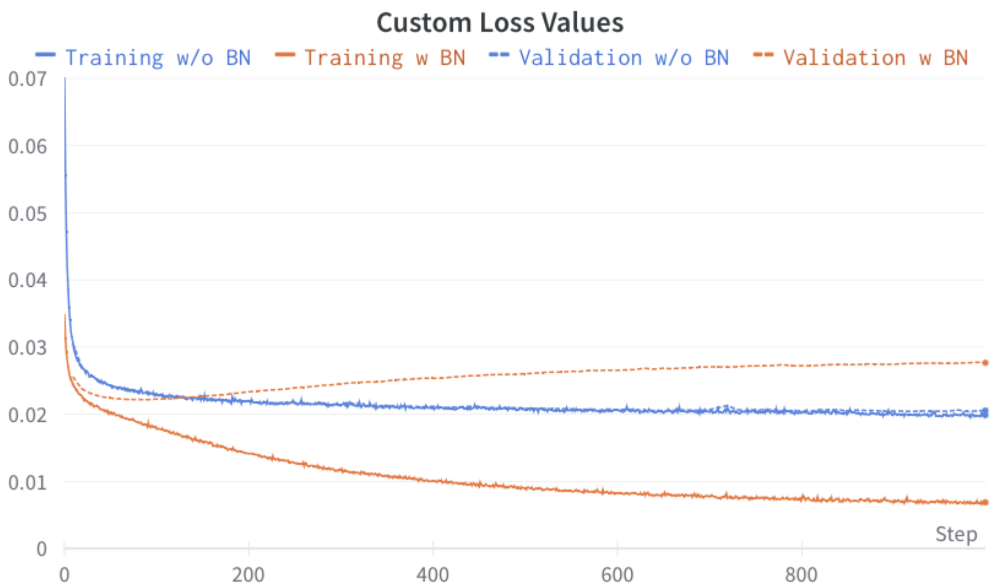


Figure 5: Loss values recorded for the 'vanilla' model w and w/o Batch Normalization layers, using a learning rate of 5×10^{-5} and the custom loss function.

that the training loss is improving yet the validation loss overfit quickly and does not reach the same values. Hence, further experiments should be done in future to improve the overfitting. Figure 5 illustrates a comparison of the custom loss values between the 'vanilla' model with and without the Batch Normalization layers for a learning rate of 5×10^{-5}.

4.3 Transformer-based Model

The hyperparameters who needs to be investigated in this model are the learning rate, number of input tokens m, the aggregation function, number of heads in the Multi-head attention, and number of sub-encoder layers in the encoder. Due to lack of time, I was able to experiment the model using the mean function as aggregation, $m = 6$ tokens (i.e., each token is of size 30), default heads and sub-encoder number (i.e., 6 each), and several learning rates. The current training comparison between the 'vanilla' method and the Transformer-based method is presented in Figure 6. For now, the 'vanilla' methods have better learning results which will implicate better accuracy.

4.4 Models' Performance Evaluation

The indicator used to evaluate the models' performance and to compare them is the root mean squared errors (RMSEs) of the estimated phases:

$$\delta_{RMSE} = \sqrt{\frac{1}{7}\sum_{n=2}^{8} min\left(|\delta_n - \hat{\delta}_n|,\ 360° - |\delta_n - \hat{\delta}_n|\right)^2}$$

where $\hat{\delta}_n$ is the estimated phase error of the nth antenna.

Currently, I achieve an averaged RMSE over the test set of $\bar{\delta}_{RMSE} = 48.85°$ and $\bar{\delta}_{RMSE} = 55.38°$ for the 'vanilla' model and Transformer-based model respectively. These results are not yet comparable to the results reported in [3] ($\bar{\delta}_{RMSE} = 7°$) and to the accuracy of the conventional methods which are similar.

4.5 Models' Generalization Performance

We would like to evaluate the models on a test set which represent an antenna array with different characteristics from the one the model have trained on, e.g., different directivity function $g(\theta)$,

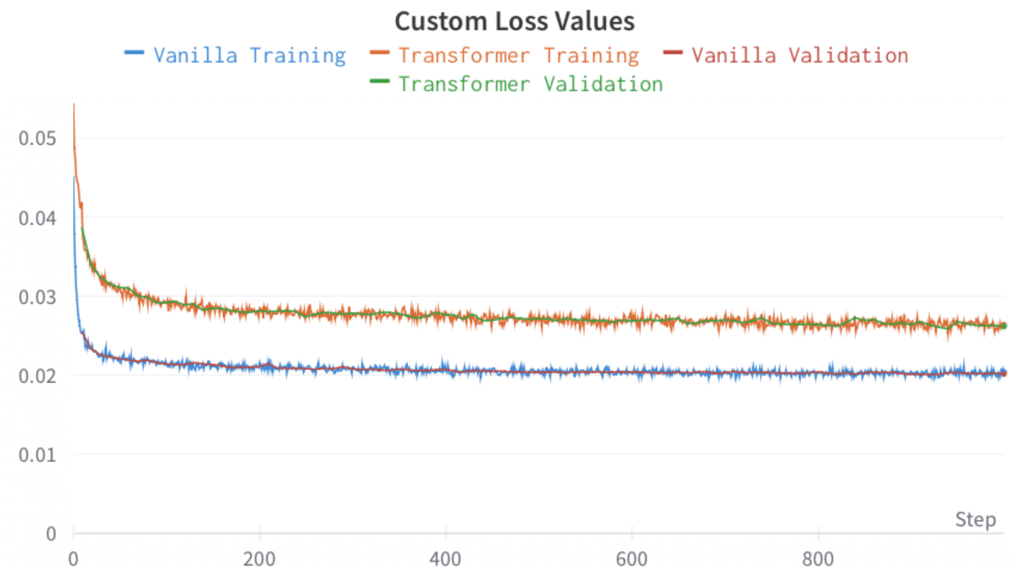


Figure 6: Loss values recorded for the 'vanilla' model w/o Batch Normalization layers and the Transformer-based model, using a learning rate of 1×10^{-4} and the custom loss function.

elements spacing d, and transmission frequency f. In that way, we can analyze a model generalization capabilities to different kinds of antenna arrays. A total of three experiments were conducted where only one of the following antenna parameter was changed while the rest remain the same: $g(\theta) = cos(\theta)$ (instead of 1), $d = 8.0 \times 10^{-3}$ $[m]$ (instead of 6.0×10^{-3} $[m]$), $f = 33$ GHz (instead of 28 GHz).

The experiments results achieved by 'vanilla' model and Transformer-based model are presented in Table 1.

Model	δ_{RMSE} Regular	δ_{RMSE} Changed $g(\theta)$	δ_{RMSE} Changed d	δ_{RMSE} Changed f
'Vanilla'	48.85°	62.52°	100.72°	96.29°
Tansformer	55.38°	60.61°	103.18°	99.98°

Table 1: Average RMSE results for changes in the antenna parameters.

Clearly, both models have trouble preforming well on data generated from different kinds of antenna. Yet, it seems that the Transformer-based model have the potential to have better generalization accuracy. These results are not so surprising due to the input (radiation pattern) sensitivity to these big changes. This sensitivity is illustrated in Figure 7. Figure 7b is closest in shape to Figure 7a in most angles, and that explains why their results are more alike.

5 Conclusion and Future Work

In this project, motivated by the fact that deep learning calibration methods, for millimeter-wave phased array antennas, have the potential to overcome many drawbacks of conventional calibration methods; I began to explore, analyze and propose new deep learning calibration methods. First, I tried to reproduce the results of a recently published paper, which uses an MLP architecture for the problem. After a deep analysis of their work, and since no code was published, I believe that their reported accuracy results and learning time are problematic. Nevertheless, my 'vanilla' methods is based on their work with replacing the MSE loss function to a custom one. Then, a new Transformer-based model is proposed. Both models' performance are evaluated and compared using the RMSE indicator. Currently the 'vanilla' method outperform the Transformer-based method, but many more

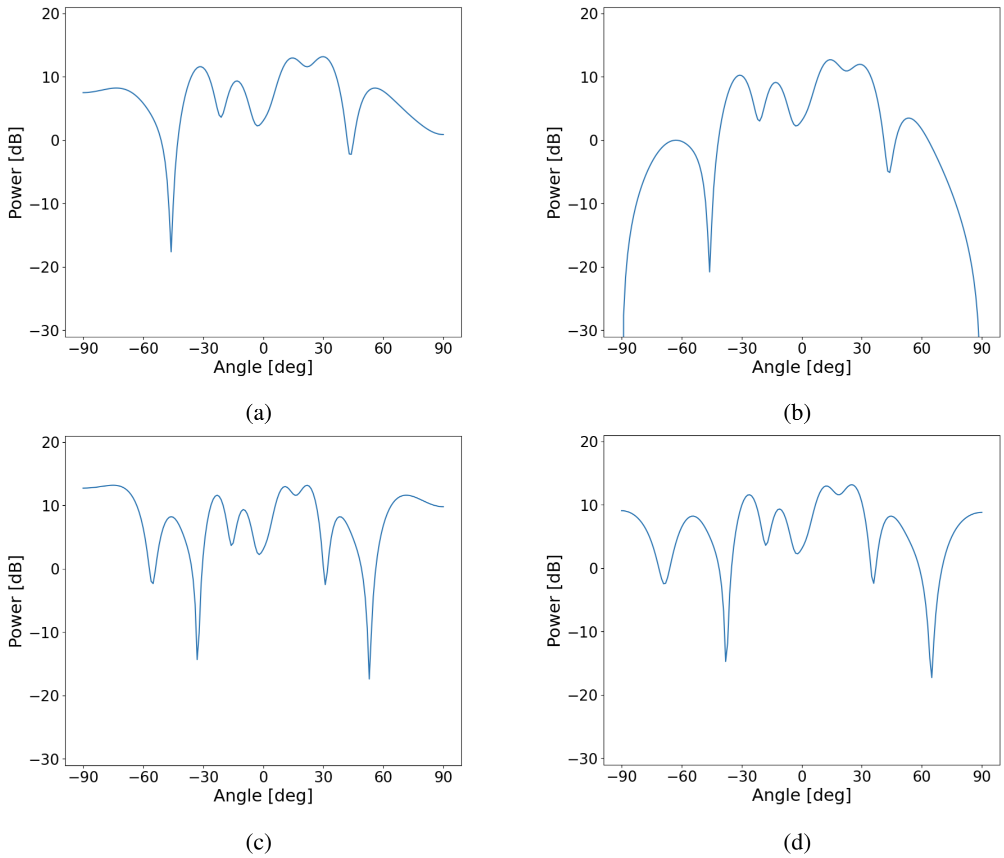


Figure 7: Effect of change in antenna parameters on the antenna radiation pattern with phase error. (a) Trained parameters. (b) Change of directivity function. (c) Change of spacing between antenna elements. (d) Change of transmission frequency.

experiments need to be done for the Transformer-based method. Lastly, an analysis of the models' generalization performance on data generated from different antenna characteristics is conducted. Both models have difficulties maintaining good accuracy because the input patterns differ a lot from the one which used for training. To conclude, the current achieved results are not yet comparable to accuracy achieved by the conventional methods, hence, does not yet motivate the move to ML-based methods. However, I am certain that with some more future research and efforts in improving the learning phase, and tweaking the hyperparameters, this would be accomplished.

The future work directions are as follows.

- Overcoming the overfitting which occurs in the 'vanilla' method when using batch normalization layers.
- Enhancing the Transformer-based method learning by further experimenting and tweaking its hyperparameters.
- Enhancing the Transformer-based method performance by implementing an MLP after the Transformer encoder aggregated vector.
- Increasing the input data resolution for better learning.
- Using the radiation pattern phase information combined with the amplitude information for better learning.
- Analyzing how the reduce of the random phase error range would affect the performance. $-180^\circ \leq \delta_n \leq 180^\circ$ might be to large for what occurs in nowadays systems.
- Proposing models which could perform similar on different kinds of antennas, instead of training a model for each antenna. I believe that a smart choice of an input should be used.

References

[1] Grant A Hampson and A Bert Smolders. A fast and accurate scheme for calibration of active phased-array antennas. In *IEEE Antennas and Propagation Society International Symposium. 1999 Digest. Held in conjunction with: USNC/URSI National Radio Science Meeting (Cat. No. 99CH37010)*, volume 2, pages 1040–1043. IEEE, 1999.

[2] Shuangfeng Han, I Chih-Lin, Zhikun Xu, and Corbett Rowell. Large-scale antenna systems with hybrid analog and digital beamforming for millimeter wave 5g. *IEEE Communications Magazine*, 53(1):186–194, 2015.

[3] Tetsuya Iye, Pieter van Wyk, Takahiro Matsumoto, Yuki Susukida, Shohei Takaya, and Yoshimi Fujii. Neural network-based phase estimation for antenna array using radiation power pattern. *IEEE Antennas and Wireless Propagation Letters*, 21(7):1348–1352, 2022.

[4] Mohsen Khalily, Rahim Tafazolli, Pei Xiao, and Ahmed A Kishk. Broadband mm-wave microstrip array antenna with improved radiation characteristics for different 5g applications. *IEEE Transactions on Antennas and Propagation*, 66(9):4641–4647, 2018.

[5] Robert J Mailloux. *Phased array antenna handbook*. Artech house, 2017.

[6] Seiji Mano and Takashi Katagi. A method for measuring amplitude and phase of each radiating element of a phased array antenna. *Electronics and Communications in Japan (Part I: Communications)*, 65(5):58–64, 1982.

[7] Takahide Murakami, Yoji Kishi, Koji Ishibashi, Keisuke Kasai, Hiroyuki Shinbo, Morihiko Tamai, Kensuke Tsuda, Masataka Nakazawa, Yu Tsukamoto, Hiroyuki Yokoyama, et al. Research project to realize various high-reliability communications in advanced 5g network. In *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–8. IEEE, 2020.

[8] Wonil Roh, Ji-Yun Seol, Jeongho Park, Byunghwan Lee, Jaekon Lee, Yungsoo Kim, Jaeweon Cho, Kyungwhoon Cheun, and Farshid Aryanfar. Millimeter-wave beamforming as an enabling technology for 5g cellular communications: Theoretical feasibility and prototype results. *IEEE communications magazine*, 52(2):106–113, 2014.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[10] Xizixiang Wei, Yi Jiang, Qingwen Liu, and Xin Wang. Calibration of phase shifter network for hybrid beamforming in mmwave massive mimo systems. *IEEE Transactions on Signal Processing*, 68:2302–2315, 2020.