# Optimization of Antenna Arrays using Deep Learning

**David Lin Yi Lu**
UCI

## Abstract

The deployment of antenna arrays in wireless communication systems has witnessed significant advancements in recent years. A well-designed antenna array can enhance the power radiated in the desired direction, achieving a higher gain than a single antenna. However, if the design is flawed, the antennas can interfere with each other. The goal of the project is to utilize deep learning to improve a given pattern.

## 1 Introduction

Optimizing antenna arrays to achieve desired patterns is a formidable challenge in the realm of wireless communication. Particularly, the application of deep learning techniques to tackle this problem remains relatively unexplored. This research paper aims to delve into this uncharted territory by investigating the potential of deep learning in addressing the complexities of antenna array optimization. The study focuses on two crucial constraints: the need to maintain a constant number of antennas and the requirement for a minimum spacing between them. By exploring these aspects, we aim to shed light on the untapped potential of deep learning for optimizing antenna arrays and contributing to advancements in the field of wireless communication.

Designing an antenna array with a specific pattern is difficult due to the intricate relationship between array geometry and radiation patterns. Additionally, the constraints of fixed antenna numbers and minimum spacing further complicate the optimization process.

To tackle this challenge, we propose a two-part approach. Firstly, we develop a deep learning predictive model to estimate the cost associated with different antenna array designs. This predictive model is trained using a dataset of known configurations and associated costs. Secondly, leveraging the insights gained from the predictive model, we employ optimization algorithms to improve the inputs and minimize the cost.

## 2 Dataset

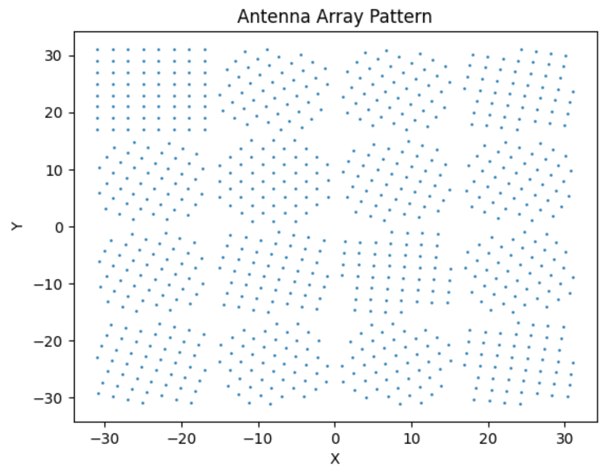
### 2.1 Exploring the dataset

The dataset utilized in this project was provided by Prof. Baldi and Prof. Boag. A total of 100,000 patterns were obtained, each consisting of a variable number of antennas ranging from 800 to 1024, where Y and Z range from -33 to 33. The dataset provides crucial information for each pattern, including the corresponding cost, which is a scalar value indicating the quality of the pattern. The cost is calculated by the function F, producing a low value if the pattern is good and a high value if the pattern has more interference. Additionally, each antenna within a pattern is characterized by its Y and Z coordinates, representing the spatial positioning.

The range of the cost values within the dataset spans from -0.07 to -40,000. The mean cost is approximately -10,000, with a standard deviation of around 9,000. These statistics indicate a considerable variation in the costs, highlighting the diverse quality of the antenna array patterns.

Preprint. Under review.

Figure 1: An example of an antenna array pattern




### 2.2 Scaling the data

To facilitate the optimization process and ensure effective utilization of the dataset, two scaling approaches were employed. The first approach, referred to as the YZ-C scale, involves scaling the Y and Z coordinates of the antennas as well as the cost.

In the YZC-scale, the mean and standard deviation of the Y and Z coordinates are calculated separately. The mean and standard deviation are calculated by the non-zero antenna coordinates. These values serve as scaling factors to normalize the Y and Z coordinates. By applying this scaling, the Y and Z coordinates are transformed to a standardized range, allowing for fair and consistent comparison across different antenna patterns.

Similarly, the cost is also scaled by utilizing its corresponding mean and standard deviation. This scaling procedure ensures the transformation of the cost into a normalized range, aligning it with the standardized Y and Z coordinates. Consequently, the scaled cost values can be interpreted as deviations from the average cost. This standardized and comparable metric greatly facilitates the optimization process. Additionally, in the C-scale approach, the cost undergoes the same scaling procedure, while the Y and Z coordinates remain in their original scale.

By employing these scaling approaches, the dataset is transformed into a more manageable and uniform representation, allowing for efficient analysis and optimization of the antenna array patterns. The scaled data provides a foundation for the subsequent stages of the research, facilitating the application of deep learning techniques and optimization algorithms to achieve optimal antenna array configurations.

## 3 Methods

### 3.1 Predicting the cost

The initial stage of this research project focuses on approximating the cost function F, which plays a pivotal role in evaluating the performance of antenna array patterns. Our primary objective is to construct a robust model that can accurately predict the cost associated with various antenna array designs. To accomplish this goal, we explore the effectiveness of two distinct models: a feedforward neural network (FNN) and a set transformer.

For training purposes, we employ the Adam optimizer with a learning rate of 1e-3. The mean squared error (MSE) is utilized as the loss function, enabling us to measure the disparity between the predicted costs and the actual costs.

### 3.1.1 Feedforward Neural Network (FNN)

For the FNN, we can use both YZC-scale and C-scale without much difference. The model takes in a 1-D vector arranged as $Y_1, Y_2, ..., Y_{1024}, Z_1, ..., Z_{1024}$, followed by two hidden layers with 8 neurons and ReLU activation to produce the cost.

### 3.1.2 Set Transformer

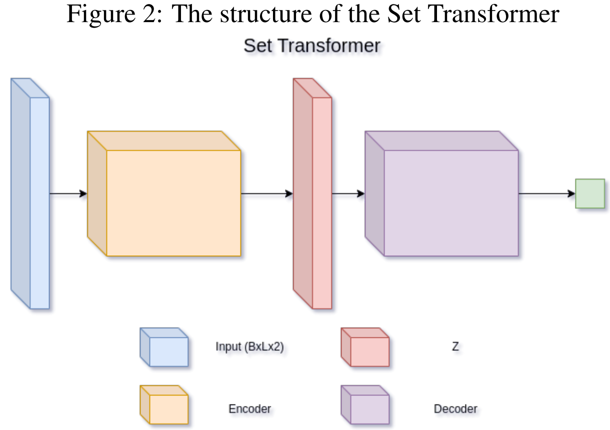
In our approach, we adopt the YZC-scale for the Set Transformer model, which takes a 2-D vector represented as $[(Y_1, Z_1), (Y_2, Z_2), ..., (Y_N, Z_N)]$. The Set Transformer architecture consists of an encoder and a decoder, each serving distinct functions.

The design of the Set Transformer is inspired by the DeepSet model, which also utilizes an encoder and decoder structure. In DeepSet, the encoder primarily encodes each element independently, followed by a set pooling function, such as sum or mean, applied by the decoder to aggregate information across dimensions.

In our case, the encoder of the Set Transformer plays a crucial role in capturing and encoding interactions between antennas within the set. To accomplish this, we employ Set-Attention-Blocks (SAB) within the encoder. These SABs utilize self-attention to encode pairwise interactions, allowing the encoder to capture not only pairwise but also higher-order interactions within the set by stacking multiple SABs.

In contrast to DeepSet, which typically employs a set pooling function after encoding each element independently, our Set Transformer decoder employs a Pooling-by-Multihead-Attention (PMA) mechanism. This approach enables pooling across dimensions without explicitly relying on a predefined set pooling function. By leveraging the PMA mechanism, the decoder can effectively aggregate information across the encoded dimensions, facilitating effective information integration and output generation.

Figure 2: The structure of the Set Transformer




Compared to DeepSet, the Set Transformer takes advantage of self-attention for capturing element interactions and learns the pooling function within the decoder. This approach allows the model to effectively encode the complex relationships among the elements of the antenna array set, enhancing its ability to predict the associated cost accurately.

## 3.2 Optimizing the inputs

In this section, we utilize the FNN, represented as G, as an approximation of the cost function. The objective is to examine how the cost changes with variations in the input by calculating the gradient of G with respect to the inputs. By leveraging the gradient and employing a learning rate, we can employ gradient descent to optimize the input space.

During the optimization process, certain considerations are taken into account to ensure the validity and feasibility of the antenna array designs. First, we apply a padding mask to maintain a consistent number of antennas throughout the optimization. This step helps maintain the structural integrity of the array.

Additionally, a constraint check is performed to enforce a minimum distance between antennas. This constraint aims to prevent antennas from being positioned too close to each other, which may lead to interference or degradation in performance. The minimum distance is set to 0.5 times the Euclidean distance between the antennas. If this constraint is violated during the optimization, the gradient step is not accepted, but the optimization process continues to explore other possibilities until a set number of iterations.

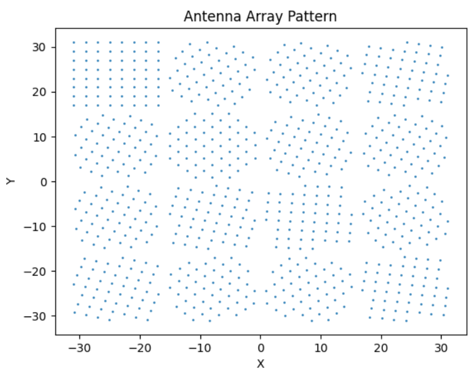



Figure 3: Given Pattern

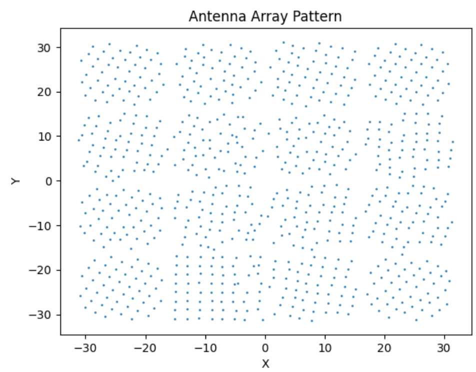



Figure 4: Optimized Pattern

## 4 Results and Analysis

In the cost prediction phase, we employed two models using YZC-scale: the FNN and the Set Transformer. Given the limited exploration of this problem, we lack established benchmarks for model performance comparison. As a preliminary benchmark, we predicted the cost $\mu$ of the training dataset. The table below displays the results, indicating that both models exhibited promising performance, with the FNN demonstrating slightly superior results. Regarding the optimization of a given pattern, we are currently awaiting confirmation from our collaborators.

| Results | |
|---|---|
| Models | MSE |
| $\mu$ | 1 |
| FNN | 0.01 |
| Set Transformer | 0.06 |

## 5 Conclusion and Discussion

In conclusion, both the FNN and the Set Transformer models demonstrated commendable performance in predicting the cost associated with antenna array patterns. While the Set Transformer offers increased complexity and the ability to capture interactions between elements using self-attention, it was the FNN that excelled in approximating the true cost function. This outcome highlights the effectiveness of the FNN architecture in this particular research context.

Moving forward, it would be valuable to explore alternative regularizers to satisfy the minimum distance constraint between antennas. The current optimization process imposes a constraint check on the minimum distance, but incorporating different regularizers could provide additional flexibility and ensure compliance with this constraint. By experimenting with various regularization techniques, such as penalty functions or custom constraints, we can further refine the optimization process and potentially improve the overall performance and robustness of the antenna array design. Exploring different regularizers offers an avenue for future research and holds the potential to enhance the practical applicability of the optimization approach.