DPD using Deep Learning

Contents

[DPD using Deep Learning 1](#_Toc111468436)

[1. Astract 3](#_Toc111468437)

[2. Introduction 3](#_Toc111468438)

[2.1 Motivation 3](#_Toc111468439)

[2.2 Objective 3](#_Toc111468440)

[2.3 contribution 3](#_Toc111468441)

[3. Literature review 3](#_Toc111468442)

[3.1 PA Model 3](#_Toc111468443)

[3.2 DPD 3](#_Toc111468444)

[3.3 MSE 3](#_Toc111468445)

[3.4 Neural Network 3](#_Toc111468446)

[4. Metodology 4](#_Toc111468447)

[5. Results 4](#_Toc111468448)

[5.1 PA Modelling 4](#_Toc111468449)

[5.2 DPD 4](#_Toc111468450)

[6. Discussion 4](#_Toc111468451)

[7. Conclusion 4](#_Toc111468452)

# Astract

בעתיד

# Introduction

## Motivation

Digital Pre Distortion (DPD) is one of the most effective techniques of PA linearization. In this technique, a digital non-linear block, known as a Predistorter, is used in front of the Analog to Digital (A/D) component. The role of the Predistorter is to distort the signal in a way that will be, in turn, compensated by the PA. Ideally, the total response of the Predistorter and PAs will be linear up to some saturation voltage.

## Objective

The aim of this project is to reduce am/am, am/pm distortions and OOB emission which caused by nonlinearity of power amplifiers. Using both classical and deep learning methods, we will create a digital pre distorter that preprocess a signal before entering a nonlinear power amplifier.

## contribution

בעתיד

# Literature review

## PA Model

Nonlinear effects – AM/AM, AM/PM, OOB emission

Power amplifiers (PAs), which are inherently nonlinear systems, are essential components in communication systems. Moreover, PAs also exhibit memory effects. This means that, the current output of the PA depends not only on the current input, but also on past input values, and makes the power amplifier a nonlinear system with memory. Different models, such as Wiener model and Hammerstein model, represent this kind of system as a two block model. The first block is a linear FIR filter that deals with linear memory effects, and the second block accounts for the nonlinear effects caused by the PA. A block diagram of such model is represented on figure 3.1.a.

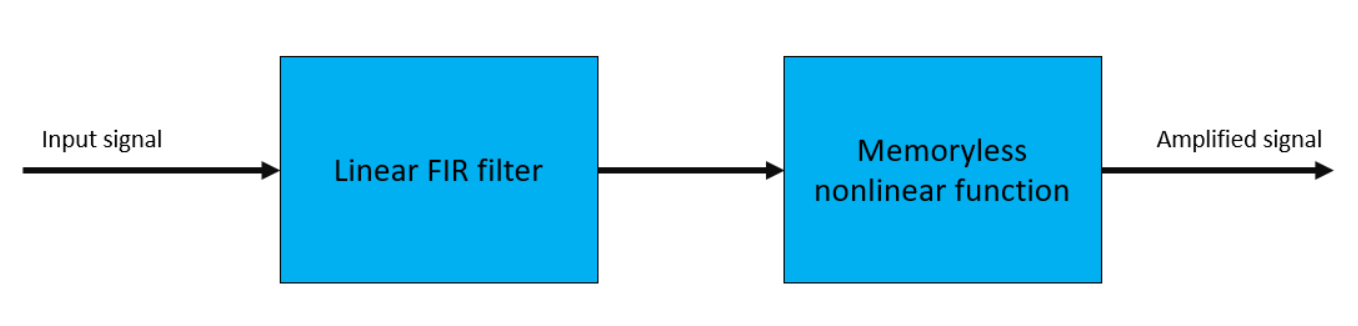


Figure 3.1.a – Block diagram of a two-block PA model

The nonlinearity causes in-band distortion and a spectral regrowth, which leads to interference and violations of the out-of-band emission requirements. The use of different transmission formats, such as wideband Code Division Multiple Access (CDMA) or Orthogonal Frequency Division Multiplexing (OFDM), which are known to have high peak to average power ratios, increases the risk of using voltages that are close to the PAs saturation point, as this will lead to a severe distortion, as mentioned above. For this reason, PA linearization methods have gained popularity and increasing interest in recent years.

The PA nonlinearity may be characterized in many ways. In this work, we will concentrate on three types of distortion of the output signal: in its amplitude (also referred to as Amplitude Modulation/ Amplitude Modulation (AM/AM) distortion), phase (also referred to as Amplitude Modulation/ Phase Modulation (AM/PM)) and out of band emission (OOB emission).

AM/AM is the relation between the amplitude of the input signal and the amplitude of the output signal, which ideally should be linear, but due to non-linear components in the PA, is usually nonlinear. This nonlinearity, has been accepted in our lab’s PA, and illustrated in figure-3.1.b, where the output amplitude is shown vs input signal amplitude.



Figure 3.1.b – illustration of AM/AM distortion caused by our lab’s nonlinear PA

AM/PM is the relation between the amplitude of the input signal and the phase difference between the input signal and the output signal. Ideally, the phase difference should be zero. It usually happens in the lower range of input signal amplitudes. This distortion, has been accepted in our lab’s PA, and illustrated in figure-3.1.c, where the phase difference between input and output is shown vs input signal amplitude. A zoomed in version of figure-3.1.c is shown in figure-3.1.d



Figure 3.1.c – illustration of AM/PM distortion caused by our lab’s nonlinear PA

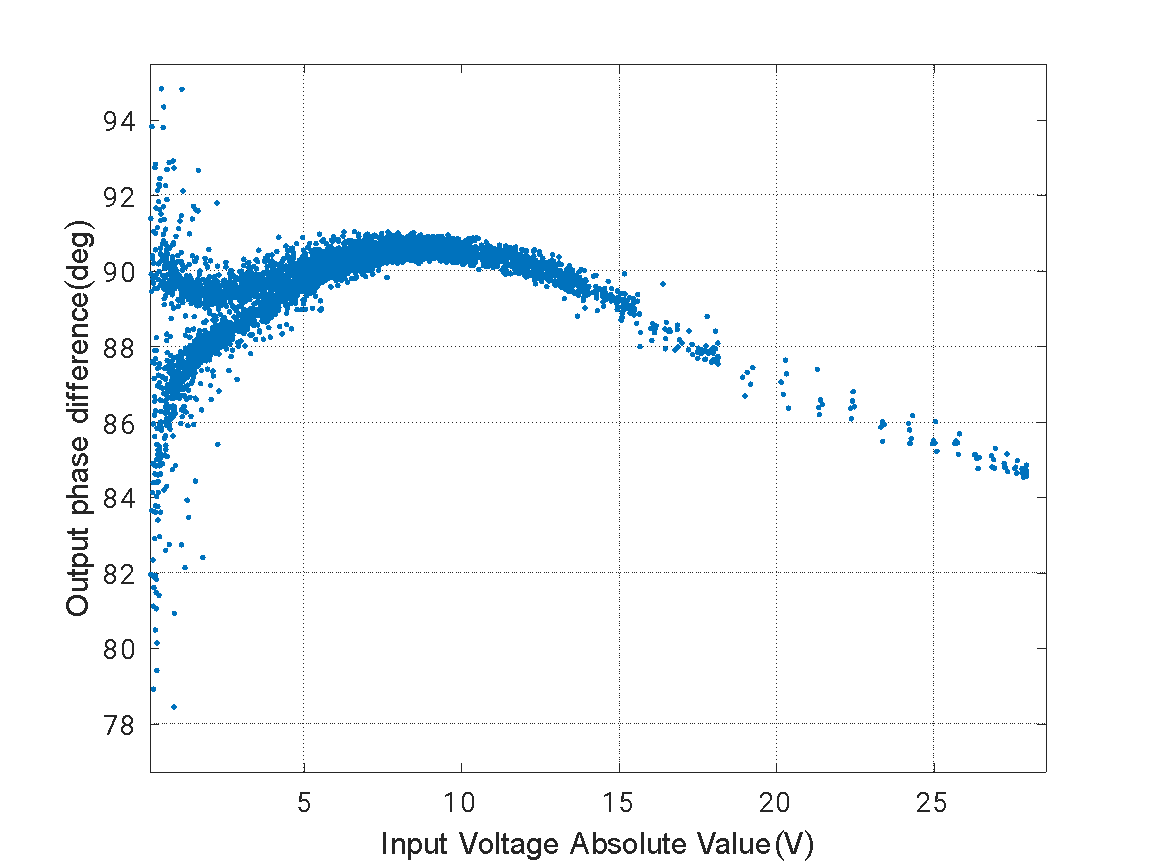


Figure 3.1.d – A zoomed in version of figure-3.1.b

OOB emission is emission on a frequency or frequencies immediately outside the necessary bandwidth which results from the nonlinearity of the PA. Due to its nonlinearity, it can create harmonics of the fundamental frequency of the input signal. For example, if a signal which has two fundamental frequencies, , is entered into a nonlinear PA, output signal will contain new frequency components, which will be in the form of , when n and m are integers. Out-of-band frequencies can be ignored by filters, however, in-band frequencies cannot. This distortion, has been accepted in our lab’s PA, and illustrated in figure-3.1.e.

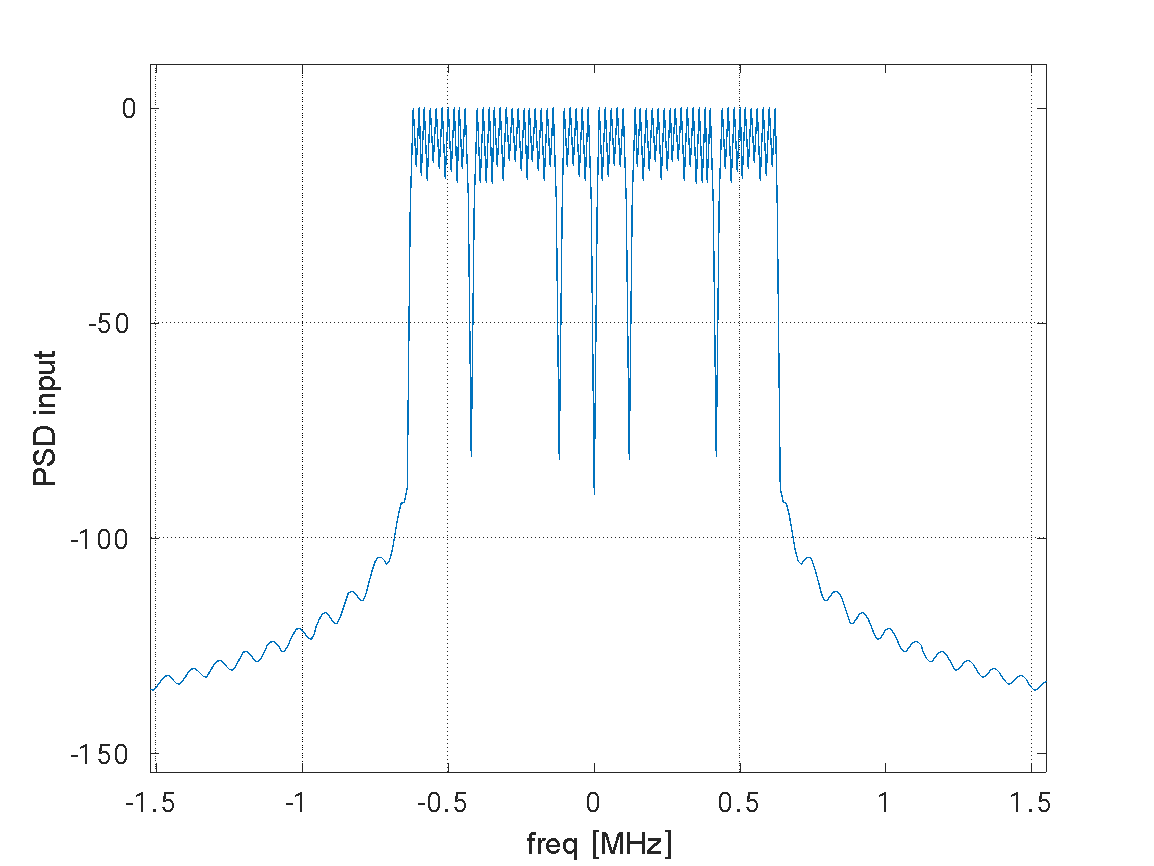
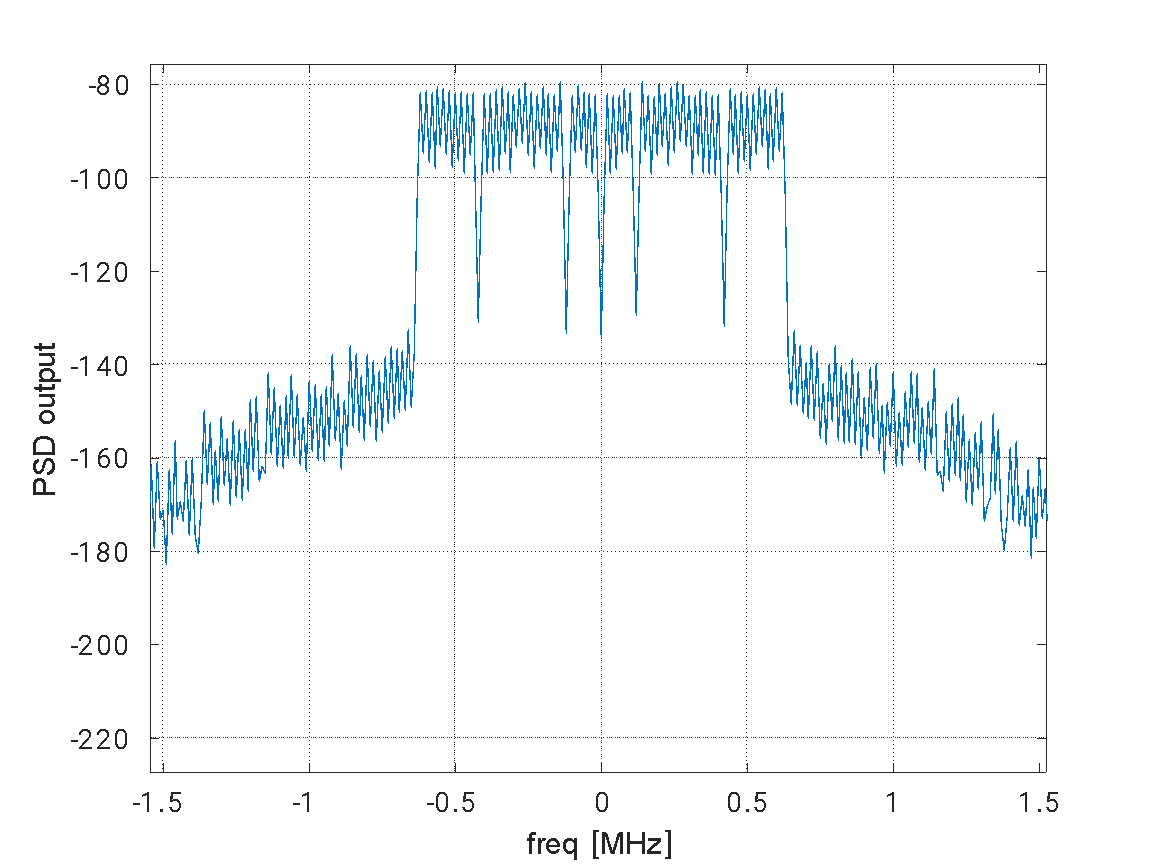
 

Figure 3.1.e – illustration of OOB emission caused by our lab’s nonlinear PA

Volterra series as a PA model (MP)

Volterra series are used in order to model systems that are both nonlinear and have memory.

As such, the Volterra series are often chosen to represent PAs response. The general form of the discrete Volterra series is given by:

where is the output sample, is the input sample, K is the order of non-linearity of the system, M is the order of memory of the system, and is a coefficient with set values as a function of k and

The extensive number of coefficients related with every combination of input sample and delayed input sample combinations within the bounds specified makes it possible to model large scale of nonlinear systems with memory effects. It is able to model systems with both large non-linearities and drastic memory effects, and by raising the nonlinear and memory orders, K and M, the model can become more accurate. However, the complexity in calculating the coefficients increases dramatically as either K or M is increased. As a result, many simplifications of the full Volterra model, with less coefficients, have been devised. One of these is the Memory Polynomial model, which is given by:

where is the output sample, is the input sample, K is the order of non-linearity of the system, M is the order of memory of the system, and is a coefficient with set values as a function of 𝑘 and 𝑚.

This model can be represented efficiently in matrix form as:

where 𝒚 is the outputs array of the MP model, is the coefficients array , and 𝑿 is an matrix containing the signal, delayed values of the signal, and their powers, which required for calculating the output. This means that the matrix can be created given only the input sample array and the order of non-linearity and memory order of the MP model to be used. The coefficient array contains the unknown coefficients that represents the power amplifier.

## DPD

After creating a realistic model of a HPA, it is also desired to create a model of an associated predistortion function that can be used to create an overall linearly behaving system. The model of the DPD should be an inverse of the model of the HPA, so it is clear that the DPD will also need to account for both non-linearities and memory effects. That means that the DPD can also be successfully modeled using the MP model, given by:

where is the DPD output signal array, is the delay matrix formed by the DPD input signal, and is an array containing all the MP model coefficients of the DPD. Order of nonlinearity and order of memory for the DPD should be found as well.

## MSE

PA modeling

As detailed before, we modeled PA output signal such that:

This model can be represented efficiently in matrix form as:

where 𝒚 is the outputs array of the MP model, is the coefficients array , and 𝑿 is an matrix containing the signal, delayed values of the signal, and their powers, which required for calculating the output.

In order to estimate this coefficients, one requires to have measurements of the output of the PA excited by a well-known input signal. The input signal should be chosen, such that it ranges over the whole bandwidth and input amplitudes that the modeled amplifier should handle. We put the measurements of the output signal that we got from our lab’s nonlinear PA in a vector . By using this vector, the coefficients vector can be found using two different methods.

The first method is the well-known method of minimizing the least squares error between the calculated output and measured output. The coefficients vector is given by:

Where 𝒚 is the measured output signal, 𝑿 is the matrix detailed above, which formed by the input signal and the superscript is the conjugate transpose operator.

The second method is SGD. We used this algorithm in order to find the parameters that minimize the least squares error between the calculated output and measured output.

We set the step size using the following formula:

While was set to a very small number - , and alpha was set to . According to this definition, step size is a matrix. We did it because the gradient has radical differences between its components, and in order to handle this with SGD, the above definition for step size is required. The full algorithm is given as:

In order to find the optimal coefficients for the MP model, it is essential to choose values of m and k which model PA optimally. In order to do that, one should find values of m and k that minimize the error between calculated output generated by coefficients calculated with those specific m and k, and measured output. This optimization was doen for our lab’s PA, and results are shown on figure-3.3.a. The error is minimized at k=9, m=2.

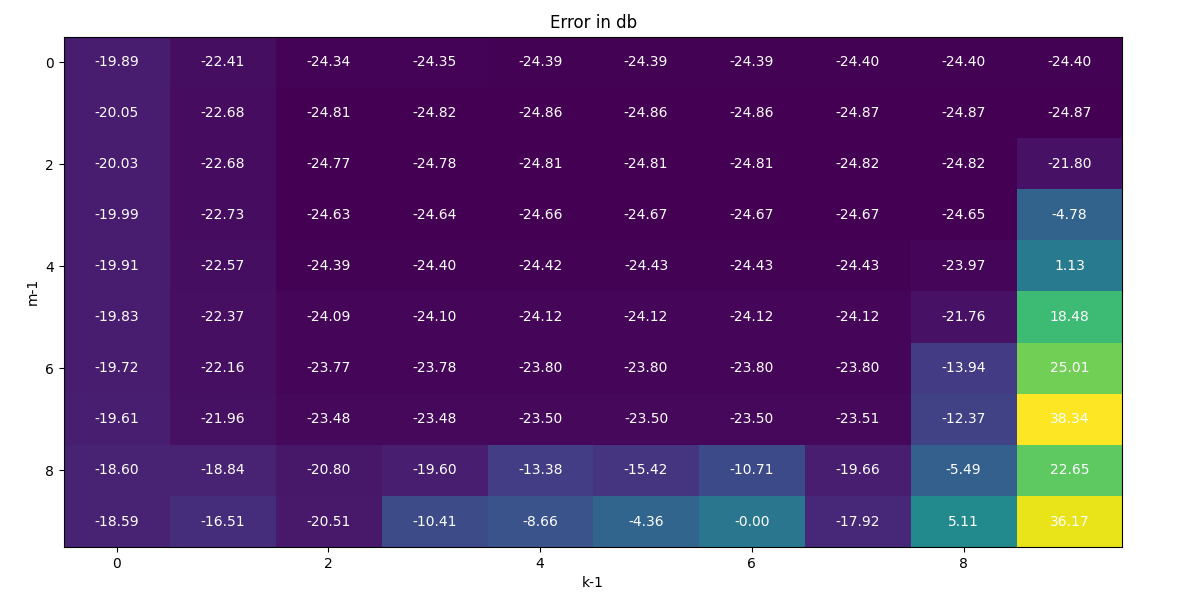


Figure 3.3.a - error between calculated and measured output in db as function of parameters m,k

After an adequate order of non-linearity and order of memory were selected, and if the range of input amplitudes and frequencies present in a general input signal fall within those represented by the calibration data input signal, then the realistic output of the HPA can be simulated (according to MP model).

DPD

As detailed before, we modeled DPD output signal such that:

where is the DPD output signal array, is the delay matrix formed by the DPD input signal, and is an array containing all the MP model coefficients of the DPD.

In order to calculate the coefficient array , the same set of measurements is used, but it is scaled and used in reverse order. The input signal array remains unscaled, but the measured output signal is rescaled in a way that its maximum magnitude equals the maximum magnitude of the input array x:

When using the well-known method of minimizing the least squares error between the calculated DPD output and measured DPD output (which is the input array x). The coefficients vector is given by:

where is the unscaled input signal array of the calibration data, is the delay matrix formed by the rescaled calibration data output signal y, and the superscript is the conjugate transpose operator.

In order to find the optimal coefficients for the MP model, it is essential to choose values of m and k which model the DPD optimally. In order to do that, one should find values of m and k that minimize the error between calculated DPD output generated by coefficients calculated with those specific m and k, and the real input signal. This optimization was doen for our lab’s PA, and results are shown on figure-3.3.b. The error is minimized at k=9, m=2, which is similar to the same parameters of the PA model.

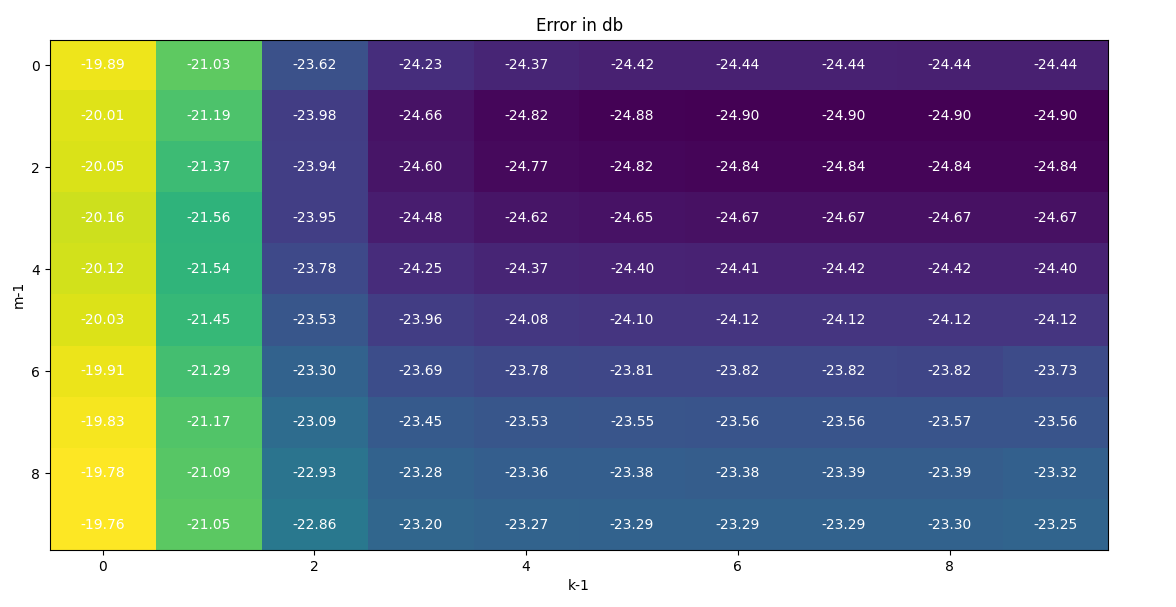


Figure 3.3.b - error between calculated and measured input in db as function of parameters m,k

## 3.4 Neural Network

Neural Network explained

NNs usually consist of an input layer, hidden layers, and an output layer. The number of hidden layers is a hyperparameter which changes according to the specific problem, and many times is determined during training of the network or by using empirical methods. The output of each neuron is equal to the bias plus the sum of the products of the input signals and corresponding weights, which is expressed as:

Were l is the index of the layer, i is the index of the neuron in the layer, j is the index of the neuron in the previous layer, p is the number of neurons in the previous layer, is the weight connecting the i-th neuron of the l-th layer to the j-th neuron of the (l-1)-th layer, is the input from the j-th neuron of the (l-1)-th layer to the i-th neuron of the l-th layer, is the bias of the i-th neuron at the l-th layer and f is the activation function. Commonly used activation functions are the log sigmoid (logsig) and relu.

After the cost function between the NN output of the first iteration and desired output is calculated, the values of the weights and biases are updated using a well-known type of the back-propagation algorithm that has been shown in literature to perform exceptionally well. The cost function is then calculated again using the updated weights and biases during the next iteration. Learning stops when the number of iteration reaches the maximum iterations allowed, or when the NN satisfies the desired performance in terms of MSE.

Modeling a PA and a DPD with a NN

Due to their strong adaptive nature and approximation capability, NNs are very attractive for the behavioral modeling of PAs. One optional architecture for such NN is a single-input single-output feedforward model, in which the network gets a complex signal as input and extracts complex PA output. However, this architecture requires the use of complex-valued weights and activation functions, which results in complicated calculations. Another proposed architecture is a polar feed forward NN, in which the network consist of two different NNs. The first NN extracts the amplitude response of the PA output, and the second extracts the phase response. However, the two NN branches in this design usually cannot converge at the same time, resulting in overtraining or undertraining.

In order to solve this, another architecture has been proposed, as shown in figure 3.4.a. This model uses the I and Q signals as inputs, and it resolves the simultaneous convergence issue mention above. However, this architecture does not take into account memory effects. These effects become dominant in wideband signals and need more consideration.

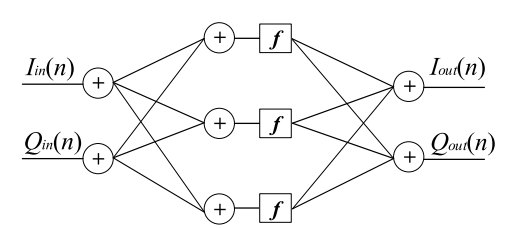


Figure 3.4.a – NN architecture for modeling a PA

The proposed model in the article (להוסיף רפרנס) is a combination of the above architectures, and it presented in figure 3.4.b. We used this model as a baseline for our proposed solution.

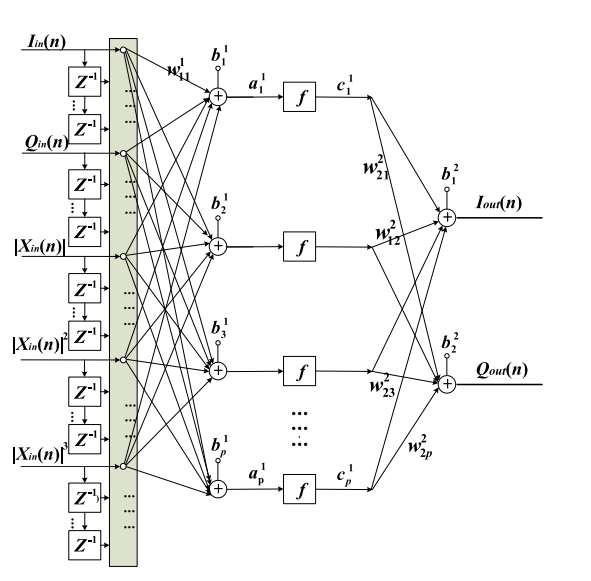


Figure 3.4.b – proposed NN architecture for modeling a PA. We used this as a baseline for our proposed solution.

It considers input’s in-phase (I) and quadrature (Q) components, the amplitudes of the input signal and nonlinear orders of it. In addition, this model include past samples of that features, and, therefore, quite suitable for the modeling of memory effects. It can be observed that the input feature vector for this model is:

And therefore input vector size is: , and output vector size is .

The memory order M and the order of nonlinearity K are determined by an optimization process and may vary depending on the PA.

The cost function is the MSE between the measured PA output and the NN output, and it defined as:

As mentioned before, we found the ideal memory order (m=2) and nonlinearity order (k=9), therefore input vector size is 22, and output vector size is 2. In the training process, we have tested different architectures (while using the architecture in figure 3.4.b as a baseline) in order to find the optimal architecture which can model the PA and the DPD. We tried to change the number of the hidden layers, and the number of neurons at each layer. After tests and comparisons, the architecture that was chosen is a feedforward NN consist of three hidden layers, which contain 22-40-2 neurons, respectively. The chosen architecture is shown on figure 3.4.c.

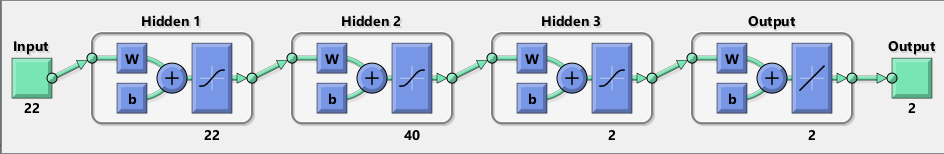
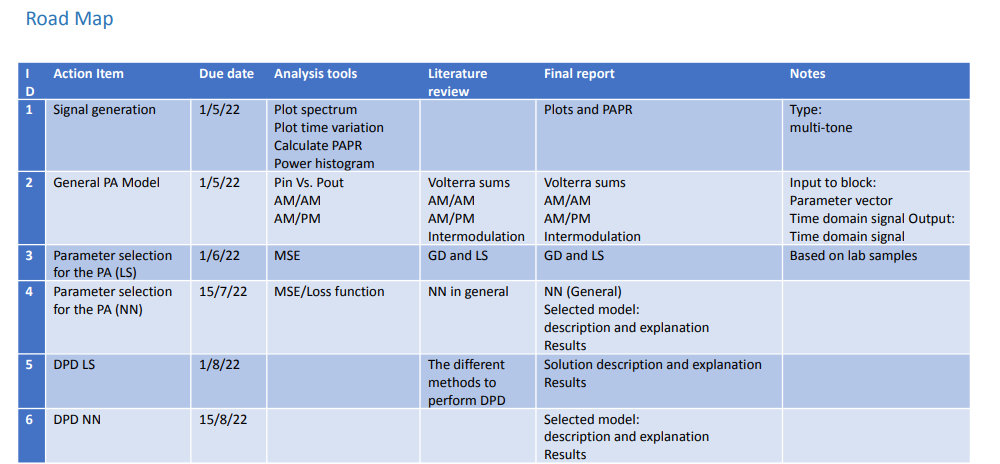


Figure 3.4.c – The chosen NN architecture we used to model PA and DPD

# Metodology

Data (lab & simulation)

In order to model a real PA, we used data recorded in our lab’s PA. We took sampling rate of 10MHz, of about 60 tones, from a bandwidth of about 1MHz.



# Results

## PA Modelling

Classical

In order to check calculated parameters accuracy, several sanity checks should be made. The first one is to show AM/AM, AM/PM and OOB emission graphs of the modeled PA’s output, and to check if they are similar to the same graphs of the measured output as shown in figures 3.1.a-d. This graphs are shown in figures 5.1.a-d respectively.

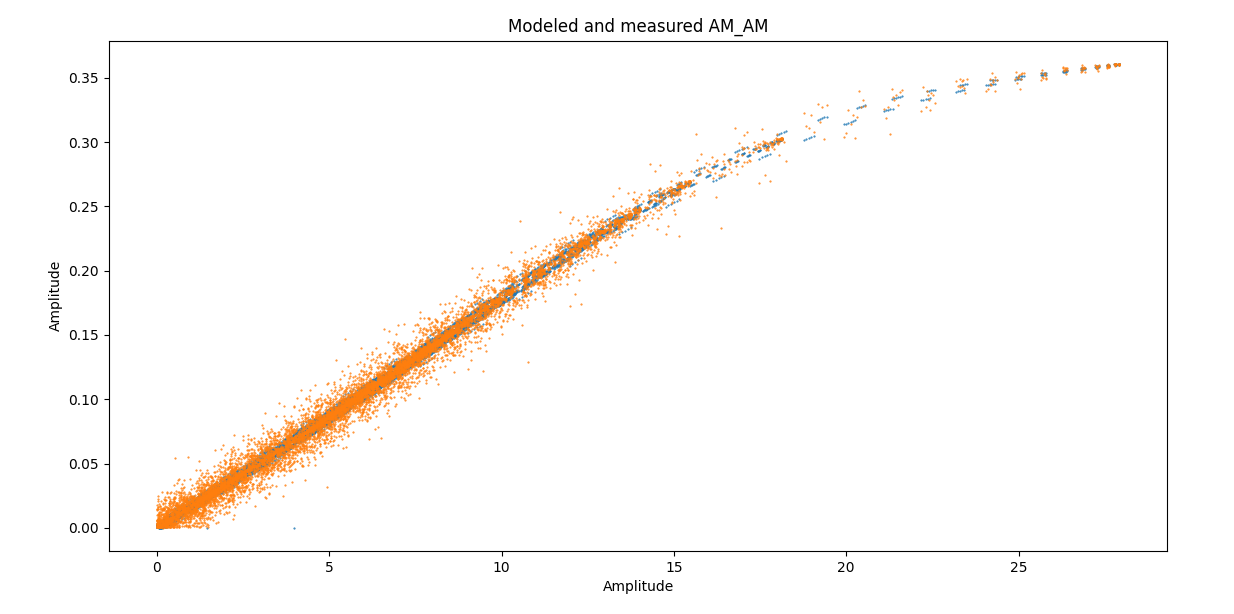


figure 5.1.a - illustration of AM/AM distortion of input signal, caused by classically modeled PA in blue, and real lab’s PA in orange

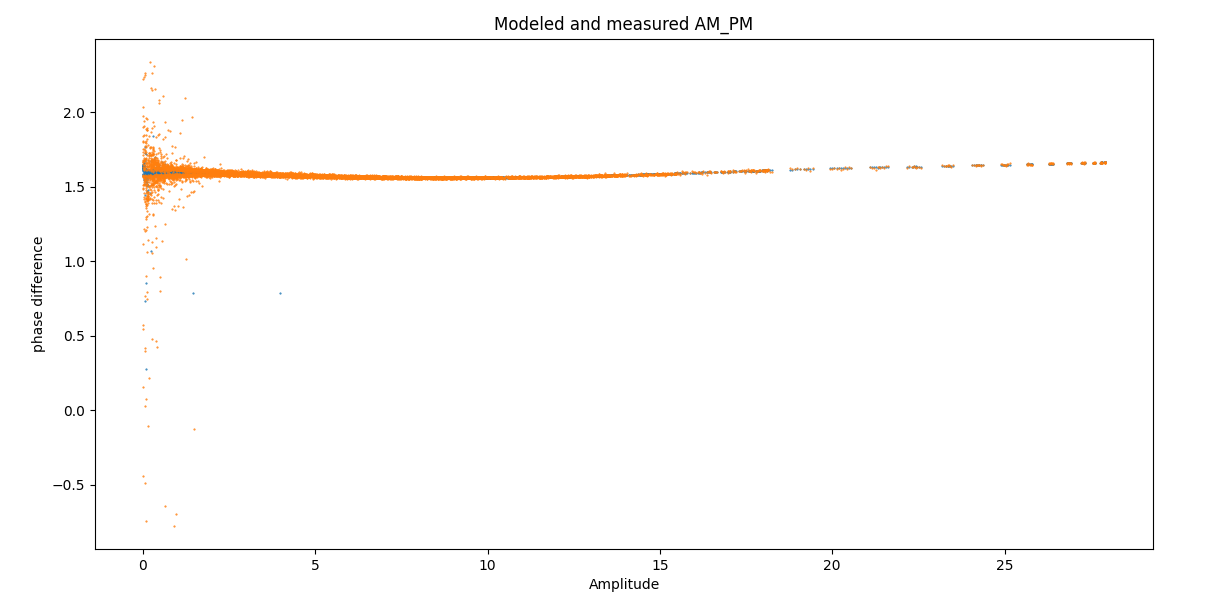


figure 5.1.b - illustration of AM/PM distortion of input signal, caused by classically modeled PA in blue, and real lab’s PA in orange

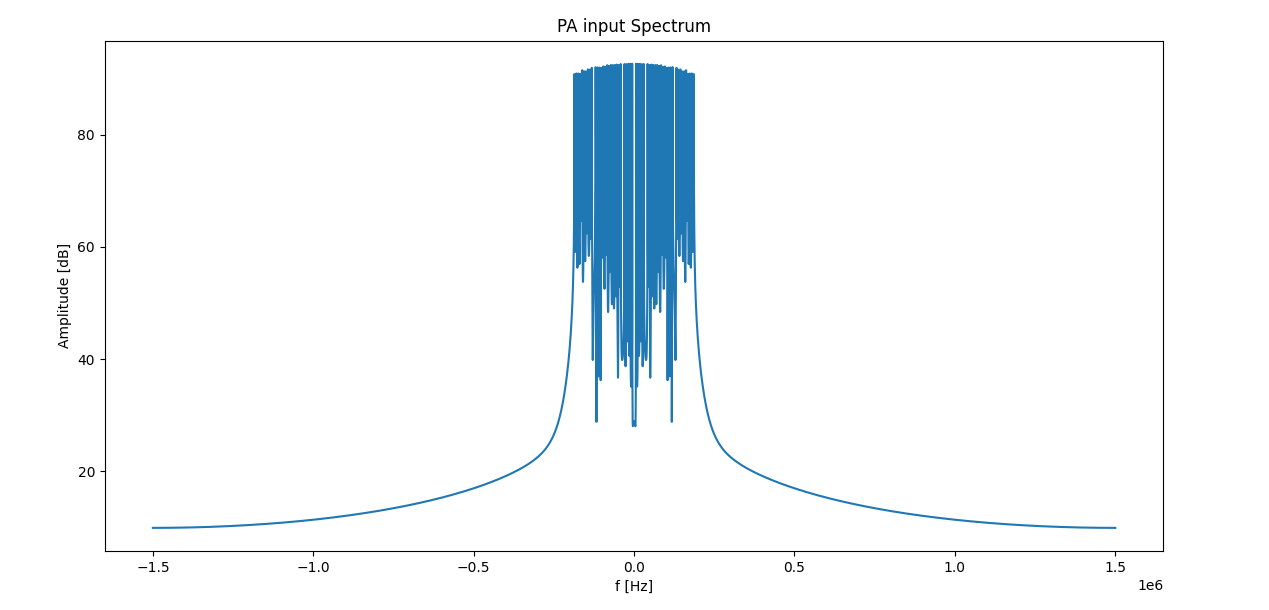


Figure 5.1.c – illustration of input signal spectrum

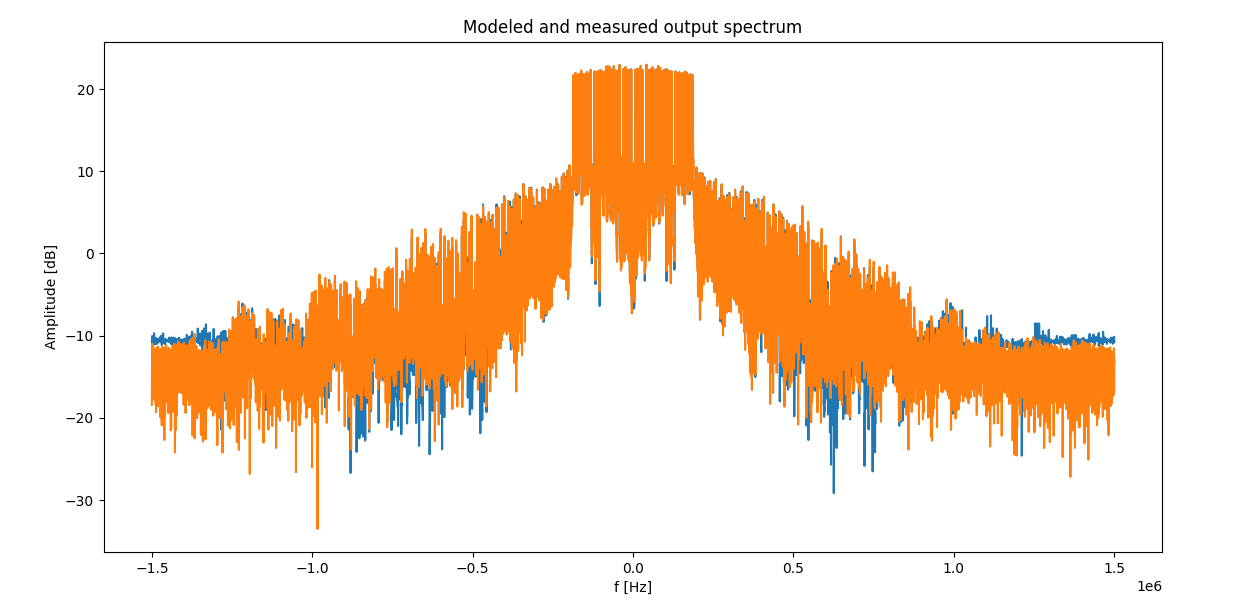


Figure 5.1.d – illustration of OOB emission caused by classically modeled PA in blue, and real lab’s PA in orange

The results of the SGD process was satisfying as well, which are shown in figure 5.1.e.

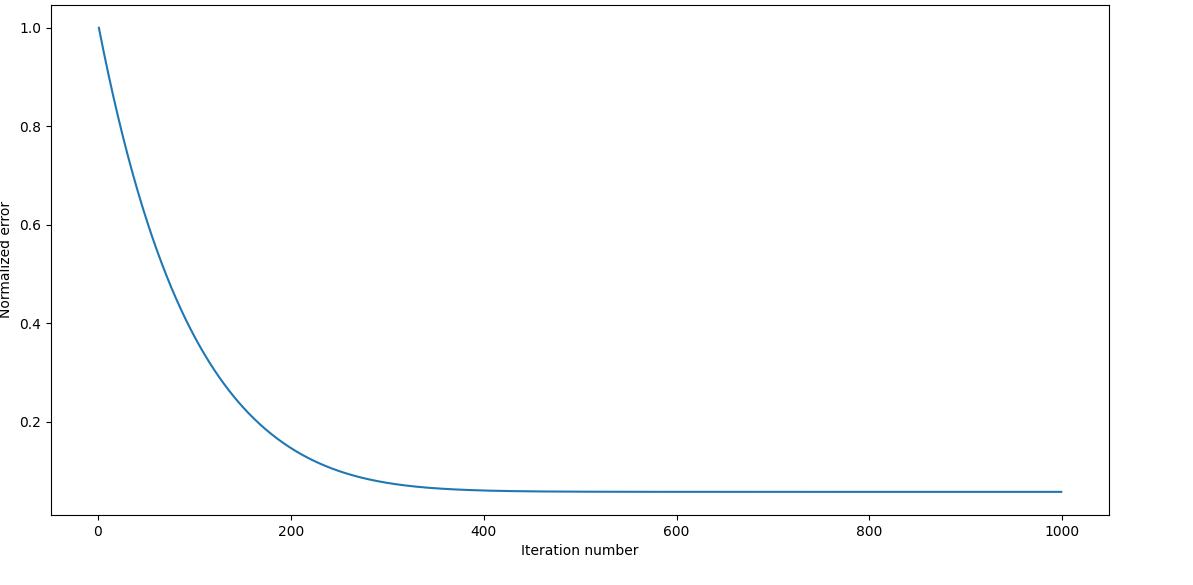


figure 5.1.e - Normalized error between SGD modeled PA output and real Lab’s PA output

Although the above problem has an analytic solution, we also used SGD to model the PA. For a sanity check, After we got the final parameter vector , we made a comparison to it with , by calculating:

And the result was , which indicates a good convergence to solution.

NN

As described before, we tried different NN architectures for modeling the PA. One of the architectures we tried was a feedforward NN that consist of three hidden layers, at the sizes of 22-40-2 neurons, respectively. AM/AM, AM/PM and OOB emission graphs of this NN output are shown in figures 5.1.f-h respectively.

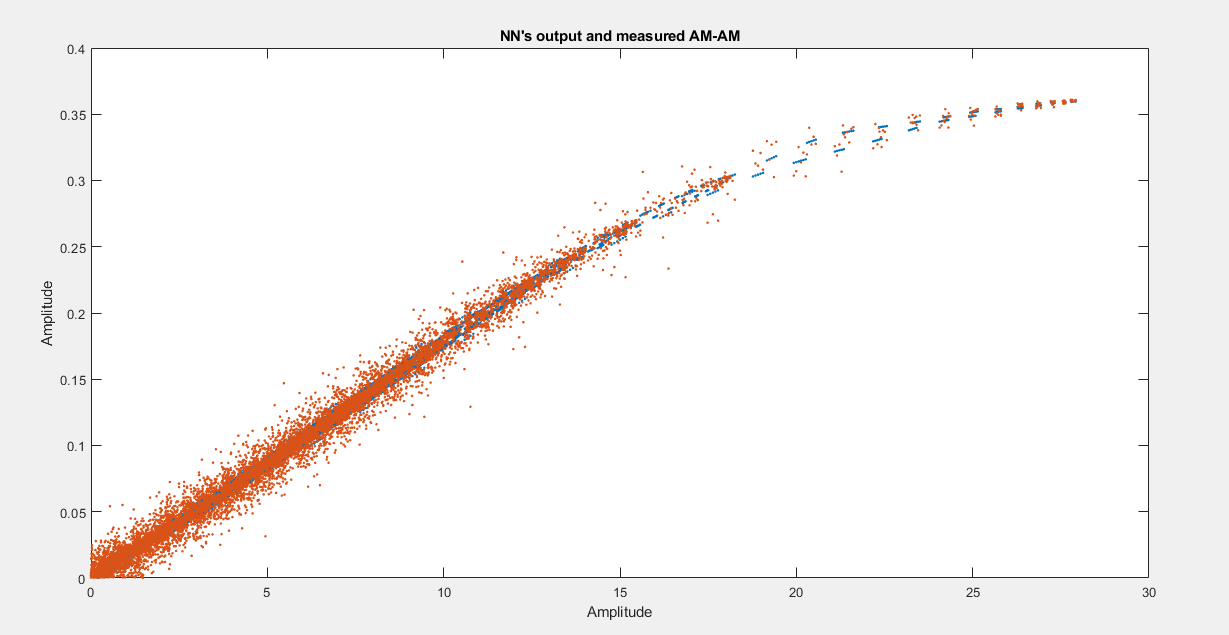


figure 5.1.f - illustration of AM/AM distortion of input signal, caused by feedforward (22-40-2) NN modeled PA in blue, and real lab’s PA in orange

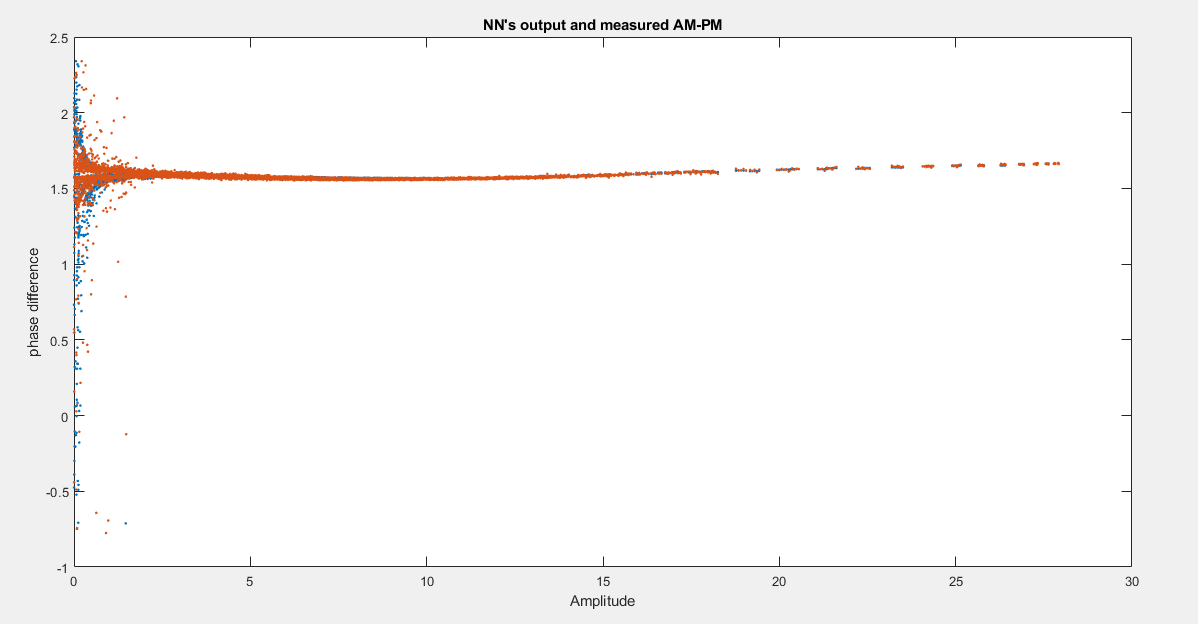


figure 5.1.g - illustration of AM/PM distortion of input signal, caused by feedforward (22-40-2) NN modeled PA in blue, and real lab’s PA in orange

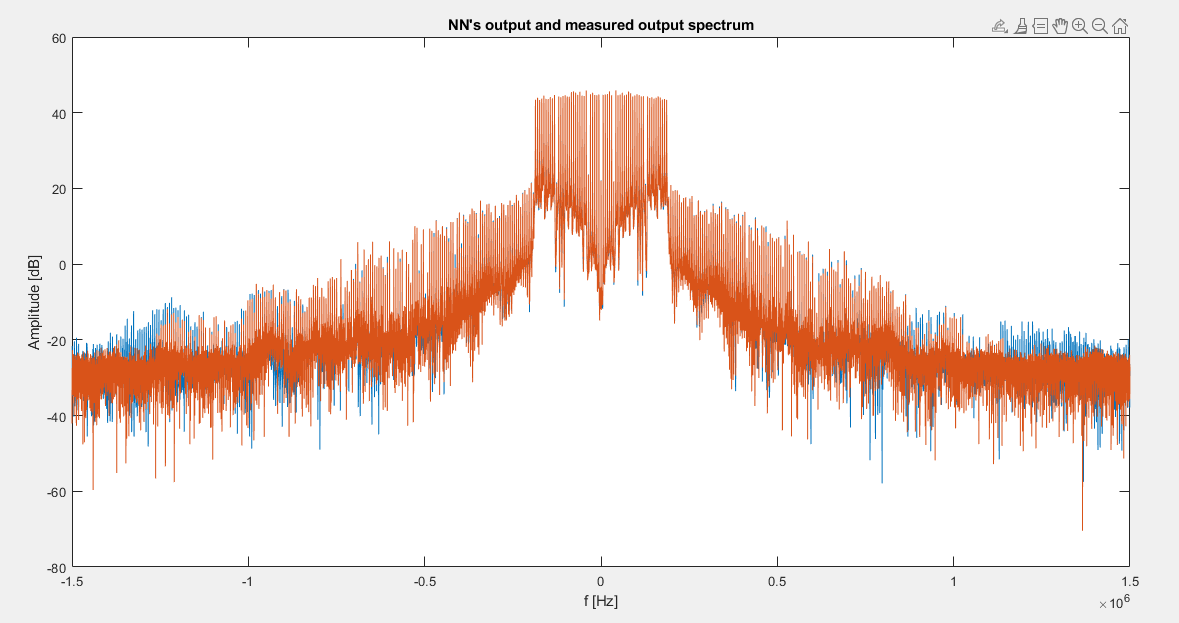


Figure 5.1.h – illustration of OOB emission caused by feedforward (22-40-2) NN modeled PA in blue, and real lab’s PA in orange

COMPARISONS to other architectures

## DPD

Classical

In order to check the effectiveness of the proposed classical DPD, we will show AM/AM, AM/PM and OOB emission graphs of the output, with and without using DPD, to illustrate that the classical DPD overcomes well the PA nonlinearity. This graphs are shown in figures 5.2.a-c respectively.

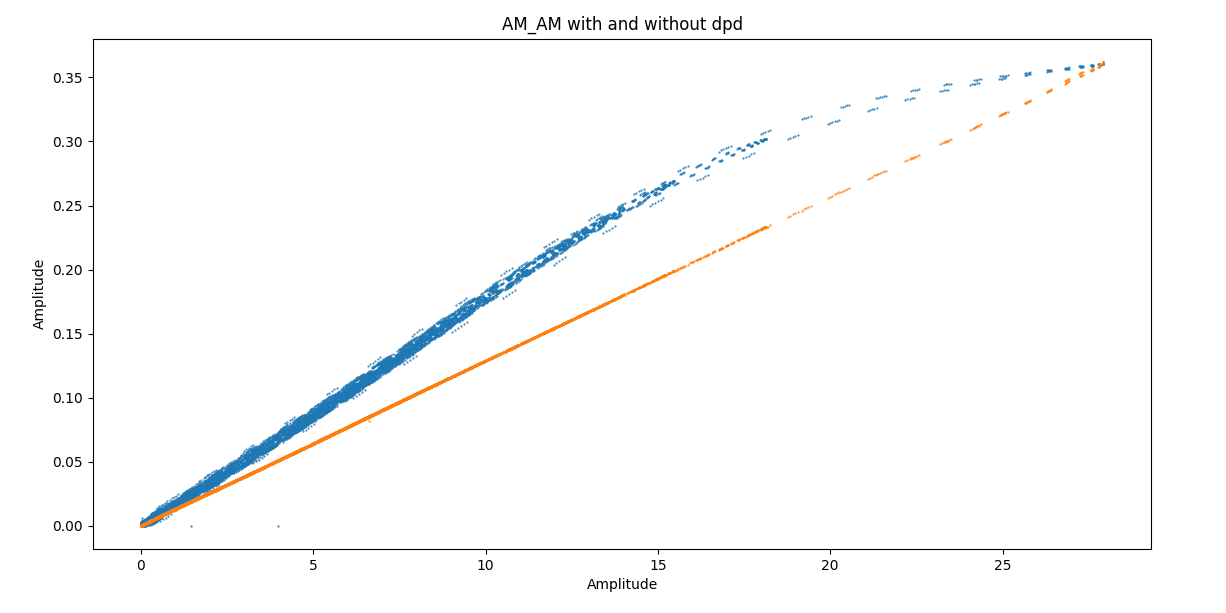


figure 5.2.a - illustration of AM/AM distortion of input signal, with (orange) and without (blue) classically modeled DPD

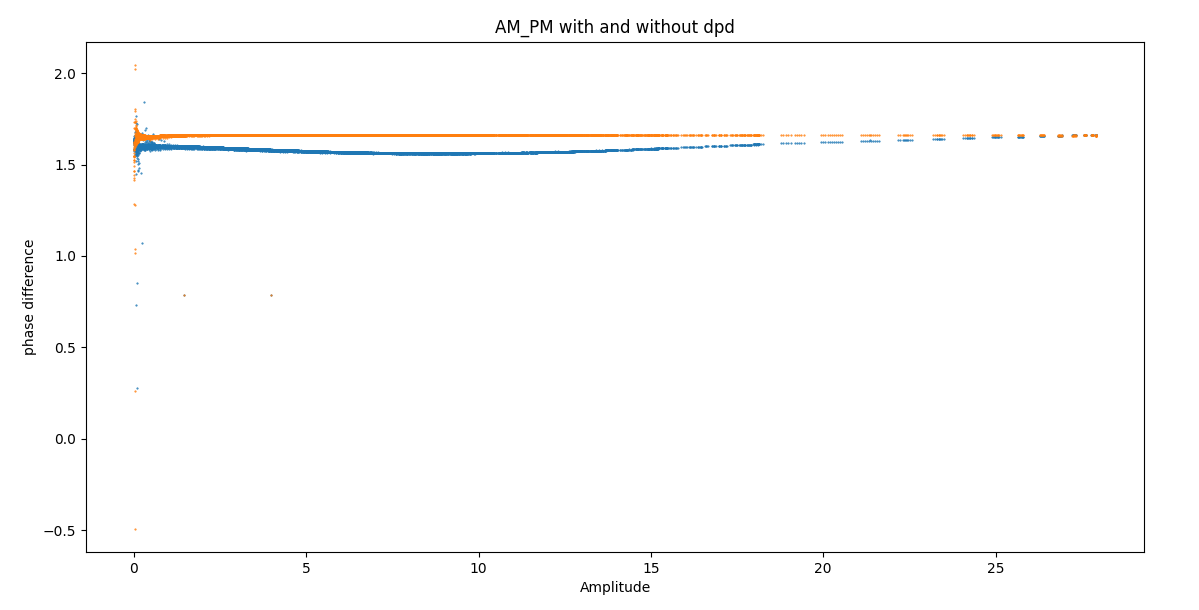


figure 5.2.b - illustration of AM/PM distortion of input signal, with (orange) and without (blue) classically modeled DPD

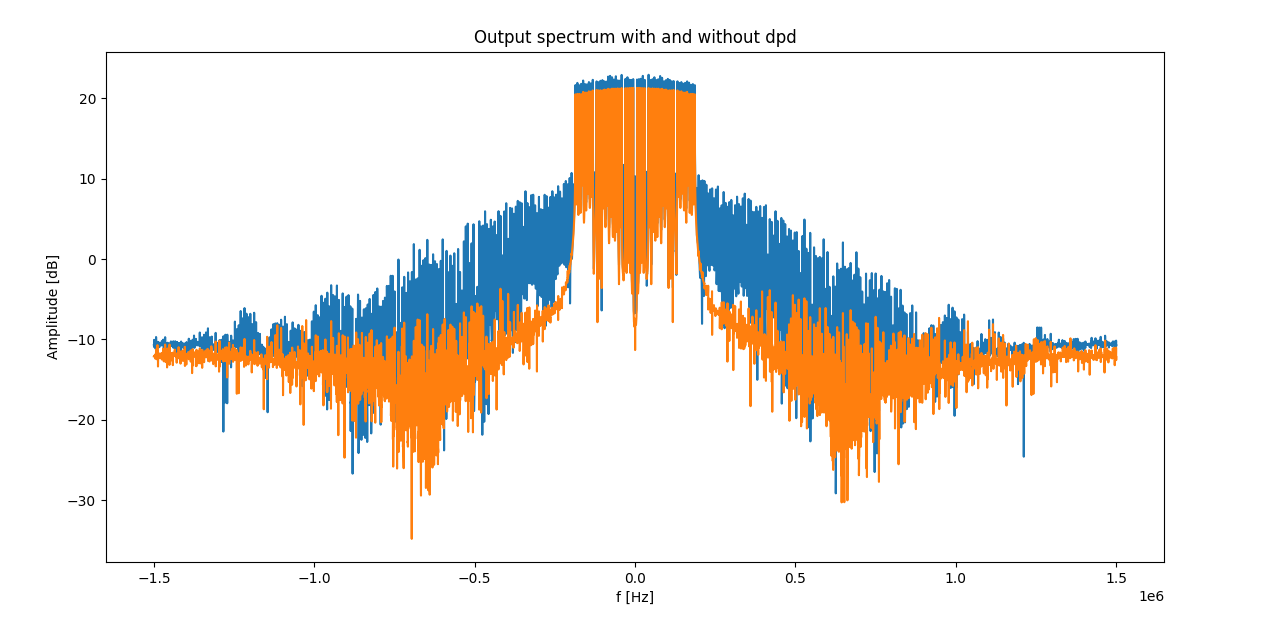


figure 5.2.c - illustration of OOB emission caused by modeled PA, with (orange) and without (blue) classically modeled DPD

NN

In order to check the effectiveness of the proposed NN DPD, we tried different NN architectures for modeling it. One of the architectures we tried was a feedforward NN that consist of three hidden layers, at the sizes of 22-40-2 neurons, respectively. We will show AM/AM, AM/PM and OOB emission graphs of the output, with and without using DPD, to illustrate that the DPD somewhat overcomes the PA nonlinearity. This graphs are shown in figures 5.2.d-f respectively.

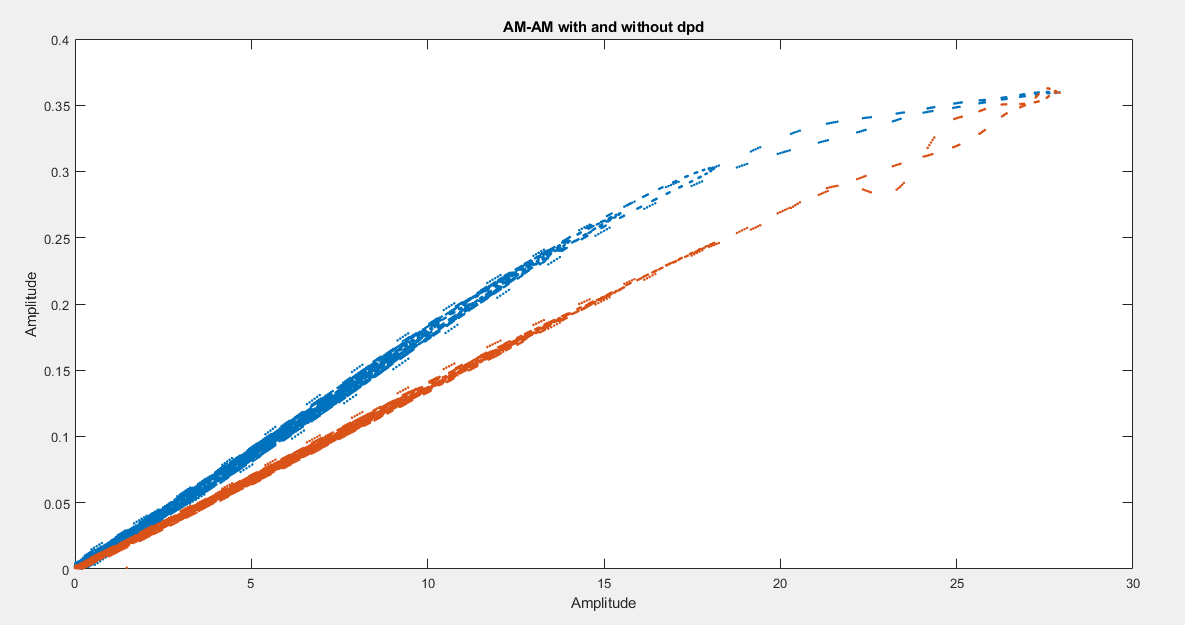


figure 5.2.d - illustration of AM/AM distortion of input signal, with (orange) and without (blue) feedforward (22-40-2) NN modeled DPD

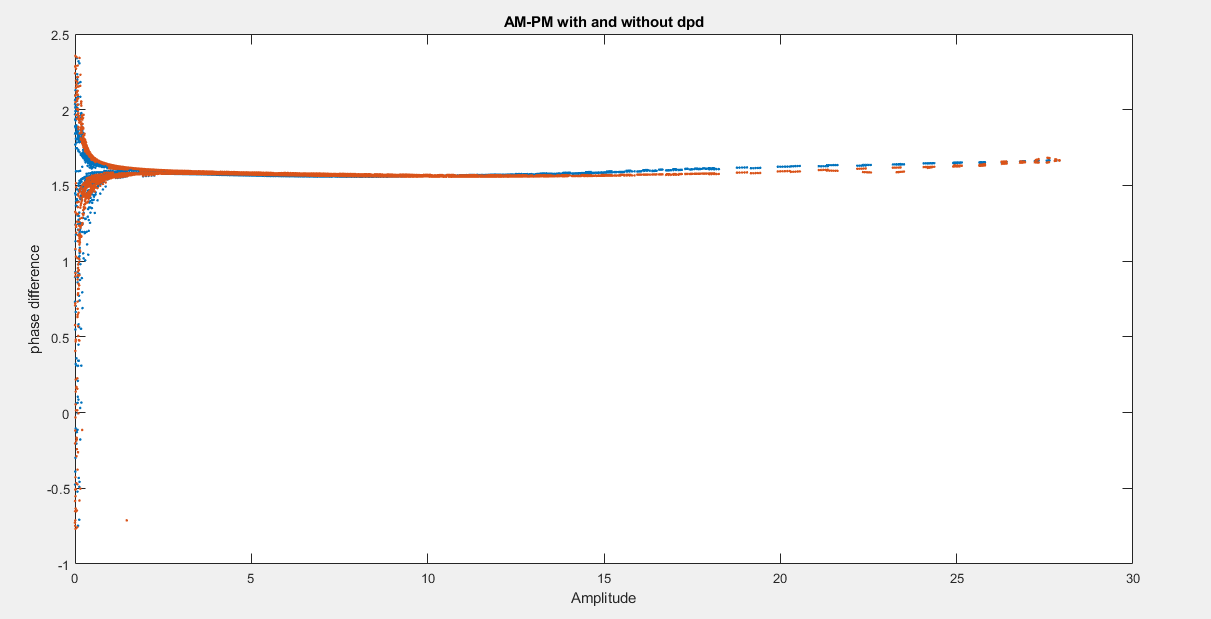


figure 5.2.e - illustration of AM/PM distortion of input signal, with (orange) and without (blue) feedforward (22-40-2) NN modeled DPD

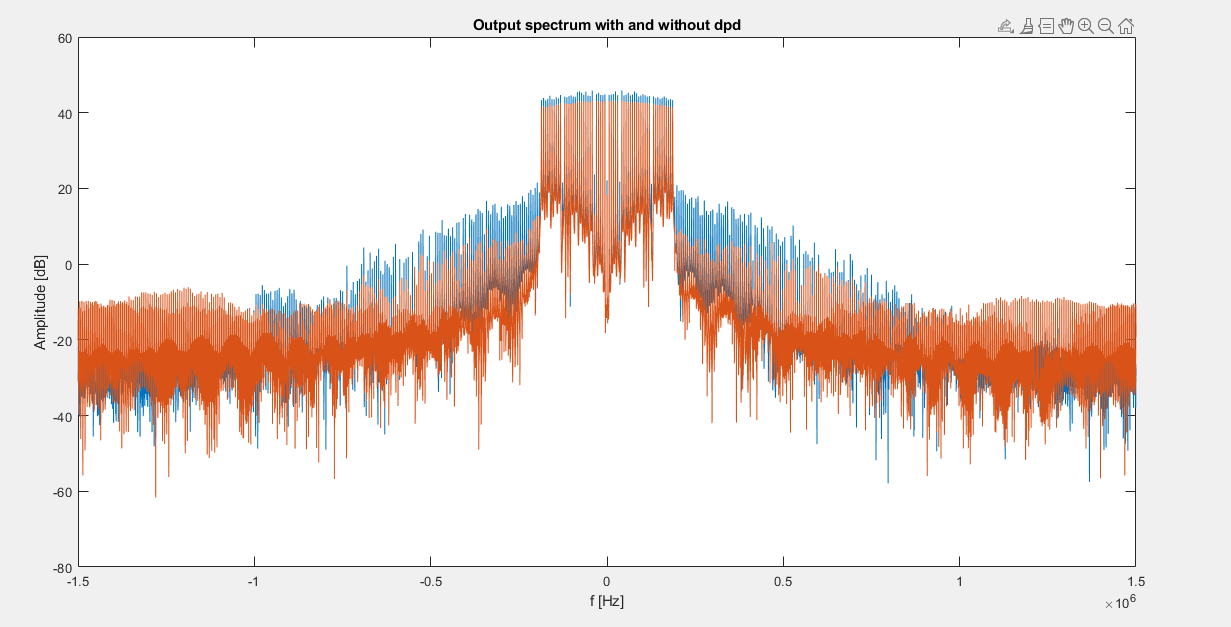


figure 5.2.f - illustration of AM/PM distortion of input signal, with (orange) and without (blue) feedforward (22-40-2) NN modeled DPD

COMPARISONS to other architectures

# Discussion

Pros and Cons of each method, Insights on stuff (e.g. overfitting, gradient normalization, complex numbers …)

# Conclusion