School of Information Science and Engineering

射频电路建模与CAD方法（双语）

编号:B0433111

Final Lab: DPD Extraction and Validation

Student Number: 04022212, 04022114
Name: 钟源，贾岩森

·Report

1. Introduction

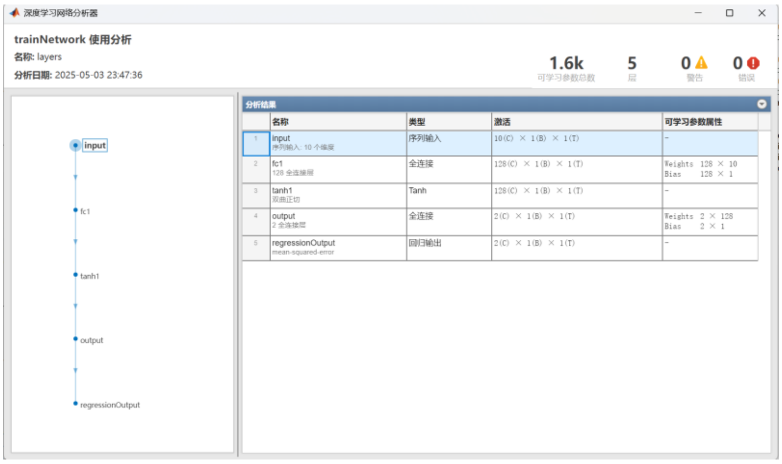
In this laboratory assignment, we are given a 3-way PA with memory effects. We aim to build a DPD model for given PA circuit using knowledge learned from the course. To achieve this goal, We have tried a lot of methods, including Feedforward Neural Network (FNN), Recurrent Neural Network(RNN) and Transformer. In the Networks we design, we have combined FCN, CNN, LSTM, Bi-LSTM, GRU and other useful network layers. These results will be shown in the upcoming section.

2. Methods Selection

2.1 Fully Connected Network (FCN)

Firstly, we use Deep Network Designer in MATLAB to design common neural networks.
As the most fundamental network structure in neural networks, the Fully Connected Network has characteristics such as full connectivity, no spatial assumptions, and hierarchical stacking. It can capture complex global relationships between input features and theoretically approximate any continuous function, making it very suitable for processing DPD input data.

We chose Adam as the optimization algorithm and constructed a simple neural network consisting entirely of fully connected layers, as follows:



Set Memory Depth (M) to 5, with a maximum training frequency of 15000 rounds (stopping around 11320 rounds when the validation conditions are met), and the fitting accuracy result is -35.315 dB.

模型的拟合精度NMSE为 -35.314857 dB

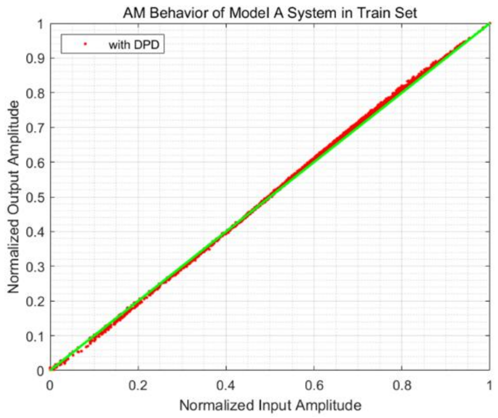
In order to better measure the generalization ability of the network model, we set up two datasets to examine whether it can be well applied to the original circuit. The selection of the two datasets is as follows. Starting the simulation, 7993 sample points were sampled at equal intervals of 0-999us as dataset 1, named Train Set (as the data obtained comes from this time period); Afterwards, 2000 sample points were sampled at equal intervals in the 250us dataset, and named Valid Set as Dataset 2. It should be noted that the Train Set is not truly the dataset for training the model, as the input data for the model's training set is the output data of PA, and the output data is the input data of PA. When DPD is applied to circuits, its input is the original input data of PA.

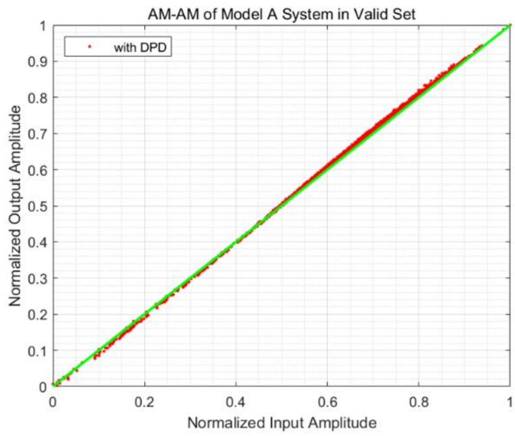
After verification, the NMSE of the full FCN network is -35.276dB on the Train Set and -35.324dB on the Valid Set.

使用Model A，系统的NMSE(Train)仿真结果为 -35.276285 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -35.323938 dB

The obtained AM-AM image is as follows:





It can be seen that compared to the absence of DPD, NMSE has decreased by about 10dB, which is reduced to 1/10 of its original size. This demonstrates the significant role of using DPD and the potential of FCN networks.

2.2 Long Short-Term Memory (LSTM)

LSTM is suitable for processing and predicting time series data, and is suitable for big data environments. LSTM can effectively capture and remember long-term dependencies in sequence data, making it highly suitable for solving PA nonlinear problems, and therefore widely used in the field of DPD optimization.

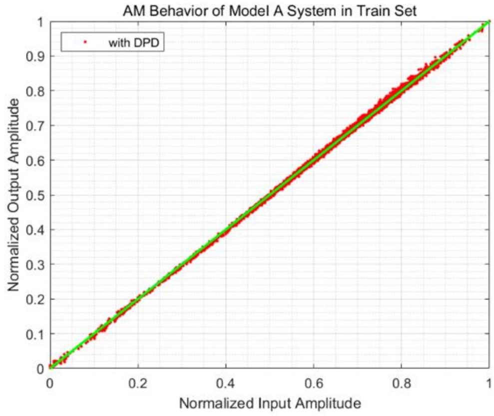
Adam was still chosen as the optimization algorithm, with Memory Depth (M) set to 5 and a maximum training frequency of 4000 rounds (stopping at around 900 rounds when the validation conditions are met). A layer of LSTM network was constructed (as shown below), and its fitted NMSE result was -42.341dB.

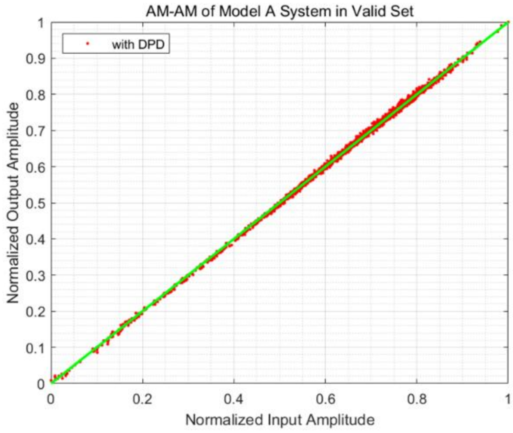
After verification, the NMSE of the LSTM network on the Train Set is -38.654dB, and the NMSE on the Valid Set is -38.929dB。

使用Model A，系统的NMSE(Train)仿真结果为 -38.654426 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -38.929237 dB

The obtained AM-AM image is as follows:



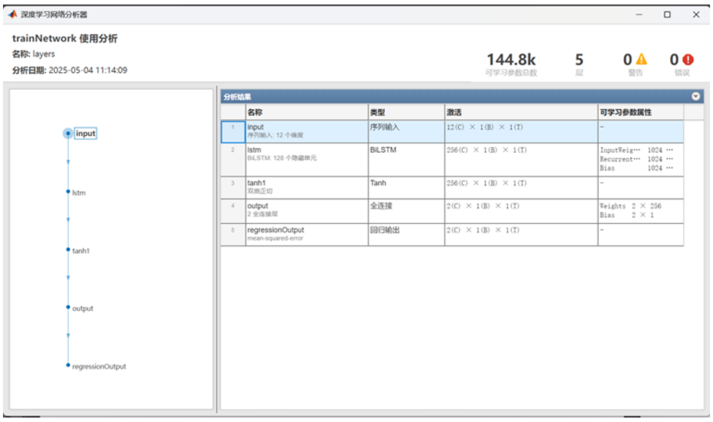


It can be seen that although the LSTM model has higher fitting accuracy, its performance actually decreases by about 4dB when used in circuits, indicating a certain degree of overfitting. In the following equally complex networks, it can be seen that the more complex the network, the more overfitting may occur. This enlightens us that we cannot endlessly increase the network complexity, otherwise the model performance may actually decline.

2.3 Bi-directional Long Short-Term Memory (Bi-LSTM)

Due to Bi LSTM's ability to better capture bidirectional sequence dependencies, we have also attempted this type of network.

Adam was still chosen as the optimization algorithm, with Memory Depth (M) set to 5 and a maximum training frequency of 4000 rounds (stopping at around 840 rounds when the validation conditions were met). A Bi LSTM network was constructed (as shown below), and its fitted NMSE result was -44.254dB。



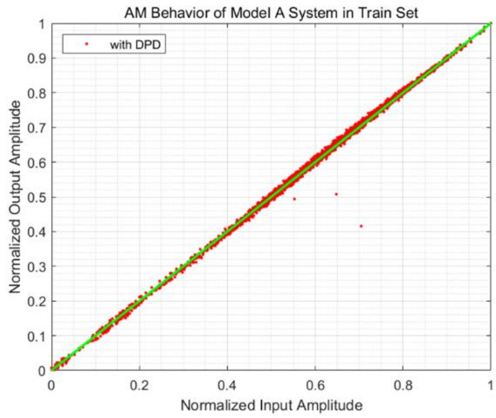
	名称	类型	激活	可学习参数属性
1	input 序列输入: 12 个维度	序列输入	12(C) × 1(B) × 1(T)	-
2	lstm BiLSTM: 128 个隐藏单元	BiLSTM	256(C) × 1(B) × 1(T)	InputWeig… 1024 … Recurrent… 1024 … Bias 1024 …
3	tanh1 双曲正切	Tanh	256(C) × 1(B) × 1(T)	-
4	output 2 全连接层	全连接	2(C) × 1(B) × 1(T)	Weights 2 × 256 Bias 2 × 1
5	regressionOutput mean-squared-error	回归输出	2(C) × 1(B) × 1(T)	-

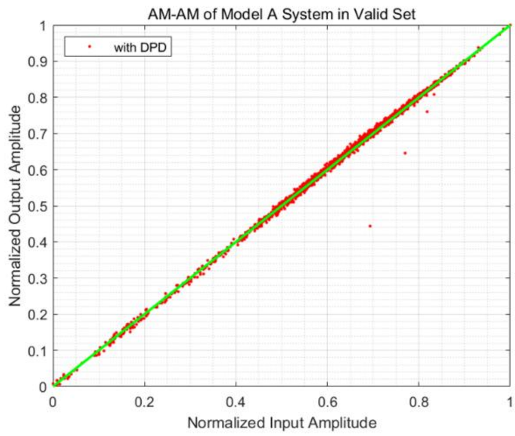
After verification, the NMSE of Bi LSTM network on Train Set is -33.932dB, and on Valid Set it is -33.205dB。

```
使用Model A，系统的NMSE(Train)仿真结果为 -33.932480 dB
```

```
使用Model A，系统的NMSE(Valid)仿真结果为 -33.205341 dB
```

The obtained AM-AM image is as follows：



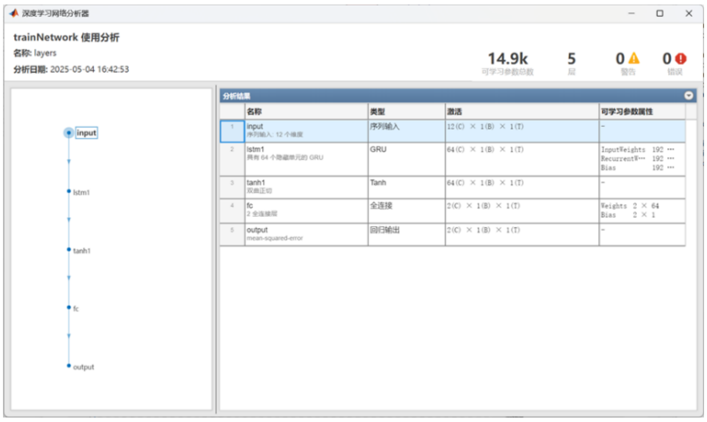


As can be seen, the network of Bi LSTM is more complex (with learning parameters exceeding 144k!), and its overfitting trend is more severe. Although the model fits NMSE to around -44dB, its actual NMSE is less than -34dB (inferior to FCN) when applied, and outliers appear in the AM image, which inspires us to search for simpler networks。

2.4 Gated Recurrent Unit (GRU)

GRU is generally regarded as a simplified version of LSTM, with 1/3 fewer parameters to train than LSTM. We also tried this network and based on the idea of preventing overfitting, we tried to reduce the number of hidden layers and found good results.

Adam was still chosen as the optimization algorithm, with Memory Depth (M) set to 5 and a maximum training frequency of 4000 rounds (stopping at around 1640 rounds when the validation conditions were met). A layer of Bi LSTM network was constructed (as shown below), and its fitted NMSE result was -47.393dB.

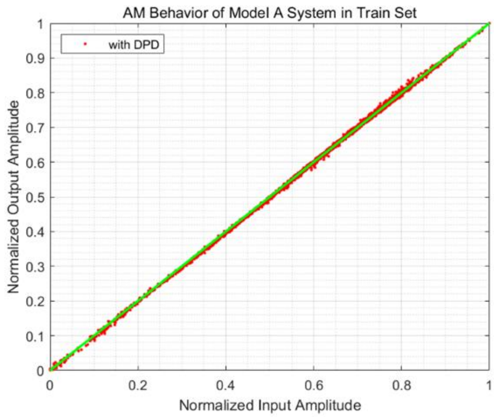


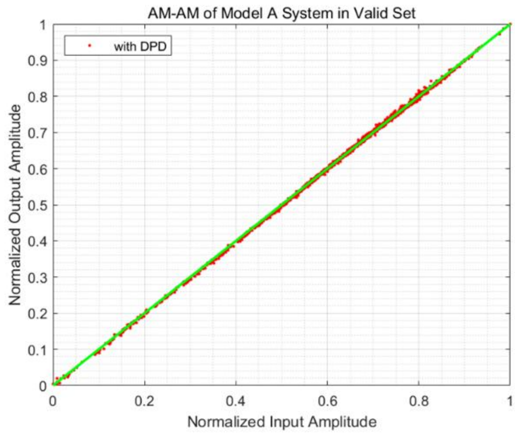
After verification, the NMSE of the GRU network on the Train Set is -43.417dB, and the NMSE on the Valid Set is -43.694dB。

```
使用Model A，系统的NMSE(Train)仿真结果为 -43.417162 dB
```

```
使用Model A，系统的NMSE(Valid)仿真结果为 -43.693773 dB
```

The obtained AM-AM image is as follows：



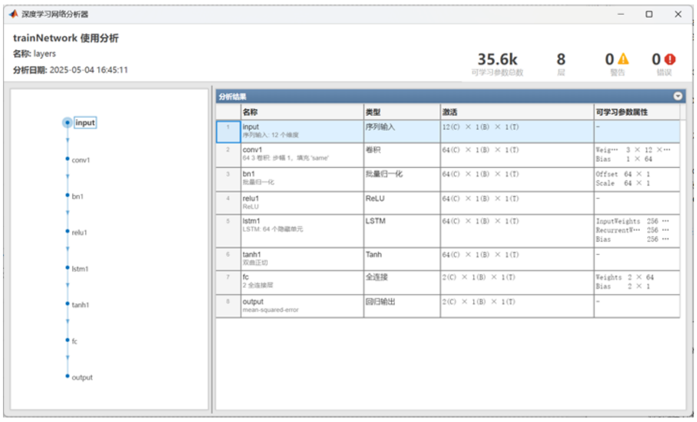


It can be seen that although GRU also suffers from overfitting, it has been improved, and its NMSE when applied can reach over -43dB, with a significant performance improvement.

2.5 CNN-LSTM

When searching for information, we found that 1D-CNN often appears together with LSTM because it can capture nonlinear distortions in the model and reduce model complexity. We are also trying to combine CNN with LSTM to generate a hybrid network.

Still choosing Adam as the optimization algorithm, with Memory Depth (M) set to 5 and a maximum training frequency of 4000 rounds (stopping after satisfying the validation conditions around 1160 rounds), a layer of CNN-LSTM network was constructed (as shown below), and its fitted NMSE result was -38.850dB.

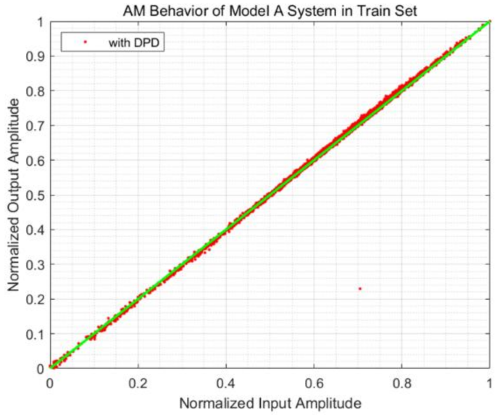


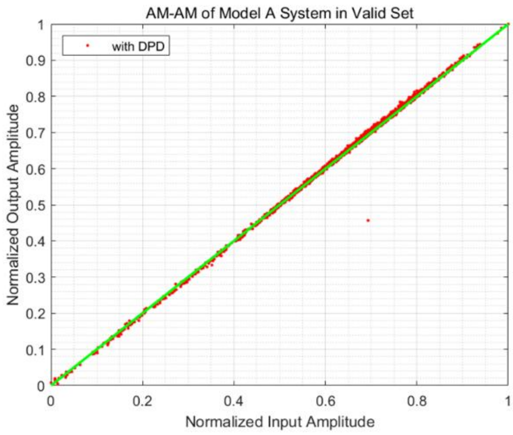
After verification, the NMSE of the CNN-LSTM network on the Train Set is -36.615dB, and on the Valid Set it is -36.823dB.

使用Model A，系统的NMSE(Train)仿真结果为 -36.615214 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -36.822592 dB

The obtained AM-AM image is as follows：

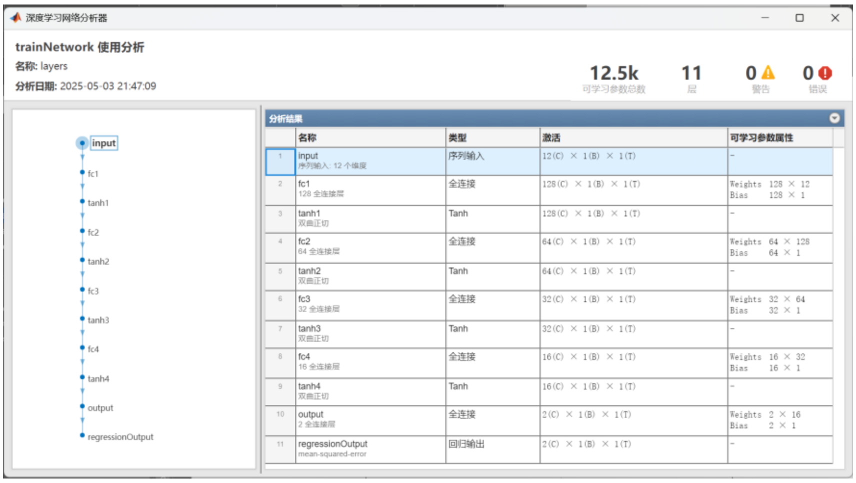




It can be seen that introducing CNN does help reduce model complexity, effectively improve overfitting, and the training time is very short. Unfortunately, the performance of the model itself is not very strong and can be applied to scenarios where DPD accuracy is not very high but requires fast modeling.

2.6 Multi-layer FCN

After some exploration, we found that generally speaking, increasing the number of hidden layers and the number of hidden layers in FCN can significantly improve the optimization effect of DPD without fitting. Based on this, we propose a multi-layer structure full FCN network using Adam optimization algorithm, as follows:



深度学习网络分析器

trainNetwork 使用分析
名称: layers
分析日期: 2025-05-03 21:47:09

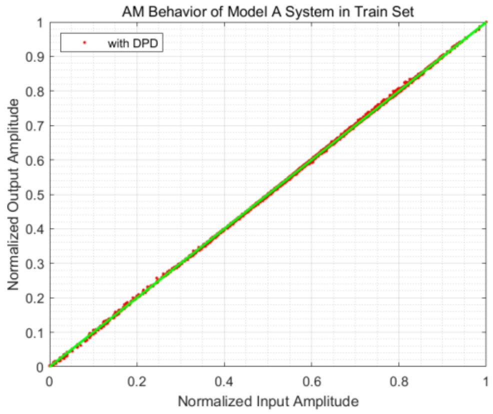
12.5k 可学习参数总数 | 11 层 | 0 警告 | 0 错误

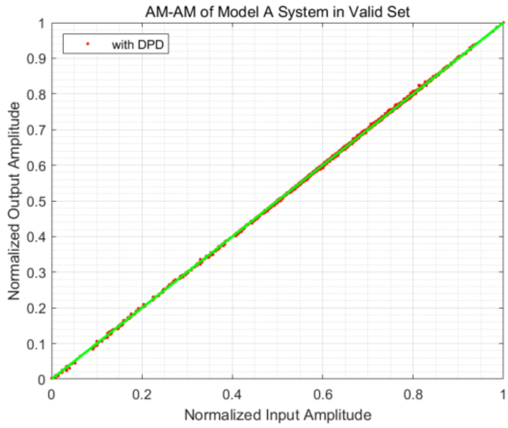
分析结果

	名称	类型	激活	可学习参数属性
1	input 序列输入: 12 个维度	序列输入	12(C) × 1(B) × 1(T)	-
2	fc1 128 全连接层	全连接	128(C) × 1(B) × 1(T)	Weights 128 × 12 Bias 128 × 1
3	tanh1 双曲正切	Tanh	128(C) × 1(B) × 1(T)	-
4	fc2 64 全连接层	全连接	64(C) × 1(B) × 1(T)	Weights 64 × 128 Bias 64 × 1
5	tanh2 双曲正切	Tanh	64(C) × 1(B) × 1(T)	-
6	fc3 32 全连接层	全连接	32(C) × 1(B) × 1(T)	Weights 32 × 64 Bias 32 × 1
7	tanh3 双曲正切	Tanh	32(C) × 1(B) × 1(T)	-
8	fc4 16 全连接层	全连接	16(C) × 1(B) × 1(T)	Weights 16 × 32 Bias 16 × 1
9	tanh4 双曲正切	Tanh	16(C) × 1(B) × 1(T)	-
10	output 2 全连接层	全连接	2(C) × 1(B) × 1(T)	Weights 2 × 16 Bias 2 × 1
11	regressionOutput mean-squared-error	回归输出	2(C) × 1(B) × 1(T)	-

Set Memory Depth (M) to 5, with an initial learning rate of 0.005 and a maximum training frequency of 15000 (stopping at around 12360 rounds when the validation conditions are met). The fitted NMSE is -46.003dB. After validation, the NMSE of the multi-layer FCN network on the Train Set is -44.859dB and on the Valid Set is -45.179dB.

The obtained AM-AM image is as follows:





2.7 Temporal Convolutional Network (TCN)

Based on the comprehensive training speed and model performance, Multi layer FCN is basically the optimal result we have explored. However, the group members also explored several other networks in their spare time. Although the model performance is not very ideal, it can still be used as a reference, as shown below.

TCN is a convolutional neural network specifically designed for processing time-series data, aimed at replacing traditional RNNs such as LSTM and GRU in sequence modeling tasks. Its core feature is to achieve efficient modeling of time series by improving convolution operations. Compared to traditional RNNs, TCNs can perform parallel computing (simultaneously processing the entire sequence) and effectively model long-range temporal relationships through multi-level dilated convolutions, while also being more stable in training.

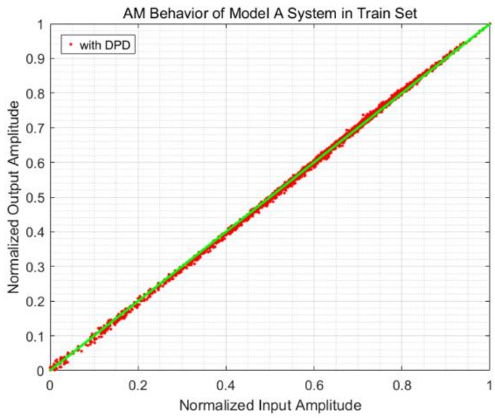
Choose Adam as the optimization algorithm, set Memory Depth (M) to 4, and set the maximum training times to 10000 rounds (stopping after about 6000 rounds when the validation conditions are met). Construct the TCN network provided by the system default in MATLAB (as shown below), and the fitted NMSE result is -42.507dB.

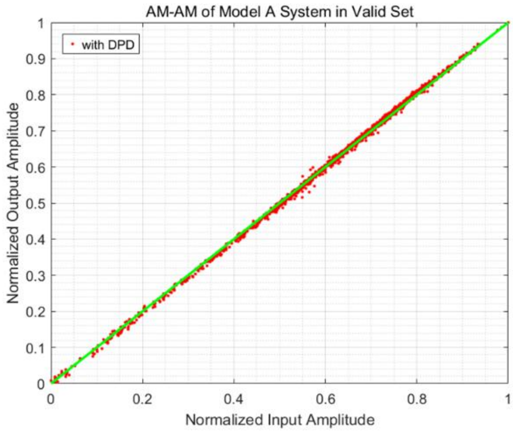
After verification, the NMSE of the CNN-LSTM network on the Train Set is -38.627dB, and on the Valid Set it is -37.308dB.

使用Model A，系统的NMSE(Train)仿真结果为 -38.626561 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -37.308487 dB

The obtained AM-AM image is as follows:





As can be seen, the results of the model were not as expected. On the one hand, this is because the model was set up in the MATLAB toolbox without any specific adjustments made to the task (also because I don't know how to adjust it); On the other hand, due to the large number of parameters, the training cost is high, and group members do not have sufficient time to adjust hyperparameters to achieve better results. Additionally, it has been noted that the model also suffers from significant overfitting issues.

2.8 Transformer

Transformer is different from RNN in that it processes the entire sequence step by step through matrix operations, greatly improving training speed. Its core is the self attention mechanism, which directly models the relationship between any two elements in the sequence, regardless of the distance, naturally solving the long-distance dependency problem, and can be stacked by multiple encoder/decoder blocks with the same structure, making it easy to expand the model capacity.

Still choose Adam as the optimization algorithm, set Memory Depth (M) to 4, and set the maximum training times to 8000 rounds (stopping when the validation conditions are met around 1000 rounds). Construct a Transformer architecture network (as shown below), and its fitted NMSE result is -28.096dB.

	名称	类型	激活	可学习参数大小
1	input 序列输入: 10 个维度	序列输入	10(C) × 1(B) × 1(T)	-
2	pos-emb 具有 10 个输出通道且最大位置为 8192 ...	位置编码	10(C) × 1(B) × 1(T)	Weights 10 ×
3	add 加法: 将 2 个输入按元素相加	加法	10(C) × 1(B) × 1(T)	-
4	selfattention_1 具有 4 个头和 128 个键和查询通道的自...	自注意力	10(C) × 1(B) × 1(T)	QueryWeig… KeyWeights ValueWeig… OutputWei… QueryBias KeyBias ValueBias OutputBias
5	layer @(x)x	函数	10(C) × 1(B) × 1(T)	-
6	fc1 32 全连接层	全连接	32(C) × 1(B) × 1(T)	Weights 32 × Bias 32 ×
7	fc_relu ReLU	ReLU	32(C) × 1(B) × 1(T)	-
8	fc2 2 全连接层	全连接	2(C) × 1(B) × 1(T)	Weights 2 × Bias 2 ×
9	output mean-squared-error	回归输出	2(C) × 1(B) × 1(T)	-

After verification, the NMSE of the Transformer architecture network on the Train Set is -25.923dB, and the NMSE on the Valid Set is -26.051dB.

使用Model A，系统的NMSE(Train)仿真结果为 -25.922682 dB

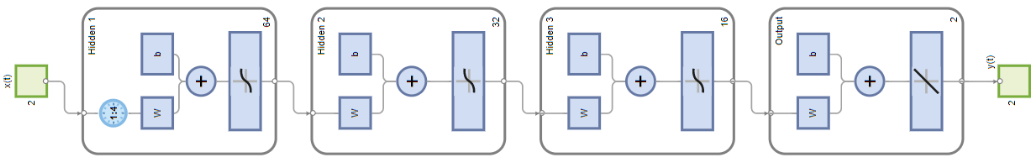
使用Model A，系统的NMSE(Valid)仿真结果为 -26.051169 dB

It can be seen that the Transformer architecture performs poorly in DPD scenarios and is the only model among many that achieves negative optimization. The core reason for the failure of training networks is mainly due to limited time and knowledge, inadequate transformer architecture settings of team members, and incorrect use of certain networks or activation functions, resulting in such results. Secondly, it can be seen that transformers are commonly used for complex problems and scenarios with large amounts of data, so they may not be suitable for DPD optimization scenarios.

2.9 Time Delay Net

Time delayed neural network is a neural network structure specifically designed for processing time series prediction in MATLAB. Its core idea is to capture temporal dependencies in a sequence by introducing time delay, and it is suitable for modeling univariate or multivariate time series.

Select LM as the optimization algorithm, set Memory Depth (M) to 4, and set the maximum training times to 4000 rounds (stopping at around 600 rounds when the validation conditions are met). Construct a time delayed neural network (as shown below), and its fitted NMSE result is -44.655dB。

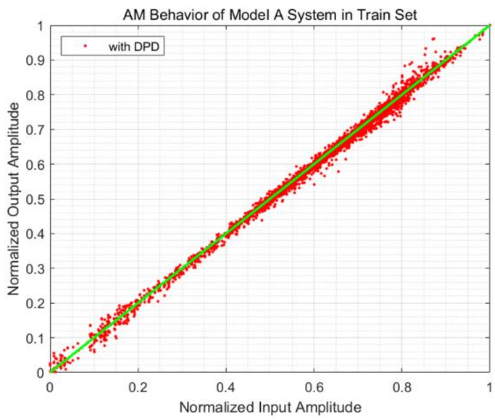


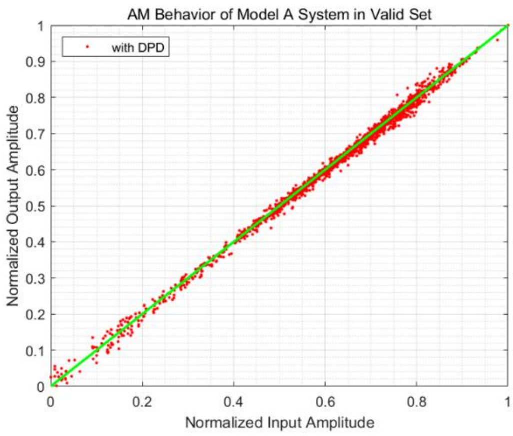
After verification, the NMSE of the time delayed neural network on the Train Set is -30.378dB, and on the Valid Set it is -30.719dB.

使用Model A，系统的NMSE(Train)仿真结果为 -30.378280 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -30.719442 dB

The obtained AM-AM image is as follows:





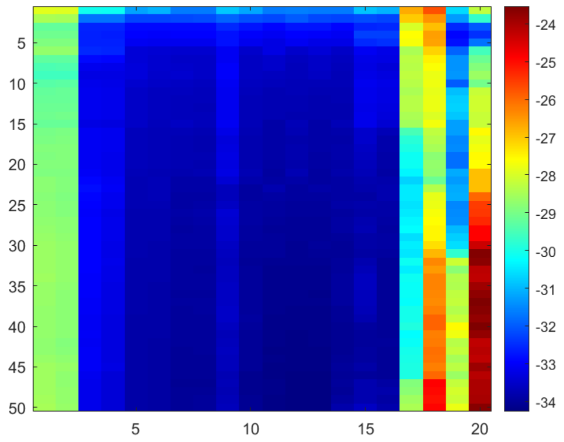
2.10 Volterra

Out of curiosity, we also tried the Volterra model commonly used in previous experiments.

The core function is shown below:

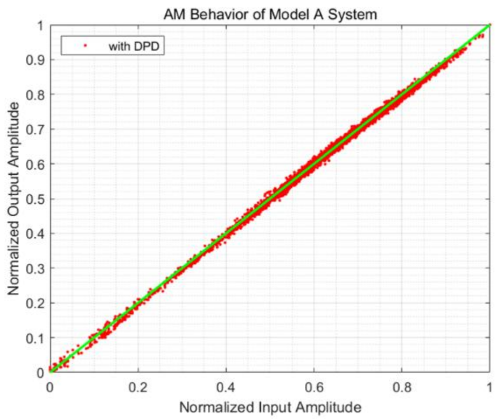
```
function DPD_coefficient=Train_Voltrra(M,N,dim,input,output)
output=output/max(abs(output));
input=input/max(abs(input));
DPD_input_matrix = zeros(dim, (M+1)*N);
for i=1:M+1
    for j= 1:N
        DPD_input_matrix(:,N*(i-1)+j) = Delay(output,i-1).*((Delay(output,i-
1)).^(2*j-2));
    end
end
DPD_coefficient=DPD_input_matrix\input(:);
end
```

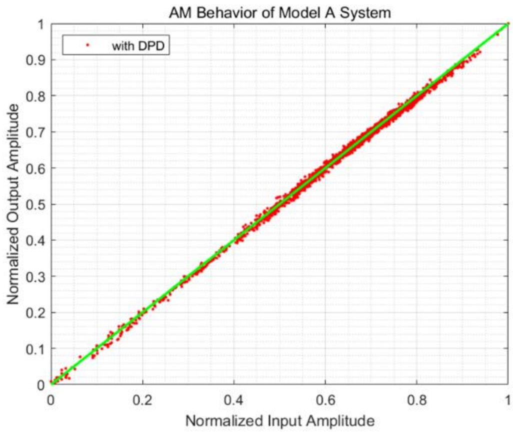
In order to determine the optimal parameters M and N (order), we conducted parameter searches on M∈[1,50]和 N∈[1,20], and the resulting heatmap is as follows:



The conclusion drawn is that when N=13, the larger the M, the better the performance. However, if M is too large, it will lead to overfitting. Finally, we take N=13 and M=100 to obtain a fitted NMSE of -34.707dB. After verification, the NMSE of the Volterra model on the Train Set is -32.241dB and on the Valid Set is -32.349dB.

The obtained AM-AM image is as follows:





It can be seen that the accuracy of Volterra models is limited, making it difficult to effectively optimize complex models.

2.11 Comparison and Summary

We compared the performance of different methods to improve the degree of system linearization and obtained the following table:

方法	拟合NMSE(dB)	Train Set NMSE(dB)	Valid Set NMSE(dB)
Original	---	-26.743	-26.794
FCN(M=5)	-35.315	-35.276	-35.324
LSTM(M=5)	-42.341	-38.654	-38.929
Bi-LSTM(M=5)	-44.254	-33.932	-33.205
GRU(M=5)	-47.393	-43.417	-43.694
CNN-LSTM(M=5)	-38.850	-36.615	-36.823

Multi-layer FCN (M=5)	-46.003	-44.859	-45.179
TCN(M=4)	-42.507	-38.627	-37.308
Transformer(M=4)	-28.096	-25.923	-26.051
Timedelay(M=4)	-44.655	-30.378	-30.719
Volterra(M=100, N=13)	-34.707	-32.241	-32.349

Note: Among all networks, only Timedelay used the Levenberg Marquardt optimization algorithm, while the other networks used the Adam optimization algorithm.

After comparison, it was found that the performance of Multi layer FCN was the best, with the shallowest degree of overfitting (among the better performing models), and the training time was also moderate. So we decided to use Multi layer FCN as the main method for building more complex models, with other methods as supplements.

3. The selection of optimization algorithm

In further exploration, we found that choosing different optimization algorithms can lead to significant differences in results. We will take Multi layer FCN as an example to explore the impact of different optimization algorithms on DPD modelling.

3.1 Adaptive Moment Estimation (adam)

Adam is a commonly used adaptive learning rate optimization algorithm in neural network optimization, which combines the advantages of Momentum and RMSprop algorithms: it can accumulate historical gradient directions through exponential weighted averaging to accelerate convergence; Adaptively adjusting the learning rate for different parameters to alleviate the problem of gradient vanishing/exploding, it performs well in scenarios with large amounts of data and multiple parameters.

The result of using Adam optimization algorithm is the same as 2.7, and will not be repeated here.

3.2 Stochastic Gradient Descent with Momentum (sgdm)

SGDM is a classic stochastic gradient descent (SGD) combined with momentum (Momentum), which accelerates convergence and reduces oscillations by accumulating historical gradient directions. The momentum factor can be understood as the "degree of memory" of past update directions. A higher momentum factor will help the optimization process maintain the previous direction to a certain extent, avoiding frequent changes in direction during the optimization process and thus accelerating convergence speed.

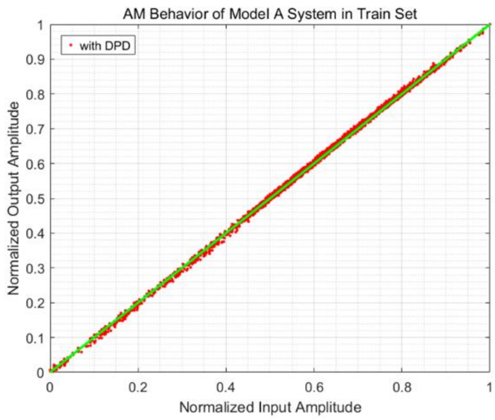
We chose sgdm as the optimization algorithm, set Memory Depth (M) to 5, adjusted the learning rate from 0.005 to 0.01, set the momentum factor to 0.9, and constructed a Multi layer FCN network with the same 2.7 training epochs. The fitted NMSE result was -42.238dB.

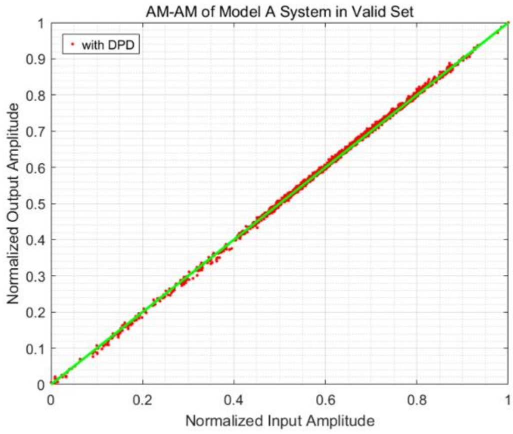
After verification, the NMSE of the Multi layer FCN network using sgdm as the optimization algorithm on the Train Set is -41.339dB, The NMSE on the Valid Set is -41.572dB.

使用Model A，系统的NMSE(Train)仿真结果为 -41.338579 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -41.571677 dB

The obtained AM-AM image is as follows:





As can be seen, when using SGDM, its iteration is faster and requires fewer iterations, but correspondingly, the iteration process is more unstable, and its validation conditions need to be relaxed to avoid ending the training prematurely.

3.3 Root-Mean-Square Propagation (rmsprop)

RMSProp is an adaptive learning rate optimization algorithm, whose main idea is to normalize the gradient to solve the problem of fixed learning rate in the gradient descent process. Its core is to adjust the learning rate by exponentially weighted moving average of the square of the gradient.

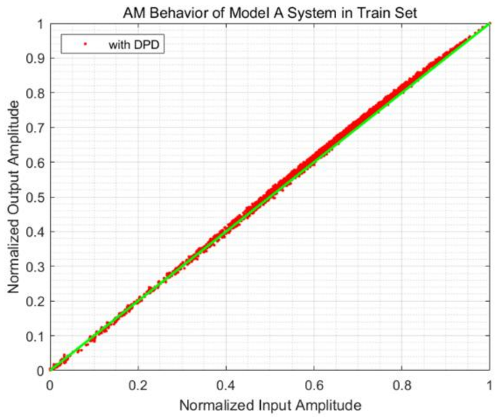
We chose rmsprop as the optimization algorithm, set Memory Depth (M) to 5, adjusted the learning rate from 0.005 to 0.001, set the momentum factor to 0.9, and set the maximum training times to 15000 rounds (stopping at around 11900 rounds when the validation conditions are met). We constructed a Multi layer FCN network with the same 2.7, and the fitted NMSE result was -31.994dB.

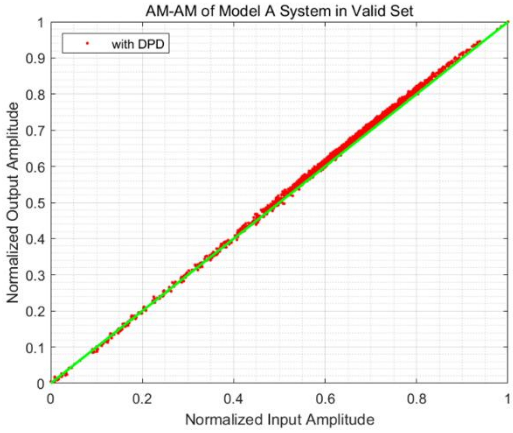
After verification, the NMSE of the Multi layer FCN network using rmsprop as the optimization algorithm on the Train Set is -33.357dB， The NMSE on the Valid Set is -33.364dB.

使用Model A，系统的NMSE(Train)仿真结果为 -33.356692 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -33.363746 dB

The obtained AM-AM image is as follows:





As can be seen, when using rmsprop, its iterations are slower and require more iterations. Under similar conditions, it requires more resources and time compared to other optimization algorithms, and the results are not ideal, making it a poor choice.

3.4 Levenberg-Marquardt (trainlm)

The Levenberg Marquardt algorithm is an optimization algorithm used to solve nonlinear least squares problems. It combines the advantages of gradient descent and Gaussian Newton method. When the parameter is far from the optimal solution, it exhibits characteristics similar to gradient descent and can steadily move towards the direction of error reduction; And when the parameters approach the optimal solution, it exhibits characteristics similar to the Gauss Newton method, which can converge quickly. And the Levenberg Marquardt algorithm is very effective in handling small and medium-sized neural network training tasks, especially in tasks such as function approximation, pattern recognition, and classification, making it perfect for DPD processing scenarios.

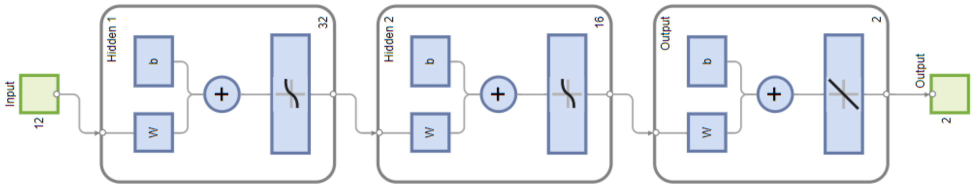
Due to MATLAB's Deep Network Designer not supporting the use of trainlm as an optimization algorithm, it is necessary to switch to the traditional neural network toolbox.

We chose trainlm as the optimization algorithm, set Memory Depth (M) to 5, and did not need to set the learning rate. We constructed a Multi layer FCN network with the same 2.7, and the fitted NMSE result was -54.500dB. After verification, the NMSE of the Multi layer FCN network using trainlm as the optimization algorithm was -45.662dB on the Train Set and -46.046dB on the Valid Set.

```
使用Model A，系统的NMSE(Train)仿真结果为 -45.661668 dB
```

```
使用Model A，系统的NMSE(Valid)仿真结果为 -46.045967 dB
```

It can be seen that the trainlm algorithm iterates very quickly and has excellent performance, which also leads to a serious overfitting problem. So we reduced the amount of input data and streamlined the number of hidden layers, and designed the network as follows:

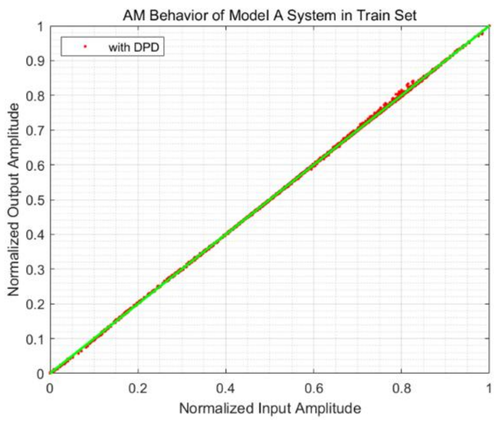


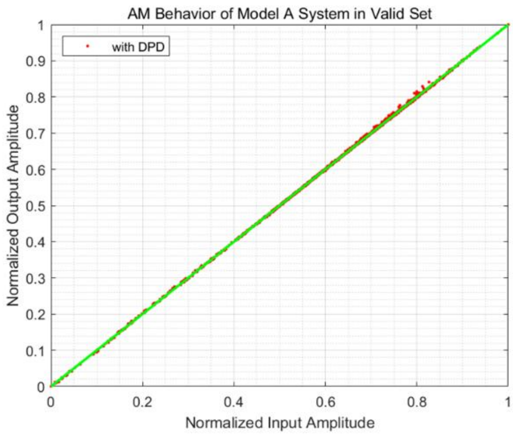
The fitted NMSE result is -54.067dB. After verification, the NMSE of the Multi layer FCN network using trainlm as the optimization algorithm is -49.004dB on the Train Set and -46.381dB on the Valid Set.

使用Model A，系统的NMSE(Train)仿真结果为 -49.004223 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -49.381393 dB

The obtained AM-AM image is as follows:





Compared to other optimization algorithms, trainlm has good stability, faster training time, and better performance. The only possible regret is that Jacobi matrices were used during its training process and do not support computation on GPUs.

3.5 Scaled Conjugate Gradient (trainscg)

The MATLAB traditional neural network toolbox also supports optimization algorithms such as scaling conjugate gradient method and Bayesian regularization, which we will further explore.

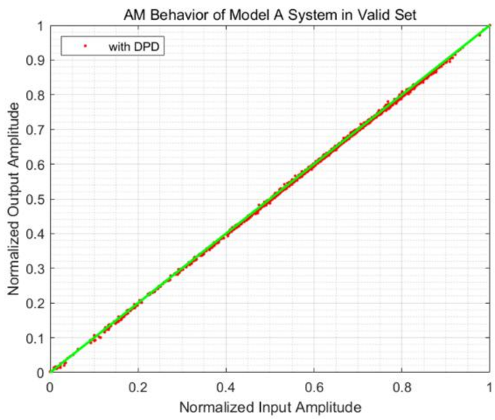
The Scaled Conjugate Gradient algorithm is an optimization algorithm used for training neural networks, proposed by Martin F. Moller in 1993. It combines the efficiency of conjugate gradient method and adaptive step size adjustment strategy, making it particularly suitable for solving medium-sized nonlinear optimization problems (such as parameter training of neural networks).

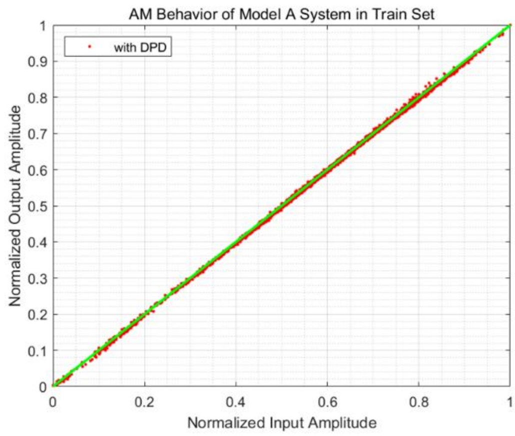
We chose trainscg as the optimization algorithm and set Memory Depth (M) to 5 without setting the learning rate. We constructed a simple Multi layer FCN network similar to 3.3, and the fitted NMSE result was -41.556dB. After verification, the NMSE of the Multi layer FCN network using trainlm as the optimization algorithm was -41.540dB on the Train Set and -41.833dB on the Valid Set.

使用Model A，系统的NMSE(Train)仿真结果为 -41.540421 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -41.832808 dB

The obtained AM-AM image is as follows:





3.6 Bayesian Regularization (trainbr)

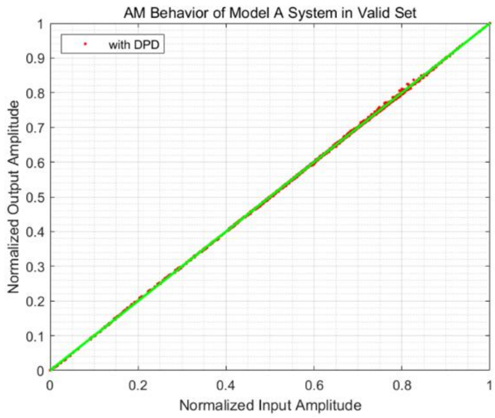
The Bayesian regularization algorithm is a high-order optimization method used in neural network training to prevent overfitting, combining the Levenberg Marquardt algorithm with Bayesian regularization techniques. Being able to suppress model complexity, improve generalization performance, and prevent overfitting through regularization; Moreover, it is suitable for small-scale data and can maintain high prediction accuracy even with limited training data. This is very consistent with our DPD optimization scenario.

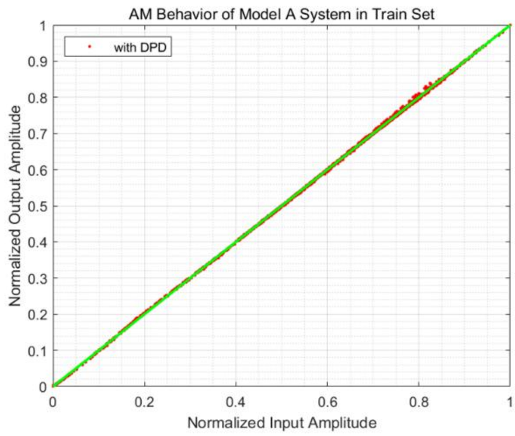
We chose trainscg as the optimization algorithm and set Memory Depth (M) to 4 without setting the learning rate. We constructed a simple Multi layer FCN network similar to 3.3, and the fitted NMSE result was -52.952dB. After verification, the NMSE of the Multi layer FCN network using trainlm as the optimization algorithm was -49.211dB on the Train Set and -49.750dB on the Valid Set.

使用Model A，系统的NMSE(Train)仿真结果为 -49.210931 dB

使用Model A，系统的NMSE(Valid)仿真结果为 -49.749538 dB

The obtained AM-AM image is as follows:





It can be seen that the Bayesian regularization algorithm performs slightly better than the LM algorithm, and the overfitting problem is alleviated. The only drawback is that its training time is much longer than the LM algorithm.

3.7 Comparison and Summary

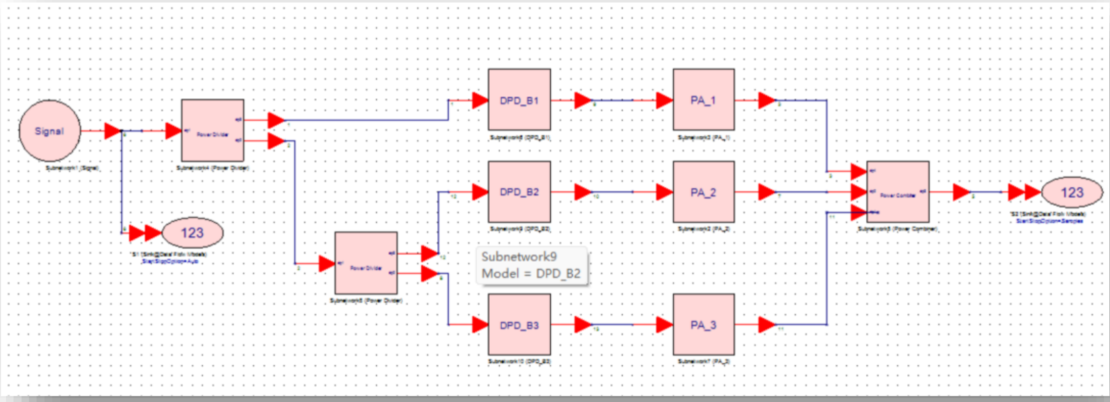
优化算法	拟合NMSE(dB)	Train Set NMSE(dB)	Valid Set NMSE(dB)
adam	-46.003	-44.859	-45.179
sgdm	-42.238	-41.339	-41.572
rmsprop	-31.994	-33.357	-33.364
trainlm with complex network	-54.500	-45.662	-46.046
trainlm with simple network	-54.067	-49.004	-49.381
trainscg with simple network	-41.556	-41.540	-41.833
trainbr with simple network	-52.952	-49.211	-49.750

4. Composite Model

Inspired by the circuit structure of the system, we found that DPD can be used not only for overall optimization, but also for optimizing three PAs separately. Therefore, we still propose the following two composite models based on the optimal network Multi layer FCN mentioned above.

4.1 3-way Optimization Model

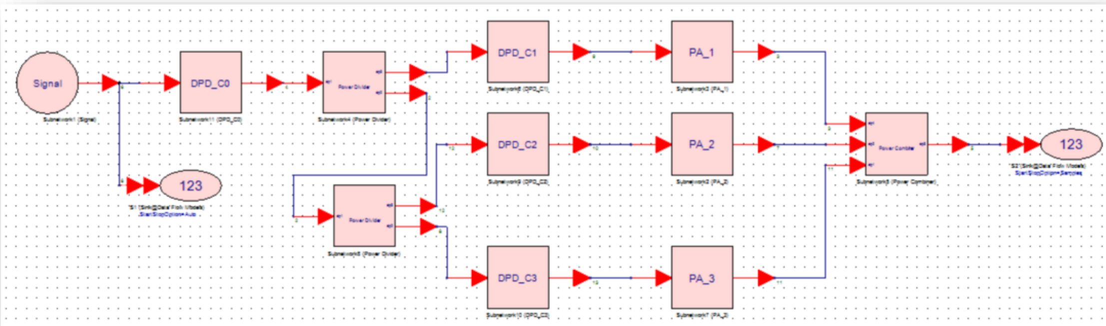
Firstly, attempt to optimize using DPD on each of the three PAs. The network structure is shown in the figure below:



Each DPD constructed the same network structure as 3.6, resulting in an NMSE of -41.131dB on the training set and -41.240dB on the validation set, but the effect was not as good as expected.

4.2 3-way Optimization Plus Overall Optimization Model

After attempting 4.1, we came up with the idea of conducting a three-way optimization first, followed by the aforementioned overall optimization. Therefore, we propose a network structure that uses four DPDs, as shown in the following figure:

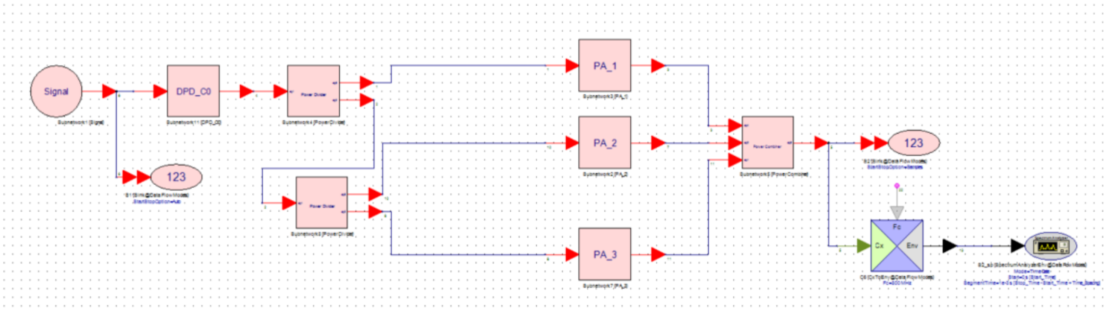


We also constructed a Multi layer FCN network for each DPD, resulting in an NMSE of -49.155dB on the training set and -49.693dB on the validation set, which is a slight improvement from 4.2 but still not as good as the optimal result of a single overall optimization.

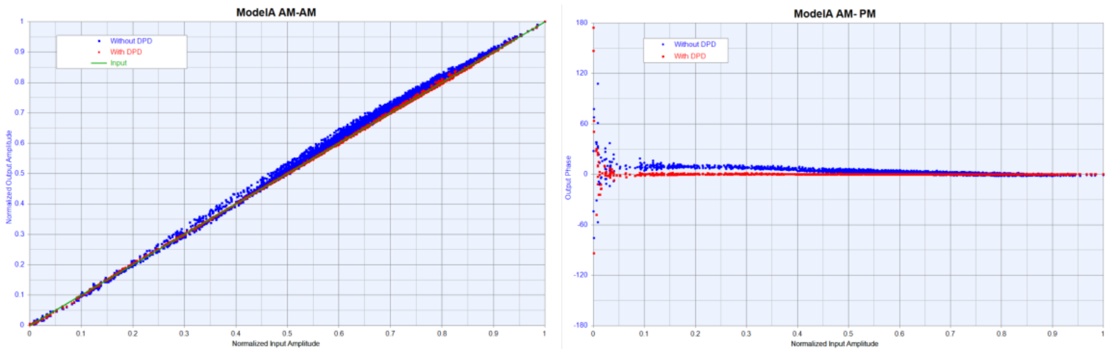
Moreover, we have found that the training results using this model are often unstable and take a long time.

5. The Best Model

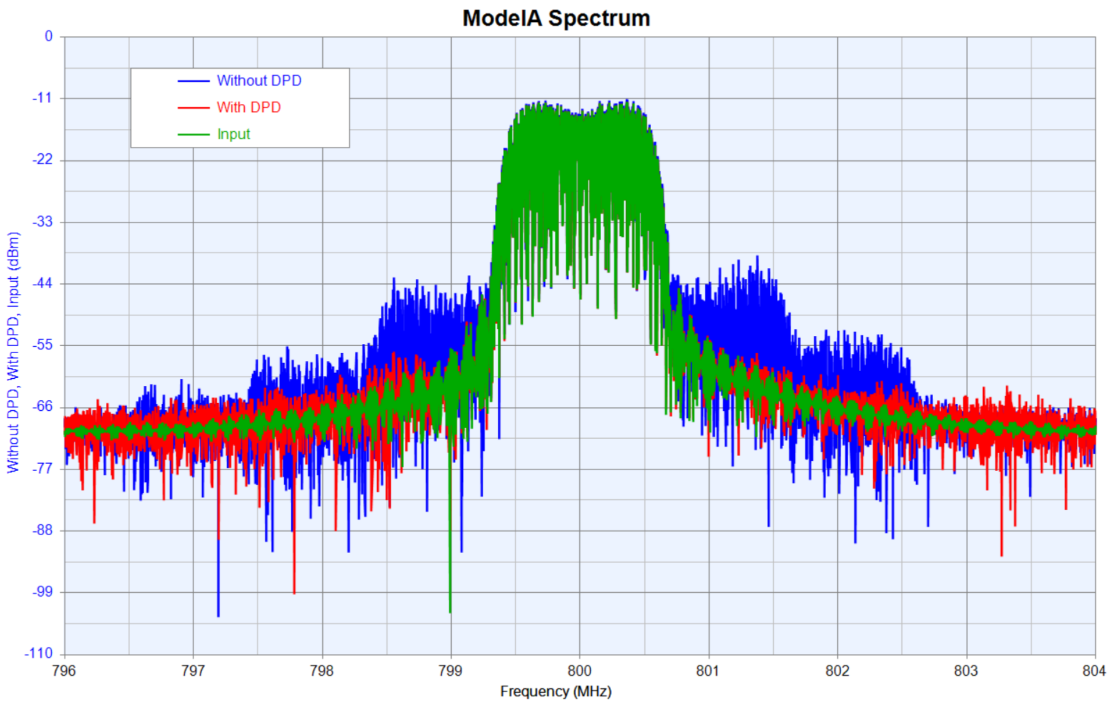
Based on the optimal network structure and optimization algorithm mentioned above, we ultimately adopt ModelA with single overall optimization and use a Multi layer FCN network structure. Among them, the Multi layer FCN uses cascade forward net ([36,24]), the algorithm uses' trainbr ', adopts data segmentation (2100 training set+1900 validation set), and increases the regularization coefficient by 0.0001. The structure of the system is shown in the following figure:



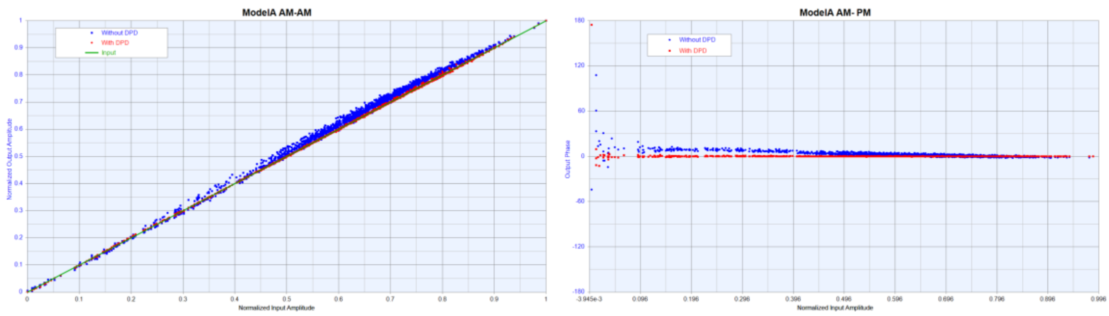
On the training set, the NMSE is -49.667dB, and the normalized input-output AM-AM and AM-PM images are shown in the following figure. It can be seen that the amplifier system optimized by DPD has significantly improved in both amplitude and phase characteristics.



The input, output before DPD optimization, and output after DPD optimization are shown in the following figure. It can be seen that the output spectrum after DPD optimization is more accurate in reproducing the frequency spectrum of the input signal.

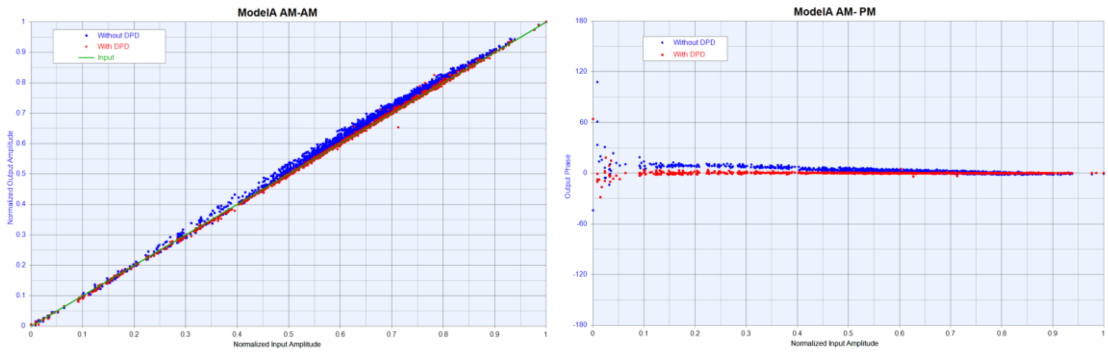


The NMSE on the test set is -49.907dB, and the AM-AM and AM-PM images are shown below, indicating that this model has also achieved considerable performance on the validation set.

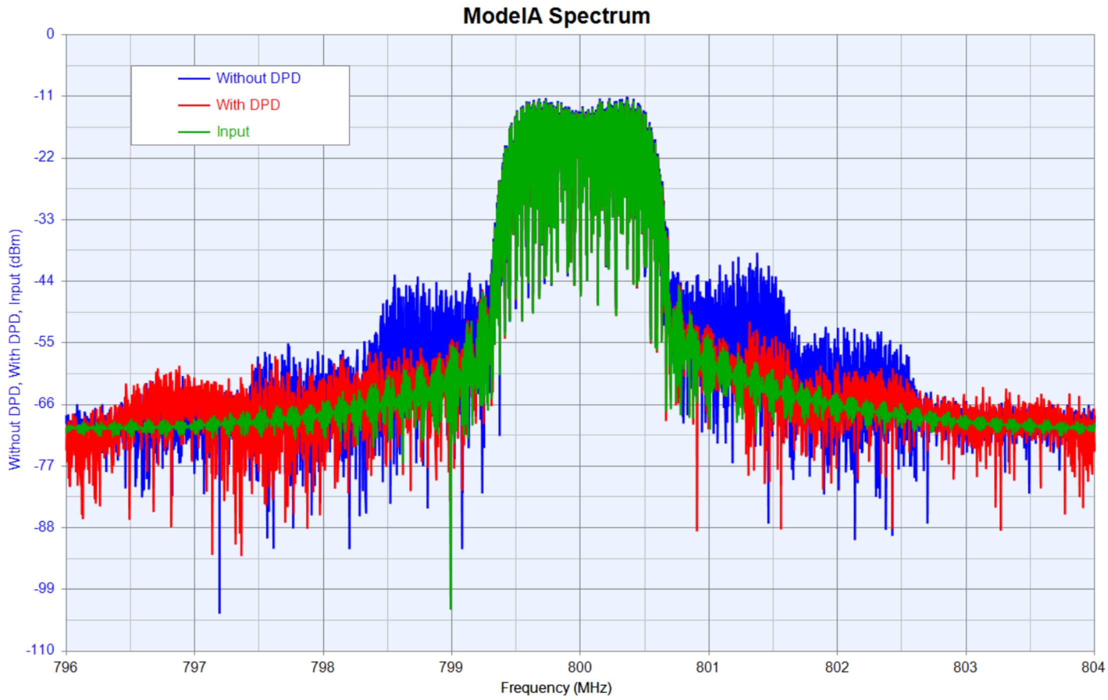


Due to the good performance of GRU with a memory depth of 5 in the previous test, we will continue to use ModelA with single overall optimization and GRU network structure for further testing.

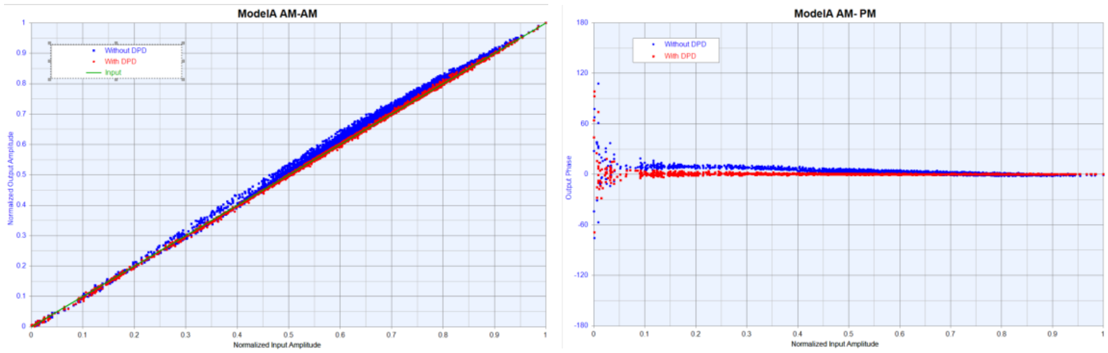
On the training set, the NMSE is -43.9dB, and the normalized input-output AM-AM and AM-PM images are shown in the following figure.



The output spectra before and after DPD optimization are shown in the following figure, indicating that the output spectrum after DPD optimization is more accurate in reproducing the frequency spectrum of the input signal.



The NMSE on the test set is -43.631dB, and the AM-AM and AM-PM images are shown below, indicating that this model has also achieved considerable performance on the validation set.



After comparison, the performance of Model A implemented by GRU is weaker than that implemented by FCN. The conclusion of this experiment is that the best performing model is Model A using FCN, which has a NMSE of up to -49dB.

6. Conclusion

This experiment focuses on modeling the digital pre distortion (DPD) of power amplifiers (PA), and explores effective methods to improve the linearization performance of PA by comparing various neural network architectures and optimization algorithms. The experimental results indicate that there is no positive correlation between model complexity and performance, and the key to improving DPD performance is to choose a reasonable network structure and optimization strategy.

Among numerous networks, the fully connected network (Multi layer FCN) performs the best in overall performance, with a validation set NMSE of -49.907 dB and the low degree of overfitting. However, although complex models such as Bi LSTM and Transformer have higher fitting accuracy, they suffer from severe overfitting due to the large number of parameters, and their actual application performance is not as good as simple models. Traditional Volterra models have limited nonlinear modeling capabilities and significantly lag behind neural network methods in performance.

In terms of optimization algorithms, Bayesian regularization (trainbr) combined with Multi layer FCN performs the best, with a validation set NMSE of -49.750 dB, effectively balancing fitting ability and generalization performance. The Levenberg Marquardt (LM) algorithm has the highest fitting accuracy and the training time is short, but it needs to simplify the network structure to avoid overfitting. Adaptive algorithms, such as Adam, exhibit a balance between training efficiency and stability, making them suitable for medium-sized data scenarios.

In this experiment, we encountered many challenges, such as the complexity of parameter adjustment and the potential significant impact of small changes in network hyperparameters (such as memory depth and hidden layers) on performance; Some algorithms, such as LM, cannot utilize GPU acceleration due to their dependence on Jacobian matrices, which limits their application in large-scale scenarios.

This experiment not only validates the potential of neural networks in DPD modeling, but also provides important references for model selection and optimization in engineering practice.

Future work can further combine hardware characteristics with real-time requirements to promote the application of DPD technology in practical communication systems.

7. A brief summary of this experiment

7.1 Member 1：04022212钟源

In this lab, my partner and I use Neural Network and Volterra model to build a DPD model. We've tried FCN, LSTM, Bi-LSTM, GRU, CNN-LSTM, TCN, Transformer and so on. After so many tries, we finally realized that "the less is the more": simple network might bring better output. In fact, adjusting the parameters of a network is a very painful process, because when you solve one problem, another problem emerges, which will affect performance of the entire model. Anyway, this lab assignment is a valuable experience for us to learn system modelling and neural network.

7.2 Member 2：04022114贾岩森

In this lab, my teammate and I designed a DPD model for power amplifiers by evaluating neural networks (e.g., FCN, LSTM, GRU) and traditional Volterra series. We observed that complex architectures like Bi-LSTM suffered from severe overfitting despite high fitting accuracy, while simpler Multi-layer FCN achieved better generalization with NMSE of -49.75 dB on validation data. Although advanced models like the Transformer showed potential, practical implementation required adjustments beyond our current scope. This experience emphasized the importance of simplicity and regularization in system modeling.