ARTIFICIAL NEURAL NETWORK FOR CROSSTALK PREDICTION IN STRIPLINE TRANSMISSION LINES

KONG CHUN LEI

UNIVERSITI SAINS MALAYSIA

2018

ARTIFICIAL NEURAL NETWORK FOR CROSSTALK PREDICTION IN STRIPLINE TRANSMISSION LINES

by

KONG CHUN LEI

Thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering (Electronic Engineering)

JUNE 2018

ACKNOWLEDGEMENTS

First and foremost, I would like to express my greatest gratitude to my final year project supervisor Dr. Patrick Goh Kuan Lye for his guiding and giving constant encouragement throughout this research. I am very grateful to him for providing me with a clear theory on my project and giving some useful suggestions to improve the project. Besides, he is willing to spend his precious time and effort to follow up my progress and discuss the project flow. It was a tough work to complete this project if without his guidance and teaching.

Next, my sincere thanks to Mr. Goay Chan Hong, a postgraduate student who assisted me on the problems encountered in Advanced Design System (ADS) and Artificial Neural Network (ANN) throughout the project. I appreciate him for his patient and willingness to support me.

Besides, I would like to thank my examiner, Dr. Intan Sorfina for giving the positive feedbacks and comments on the project.

Lastly, we would like to thank my lovely parents and friends who provided me unlimited support in finalizing this project within the limited time frame.

Table of Contents

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADS	Advanced Design System
ANN	Artificial Neural Network
CAD	Computer-Aided Design
CM	Common Mode
CPU	Central Processing Unit
DOE	Design of Experiment
DUT	Design Under Test
EM	Electromagnetic
EMC	Electromagnetic Compatibility
FEXT	Far End Crosstalk
LMS	Least Mean Square
MLP	Multilayer Perceptron
NEXT	Near End Crosstalk
RDSI	Random Displacement Spline Interpolation
RF	Radio Frequency
RMD	Random Multiple Displacement
SCLIN	Libra-Edged Coupled line in stripline
SPICE	Simulation Program with Integrated Circuit Emphasis
TEM	Transverse Electromagnetic Transmission Line
TERM	Termination Port
TLM	Transmission Line Matrix
trainbfg	BFGS Quansi-Newton Backpropagation
traincgb	Conjugate Gradient with Powell/Beale Restarts

traincgf	Fletcher-Powell Conjugate Gradient
traincgp	Polak-Ribiére Conjugate Gradient
trainlm	Levenberg-Marquardt Backpropagation
trainscg	Scaled Conjugate Gradient
trainrp	Resilient Backpropagation
trainoss	One Step Secant
VAR	Variables and Equations Component

RANGKAIAN NEURAL UNTUK MENJANGKAKAN CROSSTALK DALAM TALIAN PENGHANTARAN STRIPLINE

ABSTRAK

Crosstalk boleh menyebabkan masalah gangguan elektromagnet yang serius. Oleh itu, menjangkakan crosstalk dalam peringkat reka bentuk awal adalah penting. Beberapa kaedah pemodelan konvensional seperti RDSI dan SPICE telah dibentangkan untuk menganggarkan crosstalk dalam talian penghantaran. Walaubagaimanapun, kaedah ini memerlukan penggunaan memori CPU yang besar dan tempoh latihan yang panjang. DOE digunakan untuk memilih data latihan secara efisien dan mengurangkan bilangan simulasi EM dalam “Advanced Design System” (ADS). Momentum EM Simulator digunakan untuk mengekstrak S-parameter dari stripline dengan parameter reka bentuk yang berbeza dan menghasilkan dataset yang efisien. Matlab Neural Network Toolbox digunakan untuk mencipta model rangkaian neural. Model rangkaian neural dilatih untuk mempelajari pencirian data bagi menjangkakan crosstalk dalam stripline. Akhir sekali, model rangkaian neural disahkan dengan membandingkan keputusan simulasi dan hasil yang diramalkan dari ADS dan ANN. Penilaian prestasi menunjukkan bahawa anggaran crosstalk dalam stripline mencapai 99.9% dalam masa latihan 0.2810s. Kesimpulannya, hasil kajian ini mengesahkan bahawa ANN adalah berkesan dalam ramalan crosstalk stripline.

ARTIFICIAL NEURAL NETWORK FOR CROSSTALK PREDICTION IN STRIPLINE TRANSMISSION LINES

ABSTRACT

Crosstalk can cause serious electromagnetic interference problem and crosstalk prediction in the early design stage is important. Several conventional modeling methods such as RDSI and SPICE have previously presented to predict crosstalk in non-uniform transmission lines and it needs large CPU memory consumption and long simulation time. DOE is applied to efficiently select training data and reduce the number of EM simulations in the Advanced Design System (ADS). Momentum EM Simulator is used to extract S-parameters from coupled stripline with different design parameters and generated an efficient dataset. Matlab Neural Network Toolbox is used to create neural network models. Neural network models are trained to learn the characterization and behavior of data for crosstalk estimation in stripline. Lastly, the neural model is validated by comparing the simulated results and predicted results from ADS and ANN. The performance evaluation shows that the crosstalk prediction in stripline achieved 99.9% with training time of 0.2810s. In conclusion, this verified that the ANN is effective in the stripline crosstalk prediction.

CHAPTER 1
INTRODUCTION

1.1 Research Background

Microwave Device such as antennas, couplers, filters, power divider and etc form from transmission line with entire device existing as a pattern of metallization on the substrate is widely used in today's digital era. The demand of miniaturized microwave/RF integrated circuit, frequency increases in the state-of-the-art digital system, led to crosstalk. It is due to mutual coupling of the close proximity of parallel interconnect lines. Crosstalk caused unintentional signal integrity problems and electromagnetic interference in the electronic devices, especially between the long parallel lines.

Crosstalk is a common EMC problem and it is vital to predicting the crosstalk in the early design stage to prevent unintended malfunction on electronic equipment. The traditional analysis method of crosstalk among transmission lines depends on TLM equations and the distributing parameters such as SPICE model. Unfortunately, the complexity of coupling relations among transmission lines causes a significant challenge to compute analytical formulas to predict crosstalk. Then, Rainal computed the formulas for estimating near end and far end induced crosstalk voltages at the driver and receiver ends of two coupled, symmetric, homogeneous dielectric stripline. The far end crosstalk coefficients, K_{fe} and near end crosstalk coefficient, K_{ne} with their characteristic impedance are derived as

$$K_{fe} = \frac{1}{2}\left(Z_o C_m - \frac{L_M}{Z_o}\right) \tag{1.1}$$

$$K_{ne} = \frac{V_p}{4}\left(Z_o C_m + \frac{L_M}{Z_o}\right) \tag{1.2}$$

Where$V_p$is the propagation velocity, Z_o is the characteristic impedance of each line, C_m and L_M are the mutual capacitance and mutual inductance between the lines respectively. K_{fe} and K_{ne} used to estimate the crosstalk voltages as fractions of the driver output voltage. The crosstalk can be predicted accurately under some assumptions such as coupled and homogeneous dielectric stripline. However, Rainal's formula only applicable to modal solution for the stripline with only two coupled lines. If more than two coupled line, the crosstalk coefficient formulas cannot predict the crosstalk with high degree of accuracy [4].

Computationally electromagnetics algorithms applied to predict crosstalk when computer science developed rapidly. Some computer-aided design (CAD) tools such as RDSI and SPICE are used to calculate crosstalk in multiconductor. However, these CAD tools need large CPU memory consumption and long simulation time to extract transmission line parameters and predict crosstalk.

The performance of ANN relies on the size and training dataset. Accurate and rapid neural network model can be created in an efficient dataset because it spreads the sample points more evenly across all possible values based on the original multidimensional distribution. Many previous works applied DOE to collect dataset and generate the neural network model of transmission line structures[5]. DoE is widely used to study the correlation between the involved design factors, and affecting the outputs of the process. In addition, it can be used to determine the optimum setting of design parameters to achieve the desired output[6].

In this work, stripline training parameters are optimized by DOE method to choose training data efficiently and point out the sensitive factors in the design that have greatest effect on crosstalk. Artificial Neural Network (ANN) algorithm is applied to predict crosstalk with varying design parameters in the stripline. Neural modeling consists of high nonlinear structure, which able to model the nonlinear relation between data sets. The neural network is trained to model the electrical behavior of active and passive components. The trained neural network can be used for high-level modeling and design, provide an accurate result and reduce computational effort and time [7].

1.2 Problem Statement

Crosstalk estimation is a big challenge to model in the transmission line. This is due to the intrinsic behavior of the mutual distance among transmission line. The traditional method of using worst-case analysis to predict system performance can be very pessimistic and caused overdesigned and expensive interconnect systems.

Several statistical modeling methods have previously presented to predict crosstalk in non-uniform transmission lines. Nevertheless, the previous statistical modeling methods to predict crosstalk in transmission lines rely on simulation of many specific non-uniform transmission lines and require significant effort and time. For example, Random Displacement Spline Interpolation (RDSI) method apply Gaussian distribution to distribute all transmission line position with random numbers. Running numerous simulations of the network with a large number of distributed models using conventional circuit simulation tools to meet the specifications. It leads to increasing the modeling time and highly CPU intensive.

Moreover, Monte Carlo analysis used to determine system performance by varying the parameters using their distribution functions. However, Monte Carlo analysis is very time-consuming because of a large number of simulations. Modeling stripline involved many design parameters such as substrate thickness, conductor thickness, the spacing between conductor and conductor width. When the number of physical parameters increases, the complexity of EM stripline circuit increase, the simulation time becomes longer.

In this work, artificial neural network (ANN) approach are proposed to predict the crosstalk in the stripline. It is capable to model nonlinear relationships between channel parameters and system performance to speed up system simulations and provide an accurate result. It also can prevent a small varying in the physical parameters need a complete simulation. Hence, ANN is an efficient approach to be implemented in this project to estimate crosstalk in the stripline accurately.

1.3 Research Objectives

The objectives of this project are:

1. To investigate which design parameters in stripline has the greatest effect on crosstalk.
2. To apply artificial neural network (ANN) for crosstalk prediction in the stripline.

1.4 Scope of Research

In this project, the most popularly used neural network structure, multilayer perceptron (MLP) neural network is implemented. Determining the input parameters and output performance by distinguishing the key parameters and desired performance. The physical parameters in the stripline such as substrate thickness, conductor thickness, the spacing between conductor and conductor width are varying in EM simulation to generate S-parameter. The data collected from the EM simulated result are used as a training and validating data to estimate the performance of the neural network. Training of data is carried out. Test data are used to verify the neural network model and predict crosstalk in the stripline. The limitation of this research is ANN computationally intensive to train as it required a lot of datasets and distributed run-time to train on the very large amount of datasets. Generating numerous datasets for training and testing are a significant challenge because it requires simulating manually one by one in the EM simulator of ADS and time-consuming.

1.5 Thesis Outline

The thesis is divided into five main chapters. Generally, thesis begins with the introduction in Chapter 1. Research Background and Motivation, Problem Statement, Research Objectives, Research Scopes and Thesis Outline are described in Chapter1.

Chapter 2 is Literature Review which described the related work of title relevant background studies. It provides the summary, description, analysis and critical evaluation of the past works.

Chapter 3 covers the methodology of research which included project overall flow and experimental procedure. Various steps involved in the development of neural network model are explained. The selection of neural network architecture and training algorithm is explained. Setting training goal and train the ANN are presented. Calculation and analysis to evaluate the performance of neural network are carried out.

Chapter 4 is Results and Discussions that provide the analysis and discussion of the result and findings. Limitation of the neural network algorithm in stripline crosstalk prediction also discussed in this chapter. Comparative studies about the performance of the neural network in the different optimal solutions and different training algorithms are reviewed.

Chapter 5 contains Conclusion that summarizes the overall of the project and provides some recommendation for future improvements.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Artificial neural networks (ANN) has attracted many attentions in these recent years, an increased number of RF/microwave engineers and researchers have started taking a serious interest in this emerging technology. ANN has been used for electromagnetic-based simulations and optimization of RF and microwave system designs. Neural network are efficient alternatives to the conventional method and analytical methods which are computationally expensive and limited range and accuracy respectively. ANN used to model electromagnetic components to speed up system simulations without sacrificing accuracy and providing a fast answer to the task they learned. In this work, unintentional electromagnetic interference, crosstalk in the stripline are predicted to avoid the failure and abnormalities on the electronic equipment.

2.2 Stripline

Stripline is the earliest form of the planar transmission line and it is transverse electromagnetic (TEM) transmission line [8]. Stripline conductors are sandwiched by the surrounding dielectric material as shown in Figure 2.1. It always explained as a coaxial cable which is non-dispersive and does not consist of the cutoff frequency. Thus, transmission lines in stripline are closely spaced and densely packed, it caused the minimization of microwave frequencies.

The characteristic impedance of the stripline which is a transmission line depends on the relativity permittivity of the substrate, the thickness of the substrate and the width of the stripline. It provided enhanced noise immunity against the propagation of radiated RF emissions, at the slower propagation speeds in contrast to microstrip lines. The wave propagation only happened in the substrate and lead to the equal relative permittivity of the dielectric substrate and effective permittivity of striplines.

The advantages of stripline over microstrip are the shielded signal traces and caused less electromagnetic interference (EMI) and less crosstalk. The disadvantage of stripline compared to the microstrip line is hard to troubleshoot due to no top board access. The cost of stripline are expensive due to more layers and thicker in dimension. The propagation speed is slow due to the higher effective permittivity.

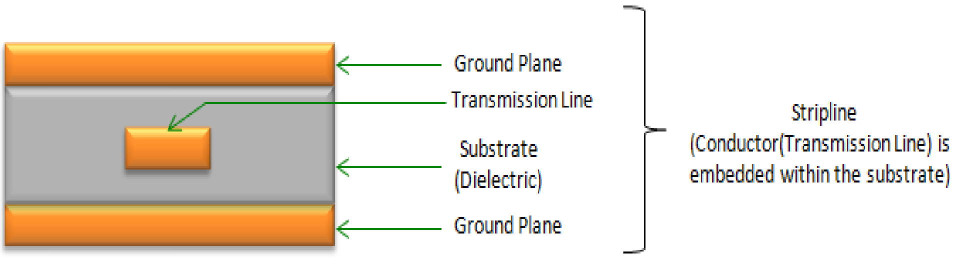

Figure 2.1: Structure of stripline.

2.3 Crosstalk

Mutual coupling exists between the parallel interconnection lines in the same plane of striplines due to close proximity to each other have caused the problem of crosstalk. It is an electromagnetic interference phenomenon between signal lines, which is a common EMC problem. The formula to compute the crosstalk coupling among transmission lines by C.R. Paul [9]. It is hard to compute crosstalk coupling with the analytical formula because of complexity coupling relation among multiconductor due to no rules for arrangements of multiconductor in real condition.

There is two type of crosstalk existed in the channel which is near-end crosstalk (NEXT) and far-end crosstalk (FEXT) as presented in Figure 2.2. The NEXT is a performance parameter measured within a single channel or link. Near end crosstalk defined as the portion of coupled power that flows backwards along the victims in an opposite direction to the forward progress of aggressive signal. NEXT is measured at both ends of the wire and between one wire pair and another in the same cable. It is expressed in decibel (dB) and changed with the frequency of transmission because the higher frequency produced high interference. The FEXT is the mutual coupling between two or more transmitting pairs as the signal propagates from send end to receive end. It is measured by the channel and expressed in decibel (dB) and varies in proportion to trace length.

In the stripline, there is no exist of far end crosstalk but near end crosstalk due to its geometries. FEXT is due to the difference between the relative capacitive coupling and relative inductive coupling between two adjacent lines. The geometry of stripline cause the equal of relative capacitance coupling and relative inductive coupling between two parallel lines. Fridging electric and magnetic effect coupling between two differential pair cause near-end crosstalk in stripline.

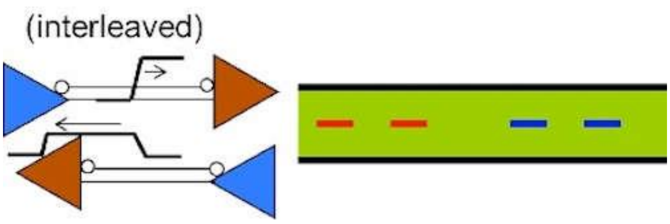

Figure 2.2: Near-end crosstalk (NEXT) in the stripline.

2.4 Previous Statistical Approaches to Predict Crosstalk

Crosstalk is an undesired electromagnetic interference phenomenon among the multiconductor, which produce anomalies and failures on electronic devices. Estimating crosstalk becomes a remarkable challenge and primary important step.

Some statistical methods have previously work on predicting crosstalk. Random Midpoint Displacement (RMD) algorithm presented in [10], the cable bundle is separated into *n* uniform segments and the wire positions are based on a fractal model.

This method allows the changing of the wires meandering degree by the means of a single parameter. Monte Carlo algorithm is proposed to predict a cumulative distribution function for crosstalk via the numerical solution of multiconductor transmission line equations. The position is changed by segmenting the harness along with its length and selecting a random position for each wire within each segment [11]. These two approaches have large discontinuities between adjacent segments are far from the real behavior of a transmission line. Unintentionally resonances of the common mode (CM) along transmission line can cause a large discontinuity.

Random displacement spline interpolation (RDSI) method is applied to Gaussian distribution to interrupt all wire positions with random numbers. The spline interpolation function is used to improve the continuity of the wires along transmission lines. Next, estimate worst-case crosstalk voltage in multiconductor interconnects is modeled through time-domain simulation program such as SPICE. The electrical parameters in a set of coupled transmission lines such as impedances, capacitances and time delays are modeled by a simulation method. The physical parameters from transmission lines such as dielectric properties, line spacing, and line length can be calculated by numerical means [12]. Unfortunately, the statistical approaches to predict crosstalk depends on many simulations on specific multi conductors and large computational effort and time are necessary.

Artificial neural network (ANN) is proposed to estimate crosstalk in the stripline transmission lines. It is an efficient and attractive method for crosstalk modeling because none of the existing methods is capable of the fully modeling the effect of the input signal spectrum crosstalk. Next, it reduces computational effort and time to predict crosstalk with high accuracy. In the addition, the effect of various uncorrelated parameters can be incorporated into single neural network model simultaneously.

2.5 ANN Approach

Neural network also called artificial neural networks (ANN) are information processing systems inspired by the studies of the ability of human brain processes a particular task from observation and generalized by abstraction. The network is implemented using electronic components or simulated in software on the digital computer [13]. The definition of neural networks described neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use [14]. It assemble brain into two parts: (1) interneuron connection strength is synaptic weights are used to store the knowledge, and (2) knowledge is obtained through a learning process.

ANN is a nonlinear network that used to solving complex problems and model the system which has a high tolerance to error. It has the nonlinear modeling and parallel processing capabilities and their ability to learn and generalize. They are universal approximator which can give a robust representation of the nonlinear mapping between design parameters and the desired performances of systems. ANN can increase the interconnect channel simulation with high accuracy due to the input and output relationships are described by a simple function.

The relationship between performance and design parameters can be defined as

$$\boldsymbol{Y} = Ƒ(\boldsymbol{X}) \tag{2.1}$$

Where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the n-dimensional input and m-dimensional output vectors that represent the design parameters and the desired performance respectively. Ƒ is the nonlinear mapping between the design parameters and the desired performance.

The ANN can be trained to learn from simulation data on a set of parameter values based on orthogonal arrays for analyzing the relationship between the parameter and performance. Orthogonal array able to capture the parameter space and acquire an optimum set of training data efficiently [15].

2.6 Neural Network versus Conventional Method

The comparison between neural network and conventional method in modeling and estimating of both linear and nonlinear system for better exploration. The neural network can learn and generalize the component behavior from the dataset of microwave components. It able to speed up the simulation process and provide prediction without sacrificing its accuracy. ANN use regression of prior simulated data as neural net inputs and implement learning rule such as the delta rule to learn the active and passive behavior. The neural network is assumed to implement to a large amount of dataset of both linear and nonlinear system [16].

The conventional method needs to identify the model structure and use the input or output data to predict the model parameters by the techniques from prediction theory. Then, compared the various time or frequency response to validate the model. The examples of conventional method are the detailed modeling approach and approximate modeling method. It is hard to obtain new devices due to limited range and accuracy. The computational method takes a long time in the simulation process and could be computationally expensive.

2.7 Structure of Neural Network

A neural network model consists of a set of interconnected nonlinear functional blocks known as neurons, which process a certain number of the incoming input signal into an output data and interconnection between them namely, links. Every link has corresponding weight parameter associated with it. Each neuron receives stimulus from other neuron connected to it, processes information and produces an output. Neural network model consists of input neurons, hidden neurons, and output neurons [17].

Input neurons are the neurons that obtain stimuli from outside the network. Neurons that provide the output externally are known as output neurons. Hidden neurons are the neurons that get the stimuli from other neurons and send output to other neurons in the network. The different structure of the neural network is made up of a different type of neurons and connecting methods [18].

Multilayer perceptron (MLP) is one of the popularly ANN structure that has been widely used in RF and microwave field. To cite an example, system identification, and nonlinear mapping. MLP can approximate an arbitrary multivariable function [19]. In the MLP neural network, the neurons are grouped into layers. The most common used in MLP formed in three layers, as shown in Figure 2. The first and last layers are input and output layer respectively and the middle layer is hidden layers (L-1). The input layer distributes input signals to the input of next layer, without any further processing. The hidden layers received information from the input layer, process it and pass it to connected neighboring neurons. The output layer whose outputs are external [19].

From Figure 2.3, let the L be the total number of layers, Lth layer is the output layer and layer 2 to L-1 are hidden layers. Let $\boldsymbol{\omega}_{ij}^{l}$ represent the weight of the link between the $\boldsymbol{j}$th neuron of the $\boldsymbol{(l-1)}$ th layer and the $\boldsymbol{i}$th neuron of the $\boldsymbol{l}$th layer. The parameters in the weight factor are initialize before MLP training. They are updated in a systematic manner during the training process [20]. The vector $\boldsymbol{\omega}$ remains constant through the usage of the neural network as a model when done the neural networks training. Activation function process is completed when each neuron received stimuli (inputs) from other neurons in MLP network. Weighted sum $\boldsymbol{\gamma}_{\mathbf{i}}^{\mathbf{l}}$ is the sum of product in the multiplication of stimuli received from neurons from (l-1)th layer, $\boldsymbol{z}_{\boldsymbol{j}}^{\boldsymbol{l-1}}$ and the weight parameter, $\boldsymbol{\omega}_{\boldsymbol{ij}}^{\boldsymbol{l}}$.

$$\gamma_{\mathrm{i}}^{\mathrm{l}} = \sum_{J=0}^{N_{I-1}} \omega_{ij}^{l} z_j^{l-1} \tag{2.2}$$

Weighted sum, γ_i^l is used to activate the neuron's activation function, σ(γ)to produce the final output of the neuron z_i^l = σ(σ(γ_i^l). The sigmoid function defined as below is the general hidden neuron activation function.

$$\sigma(\gamma) = \frac{1}{(1+e^{-\gamma})} \tag{2.3}$$

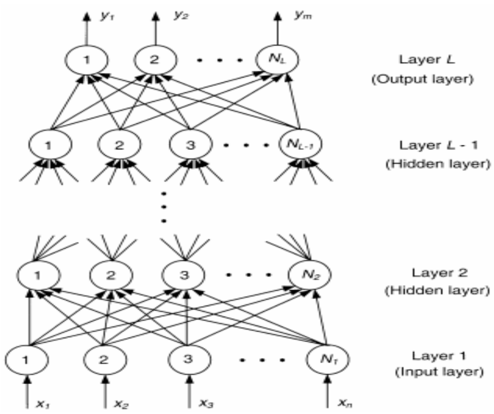

Figure 2.3: MLP neural network structure [3].

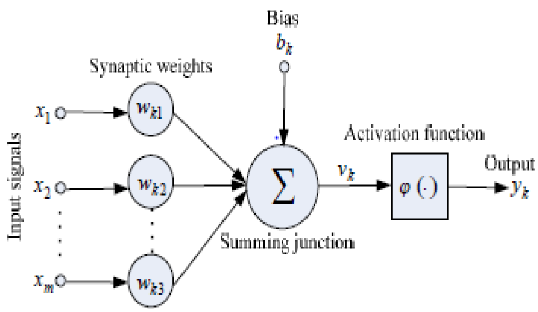

Fig 2.4: The neural network model.

2.8 Data Processing

Identification of input and output parameter is the first step in developing the neural model. The output parameter is determined based on the purpose of the neural network model. The desired data are generated using simulation software (e.g. Momentum EM simulator in Advanced Design System (ADS)). The datasets from simulation result are divided into three sets, training data, validation data and testing data. Training data is used to training the neural network like update the neural network weight parameters during training. Validation data is utilized to validate the quality of neural network model and to predict stop criteria for the training process. While testing data is used to independently test the quality of neural network model in terms of accuracy and generalization ability.

2.9 Learning rule-The Delta Rule

ANN used learning rule to modify the weight of inputs in the single-layer neural network. The most common learning rule applied in the ANN is the delta rule also known as Least Mean Square (LMS) Learning rule or Widrow-Hoff Learning rule. This rule is adjusting the strength of the input connections continuously to cut down the difference (the delta) between the desired output value and the actual output of processing element. If the difference is zero, no updating takes place. It minimized the mean square error of the network by modifying the synaptic weights. The change in weight from *ui* to *uj* is defined as:

$$d\omega_{ij} = r\,(ai)\,ej \tag{2.4}$$

where r is the learning rate, *ai* is the activation function of *ui* and *ej* is the difference between the desired output and the actual output of *uj*.

The delta rule applies a gradient descent by moving the weight vector from a point on the surface of the paraboloid down towards the lowest point, the vertex. It does not only move the weight vector closer to the ideal weight vector. The vertex refers to the point where the error is minimized and weight vector corresponding to this point is the ideal weight vector.

2.10 Data Generation

Data generation is clarified as acquire the sample pairs by using simulation or measuring process. It is produced by using simulation software such as three-dimensional EM simulator in ADS. The generated data are used for training data and testing data which used to train the neural model and check the resulting neural network model in term of accuracy respectively. Simulating result and measuring result are estimated to have small error generally. The simulated error may assign to round off or non-convergence and Measured error because of tolerance and equipment limitation. The general principle of data generation is to produce a large number of the dataset for nonlinear high-level design and a small number of the dataset for a relatively smooth low-dimensional design.

2.11 Neural Network Training

The most important step of development of the neural network is neural network training. The motivation of neural networks training is to alter neural network weight $\boldsymbol{\omega}$ and biases for the best mapping of training data output and neural model output in order to minimize the error function. The biases and weight of neural network are adjusting every cycle to estimating the error function, the difference between the simulated output and expected output [21].

Neural Network does not consist previous knowledge of the desired weight to perform better result. Thus, learning and training process play a vital position in the neural network [22]. Training data which is a set of input and output pair to update the neural network parameters during the training process. Training data produced from the result of simulations depend on orthogonal arrays. The orthogonal arrays allow better training information to ANN because of orthogonal arrays evaluate parameter space. The complexity of mapping and dimensionality of input space become the determining factors for the training data size.

Feedforward backpropagation (BP) is established as the training algorithm of MLP. Backpropagation is the most commonly used training algorithm in RF/microwave applications. Levenberg-Marquardt algorithm is implemented to enhance the convergence speed and accuracy [23]. It is used to model and interpret electromagnetic and interconnect system [24]. Other ANN training algorithm comprises conjugate-gradient, quasi-Newton, simplex, genetic and simulated annealing.

The multilayer backpropagation algorithm reduces the ANN error function, E to evaluate the performance of neural network during training process with the expression given by

$$\mathrm{E} = \frac{1}{2}\sum_{I=1}^{N}\sum_{j=1}^{m}\left\|y_{ij} - d_{ij}\right\|^2 \qquad (2.5)$$

Where m is the number of outputs, N is the number of training data, y_i is the expected ANN output obtained from full-wave simulation, and d_i is the desired output respectively. The difference of error function is computed to tune exact weight and bias of ANN. The weight is adjusting during the training process. Figure 2.3 displays the flowchart of neural network training, neural modeling, implementing training, validation and testing data sets in ANN modeling approach.

The accuracy of ANN can enhance by growing and pruning-network technique. In growing technique, fewer hidden nodes are used in the initial of the training process. Growing the number of hidden nodes gradually until the error over one epoch dropped to the desired error value. In the contrast of pruning-network technique, the large number of hidden nodes is used at the beginning and decrease the hidden nodes gradually to minimize the error to allowed error.

A number of free parameters, weights of neural network and number of hidden nodes are the key parameters for neural network accuracy approximation in input and output mapping. Sufficient training datasets are necessary to identify correct weights and fewer training datasets cannot estimate input and output mapping. After ANN is trained and verified, the parameters distribution are mapped properly to the desired distribution of performance by ANN.

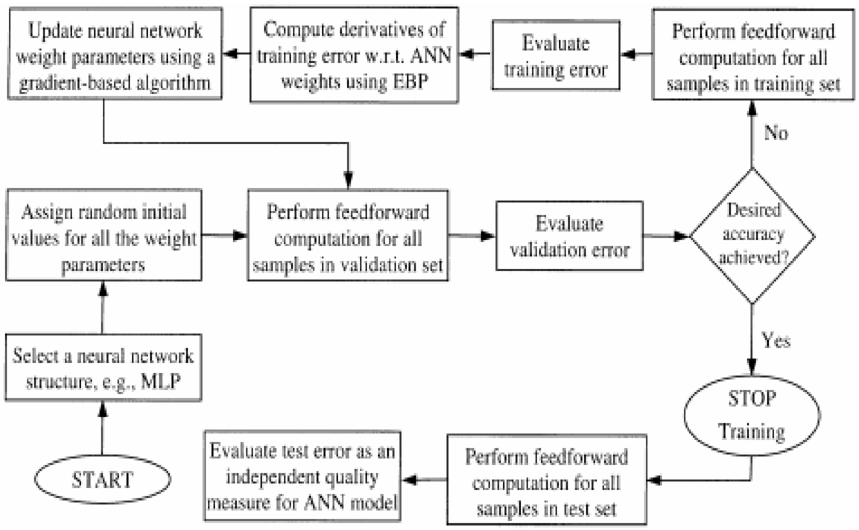

Figure 2.5: Flowchart demonstrating neural network training, neural model testing, and use of training, validation and test data sets in ANN modeling approach [3].

2.12 ANN Training Algorithms

The speed of training algorithm is hard to identify because it relies on some parameters such as total number of the testing dataset, number of weight and biases in the neural network, the error goal, component complexity and regression function. The variety of MLP training algorithms most commonly applied included Fletcher-Powell Conjugate Gradient (traincgf), Polak-Ribiére Conjugate Gradient (traincgp) and One Step Secant (trainoss) [25]. Different MLP training algorithm is used for different architectures and complexities of problems. Every training algorithms perform various accuracy levels and speed [1].

Table 2.1 showing the comparison of MLP training algorithms performance. From the result, fewer hidden nodes consume less time to train input data and more efficient in the system. Refer to the Table 2.1, the training algorithm, Conjugate Gradient with Powell/Beale Restarts (traincgb) has the best performance among other training algorithms because of the highest Signal-to-Noise (SNR) of 30 which is 89.44%.

Table 2.1: MLP training algorithms performance [1].

	MLP Training Algorithm Types							
	Train lm	***Train rp***	***Train bfg***	***Train scg***	***Train cgb***	***Train cgf***	***Train cgp***	***Train oss***
Hidden Nodes	**5**	20	**4**	14	**6**	22	**7**	19
Training (%)	100	98	100	100	100	100	100	98
Validation (%)	100	100	100	100	100	100	100	100
Testing (Ideal) (%)	98.00	88.00	98.00	92.00	**98.00**	96.00	98.00	96.00
Testing (SNR of 30) (%)	86.80	84.56	88.72	85.84	**89.44**	87.68	87.92	86.24

The Levenberg-Marquardt (trainlm) is a simple and robust algorithm for approximate a function. It gives a numerical solution to minimize the problem, especially in nonlinear function. It is a common MLP training algorithm in ANN. Various estimation in the performance in term of maximum prediction error is carried out among three training algorithms, traincgp, trainlm, and traincd. The Levenberg-Marquardt (trainlm) has verified the lowest training and test performance error as shown in Table 2.2. Thus, trainlm is used for the training process due to its high performance and accuracy [2].

Table 2.2: Comparison of the training and test performance errors among traincgp, trainlm, and traingd [2].

Training Algorithms	Performance error	
	Training data	Test data
traincgp	0.0845	0.1873
trainlm	0.0083	0.0274
traingd	0.2831	0.3165

2.13 Levenberg-Marquardt Backpropagation (trainlm)

Levenberg-Marquardt is a modification of Gauss-Newton method. It is commonly formulated as a non-linear least squares problems. It can apply to train neural network weight parameters. The performance function is the sum of squares, Hessian Matrix can be expressed as

$$H= J^{T}J \tag{2.6}$$

and the gradient can be computed as

$$g= J^{T}\mathrm{e} \tag{2.7}$$

where J is the Jacobian matrix that consist five derivatives of the network errors based on weight and biases, and e is a vector of a network error. The Jacobian matrix can be expressed via a standard backpropagation technique that is simpler than computing Hessian matrix.

The Levenberg-Marquardt implement approximation to the Hessian matrix with the following rule:

$$x_{k+1} = x_{k} \text{-} [= J^{T}J + \mu l\,]^{-1}J^{T}\mathrm{e} \tag{2.8}$$

Parameter μ make sure the matrix inversion to produce a result. This parameter relys on ethe valuation of sum of square errors [26]. Newton's method when scalar μ is zero and implement the approximate Hessian matrix. When large μ is used, gradient desceased with small step size. Newton method can speed up the process and provide accurate resuly with minimum error. Therefore, μ is reduced after each successful step and boost pnce tentative step enhance the performance function.

2.14 Network Size and Finding Number of Hidden Neurons

An optimal number of hidden neurons is crucial in determining neural network trained to be an accurate model. A number of hidden neurons based on the degree of nonlinearity of detailed physics-based input-output relationship,*f*, and dimensionality of input and output (x,y). More hidden neurons required in the highly nonlinear component and fewer neurons need for the smooth component. However, the desired hidden neuron number for a given modeling is an open question. It can estimate the number of hidden neurons through experience and trial-and-error process. The precise number of hidden neurons can be predicted via adaptive processes, which add or remove the neurons during training. Determining the ideal hidden neuron number is very important if numerous hidden neurons are used, the training process consumed a long time. In contrast, fewer hidden neurons are used, the performance of the trained neural network is not good [27]. Setting the number of hidden neurons in 1,2,4,8,16,32,64 and 128. The fine search then is used to find the neighboring range of the hidden neuron number with the lowest MSE [28].

2.15 Summary

In this chapter, the background of stripline is reviewed and included the pros and cons of the stripline. Crosstalk definition and kind of crosstalks are explained. In addition, the previous statistical methods to predict crosstalk are described and the important reasons of ANN are selected to estimate crosstalk also revealed. The research background of the artificial neural network (ANN) and application of neural network in the RF/microwave field are described. The comparison between neural network approach and conventional method also discussed. The flowchart in development of neural network model including neural network training, neural model testing, and use of training, validation and test data sets are presented. According to the research papers from previous researchers, various MLP training algorithms are compared based on the performance error and accuracy crosstalk in this project also presented.

CHAPTER 3

METHODOLOGY

3.1 Overview

This chapter reviews the implementation of Artificial Neural Network (ANN) to predict the crosstalk in the stripline. Section 3.2 described the project flow of the methodology including project implementation flow and dataset generation flow. It also discussed the Design of Experiment (DOE) used to determine the key parameters of stripline to predict crosstalk. The topics of neural network design process, data generation, and data organization are also reviewed. Furthermore, create a neural network and neural network training are presented. Performance of the ANN in the predicting crosstalk of stripline is evaluated and analyzed. Lastly, two comparative study to compare the performance of ANN training in different optimal solutions and training methods will be discussed.

3.2 Project Flow

The project starts with determining the input-output relationship of various design parameters in the stripline by Design of Experiment (DOE). It is an alternative to reduce the size of learning sets by identifying the various design parameters values in the stripline which have the greatest effect on crosstalk. The data collected through for EM simulations in the Advance Design System (ADS) by altering the physical parameters of stripline as shown in Figure 3.1.

Data collection starts with setting the component parameters of stripline (SCLIN) in the ADS schematic of a circuit simulator. Then, generated layout from schematic and define EM simulation setup included substrate properties and frequency plan. The S-parameter results generated from EM simulation are compared with S-parameters results in the circuit simulator. Once S-parameter results matched, extracted S-parameters are generated into 4-port S-parameters file block (S4P) which represented 4-terminal coupled stripline for transient modeling. The simulation result from transient modeling, near-end crosstalk voltage are recorded. The process of data collection is repeated again until the 168 sets of training data are generated.

The data generated are divided into three sets, training data, test data and validation data. Neural network architecture and training algorithm are decided. Next, the training goal of ANN is set to accomplish. Matlab Neural Network Toolbox is used to create neural network models. The neural network is trained by training data. The performance of trained neural network is evaluated based on the error function (mean square error) by evaluation data. If the neural network does not reach the allowed error goal, the training process is required to restart. More learning points are required added to data collection and varying the design parameters. The flow chart of neural network training as shown in Figure 3.2.

Finally, neural network models are trained to learn the characterization and behavior of data for crosstalk prediction in stripline. The simulated result and predicted result of crosstalk in stripline are compared and validated.

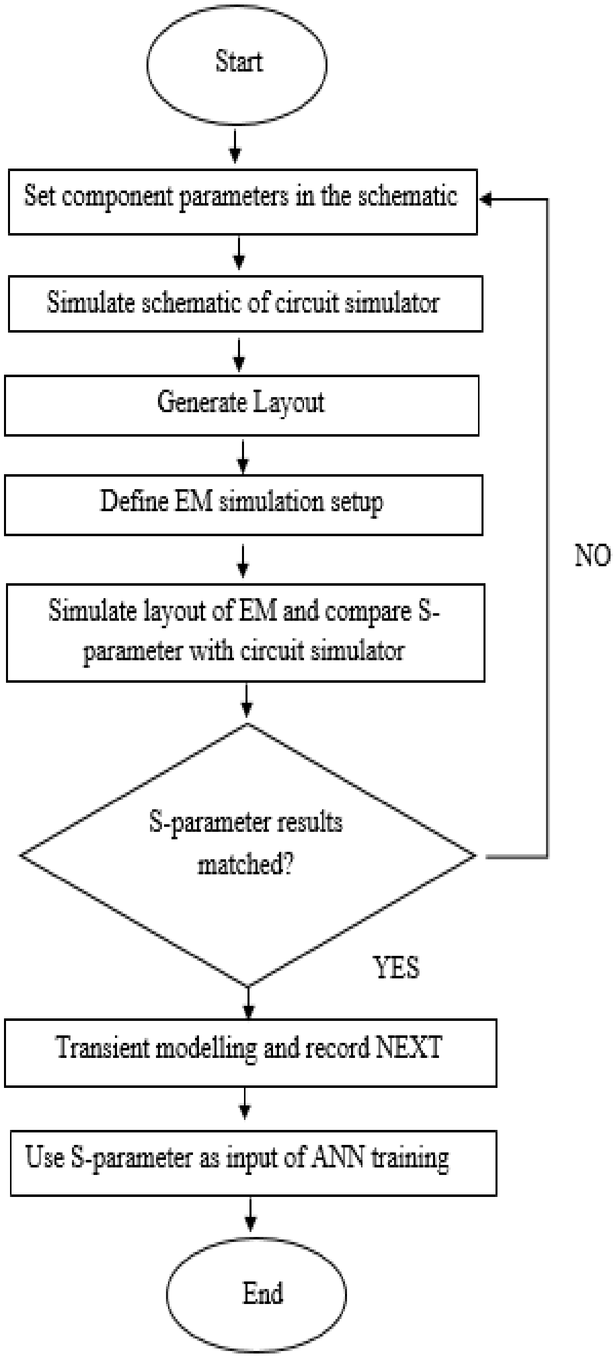

Figure 3.1: Flow of data generation