

Machine-Learning Based Semiconductor Device Characterisation and Behavioural Modelling

Weihao Yuan

A thesis **submitted** for the degree of
Doctor of Philosophy

Cardiff University
March 2026

Abstract

This thesis investigates methods to improve the efficiency of behavioural modelling workflows for Radio Frequency (RF) power amplifiers (PAs) using load-pull measurements. In practical design processes, both large-scale simulations and measurements can consume significant time, making model development and design optimisation slower.

To improve simulation efficiency, an Artificial Neural Network (ANN) model implementation framework is developed in a computer-aided design environment based on an existing Cardiff Model (CM) implementation. The framework introduces a dynamic function-generation framework within the ADS schematic, allowing the ANN model implementation to be adapted according to the required model structure. Based on this framework, a systematic runtime comparison between ANN model and CM implementations is presented under load-pull conditions of varying complexity. The results show that schematic structure has a strong effect on simulation cost. For the fundamental-frequency simulation considered in this study, removing calculations that were redundant for the selected model configuration reduced the runtime of the original CM implementation by approximately 99% while maintaining modelling accuracy.

In addition, this thesis investigates systematic (Type B) uncertainties in load-pull measurement systems that may remain hidden after calibration. A novel method is developed to characterise and assess these uncertainties. A measurement system integrity assessment workflow is introduced, enabling early detection of abnormal deviations before they affect behavioural model extraction and device performance evaluation.

The results show that the measurement-domain threshold is exceeded by systematic distortions of around 1-2% of the incident-wave magnitude, while the model-domain threshold is exceeded at around 2.6%. These results indicate that proper model implementation can significantly reduce simulation time, while early detection of systematic distortions can help reduce the time wasting on re-measurement and recalibration.

Overall, this work provides a new perspective on improving both computational and experimental efficiency in load-pull based behavioural modelling, contributing to faster and more reliable RF PA design workflows.

List of Publications

- [1] W. Yuan, M. Tian, and J. Bell, “Implementation Efficiency Comparison Between ANN and the Cardiff Model in ADS,” in *2023 International Workshop on Integrated Nonlinear Microwave and Millimetre-Wave Circuits (INMMIC)*, 2023, pp. 1–4. DOI: 10.1109/INMMIC57329.2023.10321776.

Acknowledgements

I would like to express my sincere thanks to my supervisor, Dr. James Bell. Throughout my PhD, James has always supported me with his knowledge, guidance, and care. He was always there to encourage me whenever I felt lost, and he gave me direction countless times during this journey.

I also deeply appreciate my co-supervisor, Dr. Roberto Quaglia, for his great help with the measurement work in this project, and for his advice and support throughout my studies.

My special thanks to Dr. Mengyue Tian. I am very grateful to her for introducing me to James and giving me the opportunity to begin this PhD. I also appreciate the help and support she gave me during my PhD.

Also, lots of thanks to Prof. Paul Tasker and Dr. Alexander Baddeley from the Centre for High Frequency Engineering for sharing their experience and for their help in modelling and measurement work. I would also like to profoundly thank the staff and colleagues in the Centre for High Frequency Engineering. I will always remember the times we played football and had lunch together.

Finally, I would like to express my deepest gratitude to my parents for their support, encouragement, and companionship. I am especially thankful for their financial support, which made my PhD study possible. I am also very grateful to my friends for always being with me throughout these years.

I feel very lucky to have met so many wonderful people during this important stage of my life. The experiences will always be a valuable treasure of my life.

List of Abbreviations

AEL Application Extension Language

ANN Artificial Neural Network

CAD Computer-Aided Design

CM Cardiff Model

DUT Device Under Test

DWLU Direct Waveform Lookup

GaN Gallium Nitride

GUM Guide to the Expression of Uncertainty in Measurement

HEMT High Electron Mobility Transistor

KBNN Knowledge-Based Neural Network

NMSE Normalised Mean Square Error

PA Power Amplifier

PAE Power-Added Efficiency

PHD Poly-Harmonic Distortion

RF Radio Frequency

TRL Thru-Reflect-Line

VNA Vector Network Analyser

Contents

Abstract	i
List of Publications	iii
Acknowledgements	iv
List of Abbreviations	v
List of Figures	xi
List of Tables	xv
1 Introduction	1
1.1 Transistor Models for RF/Microwave Design	2
1.2 Measurement Systems and Load-Pull Workflow	4
1.3 Thesis Objective	7
1.4 Chapter Synopsis	8
References	9
2 Literature Review	12
2.1 S-parameter and Non-linear Behaviour Models	13
2.1.1 S-parameter	13
2.1.2 Non-linear Behaviour Models	15
2.1.3 S-parameter and Non-linear Behaviour Models Discussion . . .	22
2.2 The Cardiff Model	23
2.2.1 The Cardiff DWLU Model	23

2.2.2	The Cardiff Model Formulation	25
2.2.3	The Cardiff Model Extraction	31
2.2.4	The Cardiff Model CAD Implementation	33
2.2.5	The Cardiff Model Discussion	38
2.3	ANN Model	39
2.3.1	Model Structure	39
2.3.2	Activation Function	43
2.3.3	Adding Bias	49
2.3.4	Model Training	50
2.3.5	ANN Model Discussion	51
2.4	Error and Calibration	52
2.4.1	Network Analyser	52
2.4.2	Source of Error	54
2.4.3	Error Model	55
2.4.4	Calibration Method	60
2.4.5	Calibration Discussion	68
2.5	Conclusion	69
	References	69
3	ADS Schematic Design Implementation	76
3.1	Introduction	76
3.2	Creating the Dynamic Schematic	81
3.2.1	AEL Code Working Flow on Placing the VAR	81
3.2.2	Adapting Parameter Import for Different ANN Structures	82
3.2.3	Implementation	84
3.2.4	Verification	88
3.3	Conclusion	89
	References	89

4	ANN and Cardiff Model Efficiency Comparison in ADS	91
4.1	Introduction	92
4.2	Experiment	92
4.2.1	Simulation Setup	92
4.2.2	Cardiff Model Schematic Simplification for Fundamental Fre- quency Simulations Only	94
4.2.3	Schematic Adjustments for a Reduced Model Coefficient Set .	96
4.2.4	Power Sweep	100
4.3	Results and Analysis	100
4.4	Conclusion	104
	References	104
5	Type B Error Detection and Integrity Assessment in Load-Pull Measurement Systems	106
5.1	Introduction	107
5.2	Experimental Setup	109
5.2.1	Calibration	110
5.2.2	Thru Measurements	112
5.2.3	Device Measurements	113
5.3	Results and Error Model Analysis	115
5.3.1	Thru Measurement Results and Error Model Extraction . . .	116
5.3.2	GaN HEMT Measurement Results and Error Model Application	121
5.3.3	Error Effect Analysis for Cardiff Model Coefficients	125
5.4	Calibration and Model Integrity Metrics	131
5.5	Validation Using Designer Metrics	136
5.5.1	Global Distortion	136
5.5.2	Behaviour Under Large Systematic Distortion	138
5.5.3	Validation at an Additional Input Power Level	140
5.5.4	Practical Implementation of the Integrity Metric	144
5.6	Conclusion	146

References	147
6 Conclusion and Future Work	149
6.1 Conclusion	150
6.2 Future Work	152
References	154
A AEL Implementation Details for the ANN Model in ADS	155

List of Figures

1.1	Example of a CAD-based load-pull implementation for behavioural model validation	7
2.1	S-parameter flow diagram	14
2.2	The system equivalence of a two-port DUT	15
2.3	The harmonic superposition principle	18
2.4	Definition of the Cardiff DWLU model in frequency domain	23
2.5	The input second harmonic and output second harmonic coefficient space	32
2.6	Comparison of measured and modelled b_{21} with a Cree device and different numbers of coefficients	33
2.7	FDD core of the model schematic	34
2.8	The FDD functions	34
2.9	The DAC and file variables	35
2.10	V-I to a-b translation equations and equations for phase normalisation	36
2.11	A-element calculations	36
2.12	Construction of the B-matrix using the R-matrix, and A-matrix equations	37
2.13	Flow diagram of ADS-AEL simulation	38
2.14	General MLP structure	40
2.15	KBNN structure	42
2.16	KBNN structural diagram with activation function setup	43
2.17	The Sigmoid activation plot with $x = [-5 \ 5]$ range	44

2.18	<i>Tanh()</i> activation plot with $x = [-5 \ 5]$ range	45
2.19	The Symmetric Elliot activation plot with $x = [-5 \ 5]$ range	46
2.20	The ReLU activation plot with $x = [-5 \ 5]$ range	47
2.21	The ELU activation plot with $x = [-5 \ 5]$ range	48
2.22	Structural diagram with bias added to the setup	49
2.23	ANN structure for a single output neuron with bias and activation function	50
2.24	VNA diagram [31]	53
2.25	Signal flow graph for one-port error model	56
2.26	Signal flow graph for 12-term error model	57
2.27	Signal flow graph for 8-term error model	58
2.28	Signal flow graph for 16-term error model	60
2.29	Signal flow graph for calibration standards of TRL	62
3.1	The voltages/currents receiver and sender	77
3.2	DAC for accessing data	78
3.3	Simplified calculation flow diagram of the implemented ANN model, including input normalisation and output de-normalisation.	80
3.4	AEL code working flow	82
3.5	Working flow of weights and biases import	83
3.6	Working flow of generating ANN model calculation equations	84
3.7	<i>main()</i> function	85
3.8	<i>Phase_Normalising()</i> function	86
3.9	<i>ANN_Cal(n)</i> function code	87
3.10	<i>DAC()</i> function code	88
3.11	Verification result	89
4.1	Basic load-pull circuit	92
4.2	Very dense fundamental load point grid for the 72,000 load points with $\Gamma_{max} = 0.8$	93

4.3	V-I to a-b translation equations and equations for phase normalisation	94
4.4	Full 2-input harmonics and 3 output harmonic Amatrix calculations on the ADS schematic	95
4.5	Fundamental only simplify Amatrix calculation under the fundamen- tal frequency	95
4.6	Fundamental only Amatrix calculation without exponentiation	96
4.7	2-coefficient CM	97
4.8	6-coefficient CM	97
4.9	2 hidden layer neurons ANN model	99
4.10	6 hidden layer neurons ANN model	99
4.11	Model complexity run time comparison for the CM and ANN models	102
5.1	VNA calibration setup, showing the main measurement components and the TRL calibration standards	111
5.2	Schematic diagram of the thru load-pull measurement	112
5.3	Smith chart of the thru load points	113
5.4	Load-pull measurement system used for device measurements	114
5.5	Schematic diagram of the load-pull measurement system used for de- vice measurements	115
5.6	Residual thru error maps	118
5.7	Smith chart of the device load points	121
5.8	Measured b_2 magnitude (dB) over the device Γ plane	122
5.9	Measured and corrected output power P_{out} distribution over the de- vice Γ plane	124
5.10	Output power difference ΔP_{out} after applying the correction	125
5.11	Effect of the proposed Γ -dependent correction on the extracted CM coefficients	127
5.12	Predicted performance difference between the two extracted models .	130
5.13	Additive offset stress test with 3σ statistical threshold	133
5.14	CM coefficient drift vs injected error	135

5.15	ΔP_{out} (dB) over the Γ plane for global distortion	137
5.16	Output power deviation and contour shift under global distortion with magnitude $\alpha = 8\alpha_{th}$	139
5.17	Comparison of measured and corrected output power for the dataset measured at a different source power level	141
5.18	ΔP_{out} (dB) over the Γ plane for global distortion at another input power level	143
5.19	Workflow for practical implementation of the proposed measurement system integrity assessment method	145
A.1	Travelling-wave calculation and normalisation	155
A.2	Hidden-layer parameters for data import	156
A.3	Output-layer parameters for data import	156
A.4	Weight and bias normalisation and re-normalisation	157
A.5	ANN model calculation block	158

List of Tables

2.1	Correspondence between PHD modelling formulas and X-Parameter general form	21
2.2	Phase and magnitude exponent parameters corresponding to (2.38). .	29
2.3	Phase and magnitude exponent parameters corresponding to (2.39). .	29
4.1	Run time and NMSE comparison of the ANN and CM implementa- tions using the 72,000-point dataset	100
4.2	Run time comparison of the ANN and simplified CM models	103
5.1	Magnitude and phase residual statistics	119
5.2	CM prediction accuracy before and after Γ -dependent correction . . .	128

Chapter 1

Introduction

Radio frequency (RF) and microwave power amplifiers play an important role in modern wireless communication and electronic systems. Their design depends significantly on accurate transistor behavioural models that can predict large-signal behaviour under practical operating conditions. In a Computer-Aided Design (CAD) environment, such models help designers save development time and cost by enabling circuit optimisation before hardware implementation.

However, the application and extraction of these models depend on both efficient simulation procedures and accurate measurement data. Load-pull measurement systems are commonly used to obtain the large-signal datasets required for behavioural modelling, while a CAD environment is used to implement the extracted models. Therefore, the overall behavioural modelling workflow depends not only on model accuracy and simulation time, but also on the reliability of the measured data used for model extraction.

To provide the background for the rest of the thesis, this chapter first introduces the main transistor modelling approaches used in RF and microwave design, especially the behavioural modelling methods relevant to large-signal power amplifier applications. Then it describes the load-pull measurement systems and CAD environments used for model extraction and evaluation. Finally, the objectives of this thesis are presented, followed by a summary of the chapter structure.

1.1 Transistor Models for RF/Microwave Design

Under low excitation levels, small-signal models are often sufficient to represent linear device behaviour. However, for RF power amplifier design, transistor modelling becomes more difficult under large-signal operating conditions because the device behaves non-linearly. When the device operates in the non-linear region, the model must consider the effects of input power, bias, and load impedance. This has led to the development of advanced large-signal modelling approaches to predict device response more accurately.

In general, transistor models for RF and microwave applications can be separated into physics-based models, table-based models, and behavioural models. Physics-based models are derived from the internal device physics and are usually used to study transistor mechanisms and process-related effects. Among them, compact models are widely used in circuit simulation [1], [2]. They use simplified equations to describe device behaviour. However, because these models are closely related to device structure and model formulation, they can be harder to extract and adapt for large-signal applications based directly on measured data.

Table-based models provide another approach by storing measured device responses under different operating conditions and using interpolation to predict behaviour within the measured region [3], [4]. Their main advantage is that they can reproduce measured data directly without needing a detailed analytical model of the device. However, their accuracy depends strongly on the quality and density of the measurement data. It is also affected by the interpolation capability of the CAD environment. In addition, table-based models usually have limited ability to extrapolate outside the measured operating region [3].

To solve the limitations of purely linear modelling under large-signal operation, a range of non-linear behavioural modelling approaches has been developed. Beginning with the small-signal S-parameter model, methods such as Volterra Input Output Map (VIOMAP) [5], hot S-parameters [6], Poly-Harmonic Distortion (PHD)

modelling [7], [8], and X-parameters [9] have extended the ability to represent non-linear device behaviour under large-signal conditions. These approaches provided an important foundation for later waveform-based and behavioural modelling techniques used in RF power amplifier design.

To improve the direct use of measured large-signal data in CAD environments, the Cardiff Direct Waveform Lookup Table (DWLU) approach was developed for importing measured device characteristics into simulation [10]. Building on the Cardiff DWLU and PHD approaches, the Cardiff Model (CM) was developed as a behavioural modelling framework in which the relationship between incident and scattered waves can be represented using a configurable set of model coefficients, allowing the model complexity and harmonic content to be adapted to the available measurement data [11], [12]. Unlike purely table-based approaches, it offers a formulated model structure while still remaining closely related to measured device behaviour.

At the same time, artificial neural network (ANN) techniques have also been applied to transistor behavioural modelling [13], [14]. ANN-based models provide a data-driven approach that is able to represent complex non-linear relationships directly from measured or simulated datasets. Compared with highly structured behavioural models, ANN models provide larger flexibility in model form, although their behaviour depends significantly on the selected network structure, training process, and available dataset [15], [16].

This thesis mainly focuses on the CM and ANN-based modelling approaches. The CM provides an established behavioural modelling framework for RF power amplifier applications [11], [12], while ANN models provide a flexible approach to non-linear data representation [13], [15]. Together, these two approaches provide a useful basis for investigating both the implementation efficiency of behavioural models in CAD environments and the reliability of the measurement data on which these models depend.

Because both the CM and ANN-based models rely on proper large-signal datasets

for model extraction and assessment, the measurement system is an important part of the overall modelling workflow. The load-pull measurement systems used to obtain these data are therefore introduced in the following section.

1.2 Measurement Systems and Load-Pull Workflow

In load-pull based behavioural modelling, the measurement system is not only used to obtain device data, but also affects the efficiency and reliability of the modelling workflow. For this reason, this section explains how the load-pull measurement system contributes to behavioural model extraction.

Different transistor modelling approaches require different types of measurement data for model extraction and validation. For small-signal models, the required data are usually obtained from Vector Network Analyser (VNA) measurements under linear operating conditions. In contrast, large-signal behavioural modelling requires measured datasets obtained when the device operates under non-linear operating conditions [17], [18]. For this reason, load-pull measurement systems are widely used in RF and microwave power amplifier design, because they enable the device behaviour to be characterised over a range of fundamental and, where required, harmonic impedance conditions. As a result, the measurement system is an important part of the overall behavioural modelling workflow.

In load-pull measurements, the load impedance of the device is varied so that the corresponding transistor response can be observed under different operating conditions. By changing the load condition, it is possible to evaluate changes in output power, gain, efficiency, and waveform behaviour across the Smith chart. Since device performance depends significantly on impedance conditions, this makes load-pull measurements suitable for non-linear transistor modelling and power amplifier design [17], [18]. Actually, load-pull measurements provide the large-signal datasets required for behavioural model extraction and CAD simulation.

Load-pull systems can be implemented using passive, active, or hybrid approaches [17], [18], depending on the required impedance range and device power level. Passive load-pull systems usually use tuners to synthesise the required impedance. They are relatively direct in operation. However, the available impedance range is limited by system losses, especially when load conditions are close to the boundary of the Smith chart, which is particularly critical for devices whose optimum load impedances lie in these high-reflection regions.

Active load-pull systems overcome this limitation by injecting additional signals to create the required impedance conditions at the device ports. Hybrid approaches combine passive and active load-pull to cover wider impedance ranges while keeping measurement flexibility [17], [18]. Although the detailed system structure may differ, the main purpose of these methods is to obtain device measurement data over a range of large-signal operating conditions.

For behavioural modelling, the large-signal response of the transistor is usually described with incident and reflected travelling waves. In a two-port system, the incident waves are denoted by a-waves and the reflected waves by b-waves. They provide a convenient frequency-domain description of device behaviour and are widely used in load-pull based modelling approaches. Depending on the specific measurement system, the travelling waves can be measured directly or calculated from the corresponding voltage and current signals with a reference impedance Z_0 . The voltage-current and travelling-wave relationships can be written as follows:

$$V = \frac{Z_0^* a + Z_0 b}{\sqrt{\Re(Z_0)}} \quad (1.1)$$

$$I = \frac{a - b}{\sqrt{\Re(Z_0)}} \quad (1.2)$$

$$a = \frac{V + Z_0 I}{2\sqrt{\Re(Z_0)}} \quad (1.3)$$

$$b = \frac{V - Z_0^* I}{2\sqrt{\Re(Z_0)}} \quad (1.4)$$

where V and I are the terminal voltage and current, a and b are the incident and reflected travelling waves, Z_0 is the reference impedance, and Z_0^* is the complex conjugate of Z_0 . In this thesis, Z_0 is usually treated as 50Ω . These relationships enable the measured large-signal data to be represented in a form convenient for behavioural model extraction in the frequency-domain [19], [20].

With the travelling-wave representation, the device response can be described more generally as a function of the incident wave conditions [20]. When harmonic effects are included, a generic formulation for large-signal behavioural modelling can be expressed as

$$b_{p,h} = F(a_{p,h}) \quad (1.5)$$

where p denotes the port number and h denotes the harmonic index. This expression can be used to show that the measured waves are used to describe transistor behaviour under non-linear large-signal operation. The modelling dataset is obtained under a range of specific operating conditions, such as input power, bias, frequency, and load impedance. Then it is used for model extraction and evaluation.

After measurement, the obtained data are usually processed in a numerical environment for model extraction and parameter determination. For example, platforms such as MATLAB can be used to organise the measured travelling-wave data and extract the model coefficients. The extracted model can then be imported into a CAD environment for implementation and evaluation. Modern CAD platforms such as Keysight ADS can support load-pull simulation, parameter sweeps, and behavioural model implementation, allowing the measured device data to be applied directly to circuit-level analysis [21]. For example, in ADS, a Frequency-Domain Defined Device (FDD) component can be used to implement measured or extracted model

data for load-pull simulation [21]. This provides a direct connection between the measurement system and the design process in CAD.

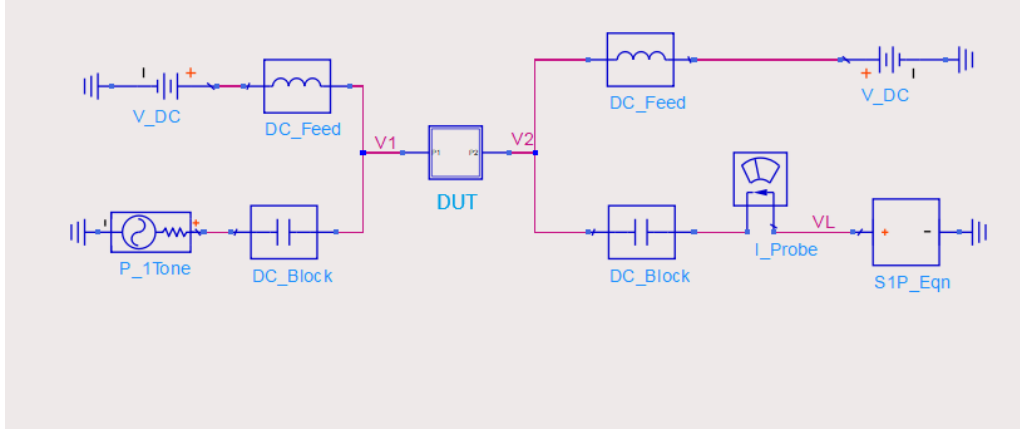


Figure 1.1: Example of a CAD-based load-pull implementation for behavioural model validation

Because of the tight connection between measurement and CAD environments, the quality of the final model depends on both simulation efficiency and the accuracy of the measured data. A modelling workflow can become inefficient when simulations take too much time or when hidden errors in the measured data require repeated checking or re-measurement. Therefore, the measurement system is not only a source of data, but also a key factor affecting the efficiency and reliability of behavioural modelling for RF power amplifier design. This motivates the thesis objective presented in the following section.

1.3 Thesis Objective

In practical RF power amplifier design workflows, behavioural modelling significantly depends on both efficient simulation and reliable measurement data. However, large-scale load-pull based modelling can take a long time, not only because of the computational cost of simulation, but also because hidden problems in measured data which may require repeated checking, re-measurement, or recalibration.

For simulation, different behavioural model structures can result in large differences in runtime under large-scale load-pull conditions. For measurements, system-

atic errors may remain hidden after calibration and may gradually increase with time after the initial calibration, before propagating into the extracted model and affecting predicted device performance. These two issues reduce the efficiency of the modelling workflow and may lead to unnecessary time and cost.

The objective of this thesis is to improve the efficiency of load-pull based behavioural modelling workflows by addressing both simulation runtime and the early detection of systematic measurement errors. To achieve this, an ANN model implementation framework is developed based on an existing CM implementation, and a dynamic schematic generation function is introduced to enable the ANN model to be created according to the required model structure. Based on this framework, a systematic runtime comparison between ANN and CM implementations is presented under load-pull conditions of varying scale.

In addition, a novel method is proposed to characterise and assess systematic uncertainties in load-pull measurement systems. A measurement system integrity assessment workflow is introduced to enable early detection of abnormal deviations before they significantly affect behavioural model extraction and device performance. In this way, the thesis addresses workflow efficiency from both simulation and measurement perspectives within the same behavioural modelling process.

1.4 Chapter Synopsis

Based on the objectives described above, the thesis is organised as follows.

Chapter 2 presents a literature review of transistor modelling techniques for RF and microwave applications. Particular attention is given to large-signal behavioural modelling approaches, including the CM and ANN-based models. The related measurement and uncertainty background is also introduced to support the later discussions on both CAD implementation and measurement reliability.

Chapter 3 describes the CAD implementation framework used in this thesis. It introduces the ANN model implementation framework developed based on an existing CM implementation, together with the simulation setup that provides the

basis for the runtime analysis presented in Chapter 4.

Chapter 4 investigates the simulation efficiency of the behavioural modelling approaches. The runtime performance of different model formulations is evaluated under load-pull conditions of varying scale in order to test the impact of model structure on computational cost.

Chapter 5 investigates systematic uncertainties in load-pull measurement systems and their effect on behavioural modelling. While Chapter 4 focuses on simulation efficiency in CAD, Chapter 5 investigates the measurement side of the workflow. A method is developed to characterise and assess systematic errors, and a measurement system integrity assessment workflow is proposed for early detection of abnormal deviations before they significantly affect model extraction and device performance evaluation.

Chapter 6 concludes the thesis and summarises the main findings of the work. Possible directions for future research are also discussed.

References

- [1] P. H. Aaen, J. A. Plá, and J. Wood, *Modeling and Characterization of RF and Microwave Power FETs* (The Cambridge RF and Microwave Engineering Series). Cambridge: Cambridge University Press, 2007.
- [2] U. Radhakrishna, T. Imada, T. Palacios, and D. Antoniadis, “MIT virtual source GaNFET-high voltage (MVSG-HV) model: A physics-based compact model for HV-GaN HEMTs,” *Physica Status Solidi C*, vol. 11, no. 3-4, pp. 848–852, 2014. DOI: 10.1002/pssc.201300392.
- [3] S. Woodington et al., “A novel measurement based method enabling rapid extraction of a RF Waveform Look-Up table based behavioral model,” in *2008 IEEE MTT-S International Microwave Symposium Digest*, 2008, pp. 1453–1456. DOI: 10.1109/MWSYM.2008.4633053.
- [4] H. Qi, J. Benedikt, and P. Tasker, “A Novel Approach for Effective Import of Nonlinear Device Characteristics into CAD for Large Signal Power Amplifier Design,” in *2006 IEEE MTT-S International Microwave Symposium Digest*, 2006, pp. 477–480. DOI: 10.1109/MWSYM.2006.249596.
- [5] F. Verbeyst and V. Bossche, “VIOMAP, the S-parameter equivalent for weakly nonlinear RF and microwave devices,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, no. 12, pp. 2531–2535, 1994. DOI: 10.1109/22.339793.

-
- [6] J. Verspecht, D. Barataud, J.-P. Teyssier, and J.-M. Nebus, “Hot S-parameter techniques: $6 = 4 + 2$,” eng, in *2005 66th ARFTG Microwave Measurement Conference : digital communication system metrics : December 1st - 2nd 2005, Washington Marriott, Washington DC*, IEEE, 2005, pp. 1–9.
 - [7] Verspecht and D. Root, “Polyharmonic distortion modeling,” *IEEE Microwave Magazine*, vol. 7, no. 3, pp. 44–57, 2006. DOI: 10.1109/MMW.2006.1638289.
 - [8] J. Verspecht and P. Van Esch, “Accurately Characterizing Hard Nonlinear Behavior of Microwave Components with the Nonlinear Network Measurement System: Introducing Nonlinear Scattering Functions,” in *Proceedings of the 5th International Workshop on Integrated Nonlinear Microwave and Millimeterwave Circuits*, Duisburg, Germany, 1998, pp. 17–26.
 - [9] D. E. Root, J. Verspecht, J. Horn, and M. Marcu, *X-Parameters: Characterization, Modeling, and Design of Nonlinear RF and Microwave Components* (The Cambridge RF and Microwave Engineering Series), English. Cambridge: Cambridge University Press, 2013.
 - [10] H. Qi, J. Benedikt, and P. J. Tasker, “Nonlinear Data Utilization: From Direct Data Lookup to Behavioral Modeling,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 6, pp. 1425–1432, 2009. DOI: 10.1109/TMTT.2009.2019996.
 - [11] P. J. Tasker and J. Benedikt, “Waveform Inspired Models and the Harmonic Balance Emulator,” *IEEE Microwave Magazine*, vol. 12, no. 2, pp. 38–54, 2011. DOI: 10.1109/MMM.2010.940101.
 - [12] S. Woodington, R. Saini, D. Williams, J. Lees, J. Benedikt, and P. J. Tasker, “Behavioral model analysis of active harmonic load-pull measurements,” in *2010 IEEE MTT-S International Microwave Symposium*, 2010, pp. 1688–1691. DOI: 10.1109/MWSYM.2010.5517261.
 - [13] V. Devabhaktuni, M. Yagoub, and Q.-J. Zhang, “A robust algorithm for automatic development of neural-network models for microwave applications,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 12, pp. 2282–2291, 2001. DOI: 10.1109/22.971611.
 - [14] F. Wang, V. K. Devabhaktuni, C. Xi, and Q.-J. Zhang, “Neural network structures and training algorithms for RF and microwave applications,” *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 9, no. 3, pp. 216–240, 1999. DOI: 10.1002/(SICI)1099-047X(199905)9:3<216::AID-MMCE7>3.0.CO;2-W.
 - [15] A. Suntives, M. S. Hossain, J. Ma, R. Mittra, and V. Veremey, “Application of artificial neural network models to linear and nonlinear RF circuit modeling,” *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 11, no. 4, pp. 231–247, 2001. DOI: <https://doi.org/10.1002/mmce.1028>.
 - [16] A. Khusro, S. Husain, M. S. Hashmi, and A. Q. Ansari, “Small signal behavioral modeling technique of GaN high electron mobility transistor using artificial neural network: An accurate, fast, and reliable approach,” *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 30, no. 4, e22112, 2020. DOI: <https://doi.org/10.1002/mmce.22112>.

-
- [17] M. S. Hashmi and F. M. Ghannouchi, “Introduction to load-pull systems and their applications,” *IEEE Instrumentation & Measurement Magazine*, vol. 16, no. 1, pp. 30–36, 2013. DOI: 10.1109/MIM.2013.6417055.
 - [18] F. Deshours, E. Bergeault, F. Blache, J.-P. Villotte, and B. Villeforceix, “Experimental comparison of load-pull measurement systems for nonlinear power transistor characterization,” *IEEE Transactions on Instrumentation and Measurement*, vol. 46, no. 6, pp. 1251–1255, 1997. DOI: 10.1109/19.668266.
 - [19] P. J. Tasker, “Practical waveform engineering,” *IEEE Microwave Magazine*, 2009. DOI: 10.1109/mm.2009.934518.
 - [20] J. Wood and D. E. Root, *Fundamentals of Nonlinear Behavioral Modeling for RF and Microwave Design* (Artech House Microwave Library), English. Boston, Mass.: Artech House, 2005.
 - [21] Keysight Technologies, *Advanced Load Pull DesignGuide*, Keysight ADS documentation.

Chapter 2

Literature Review

How to build a highly efficient and accurate behavioural model that can help researchers design and analyse prototypes is a key problem in the RF and microwave industries. A well-developed device model allows companies and laboratories to save time and money on exhaustive investigations of other possible prototypes. Currently, with the rapid development of electronic information technology and communication technology, gallium nitride high electron mobility transistors (GaN HEMTs) have been widely used in these areas. From GaN HEMT devices at the device level to circuit-level applications, the development of large-signal models has become increasingly important.

In the early stages of research, around the 1990s, researchers used relatively simple expressions to describe device behaviour, which are generally referred to as empirical models. Representative examples of this category include the Statz model [1], the Crutice model [2], and the Angelov model [3], [4]. For instance, the Angelov model makes clever use of the $\tanh()$ function to accurately describe the phenomenon in which the transconductance of a HEMT device reaches a peak as the gate voltage varies. With time, these empirical models were further developed, extended, and standardised. Amongst these developments, the Energy Equation HEMT Model (EEHEMT) [5], developed by Keysight Technologies, is one of the most validated models. It is based on the Angelov model but extends and enhances it, making the EEHEMT more comprehensive and robust. Meanwhile, other mod-

els moved beyond descriptions based mainly on mathematical expressions of GaN HEMT characteristics, toward physics-based virtual-source models. One example is the MIT virtual source GaNFET-high voltage (MVSG-HV) model [6].

Under small-signal conditions, scattering parameters (S-parameters) are commonly employed for GaN HEMT model extraction. However, when applied to large-signal conditions, linear S-parameters cannot accurately predict non-linear device behaviour. For this reason, non-linear models are indispensable. Well-known non-linear models include the Volterra Input Output Map (VIOMAP) [7], Poly-Harmonic Distortion (PHD) model [8], X-parameters [9], Cardiff Model (CM) [10], [11] and Artificial Neural Network (ANN) models [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28]. All of these approaches have been successfully validated and widely used in applications around the world. Therefore, in this chapter, both linear and non-linear models will first be reviewed. Following this, because the work in this thesis is primarily based on the CM and the ANN model, these two models will be introduced in detail. Finally, some basic knowledge of calibration and two types of errors will also be discussed, as these are used for the collection and analysis of measurement data required for the modelling focus of this thesis.

2.1 S-parameter and Non-linear Behaviour Models

2.1.1 S-parameter

At low frequencies, it is relatively straightforward to analyse circuits in terms of voltages, currents, and impedances. However, when frequency increases and the wavelength becomes shorter, transmission line effects dominate. Under these circumstances, conventional circuit analysis methods become ineffective because voltages and currents vary with position along the line. To address this, engineers employ

S-parameters [29], [30] to describe how a network or device reflects or transmits incident signals. Simply put, S-parameters characterise the reflection and transmission properties of an RF device at its ports [31].

For each port, there exist incident waves (a-waves) and reflected waves (b-waves). S-parameters define the mathematical relationship between these two quantities.

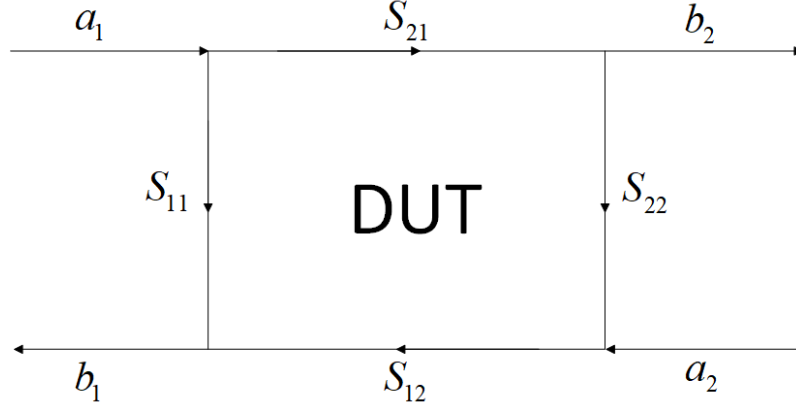


Figure 2.1: S-parameter flow diagram

Figure 2.1 illustrates an S-parameter schematic diagram for a device under test (DUT). In this figure, a_1 and a_2 represent incident waves, while b_1 and b_2 represent reflected waves, as indicated by the arrows. From these relationships, the general form of the S-parameter equations can be derived, as shown in equation (2.1). For a two-port network, the equations can be expressed as:

$$\begin{aligned}
 S_{11} &= \left. \frac{b_1}{a_1} \right|_{(a_2=0)} \\
 S_{12} &= \left. \frac{b_1}{a_2} \right|_{(a_1=0)} \\
 S_{21} &= \left. \frac{b_2}{a_1} \right|_{(a_2=0)} \\
 S_{22} &= \left. \frac{b_2}{a_2} \right|_{(a_1=0)}
 \end{aligned} \tag{2.1}$$

With equation (2.1) and S-parameters, the relationship between a-waves and b-waves at different ports are shown here.

$$\begin{aligned} b_1 &= S_{11}a_1 + S_{12}a_2 \\ b_2 &= S_{21}a_1 + S_{22}a_2 \end{aligned} \tag{2.2}$$

Equations (2.2) clearly demonstrate the linear relationship between S-parameters and the incident/reflected waves. However, when a DUT is operated under large-signal conditions, non-linear characteristics emerge. In this case, S-parameters fail to accurately capture the device behaviour beyond the linear region. Therefore, methods capable of extending analysis to non-linear behaviour are required. One such method is the Volterra Input Output Map (VIOMAP) model, which provides a modelling framework capable of including both linear and non-linear behaviour, regardless of whether the previous input signals are known.

2.1.2 Non-linear Behaviour Models

Before extracting the VIOMAP model, consider again the two-port DUT shown in Figure 2.1. Its equivalent system representation transforms the two ports into terminals, such that incident waves become the input signals and reflected waves become the output signals. Figure 2.2 illustrates this system equivalence.

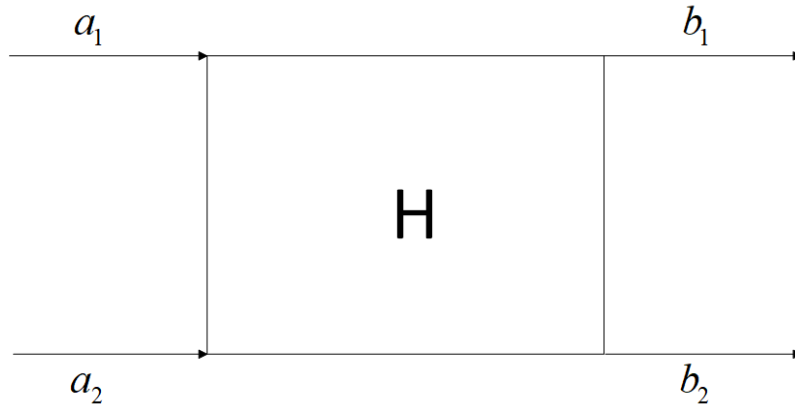


Figure 2.2: The system equivalence of a two-port DUT

For a purely linear DUT, the response can be expressed as in the equation (2.3). Similarly, the VIOMAP model, with Volterra kernel $H_{n,ji}$, can be expressed by

equation (2.4).

$$\begin{aligned}
b_1(1) &= S_{11}(1)a_1(1) + S_{12}(1)a_2(1) \\
b_2(1) &= S_{21}(1)a_1(1) + S_{22}(1)a_2(1) \\
b_1(3) &= 0 \\
b_2(3) &= 0 \quad \text{with } 1 = f_0, \text{ and } 3 = 3f_0
\end{aligned} \tag{2.3}$$

$$\begin{aligned}
b_1(1) &= H_{1,11}(1)a_1(1) + H_{1,12}(1)a_2(1) \\
b_1(3) &= 0
\end{aligned} \tag{2.4}$$

Because this is a linear system, the third-order Volterra kernel is zero. Then, for a non-linear DUT, the $H_{n,j i_1 i_2 \dots i_n}(f_1, f_2, \dots, f_n)$ represents the n th-order Volterra kernel of the system H under the frequency f_k of input i_k to the output terminal j of frequency $f_1 + f_2 + \dots + f_n$. Equations (2.5) and (2.6) illustrate this for a system including third-order non-linearities.

$$\begin{aligned}
b_1(1) &= H_{1,1}(1)a_1(1) \\
&+ H_{1,12}(1)a_2(1) \\
&+ 3H_{3,1111}(1, 1, -1)a_1(1)a_1(1)a_1(-1) \\
&+ 3H_{3,1112}(1, 1, -1)a_1(1)a_1(1)a_2(-1) \\
&+ 6H_{3,1112}(1, -1, 1)a_1(1)a_1(-1)a_2(1) \\
&+ 6H_{3,1122}(1, -1, 1)a_1(1)a_2(-1)a_2(1) \\
&+ 3H_{3,1222}(1, 1, -1)a_1(1)a_2(1)a_2(-1) \\
&+ 3H_{3,1222}(1, 1, -1)a_2(1)a_2(1)a_2(-1)
\end{aligned} \tag{2.5}$$

$$\begin{aligned}
b_1(3) = & H_{3,1111}(1, 1, 1)a_1(1)a_1(1)a_1(1) \\
& + 3H_{3,1112}(1, 1, 1)a_1(1)a_1(1)a_2(1) \\
& + 3H_{3,1122}(1, 1, 1)a_1(1)a_2(1)a_2(1) \\
& + H_{3,1222}(1, 1, 1)a_2(1)a_2(1)a_2(1)
\end{aligned} \tag{2.6}$$

Nevertheless, when applied to strongly non-linear systems, the VIOMAP model requires a highly complex calculation process. To overcome some of this complexity, the PHD modelling method provides a faster and more standardised approach to non-linear model extraction.

The key principle of PHD modelling is to drive a strongly non-linear system with a sufficiently large input signal, forcing it into a periodic steady-state large-signal operating condition. Under this condition, small perturbations such as delays around the operating point can be approximated as linear. As noted in [8], PHD modelling can be regarded as a black-box approach. The basic quantities are travelling-wave spectral components, where A denotes an incident wave and B denotes a scattered wave. Thus, the relationship between the incident spectral components and a scattered spectral component can be expressed as

$$B_{pm} = F_{pm}(A_{11}, A_{12}, A_{21}, A_{22}, \dots) \tag{2.7}$$

where $F_{pm}()$ is the black-box. Before analysing the black-box, one quantity needs to be defined: the phasor, defined as

$$P = e^{+j\phi(A_{11})}. \tag{2.8}$$

The concept of the phasor is then introduced in order to normalise the phase of the

input signal and simplify the process of model extraction. This ensures that the first input argument is always positive, as shown in equation (2.9)

$$\begin{aligned}
 B_{pm}e^{jm\theta} &= F_{pm} (A_{11}e^{j\theta}, A_{12}e^{j2\theta}, \dots, A_{21}e^{j\theta}, A_{22}e^{j2\theta}, \dots) \\
 &= F_{pm} (|A_{11}|, A_{12}P^{-2}, A_{13}P^{-3}, \dots, A_{21}P^{-1}, A_{22}P^{-2}, \dots) P^{+m}.
 \end{aligned}
 \tag{2.9}$$

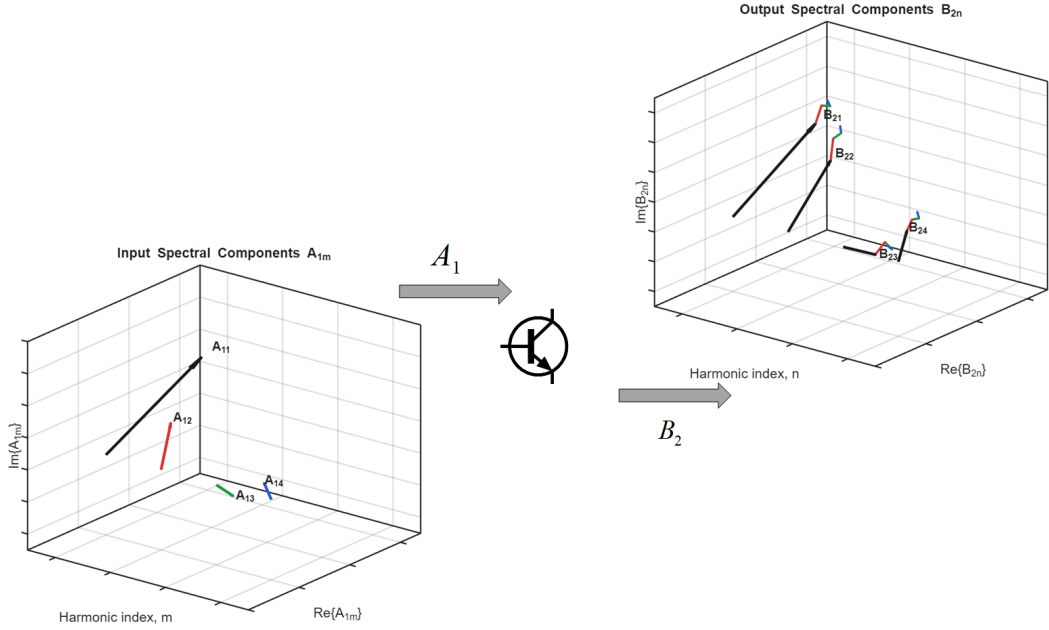


Figure 2.3: The harmonic superposition principle

Figure 2.3 illustrates the harmonic superposition principle, where A_{11} is the dominant large-signal input component and the smaller harmonic components A_{12} , A_{13} , and A_{14} produce corresponding perturbations in the output spectral components B_{2n} . Based on the harmonic superposition principle [32], most components of equation (2.9), except for the large-signal term A_{11} , can be linearised. And the equation

(2.9) becomes

$$\begin{aligned}
p_{pm} &= K_{pm}(A_{11})P^{+m} \\
&+ \sum_{qn} G_{pq,mn}(A_{11})P^{+m} \operatorname{Re} (A_{pq}P^{-n}) \\
&+ \sum_{qn} H_{pq,mn}(A_{11})P^{+m} \operatorname{Im} (A_{pq}P^{-n}) ,
\end{aligned} \tag{2.10}$$

whereby

$$\begin{aligned}
K_{pm}(A_{11}) &= F_{pm}(A_{11}, 0, \dots, 0), \\
G_{pq,mn}(A_{11}) &= \left. \frac{\partial F_{pm}}{\partial \operatorname{Re} (A_{qn}P^{-n})} \right|_{A_{11}, 0, \dots, 0}, \\
H_{pq,mn}(A_{11}) &= \left. \frac{\partial F_{pm}}{\partial \operatorname{Im} (A_{qn}P^{-n})} \right|_{A_{11}, 0, \dots, 0}.
\end{aligned} \tag{2.11}$$

By substituting the $\operatorname{Re} (A_{pq}P^{-n})$ and $\operatorname{Im} (A_{pq}P^{-n})$ with the following equations

$$\begin{aligned}
\operatorname{Re} (A_{qn}P^{-n}) &= \frac{A_{qn}P^{-n} + \operatorname{conj} (A_{qn}P^{-n})}{2}, \\
\operatorname{Im} (A_{qn}P^{-n}) &= \frac{A_{qn}P^{-n} - \operatorname{conj} (A_{qn}P^{-n})}{2j}.
\end{aligned} \tag{2.12}$$

Equation (2.9) can be transformed into

$$\begin{aligned}
B_{pm} &= K_{pm}(A_{11})P^{1+m} + \sum_{qn} C_{pq,mn}(A_{11})P^{1+m} \\
&\quad \times \left(\frac{A_{qn}P^{-n} + \overline{A_{qn}P^{-n}}}{2} \right) \\
&\quad + \sum_{qn} \Pi_{pq,mn}(A_{11}) \\
&\quad \times P^{1+m} \left(\frac{A_{qn}P^{-n} - \overline{A_{qn}P^{-n}}}{2j} \right) \\
&= \sum_{qn} S_{pq,mn}(A_{11})P^{1+m-n}A_{qn} \\
&\quad + \sum_{qn} T_{pq,mn}(A_{11})P^{1+m+n}\overline{A_{qn}}.
\end{aligned} \tag{2.13}$$

Note that two new functions, $S_{pq,mn}(\cdot)$ and $T_{pq,mn}(\cdot)$, are introduced. They are defined as

$$\begin{aligned}
S_{p1,m1}(A_{11}) &= \frac{K_{pm}(A_{11})}{|A_{11}|}, \\
T_{p1,m1}(A_{11}) &= 0 \\
\forall \{q, n\} \neq \{1, 1\} : S_{pq,mn}(A_{11}) &= \frac{C_{pq,mn}(A_{11}) - jH_{pq,mn}(A_{11})}{2}, \\
\forall \{q, n\} \neq \{1, 1\} : T_{pq,mn}(A_{11}) &= \frac{C_{pq,mn}(A_{11}) + jH_{pq,mn}(A_{11})}{2}.
\end{aligned} \tag{2.14}$$

Moreover, $T_{pq,mn}(\cdot)$ is zero because

$$P^{+m-1}A_{11} = P^{+m+1}\text{conj}(A_{11}) = |A_{11}|. \tag{2.15}$$

Building on PHD theory, X-parameters were developed as an industry-standard non-linear behavioural modelling approach. X-parameters can be seen as the practical implementation of PHD theory. The general form of the X-parameter, given

in equation (2.16), clearly shows its correspondence to the PHD modelling equation (2.13).

$$B_{p,k} = X_{p,k}^{(F)} + \sum_{q=1}^P \sum_{m=1}^N [s_{p,k;q,m} \cdot \Delta A_{q,m} + t_{p,k;q,m} \cdot \Delta A_{q,m}^*] \quad (2.16)$$

In particular, for the fundamental output response, equation (2.16) can be written as

$$B_{2,1} = X_{2,1}^{(F)} + S_{2,1;1,1} \cdot \Delta A_{1,1} + T_{2,1;1,1} \cdot \Delta A_{1,1}^* + S_{2,1;2,1} \cdot \Delta A_{2,1} + T_{2,1;2,1} \cdot \Delta A_{2,1}^* \quad (2.17)$$

Table 2.1 provides a comparison of corresponding functional components between PHD modelling formulas and the general form of X-parameters.

Table 2.1: Correspondence between PHD modelling formulas and X-Parameter general form

PHD Term	X-Parameter	Meaning
$K_{pm}(A_{11})P^{+m}$	$X_{p,m}^{(F)}$	Large-signal steady-state response
$S_{pq,mn}(A_{11})P^{+m-n}A_{qn}$	$S_{p,m;q,n} \cdot \Delta A_{q,n}$	Linear scattering term
$T_{pq,mn}(A_{11})P^{+m+n} \text{conj}(A_{qn})$	$T_{p,m;q,n} \cdot \Delta A_{q,n}^*$	non-linear dispersion term

As stated by Verspecht [8], the X-parameter model represents a linearization of a non-linear RF device about a large-signal periodic steady-state operating point. The term $X_{p,m}^{(F)}$ describes the large-signal response spectrum generated by the fundamental excitation alone. Small deviations from the large-signal operating point are expressed using incremental incident waves $\Delta A_{q,n}$, leading to the linear scattering coefficients $S_{p,m;q,n}$, which characterise the device response to small-signal perturbations in the presence of a large-signal excitation. Moreover, the conjugate coefficients $T_{p,m;q,n}$ represent the dependence of the response on the complex conjugate of the in-

cident waves, capturing the non-analytic behaviour of large-signal-driven non-linear systems. All in all, these terms set up a first-order polyharmonic representation that accurately predicts the behaviour of the device close to the large-signal operating point.

2.1.3 S-parameter and Non-linear Behaviour Models

Discussion

S-parameters can accurately represent a behavioural model of a DUT operating under the linear condition and are independent of measurement systems. S-parameters are widely supported by RF and microwave CAD and measurement software tools and have become an industry standard for the analysis and design of linear circuits [30]. However, under large-signal operating conditions, the DUT exhibits non-linear behaviour and the linear relationship between incident and scattered waves assumed by conventional S-parameters is no longer sufficient to describe its response. Models like VIOMAP were developed to solve these problems. Even though VIOMAP models can take into account both linear and non-linear characteristics, they also have their own limitations, such as complicated calculation and the lack of an industrial standard. PHD modelling, and its practical implementation through X-parameters, addresses this limitation by linearising the response to small perturbations around a large-signal operating point, thereby providing a practical framework for modelling non-linear RF behaviour.

Nevertheless, X-parameters still have some limitations. When the device response exhibits strong non-linearity within a local region of the operating space, the standard X-parameter formulation may become less flexible. Furthermore, X-parameter characterisation requires specialised large-signal measurement capability, including the generation and measurement of small perturbation, or “tickle-tone”, signals around the large-signal operating point. Such capability is not available in all measurement facilities, which can limit the practical accessibility of X-parameter extraction. The amount of measurement data and the resulting model file size may

also increase as the characterised operating space becomes larger. In this case, models such as the CM can provide a more compact representation.

2.2 The Cardiff Model

Accessing device behaviour in a CAD environment using large-signal waveforms can greatly assist researchers in design tasks. By systematically changing parameters such as input power, biasing, and fundamental frequencies, researchers can optimise designs to meet specifications. However, this process often requires exploring a large measurement space and can reduce CAD simulation speed. To address these challenges, the Cardiff Direct Waveform Lookup Table (DWLU) approach [33] was developed by H. Qi *et al.*

2.2.1 The Cardiff DWLU Model

The Cardiff DWLU model imports measurement data into CAD tools in the frequency domain. Figure 2.4 illustrates the Cardiff DWLU model [34].

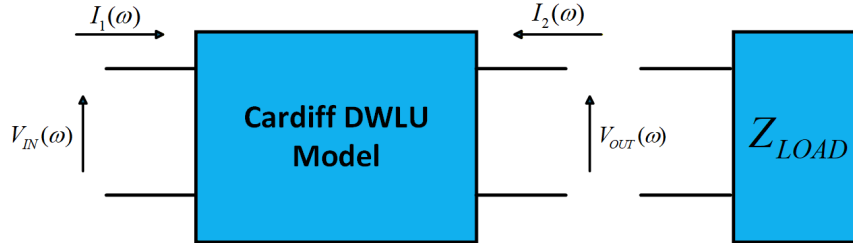


Figure 2.4: Definition of the Cardiff DWLU model in frequency domain

The port current $I_1(\omega)$ and $I_2(\omega)$ in Figure 2.4 are system responses to the input and output voltages, $V_{IN}(\omega)$ and $V_{OUT}(\omega)$. The equations (2.18) and (2.19) describe the current responses at the two ports for a given load impedance Z_{LOAD} . Here, A_0 and B_0 represent DC components, n is the order of harmonic, the V_{IN} is the input voltage stimulus, the f_0 is the fundamental frequency.

$$I_1(\omega) = A_0 \cdot \delta(\omega) + \sum_{n=1}^m A_n \cdot V_{IN}^n \cdot \delta(\omega - 2\pi \cdot n \cdot f_0) \quad (2.18)$$

$$I_2(\omega) = B_0 \cdot \delta(\omega) + \sum_{n=1}^m B_n \cdot V_{IN}^n \cdot \delta(\omega - 2\pi \cdot n \cdot f_0), \quad (2.19)$$

The coefficients A_n and B_n are related to I_1 and I_2 through equations (2.18) and (2.19). Changes in V_{IN} , Z_{LOAD} and biases affect both voltages and currents; thus, the coefficients A_n and B_n can be treated as functions of those parameters. Considering each harmonic separately, two matrices can be derived:

$$\begin{aligned} \begin{bmatrix} I_{10} \\ I_{11} \\ I_{12} \\ I_{13} \end{bmatrix} &= \begin{bmatrix} A_0 & 0 & 0 & 0 \\ 0 & A_1 & 0 & 0 \\ 0 & 0 & A_2 & 0 \\ 0 & 0 & 0 & A_3 \end{bmatrix} \begin{bmatrix} 1 \\ V_{IN} \\ V_{IN}^2 \\ V_{IN}^3 \end{bmatrix}, \\ \begin{bmatrix} I_{20} \\ I_{21} \\ I_{22} \\ I_{23} \end{bmatrix} &= \begin{bmatrix} B_0 & 0 & 0 & 0 \\ 0 & B_1 & 0 & 0 \\ 0 & 0 & B_2 & 0 \\ 0 & 0 & 0 & B_3 \end{bmatrix} \begin{bmatrix} 1 \\ V_{IN} \\ V_{IN}^2 \\ V_{IN}^3 \end{bmatrix}. \end{aligned} \quad (2.20)$$

In the matrices, the I_{kn} denotes the current at port k and harmonic index n . The equations (2.20) can be expressed as :

$$A_n = \frac{I_1(nf_0)}{V_{IN}^n(nf_0)} = F_1(|V_{IN}|, I_{LOAD}, V_{DC \text{ IN}}, V_{DC \text{ OUT}}) \quad (2.21)$$

$$B_n = \frac{I_2(nf_0)}{V_{IN}^n(nf_0)} = F_2(|V_{IN}|, I_{LOAD}, V_{DC \text{ IN}}, V_{DC \text{ OUT}}) \quad (2.22)$$

The basic modelling data consists of two-port voltages and currents, as shown in equations (2.21) and (2.22). Both a/b-waves and voltages/currents can be used to fit coefficients to measurement data. However, voltages and currents are chosen because the CAD environment (Keysight ADS) and its FDD component can only process voltages and currents. Thus, using voltages and currents ensures a smoother CAD implementation.

2.2.2 The Cardiff Model Formulation

In PHD modelling, the harmonic superposition principle assumes that A_1 is the dominant large-signal input component, while the other incident-wave components remain relatively small [32]. However, when optimum power, gain, or efficiency impedances is far from 50Ω system impedance, and the reflected wave a_2 can become large and this assumption may no longer be valid. To overcome this limitation, the Cardiff Behavioural Model was developed, extending the DWLU and PHD approaches.

First, consider a two-port network as in Figure 2.1. The describing functions $f()$ and $g()$ express the relationships between a-waves and b-waves:

$$\begin{aligned} b_1 &= f(a_1, a_2) \\ b_2 &= g(a_1, a_2). \end{aligned} \tag{2.23}$$

If b_p depends on a_1 and a_2 , then b_p can be expressed as a function of a_1 multiplied by a function of a_2 :

$$\begin{aligned} b_1 &= f(a_1)f(a_2) \\ b_2 &= g(a_1)g(a_2). \end{aligned} \tag{2.24}$$

For CAD implementation, (2.24) is approximated using n th-order polynomials.

A third-order formulation of (2.24) is illustrated here:

$$b_1 = \varphi_{10}a_1 + \varphi_{01}a_2 + \varphi_{11}a_1a_2 + \cdots + \varphi_{30}a_1^3 + \varphi_{03}a_2^3 \quad (2.25)$$

$$b_2 = \theta_{10}a_1 + \theta_{01}a_2 + \theta_{11}a_1a_2 + \cdots + \theta_{30}a_1^3 + \theta_{03}a_2^3, \quad (2.26)$$

where the $\varphi_{n_1n_2}$ and $\theta_{n_1n_2}$ are constant complex coefficients.

By regrouping and substituting [35], the equations (2.25) and (2.26) transform into:

$$\begin{aligned} b_1 &= (\varphi_{10} + 3\varphi_{30}|a_1|^2 + \varphi_{12}|a_2|^2)a_1 \\ &\quad + (\varphi_{12}|a_2|^2)a_1^*p^2 \\ &\quad + (\varphi_{01} + 3\varphi_{03}|a_2|^2 + \varphi_{21}|a_1|^2)a_2 \\ &\quad + (\varphi_{21}|a_1|^2)a_2^*q^2 \end{aligned} \quad (2.27)$$

$$\begin{aligned} b_2 &= (\theta_{00} + 3\theta_{30}|a_1|^2 + \theta_{12}|a_2|^2)a_1 \\ &\quad + (\theta_{12}|a_2|^2)a_1^*p^2 \\ &\quad + (\theta_{01} + 3\theta_{03}|a_2|^2 + \theta_{21}|a_1|^2)a_2 \\ &\quad + (\theta_{21}|a_1|^2)a_2^*q^2, \end{aligned} \quad (2.28)$$

where $p = e^{j\angle a_2}$ and $q = e^{j\angle a_1}$ and p is the phase of a_2 and q is the phase of a_1 , a_n^* is the conjugate of a_n . Summarising the fundamental terms as S and T , equations (2.27) and (2.28) reduce to

$$b_1 = S_{11}a_1 + T_{11}a_1^*p^2 + S_{12}a_2 + T_{12}a_2^*q^2 \quad (2.29)$$

$$b_2 = S_{21}a_1 + T_{21}a_1^*p^2 + S_{22}a_2 + T_{22}a_2^*q^2 \quad (2.30)$$

where S and T are

$$\begin{aligned}
S_{11} &= \varphi_{10} + 3\varphi_{30}|a_1|^2 + \varphi_{12}|a_2|^2, & T_{11} &= \varphi_{12}|a_2|^2 \\
S_{12} &= \varphi_{01} + 3\varphi_{03}|a_2|^2 + \varphi_{21}|a_1|^2, & T_{12} &= \varphi_{21}|a_1|^2 \\
S_{21} &= \theta_{10} + 3\theta_{30}|a_1|^2 + \theta_{12}|a_2|^2, & T_{21} &= \theta_{12}|a_2|^2 \\
S_{22} &= \theta_{01} + 3\theta_{03}|a_2|^2 + \theta_{21}|a_1|^2, & T_{22} &= \theta_{21}|a_1|^2.
\end{aligned} \tag{2.31}$$

Another important point raised by Qi in [35] is the renormalisation of travelling waves. Usually, the travelling waves are normalised to the system characteristic impedance, which is 50Ω . However, when the impedance of the optimum power point is far from the system characteristic impedance, the magnitude of the travelling wave a_2 can become large. Renormalising the travelling waves to an impedance close to the optimum power point can reduce the magnitude of a_2 and provide a more convenient representation of the travelling-wave space. Although this changes the numerical values of the travelling-wave and model quantities, it does not change the device behaviour. The renormalization expressions for travelling waves are shown below:

$$a_{\text{Renorm}} = \frac{V + Z_{\text{opt}}I}{2} \cdot \frac{\sqrt{\text{Re}(Z_{\text{opt}})}}{|Z_{\text{opt}}|} \tag{2.32}$$

$$b_{\text{Renorm}} = \frac{V - Z_{\text{opt}}I}{2} \cdot \frac{\sqrt{\text{Re}(Z_{\text{opt}})}}{|Z_{\text{opt}}|}. \tag{2.33}$$

Based on the work done by Qi, Woodington developed the Cardiff Behavioural Model in [11], [36], [37] and Tasker summarises the modelling efforts in [10]. Woodington renamed and defined those constant complex coefficients from (2.29) and

(2.30):

$$b_1 = S_{11}a_1 + T_{11}a_1^*Q^2 + S_{12}a_2 + T_{12}a_2^*P^2 \quad (2.34)$$

$$b_2 = S_{21}a_1 + T_{21}a_1^*Q^2 + S_{22}a_2 + T_{22}a_2^*P^2, \quad (2.35)$$

where Q and P are the respective fundamental input and output a-wave phase operators, given by

$$P = \frac{a_1}{|a_1|}, Q = \frac{a_2}{|a_2|}. \quad (2.36)$$

This work clearly shows that for every S and T coefficient, they have their own phase vector and can be extracted independently with respective measured b-waves as shown here:

$$\begin{aligned} S_{11} \cdot |a_1| &= \frac{1}{n} \sum b_1 \cdot \frac{1}{P}, & T_{11} \cdot |a_1| &= \frac{1}{n} \sum b_1 \cdot \frac{P}{Q^2} \\ S_{12} \cdot |a_2| &= \frac{1}{n} \sum b_1 \cdot \frac{1}{Q}, & T_{12} \cdot |a_2| &= \frac{1}{n} \sum b_1 \cdot \frac{Q}{P^2} \\ S_{21} \cdot |a_1| &= \frac{1}{n} \sum b_2 \cdot \frac{1}{Q}, & T_{21} \cdot |a_1| &= \frac{1}{n} \sum b_2 \cdot \frac{P}{Q^2} \\ S_{22} \cdot |a_2| &= \frac{1}{n} \sum b_2 \cdot \frac{1}{Q}, & T_{22} \cdot |a_2| &= \frac{1}{n} \sum b_2 \cdot \frac{Q}{P^2}. \end{aligned} \quad (2.37)$$

Then, extracting the phase of a_1 and keeping other parts, the equations (2.34) and (2.35) will transform into

$$b_1 = Q_{11} (T_{12}|a_2|\Phi^{-1} + S_{11}|a_1| + S_{12}|a_2|\Phi + T_{11}|a_1|\Phi^2) \quad (2.38)$$

$$b_2 = Q_{11} (T_{22}|a_2|\Phi^{-1} + S_{21}|a_1| + S_{22}|a_2|\Phi + T_{21}|a_1|\Phi^2) \quad (2.39)$$

where

$$\Phi = e^{j(\angle a_2 - \angle a_1)} = \frac{Q_{21}}{Q_{11}} \quad (2.40)$$

Using (2.38), the phase and magnitude exponent parameters can be derived systematically and are summarised in Tables 2.2 and 2.3. The fifth- and seventh-order cases are shown here as representative examples, and the same procedure can be extended to higher odd orders if required.

Table 2.2: Phase and magnitude exponent parameters corresponding to (2.38).

	Φ^{-6}	Φ^{-5}	Φ^{-4}	Φ^{-3}	Φ^{-2}	Φ^{-1}	Φ^0	Φ^1	Φ^2	Φ^3	Φ^4	Φ^5	Φ^6	Φ^7
$ a_{21} ^0$							1 st							
$ a_{21} ^1$						3 rd		1 st						
$ a_{21} ^2$					3 rd		3 rd		3 rd					
$ a_{21} ^3$				5 th		3 rd		3 rd		3 rd				
$ a_{21} ^4$			5 th		5 th		5 th		5 th		5 th			
$ a_{21} ^5$		7 th		5 th		5 th		5 th		5 th		5 th		
$ a_{21} ^6$	7 th		7 th		7 th		7 th		7 th		7 th		7 th	
$ a_{21} ^7$		7 th		7 th		7 th		7 th		7 th		7 th		7 th

Table 2.3: Phase and magnitude exponent parameters corresponding to (2.39).

	Φ^{-3}	Φ^{-2}	Φ^{-1}	Φ^0	Φ^1	Φ^2	Φ^3	Φ^4
$ a_{21} ^0$				1 st				
$ a_{21} ^1$			3 rd		1 st			
$ a_{21} ^2$		5 th		3 rd		3 rd		
$ a_{21} ^3$	7 th		5 th		3 rd		5 th	
$ a_{21} ^4$		7 th		5 th		5 th		7 th
$ a_{21} ^5$			7 th		5 th		7 th	
$ a_{21} ^6$				7 th		7 th		
$ a_{21} ^7$					7 th			

By regrouping and merging magnitude terms, the equations (2.38) and (2.39) can be combined together to get a general form for b-waves:

$$b_p = \sum_{m=0}^{\frac{n-1}{2}} C_{p,m} \left(\frac{Q_{21}}{Q_{11}} \right)^m a_1 + \sum_{m=0}^{\frac{n-1}{2}} U_{p,m} \left(\frac{Q_{11}}{Q_{21}} \right)^m a_2 \quad (2.41)$$

where n is the order of polynomial used in model, $C_{p,m}$ and $U_{p,m}$ are polynomial combinations of magnitudes of a_1 and a_2 , p is the port number. Including the harmonic factor extends the equation into:

$$b_{p,h} = Q_{11}^h \sum_{n=-\frac{ord-1}{2}}^{\frac{ord+1}{2}} K_{p,h,n} \left(\frac{Q_{21}}{Q_{11}} \right)^n \quad (2.42)$$

where

$$K_{p,h,n} = \sum_{m=0}^{m=q} M_{p,h,n,m} |q_{21}|^m \quad (2.43)$$

and the $M_{p,h,n,m}$ are the model coefficients. From this work, the general form of the CM can be expressed as:

$$B_{p,h}(|A_{1,1}|) = Q_{1,1}^h \left(\sum_{r=0}^{r=\frac{ord-h}{2}} \sum_{n=h+\frac{ord-h}{2}-r}^{n=h+\frac{ord-h}{2}-r} M_{p,h,m,n} (|A_{1,1}|) |A_{2,1}|^m \left(\frac{Q_{2,1}}{Q_{1,1}} \right)^n \right) \quad (2.44)$$

The parameters are well defined by Tasker in [10] with the mixing theory and the least squared approximation. The phase exponent parameter n , which can be calculated with the mixing order parameter ord and the harmonic h , ranges from $-\left(\frac{(ord-h)}{2} - r\right)$ to $h + \left(\frac{(ord-h)}{2} - r\right)$, where r is the magnitude restricting term that ranges from 0 to $\frac{(ord-h)}{2}$. If the input dataset has been pre-normalised by the phase of A_{11} , the equation (2.44) can be simplified into

$$B_{p,h} = \left(\sum_{r=0}^{max_r} \sum_{n=h+\frac{ord-h}{2}-r}^{n=h+\frac{ord-h}{2}-r} M_{p,h,m,n} |A_{2,1}|^m (Q_{2,1})^n \right). \quad (2.45)$$

In addition, Tasker also expands the equation (2.44) to include the additional

varying stimuli at the relevant harmonics under harmonic load-pull.

$$b_{n,h} = P_1^h \cdot \sum_{m_r=-\frac{w_r+1}{2}}^{m_r=+\frac{w_r+1}{2}} \cdots \sum_{m_1=-\frac{w_1+1}{2}}^{m_1=+\frac{w_1+1}{2}} K_{n,h,m_1,\dots,m_r} \cdot \left(\frac{Q_1}{P_1}\right)^{m_1} \cdots \left(\frac{Q_r}{P_1^r}\right)^{m_r} \quad (2.46)$$

where $P_1 = \angle a_{1,1} = \frac{a_{1,1}}{|a_{1,1}|}$ and $Q_r = \angle a_{2,r} = \frac{a_{2,r}}{|a_{2,r}|}$.

2.2.3 The Cardiff Model Extraction

In (2.44), the coefficient $M_{p,h,n,m}$ is the key parameter for implementing the CM. Several extraction methods have been proposed [37], [38]. Recently, Tasker [39] introduced a direct extraction method based on fundamental load-pull measurements, while Tian [40] applied neural networks. Nonetheless, least-squares approximation remains the standard.

Based on equation (2.44), the coefficient $M_{p,h,n,m}$ can be extracted with this expression:

$$\begin{aligned} [M_{p,h,m,n}] &= \left(|A_{2,1}|^m \left(\frac{Q_{2,1}}{Q_{1,1}} \right) \right)^H \left(|A_{2,1}|^m \left(\frac{Q_{2,1}}{Q_{1,1}} \right)^n \right)^{-1} \\ &\times \left(|A_{2,1}|^m \left(\frac{Q_{2,1}}{Q_{1,1}} \right)^n \right)^H [B_{p,h}] \end{aligned} \quad (2.47)$$

As mentioned in the last section, when input data is pre-normalised and only the fundamental frequency is considered, (2.47) simplifies to:

$$[M_{p,h,m,n}] = ([A]^H[A])^{-1} [A]^H [B_{p,h}] \quad (2.48)$$

where the $[A]$ is usually called the A-matrix consisting of $|A_{21}|^m(Q_{21})^n$ and the m, n corresponding to Table 2.2, the exponents of phase and magnitude of A_{21} . Since this thesis focuses on the fundamental frequency, determining the number of coefficients is critical. Bell [38], [41], [42] refined Woodington's work [36] by enforcing symmetry

in the coefficient distribution and applying filters to preserve model accuracy without overly increasing model complexity. The marker size in Figure 2.5 indicates the relative contribution of each coefficient term to the model response. It accounts for both the coefficient magnitude and the associated A -wave magnitude raised to the corresponding polynomial order, rather than representing the coefficient magnitude alone.

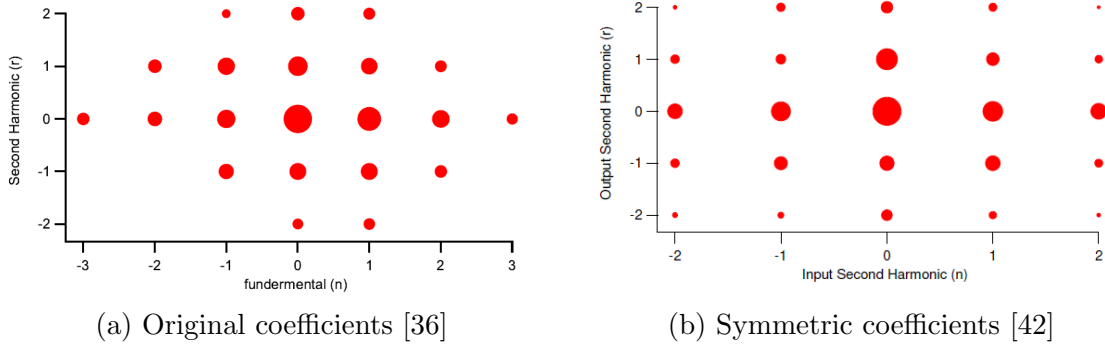
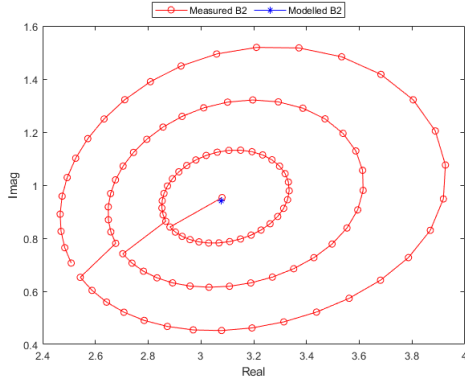
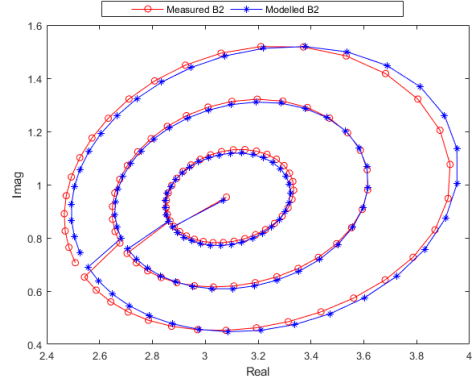


Figure 2.5: The input second harmonic and output second harmonic coefficient space

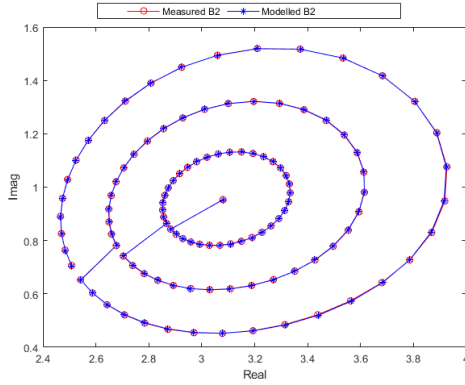
Furthermore, Bell's work shows that increasing the number of coefficients improves the model fit, but the improvement becomes limited as the model complexity increases. Figure 2.6 shows that six coefficients provide good agreement between the measured and modelled b_{21} , while increasing the number to nine gives little additional improvement.



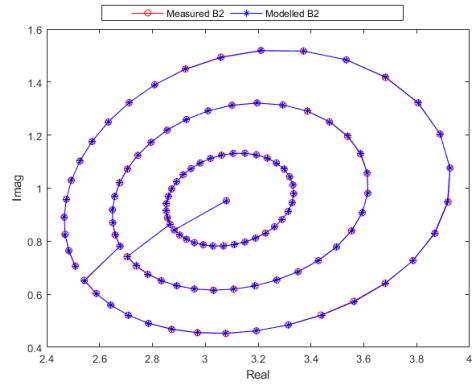
(a) 1 Coefficient



(b) 3 Coefficients



(c) 6 Coefficients



(d) 9 Coefficients

Figure 2.6: Comparison of measured and modelled b_{21} with a Cree device and different numbers of coefficients

Note that the coefficients in (2.45) are obtained by least-squares approximation.

2.2.4 The Cardiff Model CAD Implementation

In [42], Bell improves the method using Application Extension Language (AEL), a scripting and extension language used in ADS, to smoothly and automatically implement the CM into a CAD environment. It consists of different parts and starts with the core component in ADS.

Core Circuit

A four-port FDD, shown in Figure 2.7, is the core of the CM implementation.

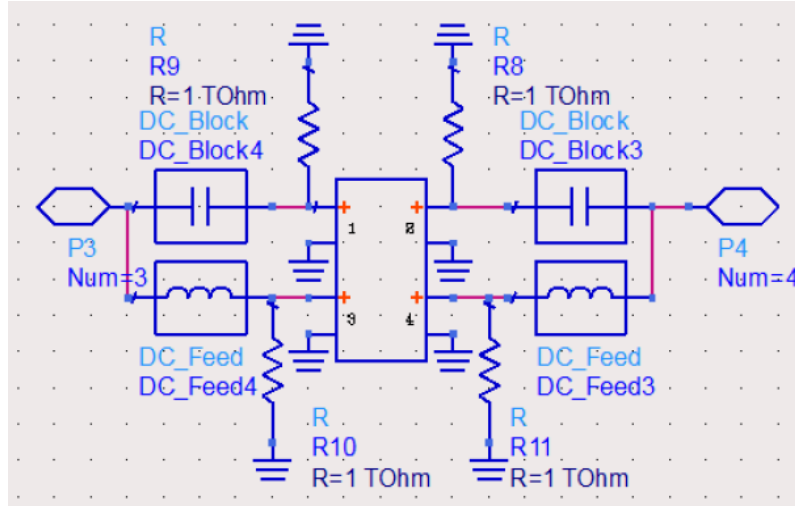


Figure 2.7: FDD core of the model schematic

The FDD acts as an interface between the ADS circuit and the behavioural model implementation. It receives the harmonically indexed port voltages and currents from the ADS circuit and returns the corresponding harmonically indexed port voltages and currents after the required model calculations have been performed. The conversion between port voltages/currents and travelling waves, together with the behavioural model calculations, is implemented separately using equations within the schematic. Figure 2.8 shows the equations used in this implementation to combine the modelled wave quantities with the phase of A_{11} .

```

VAR
VI_Functions
Veq(a,B,Zn,Pha)=(a+B)*(cos(Pha)+j*sin(Pha))
FDD4P
FDD2P1_
I[3,0]= B10
V[1,1]= Veq(V11,B11,Phase)
V[1,2]=Veq(V12,B12,Phase*2)
V[1,3]=Veq(V13,B13,Phase*3)
I[4,0]=B20
V[2,1]=Veq(V21,B21,Phase)
V[2,2]=Veq(V22,B22,Phase*2)
V[2,3]=Veq(V23,B23,Phase*3)

```

Figure 2.8: The FDD functions

Data Import

After the core part of the implementation, how the generated model files are imported and how the coefficients are assigned to ADS variables separately also matter. The Data Access Component (DAC) provides this function. As shown in Figure 2.9, the DAC can alternatively choose model file type and data import order, and also assign coefficients under different frequencies and biases.

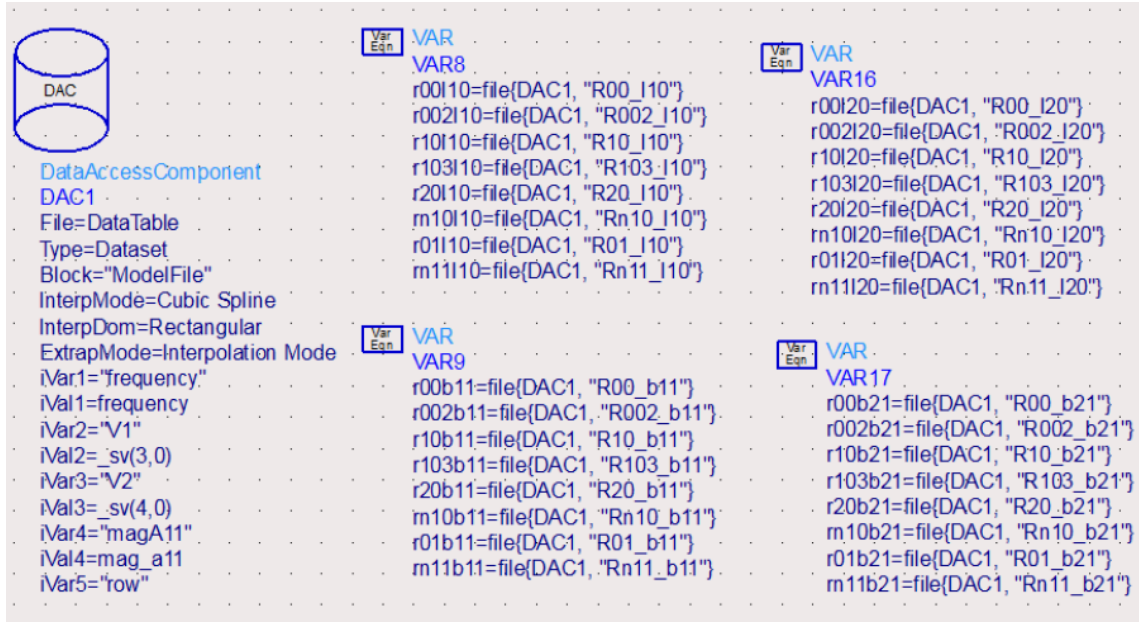


Figure 2.9: The DAC and file variables

Calculation Equations

In [42] Bell showed that the schematic window does not support Matrix-level calculation. Therefore, in order to achieve matrix calculations in the CAD environment, the matrix calculations are replaced by long-hand expressions and functions that calculate every element in the matrices.

The FDD provides a way to invoke the received voltages and currents by $_sv(port, harmonic)$ and $_si(port, harmonic)$. These allow the equations in Figure 2.10 to calculate the travelling waves and to normalise the phase of a_{11} to zero.

Var Eqn VAR Translation_Equations <pre> a11o=((_sv(1,1)+Z_11*_si(1,1))+1e-18)/2 a12o=((_sv(1,2)+Z_12*_si(1,2))+1e-18)/2 a13o=((_sv(1,3)+Z_13*_si(1,3))+1e-18)/2 a21o=((_sv(2,1)+Z_21*_si(2,1))+1e-18)/2 a22o=((_sv(2,2)+Z_22*_si(2,2))+1e-18)/2 a23o=((_sv(2,3)+Z_23*_si(2,3))+1e-18)/2 Phase=phaserad(a11o) </pre>	Var Eqn VAR PhaseNormalising <pre> a11=a11o/(cos(Phase)+j*sin(Phase)) a12=a12o/(cos(Phase*2)+j*sin(Phase*2)) a13=a13o/(cos(Phase*3)+j*sin(Phase*3)) a21=a21o/(cos(Phase)+j*sin(Phase)) a22=a22o/(cos(Phase*2)+j*sin(Phase*2)) a23=a23o/(cos(Phase*3)+j*sin(Phase*3)) </pre>
---	---

Figure 2.10: V-I to a-b translation equations and equations for phase normalisation

Note that the $1e^{-18}$ in the translation equations is used to avoid the errors caused by the appearance of zeros in further calculations.

Var Eqn VAR A_1 <pre> A1=Ap(0,0,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13) </pre>	Var Eqn VAR A_2 <pre> A2=Ap(1,1,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13) </pre>
---	---

Figure 2.11: A-element calculations

Figure 2.11 displays how to transform A-matrix calculations into long-hand formulas, and the $Ap(M, P, Aph)$ function is defined in Figure 2.12. This function cleverly avoids calculating the phase of a-waves to save simulation time. It also shows the b-wave calculations with the A-matrix and R-matrix using $[B] = [R] * [A]$. The R-matrix includes the coefficients imported from the model file through the DAC. Then, the B-matrix calculations use $Bpop(R, A)$ to calculate the b-waves corresponding to the equation (2.45).

Var Eqn	VAR Bmatrix B10=Bpop(R10,Amatrix) B11=Bpop(R11,Amatrix) B12=Bpop(R12,Amatrix) B13=Bpop(R13,Amatrix) B20=Bpop(R20,Amatrix) B21=Bpop(R21,Amatrix) B22=Bpop(R22,Amatrix) B23=Bpop(R23,Amatrix)	Var Eqn	VAR Rmatrix R10={r10u0,r10u1,r10u2,r10u3,r10u4,r10u5} R11={r11u0,r11u1,r11u2,r11u3,r11u4,r11u5} R12={r12u0,r12u1,r12u2,r12u3,r12u4,r12u5} R13={r13u0,r13u1,r13u2,r13u3,r13u4,r13u5} R20={r20u0,r20u1,r20u2,r20u3,r20u4,r20u5} R21={r21u0,r21u1,r21u2,r21u3,r21u4,r21u5} R22={r22u0,r22u1,r22u2,r22u3,r22u4,r22u5} R23={r23u0,r23u1,r23u2,r23u3,r23u4,r23u5}
Var Eqn	VAR Amatrix Ap(M,P,Aph)=(mag(Aph)**(M-P))*Aph**P Amatrix={A1,A2,A3,A4,A5,A6} Bpop(R,A)=R[1]*A[1]+R[2]*A[2]+R[3]*A[3]+R[4]*A[4]+R[5]*A[5]+R[6]*A[6]		

Figure 2.12: Construction of the B-matrix using the R-matrix, and A-matrix equations

In the end, in Figure 2.8, because the a-waves are phase normalised, the VI-Function adds the phase of A_{11} back and calculates the voltages for the respective harmonic orders.

AEL in ADS

The AEL is used to control an ADS schematic to add or remove all these components, variables, expressions, and functions in order. Usually, it will detect the target component's existence, then remove those existing components and add the required components onto the schematic. The figure below illustrates how the ADS simulator runs the AEL codes in simulations.

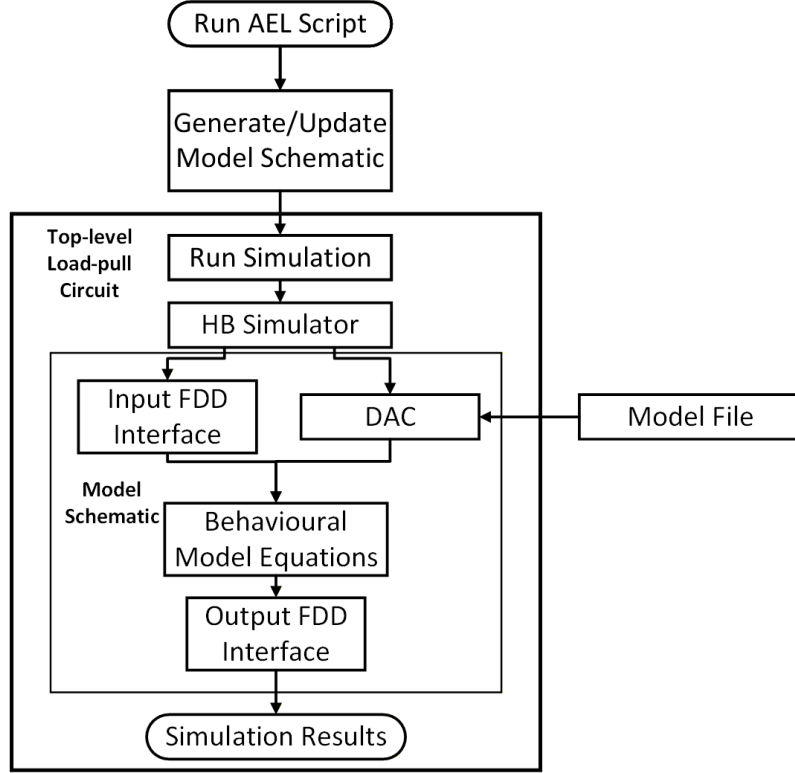


Figure 2.13: Flow diagram of ADS-AEL simulation

2.2.5 The Cardiff Model Discussion

The Cardiff DWLU model has been shown to agree very well with measurement data on a 100 W LDMOS device [43]. Its formulas directly display its original goal: importing measurement data into CAD environments for simulation. However, it does not show any relationship between the input and output of a DUT or the system response.

With the extended work based on the Cardiff DWLU model [11], [37], the Cardiff Behavioural Model has been improved. It solved the problem of increasing file sizes when facing complex non-linearities. The model can also be extracted using different methods for both fundamental and harmonic cases. In practice, the fundamental-frequency case provides a simpler and less time-consuming framework for model extraction and implementation. Additionally, selecting an appropriate number of coefficients can improve model efficiency by avoiding unnecessary model complexity, while the CAD implementation improves the usability of the model.

2.3 ANN Model

The Artificial Neural Network (ANN) is an information analysis model inspired by the biological neural networks of the human brain. Just as the brain uses neurons to process and transmit information, the ANN employs artificial neurons, also called nodes or units, to recognise patterns and learn relationships. Since the beginning of the 21st century, ANNs have been increasingly applied in RF design [44]. Early applications focused on modelling the electrical performance of devices and circuits, and later expanded to advanced design, simulation, and problem-solving tasks. Because ANNs can learn complex non-linear relationships, they serve as strong alternatives to numerical modelling methods and empirical models. Over time, ANNs have been widely used in various areas of RF and microwave applications, including embedded passives [12], transmission-line components [13], [14], [15], [16], coplanar waveguide (CPW) components [17], [18], spiral inductors [19], [20], field-effect transistors (FETs) [21], [22], and power amplifiers [23], [24]. Neural networks have also been applied in impedance matching [25], inverse modelling [26], measurements [27], and circuit synthesis [28]. Within behavioural modelling, previous studies have mainly focused on improving model structure, training methods, and prediction accuracy. In comparison, the work in this thesis focuses on the practical implementation of an ANN behavioural model in ADS and its simulation efficiency [45]. A CAD-based schematic generation framework is developed to dynamically adapt the ANN implementation according to the required model complexity. The runtime performance of the ANN and CM implementations is then compared under load-pull simulation conditions to evaluate the proposed framework and investigate the computational efficiency of both implementations.

2.3.1 Model Structure

Selecting an appropriate ANN structure offers several possibilities, such as the standard Multilayer Perceptron (MLP), the Knowledge-Based Neural Network (KBNN),

and the Combined Network [46]. In general, once a structure is chosen, the network operates in a feed-forward manner to process the input data and approximate the desired output, using different neuron topologies for a wide range of RF-related applications [45], [46], [47].

Standard Multilayer Perceptrons (MLP)

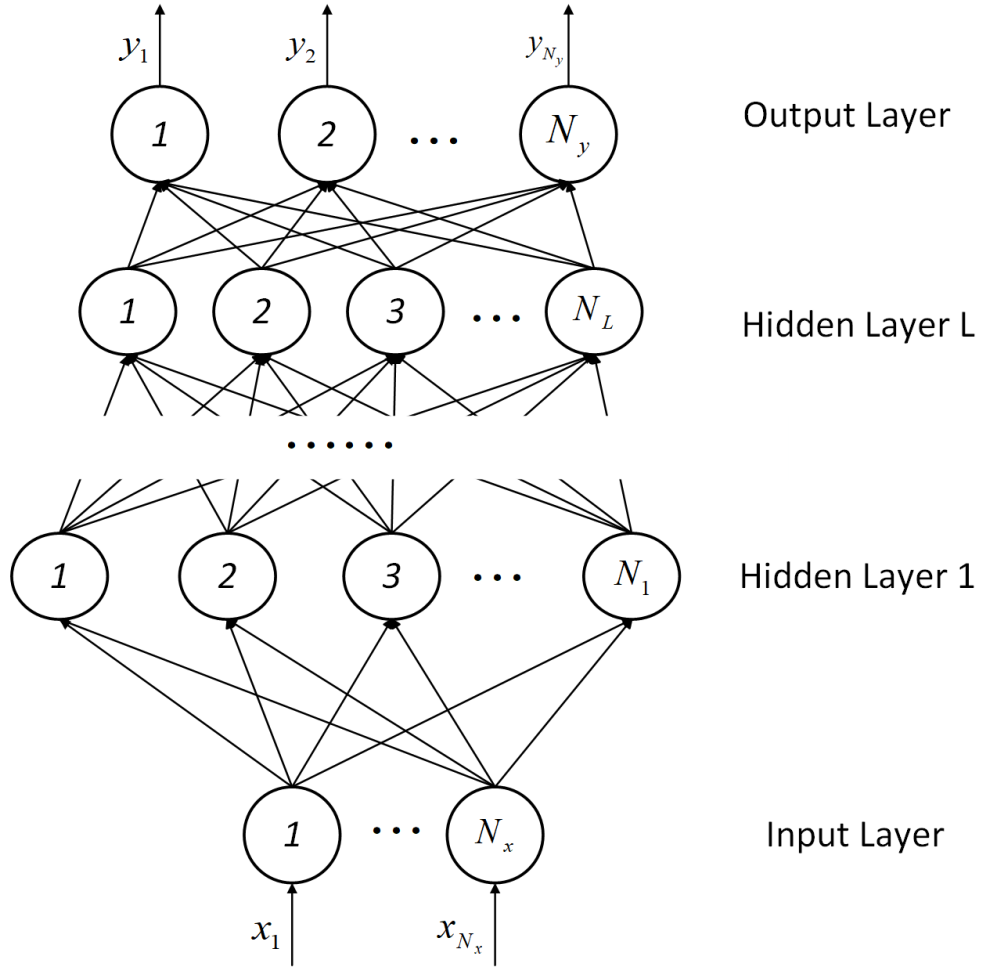


Figure 2.14: General MLP structure

Multilayer Perceptrons (MLP), a basic type of neural network capable of approximating continuous and integrable functions, represent an important class of feedforward neural networks. Typically, it includes one input layer, one output layer and several hidden layers between them, as shown in Figure 2.14.

Assume a neural network contains L hidden layers. Let the input and output

be represented as x_i and y_i . The number of neurons in hidden layer l is N_l , where $l = 1, 2, \dots, L$. For hidden layer l , the output of the i th neuron $\bar{y}_i^\ell$, is given by

$$\bar{y}_i^\ell = \sigma \left(\sum_{j=1}^{N_{l-1}} w_{ij}^\ell \cdot \bar{y}_j^{l-1} + \theta_i^\ell \right), \quad (2.49)$$

where w_{ij}^ℓ is the weight of the link between the j th neuron of the $l - 1$ th hidden layer and the i th neuron of the l th hidden layer, θ_i^ℓ is the bias parameter of the i th neuron of the l th hidden layer and $i = 1, 2, \dots, N_l$ and $\bar{y}_i^0 = x_i$. The $\sigma(\cdot)$ is a monotone squashing function, commonly called the activation function. Then let v_{ki} represent the weight of the link between i th neuron of the L th hidden layer and the k th neuron of the output layer, and let β_k represent the bias parameter of the k th neuron of the output layer. Then, the output of the MLP is given by

$$y_k = \sum_{i=1}^{N_L} v_{ki} \cdot \bar{y}_i^L + \beta_k, \quad (2.50)$$

where $k = 1, 2, \dots, N_y$.

Knowledge-Based Neural Network (KBNN)

Since MLP models act as black-box models, all information they contain is derived purely from training data and does not incorporate any task-specific knowledge. Consequently, large datasets are often required to improve model accuracy. However, obtaining sufficiently large datasets from measurements or simulations is expensive and time-consuming, as variations in materials, process parameters, or other factors can affect performance. However, less training data may also cause lower accuracy of the model. To address this problem, the Knowledge-Based Neural Network (KBNN) was introduced in [48], whose structure is shown below.

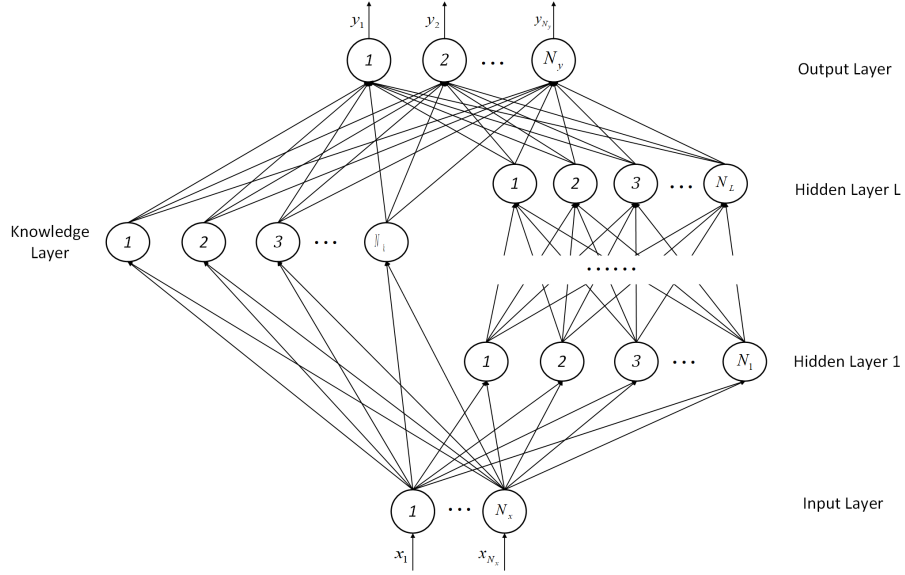


Figure 2.15: KBNN structure

The KBNN embeds task-related knowledge into the neural network while retaining the core advantages of conventional neural networks. The exact location of each switching boundary, as well as the scale and position of each knowledge function, are randomly initialised and subsequently determined during training [46]. The expression of the KBNN can be written with the result of the equation (2.51),

$$y_k = \sum_{j=1}^{N_z} w_{ji} x_i \left(\sum_{m=1}^{N_L} v_{km} \cdot \bar{y}_m^L \right) + \beta_k, \quad (2.51)$$

where $i = 1, 2, \dots, N_x$ and $k = 1, 2, \dots, N_y$. Here, v_{km} is the weight of the link between m th neuron of the L th hidden layer and the k th neuron of the output layer, w_{ji} is the weight of the link between i th input and the j th neuron of the knowledge layer.

According to the validation shown in [48], the KBNN structure consistently provides a lower model test error than standard MLP models across different examples. Its performance is superior not only within the measurement region (interpolation), but also outside the model extraction region (extrapolation).

2.3.2 Activation Function

Activation functions play a critical role in the ANN model. For example, in a neural network as shown in Figure 2.14, the input of the i th neuron of l hidden layer is the weighted output from $l - 1$ hidden layer. Assuming no activation function $\sigma()$ is used, the calculation in the neural network becomes equivalent to a linear regression model, and the equation (2.51) simplifies into

$$\bar{y}_i^\ell = \sum_{j=1}^{N_{l-1}} w_{ij}^\ell \cdot \bar{y}_j^{l-1} + \theta_i^\ell. \quad (2.52)$$

It is evident that the number of hidden layers is then irrelevant, since the combination of the linear functions remains linear. Consequently, the model cannot simulate or capture non-linear behaviour. Activation functions are therefore introduced to address this issue [49].

In feed-forward networks, activation functions are generally positioned between network layers (represented by f in Figure 2.16), where they govern the non-linear transformation of input signals as they propagate through the model.

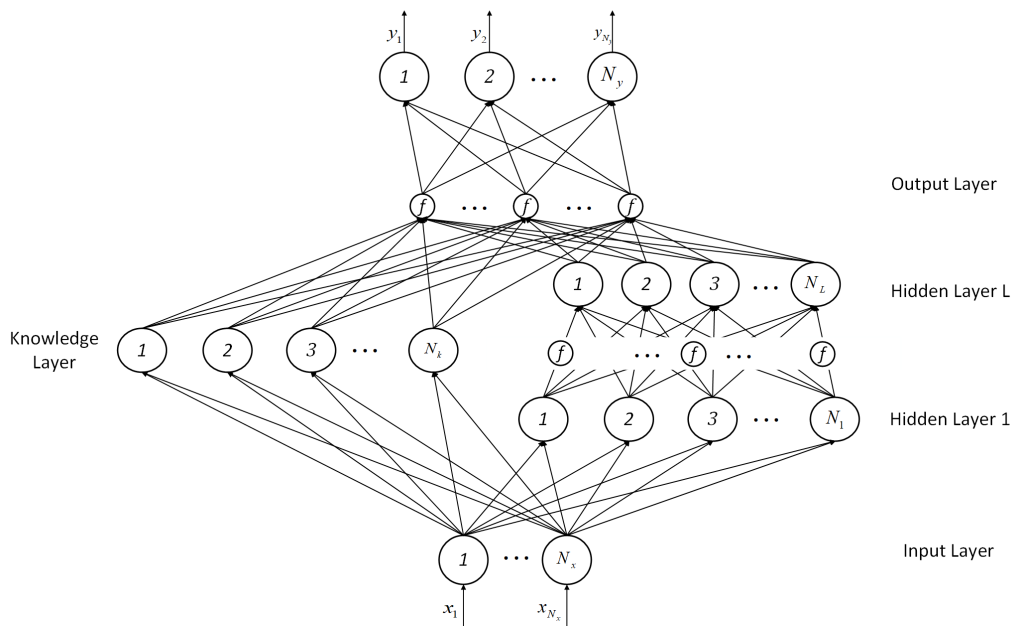


Figure 2.16: KBNN structural diagram with activation function setup

The choice of activation function in RF and microwave-related neural network models often differs from those used in image or Natural Language Processing (NLP) applications, as the tasks, such as device modelling, behavioural modelling, and load-pull prediction require smoothness, extrapolation stability, or the ability to process complex-valued data. The following subsections briefly introduce several commonly used activation functions.

Sigmoid Function

Sigmoid activation function, also known as logistic sigmoid, was one of the earliest activation functions used in neural networks. It is expressed as,

$$g(x) = \text{sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (2.53)$$

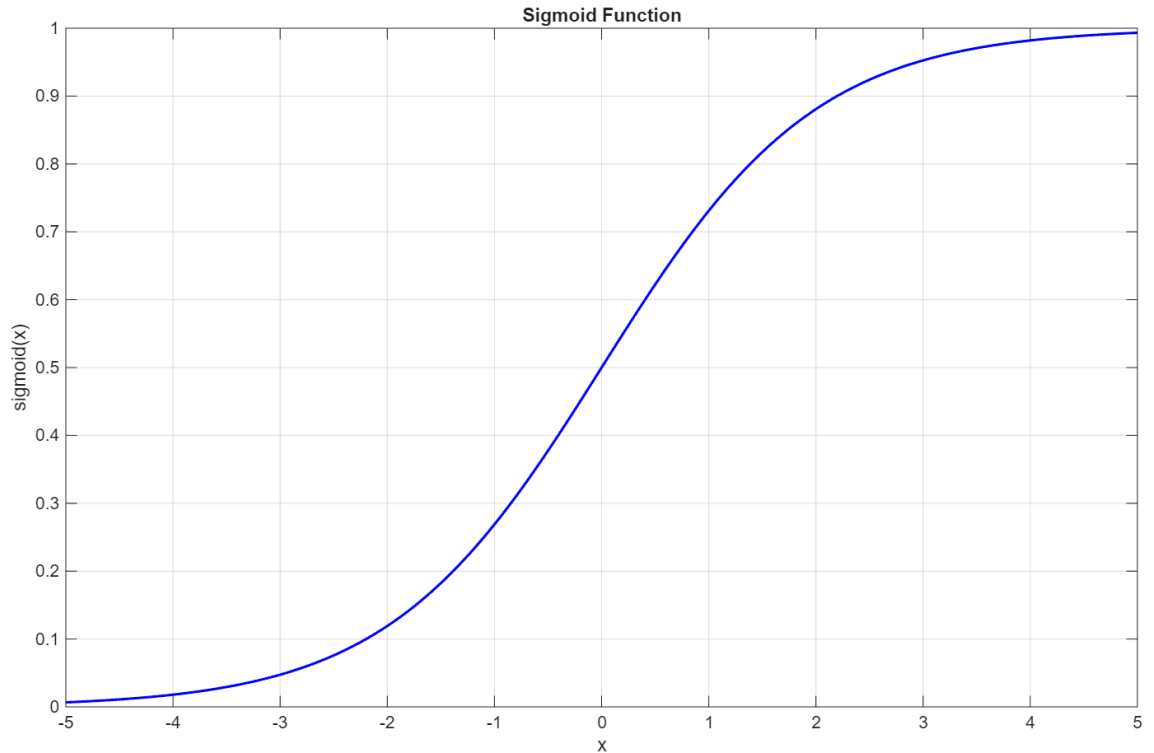


Figure 2.17: The Sigmoid activation plot with $x = [-5 \ 5]$ range

As shown in Figure 2.17, this function maps any real-valued input into the range $(0, 1)$. When the output of the function is close to one, the neuron is activated and

the information is transmitted to the next layer; when close to zero, the neuron remains inactive.

tanh() Function

The tangent hyperbolic activation function, also denoted as $\tanh(\cdot)$, is a shifted version of the sigmoid function. It maps the real-valued input into the range $(-1, 1)$ and is expressed mathematically with:

$$g(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1}. \quad (2.54)$$

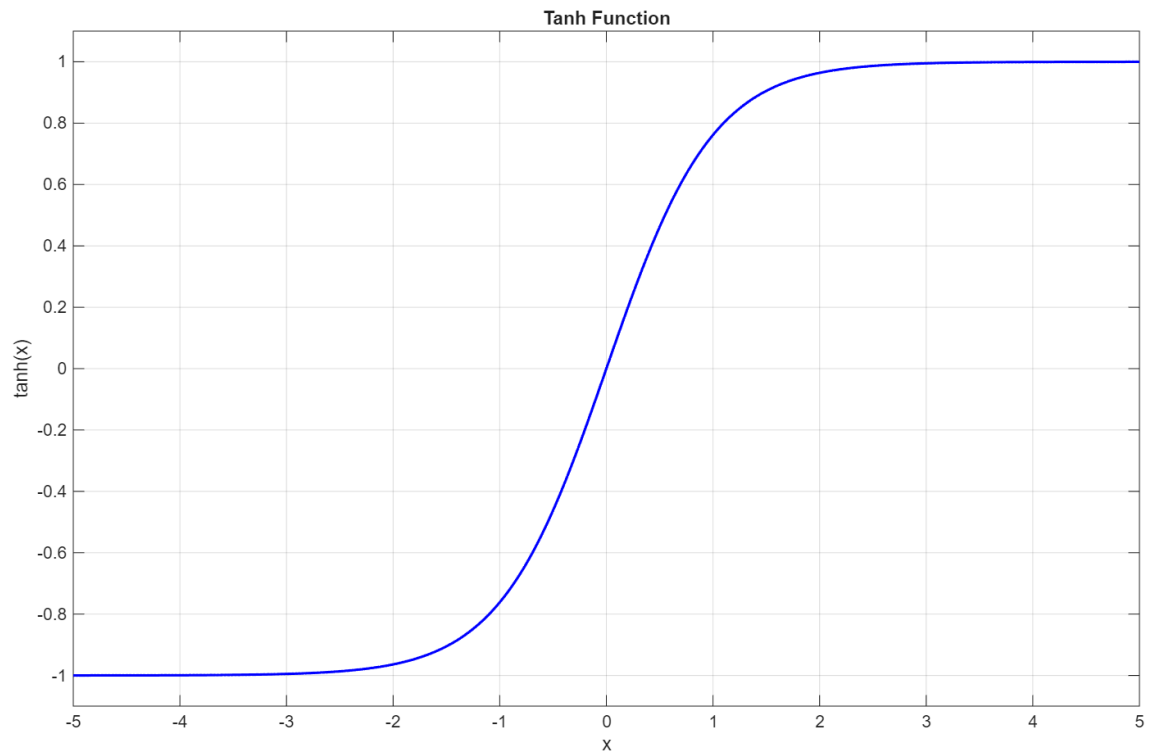


Figure 2.18: $\tanh()$ activation plot with $x = [-5 \ 5]$ range

Its zero-centred characteristic ensures faster convergence.

Symmetric Elliot Function

The Symmetric Elliot activation function provides a high-speed alternative to the sigmoid function, mapping the input real values into the range $(-1, 1)$, similar to the $\tanh$ function. Unlike $\tanh$, it is smoother and grows polynomially rather than exponentially, which helps mitigate the vanishing gradient problem. It can be expressed as:

$$g(x) = SElliot(x) = \frac{x}{1 + |x|}. \quad (2.55)$$

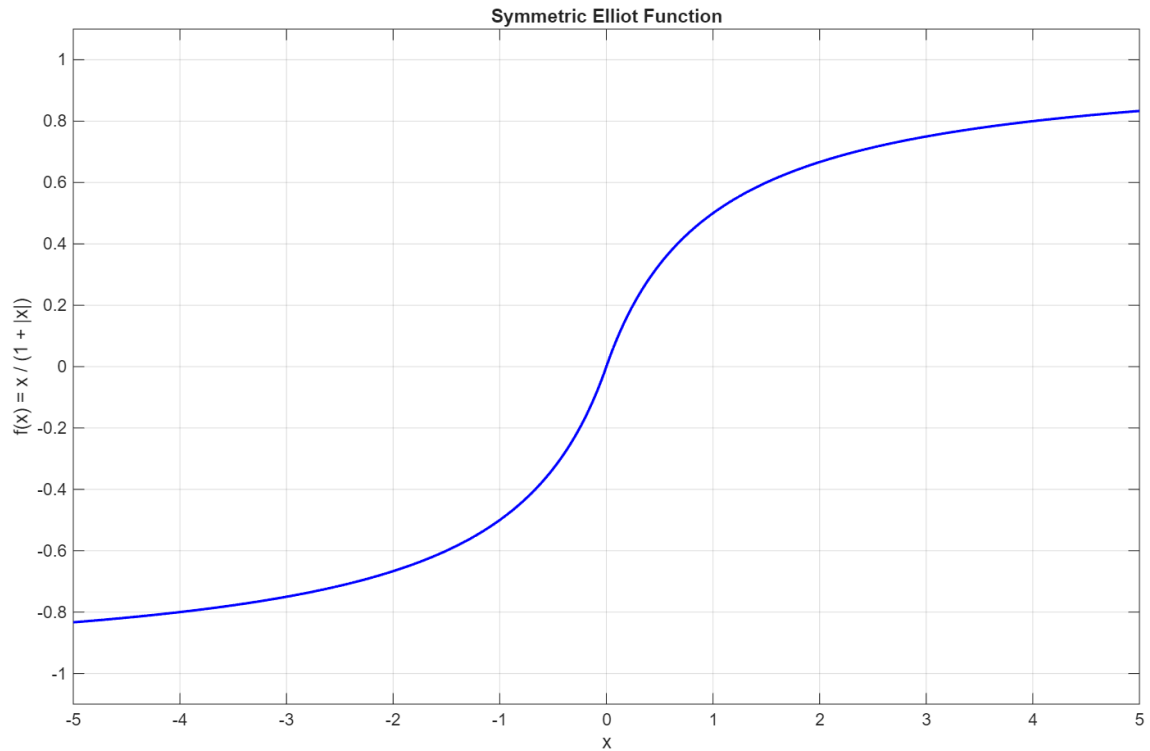


Figure 2.19: The Symmetric Elliot activation plot with $x = [-5 \ 5]$ range

However, compared with the $\tanh$ function, it has slightly higher computational complexity due to the need for derivative calculations.

ReLU Function

The Rectified Linear Unit (ReLU) function is a simple and widely used activation function that outputs the input value if it is positive, and zero otherwise. It is expressed as:

$$g(x) = \text{ReLU}(x) = \max(x, 0) = \begin{cases} 0 & \text{if } x \leq 0, \\ x & \text{if } x > 0. \end{cases} \quad (2.56)$$

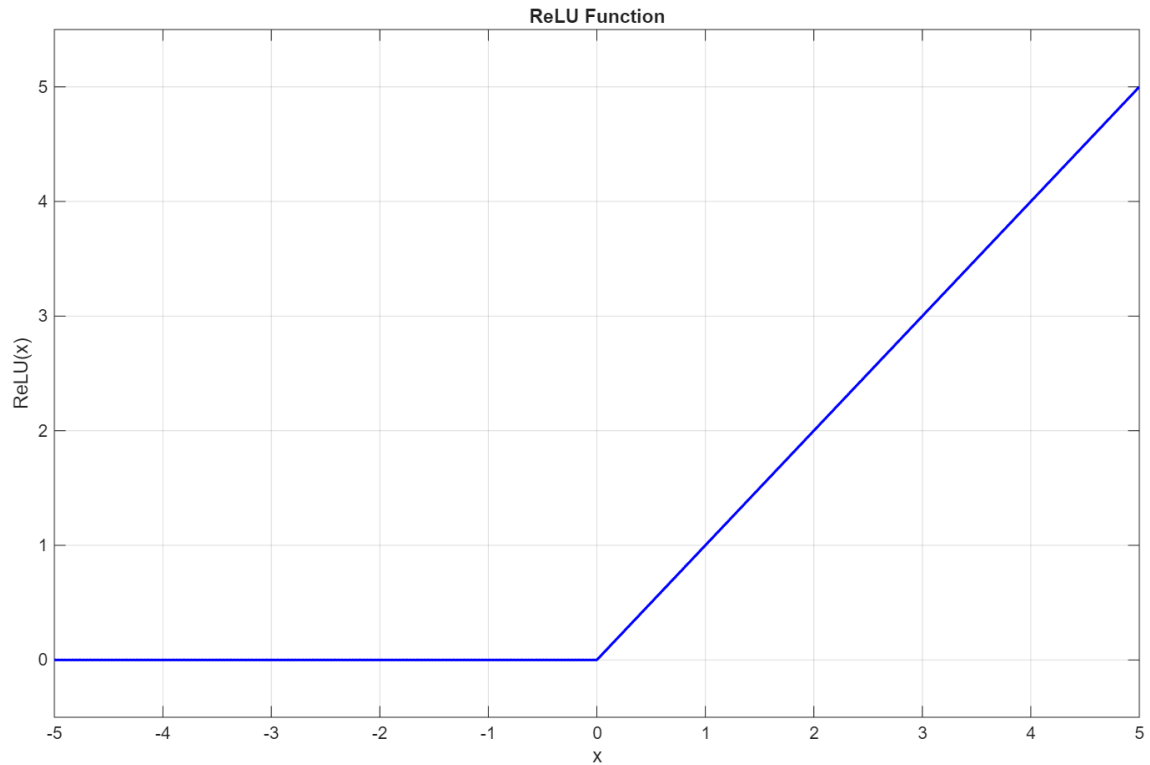


Figure 2.20: The ReLU activation plot with $x = [-5 \ 5]$ range

The ReLU function avoids the vanishing gradient problem for positive inputs, helping to train more stable neural networks. However, it also introduces a non-negligible problem: when the input is negative, the neuron output is always zero during backpropagation, preventing weight updates and rendering the neuron inactive (“dead”).

ELU Function

Exponential Linear Unit (ELU) is an advanced ReLU function. For positive inputs, its output is identical to ReLU, but for negative inputs, the ELU introduces an exponential nonlinearity. This modification resolves the “dead neuron” problem and improves training stability. However, this improvement slightly reduces computational efficiency. Additionally, the new hyperparameter α is used to control the slope of the negative region and is important for tuning the model. The mathematical form is shown below:

$$g(x) = \text{ELU}(x) = \begin{cases} x & \text{if } x > 0, \\ \alpha(e^x - 1) & \text{if } x \leq 0 \end{cases} \quad (2.57)$$

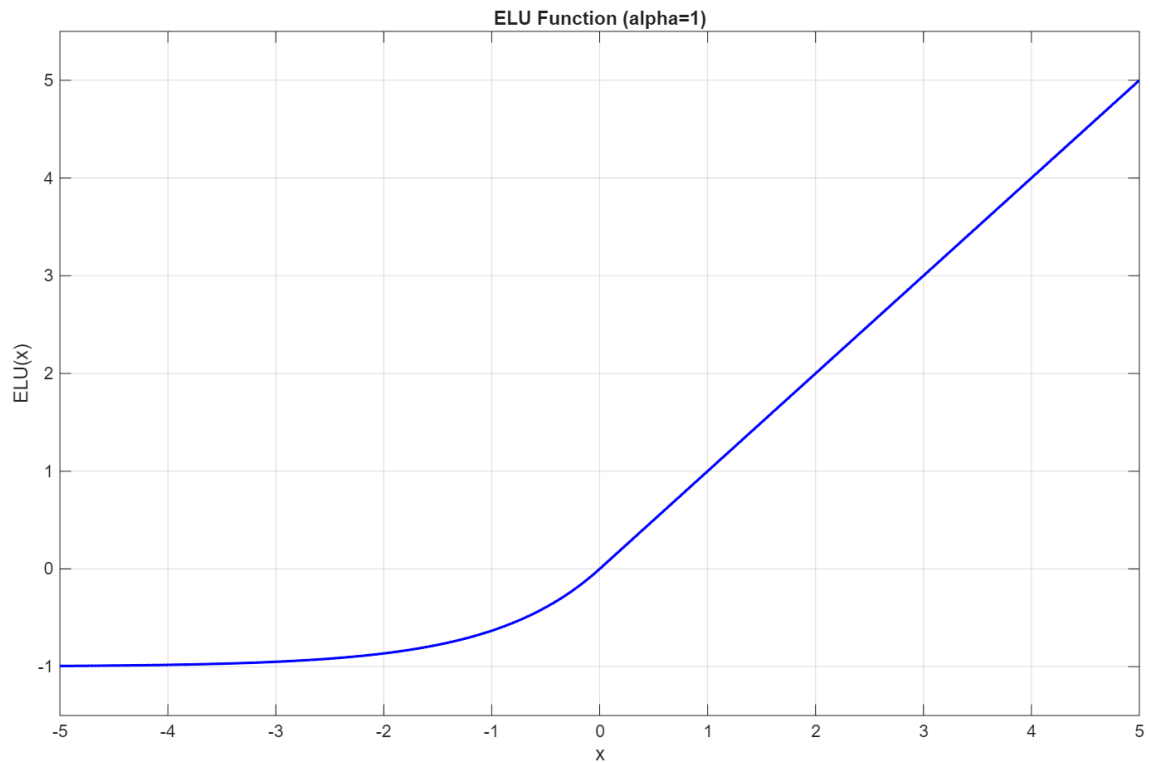


Figure 2.21: The ELU activation plot with $x = [-5 \ 5]$ range

In regression tasks, a linear activation function is typically preferred at the output

layer, as it aligns naturally with the Mean Squared Error (MSE) loss function and has been shown to simplify network training while improving convergence [47].

2.3.3 Adding Bias

In ANN, the biases play an important role in the model's expressive ability. With the bias term, the model can be more flexible and improve the learning efficiency because the centre of the activation is not always around the origin of the coordinate system [50]. As proven in [51], a neural network can approximate any continuous function only when neurons operate on affine functional of the form $x + b$, rather than purely linear ones. The figure below illustrates the KBNN with biases added.

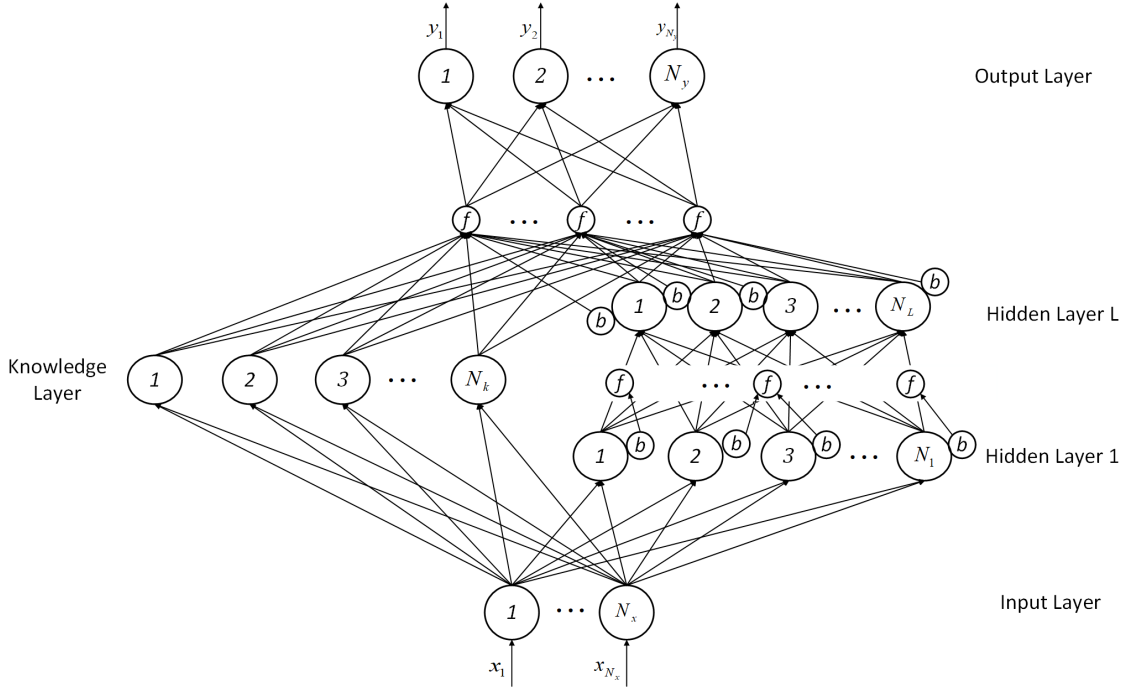


Figure 2.22: Structural diagram with bias added to the setup

Figure 2.23 illustrates the ANN structure associated with a single output neuron. The outputs from the neurons in the previous layer are multiplied by their corresponding weights and summed together with the bias b . The result is then passed through the activation function $f(\cdot)$ to produce the neuron output.

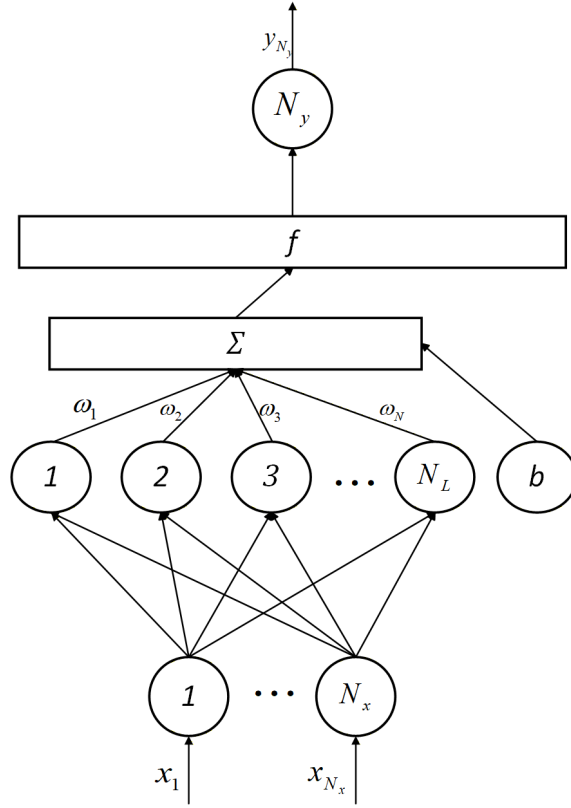


Figure 2.23: ANN structure for a single output neuron with bias and activation function

2.3.4 Model Training

After selecting the structure and activation function, and including the bias terms, the ANN model is initialised with random weights and biases. However, these initial values may be far from their optimal values. Therefore, the model must be trained to obtain more suitable weights and biases. Generally, there are three training categories: supervised training, unsupervised training, and reinforcement training.

Backpropagation (BP) is a widely used supervised training algorithm for multi-layer neural networks, and is known for its ability to iteratively improve learning and achieve higher accuracy [52]. It works by feeding input data through the network to compute outputs, then comparing these outputs with target values to compute the error. Then, the error is propagated backward through the network to update weights and biases according to the chosen optimization rule.

General gradient descent error backpropagation applies partial differentiation to

the error calculation for weight and bias updates:

$$W_{i+1} = W_i - \eta \frac{\partial E}{\partial W_i} \quad (2.58)$$

$$b_{i+1} = b_i - \eta \frac{\partial E}{\partial b_i}, \quad (2.59)$$

where, i is the index, W and b denote weights and biases and the η is a variable named learning rate which can be adjusted according to the chosen algorithm. When talking about the gradient descent error back-propagation theories, there are the Quasi-Newton [53], the Levenberg-Marquardt (LM) [54] and the Bayesian Regularization [55], [56] training algorithms, which are popular for RF related model extraction. Among these algorithms, the LM algorithm performs the fastest convergence [47].

The LM algorithm forms the foundation of the Bayesian Regularization method and is based on a Quasi-Newton approximation using the Hessian matrix. By adjusting the learning rate μ , the LM algorithm behaves like gradient descent when μ is large and like Newton's method when $\mu \rightarrow 0$ [57]. Its theoretical basis is similar to that of the Least Mean Squares (LMS) algorithm, which also underpins the CM.

2.3.5 ANN Model Discussion

As a fast, accurate, and flexible tool, neural networks have recently been applied to RF/microwave modelling, simulation, and design. As this emerging technology expands from academic research to actual applications, there is a need to address the basic conceptual issues from model structure selection to CAD implementation [58].

Compared with the basic MLP model, the KBNN model performs better compatibility in RF and microwave designs, as the neurons in the knowledge layer help the model adapt to various parameter combinations. When transferring information between different layers of the neural network, the activation function and the biases are also important. They help the model to fit and capture the non-linear charac-

teristics and improve the model to be more efficient and robust. Then, training the model properly is also important. Using the LM algorithm for network training is an effective choice for achieving fast and accurate convergence. MATLAB also provides a user-friendly tool to train and test the ANN model. However, in practical RF and microwave design experiments, extracting an ANN model defined by specific parameters often requires custom MATLAB scripts to manually adjust the model-building and training process.

The ANN model developed in this work is implemented in ADS using an FDD-based interface together with parameter-import and ANN calculation blocks. The detailed ADS implementation is presented in Chapter 3, with the full schematic blocks provided in Appendix A.

2.4 Error and Calibration

2.4.1 Network Analyser

The load-pull measurement method is commonly used in the PA design to characterise the non-linear behaviour of active devices. The principle of load-pull is to measure the device performance under non-50 Ω load conditions under large-signal excitation. The VNA is one of the most precise electronic instruments used in RF and microwave measurements. It can measure signals of high and low power signals with better accuracy compared to other power sensors and the gain response across frequency of an electronic device [59]. In [31], a simplified diagram of the VNA shown in Figure 2.24 is provided.

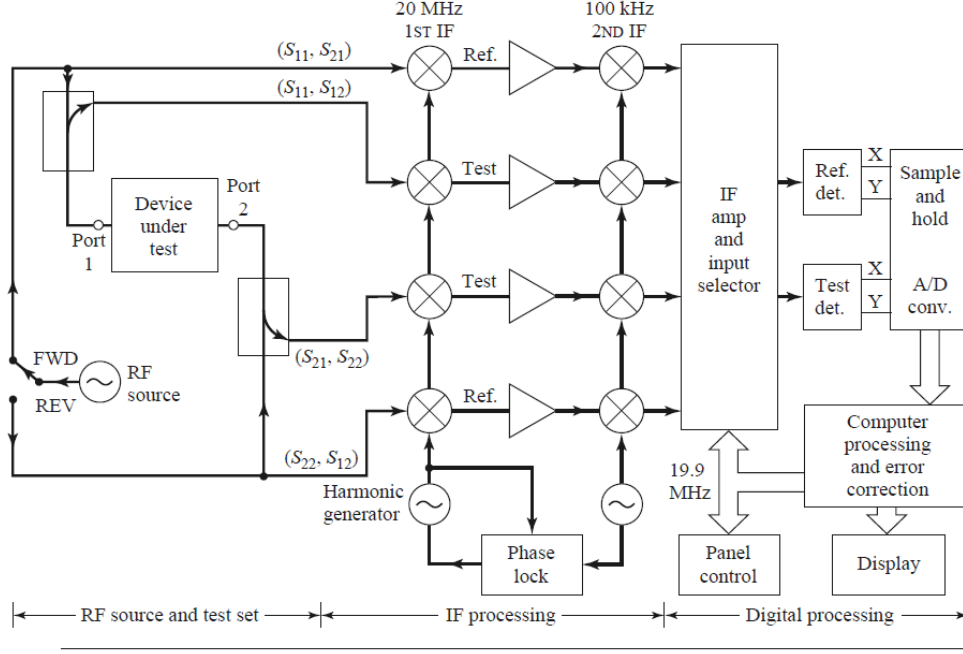


Figure 2.24: VNA diagram [31]

The VNA uses an RF source to generate electromagnetic waves that interact with the DUT, while the scattered waves are measured. Each physical port operates as part of a four-port network equipped with directional couplers or bridges to separate the incident and reflected signals. Among all the electronic measurement systems, the measurement accuracy and performance of the VNA rely most heavily on error correction.

However, when discussing “calibration” in the context of a VNA, a distinction that often causes confusion is that it differs from the usual meaning of calibration used for other instruments. In this case, the term error correction generally refers to vector correction methods.

Some modern VNAs come with the error correction process already performed at the factory, providing a built-in “factory calibration”. This ensures that the amplitude and phase of the signal are known at the VNA ports. However, adding extra cable length introduces both additional phase shift and attenuation, reducing measurement accuracy, since the original calibration reference planes are at the VNA ports. Therefore, a new calibration (or error correction) is required to shift the reference planes to the ends of the coaxial cables where the DUT will be measured. If

these cables are further connected to other components (such as waveguides) through which the signal travels before reaching the DUT, the reference planes must again be shifted to account for the additional phase change and attenuation. Moreover, it is not only the phase that may be affected. The impedance mismatches at cable connections or cable to waveguide transitions can cause part of the signal to be reflected, and these mismatch errors must also be corrected.

2.4.2 Source of Error

It is important for designers to obtain accurate S-parameters of the DUT, as this enables the development of precise system models, reducing design effort and rework. Only an ideal instrument would not require correction. In practice, all measurement instruments have imperfections, leading to less than ideal results that affect PA design. Consequently, all measurements are affected by the uncertainty, which in VNA measurements can generally be classified as systematic, random, or drift errors [60].

Random errors, as shown in the name, vary unpredictably over time and are mainly caused by instrument noise. Drift errors occur when the performance of the system changes with time after calibration, typically due to temperature variations. To minimise such errors, the setup should be placed in an environment with stable or controlled temperature. These error types cause measurement uncertainties that are difficult to remove because of their unpredictable nature.

Systematic errors, on the other hand, produce measurement uncertainties resulting from imperfections in the VNA and the test setup. Since these errors are stable over time, they can be removed using calibration and correction methods. There are six main types of systematic errors: source and load impedance mismatches (causing reflections), crosstalk and directivity errors (caused by signal leakage), and frequency response errors (resulting from reflection and transmission tracking within the test receivers) [61]. In general, signal leakage, signal reflections, and frequency response issues are considered to be the main sources of systematic errors [62].

Achieving complete correction is challenging because random fluctuations are superimposed on the measured data. To fully account for systematic measurement errors, both the raw measurement data and the corresponding error terms must be known. Otherwise, accurate correction is not possible. Therefore, the VNA calibration is essential before measurement to determine these error terms and ensure that the designer obtains accurate S-parameters [62].

2.4.3 Error Model

Calibration is a step-by-step procedure used to determine the error terms (or error model) of a VNA, which represent its systematic errors under specific conditions. To derive these error terms, the relationship between the incident and reflected waves in the measurement system must be calculated through a series of calibration measurements [62], [63]. The VNA error correction process contains two main stages. The first stage, commonly called calibration or VNA-cal, involves measuring known standards, such as open, short, and load, to identify the systematic error terms of the VNA. This stage can be formally described as error correction acquisition. The second stage includes measuring the DUT and applying the previously determined error correction algorithms to produce accurate, corrected results. This step is formally known as error correction application [59].

12-Term Error Model

The 12-term error model is derived using the signal flow graph method originally improved by Mason in 1956 [64], and was later formalised for vector network analyser calibration by Speciale in 1977 [65]. Figure 2.25 displays a signal flow graph describing a common error model for a one-port VNA [31], [66].

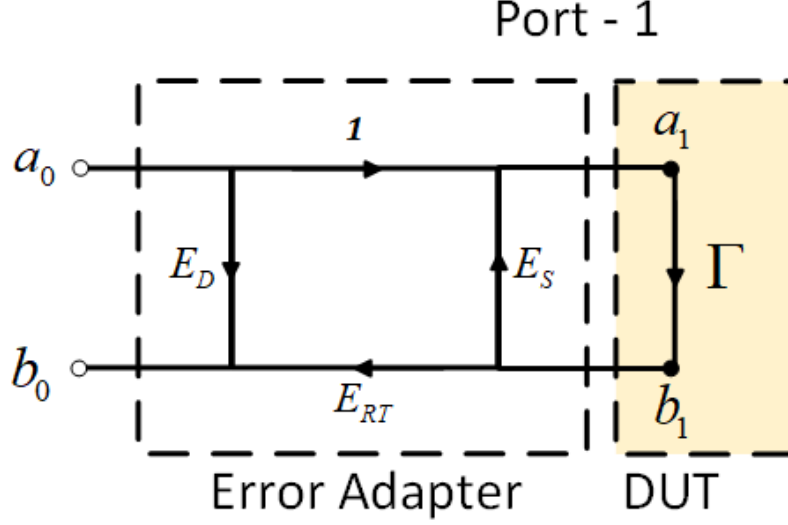


Figure 2.25: Signal flow graph for one-port error model

The signal flow graph of a one-port VNA error adapter includes three primary types of errors. Directivity error (E_D or e_{00}) refers to errors in the measured reflected signal caused by signal leakage and the limited directivity of the coupler. Source match error (E_S or e_{11}) can be defined as impedance mismatches at the source, which affects the measurement at the reference plane and the reflection tracking. Reflection tracking error (E_{RT} or $e_{10}e_{01}$) represents frequency-dependent imperfections between the reference and test channels [31].

For two-port cases, the 12-term error model describes the ability of a measurement system to characterise a DUT with two ports. This model is developed from the one-port error model. The error adapter of each port can be represented by an S-parameter network containing the terms S_{11} , S_{12} , S_{21} , S_{22} . The 12-term model mathematically defines the relationship between the corrected data and the raw measured data (taken at the reference planes), making it necessary for the calibration process and directly influencing the accuracy of the calibration results.

The 12-term model is one of the basic models employed in VNAs. It was designed for older VNA architectures, which rely on use of only three receivers [59]. Under these circumstances, there is a clear differentiation between the forward and the reverse measurement directions. Therefore, it actually consists of two separate 6-

term models and error sets for each direction. Figure 2.26 conveniently represents these two measurement phases.

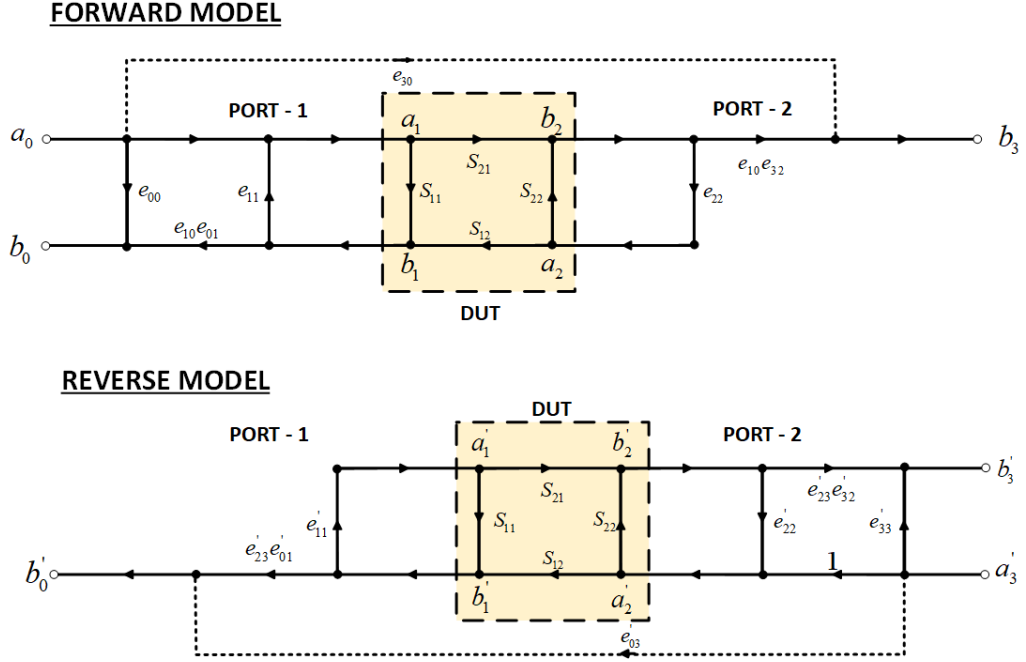


Figure 2.26: Signal flow graph for 12-term error model

As shown in Figure 2.26, there are twelve different unknown terms to describe the systematic errors in the VNA, which need to be quantified to correct the measurement system. The term e_{30} and e'_{30} are related to cross talk errors, and can be directly measured with S_{21} and S_{12} obtained when terminating port 1 and port 2. This helps to reduce the number of the unknown errors into ten. For other terms, in the forward direction, the procedure typically starts from measuring three independent one-port DUT with known coefficients (calibration standard) to evaluate e_{00} , e_{11} and $e_{10}e_{01}$. Then connecting the port 1 and port 2 together obtains the e_{22} and $e_{10}e_{32}$. For the reverse mode, the rest unknown terms can be obtained from the same procedure [66]. Also, when connecting two ports, if the S-parameter is known, the transmission line can replace the interconnection between port 1 and port 2.

8-Term Error Model

When separate receivers are used for all the scattered waves, then the forward and reverse models can be combined together. In this case, the 12-term error model

can be substituted by an equivalent error model with a lower number of error terms and no loss in accuracy [67]. First, let's assume a scenario where cross-talk leakage is zero, the switch is ideal, and load matching remains unaffected when the VNA toggles between forward and reverse operating modes. It should be noted that this assumption holds only on the condition that four separate receivers are used to capture scattered waves [67]. In this case, it is possible to use a different error model including only 8 error terms, as shown in Figure 2.27.

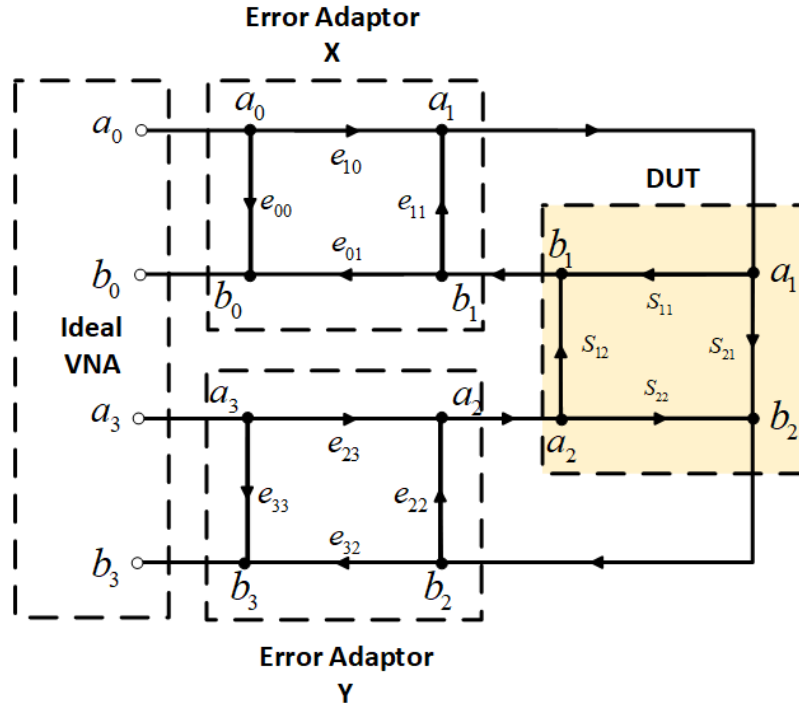


Figure 2.27: Signal flow graph for 8-term error model

The 8-term error model basically derives from two independent error adapters, one between port 1 and the input of the DUT (error adapter X in Figure 2.27) and one between port 2 and the output of the device (error adapter Y in Figure 2.27). Since the error boxes and the DUT are in a cascade configuration, it is convenient to represent the relationship mathematically using a cascade matrix. Let T_M represent

the cascade matrix of the measurement, then [67], [68]

$$T_M = X \cdot T_{DUT} \cdot \bar{Y} \quad (2.60)$$

where $\bar{Y}$ represents the transpose of the matrix Y and the X and Y are the cascade matrices of the error adaptors X and Y . Once these have been confirmed, the calibration can be easily applied by inverting the equation:

$$T_{DUT} = X^{-1} \cdot T_M \cdot \bar{Y}^{-1} \quad (2.61)$$

It is notable that both the 12-term and the 8-term error models describe exactly the same system and the same errors, which also means when VNA has four independent receivers, the two models can be interchanged. Actually, the 8-term model can be modified into 10-term model and the 12-term model can be simplified into a 10-term model by removing the crosstalk terms [66].

16-Term Error Model

The 16-term error model, used in earlier generations of VNAs (not currently used), represents the measurement system through an error adapter placed between the DUT and the actual measurement reference. This model accounted for all possible linear and stationary errors that could appear within the measurement system, as illustrated in Figure 2.28.

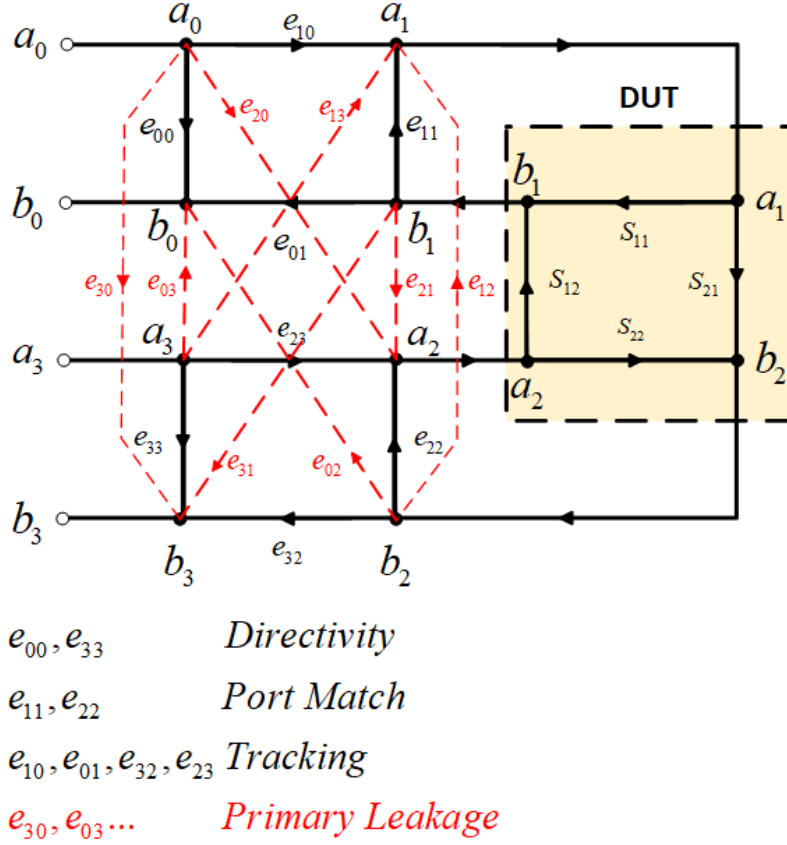


Figure 2.28: Signal flow graph for 16-term error model

This model consists of 16 terms, of which 8 are critical. These 8 cover the tracking errors, port match and directivity, and the second 8 represent the leakage terms. The 8-term model can be obtained from the 16-term model by assuming that the leakage terms are all zero.

2.4.4 Calibration Method

The vector error correction methods focus on a process of identifying systematic error terms which is achieved by measuring known standards to remove impacts from further measurements. Therefore, the calibration process requires several standards to determine error coefficients. The choice of algorithm used in calibration is dependent on the selection of the calibration standards. Several commonly employed calibration algorithms are available, such as

- **Short-Open-Load-Thru** [31]

-
- **Thru-Reflect-Line** or **Line-Reflect-Line** [69]
 - **Line-Reflect-Match** [70]
 - **Reciprocal-Short-Open-Load** [71].

Actually, there are many other VNA calibration methods, each of which have unique mathematical concepts and different implementation processes. This chapter will introduce and discuss in detail one of those methods: TRL.

TRL

TRL calibration was proposed by Engen and Hoer in 1979 [69]. The TRL calibration is used to establish a full two-port calibration using three standards: a Thru, a Reflect, and a defined Line. The thru standard is equivalent to a zero-length microstrip, requiring the fabrication of a physical device with near-zero physical dimensions at the frequencies required. The reflect standard typically takes the form of either a short or open circuit, and the phase characteristic is generally undefined in basic calibration frameworks. While the line standard has a structure similar to the thru standard, it features a non-zero length. In TRL calibration, the characteristic impedance of the line standard determines the reference impedance of the calibrated system. Therefore, the line standard is normally designed to have a characteristic impedance close to the desired system impedance, so that the calibrated reference impedance remains close to the intended system impedance and additional impedance renormalisation can be minimised. The flow graphs of these standards are shown in Figure 2.29.

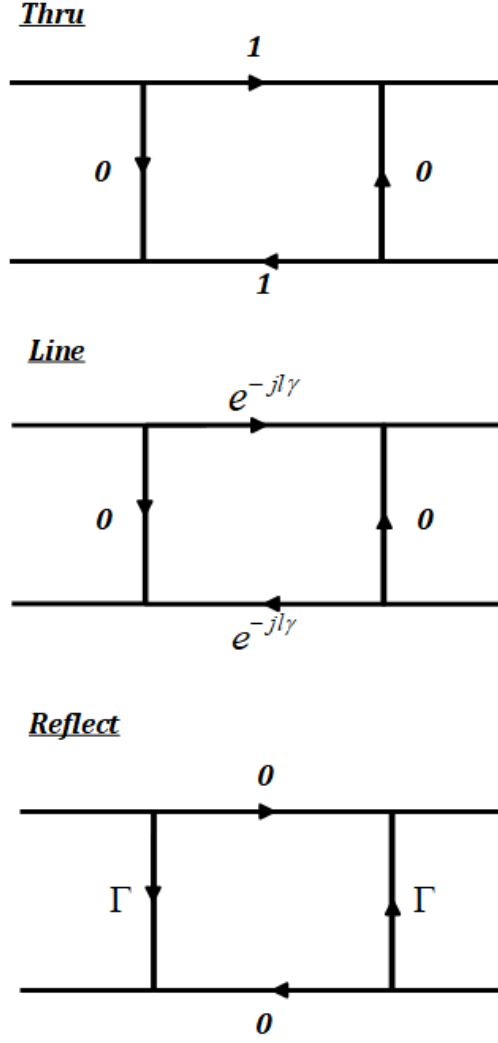


Figure 2.29: Signal flow graph for calibration standards of TRL

Before calculating the error parameters, the relationship between S-parameters and the T-matrix must first be established, since the T-matrix is more suitable when cascading a series of two-port networks.

$$[T] = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} = \begin{bmatrix} \frac{-\Delta s}{s_{21}} & \frac{s_{11}}{s_{21}} \\ \frac{s_{22}}{s_{21}} & \frac{1}{s_{21}} \end{bmatrix} \quad (2.62)$$

where $\Delta s = s_{11}s_{22} - s_{12}s_{21}$. With the T-matrix and Figure 2.27, let the T_M represent the measurement matrix and the T_D represent the performance of the DUT. The whole network can be represented as:

$$\begin{bmatrix} b_0 \\ a_0 \end{bmatrix} = [T_M] \begin{bmatrix} b_3 \\ a_3 \end{bmatrix} \quad (2.63)$$

As stated in equation (2.60), the T_M is

$$[T_M] = [T_A][T_D][T_B] \quad (2.64)$$

where T_A and T_B represent the two error adapters. Therefore, to define the relationship between raw a and b data to calibrated data the following can be stated:

$$\begin{aligned} \begin{bmatrix} b_0 \\ a_0 \end{bmatrix} &= [T_A] \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} \\ &= \begin{bmatrix} T_{11}^A & T_{12}^A \\ T_{21}^A & T_{22}^A \end{bmatrix} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} \\ &= T_{22}^A \begin{bmatrix} T_{11}^A/T_{22}^A & T_{12}^A/T_{22}^A \\ T_{21}^A/T_{22}^A & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} \\ &= T_{22}^A \begin{bmatrix} a & b \\ c & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} \end{aligned} \quad (2.65)$$

and

$$\begin{bmatrix} b_1 \\ a_1 \end{bmatrix} = [T_D] \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \quad (2.66)$$

Similarly,

$$\begin{aligned}
\begin{bmatrix} a_2 \\ b_2 \end{bmatrix} &= [T_B] \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \\
&= \begin{bmatrix} T_{11}^B & T_{12}^B \\ T_{21}^B & T_{22}^B \end{bmatrix} \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \\
&= T_{22}^B \begin{bmatrix} T_{11}^B/T_{22}^B & T_{12}^B/T_{22}^B \\ T_{21}^B/T_{22}^B & 1 \end{bmatrix} \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \\
&= T_{22}^B \begin{bmatrix} \alpha & \beta \\ \gamma & 1 \end{bmatrix} \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \tag{2.67}
\end{aligned}$$

Now, with reference to the appropriate signal flow diagrams, the calibration standards' T-matrices can be defined:

$$[T_D^{thru}] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.68}$$

$$[T_M^{thru}] = [T_A] [T_D^{thru}] [T_B] = [T_A] [T_B] \tag{2.69}$$

$$[T_D^{line}] = \begin{bmatrix} L & 0 \\ 0 & 1/L \end{bmatrix} \tag{2.70}$$

$$[T_M^{line}] = [T_A] [T_D^{line}] [T_B] \tag{2.71}$$

To simplify the line-standard analysis by cancelling the common matrix terms, the

auxiliary matrix $[X]$ is defined as

$$[X] = [T_M^{line}] [T_M^{thru}]^{-1} \quad (2.72)$$

Then

$$\begin{aligned} [X] &= [T_A] [T_D^{line}] [T_B] [[T_A] [T_B]]^{-1} \\ &= [T_A] [T_D^{line}] [T_B] [T_B]^{-1} [T_A]^{-1} \\ &= [T_A] [T_D^{line}] [T_A]^{-1} \end{aligned} \quad (2.73)$$

In the calculation of the error terms it is simplest to consider the line (L) measurements first.

$$\begin{aligned} [X][T_A] &= [T_A][T_D^{line}][T_A]^{-1}[T_A] \\ &= [T_A][T_D^{line}] \end{aligned} \quad (2.74)$$

Then solving for the terms of $[T_A]$ will provide the following set of four equations

$$X_{11}T_{11}^A + X_{12}T_{21}^A = T_{11}^A L \quad (2.75)$$

$$X_{21}T_{11}^A + X_{22}T_{21}^A = T_{21}^A L \quad (2.76)$$

$$X_{11}T_{12}^A + X_{12}T_{22}^A = T_{12}^A / L \quad (2.77)$$

$$X_{21}T_{12}^A + X_{22}T_{22}^A = T_{22}^A / L \quad (2.78)$$

By solving these equations, two roots are obtained. The remaining task is to determine which root corresponds to $\frac{T_{12}^A}{T_{22}^A}$ and which corresponds to $\frac{T_{21}^A}{T_{11}^A}$.

If $Root_1 \ll Root_2$, then

$$\frac{T_{12}^A}{T_{22}^A} = b = Root_1 \text{ and } \frac{T_{21}^A}{T_{11}^A} = c/a = 1/Root_2 \quad (2.79)$$

else, which means $Root_1 \gg Root_2$, then

$$\frac{T_{12}^A}{T_{22}^A} = b = Root_2 \text{ and } \frac{T_{21}^A}{T_{11}^A} = c/a = 1/Root_1 \quad (2.80)$$

After finishing the line calculations, it is the right opportunity to consider the thru measurements.

$$[T_B] = [T_A]^{-1} [T_M^{thru}] \quad (2.81)$$

Solving for the terms of $[T_B]$ will give

$$\frac{T_{12}^B}{T_{11}^B} = \frac{T_{12}^{thru} - \left(\frac{T_{12}^A}{T_{22}^A}\right) T_{22}^{thru}}{T_{11}^{thru} - \left(\frac{T_{12}^A}{T_{22}^A}\right) T_{21}^{thru}} = \beta/\alpha = \frac{e - b}{d - bf} = \left(-\frac{S_{11}^B}{\Delta S}\right) \quad (2.82)$$

and

$$\frac{T_{21}^B}{T_{22}^B} = \frac{\left(\frac{T_{21}^A}{T_{11}^A}\right) T_{11}^{thru} - T_{21}^{thru}}{\left(\frac{T_{21}^A}{T_{11}^A}\right) T_{12}^{thru} - T_{22}^{thru}} = \gamma = \frac{f - d(c/a)}{1 - e(c/a)} = (-S_{22}^B) \quad (2.83)$$

Finally, the reflect measurements can be considered to determine a/α .

For port 1:

$$a\Gamma_R = \frac{(\Gamma_1 - b)}{(1 - c/a \Gamma_1)} \quad (2.84)$$

where $\Gamma_1 = S_{11}^{short}$.

For port 2:

$$\alpha\Gamma_R = \frac{(\Gamma_2 + \gamma)}{(1 + \beta/\alpha\Gamma_2)} \quad (2.85)$$

where $\Gamma_2 = S_{22}^{short}$.

Therefore

$$a/\alpha = \left(\frac{\Gamma_1 - b}{1 - \frac{c}{a}\Gamma_1} \right) \left(\frac{1 + \frac{\beta}{\alpha}\Gamma_2}{\Gamma_2 + \gamma} \right) \quad (2.86)$$

Considering again the thru measurement to determine $a\alpha$.

$$[T_M^{thru}] = [T_A][T_B] \quad (2.87)$$

$$(2.88)$$

This gives

$$a\alpha = \frac{\frac{T_{11}^{thru}}{T_{22}^{thru}} - b\frac{T_{21}^{thru}}{T_{22}^{thru}}}{1 - \frac{T_{12}^{thru}}{T_{22}^{thru}}(c/a)} \quad (2.89)$$

Using (2.86) and (2.89) a^2 or α^2 can be determined, however there is still a sign issue remaining.

$$a = \pm\sqrt{(a\alpha)(a/\alpha)} \quad (2.90)$$

The sign of (2.90) is determined by the sign of Γ_R .

$$\Gamma_R = \frac{(\Gamma_1 - b)}{(1 - \frac{c}{a}\Gamma_1)} (1/a) \quad (2.91)$$

The user must supply Γ_R^{std} , which can be found in the Cal-kit data sheet.

$$\text{If } \text{real}(\Gamma_R/\Gamma_R^{std}) \ll 0, \text{ then } a = -a; \text{ otherwise, } a = a \quad (2.92)$$

So

$$\alpha = (a\alpha)/a \quad (2.93)$$

Finally, c can be determined using (2.79) or (2.80).

All terms apart from T_{22}^A are now known, however T_{22}^A is typically set to unity.

$$T_{11}^A = aT_{22}^A \quad (2.94)$$

$$T_{12}^A = bT_{22}^A \quad (2.95)$$

$$T_{21}^A = (c/a)aT_{22}^A \quad (2.96)$$

$$T_{22}^B = \frac{T_{22}^{thru}}{(1 + c\beta)T_{22}^A} \quad (2.97)$$

$$T_{11}^B = \alpha T_{22}^B \quad (2.98)$$

$$T_{12}^B = \beta T_{22}^B \quad (2.99)$$

$$T_{21}^B = \gamma T_{22}^B \quad (2.100)$$

In the end, the T-matrix is converted to the S-matrix for the final calculation of the traditional error terms.

2.4.5 Calibration Discussion

Accurate calibration is essential for VNA-based characterisation, especially for devices exhibiting high reflection coefficients. This section introduces VNA architec-

ture, measurement uncertainties, and systematic error models. Finally, the key equations of TRL calibration procedure are displayed. However, some software suppliers, for example, Mesuro, already support embedded calibration calculation functions and will be used in Chapter 5.

2.5 Conclusion

This chapter reviewed the non-linear behavioural modelling techniques including VIOMAP, PHD modelling and the X-parameters, followed by the development of the CM and the structure of KBNN, a kind of ANN model. The chapter concluded with VNA calibration theory and TRL-based error model extraction, providing a solid foundation for the modelling and error analysis presented in the following chapters.

References

- [1] H. Statz, P. Newman, I. Smith, R. Pucel, and H. Haus, “GaAs FET device and circuit simulation in SPICE,” *IEEE Transactions on Electron Devices*, vol. 34, no. 2, pp. 160–169, 1987. DOI: 10.1109/T-ED.1987.22902.
- [2] W. Curtice and M. Ettenberg, “A Nonlinear GaAs FET Model for Use in the Design of Output Circuits for Power Amplifiers,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 33, no. 12, pp. 1383–1394, 1985. DOI: 10.1109/TMTT.1985.1133229.
- [3] I. Angelov, L. Bengtsson, and M. Garcia, “Extensions of the Chalmers nonlinear HEMT and MESFET model,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 44, no. 10, pp. 1664–1674, 1996. DOI: 10.1109/22.538957.
- [4] I. Angelov, H. Zirath, and N. Rosman, “A new empirical nonlinear model for HEMT and MESFET devices,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 12, pp. 2258–2266, 1992. DOI: 10.1109/22.179888.
- [5] A. A. Kokolov and L. I. Babak, “Methodology of built and verification of non-linear EEHEMT model for GaN HEMT transistor,” *Radioelectronics and Communications Systems*, vol. 58, no. 10, pp. 435–443, Oct. 2015. DOI: 10.3103/S0735272715100015.
- [6] U. Radhakrishna, T. Imada, T. Palacios, and D. Antoniadis, “MIT virtual source GaNFET-high voltage (MVSG-HV) model: A physics-based compact model for HV-GaN HEMTs,” *Physica Status Solidi C*, vol. 11, no. 3-4, pp. 848–852, 2014. DOI: 10.1002/pssc.201300392.

-
- [7] F. Verbeyst and V. Bossche, "VIOMAP, the S-parameter equivalent for weakly nonlinear RF and microwave devices," *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, no. 12, pp. 2531–2535, 1994. DOI: 10.1109/22.339793.
- [8] Verspecht and D. Root, "Polyharmonic distortion modeling," *IEEE Microwave Magazine*, vol. 7, no. 3, pp. 44–57, 2006. DOI: 10.1109/MMW.2006.1638289.
- [9] D. E. Root, J. Verspecht, J. Horn, and M. Marcu, *X-Parameters: Characterization, Modeling, and Design of Nonlinear RF and Microwave Components* (The Cambridge RF and Microwave Engineering Series), English. Cambridge: Cambridge University Press, 2013.
- [10] P. J. Tasker and J. Benedikt, "Waveform Inspired Models and the Harmonic Balance Emulator," *IEEE Microwave Magazine*, vol. 12, no. 2, pp. 38–54, 2011. DOI: 10.1109/MMM.2010.940101.
- [11] S. Woodington et al., "A novel measurement based method enabling rapid extraction of a RF Waveform Look-Up table based behavioral model," in *2008 IEEE MTT-S International Microwave Symposium Digest*, 2008, pp. 1453–1456. DOI: 10.1109/MWSYM.2008.4633053.
- [12] Q. Zhang, L. Ton, and Y. Cao, "Microwave Modeling Using Artificial Neural Networks and Applications to Embedded Passive Modeling," in *2007 International Conference on Microwave and Millimeter Wave Technology*, 2007, pp. 1–4. DOI: 10.1109/ICMMT.2007.381409.
- [13] A. Suntives, M. S. Hossain, J. Ma, R. Mittra, and V. Veremey, "Application of artificial neural network models to linear and nonlinear RF circuit modeling," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 11, no. 4, pp. 231–247, 2001. DOI: <https://doi.org/10.1002/mmce.1028>.
- [14] F. Güneş and N. Türker, "Artificial neural networks in their simplest forms for analysis and synthesis of RF/microwave planar transmission lines," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 15, no. 6, pp. 587–600, 2005. DOI: <https://doi.org/10.1002/mmce.20103>.
- [15] J. N. Chavan, A. A. Kale, and S. R. Deore, "Transmission Line Fault Detection Using Wavelet Transform & ANN Approach," in *2022 IEEE Integrated STEM Education Conference (ISEC)*, 2022, pp. 426–432. DOI: 10.1109/ISEC54952.2022.10025222.
- [16] K. A. Alshamaileh, M. J. Almalkawi, R. R. Junuthula, V. K. Devabhaktuni, and P. H. Aaen, "ANN-based modeling of compact impedance-varying transmission lines with applications to ultra-wideband Wilkinson power dividers," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 25, no. 7, pp. 563–572, 2015. DOI: <https://doi.org/10.1002/mmce.20893>.
- [17] C. Yildiz, S. Gultekin, K. Guney, and S. Sagiroglu, "Neural Models for the Resonant Frequency of Electrically Thin and Thick Circular Microstrip Antennas and the Characteristic Parameters of Asymmetric Coplanar Waveguides Backed with a Conductor," *AEU - International Journal of Electronics and Communications*, vol. 56, no. 6, pp. 396–406, 2002, ISSN: 1434-8411. DOI: <https://doi.org/10.1078/1434-8411-54100128>.

-
- [18] Y. Hu, Y. Zhang, H. Lu, D. Tan, and J. Qi, "Scalable modeling of grounded coplanar waveguide for MMICs design using neural network with an effective sampling strategy," *Microelectronics Journal*, vol. 131, p. 105 670, 2023, ISSN: 1879-2391. DOI: <https://doi.org/10.1016/j.mejo.2022.105670>.
- [19] S. K. Mandal, S. Sural, and A. Patra, "ANN- and PSO-Based Synthesis of On-Chip Spiral Inductors for RF ICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 1, pp. 188–192, 2008. DOI: 10.1109/TCAD.2007.907284.
- [20] T. Guillod, P. Papamanolis, and J. W. Kolar, "Artificial Neural Network (ANN) Based Fast and Accurate Inductor Modeling and Design," *IEEE Open Journal of Power Electronics*, vol. 1, pp. 284–299, 2020. DOI: 10.1109/OJPEL.2020.3012777.
- [21] A. Raj and S. K. Sharma, "Device parameter prediction for GAA junction-less nanowire FET using ANN approach," *Microelectronics Journal*, vol. 147, p. 106 192, 2024, ISSN: 1879-2391. DOI: <https://doi.org/10.1016/j.mejo.2024.106192>.
- [22] R. Butola, Y. Li, S. R. Kola, C.-Y. Chen, and M.-H. Chuang, "Artificial Neural Network-Based Modeling for Estimating the Effects of Various Random Fluctuations on DC/Analog/RF Characteristics of GAA Si Nanosheet FETs," *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 11, pp. 4835–4848, 2022. DOI: 10.1109/TMTT.2022.3198659.
- [23] S. I. Yahya et al., "A New Design Method for Class-E Power Amplifiers Using Artificial Intelligence Modeling for Wireless Power Transfer Applications," *Electronics*, vol. 11, no. 21, 2022, ISSN: 2079-9292. DOI: 10.3390/electronics11213608.
- [24] Q. Lin, H.-F. Wu, L.-S. Liu, L.-N. Jia, and X.-Z. Wang, "Optimization Design of RF Power Amplifier Based on ANN," in *2021 International Conference on Microwave and Millimeter Wave Technology (ICMMT)*, 2021, pp. 1–3. DOI: 10.1109/ICMMT52847.2021.9617982.
- [25] J. Choo, T. H. A. Pho, and Y.-H. Kim, "Machine Learning Technique to Improve an Impedance Matching Characteristic of a Bent Monopole Antenna," *Applied Sciences*, vol. 11, no. 22, 2021, ISSN: 2076-3417. DOI: 10.3390/app112210829.
- [26] H. Kabir, Y. Wang, M. Yu, and Q.-J. Zhang, "Neural Network Inverse Modeling and Applications to Microwave Filter Design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 4, pp. 867–879, 2008. DOI: 10.1109/TMTT.2008.919078.
- [27] Z. Guan, P. Zhao, X. Wang, and G. Wang, "Modeling Radio-Frequency Devices Based on Deep Learning Technique," *Electronics*, vol. 10, no. 14, 2021, ISSN: 2079-9292. DOI: 10.3390/electronics10141710.
- [28] D. Lee, G. Shin, S. Lee, K. Kim, T.-H. Oh, and H.-J. Song, "Neural-Network-Based Automated Synthesis of Transformer Matching Circuits for RF Amplifier Design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 11, pp. 4726–4739, 2022. DOI: 10.1109/TMTT.2022.3199756.

-
- [29] J. Wood and D. E. Root, *Fundamentals of Nonlinear Behavioral Modeling for RF and Microwave Design* (Artech House Microwave Library), English. Boston, Mass.: Artech House, 2005.
- [30] G. Gonzalez, *Microwave transistor amplifiers : analysis and design*, eng, 2nd ed. Upper Saddle River, N.J: Prentice Hall, 1997.
- [31] D. M. Pozar, *Microwave engineering*, 4th ed. Hoboken, NJ: Wiley, 2012.
- [32] J. Verspecht and P. Van Esch, “Accurately Characterizing Hard Nonlinear Behavior of Microwave Components with the Nonlinear Network Measurement System: Introducing Nonlinear Scattering Functions,” in *Proceedings of the 5th International Workshop on Integrated Nonlinear Microwave and Millimeterwave Circuits*, Duisburg, Germany, 1998, pp. 17–26.
- [33] H. Qi, J. Benedikt, and P. J. Tasker, “Nonlinear Data Utilization: From Direct Data Lookup to Behavioral Modeling,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 6, pp. 1425–1432, 2009. DOI: 10.1109/TMTT.2009.2019996.
- [34] H. Qi, J. Benedikt, and P. Tasker, “A Novel Approach for Effective Import of Nonlinear Device Characteristics into CAD for Large Signal Power Amplifier Design,” in *2006 IEEE MTT-S International Microwave Symposium Digest*, 2006, pp. 477–480. DOI: 10.1109/MWSYM.2006.249596.
- [35] H. Qi, “Nonlinear data utilization: direct data look-up to behavioural modelling,” Ph.D. dissertation, Cardiff University, 2008.
- [36] S. Woodington, “Behavioural Model Analysis of Active Harmonic Load-pull Measurements,” Ph.D. dissertation, Cardiff University, 2011.
- [37] S. Woodington, R. Saini, D. Williams, J. Lees, J. Benedikt, and P. J. Tasker, “Behavioral model analysis of active harmonic load-pull measurements,” in *2010 IEEE MTT-S International Microwave Symposium*, 2010, pp. 1688–1691. DOI: 10.1109/MWSYM.2010.5517261.
- [38] J. J. Bell et al., “Behavioral model analysis using simultaneous active fundamental load-pull and harmonic source-pull measurements at X-band,” in *2011 IEEE MTT-S International Microwave Symposium*, 2011, pp. 1–4. DOI: 10.1109/MWSYM.2011.5972803.
- [39] P. J. Tasker, “Robust Extraction of Cardiff Model Parameters from Appropriately Tailored Measured Load-Pull Data,” in *2020 IEEE BiCMOS and Compound Semiconductor Integrated Circuits and Technology Symposium (BCICTS)*, 2020, pp. 1–5. DOI: 10.1109/BCICTS48439.2020.9392942.
- [40] M. Tian, J. J. Bell, R. Quaglia, and P. J. Tasker, “An Extraction Method Based on Artificial Neural Network Techniques for Novel Cardiff Model With Reasonable Extrapolation Behavior,” *IEEE Microwave and Wireless Technology Letters*, vol. 34, no. 1, pp. 5–8, 2024. DOI: 10.1109/LMWT.2023.3329979.
- [41] J. J. W. Bell, R. Saini, J. Lees, J. Benedikt, S. Cripps, and P. J. Tasker, “X-band behavioral model analysis using an active harmonic source- pull and load-pull measurement system,” in *Asia-Pacific Microwave Conference 2011*, 2011, pp. 1430–1433.

-
- [42] J. Bell, "Input Harmonic and Mixing Behavioural Model Analysis," Ph.D. dissertation, Cardiff University, 2014.
 - [43] H. Qi, J. Benedikt, and P. J. Tasker, "Novel Nonlinear Model for Rapid Waveform-based Extraction Enabling Accurate High Power PA Design," in *2007 IEEE/MTT-S International Microwave Symposium*, 2007, pp. 2019–2022. DOI: 10.1109/MWSYM.2007.380246.
 - [44] V. Devabhaktuni, M. Yagoub, and Q.-J. Zhang, "A robust algorithm for automatic development of neural-network models for microwave applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 12, pp. 2282–2291, 2001. DOI: 10.1109/22.971611.
 - [45] V. K. Devabhaktuni, M. C. E. Yagoub, Y. Fang, J. Xu, and Q.-J. Zhang, "Neural networks for microwave modeling: Model development issues and nonlinear modeling techniques," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 11, no. 1, pp. 4–21, 2001. DOI: [https://doi.org/10.1002/1099-047X\(200101\)11:1<4::AID-MMCE2>3.0.CO;2-I](https://doi.org/10.1002/1099-047X(200101)11:1<4::AID-MMCE2>3.0.CO;2-I).
 - [46] F. Wang, V. K. Devabhaktuni, C. Xi, and Q.-J. Zhang, "Neural network structures and training algorithms for RF and microwave applications," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 9, no. 3, pp. 216–240, 1999. DOI: 10.1002/(SICI)1099-047X(199905)9:3<216::AID-MMCE7>3.0.CO;2-W.
 - [47] Q. J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*, 1st. USA: Artech House, 2000.
 - [48] F. Wang and Q.-J. Zhang, "Knowledge-based neural models for microwave design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 12, pp. 2333–2343, 1997. DOI: 10.1109/22.643839.
 - [49] M. Ouadefli, M. Et-tolba, A. Tribak, and T. F. Ibanez, "On Selecting Activation Functions for Neural Network-based Digital Predistortion Models," *Progress In Electromagnetics Research C*, vol. 152, pp. 111–120, Jan. 2025. DOI: <https://doi.org/10.2528/pierc24120508>.
 - [50] Hecht-Nielsen, "Theory of the backpropagation neural network," in *International 1989 Joint Conference on Neural Networks*, vol. 1, 1989, pp. 593–605. DOI: 10.1109/IJCNN.1989.118638.
 - [51] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.
 - [52] H. Yeremia, N. A. Yuwono, P. Raymond, and W. Budiharto, "GENETIC ALGORITHM AND NEURAL NETWORK FOR OPTICAL CHARACTER RECOGNITION," *Journal of Computer Science*, vol. 9, no. 11, pp. 1435–1442, Sep. 2013. DOI: 10.3844/jcssp.2013.1435.1442.
 - [53] A. Antoniou and W.-S. Lu, *Practical Optimization Algorithms and Engineering Applications* (Texts in Computer Science), 2nd. New York, NY: Springer, 2021.
 - [54] M. Hagan and M. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994. DOI: 10.1109/72.329697.

-
- [55] F. Dan Foresee and M. Hagan, “Gauss-Newton approximation to Bayesian learning,” in *Proceedings of International Conference on Neural Networks (ICNN’97)*, vol. 3, 1997, pp. 1930–1935. DOI: 10.1109/ICNN.1997.614194.
 - [56] F. Burden and D. Winkler, “Bayesian regularization of neural networks,” *Artificial neural networks: methods and applications*, pp. 23–42, 2008.
 - [57] The MathWorks, Inc., *Levenberg-Marquardt Backpropagation*, [Online;] Natick, Massachusetts, United States, 2006. [Online]. Available: <https://uk.mathworks.com/help/deeplearning/ref/trainlm.html>.
 - [58] Q.-J. Zhang, K. C. Gupta, and V. K. Devabhaktuni, “Artificial neural networks for RF and microwave design: From theory to practice,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 4, pp. 1339–1350, 2003. DOI: 10.1109/TMTT.2003.809179.
 - [59] J. P. Dunsmore, “VNA Measurement Systems,” in *Handbook of Microwave Component Measurements*. 2020, ch. 2, pp. 71–126. DOI: 10.1002/9781119477167.ch2.
 - [60] Hewlett-Packard, *S-Parameter Techniques: A General Perspective*, Application Note 95-1, 1997.
 - [61] Agilent Technologies, *Applying Error Correction to Network Analyzer Measurements*, Application Note 1287-3, Literature Number 5965-7709E, 2002. [Online]. Available: <https://anlauge.umd.edu/Microwave%20Measurements%20for%20Personal%20Web%20Site/5965-7709E.pdf>.
 - [62] Agilent Technologies, *Network Analyzer Basics*, Application Note 5965-7917E, 2010. [Online]. Available: <https://anlauge.umd.edu/Microwave%20Measurements%20for%20Personal%20Web%20Site/Agilent%20NWA%20Basics%205965-7917E.pdf>.
 - [63] A. Rumiantsev and N. Ridler, “VNA calibration,” *IEEE Microwave Magazine*, vol. 9, no. 3, pp. 86–99, 2008. DOI: 10.1109/MMM.2008.919925.
 - [64] S. J. Mason, “Feedback Theory—Further Properties of Signal Flow Graphs,” *Proceedings of the IRE*, vol. 44, no. 7, pp. 920–926, 1956. DOI: 10.1109/JRPROC.1956.275147.
 - [65] R. Speciale and N. Franzen, “Super - TSD, A Generalization of the TSD Network Analyzer Calibration Procedure, Covering n - Port Measurements with Leakage,” pp. 114–117, 1977. DOI: 10.1109/MWSYM.1977.1124378.
 - [66] D. Rytting, “Network Analyzer Error Models and Calibration Methods,” in *ARFTG Conference Digest*, Agilent Technologies, 2001.
 - [67] R. B. Marks, “Formulations of the Basic Vector Network Analyzer Error Model including Switch-Terms,” in *50th ARFTG Conference Digest*, vol. 32, 1997, pp. 115–126. DOI: 10.1109/ARFTG.1997.327265.
 - [68] V. Teppati, A. Ferrero, and M. Sayed, “Two-port network analyzer calibration,” in *Modern RF and Microwave Measurement Techniques*, V. Teppati, A. Ferrero, and M. Sayed, Eds., Cambridge: Cambridge University Press, 2013, ch. 8. DOI: 10.1017/CB09781139567626.

-
- [69] G. Engen and C. Hoer, “Thru-Reflect-Line: An Improved Technique for Calibrating the Dual Six-Port Automatic Network Analyzer,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 27, no. 12, pp. 987–993, 1979. DOI: 10.1109/TMTT.1979.1129778.
- [70] D. Williams and R. Marks, “LRM probe-tip calibrations using nonideal standards,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 2, pp. 466–469, 1995. DOI: 10.1109/22.348112.
- [71] A. Ferrero and U. Pisani, “Two-port network analyzer calibration using an unknown ‘thru’,” *IEEE Microwave and Guided Wave Letters*, vol. 2, no. 12, pp. 505–507, 1992. DOI: 10.1109/75.173410.

Chapter 3

ADS Schematic Design

Implementation

As the ANN model has been created and tested by Mengyue Tian et al. [1], creating a circuit simulation in ADS is also useful for researchers to control various parameters in the model. However, implementing different types of ANN models may present challenges in adjusting the calculation equations and circuit structures. In this chapter, the design of the schematic in ADS will be introduced in detail, and the test results will also be included.

3.1 Introduction

ADS was selected as the CAD environment used in this work. However, the general implementation approach could also be adapted to other CAD environments that provide equivalent capabilities for behavioural-model implementation, parameter importing, equation definition, and automated schematic modification. The ANN model implementation in ADS can be broken down into the following five parts:

- voltages/currents at the receiver and sender
- traveling waves calculation and normalization
- weights and biases importing

- weights and biases normalization and re-normalization
- ANN model calculation

First of all, the voltages/currents receiver and sender are shown in Figure 3.1. The central component is a frequency domain device with four ports. All ports can receive voltages and currents from the external circuit and also send the calculated voltages and currents. The four high-value resistors provide a DC reference path for the relevant nodes, preventing floating-node conditions and improving numerical robustness during simulation while introducing negligible loading to the RF signal path. The far left and right ports are the two pins of this ANN model in ADS.

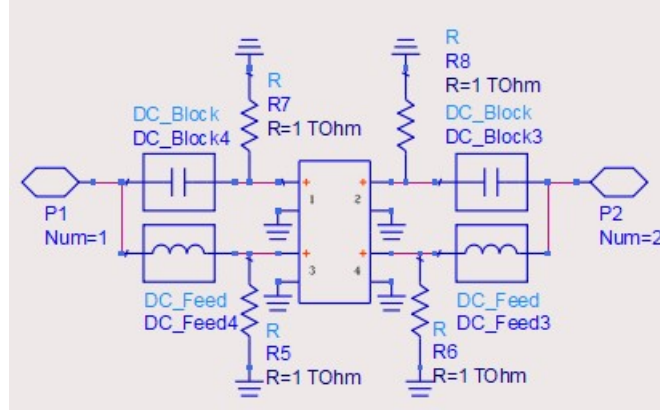


Figure 3.1: The voltages/currents receiver and sender

The next part is the travelling-wave calculation and normalization, where the schematic window capture including full equations is depicted in Figure A.1 in the appendix. The $_{sv}(p,h)$ and $_{si}(p,h)$ are voltages and currents from $P1$ and $P2$ in Figure 3.1, where p stands for port number and h for harmonics, $a_{p,h,o}$ represents for the travelling waves calculated from voltages and currents.

$$Z_{p,h} = 50 + j * 0 \quad (3.1)$$

$$a_{p,h,o} = \frac{(_{sv}(p,h) + Z_{p,h} * _{si}(p,h)) + 1e - 18}{2 * sqrt(Z_{p,h})} \quad (3.2)$$

$$input_{a_{ph},r} = real(a_{p,h,o}) \quad (3.3)$$

$$input_{a_{ph},i} = imag(a_{p,h,o}) \quad (3.4)$$

Additionally, the equations (3.3) and (3.4) prepare for ANN model calculation, as mentioned in Chapter 2, because the ANN model calculation equations only accept real numbers.

Here, Figure 3.2 and equations (3.5) and (3.6) illustrate the weights and biases importing part. It consists of three different parts: DAC, hidden layer parameter importing variables and output layer parameter importing variables.

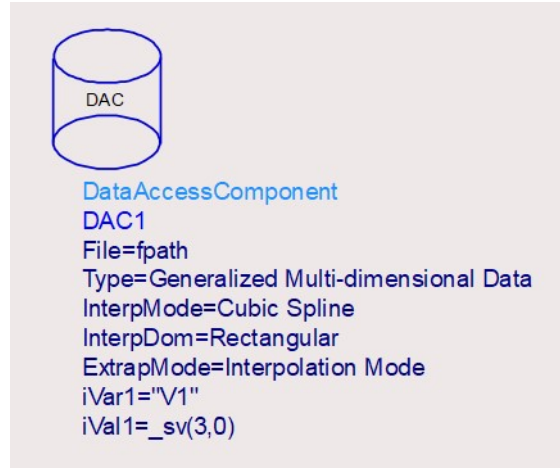


Figure 3.2: DAC for accessing data

In equation (3.5), n stands for the n -th neuron and $m \in \{1, 2\}$ which stands for the weight of the real part and the imaginary part of the input signal, $HW_{n,m}$ is the hidden layer weights and $HB_{n,1}$ is the hidden layer biases.

$$HW_{n,m} = file\{DAC1, "Hidden_Weight_nm"\}$$

$$HB_{n,1} = file\{DAC1, "Hidden_Bias_n1"\} \quad (3.5)$$

In equation (3.6), o stands for the o -th output, $CW_{n,m}$ is the cascade weights, $OW_{o,n}$ is the output layer weights and the $OB_{o,1}$ is the output layer biases.

$$\begin{aligned}
CW_{n,m} &= file\{DAC1, "Cascade_Weight_nm"\} \\
OW_{o,n} &= file\{DAC1, "Output_Weight_on"\} \\
OB_{o,1} &= file\{DAC1, "Output_Bias_o1"\}
\end{aligned} \tag{3.6}$$

Before and after the ANN model calculation, normalization and renormalization for travelling waves are necessary. The following equations (also shown in Figure A.4) display how these two steps work in ADS.

$$a_{2,1,real,norm} = \frac{2 * (input_{a_{ph},r}) - min_f}{max_f - min_f} - 1 \tag{3.7}$$

$$a_{2,1,imag,norm} = \frac{2 * (input_{a_{ph},i}) - min_f}{max_f - min_f} - 1 \tag{3.8}$$

Where the max_f and min_f are from all inputs and outputs of the test dataset used to extract the ANN model.

Also, the de-normalization has to be applied to the output of the ANN model, which is b-waves:

$$b_{2,1,real,de} = \frac{(b_{2,1,real} + 1) * (max_f - min_f)}{2} + min_f \tag{3.9}$$

$$b_{2,1,imag,de} = \frac{(b_{2,1,imag} + 1) * (max_f - min_f)}{2} + min_f \tag{3.10}$$

$$b_{2,0,de} = \frac{(b_{2,1,dc} + 1) * (max_f - min_f)}{2} + min_f \tag{3.11}$$

Finally, the most important part is ANN model calculation, as shown in equation (3.12). Since ADS does not support matrix calculation, the equations need to be applied to each element in every parameter matrix.

$$\begin{aligned}
ANNr(in_r, in_i) = & CW_{n11} \cdot in_r + CW_{n12} \cdot in_i \\
& + OW_{n11} \cdot \tanh \left(\begin{aligned} & HW_{n11} \cdot in_r + HW_{n12} \cdot in_i \\ & + HB_{n11} \end{aligned} \right) \\
& + OW_{n12} \cdot \tanh \left(\begin{aligned} & HW_{n21} \cdot in_r + HW_{n22} \cdot in_i \\ & + HB_{n21} \end{aligned} \right) \\
& + \dots \\
& + OW_{n15} \cdot \tanh \left(\begin{aligned} & HW_{n51} \cdot in_r + HW_{n52} \cdot in_i \\ & + HB_{n51} \end{aligned} \right) \\
& + OB_{n11}
\end{aligned} \tag{3.12}$$

Where $ANNr(\cdot)$ stands for the real part of the output. The ANN model also contains imaginary part and DC part calculations which is shown in Figure A.5 in the appendix.

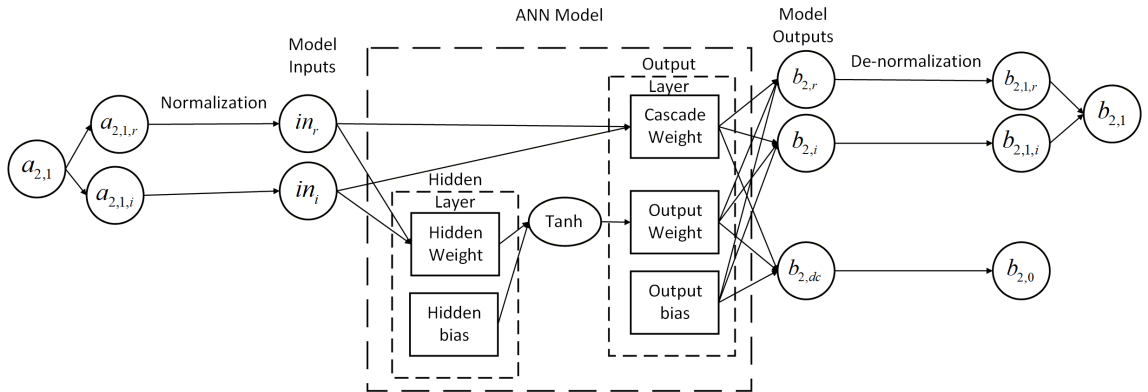


Figure 3.3: Simplified calculation flow diagram of the implemented ANN model, including input normalisation and output de-normalisation.

Figure 3.3 shows the overall calculation flow of the implemented ANN model in ADS. The input travelling wave $a_{2,1}$ is separated into its real and imaginary components, normalised, processed by the ANN, and then de-normalised to generate

the model outputs $b_{2,1}$ and $b_{2,0}$.

3.2 Creating the Dynamic Schematic

In this ADS schematic, all parameters importing and calculations are related to VAR (Variables and Equation) components. Because the central circuit remains unchanged, the AEL code focuses on modifications related to the VAR components. In ADS, a VAR component is used to define variables and mathematical expressions that can be referenced by other components in the schematic. In the developed ANN implementation, the imported model parameters and the equations used for normalisation, ANN calculation, and output processing are defined within VAR components. The AEL code therefore dynamically creates, updates, or removes the required VARs and their contents according to the ANN model structure.

3.2.1 AEL Code Working Flow on Placing the VAR

With the strict naming rules for VAR in the ADS schematic [2], every VAR must be named and checked to ensure it is registered before being placed on the schematic. Figure 3.4 illustrates how the code works to create and register a new VAR in the ADS schematic.

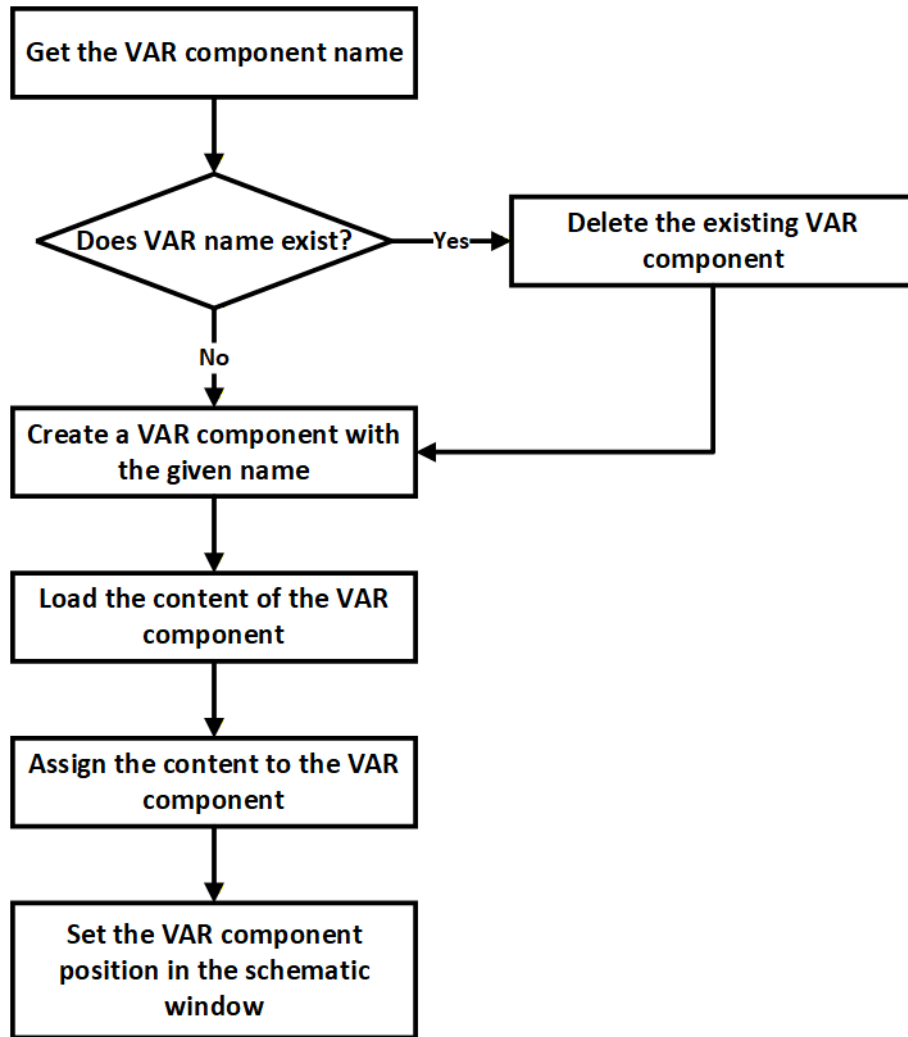


Figure 3.4: AEL code working flow

3.2.2 Adapting Parameter Import for Different ANN Structures

When changing the structure of the ANN model, two main scenarios may occur: altering the number of weights and biases, or modifying the calculation equation of the ANN model.

At the beginning, before calculating the ANN model, the weights and biases must be imported. Figure 3.5 shows how the AEL code determines and generates the VAR for the weights and biases across different layers.

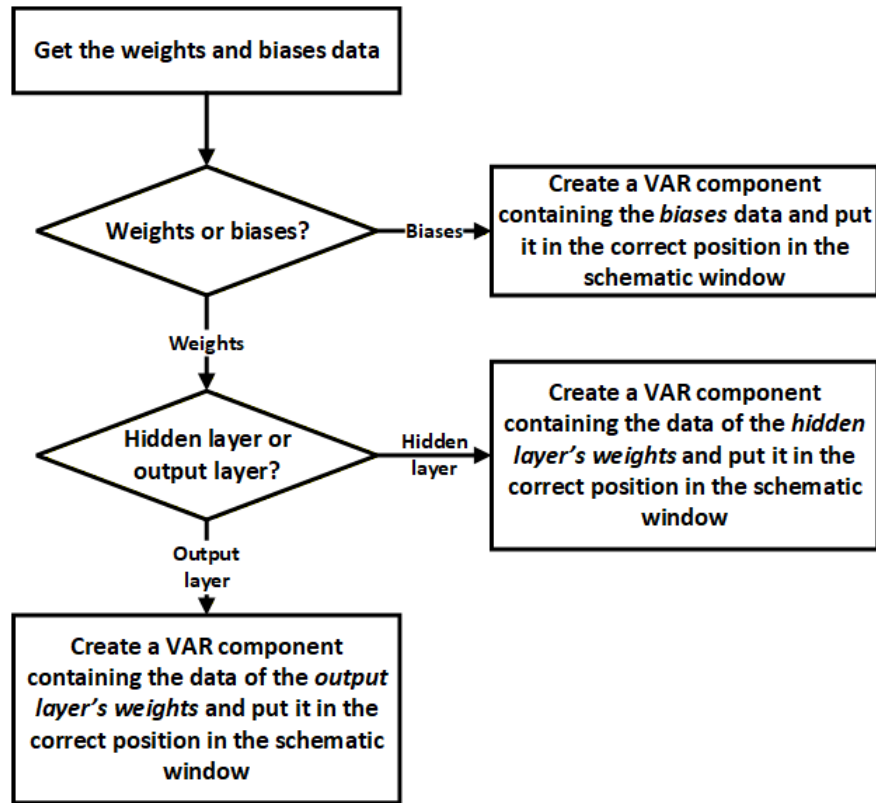


Figure 3.5: Working flow of weights and biases import

Due to the limitations on matrix calculations in ADS, the ANN calculation equations must be expanded to match the number of elements in the output matrix. Each element in the output matrix follows the same calculation process, with the only difference in the subscripts of the weights and biases. Thus, a for-loop is a convenient way to generate the long-hand equations used in the schematic. Figure 3.6 demonstrates that the equation string needs to be initialised with the cascade weight calculation part, then the for-loop is used to add the hidden layer calculation with the $\tanh()$ function and the output layer calculation. Finally, the output layer biases are added.

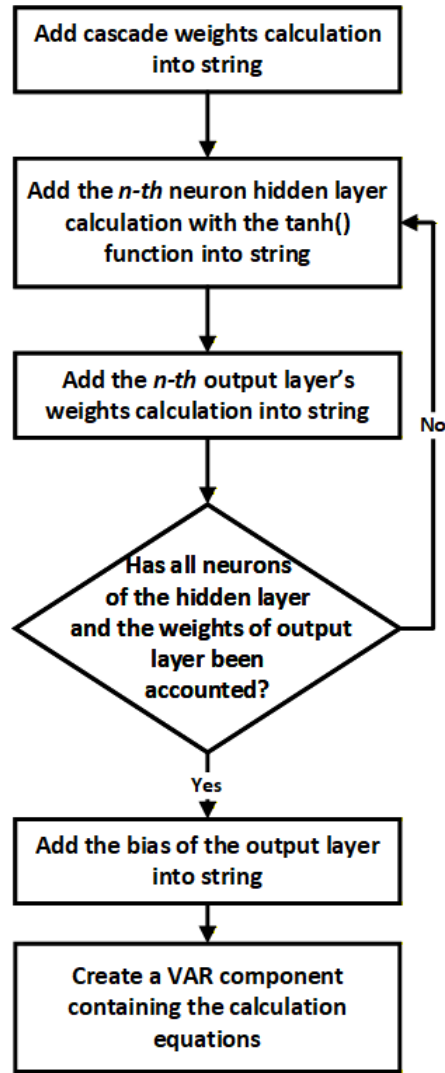


Figure 3.6: Working flow of generating ANN model calculation equations

3.2.3 Implementation

To reduce the complexity of debugging the AEL code, different parts are organised into user-defined functions, which are executed sequentially within the *main()* function (Figure 3.7). Some significant parts will be shown in the following sections. In Figure 3.7, the variables "input_item_cnt" and "output_item_cnt" are used to set the total numbers of the real and imaginary parts for the inputs and outputs.

```

5  defun main()
6  {
7      decl macroContext = de_get_design_context_from_name(
8          "cardiffmodel_ann_ael_lib:anncell_test10:schematic");
9      decl neuron = 6;
10     decl input_item_cnt=6;
11     decl output_item_cnt=12;
12
13     placeImp();
14     // extract original a_xx
15     Translation_eqn();
16     Phase_Normalising();
17     // get real/imag a_xx
18     extract_real_num();
19     ANN_Cal(neuron);
20     Get_Weight(neuron);
21     Bmatrix();
22     FDD2P();
23     // data access
24     DAC();
25 }

```

Figure 3.7: *main()* function

The functions between lines 12 and line 23 in Figure 3.7 contain three types of features: the VAR component creation and placement, ANN calculation equation generation and data importing component setting.

VAR Component Creation and Placement

The function *placeImp()*, *Translation_eqn()*, *Phase_Normalising()*, *extract_real_num()*, *Get_Weight(neuron)*, *Bmatrix()* and *FDD2P()* are used to create and place the VAR component into the correct position and assign the variables and data to the VAR component.

```

defun Phase_Normalising()
{
  /// calculation strings
  decl s10,s11,s12,s13,s21,s22,s23;
  s11 ="a11o/(cos(Phase)+j*sin(Phase))";
  s12 ="a12o/(cos(Phase*2)+j*sin(Phase*2))";
  s13 ="a13o/(cos(Phase*3)+j*sin(Phase*3))";
  s21 ="a21o/(cos(Phase)+j*sin(Phase))";
  s22 ="a22o/(cos(Phase*2)+j*sin(Phase*2))";
  s23 ="a23o/(cos(Phase*3)+j*sin(Phase*3))";

  /// place component
  decl context = de_get_current_design_context();
  decl instlst = de_get_design_instances(db_get_design_name(context)); // Getting all the instance names
  on the current design
  decl testlst=member("PhaseNormalising",instlst);
  decl var="VAR1";
  if(testlst!=NULL)
  {
    de_select_by_name("PhaseNormalising",1); /// deletes any pre-existing instances of
    PhaseNormalising
    de_delete();
  }

  decl itemInfo0SP = de_init_item("VAR");
  de_rotate_image("UP");
  de_place_item(itemInfo0SP, -7.25, -2.75);

  itemInfo0SP = de_edit_item(var);
  de_set_item_id(itemInfo0SP, "PhaseNormalising");
  de_set_item_parameters(itemInfo0SP, list(list(prm("VarFormStdForm",prm("VarNameForm","a11"),prm(
  "VarValueForm",s11)),prm("VarFormStdForm",prm("VarNameForm","a12"),prm("VarValueForm",s12)),prm(
  "VarFormStdForm",prm("VarNameForm","a13"),prm("VarValueForm",s13)),prm("VarFormStdForm",prm(
  "VarNameForm","a21"),prm("VarValueForm",s21)),prm("VarFormStdForm",prm("VarNameForm","a22"),prm(
  "VarValueForm",s22)),prm("VarFormStdForm",prm("VarNameForm","a23"),prm("VarValueForm",s23)))));
  de_end_edit_item(itemInfo0SP);
  de_end_edit_item(itemInfo0SP);
  itemInfo0SP = de_free_item(itemInfo0SP);

  return 0;
}

```

Figure 3.8: *Phase_Normalising()* function

For example, in Figure 3.8, it starts with the content string definition, which is the content assigned to the "*PhaseNormalising*" VAR component. After this, the code section containing the *if*-condition is used to check whether the VAR with name "*PhaseNormalising*" exists. If it exists in the current schematic, the current VAR component will be deleted to avoid new content assignment. Then, the code section including numbers are used to confirm its position in the schematic window. Finally, the built-in function of the AEL *de_set_item_parameters()* is used to assign the strings to the variables in the VAR components.

ANN Calculation Equation Generation

```
defun ANN_Cal(n)
{
  decl cnt,i,strtmp,strtmp1,Cascade_tmp1,Cascade_tmp2,Cascade_tmp1_label,Cascade_tmp2_label,
  Output_tmp1_label,Output_tmp2_label,HO_tmp;
  decl Cascade_tmp3,strtmp2,Output_tmp3,Output3_label,Output3;
  cnt=1;
  Cascade_tmp1="CW_n11*in_r+CW_n12*in_i";
  Cascade_tmp2="CW_n21*in_r+CW_n22*in_i";
  Cascade_tmp3="CW_n31*in_r+CW_n32*in_i";
  HO_tmp="tanh(HW_n11*in_r+HW_n12*in_i+HB_n11)";
  strtmp=strcat("OW_n11*",HO_tmp);
  strtmp1=strcat("OW_n21*",HO_tmp);
  strtmp2=strcat("OW_n31*",HO_tmp);
  for(cnt=2;cnt<=n;cnt++)
  {
    HO_tmp=strcat("tanh(", "HW_n",cnt,"1*in_r+HW_n",cnt,"2*in_i+HB_n",cnt,"1",")");
    strtmp=strcat(strtmp,"+OW_n1",cnt,"*",HO_tmp);
    strtmp1=strcat(strtmp1,"+OW_n2",cnt,"*",HO_tmp);
    strtmp2=strcat(strtmp2,"+OW_n3",cnt,"*",HO_tmp);
  }
  //// Calculate process
  decl Hidden_Output,Output,Output_tmp1,Output_tmp2,Output1,Output2,Output1_label,Output2_label;
  Output_tmp1=strtmp;
  Output_tmp2=strtmp1;
  Output_tmp3=strtmp2;
  Output1_label="ANNr(in_r,in_i)";
  Output2_label="ANNi(in_r,in_i)";
  Output3_label="ANNdc(in_r,in_i)";
  Output1 = strcat(Cascade_tmp1,"+",Output_tmp1,"+OB_n11");
  Output2 = strcat(Cascade_tmp2,"+",Output_tmp2,"+OB_n21");
  Output3 = strcat(Cascade_tmp3,"+",Output_tmp3,"+OB_n31");
}
```

Figure 3.9: $ANN_Cal(n)$ function code

For the ANN used here, only the fundamental frequency is considered. Therefore, only 3 outputs will be counted, which are b_{10} , a_{21} , and b_{21} . Based on the equation mentioned before, the changes in biases and weights only affects hidden weights and biases in the hidden layer. In Figure 3.9, the input variable n represents the number of the hidden layer neurons. Based on the equation (3.12), the cascade weights and output bias calculation are fixed and the hidden layer calculation with the output weights are changed by the number of the hidden layer neurons. However, the number of the hidden layer neurons cannot be smaller than two. The for-loop is used to generate the alterable part of the ANN model calculation.

Data Importing Component Setting

The data importing component is the DAC as mentioned before. Its setting has a few differences with the VAR components. It does not require the content to be pre-defined. Nevertheless, the DAC contains more parameters to be defined. Therefore,

the *de_set_item_parameters()* contains a large number of parameters as shown in Figure 3.10.

```

458 defun DAC()
459 {
460     /// place component
461     decl context = de_get_current_design_context();
462     decl instlst = de_get_design_instances(db_get_design_name(context)); // Getting all the instance names
463     /// on the current design
464     decl testlst=member("DAC1",instlst);
465     decl var="DAC1";
466     if(testlst!=NULL)
467     {
468         de_select_by_name("DAC1",1); /// deletes any pre-existing instances of PhaseNormalising
469         de_delete();
470     }
471
472     decl itemInfo0SP = de_init_item("DataAccessComponent");
473     de_rotate_image("RIGHT");
474     de_place_item(itemInfo0SP, 4.375, 5.375);
475
476     itemInfo0SP = de_edit_item(var);
477     de_set_item_id(itemInfo0SP, "DAC1");
478     de_set_item_parameters(itemInfo0SP, list(prm_ex("ads_datacmps", "dac", "\"test.mdf\""), prm_ex(
479         "ads_datacmps", "T16"), prm_ex("ads_datacmps", "StringAndReference", "\"ModelFile\""), prm_ex("ads_datacmps",
480         ",M3"), prm_ex("ads_datacmps", "IDAC1"), prm_ex("ads_datacmps", "Extrap1"), prm_ex("ads_datacmps",
481         "StringAndReference", "\"v1\""), prm_ex("ads_datacmps", "StdForm", "_sv(3,0)"), prm_ex("ads_datacmps",
482         "StringAndReference", "\"v2\""), prm_ex("ads_datacmps", "StdForm", "_sv(4,0)"), prm_ex("ads_datacmps",
483         "StringAndReference", "\"row\""), prm_ex("ads_datacmps", "StdForm", "0"), prm_ex("ads_datacmps",
484         "StringAndReference", ""), prm_ex("ads_datacmps", "StdForm", ""), prm_ex("ads_datacmps", "StringAndReference",
485         "", prm_ex("ads_datacmps", "StdForm", ""), prm_ex("ads_datacmps", "StringAndReference", ""), prm_ex(
486         "ads_datacmps", "StdForm", ""), prm_ex("ads_datacmps", "StringAndReference", ""), prm_ex("ads_datacmps",
487         "StdForm", ""), prm_ex("ads_datacmps", "StringAndReference", ""), prm_ex("ads_datacmps", "StdForm", ""), prm_ex(
488         "ads_datacmps", "StringAndReference", ""), prm_ex("ads_datacmps", "StdForm", "")));
489     de_end_edit_item(itemInfo0SP);
490     de_end_edit_item(itemInfo0SP);
491     itemInfo0SP = de_free_item(itemInfo0SP);
492
493     return 0;
494 }

```

Figure 3.10: *DAC()* function code

3.2.4 Verification

After applying the AEL code in ADS to generate the ANN model, a load-pull circuit is used to test whether the ANN model functions correctly. In this verification, the performance of a Wolfspeed CG2H40010F device model is predicted using the ANN model. After simulating the ANN model, all data is exported into MATLAB for comparison with the simulation data of the CG2H40010F device. Figure 3.11 shows a plot comparing these two results, and the NMSE (Normalised Mean Squared Error) is taken as acceptable when it is lower than -40 dB, consistent with prior ANN-based RF behavioural modelling studies [3], [4].

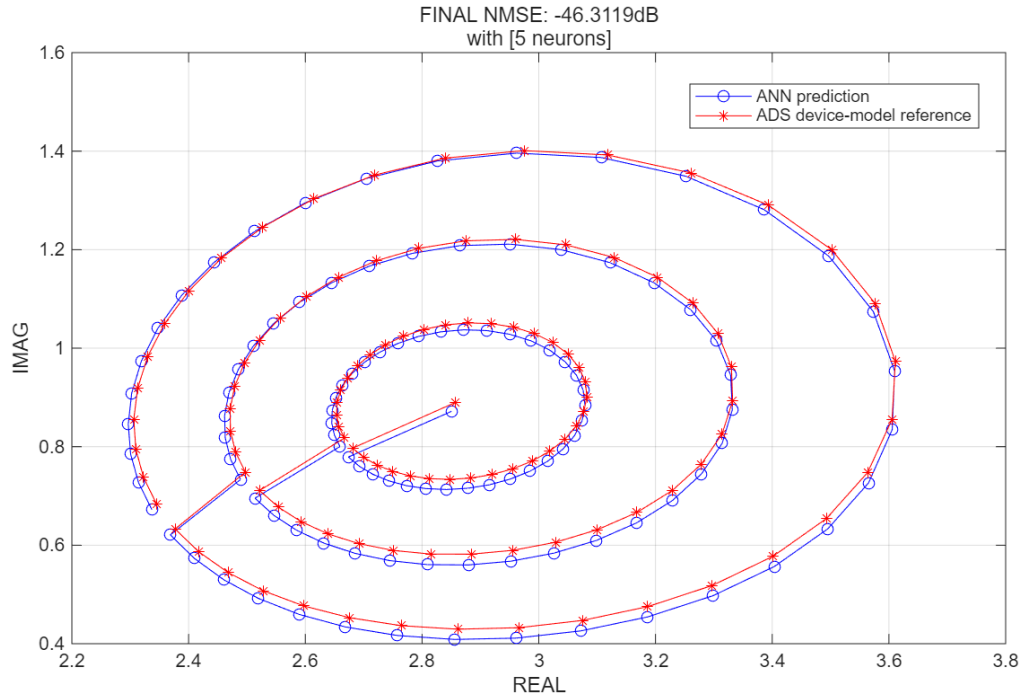


Figure 3.11: Verification result

3.3 Conclusion

Overall, the AEL code plays a significant role in dynamically generating schematics when applying different structures of the ANN model. Its for-loop and VAR generation functions efficiently address the need for schematic modifications. The final verification also demonstrates the ANN model’s ability to predict the performance of nonlinear devices. The AEL code generation function lays a solid foundation for the next part, which involves comparisons between different structures of ANN models and the Cardiff Model.

References

- [1] M. Tian, J. Bell, E. Azad, R. Quaglia, and P. Tasker, “A Novel Cardiff Model Coefficients Extraction Process Based on Artificial Neural Network,” in *2023 IEEE Topical Conference on RF/Microwave Power Amplifiers for Radio and Wireless Applications*, 2023, pp. 1–3. DOI: 10.1109/PAWR56957.2023.10046221.
- [2] I. Agilent Technologies, *Advanced Design System*, Agilent Technologies, Inc., Feb. 2011. [Online]. Available: http://edadownload.software.keysight.com/eed1/ads/2011_01/pdf/ael.pdf.

-
- [3] M. Tian, J. B. Bell, R. Quaglia, E. Azad, and P. J. Tasker, “Artificial neural network nonlinear transistor behavioral models: Structure and parameter determination process based on the Cardiff model,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 73, no. 2, pp. 745–759, 2025. DOI: 10.1109/TMTT.2024.3434959.
- [4] M. Rocio Moure, M. Casbon, M. Fernández-Barciela, and P. J. Tasker, “A Systematic Investigation of Behavioural Model Complexity Requirements,” in *2018 13th European Microwave Integrated Circuits Conference (EuMIC)*, 2018, pp. 77–80. DOI: 10.23919/EuMIC.2018.8539950.

Chapter 4

Implementation Efficiency

Comparison Between ANN and the Cardiff Model in ADS

The previous chapter presented the detailed implementation of a one-hidden-layer ANN behavioural model in ADS, including a flexible schematic structure that allows the number of hidden neurons to be varied. Building upon this implementation, and following the established Cardiff Model formulation described in J. Bell's doctoral work [1], this chapter focuses on a comparative analysis of the implementation efficiency of ANN-based [2] and CM-based behavioural models within a commercial CAD environment.

Under controlled simulation conditions, the execution time of both models is evaluated for large-scale load-pull simulations and extended excitation scenarios. In addition, the computational scaling behaviour of the ANN model is analysed by varying the number of hidden neurons under identical load conditions. Through this analysis, the chapter aims to provide practical insight into the trade-offs between data-driven behavioural modelling approaches and polynomial-based formulations from a CAD execution and implementation perspective.

4.1 Introduction

In practical PA design workflows, behavioural models are often simulated repeatedly across large sets of load impedances or power levels. When individual simulations require only seconds or minutes to complete, CAD execution efficiency may be of limited concern. However, as the number of simulated load points and excitation conditions increases, the total simulation time can extend to hours or even days, making computational efficiency a critical factor in practical large-scale CAD simulations.

Unlike studies that primarily focus on modelling accuracy, this chapter concentrates on the runtime efficiency of behavioural model implementations under varying simulation scales, e.g. different load-pull point sizes and power-level sweeps. By analysing execution time under these scenarios, the chapter aims to highlight practical differences in CAD efficiency between ANN-based and CM-based behavioural formulations.

4.2 Experiment

4.2.1 Simulation Setup

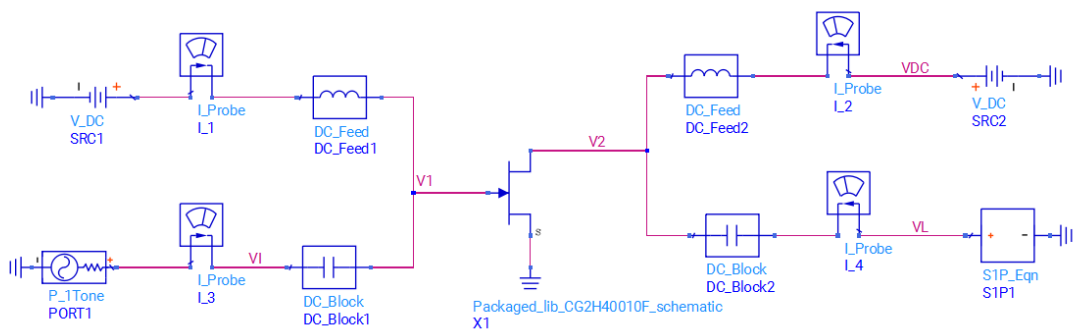


Figure 4.1: Basic load-pull circuit

The dataset used to extract the CM and ANN models was generated in Keysight ADS using the load-pull schematic shown in Figure 4.1. The active device was the

Wolfspeed CG2H40010F 10 W transistor model. In the first case, the frequency was set to 3.5 GHz and the source power was set to 28 dBm, which places the device close to saturation under the simulated matching condition. The load impedance is controlled by changing the magnitude and phase angle of the complex number, $S[1, 1]$, of an S1P (1-port S-Parameters, Equation Based) component. Additionally, the reference impedance is the same as Z_{ref} which is 50Ω set with $Z[1]$ in the S1P component. The simulated load-pull points are located as shown in Figure 4.2.

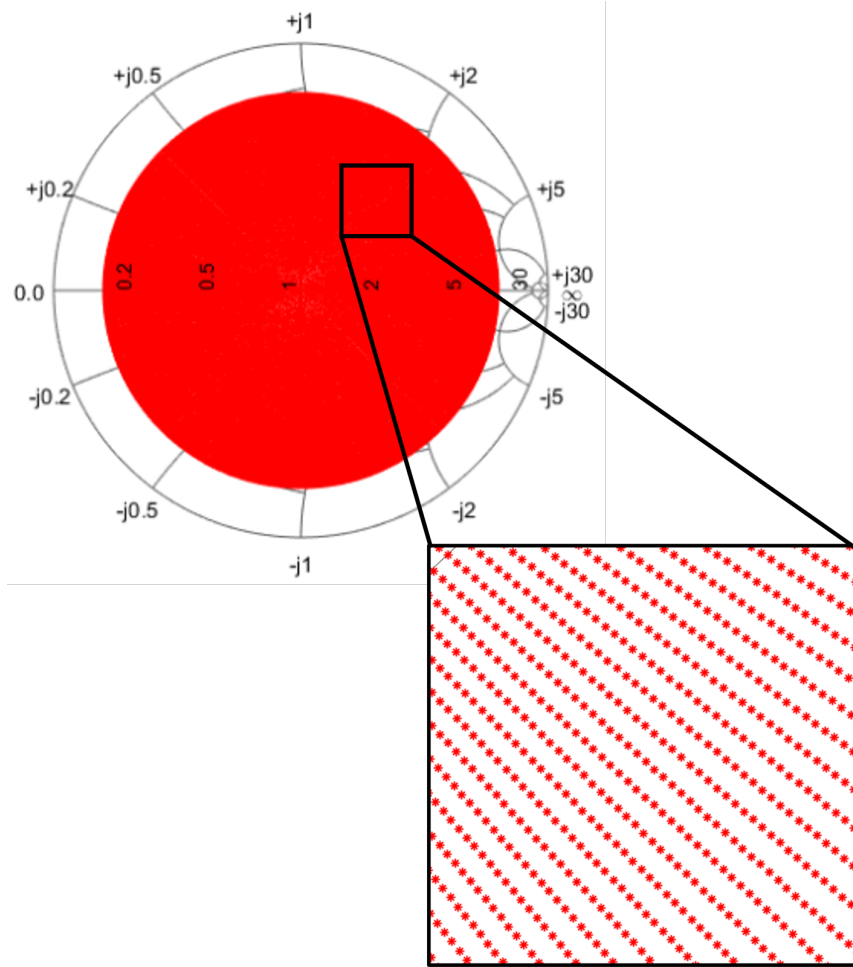


Figure 4.2: Very dense fundamental load point grid for the 72,000 load points with $\Gamma_{max} = 0.8$

The load area covered aims to be a comprehensive, yet representative, coverage of the Smith Chart, therefore a $\Gamma_{max} = 0.8$ was chosen. Specifically, the 200 sample points for magnitude and 360 sample points for phase angle result in a larger load point dataset comprising 72,000 points. This is to emulate simulation scenarios like load-pull for many simulation dimensions (e.g., DC bias, frequency, and harmonics),

but also to accentuate any differences in run time.

4.2.2 Cardiff Model Schematic Simplification for Fundamental Frequency Simulations Only

In CM CAD implementation, the raw input voltages and currents will first be calculated into travelling waves and then the phase of a_{11} will be normalised to zero, as shown in Figure 4.3.

Var Eqn	VAR	Var Eqn	VAR
	Translation Equations		PhaseNormalising
	$a_{11o}=(_sv(1,1)+Z_11*_si(1,1))+1e-18)/2$		$a_{11}=a_{11o}/(\cos(\text{Phase})+j*\sin(\text{Phase}))$
	$a_{12o}=(_sv(1,2)+Z_12*_si(1,2))+1e-18)/2$		$a_{12}=a_{12o}/(\cos(\text{Phase}^2)+j*\sin(\text{Phase}^2))$
	$a_{13o}=(_sv(1,3)+Z_13*_si(1,3))+1e-18)/2$		$a_{13}=a_{13o}/(\cos(\text{Phase}^3)+j*\sin(\text{Phase}^3))$
	$a_{21o}=(_sv(2,1)+Z_21*_si(2,1))+1e-18)/2$		$a_{21}=a_{21o}/(\cos(\text{Phase})+j*\sin(\text{Phase}))$
	$a_{22o}=(_sv(2,2)+Z_22*_si(2,2))+1e-18)/2$		$a_{22}=a_{22o}/(\cos(\text{Phase}^2)+j*\sin(\text{Phase}^2))$
	$a_{23o}=(_sv(2,3)+Z_23*_si(2,3))+1e-18)/2$		$a_{23}=a_{23o}/(\cos(\text{Phase}^3)+j*\sin(\text{Phase}^3))$
	$\text{Phase}=\text{phaserad}(a_{11o})$		

Figure 4.3: V-I to a-b translation equations and equations for phase normalisation

As introduced in chapter 2, for a phase pre-normalised input dataset [3], [4], the CM expression can be defined with:

$$B_{p,h} = \left(\sum_{r=0}^{\max_r} \sum_{n=-\left(\frac{ord-h}{2}-r\right)}^{n=h+\frac{ord-h}{2}-r} M_{p,h,m,n} |A_{2,1}|^m (Q_{2,1})^n \right). \quad (4.1)$$

To implement the travelling-wave multiplication term $|A_{2,1}|^m (Q_{2,1})^n$ in equation (4.1), the function $Ap()$ is defined as

$$Ap(M, P, A_{ph}) = |A_{ph}|^{M-P} * A_{ph}^P \quad (4.2)$$

where M is the magnitude exponent, P is the phase exponent, and A_{ph} is a travelling wave on port p under h harmonic. Therefore, the original CM schematic gives the A matrix calculation definition shown in Figure 4.4

```

Var  VAR
Eqn  A_24
A1=Ap(0,0,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)
Var  VAR
Eqn  A_22
A2=Ap(1,1,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)
Var  VAR
Eqn  A_23
A3=Ap(1,-1,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)
Var  VAR
Eqn  A_19
A4=Ap(2,0,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)
Var  VAR
Eqn  A_21
A5=Ap(2,2,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)
Var  VAR
Eqn  A_20
A6=Ap(2,-2,a21)*Ap(0,0,a22)*Ap(0,0,a23)*Ap(0,0,a12)*Ap(0,0,a13)

```

Figure 4.4: Full 2-input harmonics and 3 output harmonic Amatrix calculations on the ADS schematic

In this case, only fundamental frequency is considered. According to [1], the Amatrix calculation can be simplified into two types considering the effect of the exponentiation on run time.

```

Var  VAR
Eqn  A_25
A1=Ap(0,0,a21)
Var  VAR
Eqn  A_27
A2=Ap(1,1,a21)
Var  VAR
Eqn  A_26
A3=Ap(1,-1,a21)
Var  VAR
Eqn  A_29
A4=Ap(2,0,a21)
Var  VAR
Eqn  A_28
A5=Ap(2,2,a21)
Var  VAR
Eqn  A_30
A6=Ap(2,-2,a21)

```

Figure 4.5: Fundamental only simplify Amatrix calculation under the fundamental frequency

```

Var  VAR
Eqn  A_1
     A1=1
Var  VAR
Eqn  A_2
     A2=a21
Var  VAR
Eqn  A_3
     A3=mag_a21*mag_a21*(1/a21)
Var  VAR
Eqn  A_4
     A4=mag_a21*mag_a21
Var  VAR
Eqn  A_5
     A5=a21*a21
Var  VAR
Eqn  A_6
     A6=mag_a21*mag_a21*mag_a21*mag_a21*(1/(a21*a21))

```

Figure 4.6: Fundamental only Amatrix calculation without exponentiation

Figure 4.5 shows the simplified Amatrix calculation without other harmonic calculations and is used to test whether reducing the implementation size affects runtime. Figure 4.6 shows the direct Amatrix calculation without exponentiation and is used to test whether removing exponentiation provides further runtime improvement relative to Fig. 4.5.

4.2.3 Schematic Adjustments for a Reduced Model

Coefficient Set

When changing the number of coefficients and neurons in both models, some parts of the schematic remain unchanged, while others vary due to the way the model schematic is programmatically populated. To show this clearly on the schematic, a model with 2 coefficients and a model with 6 coefficients are shown in Figs. 4.7 and 4.8 so that the visual differences can be understood. For the CM, these two figures highlight the parts of the schematic that change between the 2-coefficient and 6-coefficient cases.

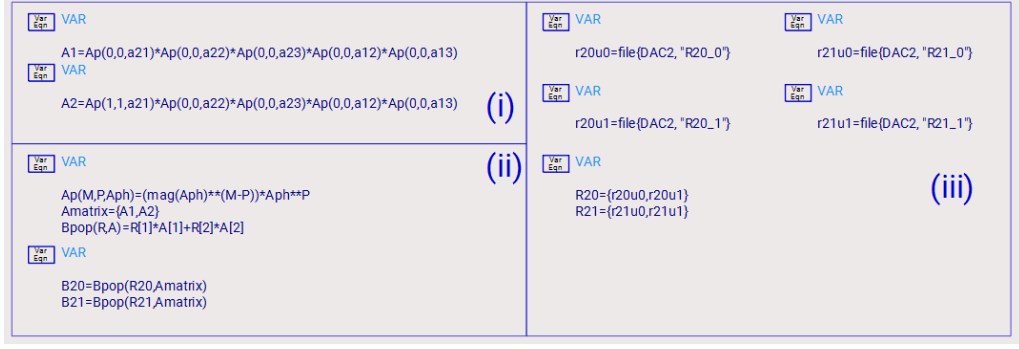


Figure 4.7: 2-coefficient CM

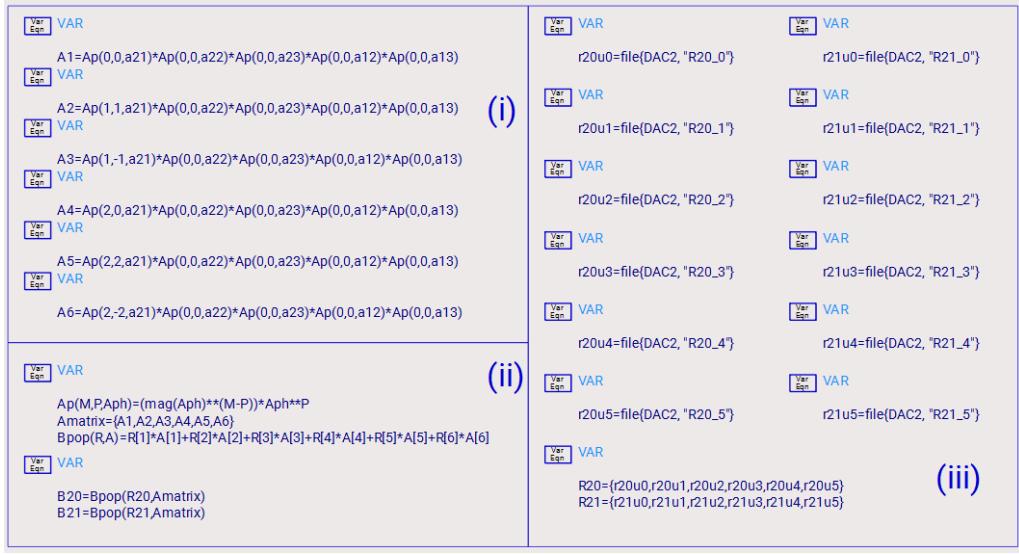


Figure 4.8: 6-coefficient CM

Figures 4.7 and 4.8 show the CM schematics for the 2-coefficient and 6-coefficient cases, respectively. In box (i), the Amatrix calculation is reduced to match the selected coefficient set. In box (ii), the b-wave calculation is simplified accordingly. In box (iii), the number of imported coefficients is also reduced. These changes show that reducing the coefficient set affects not only the coefficient importing part, but also the related Amatrix and b-wave calculations.

Figures 4.9 and 4.10 show the ANN schematics for the 2-neuron and 6-neuron hidden-layer cases, respectively. These two cases were selected as representative lower- and higher-complexity examples from the range of ANN model sizes considered in the runtime study. For consistency in the schematic comparison, the same numerical model-size range was used for the CM coefficient count and ANN

hidden-neuron count, although these quantities do not represent equivalent model complexity. Increasing the number of hidden-layer neurons changes both the hidden-layer calculations and the number of output-layer parameters that must be imported into the schematic.

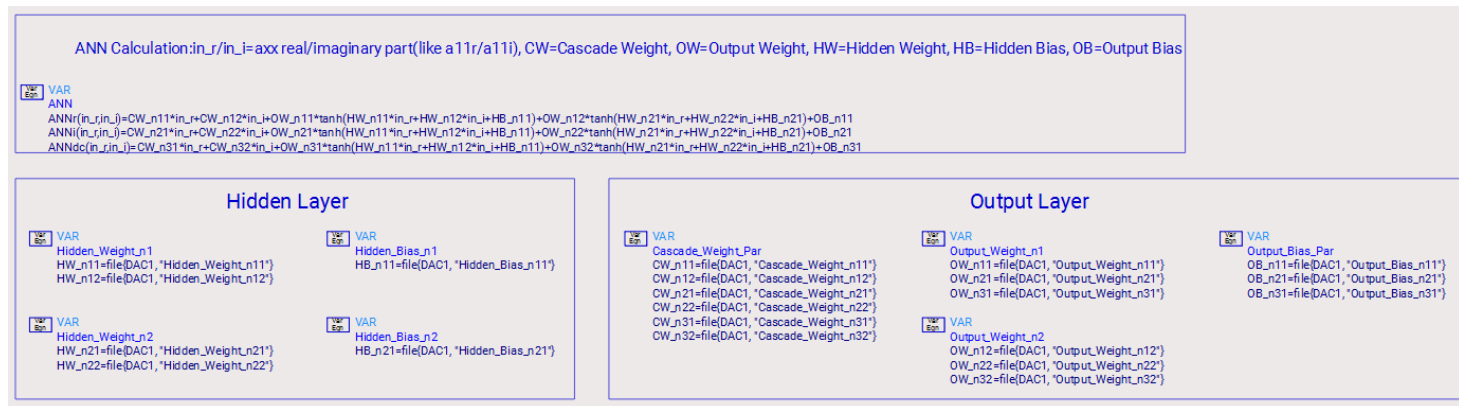


Figure 4.9: 2 hidden layer neurons ANN model

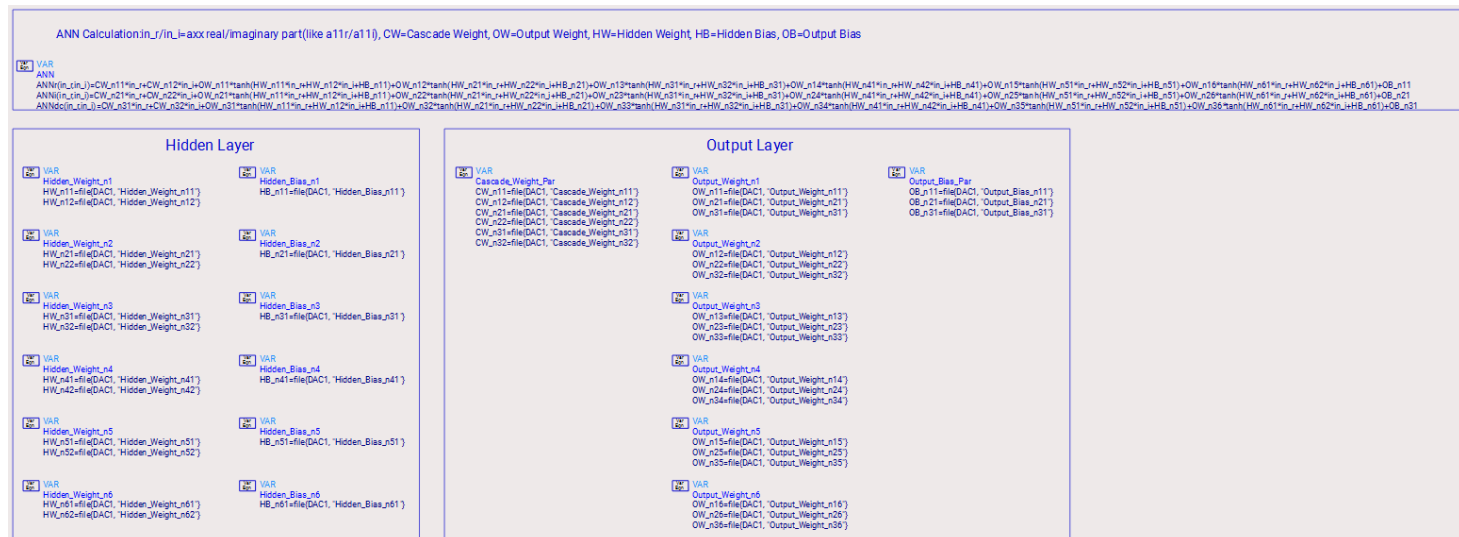


Figure 4.10: 6 hidden layer neurons ANN model

Both the CM and the ANN model ADS schematics are generated using dynamic AEL code, providing flexibility in their CAD implementation based on the model/measurement scenario and serving as a better option for time-saving purposes.

4.2.4 Power Sweep

This section introduces the power sweep simulation setup used to evaluate the runtime of the ANN and simplified CM implementations with the same number of coefficients and hidden-layer neurons, which are 6. For the power sweep tests, the input power is swept from 17 dBm to 28 dBm. This range covers operation from the small-signal region to the near-saturation region.

4.3 Results and Analysis

The first runtime comparison is carried out for the three model implementations under the same load-pull simulation condition. Table 4.1 summarises the corresponding simulation runtime and NMSE results.

Table 4.1: Run time and NMSE comparison of the ANN and CM implementations using the 72,000-point dataset

Model	Model Configuration	Run Time (s)	NMSE (dB)
ANN model	6 hidden neurons	329.19	-51.1603
Original CM	Full harmonic template	150565.18	-165.6786
Simplified CM	Fundamental-only implementation	323.23	-165.6786

The first two rows in Table 4.1 present the simulation run time for the original CM and ANN model simulated with the load points shown in Figure 4.2. It is evident that the results differ significantly due to the presence of numerous redundant equations and variables yielding 1 in the standard CM template for fundamental load-pull.

The modification of the CM focuses on replacing the zero-power calculation

parts, which relate to Figure 4.4. Experiments show that $a_{1,2}$, $a_{1,3}$, $a_{2,2}$, $a_{2,3}$ calculations can all be removed under the fundamental frequency scenario. Although ADS performs these calculations, they still consume time. Therefore, updating the elements in Amatrix gives a simplified CM as shown in Figure 4.5. Furthermore, to remove the effect of the nesting functions and confirm the source of the wasted run time, Figure 4.6 uses a more direct calculation equation. Because the runtime difference between the implementations in Fig. 4.5 and Fig. 4.6 implementations is less than 0.01 s, the simplified CM in this work is taken to refer to the CM schematic containing these simplified equations.

Table 4.1 clearly demonstrates that the simplified CM exhibits a run time that is 6 seconds faster compared to the ANN model. Also, results show a 99% saving of run time with the simplified CM when compared to the original CM. Therefore, when conducting simulations or optimizations that require numerous iterations, removing redundant equations and variables becomes a significant choice in saving simulation time.

To understand the relationship between run time and the number of model coefficients or hidden neurons more clearly, another experiment was conducted. A smaller set of load points was applied since the observed relationship between runtime and coefficient/neuron-number is consistent between large and small datasets. Also, a smaller dataset is more representative of a measured dataset. The new load points were sampled from the same area as shown in Figure 4.2, using 10 magnitude values from 0 to 0.8 and 36 phase values from 0° to 350° , resulting in 360 load points in total. The number of CM coefficients and ANN hidden-layer neurons was varied from 1 to 8 to examine the change in runtime with model size. The corresponding runtime trends are illustrated in Figure 4.11.

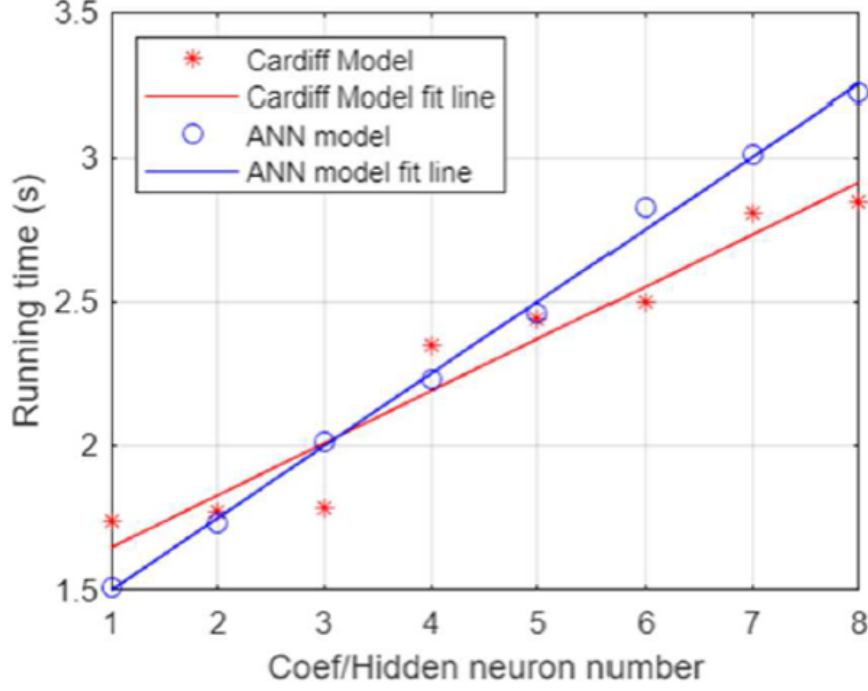


Figure 4.11: Model complexity run time comparison for the CM and ANN models

Figure 4.11 clearly shows a linear relationship between run time and the number of hidden neurons for the ANN model, and a step-shaped increase related to the CM coefficient numbers. The addition of one new neuron in the ANN model only adds on one $\tanh()$ function calculation. This can also be seen from the ANN output expression, which may be written in compact form as

$$b_2 = k + \sum_{i=0}^N OW_i \tanh(v_i) + OB, \quad (4.3)$$

where k denotes the cascade-weight contribution and v_i denotes the input to the i -th hidden neuron, as defined in (3.12). Increasing N by one adds one additional $\tanh(\cdot)$ calculation and its corresponding output-weight term. Since the same structure is repeated for each neuron, the runtime increases approximately linearly with the number of hidden-layer neurons. The reason for the run time rise between 3 coefficients and 4 coefficients for the CM is the exponents of $|a_{2,1}|$. In Figure 4.4, the exponents of $|a_{2,1}|$ for the first 3 coefficients are 0 and 1, however after that

the $b_{2,1}$ calculations including $|a_{2,1}|^2$, will be different from the ANN model. Upon adding the 7th and 8th coefficients to the CM, where the exponent value of $|a_{2,1}|$ is 3, the run time exhibits another step increase. This proves that the CM has a linear relationship between run time and coefficient number, but in a step-shaped following the increasing exponents of $|a_{2,1}|$. Additional tests based on the implementations shown in Figs. 4.5 and 4.6 show that the runtime difference between the exponentiation and repeated-multiplication forms is less than 1% under the same simulation conditions, indicating that the implementation difference contributes slightly to the stepwise runtime increase. The relationships shown in Figure 4.11 indicate that although there are similarities, data that can be modelled accurately by the simpler method should be the one chosen for best simulation runtime efficiency.

In the end, for the power sweep simulation on the ANN model and simplified CM, because they have similar run time with the large dataset containing 72,000 load points, each power-level will be tested with 6,000 load points sampled from the area shown in Figure 4.2. Thus, the total number of load points remains unchanged. Table 4.2 shows the run time difference between the two models under different conditions.

Table 4.2: Run time comparison of the ANN and simplified CM models

Model	Run Time (s)	
	72,000 Load Points	Power Sweep
ANN model	329.19	435.71
Simplified CM	323.23	226.08

The results for the 72,000 load-point and power-sweep cases indicate that the run time changes in opposite ways when simulating the power sweep for two models with the same load points. The ANN model run time increases because it imports more coefficients than CM. For the ANN model, its schematic imports 39 biases and weights for every power level. It also requires two more coefficients to load in, max_f and min_f , which are used before and after the ANN model calculation as stated in

the previous chapter, equations (3.7)-(3.11). In Figure 4.8, the schematic screenshot clearly shows that the 6-coefficient CM only requires 12 coefficients to calculate the reflect wave, which means it wastes less run time on importing coefficients. Additionally, for each power-level, the schematic only simulates 6,000 load-points, which is far less than 72,000 points, allowing the ADS software to use less memory to calculate the whole load-point set. This results in less run time for the CM in power sweep simulations.

4.4 Conclusion

In conclusion, this chapter presented a novel run time comparison between the CM and ANN model ADS implementations. The original CM template wastes 99% of the time when simulating a large load-pull dataset. However, after removing redundant equations and variables from the original CM template, the simplified CM displays similar or slightly faster run time when compared with the ANN model for the same load-point condition. For repeated simulations across different power levels, the simplified CM displays higher efficiency than the ANN model. The study also showed that the ANN run time increases linearly with the number of hidden neurons, while the CM run time increases stepwise as higher-order terms are added. Overall, the run time difference between the ANN and simplified CM implementations depends on the simulation scenario and model structure. When acceptable accuracy is maintained, the simpler implementation should be preferred for better simulation efficiency.

References

- [1] J. Bell, "Input Harmonic and Mixing Behavioural Model Analysis," Ph.D. dissertation, Cardiff University, 2014.
- [2] M. Tian, J. Bell, E. Azad, R. Quaglia, and P. Tasker, "A Novel Cardiff Model Coefficients Extraction Process Based on Artificial Neural Network," *2023 IEEE Topical Conference on RF/Microwave Power Amplifiers for Radio and Wireless Applications*, pp. 1–3, 2023. DOI: 10.1109/PAWR56957.2023.10046221.

-
- [3] P. J. Tasker, “Robust Extraction of Cardiff Model Parameters from Appropriately Tailored Measured Load-Pull Data,” in *2020 IEEE BiCMOS and Compound Semiconductor Integrated Circuits and Technology Symposium (BCICTS)*, 2020, pp. 1–5. DOI: 10.1109/BCICTS48439.2020.9392942.
 - [4] T. Hussein, A. Al-Rawachy, J. Benedikt, J. Bell, and P. Tasker, “Global Behavioural Model Generation Using Coefficients Interpolation,” in *2019 IEEE MTT-S International Microwave Symposium (IMS)*, 2019, pp. 200–203. DOI: 10.1109/MWSYM.2019.8700998.

Chapter 5

Type B Error Detection and Integrity Assessment in Load-Pull Measurement Systems

In the previous chapter, the simulation runtime efficiency of behavioural model implementations in ADS was investigated. While efficient simulation is important for circuit design, the accuracy of behavioural models strongly depends on the measured travelling-wave data used during model extraction [1], [2], [3]. In load-pull measurements, uncertainties present in the measurement system can propagate directly into extracted behavioural model coefficients. Consequently, understanding the sources of measurement uncertainty and assessing the integrity of the measurement system are necessary for ensuring reliable behavioural modelling.

Unlike Type A uncertainties, which consist of random noise and can be reduced through repeated measurements and averaging, Type B uncertainties correspond to consistent systematic deviations that remain after calibration and therefore propagate into all of the extracted travelling waves and directly affect the extraction of behavioural model coefficients. This chapter concentrates on systematic (Type B) uncertainties introduced by a calibrated load-pull measurement system using a standard thru. By measuring the travelling waves of the calibrated thru over a range

of reflection coefficients, systematic distortions in the measurement system can be identified. The extracted distortion is then used to build a reflection-dependent systematic error model, representing the systematic uncertainty as additive travelling-wave offsets. Then, this error model can be propagated to device measurements obtained from a PA using the same load-pull system. CMs generated from the adjusted dataset are then compared with models extracted directly from the original measured data, allowing the effect of systematic measurement uncertainties on behavioural model extraction to be evaluated.

Compared with traditional large-signal modelling studies that mainly evaluate behavioural models using fitting accuracy metrics, this chapter introduces a new measurement system integrity metric linking calibration standard measurement uncertainty to behavioural model coefficient stability. A systematic error model is extracted from a calibrated thru standard and propagated into device measurements. Furthermore, two integrity metrics are proposed. The first is a measurement-domain integrity metric derived from residual statistics under 3σ bootstrap resampling, which defines the acceptable level of systematic distortion in the measured travelling-wave data. The second is a model-domain Coefficient Deviation Index (CDI) that quantifies the sensitivity of CM parameters to systematic distortions. This provides a quantitative self-diagnostic method for assessing calibration robustness and measurement system integrity beyond traditional statistical assessments.

5.1 Introduction

In a load-pull measurement system, almost every component introduces systematic errors, including the VNA, couplers, bias tees, and circulators. Each component contributes different sources of uncertainty, which subsequently propagate through multiple stages of RF characterisation, such as travelling wave extraction, behavioural model coefficient determination, and power prediction calculations. For the CM, the coefficients are directly extracted from the calculated travelling waves. Consequently, any residual systematic errors present after calibration can propa-

gate into the extracted coefficients and the final model. Although vector calibration techniques such as TRL correct the dominant systematic errors represented by the calibration error model, they cannot completely remove all measurement uncertainty. Residual systematic deviations can remain because of uncertainties in the calibration standards, imperfect connector and interconnection repeatability, finite instrument directivity and tracking performance, and changes in the measurement system between calibration and subsequent load-pull measurements, such as cable movement or system drift.

Therefore, the systematic errors considered in this chapter are not the nominal error terms already removed by the TRL calibration, but the residual systematic deviations that remain in the calibrated measurement system. Even when the final NMSE remains low, these potential residual errors may still exist, since the behavioural model can adjust its coefficients to fit the distorted measurement data.

The *Guide to the Expression of Uncertainty in Measurement* (GUM) [4] classifies measurement uncertainties into two categories: Type A and Type B. Type A uncertainties are typically evaluated from repeated observations and are commonly associated with random noise. Various approaches exist to reduce their influence, including repeated measurements with statistical averaging and instrument noise reduction techniques [4], [5]. In calibrated measurement systems, the influence of Type A effects can often be reduced through repeated measurements and statistical averaging. In contrast, Type B uncertainties are evaluated based on other available information, such as calibration-standard specifications, instrument specifications, prior measurement data, or engineering knowledge.

While some instruments provide explicit systematic error bounds in their documentation, others do not. When such information is unavailable, the magnitude of systematic errors must be evaluated from measurement data, which makes the reliability of the measured results more difficult to assess. Such uncertainties may include residual systematic effects that remain after calibration and cannot be reduced simply by repeating and averaging the same measurement. These residual

effects can therefore make a significant contribution to the overall uncertainty of a calibrated load-pull measurement system.

Additionally, Type A and Type B refer to how the uncertainty is evaluated, rather than the physical source of the uncertainty. For example, temperature variation may be evaluated using Type A methods if its influence is estimated statistically from repeated measurements, or using Type B methods if it is estimated from environmental limits, instrument specifications, or prior information.

Consequently, this chapter adopts an experimental approach to estimate the Type B uncertainties of the load-pull measurement system by characterising a standard thru using available system information. These estimated uncertainties are then used to construct a systematic error model, which is applied to PA measurement data to investigate their influence on the extracted coefficients and to derive a measurement-domain distortion threshold.

Finally, this chapter introduces a measurement system integrity assessment metric to investigate the propagation of systematic errors into the extracted CM coefficients. Beyond traditional assessment metrics such as NMSE and root mean square (RMS) error, the proposed integrity assessment method evaluates systematic distortions in both the measurement-domain and the model-domain. These metrics establish thresholds for detecting systematic distortions that may remain hidden in the measurement system when traditional accuracy metrics appear acceptable. In addition, controlled distortion tests with different magnitudes are performed to investigate how systematic measurement errors affect device performance and load-pull contour behaviour.

5.2 Experimental Setup

This section first introduces the measurement system and its calibration. It then presents the thru measurements and the Cree device measurements, with figures and photographs displaying the structure of the system.

5.2.1 Calibration

As introduced in Chapter 2, several vector calibration methods are available. In this work, TRL calibration was applied to the VNA measurement system. The calibration measurements were controlled and processed using software developed by Mesuro. During the calibration, the thru, line and reflect standards were connected to the measurement ports, and the corresponding travelling-wave responses were measured by the VNA. The Mesuro software then used these measured results with the TRL algorithm to calculate the calibration error terms.

Compared with a conventional VNA calibration mainly used for corrected S-parameters, the calibration in this nonlinear vector measurement system is also used to obtain calibrated travelling-wave quantities for the following load-pull measurements. The calculated error terms are exported and applied to the measured travelling waves in the load-pull software. In addition, a separate power calibration is required to determine the absolute magnitude of the travelling waves, because TRL calibration only provides relative vector calibration.

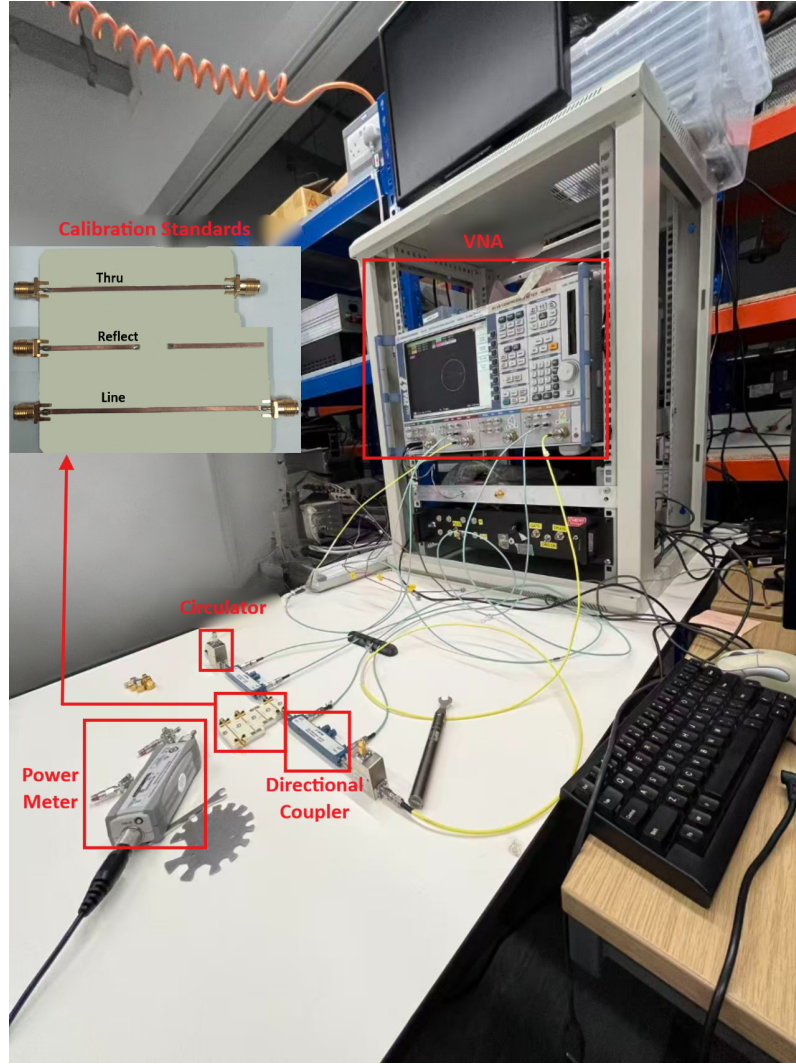


Figure 5.1: VNA calibration setup, showing the main measurement components and the TRL calibration standards

Fig. 5.1 displays one of the steps in the calibration of the VNA, which is the measurement of the thru. The VNA provides the excitation signal and measures the travelling waves through reflectometers at each port. The thru standard establishes a direct connection between the two ports, providing a known reference response that allows the calibration process to characterise the uncertainties within the measurement paths. The calibration measurement steps are as follows: power calibration, thru measurement, line measurement and reflect measurements for ports 1 and 2. They are not required to be done in a strict measurement order. For each calibration step, the corresponding calibration standard is connected to the measurement ports and the measurement is started through the Mesuro software. The

measured response is then recorded and displayed in the software. TRL calibration standard measurements are straightforward to deploy. However, power calibration requires another device, a power meter, shown at the bottom left. Power calibration is significant and necessary because the TRL algorithm only determines relative wave quantities, which means it does not establish the absolute source power [6], [7].

5.2.2 Thru Measurements

After calibration of the VNA, the error terms can be exported based on the TRL calibration algorithm. For the load-pull measurements of the thru and the devices, software named Focus Device Characterization Suite (FDCS), developed by Focus Microwave, was used. By importing the file containing the error terms, the built active load-pull circuit can use these terms to remove most of the effects arising from Type A errors. Figure 5.2 shows the schematic diagram of the measurement system used for the thru measurement.

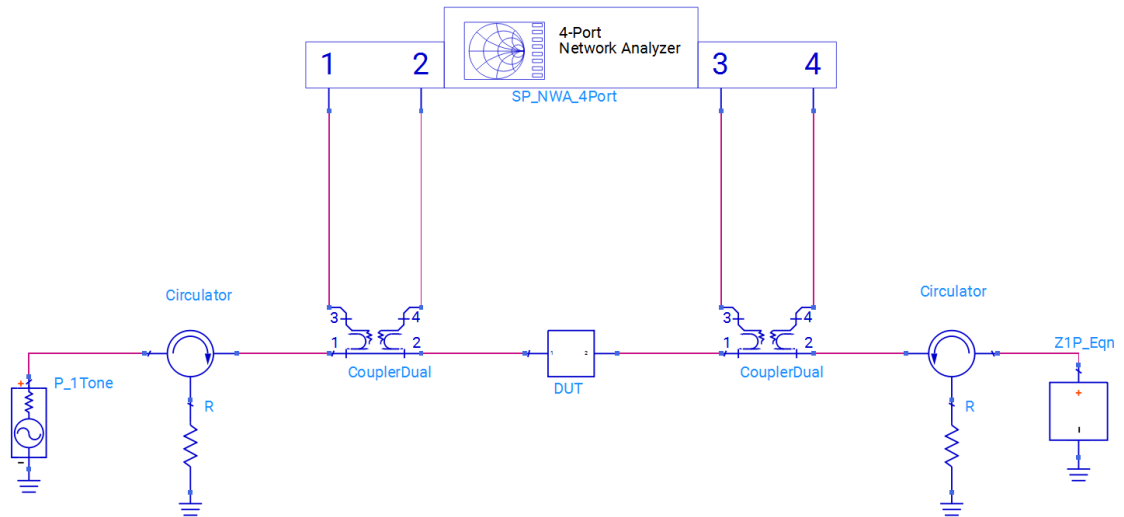


Figure 5.2: Schematic diagram of the thru load-pull measurement

The area covered by the load points is shown in Fig. 5.3. The load points were distributed across the Smith chart and extended as close as possible to the outer region of the Smith chart in order to maximise coverage, thereby improving the interpolation accuracy used in generating the Type B error model described in

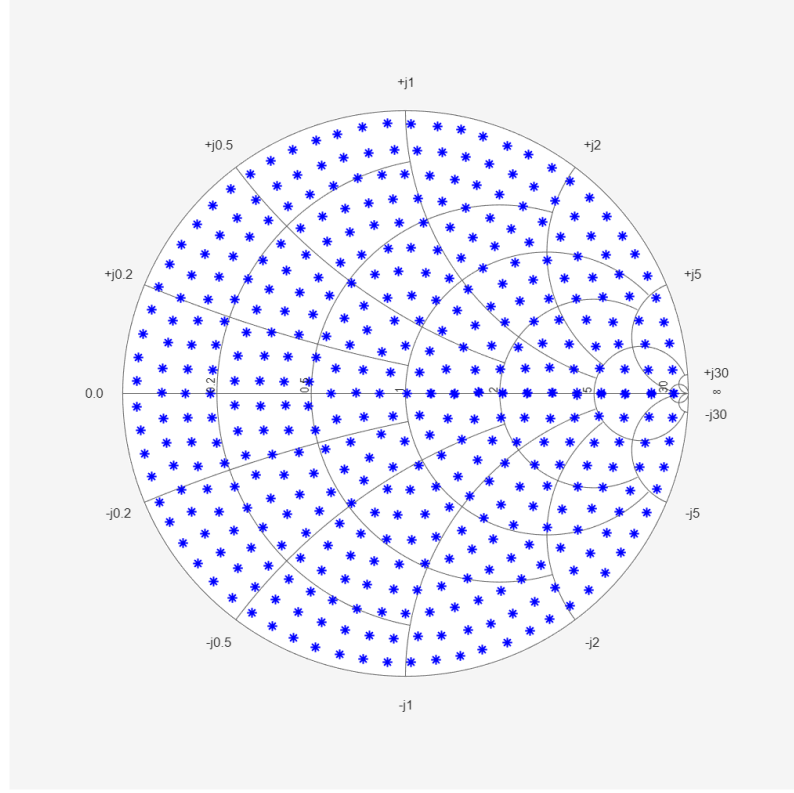


Figure 5.3: Smith chart of the thru load points

5.2.3 Device Measurements

The GaN HEMT CGHV1F006 was selected as the device under test (DUT) for measurement in the load-pull system. The measurement used the same VNA and the same FDCS software as used for the thru measurement, but the circuit is slightly different from that shown in Figure 5.2. Figure 5.4 displays the load-pull measurement system.

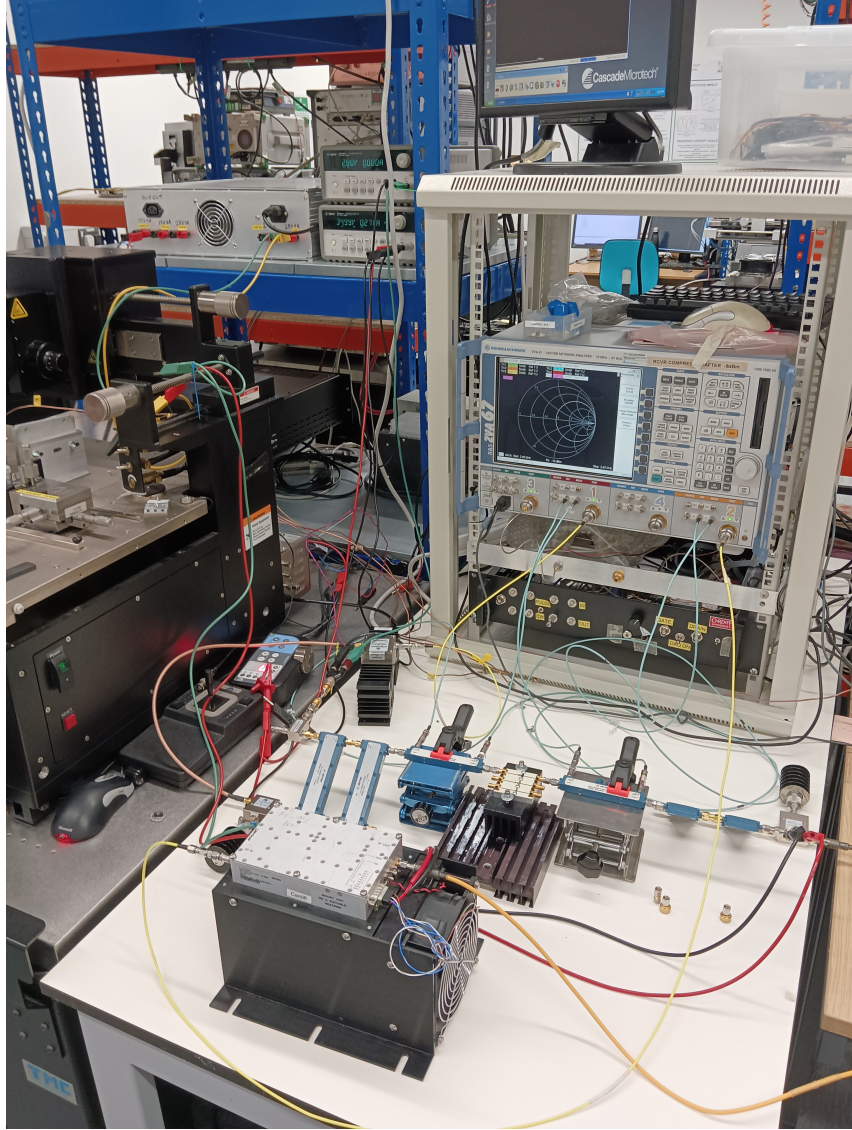


Figure 5.4: Load-pull measurement system used for device measurements

The measurement system incorporates external driver amplifiers and couplers at both ports of the DUT. In load-pull measurements, an external driver amplifier is required to provide sufficient RF power to excite the DUT into its large-signal operating region. Moreover, insertion losses introduced by couplers, bias tees, and interconnecting cables significantly reduce the available drive power at the DUT. The driver amplifier compensates for these losses, thereby enabling operation in compression and saturation. Furthermore, the amplifiers in the active loops provide the signals needed to realise the required source and load conditions during impedance sweeps [8].

Couplers are employed to inject DC bias into the DUT while maintaining RF

isolation from the VNA ports, allowing RF excitation and DC biasing to be applied simultaneously without interfering with each other. Fig. 5.5 illustrates the schematic diagram of the load-pull measurement system, including the interconnection between the DUT and the measurement instruments.

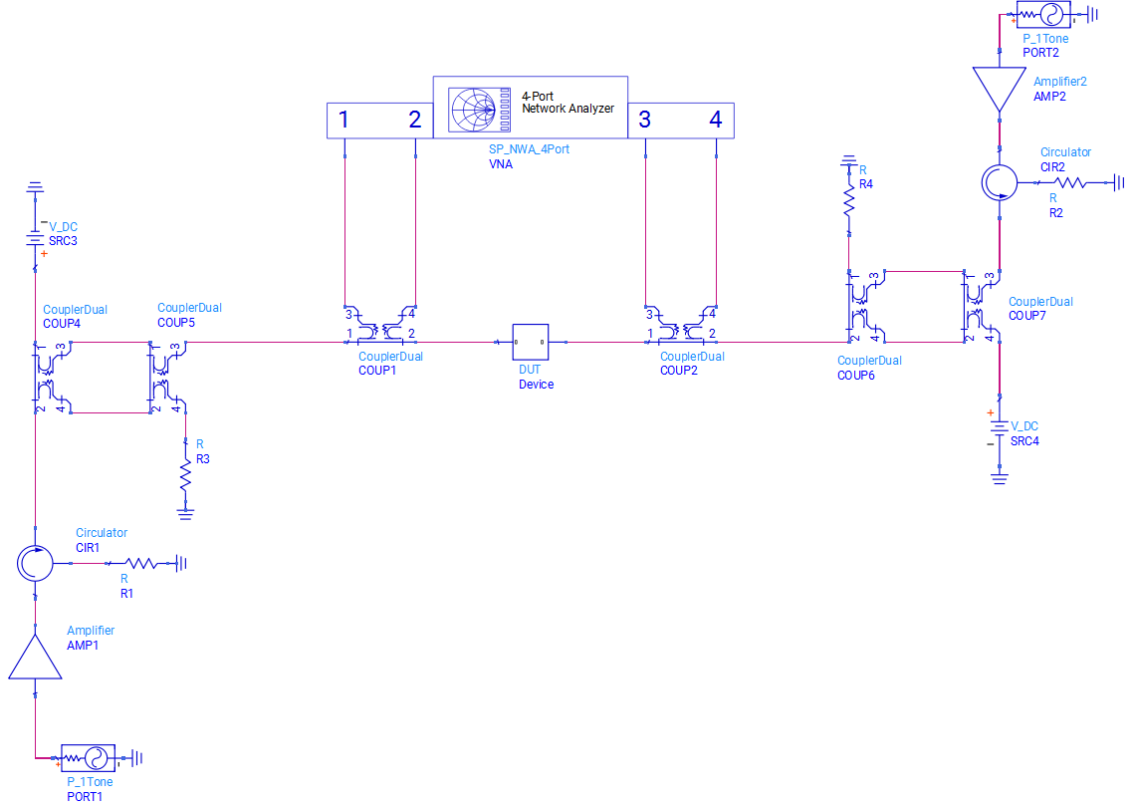


Figure 5.5: Schematic diagram of the load-pull measurement system used for device measurements

5.3 Results and Error Model Analysis

This section presents the results from the thru and GaN HEMT measurements. The thru measurements are used as a reference standard to extract error models. The error models are then applied to the GaN HEMT measurement results, and their impact on the coefficients is analysed.

5.3.1 Thru Measurement Results and Error Model Extraction

To analyse the residual measurement errors in the calibrated system, the ideal behaviour of a perfect thru is first defined as:

$$[S_{thru}] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (5.1)$$

The travelling-wave relations satisfy

$$b_2 = a_1, b_1 = a_2. \quad (5.2)$$

The travelling-wave transmission of the thru under load-pull conditions is defined as

$$H(\Gamma) = \frac{b_2}{a_1}. \quad (5.3)$$

To clarify the system loss in the measurements, the reference level of the thru can be defined as:

$$H_0 = \text{median}\{H(\Gamma_i)\} \quad (5.4)$$

In this chapter, $H_0 = \text{median}\{20\log_{10}\frac{b_2}{a_1}\} = 0.073$ dB. The median is used to obtain a robust estimate of the central magnitude level, since it is less sensitive to abnormal values or local measurement deviations in the dataset.

Since measurements are available only at a finite set of tuner impedances Γ_i , the residual error is initially obtained as a discrete set of samples, which is defined as:

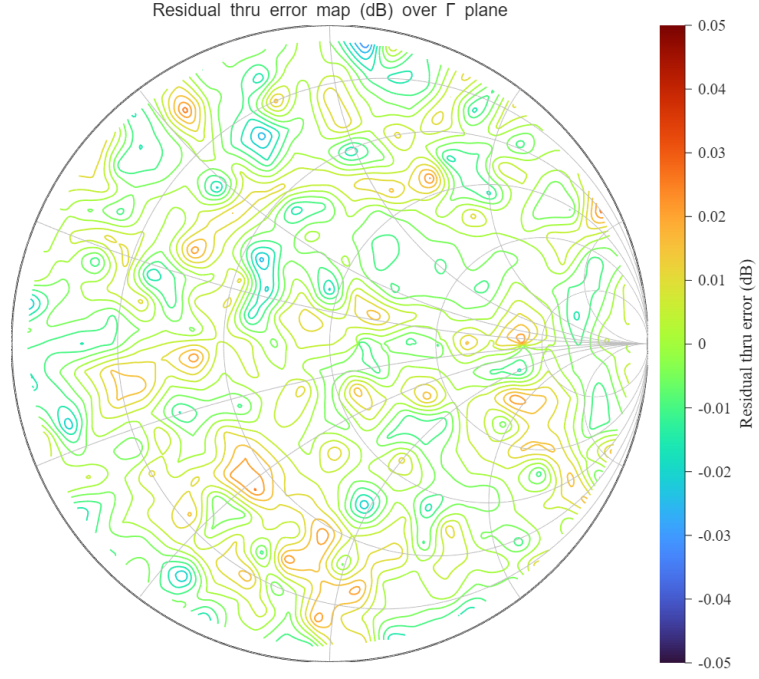
$$R(\Gamma) = \frac{H(\Gamma)}{H_0}. \quad (5.5)$$

For an ideal thru, $H(\Gamma_k) = H_0$ for all Γ_k ; hence $R(\Gamma_k) = 1$. Any deviation of $R(\Gamma_k)$ from unity therefore represents the Γ -dependent systematic error introduced by the measurement system.

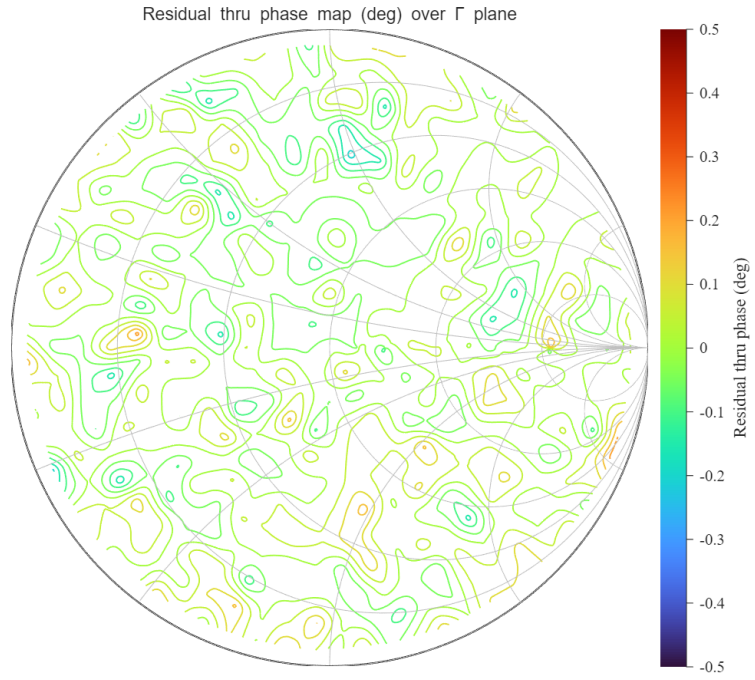
The magnitude and phase residuals are subsequently expressed as

$$R_{mag}(\Gamma_k) = 20\log_{10}|R(\Gamma_k)|, R_{pha}(\Gamma_k) = \angle R(\Gamma_k). \quad (5.6)$$

Figures 5.6a and 5.6b show the corresponding residual magnitude and phase evaluated at the measured load points.



(a) Residual thru error map (dB) over Γ plane



(b) Residual thru error map (deg) over Γ plane

Figure 5.6: Residual thru error maps

To assess the modelling performance, RMS and 95th percentile residuals are

computed for both magnitude and phase. The RMS metric evaluates the overall fitting accuracy by measuring the average magnitude of the residual error. The 95th percentile residual represents the upper range of the error distribution, capturing near worst-case behaviour while avoiding excessive sensitivity to a small number of abnormal values. These statistics are summarised in Table 5.1.

Table 5.1: Magnitude and phase residual statistics

Statistic	Magnitude (dB)	Phase (deg)
95th percentile	0.0214	0.222
RMS	0.0109	0.121

Although the residual figures and statistics provide a convenient visualisation of the Γ -dependent deviation of the measured thru response from the ideal behaviour, they are mainly used here to indicate the presence of systematic measurement errors. In an ideal case, the thru response should remain constant across different load points. Therefore, the appearance of Γ -dependent patterns in the residual figures indicates systematic deviations rather than random measurement noise. Using the ideal thru characteristics together with the measured data, the analysis is extended from a discrete set of residual samples to a Γ -dependent error description. The discrete error values are then interpolated to construct a continuous error model. This interpolation allows the systematic error to be estimated at other Γ locations, enabling the extracted error model to be applied to device measurements in Section 5.3.2, whose load points may not match the original calibration dataset. In the subsequent modelling process, the systematic error is instead represented in the travelling-wave domain as an additive offset term,

$$c(\Gamma) = b_2(\Gamma) - a_1(\Gamma) \quad (5.7)$$

where Γ represents the reflection coefficients corresponding to the measured load points in the calibration dataset. Eq. (5.7) is consistent with the Type-B uncertainty framework, where deterministic offsets are commonly adopted to characterise systematic effects. Since $|a_1|^2$ corresponds to the incident wave power at port 1, a_1 is adopted as the reference for power normalisation. Consequently, deviations from the ideal thru condition are attributed to the outgoing wave b_2 , and the Γ -dependent systematic error is therefore evaluated in the b-wave domain. This formulation enables the influence of stimulus amplitude to be largely removed, facilitating a consistent assessment of systematic measurement errors.

To evaluate the systematic error at specific Γ locations other than the measured load points used in the calibration dataset, inverse-distance weighting (IDW) interpolation is employed. The approach is simple, numerically robust, and does not require global surface fitting. The expected error at a selected Γ location is calculated as a weighted average of nearby measured points, where the weights contribute inversely to the distance in the complex Γ -plane [9]. The IDW interpolation is defined as:

$$\hat{f}(\Gamma) = \frac{\sum_{k=1}^N w_k(\Gamma) f(\Gamma_k)}{\sum_{k=1}^N w_k(\Gamma)}, \quad (5.8)$$

where N is the number of neighbouring measured points and weights w_k are defined as,

$$w_k(\Gamma) = \frac{1}{\|\Gamma - \Gamma_k\|^p}, \quad (5.9)$$

where $\|\Gamma - \Gamma_k\|$ denotes the Euclidean distance in the complex Γ -plane and p is set to 1 so that the influence of surrounding points decreases smoothly with distance, producing a stable interpolation of the error distribution.

5.3.2 GaN HEMT Measurement Results and Error Model Application

The GaN HEMT CGHV1F006 was characterised under large-signal load-pull conditions at a frequency of 2.45 GHz with a drain bias of 35 V and gate bias of -2.8 V. Travelling waves were extracted over 421 load impedances. The Γ -dependent error model obtained from the thru measurement was subsequently applied to the device data to investigate its impact on the measured responses.

Using the interpolated Γ -dependent offset term, the outgoing wave was corrected as

$$B^{corr}(\Gamma) = B^{meas}(\Gamma) - \hat{c}(\Gamma), \quad (5.10)$$

where $\hat{c}(\Gamma)$ denotes the estimated systematic offset evaluated at the device load impedances using IDW interpolation, $B^{corr}(\Gamma)$ is the corrected b -wave, and $B^{meas}(\Gamma)$ is the measured b -wave.

The measured load-pull impedance points of the device on the Γ -plane are shown in Fig. 5.7.

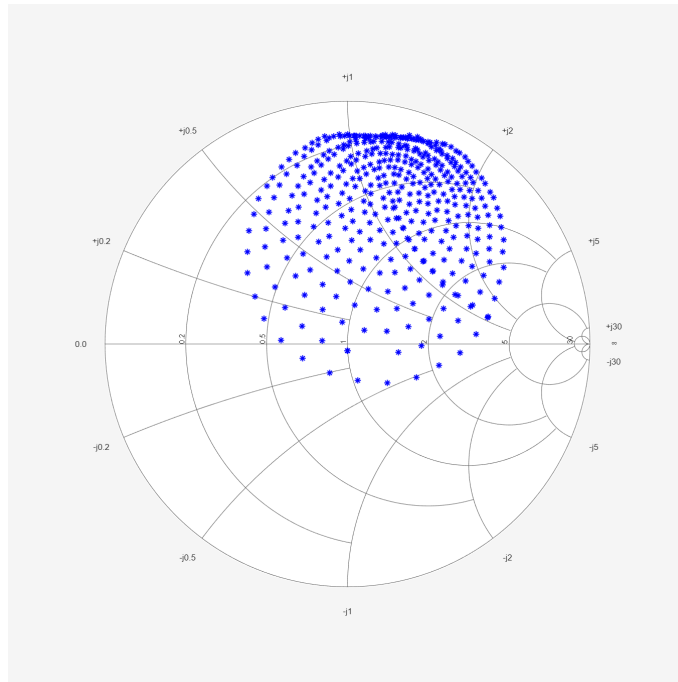


Figure 5.7: Smith chart of the device load points

Based on these measured load points, the device performance contours are interpolated across the measured Γ region. The phase comparison between the measured and corrected travelling waves is not shown, as the difference remains below approximately 1° , indicating negligible phase variation across the measured Γ region.

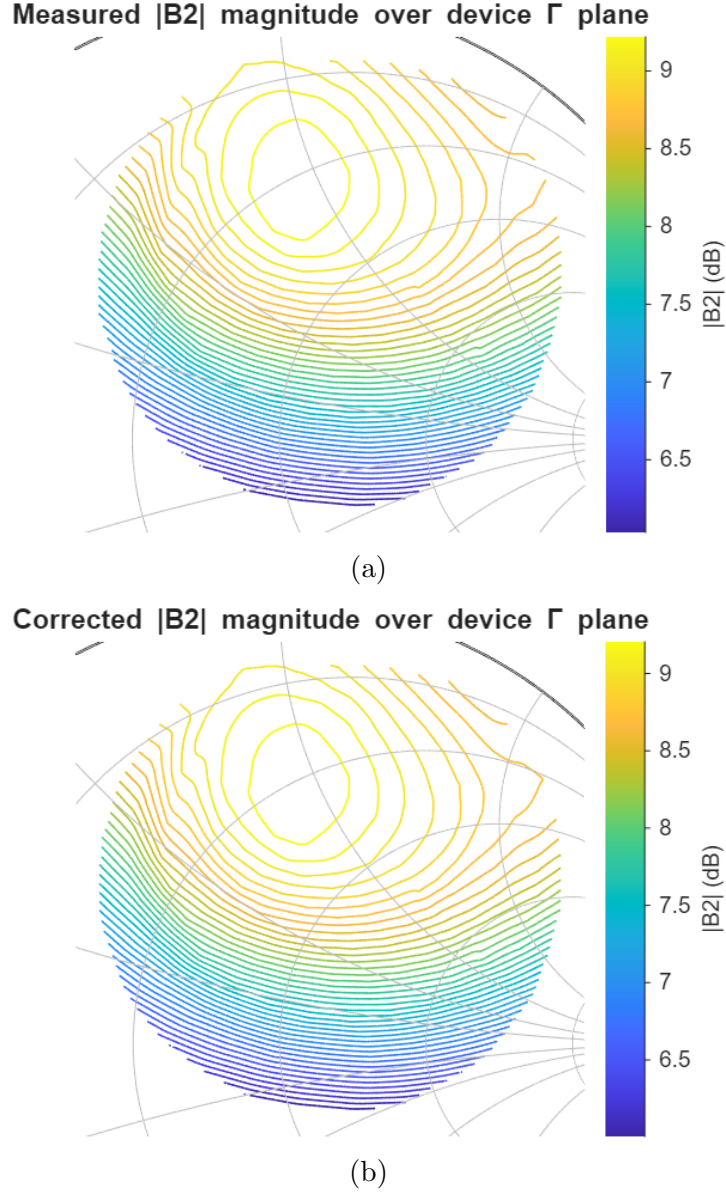


Figure 5.8: Measured b_2 magnitude (dB) over the device Γ plane

The measured $|b_2|$ magnitude displays a clear dependence on the load reflection coefficient Γ , distributed smoothly across the measured impedance area. Comparing the measured and corrected figures, the overall contour structure remains nearly unchanged. The location of the maximum $|b_2|$ region and the general gradient across the impedance area remain consistent after correction. This indicates that

the correction derived from the thru measurement produces only a small modification to the measured travelling-wave magnitude across the device load-pull dataset.

Since the device output power is directly related to the magnitude of the travelling wave b_2 , as shown in Eq. (5.11), the small change observed in the corrected $|b_2|$ distribution predicts that only a minor variation will occur in the output power.

$$P_{out} = |b_2|^2 - |a_2|^2, \quad (5.11)$$

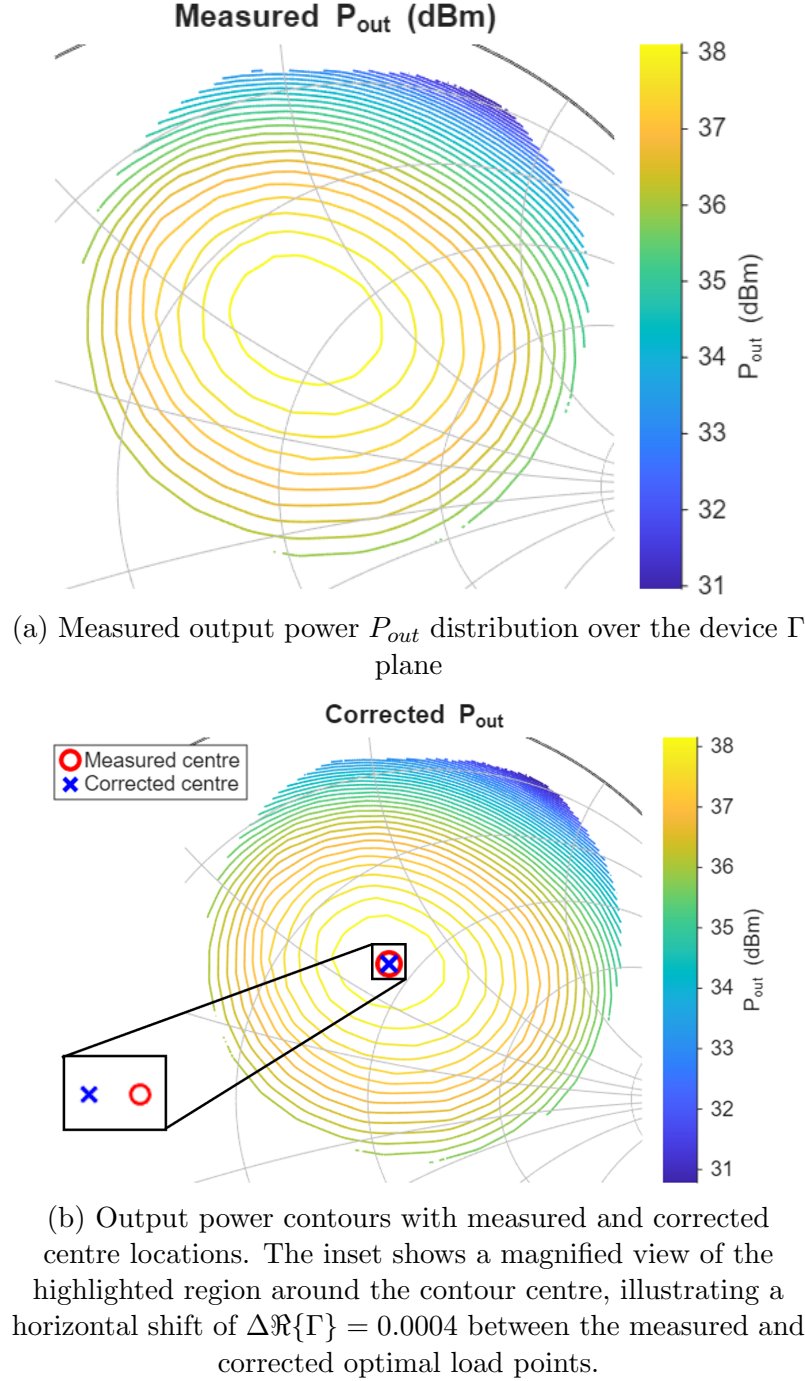


Figure 5.9: Measured and corrected output power P_{out} distribution over the device Γ plane

The measured output power P_{out} also shows a dependence on the load reflection coefficient. A high-power region can be observed within the load-pull measurement area, indicating that the device operates near its optimum power transfer point at the fundamental frequency. The overall contour structure in Fig. 5.9a and 5.9b remains similar across the impedance region. Because the shift between the contour

centres is very small relative to the Smith chart scale, a magnified inset is included to clearly illustrate the difference. However, in Fig. 5.9b, a slight shift of the maximum power region can be observed after applying the correction, indicating that the systematic error derived from the thru measurement introduces a small but measurable modification to the device power distribution.

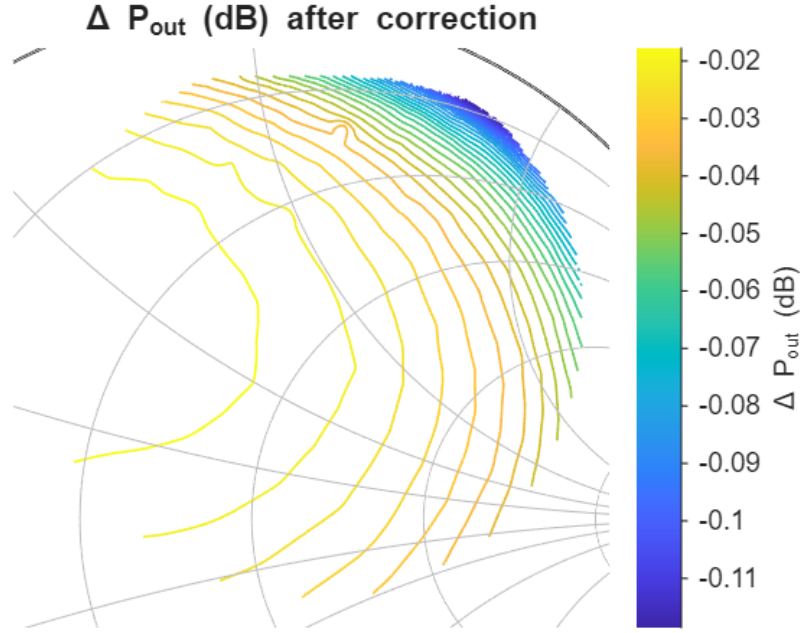


Figure 5.10: Output power difference ΔP_{out} after applying the correction

The difference between the corrected and measured output power is further illustrated in Fig. 5.10. Across most of the impedance region, the power deviation remains below approximately 0.1 dB, indicating that the systematic error correction has a small impact on the delivered power. These results indicate that the influence of the Γ -dependent systematic measurement error on the device output power is below approximately 0.1 dB over most of the tested load-pull region.

5.3.3 Error Effect Analysis for Cardiff Model Coefficients

Since the coefficients are extracted directly from the travelling waves, changes in the measured $|b_2|$ are propagated into the extracted coefficients.

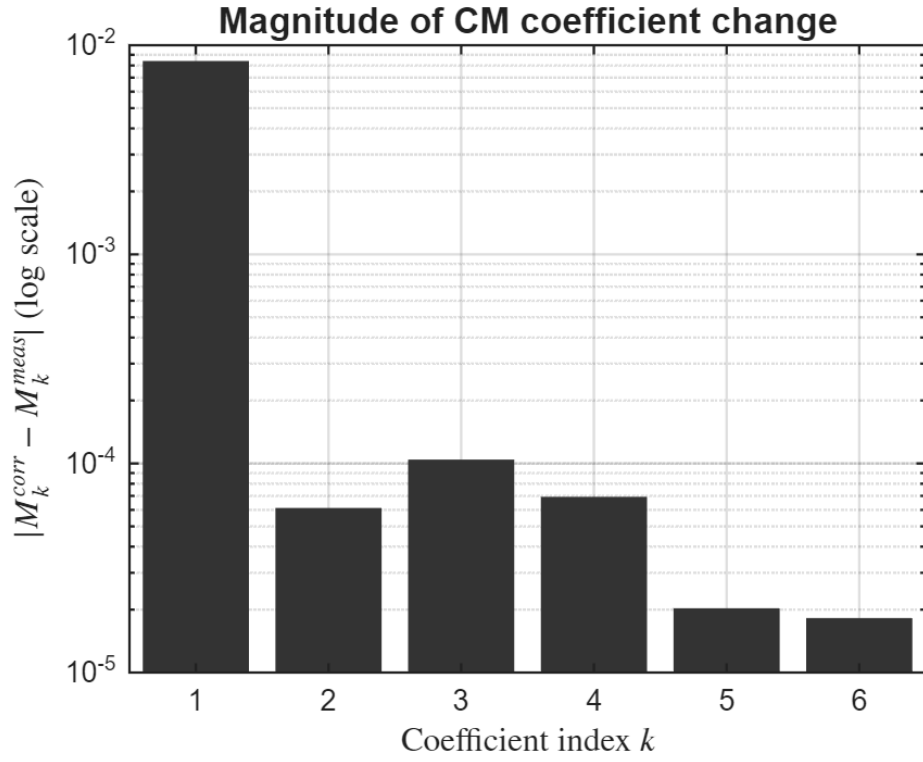
In this section, the analysis focuses on the fundamental-frequency b_2 wave and the first six CM coefficients listed in Table 2.3. From Section 5.3.2, the measured

device b_2 and corrected b_2 are obtained. With the measured a_2 , the CM coefficients can be extracted using the following equation:

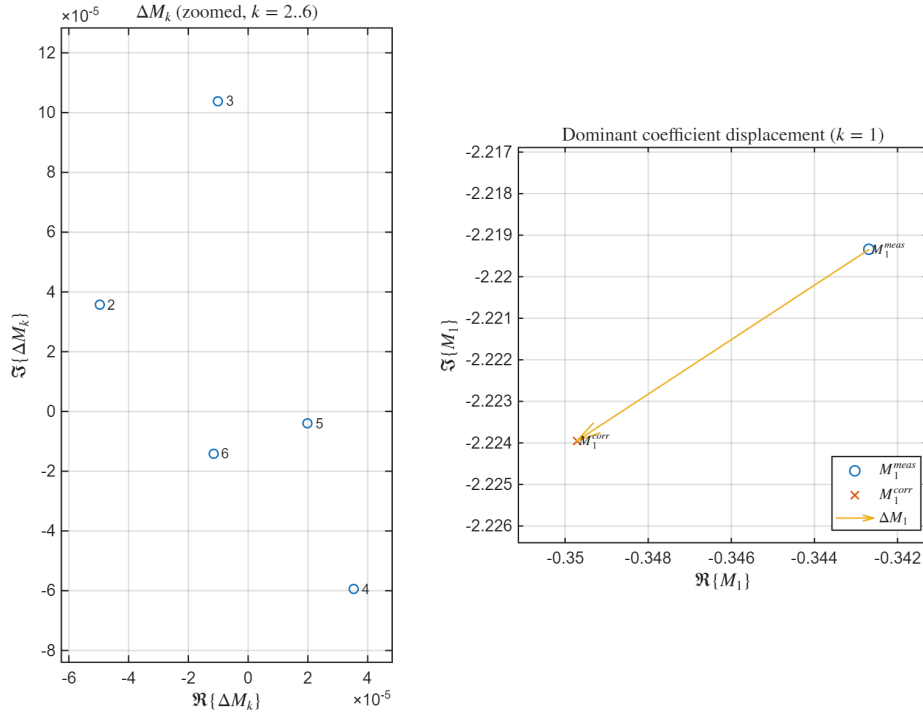
$$[M_{p,h,m,n}] = ([A]^H[A])^{-1} [A]^H[B_{p,h}]. \quad (5.12)$$

Thus, two different CMs with the same structure are obtained, with coefficients extracted from the measured data and the corrected data. From this, a delta coefficient model can be extracted:

$$\begin{aligned} [\Delta M_{p,h,m,n}] &= ([A]^H[A])^{-1} [A]^H([B_{p,h}^{corr}] - [B_{p,h}^{meas}]) \\ &= ([A]^H[A])^{-1} [A]^H[\Delta B_{p,h}] \\ &= ([A]^H[A])^{-1} [A]^H \hat{c}(\Gamma^{meas}) \end{aligned} \quad (5.13)$$



(a) Magnitude of the change in CM coefficients, $|M_k^{corr} - M_k^{meas}|$, shown on a logarithmic scale for all extracted coefficients



(b) Effect of the proposed Γ -dependent correction on the extracted CM coefficients in the complex plane. (left) View of the coefficient differences ΔM_k for $k = 2, \dots, 6$, (right) Displacement of the coefficient ($k = 1$) from the measured to the corrected value

Figure 5.11: Effect of the proposed Γ -dependent correction on the extracted CM coefficients

Figure 5.11 illustrates the effect of the measured data correction on the CM coefficients. The variations among the CM coefficients are not uniform. As shown in Fig. 5.11a, the first coefficient displays a larger variation than the other coefficients. In contrast, the remaining coefficients show variations that are orders of magnitude smaller, indicating that the correction primarily influences the first coefficient of the behavioural model.

Figure 5.11b further presents the coefficient differences in the complex plane. The first coefficient displays a clear and directed shift, while the shift of the others is comparatively minor and on the order of 10^{-5} . Although these coefficient changes are physically interpretable, their impact on fitting accuracy is suggested to be limited. This behaviour indicates that CM prediction accuracy alone cannot indicate whether systematic measurement errors are present, since the model is fitted directly to the measured dataset. The resulting NMSE and RMS statistics are therefore examined next to assess the consistency of the model before and after correction.

Table 5.2: CM prediction accuracy before and after Γ -dependent correction

Metric	CM from Measured data	Corrected CM
NMSE (dB)	-42.03	-42.06
RMS $ b_2 $ (dB)	0.0530	0.0529

Table 5.2 summarises the prediction accuracy of the CM extracted from the measured and corrected datasets, evaluated using the NMSE and RMS metrics. The differences between the two cases are minimal, which indicates that after applying the error model, the accuracy of the CM remains stable. At the same time, the results also indirectly reflect that some traditional evaluation metrics cannot fully characterise the impact of systematic uncertainties on model accuracy. This behaviour is consistent with large-signal behavioural modelling theory, while conventional accuracy metrics such as NMSE may not fully reflect these effects. [10], [11], [12].

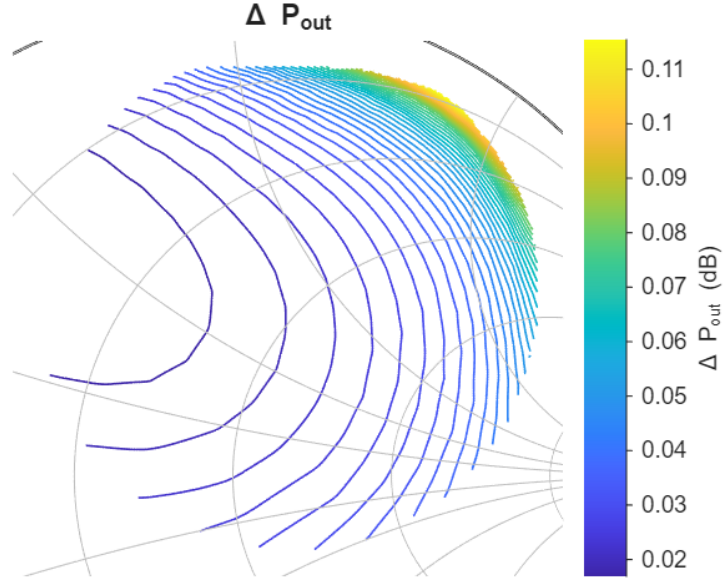
Although the coefficient-domain comparison displays variations in the extracted CM coefficients, PA design is typically based on predicted performance metrics such

as output power and efficiency. Therefore, the two extracted models are further compared using PA-related metrics to evaluate whether the observed coefficient differences result in meaningful changes in predicted device performance.

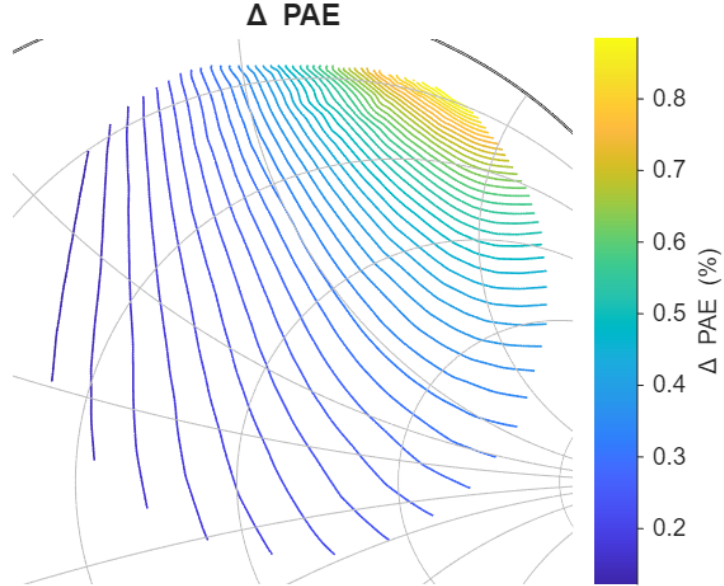
Firstly, the difference in the output power is calculated as

$$\Delta P_{out} = P_{out}^{corr-model} - P_{out}^{meas-model}, \quad (5.14)$$

where $P_{out}^{corr-model}$ is calculated using Eq. (5.11) with the b_2 wave predicted from the corrected CM, $P_{out}^{meas-model}$ is obtained from the CM extracted from the measured b_2 wave. Fig. 5.12 shows ΔP_{out} .



(a) Output power difference ΔP_{out}



(b) Power-added efficiency difference ΔPAE

Figure 5.12: Predicted performance difference between the two extracted models

Fig. 5.12 compares the predicted performance difference between the two extracted models in terms of output power and power-added efficiency (PAE). The deviation in predicted output power remains small, with a maximum deviation of about 0.12 dB. The largest differences appear near the edge of the measured Γ region, while the high-power region remains essentially unchanged. The efficiency deviation reaches a maximum of approximately 0.91%. However, the overall contour structure remains similar for both models, and the load condition corresponding to

peak performance is unchanged. These results indicate that although the extracted CM coefficients differ after applying the correction, the predicted PA performance contours change only slightly.

5.4 Calibration and Model Integrity Metrics

In Section 5.3, it was shown that Type B systematic errors may still exist even when the NMSE remains low, as behavioural models are capable of absorbing systematic errors without the user being aware that they affect the model outputs. Although the proposed error model enables the identification of residual systematic errors, it does not provide a quantitative criterion for evaluating the overall integrity of the measurement system. Therefore, a statistical metric is required to assess the reliability of the calibration procedure, the measurement system, and the measurement instruments.

In this section, two complementary approaches are introduced to establish quantitative thresholds: one based on residual statistics from the thru measurement (measurement-domain perspective); the other, based on the stability of the CM coefficients (model-domain perspective).

Even after calibration, the measurement system exhibits residual imperfections arising from VNA limitations, coupler imbalance, cable losses, and other related effects. Therefore, before calculating and defining a threshold for measurement-domain metrics, it is necessary to assess how much variation exists in the calibrated measurement system under normal operating conditions so that a proper reference baseline can be determined. Here, the baseline refers to the magnitude and phase residual statistics obtained from the calibrated thru measurements, against which abnormal behaviour can be identified.

Accordingly, the RMS residuals of the magnitude and phase obtained from the thru dataset (Table 5.1) are taken as the baseline values representing the normal operating condition of the calibrated measurement system, yielding $\mu_{mag} = 0.0109$ dB

and $\mu_{pha} = 0.121^\circ$. These baseline values define the centre of the residual statistics under normal calibrated conditions. To evaluate the statistical variation of these metrics, bootstrap resampling of the measured load points is applied [13]. The resulting standard deviation σ quantifies the expected change of the residual statistics due to finite sampling and numerical conditioning effects. Finally, the control threshold for each metric is defined as

$$\tau = \mu + 3\sigma, \quad (5.15)$$

where τ represents the upper bound of expected behaviour under stable calibrated conditions.

To evaluate the sensitivity of the proposed integrity metric to systematic measurement deviation, a controlled distortion is added to the measured travelling-wave data. For the measured b_2 wave, the added distortion can be defined as:

$$\Delta b_2 = \alpha \cdot a_{1,med} \cdot e^{j\phi_0}, \quad (5.16)$$

where α is the scaling factor, $a_{1,med}$ is the median magnitude of the incident wave and ϕ_0 defines a constant bias direction. Thus, the modified measured wave becomes:

$$b_2^{mod} = b_2^{meas} + \Delta b_2, \quad (5.17)$$

and the modified travelling wave transmission is calculated as

$$H^{mod} = \frac{b_2^{mod}}{a_1}. \quad (5.18)$$

The offset Δb_2 is constant for all load points. The normalization by $a_{1,med}$ ensures

that the modified travelling-wave magnitude is defined relative to the magnitude of the incident wave.

This test iterates the scaling factor within a selected interval to calculate the RMS and the 95th percentile for the modified travelling wave and evaluates them using the integrity index:

$$J_{thru} = \max \left(\frac{\text{RMS}_{mag}}{\tau_{mag}}, \frac{\text{P95}_{mag}}{\tau_{mag}}, \frac{\text{RMS}_{\phi}}{\tau_{\phi}}, \frac{\text{P95}_{\phi}}{\tau_{\phi}} \right) \quad (5.19)$$

where τ is defined in Eq. (5.15). The threshold corresponds to the condition $J_{thru} > 1$, which means:

$$\alpha_{th} = \min (\alpha : J_{thru}(\alpha) \geq 1) . \quad (5.20)$$

Figure 5.13 displays the result of the added offset test.

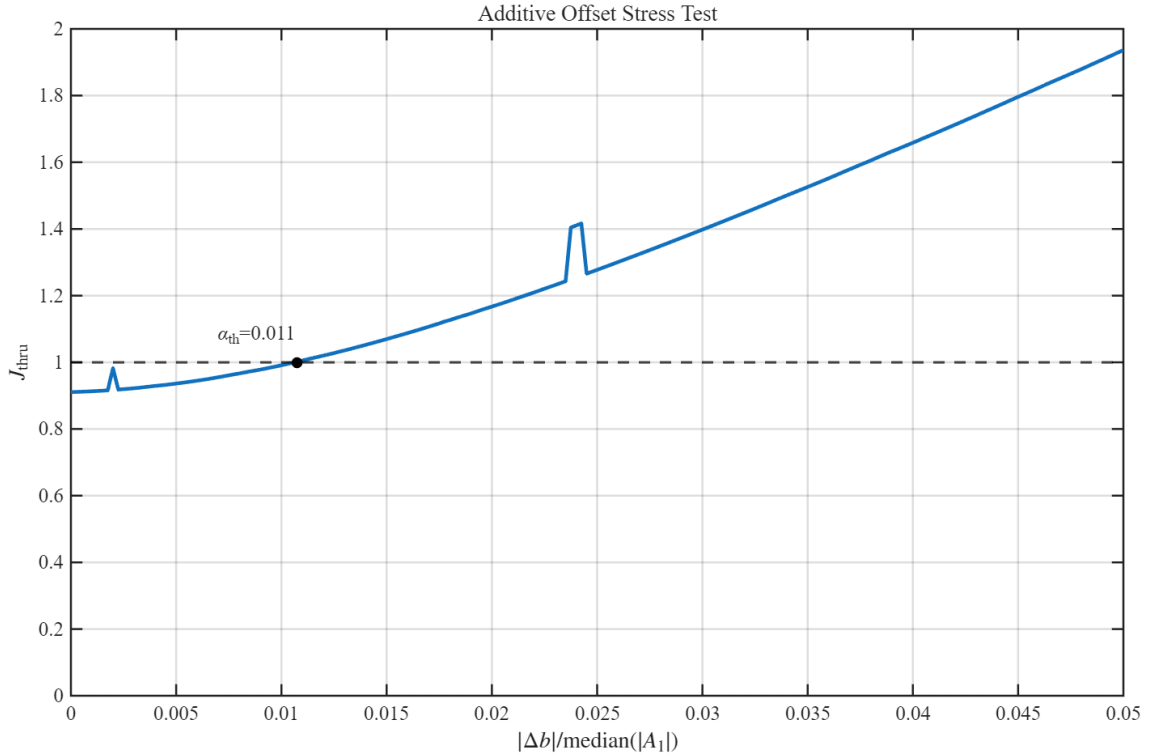


Figure 5.13: Additive offset stress test with 3σ statistical threshold

The x-axis represents the scaling factor α obtained from the variation of Eq.

(5.16):

$$|\Delta b_2| = \alpha \cdot a_{1,med} \quad (5.21)$$

The Figure shows that the integrity metric increases monotonically with the added distortion. The intersection in this figure indicates that a distortion of around 1–2% of the incident-wave magnitude is sufficient to exceed the 3σ threshold derived from the baseline residual statistics. This result demonstrates that the proposed integrity metric is sensitive to small systematic deviations in the measurement system.

The previous integrity metric evaluates the distortion in the measured travelling-wave data from a measurement-domain perspective. However, even when the residual statistics remain within acceptable ranges, systematic distortion can still propagate into the extracted behavioural model coefficients. Therefore, a model-domain evaluation is required to quantify the stability of the CM coefficients under systematic distortion.

Under normal calibration conditions, the extracted CM coefficients of the corrected measured data are denoted as M_{ref} . Using the same distortion modification method as shown in Eq. (5.16), a new modified CM model is extracted from b_2^{mod} .

The deviation metric for the CM coefficients is defined as the CDI, calculated as

$$CDI(\alpha) = \sqrt{\sum_k \left(\frac{M_k(\alpha) - M_{k,ref}}{|M_{k,ref}|} \right)^2} \quad (5.22)$$

To prevent large coefficients from dominating the result, the CDI normalises the coefficients before calculating the Euclidean distance in the coefficient space. It quantifies the relative deviation of the extracted coefficient vector from the reference.

The baseline μ and the standard deviation σ are generated using a bootstrap resampling algorithm at $\alpha = 0$ from the corrected measured data. This ensures the 3σ threshold reflects the true coefficient deviation rather than the added systematic

distortion. The thresholds are determined by

$$\alpha_{th} = \min(\alpha : \text{CDI}(\alpha) \geq \tau) \quad (5.23)$$

The test result is shown in Fig. 5.14.

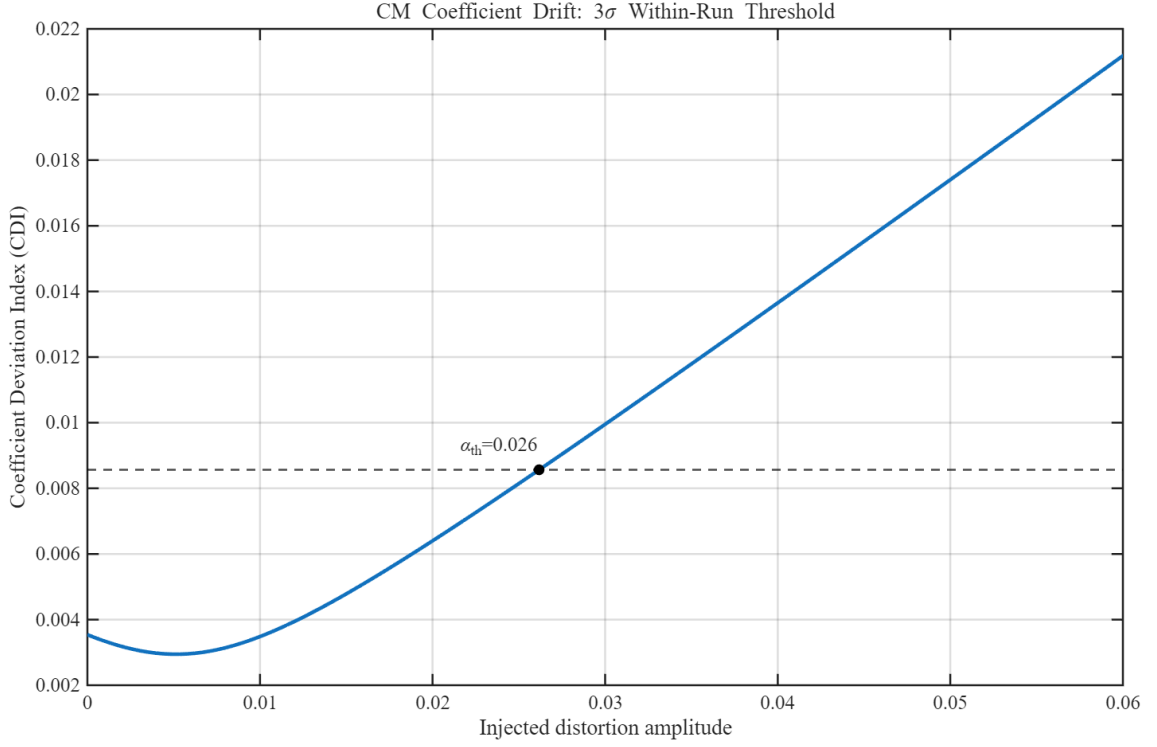


Figure 5.14: CM coefficient drift vs injected error

In this figure, the extracted CM coefficients respond proportionally to systematic distortions in the travelling-wave data. Additionally, the slight decrease near $\alpha = 0$ is caused by numerical conditioning effects in the least-squares solution. The dashed horizontal line represents the 3σ threshold under the same calibrated measurement conditions, and the intersection point defines the minimum α for which $\text{CDI}(\alpha) \geq \tau$, which is around 0.026. This indicates that a distortion of approximately 2.6% of the magnitude of the normalised incident wave is sufficient to cause coefficient changes beyond the expected variations under normal calibrated conditions. Compared with the measurement-domain metric, the CDI directly evaluates the stability of the extracted behavioural model coefficients. Figure 5.14 shows how the CM coefficients change when global distortion is applied to the measured travelling-wave dataset.

The coefficients remain relatively stable at small distortion levels, but noticeable deviations appear when the distortion exceeds approximately 2–3%.

The above analysis establishes a threshold for behavioural model coefficient extraction deviation. It quantifies the sensitivity of the extracted coefficients to systematic distortions.

5.5 Validation Using Designer Metrics

Section 5.4 introduces improved metrics and thresholds for measurement system integrity assessment. To evaluate the practical significance of the integrity threshold, controlled systematic distortions were added to the travelling-wave dataset. The resulting changes in output power were analysed. This allows the thresholds to be interpreted in terms of their practical impact on real PA designs. A similar analysis was also conducted for efficiency-related metrics, which displayed negligible variation (approximately 0.0008%) under the applied distortion levels. Therefore, only the output power contours are presented here for clarity. In addition, this section discusses how the proposed integrity metric may be applied in a practical measurement workflow for assessing abnormal system behaviour.

5.5.1 Global Distortion

First, a global offset distortion was added to the output travelling wave according to Eq. (5.16). Two distortion magnitudes were simulated: a level below the threshold ($0.5\alpha_{th}$) and a level above the threshold ($2\alpha_{th}$).

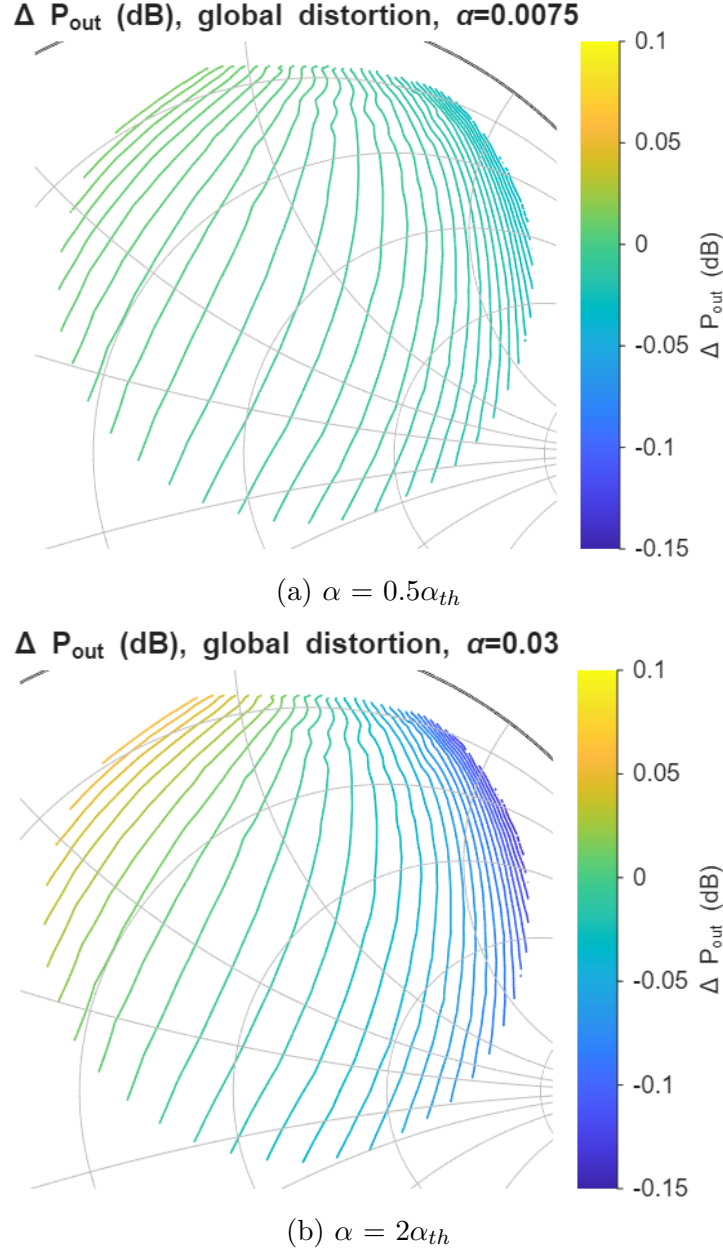


Figure 5.15: ΔP_{out} (dB) over the Γ plane for global distortion

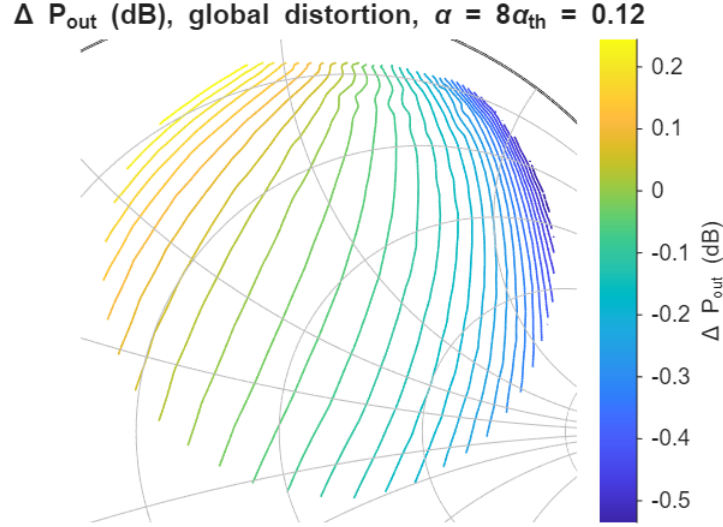
Fig. 5.15a shows the output power difference for the $0.5 \alpha_{th}$ distortion level. The added distortion results in a nearly uniform shift and keeps the contour structure unchanged. The deviation ranges from approximately -0.03 to 0.01 dB. Fig. 5.15b shows the output power difference with a distortion level equal to twice the threshold. The deviations remain relatively uniform, but they increase to approximately 0.1 dB for 95% of the data points. The global offset mainly shifts the overall power level, and only limited spatial distortion is observed.

These results give a direct explanation of the derived integrity threshold. When

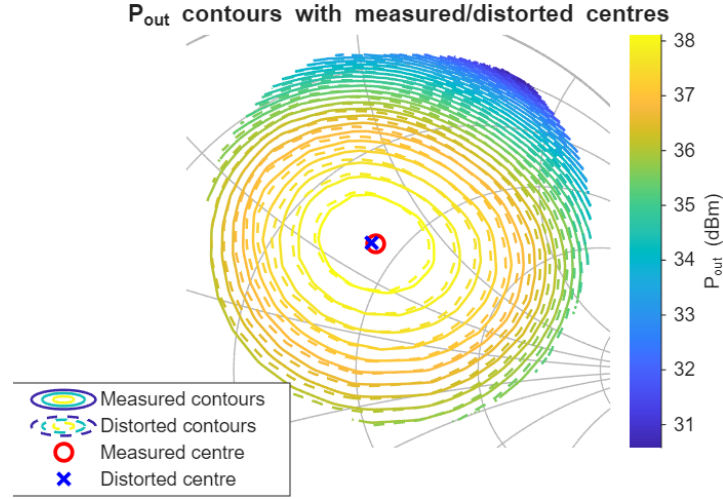
the distortion level remains below the threshold, the change in output power is small and slightly affects the contour structure or the optimal load region. In contrast, when the distortion exceeds the threshold, the deviation becomes clearly observable, with increased variation in output power that may begin to influence the accuracy of design decisions. This shows that the threshold is not only a statistical value. It also indicates that the output power contours start to change in a way that may affect practical design.

5.5.2 Behaviour Under Large Systematic Distortion

Section 5.5.1 investigated distortion levels close to the derived integrity threshold. Although distortions at $0.5\alpha_{th}$ and $2\alpha_{th}$ affect the predicted power distribution slightly, it is useful to test the behaviour of the measurement system under larger systematic distortion. Therefore, an additional test was performed by applying a global distortion with magnitude $\alpha = 8\alpha_{th}$. Although such a distortion level may not represent normal calibrated measurement conditions, it serves as a stress test to illustrate how larger systematic errors propagate through the travelling-wave dataset and affect the final load-pull contours.



(a) Output power deviation under global distortion with magnitude $\alpha = 8\alpha_{th}$



(b) Comparison of output power contours between the measured dataset and the dataset distorted by $\alpha = 8\alpha_{th}$. Solid contours correspond to the measured results, while dashed contours represent the distorted dataset. The markers indicate the centres of the highest closed contour regions, illustrating the shift of the optimal load region.

Figure 5.16: Output power deviation and contour shift under global distortion with magnitude $\alpha = 8\alpha_{th}$

The power deviation across the Γ plane is shown in Fig. 5.16a. Compared with the distortion levels tested in Section 5.5.1, larger deviations are observed across the measured impedance region, with the maximum power difference exceeding 0.5 dB. The deviation is not uniformly distributed and becomes more obvious near the edge of the load-pull region. This behaviour reflects the increased sensitivity of the

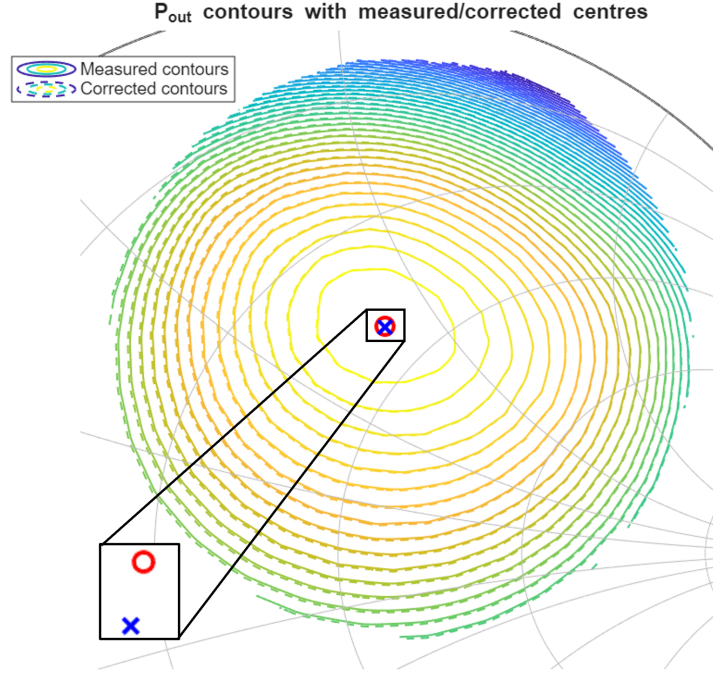
predicted output power to systematic distortion when the load impedance reaches the boundary of the measured impedance region.

The corresponding output power contour comparison between measured data and distorted data is presented in Fig. 5.16b. Although the overall contour structure remains similar, an obvious shift of the maximum power region can be observed when the distortion is applied. The centres of the maximum power contours show a slight shift, indicating that systematic distortion may change the predicted optimal load impedance. This behaviour is particularly important for PA design, where the optimal impedance location is directly used for matching network design.

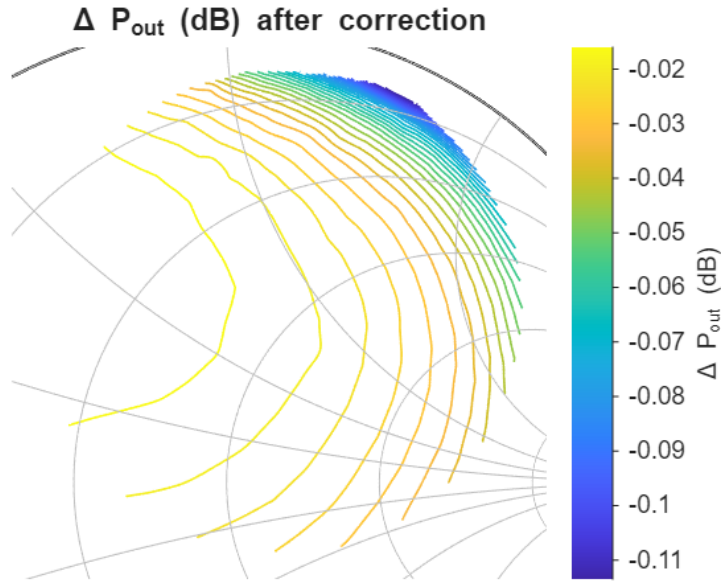
These observations indicate that sufficiently large systematic distortions can change both the power distribution and the optimal load region obtained from load-pull measurements, leading to incorrect estimation of the optimal load impedance. Although such large distortion levels are unlikely to appear under normal calibrated conditions, this stress-test scenario demonstrates the impact of severe systematic errors. In practice, the proposed integrity metric enables early detection of systematic deviations at much lower distortion levels, before visible contour shifts occur. Consequently, the metric provides a practical diagnostic tool for identifying potential calibration issues that may affect measurement system reliability and design decisions.

5.5.3 Validation at an Additional Input Power Level

To verify that the proposed measurement-domain threshold is not restricted to a specific operating condition, the same analysis is repeated using an additional dataset measured at a different source power level. Since the operating conditions are different, this comparison is used to demonstrate that the proposed integrity metric remains applicable under different excitation levels, rather than evaluating calibration drift.



(a) Comparison of measured and corrected output power contours. Solid contours represent the measured dataset, while dashed contours correspond to the corrected results. The inset shows a magnified view of the contour centre region, highlighting a shift of $\Delta\Gamma = 0.0012$ in the optimal load location.



(b) Difference in output power between corrected and measured data for the dataset measured at a different power level

Figure 5.17: Comparison of measured and corrected output power for the dataset measured at a different source power level

Fig. 5.17 displays the difference in output power between the measured data and

the corrected data for the validation measurement dataset. The inset highlights the region around the contour centre, where the small shift between the measured and corrected optimal load points becomes visible. The difference between the corrected and measured output power remains small across the same impedance region shown in Section 5.3.2, generally below approximately 0.1 dB. A small shift of the maximum power region is observed in Fig. 5.17a, showing a similar pattern at different power levels. Based on this comparison, Fig. 5.20 displays the measurement-domain threshold validation for this measurement condition.

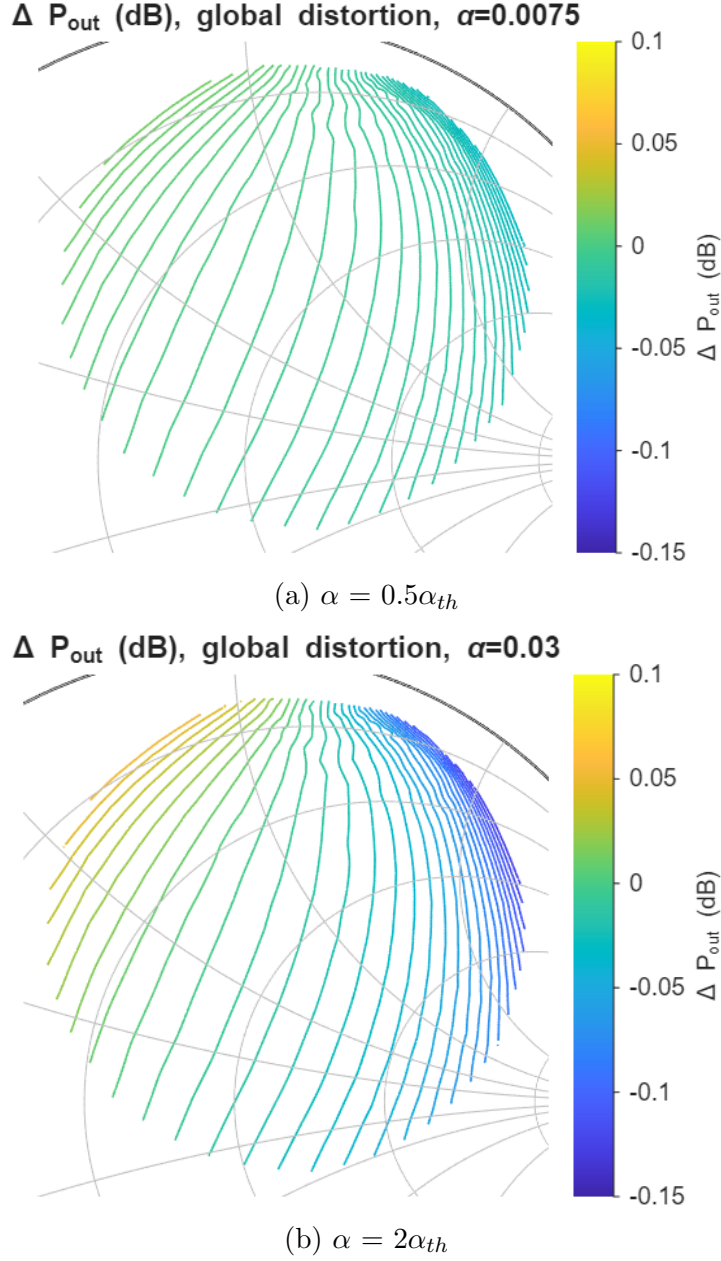


Figure 5.18: ΔP_{out} (dB) over the Γ plane for global distortion at another input power level

In Fig. 5.18a, the distortion is scaled to 0.5 times the derived threshold for this operating condition. The output power deviation remains small across the load-pull region, and the overall contour distribution keeps a structure similar to that of the corrected dataset. This indicates that distortions below the derived threshold propagate only minor modifications to the travelling-wave data and do not significantly affect the power distribution behaviour. Compared with the previous case, larger deviations in output power are observed across the impedance region when the dis-

tortion is increased to twice the threshold. The noticeable changes in Fig. 5.18b, reaching approximately ± 0.15 dB, are about three times larger than those in the previous case and begin to significantly influence the measured travelling-wave data.

Distortions below the derived threshold generate only minor differences in the output power, while distortions exceeding the threshold result in noticeable changes. This observation supports the validity of the proposed measurement-domain threshold for self-assessing measurement systems.

These results indicate that the proposed threshold shows consistent behaviour under different input power conditions. However, for practical drift detection, comparisons should be performed under the same operating conditions to ensure that any observed deviation is caused by measurement system changes rather than differences in device excitation.

5.5.4 Practical Implementation of the Integrity Metric

The previous sections demonstrated the behaviour of the proposed integrity metrics under controlled systematic distortions and different operating conditions. To apply these results to practical measurements, it is necessary to explain how the proposed method can be applied within a typical load-pull measurement workflow.

In load-pull measurements, the integrity assessment begins with a calibrated measurement system. As part of calibration verification, a reference dataset can be obtained by measuring a standard thru. From the thru measurement, baseline residual statistics are obtained, and the corresponding threshold is calculated using the proposed 3σ -based metric.

During following measurements, verification can be performed at selected stages of the measurement process by measuring the selected reference load points again. Then the measured travelling-wave data are evaluated using the same integrity metric, and the calculated value is compared with the previously established threshold. If the evaluated metric remains below the threshold, the measurement system is considered to operate within the expected range of systematic variation. If the

threshold is exceeded, this indicates the existence of an abnormal deviation in the measurement system, and further investigation or recalibration may be required. The overall workflow is shown in Fig. 5.19.

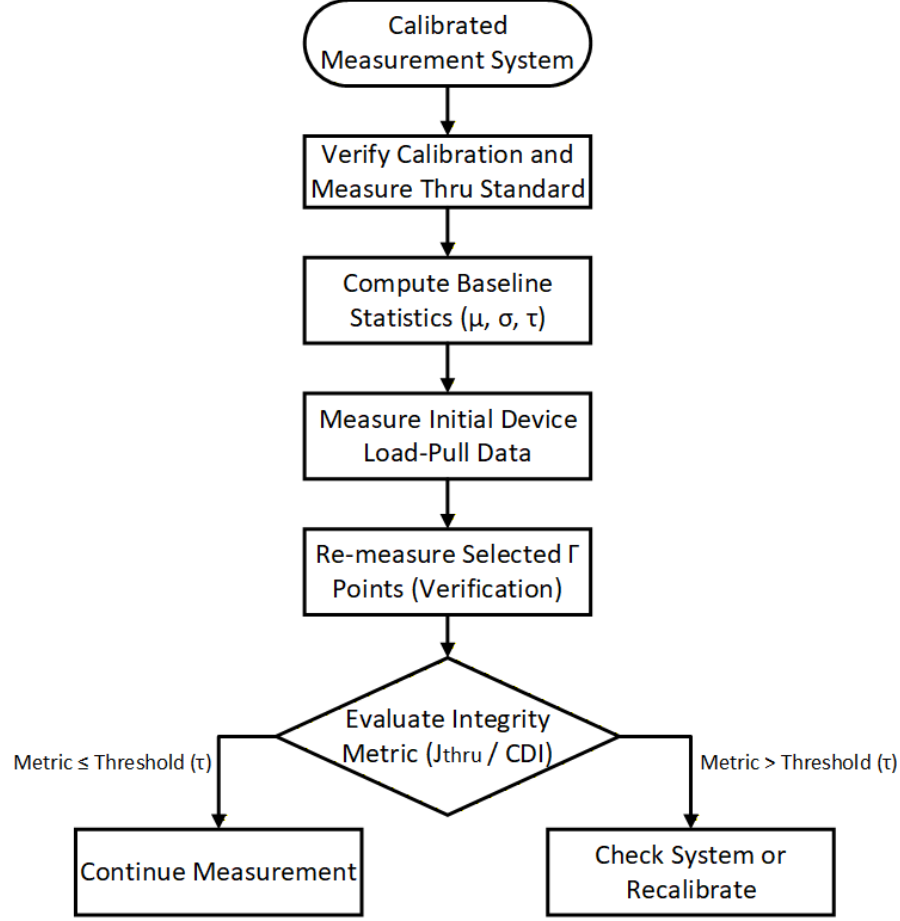


Figure 5.19: Workflow for practical implementation of the proposed measurement system integrity assessment method

In real applications, repeating a full load-pull measurement for verification may consume too much time. Therefore, a reduced set of representative load points can be selected to perform integrity verifications more efficiently. These points may include regions near the centre of the Smith chart, regions near the maximum power, and selected boundary points. This approach allows the system integrity to be assessed with reduced measurement time while the main characteristics of the Γ -dependent response can also be captured.

The proposed integrity metric is used as an early detection tool to identify systematic deviations before they cause visible distortion in power contours or shifts

in the optimal load region. However, this method assumes that the device under test remains unchanged during the comparison. Changes in device behaviour may also affect the measured data, so these effects should be considered separately when analysing the results.

Based on this workflow, the method can be used to assess measurement system integrity during load-pull measurements. It provides a practical approach and helps improve the accuracy of behavioural modelling and power amplifier design.

5.6 Conclusion

This chapter investigated the effects of systematic (Type B) uncertainties in a calibrated load-pull measurement system. A Γ -dependent error model was extracted from a calibrated thru standard measurement and represented as an additive travelling-wave offset. Then this error model was applied to device measurements to analyse its impact on output power and behavioural model extraction.

The results show that the extracted systematic uncertainties have only a small effect on output power contours and have negligible impact on conventional statistical metrics such as NMSE. However, small changes are observed in the travelling-wave data and the extracted model coefficients, indicating that systematic errors may still exist even when traditional performance metrics are acceptable.

In order to solve this problem, two integrity assessment methods were proposed. The first is a measurement-domain metric based on residual statistics from the thru standard, which is used to define a threshold using 3σ bootstrap resampling. The second is a model-domain metric, called the CDI, which evaluates how the model coefficients change under different levels of systematic distortion. These two metrics connect calibration uncertainty with model stability.

The proposed method provides a practical tool for assessing measurement system integrity during load-pull measurements. By defining a threshold for acceptable systematic distortion, it allows early detection of abnormal deviations before visible changes appear in power contours or the optimal load region. The workflow shown

in this chapter demonstrates how the proposed metric can be used in a calibration and measurement routine to improve behavioural model extraction and reduce unnecessary re-measurement during power amplifier design.

The validation in this chapter is based on controlled distortion tests rather than repeated measurements under real operating conditions. The method also assumes that the device under test remains unchanged during comparison. Further work is required to validate the method under practical measurement drift conditions and to extend the analysis to a wider range of devices and operating scenarios.

References

- [1] P. J. Tasker and J. Benedikt, “Waveform Inspired Models and the Harmonic Balance Emulator,” *IEEE Microwave Magazine*, vol. 12, no. 2, pp. 38–54, 2011. DOI: 10.1109/MMM.2010.940101.
- [2] S. Woodington, R. Saini, D. Williams, J. Lees, J. Benedikt, and P. J. Tasker, “Behavioral model analysis of active harmonic load-pull measurements,” in *2010 IEEE MTT-S International Microwave Symposium*, 2010, pp. 1688–1691. DOI: 10.1109/MWSYM.2010.5517261.
- [3] J. J. Bell et al., “Behavioral model analysis using simultaneous active fundamental load-pull and harmonic source-pull measurements at X-band,” in *2011 IEEE MTT-S International Microwave Symposium*, 2011, pp. 1–4. DOI: 10.1109/MWSYM.2011.5972803.
- [4] JCGM, *Evaluation of Measurement Data — Guide to the Expression of Uncertainty in Measurement*. Joint Committee for Guides in Metrology, 2008.
- [5] N. M. Ridler and M. J. Salter, “Evaluation of uncertainty in vector network analyzer measurements,” *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 1, pp. 45–50, 2002.
- [6] G. Engen and C. Hoer, “Thru-Reflect-Line: An Improved Technique for Calibrating the Dual Six-Port Automatic Network Analyzer,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 27, no. 12, pp. 987–993, 1979. DOI: 10.1109/TMTT.1979.1129778.
- [7] R. B. Marks and D. F. Williams, “A General Waveguide Circuit Theory,” *Journal of Research of the National Institute of Standards and Technology*, vol. 97, no. 5, pp. 533–562, 1992.
- [8] P. Colantonio, F. Giannini, E. Limiti, and V. Teppati, “An approach to harmonic load- and source-pull measurements for high-efficiency PA design,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 1, pp. 191–198, 2004. DOI: 10.1109/TMTT.2003.821276.

-
- [9] R. Franke and G. Nielson, "Smooth interpolation of large sets of scattered data," *International Journal for Numerical Methods in Engineering*, vol. 15, no. 11, pp. 1691–1704, 1980. DOI: <https://doi.org/10.1002/nme.1620151110>.
- [10] Verspecht and D. Root, "Polyharmonic distortion modeling," *IEEE Microwave Magazine*, vol. 7, no. 3, pp. 44–57, 2006. DOI: [10.1109/MMW.2006.1638289](https://doi.org/10.1109/MMW.2006.1638289).
- [11] D. Wisell, M. Isaksson, and N. Keskitalo, "A general evaluation criteria for behavioral power amplifier modeling," in *2007 69th ARFTG Conference*, 2007, pp. 1–5. DOI: [10.1109/ARFTG.2007.5456311](https://doi.org/10.1109/ARFTG.2007.5456311).
- [12] P. Landin, M. Isaksson, and P. Handel, "Comparison of evaluation criteria for power amplifier behavioral modeling," in *2008 IEEE MTT-S International Microwave Symposium Digest*, 2008, pp. 1441–1444. DOI: [10.1109/MWSYM.2008.4633050](https://doi.org/10.1109/MWSYM.2008.4633050).
- [13] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. New York: Chapman and Hall, 1993.

Chapter 6

Conclusion and Future Work

This thesis has investigated behavioural modelling of RF power amplifiers from three aspects: including schematic implementation, runtime efficiency in CAD environments, and the propagation of systematic uncertainties into behavioural model coefficients. Usually, behavioural models are evaluated through statistical assessment metrics such as NMSE and RMS error. This work shows that behavioural model stability also depends on both implementation efficiency and measurement system integrity.

The thesis first introduces the implementation of an ANN model in a CAD environment. The implemented schematic achieves several functions, including the flexible structure and automatic schematic generation. Then, it simulates the runtime efficiency of the ANN model and (CM) implementations under fundamental frequency for a dataset that has a large number of load points. The results show the influence of schematic structure and model coefficients on computational efficiency. Finally, the research investigates systematic (Type B) uncertainties in calibrated load-pull measurement systems and their effect on extracted CM coefficients. The derived thresholds for both calibration standard measurements and model coefficients are developed to detect hidden systematic deviations that remain under acceptable traditional statistical assessment metrics.

This chapter concludes the main findings and contributions of the thesis and summarises the directions for future research.

6.1 Conclusion

Chapter 3 presented the implementation of an ANN model in the ADS schematic. By using AEL code, the schematic structure was automatically generated, including travelling-waves calculations, parameter importing, normalization, and ANN calculation equations. The approach bypassed the limitation of ADS in matrix calculations by expanding neural network operations into direct calculation equation form using for-loop generation.

The implementation procedure provides several contributions. First, it builds a basic framework script to generate an ANN model schematic. This script enables flexible modification of hidden layer neuron numbers without manual adjustment. The comparison between the verification and the simulated device data confirms that the ANN model achieves acceptable predictive performance (NMSE below -40 dB) [1], [2], indicating its suitability for non-linear behavioural modelling tasks.

This contribution provides a foundation for comparison between ANN and CM implementations under controlled CAD conditions.

Chapter 4 investigated the runtime efficiency of ANN and CM implementations under different sizes of load-pull simulation datasets in ADS. The study showed that runtime efficiency is influenced not only by model formulation but also by schematic structure and redundant mathematical calculations.

The original template displayed an extremely long simulation time due to redundant harmonic calculations and unnecessary exponentiation operations under the fundamental frequency. By removing zero-power terms and simplifying the Amatrix calculation, a simplified CM structure was obtained. Under the 72,000-point fundamental-frequency load-pull condition considered in this work, this reduced the runtime by approximately 99% relative to the original CM implementation.

The coefficient analysis showed that the ANN runtime is approximately linearly related to the number of hidden neurons because the changes are uniform and only appear in $\tanh()$ calculations for different hidden neurons. In contrast, CM runtime

shows stepwise increases related to the higher-order exponentiation of travelling wave terms. Additionally, power sweep simulations indicated that the coefficient importing also affects runtime efficiency, especially for ANN implementations.

These results illustrate an important point that an efficient behavioural model requires structural optimisation under different simulation conditions. Removal of redundant operations can provide considerable performance improvement while maintaining the same model accuracy.

Chapter 5 investigated the impact of systematic uncertainties from a calibrated load-pull measurement system. By measuring a calibrated thru standard, a discrete dataset was obtained. Interpolation using inverse distance weighting built a continuous Γ -dependent error model. This error was presented as an additive travelling wave offset consistent with the Type B uncertainty characteristics. When applied to GaN HEMT load-pull measurements, the extracted correction affected the measured b_2 waves and resulted in changes in the coefficients, especially in the first coefficient. However, traditional performance metrics such as NMSE and RMS error showed minor changes before and after correction. In addition, a practical workflow was proposed to demonstrate how the metric can be applied during load-pull measurements for system integrity verification.

This observation shows that low prediction error does not always mean the measurement is reliable. Undetected systematic errors can affect design decisions. Therefore, traditional accuracy metrics are not enough to assess calibration robustness.

To solve this limitation, a statistical integrity assessment method was developed. Thresholds for different stages in load-pull measurements were obtained. The first threshold is derived from calibration standard measurements, based on magnitude and phase residual statistics evaluated under bootstrap resampling, leading to a 3σ reference limit. The second threshold relates to the behavioural model domain and is designed to quantify the deviation between corrected data and travelling waves under controlled systematic distortions. The results demonstrated that systematic

distortions of around 1–2% of the travelling wave magnitude were enough to exceed the 3σ threshold derived from normal calibrated conditions. This confirms that the integrity metric is sensitive to small systematic distortions that remain undetected by traditional assessment metrics. This means the metric can detect when systematic distortions start to affect load-pull results.

The method builds a link between calibration residual statistics and behavioural model coefficient stability. It enables self-diagnostic assessment of measurement system robustness and provides thresholds for systematic distortions that may not be detected by traditional assessment methods.

6.2 Future Work

While this thesis organises a framework for evaluating behavioural model implementation efficiency and measurement system integrity, several research directions remain open for further development. In particular, the propagation of systematic uncertainties into neural network based models and into harmonic domain formulations would be two natural extension areas from the work in this thesis. Repeated measurements under practical operating conditions are also needed to further evaluate how the proposed metric can be used in real load-pull measurement workflows.

The current investigation of systematic uncertainty focused mainly on the CM, where coefficients are extracted directly from measured travelling waves. In contrast, ANNs rely on parameter optimisation. Their response to systematic measurement errors will be different.

Future work can investigate how Γ -dependent systematic offsets propagate into ANN weight and bias parameters during the training process. By adding controlled measurement distortions into the training dataset, the variation of hidden layer weights, output layer weights, and overall prediction stability can be assessed. A parameter deviation index for ANN models can be defined to evaluate their sensitivity to systematic uncertainties. This can help to compare the robustness of ANN and CM under the same distorted measurement conditions.

In addition, it is necessary to study how systematic errors affect the convergence behaviour in ANN training. Since ANN parameters are updated through gradient descent, measurement bias may change the optimisation path. Analysing training stability under biased data can provide a deeper understanding of uncertainty effects in neural network based behavioural modelling.

Another possible direction can focus on uncertainty related ANN training. For example, uncertainty information can be injected into the loss function, or measurement distortions can be added during training to improve model robustness. This may help build ANN models that are less sensitive to systematic measurement errors.

The analysis in this thesis focused on the fundamental frequency. However, practical large-signal modelling often includes harmonic components. Therefore, the impact of systematic uncertainty on harmonic CM coefficients should be further studied.

Since harmonic CM contains higher-order exponentiation terms of travelling waves, small systematic offsets may become larger than those from the fundamental data. Future work can analyse how systematic uncertainty propagates into second and third harmonic related coefficients. The variation of higher-order harmonic coefficients with respect to measurement errors can be calculated and compared with the fundamental case.

It is also useful to investigate whether systematic errors at the fundamental frequency affect harmonic responses differently. A harmonic domain measurement system integrity metric can be developed to evaluate robustness across multiple frequency components. This can provide a more complete evaluation of behavioural model stability in real load-pull measurement systems.

Overall, this thesis has established a framework for behavioural model implementation, efficiency evaluation, and measurement system integrity assessment in load-pull based power amplifier characterisation. Further work is still needed to extend the method to other modelling approaches and practical measurements. However,

the results of this thesis provide a useful basis for improving the reliability of future measurement-driven power amplifier design.

References

- [1] M. Tian, J. B. Bell, R. Quaglia, E. Azad, and P. J. Tasker, “Artificial neural network nonlinear transistor behavioral models: Structure and parameter determination process based on the Cardiff model,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 73, no. 2, pp. 745–759, 2025. DOI: 10.1109/TMTT.2024.3434959.
- [2] M. Rocio Moure, M. Casbon, M. Fernández-Barciela, and P. J. Tasker, “A Systematic Investigation of Behavioural Model Complexity Requirements,” in *2018 13th European Microwave Integrated Circuits Conference (EuMIC)*, 2018, pp. 77–80. DOI: 10.23919/EuMIC.2018.8539950.

Appendix A

AEL Implementation Details for the ANN Model in ADS

This appendix provides supplementary AEL implementation details for the ANN model developed in ADS. The figures present the schematic blocks used for travelling-wave calculation, parameter import, normalisation, and ANN model calculation, which support the implementation procedure described in Chapter 3.

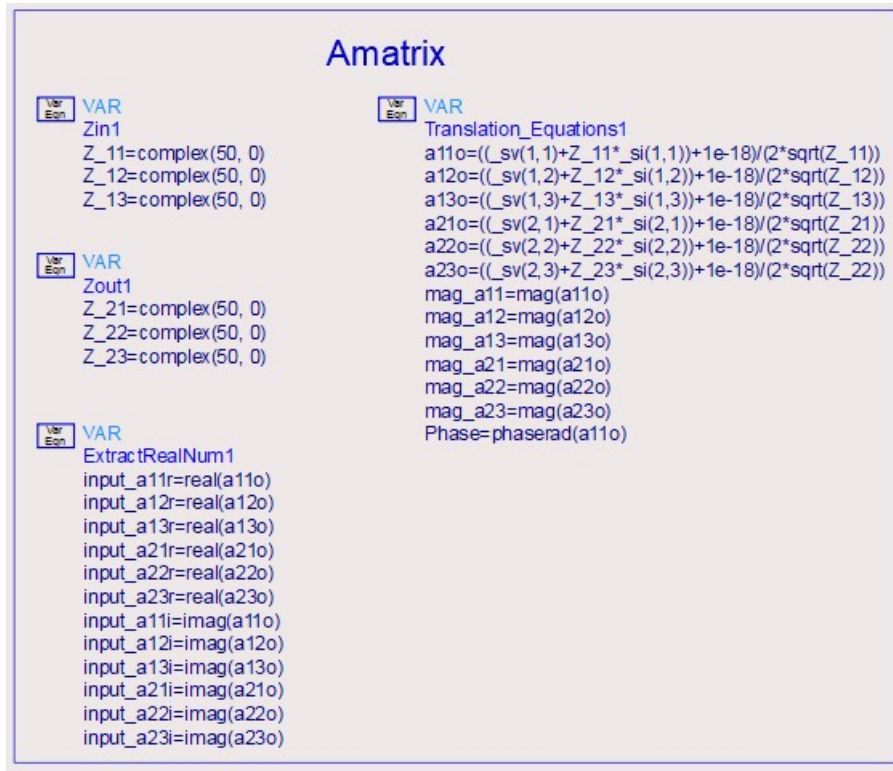


Figure A.1: Travelling-wave calculation and normalisation



Figure A.2: Hidden-layer parameters for data import

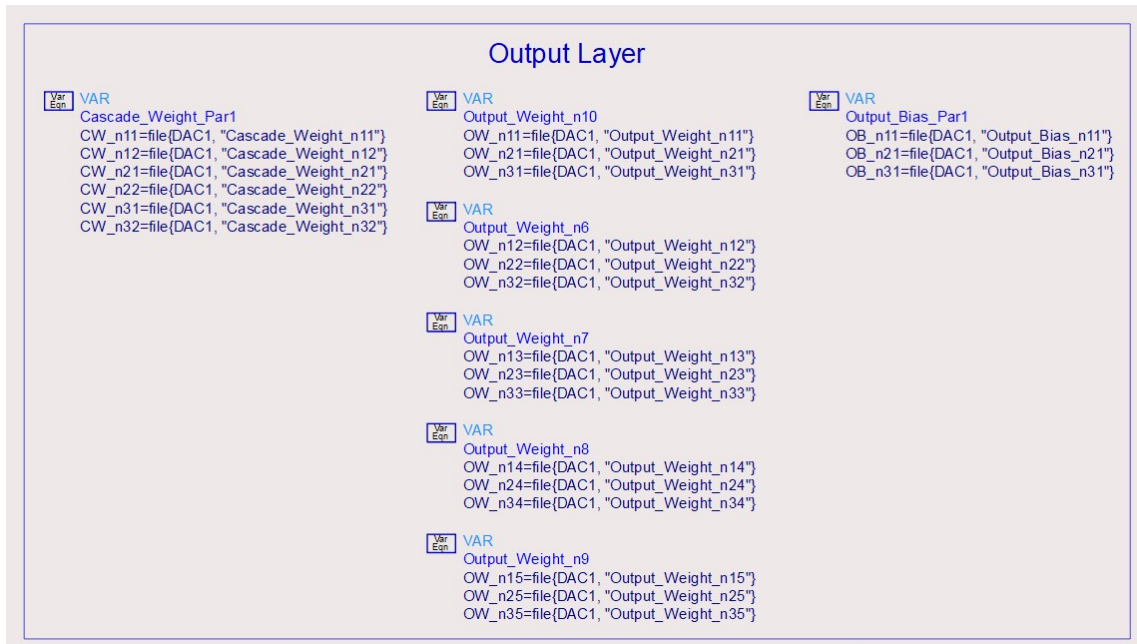


Figure A.3: Output-layer parameters for data import

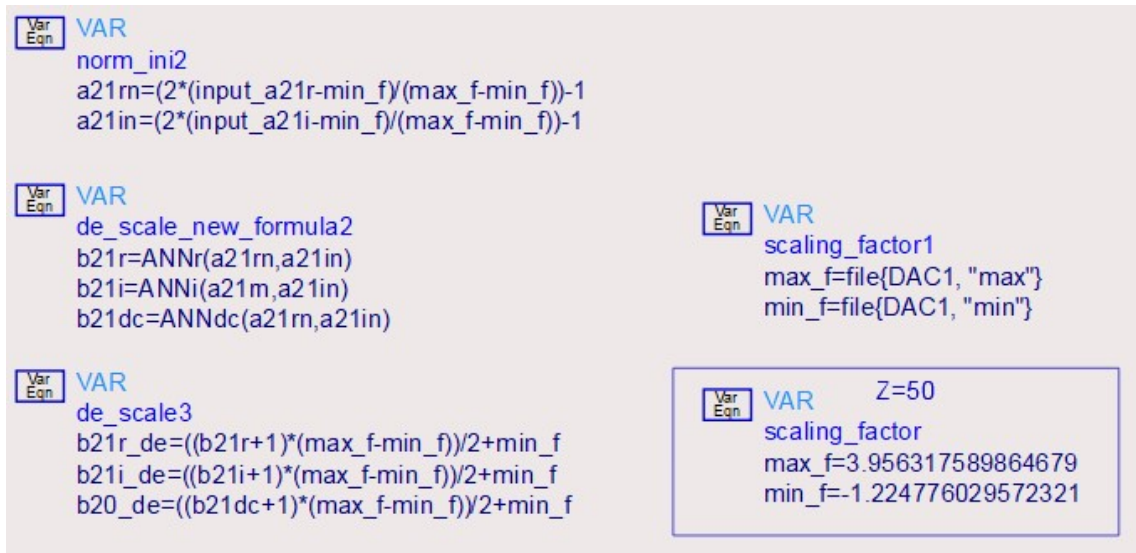


Figure A.4: Weight and bias normalisation and re-normalisation



Figure A.5: ANN model calculation block