Advanced Space Mapping and Artificial Neural Network-Based Techniques for Microwave Device Modeling

by

Zhihao Zhao, B.Eng and B.Mgt

A thesis submitted to
the Faculty of Graduate and Postdoctoral Affairs
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

Electronic Science and Technology, School of Microelectronics
Tianjin University, Tianjin, China

Ottawa-Carleton Institute for Electrical and Computer Engineering
Carleton University, Ottawa, Ontario, Canada

© 2020

Zhihao Zhao

Abstract

Microwave device modeling is an important topic in microwave area. This thesis focuses on developing advanced space mapping and artificial neural network (ANN)-based techniques for challenging microwave device modeling, addressing both microwave active and passive component modelings. Specifically, this thesis develops an advanced space mapping-based technique in combination with knowledge-based neural network (KBNN) for modeling gallium nitride (GaN) high-electron-mobility transistor (HEMT) devices. This thesis also develops advanced ANN-based transfer function mapping techniques for parametric modeling of microwave passive components.

GaN HEMTs are important for next-generation wireless communication systems and microwave power device applications. However, GaN HEMT model development can be time-consuming as the devices exhibit strong trapping effects, which often require very sophisticated model. As a consequence, a fast modeling approach for GaN HEMT devices with trapping effects is of significance for high-reliability microwave circuit design. In the first part of this thesis, we propose a novel space mapping modeling technique for GaN HEMTs with trapping effects. The proposed space mapping technique develops separate mappings for different branches inside the existing device model, such that different behaviors (i.e., trapping effects and frequency dispersion) in GaN HEMTs can be mapped separately. Through supervised learning methods, each mapping module is systematically developed to overcome the gap between each internal branch and each set of target data, accelerating the process of model development. The KBNN model is proposed for characterizing drain

current under DC, and it is used as part of the trapping mapping development. The proposed space mapping technique allows fast and systematic model development for GaN HEMTs with trapping effects to achieve a GaN HEMT large-signal model with satisfactory accuracy. Once developed, the large-signal model can be used in high-performance microwave circuit design, covering DC, PIV, S -parameter, HB, and load-pull simulations.

Neuro-transfer function (neuro-TF) approaches have become popular in parametric modeling for EM behavior of microwave components. Neuro-TF model combines neural networks and a transfer function, where transfer function is used as space mapping between the neural network and the EM responses. In the second part of this thesis, we propose a novel hybrid-based neuro-TF technique to address the discontinuity issue and the non-smoothness issue of the poles/residues when the geometrical variations of the EM structures become large. In the proposed technique, we systematically combine both pole-residue and rational formats of the transfer functions in the neuro-TF model. The proposed technique automatically identifies the poles/residues that are smooth-continuous and the poles/residues that have the discontinuity and non-smoothness issues. The proposed technique converts the poles/residues that have those issues into the coefficients of the rational-based transfer function and remains the smooth-continuous poles/residues in the pole-residue format of the transfer function. The proposed technique can obtain good model accuracy in challenging applications of large geometrical variations, addressing the discontinuity and non-smoothness issues.

In the third part of this thesis, we propose a novel decomposition technique for

the rational-based neuro-TF approach to address the high-sensitivity issue when the geometrical variations become large, where the sensitivity means the sensitivity of EM response with respect to the coefficients of the rational-based transfer function. We propose to decompose the original rational-based neuro-TF model with high-order transfer function into multiple sub-rational-based neuro-TF models with much lower-order transfer function. We then reformulate the overall model as the combination of the sub-neuro-TF models. The proposed technique automatically determines the number of sub-neuro-TF models and the order of transfer function for each sub-neuro-TF models. Using the proposed decomposition technique, we decrease the sensitivities of the overall model response with respect to the coefficients of the transfer function in each sub-neuro-TF models. With the decreased sensitivities, the overall model can be trained more robustly. The proposed modeling technique can achieve good model accuracy in challenging applications of large geometrical variations, addressing the high-sensitivity issue.

KEY WORDS: Artificial neural network (ANN), decomposition, GaN HEMT modeling, parameter extraction, parametric modeling, space mapping (SM), trapping effects, transfer function.

Acknowledgments

My deepest appreciation goes to my supervisor, Prof. Q. J. Zhang for his constant support, profound expertise, and technical discussions during the course of my research work. This thesis would not have been possible without his professional guidance, unwavering enthusiasm, innovative insights, great encouragement, and unfailing patience. It is a great honor and pleasure to work under his supervision. I would also like to express my sincere gratitude to my co-supervisor, Prof. Jianguo Ma for his invaluable counsel, technical discussion, co-supervision, and support.

My special thanks to the people of the modeling team from NXP Semiconductors, Chandler, Arizona. Especially, I would like to thank Dr. Kevin Kim for his support and technical discussions, thank Frederik Vanaverbeke and Philip Saint-erne for the measurement of the GaN HEMT device, and thank Dr. Lei Zhang and Dr. Meng Li for their technical discussions and providing the coarse charge model for the GaN HEMT device.

I would like to acknowledge Prof. Peter Aaen, Prof. Michel Nakhla, Prof. Emad Gad, Prof. Ram Achar, Prof. Qiang Liu, Prof. Yu Luo, Prof. Haipeng Fu, Prof. Pavan Gunupudi, and Prof. Derek McNamara for their valuable suggestions and corrections for my thesis.

I would like to thank Tianjin University, where I have spent eleven years accomplishing both my undergraduate and Ph.D. studies. I would also like to thank Carleton University for the Cotutelle program for my Ph.D. study.

I would like to thank all my former and present lab members especially Dr. Lin Zhu, Dr. Shuxia Yan, Chuan Zhang, Kai Ma, Bo Peng, Dr. Venu-Madhav-Reddy

Gongal-Reddy, Dr. Feng Feng, Dr. Wenyuan Liu, Dr. Chao Zhang, Dr. Weicong Na, Dr. Jianan Zhang, Dr. Wei Zhang, Shunlu Zhang, Jing Jin, and Jinyuan Cui for their valuable suggestions and helpful discussions in my thesis research work.

I would like to thank all my friends, especially Decheng Zhang, Dr. Fu Liu, Zhaopeng Wang, Wen Wen, Renfei Wang, Qianyu Zhang, Jiaqi Zhang, Tong Xue, and Yunhao Li for their friendship and encouragement.

Finally, I would like to thank my parents and grandmother for their endless love, belief, support, and encouragement throughout my Ph.D. study.

Table of Contents

Abstract	i
List of Figures	xi
List of Tables	xx
List of Symbols	xxii
1 Introduction	1
1.1 Introduction and Motivation	1
1.2 List of Contributions	7
1.3 Thesis Organization	9
2 Literature Review	11
2.1 ANN-Based Modeling Approaches	12
2.1.1 Basic Neural Network Structures	13
2.1.2 Time Delay Neural Networks	15
2.1.3 Neural Network Training	17
2.1.4 Knowledge-Based Neural Network	18

2.2	Space Mapping Modeling Approaches	19
2.2.1	Input Space Mapping	20
2.2.2	Implicit Space Mapping	21
2.2.3	Output Space Mapping	22
2.2.4	Neuro-Space Mapping	23
2.3	GaN HEMT Modeling Methods	24
2.3.1	Empirical Function/Equivalent Circuit Method	25
2.3.2	Physics-Based Compact Modeling Method	26
2.3.3	Pure-ANN Modeling Method	27
2.3.4	Discussion on Space Mapping Modeling Method	29
2.4	Parametric Modeling Methods for EM Microwave Passive Components	31
2.4.1	ANN and KBNN Modeling Methods	32
2.4.2	Pole-Residue-Based Neuro-TF Modeling Method	32
2.4.3	Rational-Based Neuro-TF Modeling Method	34
2.5	Conclusion	35

3 Space Mapping Technique Using Decomposed Mappings for GaN

	HEMT Modeling with Trapping Effects	37
3.1	Introduction	37
3.2	Proposed Space Mapping Technique for Development of a GaN HEMT Large-Signal Model	38
3.2.1	Proposed Mapping Topology for a Given GaN HEMT Large- Signal Model	38

3.2.2	Trapping Model Development Using Space Mapping	40
3.2.3	Charge Model Development Using Space Mapping	47
3.2.4	Development of Drain Current Model Using Knowledge-based Neural Network	54
3.2.5	Stepwise Algorithm for the Overall Model Development . . .	56
3.2.6	Discussions	62
3.3	Model Training and Verification	69
3.3.1	Development of Drain Current Model Using KBNN and Model Verification Using DC Data	70
3.3.2	Trapping Model Development Using Space Mapping and Model Verification Using PIV Data	73
3.3.3	Charge Model Development Using Space Mapping and Model Verification Using Pulsed S -parameter Data	77
3.3.4	Overall Model Verification Using Large-Signal Data	86
3.4	Conclusions	103
4	Parametric Modeling of EM Behavior of Microwave Components Using Combined Neural Networks and Hybrid-Based Transfer Func- tions	104
4.1	Introduction	104
4.2	The Issues of Discontinuity and Accompanied Non-Smoothness in Poles and Residues with respect to Geometrical Parameters	105

4.3	Proposed Hybrid-Based Neuro-TF Method for Parametric Model Development	108
4.3.1	Formulation of the Hybrid-Based Neuro-TF Model	108
4.3.2	Proposed Hybrid Parameter Extraction Technique	112
4.3.3	Preliminary Training of Neural Networks	118
4.3.4	Refinement Training of the Hybrid-Based Neuro-TF Model	120
4.4	Application Examples	121
4.4.1	Parametric Modeling of a Four-Pole Waveguide Filter	121
4.4.2	Parametric Modeling of a Three-Pole H -Plane Filter	130
4.5	Conclusion	138
5	Novel Decomposition Technique on Rational-Based Neuro-Transfer Function for Modeling of Microwave Components	141
5.1	Introduction	141
5.2	The High-Sensitivity Issue in the Existing Rational-Based Neuro-TF Method	142
5.3	Proposed Decomposition Technique for Development of Rational-Based Neuro-TF Model	144
5.3.1	Concept of the Decomposition Technique for Rational-Based Neuro-TF Model	144
5.3.2	Proposed Decomposition Technique for Parameter Extraction and Model Development	150
5.4	Application Examples	157

5.4.1	Three-Order Waveguide Filter Modeling	157
5.4.2	Four-Order Bandpass Filter Modeling	164
5.5	Conclusion	171
6	Conclusions and Future Research	172
6.1	Conclusions	172
6.2	Future Research	175
	Bibliography	179

List of Figures

2.1	Structure of the MLP neural network [56].	14
2.2	Structure of the carrier frequency-dependent time delay neural network [35].	16
2.3	Structure of the knowledge-based neural network model [94].	18
2.4	The mathematical representation of the input space mapping methodology [40].	21
2.5	Structure of general two-port neuro-space mapping nonlinear model [43].	23
2.6	Intrinsic topology of the GaN HEMT model [1].	25
2.7	ASM GaN HEMT model [26].	27
2.8	Intrinsic topology of the DynaFET model [2].	28
2.9	Structure of the pole-residue-based neuro-TF model [49].	33
2.10	Structure of the rational-based neuro-TF model [45].	35

3.1	Proposed mapping topology for a given GaN HEMT large-signal model with different mapping modules. The mapping module 1 will be referred to as the trapping mappings. The mapping module 2 and the mapping module 3 will be referred to as the gate charge mapping and the drain charge mapping, respectively. The coarse sub-model 1 will be referred to as the coarse dynamic trapping model. The coarse sub-model 2 and the coarse sub-model 3 will be referred to as the coarse gate charge model and the coarse drain charge model, respectively.	41
3.2	Structure of the proposed trapping mappings for a given coarse dynamic trapping model. The whole model in this figure will be referred to as the mapped-trapping-enabled drain current model.	45
3.3	Proposed knowledge-based neural network model for drain current at PIV (0 V, 0 V).	55
3.4	Flow diagram of the proposed space mapping technique, showing the steps for developing decomposed mappings for various sub-modules and the steps for the overall model development.	61
3.5	Extrinsic large-signal model topology for GaN HEMTs [117]. This topology is used to de-embed the extrinsic effect from the extrinsic measurement data in order to get the intrinsic measurement data. The obtained intrinsic data are used for the development of the intrinsic model.	70

3.6	Measured (circle) and modeled (solid line) PIV (0 V, 0 V) results at different temperatures. (a) PIV (0 V, 0 V) at 25 °C. (b) PIV (0 V, 0 V) at 75 °C. (c) PIV (0 V, 0 V) at 125 °C. The measurement data in the figure are training data.	72
3.7	Measured (circle) and modeled (solid line) DC results.	73
3.8	(a) Measured (circle) and modeled (solid line) DC results around the switching point $v_{gs_swc} = -2.6$ V. (b) Modeled (solid line) DC simulation result of g_m at $v_{ds} = 10$ V.	74
3.9	Measured (circle) and modeled (solid line) PIV results at various quiescent biases: (a) PIV (0 V, 0 V), (b) PIV (-2.84 V, 0 V), (c) PIV (-2.82 V, 28 V). Similar comparisons for other four quiescent biases are provided in Fig. 3.10.	78
3.10	Measured (circle) and modeled (solid line) PIV results at various quiescent biases: (a) PIV (-2.84 V, 48 V), (b) PIV (-5 V, 0 V), (c) PIV (-5 V, 28 V), and (d) PIV(-5 V, 48 V). This figure complements Fig. 3.9 by providing PIV results at the other four quiescent biases.	79
3.11	The charge surfaces of the developed gate charge model and the developed drain charge model. (a) The gate charge surface Q_{gs} . (b) The drain charge surface Q_{ds}	81

3.12	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-5.00 V, 48.01 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -5.00 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 48.01 V.	82
3.13	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-5.00 V, 27.02 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -5.00 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 27.02 V.	83
3.14	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-2.80 V, 28.68 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -2.80 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 28.68 V.	84
3.15	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-2.80 V, 46.72 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -2.80 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 46.72 V.	85

3.16	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-0.01 V, 9.29 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -0.01 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 9.29 V.	86
3.17	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (0.99 V, 10.57 V). The gate voltage is pulsed from quiescent point of 0 V up to pulsed point of 0.99 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 10.57 V.	87
3.18	Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (1.99 V, 5.43 V). The gate voltage is pulsed from quiescent point of 0 V up to pulsed point of 1.99 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 5.43 V.	88
3.19	Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-5.00 V, 27.02 V) and (1.99 V, 5.43 V).	89
3.20	Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-5.00 V, 48.01 V) and (0.99 V, 10.57 V).	90

3.21	Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-2.80 V, 28.68 V) and (-0.01 V, 9.29 V).	91
3.22	Measured (circle) and modeled (solid line) HB results for different load impedances at 3.6 GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$. The dash line represents the coarse model without mapping (a) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=93.79+34.63j$; (b) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=93.79+34.63j$; (c) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=82.06+54.59j$; (d) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=82.06+54.59j$	92
3.23	Measured (circle) and modeled (solid line) HB results for different load impedances at 3.6 GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$. The dash line represents the coarse model without mapping (a) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=84.28+49.18j$; (b) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=84.28+49.18j$; (c) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=89.69+31.45j$; (d) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=89.69+31.45j$	94
3.24	Measured (circle) and modeled (solid line) gate current results at different temperatures and $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V})$. (a) gate current at 25 °C. (b) gate current at 75 °C. (c) gate current at 125 °C.	97

3.25	Modeled load-pull contours at $P_{avs}=21.77$ dBm, $RF_{freq}=3.6$ GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5mA, 48 V)$. (a) PAE contours; (b) P_{out} contours	98
4.1	The structure of the proposed hybrid-based neuro-TF model.	108
4.2	The flow diagram of the proposed three-stage hybrid parameter extraction process.	119
4.3	The flow diagram of the overall hybrid-based neuro-TF model development process.	122
4.4	The geometrical parameters of the four-pole waveguide filter and the 3D configuration for EM simulation. The geometrical parameters are h_1, h_2, h_3, h_{c1} , and h_{c2}	123
4.5	The structure of the proposed hybrid-based neuro-TF model for the four-pole waveguide filter example.	123
4.6	The original poles/residue data with discontinuity issue and the non-smoothness issue versus the newly obtained rational data with improved smoothness (after using the proposed hybrid parameter extraction technique) in Case 2 for the four-pole waveguide filter example. (a) the imaginary part of p_1 versus the training sample from the original poles/residue data, (b) the real part of r_3 versus the training sample from the original poles/residue data, (c) a_2 of the numerator versus the training sample from the newly obtained rational data. .	127

4.7	Comparison of $ S_{11} $ (dB) of the models developed using different modeling methods and the <i>HFSS</i> EM data: (a) test geometrical sample #1, (b) test geometrical sample #2, and (c) test geometrical sample #3 for the four-pole waveguide filter example.	131
4.8	The geometrical parameters of the three-pole <i>H</i> -plane filter and the 3D configuration for EM simulation. The geometrical parameters are L_1 , L_2 , W_1 , and W_2 , with $a = 19.05$ mm, $b = 9.525$ mm, and $t = 2.0$ mm.	132
4.9	The structure of the proposed hybrid-based neuro-TF model for the three-pole <i>H</i> -plane filter example.	132
4.10	The original poles/residue data with discontinuity issue and the non-smoothness issue versus the newly obtained rational data with improved smoothness (after using the proposed hybrid parameter extraction technique) in Case 2 for the three-pole <i>H</i> -plane filter example. (a) the imaginary part of p_1 versus the training sample from the original poles/residue data, (b) the real part of r_7 versus the training sample from the original poles/residue data, (c) a_2 of the numerator versus the training sample from the newly obtained rational data. .	135
4.11	Comparison of $ S_{11} $ (dB) of the models developed using different modeling methods and the <i>HFSS</i> EM data: (a) test geometrical sample #1, (b) test geometrical sample #2, and (c) test geometrical sample #3 for the three-pole <i>H</i> -plane filter example.	139

5.1	Structure for the rational-based neuro-TF model incorporating the proposed decomposition technique.	149
5.2	Geometrical variables and 3D configuration of the three-order waveguide filter example. $a = 19.05$ mm, $b = 9.525$ mm, and $t = 2.0$ mm.	158
5.3	Structure for the proposed rational-based neuro-TF model with $M = 2$ of the three-order waveguide filter example.	160
5.4	Comparison of $ S_{11} $ (dB) of S_{11} different modeling approaches and EM data: (a) test sample $\mathbf{x} = [14.37 \ 14.57 \ 9.25 \ 6.17]^T$ (mm), (b) test sample $\mathbf{x} = [14.17 \ 15.20 \ 8.75 \ 6.25]^T$ (mm), and (c) test sample $\mathbf{x} = [14.56 \ 15.20 \ 8.63 \ 6.17]^T$ (mm) for the three-order waveguide filter example.	164
5.5	Geometrical variables and 3D configuration of the four-order bandpass filter example.	165
5.6	Structure for the proposed rational-based neuro-TF model with $M = 2$ of the four-order bandpass filter example.	167
5.7	Comparison of $ S_{11} $ (dB) of S_{11} for different modeling approaches and EM data: (a) test sample $\mathbf{x} = [3.52 \ 4.46 \ 3.96 \ 3.28 \ 3.07]^T$ (mm), (b) test sample , $\mathbf{x} = [3.60 \ 4.22 \ 4.20 \ 3.40 \ 3.10]^T$ (mm), and (c) test sample $\mathbf{x} = [3.36 \ 4.38 \ 4.00 \ 3.36 \ 3.02]^T$ (mm) for the four-order bandpass filter example.	170
5.8	Comparison of $\text{Re}(S_{11})$ and $\text{Im}(S_{11})$ for different modeling approaches and EM data at test sample $\mathbf{x} = [3.36 \ 4.38 \ 4.00 \ 3.36 \ 3.02]^T$ (mm) for the four-order bandpass filter example. (a) $\text{Re}(S_{11})$ and (b) $\text{Im}(S_{11})$	171

List of Tables

3.1	Mapping Spaces for Different Mapping Modules	63
3.2	Qualitative Comparison of GaN HEMT Model Development between the Proposed Space Mapping Technique and Several Existing Methods	66
3.3	Comparisons of Load-pull Figures of Merit at Maximum Drain Efficiency in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$.	95
3.4	Comparisons of Load-pull Figures of Merit at Maximum Power in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$	95
3.5	Comparisons of Load-pull Figures of Merit at Maximum Drain Efficiency in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$.	96
3.6	Comparisons of Load-pull Figures of Merit at Maximum Power in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$	96
3.7	Comparisons of Output Powers and Power Added Efficiencies for Different Load Impedances at Fixed Available Powers	99
3.8	Summary of Measurement Data Used for Model Development . . .	100
3.9	Comparison of Model Results between the Proposed Space Mapping Technique and the Existing Space Mapping Technique	101

4.1	Definition of Training and Testing Data for The Four-Pole Waveguide Filter Example	126
4.2	Comparisons of Different Modeling Methods for The Four-Pole Waveguide Filter Example	130
4.3	Definition of Training and Testing Data for The Three-Pole H -Plane Filter Example	134
4.4	Comparisons of Different Modeling Methods for The Three-Pole H -Plane Filter Example	138
5.1	Comparison of the Sensitivity of Rational-Based Neuro-TF Model Response with respect to the Coefficients with/without Decomposition.	147
5.2	Definition of Training and Test Samples for The Three-Order Waveguide Filter Example	159
5.3	Comparisons of Different Rational-Based Neuro-TF Modeling Approaches for The Three-Order Waveguide Filter Example	162
5.4	Definition of Training and Test Samples for The Four-Order Bandpass Filter Example	166
5.5	Comparisons of Different Rational-Based Neuro-TF Modeling Approaches for The Four-Order Bandpass Filter Example	168

List of Symbols

- $\mathbf{A}$ The matrix which is used as the training data for the coefficient functions $\alpha_{1j}(\cdot)$, $\alpha_{2j}(\cdot)$, $\alpha_{3j}(\cdot)$, and $\alpha_{4j}(\cdot)$
- $\hat{\mathbf{a}}$ The vector which contains all the $\hat{\mathbf{a}}_i$ of the proposed rational-based neuro-TF model
- $\hat{\mathbf{a}}(\cdot)$ The neural network which represents the relationship between $\hat{\mathbf{a}}$ and $\mathbf{x}$ in the proposed rational-based neuro-TF model
- $\hat{\mathbf{a}}_i$ The vector which contains all the $\hat{a}_{ij}$ in the i th sub-neuro-TF model of the proposed rational-based neuro-TF model
- $\mathbf{a}^h(\cdot)$ The neural network which represents the relationship between $\mathbf{a}^h$ and $\mathbf{x}$ in the proposed hybrid-based neuro-TF model
- $(\mathbf{a}^h)^{(k)}$ The vector which contains the coefficient data for the transfer function numerator converted from the discontinuous pole/residue data and the non-smooth-continuous pole/residue data

$\hat{\mathbf{a}}_i^{(k)}$ The coefficient data of the transfer function numerator for the i th sub-neuro-TF model of the proposed rational-based neuro-TF model

a_j The j th coefficients of the numerator of the rational-based transfer function

$\hat{a}_{ij}$ The j th coefficients of the numerator of the i th sub-transfer function in the proposed rational-based neuro-TF model

a_i^h The i th coefficient of the numerator of the rational-based transfer function in the proposed hybrid-based neuro-TF model

α_{1j}, α_{2j} The j th coefficients of $f_{mp1}^Q(\cdot)$

α_{3j}, α_{4j} The j th coefficients of $f_{mp2}^Q(\cdot)$

$\mathbf{B}$ The matrix containing the difference between $\mathbf{Y}_c$ and $\mathbf{Y}_d$ for the real and imaginary parts at each frequency point

$\hat{\mathbf{b}}$ The vector which contains all the $\hat{\mathbf{b}}_i$ of the proposed rational-based neuro-TF model

$\hat{\mathbf{b}}(\cdot)$ The neural network which represents the relationship between $\hat{\mathbf{b}}$ and $\mathbf{x}$ in the proposed rational-based neuro-TF model

$\hat{\mathbf{b}}_i$ The vector which contains all the $\hat{b}_{ij}$ in the i th sub-neuro-TF model of the proposed rational-based neuro-TF model

$\hat{\mathbf{b}}_i^{(k)}$ The coefficient data of the transfer function denominator for the i th sub-neuro-TF model of the proposed rational-based neuro-TF model

- $\mathbf{b}^h(\cdot)$ The neural network which represents the relationship between $\mathbf{b}^h$ and $\mathbf{x}$ in the proposed hybrid-based neuro-TF model
- $(\mathbf{b}^h)^{(k)}$ The vector which contains the coefficient data for the transfer function denominator converted from the discontinuous pole/residue data and the non-smooth-continuous pole/residue data
- b_j The j th coefficients of the denominator of the rational-based transfer function
- $b_{k\ell}$ The element in the k th row and ℓ th column of matrix $\mathbf{B}$
- $\hat{b}_{ij}$ The j th coefficients of the denominator of the i th sub-transfer function in the proposed rational-based neuro-TF model
- b_i^h The i th coefficient of the denominator of the rational-based transfer function in the proposed hybrid-based neuro-TF model
- $\mathbf{c}_k^h$ The vector which contains the hybrid parameters $(\mathbf{p}^h)^{(k)}$, $(\mathbf{r}^h)^{(k)}$, $(\mathbf{a}^h)^{(k)}$, and $(\mathbf{b}^h)^{(k)}$
- $\tilde{\mathbf{c}}_k$ The vector which contains the poles $\tilde{\mathbf{p}}^{(k)}$ and residues $\tilde{\mathbf{r}}^{(k)}$ after vector fitting
- $\mathbf{d}_k$ The k th sample of the EM data
- E_t The user-defined threshold
- $E_{Tr}(\cdot)$ The error function to be minimized when training the proposed neuro-TF model
- E_{Ts} The testing error between the neuro-TF model and the EM data

$e^{-j\omega\tau}$ The time delay term in frequency domain

$f(\cdot)$ The step function for η_i

$f_2^I(\cdot)$ The drain current model for $V_{gQ}=0$ V and $V_{dQ}=0$ V

$f_1^Q(\cdot)$ The gate charge model

$f_2^Q(\cdot)$ The drain charge model

$f_{ANN}^I(\cdot)$ The ANN drain current model for $V_{gQ}=0$ V and $V_{dQ}=0$ V

$f_c^I(\cdot)$ The coarse drain current model for $V_{gQ}=0$ V and $V_{dQ}=0$ V

$f_{c1}^{dyn-trp}(\cdot)$ The coarse dynamic trapping model which outputs v_{gsec}

$f_{c2}^{dyn-trp}(\cdot)$ The coarse dynamic trapping model which outputs v_{dsec}

$f_{c1}^Q(\cdot)$ The coarse gate charge model (coarse sub-model 2)

$f_{c2}^Q(\cdot)$ The coarse drain charge model (coarse sub-model 3)

$f_{KBNN}^I(\cdot)$ The proposed KBNN drain current model for $V_{gQ}=0$ V and $V_{dQ}=0$ V

$f_{mp1}^Q(\cdot)$ The proposed gate charge mapping (time-dependent mapping) for $f_{c1}^Q(\cdot)$

$f_{mp2}^Q(\cdot)$ The proposed drain charge mapping (time-dependent mapping) for $f_{c2}^Q(\cdot)$

$f_{mp1}^{trp}(\cdot)$ The proposed trapping mapping model for $f_{c1}^{dyn-trp}(\cdot)$

$f_{mp2}^{trp}(\cdot)$ The proposed trapping mapping model for $f_{c2}^{dyn-trp}(\cdot)$

G_p The power gain

- G_t The transducer gain
- $g(\cdot)$ The step function for σ_i^h
- g_m The transconductance
- η_i The ratio between n_i^R and n_s for $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$)
- I The index set of the effective poles/residues
- I_i The i th subset of I
- I_{ds} The drain current at $V_{gQ}=0$ V and $V_{dQ}=0$ V
- I_{gs} The gate current
- I_{ds}^{PIV} The drain current for different quiescent biases
- I_{ds_ANN} The drain current at $V_{gQ}=0$ V and $V_{dQ}=0$ V from $f_{ANN}^I(\cdot)$
- I_{dsc} The drain current at $V_{gQ}=0$ V and $V_{dQ}=0$ V from $f_c^I(\cdot)$
- J The index set of samples of $\tilde{\mathbf{p}}^{(k)}$ (or $\tilde{\mathbf{r}}^{(k)}$)
- J' The index set of the samples of continuous pole data (or continuous residue data)
- J_2 The index set of the samples of the discontinuous pole/residue data and the non-smooth-continuous pole/residue data is expressed
- J_2^a The index set of the samples of discontinuous pole data (or discontinuous residue data)

J_2^b	The index set of the samples of non-smooth-continuous pole/residue data
M	The number of the sub-neuro-TF models
μ_n^p	The mean value of the vector which contains $p_n^{(k)} \forall k \in T_r$
μ_n^r	The mean value of the vector which contains $r_n^{(k)} \forall k \in T_r$
$\tilde{\mu}_i^p$	The mean value of the i th continuous poles
$\tilde{\mu}_i^r$	The mean value of the i th continuous residues
N	The order of the transfer function
N_1	The order of the pole-residue-based transfer function in the proposed hybrid-based neuro-TF model
N_2	The order of the rational-based transfer function in the proposed hybrid-based neuro-TF model
N_d	The number of time delays
N_f	The number of frequency points
N_i	The effective order of the i th sub-transfer function
N_{d1}	The number of time delays for $f_{mp1}^Q(\cdot)$
N_{d2}	The number of time delays for $f_{mp2}^Q(\cdot)$
N_{eff}	The effective order of the transfer function

N_2^a	The number of discontinuous poles (or residues)
N_2^b	The number of the non-smooth-continuous poles/residues
n_i^R	The number of the geometrical samples where $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$) is a real pole (or residue)
n_s	The total number of training samples
Ω	The index set of frequency samples
PAE	The power-added efficiency
P_{in}	The input power
P_{out}	The output power
P_{out} (fund)	The output power at fundamental frequency
P_{out} (2nd)	The output power at 2nd harmonic
P_{out} (3rd)	The output power at 3rd harmonic
$\mathbf{p}_w$	The synaptic parameters of $f_{ANN}^I(\cdot)$
$\mathbf{p}^h(\cdot)$	The neural network which represents the relationship between $\mathbf{p}^h$ and $\mathbf{x}$ in the proposed hybrid-based neuro-TF model
$(\mathbf{p}^h)^{(k)}$	The vector which contains the smooth-continuous pole data
$\mathbf{p}^{(k)}$	The vector which contains the effective poles after sorting for the k th sample

- $\tilde{\mathbf{p}}^{(k)}$ The vector which contains the poles obtained after vector fitting at $\mathbf{x}_k$
- $\hat{\mathbf{p}}_i^{(k)}$ The pole data for the i th sub-neuro-TF model of the proposed rational-based neuro-TF model
- p_i^h The i th pole of the pole-residue-based transfer function in the proposed hybrid-based neuro-TF model
- p_j The j th effective pole of the pole-residue-based transfer function
- p_j^* The complex conjugate parts of p_j
- $p_n^{(k)}$ The n th element of $\mathbf{p}^{(k)}$
- $\tilde{p}_{\tilde{\ell}}^{(k)}$ The $\tilde{\ell}$ th element of $\tilde{\mathbf{p}}^{(k)}$
- ϕ_d The drain trap state
- ϕ_g The gate trap state
- ϕ_d' The input variable of $f_{c1}^{dyn-trp}(\cdot)$ and $f_{c2}^{dyn-trp}(\cdot)$, associated with the drain trap state
- ϕ_g' The input variable of $f_{c1}^{dyn-trp}(\cdot)$ and $f_{c2}^{dyn-trp}(\cdot)$, associated with the gate trap state
- Q_{dsc} The output of the coarse drain charge model $f_{c2}^Q(\cdot)$
- Q_{gsc} The output of the coarse gate charge model $f_{c1}^Q(\cdot)$
- Q_{ds_mp} The output of the proposed drain charge mapping $f_{mp2}^Q(\cdot)$

- Q_{gs_mp} The output of the proposed gate charge mapping $f_{mp1}^Q(\cdot)$
- Q_{ds} The drain charge
- Q_{gs} The gate charge
- $\mathbf{r}^h(\cdot)$ The neural network which represents the relationship between $\mathbf{r}^h$ and $\mathbf{x}$ in the proposed hybrid-based neuro-TF model
- $\mathbf{r}^{(k)}$ The vector which contains the effective residues after sorting for the k th sample
- $\tilde{\mathbf{r}}^{(k)}$ The vector which contains the residues obtained after vector fitting at $\mathbf{x}_k$
- $(\mathbf{r}^h)^{(k)}$ The vector which contains the smooth-continuous residue data
- $\hat{\mathbf{r}}_i^{(k)}$ The residue data for the i th sub-neuro-TF model of the proposed rational-based neuro-TF model
- r_j The j th effective residue of the pole-residue-based transfer function
- r_j^* The complex conjugate parts of r_j
- r_i^h The i th residue of the pole-residue-based transfer function in the proposed hybrid-based neuro-TF model
- $r_n^{(k)}$ The n th element of $\mathbf{r}^{(k)}$
- $\tilde{r}_{\tilde{\ell}}^{(k)}$ The $\tilde{\ell}$ th element of $\tilde{\mathbf{r}}^{(k)}$
- s The Laplace domain frequency

- s_m The m th sample of the Laplace domain frequency
- $\boldsymbol{\sigma}$ The deviation vector
- σ_n The n th element of $\boldsymbol{\sigma}$
- σ_n^p The deviation of the vector which contains $p_n^{(k)} \forall k \in T_r$
- σ_n^r The deviation of the vector which contains $r_n^{(k)} \forall k \in T_r$
- $\tilde{\sigma}_i$ The shared deviation for both i th continuous poles and residues
- $\tilde{\sigma}_0$ The user-defined threshold of the deviation
- $\tilde{\sigma}_i^p$ The deviation of the i th continuous poles
- $\tilde{\sigma}_i^r$ The deviation of the i th continuous residues
- $\mathbf{T}$ The matrix containing the combined frequency and time-delay information at each frequency point
- T_j The junction temperature
- T_r The index set of the training samples of $\mathbf{x}$
- $t_{k\ell}$ The element in the k th row and ℓ th column of matrix $\mathbf{T}$
- τ The time step
- $\hat{\mathbf{u}}$ The vector which contains the indices of the elements of $\boldsymbol{\sigma}$ after sorting
- $\hat{u}_m$ The m th element of $\hat{\mathbf{u}}$

- (V_{gQ}, V_{dQ}) The quiescent bias
- v_{ds} The intrinsic drain-source terminal voltage
- v_{gs} The intrinsic gate-source terminal voltage
- v_{dse} The effective drain-source voltage
- v_{gse} The effective gate-source voltage
- v_{gs_swc} The switching point of v_{gs} which divides the low and high v_{gs} regions
- v_{dsec} The coarse effective drain-source voltage
- v_{gsec} The coarse effective gate-source voltage
- v_{ds_mp} The output of the trapping mapping $f_{mp2}^{trp}(\cdot)$
- v_{gs_mp} The output of the trapping mapping $f_{mp1}^{trp}(\cdot)$
- $(\tilde{v}_{gs}, \tilde{v}_{ds})$ The dynamic bias
- $W(\cdot)$ The weighting function of the proposed KBNN drain current model for $V_{gQ}=0$ V and $V_{dQ}=0$ V
- $\mathbf{w}_a$ The neural network weights of $\hat{\mathbf{a}}(\cdot)$
- $\mathbf{w}_b$ The neural network weights of $\hat{\mathbf{b}}(\cdot)$
- $\mathbf{w}^h$ The vector which contains all the neural network weights in the proposed hybrid-based neuro-TF model

$\mathbf{w}_a^h$ The neural network weights of $\mathbf{a}^h(\cdot)$

$\mathbf{w}_b^h$ The neural network weights of $\mathbf{b}^h(\cdot)$

$\mathbf{w}_p^h$ The neural network weights of $\mathbf{p}^h(\cdot)$

$\mathbf{w}_r^h$ The neural network weights of $\mathbf{r}^h(\cdot)$

ω The angular frequency

$\mathbf{x}$ The vector which contains the geometrical variables

$\mathbf{x}_k$ The k th sample of the geometrical variables

$(\mathbf{x}_k, \mathbf{d}_k)$ The training data for the overall proposed rational-based neuro-TF model

$(\mathbf{x}_k, (\mathbf{a}^h)^{(k)})$ The training data for $\mathbf{a}^h(\cdot)$ in the proposed hybrid-based neuro-TF model

$(\mathbf{x}_k, (\mathbf{b}^h)^{(k)})$ The training data for $\mathbf{b}^h(\cdot)$ in the proposed hybrid-based neuro-TF model

$(\mathbf{x}_k, (\mathbf{p}^h)^{(k)})$ The training data for $\mathbf{p}^h(\cdot)$ in the proposed hybrid-based neuro-TF model

$(\mathbf{x}_k, (\mathbf{r}^h)^{(k)})$ The training data for $\mathbf{r}^h(\cdot)$ in the proposed hybrid-based neuro-TF model

$(\mathbf{x}_k, \hat{\mathbf{a}}^{(k)})$ The training data which are used to train the neural network $\hat{\mathbf{a}}(\cdot)$ of the proposed rational-based neuro-TF model

$(\mathbf{x}_k, \hat{\mathbf{b}}^{(k)})$ The training data which are used to train the neural network $\hat{\mathbf{b}}(\cdot)$ of the proposed rational-based neuro-TF model

$\mathbf{Y}_c$ The Y -parameters from the model which includes the given gate current model, the developed mapped-trapping-enabled drain current model, and the given coarse gate and drain charge models.

$\mathbf{Y}_d$ The Y -parameter data which are converted from the pulsed S -parameter data at $V_{gQ}=0$ V and $V_{dQ}=0$ V.

y The frequency response of the neuro-TF model, e.g., S -parameter

y_i The frequency response of the i th sub-neuro-TF model

Z_{in} The input impedance

Z_{load} The load impedance

Z_{src} The source impedance

$\|\cdot\|$ L_2 norm

Chapter 1

Introduction

1.1 Introduction and Motivation

Microwave device modeling is an important topic in microwave area. With the increasing demand on the performance and functionality of the microwave systems, the microwave devices become more advanced and the device behaviors become more complicated. The need to accurately model many different aspects of the device behaviors makes the modeling task more and more challenging. Therefore, advanced modeling methods are needed in order to accurately model the microwave devices with complicated behaviors. This thesis focuses on developing advanced space mapping and artificial neural network (ANN)-based techniques for challenging microwave device modeling, addressing both microwave active and passive component modelings. Specifically, this thesis develops an advanced space mapping-based technique in combination with knowledge-based neural network (KBNN) for modeling gallium nitride (GaN) high-electron-mobility transistor (HEMT) devices as a microwave active component modeling application. This thesis also develops ad-

vanced ANN-based transfer function mapping techniques for parametric modeling of microwave passive components as microwave passive component modeling applications. We first introduce the existing modeling techniques for GaN HEMT devices.

GaN devices, such as GaN HEMTs, are important for next-generation wireless communication systems and microwave power device applications [1-5]. However, the GaN HEMT devices exhibit strong trapping effects [6-12] that will deteriorate the performance of the RF and microwave circuits. As a consequence, an accurate large-signal model for GaN HEMT devices capable of characterizing the sophisticated trapping effects is essential for high-performance RF and microwave circuit design [13],[14].

Various modeling techniques for GaN HEMTs with trapping effects have been investigated [1],[2],[5],[10],[14-20]. A commonly used approach is the empirical function/equivalent circuit approach, such as [1],[8],[9],[21-24]. As an example, good accuracy of the large-signal model in [1] has been achieved based on carefully developed formulas/circuit topologies and the optimized parameters inside the model. However, the development of an accurate empirical function/equivalent circuit model can be time-consuming, as it requires manual trial-and-errors in finding a particular type of formulas/circuit topologies in order to accurately characterize the GaN HEMTs being modeled. In addition, the existing empirical function/equivalent circuit models which can fit the existing GaN HEMT data may sometimes not fit new GaN HEMT data well. Thus, the existing empirical function/equivalent circuit models need to be modified in order to fit the new GaN HEMT data. However,

such modification may involve manual trial-and-errors and be time-consuming and require expert knowledge.

Physics-based compact modeling approach [4],[17],[25-33], is another type of GaN HEMT modeling technique. In this approach, the model has physics insight, good accuracy, and reasonably fast computational speed. However, this modeling approach often needs a sophisticated parameter extraction procedure, requiring process information. Thus, an alternative modeling approach with a systematic computation flow for GaN HEMTs will be desired.

Another type of GaN HEMT modeling approach is the pure artificial neural network (ANN) technique [2],[6],[16]. ANNs have been used for a wide variety of microwave active component applications, such as power amplifier modeling [34],[35], field-effect transistor (FET) modeling [16],[36], HEMT modeling [6],[36-39], etc. The DynaFET model in [16] is trained using the large-signal measurement data in time-domain. This DynaFET model, as a pure ANN model, fits very well with the measurement data thanks to the ANN's great generalization and characterization capabilities. The modeling technique in [16] is suitable for the GaN HEMT when sufficient measurement data for training can be obtained. If the measurement data are not sufficient, the efficiency of pure ANN models may be compromised.

Space mapping [40-43] is one of the fast modeling approaches to overcome the time-consuming issue in empirical function/equivalent circuit model development, and the data-hungry issue in pure ANN model development. Instead of starting from scratch, the model development in this approach usually begins with a best available model as the given coarse model. By developing mappings, the coarse model is

mapped to match the fine device data. In space mapping technique, mappings are formulated by known mathematical equations, such as linear equations and ANN equations. Those known equations are determined systematically without the need for manual trial-and-errors, speeding up the model development process. Moreover, space mapping can be applied to device modeling even when fewer measurement data are available, lowering the cost of data generation from the measurement. Conventionally, the space mapping technique for modeling nonlinear microwave devices has been employed to externally map the coarse model as a whole [43]. For simplicity, we call this technique external space mapping. This approach works well when the nonlinear devices are typical transistors, such as silicon germanium (SiGe) heterojunction bipolar transistors (HBTs), gallium arsenide (GaAs) metal-semiconductor field-effect transistors (MESFETs), and GaAs HEMTs. However, when behaviors, such as trapping effects and frequency dispersion in GaN HEMTs, are very different for different internal branches of the coarse model, it is hard for one external mapping to simultaneously map different behaviors.

The focus of the first part of this thesis is on developing a new and effective space mapping modeling technique for GaN HEMT where mappings can simultaneously map different behaviors, addressing the time-consuming issue and the data-hungry issue.

The second topic of this thesis is the parametric modeling techniques for microwave passive components. Parametric modeling of electromagnetic (EM) behaviors for microwave components has become important for EM-based design [44-53]. EM-based design can be time-consuming, since repetitive simulations of EM are

usually required due to value adjustments of geometrical parameters. Parametric models are developed to characterize the relationship between geometrical variables and EM responses. The developed parametric models allow faster simulation and optimization with varying values of geometrical parameters and can be subsequently implemented in high-level design of circuit and system.

ANN has also been an important vehicle for parametric modeling of EM behavior in radio frequency and microwave area [44-55]. ANN benefits from its strong learning and generalization capabilities and it has been used for a wide variety of microwave applications [56-77]. The universal approximation theorem [78] of ANNs provides the theoretical foundation that if sufficient data are used in training, the good accuracy of ANN models can be achieved within the training region. To develop an accurate parametric model, we train ANNs to learn the nonlinear relationship between geometrical variables and EM behavior.

Another parametric modeling approach is the knowledge-based approach. The knowledge-based neural network (KBNN) is a recognized knowledge-based approach for parametric modeling. In the KBNN approach, KBNN models are developed using neural networks combined with knowledge models such as analytical formulas [79], empirical functions [80], or equivalent circuit models [81],[82]. The knowledge models can help speed up model development and enhance the capability for learning and generalization of the KBNN models [83]. Space mapping [40-43],[84] is also a recognized knowledge-based approach for parametric modeling. Research endeavors on space mapping have focused on several areas, such as input space mapping[40],[81], implicit space mapping [86-88],[90], output space mapping [89-

91], neural space mapping [38],[42],[43],[92],[93], etc.

Another popular parametric modeling method is the neuro-transfer function (neuro-TF) method [45],[48],[49],[51], which combines neural networks and transfer functions. In this method, EM responses of passive components versus frequency are represented by the transfer functions. This approach can be employed even if accurate knowledge models are not available. One of the popular neuro-TF methods is the pole-residue-based neuro-TF method [49],[51]. This method can function well with small geometrical variations. However, with large geometrical variations, some poles and residues obtained from vector fitting do not change continuously with respect to geometrical variables, resulting in the discontinuity issue. Along with the discontinuity issue in the poles/residues, the smoothness of the neighboring poles/residues with respect to the geometrical parameters are also affected, resulting in the non-smoothness issue. These issues lead to the difficulty of training for the overall model, decreasing the effectiveness of the method.

Rational-based neuro-TF method [45] is another popular neuro-TF method, which does not have the discontinuity issue of poles and residues. The rational-based neuro-TF works well when the order in the neuro-TF model is low. However, as the order increases, the transfer function response is very sensitive to the coefficients of the transfer function, leading to the high-sensitivity issue. The high-sensitivity issue can result in the difficulty of training for the overall model when the order in the neuro-TF model is high and the geometrical variations becomes large.

The focus of the second part of this thesis is on developing a new type of neuro-TF parametric modeling technique for EM microwave passive components,

addressing the discontinuity issue and the associated non-smoothness issue when the geometrical variations become large. The focus of the third part of this thesis is on developing a new rational-based neuro-TF parametric modeling technique for EM microwave passive components, addressing the high-sensitivity issue when the order in the neuro-TF model is high and the geometrical variations become large.

1.2 List of Contributions

The main objective of this thesis is to investigate and develop advanced space mapping and ANN-based techniques for challenging microwave device modeling, addressing both microwave active and passive components. Specifically, this thesis investigates and develops a fast modeling technique for GaN HEMTs with trapping effects and two accurate parametric modeling techniques for microwave passive components. Toward this objective, the following significant contributions are made:

- A novel space mapping modeling technique using decomposed mappings for GaN HEMTs with trapping effects is proposed. Separate mappings are developed for different branches inside the existing device model, such that different behaviors (i.e., trapping effects and frequency dispersion) in GaN HEMTs can be mapped separately. Each mapping module is formulated to map a specific behavior in the overall model. Through supervised learning methods, each mapping module is systematically developed to overcome the gap between each internal branch and each set of target data, accelerating the process of model development. The proposed space mapping technique allows fast and systematic large-signal model development for GaN HEMTs with trapping

effects. The proposed technique employs less training data than the existing ANN approach, lowering the cost of data generation from measurements. Once developed, the large-signal model can be used in high-performance microwave circuit design, covering DC, PIV, S -parameter, HB, and load-pull simulations.

- A novel hybrid-based neuro-TF technique is developed to address the discontinuity issue and the associated non-smoothness issue of the poles/residues when the geometrical variations become large. In the proposed technique, we systematically combine both pole-residue and rational formats of the transfer functions in the neuro-TF model. The proposed technique automatically identifies the poles/residues that are smooth-continuous and the poles/residues that have the discontinuity and non-smoothness issues. The proposed technique converts the poles/residues that have those issues into the coefficients of the rational-based transfer function and remains the smooth-continuous poles/residues in the pole-residue format of the transfer function. The proposed technique can obtain good model accuracy in challenging applications of large geometrical variations, addressing the discontinuity and non-smoothness issues.
- A novel decomposition technique on the rational-based neuro-TF is developed to address the high-sensitivity issue. In the proposed technique, we decompose the original rational-based neuro-TF model with high order of transfer function into multiple sub-rational-based neuro-TF models with much lower order

of transfer function. The overall model is reformulated as the combination of the sub-neuro-TF models. The proposed technique automatically determines the number of sub-neuro-TF models and the order of transfer function for each sub-neuro-TF models. Using the proposed decomposition technique, we can decrease the sensitivities of the overall model response with respect to the coefficients of the transfer function in each sub-neuro-TF models. With the decreased sensitivities, the overall model can be trained more robustly. The proposed modeling technique can achieve good model accuracy in challenging applications of large geometrical variations, addressing the high-sensitivity issue.

1.3 Thesis Organization

The rest of the thesis is organized as follows: Chapter 2 presents a literature review of popularly used GaN HEMT modeling approaches and parametric modeling approaches for EM microwave passive components. An overview of ANN modeling methods and space mapping modeling methods are provided. The existing GaN HEMT modeling approaches including the empirical function/equivalent circuit method, the physics-based compact modeling method, and the pure-ANN method are reviewed. The existing neuro-TF parametric modeling methods including the ANN method, the KBNN method, the pole-residue-based neuro-TF method and the rational-based neuro-TF method are reviewed.

Chapter 3 presents a novel space mapping technique which develops separate mappings for different branches inside the existing device model, such that different

behaviors in GaN HEMTs can be mapped separately. The proposed technique allows fast model development for GaN HEMTs with trapping effects to achieve a GaN HEMT model with satisfactory accuracy. This proposed technique is illustrated by a GaN HEMT example with measurement data.

Chapter 4 presents a novel hybrid-based neuro-TF technique that addresses the discontinuity issue and the associated non-smoothness issue with large geometrical variations. By systematically combining both pole-residue and rational formats of the transfer functions in the neuro-TF model, the proposed technique can resolve the discontinuity and non-smoothness issues, achieving good model accuracy for applications of large geometrical variations. This proposed technique is illustrated by two EM microwave component examples.

Chapter 5 presents a novel decomposition technique on the rational-based neuro-TF approach, addressing the high-sensitivity issue with large geometrical variations. By decomposing the original neuro-TF model into multiple sub-neuro-TF models, the proposed technique can resolve the high-sensitivity issue and increase the training robustness of the overall model, achieving good model accuracy for high-order applications. This proposed technique is demonstrated by two EM microwave component examples.

Finally, Chapter 6 presents the conclusions and possible future research directions.

Chapter 2

Literature Review

With the increasing demand on the performance and functionality of the microwave systems, the microwave devices become more advanced and the device behaviors become more complicated. The need to accurately model many different aspects of the device behaviors makes the modeling task more and more challenging. Therefore, advanced modeling methods are needed in order to accurately model the microwave devices with complicated behaviors.

GaN HEMTs are microwave active devices which exhibit strong trapping effects that would impact the performance of the RF and microwave circuits. As a consequence, an accurate large-signal model for GaN HEMT devices capable of characterizing the sophisticated trapping effects is the key for high-performance RF and microwave circuit design. On the other hand, parametric modeling of EM behaviors for microwave passive components has become important for EM-based design in microwave area. Parametric models can be developed to characterize the relationship between geometrical variables and EM responses. Instead of running repetitive EM simulations which can be time-consuming for EM-based design, the

developed parametric models allow faster simulation and optimization with varying values of geometrical parameters and can be subsequently implemented in high-level design of circuit and system.

This literature review first provides an overview of ANN modeling approaches and space mapping modeling approaches. Then, some popular existing GaN HEMT modeling methods are reviewed. Finally, some popular existing parametric modeling methods are reviewed.

2.1 ANN-Based Modeling Approaches

Artificial neural networks (ANNs) have been recognized as important vehicles for device modeling in RF and microwave area [56-64],[66-77],[80]. As important data-driven models in the passive and active RF and microwave device modeling area, ANN models benefit from their strong learning and generalization capabilities [78]. According to the universal approximation theorem of ANNs, ANNs can approximate arbitrary multidimensional continuous functions to user desired accuracy [78]. This theorem provides the theoretical foundation that if sufficient data are used in training, the good accuracy of ANN models can be achieved within the training region. ANN technique is also a fast and systematic modeling approach. ANN models can be developed through a systematic training algorithm, i.e., neural network training algorithm [37]. Through training process, ANNs can automatically learn from the device data.

ANNs have been used for modeling a wide variety of microwave components [56-64],[66-77],[80], including microstrip filters [47],[68], antennas [57], vias [58],

high-speed interconnects [59],[60], transmission line components [61],[94], coplanar waveguide (CPW) components [62], waveguide filters [63],[64], embedded passives [54],[67], voltage-controlled oscillators [69], mixers [71], bends [80], spiral conductors [66], power amplifiers [34],[35], field-effect transistors (FETs) [16],[36], high-electron-mobility transistors (HEMTs) [6],[37-39], etc.

2.1.1 Basic Neural Network Structures

Diverse ANN structures have been researched for different microwave applications over the years, such as multilayer perception (MLP) neural networks [95], radial basis function (RBF) neural networks [44],[96],[97], recurrent neural networks (RNNs) [70-72], dynamic neural networks (DNNs) [74],[75], time delay neural networks (TDNNs) [35],[38],[73], state-space dynamic neural networks (SSDNNs) [76],[77], and deep neural networks [98].

Multilayer Perceptrons (MLP) [56],[95] are the most popular used neural network type in many different applications. In the structure of MLP, the neurons are divided into different layers. The first layer is called input layer and the last layer is called output layer. The rest of layers in between of these two layers are called hidden layers. Input neurons are neurons in the input layer, receiving stimulus outside the neural networks. Output neurons are neurons in the output layer, sending stimulus outside the neural networks. Hidden neurons are neurons in the hidden layer, receiving and sending stimulus from and for other neurons in the neural network. Typically, an MLP consists of one input layer, one or more hidden layers, and one output layer, as shown in Fig. 2.1.

Figure 2.1: Structure of the MLP neural network [56].

Radial basis function (RBF) neural network is another popular neural network structure where radial basis activation functions are used for hidden neurons [44],[96],[97]. A typical RBF neural network consists of one input layer, one output layer, and one RBF hidden layer. The Gaussian and multiquadratic functions are the most commonly used radial basis activation functions. Wavelet network [99-101] is another type of neural network structure, where wavelet activation functions are used for hidden neurons. MLP, RBF neural network, and wavelet network are the feedforward neural network structures for microwave active and passive component modelings.

Apart from the basic neural network structures, there is another type of neural network structure which allows time-domain behaviors of a dynamic system, including recurrent neural networks (RNNs), dynamic neural networks (DNNs), time delay neural networks (TDNNs), state-space dynamic neural networks (SSDNNs). In the next subsection, we will describe the time delay neural network as an example of this type of neural network structure.

2.1.2 Time Delay Neural Networks

Time delay neural network (TDNN)[35],[38],[73] is a neural network structure which allows time-domain behaviors of a dynamic system. This type of neural network is established depending not only on the present inputs but also on the history of the system states and inputs. TDNN has been applied to nonlinear behavior modeling. In [35], a carrier frequency-dependent TDNN model is developed for behavioral modeling of a memory power amplifier (PA) under multi-band operation.

Let f_c represent the carrier frequency for a memory PA. Let $I_{in}(n)$ and $Q_{in}(n)$ represent the in-phase and quadrature-phase components of the input baseband signal at n th time sequence, respectively. Let $I_{out}(n)$ and $Q_{out}(n)$ represent the in-phase and quadrature-phase components of the output baseband signal at n th time sequence, respectively. Let M denote the memory depth for the TDNN model. Fig. 2.2 illustrates the structure of this TDNN model for PA behavioral modeling. In this figure, neural network is established depending not only on the present inputs $I_{in}(n)$ and $Q_{in}(n)$, but also on the history of the inputs (or time-delayed inputs) $I_{in}(n-1)$, $Q_{in}(n-1)$, ..., $I_{in}(n-M+1)$ and $Q_{in}(n-M+1)$. These delayed inputs

Figure 2.2: Structure of the carrier frequency-dependent time delay neural network [35].

form as extra inputs in the MLP neural network. A feedforward neural network (i.e., the MLP) together with the time delay in the inputs results in a TDNN structure.

Based on the structure of Fig. 2.2, the carrier frequency-dependent TDNN model is formulated as

$$(I_{out}(n), Q_{out}(n)) = f_{ANN}(f_c, I_{in}(n), Q_{in}(n), I_{in}(n-1), Q_{in}(n-1), \dots, I_{in}(n-M+1), Q_{in}(n-M+1), \mathbf{w}) \quad (2.1)$$

where $\mathbf{w}$ represents a vector containing all neural network weights.

In the TDNN model, the dynamic information is introduced by the histories (or the time-delayed inputs) and the nonlinearity is represented by the feedforward neural network. Therefore, the TDNN model has strong learning ability of characterizing the complicated nonlinear and dynamic properties of microwave devices. The TDNN model has been applied for behavioral modeling various microwave devices, such as GaAs MESFET, GaAs HEMT, GaAs pHEMT, GaN HEMT PA.

2.1.3 Neural Network Training

A neural network cannot represent any device/circuit behavior unless it is well trained with corresponding data [78]. A systematic model development process for neural network consists of data generation, data scaling, training, validation, and testing.

Let $\mathbf{x}$ and $\mathbf{y}$ represent an input vector and an output vector, respectively. A neural network is used to represent the relationship between $\mathbf{x}$ and $\mathbf{y}$, formulated as

$$\mathbf{y} = f_{ANN}(\mathbf{x}, \mathbf{w}) \quad (2.2)$$

where $\mathbf{w}$ represents a vector containing all neural network weights.

A neural network model is developed by optimizing the neural network weights with a set of simulated/measured data called the training data. The training data is denoted as $(\mathbf{x}_k, \mathbf{d}_k)$ $k \in T_r$, where $\mathbf{d}_k$ represents the measured/simulated output data for the input $\mathbf{x}_k$, and T_r represents the index set of training data. For training purpose, the error function is defined as,

$$E(\mathbf{w}) = \frac{1}{2} \sum_{k \in T_r} \|f_{ANN}(\mathbf{x}_k, \mathbf{w}) - \mathbf{d}_k\|^2 \quad (2.3)$$

The primary objective of neural network training is to minimize $E(\mathbf{w})$ by adjusting $\mathbf{w}$. Many training algorithms are developed for neural network training, such as back-propagation, conjugate gradient, Quasi-Newton, Levenberg-Marquardt, Genetic Algorithms and Simulated Annealing. Once trained, an independent set of data, called testing data, are used to test the accuracy of the neural network model.

Figure 2.3: Structure of the knowledge-based neural network model [94].

 Fig. 2-6 Structure of a knowledge-based neural network^[8].

If both training and testing results are satisfied with the user desired accuracy, the neural network model are well trained and can be used in circuit simulation and design optimization. The space mapping technique establishes a mathematical link between the coarse and the fine models and directs the bulk of the CPU-intensive computations to the coarse model, while preserving the accuracy offered by the fine model.

2.1.4 Knowledge-Based Neural Network

Let $\mathbf{x}_c$ and $\mathbf{x}_f$ represent the input parameters of the coarse and fine models, respectively, and let $\mathbf{R}_c(\mathbf{x}_c)$ and $\mathbf{R}_f(\mathbf{x}_f)$ represent the corresponding model responses. $\mathbf{R}_c$ is much combines prior knowledge with the learning ability of neural networks by incorporating the empirical information into the internal structure of neural networks.

Fig. 2.3 illustrates the comprehensive structure of the KBNN. In KBNN, the knowledge is embedded as part of the neural network internal structure. The KBNN

structure includes six layers that are not fully connected to each other, called the input layer, the knowledge layer, the boundary layer, the region layer, the normalized region layer, and the output layer. The knowledge layer contains the knowledge, complementing the ability of learning and generalization of neural networks with additional knowledge. The boundary layer incorporates knowledge in the form of problem dependent boundary functions. The region layer contains neurons for constructing regions from boundary neurons. The normalized region layer contains rational function based neurons to normalize the outputs of the region layer. The output layer consists of second-order neurons combining normalized region neurons and knowledge neurons.

The KBNN exploits existing knowledge in the form of empirical functions or equivalent circuit model together with a neural network model to develop faster and more accurate models. Extrapolation capability of KBNN is also enhanced by embedding knowledge in the model.

2.2 Space Mapping Modeling Approaches

Space mapping [40-43],[84], as a recognized knowledge-based modeling methodology, is a fast and systematic modeling approach for modeling RF and microwave devices. Space mapping establishes a mathematical link between a coarse model and a fine model. The coarse model is a low-fidelity yet computationally inexpensive model. The fine model is a high-fidelity and expensive model. Mappings are developed to map the coarse model such that the mapped coarse model matches the fine model. Space mapping was originally an optimization approach [40],[87],[102],[103], and it

has been subsequently used for passive and active device modeling. Research on space mapping has focused on several areas, such as input space mapping [40],[81], implicit space mapping [86-88],[90], output space mapping [89-91], neural space mapping [38],[42],[43],[52],[92],[93], generalized space mapping [84], tuning space mapping [104],[105], portable space mapping [106], parallel space mapping [107], coarse and fine mesh space mapping [108], aggressive space mapping [102], etc.

In the space mapping modeling approach, the examples of the coarse model are analytical formulas [103], empirical functions [43], and equivalent circuit models [83]. Given the coarse model, mappings can be systematically developed to overcome the gap between the coarse model and the fine device data, and hence improve the accuracy of the overall model.

2.2.1 Input Space Mapping

Input space mapping is the most standard space mapping methodology [40],[81]. Input space mapping focus on reducing the misalignment between the fine and coarse models by establishing a mapping between the input spaces (e.g., design parameter spaces) of the fine and coarse models. Input mapping with the linear mapping function is also called as original space mapping, defined as

$$\mathbf{x}_c = \mathbf{P}_{SM}(\mathbf{x}_f) = \mathbf{B}_{SM}\mathbf{x}_f + \mathbf{c}_{SM} \quad (2.4)$$

where $\mathbf{B}_{SM}$ and $\mathbf{c}_{SM}$ represent the coefficients of a linear mapping function; $\mathbf{x}_c$ is a vector containing the inputs of the coarse model; $\mathbf{x}_f$ is a vector containing the inputs of the fine model; and $\mathbf{P}_{SM}$ represents the mapping function. Fig. 2.4 shows the

Figure 2.4: The mathematical representation of the input space mapping methodology [40].

mathematical representation of the input space mapping methodology. Let $\mathbf{R}_c(\mathbf{x}_c)$ denote the response of the coarse model. Let $\mathbf{R}_f(\mathbf{x}_f)$ denote the response of the fine model. We establish mapping for the coarse model such that the mapped coarse model (or the surrogate model) matches the fine model, expressed as

$$\mathbf{R}_f(\mathbf{x}_f) = \mathbf{R}_c(\mathbf{P}_{SM}(\mathbf{x}_f)). \quad (2.5)$$

2.2.2 Implicit Space Mapping

Implicit space mapping [86-88],[90] explores the flexibility of the preassigned parameters such as dielectric constant, substrate height as auxiliary input variables for the mapping. The auxiliary input variables are selected based on the analysis of the coarse model. Implicit space mapping usually offers greater flexibility in creating and enhancing mapped models (or surrogate models).

Let $\mathbf{x}_{aux}$ represent the auxiliary input variables (i.e., preassigned parameters) for the coarse model. The coarse model response $\mathbf{R}_c$ is then expressed as $\mathbf{R}_c(\mathbf{x}_c, \mathbf{x}_{aux})$.

Implicit space mapping focuses on developing an implicit mapping $\mathbf{P}_{SM}$ between the input spaces $\mathbf{x}_f$, $\mathbf{x}_c$, and $\mathbf{x}_{aux}$, defined as

$$\mathbf{P}_{SM}(\mathbf{x}_f, \mathbf{x}_c, \mathbf{x}_{aux}) = 0 \quad (2.6)$$

such that

$$\mathbf{R}_f(\mathbf{x}_f) = \mathbf{R}_c(\mathbf{x}_c, \mathbf{x}_{aux}). \quad (2.7)$$

Implicit mapping can provide a good match between the coarse and fine models when input space mapping itself cannot obtain a good match.

2.2.3 Output Space Mapping

Output space mapping technique [89-91] is introduced to enhance the robustness of the space mapping modeling process in cases where input or implicit space mapping cannot provide sufficient matching with the fine model. Output space mapping enhances the mapped model (or surrogate model) by a difference term, i.e., the difference between the fine model and the coarse model, defined as,

$$\mathbf{R}_f(\mathbf{x}_f) = \mathbf{P}_{SM}(\mathbf{x}_f) + \mathbf{R}_c(\mathbf{x}_f). \quad (2.8)$$

where $\mathbf{P}_{SM}$ provides a difference term at the output space in the mapped model (or surrogate model). Typically, output space mapping improves the performance of the space-mapping algorithm when applied as an auxiliary mapping. If the range of the coarse model is substantially different from that of the fine model, output space mapping is practically mandatory to facilitate the space-mapping algorithm.

Figure 2.5: Structure of general two-port neuro-space mapping nonlinear model [43].

2.2.4 Neuro-Space Mapping

Neuro-space mapping technique [38],[43],[93] has been applied as a fast and systematic approach to microwave active device modeling. According to [43], the starting point for the neuro-space mapping technique is when the existing/available device model cannot match the data of a new device. The existing/available nonlinear device model is called the coarse model. The fine model is a fictitious model implied by actual device data from measurement or detailed/expensive device simulation.

The neuro-space mapping technique for modeling nonlinear RF and microwave devices is employed to externally map the coarse model as a whole [43]. Fig. 2.5 illustrates the structure of general two-port neuro-space mapping nonlinear model, where a neural network is used to provide mapping between coarse input signals and fine input signals. In this figure, the terminal voltage signals are mapped onto the voltage signals of the coarse model. By such mapping, the current signals of

the modified coarse model will match the fine current signals. Since the mapping relationship between the voltage signals are unknown and can be nonlinear, the neural network is used to learn the mapping. This neuro-space mapping model will match the fine device data more closely than that possible from the coarse model alone.

This space mapping modeling approach works well for many typical transistors, such as SiGe HBTs, GaAs MESFETs, and GaAs HEMTs.

2.3 GaN HEMT Modeling Methods

GaN-based RF and power devices, such as GaN HEMTs, have attracted tremendous interests in recent years for next-generation wireless communication systems and microwave power device applications [1-5], such as power amplifiers, mixers, and oscillators. However, the GaN HEMT devices exhibit strong trapping effects [6-12] that would deteriorate the performance of the RF and microwave circuits. As a consequence, an accurate large-signal model for GaN HEMT devices capable of characterizing the sophisticated trapping effects is the key for high-reliability RF and microwave circuit design [13],[14].

Various modeling methods for GaN HEMTs with trapping effects have been investigated [1],[2],[5],[10],[14-20], such as the empirical function/equivalent circuit method, the physics-based compact modeling method, and the pure-ANN modeling method.

Figure 2.6: Intrinsic topology of the GaN HEMT model [1].

2.3.1 Empirical Function/Equivalent Circuit Method

A commonly used modeling approach for GaN HEMT devices is the empirical function/equivalent circuit method, such as [1],[8],[9],[21-24]. As an example of the empirical function/equivalent circuit approach, the large-signal model developed in [1] has successfully characterized the GaN HEMT with trapping effects.

Fig. 2.6 illustrates the topology of the GaN HEMT model based on a modified version of the Angelov-Chalmers model. This topology consists of the main nonlinear part, a thermal subcircuit, and a drain source to account for trapping effects. The modified version of the Angelov-Chalmers model for GaN HEMT is with a total of 24 parameters, formulated as

$$\begin{aligned}
 i_{ds} &= f(v_{DS}, v_{GSE}, T_j) \\
 &= I_{pk0}(T_j) \cdot [1 + M_{pk}(v_{DS}, v_{GSE}) \cdot \tanh(\psi(v_{DS}, v_{GSE}))] \cdot \tanh(\alpha \cdot v_{DS})
 \end{aligned} \tag{2.9}$$

$$C_{gs} = C_{gsp} + C_{gs0} \cdot [1 + \tanh(Q_{10} + Q_{11} \cdot v_{GS})] \cdot [1 + \tanh(Q_{20} + Q_{21} \cdot v_{DS})] \tag{2.10}$$

$$C_{gd} = C_{gdp} + C_{gd0} \cdot [1 + \tanh(Q_{30} + Q_{31} \cdot v_{GS})] \cdot [1 + \tanh(Q_{40} + Q_{41} \cdot v_{DS})] \quad (2.11)$$

where

$$\begin{aligned} v_{GSE} = & V_{GSQ} + v_{gs} + k_{g1} \cdot (V_{GSQ} + v_{gs} - V_{GSQ0}) \cdot V_{DSQ} \cdot v_{ds} \\ & + k_{g2} \cdot (V_{GSQ} - V_{GSQ0}) \cdot (V_{GSQ} + v_{gs} - V_{GSQ0}). \end{aligned} \quad (2.12)$$

v_{GSE} is called the effective voltage, which is quiescent bias dependent to be used to allow the drain characteristic to take account of bias-dependent dispersion (trapping effects).

Good accuracy of the large-signal model in [1] has been achieved based on the carefully developed formulas/circuit topologies and the optimized parameters inside the model. However, the development of an accurate empirical function/equivalent circuit model could be time-consuming, as it requires lots of manual development of formulas/circuit topologies to accurately model the GaN HEMTs. In addition, the existing empirical function/equivalent circuit models that can approximate an existing GaN HEMT data may sometimes not fit new GaN HEMT data well.

2.3.2 Physics-Based Compact Modeling Method

Another type of GaN HEMT modeling technique is the physics-based compact modeling method [4],[17],[25-33]. In this approach, the model is developed based on the analytical solution of the physics equations.

In [17],[26], an ASM model is developed for GaN HEMT devices. As shown in 2.7, the ASM GaN HEMT model consists of core models of several real-device effects in the GaN device, including self-heating, trapping effects, velocity saturation, etc.

Figure 2.7: ASM GaN HEMT model [26].

The ASM model is a physics-based compact model where closed form expressions for drain current and intrinsic charges are derived in terms of surface potential.

The model has physics insight, good accuracy, and reasonably fast computational speed. This modeling approach usually takes long time to develop a good model for a new device. Another disadvantage of this modeling approach is that it often needs a sophisticated parameter extraction procedure, requiring process information. Thus, an alternative modeling approach with a systematic computational flow for GaN HEMTs will be desired.

2.3.3 Pure-ANN Modeling Method

Another type of GaN HEMT modeling approach is the pure ANN technique [16],[2],[6]. In [16], the large-signal model named DynaFET model is developed by a pure ANN modeling approach for GaN HEMTs. Fig. 2.8 illustrates the topology of the DynaFET model. This topology includes a electrical circuit (top), a thermal circuit (middle), and two trapping circuits (bottom). Based on this topology, the DynaFET

Figure 2.8: Intrinsic topology of the DynaFET model [2].

model is formulated as

$$I_D(t) = I_D(V_{gs}(t), V_{ds}(t), T_j(t), \Phi_1(t), \Phi_2(t)) + \frac{d}{dt} Q_D(V_{gs}(t), V_{ds}(t), T_j(t), \Phi_1(t), \Phi_2(t)) \quad (2.13)$$

$$I_G(t) = I_G(V_{gs}(t), V_{ds}(t), T_j(t)) + \frac{d}{dt} Q_G(V_{gs}(t), V_{ds}(t), T_j(t)) \quad (2.14)$$

$$T_j(t) = T_0 + \langle I(t)V(t) \rangle \cdot R_{th} \quad (2.15)$$

$$\Phi_1(t) = \min(V_{gs}(t)) \quad (2.16)$$

$$\Phi_2(t) = \max(V_{ds}(t)) \quad (2.17)$$

where $V_{gs}(t)$ and $V_{ds}(t)$ are the instantaneous intrinsic gate-source and drain-source terminal voltages; $T_j(t)$ is the dynamic junction temperature; and $\Phi_1(t)$ and $\Phi_2(t)$

are trap states associated with gate-lag and drain-lag, respectively. In this method, trapping effects are incorporated into the pure ANN-based drain current model as part of the large-signal model. The DynaFET model in [16] is trained using the large-signal measurement data in time domain.

This DynaFET model, as a pure ANN model, fits very well with the measurement data thanks to the ANN's great generalization and characterization capabilities. This ANN model is trained by a systematic computation flow, i.e., ANN training. The modeling technique in [16] is suitable for the GaN HEMT modeling when sufficient measurement data for training can be obtained. If the measurement data are not sufficient, the efficiency of pure ANN models may be compromised. The time-consuming issue in developing the aforementioned empirical function/equivalent circuit, and the data-hungry issue in developing the pure ANN model are major issues in developing accurate models for GaN HEMTs.

2.3.4 Discussion on Space Mapping Modeling Method

Space mapping is one of the methods that can be used to overcome the time-consuming issue in empirical function/equivalent circuit model development, and the data-hungry issue in pure ANN model development. Instead of starting from scratch, the model development usually begins with a best available model as the given coarse model. By developing mappings, the coarse model is mapped to match the fine device data. In space mapping technique, mappings are formulated by known mathematical equations, such as linear equations and ANN equations. Those known equations are determined systematically without the need for manual trial-

and-errors, speeding up the model development process. Moreover, space mapping can be applied to device modeling even when fewer measurement data are available, lowering the cost of data generation from the measurement. Realignment between the coarse model and the fine model (which costs less data) is relatively easier than creating the nonlinear model from scratch (which requires more data). To be realigned with the fine model, the coarse model is tuned by mappings which slightly “shift and/or scale” the inputs and/or outputs of the coarse model. More specifically, the values of the inputs are modified into new values. Similarly, the values of the outputs are modified into new values. By such input-input modification and such output-output modification, the coarse model is realigned with the fine model. The input-input and/or output-output relationships from these modifications are relatively simpler, if compared to the nonlinear input-output relationship from the coarse model. Therefore, less measurement data are needed to develop mappings for the characterization of those adjustments.

Conventionally, the space mapping technique for modeling nonlinear RF and microwave devices is employed to externally map the coarse model as a whole [43]. For simplicity, we call this technique external space mapping. The existing external space mapping modeling approach works well when the nonlinear devices being modeled are typical transistors, such as SiGe HBTs, GaAs MESFETs, and GaAs HEMTs. However, this external space mapping technique is not directly suitable for GaN HEMT model development. This is because that the behaviors, such as trapping effects and frequency dispersion in GaN HEMTs, are very different for different internal branches of the coarse model. It is hard for one external mapping

to simultaneously map different behaviors from outside.

The focus of first of part of this thesis is on developing a new and effective space mapping technique for GaN HEMT model development, addressing the time-consuming issue and the data-hungry issue.

2.4 Parametric Modeling Methods for EM Microwave Passive Components

The second topic of this thesis is the parametric modeling techniques for EM microwave passive components. Parametric modeling of EM behaviors for microwave passive components has become important for EM-based design in microwave area [44-46],[49],[50],[109-112]. EM-based design can be time-consuming, since repetitive simulations of EM are usually required due to value adjustments of geometrical parameters. Parametric models are developed to characterize the relationship between geometrical variables and EM responses. The developed parametric models allow faster simulation and optimization than EM simulations with varying values of geometrical parameters and can be subsequently implemented in a circuit simulator.

Various parametric modeling methods for EM microwave passive components have been investigated [44-46],[49],[50], such as the ANN modeling method, the KBNN modeling method, the pole-residue-based neuro-TF modeling method, and the rational-based neuro-TF modeling method.

2.4.1 ANN and KBNN Modeling Methods

ANN has also been an important vehicle for parametric modeling of EM behavior in radio frequency and microwave area [44-46],[49],[50],[109-112]. ANN benefits from its strong learning and generalization capabilities and it has been used for a wide variety of microwave applications [56-64],[66-77],[80]. The universal approximation theorem [78] of ANNs provides the theoretical foundation that if sufficient data are used in training, the good accuracy of ANN models can be achieved within the training region. To develop an accurate parametric model, ANNs are trained to learn the nonlinear relationship between geometrical variables and EM behavior.

Another parametric modeling approach is the knowledge-based approach. The knowledge-based neural network (KBNN) is a recognized knowledge-based approach for parametric modeling. In the KBNN approach, KBNN models are developed using neural networks combined with knowledge models such as analytical formulas [79], empirical functions [80], or equivalent circuit models [81],[82]. The knowledge models can help speed up model development and enhance the capability for learning and generalization of the KBNN models [83].

2.4.2 Pole-Residue-Based Neuro-TF Modeling Method

Another popular parametric modeling method is the neuro-transfer function (neuro-TF) method [45],[48],[49],[51], which combines neural networks and transfer functions. This approach can be employed even if accurate knowledge models are not available. In this method, EM responses of passive components versus frequency are represented by the transfer functions. Neural networks are trained to learn the

Figure 2.9: Structure of the pole-residue-based neuro-TF model [49].

relationship between the poles/residues (or coefficients) in the transfer functions and geometrical parameters.

One of the popular neuro-TF methods is the pole-residue-based neuro-TF method [49],[51]. Fig. 2.9 illustrates the structure of the pole-residue-based neuro-TF model [49]. The model consists of pole-residue-based transfer functions and neural networks. $\mathbf{x}$ represents the geometrical variables, $\mathbf{y}$ represents the model outputs, i.e., EM responses (e.g., S -parameters) of the microwave component, and $\mathbf{d}$ represents the outputs of the EM simulations. Two training processes are performed for the neuro-TF model, i.e., preliminary training of neural networks and the refinement training of the overall neuro-TF model.

This neuro-TF method can address the challenge of high-sensitivities of the overall model response with respect to the coefficients in the rational-based transfer functions for high-order application. This method can function well with small geometrical variations. However, with large geometrical variations, some poles and

residues obtained from vector fitting may not change continuously with respect to geometrical variables, resulting in the discontinuity issue. Along with the discontinuity issue in the poles/residues, the smoothness of the neighboring poles/residues with respect to the geometrical parameters are also affected, resulting in the non-smoothness issue. These issues lead to the difficulty of training for the overall model, decreasing the effectiveness of the method.

2.4.3 Rational-Based Neuro-TF Modeling Method

Rational-based neuro-TF method [45] is another popular neuro-TF method, which does not have the discontinuity issue of poles and residues.

Fig. 2.10 illustrates the structure of the rational-based neuro-TF model [45]. The model consists of rational-based transfer functions and neural networks. $\mathbf{x}$ represents the geometrical variables, $\mathbf{y}$ represents the model outputs, i.e., EM responses (e.g., S -parameters) of the microwave component, and $\mathbf{d}$ represents the outputs of the EM simulations. Similar to the pole-residue-based neuro-TF method, two training processes are performed for the rational-based neuro-TF model, i.e., preliminary training of neural networks and the refinement training of the overall neuro-TF model.

The rational-based neuro-TF works well when the order in the neuro-TF model is low. However, as the order increases, the transfer function response is very sensitive to the coefficients of the transfer function, leading to the high-sensitivity issue. The high-sensitivity issue can result in the difficulty of training for the overall model when the order in the neuro-TF model is high and the geometrical variations be-

Figure 2.10: Structure of the rational-based neuro-TF model [45].

comes large.

The focus of the second part of this thesis is on developing a new type of neuro-TF parametric modeling technique for EM microwave passive components, addressing the discontinuity issue and the associated non-smoothness issue when the geometrical variations become large. The focus of the third part of this thesis is on developing a new rational-based neuro-TF parametric modeling technique for EM microwave passive components, addressing the high-sensitivity issue when the order in the neuro-TF model is high and the geometrical variations become large.

2.5 Conclusion

In this chapter, a literature review of popularly used GaN HEMT modeling methods and parametric modeling methods for microwave passive components has been provided. First an overview of ANN modeling approaches and space mapping modeling

approaches has been presented as the general methods for device modeling. Existing GaN HEMT modeling approaches including the empirical function/equivalent circuit method, the physics-based compact modeling method, and the pure-ANN method have been reviewed and their relative advantages and disadvantages are discussed. The popular existing neuro-TF parametric modeling methods including the ANN method, the KBNN method, the pole-residue-based neuro-TF method and, the rational-based neuro-TF method have been reviewed and discussed. In the next chapter, a novel space mapping technique using decomposed mappings for GaN HEMT modeling with trapping effects is proposed.

Chapter 3

Space Mapping Technique Using Decomposed Mappings for GaN HEMT Modeling with Trapping Effects

3.1 Introduction

A novel space mapping modeling approach for gallium nitride (GaN) high-electron-mobility transistors (HEMTs) with trapping effects is presented in this chapter, advancing the space mapping technique for nonlinear device modeling. The existing space mapping modeling approach uses an external mapping to map an existing device model onto device data. When different branches inside the existing device model need to address very different behaviors, such as trapping effects and frequency dispersion in GaN HEMTs, it is hard for one external mapping to simultaneously map different behaviors. The proposed space mapping technique develops separate mappings for different branches, such that different behaviors can be mapped separately. Each mapping module is formulated to map a specific behav-

ior in the overall model. Each mapping module is developed through supervised learning methods to systematically overcome the gap between each internal branch and each set of target data, accelerating the process of model development. The proposed space mapping technique is a fast and systematic modeling approach, compared with the existing empirical function/equivalent circuit approach. Compared to the pure neural network modeling approach, the proposed space mapping technique employs less training data. Measurement data of a $2 \times 350 \mu\text{m}$ GaN HEMT device are employed for model training and verification. Good agreement can be achieved between the developed large-signal model and the measurement data including DC, pulsed I-V (PIV) at seven quiescent biases, S -parameters, and power characteristics. Reasonably close predictions of load-pull figures of merit are achieved by the developed model. The model development illustrated in the example shows that the proposed space mapping technique is a fast modeling approach to develop an accurate large-signal model for GaN HEMTs.

3.2 Proposed Space Mapping Technique for Development of a GaN HEMT Large-Signal Model

3.2.1 Proposed Mapping Topology for a Given GaN HEMT Large-Signal Model

The proposed mapping topology for a given GaN HEMT large-signal model (which is referenced to the intrinsic plane) is shown in Fig. 3.1. A commonly used intrinsic large-signal model topology consists of a gate current branch, a gate charge branch, a drain current branch, and a drain charge branch [36]. Let this model serve as a

coarse model. Let each internal branch of the coarse model be called a “coarse sub-model”. Let each mapped internal branch be called a “mapped coarse sub-model”. The “fine sub-model” is a fictitious sub-model implied by the device data from the measurement.

As can be seen from Fig. 3.1, the proposed mapping topology contains three mapping modules (within dash-dot line boxes) for three given coarse sub-models (within dashed line boxes). Suppose that the given coarse dynamic trapping model (coarse sub-model 1) cannot accurately represent the trapping effects in GaN HEMTs. More specifically, suppose that the given coarse dynamic trapping model (coarse sub-model 1) cannot accurately produce the effective voltages which are then used to replace the terminal voltages of the drain current model (sub-model 0) in order to characterize the trapping effects. To improve the accuracy of the overall dynamic trapping model (i.e., to obtain accurate effective voltages), mapping module 1 is applied to map the coarse dynamic trapping model (coarse sub-model 1). After mapping, the mapped coarse dynamic trapping model (mapped coarse sub-model 1) can accurately produce the effective voltages. With the accurate effective voltages as the inputs of the drain current model (sub-model 0), the output from the modified drain current model can accurately match the pulsed I-V (PIV) data for different quiescent biases, simultaneously. In this way, the mapping module 1 improves the accuracy of the overall dynamic trapping model. Suppose that the given coarse gate charge model (coarse sub-model 2) and the given coarse drain charge model (coarse sub-model 3) can characterize, yet not accurately the frequency dispersion in GaN HEMTs. The mapping module 2 and the mapping module 3 are

applied to map the coarse gate charge model and the coarse drain charge model, respectively. After mapping, the mapped coarse charge models (mapped coarse sub-model 2 and mapped coarse sub-model 3), the gate and drain current models, and the rest of the mapped models together will construct the mapped overall large-signal model. This mapped overall large-signal model can produce accurate S -parameter responses which will match the S -parameter data from the measurement. In this way, the mapping module 2 and the mapping module 3 together improve the accuracy of the overall charge model. In addition, the three given sub-circuits (within a dash-dot-dot box) at the bottom of Fig. 3.1 include a gate trapping sub-circuit in the bottom left, a drain trapping sub-circuit in the bottom right, and a thermal sub-circuit in the top left. The thermal sub-circuit in Fig. 3.1 is a first-order RC thermal sub-circuit [2] used to capture the dynamic variable T_j (junction temperature), for modeling the dynamic behavior of thermal effect. Trapping sub-circuits [2] in Fig. 3.1 are employed to extract dynamic variables ϕ_g and ϕ_d , i.e., trap states that are associated with gate lag and drain lag, respectively, for modeling the dynamic behavior of gate and drain trapping effects.

3.2.2 Trapping Model Development Using Space Mapping

Trapping effects are peculiar phenomena in GaN HEMTs, resulting in current collapse. The current collapse can be observed from the PIV curves at different quiescent biases. Trapping effects in GaN HEMT are caused by the charge trapping and releasing processes of the deep donors in the semi-insulating buffers. Trapping is a fast process of several nanoseconds while releasing is a slow process of microsec-

Figure 3.1: Proposed mapping topology for a given GaN HEMT large-signal model with different mapping modules. The mapping module 1 will be referred to as the trapping mappings. The mapping module 2 and the mapping module 3 will be referred to as the gate charge mapping and the drain charge mapping, respectively. The coarse sub-model 1 will be referred to as the coarse dynamic trapping model. The coarse sub-model 2 and the coarse sub-model 3 will be referred to as the coarse gate charge model and the coarse drain charge model, respectively.

ond level [6]. Trapping effects are dynamic, which can be observed when a GaN HEMT is pulsed up/down. Many dynamic trapping models are developed using equivalent circuits/empirical functions. In these existing techniques, the dynamic trapping model is often built upon a given drain current model. Let (V_{gQ}, V_{dQ}) represent the quiescent bias. Quiescent biases (V_{gQ}, V_{dQ}) and the dynamic variables ϕ_g and ϕ_d (extracted from the trapping sub-circuits) are often used as inputs of the existing dynamic trapping model. In this way, the existing dynamic trapping model with the drain current model together can characterize the different PIV curves at different quiescent biases as well as the two different transient behaviors resulting from trapping and releasing processes.

However, it is not always possible to accurately fit the entire PIV curves for many different quiescent biases at the same time for the existing dynamic trapping model with the given drain current model. For some GaN HEMT devices, the shapes of the PIV curves for different quiescent biases are no longer similar. Therefore, it becomes harder to use the circuits/equations that can accurately fit the PIV curves at one quiescent bias to represent the PIV curves at other different quiescent biases.

The starting point of the proposed technique for trapping model development is when the existing dynamic trapping model with the given drain current model cannot match the PIV data for many different quiescent biases. Let the given existing dynamic trapping models be called coarse dynamic trapping models. We propose trapping mappings (mapping module 1) for the coarse dynamic trapping models (coarse sub-model 1) to improve the accuracy of the overall dynamic trapping model. The trapping mappings are systematically developed through supervised

learning methods to learn trapping effects in GaN HEMTs so that the mapped coarse dynamic trapping models with the given drain current model can accurately fit the entire PIV curves for all the quiescent biases at the same time. Suppose that the drain current model and the existing dynamic trapping models (which are coarse) are both given. Suppose that the given drain current model (sub-model 0) is accurate for PIV (0 V, 0 V). The model is represented as

$$I_{ds}^{PIV} = f_2^I(v_{gse}, v_{dse}, T_j) \quad (3.1)$$

where v_{gse} and v_{dse} are the effective gate-source voltage and the effective drain-source voltage, respectively, T_j represents the junction temperature, and I_{ds}^{PIV} represents the drain current for different quiescent biases. Let v_{gsec} and v_{dsec} represent the coarse effective gate-source and drain-source voltages which are the outputs from the coarse dynamic trapping models. Let v_{gs_mp} and v_{ds_mp} be the outputs of the trapping mappings. After mapping, the v_{gsec} and v_{dsec} are mapped onto the v_{gse} and v_{dse} . In other words, the effective gate-source and drain-source voltages of low accuracy are mapped onto those of a high-accuracy version through mapping. The accurate v_{gse} and v_{dse} are then supplied to (3.1) to produce accurate I_{ds}^{PIV} that simultaneously fits the entire PIV curves for many different quiescent biases. The relationships between the trapping mappings and the coarse dynamic trapping models are formulated as

$$v_{gse} = v_{gsec} + v_{gs_mp} \quad (3.2)$$

$$v_{dse} = v_{dsec} + v_{ds_mp}. \quad (3.3)$$

The given coarse dynamic trapping models are expressed as

$$v_{gsec} = f_{c1}^{dyn-trp}(v_{gs}, v_{ds}, \phi_g, \phi_d, V_{gQ}, V_{dQ}) \quad (3.4)$$

and

$$v_{dsec} = f_{c2}^{dyn-trp}(v_{gs}, v_{ds}, \phi_g, \phi_d, V_{gQ}, V_{dQ}) \quad (3.5)$$

where v_{gs} and v_{ds} are the intrinsic gate-source and drain-source terminal voltages. In order to guarantee that the $f_{c1}^{dyn-trp}(\cdot)$ and the $f_{c2}^{dyn-trp}(\cdot)$ do not affect the drain current of the given drain current model $f_2^I(\cdot)$ under DC operation, the coarse dynamic trapping models are required to satisfy the following conditions:

$$V_{gQ} = f_{c1}^{dyn-trp}(V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}) \quad (3.6)$$

and

$$V_{dQ} = f_{c2}^{dyn-trp}(V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}). \quad (3.7)$$

The proposed trapping mappings can be formulated as

$$v_{gs_mp} = f_{mp1}^{trp}(v_{gs}, v_{ds}, \phi'_g, \phi'_d, V_{gQ}, V_{dQ}) \quad (3.8)$$

and

$$v_{ds_mp} = f_{mp2}^{trp}(v_{gs}, v_{ds}, \phi'_g, \phi'_d, V_{gQ}, V_{dQ}) \quad (3.9)$$

where ϕ'_g and ϕ'_d are expressed as

$$\phi'_g = \min\{v_{gs}, V_{gQ}\} \quad (3.10)$$

$$\phi'_d = \max\{v_{ds}, V_{dQ}\}. \quad (3.11)$$

Figure 3.2: Structure of the proposed trapping mappings for a given coarse dynamic trapping model. The whole model in this figure will be referred to as the mapped-trapping-enabled drain current model.

Similarly, to ensure that the $f_{mp1}^{trp}(\cdot)$ and the $f_{mp2}^{trp}(\cdot)$ do not affect the drain current of the given drain current model $f_2^I(\cdot)$ under DC operation, the proposed trapping mappings should satisfy the following conditions:

$$0 = f_{mp1}^{trp}(V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}) \quad (3.12)$$

and

$$0 = f_{mp2}^{trp}(V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}, V_{gQ}, V_{dQ}). \quad (3.13)$$

Since obtaining the equivalent circuit/empirical models for the trapping map-

pings are expensive by manual trial-and-errors, a supervised learning method is employed. The trapping mappings $f_{mp1}^{trp}(\cdot)$ and $f_{mp2}^{trp}(\cdot)$, as shown in (3.8) and (3.9), are formulated by ANN or other data-driven models, such as [114-116]. Such data-driven models can systematically characterize the gap between the coarse dynamic trapping models $f_{c1}^{dyn-trp}(\cdot)$ and $f_{c2}^{dyn-trp}(\cdot)$ (combined with the given drain current model $f_2^I(\cdot)$) and the PIV data for all the quiescent biases (V_{gQ}, V_{dQ}) . The trapping mappings are trained to learn the differences between the coarse dynamic trapping model (combined with the given drain current model) and the PIV data for all the quiescent biases (V_{gQ}, V_{dQ}) . After learning, the mapped coarse dynamic trapping models (combined with the given drain current model) can accurately fit the entire PIV curves for many different quiescent biases (V_{gQ}, V_{dQ}) at the same time. The formulation of the trapping mappings can be established by data-driven models, such as polynomial, ANNs, Kriging, etc. Based on a certain type of data-driven model, a supervised learning process systematically determines the specific mathematical equations with trained coefficients. The specific data-driven models chosen as the realization of the trapping mappings $f_{mp1}^{trp}(\cdot)$ and $f_{mp2}^{trp}(\cdot)$ will be illustrated for the GaN HEMT example in Subsection 3.3.2. After the trapping mappings are developed, they will be combined with the coarse dynamic trapping model and the given drain current model to construct the drain current model for different quiescent biases, according to Fig. 3.2. This constructed drain current model in Fig. 3.2 will be referred to as the mapped-trapping-enabled drain current model.

3.2.3 Charge Model Development Using Space Mapping

The GaN HEMTs are often required to operate at high frequency. At high frequency, the responses of the devices deviate from those at relatively low frequency. The starting point for the space mapping technique is when the existing charge models cannot match the frequency response of a new device simultaneously for the entire new (or enlarged) frequency range with good accuracy.

By developing suitable mappings on top of the coarse charge model, the mapped coarse charge model matches the fine charge model over the entire new (or enlarged) frequency range. As the frequency response pattern of the coarse charge model changes with frequency differently from that of the fine charge model, mappings with different values are needed for different frequencies. In this scenario, frequency information is required from the mappings.

Due to the lack of the frequency information in the mapping formulation, the conventional static (time-invariant) mapping [43] provides the same mapping values for different frequencies. As a result, the mapped coarse charge model using the conventional time-invariant mapping may not be accurate to represent the fine charge model over the entire frequency range. Conventional frequency-dependent mapping [55] has frequency information in its formulation, which is suitable for passive device modeling. However, such mapping is not suitable for nonlinear device modeling. The reason is that such mapping is not directly suitable for time-domain simulations, since the frequency is explicitly used as an input variable in the mapping. GaN HEMT models, like many other transistor models, are commonly evaluated in time domain rather than frequency domain for circuit simulations. As a result, the

existing method of explicit frequency-dependent mapping is not directly suitable for GaN HEMT intrinsic nonlinear modeling.

We propose a time-dependent space mapping approach to develop the overall charge model for GaN HEMTs. The time-dependent mapping in the proposed approach is directly suitable for transient/steady-state time-domain simulations, since frequency is not explicitly used as a variable in the mapping. Moreover, the proposed time-dependent mapping has rich frequency information (in frequency domain) even though frequency does not explicitly appear in the mapping. The proposed time-dependent mapping can produce different mapping values for different frequencies, becoming a suitable mapping to improve the overall charge model with the desired accuracy. The proposed modeling approach starts with the coarse charge model. Suppose that the given charge models are accurate in some frequency ranges, but are inaccurate in other frequency ranges. Let the given charge models be treated as the coarse charge models, which are represented as

$$Q_{gsc} = f_{c1}^Q(v_{gs}, v_{ds}) \quad (3.14)$$

$$Q_{dsc} = f_{c2}^Q(v_{gs}, v_{ds}) \quad (3.15)$$

where $f_{c1}^Q(\cdot)$ and $f_{c2}^Q(\cdot)$ represent the coarse gate charge model (coarse sub-model 2) and coarse drain charge model (coarse sub-model 3), respectively.

To enhance the accuracy of the overall charge model, two separate time-dependent mappings (mapping module 2 and mapping module 3) formulated by time delay-based models are proposed to map the coarse gate charge model and the coarse drain charge model, respectively. Let $e^{-j\omega\tau}$ represent time delay term in frequency do-

main, where ω is the frequency and τ is the time step. The time delay terms in the time delay-based models are the nonlinear functions of frequency in the frequency domain. Therefore, the two proposed time-dependent mappings are both nonlinear with respect to frequency when used in frequency domain. Moreover, the time delay terms with different delay times can produce frequency domain responses with different combinations of the nonlinearity with respect to frequency. By providing different mapping values for different frequencies to the coarse charge models, the two proposed time-dependent mappings have the capability to accurately overcome the gap between two different response-frequency patterns from the coarse charge models and the fine charge models. In this way, the proposed time-dependent space mapping approach can improve the overall charge model accuracy over the entire frequency range.

The general forms of the two time-dependent mappings (i.e., the gate charge mapping and the drain charge mapping) can be expressed as

$$Q_{gs_mp} = f_{mp1}^Q(v_{gs}(t), v_{gs}(t - \tau), \dots, v_{gs}(t - N_{d1}\tau), v_{ds}(t), v_{ds}(t - \tau), \dots, v_{ds}(t - N_{d1}\tau)) \quad (3.16)$$

$$Q_{ds_mp} = f_{mp2}^Q(v_{gs}(t), v_{gs}(t - \tau), \dots, v_{gs}(t - N_{d2}\tau), v_{ds}(t), v_{ds}(t - \tau), \dots, v_{ds}(t - N_{d2}\tau)) \quad (3.17)$$

where N_{d1} and N_{d2} represent the number of time delays for time-dependent mappings $f_{mp1}^Q(\cdot)$ and $f_{mp2}^Q(\cdot)$, respectively; and τ is the time step. The two proposed time-dependent mappings are both difference mappings to their corresponding coarse charge models. The relationships between the time-dependent mappings and

the coarse charge models are formulated as

$$Q_{gs} = f_1^Q(\cdot) = Q_{gsc} + Q_{gs_mp} \quad (3.18)$$

$$Q_{ds} = f_2^Q(\cdot) = Q_{dsc} + Q_{ds_mp}. \quad (3.19)$$

Since the coarse charge models already characterize the nonlinear behaviors of the charges to some degrees, the time delay-based models in the proposed time-dependent mappings $f_{mp1}^Q(\cdot)$ and $f_{mp2}^Q(\cdot)$ could be simplified. The two time-dependent mappings with simplified time delay-based models are given by

$$\begin{aligned} Q_{gs_mp} = f_{mp1}^Q(\cdot) &= \sum_{j=0}^{N_{d1}} (\alpha_{1j}(\tilde{v}_{gs}, \tilde{v}_{ds}) \cdot v_{gs}(t - j\tau)) \\ &+ \sum_{j=0}^{N_{d1}} (\alpha_{2j}(\tilde{v}_{gs}, \tilde{v}_{ds}) \cdot v_{ds}(t - j\tau)) \end{aligned} \quad (3.20)$$

and

$$\begin{aligned} Q_{ds_mp} = f_{mp2}^Q(\cdot) &= \sum_{j=0}^{N_{d2}} (\alpha_{3j}(\tilde{v}_{gs}, \tilde{v}_{ds}) \cdot v_{gs}(t - j\tau)) \\ &+ \sum_{j=0}^{N_{d2}} (\alpha_{4j}(\tilde{v}_{gs}, \tilde{v}_{ds}) \cdot v_{ds}(t - j\tau)). \end{aligned} \quad (3.21)$$

Since the nonlinear relationship between response and frequency is bias-dependent, the coefficients α_{1j} , α_{2j} , α_{3j} , and α_{4j} in (3.20) and (3.21) should be different for different biases. Therefore, the coefficients α_{1j} , α_{2j} , α_{3j} , and α_{4j} should be functions of bias. Since these coefficient functions are bias-dependent, they can characterize different nonlinear relationships between response and frequency for different biases. Moreover, since the pulsed S -parameter data from measurements are measured for many different pulsed biases at each quiescent bias, the biases that are swept in the

S -parameter simulations should be pulsed biases at each user given quiescent bias. In this scenario, the coefficients α_{1j} , α_{2j} , α_{3j} , and α_{4j} should be functions of pulsed bias instead of quiescent bias. In order to represent the pulsed bias for the coefficient functions in S -parameter simulations, dynamic bias $(\tilde{v}_{gs}, \tilde{v}_{ds})$ is introduced. The dynamic biases $\tilde{v}_{gs}$ and $\tilde{v}_{ds}$ are the DC coefficients of the discrete Fourier transform of the terminal signals v_{gs} and v_{ds} with a period of the fundamental signal as the fundamental period in the discrete Fourier transform. In other words, the $\tilde{v}_{gs}$ and $\tilde{v}_{ds}$ are the average values of the terminal signals v_{gs} and v_{ds} , respectively, within a period of the fundamental signal in time domain. The dynamic biases are slow changing signals that can be extracted from the filtered terminal signals v_{gs} and v_{ds} with a low-pass filter. In practice, the coefficients α_{1j} , α_{2j} , α_{3j} , and α_{4j} become functions of dynamic biases $\tilde{v}_{gs}$ and $\tilde{v}_{ds}$.

The time-dependent mappings $f_{mp1}^Q(\cdot)$ and $f_{mp2}^Q(\cdot)$ formulated by (3.20) and (3.21) have the capability to accurately characterize the nonlinear relationships between response and frequency exhibited in the GaN HEMTs. Therefore, the proposed time-dependent space mapping technique can improve the accuracy of characterizing frequency dispersion in the GaN HEMT devices. The dynamic information is introduced by the histories $v_{gs}(t - \tau), \dots, v_{gs}(t - N_d\tau)$ and $v_{ds}(t - \tau), \dots, v_{ds}(t - N_d\tau)$, and the nonlinear characteristic is characterized by the coefficient functions $\alpha_{1j}(\cdot)$, $\alpha_{2j}(\cdot)$, $\alpha_{3j}(\cdot)$, and $\alpha_{4j}(\cdot)$. The coefficient functions are nonlinear models that can be developed accurately and systematically by supervised learning methods, such as ANNs and Kriging method. The number of the time delays determines the degree to which the model is needed to sufficiently capture the complicated dynamic be-

haviors of the GaN HEMT devices. More complicated dynamic behaviors in general needs to be addressed with more delays. Therefore, the number of the time delays needs to be appropriately selected to adequately depict the dynamic behavior of the GaN HEMT devices.

Further, the training data for the coefficient functions $\alpha_{1j}(\cdot)$, $\alpha_{2j}(\cdot)$, $\alpha_{3j}(\cdot)$, and $\alpha_{4j}(\cdot)$ can be obtained by solving a set of linear equations. Let N_d represent the number of time delays, and let N_f represent the number of frequency points. The set of linear equations is formulated as

$$\mathbf{T}\mathbf{A}^T = \mathbf{B} \quad (3.22)$$

where $\mathbf{T}$ is a $(2N_f) \times (N_d + 1)$ matrix, $\mathbf{A}$ is a $4 \times (N_d + 1)$ matrix, and $\mathbf{B}$ is a $(2N_f) \times 4$ matrix. Let $t_{k\ell}$ be an element of the $\mathbf{T}$ matrix. The value of the $t_{k\ell}$ is calculated as

$$t_{k\ell} = \begin{cases} \omega_k \cdot \sin(\omega_k(\ell - 1)\tau), & \text{if } 1 \leq k \leq N_f \\ \omega_{k-N_f} \cdot \cos(\omega_{k-N_f}(\ell - 1)\tau), & \text{if } N_f < k \leq 2N_f \end{cases} \quad (3.23)$$

where $k = 1, 2, \dots, 2N_f$, $\ell = 1, 2, \dots, N_d + 1$, and ω_k represents the frequency at k -th frequency sample point. In order to explain the calculation of the $\mathbf{B}$ matrix, we define the following symbols. Suppose that $\mathbf{Y}_d$ represents the Y -parameter data which are converted from the pulsed S -parameter data at $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V})$. Suppose that $\mathbf{Y}_c$ represents the Y -parameters from the model which includes the given gate current model, the developed mapped-trapping-enabled drain current model, and the given coarse gate and drain charge models. Let $Y_{11_d}^R(\omega_k)$,

$Y_{12_d}^R(\omega_k)$, $Y_{21_d}^R(\omega_k)$, and $Y_{22_d}^R(\omega_k)$ represent the real parts of the $\mathbf{Y}_d$ at k -th frequency point ω_k . Let $Y_{11_d}^I(\omega_k)$, $Y_{12_d}^I(\omega_k)$, $Y_{21_d}^I(\omega_k)$, and $Y_{22_d}^I(\omega_k)$ represent the imaginary parts of the $\mathbf{Y}_d$ at frequency point ω_k . Similarly, let $Y_{11_c}^R(\omega_k)$, $Y_{12_c}^R(\omega_k)$, $Y_{21_c}^R(\omega_k)$, and $Y_{22_c}^R(\omega_k)$ represent the real parts of the $\mathbf{Y}_c$ at frequency point ω_k . Let $Y_{11_c}^I(\omega_k)$, $Y_{12_c}^I(\omega_k)$, $Y_{21_c}^I(\omega_k)$, and $Y_{22_c}^I(\omega_k)$ represent the imaginary parts of the $\mathbf{Y}_c$ at frequency point ω_k . Let $b_{k\ell}$ be an element of the $\mathbf{B}$ matrix. The value of the $b_{k\ell}$ is calculated as

$$b_{k\ell} = \begin{cases} Y_{11_d}^R(\omega_k) - Y_{11_c}^R(\omega_k), & \text{if } 1 \leq k \leq N_f \text{ and } \ell = 1 \\ Y_{11_d}^I(\omega_{k-N_f}) - Y_{11_c}^I(\omega_{k-N_f}), & \text{if } N_f < k \leq 2N_f \\ & \text{and } \ell = 1 \\ Y_{12_d}^R(\omega_k) - Y_{12_c}^R(\omega_k), & \text{if } 1 \leq k \leq N_f \text{ and } \ell = 2 \\ Y_{12_d}^I(\omega_{k-N_f}) - Y_{12_c}^I(\omega_{k-N_f}), & \text{if } N_f < k \leq 2N_f \\ & \text{and } \ell = 2 \\ Y_{21_d}^R(\omega_k) - Y_{21_c}^R(\omega_k), & \text{if } 1 \leq k \leq N_f \text{ and } \ell = 3 \\ Y_{21_d}^I(\omega_{k-N_f}) - Y_{21_c}^I(\omega_{k-N_f}), & \text{if } N_f < k \leq 2N_f \\ & \text{and } \ell = 3 \\ Y_{22_d}^R(\omega_k) - Y_{22_c}^R(\omega_k), & \text{if } 1 \leq k \leq N_f \text{ and } \ell = 4 \\ Y_{22_d}^I(\omega_{k-N_f}) - Y_{22_c}^I(\omega_{k-N_f}), & \text{if } N_f < k \leq 2N_f \\ & \text{and } \ell = 4 \end{cases} \quad (3.24)$$

where $k = 1, 2, \dots, 2N_f$, and $\ell = 1, 2, 3, 4$.

The $\mathbf{A}$ matrix is the solution of the set of linear equations (3.22). The $\mathbf{A}$ matrix is used as the training data for the coefficient functions $\alpha_{1j}(\cdot)$, $\alpha_{2j}(\cdot)$, $\alpha_{3j}(\cdot)$, and $\alpha_{4j}(\cdot)$.

3.2.4 Development of Drain Current Model Using Knowledge-based Neural Network

The nonlinear drain current model (sub-model 0) used in (3.1) characterizes the behavior of the drain current I_{ds} for GaN HEMT devices at PIV (0 V, 0 V) with thermal effects. The nonlinear drain current model can be represented by

$$I_{ds} = f_2^I(v_{gs}, v_{ds}, T_j) \quad (3.25)$$

where T_j is the junction temperature.

We propose to use the knowledge-based neural network (KBNN) [94], [50] concept to develop a KBNN drain current model to represent the drain current of a GaN HEMT at PIV (0 V, 0 V). Its detailed structure is shown in Fig. 3.3. Here we define two v_{gs} regions: low v_{gs} region is a region where the measurement data of drain current needs to be fitted smoothly and the model should be physically meaningful. High v_{gs} region is a region where the measurement data of drain current needs to be fitted both accurately and smoothly. The model in low v_{gs} region does not have to be very accurate, since the measurement data of drain current in low v_{gs} region are much lower than that in high v_{gs} region. We define switching point v_{gs_swc} as a v_{gs} point to divide the low and high v_{gs} regions.

Suppose that the available knowledge model is not accurate enough yet smooth.

Figure 3.3: Proposed knowledge-based neural network model for drain current at PIV (0 V, 0 V).

Let this model be the coarse drain current model for PIV (0 V, 0 V). We use the KBNN model to realize the drain current model $f_2^I(\cdot)$ in (3.25). The formulation of the proposed KBNN drain current model for PIV (0 V, 0 V) is

$$I_{ds} = f_{KBNN}^I(v_{gs}, v_{ds}, T_j) = I_{dsc} \cdot (1 - W) + I_{ds_ANN} \cdot W \quad (3.26)$$

where $f_{KBNN}^I(\cdot)$ represents the proposed KBNN drain current model for PIV (0 V, 0 V), I_{ds} represents the drain current at PIV (0 V, 0 V), I_{dsc} represents the drain current at PIV (0 V, 0 V) from the coarse drain current model, I_{ds_ANN} represents the drain current at PIV (0 V, 0 V) from the ANN model, and W represents the weighting function. The ANN model is combined with the coarse drain current model by the weighting function to achieve good model smoothness.

The general equations for the coarse drain current model, the ANN model, and

the weighting function are expressed as

$$I_{dsc} = f_c^I(v_{gs}, v_{ds}, T_j) \quad (3.27)$$

$$I_{ds_ANN} = f_{ANN}^I(v_{gs}, v_{ds}, T_j, \mathbf{p}_w) \quad (3.28)$$

$$W = f_{swc}(v_{gs}, v_{gs_swc}) \quad (3.29)$$

where $f_c^I(\cdot)$ represents the coarse drain current model, $f_{ANN}^I(\cdot)$ represents the ANN model, and $\mathbf{p}_w$ denote the synaptic parameters of the neural networks. The weighting function W can be expressed by a smooth switch function f_{swc} . The coarse drain current model in (3.27) is good for the low v_{gs} region with good smoothness while the ANN model in (3.28) is good for the high v_{gs} region with good accuracy. Using the weighting function W , the coarse drain current model in (3.27) is emphasized in the low v_{gs} region while the ANN model in (3.28) is emphasized in the high v_{gs} region. The proposed KBNN drain current model takes the advantages of accuracy of the ANN model in the high v_{gs} region and smoothness of coarse drain current model in the low v_{gs} region.

3.2.5 Stepwise Algorithm for the Overall Model Development

Suppose that an existing large-signal model at the intrinsic plane for GaN HEMTs is available. Suppose that the measurement data used for model development are PIV data at different temperatures, DC data, PIV data at different quiescent biases, pulsed S -parameter data, large-signal data, and load-pull data.

The computational steps for implementing the proposed space mapping tech-

nique using decomposed mappings can be summarized as follows.

Step 0: Compare the available large-signal model with the measurement data. If the available model can simultaneously match the DC data, the PIV data at different quiescent biases, the pulsed S -parameter data, the large-signal data, and the load-pull data, the available model is satisfactory. In this scenario, exit without going to *Step 1* and there is no need to proceed to the proposed space mapping. Otherwise, if the available model cannot simultaneously fit all these data, proceed to the proposed space mapping technique by going to *Step 1*.

Step 1: Let this available model be called a coarse model. Decompose the given coarse model into coarse sub-models. The decomposed coarse sub-models are: the coarse drain current model $f_c^I(\cdot)$ of (3.27), the coarse dynamic trapping models $f_{c1}^{dyn-trp}(\cdot)$ of (3.4) and $f_{c2}^{dyn-trp}(\cdot)$ of (3.5), the gate current model, the coarse gate charge model $f_{c1}^Q(\cdot)$ of (3.14), and the coarse drain charge model $f_{c2}^Q(\cdot)$ of (3.15).

Step 2: Check the error between the coarse drain current model $f_c^I(\cdot)$ and the PIV (0 V, 0 V) data at different temperatures. If the error is small enough, let this model be used as the drain current model $f_2^I(\cdot)$ of (3.25), and go to *Step 6*. Otherwise, go to *Step 3*.

Step 3: Train the ANN model $f_{ANN}^I(\cdot)$ from (3.28) to learn the drain current in the PIV (0 V, 0 V) data for different pulsed biases at different temperatures. If the training error satisfies the stop criteria, go to *Step 4*. Otherwise, adjust the number of hidden neurons and repeat *Step 3*.

- Step 4:* Check the smoothness of the trained ANN model $f_{ANN}^I(\cdot)$ in the low v_{gs} region. If the trained ANN model $f_{ANN}^I(\cdot)$ is smooth, let this model be used as the drain current model $f_2^I(\cdot)$ of (3.25) and go to *Step 6*. Otherwise, determine the switching point v_{gs_swc} between the low v_{gs} region and the high v_{gs} region. Formulate the weighting function W from (3.29).
- Step 5:* Combine the trained ANN model $f_{ANN}^I(\cdot)$ and the coarse drain current model $f_c^I(\cdot)$ using the weighting function W to construct the KBNN model $f_{KBNN}^I(\cdot)$ from (3.26). This combination is based on Fig. 3.3. Let the KBNN model $f_{KBNN}^I(\cdot)$ be used as the drain current model $f_2^I(\cdot)$ of (3.25).
- Step 6:* Use the DC data to verify the developed drain current model $f_2^I(\cdot)$.
- Step 7:* Divide the PIV data into training data and testing data, with each type of data covering all the quiescent biases. Let n^{PIV} be the number of the PIV training data. Let $v_{gs_mp}^d$ and $v_{ds_mp}^d$ be variables. Replace the v_{gs} and v_{ds} in the developed drain current model $f_2^I(v_{gs}, v_{ds}, T_j)$ by $v_{gs_mp}^d + v_{gsec}$ and $v_{ds_mp}^d + v_{dsec}$, respectively. Optimize $v_{gs_mp}^d$ and $v_{ds_mp}^d$ such that the modified drain current model $f_2^I(v_{gs_mp}^d + v_{gsec}, v_{ds_mp}^d + v_{dsec}, T_j)$ matches the PIV training data at different quiescent biases. Let $\mathbf{v}_{gs_mp}^d$ and $\mathbf{v}_{ds_mp}^d$ be vectors of size n^{PIV} , containing the resulting solutions of $v_{gs_mp}^d$ and $v_{ds_mp}^d$, respectively, for all the n^{PIV} training samples.
- Step 8:* Use the data $\mathbf{v}_{gs_mp}^d$ and $\mathbf{v}_{ds_mp}^d$ to train the trapping mappings $f_{mp1}^{trp}(\cdot)$ from (3.8) and $f_{mp2}^{trp}(\cdot)$ from (3.9). If the training error satisfies the stop criteria, go to *Step 9*. Otherwise, adjust the structures of the ANN or other data-driven models (which are the trapping mappings $f_{mp1}^{trp}(\cdot)$ and $f_{mp2}^{trp}(\cdot)$),

such as the number of hidden neurons in the ANN model, or the order of the polynomials. Repeat *Step 8*.

Step 9: Combine the developed f_{mp1}^{trp} and f_{mp2}^{trp} , the given $f_{c1}^{dyn-trp}(\cdot)$ and $f_{c2}^{dyn-trp}(\cdot)$, and the developed $f_2^I(\cdot)$ to build the mapped-trapping-enabled drain current model. This combination is based on Fig. 3.2.

Step 10: Use the PIV test data to verify the developed mapped-trapping-enabled drain current model for different quiescent biases.

Step 11: Convert the pulsed S -parameter data at $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V})$ into the Y -parameter data $\mathbf{Y}_d$. Calculate the Y -parameter $\mathbf{Y}_c$ from the entire original large-signal model where the $f_c^I(\cdot)$, the $f_{c1}^{dyn-trp}(\cdot)$, and the $f_{c2}^{dyn-trp}(\cdot)$ together are replaced by the developed mapped-trapping-enabled drain current model. Compute the $\mathbf{B}$ matrix with the $\mathbf{Y}_d$ and the $\mathbf{Y}_c$ by (3.24).

Step 12: Set the number of time delays N_d in the charge mappings $f_{mp1}^Q(\cdot)$ from (3.20) and $f_{mp2}^Q(\cdot)$ from (3.21). Calculate the $\mathbf{T}$ matrix by (3.23). Solve the set of linear equations (3.22) to get the $\mathbf{A}$ matrix.

Step 13: Use the $\mathbf{A}$ matrix as the data to train the coefficient functions $\alpha_{1j}(\cdot)$, $\alpha_{2j}(\cdot)$, $\alpha_{3j}(\cdot)$, and $\alpha_{4j}(\cdot)$, which are shown in (3.20) and (3.21). If the training error satisfies the stop criteria, go to *Step 14*. Otherwise, adjust the structures of the ANN or other data-driven models (which are the coefficient functions), such as the number of hidden neurons in the ANN model, or the number of kernels in the Kriging model. Repeat *Step 13*.

Step 14: Use the trained coefficient functions to obtain charge mappings $f_{mp1}^Q(\cdot)$ and $f_{mp2}^Q(\cdot)$ by (3.20) and (3.21). Combine the obtained $f_{mp1}^Q(\cdot)$ and $f_{mp2}^Q(\cdot)$,

and the given coarse charge models $f_{c1}^Q(\cdot)$ and $f_{c2}^Q(\cdot)$ to construct the overall charge model which consists of $f_1^Q(\cdot)$ and $f_2^Q(\cdot)$ by (3.18) and (3.19). Combine the constructed overall charge model, the given gate current model, and the developed mapped-trapping-enabled drain current model to obtain the final overall large-signal model.

Step 15: Use the pulsed S -parameter test data to verify the developed final overall large-signal model.

Step 16: Use the DC data and the PIV test data to verify the developed final overall large-signal model.

Step 17: Perform HB simulations for the final overall large-signal model to investigate its power characteristics, such as output power, power-added efficiency, gain, and output power at various harmonics. Compare the power characteristics from the model with that from the large-signal data. Perform HB simulations for the model, driven beyond 3 dB gain compression points to investigate its load-pull figures of merit. Compare the load-pull figures of merit from the model with that from the load-pull data. In this way, we verify the final overall large-signal model with the large-signal data and the load-pull data.

Fig. 3.4 shows the flow diagram of the proposed space mapping technique for GaN HEMT model development.

Figure 3.4: Flow diagram of the proposed space mapping technique, showing the steps for developing decomposed mappings for various sub-modules and the steps for the overall model development.

3.2.6 Discussions

The proposed space mapping focuses on the intrinsic model development and the technique is applied after the extrinsic extraction. It is possible to use the proposed technique if the de-embedding is achieved using different extrinsic topologies. Besides, the mapped model is temperature-dependent. For temperature-dependent measurement data, the de-embedded data are also temperature-dependent. It is also possible to use the proposed technique if the de-embedding is achieved using the extrinsic topologies with temperature-dependent parameters such as the temperature-dependent R_s and R_d .

As additional information for the decomposed mappings, the mapping spaces for different mapping modules are shown in Table 3.1. The decomposed mappings are implicit mappings and they are used to map the “response spaces”. The mapping spaces for the decomposed mappings are the transistor internal electrical voltage and charge spaces (equivalent to the “response spaces” in the space mapping terminology). The decomposed mappings are difference mappings which can be categorized as output space mapping structure with nonlinear mapping functions. The input variables for the mapping functions include the inputs of the coarse sub-models as well as other variables from the equivalent circuit models. Because of the use of other variables (in addition to the input variables of the coarse sub-models) as the inputs to the mapping functions, the decomposed mappings can be considered as implicit mappings.

In this chapter, we use output difference mappings in an addition form to formulate the trapping mappings and to express the relationships between the trapping

Table 3.1: Mapping Spaces for Different Mapping Modules

	Coarse Output Space	Fine Output Space	Input Variables in the Mapping Functions
Mapping Module 1 (Trapping Mapping)	(v_{gsec}, v_{dsec})	(v_{gse}, v_{dse})	$(v_{gs}, v_{ds}, \phi'_g, \phi'_d, V_{gQ}, V_{dQ})$
Mapping Module 2 (Charge Mapping)	Q_{gsc}	Q_{gs}	$(v_{gs}(t), v_{gs}(t - \tau), \dots, v_{ds}(t), v_{ds}(t - \tau), \dots)$
Mapping Module 3 (Charge Mapping)	Q_{dsc}	Q_{ds}	$(v_{gs}(t), v_{gs}(t - \tau), \dots, v_{ds}(t), v_{ds}(t - \tau), \dots)$

mappings and the coarse dynamic trapping models. It is possible to select either the input mapping [43] or output difference mapping (this chapter) for the decomposed mapping. We used output difference mapping in this chapter because it is simpler for training and leads to better model convergence.

Before starting training, we get the data from original devices. The amount of data for the proposed technique can be similar as that for empirical models. The advantages of the proposed technique are systematic process and fast speed of model development. If the available empirical model with optimized coefficients is already accurate enough relative to the data, there is no need to use the proposed technique.

The coarse model can be either the same or different for different GaN HEMTs. The coarse model [36] used in this chapter is applicable to a variety of transistor devices including GaN HEMTs. For different GaN HEMTs, we may use the same coarse model but may need to reoptimize the parameters of the coarse model before using the proposed technique. In another situation, if the given coarse model is different for different GaN HEMTs, the proposed decomposed mapping approach is

still applicable, but the mapping formulas have to be adjusted to be applied to new branches in the new coarse model.

The starting point of the proposed space mapping technique is when the coarse model (i.e., the knowledge) is available and is not accurate enough. The trainings are applied to multiple mapping modules. Normally, more mappings need more data. However, in our proposed technique, one complex mapping is decomposed into multiple simpler mapping modules. In this scenario, multiple simpler mapping modules may not need much more data than those needed in one complex mapping.

The mappings for coarse sub-models are trained by supervised learning methods. The objective for training the ANN or other data-driven models is to maximize the mapping accuracy. For example, the mapping ANN is trained such that the mapped model can best match the device data. This training process is iterative. After training is finished, the mapping does not need additional iterative updates.

A qualitative comparison of GaN HEMT model development between the proposed technique and several existing techniques is provided in Table 3.2. As explained from the table, the model development in the empirical function/equivalent circuit approach can be time-consuming, because it is a manual trial-and-error process to modify the model topologies and equations. By contrast, in the proposed technique, the model development is based on systematic computer training, which is much faster than the manual process. The computer-based training for the proposed technique is fast, because the decomposed mappings have simpler structures over a non-decomposed mapping. In the conventional space mapping technique, one single mapping for different behaviors can be too complicated to be trained

effectively for GaN HEMTs, resulting in limited model accuracy. By contrast, the proposed technique decomposes one mapping into separate mappings for separate internal branches. The decomposed mappings are much simpler, which can be trained more effectively, resulting in good model accuracy. In the pure ANN technique (which is a data-driven technique), the model development needs large amount of training data (e.g., hundreds of thousands [113]), because that the information of the entire nonlinear relationship needs to be provided by the data. By contrast, the proposed approach needs much less training data (e.g., several thousands) because of the availability of the knowledge of the device in the form of the coarse model. Other data-driven techniques such as the Kriging and SVM techniques [114-116] may have good generalization capability when training data are limited. However, the use of the SVM and Kriging techniques for GaN HEMT model development, especially for trapping effects, is not yet available in the existing literature.

Here we also provide a further explanation of the novelty of the proposed technique relative to the direct combination of the existing techniques. The direct combination of existing space mapping with supervised learning methods is to employ one mapping to externally map the existing device model (coarse model), and to employ supervised learning methods to develop this mapping. However, this standard space mapping technique is not directly suitable for GaN HEMT model development. This is because that widely different behaviors within the internal structures of the same device cannot be effectively addressed by a single mapping from outside. By contrast, the proposed technique goes internally into the coarse model to separate different behaviors in order to map the internal variables. Moreover, the

Table 3.2: Qualitative Comparison of GaN HEMT Model Development between the Proposed Space Mapping Technique and Several Existing Methods

	Model Development Time	Amount of Data for Model Development	Modeling Accuracy
Proposed Space Mapping Technique	Computer training time (Normally hours/a few days)	Small/Moderate, because of the knowledge of the coarse model	Good
Empirical Function/Equivalent Circuit Development	Time for manual trial-and-error process (Normally days/weeks)	Small/Moderate, because of the human knowledge of the device	Acceptable
Pure ANN Technique (E.g., DynaFET [2])	Computer training time (Normally hours/days)	Large Amount	High
Existing Space Mapping Technique	Computer training time (Normally hours/a few days)	Small/Moderate, because of the knowledge of the coarse model	Limited accuracy for GaN HEMT
SVM Technique	Not yet available in the existing literature for GaN HEMT	Moderate	Expected to be acceptable
Kriging Technique	Not yet available in the existing literature for GaN HEMT	Moderate	Expected to be acceptable

standard space mapping technique uses a single mapping to map all variables for different behaviors. By contrast, the proposed technique reformulates a complex mapping into multiple simpler mappings to separately map different subsets of internal variables of the coarse model. In this new way, different internal branches can be mapped according to different needs, leading to fast and effective mapping developments. Furthermore, among the multiple decomposed mappings, the training of some mappings are dependent on the training results of other mappings. Therefore, the proposed technique has a stepwise algorithm to systematically develop the multiple mappings. This is different from the standard space mapping technique, which trains all mapping variables at the same time. As can be seen in Fig. 3.1,

different mapping variables are chosen for different mapping modules.

A brute-force use of dynamic neuro-space mapping for GaN HEMTs is to employ a single dynamic mapping to externally map the coarse model. As a single mapping, dynamic mapping still cannot effectively map different behaviors in different internal parts of GaN HEMTs. For this reason, the brute-force dynamic neuro-space mapping for GaN HEMT modeling has similar limitations as the static space mapping of [43] in terms of mapping topology and mapping performance.

The main difference between the proposed space mapping modeling technique and the conventional supervised learning methods for modeling is explained as follows. The proposed space mapping is to find the relationship between coarse and fine models. By contrast, the conventional supervised learning method, e.g., the pure ANN technique, is to find the relationship between inputs and outputs of the fine model. Besides, the proposed space mapping modeling approach is a knowledge-based modeling approach and the starting point of the technique is the availability of the knowledge models (i.e., coarse models). By contrast, the conventional supervised learning method, e.g., the pure ANN technique, does not involve any knowledge models. In the proposed space mapping modeling approach, the mapping relationship between each coarse sub-model and each fine sub-model does not have available formulas. Therefore, ANN or other data-driven models are used to learn the mapping relationships. In this way, the supervised learning method is used as part of the mapping structures in the proposed space mapping technique.

The mapped-trapping-enabled drain current model for different quiescent biases is verified by PIV test data in *Step 10* of the stepwise algorithm in Subsection 3.2.5.

The algorithm continues if the test error satisfies accuracy requirement. However, if the test error can not satisfy accuracy requirement, the amount of PIV training data needs to be increased to perform the training in *Step 7*.

The validations of the decomposed mappings so far are conducted for the mapped-trapping-enabled drain current model and the final overall large-signal model. It is also possible to validate each developed mapping module using testing data. The testing data for each mapping module can be generated in the same way as the training data, which are processed from the measurement data such as the PIV data, pulsed S -parameter data, etc. After the validation of each mapping module, the DC data, PIV data, S -parameter data, large signal data, and load-pull data of the overall device are used to validate the overall model as the final validation. The final validation can be conducted as described in the stepwise algorithm in Subsection 3.2.5.

The coarse charge model described in the proposed technique so far consists of the Qgs model and the Qds model (e.g., [36] and [2]). The Qgs model combines the effects of both the gate-source charge and the gate-drain charge in physics. The Qds model includes the effects of the gate-drain charge in physics. The Qgs and Qds models can be validated by the imaginary part of Y -parameter data (converted from the measured S -parameter data). It is noticed that the Y -parameter data include the effects of both the gate-source capacitance and the gate-drain capacitance.

If the frequency range of the S -parameter data is increased, more time delay steps can be used in the input variables of the dynamic mappings during the charge model development, in order to accurately match the S -parameter data with enlarged

frequency range.

3.3 Model Training and Verification

In this chapter, measurement data of a $2 \times 350 \mu\text{m}$ GaN HEMT device from NXP Semiconductors N.V. are employed for the model training and verification. The extrinsic parasitic parameters of the GaN HEMT device are extracted using cold-FET method [117]. The topology used in the example, as shown in Fig. 3.5, is general for different technologies, and has been used for GaN technology, such as [6]. The extrinsic network for this example is shown in Fig. 3.5 based on the cold-FET method. The extrinsic parameter values in Fig. 3.5 are determined through optimization over a broad frequency range. Based on this extrinsic network in Fig. 3.5 with the determined extrinsic parameter values, the measured extrinsic data are de-embedded to obtain the intrinsic measurement data.

The proposed space mapping technique is employed to develop a large-signal model by using these intrinsic measurement data of the GaN HEMT device. The mapping modules are developed separately for the different coarse sub-models during model training, covering DC, PIV, and small-signal training. Large-signal measurement data and load-pull measurement data are used for further model verification. The model verifications are performed in *Keysight Advanced Design System (ADS)*.

Figure 3.5: Extrinsic large-signal model topology for GaN HEMTs [117]. This topology is used to de-embed the extrinsic effect from the extrinsic measurement data in order to get the intrinsic measurement data. The obtained intrinsic data are used for the development of the intrinsic model.

3.3.1 Development of Drain Current Model Using KBNN and Model Verification Using DC Data

The specific formulas of the coarse drain current model in (3.27) for drain current at PIV (0 V, 0 V) and the smooth switch function in (3.29) used in this example are

$$f_c^I(v_{gs}, v_{ds}, T_j) = \arctan(k_1 \cdot v_{ds} + \beta_1) \cdot \left(\frac{2}{\pi} \cdot \arctan(k_2 \cdot v_{gs} + \beta_2) + 1 \right) \cdot (1 - \alpha_0 \cdot (T_j - T_A)) \quad (3.30)$$

and

$$f_{swc}(v_{gs}, v_{gs_swc}) = \frac{1}{\pi} \cdot \arctan(k_0 \cdot (v_{gs} - v_{gs_swc})) + 0.5 \quad (3.31)$$

where k_1 , k_2 , β_1 , β_2 , and α_0 are coefficients to be determined through curve fitting, and k_0 is a user-given parameter. The ideal Shockley diode models [1] with thermal

consideration are used to model gate current I_{gs} .

The KBNN drain current model is trained based on Steps 3–5 with the measurement data of PIV (0 V, 0 V) at different temperatures. The training range of PIV (0 V, 0 V) measurement data is selected as follows: gate voltage v_{gs} from -7 V to -4 V with step size of 0.5 V and from -3.8 V to 2 V with step size of 0.2 V; drain voltage v_{ds} from 0 V to 50 V with step size of 3 V and junction temperature T_j at 25 °C, 75 °C and 125 °C. The testing range of DC measurement data is selected as follows: v_{gs} from -7 V to -4 V with step size of 0.5 V and from -3.8 V to -1 V with step size of 0.2 V; v_{ds} from 0 V to 50 V with step size of 3 V. The comparison of PIV (0 V, 0 V) results at 25 °C, 75 °C and 125 °C between measurement and the developed (KBNN) drain current model is shown in Fig. 3.6 (a), Fig. 3.6 (b), and Fig. 3.6 (c), respectively. For DC verification, the drain current I_{ds} fitting under DC simulation in *ADS* is shown in Fig. 3.7.

Furthermore, smooth DC IV curves with the good match in the low v_{gs} region are shown in Fig. 3.8 (a). As can be seen, each DC IV curve is smooth and physically meaningful in the low v_{gs} region. The transconductance g_m result of the developed drain current model at $v_{ds} = 10$ V under DC simulation is shown in Fig. 3.8 (b). This figure shows that the model is smooth when crossing the boundary between the low v_{gs} region and the high v_{gs} region. The smooth g_m connection around the switching point $v_{gs_swc} = -2.6$ V can be observed and there is no kink in the g_m curve. Smooth IV curves in both the low v_{gs} region ($v_{gs} \leq -2.6$ V) and the high v_{gs} region ($v_{gs} > -2.6$ V) are required for good convergence under large-signal simulations. The proposed KBNN modeling approach to develop the drain current

(a)

(b)

(c)

Figure 3.6: Measured (circle) and modeled (solid line) PIV (0 V, 0 V) results at different temperatures. (a) PIV (0 V, 0 V) at 25 °C. (b) PIV (0 V, 0 V) at 75 °C. (c) PIV (0 V, 0 V) at 125 °C. The measurement data in the figure are training data.

Figure 3.7: Measured (circle) and modeled (solid line) DC results.

model can produce IV curves with good smoothness in these two v_{gs} regions.

So far, the developed drain current model is capable of accurately capturing DC IV with thermal effect of the GaN HEMT device and remains smooth under DC simulation. Since the coarse drain current model is given and the ANN training is a fast and systematic process, the development of the (KBNN) drain current model for PIV (0 V, 0 V) is systematic and fast.

3.3.2 Trapping Model Development Using Space Mapping and Model Verification Using PIV Data

The coarse dynamic trapping models of (3.4) and (3.5) used in this example are given as

$$\begin{aligned}
 v_{gsec} &= f_{c1}^{dyn-trp}(v_{gs}, v_{ds}, \phi_g, \phi_d, V_{gQ}, V_{dQ}) \\
 &= v_{gs} + \alpha_g(V_{gQ}) \cdot (v_{gs} - \phi_g) \\
 &\quad + \alpha_d(V_{gQ}, V_{dQ}) \cdot (\phi_d - v_{ds})
 \end{aligned} \tag{3.32}$$

Figure 3.8: (a) Measured (circle) and modeled (solid line) DC results around the switching point $v_{gs_swc} = -2.6$ V. (b) Modeled (solid line) DC simulation result of g_m at $v_{ds} = 10$ V.

and

$$\begin{aligned}
 v_{dsec} &= f_{c2}^{dyn_trp}(v_{gs}, v_{ds}, \phi_g, \phi_d, V_{gQ}, V_{dQ}) \\
 &= v_{ds}
 \end{aligned} \tag{3.33}$$

where α_g and α_d are the coefficients of the dynamic terms $(v_{gs} - \phi_g)$ and $(v_{ds} - \phi_d)$, respectively. They reflect the degree to which the coarse gate effective voltage changes due to gate trapping effects and drain trapping effects.

To realize the equations (3.8) and (3.9) in Subsection 3.2.2, polynomial functions

are used as the specific formulation for the trapping mappings $f_{mp1}^{trp}(\cdot)$ from (3.8) and $f_{mp2}^{trp}(\cdot)$ from (3.9) in this example. For simplicity, following [1] where the coarse effective drain-source voltage v_{dsec} is the same as the terminal drain-source voltage v_{ds} in the coarse dynamic trapping models, we only map the gate voltage without mapping the drain voltage in this case. Therefore, the trapping mapping $f_{mp1}^{trp}(\cdot)$ is used, and the trapping mapping $f_{mp2}^{trp}(\cdot)$ is given as

$$\begin{aligned} v_{ds_mp} &= f_{mp2}^{trp}(v_{gs}, v_{ds}, \phi'_g, \phi'_d, V_{gQ}, V_{dQ}) \\ &= 0 \end{aligned} \quad (3.34)$$

Moreover, since gate trapping effect is not as remarkable as drain trapping effect, it can be characterized by α_g from the given coarse dynamic trapping model in (3.32). Therefore, the term ϕ'_g can be ignored, leading to the simplification of the formula. Now the inputs of the trapping mapping $f_{mp1}^{trp}(\cdot)$ is reduced to v_{gs} , v_{ds} , ϕ'_d , V_{gQ} and V_{dQ} after such simplification.

The formulation of mapping $f_{mp1}^{trp}(\cdot)$ in this example is given by

$$\begin{aligned} v_{gs_mp} &= f_{mp1}^{trp}(v_{gs}, v_{ds}, \phi'_d, V_{gQ}, V_{dQ}) = \\ &\sum_{i,j=0}^3 \lambda_{ij}(V_{gQ}, V_{dQ}) \cdot (v_{gs})^i \cdot (\phi'_d - v_{ds})^j, \quad i + j \leq 3 \end{aligned} \quad (3.35)$$

where the coefficient λ_{ij} is expressed as

$$\begin{aligned} \lambda_{ij}(V_{gQ}, V_{dQ}) &= k_{1_ij} \cdot V_{gQ} + k_{2_ij} \cdot V_{dQ} \\ &\quad + k_{3_ij} \cdot V_{gQ} \cdot V_{dQ} + k_{4_ij} \cdot V_{gQ}^2 + \beta_{ij}. \end{aligned} \quad (3.36)$$

The polynomial order in (3.35) in this example is set to the 3rd order.

The PIV measurement data in this example are split into training data and testing data for developing the trapping mappings. The training data contain 192 samples and the testing data contain 6578 samples. PIV measurements are performed when the GaN HEMT is biased at the seven quiescent bias points of $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V}), (-2.84 \text{ V}, 0 \text{ V}), (-2.82 \text{ V}, 28 \text{ V}), (-2.84 \text{ V}, 48 \text{ V}), (-5 \text{ V}, 0 \text{ V}), (-5 \text{ V}, 28 \text{ V}),$ and $(-5 \text{ V}, 48 \text{ V})$, resulting in seven sets of PIV data. Trapping effects mainly account for the different shapes between PIV curves at different quiescent biases (V_{gQ}, V_{dQ}) . The trapping mapping $f_{mp1}^{trp}(\cdot)$ is trained based on Steps 7–8 in Subsection 3.2.5, such that the mapped coarse dynamic trapping model (with the developed drain current model) fits the entire PIV curves for the seven target quiescent biases. The development of the trapping mapping $f_{mp1}^{trp}(\cdot)$ is fast and systematic, since the trapping mapping is developed by the training of the polynomial models systematically.

Following *Step 9* in Subsection 3.2.5, we developed the mapped-trapping-enabled drain current model. The developed model is simulated in *ADS* by the “pulsed IV FET tester” test bench at these seven target quiescent biases for PIV verification. The drain voltage v_{ds} is swept from 0 to 50 V with step size of 3 V, and the gate voltage v_{gs} is swept from -7 V to 2 V with step size of 0.2 V. The testing data consist of 6578 data samples from the PIV measurement data at the seven quiescent biases for PIV verification. Such testing data have never been used for training. Fig. 3.9 shows the PIV simulation results of the developed mapped-trapping-enabled drain current model at three of the seven quiescent biases, while the PIV results at other four quiescent biases are shown in Fig. 3.10. Good agreement between model

and measured testing data can be observed from each sub-figure. The developed mapped-trapping-enabled drain current model is capable of accurately modeling PIV curves at those seven quiescent biases. The good PIV fittings suggest that the proposed mapping can accurately overcome the gap between coarse dynamic trapping model and the PIV data. Moreover, since the development of the trapping mapping $f_{mp1}^{trp}(\cdot)$ is fast and systematic, and the coarse dynamic trapping model is given, the model development of the overall dynamic trapping model is systematic and fast.

3.3.3 Charge Model Development Using Space Mapping and Model Verification Using Pulsed S -parameter Data

The charge models $f_1^Q(\cdot)$ and $f_2^Q(\cdot)$ are developed by the proposed time-dependent space mapping technique. The given coarse charge models in (3.14) and (3.15) are developed from the ANN-based Root modeling approach [117]. In the Root model extraction, the conservative gate and drain charge data are obtained by integration of the Y -parameter data (which are converted from the pulsed intrinsic S -parameter measurement data). These gate and drain charge data are then approximated by ANNs, resulting in smooth and infinitely differentiable 2-D charge functions. The ANN-based Root charge models are formulated as

$$Q_{gsc} = f_{1c}^Q(v_{gs}, v_{ds}) = f_{r_ANN1}(v_{gs}, v_{ds}, \mathbf{w}_0) \quad (3.37)$$

$$Q_{dsc} = f_{2c}^Q(v_{gs}, v_{ds}) = f_{r_ANN2}(v_{gs}, v_{ds}, \mathbf{w}_0) \quad (3.38)$$

(a)

(b)

(c)

Figure 3.9: Measured (circle) and modeled (solid line) PIV results at various quiescent biases: (a) PIV (0 V, 0 V), (b) PIV (-2.84 V, 0 V), (c) PIV (-2.82 V, 28 V). Similar comparisons for other four quiescent biases are provided in Fig. 3.10.

Figure 3.10: Measured (circle) and modeled (solid line) PIV results at various quiescent biases: (a) PIV (-2.84 V, 48 V), (b) PIV (-5 V, 0 V), (c) PIV (-5 V, 28 V), and (d) PIV(-5 V, 48 V). This figure complements Fig. 3.9 by providing PIV results at the other four quiescent biases.

where $f_{r_ANN1}(\cdot)$ and $f_{r_ANN2}(\cdot)$ represent the ANN equations for the coarse gate charge model and the drain charge model, respectively, and $\mathbf{w}$ denote the weights of neural networks. The ANN-based Root charge models are directly learned from the charge data, which are extracted by Root modeling method at $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V})$.

The development of the proposed time-dependent mappings is a systematic and fast process. The time-dependent mappings employ the converted intrinsic Y-parameter data at the quiescent bias $(V_{gQ}, V_{dQ}) = (0 \text{ V}, 0 \text{ V})$ as the training data. The time-dependent mappings use time delay-based models to characterize the differences between the coarse charge models and the fine charge models. The coefficients α_{1j} , α_{2j} , α_{3j} , and α_{4j} in (3.20) and (3.21) are determined through linear regressions by (3.22). These coefficients are the functions of the dynamic bias $(\tilde{v}_{gs}, \tilde{v}_{ds})$. Then, these coefficient functions are developed accurately and systematically by Kriging method. The training of these coefficient functions by Kriging method is fast and the training time is around two minutes. The number of time delays N_d used for this example is six. Following *Step 14* in Subsection 3.2.5, we developed the charge mappings, the overall charge model, and the (final) overall large-signal model.

Fig. 3.11 (a) and Fig. 3.11 (b) show the charge surfaces of the developed gate charge model and the developed drain charge model, respectively. The smooth charge surfaces of the charge models indicate that the overall charge model is smooth and has the potential to converge robustly under large-signal simulations.

The pulsed S -parameter measurement data at the quiescent bias $(V_{gQ}, V_{dQ}) =$

Figure 3.11: The charge surfaces of the developed gate charge model and the developed drain charge model. (a) The gate charge surface Q_{gs} . (b) The drain charge surface Q_{ds} .

(0 V, 0 V) from 0.5 GHz up to 18 GHz are used for the model validation. The *ADS* *S*-parameter simulation results of the developed overall large-signal model versus the corresponding measurement data are shown in Fig. 3.12 - Fig. 3.18. These figures show the *S*-parameter simulation results at pulsed biases of (-5.00 V, 27.02 V), (-5.00 V, 48.01 V), (-2.80 V, 28.68 V), (-2.80 V, 46.72 V), (-0.01 V, 9.29 V), (0.99 V, 10.57 V), and (1.99 V, 5.43 V). Good agreement between model and data indicates that the overall large-signal model is capable of accurately characterizing

the small-signal behaviors of the GaN HEMT device. The model development of the overall charge model is systematic and fast.

Figure 3.12: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-5.00 V, 48.01 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -5.00 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 48.01 V.

Further, we illustrate the need of utilizing time delay terms in the mapping formulation to improve the accuracy of the overall charge model over the entire frequency range by showing the following comparisons. The comparisons are the Y -parameter responses between the proposed time-dependent space mapping (SM)-based charge model and the static SM-based charge model. The mappings in the static SM-based charge model [43] are static voltage mappings, where terminal voltages v_{gs} and v_{ds} are modified to different values v_{gc} and v_{dc} , which are then supplied to the inputs of the given coarse charge models. Fig. 3.19, Fig. 3.20, and Fig. 3.21 show the comparisons of Y -parameters between the measurement

Figure 3.13: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-5.00 V, 27.02 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -5.00 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 27.02 V.

data, the proposed time-dependent SM-based charge model, and the static SM-based charge model at pulsed biases of (-5.00 V, 27.02 V), (1.99 V, 5.43 V), (-5.00 V, 48.01 V), (0.99 V, 10.57 V), (-2.80 V, 28.68 V), and (-0.01 V, 9.29 V).

As can be seen from Fig. 3.19, Fig. 3.20, and Fig. 3.21, the static SM-based charge model is linear and monotonic, which cannot accurately fit the measurement data over the entire frequency range, while the proposed time-dependent SM-based charge model can match the measurement data well over the entire frequency range. The given coarse charge model, which is linear and monotonic with respect to frequency in the example, produces a different pattern of Y -parameter response from that of the fine charge model over the entire frequency range. These figures indicate that the proposed time-dependent SM-based charge model can produce Y -

Figure 3.14: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-2.80 V, 28.68 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -2.80 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 28.68 V.

parameter response pattern which changes with frequency exactly the same way as the fine charge model, even though the Y -parameter response pattern of the coarse charge model which changes with frequency is different from that of the fine charge model. Therefore, the proposed time-dependent space mapping modeling approach can be utilized to improve the accuracy of the overall charge model and to obtain accurate characterization of the frequency dispersion in GaN HEMTs.

Next, we use the DC data and the PIV test data to recheck the DC and PIV performances of the overall large-signal model developed using the proposed space mapping technique. We confirmed that the DC results of the overall large-signal model are consistent with those of the developed drain current model. The overall large-signal model matches the DC data in the same way as Fig. 3.7 and Fig. 3.8.

Figure 3.15: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-2.80 V, 46.72 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -2.80 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 46.72 V.

Since the coarse dynamic trapping models and the trapping mappings used in this example have satisfied the conditions in (3.6), (3.7), (3.12), and (3.13), the mapped coarse dynamic trapping models do not affect the DC performance of the developed drain current model under DC operation. We also confirmed that the PIV results of the overall large-signal model are consistent with those of the mapped-trapping-enabled drain current model. The overall large-signal model matches the PIV data in the same way as Fig. 3.9. Since the overall large-signal model accurately matches the S -parameter data, the developed overall charge model accurately represents the nonlinear charge behavior of the GaN HEMT. Since the capacitance from the overall charge model (similar as that from the GaN HEMT) in the example is much less than a nanofarad, the drain current will finish transient quickly at the beginning

Figure 3.16: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (-0.01 V, 9.29 V). The gate voltage is pulsed from quiescent point of 0 V down to pulsed point of -0.01 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 9.29 V.

of the pulse period of several microseconds, and will remain in steady-state for substantially most of the time of the pulse period. Therefore, the overall charge model does not affect the PIV performance of the developed mapped-trapping-enabled drain current model under PIV operation.

3.3.4 Overall Model Verification Using Large-Signal Data

On-wafer load-pull data are measured under *Active Load-pull System (ALPS)* with different load impedances for the GaN HEMT device. These measurement data are employed to further verify the large-signal performances of the overall large-signal model developed using the proposed space mapping technique. The GaN HEMT device operates at frequency $f = 3.6$ GHz under two Class AB bias conditions:

Figure 3.17: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (0.99 V, 10.57 V). The gate voltage is pulsed from quiescent point of 0 V up to pulsed point of 0.99 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 10.57 V.

1) $I_{dq} = 10.5$ mA, $V_{dQ} = 28$ V; and 2) $I_{dq} = 10.5$ mA, $V_{dQ} = 48$ V. The overall large-signal model is simulated in *ADS* with different load impedances, under the aforementioned two Class AB bias conditions. Let “ P_{out} ” be the output power, “ P_{in} ” be the input power, and “PAE” be the power-added efficiency. Let “ P_{out} (fund)” represent the output power at fundamental frequency, “ P_{out} (2nd)” represent the output power at 2nd harmonic, and “ P_{out} (3rd)” represent the output power at 3rd harmonic. P_{out} versus P_{in} , Gain versus P_{in} , PAE versus P_{in} , P_{out} (fund) versus P_{in} , P_{out} (2nd) versus P_{in} , and P_{out} (3rd) versus P_{in} are the power characteristics being investigated for the comparisons between model and measurement data.

Under Class AB bias condition 1), i.e., $(I_{dq}, V_{dQ}) = (10.5$ mA, 28 V): The P_{out} versus P_{in} , Gain versus P_{in} , and PAE versus P_{in} for various load impedances

Figure 3.18: Measured (circle) and modeled (solid line) S -parameter simulation results at pulsed bias (1.99 V, 5.43 V). The gate voltage is pulsed from quiescent point of 0 V up to pulsed point of 1.99 V, and the drain voltage is pulsed from quiescent point of 0 V up to pulsed point of 5.43 V.

are shown in Fig. 3.22 (a) and (c). The fundamental and harmonic powers P_{out} (fund) versus P_{in} , P_{out} (2nd) versus P_{in} , and P_{out} (3rd) versus P_{in} for various load impedances are shown in Fig. 3.22 (b) and (d). According to the HB simulations in Fig. 3.22, the overall large-signal model has good accuracy for the power characteristics under large-signal conditions.

Under Class AB bias condition 2), i.e., $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$: The P_{out} versus P_{in} , Gain versus P_{in} , and PAE versus P_{in} for various load impedances are shown in Fig. 3.23 (a) and (c). The fundamental and harmonic powers P_{out} (fund) versus P_{in} , P_{out} (2nd) versus P_{in} , and P_{out} (3rd) versus P_{in} for various load impedances are shown in Fig. 3.23 (b) and (d). The good agreement between model and measurement over those power characteristics observed in Fig. 3.22 and

Figure 3.19: Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-5.00 V, 27.02 V) and (1.99 V, 5.43 V).

Figure 3.20: Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-5.00 V, 48.01 V) and (0.99 V, 10.57 V).

Figure 3.21: Comparisons of Y -parameters between the measurement data, the static SM-based charge model, and the proposed time-dependent SM-based charge model for two different pulsed biases (-2.80 V, 26.68 V) and (-0.01 V, 9.29 V).

Figure 3.22: Measured (circle) and modeled (solid line) HB results for different load impedances at 3.6 GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$. The dash line represents the coarse model without mapping (a) P_{out} , Gain, and PAE versus P_{in} at $Z_{load} = 93.79 + 34.63j$; (b) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load} = 93.79 + 34.63j$; (c) P_{out} , Gain, and PAE versus P_{in} at $Z_{load} = 82.06 + 54.59j$; (d) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load} = 82.06 + 54.59j$.

Fig. 3.23 indicates that the overall large-signal model can accurately predict the large-signal behaviors of the GaN HEMT device.

The developed model using the proposed space mapping technique is a nonlinear model, which is trained by the bias-dependent S -parameter data at different biases as well as DC and PIV data which cover a large signal range of the (v_{gs}, v_{ds}) . When P_{in} is relatively low or moderate, the developed model is very accurate as shown in the model-data comparisons in Fig. 3.22 and Fig. 3.23. If the input power is too high, the accuracy of the developed model is decreased. This is because that the developed model is not directly trained by the large signal data.

To further test the performance of the developed overall large-signal model, the load-pull figures of merit of the model under the two aforementioned Class AB conditions are investigated. The model is verified in *ADS* under HB simulations, driven beyond 3 dB gain compression points. The source impedance used is $Z_{src} = 50.51 + 9.51j$. Two sets of load impedances Z_{load} are used in the simulations: one set of load impedances is at maximum drain efficiency and the other set of load impedances is at maximum power.

The comparisons of load-pull figures of merit under Class AB (10.5 mA, 28 V) at maximum drain efficiency and maximum power are listed in Table 3.3 and Table 3.4, respectively. The comparisons of load-pull figures of merit under Class AB (10.5 mA, 48 V) at maximum drain efficiency and maximum power are listed in Table 3.5 and Table 3.6, respectively. Reasonable maximum drain efficiency difference and reliable maximum power between the developed model and the measurement data for both two Class AB bias conditions suggest that the overall large-signal

Figure 3.23: Measured (circle) and modeled (solid line) HB results for different load impedances at 3.6 GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$. The dash line represents the coarse model without mapping (a) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=84.28+49.18j$; (b) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=84.28+49.18j$; (c) P_{out} , Gain, and PAE versus P_{in} at $Z_{load}=89.69+31.45j$; (d) P_{out} (fund), P_{out} (2nd), and P_{out} (3rd) versus P_{in} at $Z_{load}=89.69+31.45j$.

Table 3.3: Comparisons of Load-pull Figures of Merit at Maximum Drain Efficiency in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$

	Measurement Data	Developed Model
$f(\text{GHz})$	3.6	3.6
$Z_{src}(\Omega)$	50.54+9.51j	50.54+9.51j
$Z_{load}(\Omega)$	59.86+68.97j	59.86+68.97j
$Z_{in}(\Omega)$	5.94-20.16j	6.15-23.67j
$P_{out}(\text{dBm})$	35.23	34.47
<i>Drain Efficiency (%)</i>	65.28	52.57
$G_t(\text{dB})$	11.16	11.00
$G_p(\text{dB})$	15.56	16.38

Table 3.4: Comparisons of Load-pull Figures of Merit at Maximum Power in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 28 \text{ V})$

	Measurement Data	Developed Model
$f(\text{GHz})$	3.6	3.6
$Z_{src}(\Omega)$	50.54+9.51j	50.54+9.51j
$Z_{load}(\Omega)$	54.95+27.72j	54.95+27.72j
$Z_{in}(\Omega)$	6.34-23.41j	7.03-26.89j
$P_{out}(\text{dBm})$	36.71	36.28
<i>Drain Efficiency (%)</i>	58.16	49.19
$G_t(\text{dB})$	10.03	9.56
$G_p(\text{dB})$	14.31	13.61

model developed using the proposed approach can correctly predict the large-signal behaviors of the GaN HEMT device. In addition, other load-pull figures of merit, i.e., input impedance Z_{in} , transducer gain G_t , and power gain G_p , are reasonably close between the developed model and the measurement data.

Additional information for model validations are provided here in terms of the

Table 3.5: Comparisons of Load-pull Figures of Merit at Maximum Drain Efficiency in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$

	Measurement Data	Developed Model
$f(\text{GHz})$	3.6	3.6
$Z_{src}(\Omega)$	50.54+9.51j	50.54+9.51j
$Z_{load}(\Omega)$	84.25+95.47j	84.25+95.47j
$Z_{in}(\Omega)$	6.34-21.82j	6.43-25.52j
$P_{out}(\text{dBm})$	38.38	37.72
<i>Drain Efficiency (%)</i>	64.20	51.43
$G_t(\text{dB})$	12.76	13.19
$G_p(\text{dB})$	16.99	17.49

Table 3.6: Comparisons of Load-pull Figures of Merit at Maximum Power in Class AB Operation with $(I_{dq}, V_{dQ}) = (10.5 \text{ mA}, 48 \text{ V})$

	Measurement Data	Developed Model
$f(\text{GHz})$	3.6	3.6
$Z_{src}(\Omega)$	50.54+9.51j	50.54+9.51j
$Z_{load}(\Omega)$	82.14+53.88j	82.14+53.88j
$Z_{in}(\Omega)$	6.90-24.06j	7.06-27.86j
$P_{out}(\text{dBm})$	39.16	38.55
<i>Drain Efficiency (%)</i>	58.32	48.86
$G_t(\text{dB})$	11.93	11.56
$G_p(\text{dB})$	15.94	15.64

gate current and the load-pull at fixed available powers (P_{avs}). Fig. 3.24 provides the gate leakage current versus different pulsed biases at 25 °C, 75 °C, and 125 °C for quiescent bias at (0 V, 0 V). Fig. 3.24 shows that the developed model matches the data.

Table 3.7 shows a list of model-data comparisons of P_{out} and PAE at different

(a)

(b)

(c)

Figure 3.24: Measured (circle) and modeled (solid line) gate current results at different temperatures and $(V_{gQ}, V_{dQ}) = (0\text{ V}, 0\text{ V})$. (a) gate current at $25\text{ }^{\circ}\text{C}$. (b) gate current at $75\text{ }^{\circ}\text{C}$. (c) gate current at $125\text{ }^{\circ}\text{C}$.

Z_{load} for certain Pavs. The table indicates that the model developed using proposed technique can predict the P_{out} and PAE for different Z_{load} in the example. In addition, P_{out} contours and PAE contours from the model at Pavs = 21.77 dBm are provided in Fig. 3.25.

Figure 3.25: Modeled load-pull contours at Pavs=21.77 dBm, RFfreq=3.6 GHz under Class AB operation with $(I_{dq}, V_{dQ}) = (10.5\text{mA}, 48\text{ V})$. (a) PAE contours; (b) P_{out} contours

The measurement data used for model training and verification in the GaN HEMT example are summarized in Table 3.8. The pulsed data are measured with a short pulse width in order to make the measurement isothermal and isotrapping. The GaN HEMT device in the example is a GaN HEMT on silicon carbide. The large-signal data were measured under on-wafer *Active Load-pull System* with harmonic terminations close to $50\ \Omega$. In the validation, the *ADS* simulation used the same impedances for harmonic terminations as well.

Further, we performed a comparison of the GaN HEMT modeling results be-

Table 3.7: Comparisons of Output Powers and Power Added Efficiencies for Different Load Impedances at Fixed Available Powers

$Z_{load} (\Omega)$	$P_{out} \text{ (dBm)}$		$PAE \text{ (\%)}$	
	Measured	Modeled	Measured	Modeled
$P_{avs} = 16.82 \text{ dBm}$				
82.37+54.48j	31.76	31.90	27.80	26.39
77.52+51.12j	31.65	31.66	26.76	24.80
82.78+62.85j	31.93	32.10	29.22	28.01
102.48+53.94j	31.65	32.28	27.68	29.40
72.76+42.42j	31.32	31.25	24.17	22.31
77.90+27.13j	30.84	30.96	21.17	20.88
63.20+40.69j	31.13	30.81	22.75	19.89
64.65+29.86j	30.72	30.52	20.28	18.60
69.16+10.75j	30.06	30.14	17.05	17.03
77.70+44.06j	31.43	31.46	25.92	23.59
$P_{avs} = 18.75 \text{ dBm}$				
83.73+94.96j	34.42	34.82	46.66	43.49
84.07+49.54j	33.50	33.79	32.85	32.22
73.01+95.01j	34.44	34.58	46.55	41.05
83.00+72.38j	34.08	34.36	39.63	37.49
67.59+59.82j	33.74	33.58	34.84	30.36
82.04+123.15j	34.38	34.83	51.55	47.99
81.55+145.73j	34.11	34.51	53.05	48.54
89.71+31.69j	32.97	33.43	28.28	29.60
73.26+114.11j	34.49	34.75	51.42	45.37
104.98+90.34j	34.11	34.93	42.30	45.16
$P_{avs} = 21.77 \text{ dBm}$				
75.80+58.38j	36.38	36.48	46.32	43.53
90.31+57.78j	36.36	36.77	47.00	47.21
90.09+89.62j	36.77	37.02	56.57	53.75
98.75+74.93j	36.60	37.06	52.13	52.64
92.79+42.74j	36.02	36.51	42.18	43.85
117.40+124.14j	36.34	36.58	56.81	55.65
58.15+69.18j	36.43	36.16	47.86	40.10
152.50+154.14j	35.68	35.92	49.81	52.98
87.02+9.23j	35.02	35.55	31.71	34.30
55.41+33.69j	35.15	34.91	32.17	29.12

Table 3.8: Summary of Measurement Data Used for Model Development

	Data for Model Training	Data for Model Verification
DC Operation	PIV (0 V, 0 V) data at 25 °C, 75 °C, and 125 °C. The number of samples is 3110.	DC data. The number of samples is 539.
PIV Operation	Training set of the PIV data at seven different quiescent biases. The number of samples is 192.	Testing set of the PIV data at seven different quiescent biases. The number of samples is 6578.
S-parameter Operation	Training set of the pulsed S-parameter data at (0 V, 0 V) quiescent bias. The number of samples is 84.	Testing set of the PIV data at (0 V, 0 V) quiescent bias. The number of samples is 84.
Large-signal Operation	/	HB data at two different quiescent biases for different load impedances. The number of samples is 2288 for (Idq, VdQ) = (10.5 mA, 28 V) and 1430 for (Idq, VdQ) = (10.5 mA, 48 V).
		Pout and PAE data at two different quiescent biases for different load impedances. The number of samples is 1434.

tween the proposed technique and the existing space mapping technique as shown in Table 3.9. The existing technique used for the comparison is from [43], which is a static space mapping technique. As can be seen from the table, the developed model using the proposed space mapping technique has better model accuracy than that using the existing space mapping technique, in terms of DC, PIV, and S -parameter responses.

Table 3.9: Comparison of Model Results between the Proposed Space Mapping Technique and the Existing Space Mapping Technique

	Number of Hidden Neurons	Overall Training Error	Overall Testing Error	Modeling Error for Subsets of Data		
				DC Data (PIV (0, 0) @ 25 °C)	PIV Data	S-Parameter Data (at Seven Biases)
Coarse Sub-Models (before Using Space Mapping Techniques)	/	/	/	0.366%	7.9881%	19.3417%
Model Developed by Existing Space Mapping Technique	5	3.273%	6.321%	2.3643%	14.619%	22.474%
	10	2.394%	6.630%	0.3672%	7.991%	19.293%
	30	2.326%	64.642%	0.2601%	149.541%	29.485%
Model Developed by Proposed Space Mapping Technique	/	0.396%	0.555%	0.366%	1.025%	5.754%

Here we provide a further explanation for the performance of the existing space mapping approach in Table 3.9. The model developed using the existing space mapping technique has relatively large errors. Neural networks with different number of hidden neurons are trained for the mapping. When the number of hidden neurons is 5 (which is small), both training error and testing error are large, suggesting that the neural networks are in the under-learning state. As the number of hidden neuron increases, the training errors slowly decrease while the testing errors quickly increase, leading to overlearning for neural networks. In both scenarios, the developed model has large testing errors. This is because that the one single mapping cannot effectively map different behaviors at the same time, resulting in the trade-off among errors for mapping different behaviors. Besides, the mapping is static, which distributes the mapping errors over the entire frequency range, compromising the

model accuracy of S -parameters between low and high frequencies. That is why the developed model using existing space mapping has large errors for S -parameters.

The proposed space mapping technique develops separate mappings to map different behaviors separately, so that each and every one of the behaviors can be mapped reliably, leading to much improved solutions for the overall model. Moreover, the proposed time-dependent mappings for charge models are dynamic and frequency-dependent. The mapping at low frequency can be different from the mapping at high frequency. Therefore, the proposed mappings can simultaneously improve the model accuracy of S -parameters at both low frequency and high frequency, ensuring good model accuracy of S -parameters in the whole frequency range. That is why the developed model using proposed space mapping has lower errors for S -parameters than that using the existing space mapping technique.

From all the above demonstrated comparisons, good performance of the overall large-signal model developed using the proposed approach has been verified in DC, PIV, small-signal, and large-signal operations for the GaN HEMT example. The proposed space mapping modeling approach is systematic and fast. By developing different mapping modules, coarse sub-models are mapped separately to address diversely different behaviors such as trapping effects and frequency dispersion. Each mapping module is developed through supervised learning methods to systematically overcome the gap between the coarse sub-model and its corresponding measurement data. Moreover, the proposed modeling approach uses less training data than pure ANN modeling approach.

3.4 Conclusions

This chapter has presented a novel space mapping modeling approach for GaN HEMTs with trapping effects. The proposed technique develops separate mappings for different branches inside the existing device model, such that different features such as trapping effects and frequency dispersion can be mapped separately. By developing different mappings, different features can be addressed more efficiently and accurately. The overall large-signal model developed using the proposed approach has been verified by the measurement data of a $2 \times 350 \mu\text{m}$ GaN HEMT device. Good accuracy and convergence of the developed model under DC, PIV, small signal, and large-signal simulations have been achieved. The proposed space mapping technique is a fast and systematic modeling approach to develop a large-signal model for GaN HEMTs with satisfactory accuracy. The proposed space mapping technique exploits supervised learning methods to systematically develop separate mappings to achieve desired model accuracy, speeding up the model development process over the manual process of developing equivalent circuit/empirical models. Compared to the pure ANN modeling approach, the proposed space mapping technique employs less training data, lowering the cost of data generation from measurements. The potential use of the SVM and Kriging techniques for GaN model development, especially for trapping effects, is a possible future direction. Another future direction is to explore the validation of the mapped model with other types of data, such as dynamic drain currents at different biases, drain current versus bias at RF excitation, and memory effects or two-tone data.

Chapter 4

Parametric Modeling of EM Behavior of Microwave Components Using Combined Neural Networks and Hybrid-Based Transfer Functions

4.1 Introduction

Neuro-transfer function (neuro-TF) approaches have become more and more popular in parametric modeling for electromagnetic (EM) behavior of microwave components. Existing pole-residue-based neuro-TF approach has better capability of dealing with high-order problem than the rational-based neuro-TF approach, but has the discontinuity issue and the associated non-smoothness issue of the poles/residues when the geometrical variations become large while the rational-based neuro-TF approach does not have. This chapter addresses this situation and presents a novel hybrid-based neuro-TF technique which systematically combines both pole-residue

and rational formats of the transfer functions. Starting with the pole-residue-based transfer functions, we propose a novel technique to automatically identify the poles/residues that are smooth-continuous and the poles/residues that have the discontinuity and non-smoothness issues. The proposed technique converts the poles/residues that have those issues into the coefficients of the rational-based transfer function to solve the discontinuity and non-smoothness issues in the existing pole-residue-based neuro-TF approach. The proposed technique remains the smooth-continuous poles/residues in the pole-residue format of the transfer function to maintain the capability of handling high-order problem. Compared with the existing neuro-TF modeling methods, the proposed technique can obtain better accuracy in challenging applications of large geometrical variations and high order. The proposed technique is illustrated by two examples of parametric modeling of microwave components.

4.2 The Issues of Discontinuity and Accompanied Non-Smoothness in Poles and Residues with respect to Geometrical Parameters

In the existing pole-residue-based neuro-TF modeling approach [49], a parameter extraction process is required to extract the poles/residues in the transfer function for each geometrical parameter sample. All the extracted poles/residues are required to be changed continuously in terms of the types of their values as the geometrical parameters change. The method can work well when the geometrical variations are small. It is because that all the poles/residues vary within complex

values, which are continuous as the geometrical parameters change. However, as the geometrical variations become large, not all the poles/residues vary within complex values as the geometrical parameters change. In this case, the discontinuity issue occurs in the poles/residues and becomes a major issue for parametric modeling. More specifically, some poles/residues vary within complex values, while other poles/residues vary between real values and complex values, as the geometrical parameters change. The discontinuity issue is also the cause for the high nonlinearity of pole/residues from sample to sample, resulting in the difficulty of the training for the pole-residue-based neuro-TF model. For this reason, the effectiveness of the method is compromised when the discontinuity issue exists in the poles/residues as the geometrical variations become large. Along with the discontinuity issue in the poles/residues, the smoothness of the neighboring poles/residues with respect to the geometrical parameters are also affected. Although some poles/residues do not have the discontinuity issue as other poles/residues, some of these continuous poles/residues are not smooth with respect to the geometrical parameters, leading to the non-smoothness issue. This issue is also the cause for the high nonlinearity of pole/residues from sample to sample, resulting in the difficulty of the training for the pole-residue-based neuro-TF model. This is another reason why the effectiveness of the method is compromised when the discontinuity issue exists in the poles/residues as the geometrical variations become large.

Rational-based neuro-TF modeling approach is one of the methods that does not have the discontinuity issue existed in the pole-residue-based neuro-TF approach. In the existing rational-based neuro-TF modeling approach [45], the poles/residues

are reformatted into the coefficients of the numerators and the denominators of the rational-based transfer function. These coefficients always vary in real values as the geometrical parameters change. In other words, the coefficients are continuous and does not have the discontinuity issue. However, as the order of the transfer function increases, the transfer function is more sensitive to the coefficients, resulting in the difficulty of the training for the rational-based neuro-TF model. For this reason, the effectiveness of the method is compromised when the order of the transfer function is high.

The focus of this chapter is on developing a new and effective parametric modeling technique, addressing the discontinuity issue and the accompanied non-smoothness issue. In the subsequent section, we introduce a novel hybrid-based neuro-TF modeling technique which systematically combines both pole-residue and rational formats of the transfer functions in order to overcome those issues in the poles/residues. The main idea is that we convert the poles/residues that have the issues into the coefficients of the rational-based transfer function to effectively address the discontinuity and non-smoothness issues and we remain the smooth-continuous poles/residues in the pole-residue format of the transfer function to maintain the capability of handling high-order problem. In this way, the discontinuity and non-smoothness issues can be solved by the proposed hybrid-based neuro-TF modeling technique and high-order modeling capability can be maintained by the overall model.

Figure 4.1: The structure of the proposed hybrid-based neuro-TF model.

4.3 Proposed Hybrid-Based Neuro-TF Method for Parametric Model Development

4.3.1 Formulation of the Hybrid-Based Neuro-TF Model

The structure of the proposed hybrid-based neuro-TF model is illustrated in Fig. 4.1. Let $\mathbf{x}$ be the model input, representing the geometrical variables. Let y be the model output, representing the EM responses (e.g., S -parameters). The proposed model consists of the pole-residue-based transfer function, the rational-based transfer function, and four neural networks. The structure of the proposed model is a little bit more complicated than the existing pole-residue-based neuro-TF or the existing rational-based neuro-TF. The outputs of the model are the EM behaviors e.g., S -parameters, of the microwave components. The inputs of the model are the geometrical parameters of the microwave components and frequency. As the values

of geometrical parameters change, some of the pole/residues may vary between real and complex values, resulting in the discontinuity issue. Along with the discontinuity issue in the poles/residues, the smoothness of the neighboring poles/residues with respect to the geometrical parameters are also affected, leading to the non-smoothness issue. Here, we define the continuous pole/residue to be the one whose value is always complex or real as the values of the geometrical parameters change. We define the discontinuous pole/residue to be the one whose value varies between real and complex as the value of the geometrical parameters change. Among the continuous poles/residues, there may exist one or more poles/residues whose values do not change smoothly as the values of the geometrical parameters change. The non-smoothness issue happens in these continuous poles/residues whose smoothness with respect to the geometrical parameters does not satisfy the smoothness requirement. Let these poles/residues be called the non-smooth-continuous poles/residues. The rest of the continuous poles/residues satisfy the smoothness requirement. Let the remaining smooth and continuous poles/residues be called the smooth-continuous poles/residues.

We propose to maintain the smooth-continuous poles/residues to formulate the pole-residue-based transfer function, while to reformulate the discontinuous poles/residues and the non-smooth-continuous poles/residues together into rational-based transfer function. The pole-residue-based transfer function and the rational-format-based transfer function are combined together to formulate the proposed hybrid-based transfer function. By such systematical combination of the two transfer functions, the discontinuity and non-smoothness issues can be effectively addressed by

the rational-based transfer function with a relatively low order and the capability of handling high-order problem can be maintained by the pole-residue-based transfer function without the issues. In this way, the discontinuity and non-smoothness issues can be solved by the proposed hybrid-based neuro-TF modeling technique and high-order modeling capability can be maintained by the overall model. The parameters in the proposed hybrid-based transfer function are defined as the hybrid parameters, containing the poles/residues of the pole-residue-based transfer function and coefficients in the numerator/denominator of the rational-based transfer function. Since the relationship between the hybrid parameters and the geometrical parameters is nonlinear and unknown, we propose to use neural networks to learn and represent this nonlinear relationship.

Let N_1 and N_2 represent the orders of the pole-residue-based transfer function and the rational-based transfer function, respectively. Let N represent the total order of the hybrid-based transfer function, which equals the sum of the N_1 and N_2 , i.e., $N = N_1 + N_2$. In the propose model, four different neural networks are used to learn and represent the nonlinear and unknown relationship between the hybrid parameters and the geometrical parameters. More specifically, these neural networks are used to learn and represent the nonlinear and unknown relationship between the poles/residues of the pole-residue-based transfer function and the geometrical parameters, and the nonlinear and unknown relationship between the coefficients of the rational-based transfer function and the geometrical parameters. Let p_i^h represent the i th output of the neural network for poles corresponding to geometrical variables and neural network weights $\mathbf{w}_p^h$. Let r_i^h represent the i th output of the

neural network for residues corresponding to geometrical variables and neural network weights $\mathbf{w}_r^h$. Let a_i^h represent the i th output of the neural network for the coefficients in the numerator of the rational-based transfer function corresponding to geometrical variables and neural network weights $\mathbf{w}_a^h$. Let b_i^h represent the i th output of the neural network for the coefficients in the denominator of the rational-based transfer function corresponding to geometrical variables and neural network weights $\mathbf{w}_b^h$. Let $\mathbf{w}^h$ represent a vector containing all the weights in the proposed hybrid-based neuro-TF model, defined as

$$\mathbf{w}^h = [(\mathbf{w}_p^h)^T \ (\mathbf{w}_r^h)^T \ (\mathbf{w}_a^h)^T \ (\mathbf{w}_b^h)^T]^T. \quad (4.1)$$

Let $\mathbf{x}$ be a vector containing the geometrical variables, representing the inputs of the proposed hybrid-based neuro-TF model. Let y represent the output frequency response, e.g., S -parameter of the proposed model. For convenience of representation, we define $\mathbf{p}^h$ and $\mathbf{r}^h$ to be vectors containing the smooth-continuous poles and residues, respectively, i.e., $\mathbf{p}^h = [p_1^h \ p_2^h \ \cdots \ p_{N_1}^h]^T$ and $\mathbf{r}^h = [r_1^h \ r_2^h \ \cdots \ r_{N_1}^h]^T$. We define $\mathbf{a}^h$ and $\mathbf{b}^h$ to be vectors containing coefficients in the numerator and denominator of the rational-based transfer function, respectively, i.e., $\mathbf{a}^h = [a_1^h \ a_2^h \ \cdots \ a_{N_2}^h]^T$ and $\mathbf{b}^h = [b_1^h \ b_2^h \ \cdots \ b_{N_2}^h]^T$. As is known to all, the pole-residue-based transfer function and the rational-based transfer function are interconvertible to represent the frequency response in general. Instead of using these two classical transfer functions, we propose to convert some of the pole-residue terms from the original pole-residue-based transfer function into the rational-based transfer function in order to formulate the hybrid-based neuro-TF model. The output y of the hybrid-based neuro-TF model

can be formulated as

$$y(\mathbf{x}, \mathbf{w}^h, s) = \sum_{i=1}^{N_1} \frac{r_i^h(\mathbf{x}, \mathbf{w}_r^h)}{s - p_i^h(\mathbf{x}, \mathbf{w}_p^h)} + \frac{\sum_{i=1}^{N_2} a_i^h(\mathbf{x}, \mathbf{w}_a^h) s^{i-1}}{1 + \sum_{i=1}^{N_2} b_i^h(\mathbf{x}, \mathbf{w}_b^h) s^i}, \quad (4.2)$$

where s represents the frequency in Laplace domain.

4.3.2 Proposed Hybrid Parameter Extraction Technique

The proposed hybrid parameter extraction technique begins with samples of EM data $\mathbf{d}_k$, e.g., S -parameters, for different values of geometrical parameters $\mathbf{x}_k$, where the subscript k represents the index indicating the k th sample of geometrical parameters, i.e., $k \in T_r = \{1, 2, \dots, n_s\}$, where T_r is the index set of training samples of geometrical parameters and n_s is the total number of training samples. Training samples are generated with respect to the geometrical parameters while frequency is a separate variable swept by the EM simulator during data generation. In our method, frequency is an additional input of the hybrid-based transfer function.

Vector fitting process [118],[119] is performed to obtain a group of poles and residues for each geometrical sample. In the vector fitting process, the given information is EM data $\mathbf{d}$ versus frequency for a certain geometrical sample. Expected solutions are poles and residues of the transfer function. Let $\tilde{\mathbf{c}}_k$ represent a vector containing the poles $\tilde{\mathbf{p}}^{(k)}$ and residues $\tilde{\mathbf{r}}^{(k)}$ of the transfer function for the k th

geometrical sample obtained after vector fitting, defined as

$$\tilde{\mathbf{c}}_k = \begin{bmatrix} \tilde{\mathbf{p}}^{(k)} \\ \tilde{\mathbf{r}}^{(k)} \end{bmatrix} = \begin{bmatrix} [\tilde{p}_1^{(k)} \ \tilde{p}_2^{(k)} \ \cdots \ \tilde{p}_N^{(k)}]^T \\ [\tilde{r}_1^{(k)} \ \tilde{r}_2^{(k)} \ \cdots \ \tilde{r}_N^{(k)}]^T \end{bmatrix}. \quad (4.3)$$

We propose a hybrid parameter extraction technique to identify the discontinuous poles/residues, the smooth-continuous poles/residues, and the non-smooth-continuous poles/residues. After the identification, the smooth-continuous poles/residues will be maintained into the pole-residue-based transfer function, while the discontinuous poles/residues and the non-smooth-continuous poles/residues together will be re-formulated into rational-based transfer function. Before using the proposed method, the existing pole sorting technique [49] is used to solve the pole mis-match problem from vector fitting. If the matched poles still have the discontinuity and non-smoothness issues, we perform the proposed modeling process.

Suppose that we have already obtained the pole and residue data (i.e., $\tilde{\mathbf{p}}^{(k)}$ and $\tilde{\mathbf{r}}^{(k)}$) for all the geometrical samples after vector fitting. Since each pole $\tilde{p}_i^{(k)}$ usually has a related residue $\tilde{r}_i^{(k)}$ after vector fitting, we suppose that the residue data $\tilde{\mathbf{r}}^{(k)}$ are listed in the same way as their corresponding pole data $\tilde{\mathbf{p}}^{(k)}$. The proposed hybrid parameter extraction technique consists of three stages.

The first stage of the proposed hybrid parameter extraction technique is to identify the discontinuous poles/residues and to identify the continuous poles/residues for further identification in the second stage. Let N be the total order of the hybrid-based transfer function. Let n_s be the total number of the geometrical samples. Usually, the discontinuous residues appear together with their corresponding

discontinuous poles. Similarly, the continuous residues appear together with their corresponding continuous poles. For simplicity of representation, we use n_i^R to represent the number of the geometrical samples where the i th pole $\tilde{p}_i^{(k)}$ (or the i th residue $\tilde{r}_i^{(k)}$) is a real pole (or a real residue), i.e., $i \in J = \{1, 2, \dots, N\}$, where J is the index set of samples of pole data $\tilde{\mathbf{p}}^{(k)}$ (or residue data $\tilde{\mathbf{r}}^{(k)}$). Let η_i represent the ratio between n_i^R and n_s for $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$), where $n_i^R \leq n_s$, defined as

$$\eta_i = \frac{n_i^R}{n_s}. \quad (4.4)$$

Here we define a step function $f(\cdot)$ for η_i , expressed as

$$f(\eta_i) = \begin{cases} 1, & \text{if } 0 < \eta_i < 1 \\ 0, & \text{otherwise} \end{cases}. \quad (4.5)$$

If $0 < \eta_i < 1$, it indicates that $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$) is a real pole (or residue) for some geometrical samples and is a complex pole (or residue) for other geometrical samples. In this case, $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$) is a discontinuous pole (or residue) and $f(\eta_i) = 1$. Otherwise, $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$) is either a real pole (or residue) or a complex pole (or residue) for all the geometrical samples. In this case, $\tilde{p}_i^{(k)}$ (or $\tilde{r}_i^{(k)}$) is a continuous pole (or residue) and $f(\eta_i) = 0$. Let N_2^a be the number of discontinuous poles (or residues), formulated as

$$N_2^a = \sum_{i=1}^N f(\eta_i). \quad (4.6)$$

The index set J_2^a of the samples of discontinuous pole data (or discontinuous residue

data) are formulated as

$$J_2^a = \{i \mid f(\eta_i) = 1, i \in J\}. \quad (4.7)$$

Subsequently, the index set J' of the samples of continuous pole data (or continuous residue data) can be expressed as $J' = J \setminus J_2^a$. Both index sets J_2^a and J' are the subsets of the index set J . So far, we have identified the discontinuous pole/residue data and the continuous pole/residue data through calculation of N_2^a , J_2^a , and J' .

In the second stage, we identify the non-smooth continuous poles/residues from the identified continuous poles/residues with the index set J' . Here we use $\tilde{\sigma}_i^p$ and $\tilde{\sigma}_i^r$ to represent the deviations of the i th continuous poles $\tilde{\mathbf{p}}_i$ and the i th continuous residues $\tilde{\mathbf{r}}_i$, respectively, covering all the geometrical samples, where $i \in J'$. The deviations $\tilde{\sigma}_i^p$ and $\tilde{\sigma}_i^r$ are expressed as

$$\tilde{\sigma}_i^p = \max_{k \in T_r} \{ | \tilde{p}_i^{(k)} - \tilde{\mu}_i^p | \} \quad (4.8)$$

and

$$\tilde{\sigma}_i^r = \max_{k \in T_r} \{ | \tilde{r}_i^{(k)} - \tilde{\mu}_i^r | \} \quad (4.9)$$

where $\tilde{\mu}_i^p$ and $\tilde{\mu}_i^r$ are expressed as

$$\tilde{\mu}_i^p = \frac{1}{n_s} \sum_{k=1}^{n_s} \tilde{p}_i^{(k)} \quad (4.10)$$

and

$$\tilde{\mu}_i^r = \frac{1}{n_s} \sum_{k=1}^{n_s} \tilde{r}_i^{(k)}. \quad (4.11)$$

Here we define a shared deviation σ_i^h for both i th continuous poles and residues $\tilde{\mathbf{p}}_i$

and $\tilde{\mathbf{r}}_i$ in order to perform the identification, formulated as

$$\tilde{\sigma}_i = \max\{\tilde{\sigma}_i^p, \tilde{\sigma}_i^r\} \quad (4.12)$$

where $i \in J'$. Let $\tilde{\sigma}_0$ be the user-defined threshold of the deviation. We define a step function $g(\cdot)$ for $\tilde{\sigma}_i$, expressed as

$$g(\tilde{\sigma}_i) = \begin{cases} 1, & \text{if } \tilde{\sigma}_i > \tilde{\sigma}_0 \\ 0, & \text{otherwise} \end{cases}. \quad (4.13)$$

If $\tilde{\sigma}_i > \tilde{\sigma}_0$, the i th continuous poles and residues $\tilde{\mathbf{p}}_i$ and $\tilde{\mathbf{r}}_i$ are considered non-smooth and $g(\tilde{\sigma}_i) = 1$. Otherwise, the i th continuous poles and residues $\tilde{\mathbf{p}}_i$ and $\tilde{\mathbf{r}}_i$ are considered smooth and $g(\tilde{\sigma}_i) = 0$. Let N_2^b represent the number of the non-smooth-continuous poles/residues, formulated as

$$N_2^b = \sum_{i \in J'} g(\tilde{\sigma}_i). \quad (4.14)$$

It is noticed that the order N_2 of rational-based transfer function cannot be high, since the transfer function becomes more sensitive to its coefficients if the order is higher. For this reason, N_2 should be less than a certain user-defined order N_2^{max} (e.g., N_2^{max} equals eight). Suppose that the number of discontinuous poles (or residues) N_2^a is less than N_2^{max} . Therefore, N_2^b should satisfy the condition

$$N_2^b \leq N_2^{max} - N_2^a. \quad (4.15)$$

If this condition is not satisfied, then we increase the value of $\tilde{\sigma}_0$, recalculate N_2^b and recheck the condition with the updated N_2^b . If the condition is satisfied, we

obtain the index set J_2^b of the samples of non-smooth-continuous pole data and non-smooth-continuous residue data, formulated as

$$J_2^b = \{i \mid g(\tilde{\sigma}_i) = 1, i \in J'\}. \quad (4.16)$$

Subsequently, the index set J_1 of the samples of smooth-continuous pole data and smooth-continuous residue data can be expressed as $J_1 = J' \setminus J_2^b$. Both index sets J_2^b and J_1 are the subsets of the index set J . So far, we have identified the smooth-continuous pole/residue data and the non-smooth-continuous pole/residue data through the calculation of N_2^b , J_2^b , and J_1 . The order N_2 is subsequently calculated as

$$N_2 = N_2^a + N_2^b. \quad (4.17)$$

The order N_1 is subsequently calculated as

$$N_1 = N - N_2. \quad (4.18)$$

The total index set J_2 of the samples of the discontinuous pole/residue data and the non-smooth-continuous pole/residue data is expressed as $J_2 = J_2^a \cup J_2^b$.

In the third stage, we gather the discontinuous poles/residues and the non-smooth-continuous poles/residues with the index set J_2 from the original pole/residue data into the new pole/residue data. These new pole/residue data are reformatted into the coefficient data for the rational-format-based transfer function. After reformatting, the coefficient data are defined as $(a_i^h)^{(k)}$ and $(b_i^h)^{(k)}$ for the k th geometrical sample, where $i = 1, 2, \dots, N_2$. We maintain the smooth-continuous pole/residue data for the pole-residue-based transfer function. These smooth-continuous pole/re-

sidue data are redefined as $(p_i^h)^{(k)}$ and $(r_i^h)^{(k)}$ for the k th geometrical sample, where $i = 1, 2, \dots, N_1$. Let $\mathbf{c}_k^h$ represent a vector containing the hybrid parameters of the hybrid-based transfer function for the k th geometrical sample, defined as

$$\mathbf{c}_k^h = [((\mathbf{p}^h)^{(k)})^T \quad ((\mathbf{r}^h)^{(k)})^T \quad ((\mathbf{a}^h)^{(k)})^T \quad ((\mathbf{b}^h)^{(k)})^T]^T \quad (4.19)$$

where $(\mathbf{p}^h)^{(k)}$, $(\mathbf{r}^h)^{(k)}$, $(\mathbf{a}^h)^{(k)}$, and $(\mathbf{b}^h)^{(k)}$ are defined as $(\mathbf{p}^h)^{(k)} = [(p_1^h)^{(k)} \ (p_2^h)^{(k)} \ \dots \ (p_{N_1}^h)^{(k)}]^T$, $(\mathbf{r}^h)^{(k)} = [(r_1^h)^{(k)} \ (r_2^h)^{(k)} \ \dots \ (r_{N_1}^h)^{(k)}]^T$, $(\mathbf{a}^h)^{(k)} = [(a_1^h)^{(k)} \ (a_2^h)^{(k)} \ \dots \ (a_{N_2}^h)^{(k)}]^T$, and $(\mathbf{b}^h)^{(k)} = [(b_1^h)^{(k)} \ (b_2^h)^{(k)} \ \dots \ (b_{N_2}^h)^{(k)}]^T$, respectively. Fig. 4.2 provides a flow diagram illustrating the proposed three-stage hybrid parameter extraction technique.

4.3.3 Preliminary Training of Neural Networks

We propose a three-stage training process. In the first stage, we perform a preliminary training process for characterizing the relationships between the poles/residues of the transfer function and the geometrical parameters. During this process, neural networks are trained to learn these relationships. The training data for this stage is $(\mathbf{x}_k, (\mathbf{p}^h)^{(k)})$ and $(\mathbf{x}_k, (\mathbf{r}^h)^{(k)})$, $k \in T_r$ (i.e., geometrical parameters as model inputs and poles/residues as model outputs). The nonlinear relationship between poles and geometrical parameters are usually different from that between residues and geometrical parameters. Subsequently, poles and residues are provided by separate neural networks.

In the second stage, we perform a preliminary training process for characterizing the relationships between the coefficients of the transfer function and the geometrical parameters. During this process, different sets of neural networks are

Figure 4.2: The flow diagram of the proposed three-stage hybrid parameter extraction process.

trained to learn these relationships. The training data for this stage is $(\mathbf{x}_k, (\mathbf{a}^h)^{(k)})$ and $(\mathbf{x}_k, (\mathbf{b}^h)^{(k)})$, $k \in T_r$ (i.e., geometrical parameters as model inputs and coefficients as model outputs). The nonlinear relationship between the coefficients in the numerator and geometrical parameters are usually different from that between the coefficients in the denominator and geometrical parameters. Similarly, separate neural networks are used to train the coefficients in the numerator and denominator.

During the two stages of preliminary training, a certain amount of training error (e.g., 5%-10%) can be tolerated. With this relaxed error criteria, less hidden neurons can be used in the neural networks. The neural networks thus have lower nonlinearity which makes the hybrid-based neuro-TF more robust. After preliminary training of the neural networks, an overall model refinement process is performed to further refine the preliminary-trained model.

4.3.4 Refinement Training of the Hybrid-Based Neuro-TF Model

In the third stage of the training process, a model refinement is performed to further refine the overall hybrid-based neuro-TF model. The training data for this stage is $(\mathbf{x}_k, \mathbf{d}_k)$, $k \in T_r$ (i.e., geometrical parameters as model inputs and EM responses (i.e., S -parameters) as model outputs). This model refinement process consists of both training and testing of the model. In the refinement training process, the overall hybrid-based neuro-TF model consists of the hybrid-based transfer function of (4.2) and the neural networks whose initial values are the optimal solutions from the two-stage preliminary training. The objective of training is to minimize the

training error of the overall model. Training is performed by optimizing the weights inside the neural networks to minimize the error function

$$\begin{aligned}
& E_{Tr}(\mathbf{w}_p^h, \mathbf{w}_r^h, \mathbf{w}_a^h, \mathbf{w}_b^h) \\
&= \frac{1}{2n_s} \sum_{k \in T_r} \sum_{m \in \Omega} \|y(\mathbf{w}_p^h, \mathbf{w}_r^h, \mathbf{w}_a^h, \mathbf{w}_b^h, \mathbf{x}_k, s_m) - \mathbf{d}_{k,m}\|^2
\end{aligned} \tag{4.20}$$

where n_s is the total number of training samples; Ω is the index set of frequency samples; y is the outputs of the overall model, which ultimately is a function of geometrical variables $\mathbf{x}_k$, frequency s_m , and neural network weights $\mathbf{w}_p^h$, $\mathbf{w}_r^h$, $\mathbf{w}_a^h$ and $\mathbf{w}_b^h$.

The training process terminates when the training error becomes lower than a user-defined threshold E_t . After the training process, an independent set of testing data which are never used in training is used for testing the quality of the trained hybrid-based neuro-TF model. The testing error E_{Ts} is defined as the error between the model response and the testing data. If the testing error is also lower than the threshold error E_t , the model refinement process terminates and the proposed model is ready to be used for high-level design. Otherwise, the overall model training process will be repeated with different numbers of hidden neurons. A flow diagram illustrating the proposed overall model development process is shown in Fig. 4.3.

4.4 Application Examples

4.4.1 Parametric Modeling of a Four-Pole Waveguide Filter

In this example, the development of a parametric model for the EM behavior of a four-pole waveguide filter [120] using the proposed hybrid-based neuro-TF technique

Figure 4.3: The flow diagram of the overall hybrid-based neuro-TF model development process.

is illustrated. As shown in Fig. 4.4, the geometrical parameters of the four-pole waveguide filter are $\mathbf{x} = [h_1 \ h_2 \ h_3 \ h_{c1} \ h_{c2}]^T$.

Figure 4.4: The geometrical parameters of the four-pole waveguide filter and the 3D configuration for EM simulation. The geometrical parameters are h_1 , h_2 , h_3 , h_{c1} , and h_{c2} .

Figure 4.5: The structure of the proposed hybrid-based neuro-TF model for the four-pole waveguide filter example.

In the proposed technique, a hybrid-based neuro-TF model is developed as the parametric model for the four-pole waveguide filter example. As defined in the model structure of Fig. 4.5, the model has five input geometrical variables and a frequency variable (as an additional input). The model has two outputs, i.e., $\mathbf{y} = [RS_{11} \ IS_{11}]^T$, which are the real and imaginary parts of S_{11} , respectively. Full-

wave EM simulation are performed by the *ANSYS HFSS* EM simulator to generate the training and testing data for parametric modeling. A parallel computation method is used in data generation. Message passing interface (MPI) is used for parallel processing with distributed memory. Design of experiments (DOE) [121] method is used to sample both training data and testing data.

As can be seen in Table 4.1, the proposed parametric modeling technique is applied to two different cases, in terms of the parameter range. Case 1 considers a narrower parameter range and Case 2 considers an increased parameter range. In both cases, the total order N is set to twelve for all the training and test samples. The number of frequency samples for each geometrical sample is 101. In both cases, nine levels of DOE are used to sample the training data, resulting in a total number of 81 samples, while eight levels of DOE are used to sample the testing data, resulting in a total number of 64 samples. Table 4.1 shows the detailed ranges of training data and testing data for the two cases.

The original frequency range in this example is 10.5 GHz - 11.5 GHz. Scaling and shifting of the frequency range are performed. The total order N is split into the order N_1 of the pole-residue-based transfer function and the order N_2 of the rational-based transfer function by using the proposed hybrid parameter extraction technique. Based on the pole/residue data (which are directly obtained from the vector fitting of training data), we identify the discontinuous poles/residues, non-smooth-continuous poles/residues, and the smooth-continuous poles/residues for both cases. The pole-residue data in Case 1 are relatively smooth as the values of the geometrical parameters change. By contrast, the pole-residue data in Case

2 have discontinuity issue in some of the imaginary parts of poles and residues, and have the accompanied non-smoothness issue in other poles and residues, as the values of the geometrical parameters change. The discontinuity issue and the non-smoothness issue in Case 2 can be observed in Fig. 4.6. As shown in the figure, the original poles/residue data have the discontinuity issue and the non-smoothness issue. After using the proposed extraction technique, the newly obtained rational data have better smoothness than the original poles/residue data. Based on the proposed hybrid parameter extraction technique illustrated in Subsection 4.3.2, N_1 is set to eight and N_2 is set to four in Case 1. N_1 is set to six and N_2 is set to six in Case 2. The hybrid-based neuro-TF model is trained using the *NeuroModelerPlus* software for both preliminary training and refinement training. The average training error is 0.909% in Case 1 and the average testing error is 1.620% in Case 2.

For comparison purpose, the recent pole-residue-based neuro-TF modeling method and the existing rational-based neuro-TF modeling method [45] are applied to the two cases. Table 4.2 shows the comparisons of ANN structures, number of hidden neurons, and average training and testing errors between several parametric modeling methods and EM data.

In Case 1, since the geometrical parameters vary within a small range, discontinuity and non-smoothness issues do not exist and the pole-residue data are relatively smooth. Both the pole-residue-based neuro-TF method and the proposed method obtain comparatively small training and testing errors. The rational-based neuro-TF method obtains relatively large training and testing errors, since the transfer function order is relatively high.

Table 4.1: Definition of Training and Testing Data for The Four-Pole Waveguide Filter Example

Geometrical Variables		Training Data (81 samples)			Testing Data (64 samples)		
		Min	Max	Step	Min	Max	Step
Case 1 (Narrower Range)	h_1 (mm)	3.4	3.56	0.02	3.41	3.55	0.02
	h_2 (mm)	4.3	4.46	0.02	4.31	4.45	0.02
	h_3 (mm)	4.0	4.16	0.02	4.01	4.15	0.02
	h_{c1} (mm)	3.2	3.36	0.02	3.21	3.35	0.02
	h_{c2} (mm)	2.9	3.06	0.02	2.91	3.05	0.02
Case 2 (Increased Range)	h_1 (mm)	3.3	3.62	0.04	3.32	3.6	0.04
	h_2 (mm)	4.2	4.52	0.04	4.22	4.5	0.04
	h_3 (mm)	3.9	4.22	0.04	3.92	4.2	0.04
	h_{c1} (mm)	3.1	3.42	0.04	3.12	3.4	0.04
	h_{c2} (mm)	2.8	3.12	0.04	2.82	3.1	0.04

In Case 2, the geometrical parameters vary within an increased range and the pole/residue data have the discontinuity issue and the accompanied non-smoothness issue. The pole-residue-based neuro-TF method encounters the discontinuity issue and the non-smoothness issue, resulting in large testing error. This is because that the pole/residue data (which are used in the preliminary training) have the discontinuity and non-smoothness issues. These issues are also the cause for the high nonlinearity of pole/residues from sample to sample. To achieve good training result, the number of hidden neurons is increased from 10 to 50. Such increase of the number of hidden neurons, however, leads to the overlearning of the ANN models, making the testing error even worse. The rational-based neuro-TF method obtains relatively smaller testing error than that obtained by the pole-residue-based

Figure 4.6: The original poles/residue data with discontinuity issue and the non-smoothness issue versus the newly obtained rational data with improved smoothness (after using the proposed hybrid parameter extraction technique) in Case 2 for the four-pole waveguide filter example. (a) the imaginary part of p_1 versus the training sample from the original poles/residue data, (b) the real part of r_3 versus the training sample from the original poles/residue data, (c) a_2 of the numerator versus the training sample from the newly obtained rational data.

neuro-TF method with the same number of hidden neurons. This is because that the rational data (which are converted from the pole-residue data for preliminary training) have better smoothness. However, the order of the rational-based transfer function is relatively high. The rational-based transfer function is more sensitive to its coefficients, and the coefficients are more nonlinear than the pole/residues over the increased geometrical parameter range from sample to sample. Therefore, the testing error of the rational-based neuro-TF model is still high. These are the reasons why the parametric models trained by the two existing methods cannot obtain good training and testing accuracy.

The proposed hybrid-based neuro-TF method can achieve better accuracy in training and testing than the two existing methods for both cases, as shown in the comparison in Table 4.2. It can be also noticed that the amount of hidden neurons used in the hybrid-based neuro-TF model is small in both cases. This indicates that the good learning of the model is achieved, leading to good training and testing results. The reason behind is that the poles/residues which have the discontinuity and non-smoothness issues in the original pole/residue data are reformatted into the new rational data (of which the smoothness is improved) with less order. The poles/residues which are smooth and continuous in the original pole/residue data are gathered together to format into the new pole/residue data (which are smooth) with less order. These new pole/residue data are used to train the pole-residue-based neuro-TF model and the new rational data are used to train the rational-based neuro-TF model during the preliminary training. Since there is no discontinuity and non-smoothness issues in the new pole/residue data, the pole-residue-based

neuro-TF model can achieve good learning with small amount of hidden neurons. Since the new rational data are smoother and with less order, the rational-based transfer function is less sensitive to its coefficients, leading to good learning of the rational-based neuro-TF model with small amount of hidden neurons as well. The preliminary trained pole-residue-based neuro-TF and rational-based neuro-TF models are then assembled into the hybrid-based neuro-TF model for the refinement training. These are the reasons why the parametric models trained by the proposed method can obtain good training and testing accuracy.

Fig. 4.7 shows the output S_{11} of the proposed hybrid-based neuro-TF model for three different test geometrical samples in the four-pole waveguide filter example. The S_{11} response from the proposed model is compared with the model responses using different methods and EM data. The geometrical variables for the three samples (#1, #2, and #3) are listed follows.

Test geometrical sample #1: $\mathbf{x} = [3.44 \ 4.30 \ 4.12 \ 3.12 \ 3.06]^T$ (mm)

Test geometrical sample #2: $\mathbf{x} = [3.52 \ 4.46 \ 3.96 \ 3.28 \ 3.06]^T$ (mm)

Test geometrical sample #3: $\mathbf{x} = [3.48 \ 4.46 \ 4.04 \ 3.32 \ 2.90]^T$ (mm)

It is observed from Fig. 4.7 that the model using the proposed hybrid-based neuro-TF method can achieve good model accuracy for different geometrical samples even though these test samples are never used in training. The good model accuracy was achieved at the cost of more CPU time in training, compared with existing pole-residue-based neuro-TF method. The amount of EM data and the EM data generation time in the proposed method are the same in the existing neuro-TF methods.

Table 4.2: Comparisons of Different Modeling Methods for The Four-Pole Waveguide Filter Example

Training Method		Number of Hidden Neurons		Average Training Error	Average Testing Error
Case 1 (Narrower Range)	Pole-Residue Neuro-TF Method	Pole NN	6	0.966 %	0.986 %
		Residue NN	6		
	Rational Neuro-TF Method	Numerator NN	10	5.094 %	5.221 %
		Denominator NN	10		
		Numerator NN	40	2.562 %	9.121 %
		Denominator NN	40		
	Proposed Hybrid Neuro-TF Method	Pole NN	4	0.837 %	0.909 %
		Residue NN	4		
		Numerator NN	6		
		Denominator NN	6		
Case 2 (Increased Range)	Pole-Residue Neuro-TF Method	Pole NN	10	8.823 %	9.413 %
		Residue NN	10		
		Pole NN	50	1.084 %	1299.1 %
		Residue NN	50		
	Rational Neuro-TF Method	Numerator NN	10	5.621 %	6.430 %
		Denominator NN	10		
		Numerator NN	50	1.951 %	24.263 %
		Denominator NN	50		
	Proposed Hybrid Neuro-TF Method	Pole NN	10	0.986 %	1.620 %
		Residue NN	10		
		Numerator NN	10		
		Denominator NN	10		

4.4.2 Parametric Modeling of a Three-Pole H -Plane Filter

In this example, the development of a parametric model for the EM behavior of a three-pole H -plane filter [53] using the proposed hybrid-based neuro-TF technique is illustrated. As shown in Fig. 4.8, the geometrical parameters of the three-pole

Figure 4.7: Comparison of $|S_{11}|$ (dB) of the models developed using different modeling methods and the *HFSS* EM data: (a) test geometrical sample #1, (b) test geometrical sample #2, and (c) test geometrical sample #3 for the four-pole waveguide filter example.

H -plane filter are $\mathbf{x} = [L_1 \ L_2 \ W_1 \ W_2]^T$. As defined in the model structure for the three-pole H -plane filter example in Fig. 4.9, the model has four input geometrical variables and a frequency variable (as an additional input). The model has two outputs, i.e., $\mathbf{y} = [RS_{11} \ IS_{11}]^T$, which are the real and imaginary parts of S_{11} , respectively. The *ANSYS HFSS* EM simulator is used to perform the full-wave EM simulation for generation of training and testing data. A parallel computation

Figure 4.8: The geometrical parameters of the three-pole H -plane filter and the 3D configuration for EM simulation. The geometrical parameters are L_1 , L_2 , W_1 , and W_2 , with $a = 19.05$ mm, $b = 9.525$ mm, and $t = 2.0$ mm.

Figure 4.9: The structure of the proposed hybrid-based neuro-TF model for the three-pole H -plane filter example.

method is used in data generation. DOE method is used as sampling method.

As can be seen in Table 4.3, the proposed parametric modeling technique is applied to two different cases, similar to the four-pole waveguide filter example. Case 1 considers a narrower parameter range and Case 2 considers an increased

parameter range. In both cases, the total order N of the hybrid-based transfer functions is set to ten for all the training and test samples. The number of frequency samples for each geometrical sample is 71. In both cases, seven levels of DOE are used to sample the training and testing data, resulting in a total number of 49 training samples and 49 testing samples. The detailed ranges of training data and testing data for the two cases are listed in Table 4.3.

The original frequency range in this example is 11.6 GHz - 12.4 GHz. Scaling and shifting of the frequency range are performed. The total order N is split into the order N_1 of the pole-residue-based transfer function and the order N_2 of the rational-based transfer function by using the proposed hybrid parameter extraction technique. Based on the pole/residue data (which are directly obtained from the vector fitting of training data), we identify the discontinuous poles/residues, non-smooth-continuous poles/residues, and the smooth-continuous poles/residues for both cases. The pole-residue data in Case 1 are relatively smooth as the values of the geometrical parameters change. By contrast, the pole-residue data in Case 2 have discontinuity issue in some of the imaginary parts of poles and residues, and have the accompanied non-smoothness issue in other poles and residues, as the values of the geometrical parameters change. The discontinuity issue and the non-smoothness issue in Case 2 can be observed in Fig. 4.10. As shown in the figure, the original poles/residue data have the discontinuity issue and the non-smoothness issue. After using the proposed extraction technique, the newly obtained rational data have better smoothness than the original poles/residue data. Based on the proposed hybrid parameter extraction technique illustrated in Subsection 4.3.2, N_1

Table 4.3: Definition of Training and Testing Data for The Three-Pole H -Plane Filter Example

Geometrical Variables		Training Data (49 samples)			Testing Data (49 samples)		
		Min	Max	Step	Min	Max	Step
Case 1 (Narrower Range)	L_1 (mm)	13.84	14.12	0.05	13.86	14.10	0.04
	L_2 (mm)	15.05	15.35	0.05	15.07	15.33	0.04
	W_1 (mm)	8.91	9.09	0.03	8.93	9.08	0.03
	W_2 (mm)	5.94	6.06	0.02	5.95	6.05	0.02
Case 2 (Increased Range)	L_1 (mm)	13.70	14.26	0.09	13.75	14.21	0.08
	L_2 (mm)	14.90	15.50	0.10	14.95	15.45	0.08
	W_1 (mm)	8.82	9.18	0.06	8.85	9.15	0.05
	W_2 (mm)	5.88	6.12	0.04	5.90	6.10	0.03

is set to six and N_2 is set to four in both cases. Similarly, the hybrid-based neuro-TF model is trained using the *NeuroModelerPlus* software for both preliminary training and refinement training. The average training error is 0.356% in Case 1 and the average testing error is 1.729% in Case 2.

For comparison purpose, the recent pole-residue-based neuro-TF modeling method and the existing rational-based neuro-TF modeling method are also applied to the two cases of this example. Table 4.4 shows the comparisons of ANN structures, number of hidden neurons, and average training and testing errors between several parametric modeling methods and EM data.

In Case 1, since the geometrical parameters vary within a small range, discontinuity and non-smoothness issues do not exist and the pole-residue data are relatively

Figure 4.10: The original poles/residue data with discontinuity issue and the non-smoothness issue versus the newly obtained rational data with improved smoothness (after using the proposed hybrid parameter extraction technique) in Case 2 for the three-pole H -plane filter example. (a) the imaginary part of p_1 versus the training sample from the original poles/residue data, (b) the real part of r_7 versus the training sample from the original poles/residue data, (c) a_2 of the numerator versus the training sample from the newly obtained rational data.

smooth. The transfer function order is moderate. For these reasons, all the methods in this case achieve comparatively small training and testing errors.

In Case 2, the geometrical parameters vary within an increased range and the pole/residue data have the discontinuity and non-smoothness issues. The pole-residue-based neuro-TF method, again, encounters these issues, resulting in large testing error. To achieve good training result, the number of hidden neurons is increased from 5 to 40. Such increase of the number of hidden neurons, however, leads to the overlearning of the ANN models, making the testing error even worse.

The rational-based neuro-TF method obtains relatively smaller testing error than that obtained by the pole-residue-based neuro-TF method. However, the order of the rational-based transfer function is not low enough and the geometrical variations are larger than those in Case 1. The rational-based transfer function is still very sensitive to its coefficients, and the coefficients are more nonlinear than the pole/residues over the increased geometrical parameter range from sample to sample. Thus, the testing error of the rational-based neuro-TF model is still high.

The proposed hybrid-based neuro-TF method can achieve better accuracy in training and testing than the two existing methods for both cases, as shown in the comparison in Table 4.2. It can be also noticed that the amount of hidden neurons used in the hybrid-based neuro-TF model is small in both cases. This suggests that the good learning of the model is achieved, leading to good training and testing results. In the proposed technique, there is no discontinuity and non-smoothness issues in the new pole/residue data and the new rational data are smoother and with less order. The poles/residues which are non-smooth and discontinuous in the

original pole/residue data are gathered together to reformat into the new rational data (which are much smoother) with less order. Since the new rational data are smoother and with less order, the rational-based transfer function is less sensitive to its coefficients. Therefore, both neuro-TF models in the hybrid-based neuro-TF model can achieve good learning with small amount of hidden neurons, leading to good training and testing accuracy of the overall model.

Fig. 4.11 shows the output S_{11} of the proposed hybrid-based neuro-TF model for three different test geometrical samples in the three-pole H -plane filter example. The S_{11} response from the proposed model is compared with the model responses using different methods and EM data. The geometrical variables for the three samples (#1, #2, and #3) are listed follows.

Test geometrical sample #1: $\mathbf{x} = [14.08 \ 15.12 \ 9.00 \ 5.93]^T$ (mm)

Test geometrical sample #2: $\mathbf{x} = [14.06 \ 15.20 \ 8.90 \ 6.10]^T$ (mm)

Test geometrical sample #3: $\mathbf{x} = [13.98 \ 15.37 \ 9.05 \ 5.90]^T$ (mm)

It is observed from Fig. 4.11 that the model using the proposed neuro-TF method can achieve good model accuracy for different geometrical samples even though these test samples are never used in training. The good model accuracy was achieved at the cost of more CPU time in training, compared with existing pole-residue-based neuro-TF method. The amount of EM data and the EM data generation time in the proposed method are the same in the existing neuro-TF methods. So far, the proposed method has been used to model the S_{11} responses of the two example microwave components, it is also possible to model S_{21} or other S -parameter responses by the similar modeling process.

Table 4.4: Comparisons of Different Modeling Methods for The Three-Pole H -Plane Filter Example

Training Method		Number of Hidden Neurons		Average Training Error	Average Testing Error
Case 1 (Narrower Range)	Pole-Residue Neuro-TF Method	Pole NN	10	0.128 %	0.174 %
		Residue NN	10		
	Rational Neuro-TF Method	Numerator NN	10	0.630 %	0.718 %
		Denominator NN	10		
	Proposed Hybrid Neuro-TF Method	Pole NN	10	0.100 %	0.356 %
		Residue NN	10		
		Numerator NN	10		
		Denominator NN	10		
Case 2 (Increased Range)	Pole-Residue Neuro-TF Method	Pole NN	5	21.751 %	29.376 %
		Residue NN	5		
		Pole NN	40	1.559 %	25927 %
		Residue NN	40		
	Rational Neuro-TF Method	Numerator NN	10	3.680 %	3.689 %
		Denominator NN	10		
		Numerator NN	50	1.596 %	4.340 %
		Denominator NN	50		
	Proposed Hybrid Neuro-TF Method	Pole NN	10	1.708 %	1.729 %
		Residue NN	10		
		Numerator NN	10		
		Denominator NN	10		

4.5 Conclusion

This chapter has presented a novel hybrid-based neuro-TF parametric modeling technique which systematically combines both pole-residue and rational formats of the transfer functions, addressing the discontinuity and non-smoothness issues. The proposed technique identifies the discontinuous poles/residues, the smooth-continuous poles/residues, and non-smooth-continuous poles/residues. The discon-

Figure 4.11: Comparison of $|S_{11}|$ (dB) of the models developed using different modeling methods and the *HFSS* EM data: (a) test geometrical sample #1, (b) test geometrical sample #2, and (c) test geometrical sample #3 for the three-pole *H*-plane filter example.

tinuous and non-smooth-continuous poles/residues are gathered together to reformat into the coefficients of the rational-based transfer function to effectively address the discontinuity and non-smoothness issues. The smooth-continuous poles/residues are gathered together to be remained in the pole-residue format of the pole-residue-based transfer function to maintain the capability of handling high-order problem. In this way, the discontinuity and non-smoothness issues are solved by the proposed

hybrid-based neuro-TF modeling technique and high-order modeling capability is maintained by the overall model. The developed model can provide accurate and fast prediction of the EM behavior of microwave components. As illustrated in the examples and compared with the existing modeling methods, the proposed technique can obtain better accuracy in challenging applications of large geometrical variations and high order, addressing the discontinuity and non-smoothness issues.

Chapter 5

Novel Decomposition Technique on Rational-Based Neuro-Transfer Function for Modeling of Microwave Components

5.1 Introduction

Rational-based neuro-transfer function (neuro-TF) method is a popular method for parametric modeling of electromagnetic (EM) behavior of microwave components. However, when the order in the neuro-TF becomes high, the sensitivities of the model response with respect to the coefficients of the transfer function become high. Due to this high-sensitivity issue, small training errors in the coefficients of the transfer function will result in large errors in the model output, leading to the difficulty in training of the neuro-TF model. This chapter proposes a new decomposition technique to address this high-sensitivity issue. In the proposed technique, we decompose the original neuro-TF model with high order of transfer function into multiple sub-neuro-TF models with much lower order of transfer function. We then

reformulate the overall model as the combination of the sub-neuro-TF models. New formulations are derived to determine the number of sub-models and the order of transfer function for each sub-model. Using the proposed decomposition technique, we can decrease the sensitivities of the overall model response with respect to the coefficients of the transfer function in each sub-model. Therefore, the modeling approach using the proposed decomposition technique can increase the modeling accuracy. Two EM parametric modeling examples are used to demonstrate the proposed decomposition technique.

5.2 The High-Sensitivity Issue in the Existing Rational-Based Neuro-TF Method

The existing rational-based neuro-TF modeling method [45] requires a parameter extraction process for extraction of the coefficients of the transfer function numerator and denominator in the neuro-TF model for each geometrical parameter sample. The existing method can be well-functioning if the geometrical variations are small and/or the order of the transfer function in the neuro-TF model is low. However, the existing neuro-TF method may not be suitable to address the situations where geometrical variations become large and/or where the order of the transfer function becomes high.

When the geometrical variations are small, the relationship between the geometrical variables and the coefficients of the transfer function reveals less nonlinearity. Since neural networks can easily learn this relationship, the training accuracy of the neural networks is high. This results in the high training accuracy of the overall

neuro-TF model. When the geometrical variations becomes larger, the relationship between the geometrical variables and the coefficients of the transfer function reveals more nonlinearity. The accuracy of the neural networks becomes lower. Under this situation, the order of the neuro-TF model becomes an important factor to the accuracy of the overall model. When the transfer function order becomes high, the sensitivities of the neuro-TF model response with respect to the coefficients of the transfer function become high. With the high sensitivity, a relatively small error of the neural networks will result in a large error of the overall neuro-TF model. How to solve this high-sensitivity issue to increase the accuracy of the rational-based neuro-TF is a challenging topic.

To address the high-sensitivity issue, this chapter proposes a novel decomposition technique for the rational-based neuro-TF modeling method. The main idea is that we decompose the neuro-TF model into sub-neuro-TF models and reformulate the overall neuro-TF model as the combination of the sub-neuro-TF models. In this way, the order of the overall neuro-TF model is decomposed into lower orders of the sub-neuro-TF models, decreasing the sensitivities of the overall model response with respect to the coefficients of the transfer function. By decreasing sensitivities of the overall model response with respect to coefficients of the transfer function using the proposed decomposition technique, the accuracy of the overall neuro-TF model can be improved.

5.3 Proposed Decomposition Technique for Development of Rational-Based Neuro-TF Model

5.3.1 Concept of the Decomposition Technique for Rational-Based Neuro-TF Model

Let $\mathbf{x}$ be a vector which contains the geometrical variables. Let y represent the frequency response, e.g., S -parameter. A general rational-based neuro-TF model [45] in frequency domain can be expressed as

$$y(\mathbf{x}, \mathbf{w}_a, \mathbf{w}_b, s) = \frac{\sum_{j=1}^N a_j(\mathbf{x}, \mathbf{w}_a) s^{j-1}}{1 + \sum_{j=1}^N b_j(\mathbf{x}, \mathbf{w}_b) s^j} \quad (5.1)$$

where N represents the transfer function order; a_j and b_j represent the j th coefficients of the transfer function numerator and denominator, respectively; $\mathbf{w}_a$ and $\mathbf{w}_b$ represent the neural network weights; and s is the Laplace domain frequency.

As can be seen from (5.1), the direct decomposition of the rational-based neuro-TF is not easy. In this case, we propose to reformat the rational-based transfer function and decompose the neuro-TF model indirectly. To reformat the rational-based transfer function, we use the pole-residue-based transfer function [51], which is another format of the transfer function. Here we define the effective poles and residues as the poles and residues whose imaginary parts are positive [51]. We define the effective order as the order of the pole-residue transfer function which consists only the effective poles and residues. In order to derive the decomposition technique for the rational-based neuro-TF model, we need to first convert the rational-based

transfer function into pole-residue format, expressed as

$$y = \frac{\sum_{j=1}^N a_j s^{j-1}}{1 + \sum_{j=1}^N b_j s^j} = \sum_{j=1}^{N_{eff}} \frac{r_j}{s - p_j} + \sum_{j=1}^{N_{eff}} \frac{r_j^*}{s - p_j^*} \quad (5.2)$$

where N_{eff} is the effective order of the pole-residue-based transfer function; p_j and r_j are the j th effective pole and residue of the pole-residue-based transfer function, respectively; and p_j^* and r_j^* are the complex conjugate parts of j th effective pole and residue, respectively. To ensure that all the poles and residues in (5.2) are in complex values, order N should be an even number and thus the effective order $N_{eff} = N/2$. It is noticed that the pole-residue-based transfer function itself is expressed as a summation of sub-transfer functions. We can easily decompose the pole-residue-based transfer function into multiple sub-transfer functions in pole-residue format. Let M represent the number of the sub-transfer functions. The decomposition of the pole-residue-based transfer function can be expressed as

$$y = \sum_{i=1}^M y_i = \sum_{i=1}^M \left(\sum_{j=1}^{N_i} \frac{r_{n(i,j)}}{s - p_{n(i,j)}} + \sum_{j=1}^{N_i} \frac{r_{n(i,j)}^*}{s - p_{n(i,j)}^*} \right) \quad (5.3)$$

where

$$n(i, j) = \begin{cases} j, & \text{if } i = 1 \\ j + \sum_{t=1}^{i-1} N_t, & \text{if } i > 1 \end{cases}, \quad (5.4)$$

y_i represents the frequency response of the i th sub-transfer function; and N_i represents the effective order of the i th sub-transfer function. The relationship between

N_{eff} , N_i , and M can be expressed as

$$N_{eff} = \sum_{i=1}^M N_i. \quad (5.5)$$

After the decomposition, we convert these sub-transfer functions back into rational format, expressed as

$$y_i = \sum_{j=1}^{N_i} \frac{r_{n(i,j)}}{s - p_{n(i,j)}} + \sum_{j=1}^{N_i} \frac{r_{n(i,j)}^*}{s - p_{n(i,j)}^*} = \frac{\sum_{j=1}^{2N_i} \hat{a}_{ij} s^{j-1}}{1 + \sum_{j=1}^{2N_i} \hat{b}_{ij} s^j} \quad (5.6)$$

where $\hat{a}_{ij}$ and $\hat{b}_{ij}$ represent the j th coefficients of the numerator and denominator for the i th sub-transfer function, respectively. Now the rational-based transfer function can be formulated as the summation of M sub-rational-based transfer functions. Substituting (5.6) into (5.3), we derive the rational-based neuro-TF model incorporating decomposition technique, formulated as

$$y(\mathbf{x}, \mathbf{w}_a, \mathbf{w}_b, s) = \sum_{i=1}^M y_i(\mathbf{x}, \mathbf{w}_a, \mathbf{w}_b, s) = \sum_{i=1}^M \frac{\sum_{j=1}^{2N_i} \hat{a}_{ij}(\mathbf{x}, \mathbf{w}_a) s^{j-1}}{1 + \sum_{j=1}^{2N_i} \hat{b}_{ij}(\mathbf{x}, \mathbf{w}_b) s^j}. \quad (5.7)$$

Table 5.1 illustrates the sensitivities of the proposed and the existing neuro-TF model responses with respect to coefficients of the transfer function. In Table 5.1, for the existing neuro-TF model formulated in (5.2), when the frequency range is small (e.g., varying around 1 to ignore the influence of s^j or s^{j-1}), all the effective poles (i.e., $p_j, \forall j \in N$) are close to each other. When the value of frequency s is in the

Table 5.1: Comparison of the Sensitivity of Rational-Based Neuro-TF Model Response with respect to the Coefficients with/without Decomposition.

	Coefficient	Transfer Function	Sensitivity of Transfer Function Response w.r.t the Coefficients
Existing Rational-Based Neuro-TF model (without Decomposition)	a_j	$y = \frac{\sum_{j=1}^N a_j s^{j-1}}{1 + \sum_{j=1}^N b_j s^j}$	$\frac{\partial y}{\partial a_j} = \frac{s^{j-1}}{1 + \sum_{j=1}^N b_j s^j} = \frac{s^{j-1}}{\prod_{j=1}^{N_{eff}} (s - p_j) \cdot \prod_{j=1}^{N_{eff}} (s - p_j^*)}$
	b_j		$\frac{\partial y}{\partial b_j} = \frac{-s^j \sum_{j=1}^N a_j s^{j-1}}{\left(1 + \sum_{j=1}^N b_j s^j\right)^2} = y \cdot \frac{-s^j}{\prod_{j=1}^{N_{eff}} (s - p_j) \cdot \prod_{j=1}^{N_{eff}} (s - p_j^*)}$
Proposed Rational-Based Neuro-TF model (with Decomposition)	$\hat{a}_{ij}$	$y = \sum_{i=1}^M y_i = \sum_{i=1}^M \frac{\sum_{j=1}^{2N_i} \hat{a}_{ij} s^{j-1}}{1 + \sum_{j=1}^{2N_i} \hat{b}_{ij} s^j}$	$\frac{\partial y}{\partial \hat{a}_{ij}} = \frac{s^{j-1}}{1 + \sum_{j=1}^{2N_i} \hat{b}_{ij} s^j} = \frac{s^{j-1}}{\prod_{j=1}^{N_i} (s - p_{n(i,j)}) \cdot \prod_{j=1}^{N_i} (s - p_{n(i,j)}^*)}$
	$\hat{b}_{ij}$		$\frac{\partial y}{\partial \hat{b}_{ij}} = \frac{-s^j \sum_{j=1}^{2N_i} \hat{a}_{ij} s^{j-1}}{\left(1 + \sum_{j=1}^{2N_i} \hat{b}_{ij} s^j\right)^2} = y_i \cdot \frac{-s^j}{\prod_{j=1}^{N_i} (s - p_{n(i,j)}) \cdot \prod_{j=1}^{N_i} (s - p_{n(i,j)}^*)}$

middle of all the effective poles, the value of $\prod_{j=1}^{N_{eff}} (s - p_j)$ becomes very small as N_{eff} is high, which makes the y too sensitive with respect to the coefficients a_j and b_j . This high-sensitivity issue can be effectively overcome by the neuro-TF model incorporating the proposed decomposition technique, formulated in (5.7). For the proposed neuro-TF model, the value of N_i is much smaller than that of N_{eff} . In that case, the value of $\prod_{j=1}^{N_i} (s - p_{n(i,j)})$ is much larger than that of $\prod_{j=1}^{N_{eff}} (s - p_j)$. Therefore, the sensitivities of y with respect to the coefficients $\hat{a}_{ij}$ and $\hat{b}_{ij}$ of the proposed neuro-TF model are much lower than those with respect to the coefficients a_j and b_j of the existing neuro-TF model. With lower sensitivities of the model response with respect to its coefficients of the transfer function, the proposed neuro-TF model can achieve higher accuracy than the existing neuro-TF model.

We use $\hat{\mathbf{a}}_i$ and $\hat{\mathbf{b}}_i$ to represent vectors which contain all the coefficients of the transfer function numerator and denominator in the i th sub-neuro-TF model, respectively, defined as

$$\hat{\mathbf{a}}_i = [\hat{a}_{i1} \ \hat{a}_{i2} \ \cdots \ \hat{a}_{ij} \ \cdots \ \hat{a}_{i(2N_i)}]^T \quad (5.8)$$

and

$$\hat{\mathbf{b}}_i = [\hat{b}_{i1} \ \hat{b}_{i2} \ \cdots \ \hat{b}_{ij} \ \cdots \ \hat{b}_{i(2N_i)}]^T \quad (5.9)$$

where $i = 1, 2, \dots, M$ and $j = 1, 2, \dots, 2N_i$. Based on the definition of coefficients $\hat{\mathbf{a}}_i$ and $\hat{\mathbf{b}}_i$, we use $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$ to represent vectors which contain all the coefficients of the transfer function numerators and denominators in all the sub-neuro-TF models, respectively, defined as

$$\hat{\mathbf{a}} = [\hat{\mathbf{a}}_1^T \ \hat{\mathbf{a}}_2^T \ \cdots \ \hat{\mathbf{a}}_i^T \ \cdots \ \hat{\mathbf{a}}_M^T]^T \quad (5.10)$$

and

$$\hat{\mathbf{b}} = [\hat{\mathbf{b}}_1^T \ \hat{\mathbf{b}}_2^T \ \cdots \ \hat{\mathbf{b}}_i^T \ \cdots \ \hat{\mathbf{b}}_M^T]^T \quad (5.11)$$

where $i = 1, 2, \dots, M$.

Fig. 5.1 shows the structure of the neuro-TF model incorporating the proposed decomposition technique. As can be seen in Fig. 5.1, the proposed model consists of two separate neural networks. Neural network $\hat{\mathbf{a}}(\mathbf{x}, \mathbf{w}_a)$ is used to learn and represent the nonlinear relationship between the geometrical variables $\mathbf{x}$ and the coefficients $\hat{\mathbf{a}}$ of the transfer function numerators in all the sub-neuro-TF models. Neural network $\hat{\mathbf{b}}(\mathbf{x}, \mathbf{w}_b)$ is used to learn and represent the nonlinear relationship

Figure 5.1: Structure for the rational-based neuro-TF model incorporating the proposed decomposition technique.

between the geometrical variables $\mathbf{x}$ and the coefficients $\hat{\mathbf{b}}$ of the transfer function denominators in all the sub-neuro-TF models.

In next subsection, we propose a novel decomposition technique for the parameter extraction and development of the proposed rational-based neuro-TF model. New formulations are derived to determine the number M of sub-neuro-TF models and the effective order N_i of the transfer function for each sub-neuro-TF model. Using the proposed decomposition technique, we can decrease the sensitivities of the overall model response with respect to the coefficients of the transfer function in each sub-model, achieving better overall model accuracy.

5.3.2 Proposed Decomposition Technique for Parameter Extraction and Model Development

In this subsection, we propose a decomposition technique for parameter extraction and model development. The proposed decomposition technique starts with the EM data samples, e.g., S -parameter data for different training samples of geometrical parameter. Let n_s represent the total number of training samples. Let T_r denote the index set of the training samples of the geometrical variables, i.e., $T_r = \{1, 2, \dots, n_s\}$. Let $\mathbf{x}_k$ and d_k represent the k th sample of the geometrical variables and EM data, respectively, where $k \in T_r$. During data generation, frequency is swept by the EM simulator as a separate variable.

We perform the vector fitting process [118], [119] to obtain poles and residues for each geometrical parameter sample. A scaling-and-shifting process for the frequency range with an even number of order N are set up during the vector fitting process to get all the poles and residues in complex values. The effective order N_{eff} is obtained as $N_{eff} = N/2$. Let $\tilde{\mathbf{p}}^{(k)}$ and $\tilde{\mathbf{r}}^{(k)}$ represent vectors which contain the poles and the residues obtained after vector fitting, respectively, at the k th geometrical sample, defined as

$$\tilde{\mathbf{p}}^{(k)} = [\tilde{p}_1^{(k)} \ \tilde{p}_2^{(k)} \ \dots \ \tilde{p}_{\tilde{\ell}}^{(k)} \ \dots \ \tilde{p}_N^{(k)}]^T \quad (5.12)$$

and

$$\tilde{\mathbf{r}}^{(k)} = [\tilde{r}_1^{(k)} \ \tilde{r}_2^{(k)} \ \dots \ \tilde{r}_{\tilde{\ell}}^{(k)} \ \dots \ \tilde{r}_N^{(k)}]^T \quad (5.13)$$

where $\tilde{p}_{\tilde{\ell}}^{(k)}$ and $\tilde{r}_{\tilde{\ell}}^{(k)}$ denote the $\tilde{\ell}$ th related pole and residue, respectively; and $\tilde{\ell} \in \tilde{I} = \{1, 2, \dots, N\}$.

In the first stage of the proposed decomposition technique, we sort the effective poles and residues in ascending sequence according to the values of their imaginary parts. Let the effective poles and residues after sorting be denoted as $\mathbf{p}^{(k)}$ and $\mathbf{r}^{(k)}$, respectively. The n th elements of $\mathbf{p}^{(k)}$ and $\mathbf{r}^{(k)}$ are denoted as $p_n^{(k)}$ and $r_n^{(k)}$, respectively, calculated as

$$p_n^{(k)} = \tilde{p}_{\ell_n}^{(k)} \quad (5.14)$$

and

$$r_n^{(k)} = \tilde{r}_{\ell_n}^{(k)} \quad (5.15)$$

where

$$\ell_n = \arg \min_{\tilde{\ell} \in D \setminus D_n} \left\{ \text{Im}(\tilde{p}_{\tilde{\ell}}^{(k)}) \right\} \quad (5.16)$$

$$D = \left\{ \tilde{\ell} \mid \text{Im}(\tilde{p}_{\tilde{\ell}}^{(k)}) > 0, \tilde{\ell} \in \tilde{I} \right\} \quad (5.17)$$

$$D_n = \begin{cases} \emptyset, & \text{if } n = 1 \\ D_{n-1} \cup \{\ell_{n-1}\}, & \text{if } n > 1 \end{cases} \quad (5.18)$$

and $n \in I = \{1, 2, \dots, N_{eff}\}$.

The second stage of the proposed decomposition technique is to quantify the degree of smoothness of the effective pole/residue data with respect to the changing of the geometrical variables. The quantified degree of smoothness can be used as indicators for decomposition. Let σ_n^p represent the deviation of the n th effective pole as the geometrical parameter changes. Similarly, let σ_n^r represent the deviation of the n th effective residue as the geometrical parameter changes. The deviations

σ_n^p and σ_n^r are formulated as

$$\sigma_n^p = \max_{k \in T_r} \left\{ \left(\left| \frac{\text{Re}(p_n^{(k)}) - \text{Re}(\mu_n^p)}{\text{Re}(\mu_n^p)} \right|^2 + \left| \frac{\text{Im}(p_n^{(k)}) - \text{Im}(\mu_n^p)}{\text{Im}(\mu_n^p)} \right|^2 \right)^{1/2} \right\} \quad (5.19)$$

and

$$\sigma_n^r = \max_{k \in T_r} \left\{ \left(\left| \frac{\text{Re}(r_n^{(k)}) - \text{Re}(\mu_n^r)}{\text{Re}(\mu_n^r)} \right|^2 + \left| \frac{\text{Im}(r_n^{(k)}) - \text{Im}(\mu_n^r)}{\text{Im}(\mu_n^r)} \right|^2 \right)^{1/2} \right\} \quad (5.20)$$

where μ_n^p and μ_n^r are expressed as

$$\mu_n^p = \frac{1}{n_s} \sum_{k=1}^{n_s} p_n^{(k)} \quad (5.21)$$

and

$$\mu_n^r = \frac{1}{n_s} \sum_{k=1}^{n_s} r_n^{(k)}. \quad (5.22)$$

Here we define the deviation vector $\boldsymbol{\sigma}$ as

$$\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \dots, \sigma_n, \dots, \sigma_{N_{eff}}]^T \quad (5.23)$$

where

$$\sigma_n = \sqrt{(\sigma_n^p)^2 + (\sigma_n^r)^2} \quad (5.24)$$

and $n \in I$. The deviation vector $\boldsymbol{\sigma}$ reflects the degree of smoothness of the effective pole/residue data ($\boldsymbol{p}^{(k)}$ and $\boldsymbol{r}^{(k)}$). It can be seen that the size of $\boldsymbol{\sigma}$ is consistent with the size of index set $I = \{1, 2, \dots, N_{eff}\}$. Based on the different values of σ_n , the index set I will be decomposed into multiple subsets for parameter extraction.

In the third stage, we decompose the index set $I = \{1, 2, \dots, N_{eff}\}$ into M

subsets based on σ and M . The initial value of M is set to two. Since the size of the i th subset depends on the effective order N_i of the i th sub-neuro-TF model, we determine the effective order N_i , formulated as

$$N_i = \begin{cases} \left\lceil \frac{N_{eff}}{M} \right\rceil, & \text{if } i = 1 \\ \left\lceil \frac{N_{eff} - \sum_{j=1}^{i-1} N_j}{M - i + 1} \right\rceil, & \text{if } i > 1 \end{cases} \quad (5.25)$$

where $i = 1, 2, \dots, M$. By using (5.25), N_i is distributed as evenly as possible for each sub-neuro-TF model.

After the determination of N_i , we sort the elements of the deviation vector σ in descending sequence. Let $\hat{\mathbf{u}}$ denote a vector which contains the indices of the elements of σ after sorting. Let $\hat{u}_m$ denote the m th element of $\hat{\mathbf{u}}$, formulated as

$$\hat{u}_m = \arg \max_{u \in I \setminus J_m} \{\sigma_u\} \quad (5.26)$$

where

$$J_m = \begin{cases} \emptyset, & \text{if } m = 1 \\ J_{m-1} \cup \{\hat{u}_{m-1}\}, & \text{if } m > 1 \end{cases} \quad (5.27)$$

and $m \in I$. We define the i th subset $I_i \subseteq I$, where $i = 1, 2, \dots, M$; $|I_i| = N_i$; $\bigcup_{i=1}^M I_i = I$; and $\left| \bigcup_{i=1}^M I_i \right| = |I| = N_{eff}$. Using the information of the index vector $\mathbf{u}$ and the effective order N_i , the i th subset I_i can be expressed as

$$I_i = \{\hat{u}_{q+1}, \hat{u}_{q+2}, \dots, \hat{u}_{q+N_i}\} \quad (5.28)$$

where q can be formulated as

$$q = \begin{cases} 0, & \text{if } i = 1 \\ \sum_{t=1}^{i-1} N_t, & \text{if } i > 1 \end{cases}. \quad (5.29)$$

In the fourth stage, we group the effective poles $\mathbf{p}^{(k)}$ and their complex conjugate parts $(\mathbf{p}^*)^{(k)}$ into pole subsets and group the effective residues $\mathbf{r}^{(k)}$ and their complex conjugate parts $(\mathbf{r}^*)^{(k)}$ into residue subsets, based on the information of subsets $I_i, \forall i = 1, 2, \dots, M$. Let $\hat{\mathbf{p}}_i^{(k)}$ and $\hat{\mathbf{r}}_i^{(k)}$ denote the pole and residue data for the i th sub-neuro-TF model, respectively, expressed as

$$\hat{\mathbf{p}}_i^{(k)} = \begin{bmatrix} p_{\hat{u}_{q+1}}^{(k)} & (p_{\hat{u}_{q+1}}^*)^{(k)} & p_{\hat{u}_{q+2}}^{(k)} & (p_{\hat{u}_{q+2}}^*)^{(k)} & \dots & p_{\hat{u}_{q+N_i}}^{(k)} & (p_{\hat{u}_{q+N_i}}^*)^{(k)} \end{bmatrix}_{1 \times 2N_i}^T \quad (5.30)$$

and

$$\hat{\mathbf{r}}_i^{(k)} = \begin{bmatrix} r_{\hat{u}_{q+1}}^{(k)} & (r_{\hat{u}_{q+1}}^*)^{(k)} & r_{\hat{u}_{q+2}}^{(k)} & (r_{\hat{u}_{q+2}}^*)^{(k)} & \dots & r_{\hat{u}_{q+N_i}}^{(k)} & (r_{\hat{u}_{q+N_i}}^*)^{(k)} \end{bmatrix}_{1 \times 2N_i}^T \quad (5.31)$$

where $i = 1, 2, \dots, M$; and q is calculated by (5.29).

We convert these grouped pole and residue data (i.e., $\hat{\mathbf{p}}_i^{(k)}$ and $\hat{\mathbf{r}}_i^{(k)}$) into the coefficient data of the transfer function numerator $\hat{\mathbf{a}}_i^{(k)}$ and the transfer function denominator $\hat{\mathbf{b}}_i^{(k)}$ for the i th sub-neuro-TF model. Since the nonlinear relationships between the geometrical variables $\mathbf{x}$ and the coefficients of the numerators $\hat{\mathbf{a}}_i$ are usually similar among different sub-neuro-TF models, we use one single neural network $\hat{\mathbf{a}}(\mathbf{x}, \mathbf{w}_a)$ to learn the relationships between the geometrical variables $\mathbf{x}$ and the coefficients of the numerators $\hat{\mathbf{a}}$ of all the sub-neuro-TF models. We use another single neural network $\hat{\mathbf{b}}(\mathbf{x}, \mathbf{w}_b)$ to learn the relationships between the geometrical

variables $\mathbf{x}$ and the coefficients of the denominators $\hat{\mathbf{b}}$ of all the sub-neuro-TF models. Let $(\mathbf{x}_k, \hat{\mathbf{a}}^{(k)})$ and $(\mathbf{x}_k, \hat{\mathbf{b}}^{(k)})$ denote the training data which are used to train the two neural networks $\hat{\mathbf{a}}(\mathbf{x}, \mathbf{w}_a)$ and $\hat{\mathbf{b}}(\mathbf{x}, \mathbf{w}_b)$, respectively, expressed as

$$\hat{\mathbf{a}}^{(k)} = \left[(\hat{\mathbf{a}}_1^{(k)})^T \quad (\hat{\mathbf{a}}_2^{(k)})^T \quad \dots \quad (\hat{\mathbf{a}}_i^{(k)})^T \quad \dots \quad (\hat{\mathbf{a}}_M^{(k)})^T \right]^T \quad (5.32)$$

and

$$\hat{\mathbf{b}}^{(k)} = \left[(\hat{\mathbf{b}}_1^{(k)})^T \quad (\hat{\mathbf{b}}_2^{(k)})^T \quad \dots \quad (\hat{\mathbf{b}}_i^{(k)})^T \quad \dots \quad (\hat{\mathbf{b}}_M^{(k)})^T \right]^T \quad (5.33)$$

where $i = 1, 2, \dots, M$.

In the fifth stage, we train the proposed neuro-TF model. First, a preliminary training process[49] is performed for the two neural networks $\hat{\mathbf{a}}(\mathbf{x}, \mathbf{w}_a)$ and $\hat{\mathbf{b}}(\mathbf{x}, \mathbf{w}_b)$ using the obtained training data $(\mathbf{x}_k, \hat{\mathbf{a}}^{(k)})$ and $(\mathbf{x}_k, \hat{\mathbf{b}}^{(k)})$, respectively. The two neural networks are trained separately to learn the relationships between the geometrical variables $\mathbf{x}$ and the coefficients (i.e., $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$). Next, we perform a model refinement training process[49] to refine the overall proposed model. The training data for the proposed overall model are $(\mathbf{x}_k, \mathbf{d}_k)$, where $k \in T_r$. The objective of training is to minimize the training error E_{Tr} of the overall model by optimizing the neural network weights $\mathbf{w}_a$ and $\mathbf{w}_b$. The training error function E_{Tr} is formulated as

$$E_{Tr}(\mathbf{w}_a, \mathbf{w}_b) = \frac{1}{2n_s} \sum_{k \in T_r} \sum_{m \in \Omega} \left\| \sum_{i=1}^M y_i(\mathbf{w}_a, \mathbf{w}_b, \mathbf{x}_k, s_m) - d_{k,m} \right\|^2 \quad (5.34)$$

where n_s is the total number of training samples; Ω is the index set of frequency samples; M is the number of the sub-neuro-TF models; and y_i is the output of the i th sub-neuro-TF model, which is also a function of $\mathbf{x}_k$, $\mathbf{w}_a$ and $\mathbf{w}_b$, and s_m .

We use the training data $(\mathbf{x}_k, \mathbf{d}_k)$ to verify the quality of trained overall model. If the training error E_{Tr} is lower than a user-defined threshold E_t , the whole training process is terminated and the model is ready for testing. Otherwise, the two neural networks are underlearned. In that case, we should increase the number of hidden neurons for the two neural networks and repeat the two training processes.

After the model is trained, we test the quality of model using independent testing data which are never used in training. We define E_{Ts} to be the testing error. If the testing error E_{Ts} is also lower than the threshold E_t , the model development is finished and the current value of M is regarded as the final M . Otherwise, we increase the number M of sub-neuro-TF models by one (i.e., $M = M + 1$) and repeat the parameter extraction and training processes from the third stage to the fifth stage (5.25)–(5.34) of the proposed decomposition technique.

Here we summarize the proposed decomposition technique in a stepwise algorithm as follows.

Step 1: Perform the first stage of the proposed decomposition technique to select the effective poles and residues $(\mathbf{p}^{(k)})$ and $(\mathbf{r}^{(k)})$ by (5.12)–(5.18).

Step 2: Perform the second stage of the proposed decomposition technique to obtain the deviation vector $\boldsymbol{\sigma}$ by (5.19)–(5.24).

Step 3: Initialize the number M of the sub-neuro-TF models to two.

Step 4: Perform the third stage of the proposed decomposition technique with the current value of M to obtain the effective order N_i by (5.25) and subset I_i by (5.26)–(5.29) for the i th sub-neuro-TF model.

Step 5: Perform the fourth stage of the proposed decomposition technique to obtain

the pole and residue data $\hat{\mathbf{p}}_i^{(k)}$ and $\hat{\mathbf{r}}_i^{(k)}$ by (5.30)–(5.31) for the i th sub-neuro-TF model. Convert data $\hat{\mathbf{p}}_i^{(k)}$ and $\hat{\mathbf{r}}_i^{(k)}$ into data $\hat{\mathbf{a}}_i^{(k)}$ and $\hat{\mathbf{b}}_i^{(k)}$. Obtain the training data $(\mathbf{x}_k, \hat{\mathbf{a}}^{(k)})$ and $(\mathbf{x}_k, \hat{\mathbf{b}}^{(k)})$ by (5.32)–(5.33) for the two neural networks $\hat{\mathbf{a}}(\mathbf{x}, \mathbf{w}_a)$ and $\hat{\mathbf{b}}(\mathbf{x}, \mathbf{w}_b)$.

Step 6: Perform the fifth stage of the proposed decomposition technique, i.e. the preliminary training of the two neural networks and refinement training of the overall model by (5.34).

Step 7: Use the training data $(\mathbf{x}_k, \mathbf{d}_k)$ to verify the trained overall model. If the training error E_{Tr} is lower than a user-defined threshold E_t , go to *Step 8*. Otherwise, increase the number of hidden neurons and go to *Step 6*.

Step 8: Use the testing data to verify the overall model. If the testing error E_{Ts} is lower than the user-defined threshold E_t , go to *Step 9*. Otherwise, increase the number M by one (i.e., $M = M + 1$) and go to *Step 4*.

Step 9: Stop the modeling process.

5.4 Application Examples

5.4.1 Three-Order Waveguide Filter Modeling

A three-order waveguide filter [53] is used to illustrate the rational-based neuro-TF method using the proposed decomposition technique. As can be seen in Fig. 5.2, the geometrical variables are $\mathbf{x} = [L_1 \ L_2 \ W_1 \ W_2]^T$ for the three-order waveguide filter. Four geometrical variables, i.e., $\mathbf{x} = [L_1 \ L_2 \ W_1 \ W_2]^T$, are used as model inputs and the real and imaginary parts of S_{11} , i.e., $\mathbf{y} = [RS_{11} \ IS_{11}]^T$, are used as model outputs. Full-wave EM simulations are performed using the *ANSYS HFSS*

Figure 5.2: Geometrical variables and 3D configuration of the three-order waveguide filter example. $a = 19.05$ mm, $b = 9.525$ mm, and $t = 2.0$ mm.

EM simulator to generate training and test samples for parametric modeling. Both training samples and test samples are generated using the design of experiments (DOE) [121] sampling method.

As shown in Table 5.2, the proposed parametric modeling technique is performed for three cases according to different geometrical parameter ranges. Case 1 is with a narrower range, Case 2 is with an increased range, and Case 3 is with a wider range. In all cases, we set the total order N of the overall transfer function to 10 for every samples. Subsequently, N_{eff} equals five for the three cases. In Case 1 and Case 2, the frequency range is from 11.65 GHz to 12.35 GHz and the number of frequency samples per geometrical sample equals 71. In Case 3, the frequency range is from 11.0 GHz to 13.0 GHz and the number of frequency samples per geometrical sample equals 201. We use seven levels of DOE for all cases to generate the training and test samples. The numbers for training samples and testing samples are both 49. Table 5.2 lists the specific ranges of training samples and test samples for the three cases. We scale and shift the frequency range and set the total order N to

Table 5.2: Definition of Training and Test Samples for The Three-Order Waveguide Filter Example

Geometrical Parameters (mm)		Training Samples (49 samples)			Test Samples (49 samples)		
		Min	Max	Steps	Min	Max	Steps
Case 1 (Narrower Range)	L_1	13.84	14.12	0.05	13.86	14.10	0.04
	L_2	15.05	15.35	0.05	15.07	15.33	0.04
	W_1	8.91	9.09	0.03	8.93	9.08	0.03
	W_2	5.94	6.06	0.02	5.95	6.05	0.02
Case 2 (Increased Range)	L_1	13.70	14.26	0.09	13.75	14.21	0.08
	L_2	14.90	15.50	0.10	14.95	15.45	0.08
	W_1	8.82	9.18	0.06	8.85	9.15	0.05
	W_2	5.88	6.12	0.04	5.90	6.10	0.03
Case 3 (Wider Range)	L_1	13.28	14.68	0.23	13.40	14.56	0.19
	L_2	14.44	15.96	0.25	14.57	15.83	0.21
	W_1	8.55	9.45	0.15	8.63	9.38	0.13
	W_2	5.70	6.30	0.10	5.75	6.25	0.08

ten (an even number) during the vector fitting process, ensuring that the obtained poles and residues are in complex values.

Since the initial value of the number M of sub-neuro-TF models is set to two, the proposed model is initially trained with $M=2$. Using the proposed decomposition technique with $M=2$, the effective orders are determined as $N_1=3$ and $N_2=2$ for all the three cases and the subsets are determined as $I_1=\{2, 4, 5\}$ and $I_2=\{1, 3\}$ for Case 1 and Case 2 and $I_1=\{2, 3, 4\}$ and $I_2=\{1, 5\}$ for Case 3. Fig. 5.3 illustrates the

Figure 5.3: Structure for the proposed rational-based neuro-TF model with $M = 2$ of the three-order waveguide filter example.

model structure with $M=2$ for the three-order waveguide filter example. As defined in Fig. 5.3, the model has a frequency variable for each sub-neuro-TF model, four input geometrical variables, and two outputs. The overall transfer function consists of two sub-transfer functions. The high effective order N_{eff} of 5 is decomposed into lower effective orders N_1 of 3 and N_2 of 2.

NeuroModelerPlus software is used for training of the proposed model. The user-defined threshold E_t for the three-order waveguide filter example is set to 1%. Three parametric models are developed for the three cases using the proposed technique. The average training errors of the trained models with $M = 2$ are 0.239%, 0.467%, and 1.490% for Case 1, Case 2, and Case 3, respectively. The average testing errors of the trained models with $M = 2$ are 0.267%, 0.496%, and 1.840% for Case 1, Case

2, and Case 3, respectively. On the one hand, since the training and testing errors for both Case 1 and Case 2 are lower than E_t , the two trained models with $M = 2$ are the final models for both Case 1 and Case 2. On the other hand, since the testing error for Case 3 is higher than E_t , we increased M to three and redeveloped the model with $M = 3$ for Case 3. For the proposed neuro-TF model with $M=3$ in the three-order waveguide filter example, the overall transfer function consists of three sub-transfer functions. The high effective order N_{eff} of 5 is decomposed into lower effective orders N_1 of 2, N_2 of 2, and N_3 of 1. The subsets are determined as $I_1=\{2, 4\}$, $I_2=\{3, 5\}$, and $I_3=\{1\}$ for Case 3. The average training and testing errors of the trained models with $M = 3$ are 0.746% and 0.962%, which are lower than E_t . Therefore, the trained model with $M = 3$ is the final model for Case 3.

For comparison purpose, we apply the existing modeling approach using rational-based neuro-TF [45] to the three separated cases of this example. The comparisons of number M , effective order N_i , hidden neuron numbers, and average errors for training and testing between different rational-based neuro-TF modeling methods and EM data are shown in Table 5.3.

In Case 1, since the geometrical variations are small, the relatively low nonlinear relationship between the geometrical variables and the coefficients of the transfer function can be easily represented by the neural networks with less hidden neurons of both the existing model and the proposed model in the preliminary training. With the high training accuracy of the neural networks of both models, both models can have high training accuracy in the refinement training. For these reasons, both methods in this case can obtain good training and testing accuracy.

Table 5.3: Comparisons of Different Rational-Based Neuro-TF Modeling Approaches for The Three-Order Waveguide Filter Example

Training Method		No. of sub-Models	N_{eff} or N_i	No. of Hidden Neurons		Average Training Error	Average Testing Error
Case 1 (Narrower Range)	Existing Rational Neuro-TF Method	1	5	Numerator NN	10	0.630 %	0.718 %
				Denominator NN	10		
	Proposed Rational Neuro-TF Method	2	3 2	Numerator NN	10	0.239 %	0.267 %
				Denominator NN	10		
Case 2 (Increased Range)	Existing Rational Neuro-TF Method	1	5	Numerator NN	10	1.953 %	2.017 %
				Denominator NN	10		
	Proposed Rational Neuro-TF Method	2	3 2	Numerator NN	10	0.467 %	0.496 %
				Denominator NN	10		
Case 3 (Wider Range)	Existing Rational Neuro-TF Method	1	5	Numerator NN	10	5.073 %	6.604 %
				Denominator NN	10		
		1	5	Numerator NN	40	3.222 %	50.69 %
				Denominator NN	40		
	Proposed Rational Neuro-TF Method	2	3 2	Numerator NN	10	1.490 %	1.840 %
				Denominator NN	10		
		3	2 2 1	Numerator NN	10	0.746 %	0.962 %
				Denominator NN	10		

In Case 2, since the geometrical variables vary within an increased range, the relationship between the geometrical variables and the coefficients of the transfer function reveals more nonlinearity than that in Case 1. In this case, the sensitivities of the overall model response with respect to the coefficients of the transfer function have higher effects on the accuracy of the overall model. Since the orders and the sensitivities for the proposed model are much lower than those for the existing model, the proposed model can obtain more accurate training and testing results than the existing method.

In Case 3, the geometrical variables vary within a wider range, the relationship between the geometrical variables and the coefficients of the transfer function reveals much higher nonlinearity. In this case, the sensitivities of the overall model response with respect to the coefficients of the transfer function have large effects on the accuracy of the overall model. The proposed model with $M = 2$ have smaller training and testing errors than the existing model but still can not satisfy the model accuracy criteria. Therefore, we increased the M to be 3. With $M = 3$, the proposed model can satisfy the model accuracy criteria.

It can be concluded from the above three cases that the neuro-TF model using the proposed decomposition technique has much lower sensitivities of the overall model response with respect to the coefficients of the transfer function, improving the modeling accuracy over the existing method.

Fig. 5.4 shows the output $|S_{11}|$ in dB of the proposed neuro-TF model for three different test geometrical samples in Case 3 of the three-order waveguide filter example. We compare the model response using the proposed technique with $M = 3$, using the proposed technique with $M = 2$, using the existing neuro-TF method, and EM data. The geometrical variables for the three test samples are $\mathbf{x} = [14.37 \ 14.57 \ 9.25 \ 6.17]^T$ (mm), $\mathbf{x} = [14.17 \ 15.20 \ 8.75 \ 6.25]^T$ (mm), and $\mathbf{x} = [14.56 \ 15.20 \ 8.63 \ 6.17]^T$ (mm). It is observed from Fig. 5.4 that the neuro-TF model using the proposed decomposition method can achieve better model accuracy for different geometrical samples which are never used in training.

Figure 5.4: Comparison of $|S_{11}|$ (dB) of S_{11} different modeling approaches and EM data: (a) test sample $\mathbf{x} = [14.37 \ 14.57 \ 9.25 \ 6.17]^T$ (mm), (b) test sample $\mathbf{x} = [14.17 \ 15.20 \ 8.75 \ 6.25]^T$ (mm), and (c) test sample $\mathbf{x} = [14.56 \ 15.20 \ 8.63 \ 6.17]^T$ (mm) for the three-order waveguide filter example.

5.4.2 Four-Order Bandpass Filter Modeling

A four-order bandpass filter [120] is used to illustrate the rational-based neuro-TF method using the proposed decomposition technique. As can be seen in Fig. 5.5, the geometrical variables are $\mathbf{x} = [h_1 \ h_2 \ h_3 \ h_{c1} \ h_{c2}]^T$ for the four-order bandpass filter. Five geometrical variables, i.e., $\mathbf{x} = [h_1 \ h_2 \ h_3 \ h_{c1} \ h_{c2}]^T$, are used as model inputs and the real and imaginary parts of S_{11} , i.e., $\mathbf{y} = [RS_{11} \ IS_{11}]^T$ are used as model

Figure 5.5: Geometrical variables and 3D configuration of the four-order bandpass filter example.

outputs. Similar to the first example, full-wave EM simulations are performed using the *ANSYS HFSS* EM simulator to generate training and test samples for parametric modeling. Both training samples and test samples are generated using DOE sampling method.

As shown in Table 5.4, the proposed parametric modeling technique is performed for two different cases according to different geometrical parameter ranges. Case 1 is with a narrower parameter range and Case 2 is with a wider parameter range. In both cases, we set the total order N of the overall transfer function to 12 for every samples. Subsequently, N_{eff} equals six for the two cases. In both cases, the frequency range is from 10.50 GHz to 11.50 GHz and the number of frequency samples per geometrical sample equals 101. We use nine levels of DOE to generate the training samples and eight levels of DOE to generate the test samples for both cases. The numbers for the training samples and the testing samples are 81 and 64, respectively. Table 5.4 lists the specific ranges of training samples and test samples for the two cases. We scale and shift the frequency range to ensure that the obtained poles and residues are in complex values.

Table 5.4: Definition of Training and Test Samples for The Four-Order Bandpass Filter Example

Geometrical Parameters (mm)		Training Samples (81 samples)			Test Samples (64 samples)		
		Min	Max	Steps	Min	Max	Steps
Case 1 (Narrower Range)	h_1	3.4	3.56	0.02	3.41	3.55	0.02
	h_2	4.3	4.46	0.02	4.31	4.45	0.02
	h_3	4.0	4.16	0.02	4.01	4.15	0.02
	h_{c1}	3.2	3.36	0.02	3.21	3.35	0.02
	h_{c2}	2.9	3.06	0.02	2.91	3.05	0.02
Case 2 (Wider Range)	h_1	3.3	3.62	0.04	3.32	3.6	0.04
	h_2	4.2	4.52	0.04	4.22	4.5	0.04
	h_3	3.9	4.22	0.04	3.92	4.2	0.04
	h_{c1}	3.1	3.42	0.04	3.12	3.4	0.04
	h_{c2}	2.8	3.12	0.04	2.82	3.1	0.04

Similar to the first example, the proposed model is initially trained with $M = 2$. With $M = 2$, the effective orders are determined as $N_1 = 3$ and $N_2 = 3$ for both cases and the subsets are determined as $I_1 = \{2, 4, 5\}$ and $I_2 = \{1, 3, 6\}$ for both cases. Fig. 5.6 illustrates the model structure with $M=2$ for the four-order bandpass filter example. As defined in Fig. 5.6, the model has a frequency variable for each sub-neuro-TF model, five input geometrical variables, and two outputs. The overall transfer function consists of two sub-transfer functions. The high effective order N_{eff} of 6 is decomposed into lower effective orders N_1 of 3 and N_2 of 3.

NeuroModelerPlus software is used for training of the proposed model. The user-defined threshold E_t for the four-order bandpass filter example is set to 2%. Two parametric models are developed for the two cases using the proposed technique.

Figure 5.6: Structure for the proposed rational-based neuro-TF model with $M = 2$ of the four-order bandpass filter example.

The average training errors of the trained models with $M = 2$ are 1.487% and 2.252% for Case 1 and Case 2, respectively. The average testing errors of the trained models with $M = 2$ are 1.672% and 4.397% for Case 1 and Case 2, respectively. On the one hand, since the training and testing errors for Case 1 are lower than E_t , the trained model with $M = 2$ is the final model for Case 1. On the other hand, since the testing error for Case 2 is higher than E_t , we increased M to three and redeveloped the model with $M = 3$ for Case 2. For the proposed neuro-TF model with $M=3$ in the four-order bandpass filter example, the overall transfer function consists of three sub-transfer functions. The high effective order N_{eff} of 6 is decomposed into lower effective orders N_1 of 2, N_2 of 2, and N_3 of 2. The subsets are determined as $I_1 = \{2, 5\}$, $I_2 = \{3, 4\}$, and $I_3 = \{1, 6\}$ for Case 2. The average training and testing errors of the trained model with $M = 3$ are 1.624% and 1.982%, respectively, which

Table 5.5: Comparisons of Different Rational-Based Neuro-TF Modeling Approaches for The Four-Order Bandpass Filter Example

Modeling Methods		No. of Sub-Models	N_{eff} or N_i	No. of Hidden Neurons		Average Training Error	Average Testing Error
Case 1 (Narrower Range)	Existing Rational Neuro-TF Method	1	6	Numerator NN	10	4.448 %	4.674 %
				Denominator NN	10		
		1	6	Numerator NN	40	2.562 %	9.121 %
				Denominator NN	40		
	Proposed Rational Neuro-TF Method	2	3 3	Numerator NN	10	1.487 %	1.672%
				Denominator NN	10		
		3	2 2 2	Numerator NN	10	0.876 %	1.015 %
				Denominator NN	10		
Case 2 (Wider Range)	Existing Rational Neuro-TF Method	1	6	Numerator NN	10	6.809 %	8.382 %
				Denominator NN	10		
		1	6	Numerator NN	40	4.791 %	32.23 %
				Denominator NN	40		
	Proposed Rational Neuro-TF Method	2	3 3	Numerator NN	10	2.252 %	4.397 %
				Denominator NN	10		
		2	3 3	Numerator NN	40	1.877 %	16.21 %
				Denominator NN	40		
		3	2 2 2	Numerator NN	10	1.624 %	1.982 %
				Denominator NN	10		

are lower than E_t . Therefore, the trained model with $M = 3$ is the final model for Case 2.

For comparison purpose, we apply the existing modeling approach using rational-based neuro-TF to the two separated cases of this example. The comparisons of number M , effective order N_i , hidden neuron numbers, and average errors for training and testing between different rational-based neuro-TF modeling methods and EM data are shown in Table 5.5.

In Case 1, since the geometrical variations are relatively small, the relationship between the geometrical variables and the coefficients of the transfer function reveals moderate nonlinearity. In this case, the sensitivities of the overall model response with respect to the coefficients of the transfer function have moderate effects on the accuracy of the overall model. Since the orders and the sensitivities for the proposed model are much lower than those for the existing model, the proposed model can obtain more accurate training and testing results than the existing method.

In Case 2, the geometrical variables vary within a wider range, the relationship between the geometrical variables and the coefficients of the transfer function reveals more nonlinearity. In this case, the sensitivities of the overall model response with respect to the coefficients of the transfer function have large effects on the accuracy of the overall model. The proposed model with $M = 2$ have smaller training and testing errors than the existing model but still can not satisfy the model accuracy criteria. Therefore, we increased the M to be 3. With $M = 3$, the proposed model can satisfy the model accuracy criteria.

It can be again concluded from the above two cases that the neuro-TF model using the proposed decomposition technique has much lower sensitivities of the overall model response with respect to the coefficients of the transfer function, improving the modeling accuracy over the existing method.

Fig. 5.7 shows the output $|S_{11}|$ in dB of the proposed neuro-TF model for three different test geometrical samples in Case 2 of the four-order bandpass filter example. We compare the model response using the proposed technique with $M = 3$, using the proposed technique with $M = 2$, using the existing neuro-TF method,

Figure 5.7: Comparison of $|S_{11}|$ (dB) of S_{11} for different modeling approaches and EM data: (a) test sample $\mathbf{x} = [3.52 \ 4.46 \ 3.96 \ 3.28 \ 3.07]^T$ (mm), (b) test sample , $\mathbf{x} = [3.60 \ 4.22 \ 4.20 \ 3.40 \ 3.10]^T$ (mm), and (c) test sample $\mathbf{x} = [3.36 \ 4.38 \ 4.00 \ 3.36 \ 3.02]^T$ (mm) for the four-order bandpass filter example.

and EM data. We provide the comparisons of $\text{Re}(S_{11})$ and $\text{Im}(S_{11})$, as shown in Fig. 5.8. The geometrical variables for the three test samples are $\mathbf{x} = [3.52 \ 4.46 \ 3.96 \ 3.28 \ 3.07]^T$ (mm), $\mathbf{x} = [3.60 \ 4.22 \ 4.20 \ 3.40 \ 3.10]^T$ (mm), and $\mathbf{x} = [3.36 \ 4.38 \ 4.00 \ 3.36 \ 3.02]^T$ (mm). It is observed from Fig. 5.7 and Fig. 5.8 that the neuro-TF model using the proposed decomposition method can achieve better model accuracy. So far, the proposed method has been used to model the S_{11} responses

Figure 5.8: Comparison of $\text{Re}(S_{11})$ and $\text{Im}(S_{11})$ for different modeling approaches and EM data at test sample $\mathbf{x} = [3.36 \ 4.38 \ 4.00 \ 3.36 \ 3.02]^T$ (mm) for the four-order bandpass filter example. (a) $\text{Re}(S_{11})$ and (b) $\text{Im}(S_{11})$.

of the two example microwave components, it is also possible to model S_{21} or other S -parameter responses by the similar modeling process.

5.5 Conclusion

This chapter has proposed a novel decomposition technique for rational-based neuro-TF modeling method. The proposed technique has decomposed the single neuro-TF model with high order into multiple sub-neuro-TF models with much lower order and reformulated the overall model as the combination of the sub-models. New formulations have been derived for determination of the number of sub-models and the order of each sub-model. Since the high order of the overall model has been decomposed into multiple lower orders of the sub-models, the proposed technique has decreased the sensitivities of the overall model response with respect to the coefficients of the transfer function, improving the overall model accuracy.

Chapter 6

Conclusions and Future Research

6.1 Conclusions

In this thesis, advanced space mapping and ANN-based techniques have been proposed for challenging microwave device modeling, addressing both microwave active and passive component modelings. Specifically, a novel space mapping technique using decomposed mappings for GaN HEMT modeling with trapping effects has been proposed, allowing fast and systematic model development. To address the discontinuity issue and the associated non-smoothness issue in the existing pole-residue-based neuro-TF approach, a novel hybrid-based neuro-TF technique for parametric modeling of EM microwave passive components has been proposed. To address the high-sensitivity issue in the existing rational-based neuro-TF approach, a novel decomposition technique on the rational-based neuro-TF approach for parametric modeling of EM microwave passive components has been proposed.

This thesis has first presented a novel space mapping technique using decomposed mappings for GaN HEMT modeling with trapping effects. To address the dif-

difficulty of using an external mapping to map an existing device model for very different behaviors of GaN HEMTs in the existing space mapping modeling approach, the proposed technique has developed separate mappings for different branches inside the existing device model. In this way, different behaviors such as trapping effects and frequency dispersion are mapped separately. By developing different mappings, different behaviors are addressed more efficiently and accurately. The KBNN model is proposed for characterizing drain current under DC, and it is used as part of the trapping mapping development. The proposed space mapping technique can obtain accuracy large-signal model for GaN HEMTs with trapping effects. Once developed, the large-signal model can be used in circuit simulators such as ADS for high-performance microwave circuit design, covering DC, PIV, S -parameter, HB, and load-pull simulations. The proposed space mapping technique is a fast and systematic modeling approach for GaN HEMTs. The proposed technique exploits supervised learning methods to systematically develop separate mappings to achieve desired model accuracy, speeding up the model development process over the manual process of developing equivalent circuit/empirical models. Compared to the pure ANN modeling approach, the proposed technique employs less training data, lowering the cost of data generation from measurements.

A novel hybrid-based neuro-TF parametric modeling technique for EM microwave passive components has been proposed. To address the discontinuity and non-smoothness issues in the existing pole-residue-based neuro-TF approach, the proposed neuro-TF technique has systematically combined both pole-residue and rational formats of the transfer functions in the proposed neuro-TF model. The

proposed technique has identified the discontinuous poles/residues, the smooth-continuous poles/residues, and non-smooth-continuous poles/residues. The proposed technique has reformatted the discontinuous and non-smooth-continuous poles/residues together into the coefficients of the rational-based transfer function to effectively address the discontinuity and non-smoothness issues. The proposed technique has gathered the smooth-continuous poles/residues to be remained in the pole-residue format of the pole-residue-based transfer function to maintain the capability of handling high-order problem. In this way, the proposed technique can resolve the discontinuity and non-smoothness issues and can also maintain the high-order modeling capability. Compared with the existing neuro-TF modeling approaches, the proposed technique can obtain better model accuracy in challenging applications of large geometrical variations and high order, addressing the discontinuity and non-smoothness issues. With the good model accuracy, the developed parametric model can be used to accelerate the microwave design optimization process.

To address the high-sensitivity issue in the existing rational-based neuro-TF approach, a novel decomposition technique on the rational-based neuro-TF approach for parametric modeling of EM microwave passive components has been proposed. The proposed technique has decomposed the single rational-based neuro-TF model with high order into multiple sub-rational-based neuro-TF models with much lower order and reformulated the overall neuro-TF model as the combination of the sub-neuro-TF models. The proposed technique can automatically determine the number of sub-neuro-TF models and the order of each sub-neuro-TF model during the model development. The proposed decomposition technique has decreased the sensitivi-

ties of the overall model response with respect to the coefficients of the transfer function in each sub-neuro-TF models. With the decreased sensitivities, the overall model can be trained more robustly. In this way, the accuracy of the overall neuro-TF model has been improved. Compared with the existing rational-based neuro-TF approach, the proposed modeling technique can achieve better model accuracy in challenging applications of large geometrical variations, addressing the high-sensitivity issue. With the good model accuracy, the developed parametric model can be used to accelerate the microwave design optimization process.

6.2 Future Research

Some possible future research directions based on the extension of the techniques proposed in this thesis are identified as follows

- One of the future research directions is to expand the proposed space mapping technique for GaN HEMT devices into a more general modeling approach, which does not require the presence of coarse models (or knowledge models). Support vector machine (SVM) and Kriging are popular data-driven modeling methods which do not require the presence of coarse models (or knowledge models). These methods can achieve good model accuracy if sufficient data are given during training. Compared with the large amount of data used in ANN-based method for device modeling, the SVM and Kriging methods usually require moderate amount of data for model development. Currently, the SVM and Kriging methods have not been used for GaN HEMT modeling

with trapping effects. It would be a very interesting future direction to be explored.

- Another future research direction for the proposed space mapping technique is to explore the validation of the overall model with other types of large-signal data, such as dynamic drain currents at different biases, drain current versus bias at RF excitation, and memory effects or two-tone data. The potential issue of applying the proposed technique to develop the overall model which is validated by these types of large-signal data is the accuracy problem, since the overall model is not trained by the large-signal data. To improve the overall model accuracy for these types of large-signal data, the proposed technique should involve large-signal training for the overall model. However, large-signal training for the overall model can affect the model responses under DC, PIV, and S -parameter simulations. The solution is that we develop a new type of mapping on top of the overall model. This mapping only affect the model response under large-signal simulation rather than that under DC, PIV, and S -parameter simulations. In this way, the newly mapped model can not only accurately fit the DC, PIV, and S -parameter data, but also accurately fit the large-signal data of the aforementioned types.
- Another future research direction is to expand the proposed hybrid-based neuro-TF technique to address the discontinuity and non-smoothness issues coming from order changing. Currently, the proposed hybrid-based neuro-TF technique addressed the discontinuity and non-smoothness issues from

unchanged order. Order changing is needed for parametric modeling especially when different responses cannot be effectively represented with a same order from vector fitting, as the geometrical variables change. We can unify the order of the poles and residues for each geometrical parameter sample by splitting the poles and residues of lower order into those of the highest order. After the unification, all the orders of poles and residues for each geometrical parameter sample become the same. Therefore, we can still use the proposed hybrid-based neuro-TF technique to address the discontinuity and non-smoothness issues even though the order is different for different geometrical parameter sample.

- The proposed decomposition technique on the rational-based neuro-TF approach for parametric modeling has been demonstrated by the waveguide filter examples in this thesis. Other microwave passive components such as diplexers and antennas which usually have even higher order than the waveguide filter examples are too challenging to be accurately modeled by the existing rational-based neuro-TF approach. Another future direction is to apply the proposed rational-based neuro-TF technique to other types of microwave passive components, such as antennas. The potential issue of applying the proposed technique to the diplexers and antennas is the discontinuity issue. Although the coefficient data for the original rational-based neuro-TF model do not have the discontinuity issue, however, the coefficient data for the sub-rational-based neuro-TF model may have the discontinuity issue if the data are not appropriately extracted by the proposed decomposition technique. Since

the diplexers and antennas usually have more complicated EM responses than the waveguide filters, the discontinuity issue could be more intense, compromising the proposed modeling technique. Therefore, the proposed decomposition technique should be advanced in order to avoid the discontinuity issue.

References

- [1] J. B. King and T. J. Brazil, "Nonlinear electrothermal GaN HEMT model applied to high-efficiency power amplifier design," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 1, pp. 444-454, Jan. 2013.
- [2] J. Xu, R. Jones, S. A. Harris, T. Nielsen, and D. E. Root, "Dynamic FET model – DynaFET - for GaN transistors from NVNA active source injection measurements," in *IEEE MTT-S Int. Microw. Symp.*, Tampa, FL, Jun. 2014, pp. 1-3.
- [3] R. Trew, "Wide bandgap semiconductor transistors for microwave power amplifiers," *IEEE Microw. Mag.*, vol. 1, no. 1, pp. 46-54, Mar. 2000.
- [4] U. Radhakrishna, P. Choi, D. A. Antoniadis, "Facilitation of GaN-based RF- and HV-circuit designs using MVS-GaN HEMT compact model," *IEEE Trans. Electron Devices*, vol. 66, no. 1, pp. 95-105, Jul. 2019.
- [5] T. Yasui, R. Ishikawa, and K. Honjo, "GaN HEMT DC I–V Device Model for Accurate RF Rectifier Simulation," *IEEE Microw. Wireless Compon. Lett.*, vol. 27, no. 10, pp. 930-932, Sep. 2017.

- [6] A. Huang, Z. Zheng, W. Wu, and Y. Guo, "An artificial neural network-based electrothermal model for GaN HEMTs with dynamic trapping effects consideration," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 8, pp. 2519-2528, Aug. 2016.
- [7] S. C. Binari, P. B. Klein, and T. E. Kazior, "Trapping effects in GaN and SiC microwave FETs," *Proc. IEEE*, vol. 90, no. 6, pp. 1048-1058, Jun. 2002.
- [8] K. S. Yuk, G. R. Branner, and D. J. McQuate, "A wideband multiharmonic empirical large-signal model for high-power GaN HEMTs with self-heating and charge trapping effects," *IEEE Trans. Microw. Theory Techn.*, vol. 57, no. 12, pp. 3322-3332, Dec. 2009.
- [9] O. Jardel, F. De Groote, T. Reveyrand, J. C. Jacquet, C. Charbonniaud, J. P. Teyssier, D. Floriot, and R. Quéré, "An electrothermal model for AlGaN/GaN power HEMTs including trapping effects to improve large-signal simulation results on high VSWR," *IEEE Trans. Microw. Theory Techn.*, vol. 55, no. 12, pp. 2660-2669, Dec. 2007.
- [10] F. van Raay, R. Quay, M. Seelmann-Eggebert, D. Schwantuschke, D. Peschel, M. Schlechtweg, and O. Ambacher, "New low-frequency dispersion model for AlGaN/GaN HEMTs using integral transform and state description," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 1, pp. 154-167, Jan. 2013.
- [11] A. Jarndal and G. Kompa, "Large-signal model for AlGaN/GaN HEMTs accurately predicts trapping- and self-heating-induced dispersion and intermodula-

- tion distortion," *IEEE Trans. Electron Devices*, vol. 54, no. 11, pp. 2830-2836, Nov. 2007.
- [12] A. Jarndal, A. Z. Markos, and G. Kompa, "Improved modeling of GaN HEMTs on Si substrate for design of RF power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 3, pp. 644-651, Mar. 2011.
- [13] D. E. Root, "Future device modeling trends," *IEEE Microw. Mag.*, vol. 13, no. 7, pp. 45-59, Nov./Dec. 2012.
- [14] T. Gasseling, "Compact transistor models: the roadmap to first-pass amplifier design success," *Microw. J.*, vol. 55, no. 3, pp. 74-86, Mar. 2012.
- [15] Y. Xu, C. Wang, H. Sun, W. Zhang, Y. Wu, R. Xu, X. Yu, C. Ren, Z. Wang, B. Zhang, T. Chen, and T. Gao "A scalable large-signal multiharmonic model of AlGaIn/GaN HEMTs and its application in C-band high power amplifier MMIC," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 8, pp. 2836-2846, Aug. 2017.
- [16] J. Xu, J. Horn, M. Iwamoto, and D. E. Root, "Large-signal FET model with multiple time scale dynamics from nonlinear vector network analyzer data," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Anaheim, CA, May 2010, pp. 417-420.
- [17] S. A. Albahrani, D. Mahajan, J. Hodges, Y. S. Chauhan, and S. Khandelwal, "ASM GaN: industry standard model for GaN RF and power devices—part-II: modeling of charge trapping," *IEEE Trans. Electron Devices*, vol. 66, no. 1, pp. 87-94, Jan. 2019.

- [18] J. X. Zheng, X. H. Ma, Y. Lu, B. C. Zhao, H. S. Zhang, M. Zhang, and Y. Hao, "A scalable active compensatory sub-circuit for accurate GaN HEMT large signal models," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 6, pp. 431-433, May 2016.
- [19] Z. Chen, Y. Xu, C. Wang, Z. Wen, Y. Wu, and R. Xu, "A large-signal statistical model and yield estimation of GaN HEMTs based on response surface methodology," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 9, pp. 690-692, Aug. 2016.
- [20] Z. Chen, Y. Xu, B. Zhang, T. Chen, T. Gao, and R. Xu, "A GaN HEMTs non-linear large-signal statistical model and its application in S-band power amplifier design," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 2, pp. 128-130, Jan. 2016.
- [21] I. Angelov, M. Thorsell, M. Gavel, and O. Barrera, "On the modeling of high power FET transistors," in *Proc. European Microw. Integrated Circuits Conf. (EuMIC)*, London, UK, Oct. 2016, pp. 245-248.
- [22] M. Rudolph, C. Fager, and D. E. Root, *Nonlinear Transistor Model Parameter Extraction Techniques*. Cambridge, UK: Cambridge Univ. Press, 2011.
- [23] F. Filicori, G. Vannini, A. Santarelli, A. M. Sanchez, A. Tazon, and Y. Newport, "Empirical modeling of low-frequency dispersive effects due to traps and thermal phenomena in III-V FET's," *IEEE Trans. Microw. Theory Techn.*, vol. 43, no. 12, pp. 2972-2981, Dec. 1995.

- [24] K. S. Yuk and G. R. Branner, "An empirical large-signal model for SiC MES-FETs with self-heating thermal model," *IEEE Trans. Microw. Theory Techn.*, vol. 56, no. 11, pp. 2671-2680, Nov. 2008.
- [25] 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, Feb. 2014.
- [26] S. Khandelwal, Y. S. Chauhan, T. A. Fjeldly, S. Ghosh, A. Pampori, D. Mahajan, R. Dangi, and S. A. Ahsan, "ASM GaN: Industry standard model for GaN RF and power devices—Part 1: DC, CV, and RF model," *IEEE Trans. Electron Devices*, vol. 66, no. 1, pp. 80-86, Jan. 2019.
- [27] W. Deng, J. Huang, X. Ma, and J. J. Liou, "An explicit surface potential calculation and compact current model for AlGa_N/Ga_N HEMTs," *IEEE Electron Device Lett.*, vol. 36, no. 2, pp. 108-110, Jan. 2015.
- [28] N. Karumuri, G. Dutta, N. DasGupta, and A. DasGupta, "A compact model of drain current for Ga_N HEMTs based on 2-DEG charge linearization," *IEEE Trans. Electron Devices*, vol. 63, no. 11, pp. 4226-4232, Sep. 2016.
- [29] S. Turuvekere, N. Karumuri, A. A. Rahman, A. Bhattacharya, A. DasGupta, and N. DasGupta, "Gate leakage mechanisms in AlGa_N/Ga_N and AlIn_N/Ga_N HEMTs: comparison and modeling," *IEEE Trans. Electron Devices*, vol. 60, no. 10, pp. 3157-3165, Jul. 2013.

- [30] A. Debnath, N. DasGupta, and A. DasGupta, "Charge-Based Compact Model of Gate Leakage Current for AlInN/GaN and AlGaIn/GaN HEMTs," *IEEE Trans. Electron Devices*, vol. 67, no. 3, pp. 834-840, Feb. 2020.
- [31] S. Khandelwal, N. Goyal, and T. A. Fjeldly, "A physics-based analytical model for 2DEG charge density in AlGaIn/GaN HEMT devices," *IEEE Trans. Electron Devices*, vol. 58, no. 10, pp. 3622-3625, Aug. 2011.
- [32] R. Vetury, N. Q. Zhang, S. Keller, and U. K. Mishra, "The impact of surface states on the DC and RF characteristics of AlGaIn/GaN HFETs," *IEEE Trans. Electron Devices*, vol. 48, no. 3, pp. 560-566, Mar. 2001.
- [33] O. Mitrofanov and M. Manfra, "Mechanisms of gate lag in GaN/AlGaIn/GaN high electron mobility transistors," *Superlattices Microstruct.*, vol. 34, nos. 1-2, pp. 33-53, Jul./Aug. 2003.
- [34] F. Mkadem and S. Boumaiza, "Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 4, pp. 913-923, Apr. 2011.
- [35] Z. Zhao, W. Na, V.-M.-R. Gongal-Reddy, and Q. J. Zhang, "Multi-band behavioral modeling of power amplifier using carrier frequency-dependent time delay neural network model," *IEEE MTT-S International Conference on Numerical Electromagnetic Multiphysics Modeling and Optimization (NEMO)*, pp. 82-84, May 2017.

- [36] P. Aaen, J. A. Plá, and J. Wood, *Modeling and Characterization of RF and Microwave Power FETs*. Cambridge, UK: Cambridge Univ. Press, 2007.
- [37] W. Liu, W. Na, L. Zhu, J. Ma, and Q. J. Zhang, "A Wiener-type dynamic neural network approach to the modeling of nonlinear microwave devices," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 6, pp. 2043- 2062, Jun. 2017.
- [38] L. Zhu, Q. J. Zhang, K. Liu, Y. Ma, B. Peng, and S. Yan, "A novel dynamic neuro-space mapping approach for nonlinear microwave device modeling," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 2, pp. 131-133, Feb. 2016.
- [39] Y. Long, Z. Zhong, and Y. Guo, "A novel 4-D artificial-neural-network-based hybrid large-signal model of GaAs pHEMTs," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 6, pp. 1752-1762, Jun. 2016.
- [40] J. W. Bandler, Q. S. Cheng, S. A. Dakroury, A. S. Mohamed, M. H. Bakr, K. Madsen, and J. Søndergaardet, "Space mapping: The state of the art," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 337-361, Jan. 2004.
- [41] S. Koziel, Q. S. Cheng, and J. W. Bandler, "Space mapping," *IEEE Microw. Mag.*, vol. 9, no. 6, pp. 105-122, Dec. 2008.
- [42] M. H. Bakr, J. W. Bandler, M. A. Ismail, J. E. Rayas-Sánchez, and Q. J. Zhang, "Neural space mapping optimization for EM-based design," *IEEE Trans. Microw. Theory Techn.*, vol. 48, no. 12, pp. 2307-2315, Dec. 2000.
- [43] L. Zhang, J. Xu, M. C. E. Yagoub, R. Ding, and Q. J. Zhang, "Efficient analytical formulation and sensitivity analysis of neuro-space mapping for nonlinear

- microwave device modeling," *IEEE Trans. Microw. Theory Tech.*, vol. 53, no. 9, pp. 2752-2767, Sep. 2005.
- [44] I. S. Stievano, I. A. Maio, and F. G. Canavero, "Parametric macromodels of digital I/O ports," *IEEE Trans. Adv. Packag.*, vol. 25, no. 5, pp. 255-264, May 2002.
- [45] Y. Cao, G. Wang, and Q. J. Zhang, "A new training approach for parametric modeling of microwave passive components using combined neural networks and transfer functions," *IEEE Trans. Microw. Theory Techn.*, vol. 57, no. 11, pp. 2727-2742, Nov. 2009.
- [46] S. A. Sadrossadat, Y. Cao, and Q. J. Zhang, "Parametric Modeling of Microwave Passive Components Using Sensitivity-Analysis-Based Adjoint Neural-Network Technique," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 5, pp. 1733-1747, May 2013.
- [47] W. C. Na and Q. J. Zhang, "Automated parametric modeling of microwave components using combined neural network and interpolation techniques," in *IEEE MTT-S Int. Dig.*, Seattle, WA, USA, Jun. 2013.
- [48] V.-M.-R. Gongal-Reddy, F. Feng, and Q.J. Zhang, "Parametric modeling of millimeter-wave passive components using combined neural networks and transfer functions," in *Global Symposium On Millimeter Waves (GSMM)*, Montreal, QC, Canada, May 2015, pp. 1-3.

- [49] F. Feng, C. Zhang, J. Ma, and Q. J. Zhang, "Parametric modeling of EM behavior of microwave components using combined neural networks and pole-residue-based transfer functions," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 1, pp. 60-77, Jan. 2016.
- [50] W. Na, F. Feng, C. Zhang, and Q. J. Zhang, "A unified automated parametric modeling algorithm using knowledge-based neural network and l_1 optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 3, pp. 729-745, Mar. 2017.
- [51] F. Feng, V.-M.-R. Gongal-Reddy, C. Zhang, J. Ma, and Q.-J. Zhang, "Parametric modeling of microwave components using adjoint neural networks and pole-residue transfer functions with EM sensitivity analysis," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 6, pp. 1955-1975, Jun. 2017.
- [52] W. Zhang, F. Feng, V. M. R. Gongal-Reddy, J. Zhang, S. Yan, J. Ma, and Q. J. Zhang, "Space mapping approach to electromagnetic centric multiphysics parametric modeling of microwave components," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 7, pp. 3169-3185, Jul. 2018.
- [53] J. Zhang, F. Feng, W. Zhang, J. Jin, J. Ma, and Q. J. Zhang, "A novel training approach for parametric modeling of microwave passive components using Padé via Lanczos and EM sensitivities," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 6, pp. 2215-2233, Jun. 2020.
- [54] X. Ding, V. K. Devabhaktuni, B. Chattaraj, M. C. E. Yagoub, M. Deo, J. Xu, and Q. J. Zhang, "Neural-network approaches to electromagnetic based modeling

- of passive components and their applications to high frequency and high-speed nonlinear circuit optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 436-449, Jan. 2004.
- [55] J. E. Rayas-Sánchez, "EM-based optimization of microwave circuits using artificial neural networks: The state-of-the-art," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 420-435, Jan. 2004.
- [56] Q. J. Zhang, K. C. Gupta, and V. K. Devabhaktuni, "Artificial neural networks for RF and microwave design - from theory to practice," *IEEE Trans. Microw. Theory Techn.*, vol. 51, no. 4, pp. 1339-1350, Apr. 2003.
- [57] G. Gosal, E. Almajali, D. McNamara and M. Yagoub, "Transmitarray antenna design using forward and inverse neural network modeling," *IEEE Ant. and Wireless Prop. Lett.*, vol. 15, no. 1, pp. 1483-1486, Dec. 2015.
- [58] P. M. Watson and K. C. Gupta, "EM-ANN models for microstrip vias and interconnects in multilayer circuits," *IEEE Trans. Microw. Theory Techn.*, vol. 44, no. 12, pp. 2495-2503, Dec. 1996.
- [59] Q. J. Zhang and M. S. Nakhla, "Signal integrity analysis and optimization of VLSI interconnects using neural network models," *IEEE Int. Circuits Syst. Symp.*, London, England, May 1994, pp. 459-462.
- [60] A. Veluswami, M. S. Nakhla, and Q. J. Zhang, "The application of neural networks to EM-based simulation and optimization of interconnects in high-

- speed VLSI circuits," *IEEE Trans. Microw. Theory Techn.*, vol. 45, no. 5, pp. 712-723, May 1997.
- [61] V. K. Devabhaktuni, C. Xi, F. Wang, and Q. J. Zhang, "Robust training of microwave neural models," *Int. J. RF Microwave CAE*, vol. 12, no. 1, pp. 109-124, Jan. 2002.
- [62] P. M. Watson and K. C. Gupta, "Design and optimization of CPW circuits using EM-ANN models for CPW components," *IEEE Trans. Microw. Theory Techn.*, vol. 45, no. 12, pp. 2515-2523, Dec. 1997.
- [63] S. Liao, H. Kabir, Y. Cao, J. Xu, Q. J. Zhang, and J. Ma, "Neural-network modeling for 3-D substructures based on spatial EM-field coupling in finite-element method," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 1, pp. 21-38, Jan. 2011.
- [64] G. Fedi, S. Manetti, G. Pelosi, and S. Selleri, "Design of cylindrical posts in rectangular waveguide by neural network approach," *IEEE AP-S Int. Symp. Digest*, Salt Lake City, UT, Jul. 2000, pp. 1054-1057.
- [65] J. W. Bandler, M. A. Ismail, J. E. Rayas-Sanchez, and Q. J. Zhang, "Neuromodeling of microwave circuits exploiting space-mapping technology," *IEEE Trans. Microw. Theory Techn.*, vol. 47, no. 12, pp. 2417-2427, Dec. 1999.
- [66] G. L. Creech, B. J. Paul, C. D. Lesniak, T. J. Jenkins, and M. C. Calcaterra, "Artificial neural networks for fast and accurate EM-CAD of microwave circuits," *IEEE Trans. Microwave Theory Techn.*, vol. 45, no. 5, pp. 794-802, May 1997.

- [67] V. K. Devabhaktruni, M. Yagoub, and Q. J. Zhang, "A robust algorithm for automatic development of neural-network models for microwave applications," *IEEE Trans. Microw. Theory Techn.*, vol. 49, no. 12, pp. 2282-2291, Dec. 2001.
- [68] W. Na and Q. J. Zhang, "Automated knowledge-based neural network modeling for microwave applications," *IEEE Microw. Wireless Compon. Lett.*, vol. 24, no. 7, pp. 499-501, Jun. 2014.
- [69] H. Yu, H. Chalamalasetty, and M. Swaminathan, "Modeling of voltage-controlled oscillators including I/O behavior using augmented neural networks," *IEEE Access*, vol. 7, pp. 38973-38982, Mar. 2019.
- [70] B. O'Brien, J. Dooley, and T. J. Brazil, "RF power amplifier behavioral modeling using a globally recurrent neural network," *IEEE MTT-S Int. Microw. Symp. Dig.*, San Francisco, CA, Jun. 2006, pp. 1089-1092.
- [71] Y. H. Fang, M. C. E. Yagoub, F. Wang, and Q. J. Zhang, "A new macro-modeling approach for nonlinear microwave circuits based on recurrent neural networks," *IEEE Trans. Microw. Theory Tech.*, vol. 48, no. 12, pp. 2335-2344, Dec. 2000.
- [72] S. Yan, C. Zhang, and Q. J. Zhang, "Recurrent neural network technique for behavioral modeling of power amplifier with memory effects," *Int. J. RF Microwave CAE*, vol. 25, no. 4, pp. 289-298, Oct. 2014.

- [73] T. Liu, S. Boumaiza, and F. M. Ghannouchi, "Dynamic behavioral modeling of 3G power amplifiers using real-valued time-delay neural networks," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 3, pp. 1025-1033, Mar. 2004.
- [74] Y. Cao, J. J. Xu, V. K. Devabhaktuni, R. T. Ding, and Q. J. Zhang, "An adjoint dynamic neural network technique for exact sensitivities in nonlinear transient modeling and high-speed interconnect design," *IEEE MTT-S Int. Microw. Symp. Dig.*, PA, Philadelphia, Jun. 2003, pp. 165-168.
- [75] J. Xu, M. C. E. Yagoub, R. Ding, and Q. J. Zhang, "Neural based dynamic modeling of nonlinear microwave circuits," *IEEE Trans. Microw. Theory Techn.*, vol. 50, no. 12, pp. 2769-2780, Dec. 2002.
- [76] Y. Cao, R. T. Ding, and Q. J. Zhang, "A new nonlinear transient modeling technique for high-speed integrated circuit applications based on state-space dynamic neural network," *IEEE MTT-S Int. Microw. Symp. Dig.*, Fort Worth, TX, Jun. 2004, pp. 1553-1556.
- [77] Y. Cao, R. T. Ding, and Q. J. Zhang, "State-space dynamic neural network technique for high-speed IC applications: Modeling and stability analysis," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 6, pp. 2398-2409, Jun. 2006.
- [78] Q. J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*. Norwood, MA, USA: Artech House, 2000.
- [79] V. K. Devabhaktuni, B. Chattaraj, M. C. E. Yagoub, and Q. J. Zhang, "Advanced microwave modeling framework exploiting automatic model generation,

- knowledge neural networks, and space mapping," *IEEE Trans. Microw. Theory Techn.*, vol. 51, no. 7, pp. 1822-1833, Jul. 2003.
- [80] J. W. Bandler, M. A. Ismail, J. E. Rayas-Sánchez, and Q. J. Zhang, "Neuromodeling of microwave circuits exploiting space-mapping technology," *IEEE Trans. Microw. Theory Techn.*, vol. 47, no. 12, pp. 2417-2427, Dec. 1999.
- [81] J. E. Rayas-Sánchez and V. Gutiérrez-Ayala, "EM-based Monte Carlo analysis and yield prediction of microwave circuits using linear-input neural-output space mapping," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 12, pp. 4528-4537, Dec. 2006.
- [82] Y. Cao and G. Wang, "A wideband and scalable model of spiral inductors using space-mapping neural network," *IEEE Trans. Microw. Theory Techn.*, vol. 55, no. 12, pp. 2473-2480, Dec. 2007.
- [83] H. Kabir, L. Zhang, M. Yu, P. H. Aaen, J. Wood, and Q. J. Zhang, "Smart modeling of microwave device," *IEEE Microw. Mag.*, vol. 11, no. 3, pp. 105-118, May 2010.
- [84] S. Koziel, J. W. Bandler, and K. Madsen, "A space mapping framework for engineering optimization: theory and implementation," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 10, pp. 3721-3730, Oct. 2006.
- [85] J. W. Bandler, Q. S. Cheng, N. K. Nikolova, and M. A. Ismail, "Implicit space mapping optimization exploiting preassigned parameters," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 378-385, Jan. 2004.

- [86] J. W. Bandler, M. A. Ismail, and J. E. Rayas-Sánchez, "Expanded space-mapping EM-based design framework exploiting preassigned parameters," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 49, no. 12, pp. 1833-1838, Dec. 2002.
- [87] S. Koziel, J. W. Bandler, and Q. S. Cheng, "Constrained parameter extraction for microwave design optimisation using implicit space mapping," *IET Microw. Antennas Propag.*, vol. 5, no. 10, pp. 1156-1163, Jul. 2011.
- [88] S. Koziel, J. W. Bandler, and Q. S. Cheng, "Constrained parameter extraction for microwave design optimisation using implicit space mapping," *IET Microw. Antennas Propag.*, vol. 5, no. 10, pp. 1156-1163, Jul. 2011.
- [89] S. Koziel, J. W. Bandler, and K. Madsen, "Space mapping with adaptive response correction for microwave design optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 57, no. 2, pp. 478-486, Feb. 2009.
- [90] J. W. Bandler, D. M. Hailu, K. Madsen, and F. Pedersen, "A space mapping interpolating surrogate algorithm for highly optimized EM-based design of microwave devices," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 11, pp. 2593-2600, Nov. 2004.
- [91] R. B. Ayed, J. Gong, S. Brisset, F. Gillon, and P. Brochet, "Three-level output space mapping strategy for electromagnetic design optimization," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 671-674, Feb. 2012.

- [92] V. Gutierrez-Ayala and J. E. Rayas-Sánchez, "Neural input space mapping optimization based on nonlinear two-layer perceptrons with optimized nonlinearity," *Int. J. RF Microw. Comput.-Aided Eng.*, vol. 20, no. 5, pp. 512-526, Sep. 2010.
- [93] D. Gorissen, L. Zhang, Q. J. Zhang, and T. Dhaene, "Evolutionary neuro-space mapping technique for modeling of nonlinear microwave devices," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 2, pp. 213-229, Feb. 2011.
- [94] F. Wang and Q. J. Zhang, "Knowledge-based neural models for microwave design," *IEEE Trans. Microw. Theory Techn.*, vol. 45, no. 12, pp. 2333-2343, Dec. 1997.
- [95] V. Rizzoli, A. Neri, D. Masotti, and A. Lipparini, "A new family of neural network-based bidirectional and dispersive behavioral models for nonlinear RF/microwave subsystems," *Int. J. RF Microw. Computer-Aided Eng.*, vol. 12, no. 1, pp. 51-70, Jan. 2002.
- [96] J. A. Garcia, et al., "Modeling MESFET's and HEMT's intermodulation distortion behavior using a generalized radial basis function network," *Int. Journal of RF and Microwave CAE, Special Issue on Applications of ANN to RF and Microwave Design*, Vol. 9, pp. 261-276, 1999.
- [97] M. Isaksson, D. Wisell, and D. Ronnow, "Wide-band dynamic modeling of power amplifiers using radial-basis function neural networks," *IEEE Trans. Microw. Theory Techn.*, vol. 53, no. 11, pp. 3422-3428, Nov. 2005.

- [98] J. Jin, C. Zhang, F. Feng, W. Na, J. Ma, and Q.-J. Zhang, "Deep neural network technique for high-dimensional microwave modeling and applications to parameter extraction of microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 67, no. 10, pp. 4140-4155, Oct. 2019.
- [99] Q. H. Zhang, "Using wavelet network in nonparametric estimation," *IEEE Trans. Neural Networks*, vol. 8, no. 2, pp. 227-236, Mar. 1997.
- [100] Q. H. Zhang and A. Benvensite, "Wavelet networks," *IEEE Trans. Neural Networks*, vol. 3, no. 6, pp. 889-898, Nov. 1992.
- [101] Y. C. Pati and P. S. Krishnaprasad, "Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations," *IEEE Trans. Neural Networks*, vol. 4, no. 1, pp. 73-85, Jan. 1993.
- [102] M. Sans, J. Selga, P. Vélez, A. Rodríguez, J. Bonache, V. E. Boria, and F. Martin, "Automated design of common-mode suppressed balanced wideband bandpass filters by means of aggressive space mapping," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 12, pp. 3896-3908, Dec. 2015.
- [103] S. Koziel, Q. S. Chen, and J. W. Bandler, "Fast EM modeling exploiting shape-preserving response prediction and space mapping," *IEEE Trans. Microw. Theory Techn.*, vol. 62, no. 3, pp. 399-407, Mar. 2014.
- [104] S. Koziel, J. W. Bandler, and Q. S. Cheng, "Tuning space mapping design framework exploiting reduced electromagnetic models," *IET Microw. Antennas Propag.*, vol. 5, no. 10, pp. 1219-1226, Jul. 2011.

- [105] J. Meng, S. Koziel, J. W. Bandler, M. H. Bakr, and Q. S. Cheng, "Tuning space mapping: A novel technique for engineering design optimization," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Atlanta, Georgia, Jun. 2008, pp. 991-994.
- [106] L. Zhang, P. H. Aaen, and J. Wood, "Portable space mapping for efficient statistical modeling of passive components," *IEEE Trans. Microw. Theory Techn.*, vol. 60, no. 3, pp. 441-450, Mar. 2012.
- [107] F. Feng, C. Zhang, V. M. R. Gongal-Reddy, and Q. J. Zhang, "Parallel space-mapping approach to EM optimization," *IEEE Trans. Microw. Theory Techn.*, vol. 62, no. 5, pp. 1135-1148, May 2014.
- [108] F. Feng, J. Zhang, W. Zhang, Z. Zhao, J. Jin, and Q. J. Zhang, "Coarse and fine mesh space mapping for EM optimization incorporating mesh deformation," *IEEE Microw. Wireless Compon. Lett.*, vol. 29, no. 8, pp. 510-512, Aug. 2019.
- [109] C. Zhang, J. Jin, W. Na, Q.-J. Zhang, and M. Yu, "Multivalued neural network inverse modeling and applications to microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 8, pp. 3781-3797, Aug. 2018.
- [110] V. Rizzoli, A. Costanzo, D. Masotti, A. Lipparini, and F. Mastri, "Computer-aided optimization of nonlinear microwave circuits with the aid of electromagnetic simulation," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 1, pp. 362-377, Jan. 2004.

- [111] M. B. Steer, J. W. Bandler, and C.M. Snowden, "Computer-aided design of RF and microwave circuits and systems," *IEEE Trans. Microw. Theory Techn.*, vol. 50, no. 3, pp. 996-1005, Mar. 2002.
- [112] P. Burrascano, S. Fiori, and M. Mongiardo, "A review of artificial neural networks applications in microwave computer-aided design," *Int. J. RF Microw. Comput.-Aided Eng.*, vol. 9, no. 3, pp. 158-174, May 1999.
- [113] S. Zhang, J. Xu, Q. J. Zhang, and D. E. Root, "Parallel matrix neural network training on cluster systems for dynamic FET modeling from large datasets," in *IEEE MTT-S Int. Microw. Symp. Dig.*, San Francisco, CA, USA, May 2016, pp. 1-3.
- [114] S. Koziel, S. Ogurtsov, I. Couckuyt, and T. Dhaene, "Variable-fidelity electromagnetic simulations and co-Kriging for accurate modeling of antennas," *IEEE Trans. Antennas Propag.*, vol. 61, no. 3, pp. 1301 - 1308, Mar. 2013.
- [115] P. Barmuta, F. Ferranti, K. Lukasik, G. P. Gibiino and D. Schreurs, "Hybrid nonlinear model for microwave active devices using kriging," *2017 Integrated Nonlinear Microwave and Millimetre-wave Circuits Workshop (INMMiC)*, Graz, Austria, April 2017, pp. 1-3.
- [116] J. P. Jacobs, S. Koziel, and S. Ogurtsov, "Computationally efficient multi-fidelity bayesian support vector regression modeling of planar antenna input characteristics," *IEEE Trans. Antennas Propag.*, vol. 61, no. 2, pp. 980-984, Feb. 2013.

- [117] J. Wood, P. H. Aaen, D. Bridges, D. Lamey, M. Guyonnet, D. S. Chan, and N. Monsauret, "A nonlinear electro-thermal scalable model for high-power RF LDMOS transistors," *IEEE Trans. Microw. Theory Techn.*, vol. 57, no. 2, pp. 282-292, Feb. 2009.
- [118] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052-1061, Jul. 1999.
- [119] T. Dhaene and D. Deschrijver, "Stable parametric macromodeling using a recursive implementation of the vector fitting algorithm", *IEEE Microw. Wireless Compon. Lett.*, vol. 19, no. 2, pp. 59-61, Feb. 2009.
- [120] C. Zhang, F. Feng, V.-M.-R. Gongal-Reddy, Q. J. Zhang, and J. W. Bandler, "Cognition-driven formulation of space mapping for equal-ripple optimization of microwave filters," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 7, pp. 2154-2165, Jul. 2015.
- [121] S. R. Schmidt and R. G. Launsby, *Understanding Industrial Designed Experiments*. Colorado Springs, CO: Air Force Academy, 1992.