Sequential Sampling Strategies for Automated Parametric Macromodeling of High-Frequency Electrical Systems

Strategieën van sequentiële bemonstering voor geautomatiseerde geparametriseerde macromodellering van hoogfrequente elektrische systemen

Krishnan Chemmangat Manakkal Cheriya

Promotoren: prof. dr. ir. T. Dhaene, prof. dr. ir. L. Knockaert
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Elektrotechniek

Vakgroep Informatietechnologie
Voorzitter: prof. dr. ir. D. De Zutter
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2013 - 2014

ISBN 978-90-8578-685-6
NUR 959
Wettelijk depot: D/2014/10.500/31

Acknowledgements

The four years of my doctoral research was an overwhelming experience, enriched with a fascinating international exposure which opened up a new world around me. It enabled me immensely to evaluate myself and the little bubble in which I am contained, from a completely new vantage point.

First and foremost, I would like to thank my promoters Prof. Tom Dhaene and Prof. Luc Knockaert. Without their excellent guidance and support, it would not have been possible for me to have a successful research work. I have my immense gratitude to both of them for creating a homely atmosphere and making my stay here an enriching experience, not only at a professional level but also at a personal level with both formal and informal conversations. I sincerely believe that their continuous support will always be there for me throughout my future career.

I am deeply grateful to Dr. Francesco Ferranti whose invaluable support helped me immensely in performing a good research work. I thank him for all the wonderful and thought-provoking research discussions we had over these four years, and the insightful comments and suggestions he gave, which I believe had tremendously improved the quality of my research.

I would like to offer my special thanks to Prof. Yves Rolain for the interesting discussions we had during our joint research work. I also thank him for being a part of my PhD jury, going through my thesis in great detail and giving excellent comments and suggestions to improve its quality.

I also thank all other jury members, Dr. Nobby Stevens, Prof. Luc Dupré, Prof. Eric Laermans and Dr. Dirk Deschrijver for thoroughly evaluating my work and for their valuable comments on my PhD thesis.

I thank my colleagues Domenico Spina, Elizabeth Rita Samuel, Matthias Caenepeel and Dr. Dirk Deschrijver for their excellent contribution to my doctoral research. I am in debt to all the members of the SUMO group for their help and support during my stay in Gent.

I express my sincere gratitude to Ghent University and to all the members of IBCN research group for providing me the platform to successfully carry out my research work. I would also like to thank the Research Foundation Flanders (FWO) and the interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office for releasing funds to support my PhD research.

I also wish to thank all my office members of 3.19 for providing me a pleas-

ing office atmosphere and for all their cooperations, which had a very positive influence on my PhD life.

My heartfelt thanks to all my dearly friends; Selva for being a younger brother and a part of my extended family here, Dinesh for his excellent brotherly guidance and support, Farhan for aiding me in smoothly adapting to IBCN and Belgium life, Rajesh for being a good friend motivating me from thousands of kilometers away, Prashant for his innocent and unstoppable smiles, Elizabeth for the casual talks and her excellent and easy cooking tips, Domenico for his cheerful leg-pulling remarks and Sridhar, Vijaya, Lalithya, Sreegovind, Srikanth, Srinath, Mithra, Sreeparvathy, Swarna, Prasanna, Anesh, Saravana, Ramanan, Sriram, Vidhya, Manu ettan, Achukuttan, Camille, Gaurav, Camille's mom and dad, Neelam Farhan, Sonia Heras, Arun, Sreelakshmi, Adil, Sandeep, Vishvas, Divya kurup, Jay, Sidharth, Abhishek, Sachin, Happy, and Divya for all the get together and fun during my sojourn here.

This acknowledgement is incomplete without mentioning my dearest and beloved Sreeda for being a part of me in all the happiness and sorrow. I dedicate this thesis to my family: my wife, my parents, my in-laws and my wonderful brother Sasank. They had always been there for me with their constant care, love, support, guidance, and trust, without which this PhD would have been nothing more than a beautiful early morning dream.

Gent, May 2014
Krishnan Chemmangat

Table of Contents

List of Figures

List of Tables

List of Acronyms

A

ADS	Advanced Design System
AFS	Adaptive Frequency Sampling

C

CAD	Computer-Aided Design
CS	Cubic Spline

D

DIRECT	DIviding RECTangle
DoE	Design of Experiments

E

EDA	Electronic Design Automation
EM	Electromagnetic

F

FAST	Fourier Amplitude Sensitivity Testing
FRF	Frequency Response Function

L

LLE	Locally-Linear Embedding
LMI	Linear Matrix Inequality
LTI	Linear Time-Invariant

M

MAE	Mean Absolute Error

P

PCA	Principal Component Analysis
PCHIP	Piecewise Cubic Hermite Interpolating Polynomial

S

SBO	Surrogate-Based Optimization
SM	Space Mapping
SPICE	Simulation Program with Integrated Circuit Emphasis

T

TEM	Transverse Electromagnetic
TL	Transmission Line
TLS	Total Least Squares

V

VF	Vector Fitting

Samenvatting
– Summary in Dutch –

Omwille van de recente vooruitgang op technologisch vlak is het analyzeren van vele technische systemen zeer complex geworden. De conventionele benadering waarbij men prototypes bouwt en metingen uitvoert leidt tot een traag en duur ontwerpproces. Dit spoort de onderzoekers en technische specialisten in Computer Aided Design (CAD) aan om elektromagnetische simulatoren te ontwikkelen die het gedrag van systemen nauwkeurig modelleren. Hierdoor kan het bouwen van dure prototypes voorkomen worden, waardoor men tijd en kosten bespaart. Dergelijke simulatiegedreven benaderingen zijn tegenwoordig heel populair en veel probleemspecifieke CAD-tools zijn al ontwikkeld.

Hoewel CAD tools zeer nauwkeurig zijn, is het bouwen van de aangemaakte modellen vaak duur in termen van de computer rekentijd. Veelvuldige simulaties zijn vaak vereist, aangezien deze CAD modellen gebruikt worden tijdens het ontwerpproces van complexe technische systemen. De resulterende simulaties kunnen gebruikt worden voor verschillende ontwerpactiviteiten zoals het verkennen van de parameterruimte, gevoeligheidsanalyse, variabiliteit en robuustheidsstudies, enz. Simulaties met CAD-technieken zijn echter beperkt toepasbaar in de praktijk omdat ze rekenintensief zijn en het totale ontwerpproces aanzienlijk vertragen. Als alternatief kan men ook gedrags- of surrogaatmodellen bouwen op basis van een beperkt aantal goedgekozen CAD simulaties, met als doel om deze complexe modellen zo nauwkeurig mogelijk na te bootsen. Deze surrogaatmodellen moeten zeer efficiënt en goedkoop te evalueren zijn. Daardoor vereisen ze veel minder rekenkracht dan het uitvoeren van directe simulaties met een CAD tool.

Surrogaatmodellen voor dure scalaire functies worden typisch berekend met behulp van technieken zoals Artificiële Neurale Netwerken, Support Vector Machines, veelterminterpolatie, Kriging, enz. In dit proefschrift ligt de focus op het geautomatiseerd berekenen van surrogaatmodellen voor rekenintensieve CAD simulatoren. De algemene klasse van lineaire tijds-invariante (LTI) systemen wordt gebruikt. De specifieke focus voor de toepassingen ligt op hoogfrequente elektromagnetische (EM) systemen. Surrogaatmodellen van hoogfrequente EM systemen moeten de frequentierespons efficiënt kunnen parameteriseren in functie van de ontwerpparameters. Echter, het parametriseren van de meerpoort frequentierespons van EM-systemen is veel uitdagender dan de parametrisatie van scalaire functies en moet anders worden behandeld. In de werkwijze die men hier hanteert, worden rationale modellen berekend voor goedgeselecteerde parameterwaarden

met behulp van een systeemidentificatieaanpak. De parameters van deze modellen zijn naast de frequentie ook de ontwerpparameters. Het uiteindelijk model van een willekeurige combinatie van parameters wordt bekomen door verschillende interpolatieschema's te gebruiken. Aldus wordt de frequentie als een "bijzondere"parameter beschouwd die geen deel uitmaakt van de ontwerpparameterruimte. Deze benadering van modelleren heeft twee belangrijke voordelen: 1) door de frequentierespons van LTI systemen te modelleren met rationale functies wordt het gemakkelijker om fysische eigenschappen zoals stabiliteit en causaliteit te handhaven, 2) door het scheiden van de frequentie en de parameters in de ontwerpruimte wordt de complexiteit van het parametriseren van een EM systeem significant verlaagd. Dergelijke surrogaat- of gedragsmodellen voor een meerpoort frequentierespons worden in de literatuur ook geparametriseerde macromodellen of schaalbare macromodellen genoemd. Omwille van uniformiteit wordt de term *geparametriseerd macromodel* verder gebruikt doorheen dit proefschrift. Dit proefschrift behandelt verschillende aspecten van een efficiënte en geautomatiseerde modellering zoals het ramen van de modelcomplexiteit voor de frequentieafhankelijke rationale functies, het kiezen van een geschikte strategie voor de parametrisatie en een goede selectie van foutmaten om de nauwkeurigheid van de berekende geparametriseerde macromodellen te valideren.

Daarnaast worden ook een aantal verbeteringen van bestaande geparameteriseerde macromodelingstechnieken besproken in dit proefschrift. Bovenop deze automatisering kunnen de sequentiële bemonsteringsmethoden samen met de geparameteriseerde macromodelingstechnieken parallel worden uitgevoerd en ook uitgebreid worden naar andere technische gebieden die gerelateerd zijn aan het gebruik van LTI systemen.

Dit proefschrift is als volgt opgebouwd:

Hoofdstuk 1 geeft een algemene introductie van het doctoraatswerk en een bespreking van de context en de motivatie voor dit werk.

Een overzicht van de huidige state-of-the-art van geparametriseerde macromodelleringsmethoden wordt besproken in hoofdstuk 2. Ook een bespreking van hun relatieve voordelen, hun beperkingen en hun geschiktheid om te worden uitgebreid met efficiënte sequentiële bemonsteringsschema's wordt bestudeerd.

In hoofdstuk 3 wordt een state-of-the-art techniek voor geparameteriseerde macromodellering, gebaseerd op interpolatie van de state-space matrices, veralgemeend om ook geparametriseerde gevoeligheidsanalyses te kunnen uitvoeren met modellen. Deze modellen kunnen analytisch de gevoeligheid van het systeem berekenen voor alle waarden van de ontwerpparameters zodat ze gebruikt kunnen worden in gradiëntgebaseerde ontwerpoptimalisaties en gevoeligheidsanalyses.

Hoofdstuk 4 behandelt het automatiseren van de extractie van geparameteriseerde macromodellen. Dit kan beschouwd worden als de eerste stap van het geparametriseerde macromodelingsproces. De bemonsteringsschema's ontwikkeld in hoofdstuk 4 zijn geschikt voor lokaal geparametriseerde macromodellen. Het basisidee bestaat erin om de bemonstering van de ontwerpruimte lokaal te verfijnen. In hoofdstuk 4 worden twee verschillende grid-gebaseerde bemonsteringstechnieken ontwikkeld om de ontwerpruimte adaptief te bemonsteren.

Hoofdstuk 5 bevat de toepassing van het geautomatiseerde geparametriseerde macromodelingsproces met sequentiële bemonstering bij het ontwerp van een real-life voorbeeld (microgolffilter) te versnellen. In dit hoofdstuk wordt het volledige ontwerp van de filter besproken, gebruik makend van geparametriseerde macromodellen. Het hoofdstuk toont ook aan hoeveel ontwerpcycli bespaard worden door een geparametriseerd macromodel te gebruiken in plaats van directe EM simulaties uit te voeren tijdens een extensieve ontwerpoptimalisatie en een Monte-Carlo gebaseerde variabiliteitsanalyse. De resultaten van de optimalisatie worden uiteindelijk gecontroleerd met behulp van metingen die uitgevoerd worden op de verwezenlijking van verschillende optimale filterconfiguraties. Bovendien stelt hoofdstuk 5 ook een verbeterde geparameteriseerde macromodelingsstrategie voor. Hierbij worden verschillende frequentieschalingcoëfficiënten gebruikt om het berekenen van de overdrachtsfunctie van geparametriseerde frequentieresponsies te verbeteren. In hoofdstuk 6 wordt de sequentiële manier van bemonsteren uit hoofdstuk 4 uitgebreid naar verspreide roosters met behulp van een goed geconditioneerde simplex verfijning, genaamd *pad-simplex verdeling*. Hoofdstuk 7 geeft een slotbeschouwing over het proefschrift en enkele overpeinzingen van de belangrijkste bijdragen, alsook de mogelijke uitbreidingen en toekomstige onderzoekslijnen. Tenslotte wordt in Appendix A een algoritme voor de sequentiële bemonsteringsschema's in detail uitgewerkt als hulp voor de lezer.

Summary

With recent advancements in technology, the analysis of many engineering problems has become very complex. The conventional approach of generating prototypes and performing measurements in the overall design process became slow and very expensive. This motivated the researchers and the engineering specialists to develop Computer Aided Design (CAD) tools such as electromagnetic simulators, which model the behavior of systems accurately such that expensive prototyping can be avoided while saving time and cost. Such simulation-driven approaches are very popular nowadays and many problem specific CAD tools have been developed.

Even though, such CAD tools are very accurate, the models generated using these tools are often very expensive in terms of the computational resources. Since, ultimately these CAD tools are used in the design process of complex engineering systems, multiple simulations is necessary. The resulting simulations can then be used in several design activities such as design space exploration, sensitivity analysis, variability and robustness studies, etc. However, as the CAD tool simulations are computationally very expensive, the overall design process might be considerably slow, limiting their applicability in practice. An alternative is to build behavioral or surrogate models using a limited number of judiciously chosen CAD simulations to mimic these complex models as accurate as possible. These surrogate models must be very efficient and cheap to evaluate and hence can be used in the design process with much reduced computational expenses in comparison with direct CAD tool simulations.

Conventionally, building surrogate models for expensive scalar functions is tackled with the help of tools such as Artificial Neural Networks, Support Vector Machines, Polynomial Interpolation, Kriging, Response Surface Methodology, etc. In this PhD thesis, the focus is towards automated generation of surrogate models for expensive CAD simulators. The general class of Linear Time Invariant (LTI) systems are modeled, with specific focus towards high-frequency electromagnetic (EM) systems. Such surrogate models should be able to efficiently parameterize system frequency responses as a function of design parameters. However, parameterizing multiport frequency responses of EM systems are much more challenging than parameterizing scalar functions and must be treated differently. Here, the methods generate frequency-dependent rational models for the selected samples using a system identification technique and then parameterize these rational models with respect to the design parameters using several interpolation schemes. Thus, frequency is considered as a "special" parameter and it is

not considered as the part of the the design parameter space. This approach to modeling has two main advantages: firstly, by modeling the frequency responses of LTI systems as rational function with respect to frequency, it is easier to preserve physical properties such as stability and causality; secondly, by separating the frequency parameter from the design parameter space, the complexity in parameterizing the EM system with respect to the design parameters is significantly reduced. Such surrogate or behavior models for multiport frequency responses are referred to as parametric macromodels or scalable macromodels in the literature. For uniformity, the terminology *parametric macromodel* has been used throughout this PhD thesis. This PhD thesis discusses different aspects of a fully automated parametric macromodeling technique, such as estimating the system order for the frequency-dependent rational functions, choosing an appropriate parameterization strategy, and proper error criteria for assessing the accuracy of the generated parametric macromodels. Additionally, some enhancements to the existing parametric macromodeling techniques are also discussed in this PhD thesis. In addition to being fully automated, the sequential sampling methods along with the parametric macromodeling schemes can be parallelized and also extended to other engineering fields which work in the framework of general LTI systems.

This PhD thesis is organized as follows: Chapter 1 gives a general introduction of the PhD work discussing the context as well as the motivation for this work. An overview of the existing state-of-the-art parametric macromodeling methods is given in Chapter 2 with a discussion on their relative merits and limitations and more importantly their suitability to be combined with efficient sequential sampling schemes for automation. In Chapter 3, an extension of the state-of-the-art parametric macromodeling method based on interpolation of state-space matrices is performed to generate parametric sensitivity macromodels. These parametric sensitivity macromodels can generate analytic sensitivities or gradients of the system responses with respect to the design parameters which can be used in gradient-based design optimizations and sensitivity analysis. Chapter 4 is concerned with bringing automation into the state-of-the-art parametric macromodeling techniques. This chapter can be considered as the first step towards the automation of parametric macromodeling process. The sampling schemes developed in Chapter 4 is tailored towards the local parametric macromodeling schemes which refine the design space using local regions. In Chapter 4, two different grid-based sampling schemes have been developed to sample the design space adaptively. Chapter 5 is application oriented wherein the automated parametric macromodeling process with sequential sampling is used to speed-up the design cycle of a real-life example of a microwave filter. In this chapter, the complete design flow of the filter is described which is assisted with the help of parametric macromodels, thereby gaining insight into the design with very little computational cost. The chapter also shows the amount of design cycle time gained by using a parametric macromodel instead of direct EM simulations in a robust global design optimization and a Monte-Carlo based variability analysis. The results of the optimization are finally verified with the help of measurements on realization of different optimal filter configurations. Additionally, Chapter 5 also proposes an enhanced para-

metric macromodeling strategy which uses multiple frequency scaling coefficients for parameterizing the frequency response transfer function to further improve the modeling capability. In Chapter 6, the sequential sampling method of Chapter 4 is extended to deal with scattered grids using a well-conditioned simplicial refinement called path-simplex division. Chapter 7 gives concluding remarks on the PhD thesis work with some discussion on the major contribution of this work along with possible extensions and future directions. Finally, in Appendix A, an algorithm description of the sequential sampling schemes developed during this work is given to help the reader in its implementation.

1 Introduction

The Design and the development cycle for complex engineering problems are conventionally tackled with the help of carefully planned experiments. These experiments however used to require the realization and the measurement of actual systems which can be very expensive and time consuming. To overcome these difficulties, scientists and engineers around the world have developed Computer Aided Design (CAD) tools (for example, electromagnetic (EM) simulators), which mathematically model the behavior of the actual systems as accurate as possible based on first principles. By using these CAD tools the overall cost of the design and development cycle can reduce considerably by minimizing the resources and requires a minimal number of expensive prototyping. Unfortunately, these accurate simulation tools also have certain disadvantages such as being computationally very expensive. For a typical design cycle which includes design space exploration, design optimization, robustness analysis and variability assessment, multiple CAD simulations are required, hence the computational expense can be considerable. In some cases, it may not be acceptable in practice.

Building efficient and accurate replacement models for the computationally expensive CAD tools allows to significantly improve the design process. This has been investigated in many engineering fields and a large number of general modeling tools have been developed over the recent years. With the help of judiciously chosen samples from the expensive CAD simulator, accurate replacement models are built which are computationally very efficient and thus used in the design flow to further reduce the computational complexity and consequently the design cycle time. Throughout this PhD thesis such models which act as an efficient surrogate

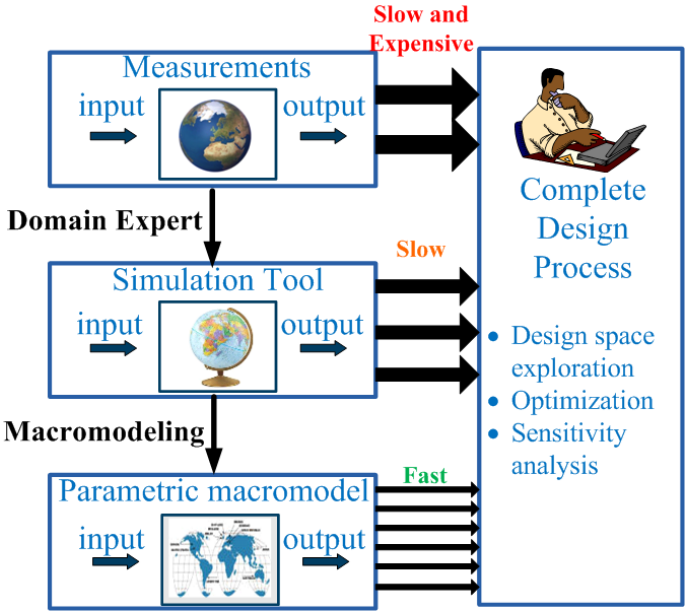

Figure 1.1: Role of efficient parametric macromodeling in the overall design process.

for the complex CAD tool are called as *parametric macromodels*. Fig. 1.1 shows the role of an efficient parametric macromodeling process in the overall design cycle. As explained before, the parametric macromodels are replacement models used to avoid the repetitive use of simulation tools as much as possible. This helps reduce the computational burden on the designer. The time required for each simulation or measurement needed to perform design activities is at least an order of magnitude lower for the parametric macromodel. It is represented by several thin arrows in Fig. 1.1.

Many engineering fields such as electrical engineering, mechanical engineering, chemical engineering, signal processing, etc., use the widely studied Linear Time Invariant (LTI) systems as a cornerstone for design activities. In this PhD thesis, the effort was focussed to investigate parametric macromodeling of complex high-frequency EM systems. The work develops efficient strategies to build accurate parametric macromodels which characterize the system frequency response of the system under test in the form of Scattering (S), Admittance (Y), or Impedance (Z) parameters. The EM systems are modeled both as a function of the frequency and several other design parameters, such as geometric parameters or substrate features for example.

The parameterization of system frequency responses of multiport systems should

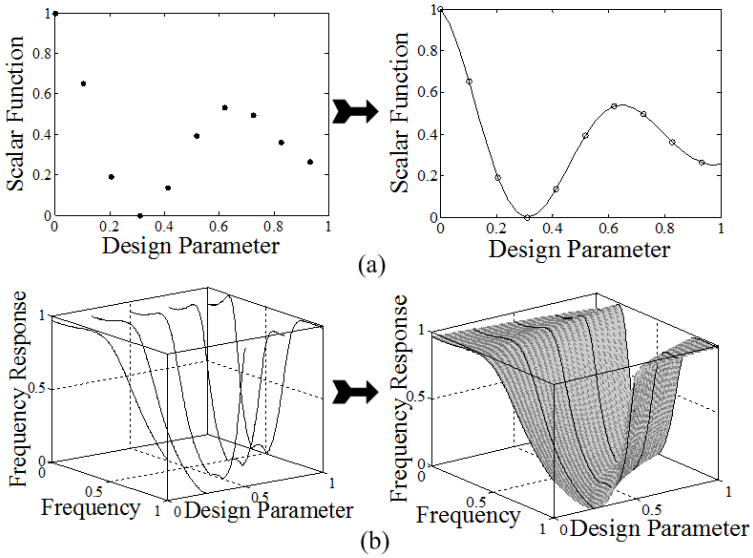

Figure 1.2: (a) Scalar function parameterization versus (b) frequency response parameterization.

be clearly distinguished from the parameterization of scalar functions. Fig. 1.2 clarifies the difference between the two different approaches. Building parametric macromodels for real scalar functions can be done using a large set of well-known tools such as Response Surface Methodology (RSM), Artificial Neural Networks (ANN), Support Vector Machines (SVM), Kriging, Polynomial interpolation schemes, etc. [1]. However, using such methods to build parameterized system frequency responses can be quite challenging due to the following facts:

1. The frequency responses are complex functions with a real part and an imaginary part. Modeling the real part and the imaginary part separately does not guarantee preservation of the causality of the system in the model representation. The real part and the imaginary part of the analytic frequency response function are related by the Kramers-Kronig relations [2].

2. It is difficult to ensure other physical properties such as stability and passivity using these modeling frameworks. This limits their applicability to time-domain analysis where passivity and stability are mandatory to obtain a simulated response such as transient or circuit simulations (for example by using SPICE [3]).

3. Throughout this PhD thesis we deal with parametric macromodels of multiport systems. They model the matrix-valued frequency response functions,

and further increase the modeling complexity.

Keeping in mind the above mentioned problems, the choice was made to directly parameterize the frequency-dependent rational functions with respect to all the parameters except frequency over the complete design space. Remember that this space is spanned by all the design parameters except frequency. This means that the frequency is kept as a "special" parameter and the system frequency responses are parameterized with respect to frequency as rational forms with the help of a system identification method. In this PhD thesis, the frequency-dependent rational models are referred to as *root macromodels*. In the next step, these *root macromodels* are parameterized with respect to the design parameters using some carefully crafted interpolation schemes. The methods developed here are more generally applicable and the formulations can be extended to other engineering areas which can be analyzed using framework of LTI systems.

In the literature, a considerable amount of work has been done regarding the parametric macromodeling of the frequency response of EM systems. Almost all of these methods fall into two main categories, *global* and *local* parametric macromodeling methods. *Global* modeling schemes build a single model for the complete design space which includes frequency as well [4–10]. Multi-dimensional Cauchy methods are one of the popular global modeling schemes for parametric macromodeling of system frequency responses [5–7, 10]. An adaptive multivariate rational fitting is reported in [7] which uses Tchebyshev orthogonal polynomials to improve the conditioning of the matrices to be solved. Neural network and radial basis function-based methods are also found in the literature [8, 9] which are also global modeling schemes. However, such global modeling schemes [4–10] suffer from the following limitations:

1. For these methods, it is difficult to preserve physical properties such as stability and passivity of the generated macromodels over the design space of interest.

2. For relatively high dimensions the memory requirement can be prohibitively high limiting their applicability. Dimensionality reduction techniques such as [11] could help improve this scenario to a certain extend.

3. It is also difficult to automate these modeling schemes in a robust way.

Local parametric macromodeling schemes distinguish themselves from the *global* modeling schemes in many ways. In contrast to the *global* modeling schemes, the *local* schemes work on local regions of the design space even though they are valid for the complete design space. It should be noted here that this definition is not equivalent to other definitions of *local* and *global* schemes such as [12] where they are defined as "locally" accurate or "globally" accurate models respectively. Fig. 1.3 shows a two parameter design space region (which excludes

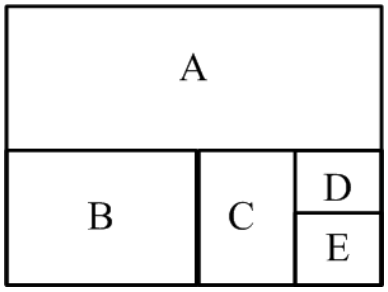

Figure 1.3: Local parametric macromodeling on a two parameter design space

frequency) divided into different sections in case of a *local* parametric macromodeling method. Each of these subregions has its own local parametric macromodel which is independent of the others. This makes preserving system properties such as stability and passivity much easier. Also, since these models can be built sequentially (with different regions having different parametric macromodels), there is simply no need for huge matrices to solve for the response of the parameters at once as in the case of *global* modeling schemes. However, the *local* schemes do have other limitations such as limited extrapolation capability, compactness in terms of the modeling parameters and limited differentiation capability. Despite these limitations, the *Local* parametric macromodeling methods [13–18], if used properly, can tackle all the above stated problems associated with the *global* methods. Also, *local* schemes can be easily used in a automated modeling procedure as it will be shown in this PhD thesis. The work done in this PhD thesis is focused towards sequential sampling strategies for existing local parametric macromodeling methods.

As discussed above, the local parametric macromodeling schemes are a very efficient way of generating accurate parametric macromodels for system frequency responses. To obtain high quality models, such methods assume a priori knowledge on several modeling parameters such as the initial number of expensive EM simulations needed to build the model, the distribution of these EM simulations over the design space and the model order for the *root macromodels*. However, this information is generally not available to the system designer. Hence, these efficient methods can only become applicable in a design process, if automation is provided to obtain the information. The majority of the work done in this PhD thesis is motivated by the need of automation for such parametric macromodeling methods which otherwise would remain unused despite their straightforward possible use in an industrial context.

The selection of the amount and distribution of expensive EM simulations in the design space are of great importance to an automated modeling strategy as

these define the accuracy of the generated parametric macromodel. If the number of EM simulations is too low, it might lead to *undersampling*. This results in an inaccurate parametric macromodel. On the contrary, if the number of EM simulations performed is too high, leading to an *oversampling*, a large amount of computational resources will be wasted, and this should be avoided at any cost. Despite the importance of these selections, there is very little work done in the literature to automate these efficient modeling methods. This lack inspired the PhD thesis.

The sequential sampling strategies are generally classified into three major schemes:

Input-Based Methods: fill the design parameter space according to some measure depending on the density of the selected samples [12, 19]. For example, the sequential Design of Experiments (DoE) method falls into this category.

Output-Based Methods: select the EM simulation samples in the design space depending on the system frequency response [20, 21]. That is, the raw system frequency responses are analyzed and some measure is generated (for example, say, the derivatives of the frequency responses) to know where the new EM simulation has to be performed.

Model-Based Methods: depend on intermediate macromodels to select the distribution of new samples as described in [5–9, 22]. Even though the work done in [5–9, 22] treats global parametric macromodeling methods, the model-based methods can easily be adapted to local parametric macromodeling methods too.

Since, efficient local parametric macromodeling schemes are already available, the best way to go forward is to use them in the sequential sampling process. The PhD thesis develops model-based sequential sampling strategies which are coupled with local parametric macromodeling methods. Fig. 1.4 shows an example of such sequential sampling strategies with local parametric macromodeling. Fig. 1.4 is the sequential sampling evolution that leads to Fig. 1.3 which is represented in a tree-like fashion. It can be seen from Fig. 1.4 that these methods start with the corner *root macromodels* and then divide the design space into subregions depending on the used-imposed accuracy target. The user can define varying levels of accuracy targets thereby generating variable accuracy or multiple fidelity models.

1.1 Research contributions

The PhD thesis investigates the use and develops different new property-preserving, local parametric macromodeling schemes that can be used in a complete design

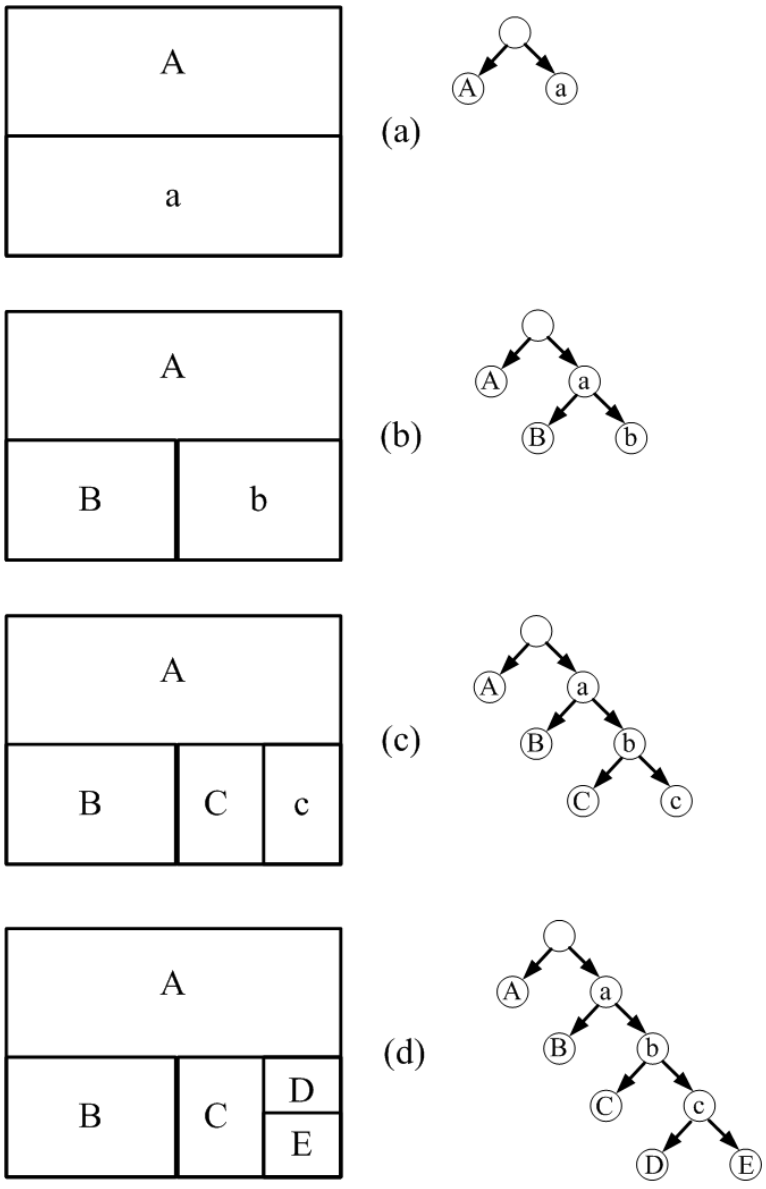

Figure 1.4: Local parametric macromodeling with two parameter design space regions (left) and the corresponding tree-based structure (right).

procedure for EM systems. Specifically, the aim was to find an automated, robust, parametric macromodeling tool which can help the designer to have an efficient design cycle with very little a priori knowledge on selecting the number and the distribution of EM simulation samples. This automated modeling tool can therefore be integrated in the design as a CAD tool. The goal is that with the push of a button, an efficient and accurate parametric macromodel is generated. The generated model serves as a replacement model for the EM solver evaluation, speeding-up the multiple evaluations in the design procedure. In addition, the macromodel will also give an insight into the design which can be of tremendous use to the designer.

In this PhD thesis, Chapter 2 gives a primer on the state-of-the-art property preserving local parametric macromodeling strategies. Chapter 3 is an initial work on the possibility of extension of the parametric macromodeling schemes to model parameterized sensitivities. From Chapter 4 onwards this PhD thesis describes the work on efficient parametric macromodel generation. A brief description of each chapter is presented here with their contribution in the overall research as shown in Fig. 1.5.

Chapter 2

Chapter 2 gives an introduction of the data-driven parametric macromodeling process, such as the definition of the design parameter space, frequency-response identification, different parameterization strategies etc. The chapter also discusses the motivation for different choices on the error measures, and macromodeling strategies that maximize the accuracy, flexibility, automation and efficiency of the entire modeling process. The chapter concludes with the description of a complete parametric macromodeling flow.

Chapter 3

This chapter extends the state-of-the-art local parametric macromodeling techniques towards the generation of the parameterized output sensitivity responses of multiport systems. First, the state-space models of the frequency response data are identified at different design space points using a system identification technique. Then, the sensitivities are calculated by parameterizing the state-space models over the design parameters using continuously differentiable interpolation schemes. Analytical examples are used to validate the parameterized sensitivities. The models are then used in a microwave filter design context. The calculated sensitivities are supplied to the gradient-based optimizers to improve the optimization efficiency.

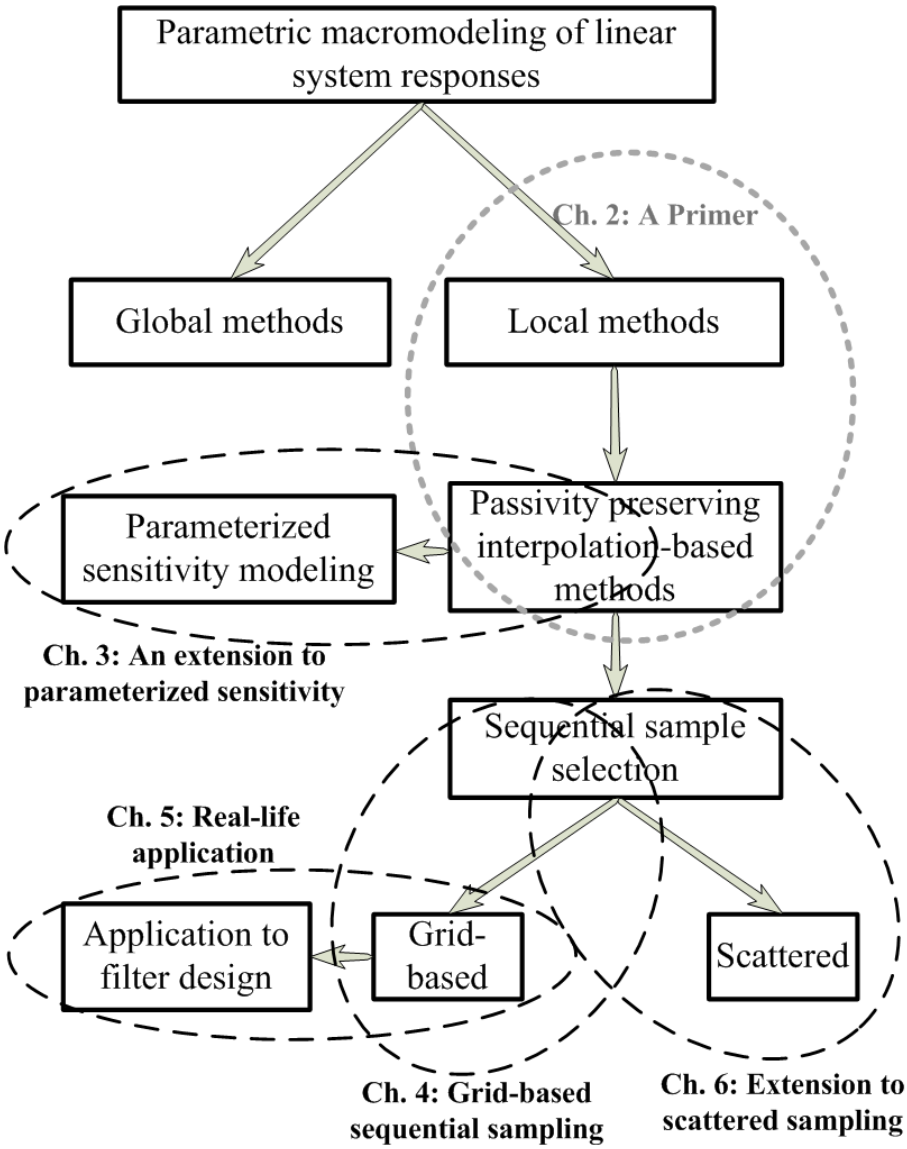

Figure 1.5: Research contribution.

Chapter 4

The state-of-the-art parametric macromodeling techniques are very efficient alternatives to direct EM solver evaluations in the design process. However, their applicability is questionable if these methods can not be automated. This chapter investigates the use of sequential sampling strategies to make the overall modeling effective by automating the complete modeling process.

The sampling methods discussed in this chapter use local refinements of the design parameter space separately in a tree-like fashion with independent branches forming different refinement regions as shown in Fig. 1.4. Different sampling schemes are described which refine the local N-dimensional hyperrectangular regions of the design space. This implementation can thus be very easily used for parallel processing. Also, the chapter investigates several error measures that can be used to assess the accuracy of the generated macromodels as well as building multiple fidelity models (Fig. 1.4). Therefore, this chapter can be considered as a first step towards the automation of several state-of-the-art local parametric macromodeling techniques.

Chapter 5

The aim of this chapter is to validate the automated parametric macromodeling methods developed in Chapter 4 by applying them to a complete modeling and design flow for a real life, microwave filter application. The chapter also develops a new parametric macromodeling method which enhances upon a previously proposed modeling scheme to further improve the macromodeling capability. A detailed explanation is given on how the models are obtained and how they are used in a design process. Multiple optimization scenarios are tackled here with the parametric macromodel of the system. Also the parametric macromodel is used for Monte-Carlo variability analysis on each of the obtained solutions, and is further validated with measurements.

Chapter 6

This chapter investigates how the grid-based sampling schemes discussed in Chapter 4 can be extended to simplicial refinements. That is, instead of refining the N-dimensional hyperrectangular girds, the N-simplex regions are refined. The refinement is kept local so as to keep all the advantages of the local parametric macromodeling discussed in Chapter 2. A well-conditioned simplicial refinement known as path-simplex division [23] is investigated here. This method is in contrast with the other refinement procedures, such as the well known Delaunay refinement [24], since it a local refinement. This ensures the possibility of parallelization, whereas with Delaunay refinement many neighboring simplicial regions

change, thereby changing the overall model structure. This is explained in detail in this chapter.

Also, this chapter compares the grid-based and path-simplex refinement schemes. A hybrid sampling scheme which combines both the refinement strategies is also discussed here to show, with the help of numerical examples, that an improvement in sampling distribution and consequently modeling efficiency is indeed possible.

1.2 Publications

The research results obtained during this PhD have been published in scientific journals and presented at a series of international conferences. The following list provides an overview of the publications during the PhD research.

1.2.1 Publications in international journals (listed in the Science Citation Index [1])

1. **Krishnan Chemmangat**, Francesco Ferranti, Luc Knockaert, and Tom Dhaene. **Parametric macromodeling for sensitivity responses from tabulated data,** *IEEE Microwave and Wireless Components Letters*, 21, no. 8 (2011): pp. 397-399.

2. **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene, and Luc Knockaert. **Gradient-based optimization using parametric sensitivity macromodels,** *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 25, no. 4 (2012): pp. 347-361.

3. **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene, and Luc Knockaert. **Scalable models of microwave system responses using sequential sampling on unstructured grids,** *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 27, no. 1 (2014): pp. 122-137.

4. **Krishnan Chemmangat**, Tom Dhaene, and Luc Knockaert. **Scalable Macromodeling of Microwave Systems Responses using Sequential Sampling with Path-Simplexes,** *Electronics Letters* , 45, no. 15 (2013): pp. 950-952.

5. **Krishnan Chemmangat**, Tom Dhaene, and Luc Knockaert. **Auto-generation of Passive Scalable Macromodels for Microwave Components using Scattered Sequential Sampling,** *International Journal of Microwave and Wireless Technologies*, Accepted, Online Available: http://dx.doi.org/10.1017/S1759078714000038 (2014).

6. **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene and Luc Knockaert. **Parametric Macromodeling of Linear High-Frequency Systems using Multiple Frequency Scaling and Sequential Sampling,** *Electronics Letters*, 50, no.6 (2014): pp. 475-476.

[1] The publications listed are recognized as 'A1 publications', according to the following definition used by Ghent University: A1 publications are articles listed in the Science Citation Index, the Social Science Citation Index or the Arts and Humanities Citation Index of the ISI Web of Science, restricted to contributions listed as article, review, letter, note or proceedings paper.

7. Domenico Spina, **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene, Luc Knockaert and Flavio Canavero. **A Comparative Study of Polynomial Chaos and Stochastic Collocation Methods for Variability Analysis of Multi-port Systems** *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, Submitted (2014).

8. Mostafa Pakparvar, David Plets, Emmeric Tanghe, Dirk Deschrijver, Wei Liu, **Krishnan Chemmangat**, Ingrid Moerman, Tom Dhaene, Luc Martens, and Wout Joseph. **A Cognitive QoS Management framework for WLANs,** *Wireless Communications and Mobile Computing*, Submitted (2013).

1.2.2 Publications in book chapters

1. **Krishnan Chemmangat**, Francesco Ferranti, Luc Knockaert, and Tom Dhaene. **Parametric Sensitivity Macromodels for Gradient-Based Optimization,** *Simulation-Driven Design Optimization and Modeling for Microwave Engineering*, Imperial College Press, pp. 247-270. 2013.

1.2.3 Publications in international conferences (listed in the Science Citation Index [2])

1. **Krishnan Chemmangat**, Francesco Ferranti, Luc Knockaert, and Tom Dhaene. **Sensitivity analysis using data-driven parametric macromodels,** *15th IEEE Workshop on Signal Propagation on Interconnects (SPI-2011), Naples, Italy*, pp. 111-114. IEEE, 2011.

2. Francesco Ferranti, Giulio Antonini, **Krishnan Chemmangat**, Luc Knockaert, and Tom Dhaene. **Partial Element Equivalent Circuit models in the solution of the electric field integral equation,** *International Conference on Electromagnetics in Advanced Applications (ICEAA), 2012, Cape Town, South Africa*, pp. 329-332. IEEE, 2012.

3. **Krishnan Chemmangat**, Dirk Deschrijver, Ivo Couckuyt, Tom Dhaene, and Luc Knockaert. **Fast optimization of microwave filters using surrogate-based optimization methods,** *International Conference on Electromagnetics in Advanced Applications (ICEAA), 2012, Cape Town, South Africa*, pp. 212-215. IEEE, 2012.

[2]The publications listed are recognized as 'P1 publications', according to the following definition used by Ghent University: P1 publications are proceedings listed in the Conference Proceedings Citation Index - Science or Conference Proceedings Citation Index - Social Science and Humanities of the ISI Web of Science, restricted to contributions listed as article, review, letter, note or proceedings paper, except for publications that are classified as A1.

4. **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene, and Luc Knockaert. **Optimization of high-speed electromagnetic systems with accurate parametric macromodels generated using sequential sampling of the design space,** *International Conference on Electromagnetics in Advanced Applications (ICEAA), 2012, Cape Town, South Africa* , pp. 128-131. IEEE, 2012.

5. Francesco Ferranti, **Krishnan Chemmangat**, Tom Dhaene, and Luc Knockaert. **Self-Constructing Compact Parametric Macromodels for Efficient Electronic Design,** *IEEE 17th Workshop on Signal and Power Integrity (SPI-2013), Paris, France*, pp. 1-4. IEEE 2013.

6. Elizabeth R Samuel, **Krishnan Chemmangat**, Dirk Deschrijver, Tom Dhaene, and Luc Knockaert. **Model Order Reduction of Parameterized State-Space Systems with Sequential Sampling,** *International Symposium on Electromagnetic Compatibility (EMC Europe-2013), Brugges, Belgium*, pp. 342-347. IEEE 2013.

1.2.4 Publications in other international conferences

1. **Krishnan Chemmangat**, Francesco Ferranti, Tom Dhaene, and Luc Knockaert. **Tree-based sequential sampling algorithm for scalable macromodeling of high-speed systems,** *IEEE 16th Workshop on Signal and Power Integrity (SPI-2012), Sorrento, Italy*, pp. 49-56. IEEE, 2012.

References

[1] D. Gorissen, I. Couckuyt, P. Demeester, T. Dhaene, and K. Crombecq, "A surrogate modeling and adaptive sampling toolbox for computer based design," *The Journal of Machine Learning Research*, vol. 99, pp. 2051–2055, 2010.

[2] J. S. Toll, "Causality and the dispersion relation: Logical foundations," *Phys. Rev.*, vol. 104, pp. 1760–1770, Dec 1956. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRev.104.1760

[3] L. Nagel, "Spice2: A computer program to simulate semiconductor circuits," *University of California, Berkeley, Electr. Res. Lab. Report ERL M520*, 1975.

[4] D. Deschrijver, T. Dhaene, and D. De Zutter, "Robust parametric macromodeling using multivariate orthonormal vector fitting," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 7, pp. 1661–1667, 2008.

[5] S. Peik, R. Mansour, and Y. Chow, "Multidimensional Cauchy method and adaptive sampling for an accurate microwave circuit modeling," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 46, no. 12, pp. 2364 –2371, Dec. 1998.

[6] A. Lamecki, P. Kozakowski, and M. Mrozowski, "Efficient implementation of the Cauchy method for automated CAD-model construction," *IEEE Microwave and Wireless Components Letters,*, vol. 13, no. 7, pp. 268 –270, July 2003.

[7] A. Cuyt, R. Lenin, S. Becuwe, and B. Verdonk, "Adaptive multivariate rational data fitting with applications in electromagnetics," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 54, no. 5, pp. 2265 – 2274, May 2006.

[8] V. Devabhaktuni, B. Chattaraj, M. Yagoub, and Q.-J. Zhang, "Advanced microwave modeling framework exploiting automatic model generation, knowledge neural networks, and space mapping," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 51, no. 7, pp. 1822 – 1833, July 2003.

[9] A. Lamecki, L. Balewski, and M. Mrozowski, "Adaptive CAD-model construction schemes," *IEEE Transactions on Magnetics,*, vol. 45, no. 3, pp. 1538 –1541, March 2009.

[10] P. Basl, R. Gohary, M. Bakr, and R. Mansour, "Modelling of electromagnetic responses using a robust multi-dimensional Cauchy interpolation technique," *IET Microwaves, Antennas Propagation,*, vol. 4, no. 11, pp. 1955 –1964, Nov. 2010.

[11] D. Deschrijver, A. Narbudowicz, E. Laermans, and T. Dhaene, "On the application of dimensional analysis to parametric macromodeling," *IEEE Microwave and Wireless Components Letters*, vol. 20, no. 4, pp. 190–192, 2010.

[12] K. Crombecq, E. Laermans, and T. Dhaene, "Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling," *European Journal of Operational Research*, vol. 214, no. 3, pp. 683 – 696, Nov. 2011.

[13] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *IEEE Microwave and Wireless Component Letters*, vol. 19, no. 10, pp. 608–610, Oct. 2009.

[14] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *IEEE International Conference on Signal Propagation and Interconnects*, pp. 117 –120, May 2010.

[15] F. Ferranti, L. Knockaert, and T. Dhaene, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623–629, Aug. 2010.

[16] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[17] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2394 –2403, Oct. 2011.

[18] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139 –151, March/April 2012.

[19] D. Montgomery, *Design and analysis of experiments*, ser. Student solutions manual. Wiley, 2008. [Online]. Available: http://books.google.be/books?id=kMMJAm5bD34C

[20] D. Deschrijver, K. Crombecq, H. Nguyen, and T. Dhaene, "Adaptive sampling algorithm for macromodeling of parameterized S-parameter responses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 1, pp. 39 –45, Jan. 2011.

[21] K. Crombecq, D. Gorissen, D. Deschrijver, and T. Dhaene, "A novel hybrid sequential design strategy for global surrogate modeling of computer experiments," *SIAM Journal on Scientific Computing*, vol. 33, no. 4, pp. 1948–1974, Aug. 2011.

[22] R. Lehmensiek and P. Meyer, "Creating accurate multivariate rational interpolation models of microwave circuits by using efficient adaptive sampling to minimize the number of computational electromagnetic analyses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 8, pp. 1419 –1430, Aug. 2001.

[23] J. Brandts, S. Korotov, and M. Krizek, "Dissection of the R^npath-simplex in into n path-subsimplices," *Linear Algebra and its Applications*, vol. 421, no. 23, pp. 382 – 393, 2007.

[24] A. Constantiniu, P. Steinmann, T. Bobach, G. Farin, and G. Umlauf, "The adaptive delaunay tessellation: a neighborhood covering meshing technique," *Computational Mechanics*, vol. 42, no. 5, pp. 655–669, 2008.

2

An Introduction to Parametric Macromodeling of Linear Multiport High-Frequency Systems

In this chapter, an introduction to the parametric macromodeling of passive high-frequency systems is given. The main aim of this chapter is to give the context of this PhD thesis and the motivation for this work by describing the state-of-the-art local parametric macromodeling techniques and the need for the opportunities created by further improvement with automated sequential sampling schemes. These techniques for building efficient models are discussed here with their corresponding merits and limitations. The chapter also discusses how automation can be introduced into the complete model building process. This serves as the source of inspiration for the rest of this PhD thesis.

2.1 Introduction

Generating accurate and efficient parametric macromodels for the frequency response of EM systems is an active field of research. In this chapter, different property preserving parametric macromodeling methods are discussed briefly. They consist of different steps such as the rational identification of the complex frequency responses, the passivity preserving interpolation schemes for parameterizing these rational forms with respect to other design variables and selection of

samples in the design space to build the accurate parametric macromodel. The state-of-the-art of the local parametric macromodeling is discussed in this chapter along with the limitations of such methods which are the driving points for the work described in this PhD thesis. Finally, the complete automated and efficient parametric macromodeling process which incorporates all the steps listed above is briefly explained. This is later explained in the PhD thesis.

2.2 Rational identification of the system frequency responses

The first and foremost step of the parametric macromodeling process is to identify frequency response functions parameterized as a rational model (also called transfer functions) from known frequency response samples. Such rational models should be able to preserve system properties such as stability and passivity. In the case of a rational model, stability means that the poles of the rational form lie on the left half of the Laplace plane. Passivity refers to the property that systems that cannot generate more energy than they absorb through their electrical ports. In this PhD thesis the well-known rational identification scheme called Vector Fitting (VF) [1–3] is used. The method starts from a set of data samples given by,

$$(s_i, \vec{g}_k), \mathbf{H}(s_i, \vec{g}_k), \tag{2.1}$$

which depend on the complex frequency $s = j\omega$ and several design parameters $\vec{g} = (g^{(1)}, g^{(2)}, \ldots, g^{(N)})$, such as layout features or substrate parameters. The frequency response $\mathbf{H}(s_i, \vec{g}_k) \in \mathbb{C}^{N_{port} \times N_{port}}$ is a matrix valued complex frequency response with N_{port} number of ports. In (2.1), $i = 1, 2, \ldots, N_s$ are the samples along the frequency axis and $k = 1, \ldots, K_{tot}$ are the samples in the rest of the design parameter space $\vec{g}$ which excludes frequency. From these data samples, for every design space point $\vec{g}_k$, frequency-dependent rational macromodels called *root macromodels* are built using the VF technique [1–3]. Each *root macromodel* $\mathbf{R}_{\vec{g}_k}(s)$ has the following pole-residue rational form:

$$\mathbf{R}_{\vec{g}_k}(s) = \sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\mathbf{c}_n^{\vec{g}_k}}{s - a_n^{\vec{g}_k}} + \mathbf{d}^{\vec{g}_k} \tag{2.2}$$

The parameters in the rational model (2.2), $a_n^{\vec{g}_k} \in \mathbb{C}$, $\mathbf{c}_n^{\vec{g}_k} \in \mathbb{C}^{N_{port} \times N_{port}}$ and $\mathbf{d}^{\vec{g}_k} \in \mathbb{C}^{N_{port} \times N_{port}}$ represent poles, residues and feed forward terms respectively at the design point $\vec{g}_k = (g_k^{(1)}, g_k^{(2)} ..., g_k^{(N)})$. $N_P^{\vec{g}_k}$ is the number of rational terms in the pole-residue form of the *root macromodel* $\mathbf{R}_{\vec{g}_k}(s)$. The problem of finding the unknown coefficients in (2.2) is nonlinear, since the poles $a_n^{\vec{g}_k}$ appear in the denominator and the residues appear in the nominator. VF solves this nonlinear

problem by separating it into two linear problems following the lines of the procedure described by Sanathanan and Koerner [4]. This is done by introducing a set of starting poles b_n and an unknown rational function $\sigma(s)$ in a pole-residue form such that:

$$\begin{bmatrix} \sigma(s)\mathbf{R}_{\vec{g}_k}(s) \\ \sigma(s) \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\mathbf{c}_n^{\vec{g}_k}}{s-b_n} + \mathbf{d}^{\vec{g}_k} \\ \sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\tilde{\mathbf{c}}_n^{\vec{g}_k}}{s-b_n} + 1 \end{bmatrix} \tag{2.3}$$

From (2.3) we have:

$$\sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\mathbf{c}_n^{\vec{g}_k}}{s-b_n} + \mathbf{d}^{\vec{g}_k} = \left[\sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\tilde{\mathbf{c}}_n^{\vec{g}_k}}{s-b_n} + 1 \right] \mathbf{R}_{\vec{g}_k}(s). \tag{2.4}$$

It should be noted in (2.4) that the rational approximation for $\sigma(s)$ as well as $\sigma(s)\mathbf{R}_{\vec{g}_k}(s)$ has the same number of poles. In the VF formulation this is achieved since the zeros of the rational function $\sigma(s)$ are equated to the poles of the original function $\mathbf{R}_{\vec{g}_k}(s)$.

As stated before, VF solves the nonlinear problem as a sequence of linear problems in two stages. In the first stage, the unknowns $\mathbf{c}_n^{\vec{g}_k} \in \mathbb{C}^{N_{port} \times N_{port}}$, $\tilde{\mathbf{c}}_n^{\vec{g}_k} \in \mathbb{C}^{N_{port} \times N_{port}}$ and $\mathbf{d}^{\vec{g}_k} \in \mathbb{C}^{N_{port} \times N_{port}}$ in (2.4) are found by solving an over-determined linear least squares problem over several frequency samples s_i, $i = 1, 2, \ldots, N_s$ as,

$$\sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\mathbf{c}_n^{\vec{g}_k}}{s_i-b_n} + \mathbf{d}^{\vec{g}_k} = \left[\sum_{n=1}^{N_P^{\vec{g}_k}} \frac{\tilde{\mathbf{c}}_n^{\vec{g}_k}}{s_i-b_n} + 1 \right] \mathbf{H}_{\vec{g}_k}(s_i). \tag{2.5}$$

Note that in (2.5), the rational form $\mathbf{R}_{\vec{g}_k}(s)$ is replaced by the data sample $\mathbf{H}_{\vec{g}_k}(s_i)$. In the second stage, the starting poles b_n are updated as the zeros of the current estimate of the rational function $\sigma(s)$. This new set of poles are calculated from the old starting poles and the residues $\tilde{\mathbf{c}}_n^{\vec{g}_k}$ of the $\sigma(s)$ function. These two stages are repeated iteratively until a convergence is observed [1].

The set of starting poles b_n in case of smooth data sets are selected as real poles either linearly or logarithmically placed over the frequency range. In case of functions with distinct resonant peaks, complex-conjugate pole pairs with the imaginary part distributed across the frequency range, are selected. Also, the real part of each complex pole is selected such that they are significantly smaller (typically $\frac{1}{100}^{\text{th}}$) in magnitude in comparison with the imaginary part. More details of this procedure are described in [1].

If unstable poles are generated during an iteration (that is, the zeros of the estimated $\sigma(s)$ lies in the right half of the Laplace plane), a pole-flipping scheme is used to enforce stability [1]. The passivity assessment is performed on the system

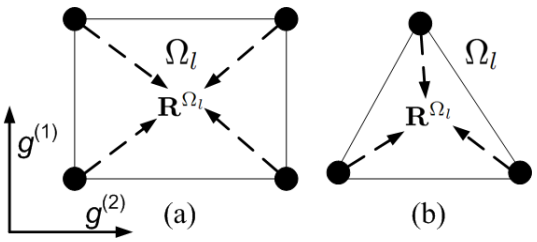

Figure 2.1: Local parametric macromodeling region in 2 D design space.

by calculating the Hamiltonian matrix eigenvalues as described in [5, 6]. The passivity enforcement is accomplished using the standard techniques [7, 8] which remove the passivity violations by perturbation of the residue matrix eigenvalues. Other passivity enforcement schemes which work directly on pole perturbation [9] and residue matrix perturbation [10] of system state-space matrices also exists in the literature which can be used. This initial step of the proposed method results in a set of stable and passive *root macromodels*. The *root macromodels* in (2.2) can also be represented in another well-known form called the state-space form given as,

$$\mathbf{R}_{\vec{g}_k}(s) = \mathbf{C}_{\vec{g}_k}(s\mathbf{I} - \mathbf{A}_{\vec{g}_k})^{-1}\mathbf{B}_{\vec{g}_k} + \mathbf{D}_{\vec{g}_k} \tag{2.6}$$

In (2.6), the matrices $\mathbf{A}_{\vec{g}_k} \in \mathbb{R}^{N_P^{\vec{g}_k} N_{port} \times N_P^{\vec{g}_k} N_{port}}$, $\mathbf{B}_{\vec{g}_k} \in \mathbb{R}^{N_P^{\vec{g}_k} N_{port} \times N_{port}}$, $\mathbf{C}_{\vec{g}_k} \in \mathbb{R}^{N_{port} \times N_P^{\vec{g}_k} N_{port}}$, and $\mathbf{D}_{\vec{g}_k} \in \mathbb{R}^{N_{port} \times N_{port}}$ are the system state-space matrices and $\mathbf{I} \in \mathbb{R}^{N_P^{\vec{g}_k} N_{port} \times N_P^{\vec{g}_k} N_{port}}$ is the identity matrix.

In this text, the local parametric macromodeling schemes either perform interpolation on the rational models given in (2.2) or on the state-space matrices of (2.6). This will be explained in detail in the next section.

2.3 Overview of the local parametric macromodeling methods

An overview of the state-of-the-art local parametric macromodeling schemes is given in the following subsection. These parametric macromodeling frameworks can be coupled with the sequential sampling schemes proposed in this PhD thesis to have a robust modeling strategy. As stated before, the word "local" is used to emphasize the fact that, in these methods, different parametric macromodels are built for different regions of the design space even though they are "globally" accurate over the complete design space of interest.

Fig. 2.1 shows two dimensional design space formulation with two design parameters $\vec{g} \in (g^{(1)}, g^{(2)})$ which can form either a rectangular (or regular grid) as in

Fig. 2.1.a or a triangular (or scattered grid) region as in Fig. 2.1.b. The region with index l is denoted as Ω_l. The corner points which span these regions are frequency-dependent LTI *root macromodels* which can be represented in a rational form (2.2) or a state-space form (2.6). The design space Ω consists of a set of several such regions $\Omega = \{\Omega_l;\ l = 1, \ldots, L\}$. Each region has a separate parametric macromodel $\mathbf{R}^{\Omega_l}(s, \vec{g})$. Note that in a general N-dimensional design space, a regular grid will consist of several N-dimensional hyperrectangular regions. Each region has 2^N corner *root macromodels*. For scattered grids the design space is built-up by N-simplexes with $N + 1$ corner *root macromodels*.

Fig. 2.2 shows the different local parametric macromodeling schemes. Table 2.1 gives a comparison of the local parametric macromodeling schemes shown in Fig. 2.2 in terms of the measures listed below.

> *Model Compactness*: refers to the model order of the resulting parametric macromodel rational form which is given scores from low to high. A high score on the compactness indicates an efficient parametric macromodeling method.
>
> *Modeling power*: refers to the ability of the parametric macromodeling method to generate an accurate model with a minimum number of EM simulations. This is measured in terms of the number of model parameters (for example, residues of the rational form) being interpolated over the design parameter space. A high score on modeling power means a good modeling scheme. This means that, for a given number of model generation samples, the one with the low score is less accurate in comparison with the one with high score.
>
> *Passivity preservation*: measure refers to the mathematical complexity for ensuring that the generated models are passive. It is categorized into two which are namely "cheap" and "expensive" and a detailed explanation is given in the following subsections.

A brief description of each of the parametric macromodeling methods is given below.

2.3.1 Interpolation of the system transfer functions

In these methods [11, 12], the *root macromodels* which are the frequency-dependent rational functions, identified with the help of VF [1–3] are interpolated along the design parameters using some special interpolation operators. In the case of Y or Z-parameter representations, the use of interpolation techniques with positive interpolation kernels are used. These positive interpolation kernel ensures that the interpolation weights given to each *root macromodel* for the parameterization is

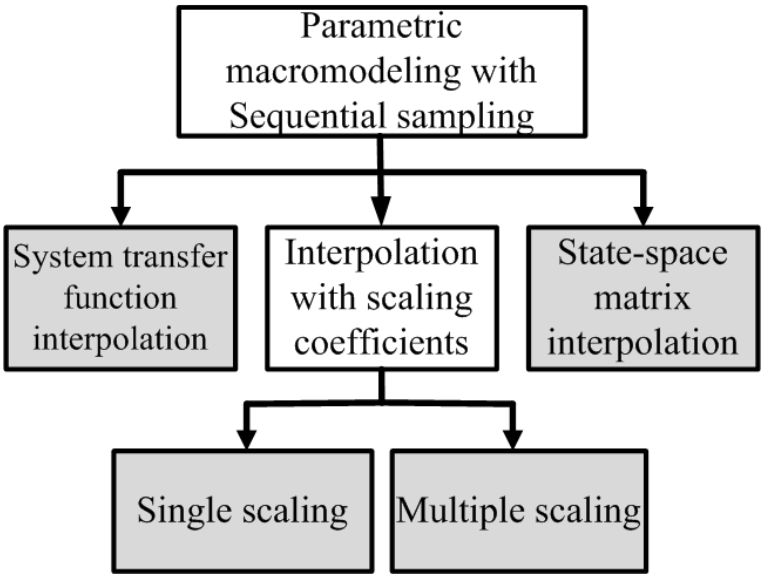

Figure 2.2: Overview of different state-of-the-art local parametric macromodeling techniques.

positive. For S-parameter representation, positive interpolation operator which is also bounded is used. This special class of interpolation operators ensures that the generated parametric macromodels are stable and passive over the design space of interest if the *root macromodels* are stable and passive.

The method is quite robust but all the *root macromodels* of the local region are interpolated and hence the compactness of the model reduces. This is due to the fact that the total model order is the sum of the orders of the corner *root macromodels*. In case of the grid-based scheme where there are 2^N corner *root macromodels* it is less compact than a simplicial refinement scheme with $N + 1$ corners as seen in Table 2.1. This gives a low and medium scores for grid-based and simplicial refinement schemes respectively. Since this method is the direct interpolation of the transfer functions, the modeling power of such methods might be limited with respect to other schemes as only the residues and direct terms are parameterized. It is relatively easy to ensure passivity property for the models generated using positive/bounded interpolation schemes which are extremely cheap to evaluate.

Interpolation scheme	Model Compactness		Modeling Power		Passivity preservation	
	Rational model order		Parameterization (refer eq. (2.2), (2.6) & (2.8))		Interpolation condition	
Transfer function	$N_P^{\min} \leq n_P \leq 2^N N_P^{\max}$ (Grid) $N_P^{\min} \leq n_P \leq (N+1) N_P^{\max}$ (Scat.)	low medium	$\mathbf{c}_n^{\vec{g}_k}$ and $\mathbf{d}^{\vec{g}_k}$	low	positive (Y, Z) /bounded (S)	cheap
State-space matrix	$n_p = N_P$	high	$\mathbf{A}_{\vec{g}_k}$, $\mathbf{B}_{\vec{g}_k}$, $\mathbf{C}_{\vec{g}_k}$, and $\mathbf{D}_{\vec{g}_k}$	high	positive (Y, Z) /bounded (S) + LMI solution	expensive
single scaling coefficient	$N_P^{\min} \leq n_P \leq 2^N N_P^{\max}$ (Grid) $N_P^{\min} \leq n_P \leq (N+1) N_P^{\max}$ (Scat.)	low medium	$\mathbf{c}_n^{\vec{g}_k}$, $\mathbf{d}^{\vec{g}_k}$, and $[\alpha_1,\ \alpha_2]$ (limited DOF)	medium	positive (Y, Z) /bounded (S)	cheap
multiple scaling coefficients	$N_P^{\min} \leq n_P \leq 2^N N_P^{\max}$ (Grid) $N_P^{\min} \leq n_P \leq (N+1) N_P^{\max}$ (Scat.)	low medium	$\mathbf{c}_n^{\vec{g}_k}$, $\mathbf{d}^{\vec{g}_k}$, and $[\alpha_1,\ \beta_n]$ (multiple DOF)	medium /high	-	-

Table 2.1: Comparison of several state-of-the-art local parametric macromodeling methods; in the table, DOF is the degrees of freedom. The quantities $N_P^{\min} = \min_k N_P^{\vec{g}_k}$ and $N_P^{\max} = \max_k N_P^{\vec{g}_k}$ are the minimum and maximum orders of the corner root macromodel rational forms.

2.3.2 Interpolation of the state-space matrices

The system state-space matrices of the obtained *root macromodels* (refer equation (2.6)) can also be parameterized as in [13–17] instead of the system transfer functions. The idea here is to first convert each *root macromodels* obtained in pole-residue form (2.2) to a state-space form (2.6) and then parameterize them using special interpolation schemes over the complete design space. Here, the state-space matrices of the same order are interpolated and hence the model order is kept same throughout the design parameter space. Thus the model is very compact and is given a high score in Table 2.1. These methods allow to parameterize poles, residues and direct terms indirectly by parameterizing the states-pace matrices, hence their modeling capability is increased with respect to the previous methods [11, 12], where only residues and direct term are parameterized. So, they are given a high score in their modeling power (see Table 2.1). The methods [13–17] are ensured to be stable if positive interpolation operators are used. However, in addition, the computationally expensive linear matrix inequalities (LMI) have to be solved to guarantee preservation of passivity. In case of Y or Z-parameter representation, the LMI for positive-real systems are solved, whereas the LMI for bounded-real systems are solved for S-parameter representation [18].

It is a well known problem that, interpolating state-space matrices is not a very robust parameterization method as there is no unique state-space realization for rational models. So, interpolating these state-space models becomes difficult. Several methods, such as barycentric [16] and Sylvester realizations [19] have been suggested to have a smooth realization for the state-space matrices. The downside is that these smooth interpolation schemes are still an ongoing research area. Other methods which first convert the rational models from a pole-residue form to a fractional form with numerator and denominator polynomial and interpolating these coefficients separately also exist [20]. This methods does not depend on a specific state-space realization.

Another issue associated with state-space interpolation methods is the fact that they assume that all the *root macromodels* have the same modeling order so that the matrices can be interpolated. However, this assumption can lead to overfitting of some of the *root macromodels* deteriorating the smoothness of the matrix elements. If a local sequential sampling strategy is used, each of the local regions of the design space (hyperrectangular or simplicial regions) would require that the corner *root macromodels* have the same order. This means that for each local region, the order has to be calculated in advance, hindering the automation process.

2.3.3 Interpolation with amplitude and frequency scaling coefficients

This method is a kind of hybridization of the two methods discussed above. The interpolation is performed here on the transfer function models to have a robust model. But to improve the modeling capability with respect to the simple transfer function interpolation, two additional coefficients, namely an amplitude and a frequency scaling coefficients are introduced [21, 22]. This ensures that the complex behavior of the frequency response with respect to the design parameters is better captured by indirectly parameterizing the poles of the systems. The behavior of the poles is captured in the frequency scaling coefficient and not directly interpolated as in the interpolation of state-space matrices ensuring a robust interpolation. For readability, a brief idea of this method is presented in this section. More information on this macromodeling can be found in Chapter 4.

Starting from the *root macromodels* of (2.2), parametric macromodeling for each and every design region $\Omega = \{\Omega_l;\ l = 1, \ldots, L\}$ is done in the following way. A two parameter description is presented here for clarity and ease of notation, even though the method is general for any dimension N of the design space. The design space region Ω_l is defined by four bounding corners $\vec{g}_1{}^{\Omega_l} = (g_1^1, g_2^1)$, $\vec{g}_2{}^{\Omega_l} = (g_1^2, g_2^1)$, $\vec{g}_3{}^{\Omega_l} = (g_1^1, g_2^2)$, and $\vec{g}_4{}^{\Omega_l} = (g_1^2, g_2^2)$ as in Fig. 2.3. Each corner possesses a different *root macromodels* $\mathbf{R}^{\Omega_l}(s, \vec{g}_k)$, $k = 1, \ldots, 4$. We will discuss the interpolation of the *root macromodels* next. For simplicity and ease of notation we omit the superscript Ω_l. In [21, 22], one amplitude scaling and one frequency

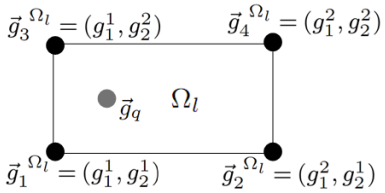

Figure 2.3: A two dimensional design space with four root macromodels.

scaling coefficient (α_1, α_2) are calculated using the optimization

$$(\alpha^*_{1,km}, \alpha^*_{2,km}) = \underset{(\alpha_{1,km}, \alpha_{2,km})}{\operatorname{argmin}} \left[\operatorname{Err}(\tilde{\mathbf{R}}^m(s, \vec{g}_k), \mathbf{H}(s, \vec{g}_m)) \right]. \qquad (2.7)$$

In (2.7), $\tilde{\mathbf{R}}^m(s, \vec{g}_k) = \alpha_{1,km}\mathbf{R}(s\alpha_{2,km}, \vec{g}_k)$, is the scaled response of $\mathbf{R}(s, \vec{g}_k)$ obtained to match $\mathbf{H}(s, \vec{g}_m)$ and $\operatorname{Err}(\cdot)$ is a suitable error measure between the two responses [21, 22]. The evaluation of the model taken at a generic point $\vec{g}_q$ in the design space (Fig. 2.3) is done similarly to [21, 22] as:

i For each *root macromodel* $\mathbf{R}(s, \vec{g}_k)$, $k = 1, \ldots, 2^N$, the amplitude scaling coefficient $\alpha_{1,km}$ and frequency scaling coefficient $\alpha_{2,km}$ are interpolated

using a multilinear interpolation [23] over $\vec{g}$ at the point $\vec{g}_q$ to find $\alpha_{1,kq}$ and $\alpha_{2,kq}$. This results in the modified *root macromodels*, $\tilde{\mathbf{R}}_q(s,\vec{g}_k) = \alpha_{1,kq}\mathbf{R}(s\alpha_{2,kq},\vec{g}_k)$ at $\vec{g}_q$,

ii Then the models $\tilde{\mathbf{R}}_q(s,\vec{g}_k)$ are interpolated using the multilinear interpolation [23] over $\vec{g}$ to get the final model $\mathbf{R}(s,\vec{g}_q)$ at the point $\vec{g}_q$.

This parametric macromodeling approach is performed for each region Ω_l to cover the complete design space.

In Table 2.1, this is given a medium score in its modeling power. Passivity is preserved here with the help of positive/bounded interpolation scheme and putting some constraints on the two scaling coefficients [21, 22] which is relatively cheap to do in comparison with passivity preservation of state-space matrix interpolation.

The modeling power can be further enhanced by defining multiple frequency scaling coefficients as it is proposed in Chapter 5, Section 5.1, to parameterize the movement of different poles independently. Here, the calculated frequency scaling coefficients $\alpha_{2,km}$ are further refined and improved by defining a separate frequency scaling coefficient $\beta_{n,km}$ for every term n in the rational model in a pole-residue form (2.2). The modified $\tilde{\mathbf{R}}^m(s,\vec{g}_k)$ is given by:

$$\tilde{\mathbf{R}}^m(s,\vec{g}_k) = \alpha_{1,km}\sum_{n=1}^{N_P^{\vec{g}_k}}\frac{\mathbf{c}_n^{\vec{g}_k}}{s\beta_{n,km}-a_n^{\vec{g}_k}} + \mathbf{d}^{\vec{g}_k} \tag{2.8}$$

The optimal $\beta_{n,km}, n = 1,\ldots,N_P^{\vec{g}_k}$ are found by performing an optimization step similar to (2.7). The $\alpha^*_{2,km}$ obtained from (2.7) are used as an initial value for all $\beta_{n,km}$. For the complex-conjugate pole pairs (n_1, n_2) the coefficients satisfy $\beta_{n_1,km} = \beta_{n_2,km}$ to preserve the symmetry. This also reduces the number of coefficients to be optimized. However, ensuring passivity can be a difficult task for such models.

As in case of the interpolation of system transfer functions, these methods also suffers from reduction in model compactness in terms of the model order (see Table 2.1).

2.4 Automated and efficient parametric macromodeling

Fig. 2.4 illustrates a complete parametric macromodeling flow with the sequential sampling strategy. The local parametric macromodeling method of Section 2.3.3 is selected as the most suitable candidate for an efficient sequential sampling since it scores well in robustness as well as modeling power with moderate complexity.

As shown in Fig. 2.4, at first, the *initialization of the parametric macromodel* starts in the definition of the design parameter space and performing some EM

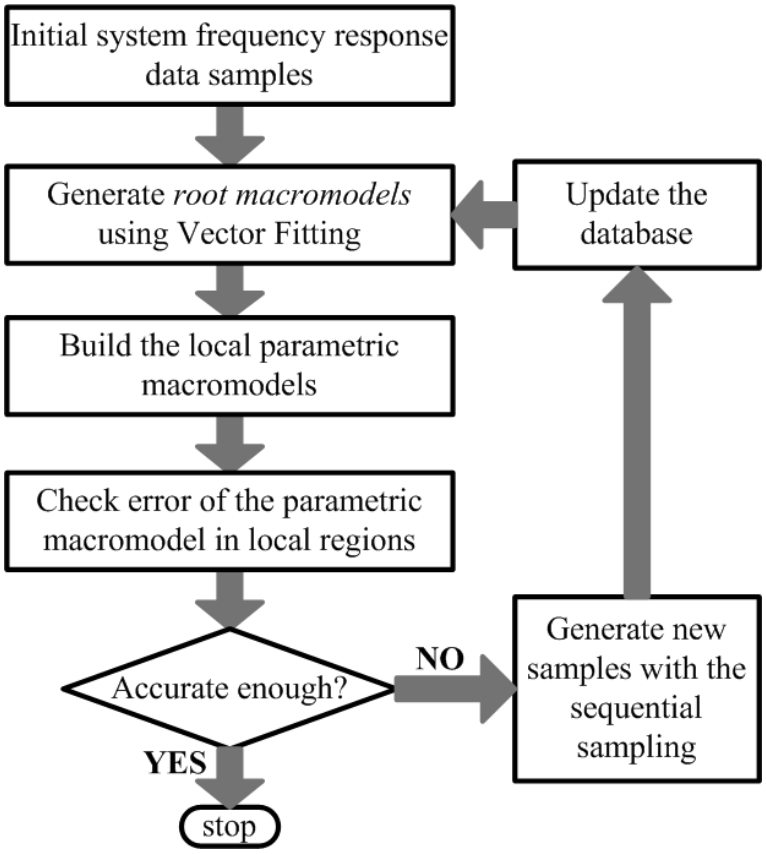

Figure 2.4: Complete flow of the automated parametric macromodeling.

simulations on the corner points which defines the convex hull the design space. Then, *root macromodels* are generated using rational frequency response identification techniques as introduced before. The set of *root macromodels* are then *interpolated* using a local interpolation technique to generate local parametric macromodels. Each of the local design space regions (as shown in Fig. 2.1) are *validated* with EM validation samples. If they are not accurate enough, *new EM samples are generated, the local region is refined and the parametric macromodel is rebuilt* for the inaccurate regions. Each of these stages in the flowchart will be explained in detail during the course of this PhD thesis.

2.5 Conclusisons

This chapter introduces the state-of-the-art of the local parameteric macromodeling as applicable to passive high-frequency electrical systems. The context of this PhD thesis and the motivation for the research work by describing the state-of-the-art local parametric macromodeling techniques is given here. These techniques for building efficient models are discussed here with their corresponding merits and limitations. The chapter also discusses how automation can be introduced into the complete model building process. This serves as the source of inspiration for the rest of this PhD thesis.

References

[1] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," vol. 14, no. 3, pp. 1052–1061, July 1999.

[2] B. Gustavsen, "Improving the pole relocating properties of vector fitting," vol. 21, no. 3, pp. 1587–1592, July 2006.

[3] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 6, pp. 1587–1592, June 2008.

[4] C. K. Sanathanan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Transactions on Automatic Control*, vol. 8, no. 1, pp. 56–58, Jan 1963.

[5] B. Gustavsen, "Fast passivity enforcement for pole-residue models by perturbation of residue matrix eigenvalues," vol. 23, no. 4, pp. 2278–2285, Oct. 2008.

[6] B. Gustavsen and A. Semlyen, "Fast passivity assessment for S-parameter rational models via a half-size test matrix," vol. 56, no. 12, pp. 2701–2708, Dec. 2008.

[7] ——, "A half-size singularity test matrix for fast and reliable passivity assessment of rational models," vol. 24, no. 1, pp. 345–351, Jan. 2009.

[8] B. Gustavsen, "Fast passivity enforcement for S-parameter models by perturbation of residue matrix eigenvalues," vol. 33, no. 1, pp. 257–265, Feb. 2010.

[9] D. Deschrijver and T. Dhaene, "Fast passivity enforcement of S-parameter macromodels by pole perturbation," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 3, pp. 620–626, March 2009.

[10] T. Dhaene, D. Deschrijver, and N. Stevens, "Efficient algorithm for passivity enforcement of S-parameter-based macromodels," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 2, pp. 415–420, Feb 2009.

[11] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *Microwave and Wireless Components Letters, IEEE*, vol. 19, no. 10, pp. 608 –610, Oct. 2009.

[12] ——, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623 –629, Aug. 2010.

[13] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[14] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *2010 IEEE 14th Workshop on Signal Propagation on Interconnects (SPI)*, pp. 117 –120, May 2010.

[15] ——, "Passive parametric modeling of interconnects and packaging components from sampled impedance, admittance or scattering data," in *3rd Electronic System-Integration Technology Conference (ESTC)*, sept. Sept. 2010, pp. 1 –6.

[16] F. Ferranti, L. Knockaert, T. Dhaene, G. Antonini, and D. De Zutter, "Parametric macromodeling for tabulated data based on internal passivity," *IEEE Microwave and Wireless Components Letters*, vol. 20, no. 10, pp. 533 –535, Oct. 2010.

[17] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling for S-parameter data based on internal nonexpansivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 26, no. 1, pp. 15–27, 2013.

[18] L. Knockaert, T. Dhaene, F. Ferranti, and D. De Zutter, "Model order reduction with preservation of passivity, non-expansivity and markov moments," *Systems & Control Letters*, vol. 60, no. 1, pp. 53–61, 2011.

[19] E. R. Samuel, L. Knockaert, F. Ferranti, and T. Dhaene, "Guaranteed passive parameterized macromodeling by using sylvester state-space realizations," *IEEE Transactions on Microwave Theory Techniques*, vol. 61, pp. 1444–1454, 2013.

[20] D. Deschrijver and T. Dhaene, "Fully parameterized macromodeling of S-parameter data by interpolation of numerator & denominator," *IEEE Microwave and Wireless Components Letters*, vol. 22, no. 6, pp. 309–311, June 2012.

[21] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2394 –2403, Oct. 2011.

[22] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed

passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139–151, 2012.

[23] A. Weiser and S. E. Zarantonello, "A note on piecewise linear and multilinear table interpolation in many dimensions," *Mathematics of Computation*, vol. 50, no. 181, pp. 189–196, Jan. 1988.

3

Parametric Macromodeling for Sensitivity Responses from Tabulated Data

K. Chemmangat, F. Ferranti, L. Knockaert and T. Dhaene

Published in IEEE Microwave and Wireless Components Letters, vol. 21, no. 8, pp. 397-399, 2011

Published in International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, vol. 25, no. 4, pp. 347-361, 2012

⋆ ⋆ ⋆

In this chapter, we extend the state-of-the-art parametric macromodeling technique which accurately describes the parameterized frequency behavior of electromagnetic systems to also model their corresponding parameterized sensitivity responses with respect to the design parameters. The technique is based on the interpolation of a set of state-space matrices with a proper choice of the interpolation scheme, so that parametric sensitivity macromodels can be computed. These parametric macromodels, along with the corresponding parametric sensitivity macromodels, can be used in a gradient-based design optimization process.

The numerical example section is divided into two parts. In the first part, two analytical examples are used to check the accuracy of the generated parametric sensitivity macromodels. In the second part, the parameterized sensitivity information is used in an efficient and accurate gradient-based design optimization for two numerical microwave filter examples.

3.1 Introduction

When designing high-speed microwave systems, one aims at obtaining the optimal values of the design parameters for which the system responses satisfy the design specifications. This process is usually carried out through electromagnetic (EM) simulations. Optimal values of the design parameters are often determined using optimization algorithms (optimizers) which drive the EM simulator to obtain the responses and their sensitivities in consecutive optimization iteration. Unfortunately, multiple consecutive EM simulations are often computationally expensive. An alternative approach is to generate accurate parametric macromodels up first, which capture the parameterized frequency behavior of the EM systems and their corresponding parameterized sensitivity responses with respect to design parameters, such as layout and substrate parameters. Efficient and accurate parametric sensitivity information is required by optimizers which employ state-of-the-art gradient-based techniques to calculate the optimal design parameters. Parametric sensitivity macromodels are able to describe sensitivity responses not only in the vicinity of a single operating point (local sensitivity), but over the entire design space of interest.

One of the most common approaches to calculate local sensitivities is the adjoint variable method. The main attractiveness of this approach is that sensitivity information can be obtained from at most two systems analyses regardless of the number of designable parameters [1–3]. However, these methods involve the calculation of the system matrix derivatives with respect to each design parameters, which is mostly done by finite difference approximations.

Recently, some interpolation-based parametric macromodeling techniques have been developed [4–9], which interpolate a set of initial univariate macromodels, called *root macromodels*. In [4–6], a parametric macromodel is built by inter-

polating a set of *root macromodels* at an input-output level, while in [7–9] the interpolation process is applied to the internal state-space matrices of the *root macromodels*, therefore at a deeper level than in the transfer function-based interpolation approaches [4–6]. The methods [7–9] allow to parameterize both poles and residues, hence their modeling capability is increased with respect to [4–6], where only residues are parameterized.

A parametric macromodeling method which apart from building an accurate model of the responses but also calculate the response sensitivities with respect to several design parameter is presented in this chapter. As in [7–9], an interpolation process on the internal state-space matrices of the *root macromodels* is performed. However, in [7–9], the focus is on parametric macromodeling which ensures stability and passivity over the design space of interest. This is not strictly necessary for the calculation of parametric sensitivities, which allows the use of more powerful interpolation schemes which are at least continuously differentiable. The parametric sensitivity macromodels avoid the use of finite difference approximations in the optimization process. Also, in [7–9] computationally expensive linear matrix inequalities (LMI) are solved to guarantee preservation of passivity, which can be avoided in the present work. The proposed parametric sensitivity macromodeling method is first verified using two analytical examples so that a direct comparison of the sensitivity information is available. as a second part of the numerical section, the parametric macromodels along with the corresponding parametric sensitivities are used in two pertinent gradient-based design optimization examples of microwave filters which confirm the applicability of the proposed technique. The importance of parameterized sensitivity information to speed up the design optimization process is shown in these examples.

3.2 Generation of *root macromodels*

Starting from a set of data samples $\{(s_i, \vec{g}_k), \mathbf{H}(s_i, \vec{g}_k),\ i = 1, \ldots, N_s,\ k = 1, \ldots, K_{tot}\}$ which depend on the complex frequency $s = j\omega$ and several design parameters $\vec{g} = (g^{(1)}, g^{(2)}, \ldots, g^{(N)})$, such as layout features or substrate parameters, a set of frequency-dependent rational macromodels is built for some design space points by means of the Vector Fitting (VF) technique [10–12]. Each *root macromodel* has the following form:

$$\mathbf{R}_{\vec{g}_k}(s) = \sum_{n=1}^{N_P} \frac{\mathbf{c}_n^{\vec{g}_k}}{s - a_n^{\vec{g}_k}} + \mathbf{d}^{\vec{g}_k} \tag{3.1}$$

The terms in the rational model (3.1), $a_n^{\vec{g}_k}$, $\mathbf{c}_n^{\vec{g}_k}$ and $\mathbf{d}^{\vec{g}_k}$ represent poles, residues and feed forward terms respectively at the design point $\vec{g}_k = (g_k^{(1)}, ..., g_k^{(N)})$ as explained before in Chapter 2. This initial step of the proposed method results

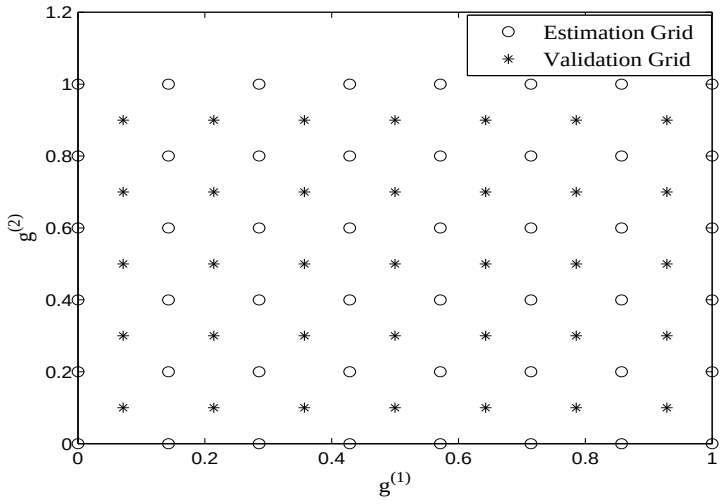

Figure 3.1: Estimation and Validation grids for a general two parameter design space.

in a set of stable and passive frequency dependent rational models, called *root macromodels*.

Two design space grids are used in the modeling process: an estimation grid and a validation grid. The first grid is utilized to build the parametric macromodels. The second grid is utilized to assess the capability of parametric macromodels of describing the system under study in a set of points of the design space previously not used for their construction. To clarify the use of these two design space grids, we show in Fig. 3.1 a possible estimation and validation design space grid in the case of two design parameters $\vec{g} = (g^{(1)}, g^{(2)})$. A *root macromodel* is built for each estimation grid point in the design space. This set of *root macromodels* is interpolated, as explained in the following section, to build a parametric macromodel that is evaluated and compared with original data related to the validation design space points.

3.3 Parametric macromodeling

Each *root macromodel* $\mathbf{R}_{\vec{g}_k}(s)$, corresponding to a specific design space point $\vec{g}_k = (g_k^{(1)}, ..., g_k^{(N)})$, is converted from the pole-residue form (3.1) into a state-space form:

$$\mathbf{R}_{\vec{g}_k}(s) = \mathbf{C}_{\vec{g}_k}(s\mathbf{I} - \mathbf{A}_{\vec{g}_k})^{-1}\mathbf{B}_{\vec{g}_k} + \mathbf{D}_{\vec{g}_k} \tag{3.2}$$

Then, this set of state-space matrices $\mathbf{A}_{\vec{g}_k}$, $\mathbf{B}_{\vec{g}_k}$, $\mathbf{C}_{\vec{g}_k}$, $\mathbf{D}_{\vec{g}_k}$ is interpolated entry-by-entry and the multivariate models $\mathbf{A}(\vec{g}), \mathbf{B}(\vec{g}), \mathbf{C}(\vec{g}), \mathbf{D}(\vec{g})$ are built using multivariate interpolation schemes to generate a parametric macromodel $\mathbf{R}(s, \vec{g})$ for the entire design space [7, 8]:

$$\mathbf{R}(s, \vec{g}) = \mathbf{C}(\vec{g})(s\mathbf{I} - \mathbf{A}(\vec{g}))^{-1}\mathbf{B}(\vec{g}) + \mathbf{D}(\vec{g}). \tag{3.3}$$

The computationally cheap piecewise linear interpolation can not be used to generate parametric sensitivity macromodels, since it is not continuously differentiable. A proper choice of interpolation schemes which are at least continuously differentiable is necessary. In this work, three interpolation methods are investigated, namely the cubic spline (CS) interpolation, the piecewise cubic Hermite interpolation (PCHIP) and the shape preserving C2 cubic spline interpolation (SPC2). They are briefly described in what follows.

3.3.1 Cubic Spline Interpolation

Given some data samples $(x_i, y_i)_{i=1}^n$, the CS interpolation method builds a cubic polynomial for each interval of the dataset $x_i \leq x \leq x_{i+1}$, $i = 1, \ldots, n$.

$$s_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i \tag{3.4}$$

The coefficients of the cubic polynomials are obtained by imposing the first and second order derivative continuity at each data point along with a $not\text{-}a\text{-}knot$ end condition, and then solving a tridiagonal linear system [13]. Once these coefficients are computed, the derivatives of the overall spline interpolation function can be analytically calculated in terms of its coefficients a_i, b_i and c_i for $x_i \leq x < x_{i+1}$, $i = 1, 2, ..., n-1$. If the data under interpolation is in a matrix form, each entry of the matrices is independently interpolated.

The univariate CS interpolation can be extended to higher dimensions by means of a tensor product implementation [13].

3.3.2 Piecewise Cubic Hermite Interpolation

The PCHIP method is a monotonic shape preserving interpolation scheme. As in the CS interpolation, each data interval is modeled by a cubic polynomial similar to (3.4):

$$p_i(x) = f_i H_1(x) + f_{i+1} H_2(x) + d_i H_3(x) + d_{i+1} H_4(x), \tag{3.5}$$

where $d_j = \frac{dp(x_j)}{dx}$, $j = i, i+1$, and the $H_k(x)$ are the usual cubic Hermite basis functions for the interval $x_i \leq x < x_{i+1}$, $i = 1, 2, ..., n-1$: $H_1(x) = \phi((x_{i+1} - x)/h_i)$, $H_2(x) = \phi((x - x_i)/h_i)$, $H_3(x) = -h_i\psi((x_{i+1} - x)/h_i)$, $H_4(x) = h_i\psi((x - x_i)/h_i)$, where $h_i = x_{i+1} - x_i$, $\phi(t) = 3t^2 - 2t^3$, $\psi(t) =$

$t^3 - t^2$. The first order derivative d_i at each data point x_i is calculated such that the local monotonicity is preserved [14]. An extension to higher dimension can be performed by a tensor product implementation [13]. The calculation of derivatives is done in the same way as in the CS interpolation case. This interpolation scheme works better for non-smooth datasets, wherein the CS scheme could result in overshoots or oscillatory behavior of the derivatives. However, the PCHIP method is only continuous in first derivatives, which affects the smoothness of the derivatives [14].

3.3.3 Shape Preserving C2 Cubic Spline Interpolation

The SPC2 interpolation is a monotonicity preserving interpolation scheme similar to PCHIP. However, in contrast to the PCHIP method which is only continuous in first derivative, the SPC2 method is a second order derivative continuous interpolation scheme. The idea here is to add two extra break points in each subinterval of the data, such that enough degrees of freedom are generated to construct a cubic spline interpolant, which is globally second order derivative continuous [15]. Since the monotonicity of the data is preserved, this scheme works better with respect to the CS method for non-smooth data sets. Similar to the CS and the PCHIP interpolation schemes, a multivariate SPC2 interpolation is performed using a tensor product implementation [13].

3.4 Parametric Sensitivity Macromodels

Once the parametric macromodel $\mathbf{R}(s, \vec{g})$ is built, the corresponding sensitivities can be computed by differentiating (3.3) with respect to the design parameters $\vec{g}$

$$\frac{\partial}{\partial \vec{g}}\mathbf{R}(s,\vec{g}) = \frac{\partial \mathbf{C}(\vec{g})}{\partial \vec{g}}(s\mathbf{I}-\mathbf{A}(\vec{g}))^{-1}\mathbf{B}(\vec{g}) + \mathbf{C}(\vec{g})(s\mathbf{I}-\mathbf{A}(\vec{g}))^{-1}\frac{\partial \mathbf{A}(\vec{g})}{\partial \vec{g}} \times (s\mathbf{I}-\mathbf{A}(\vec{g}))^{-1}\mathbf{B}(\vec{g}) + \mathbf{C}(\vec{g})(s\mathbf{I}-\mathbf{A}(\vec{g}))^{-1}\frac{\partial \mathbf{B}(\vec{g})}{\partial \vec{g}} + \frac{\partial \mathbf{D}(\vec{g})}{\partial \vec{g}} \quad (3.6)$$

In (3.6), $\frac{\partial}{\partial \vec{g}}\mathbf{R}(s, \vec{g})$ is based on the parameterized state-space matrices $\mathbf{A}(\vec{g})$, $\mathbf{B}(\vec{g})$, $\mathbf{C}(\vec{g})$, $\mathbf{D}(\vec{g})$ and the corresponding derivatives, which are computed efficiently and analytically using the interpolation methods described in Section 3.3.

3.5 Gradient-based Minimax Optimization

Parametric sensitivity macromodels can be used in the optimization process of electromagnetic systems. Considering microwave filters, a typical optimization process begins by defining passband and stopband specifications in terms of the

frequency responses, which are reformulated in the form of a cost function $F_i(\vec{g})$, at optimization frequency samples s_i, $i = 1, 2, ..N_s$ to be minimized:

$$F_i(\vec{g}) = R^i_{\mathrm{L}} - R(s_i, \vec{g}) \;\text{ or }\; R(s_i, \vec{g}) - R^i_{\mathrm{U}}, \;\; i = 1, 2, ..., N_S. \tag{3.7}$$

In (3.7), R^i_{L} and R^i_{U} represent lower and upper frequency response thresholds, respectively, at frequency samples s_i spread over the frequency range of interest. A negative cost indicates that the corresponding specification is satisfied, while a positive cost denotes that the specification is violated. The minimization (3.7) can be performed by several state-of-the-art optimization algorithms. In this chapter, we use a minimax optimization algorithm [16] which uses the cost function (3.7) and its gradients with respect to design parameter $\vec{g}$ giving the optimum design parameters $\vec{g^*}$ as,

$$\vec{g^*} = \underset{\vec{g}}{\operatorname{argmin}}\{\max_i[F_i(\vec{g})]\} \tag{3.8}$$

The complete optimization process starting from the proposed parametric macromodeling technique is depicted in Fig. 3.2.

3.6 Numerical Examples

3.6.1 Analytical examples

3.6.1.1 Coaxial cable

An open-ended coaxial cable is modeled with cross section shown in Fig. 3.3. The relative permittivity ϵ_{rel} of the dielectric is chosen equal to 2.5. The impedance matrix $Z(s, a, L)$ of the model is calculated analytically [17] as a function of the radius of the inner conductor a and the length L, in addition to frequency, on a grid of $150 \times 15 \times 15$ samples $(freq, a, L)$. The corresponding ranges of these parameters are shown in Table 3.1.

Parameter	Min	Max
Frequency ($freq$)	10 KHz	2 GHz
Inner radius (a)	2 mm	3 mm
Length (L)	100 mm	110 mm

Table 3.1: Coaxial cable: Design parameters

A set of stable and passive *root macromodels* has been built for 8 values of a and 8 values of L using VF. The remaining data are used for validation. The number of poles N_P for each *root macromodel* is 18, selected using an error-based bottom up approach (see Appendix A, section A.2 for more details). Each *root macromodel* has been converted to state-space form and the state-space matrices

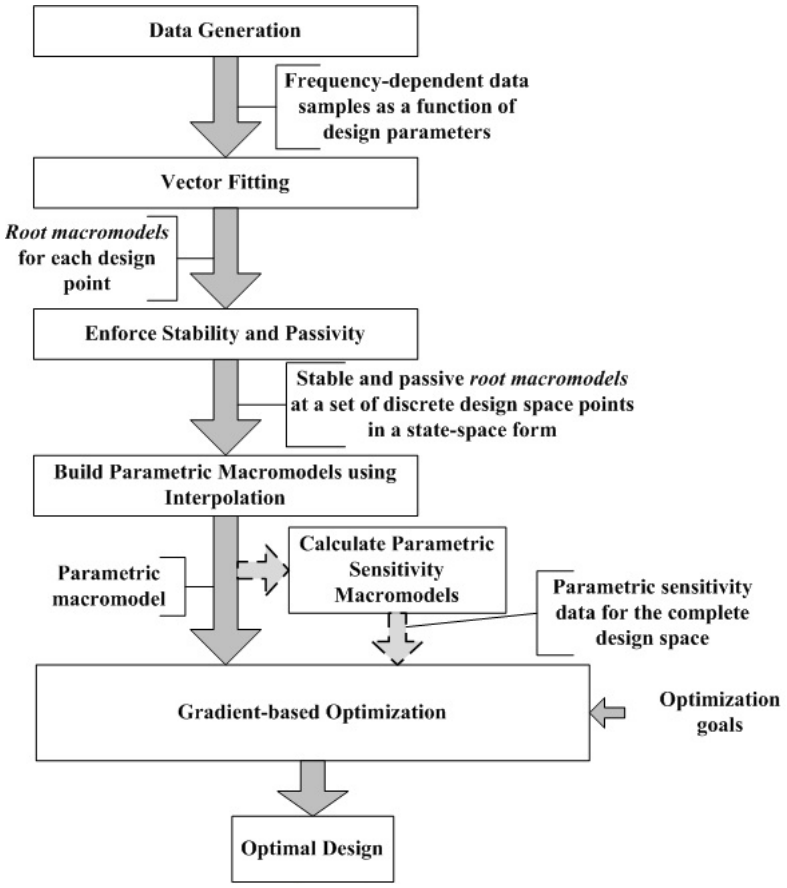

Figure 3.2: Complete optimization process flowchart.

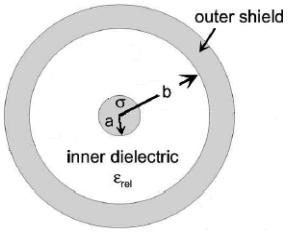

Figure 3.3: Cross section of the coaxial cable.

have been interpolated by the CS and PCHIP interpolation methods. Next, the parametric sensitivity of $Z(s, a, L)$ with respect to a and L has been computed by means of the derivatives (3.6) of the trivariate macromodel and the analytical model computed from [17]. The accuracy of the model and its derivatives for the two interpolation methods are measured in terms of the weighted rms-error defined as:

$$\mathrm{E}_{\mathrm{RMS}}(\vec{g}) = \sqrt{\frac{\sum_{i=1}^{P^2}\sum_{k=1}^{K_s} |w_{Z_i}(s,\vec{g})(R_i(s_k,\vec{g}) - Z_i(s_k,\vec{g}))|^2}{P^2 K_s}}. \tag{3.9}$$

In (3.9) P is the number of ports, K_s is the number of frequency samples and $w_{Z_i} = |Z_i(s_k, \vec{g})|^{-1}$ is the weighting function for the error. The worst case rms-error over the validation grid is chosen to assess the accuracy and the quality of the parametric sensitivity macromodels

$$\mathrm{E}_{\mathrm{RMS}}^{\mathrm{Max}} = \max_{\vec{g}} \mathrm{E}_{\mathrm{RMS}}(\vec{g}),\ \vec{g} \in \text{validation grid} \tag{3.10}$$

The maximum weighted rms-error calculated from (3.10) for the model and its sensitivities is tabulated in Table 3.2. Fig. 3.4 shows the magnitude of Z_{11}

Quantity	Method	
	CS	PCHIP
$\mathbf{Z}(s,a,L)$	0.0054	0.0054
$\frac{\partial \mathbf{Z}(s,a,L)}{\partial a}$	0.0061	0.0325
$\frac{\partial \mathbf{Z}(s,a,L)}{\partial L}$	0.0119	0.0194

Table 3.2: Coaxial cable: Modeling accuracy of the proposed method

as a function of frequency and a for $L = 105\ mm$, while Fig. 3.5 shows the magnitude of the corresponding parametric sensitivity $\frac{\partial Z_{11}}{\partial a}$ obtained by the CS scheme. Fig. 3.6 shows the magnitude of Z_{12} as a function of frequency and L for $a = 2.5\ mm$, while Fig. 3.7 shows the magnitude of the corresponding parametric

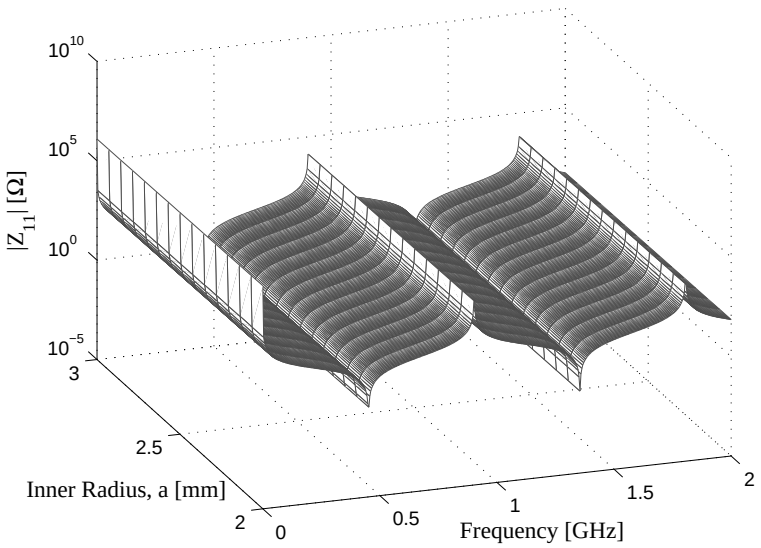

Figure 3.4: Coaxial cable: Magnitude of Z_{11} for $L = 105\ mm$.

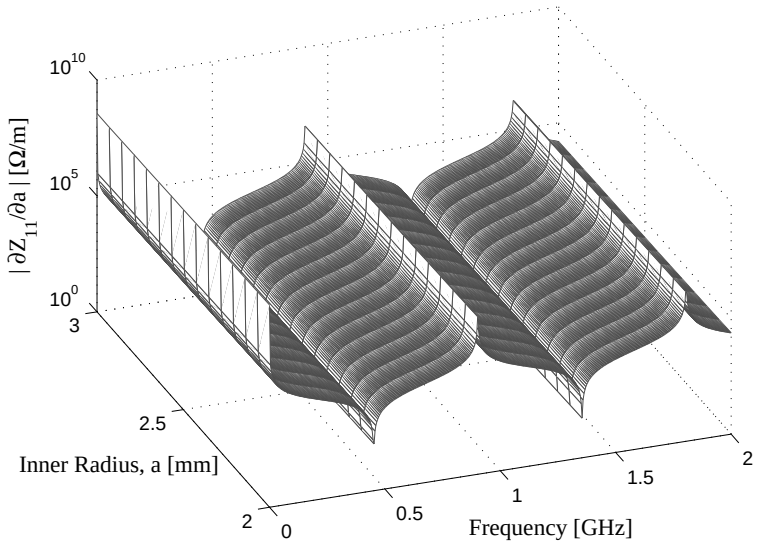

Figure 3.5: Coaxial cable: Magnitude of $\frac{\partial Z_{11}}{\partial a}$ (CS) for $L = 105\ mm$.

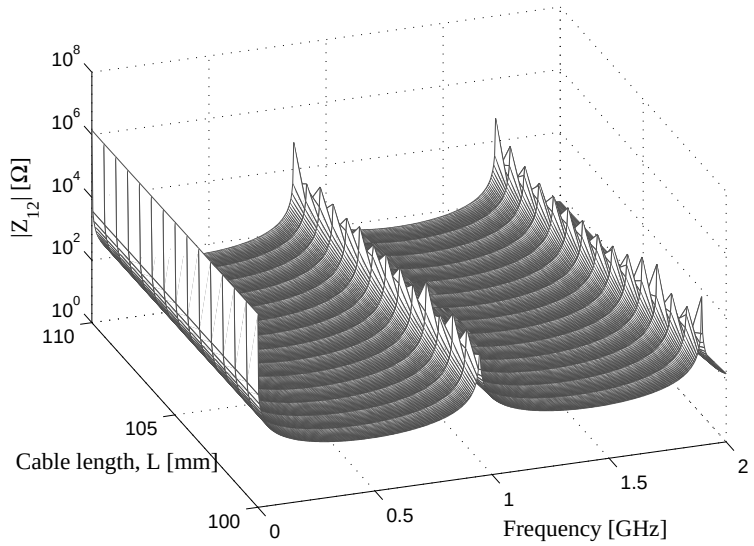

Figure 3.6: Coaxial cable: Magnitude of Z_{12} for $a = 2.5\ mm$.

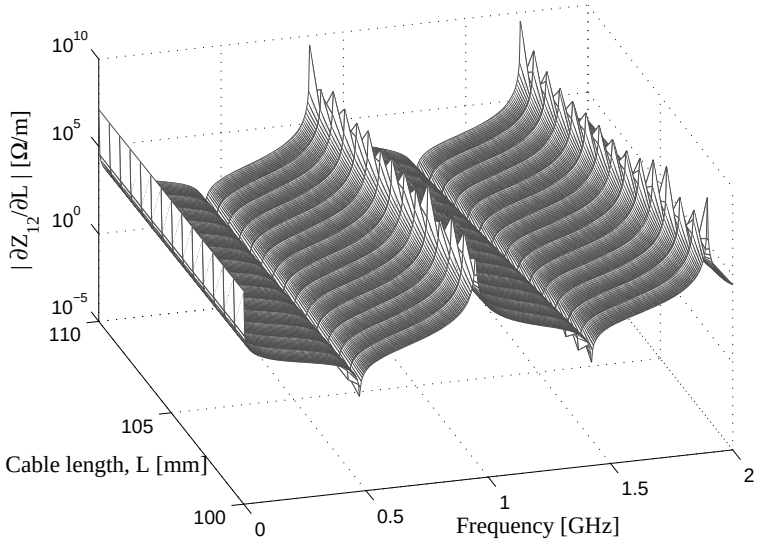

Figure 3.7: Coaxial cable: Magnitude of $\frac{\partial Z_{12}}{\partial L}$ (CS) for $a = 2.5\ mm$.

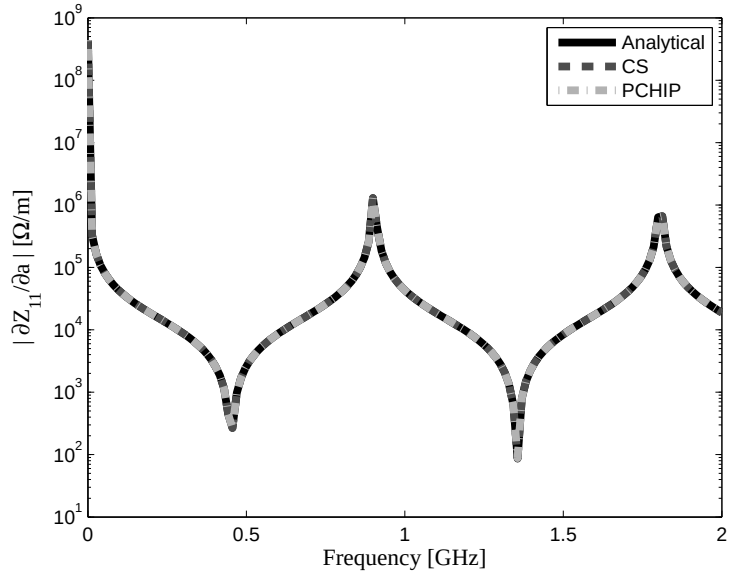

Figure 3.8: Coaxial cable: Magnitude of $\frac{\partial Z_{11}}{\partial a}$ for $a = 2.5\ mm$ and $L = 105\ mm$.

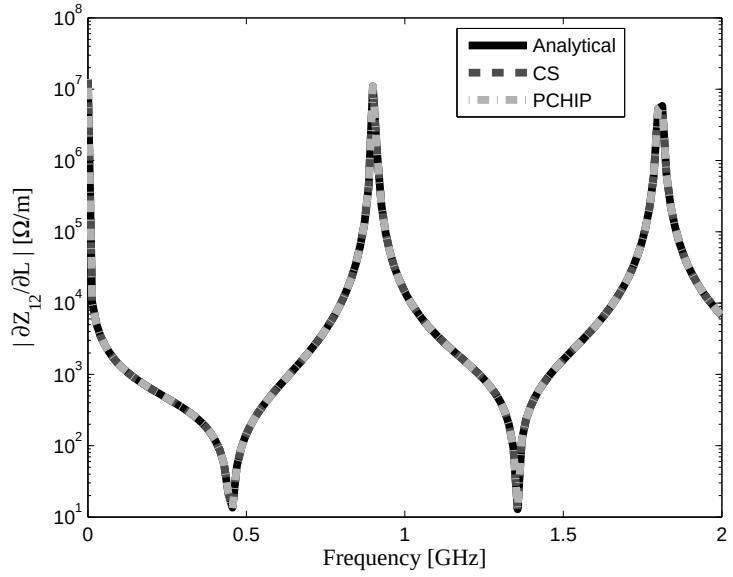

Figure 3.9: Coaxial cable: Magnitude of $\frac{\partial Z_{12}}{\partial L}$ for $a = 2.5\ mm$ and $L = 105\ mm$.

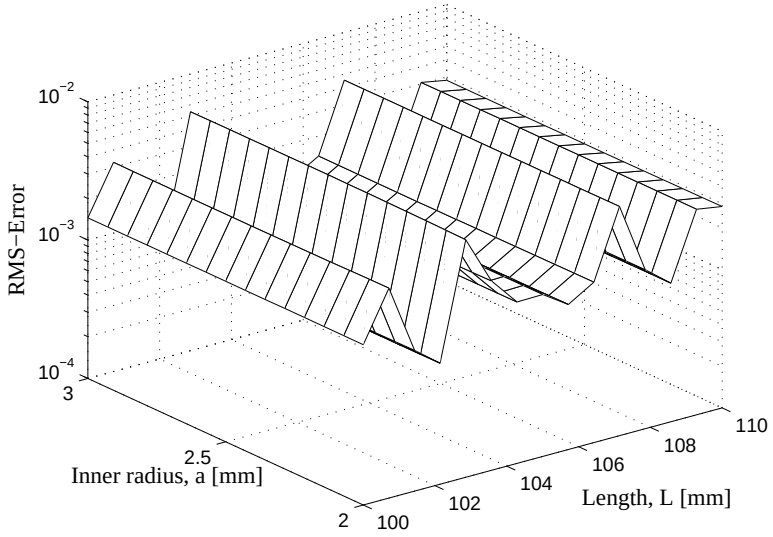

Figure 3.10: Coaxial cable: RMS-Error of Z for the entire design space (CS).

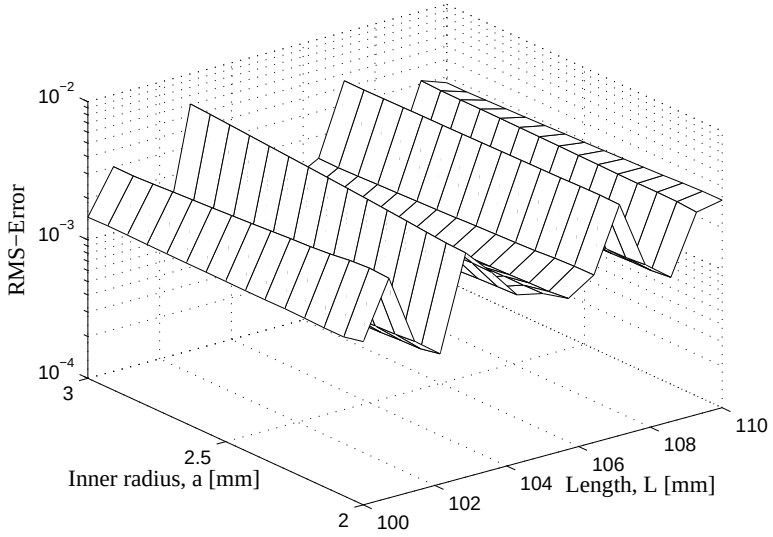

Figure 3.11: Coaxial cable: RMS-Error of $\frac{\partial Z}{\partial L}$ for the entire design space (CS).

sensitivity $\frac{\partial Z_{12}}{\partial L}$ obtained by the CS scheme. Fig. 3.8 compares the magnitude of $\frac{\partial Z_{11}}{\partial a}$ obtained by the analytical model, and the CS and PCHIP methods for the values $a = 2.5\ mm$ and $L = 105\ mm$, which have not been used for the generation of the root macromodels. Fig. 3.9 shows the magnitude of $\frac{\partial Z_{12}}{\partial L}$ for the same values of a and L. A very good agreement between the methods can be observed.

Fig. 3.10 shows the rms-error distribution of the parametric macromodel using the CS interpolation scheme with respect to the analytical model of [17] over the complete design space. Fig. 3.11 shows the rms-error distribution of the parametric sensitivity macromodel of $\frac{\partial \mathbf{Z}}{\partial L}$ with respect to the analytical model. Similar results were obtained for other cases in Table 3.2. We note that a good accuracy is achieved by both interpolation methods, but the CS scheme leads to a lower average error, probably due to the continuity of the second derivatives.

3.6.1.2 Microstrip

In this example, a microstrip with a length of 2 cm is modeled. Its cross section is shown in Fig. 3.12. The relative permittivity of the substrate is equal to $\epsilon_r = 4.1$. A trivariate macromodel is built as a function of the width W of the strip and the height h of the substrate in addition to frequency $freq$. Their corresponding ranges are shown in Table 3.3.

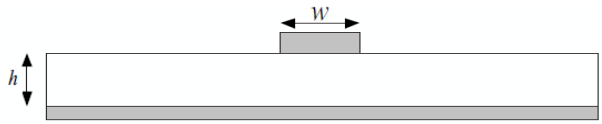

Figure 3.12: Cross section of the microstrip.

Parameter	Min	Max
Frequency ($freq$)	1 GHz	5 GHz
Width (W)	200 μm	300 μm
Height (h)	400 μm	500 μm

Table 3.3: Microstrip: Design parameters

The two-port open-circuit impedance parameter matrix $\mathbf{Z}(s, W, h)$ has been computed by means of the analytical quasi-TEM model presented in [18] on a grid of $150 \times 15 \times 15$ samples $(freq, W, h)$. The accuracy of the model $\mathbf{R}(s, \vec{g})$ and its derivatives with respect to the original analytical quasi-TEM model $\mathbf{Z}(s, \vec{g})$ for

the two interpolation methods is measured in terms of the relative error defined as:

$$\mathrm{E_{rel}} = \left| \frac{\mathbf{R}(s,\vec{g}) - \mathbf{Z}(s,\vec{g})}{\mathbf{Z}(s,\vec{g})} \right| ;\ \vec{g} = (W,h) \in \text{validation grid.} \tag{3.11}$$

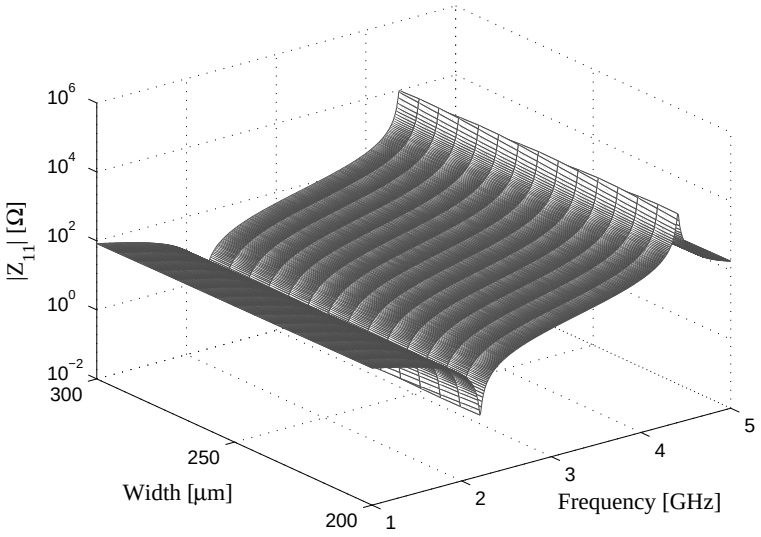

Figure 3.13: Microstrip: Magnitude of Z_{11} for $h = 450\ \mu m$.

A set of stable and passive *root macromodels* has been built for 8 values of W and 8 values of h using VF, and 14 poles were selected for the *root macromodels* using an error-based bottom up approach. The remaining data are used for validation. Each *root macromodel* has been converted to a state-space form (3.2) and the state-space matrices have been interpolated using the CS and PCHIP interpolation methods. The maximum relative error (3.11) of the parametric macromodel of $\mathbf{Z}(s,W,h)$ is -62.23 dB and -57.78 dB, respectively, using the CS and PCHIP schemes. Then, the parametric sensitivities of $\mathbf{Z}(s,W,h)$ with respect to W and h has been computed by means of the derivatives (3.6) of the trivariate macromodels and the analytical quasi-TEM model. Figs. 3.13-3.14 are plotted to show the parameterization of the $\mathbf{Z}$ and $\frac{\partial \mathbf{Z}}{\partial W}$ matrices. Fig. 3.13 shows the parametric behavior of the magnitude of the $(1,1)$ entry of the $\mathbf{Z}(s,W,h)$ matrix (Z_{11}) as a function of frequency and W for $h = 450\ \mu m$, while Fig. 3.14 shows the magnitude of the corresponding parametric sensitivity $\frac{\partial Z_{11}}{\partial W}$ obtained by the CS scheme. In order to visualize the modeling capability of the proposed method, the Fig. 3.15 shows sensitivity $\frac{\partial Z_{11}}{\partial W}$ obtained by the analytical quasi-TEM model, the CS and PCHIP

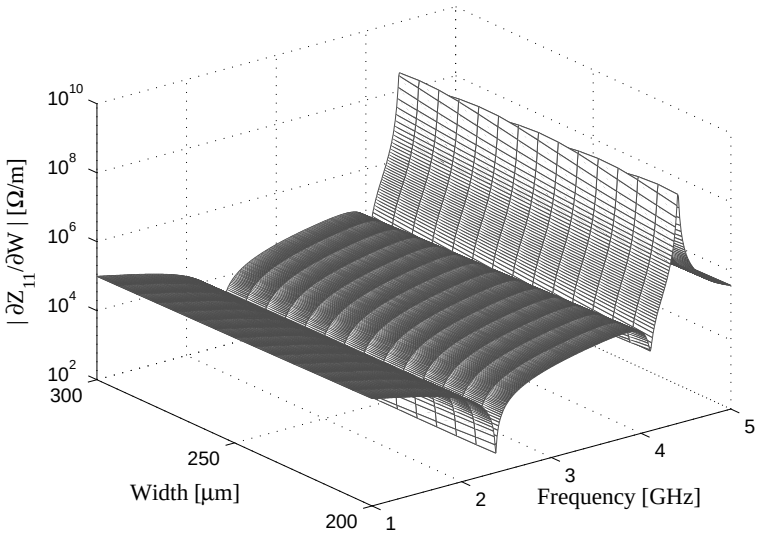

Figure 3.14: Microstrip: Magnitude of $\frac{\partial Z_{11}}{\partial W}$ for $h = 450\ \mu m$.

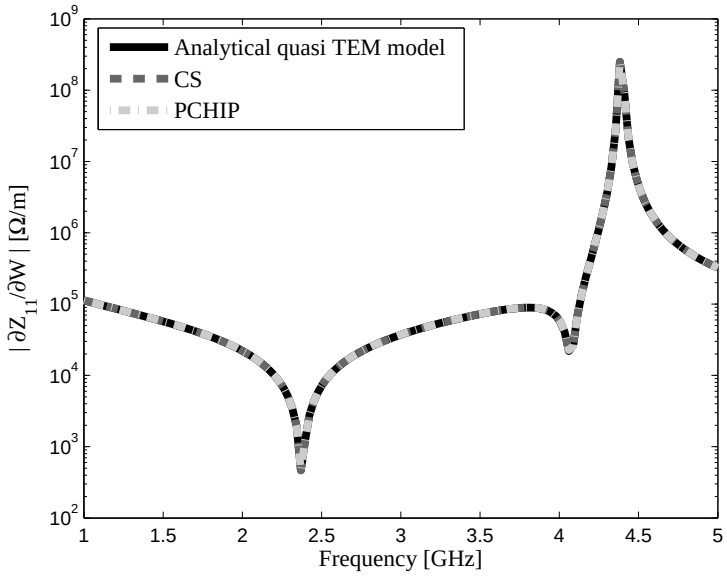

Figure 3.15: Microstrip: Magnitude of $\frac{\partial Z_{11}}{\partial W}$ for $W = 250\ \mu m$ and $h = 450\ \mu m$.

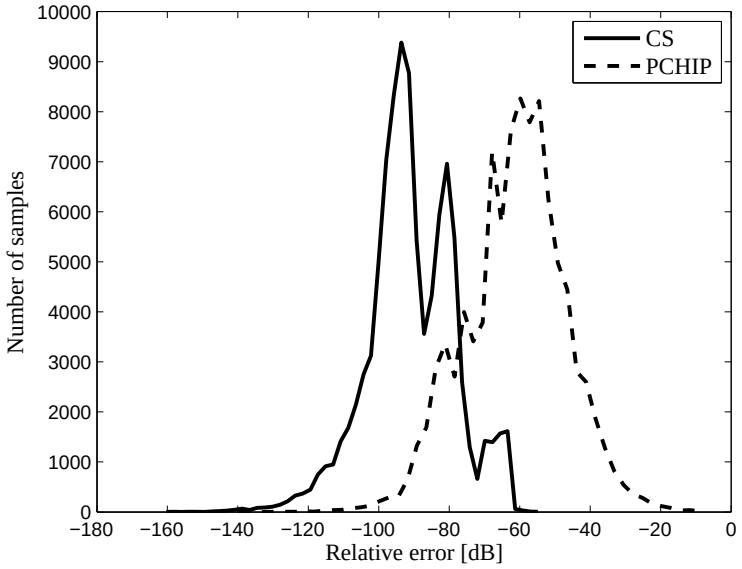

Figure 3.16: Microstrip: Error distribution histogram for $\frac{\partial \mathbf{Z}}{\partial W}$.

methods as a function of frequency for the values $W = 250\ \mu m$ and $h = 450\ \mu m$, which have not been used for the generation of the *root macromodels*. A very good agreement between the methods can be observed.

Fig. 3.16 shows the relative error distribution (3.11) of the parametric sensitivity macromodel $\frac{\partial \mathbf{Z}}{\partial W}$ over the grid of $150 \times 15 \times 15$ samples $(freq, W, h)$. Similar results are obtained for $\frac{\partial \mathbf{Z}}{\partial h}$. We note that a good accuracy is achieved by both interpolation methods, but the CS scheme leads to a lower average error due to the continuity of the second derivative. However, in cases where the interpolation is performed on nonsmooth data sets, the CS scheme may result in oscillations of the derivatives. In those cases, PCHIP will result in a better accuracy.

3.6.2 Gradient-based optimization of Microwave Filters

3.6.2.1 Double folded stub microwave filter

A Double Folded Stub (DFS) microwave filter on a substrate with relative permitivity $\epsilon_r = 9.9$ and a thickness of $0.127\ mm$ is modeled in this example. The layout of this DFS filter is shown in Fig. 3.17. The spacing S between the stubs and the length L of the stub are chosen as design parameters in addition to frequency. Their corresponding ranges are shown in Table 3.4. The design specifications of

this band-stop filter are given in terms of the scattering parameter, similarly to [19],

$$|S_{21}| \geq -3\,\text{dB} \quad \text{for} \quad freq \leq 9\,\text{GHz or } freq \geq 17\,\text{GHz} \tag{3.12a}$$

$$|S_{21}| \leq -30\,\text{dB} \quad \text{for} \quad 12\,\text{GHz} \leq freq \leq 14\,\text{GHz} \tag{3.12b}$$

From the design specifications (3.12), a cost function (3.7) is formulated in terms of S_{21} and $\vec{g} = (S, L)$.

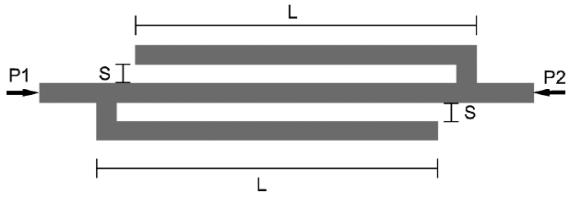

Figure 3.17: Layout of the DFS band-stop filter.

Parameter	Min	Max
Frequency ($freq$)	5 GHz	20 GHz
Spacing (S)	0.1 mm	0.25 mm
Length (L)	2.0 mm	3.0 mm

Table 3.4: DFS Filter: Design parameters

The scattering matrix $\mathbf{S}(s, S, L)$ has been computed using the ADS Momentum[1] software. The number of frequency samples were chosen to be equal to 31. The estimation and validation grid points for the design parameters are shown in Fig. 3.18. The average simulation time for each design space point (S, L) has been found to be equal to, $T_{SimAvg} = 32.87$ sec. A set of stable and passive *root macromodels* has been built for all design space points in the estimation grid of Fig. 3.18 by means of VF with a fixed number of poles $N_P = 18$, selected using an error-based bottom-up approach. Each *root macromodel* has been converted into a state-space form and the set of state-space matrices has been interpolated using the CS, PCHIP and SPC2 methods. Let us define the absolute error

$$Err(\vec{g}) = \max_{i,j,k} \Big(\Big| \Big(R_{i,j}(s_k, \vec{g}) - H_{i,j}(s_k, \vec{g}) \Big| \Big) \tag{3.13}$$

$$i = 1, \ldots, P_{in},\ j = 1, \ldots, P_{out},\ k = 1, \ldots, N_s$$

[1] Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

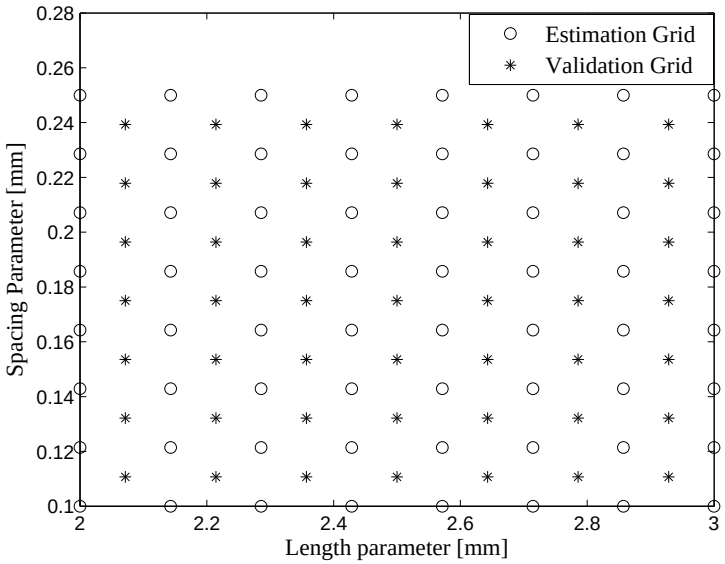

Figure 3.18: DFS Filter: Estimation and Validation grids for the parametric macromodeling.

where P_{in} and P_{out} are the number of inputs and outputs of the system, respectively, and N_s is equal to the number of frequency samples. The worst case absolute error over the validation grid is chosen to assess the accuracy and the quality of parametric macromodels

$$\vec{g}_{max} = \underset{\vec{g}}{\operatorname{argmax}} \left[Err(\vec{g}) \right], \ \vec{g} \in validation\ grid \tag{3.14}$$

$$Err_{max} = Err(\vec{g}_{max}) \tag{3.15}$$

The maximum absolute error (3.15) for the parametric macromodel over the validation grid of Fig. 3.18 is -58.45 dB, -50.23 dB and -50.23 dB, respectively using the CS, PCHIP and SPC2 interpolation schemes. The CS interpolation scheme gives the minimum worst-case error (3.15) for this specific example and hence it has been used in the optimization process to generate the parametric macromodel and corresponding sensitivities. Fig. 3.19 shows the parametric behavior of the magnitude of S_{21} as a function of frequency for five different values of S, and $L = 2.5$ mm. In Fig. 3.19, the darkest curve of S_{21} corresponds to the largest

value of S. Similarly, Fig. 3.20 shows the magnitude of S_{21} for five different values of L, with $S = 0.175$ mm.

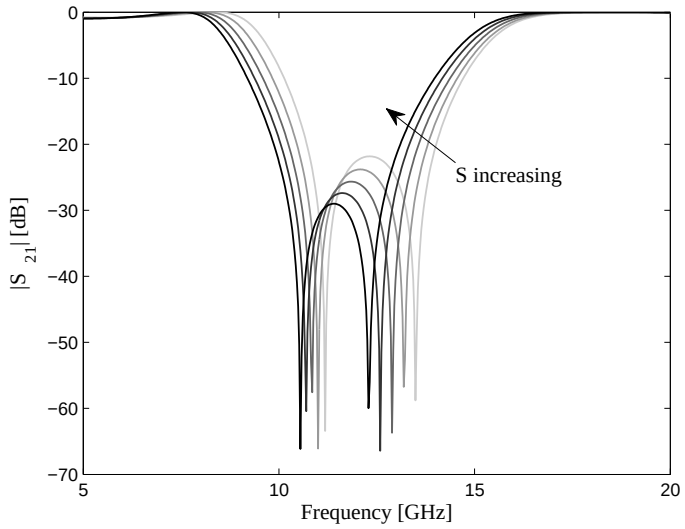

Figure 3.19: DFS filter: Magnitude of S_{21} as a function of frequency for five values of S and $L = 2.5$ mm.

The cost function (3.7) and its gradients calculated using (3.6), have been supplied to the minimax optimization algorithm (3.8), resulting in the optimum design parameter values S^* and L^*. To show the advantage of supplying parametric sensitivity information to the optimizer to speed up the optimization process, two cases have been considered:

- Case I: No sensitivity information is supplied to the minimax algorithm and the algorithm estimates it with the help of finite difference approximation computed using the parametric macromodel.
- Case II: The sensitivity information is calculated from (3.6) and supplied to the minimax algorithm.

In addition to that, in order to show the advantage of using a parametric macromodel, the same optimization problem has been performed using the gradient-based minimax optimization routine in the ADS Momentum software. Table 3.5 compares these three optimization approaches in terms of optimization time for a particular optimization case. Table 3.5 shows the relevant speed-up in the optimization process obtained using the parametric macromodel. We note that the

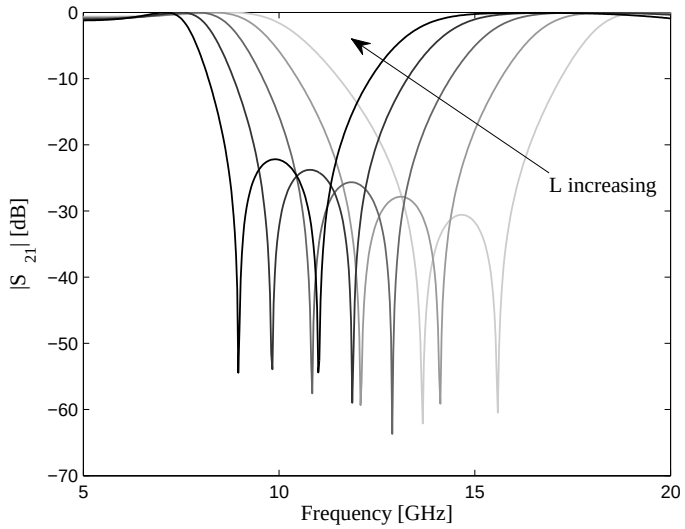

Figure 3.20: DFS filter: Magnitude of S_{21} as a function of frequency for five values of L and $S = 0.175$ mm.

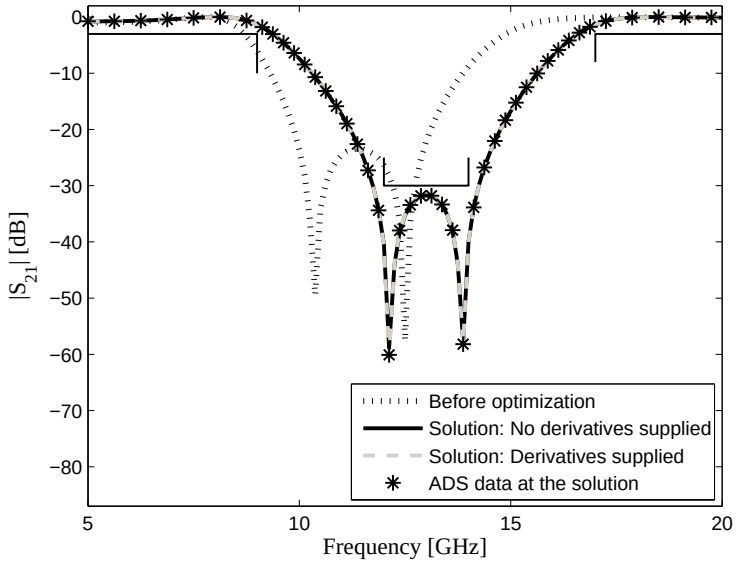

Figure 3.21: DFS filter: Magnitude of S_{21} before and after optimization.

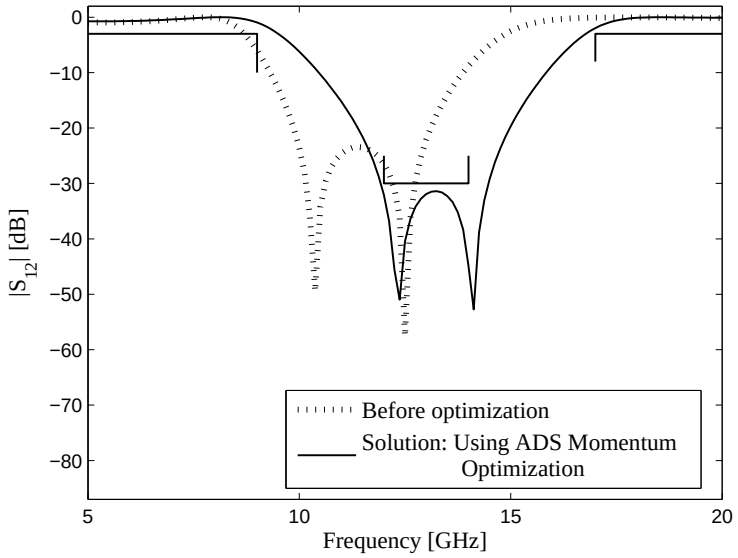

Figure 3.22: DFS filter: Magnitude of S_{21} before and after optimization.

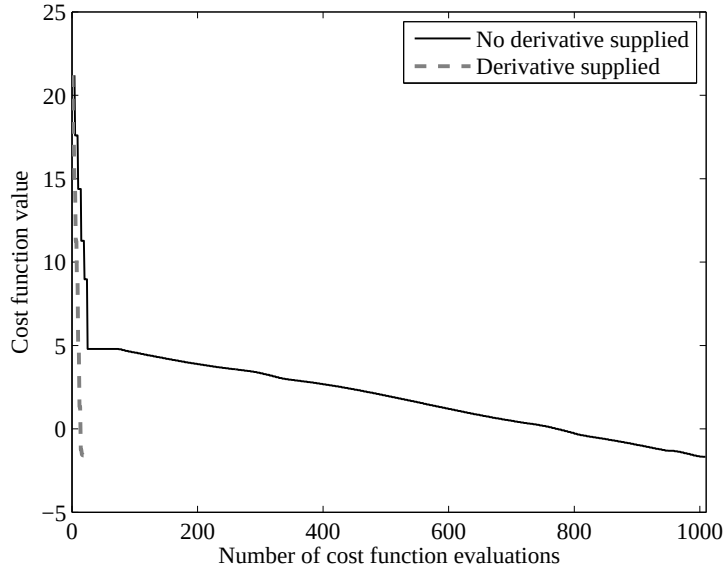

Figure 3.23: DFS filter: Cost function during optimization.

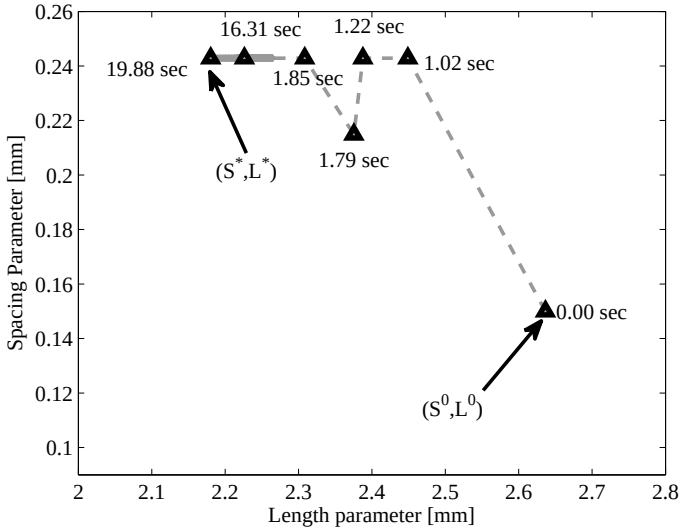

Figure 3.24: DFS filter: The trajectory of the optimal design space point (S^*, L^*) during optimization for the Case I.

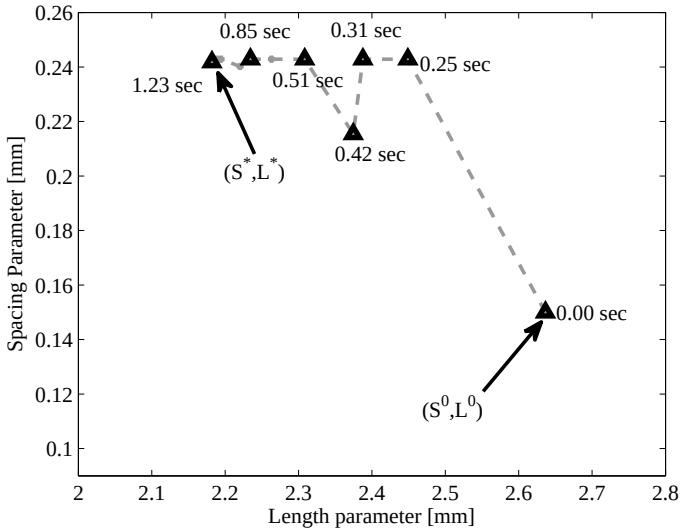

Figure 3.25: DFS filter: The trajectory of the optimal design space point (S^*, L^*) during optimization for the Case II.

method	Parametric macromodel		ADS Momentum
	Case I	Case II	
(S^0, L^0) [mm]	(0.1500, 2.6364)	(0.1500, 2.6364)	(0.1500, 2.6364)
(S^*, L^*) [mm]	(0.2408, 2.1802)	(0.2408, 2.1802)	(0.2303, 2.1580)
Optimization time [sec]	19.88	1.23	5115.00
Cost function at (S^*, L^*) [dB][1]	-1.67	-1.67	0.00

Table 3.5: DFS filter: Optimization using parametric macromodel and ADS Momentum software.

generation of the parametric macromodel requires some initial ADS Momentum simulations and therefore an initial computational effort of 3714.31 sec (for the estimation and validation design space points in Fig. 3.18). However, once the parametric macromodel is generated and validated, it acts as an accurate and efficient surrogate of the original system and can be used for multiple design optimization scenarios, for instance, changing filter specifications. Fig. 3.21 shows the magnitude of S_{21} for the optimization case of Table 3.5. The actual data generated by the ADS Momentum software at the optimum design space point (S^*, L^*) obtained using the parametric macromodel is shown by asterisks in Fig. 3.21. As seen, this is in good agreement with the parametric macromodel response. Fig. 3.22 shows the solution obtained using the gradient-based minimax optimization routine of ADS Momentum software. As clearly seen in Figs. 3.21-3.22, the optimal solutions satisfy all the requirements (3.12), which are shown by thin solid black lines.

Fig. 3.23 shows the value of cost function with respect to the number of cost function evaluations in the Case I and II, which confirms the improved convergence of the optimization when parametric sensitivity information are provided. The convergence time taken by the Case I and II are 13.76 sec and 1.73 sec, respectively. The trajectory of the optimum design space point (S^*, L^*) during optimization for the Case I is shown in Fig. 3.24 with different points showing the output of some particular iterations along with the elapsed time. A similar curve is plotted in Fig. 3.25 for the Case II. Comparing Figs. 3.24-3.25, it is seen that the time for convergence of the Case II is considerably less compared with the Case I.

Table 3.6 shows the comparison of the Case I and II for some important optimization measures which are related to 200 optimization trials, starting from dif-

[1]The Gradient-based minimax optimization routine of ADS Momentum software uses the minimax L1 error function which cannot take a value less than zero.

Method	Number of cost function evaluations			Optimization time [sec]		
	Max	Mean	STD	Max	Mean	STD
Case I	20004	861.33	2360.10	135.30	6.05	16.31
Case II	1601	50.56	138.82	41.10	1.43	3.62

Table 3.6: DFS filter: Comparison between the Cases I and II.

ferent initial design points spread over the design space. Table 3.6 confirms that, there is a considerable reduction in the number of cost function evaluations and the optimization time if derivatives are supplied (Case II).

3.6.2.2 Hairpin bandpass microwave filter

In this example, a hairpin bandpass filter with the layout shown in Fig. 3.26 is modeled [19] . The relative permittivity of the substrate is $\epsilon_r = 9.9$, while its thickness is equal to $0.635\ mm$. The specifications for the bandpass filter are given in terms of the scattering parameters S_{21} and S_{11}:

$$|S_{21}| > -2.5\,\text{dB} \quad \text{for} \quad 2.4\,\text{GHz} < freq < 2.5\,\text{GHz} \tag{3.16a}$$
$$|S_{11}| < -7\,\text{dB} \quad \text{for} \quad 2.4\,\text{GHz} < freq < 2.5\,\text{GHz} \tag{3.16b}$$
$$|S_{21}| < -40\,\text{dB} \quad \text{for} \quad freq < 1.7\,\text{GHz} \tag{3.16c}$$
$$|S_{21}| < -25\,\text{dB} \quad \text{for} \quad 3.0\,\text{GHz} < freq. \tag{3.16d}$$

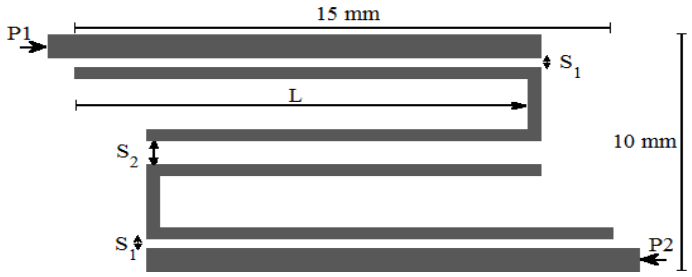

Figure 3.26: Layout of the Hairpin Bandpass Filter.

As shown in Fig. 3.26, three design parameters have been chosen for the design process namely, the spacing between the port lines and the filter lines S_1, the spacing between the two filter lines S_2 and the overlap length L in addition to frequency. The ranges of the different design parameters are shown in Table 3.7.

Parameter	Min	Max
Frequency ($freq$)	1.5 GHz	3.5 GHz
Length (L)	12.0 mm	12.5 mm
Spacing (S_1)	0.27 mm	0.32 mm
Spacing (S_2)	0.67 mm	0.72 mm

Table 3.7: Hairpin filter: Design parameters

The scattering matrix $\mathbf{S}(s, S, L)$ has been computed using the ADS Momentum[2] software. The number of frequency samples were chosen to be equal to 41. As in the first example, two design space grids are used in the modeling process. The average simulation time for each design space point (L, S_1, S_2) has been found to be equal to $T_{SimAvg} = 34.30$ sec. A set of stable and passive *root macromodels* has been built for the estimation grid of $6 \times 4 \times 4$ $(L \times S_1 \times S_2)$ samples by means of VF with a fixed number of poles, $N_P = 18$, selected using an error-based bottom-up approach. Each *root macromodel* has been converted into a state-space form and the set of state-space matrices has been interpolated by the CS, PCHIP and SPC2 methods. The maximum absolute error (3.15) of the models over the validation grid of $5 \times 3 \times 3$ $(L \times S_1 \times S_2)$ samples is equal to -42.57 dB, -38.27 dB and -38.27 dB for the CS, PCHIP and SPC2 methods, respectively. The CS technique has been used in the optimization process of this example, since it gives the best accuracy. Fig. 3.27 shows the parametric behavior of the magnitude of S_{21} as a function of frequency for five different values of L and $S_1 = 0.295$ mm, $S_2 = 0.695$ mm. Fig. 3.28 shows the magnitude of S_{21} when the parameter S_1 changes.

The cost function (3.7) and its gradients calculated using (3.6), have been supplied to the minimax optimization algorithm (3.8), resulting in the optimum design parameter values L^*, $S_1^*$and S_2^*. To show the improved convergence of the optimization when derivatives are supplied, two cases are considered as in the previous example. In addition to that, in order to show the advantage of using a parametric macromodel, the same optimization problem has been performed using the gradient-based minimax optimization routine in the ADS Momentum software. Table 3.8 compares these three optimization approaches in terms of optimization time for a particular optimization case. Table 3.8 shows the relevant speed-up in the optimization process obtained using the parametric macromodel. As mentioned in the previous example, the generation of the parametric macromodel requires an initial ADS Momentum simulation cost of 4836.30 sec. However, once the parametric macromodel is generated and validated, it acts as an accurate and

[2]Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

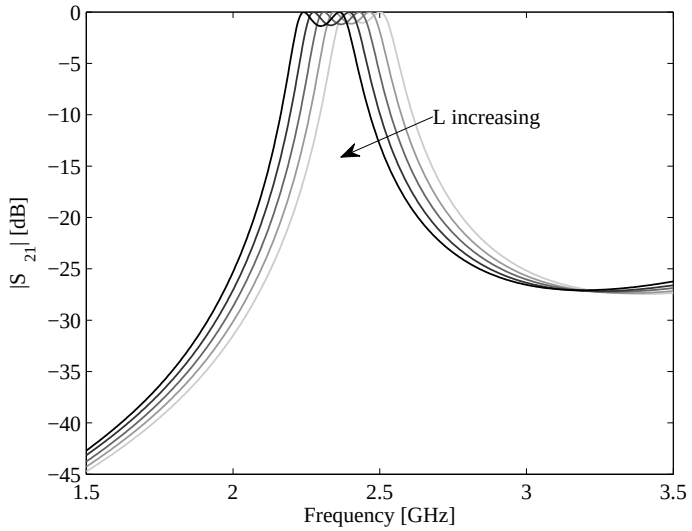

Figure 3.27: Hairpin filter: Magnitude of S_{21} as a function of frequency for five values of L with $(S_1, S_2) = (0.295, 0.695)$ mm.

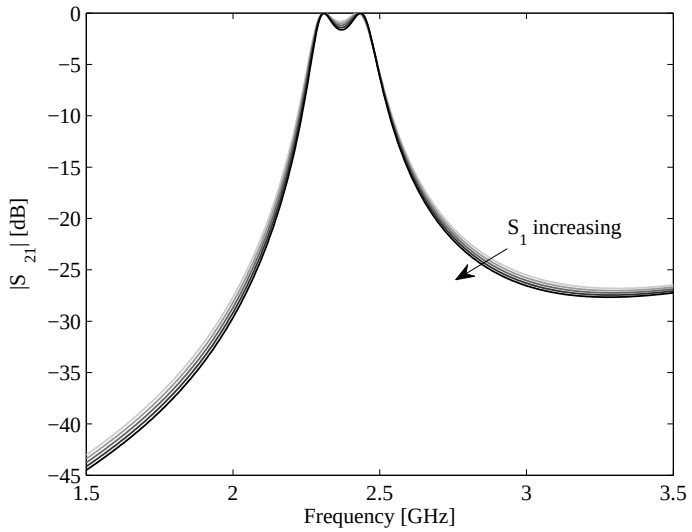

Figure 3.28: Hairpin filter: Magnitude of S_{21} as a function of frequency for five values of $S_1$with $(L, S_2) = (12.25, 0.695)$ mm.

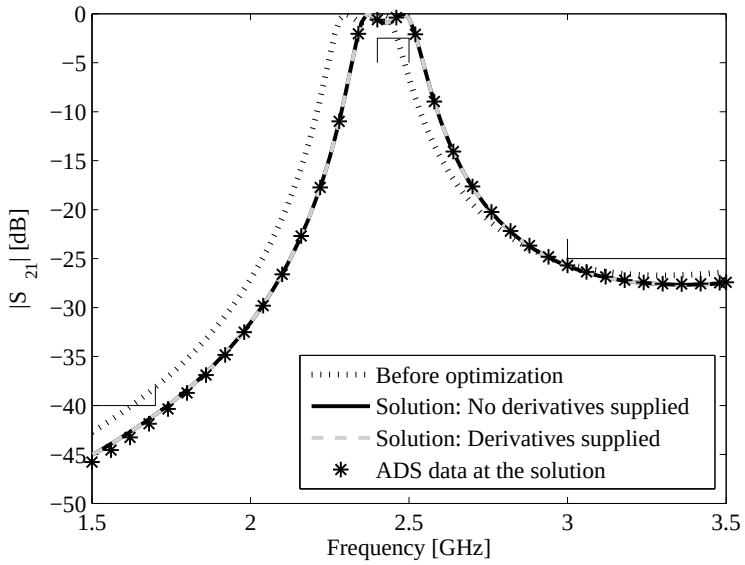

Figure 3.29: Hairpin filter: Magnitude of S_{21} before and after optimization.

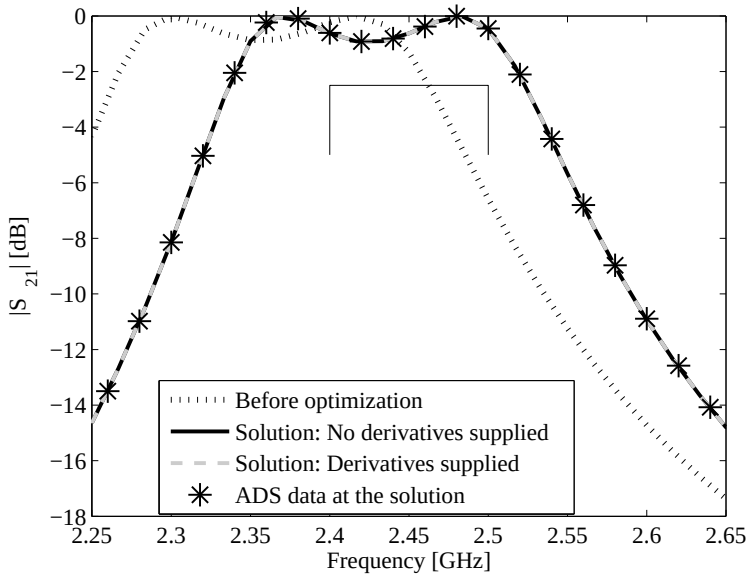

Figure 3.30: Hairpin filter: A zoomed in view of Fig. 3.29.

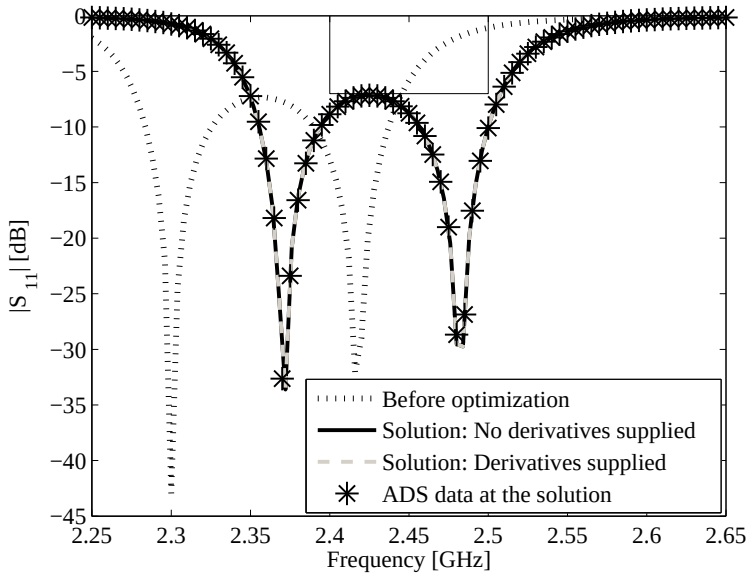

Figure 3.31: Hairpin filter: Magnitude of S_{11} before and after optimization.

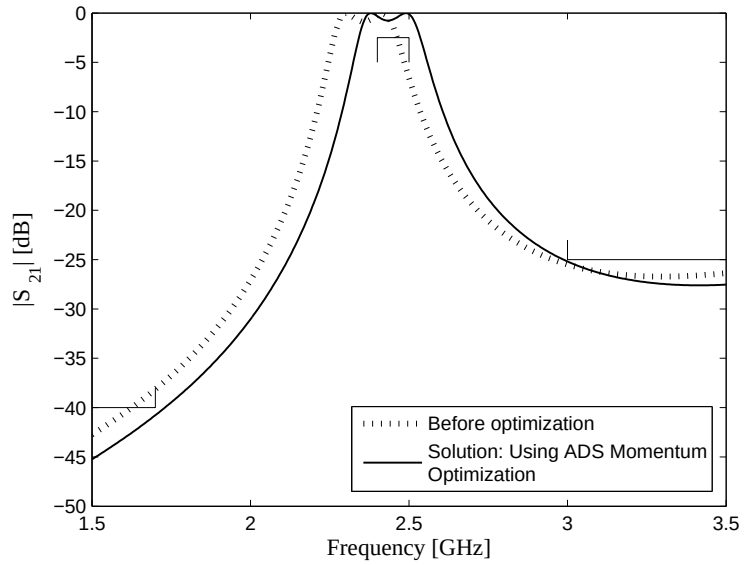

Figure 3.32: Hairpin filter: Magnitude of S_{21} before and after optimization.

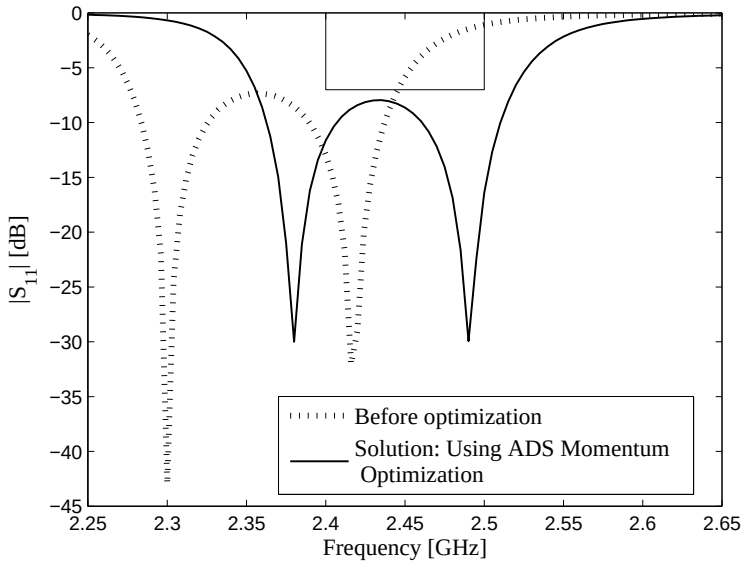

Figure 3.33: Hairpin filter: Magnitude of S_{11} before and after optimization.

efficient surrogate of the original system and can be used for multiple design optimization scenarios, for instance, changing filter specifications. Fig. 3.29 shows the magnitude of S_{21} for the optimization case of Table. 3.8. The actual data generated by the ADS Momentum software at the optimum design space point (L^*, S_1^*, S_2^*) obtained using the parametric macromodel is shown by asterisks in Fig. 3.29. As seen, this is in agreement with the parametric macromodel response. The requirements (3.16) are shown by the thin black solid lines. A magnified view of the passband is shown in Fig. 3.30 for clarity. Similar results are given for the magnitude of S_{11} in Fig. 3.31. As clearly seen, all the requirements are met for the optimal design point. Figs. 3.32-3.33 shows similar results for the solution obtained using the gradient-based minimax optimization routine of ADS Momentum software. Here also, all the design specifications (3.16) are met. Some important measures of the optimization process related to 1000 trial runs are shown in Table 3.9, which clearly shows the better convergence properties of the Case II.

Another important thing to be noted here is that, with the increase in the number of design parameters, the initial number of EM simulations increases considerably due to the curse of dimensionality. Adaptive sampling schemes, which take into account influence of the design parameters on the system behavior, could be used to properly sample the design space prior to the parametric macromodeling and help resolve this issue. For instance, in the second example, from Figs. 3.27

method	Parametric macromodel		ADS Momentum
	Case I	Case II	
(L^0, S_1^0, S_2^0) [mm]	(12.2778, 0.2700, 0.6867)	12.2778, 0.2700, 0.6867)	12.2778, 0.2700, 0.6867)
(L^*, S_1^*, S_2^*) [mm]	(12.0568, 0.2984, 0.7200)	(12.0568, 0.2984, 0.7200)	(12.0036, 0.2700, 0.6826)
Optimization time [sec]	13.46	0.79	1251.00
Cost function at (S^*, L^*) [dB][1]	-0.76	-0.76	0.00

Table 3.8: Hairpin filter: Optimization using parametric macromodel and ADS Momentum software.

Method	Number of cost function evaluations			Optimization time [sec]		
	Max	Mean	STD	Max	Mean	STD
Case I	53183	114.39	1689.71	565.14	1.29	17.95
Case II	99	13.73	5.79	4.69	0.76	0.28

Table 3.9: Hairpin filter: Comparison between the Cases I and II.

and 3.28 it is seen that the overlap length L of the Hairpin Filter is more influential than the Spacing S_1, which allows one to sample along S_1 more sparsely using a wise adaptive sampling scheme, thereby reducing the number of initial EM simulations needed for the construction of parametric macromodels and the corresponding computational effort.

3.7 Conclusions

Gradient-based design optimization of microwave systems using parametric sensitivity macromodels has been presented in this chapter. Parameterized frequency-domain data samples are used to build a set of *root macromodels* in a state-space form. Then, this set of state-space matrices is parameterized using suitable interpolation schemes which are continuously differentiable. This allows to accurately and efficiently calculate parametric sensitivities over the entire design space of interest. First, two analytical examples are used to show the accuracy of this approach in calculating parameterized sensitivities. Then, parametric macromodels and corresponding sensitivities are used for the gradient-based design optimization in two proposed numerical examples, which confirm the applicability of the

proposed technique to the optimization process of microwave filters. Also, the importance of parameterized sensitivity information to speed up the design optimization process has been shown in the examples.

3.8 Acknowledgements

This work was supported by the Research Foundation Flanders (FWO).

References

[1] D. Li, J. Zhu, N. K. Nikolova, M. H. Bakr, and J. Bandler, "Electromagnetic optimisation using sensitivity analysis in the frequency domain," *IET Microwaves, Antennas Propagation*, vol. 1, no. 4, pp. 852 –859, Aug. 2007.

[2] A. Dounavis, R. Achar, and M. Nakhla, "Efficient sensitivity analysis of lossy multiconductor transmission lines with nonlinear terminations," vol. 49, no. 12, pp. 2292–2299, Dec. 2001.

[3] N. K. Nikolova, J. W. Bandler, and M. H. Bakr, "Adjoint techniques for sensitivity analysis in high-frequency structure CAD," vol. 52, no. 1, pp. 403–419, Jan. 2004.

[4] D. Deschrijver and T. Dhaene, "Stability and passivity enforcement of parametric macromodels in time and frequency domain," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 11, pp. 2435 –2441, Nov. 2008.

[5] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *Microwave and Wireless Components Letters, IEEE*, vol. 19, no. 10, pp. 608 –610, Oct. 2009.

[6] ——, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623 –629, Aug. 2010.

[7] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[8] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *2010 IEEE 14th Workshop on Signal Propagation on Interconnects (SPI)*, pp. 117 –120, May 2010.

[9] ——, "Passive parametric modeling of interconnects and packaging components from sampled impedance, admittance or scattering data," in *3rd Electronic System-Integration Technology Conference (ESTC)*, sept. Sept. 2010, pp. 1 –6.

[10] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," vol. 14, no. 3, pp. 1052–1061, July 1999.

[11] B. Gustavsen, "Improving the pole relocating properties of vector fitting," vol. 21, no. 3, pp. 1587–1592, July 2006.

[12] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 6, pp. 1587–1592, June 2008.

[13] C. de Boor, *A Practical Guide to Splines*. New York, USA: Springer-Verlag New York, Inc., 2001.

[14] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, pp. 238–246, Apr. 1980.

[15] S. Pruess, "Shape preserving C2 cubic spline interpolation," *IMA Journal on Numerical Analysis*, vol. 13, no. 4, pp. 493–507, 1993.

[16] D. Z. Du and P. M. Pardalos, *Minimax and Applications*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1995.

[17] F. M. Tesche, "A simple model for the line parameters of a lossy coaxial cable filled with a nondispersive dielectric," *Electromagnetic Compatibility, IEEE Transactions on*, vol. 49, no. 1, pp. 12–17, 2007.

[18] J. Hong and M. J. Lancaster, *Microstrip Filters for RF/Microwave Applications*. pp. 77-82, John Wiley & Sons, Inc., 2001.

[19] J. Ureel and D. De Zutter, "Gradient-based minimax optimization of planar microstrip structures with the use of electromagnetic simulations," *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, vol. 7, no. 1, pp. 29–36, Jan. 1997.

4

Parametric Models of Microwave System Responses Using Sequential Sampling on Unstructured Grids

K. Chemmangat, F. Ferranti, T. Dhaene and L. Knockaert

Published in International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, vol. 27, no. 1, pp. 122-137, 2014

⋆⋆⋆

This chapter presents two sequential sampling algorithms for the macromodeling of parameterized system responses in model-dependent sampling frameworks. The construction of efficient algorithms for the automatic selection of samples for building parametric macromodels of frequency-domain responses is addressed in this chapter. The methods described here are the initial step towards the complete automation of the parametric macromodeling process. The sequential sampling algorithms proposed here are tailored towards the application of local parametric macromodeling schemes on unstructured design space grids. Two pertinent examples are considered. For the first one, different algorithms are applied and a comparison is made in terms of the number of samples generated, accuracy and CPU time. As a second example, four design parameters are taken into account with one of the proposed algorithms and the generated model is used in a frequency-domain optimization.

4.1 Introduction

Design space exploration, design optimization and sensitivity analysis of electromagnetic (EM) systems often require expensive simulations using EM solvers which normally provide high accuracy at a significant cost in terms of memory storage and computing time. The computational complexity of these EM solvers often results in a design cycle that is costly in terms of execution time, which might not be acceptable in practice.

An alternative is to develop accurate and efficient parametric macromodels which approximate the complex behavior of EM systems, characterized by frequency and additional design parameters, such as geometrical or substrate features. Parametric macromodeling of EM systems has attracted a lot of attention during recent years [1–15]. However, one of the key challenges in these modeling approaches, which requires further research, is the optimal selection of data samples over the design parameter space, in order to limit the total number of expensive EM simulations [1–5, 7, 14].

Sequential sampling techniques can be classified into three main categories, i.e., the input-based methods, the output-based methods and the model-based methods. The sequential Design of Experiments (DoE) method falls into the *input-based* category, where the input design space is filled according to some measure depending on the density of the selected samples [16, 17]. The *output-based* sampling schemes depend on the output system response in order to sequentially select new samples in the design space [14, 18]. A recent work on sequential sampling of scattering parameter responses (S-responses) uses output S-responses along with an exploration-exploitation-based approach [14]. In the exploration phase, which searches the design space for unidentified regions, a space filling technique is used to fill the design space uniformly, whereas in the exploitation phase, which iden-

tifies potentially interesting and dynamic regions of the design space, a local estimate of the gradient is used to select new samples. In contrast to the other two categories, *model-based* sampling schemes depend on intermediate macromodels to select the distribution of new samples [1–5, 7].

Apart from the above mentioned categories, there are global and local sequential sampling methods. *Global* sequential sampling algorithms build a single model for the complete design space. Multi-dimensional Cauchy methods are one of the popular global sequential sampling schemes [1–3, 6]. In [1], two different sampling approaches are mentioned. The first approach which uses a stable recursive Burlisch-Stoer algorithm has an inherent limitation since only the last parameter can be sampled and the rest of the parameters must lie on a fully filled grid. The second method, which uses a multi-dimensional rational function expansion can be very ill-conditioned at higher dimensions, limiting the applicability of this method to relatively simple problems. Total Least Square (TLS) algorithm was used in [2] to solve for the coefficients of the multivariate rational model with a QR factorization, and for each added samples a QR update is used instead of full QR decomposition to gain computational time. An adaptive multivariate rational fitting is reported in [3] which uses Tchebyshev orthogonal polynomials to improve the conditioning of the matrices to be solved. In [6], a multi-dimensional rational approximation is built using convex optimization with linear constraints. These constrains ensure that the algorithm is stable and the value of the interpolated function lies within the physical bounds. Neural network and radial basis function-based sequential sampling method are also found in the literature [4, 5] which are also global modeling schemes. However, such global modeling schemes [1–6] suffer from the following limitations:

1. For these methods, it is difficult to preserve physical properties such as stability and passivity of the generated macromodels over the design space of interest which are very important while performing time-domain simulations [19].

2. For most of these methods, the initial sampling of the design space is very important since they use the difference between two models with different modeling order to estimate the accuracy. The design space point where the two models differ the most is sampled. However, if the initial sampling is not adequate, the estimation of error may not be accurate, which might lead to a divergence scenario.

3. For relatively high dimensions the memory requirement can be prohibitively high limiting their applicability. Dimensionality reduction techniques such as [20] could help improve this scenario to a certain extend.

This chapter describes two sequential sampling algorithms for selecting the

optimum number of samples such that accurate parametric models for parameterized system responses can be generated. The methods presented here are quite different from the approach in [1–6, 14], since the proposed algorithms are *local* and work on local N-dimensional hyperectangular (n-box) regions of the design space. This creates the possibility of a tree-based implementation of the algorithms similar to [21, 22], reducing the model evaluation time and making them portable to parallel computing platforms. The algorithms based on exploitation of the design space result in a design space which is suitable for the application of different passivity-preserving local parametric macromodeling algorithms [9–13, 15]. Also, in contrast to the global sequential sampling methods [1–6], once the passivity is enforced on the sampled points, the local interpolation ensures that the models are stable and passive over the complete design space of interest and hence can be used in time-domain simulations. Also, the two sequential sampling algorithms described here are applied on model-based frameworks.

The chapter is organized as follows. Section 2 presents some important aspects of sampling schemes. Section 3 describes the two proposed sampling algorithms in detail. Different pertinent numerical examples are presented in order to validate and compare the different schemes discussed in the chapter, and the corresponding results are presented in Section 4.

4.2 Sequential Sampling

A sequential sampling algorithm selects those samples which allow to build an accurate model with respect to some error measures. A design space consists of all the design parameters such as layout variables or substrate features. Since frequency is a special parameter whose behavior can be accurately modeled using a rational function, it is sampled separately and is not considered as a part of the design space. If needed, an Adaptive Frequency Sampling (AFS) can be used to sample the frequency axis. In this chapter, the word *design sample point* denotes the frequency-domain response of a microwave system for a particular design configuration. The sampled system response data obtained through an EM solver are used to build macromodels which accurately describe the parameterized input-output behavior of the original complex system with a predefined level of accuracy. Some important aspects of a suitable sequential sampling strategy are briefly explained in subsections 4.2.1 and 4.2.2.

4.2.1 Error measure

An important aspect of a suitable sequential sampling algorithm is to define proper error criteria for choosing new samples. During the evolution of the design space any new point added in the design space considerably reduces a judiciously chosen

error measure between the system response of the original simulation model and the intermediate macromodel. Several error measures can be used to compare frequency-domain responses. The maximum absolute error between the original frequency response $H_{i,j}$ and the macromodel $R_{i,j}$ can be used,

$$E^{\mathrm{MaxAbs}}(\vec{g}) = \max_{i,j,k}\Big(\Big|R_{i,j}(s_k,\vec{g}) - H_{i,j}(s_k,\vec{g})\Big|\Big) \qquad (4.1)$$
$$i = 1,\ldots,P_{in},\ j = 1,\ldots,P_{out},\ k = 1,\ldots,N_s$$

with number of input ports P_{in}, output ports P_{out} and frequency samples N_s (s_k is the complex frequency or Laplace variable). However, the maximum absolute error can be misleading when comparing two frequency responses since this can give extreme values, e.g. at the resonance peaks. An alternative is to use mean error measures like the mean absolute error (MAE):

$$E^{\mathrm{MAE}}(\vec{g}) = \frac{\sum_{i=1}^{P_{in}}\sum_{j=1}^{P_{out}}\sum_{k=1}^{N_s}|R_{i,j}(s_k,\vec{g}) - H_{i,j}(s_k,\vec{g})|}{P_{in}P_{out}N_s}. \qquad (4.2)$$

This chapter uses the mean of absolute error (4.2) for the sequential sampling algorithms, as suggested in [23]. In addition, mean error measures capture the global trend of the underlying response function and hence it is rather smooth over the design space. This also gives better convergence properties for the adaptive sampling schemes.

In addition to the absolute error measures discussed above, weighted or relative error measures such as the relative mean error:

$$E^{\mathrm{MAE}}_{\mathrm{Rel}}(\vec{g}) = \frac{\sum_{i=1}^{P_{in}}\sum_{j=1}^{P_{out}}\sum_{k=1}^{N_s}\frac{|R_{i,j}(s_k,\vec{g}) - H_{i,j}(s_k,\vec{g})|}{|H_{i,j}(s_k,\vec{g})|}}{P_{in}P_{out}N_s} \qquad (4.3)$$

can be used while modeling impedance or admittance parameters, as these parameters are not bounded so that good accuracy is obtained over the complete dynamic range. Even though this gives a basic idea of how to choose proper error measures, the selection should be problem specific. For example, in a case where the modeling of resonances are important, the absolute maximum is preferred over the mean. This means that ultimately the selection of error measure should be driven from the modeling problem under consideration.

4.2.2 Selection of parametric macromodeling method

There are several choices available for a macromodeling scheme to be used in the sequential sampling and the final design space will depend on this selection. As discussed in the introduction, the sequential sampling algorithms considered in this work are local and hence any macromodeling scheme suitable to this scenario could be used.

One choice could be to use multivariate local interpolation methods, e.g., the piecewise multilinear and multivariate simplicial methods [24] on the system frequency response data and build a parametric macromodel. This choice makes the sequential sampling depend on the system frequency response data and the used interpolation scheme.

Another alternative is to use one of the local parametric macromodeling schemes which use the Vector Fitting (VF) technique [25–27] to build frequency-dependent rational models called *root macromodels* at the selected design space samples and then parameterize them, see [9–13, 15]. The parametric macromodeling process starts with a set of multivariate data samples $\{(s_i, \vec{g}_k), \mathbf{H}(s_i, \vec{g}_k), i = 1, 2, \ldots, N_s, k = 1, 2, \ldots, K_{tot}\}$ which depends on frequency and additional design parameters. From these data samples, a set of *root macromodels* in pole-residue form are built for a set of design space samples $\vec{g}_k$ by means of VF yielding a set of *root macromodels* $\mathbf{R}(s, \vec{g}_k)$. A pole-flipping scheme is used to enforce strict stability [25] and passivity assessment and enforcement is achieved using robust standard techniques [28, 29] resulting in a set of stable and passive *root macromodels*. The next step of these parametric macromodeling algorithms is the parameterization of the set of *root macromodels* $\mathbf{R}(s, \vec{g}_k)$. In [10, 11], a parametric macromodel is built by interpolating a set of *root macromodels* at an input-output level, while in [9, 12, 13], both poles and residues are parameterized by interpolating the internal state-space matrices, resulting in higher modeling capability with respect to [10, 11]. In [15], a novel enhanced interpolation of *root macromodels* at an input-output level is described, which is based on the use of some coefficients: one coefficient as a multiplicative factor at the input/output level of the system and the other coefficient as a compression or expansion term for the Laplace variable s. It results in high modeling capability and robustness.

Using stability and passivity enforced VF-based parametric macromodeling schemes have the following advantages:

1. They generate rational models which are stable and passive over the complete design space, and therefore suitable for time-domain simulations, which is difficult to achieve using the interpolation of raw frequency response data.

2. By selecting a powerful parametric macromodeling scheme, the number of EM simulations can be reduced considerably as shown by the numerical results in Section 4.4.

4.3 Sequential Sampling algorithms

This section describes two sequential sampling algorithms developed in this work. The proposed sequential sampling algorithms preserve the rectangular nature of

the sampling grid such that the existing local parametric macromodeling methods [10, 11, 15] can be directly applied. We call a n-box region of the design space as a subspace or a node.

The following aspects of the proposed sequential sampling methods distinguishes them from the previous methods [1–6]:

1. The sequential sampling described here preserve the physical properties such as stability and passivity for the entire design space of interest, which is fundamental for time-domain simulations.

2. The methods presented here are local and hence can efficiently be implemented on parallel computing platforms in a tree-based structure.

3. In contrast to the multivariate rational fitting, several frequency-dependent models are identified and then parameterized with respect to design parameters. This considerably reduces the utilization of memory resources in high dimensional design spaces.

4.3.1 Algorithm I: Division at the center of a subspace

Here we assume that the model is prone to a high error value at the center of a subspace. Hence, the idea is to subdivide a subspace at its center, when the error measure is greater than a predefined threshold. For example, let us consider a rectangular subspace for a two dimensional design space with four samples as shown in Fig. 4.1-a for which a bivariate macromodel is generated. A test sample is selected at the geometric center of the subspace (shown by a gray circle in Fig. 4.1-b) and the frequency responses between the original EM solver and macromodel are compared at this test point. If the modeling error (4.2) is found to be greater than a predefined threshold Δ, the subspace is subdivided, generating four new rectangular subspaces. This procedure is repeated until the error at the geometrical center of each subspace is less than the predefined threshold (Fig. 4.1-c) and the final design space can for example look like Fig. 4.1-d.

Fig. 4.2 shows the design space of Fig. 4.1-d as a tree structure with each node (circle) representing a particular subspace. The branches (dotted lines) represent the connection between a node and its subregions or *child nodes*. Note that each node is divided into four child nodes or in general into 2^N nodes for an N-dimensional subspace. The terminal nodes are represented in gray circles where the required accuracy is achieved. With a tree-based implementation, the advantage is the portability to parallel computing platforms with a reduction of computing time.

The sequential sampling process of *Algorithm I* consists of the following steps:

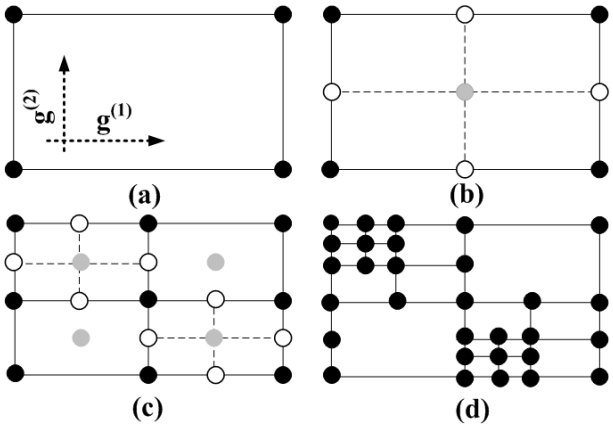

Figure 4.1: Algorithm I*: Division of the design space.*

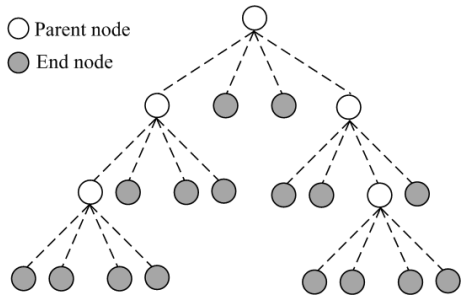

Figure 4.2: Fig. 4.1-d represented in a tree structure.

I) Generating an initial n-box design space by defining 2^N corner points that define the convex hull of the design space, where N is the number of design parameters $\vec{g} = (g^{(1)}, \ldots, g^{(N)})$. The number of subspaces $Q = 1$.

II) Building a macromodel $\mathbf{R}(s, \vec{g})$ for the entire design space with Q subspaces.

III) For each particular subspace $q = 1, \ldots, Q$, checking the error criteria at the center of the subspace.

   i. IF($Err_q > \Delta$):

      i. Divide the subspace q into 2^N new subspaces and increment $Q = Q + 2^N - 1$.

      ii. Increment $q = q + 1$ and go to Step II.

   ii. ELSE: increment $q = q + 1$.

i. IF $(q <= Q)$: Not all subspaces are checked for the error criteria. Go to Step II.

ii. ELSE: Go to Step IV.

IV) Terminating the sequential sampling algorithm.

4.3.1.1 Demonstration of *Algorithm I*: four coupled microstrips

Four coupled microstrips, shown in Fig. 4.3, are modeled, where the spacing S between the lines and the length L of the lines are chosen as the design parameters in addition to frequency. Table 4.1 shows the ranges of all parameters. The substrate is chosen with relative permittivity $\epsilon_r = 9.6$ and thickness equal to $0.125\ mm$

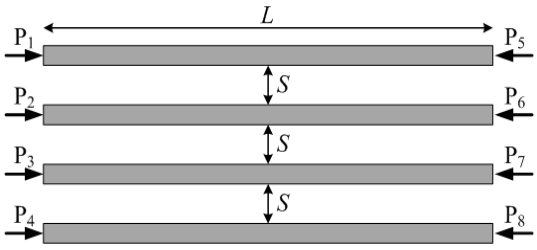

Figure 4.3: Top view of the layout of four coupled microstrips.

Parameter	Min	Max
Frequency $(freq)$	0 GHz	10 GHz
Spacing (S)	0.05 mm	0.15 mm
Length (L)	5 mm	10 mm

Table 4.1: Design parameters of four coupled microstrips.

The scattering matrix $\mathbf{S}(s, S, L)$ has been computed using the ADS Momentum[1] software, and the number of frequency samples has been chosen to be equal to $N_s = 51$. Fig. 4.4 shows the parametric behavior of the magnitude of $\mathbf{S}_{11}(s, S, L)$ with respect to L and frequency for $S = 0.10$ mm. Similarly, the parametric behavior of the magnitude of $\mathbf{S}_{81}(s, S, L)$ with respect to S and frequency is shown in Fig. 4.5 for $L = 7.5$ mm. Matlab R2010a[2] is used to drive the ADS Momentum simulations to generate Scattering parameter responses (**S**-responses) which are then supplied to the proposed sequential sampling algorithm.

[1]Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.
[2]The Mathworks, Inc., Natick, MA, USA

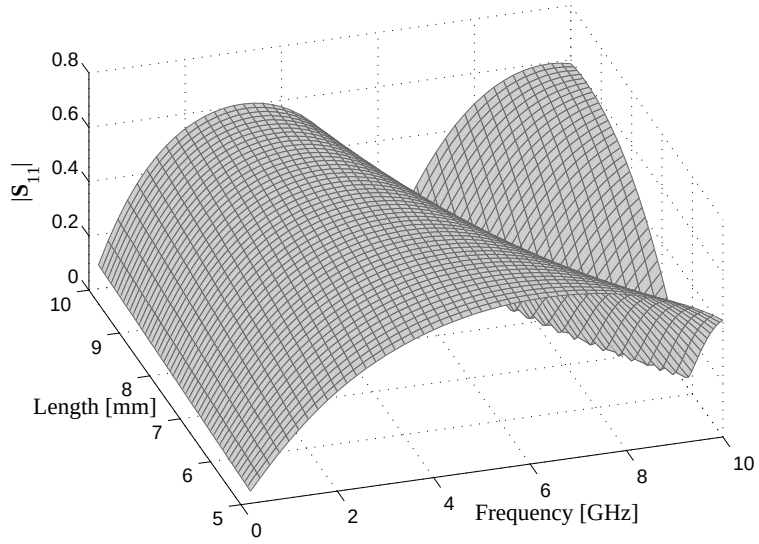

Figure 4.4: Parametric behavior of the magnitude of $\boldsymbol{S}_{11}$ for $S = 0.10$ mm.

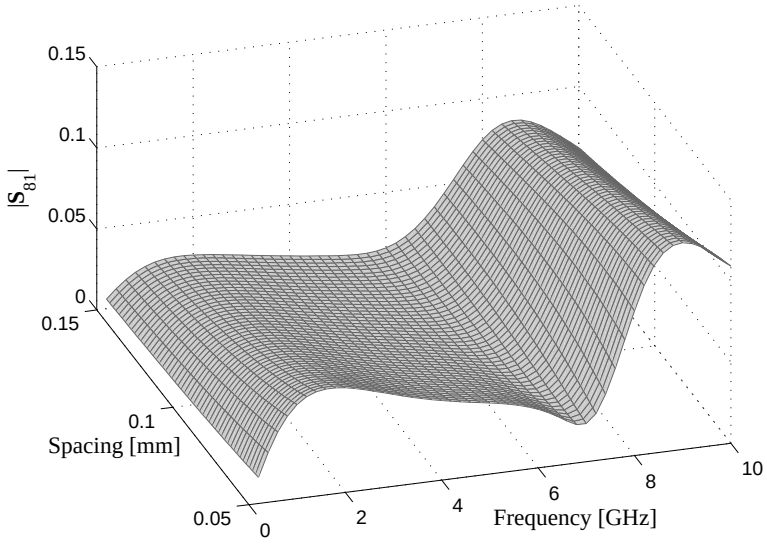

Figure 4.5: Parametric behavior of the magnitude of $\boldsymbol{S}_{81}$ for $L = 7.5$ mm.

Modeling accuracy		Number
Target [dB]	Achieved [dB]	of points
-40	-47.98	9
-45	-47.98	9
-50	-50.80	14
-55	-55.88	22
-60	-60.07	42

Table 4.2: Sequential sampling results.

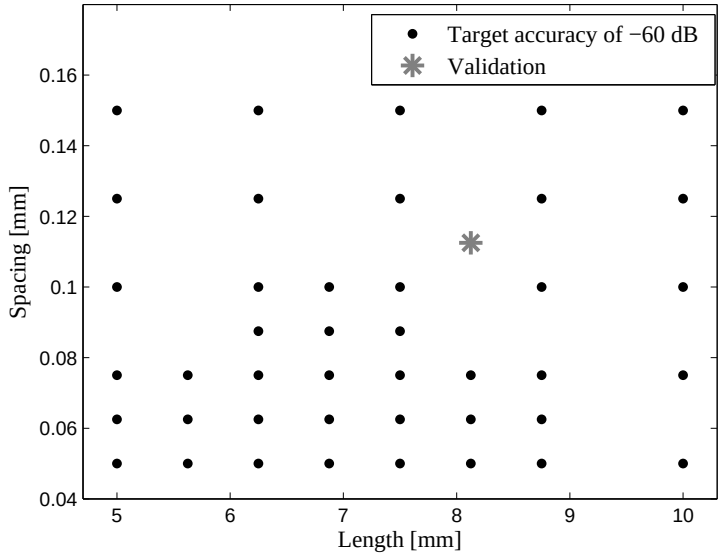

Figure 4.6: Design space generated by the proposed sequential sampling algorithm.

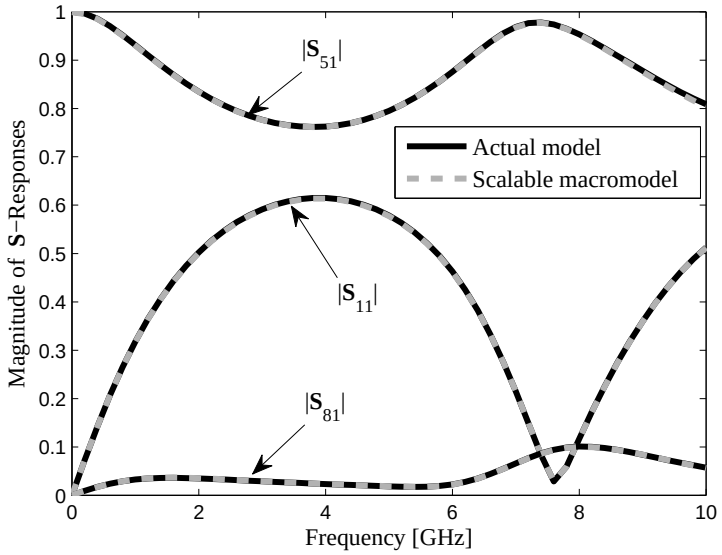

*Figure 4.7: Magnitude of some **S**-matrix entries at* $(S, L) = (0.1125, 8.125)$ *mm.*

Table 4.2 shows the number of design space points generated for varying levels of target accuracy in terms of the MAE (4.2). As discussed previously, in the algorithm the validation points are located at the geometrical center of each subspace. The final design space generated using the proposed sequential sampling algorithm is shown in Fig. 4.6 for the accuracy level equal to -60 dB. Some of the **S**-matrix entries at one of the validation point (gray asterisk in Fig. 4.6) are plotted for the parametric macromodel and compared with the ADS momentum data in Fig. 4.7, showing the achieved accuracy. One important point to be noted here is the fact that the decision on the target accuracy should be left to the designer who uses these models. For examples, considering microwave filter design where we need to optimize a response with a stop band requirement of -30 dB, an error target of -40 dB could be set. Also, with these sequential sampling schemes, multiple fidelity models can be extracted. Hence for the above stated example, the designer can aim initially for a -30 dB accuracy and use the model in the optimization and later use the high fidelity model to confirm the design.

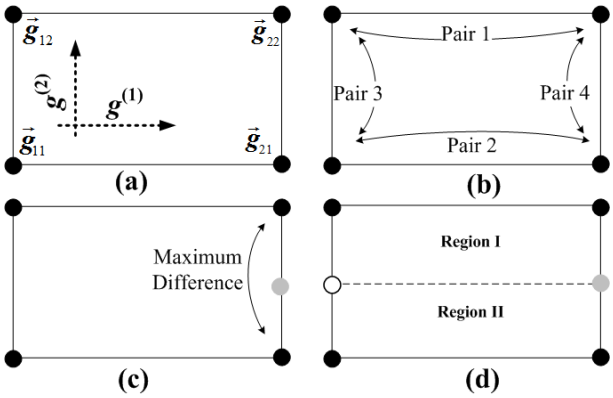

Figure 4.8: Algorithm II*: Division of the design space.*

4.3.2 Algorithm II: Division at the edge of a subspace followed by final refinement using Algorithm I

As in *Algorithm I*, the rectangular grid structure is preserved here. However, in contrast with *Algorithm I*, a subspace is subdivided into two equal halves along the maximum sensitive design parameter instead of the geometric center. The maximum sensitive parameter in a particular subspace is found in the following way. Consider a bivariate case with parameter vector $\vec{g} = (g^{(1)}, g^{(2)})$ as shown in Fig. 4.8-a, where the four initial samples are marked by $\vec{g}_{ij} = (g_i^{(1)}, g_j^{(2)})$; $i, j = 1, 2$. Now these points are paired for all four combinations as in Fig. 4.8-b, and the difference between the macromodel responses $\mathbf{R}(s, \vec{g}_{ij})$ are measured for each pair using the error measure given by (4.2). Let us assume that the difference between the pairs $\mathbf{R}(s, \vec{g}_{21})$ and $\mathbf{R}(s, \vec{g}_{22})$:

$$Diff(\vec{g}_{21}, \vec{g}_{22}) = \frac{\sum_{i=1}^{P_{in}} \sum_{j=1}^{P_{out}} \sum_{k=1}^{N_s} |R_{i,j}(s_k, \vec{g}_{21}) - R_{i,j}(s_k, \vec{g}_{22})|}{P_{in} P_{out} N_s}, \quad (4.4)$$

is the biggest as shown in Fig. 4.8-c, then a new point is taken at the middle of the range $[\vec{g}_{21}, \vec{g}_{22}]$, shown by the gray circle, and this is used as a test point to check the accuracy of the macromodel with respect to the original EM solver. If the error is found to be greater than a predefined threshold Δ, the subspace is divided into two *child nodes* along this design parameter which happens to be $g^{(2)}$ in Fig. 4.8.d, by taking additional points (white circles). Finally, the generated design space is further refined using *Algorithm I*.

The sequential sampling process of *Algorithm II* consists of the following steps:

I) Same as Step I of *Algorithm I*.

II) Same as Step II of *Algorithm I*.

III) For each particular subspace $q = 1, \ldots, Q$, finding the maximum sensitive design parameter and hence the test point as previously explained.

IV) Checking the error criteria at the test point.

i. IF($Err_q > \Delta$):

i. Divide the subspace q into 2 new subspaces and increment $Q = Q + 1$.

ii. Increment $q = q + 1$ and go to Step II.

ii. ELSE: increment $q = q + 1$.

i. IF ($q <= Q$): Not all subspaces are checked for the error criteria. Go to Step II.

ii. ELSE: Go to Step V.

V) Performing final refinement using *Algorithm I*.

VI) Terminating the sequential sampling algorithm.

The main advantage of *Algorithm II* as compared to *Algorithm I* is that the number of points added for each subspace division is significantly reduced. Also, if some of the parameters are highly influential, the algorithm divides along that direction, ensuring fine refinement along the highly dynamic parameters instead of dividing all the parameters at the same time as in *Algorithm I*. Please note that, the capability to implement the algorithm as a tree structure is preserved.

4.4 Numerical simulations

The proposed algorithms have been implemented in Matlab R2010a[3] and used to drive the ADS Momentum[4] simulations to generate S-responses at selected samples. The numerical simulations have been performed on a Windows 7 platform on Intel(R) Core(TM)2 Duo P8700 2.53 GHz machine with 2 GB RAM. The proposed sequential sampling algorithms (Section 4.3) are compared and the effect of the two macromodeling schemes on the sampling is studied on pertinent numerical examples.

[3]The Mathworks, Inc., Natick, MA, USA

[4]Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

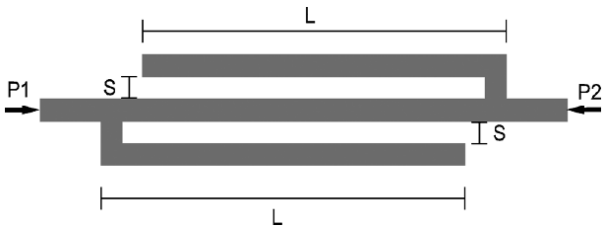

Figure 4.9: Layout of the DFS band-stop filter.

4.4.1 Double folded stub microwave filter

In this example, a Double Folded Stub (DFS) band-stop microwave filter on a substrate with relative permittivity $\epsilon_r = 9.9$ and a thickness of $0.127\ mm$ is modeled. The layout of this DFS filter is shown in Fig. 4.9. The spacing S between the stubs and the length L of the stubs are chosen as design parameters in addition to frequency whose ranges are $S \in [0.15, 0.25]$ mm, $L \in [1.0, 2.5]$ mm and frequency $\in [5, 20]$ GHz.

The S-response matrix $\mathbf{S}(s, S, L)$ has been computed using the ADS Momentum solver and the number of frequency samples has been chosen equal to 31. Fig. 4.10 shows the parametric behavior of the magnitude of S_{11} as a function of S and frequency for $L = 1.75$ mm. Similarly, Fig. 4.11 shows the magnitude of S_{21} as a function of L and frequency for $S = 0.20$ mm. Figs. 4.12 and 4.13 show the surface plots for the Figs. 4.10 and 4.11 respectively.

The two sequential sampling algorithms described in Section 4.3 are implemented using two different macromodeling schemes. In the first implementation, a local multilinear interpolation of the raw frequency response data is used to build the intermediate macromodels. The second implementation uses the parametric macromodeling technique described in [15] which perfectly fits into our proposed sequential sampling algorithms. The sequential sampling algorithms are applied on each of these cases, as seen in Table 4.3. Table 4.3 compares all proposed schemes in terms of the total number of samples, the worst case mean absolute error over the design space, the CPU time needed to run all the ADS momentum simulations and the CPU time needed for the sequential sampling algorithms.

The different sequential sampling experiments performed on the DFS structure are summarized in Table 4.3. The overall target accuracy was set to $\Delta = -50$ dB. The number of poles of the *root macromodels* were calculated using an error-based bottom-up approach (see Appendix A, section A.2 for more details). The initial number of poles for this bottom-up approach is selected as the smallest order of the *root macromodels* at the corners of the n-box design space. The accuracy of the final sampling grid is assessed using validation samples generated at the midpoint of each subspace of the final grid where the response from the EM solver is com-

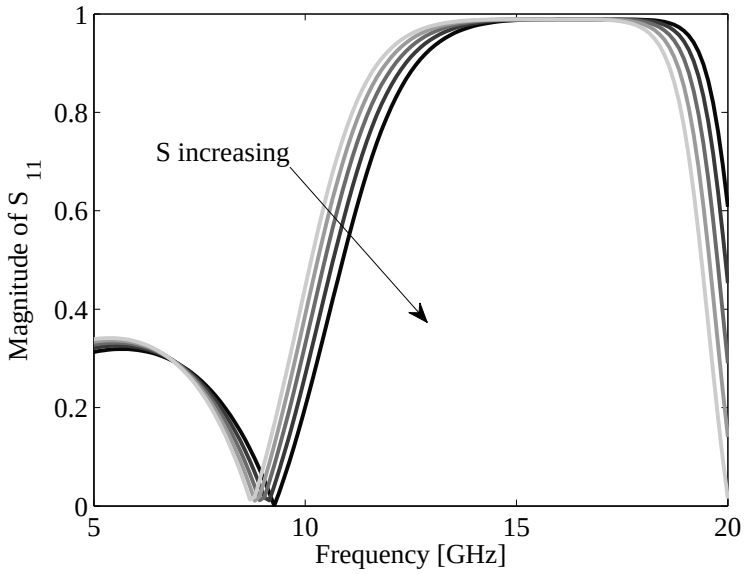

Figure 4.10: DFS Filter: Magnitude of S_{11} for $L = 1.75$ mm with parametric macromodel generated using Algorithm II.

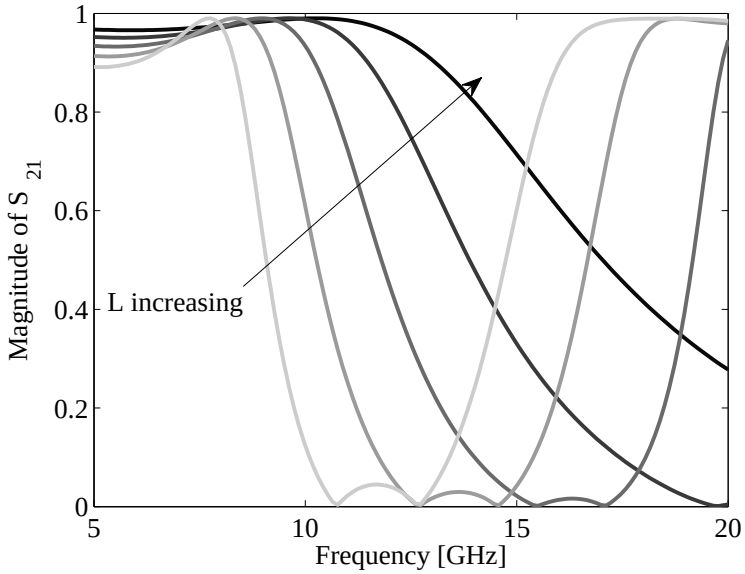

Figure 4.11: DFS Filter: Magnitude of S_{21} for $S = 0.20$ mm with parametric macromodel generated using Algorithm II.

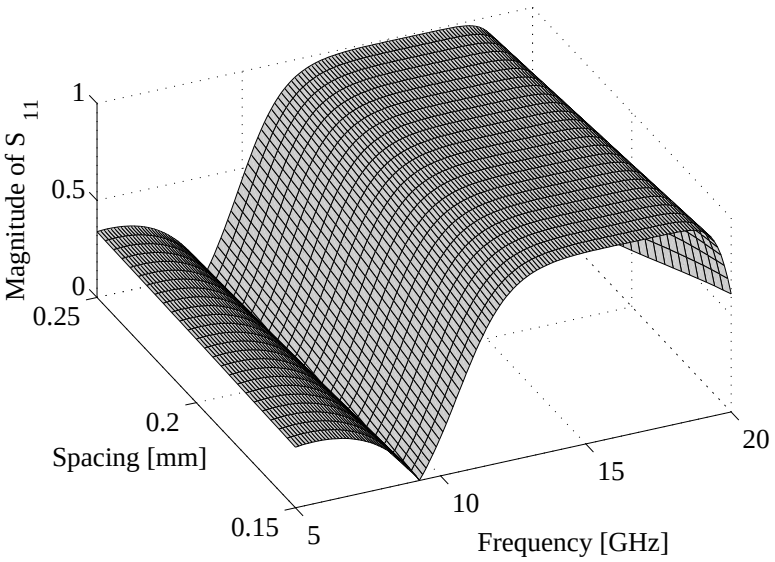

Figure 4.12: DFS Filter: Magnitude of S_{11} for $L = 1.75$ mm with parametric macromodel generated using Algorithm II.

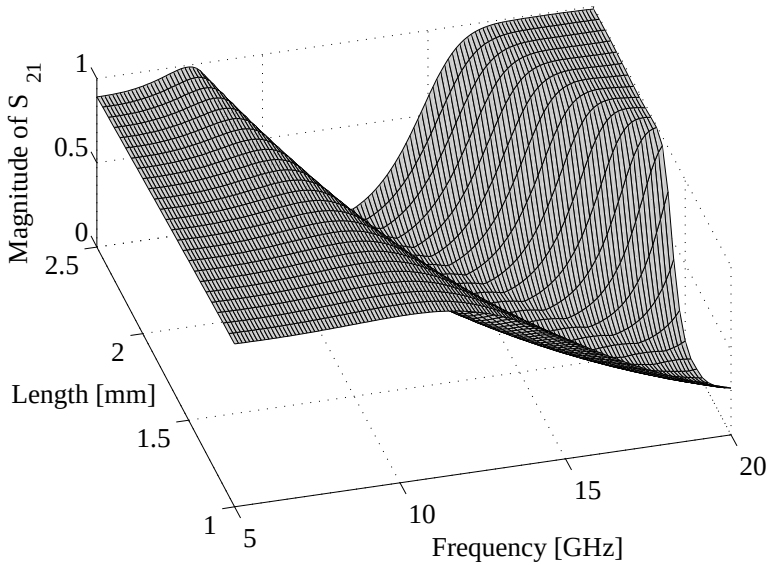

Figure 4.13: DFS Filter: Magnitude of S_{21} for $S = 0.20$ mm with parametric macromodel generated using Algorithm II.

Sampling method		Output data interpolation		Parametric macro-modeling [15]	
Sampling algorithm		I	II	I	II
Number of samples		1460	286	81	51
Accuracy (MAE) [dB]		-50.00	-50.01	-52.01	-50.25
CPU time [s]	Data generation	171363	56962	15908	10968
	Alg. execution	372	35	502	334

Table 4.3: DFS Filter: Comparison of different sampling strategies.

pared with the macromodel using the mean absolute error (4.2). Table 4.3 shows the results of the sequential sampling algorithms with some important parameters.

The following observations can be made from the tabulated results:

1) *Algorithm II* generates less number of points in comparison with *Algorithm I*. This is in accordance with the expectations of Section 4.3.

2) Using the parametric macromodeling technique of [15], the number of samples generated is very low in comparison with the approach where the raw frequency response data is interpolated as shown in Fig. 4.14. This indicates the high modeling capability of the parametric macromodeling method [15] and its effect on the sampling.

As a rule of thumb, a good choice is to use the *Algorithm II* (Section 4.3.2) to sample the design space such that the influence of each design parameter is taken into consideration and a good accuracy is achieved. The parametric macromodeling scheme described in [15] can be used to achieve a considerable gain in terms of the number of required EM simulations.

4.4.2 Hairpin bandpass microwave filter

A microwave hairpin bandpass filter on a substrate with relative permittivity $\epsilon_r = 9.9$ and a thickness of 0.635 mm is modeled in this example. The layout of this filter is shown in Fig. 4.15. Two spacings S_1 and S_2 and two lengths L_1 and L_2 are chosen as design parameters (see Fig. 4.15) in addition to frequency whose ranges are $S_1 \in [0.25, 0.35]$ mm, $S_2 \in [0.65, 0.75]$ mm, $L_1 \in [9.5, 14.5]$ mm, $L_2 \in [2.75, 3.25]$ mm and frequency $\in [1.5, 3.5]$ GHz.

Fig. 4.16 shows the parametric behavior of the magnitude of S_{11} as a function of S_1 and frequency, other values being kept at the mean value of the design space. Similarly, Fig. 4.17 shows the magnitude of S_{21} as a function of L_1 and frequency. Figs. 4.18 and 4.19 show the surface plots for the Figs. 4.16 and 4.17 respectively.

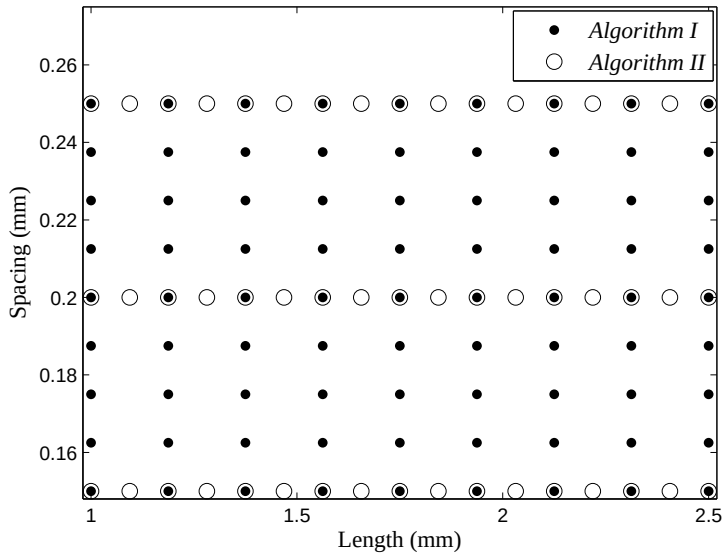

Figure 4.14: DFS Filter: Design space for Algorithm I *and* Algorithm II *using parametric macromodeling method [15].*

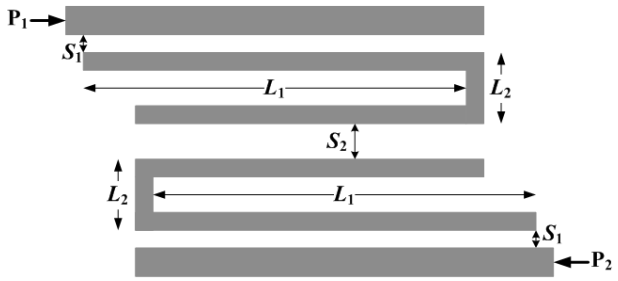

Figure 4.15: Layout of the microwave hairpin bandpass filter.

To model the hairpin filter, *Algorithm I* and *Algorithm II* are used along with the parametric macromodeling method presented in [15]. The MAE error measure (4.2) is used to assess the accuracy of the models generated with a target accuracy of -50 dB. This it resulted in 513 design space points for *Algorithm II*, with an achieved accuracy of -50.67 dB, whereas *Algorithm I* did not converge even with 13071 points. As evident from these results, *Algorithm I* samples all the parameters equally while *Algorithm II* selects the highly influential parameters carefully, requiring much fewer points. This difference between the two algorithms becomes severe with higher dimensions.

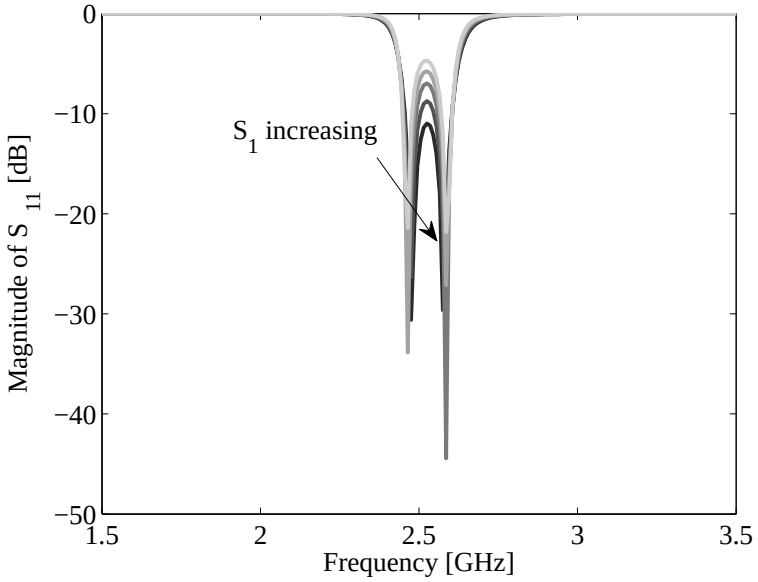

Figure 4.16: Hairpin Filter: Magnitude of S_{11} as a function of S_1 with parametric macromodel generated using Algorithm II.

Fig. 4.20 shows the design space points selected by *Algorithm II* respectively using a *parallel coordinates* plot [30]. In Fig. 4.20, the horizontal axis represents the four design parameters S_1, S_2, L_1 and L_2 and the vertical axis represents their normalized values. Also, the black dots represent the sample points selected for each design parameters and the gray dashed lines represent different design sample points. For instance, the bottom most horizontal line connecting the four black dots in Fig. 4.20 represent the design space point $[S_1, S_2, L_1, L_2] = [0, 0, 0, 0]$. In Fig. 4.20, the parameter L_1 has the maximum number of samples selected on its axis using *Algorithm II*, as it is the most dynamic parameter.

To illustrate the usefulness of the proposed method, the parametric macromodel generated with *Algorithm II* is used to optimize the bandpass filter. The specifications for the bandpass filter are given in terms of the scattering parameters S_{21} and S_{11}:

$$|S_{21}| > -2.5\,\text{dB} \quad \text{for} \quad 2.4\,\text{GHz} < freq < 2.5\,\text{GHz} \tag{4.5a}$$

$$|S_{11}| < -10\,\text{dB} \quad \text{for} \quad 2.4\,\text{GHz} < freq < 2.5\,\text{GHz} \tag{4.5b}$$

$$|S_{21}| < -40\,\text{dB} \quad \text{for} \quad freq < 1.7\,\text{GHz} \tag{4.5c}$$

$$|S_{21}| < -25\,\text{dB} \quad \text{for} \quad freq > 3.1\,\text{GHz}. \tag{4.5d}$$

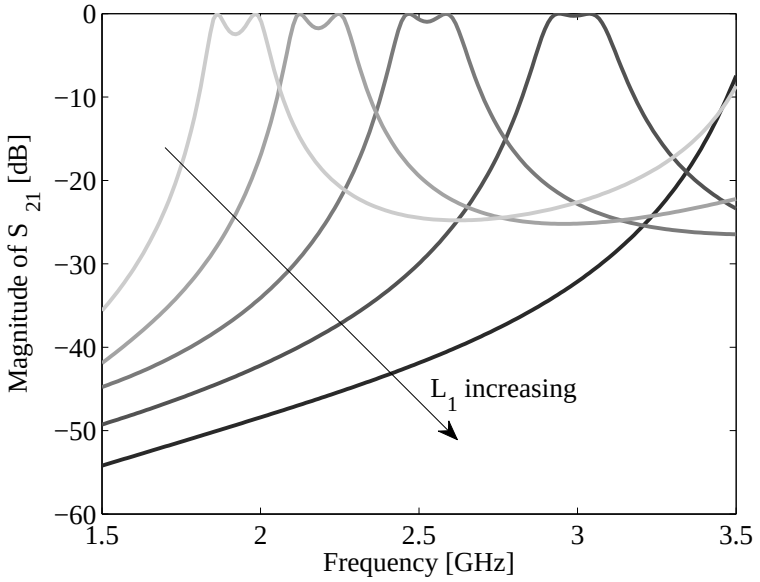

Figure 4.17: Hairpin Filter: Magnitude of S_{21} as a function of L_1 with parametric macromodel generated using Algorithm II.

Initial Design (S_1, S_2, L_1, L_2) [mm]	Optimal Design $(S_1^*, S_2^*, L_1^*, L_2^*)$ [mm]	# Function Evaluations	Optimal Cost	Processor Time [sec]
$[0.30, 0.70,$ $12.00, 3.00]$	$[0.27, 0.75,$ $12.10, 3.25]$	538	-8.4×10^{-4}	200.29
$[0.34, 0.69,$ $14.10, 2.90]$	$[0.28, 0.75,$ $12.19, 3.20]$	444	-6.3×10^{-4}	139.80
$[0.33, 0.68,$ $11.50, 3.20]$	$[0.28, 0.75,$ $12.10, 3.24]$	353	-10.9×10^{-4}	107.82

Table 4.4: Hairpin Filter: Optimization results.

The minimax optimization function *fminimax* in Matlab R2010a is used to perform an optimization with a cost function generated using the requirements of (4.5). Three optimization cases were considered with different starting conditions and are tabulated in Table 4.4. The S-response calculated using the parametric macromodel is supplied to the minimax optimization routine, resulting in optimal design space points $[S_1^*, S_2^*, L_1^*, L_2^*]$ which satisfy the constraints in all three cases.

Fig. 4.21 shows the magnitude of S_{21} for the second optimization case in

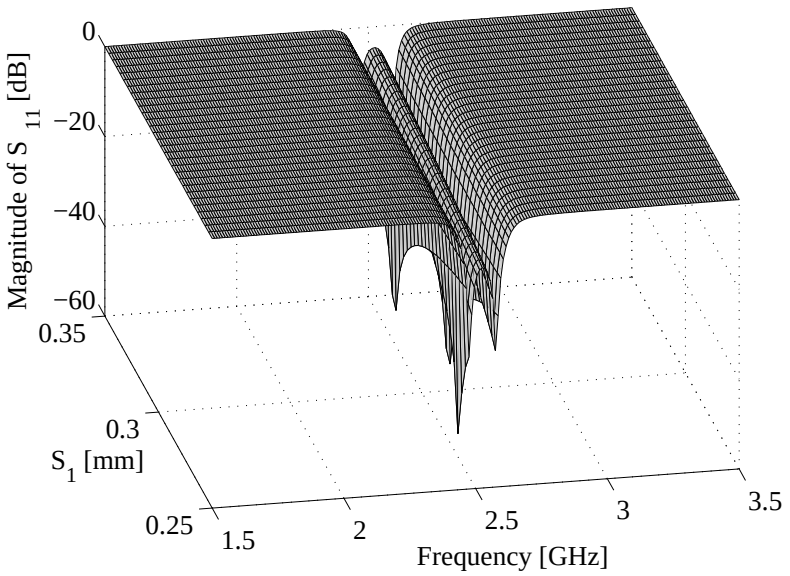

Figure 4.18: Hairpin Filter: Magnitude of S_{11} as a function of S_1 with parametric macromodel generated using Algorithm II.

the Table 4.4. The actual data generated by the ADS Momentum software and that obtained using the parametric macromodel at the optimum design space point $[S_1^*, S_2^*, L_1^*, L_2^*]$ are shown in Fig.4.21. As seen, both are in good agreement. The requirements (4.5) are shown by the thin black solid lines. Similar results are given for the magnitude of S_{11} in Fig. 4.22. As clearly seen, all the filter specifications are met for the optimal design point.

Since the cheap parametric macromodel is used in the minimax optimization, the CPU time per optimization is very small. This means that also global optimization schemes, which might require large number of function evaluations, can be used to avoid local optima without heavily increasing the CPU time. The generated parametric macromodel can also be used in other design activities such as sensitivity analysis, design space exploration, etc. Instead, if the ADS Momentum simulations had been used for the optimization, the CPU time would have been very high. For instance, in case of the hairpin filter structure a single frequency sweep with 31 frequency samples requires approximately 145 seconds in ADS Momentum and 0.29 seconds with the generated parametric macromodel with around 500 times speed-up.

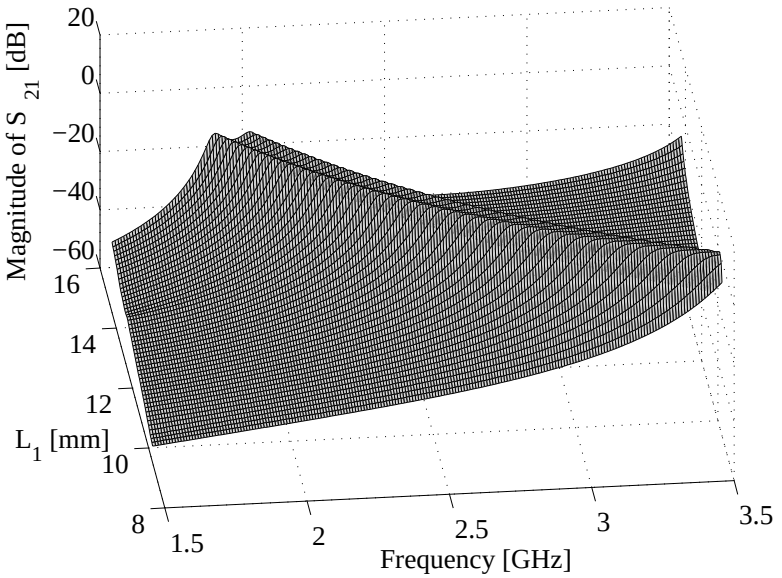

Figure 4.19: Hairpin Filter: Magnitude of S_{21} as a function of L_1 with parametric macromodel generated using Algorithm II.

4.5 Conclusions

We have presented two sequential sampling algorithms for the macromodeling of parameterized system responses in model-based sequential sampling frameworks. Two sequential sampling algorithms for the automated generation of parametric macromodels have been discussed and investigated. All the proposed techniques have been validated and compared on pertinent numerical examples in terms of the number of points needed to cover the complete design space, modeling accuracy and CPU time. Also, one of the proposed algorithms has been used in the generation of a parametric macromodel for a microwave filter example with four design parameters, and the generated parametric model is used in minimax optimization of the filter, validating the proposed sequential sampling method. However, it should be noted that these schemes work on grid-based design samples and hence susceptible to the curse of dimensionality when the number of design parameters increases.

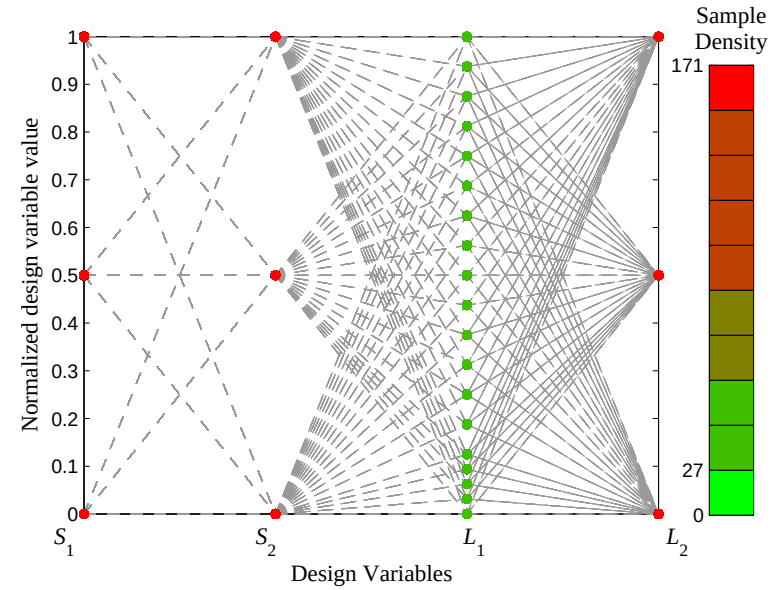

Figure 4.20: Hairpin Filter: Design space generated for Hairpin Filter using Algorithm II.

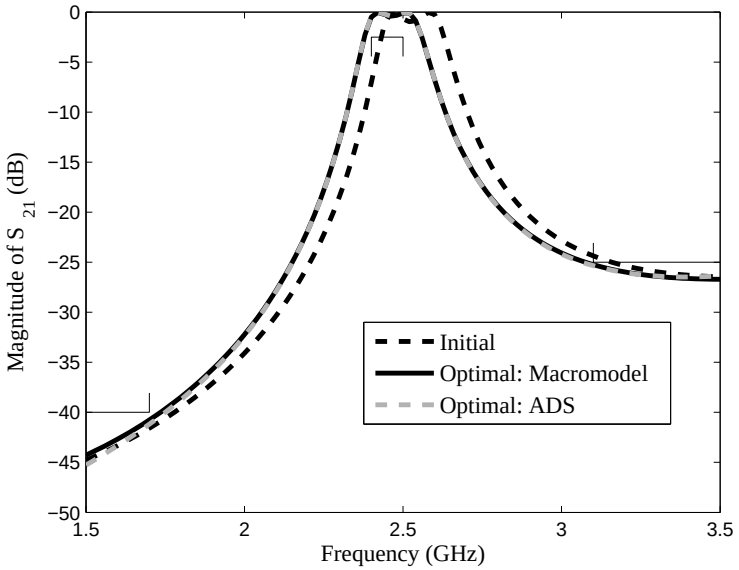

Figure 4.21: Hairpin Filter: Magnitude of S_{21} before and after optimization.

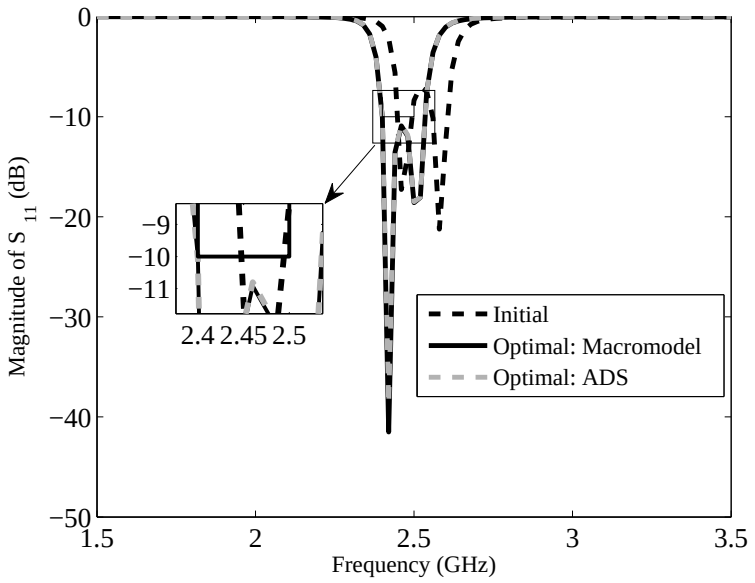

Figure 4.22: Hairpin Filter: Magnitude of S_{11} before and after optimization.

4.6 Acknowledgements

This work was supported by the Research Foundation Flanders (FWO) and by the Interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office.

References

[1] S. Peik, R. Mansour, and Y. Chow, "Multidimensional Cauchy method and adaptive sampling for an accurate microwave circuit modeling," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 46, no. 12, pp. 2364 –2371, Dec. 1998.

[2] A. Lamecki, P. Kozakowski, and M. Mrozowski, "Efficient implementation of the Cauchy method for automated CAD-model construction," *IEEE Microwave and Wireless Components Letters,*, vol. 13, no. 7, pp. 268 –270, July 2003.

[3] A. Cuyt, R. Lenin, S. Becuwe, and B. Verdonk, "Adaptive multivariate rational data fitting with applications in electromagnetics," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 54, no. 5, pp. 2265 – 2274, May 2006.

[4] V. Devabhaktuni, B. Chattaraj, M. Yagoub, and Q.-J. Zhang, "Advanced microwave modeling framework exploiting automatic model generation, knowledge neural networks, and space mapping," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 51, no. 7, pp. 1822 – 1833, July 2003.

[5] A. Lamecki, L. Balewski, and M. Mrozowski, "Adaptive CAD-model construction schemes," *IEEE Transactions on Magnetics,*, vol. 45, no. 3, pp. 1538 –1541, March 2009.

[6] P. Basl, R. Gohary, M. Bakr, and R. Mansour, "Modelling of electromagnetic responses using a robust multi-dimensional Cauchy interpolation technique," *IET Microwaves, Antennas Propagation,*, vol. 4, no. 11, pp. 1955 –1964, Nov. 2010.

[7] R. Lehmensiek and P. Meyer, "Creating accurate multivariate rational interpolation models of microwave circuits by using efficient adaptive sampling to minimize the number of computational electromagnetic analyses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 8, pp. 1419 –1430, Aug. 2001.

[8] D. Deschrijver, T. Dhaene, and D. De Zutter, "Robust parametric macromodeling using multivariate orthonormal vector fitting," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 7, pp. 1661 –1667, July 2008.

[9] P. Triverio, S. Grivet-Talocia, and M. Nakhla, "A parameterized macromodeling strategy with uniform stability test," *IEEE Transactions on Advanced Packaging*, vol. 32, no. 1, pp. 205 –215, Feb. 2009.

[10] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *IEEE Microwave and Wireless Component Letters*, vol. 19, no. 10, pp. 608–610, Oct. 2009.

[11] ——, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623–629, Aug. 2010.

[12] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *IEEE International Conference on Signal Propagation and Interconnects*, pp. 117 –120, May 2010.

[13] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[14] D. Deschrijver, K. Crombecq, H. Nguyen, and T. Dhaene, "Adaptive sampling algorithm for macromodeling of parameterized S-parameter responses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 1, pp. 39 –45, Jan. 2011.

[15] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139 –151, March/April 2012.

[16] D. Montgomery, *Design and analysis of experiments*, ser. Student solutions manual. Wiley, 2008. [Online]. Available: http://books.google.be/books?id=kMMJAm5bD34C

[17] K. Crombecq, E. Laermans, and T. Dhaene, "Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling," *European Journal of Operational Research*, vol. 214, no. 3, pp. 683 – 696, Nov. 2011.

[18] K. Crombecq, D. Gorissen, D. Deschrijver, and T. Dhaene, "A novel hybrid sequential design strategy for global surrogate modeling of computer experiments," *SIAM Journal on Scientific Computing*, vol. 33, no. 4, pp. 1948–1974, Aug. 2011.

[19] R. Rohrer and H. Nosrati, "Passivity considerations in stability studies of numerical integration algorithms," *IEEE Transactions on Circuits and Systems,*, vol. 28, no. 9, pp. 857 – 866, Sep. 1981.

[20] D. Deschrijver, A. Narbudowicz, E. Laermans, and T. Dhaene, "On the application of dimensional analysis to parametric macromodeling," *IEEE Microwave and Wireless Components Letters*, vol. 20, no. 4, pp. 190–192, 2010.

[21] D. Lewis, "Device model approximation using 2N trees," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,*, vol. 9, no. 1, pp. 30 –38, Jan. 1990.

[22] G. P. Fang, D. C. Yeh, D. Zweidinger, L. A. Arledge, and V. Gupta, "Fast, accurate MOS table model for circuit simulation using an unstructured grid and preserving monotonicity," in *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, ser. ASP-DAC '05. New York, NY, USA: ACM, 2005, pp. 1102–1106.

[23] X. R. Li and Z. Zhao, "Evaluation of estimation algorithms part I: incomprehensive measures of performance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4, pp. 1340 –1358, Oct. 2006.

[24] A. Weiser and S. E. Zarantonello, "A note on piecewise linear and multilinear table interpolation in many dimensions," *Mathematics of Computation*, vol. 50, no. 181, pp. 189–196, Jan. 1988.

[25] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," vol. 14, no. 3, pp. 1052–1061, July 1999.

[26] B. Gustavsen, "Improving the pole relocating properties of vector fitting," vol. 21, no. 3, pp. 1587–1592, July 2006.

[27] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 6, pp. 1587–1592, June 2008.

[28] B. Gustavsen and A. Semlyen, "Fast passivity assessment for S-parameter rational models via a half-size test matrix," vol. 56, no. 12, pp. 2701–2708, Dec. 2008.

[29] B. Gustavsen, "Fast passivity enforcement for S-parameter models by perturbation of residue matrix eigenvalues," vol. 33, no. 1, pp. 257–265, Feb. 2010.

[30] A. Inselberg, "FT-1 the plane with parallel coordinates," in *Parallel Coordinates*. Springer New York, 2009, pp. 49–61.

5

Efficient Design Cycle for Microwave Filters with Parametric Macromodels

K. Chemmangat, F. Ferranti, T. Dhaene and L. Knockaert

Electronics Letters, vol. 50, no. 6, pp. 475-476, 2014.

K. Chemmangat, M. Caenepeel, F. Ferranti, Y. Rolain, T. Dhaene and L. Knockaert

IEEE Transactions on Microwave Theory and Techniques, Submitted 2013

⋆ ⋆ ⋆

This chapter is divided into two parts. In the first part of this chapter, an enhanced parametric macromodeling scheme for linear high-frequency systems based on the use of multiple frequency scaling coefficients is presented along with a sequential sampling algorithm to fully automate the entire modeling process. The proposed method is applied on a ring resonator bandpass filter example and compared with another state-of-the-art macromodeling method to show its improved modeling capability and reduced setup time.

In the second part, the enhanced parametric macromodeling scheme along with the sequential sampling is used in the complete design cycle of a microwave filter. The design of transmission line (TL) microwave filters nowadays often boils down to a brute-force optimization of the design using electromagnetic (EM) simulations. The drawback of this approach is that it is numerically expensive -hence time consuming- and it does not provide any insight in the design. The parametric macromodeling scheme is used in the design process here instead. These models can be obtained in an automated way at a fraction of the numerical cost of the EM simulations and also can be used and re-used in successive designs. They also can assist the designer to get more insight in to the design. This proposed approach is illustrated here with a design example which is validated with measurements.

5.1 Parametric Macromodeling of Linear High-Frequency Systems using Multiple Frequency Scaling and Sequential Sampling

5.1.1 Introduction

Design activities of electromagnetic (EM) systems such as design space exploration, optimization, sensitivity analysis, etc., often require a substantial number of computationally expensive EM simulations. The development of parametric macromodels acting as accurate and efficient surrogate models for complex EM systems is an active field of research [1–5]. These models tend to be good approximations of the EM system behavior, characterized by frequency and additional design parameters (such as geometrical or substrate features) and can be used to speed-up the design process. Robust interpolation-based parametric macromodeling methods have been proposed over the recent years, based on the parameterization of a set of frequency-dependent rational models called *root macromodels* [1, 2, 5]. In [5], interpolation of *root macromodels* at the input-output level, based on two scaling coefficients was presented : one of the coefficients is a multiplicative factor at the input/output level of the system (amplitude scaling) and the other coefficient is a compression or expansion term for the Laplace variable s (frequency scaling). The approach of [5] results in high modeling capability and

robustness.

In this section, the parametric macromodeling method proposed in [5] is generalized by using multiple frequency scaling coefficients for all partial fractions of the *root macromodels*. This allows to model the behavior of the partial fractions of the *root macromodels* independently, in order to achieve a more flexible modeling capability. The proposed method is compared with the approach described in [5] to show its enhanced modeling capability and reduced CPU setup time.

5.1.2 Proposed Parametric Macromodeling Method

A brief recast of the parametric macromodeling process is presented, by which a parameterized system response of the microwave filter is generated with respect to the design parameters, similarly to [5, 6]. The first step of the macromodeling process is to generate a set of parameter-dependent frequency-domain responses, which we will call the set of multivariate data samples $\{(s_n, \vec{g}_k), \mathbf{H}(s_n, \vec{g}_k)\}$, $n = 1, \ldots, N_s$, $k = 1, \ldots, K^{\text{tot}}$. This set depends on the complex frequency $s = j\omega$ and additional N design parameters $\vec{g} = (g^{(1)}, \ldots, g^{(N)})$. For the specific filter case, these design parameters describe the geometry of the system and/or the properties of the substrate that one can vary during the design. An efficient sampling algorithm [7] is used to gather data samples located at maximally informative positions, i.e. spots in the design space where the response changes rapidly. Both the parametric macromodeling method and the sequential sampling algorithm are coupled to provide an automated modeling process. N dimensional hyperrectangular (N-box) regions are used as building blocks for the design space. The design space consists of the concatenation of several of such regions which are denoted as Ω_l, $l = 1, \ldots, L$. Each of these Ω_l contains 2^N frequency-dependent rational models called *root macromodels* at the corresponding corner points. Hence each bounding corner has a *root macromodel* of the N dimension hyperrectangular region. This corresponds to 2^N rational models. This means that the complex Laplace frequency s is not considered as an element of the design space. It is modeled separately using *root macromodels*. These are rational forms of the Laplace variable also known as frequency response functions (FRFs) which are identified from the data samples using the well known Vector Fitting (VF) identification technique [8].

The rational *root macromodels* $\mathbf{R}^{\Omega_l}(s, \vec{g}_i^{\,\Omega_l})$, $i = 1, \ldots, 2^N$ contained in the N-box region Ω_l are represented in a pole-residue form:

$$\mathbf{R}^{\Omega_l}(s, \vec{g}_i^{\,\Omega_l}) = \sum_{p=1}^{P_i^{\Omega_l}} \frac{C_{p,i}^{\Omega_l}}{s - a_{p,i}^{\Omega_l}} + D_i^{\Omega_l} \quad ; p = 1, \ldots, P_i^{\Omega_l} \tag{5.1}$$

where $C_{p,i}^{\Omega_l}$ represents the residue matrices, $a_{p,i}^{\Omega_l}$ denotes the poles $P_i^{\Omega_l}$ and $D_i^{\Omega_l}$ is the direct-term matrix.

Later, the parametric macromodeling technique is applied to these N-box regions Ω_l. A two parameter description is presented here for clarity and ease of notation, even though the method is general for any dimension N of the design space. The design space region Ω_l is defined by four bounding corners $\vec{g}_1{}^{\Omega_l} = (g_1^1, g_2^1)$, $\vec{g}_2{}^{\Omega_l} = (g_1^2, g_2^1)$, $\vec{g}_3{}^{\Omega_l} = (g_1^1, g_2^2)$, and $\vec{g}_4{}^{\Omega_l} = (g_1^2, g_2^2)$ as in Fig. 5.1. Each corner possesses a different *root macromodels* $\mathbf{R}^{\Omega_l}(s, \vec{g}_i)$, $i = 1, \ldots, 4$. We will discuss the interpolation of the *root macromodels* next. For simplicity and ease of notation we omit the superscript Ω_l. In [5, 6], one amplitude scaling and one frequency

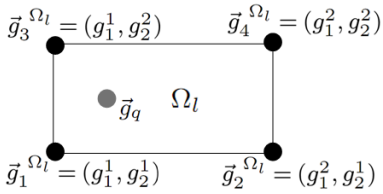

Figure 5.1: A two dimensional design space with four root macromodels.

scaling coefficient (α_1, α_2) are calculated using the optimization

$$(\alpha_{1,ij}^*, \alpha_{2,ij}^*) = \underset{(\alpha_{1,ij}, \alpha_{2,ij})}{\operatorname{argmin}} \left[\operatorname{Err}(\tilde{\mathbf{R}}^j(s, \vec{g}_i), \mathbf{H}(s, \vec{g}_j)) \right]. \tag{5.2}$$

In (5.2), $\tilde{\mathbf{R}}^j(s, \vec{g}_i) = \alpha_{1,ij} \mathbf{R}(s\alpha_{2,ij}, \vec{g}_i)$, is the scaled response of $\mathbf{R}(s, \vec{g}_i)$ obtained to match $\mathbf{R}(s, \vec{g}_j)$ and $\operatorname{Err}(\cdot)$ is a suitable error measure between the two responses. The Mean Absolute Error (MAE) measure or the L_1-norm per port is used to assess the accuracy of the model in every N-box region of the design space:

$$E^{\mathrm{MAE}}(\vec{g}) = \max_{\substack{u=1,\ldots,P \\ v=1,\ldots,P}} \frac{1}{N_s} \left(\sum_{n=1}^{N_s} |R_{u,v}(s_n, \vec{g}) - H_{u,v}(s_n, \vec{g})| \right). \tag{5.3}$$

The method compares the EM simulation response $H_{u,v}(s, \vec{g})$ with the parametric macromodel response $R_{u,v}(s, \vec{g})$, where P is the number of system ports. The MAE error measure or the L_1-norm gives a global view on the error between the two frequency responses and hence it is preferred over other error measures such as the L_∞-norm that is known to be not very smooth.

The calculated frequency scaling coefficients $\alpha_{2,ij}$ are further refined and improved by defining a separate frequency scaling coefficient $\beta_{p,ij}$ for every term p in the rational model in a pole-residue form (5.1). The modified $\tilde{\mathbf{R}}^j(s, \vec{g}_i)$ is given by:

$$\tilde{\mathbf{R}}^j(s, \vec{g}_i) = \alpha_{1,ij} \sum_{p=1}^{P_i} \frac{C_{p,i}}{s\beta_{p,ij} - a_{p,i}} + D_i \tag{5.4}$$

The optimal $\beta_{p,ij}$, $p = 1, \ldots, P_i$ are found by performing an optimization step similar to (5.2). The $\alpha^*_{2,ij}$ obtained from (5.2) are used as an initial value for all $\beta_{p,ij}$. For the complex-conjugate pole pairs (p_1, p_2) the coefficients satisfy $\beta_{p_1,ij} = \beta_{p_2,ij}$ to preserve the symmetry. This also reduces the number of coefficients to be optimized. The evaluation of the model taken at a generic point $\vec{g}_q$ in the design space (Fig. 5.1) is done similarly to [5, 6] as:

i For each *root macromodel* $\mathbf{R}(s, \vec{g}_i)$, $i = 1, \ldots, 2^N$, the amplitude scaling coefficient $\alpha_{1,ij}$ and frequency scaling coefficients $\beta_{p,ij}$, $p = 1, \ldots, P_i$ are interpolated using a multilinear interpolation [9] over $\vec{g}$ at the point $\vec{g}_q$ to find $\alpha_{1,iq}$ and $\beta_{p,iq}$, $p = 1, 2, ...P_i$. This results in the modified *root macromodels*, $\tilde{\mathbf{R}}_q(s, \vec{g}_i) = \alpha_{1,iq} \sum_{p=1}^{P_i} \frac{C_{p,i}}{s\beta_{p,iq} - a_{p,i}} + D_i$ at $\vec{g}_q$,

ii Then the models $\tilde{\mathbf{R}}_q(s, \vec{g}_i)$ are interpolated using the multilinear interpolation [9] over $\vec{g}$ to get the final model $\mathbf{R}(s, \vec{g}_q)$ at the point $\vec{g}_q$.

This parametric macromodeling approach is performed for each region Ω_l to cover the complete design space.

5.1.3 Sequential Sampling Algorithm

This section briefly describes the sequential sampling algorithm used in this work to scan the design space in an automated way. Fig. 5.2 shows the flowchart of the algorithm and the different steps are discussed below:

1. *Initialization*: During this step the design space is defined. It contains N design parameters of interest $\vec{g} = (g^{(1)}, \ldots, g^{(N)})$. The initial design space is defined by the 2^N corner points to form one single N-box region Ω_l with $l = L = 1$.

2. *Parametric macromodeling* : For each elementary region Ω_l, $l = 1, \ldots, L$, a parametric macromodel $\mathbf{R}^{\Omega_l}(s, \vec{g})$ is built with the 2^N corner points as *root macromodels* as explained in Section 5.1.2.

3. *Model validation*: The selected region Ω_l is checked with respect to the actual EM solver. This requires a subset of EM simulations that are not used during the estimation. This is done in two steps: first the EM solver is compared with the macromodel using the MAE measure (5.3) at the center of the maximum sensitive edge (which is the most difficult-to-model-edge such that the optimization of (5.2) returns the highest error, i.e., in some sense the design parameter which is the most difficult to parameterize) and if found to be accurate, a second level of check is performed at the geometric center (as done similarly to [7]). Only then the decision is made whether the region Ω_l is accurate or not.

4. *Refinement*: If the accuracy of the region Ω_l lies within the desired threshold Δ, it is not divided, else the region is split into two sub regions along the maximum sensitive edge [7] as shown in Fig. 5.2. The accuracy threshold Δ can be decided based on the application of the scheme. For example, considering microwave filters, if a passband requirement of -30 dB is required, the parametric macromodel should be able to describe the filter characteristics up to an accuracy of -30 dB.

Then after updating the number of regions the algorithm is repeated from Step 2 until all regions Ω_l are covered, i.e., $l = L$.

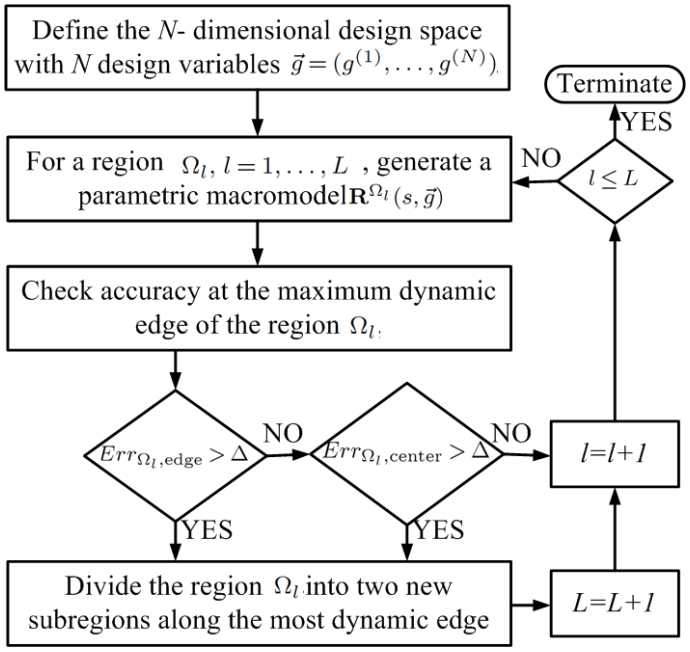

Figure 5.2: Flowchart of the sequential sampling algorithm.

5.1.4 Numerical Example

The scattering parameter response of a ring resonator bandpass filter (see Fig. 5.3) has been modeled. ADS Momentum[1] has been used as EM solver. The substrate has a relative permittivity $\epsilon_r = 4.32$, a loss tangent $\delta = 0.002$ and a thickness equal to 1.52 mm. The lengths $L_1 \in [20.0, 23.0]$ mm, $L_2 \in [20.0, 23.0]$ mm and the spacing $S_1 \in [0.05, 0.3]$ mm (see Fig. 5.3) are chosen as three design parameters in addition to frequency $\in [1.0, 3.0]$ GHz. The parametric behavior

[1] Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

of the filter is shown in Fig. 5.4. For the sequential sampling algorithm the

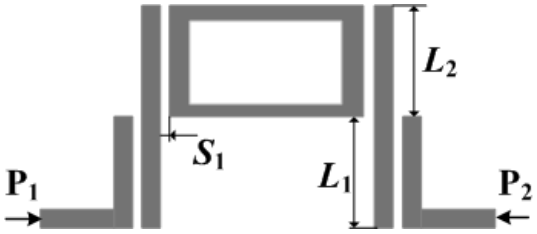

Figure 5.3: Layout of the ring resonator bandpass filter.

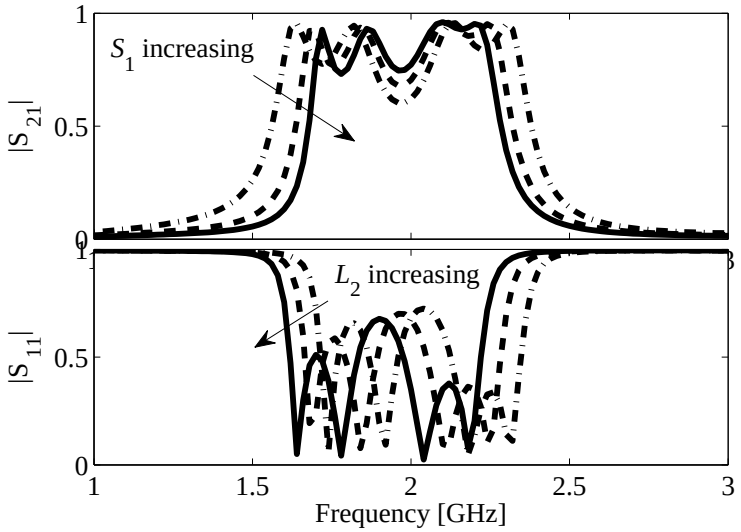

Figure 5.4: Magnitude of $\mathbf{S}_{21}$ and $\mathbf{S}_{11}$ as a function of S_1 and L_2.

Mean Absolute Error (MAE) measure per port is used to assess the accuracy of the parametric macromodel in every N-box region of the design space:

$$E^{\mathrm{MAE}}(\vec{g}) = \max_{\substack{u=1,\ldots,P_{in} \\ v=1,\ldots,P_{out}}} \frac{1}{N_s} \left(\sum_{n=1}^{N_s} |R_{u,v}(s_n, \vec{g}) - H_{u,v}(s_n, \vec{g})| \right). \qquad (5.5)$$

where the EM simulation response $H_{u,v}(s, \vec{g})$ is compared with the parametric macromodel response $R_{u,v}(s, \vec{g})$, and P_{in} and P_{out} are the number of input and output ports, respectively. The new parametric macromodeling method as well as the recent method in [5] have been used along with the grid-based sequential sampling algorithm [7] to build accurate parametric macromodels. The MAE measure

(5.5) was kept at a target accuracy of $\Delta = -40$ dB for the sequential sampling. Some comparison results are shown in Table 5.1. The total number of generation (Gen.) and validation (Val.) points are shown in the table. As it can be seen, the new method needs significantly less data samples for the same accuracy target, proving its improved modeling capability. The overall CPU time to build the corresponding model is consequently reduced.

Method	# Samples		CPU Time		Accuracy
	Gen.	Val.	Modeling	Data Gen.	[dB]
proposed	32	18	6 m 56 s	36 m 41 s	-40.4
[5]	116	95	22 m 30 s	2 h 31 m 29 s	-40.1

Table 5.1: Comparison: proposed method versus the method of [5]

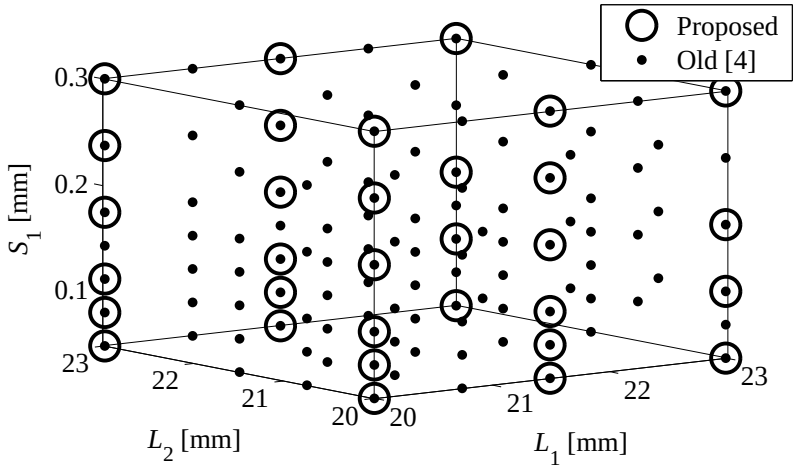

Figure 5.5: Model generation points in the design space

Fig. 5.5 shows the distribution of design space points selected using the two parametric macromodeling methods. The method [5] finds more difficult to model the filter behavior with respect to the design parameters and needs significantly larger number of points along every design space dimension. Fig. 5.6 compares the magnitude of $\mathbf{S}_{21}$ between the EM solver and the proposed parametric macromodel for three random validation points. The different responses show a very good agreement.

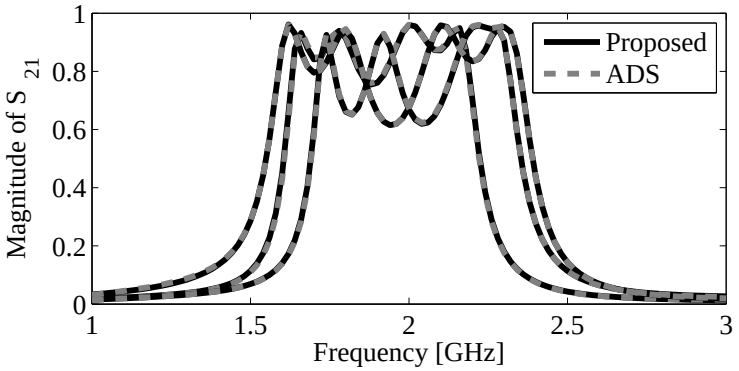

Figure 5.6: Magnitude of $\mathbf{S}_{21}$ *at three random validation points.*

5.2 Efficient Design Cycle for Microwave Filters using Parametric macromodels

5.2.1 Introduction

The design of microwave filters remains important for both research and applications. Several methods have been proposed to address the design of distributed filters [10, 11]. As the design process relies on idealized components and behavior, most designs require a post-design fine-tuning to meet the desired specifications. This tuning process involves optimization based on multiple electromagnetic (EM) simulations using accurate EM solvers. This accuracy comes at a price however, these solvers are known to be numerically expensive and hence time-consuming. Moreover, this brute-force optimization does not provide the designer with any insight about the behavior of the non-idealities. Even though this results in accurate designs, the process is typically time consuming and expensive. To speed up this process, the EM solver can be replaced by a computationally efficient model. In this chapter, parametric macromodels are used as a good compromise between model accuracy and complexity. Generating parametric macromodels to represent the parameterized response of microwave systems with respect to different geometrical variables and material properties is an active field of research [1–6, 12–22].

The two main advantages of using such an approach are:

1. The parametric macromodels replace the expensive EM solver to evaluate the filter response as a function of the design parameters of interest (e.g. geometrical parameters) for certain parameter ranges. Therefore, these parametric macromodels can be used in different optimization scenarios where changes of the specifications of the filter (e.g. the bandwidth of interest, the

selectivity, etc.), need to be examined.

2. The parametric macromodels can also be used in other computationally expensive design activities, such as design variability analysis that is a key factor for circuit reliability or design space exploration that leads to an intuitive understanding of the circuit's behavior apart from the design optimization. All these analyses are obtained without much additional computational effort.

Even though the extraction process of parametric macromodels is automated using adaptive sampling approaches [7], it still requires information from the designer. The designer must choose the ranges of the variable design parameters, for which the model is built. This means that the models must be extracted during the design process itself.

Including these parametric macromodels early in the design flow can significantly reduce the overall design cycle time. Considering design optimization of microwave filters, one such method is the Space Mapping (SM) technique which locates an optimum given a user-defined performance measure [23–25]. The main idea in this method is to reduce the computational complexity of optimization by reducing the number of expensive EM *fine* model simulations and using a *coarse* model as a SM surrogate for the output space of the EM system. The *coarse* model is used to drive the optimization and it is corrected for any possible mismatch with respect to the *fine* model so that an optimum is found with less computational cost [23–25]. Surrogate-Based Optimization (SBO) schemes which model the performance space of the EM systems are also an active field of research [26, 27]. These methods accurately model a scalar performance measure function at the regions of interest, thereby finding an optimum.

In this chapter, we incorporate the parametric macromodels not only for the design optimization itself but for the complete design flow, where multiple design optimization and variability analysis steps are performed. This distinguishes the proposed method from other methods [23–27] that aim at optimizing a particular performance measure, which leads to restart of the modeling step for optimization each time the specifications are changed. The parametric macromodels accurately and efficiently describe the parameterized behavior of EM systems with respect to frequency and several other design parameters, such as layout parameters and substrate features. Since a direct scanning or exploration of the design space using the EM solver becomes very time consuming with an increasing number of parameters, the use of an intermediate parametric macromodel is a very efficient alternative. A hairpin filter is used in this chapter to validate the proposed design approach. This filter is a space-saving alternative of the well known parallel-coupled transmission-line-resonator filter. Design equations for the latter type of filter exist and can be used to obtain an initial design for the hairpin filter. Nev-

ertheless optimization and variability analysis are needed to obtain a design that fulfills the desired specifications. This makes the design of a hairpin filter a good example to show the advantages of the macromodels over the conventional EM solver-based modeling approach.

5.2.2 Parametric Macromodels for Microwave Filters

The parametric macromodeling method is described in detail in Section 5.1.2 along with the automatic design space scanning using sequential sampling described in Section 5.1.3 are used to build the parametric macromodel for the filter under consideration. Once an efficient and accurate parametric macromodel is generated, it is used in the design process.

5.2.3 Design process

This section illustrates the macromodeling-based design procedure using as an example the design of a parallel-coupled hairpin resonator filter [28] consisting of three hairpin resonators (see Fig. 5.9). Such a filter is a space-saving alternative for the well-known parallel-coupled transmission-line-resonator (TL) filter [29]. This hairpin-line filter is a good example to illustrate macromodel-based design. On the one hand design equations exist to generate a filter for an approximative geometry replacing the hairpin by the conventional $\frac{\lambda}{2}$ TL resonator. On the other hand, optimization of the geometry is needed to compensate for the approximation error and obtain a solution that fulfills the specifications. In the current state of the art, the latter often uses an iterative optimization loop based on a succession of computationally expensive EM simulations. We use a macromodel not only for the optimization of the design, but also for the design sensitivity analysis that leads to a variability assessment of the performance for reliability evaluation.

5.2.3.1 Overview of the design process and performance assessment

Fig. 5.7 synthesizes the design flow that is used here. The design starts from a user-specified frequency response template. In the preliminary design step, we design a parallel-coupled TL that meets the template specifications using design equations given in [29, 30]. Next, we define the design space. Therefore, we determine the variable and the fixed geometric design parameters to be used for the hairpin resonator filter. Then, we generate an approximative starting geometry for the hairpin resonator filter and eventually we determine the ranges of the design parameters in the design space. Next, we build a parametric macromodel of the hairpin resonator filter for the chosen design parameters and ranges. The macromodel is used to optimize the approximative starting geometry to match the desired response and to perform a design variability analysis. If the optimized solution is

robust enough to fabrication process variation, the filter is realized and measured with a Vectorial Network Analyzer of type E8364B (PNA series) that is calibrated using the electronic Ecal module.

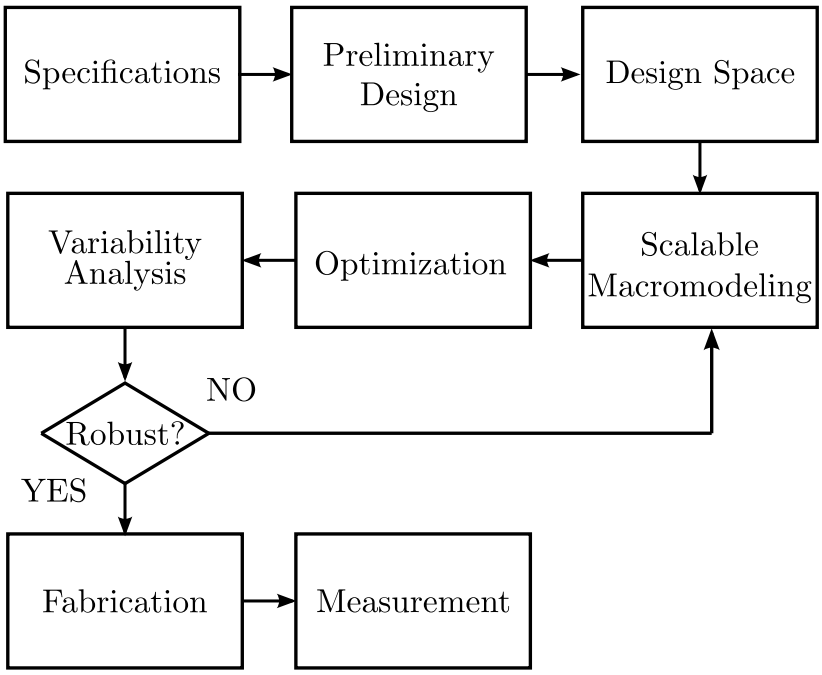

Figure 5.7: Schematic overview of the design process.

5.2.3.2 Specifications

The specifications of a filter are expressed by a frequency template (Fig. 5.8). The specifications given on the S-parameters of the filter are:

$$|S_{21}| > -L_{A,max}(dB), \qquad f_{p1} \leq freq \leq f_{p2}, \tag{5.6a}$$
$$|S_{11}| < -L_{R,min}(dB), \qquad f_{p1} \leq freq \leq f_{p2}, \tag{5.6b}$$
$$|S_{21}| < -L_{A,stop1}(dB), \qquad freq < f_{s1}, \tag{5.6c}$$
$$|S_{21}| < -L_{A,stop2}(dB), \qquad freq > f_{s2} \tag{5.6d}$$

where $L_{A,max}$ is the maximal insertion loss in the passband, $L_{A,stop1}$ and $L_{A,stop2}$ are the minimal attenuation in the stop bands. We also put specifications on S_{11}, since some filters require a limitation on the return loss. $L_{R,min}$ is the

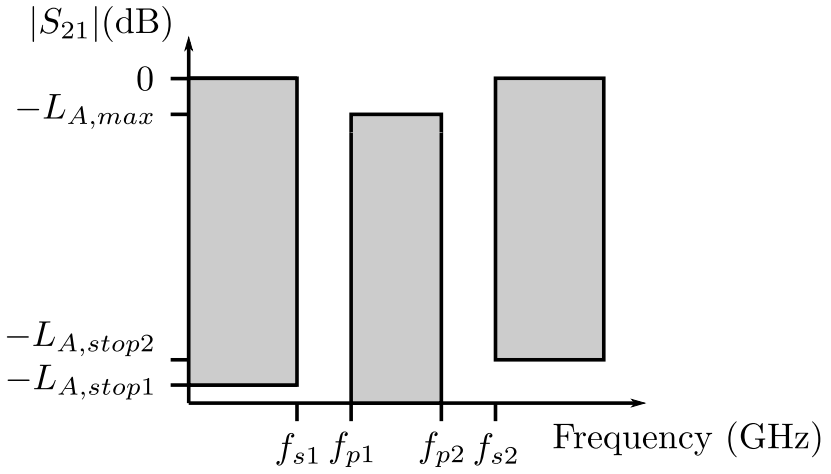

Figure 5.8: Specifications on S_{21}.

minimal return loss in the passband. f_{p1} and f_{p2} are the passband corner frequencies (f_c is the center frequency) and f_{s1} and f_{s2} the stop band corner frequencies. In general $L_{A,stop1} = L_{A,stop2}$, but we allow the stopband losses to be different.

5.2.3.3 Preliminary Design

The first step in the preliminary design process is to find a rational approximation that fulfills the demands of the frequency template. Next a synthesis is performed to obtain a lumped-element normalized lowpass prototype network. Eventually an equivalence relation [29, 31] between the lumped-element network and the parallel-coupled TL filter is used to physically dimension the distributed microwave filter. This last step can briefly be summarized as follows:

- The ratio between the width of the TL and the thickness of the dielectric h determines the characteristic impedance Z_0 of the TL.
- The length of the TL resonators determines the center frequency f_c of the filter in first order and it should be chosen equal to half the wavelength $\frac{\lambda_c}{2}$.
- The value of the spacing between the lines determines the coupling between the lines.

These design equations only hold for the parallel-coupled TL resonator filter. They act as approximations for the hairpin line filter.

5.2.3.4 Design Space

We determine the design space over which we build the design space in three steps. First we determine the design parameters, next we generate initial values for these parameters and finally we choose the ranges for these parameters.

Choice of the design parameters

The layout of a hairpin resonator filter shows the geometrical parameters (Fig. 5.9) :

- W: the width of the lines
- h: the height of the substrate
- D: the length of the legs of the hairpin
- S_h: the spacing between the legs of the hairpin
- S_1: the spacing between the feed line and the resonators
- S_2: the spacing between the resonators

We want to minimize the number of variable design parameters to reduce the number of dimensions of the design space. Some geometrical parameters are known to be related and will therefore be grouped:

- $\frac{W}{h}$ determines the characteristic impedance of the lines.
- D and S_h determine the total length L_{total} of the line and therefore the resonance frequency of the resonator. Instead of allowing both parameters to be variable, we fix S_h and only allow D to vary.
- The remaining geometrical parameters S_1 and S_2 can not be grouped, since they determine the coupling between different lines.

Since $\frac{W}{h}$ is not affected by the replacement of the TL resonator by a hairpin resonator, we will use the same design equation to determine it. This leaves us with 3 variable geometrical design parameters: D, S_1 and S_2.

Starting geometry

Now that we have parameterized the structure, we generate a starting geometry for it. First we determine values for the fixed geometric parameters: W, h and S_h next D, S_1 and S_2:

- We calculate $\frac{W}{h}$ to obtain a desired Z_0 using empirical laws [32].

- On the one hand, S_h must be large enough such that there is not too much coupling between the legs of the hairpin. On the other hand it can not be too large either, as otherwise the length D over which the hairpins are coupled becomes too small. In the numerical example (section 5.2.4), we therefore choose it equal to 1.307 mm for an approximate total length of 42 mm.

- We initialize D such that the total length of the hairpin is $\frac{\lambda_c}{2}$: $D_0 = \frac{\frac{\lambda_c}{2} - S_h}{2}$

- The lengths over which the straight TL and the hairpin resonators are coupled differ, so we rescale the spacings found for the parallel-coupled TL filter to compensate for this difference in length: $S_{1,hairpin} = S_{1,TL} \frac{D}{\frac{\lambda_c}{2}}$

Ranges of the design space

The ranges of the design space are the boundary values of the variable design parameters. They must be chosen such that the design space contains the initial design point and most (preferably all) design points that satisfy the constraints. Therefore these ranges must be wide enough. Nevertheless we have to avoid to model for values that are physically impractical (due to the fabrication process) or that lead to irrelevant S-parameters (e.g. too much insertion loss). The ranges for the design parameters D, S_1, S_2 are chosen as follows:

- We choose the range of D such that the initial value for D_0 and $\frac{\lambda_c}{4}$ are within it. Since we a priori do not know how the folding of the TL resonator affects the resonance frequency, we consider a range wide enough to make sure that the desired resonance frequency is certainly captured. The range of D is chosen such that $\frac{l_{total}}{\frac{\lambda_c}{2}} = [0.95 : 1.05]$.

- The initial value of S_1 is very close to the minimal distance that is physically feasible, therefore this minimal distance is chosen as the lower boundary. The upper boundary is chosen large enough to allow one to check whether there are also solutions in the lower coupling region of the design space. However it can not become too high, because then almost no energy is coupled from the feedline to the first resonator. The range of S_1 is chosen such that the normalized spacing is approximately $\frac{S_1}{h} = [0.16 : 1.75]$.

- The range of S_2 is chosen wide enough to capture as many desired solutions as possible. The higher value is chosen not to be too large, otherwise there is too much insertion loss. The range of S_2 is chosen such that the normalized spacing is approximately $\frac{S_2}{h} = [0.9 : 2.6]$.

5.2.3.5 Generate the Parametric Macromodel

Section 5.2.2 shows that the generation of the parametric model is automated. Nevertheless the user has to specify the design space, the frequency span and the accuracy for the model. The choice of the design space is already discussed in subsection 5.2.3.4. The frequency span is chosen wide enough such that f_{s1} and f_{s2} are within it. The choice of the accuracy is based on the minimal attenuation desired in the stopband. In the numerical example this is -30 dB. The accuracy of the model is therefore set to -40 dB. Next the macromodel is generated as is explained in Section 5.2.2. We denote the macromodel as $\mathbf{R}(s, \vec{g})$, where s is the Laplace variable and where $\vec{g} = [S_1, S_2, D]$.

5.2.3.6 Optimization

The parametric macromodel of the microwave filter $\mathbf{R}(s, \vec{g})$ is used to optimize the initial design such that it fulfills the specifications. The global optimization function *GlobalSearch* in Matlab R2012a[2] is used to perform an optimization with a cost function generated using the requirements of (5.6). The *GlobalSearch* routine first generates scattered points in the design space with a scatter search. Starting from these points, several constraint nonlinear optimizations using a gradient-based local nonlinear optimizer are performed, generating multiple solutions. The function then ranks the solutions in terms of their values in ascending order. More details of the method can be found in [33].

Considering microwave systems, a typical optimization process begins by defining passband and stopband specifications in terms of the S-parameter responses, which are reformulated in the form of a cost function $F(s_m, \vec{g})$, at optimization frequency samples s_m, $m = 1, 2, ..N_s^{\mathrm{OP}}$ to be minimized:

$$F(s_m, \vec{g}) = R_{\mathrm{L}}^m - R(s_m, \vec{g}) \;\text{ or }\; R(s_m, \vec{g}) - R_{\mathrm{U}}^m. \tag{5.7}$$

In (5.7), R_{L}^m and R_{U}^m represents the lower and upper frequency response thresholds, respectively, at frequency samples s_m, $m = 1, \ldots, N_s^{\mathrm{OP}}$, spread over the frequency range of interest. Also note in (5.7) shows that the optimization frequency samples s_m, $m = 1, \ldots, N_s^{\mathrm{OP}}$ can be different from the modeling frequency samples s_n, $n = 1, \ldots, N_s$ described in Section 5.1.2. The frequency samples s_n are generated by the EM solver and hence very expensive whereas the optimization frequency samples s_m are generated from the parametric macromodel $\mathbf{R}(s, \vec{g})$ which is cheap to evaluate. This means that, for the optimization case, more frequency samples could be selected to have a better estimation of the cost function $F(s_m, \vec{g})$ (i.e., $N_s^{\mathrm{OP}} > N_s$).

For the specifications of the filter given in (5.6), $-L_{A,max}$ is a lower bound R_{L}^m and $-L_{R,min}$, $-L_{A,stop1}$ and $-L_{A,stop2}$ are upper bounds R_{U}^m. A negative

[2]The Mathworks, Inc., Natick, MA, USA

cost indicates that the corresponding specification is satisfied, while a positive cost denotes that the specification is violated. The final cost to be optimized at a design space point $\vec{g}$ is then given as

$$F(\vec{g}) = \max_{s_m} F(s_m, \vec{g}). \tag{5.8}$$

It should be noted that, the cost function described in (5.8) is derived from the constraint on the frequency response of a specific port. If optimization constraints are given for different ports, the worst-case violation over all the ports has to be calculated as the final cost as given below:

$$F(\vec{g}) = \max_{i,j} \max_{s_m} F_{i,j}(s_m, \vec{g}). \tag{5.9}$$

where, $F_{i,j}$ represents the cost function on the $(i,j)^{th}$ port. The cost function (5.9) is then supplied to the *GlobalSearch* optimization routine, resulting in multiple optimal design space points $[S_1^*, S_2^*, D^*]$ which satisfy the constraints. Such an optimization is illustrated in section 5.2.4.

5.2.3.7 Variability Analysis

The *GlobalSearch* optimization routine results in multiple optimal design space points $[S_1^*, S_2^*, D^*]$ which satisfy the constraints (Subsection 5.2.3.6). These points are ranked according in terms of the cost function value at these points. Instead of choosing the point for which the cost function value is the least, we also check how robust the points are with respect to fabrication process variation. The specifications of the milling machine to realize the filters, determine the maximal possible variation of the design parameters. Using this maximal variations we obtain a 3 dimensional region around each of the found feasable optimal values $[S_1^*, S_2^*, D^*]$. The aim is to perform a Monte Carlo analysis in the 3 dimensional region of the design space around $[S_1^*, S_2^*, D^*]$, assuming a uniform distribution of the design parameters in the 3 dimensional region. This analysis however requires a more accurate macromodel than the macromodel built in Subsection 5.2.3.5 to capture the variations of the S-parameters due to tiny variations of the design parameters. Since the region for which we have to build these more accurate models is small, it only requires a small number of additional EM simulations to boost the accuracy level locally of the tree-based parametric macromodel. Since the evaluation of the macromodel only takes little time, we can perform a Monte Carlo analysis with a high number of realization samples to obtain the robustness of the obtained solutions and its distribution. A limitation of this variability analysis is that it does not take into account the eventual variation on other parameters such as ϵ_r, W, h and S_h. If this would be required, these parameters can be added in the analysis.

5.2.3.8 Realization and Measurements

The last step in the design process is to realize the filter. The hairpin filter is realized with a milling machine of type ProtoMat C100/HF[3] in a RO4003 material. The filter is measured with a Vectorial Network Analyzer of type E8364B (PNA series) calibrated using the electronic calibration Ecal module. The frequency span of the measurements is chosen to be larger than the frequency span of the model to be sure that we can compare our model to the measurements and to see what happens at frequencies that are not modeled.

5.2.4 Numerical examples

In this section, the steps outlined in Section 5.2.3 are illustrated with three numerical examples for three optimization cases. The examples are chosen such that the same macromodel can be used for all three optimizations.

5.2.4.1 Specifications

The specifications on the S-parameters of the filters are:

$$|S_{21}| > -3\,\mathrm{dB}, \qquad f_{p1} \leq freq \leq f_{p2}, \tag{5.10a}$$
$$|S_{11}| < -8\,\mathrm{dB}, \qquad f_{p1} \leq freq \leq f_{p2}, \tag{5.10b}$$
$$|S_{21}| < -30\,\mathrm{dB}, \qquad freq < f_{s1}, \tag{5.10c}$$
$$|S_{21}| < -30\,\mathrm{dB}, \qquad freq > f_{s2}, \tag{5.10d}$$

with three optimization cases:

I. $(f_{p1}, f_{p2}, f_{s1}, f_{s2}) = (1.87, 1.94, 1.75, 2.06)$ GHz.

II. $(f_{p1}, f_{p2}, f_{s1}, f_{s2}) = (1.86, 1.92, 1.76, 2.03)$ GHz.

III. $(f_{p1}, f_{p2}, f_{s1}, f_{s2}) = (1.89, 1.95, 1.78, 2.07)$ GHz.

5.2.4.2 Preliminary Design

We generate starting geometries for the hairpin bandpass filter (see Fig. 5.9) for the three cases using the design equations given in Section 5.2.3.3. The substrate is the standard RO4003 with a relative permitivity $\epsilon_r = 3.55$, a loss tangent $\delta = 0.0021$ and a thickness h of $1.524\,mm$. The characteristic impedance of the lines is chosen to be 50 Ω, which corresponds to a width $W = 3.4$ mm. Table 5.2 shows the starting values for the variable design parameters $[S_1, S_2, D]$ for the three cases.

[3]LPKF Laser&Electronics AG, Osteriede 7, 30827 Garbsen, Germany

Parameter	D (mm)	S_1 (mm)	S_2 (mm)
Case I	20.348	0.790	3.378
Case II	20.474	0.846	3.780
Case III	20.102	0.835	3.583

Table 5.2: Initial values for the design parameters

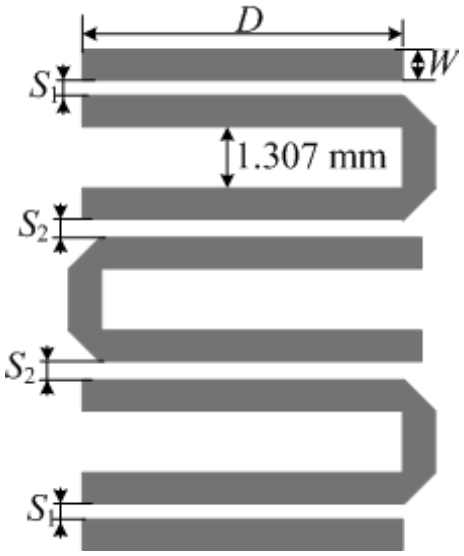

Figure 5.9: Layout of the hairpin bandpass filter.

Parameter	Min	Max
Frequency ($freq$)	1.5 GHz	2.5 GHz
Spacing 1 (S_1)	0.25 mm	2.0 mm
Spacing 2 (S_2)	1.5 mm	4.0 mm
Length (D)	20.0 mm	22.0 mm

Table 5.3: Hairpin Filter: Design parameters

5.2.4.3 Design space and parametric macromodeling of the filter

Section 5.2.3.4 shows that the two spacings S_1 and S_2 along with the length D are chosen as design parameters in addition to the frequency. The corresponding ranges are shown in Table 5.3.

The scattering parameter response (S-response) matrix $\mathbf{S}(s, S_1, S_2, D)$ has been computed using the ADS Momentum[4] software with Adaptive Frequency Sam-

[4]Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

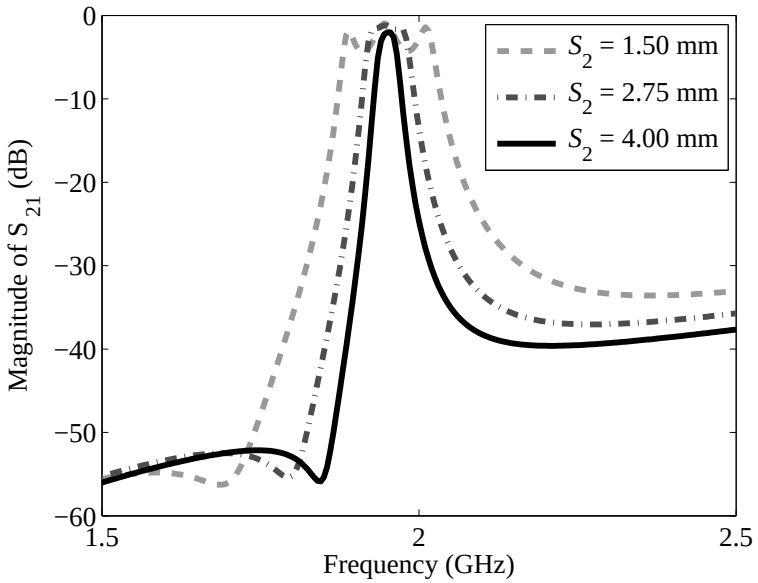

Figure 5.10: Magnitude of S_{21} for three different values of S_2.

pling (AFS). The number of simulated frequency samples is extracted from AFS. The number of frequency samples N_s has been chosen equal to 201 for the parametric macromodeling so that the sharp behavior of S-response of the microwave filter with respect to frequency is well captured. Fig. 5.10 shows the parametric behavior of the magnitude of S_{11} as a function of S_2 and frequency for $S_1 = 0.5$ mm and $D = 21$ mm. Similarly, Fig. 5.11 shows the magnitude of S_{21} as a function of D for $S_1 = 0.5$ mm and $S_2 = 2.5$ mm.

The parametric macromodeling along with the sequential sampling scheme discussed in Section 5.2.2 have been implemented in Matlab R2012a and used to drive the ADS Momentum simulations to generate S-parameters at selected design space samples. The MAE error measure (5.3) is used to asses the accuracy of the parametric macromodel. The numerical simulations have been performed on a Linux platform on Intel(R) Xeon(R) CPU E5504 @ 2.00 GHz machine with 6 GB RAM. Table 5.4 reports the total number of design space samples, the worst case mean absolute error (5.3) over the design space, the CPU time needed to run all the ADS Momentum simulations and the CPU time needed for the parametric macromodeling with the sequential sampling. The CPU time needed by ADS Momentum (using AFS) and the parametric macromodel for one frequency sweep is also shown in Table 5.4. This measure is critical in judging the advantage of such an approach while performing the design optimization.

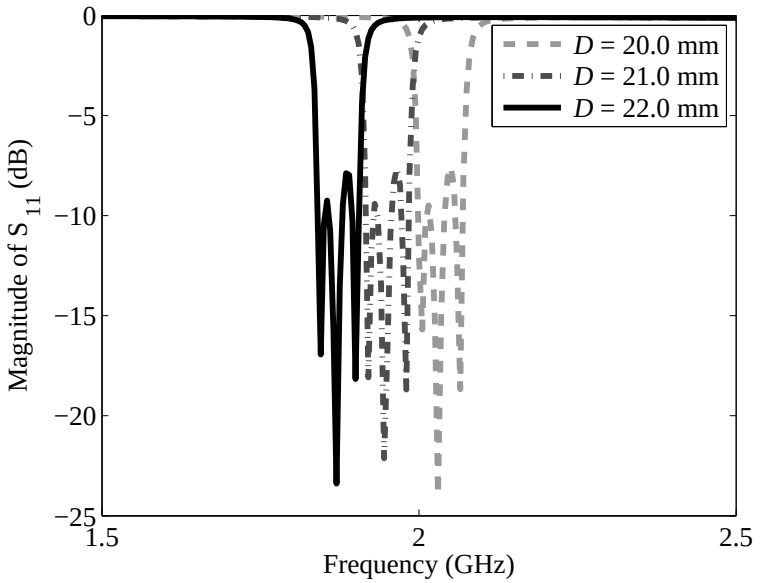

Figure 5.11: Magnitude of S_{11} for three different values of D.

# Samples		CPU Time		Accuracy
Generation	Validation	Modeling	Data Gen.	[dB]
104	86	29 min 48 s	2 h 27 min 38 s	-40
CPU Time for one ADS Momentum frequency sweep = 46.6 s CPU Time for one parametric macromodel frequency sweep = 0.0209 s Speed-up = 2230×				

Table 5.4: Hairpin Filter: Parametric macromodeling

5.2.4.4 Design optimization

The generated parametric macromodel is used in the design optimization of the filter. As explained in Section 5.2.3.6, the global optimization function *GlobalSearch* in Matlab R2012a is used to perform an optimization with a cost function formulated in (5.9) using the requirements of (5.6). The results of the three optimization cases are tabulated in Table 5.5. For each of the three optimization cases, the function *GlobalSearch* found three different optimization solutions (*Solution i* to *Solution iii*) $[S_1^{*(x)}, S_2^{*(x)}, D^{*(x)}]$, $x = i, ii, iii$, which satisfy the constraints in all three cases. The *Solution i* to *Solution iii* are ranked according in terms of the cost function value at these points.

Figs. 5.12 and 5.13 show the optimization results for *Solution i* of *Case I*

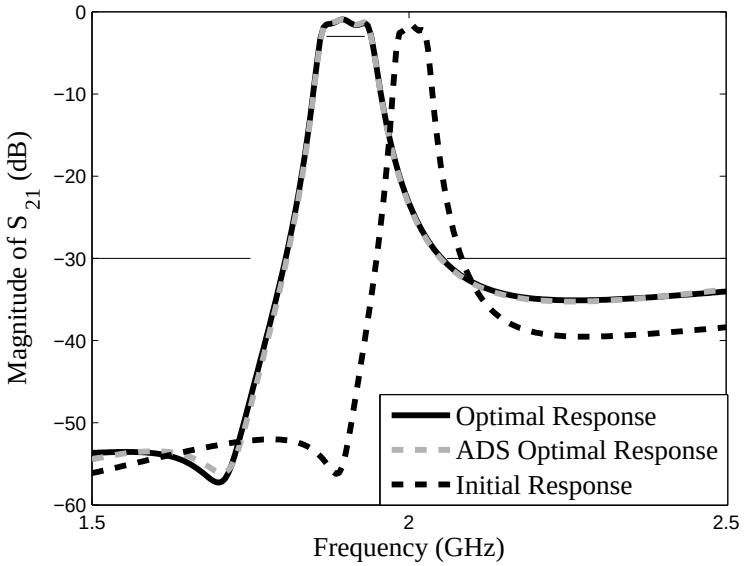

Figure 5.12: Case I, Solution i*: Magnitude of S_{21} before and after optimization. ADS optimal response is the EM simulation performed at the optimal solution point to verify that the model is accurate.*

wherein the different S-responses are shown before and after optimization. The constraints are shown by the solid black lines and as evident from the two figures, all the constraints are satisfied. Similar curves were observed for *Case II* and *Case III*.

5.2.4.5 Variability Analysis

Once multiple optimization solutions (*Solution i* to *iii*) are found for the three optimization cases, the next step is to see which solution is the most robust in terms of a variability analysis with respect to the three design parameters $[S_1, S_2, D]$ around the solutions $[S_1^{*(x)}, S_2^{*(x)}, D^{*(x)}]$, $x = i, ii, iii$. Assuming a uniform distribution of the three design parameters around the found optima, a Monte Carlo analysis with 10000 samples is performed to see the robustness of the obtained solutions of Table 5.5. For this, a locally refined parametric macromodel is built with a higher accuracy of -50 dB of MAE (5.3) on a 3D N-box region of the design space around each of the found solutions with ± 10 μm of tolerance on all the three design parameters. Each additional parametric macromodel requires only 8 generation and 1 validation points, defined by the 8 corner points $[S_1^{*(x)} \pm 10$ $\mu\text{m}, S_2^{*(x)} \pm 10$ $\mu\text{m}, D^{*(x)} \pm 10$ $\mu\text{m}]$. The Monte Carlo is then performed on these

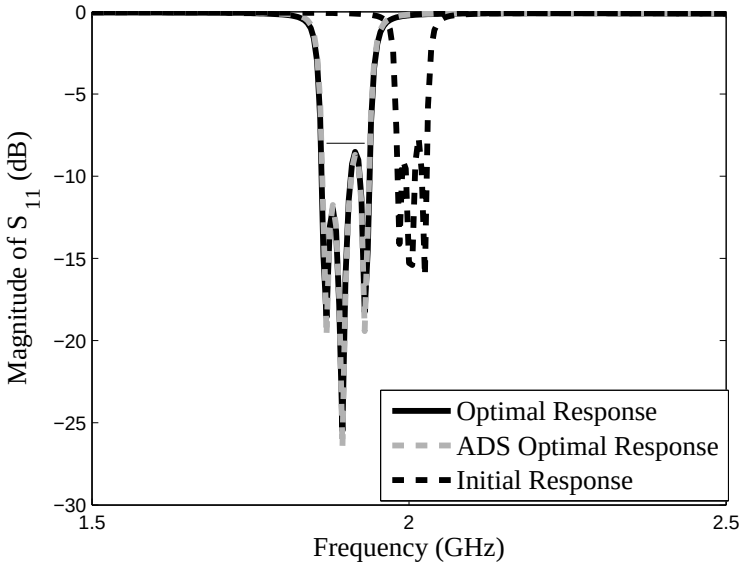

Figure 5.13: Case I, Solution i*: Magnitude of S_{11} before and after optimization. ADS optimal response is the EM simulation performed at the optimal solution point to verify that the model is accurate.*

local parametric macromodels in terms of the cost function to see the robustness of the found solutions. The Monte Carlo results are also tabulated in Table 5.5. Here, the robustness can be judged looking at the mean value and the standard deviation (STD) of the Monte Carlo solutions. A lower mean and a smaller STD is always preferred as the robust solution over the other solutions (in bold font, Table. 5.5)

Case	Initial design (S_1, S_2, D) [mm]	Solution	Optimal design (S_1^*, S_2^*, D^*) [mm]	# Function evaluations	Optimization time [s]	Variability modeling time [s]	Monte Carlo, cost [dB]		Monte Carlo time [s]
							Mean	STD	
I	$[0.790, 3.378, 20.348]$	i	$[\mathbf{0.374}, \mathbf{2.30}, \mathbf{21.629}]$	9598	3 min, 13 s	7 min, 11 s	$\mathbf{-0.659}$	$\mathbf{0.105}$	2 min, 50 s
		ii	$[0.388, 2.318, 21.628]$			7 min, 6.5 s	-0.528	0.138	2 min, 50 s
		iii	$[0.383, 2.256, 21.602]$			7 min, 19 s	-0.149	0.126	2min, 50 s
II	$[0.864, 3.780, 20.474]$	i	$[\mathbf{0.439}, \mathbf{2.504}, \mathbf{21.768}]$	6101	1 min, 58 s	7 min, 36 s	$\mathbf{-0.950}$	$\mathbf{0.100}$	2 min, 52 s
		ii	$[0.439, 2.501, 21.770]$			7 min, 33 s	-0.929	0.110	2 min, 52 s
		iii	$[0.439, 2.499, 21.763]$			7 min, 33 s	-0.893	0.099	2 min, 52 s
III	$[0.835, 3.583, 20.102]$	i	$[0.408, 2.419, 21.376]$	7269	2 min, 11 s	6 min, 55 s	-0.865	0.101	2 min, 51 s
		ii	$[\mathbf{0.406}, \mathbf{2.420}, \mathbf{21.375}]$			6 min, 55 s	$\mathbf{-0.898}$	$\mathbf{0.08}$	2 min, 51 s
		iii	$[0.381, 2.434, 21.37]$			6 min, 56 s	-0.816	0.073	2 min, 51 s

Table 5.5: Hairpin Filter: Optimization results.

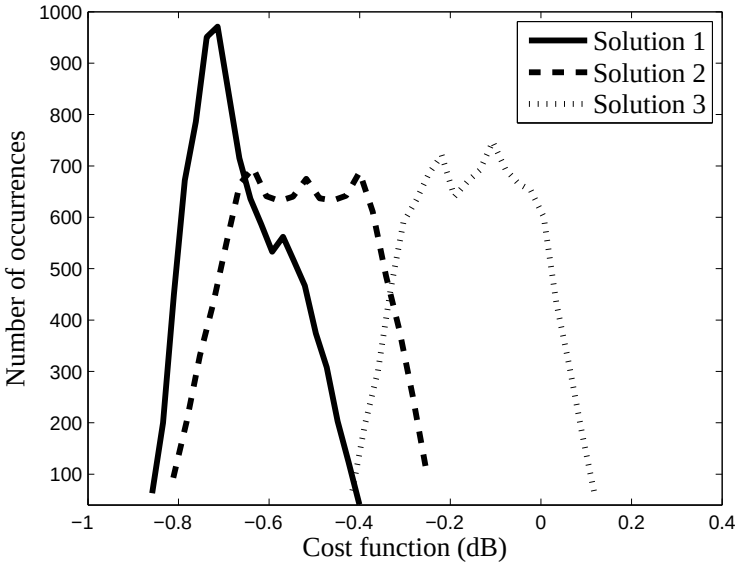

Figure 5.14: Case I*: Distribution of the cost function with the Monte Carlo analysis.*

Figs. 5.14, 5.15 and 5.16 show the distribution of the cost functions after performing the Monte Carlo analysis for *Case I*, *Case II* and *Case III* respectively. In Fig. 5.14, for *Case I*, it can be easily seen that *Solution i* is the most robust solution followed by *Solution ii* and *Solution iii* as also seen in Table. 5.5. It is interesting to note that for *Case III* in Fig. 5.16, even though the minimum cost was obtained for *Solution i*, *Solution ii* turns out to be the most robust solution.

5.2.4.6 Fabrication and measurements

After the optimization process and the variability analysis, the filters are fabricated. For each case, 2 filters are realized. The difference between the two realizations is the length of the microstripline that is added to the feed lines in order to place the connectors. For the first realization this length is 5 mm and for the second one it is equal to 15 mm. We expect that the behavior in the pass band is not influenced by this difference in length, since for the passband frequencies the filter is matched. Nevertheless we want to check how the stopbands are influenced by this difference. Fig. 5.17 shows a picture of the realization of the filter for *Case I* with 5 mm feedline.

The S-parameters of each realization are measured with a Vectorial Network Analyzer from 1 GHz to 3 GHz with a resolution of 1 MHz. Since the frequencies of the optimization and the measurements do not coincide, the measurement

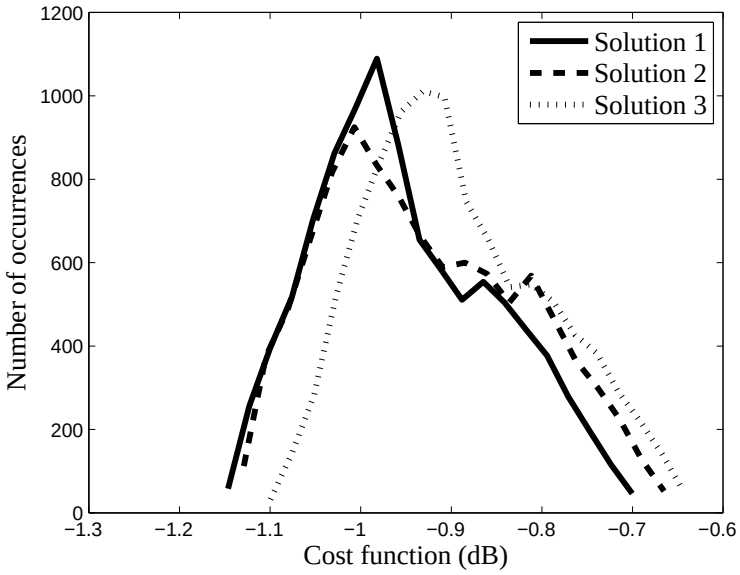

Figure 5.15: Case II*: Distribution of the cost function with the Monte Carlo analysis.*

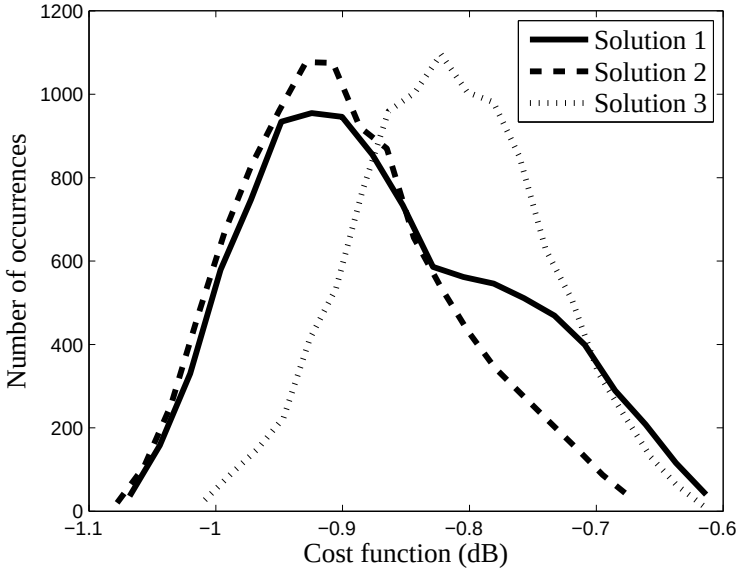

Figure 5.16: Case III*: Distribution of the cost function with the Monte Carlo analysis.*

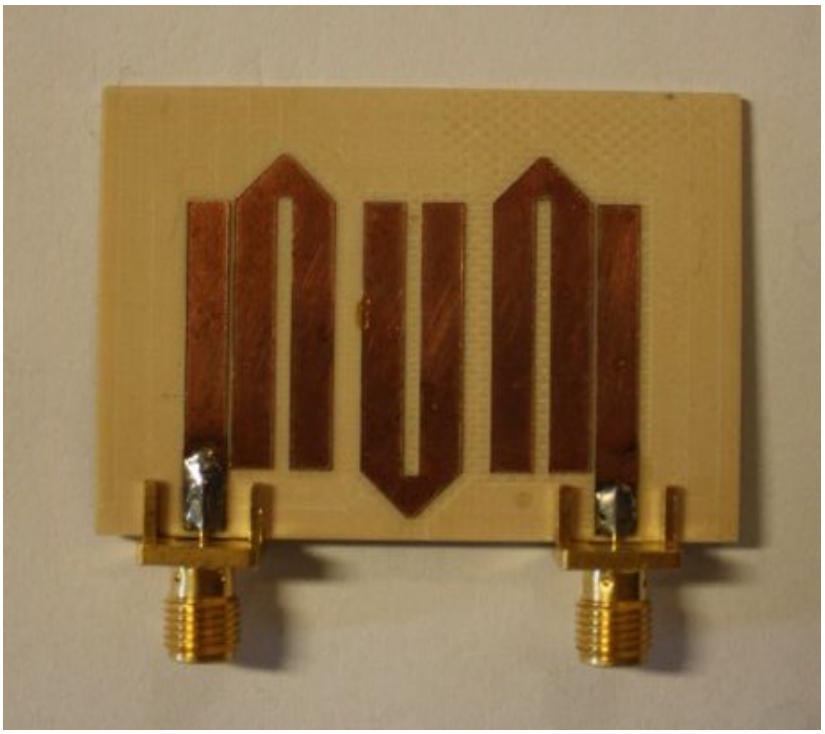

Figure 5.17: Realization of the hairpin filter for Case I *with 5 mm feedline.*

points are interpolated using cubic splines such that we can compare the results of the model to the measurements. We neglect the error due to the cubic splines interpolation. Fig. 5.18 shows the interpolation of the measurements for the first realization of *Case I*.

The filters are fabricated with a milling machine of type ProtoMat C100/HF. It is possible that due to the fabrication technique a part of the top layer of the dielectric that is not covered by stripline is milled away, which changes the height of the dielectric at ranges where the dielectric is not covered by metal. Also we assume that the relative permittivity of the substrate ϵ_r is 3.55, whereas the exact value might be different. This might introduce a frequency shift. Since we do not take into account these effects in our variability analysis, we do not know how they effect the S-parameters.

Tables 5.6 and 5.7 compare the measurement results to the optimization specifications. For each of the filters we observe a frequency shift of 10 to 25 MHz, which can be explained by the difference in height and relative permittivity. In our performance evaluation we take this shift into account. As an estimate of the shift we take the difference between the mean of the observed f_{p1} and f_{p2}

Case, realization	f_{p1} (GHz)	f_{p2} (GHz)	f_{s1} (GHz)	f_{s2} (GHz)
Spec I	1.87	1.94	1.75	2.06
Case I,1	1.898	1.961	1.823	2.115
Case I,2	1.874	1.914	1.800	2.170
Spec II	1.86	1.92	1.76	2.03
Case II,1	1.892	1.924	1.814	2.075
Case II,2	1.890	1.936	1.822	2.101
Spec III	1.89	1.95	1.78	2.07
Case III,1	1.905	1.965	1.833	2.104
Case III,2	1.911	1.972	1.845	2.323

Table 5.6: Measurement results: S_{21}

namely $f_{c,observed} = \frac{f_{p1,observed}+f_{p2,observed}}{2}$ (Table 5.6) and the specified f_{p1} and f_{p2}, namely $f_{c,spec} = \frac{f_{p1,spec}+f_{p2,spec}}{2}$. Thus the estimated frequency shift is $f_{c,spec} - f_{c,observed}$. For f_{p1} we take the highest of Table 5.6 and 5.7 and for f_{p2} the lowest so that we consider the worst-case violation. We evaluate the performance of the filters by comparing the positions of f_{p1}, f_{p2}, f_{s1} and f_{s2} of the measurements to the specifications. If the observed f_{p1} and f_{s2} are lower than and the observed f_{p2} and f_{s1} are higher than the specified frequencies the specifications are fulfilled. Table 5.8 contains the frequency shifts and the compensated frequencies for each of the filters. When we analyze these results we notice the following behavior:

- The specification on f_{s2} is always violated. Moreover we observe that the for the cases where a line 15 mm is added the difference between f_{s2} and $f_{s2,measured}$ becomes larger. This expected since out of the pass band, the structure is no longer matched.
- The specification on f_{s1} is always met.
- For Case I and Case II, the width of the pass bands is smaller than the specified width. Case III fulfills the specifications on the pass band.
- The specifications on S_{11} are always met.

Considering the fact that the fabrication process introduces some unmodeled effects, we conclude that filters perform well.

5.3 Conclusions

This chapter is divided into two parts: In the first part, we have presented an enhanced parametric macromodeling method for linear high-frequency systems. It is

Case, realization	f_{p1} (GHz)	f_{p2} (GHz)
Spec I	1.87	1.94
Case I,1	1.891	1.971
Case I,2	1.862	1.944
Spec II	1.86	1.92
Case II,1	1.871	1.942
Case II,2	1.878	1.949
Spec III	1.89	1.95
Case III,1	1.897	1.975
Case III,2	1.903	1.981

Table 5.7: Measurement results: S_{11}

Case, realization	f_{p1} (GHz)	f_{p2} (GHz)	f_{s1} (GHz)	f_{s2} (GHz)	Shift (MHz)
Spec I	1.87	1.94	1.75	2.06	0
Case I,1	1.874	1.937	1.799	2.091	-24.5
Case I,2	1.885	1.925	1.811	2.181	+11.0
Spec II	1.86	1.92	1.76	2.03	0
Case II,1	1.874	1.906	1.796	2.057	-18.0
Case II,2	1.867	1.913	1.799	2.078	-23.0
Spec III	1.89	1.95	1.78	2.07	0
Case III,1	1.890	1.950	1.818	2.089	-15.0
Case III,2	1.890	1.951	1.824	2.302	-21.5

Table 5.8: Performance evaluation

combined with a sequential sampling scheme is able to generate accurate parametric macromodels in an efficient and fully automated way. A comparison is made with state-of-the-art modeling approach on a pertinent numerical example to show the improved modeling capability and efficiency of the new method.

In the second part, we introduce parametric macromodels, developed in the first part of the chapter, in the design cycle of microwave filters. The main advantage of these macromodels is that they are cheap to evaluate with a suitable accuracy. Hence, the macromodels can replace the expensive EM solver in the optimization step of the filter, which makes this step much less time-consuming. Moreover the macromodels can be used to perform a variability analysis. We must however keep in mind that it also takes time to generate the macromodel and that this must be done during the design cycle. But even when we take this model generation time into account, the overall design cycle time for design activities (optimization and variability analysis) is much less than in the case an EM solver is used. We have illustrated this macromodeling-based design approach by applying it to the design of a hairpin resonator filter. Although the generation of the macro-

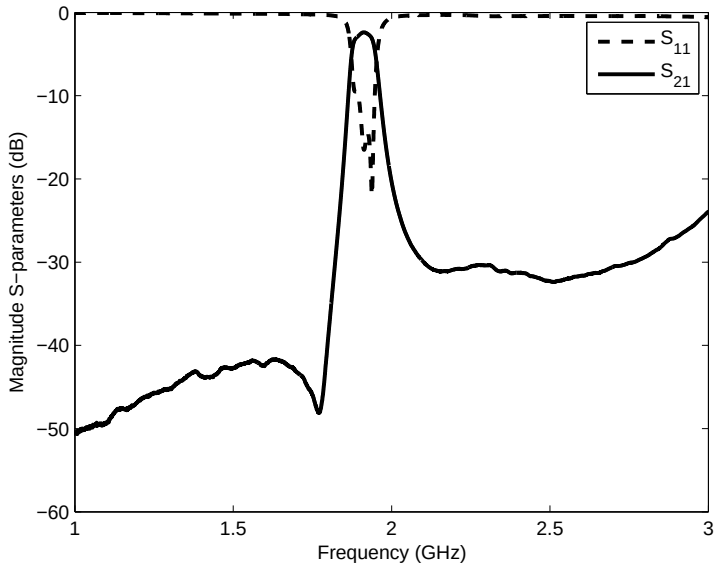

Figure 5.18: Measurement results Case I,1.

models is an automated process, it still requires some information from the user, namely the ranges of the design parameters, the desired accuracy and the frequency span. How a designer can hand this information to the macromodel generation process, is explained in detail in this chapter. Next, we show how the macromodel can be used for optimization and variability analysis. An important remark about the variability analysis, is that in general a more accurate macromodel must be generated in addition to the macromodel initially used to represent the parameterized system behavior over the complete design space.Since the design space regions for which these more accurate macromodels have to be generated are small, a very limited number of EM-simulations is required. Eventually three different filters are designed, fabricated and measured using this design approach. The designs are validated with measurements. The found performances of the filters confirms that macromodeling-based design approach works very well.

5.4 Acknowledgements

This work was supported by the Research Foundation Flanders (FWO), the Interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office, the Research Council of the Vrije Universiteit Brussel, the Flemish Government (Methusalem Fund METH1), and by the Belgian Federal

Government (Interuniversity Attraction Poles programme IAP VII, DYSCO).

References

[1] D. Deschrijver and T. Dhaene, "Stability and passivity enforcement of parametric macromodels in time and frequency domain," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 56, no. 11, pp. 2435–2441, 2008.

[2] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *IEEE Microwave and Wireless Component Letters*, vol. 19, no. 10, pp. 608–610, Oct. 2009.

[3] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *IEEE International Conference on Signal Propagation and Interconnects*, pp. 117 –120, May 2010.

[4] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[5] ——, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139 –151, March/April 2012.

[6] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2394 –2403, Oct. 2011.

[7] K. Chemmangat, F. Ferranti, T. Dhaene, and L. Knockaert, "Scalable models of microwave system responses using sequential sampling on unstructured grids," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 27, no. 1, pp. 122–137, 2014.

[8] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 1052–1061, July 1999.

[9] A. Weiser and S. E. Zarantonello, "A note on piecewise linear and multilinear table interpolation in many dimensions," *Mathematics of Computation*, vol. 50, no. 181, pp. 189–196, Jan. 1988.

[10] R. Levy, R. V. Snyder, and G. Matthaei, "Design of microwave filters," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 50, no. 3, pp. 783–793, 2002.

[11] D. Swanson and G. Macchiarella, "Microwave filter design by synthesis and optimization," *Microwave Magazine, IEEE*, vol. 8, no. 2, pp. 55–69, 2007.

[12] S. Peik, R. Mansour, and Y. Chow, "Multidimensional Cauchy method and adaptive sampling for an accurate microwave circuit modeling," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 46, no. 12, pp. 2364 –2371, Dec. 1998.

[13] A. Lamecki, P. Kozakowski, and M. Mrozowski, "Efficient implementation of the Cauchy method for automated CAD-model construction," *IEEE Microwave and Wireless Components Letters,*, vol. 13, no. 7, pp. 268 –270, July 2003.

[14] A. Cuyt, R. Lenin, S. Becuwe, and B. Verdonk, "Adaptive multivariate rational data fitting with applications in electromagnetics," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 54, no. 5, pp. 2265 – 2274, May 2006.

[15] V. Devabhaktuni, B. Chattaraj, M. Yagoub, and Q.-J. Zhang, "Advanced microwave modeling framework exploiting automatic model generation, knowledge neural networks, and space mapping," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 51, no. 7, pp. 1822 – 1833, July 2003.

[16] A. Lamecki, L. Balewski, and M. Mrozowski, "Adaptive CAD-model construction schemes," *IEEE Transactions on Magnetics,*, vol. 45, no. 3, pp. 1538 –1541, March 2009.

[17] P. Basl, R. Gohary, M. Bakr, and R. Mansour, "Modelling of electromagnetic responses using a robust multi-dimensional Cauchy interpolation technique," *IET Microwaves, Antennas Propagation,*, vol. 4, no. 11, pp. 1955 –1964, Nov. 2010.

[18] R. Lehmensiek and P. Meyer, "Creating accurate multivariate rational interpolation models of microwave circuits by using efficient adaptive sampling to minimize the number of computational electromagnetic analyses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 8, pp. 1419 –1430, Aug. 2001.

[19] D. Deschrijver, T. Dhaene, and D. De Zutter, "Robust parametric macromodeling using multivariate orthonormal vector fitting," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 7, pp. 1661 –1667, July 2008.

[20] P. Triverio, S. Grivet-Talocia, and M. Nakhla, "A parameterized macromodeling strategy with uniform stability test," *IEEE Transactions on Advanced Packaging*, vol. 32, no. 1, pp. 205 –215, Feb. 2009.

[21] F. Ferranti, L. Knockaert, and T. Dhaene, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623–629, Aug. 2010.

[22] D. Deschrijver, K. Crombecq, H. Nguyen, and T. Dhaene, "Adaptive sampling algorithm for macromodeling of parameterized S-parameter responses," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 1, pp. 39 –45, Jan. 2011.

[23] S. Koziel, J. Bandler, and K. Madsen, "A space-mapping framework for engineering optimization; theory and implementation," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 54, no. 10, pp. 3721 –3730, Oct. 2006.

[24] S. Koziel, J. Bandler, and Q. Cheng, "Robust trust-region space-mapping algorithms for microwave design optimization," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 58, no. 8, pp. 2166 –2174, Aug. 2010.

[25] S. Koziel, "Shape-preserving response prediction for microwave design optimization," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 58, no. 11, pp. 2829 –2837, Nov. 2010.

[26] I. Couckuyt, F. Declercq, T. Dhaene, H. Rogier, and L. Knockaert, "Surrogate-based infill optimization applied to electromagnetic problems," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 20, no. 5, pp. 492–501, 2010.

[27] I. Couckuyt, D. Deschrijver, and T. Dhaene, "Towards efficient multiobjective optimization: Multiobjective statistical criterions," in *2012 IEEE Congress on Evolutionary Computation (CEC),*. IEEE, 2012, pp. 1–8.

[28] E. Cristal and S. Frankel, "Hairpin-line and hybrid hairpin-line/half-wave parallel-coupled-line filters," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 20, no. 11, pp. 719–728, 1972.

[29] S. B. Cohn, "Parallel-coupled transmission-line-resonator filters," *Microwave Theory and Techniques, IRE Transactions on*, vol. 6, no. 2, pp. 223–231, 1958.

[30] J.-S. G. Hong and M. J. Lancaster, *Microstrip filters for RF/microwave applications*. Wiley-interscience, 2001, vol. 126.

[31] S. Cohn, "Direct-coupled-resonator filters," *Proceedings of the IRE*, vol. 45, no. 2, pp. 187–196, 1957.

[32] E. Fooks *et al.*, *Microwave engineering using microstrip circuits*. Prentice-Hall, Inc., 1990.

[33] Z. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly, and R. Martí, "Scatter search and local NLP solvers: A multistart framework for global optimization," *INFORMS Journal on Computing*, vol. 19, no. 3, pp. 328–340, 2007.

6

Sequential Sampling Generalizations to Scattered Grids

K. Chemmangat, T. Dhaene and L. Knockaert

Published in Electronics Letters, vol. 49, no. 15, pp. 950-952, 2013

International Journal of Microwave and Wireless Technologies, Accepted, In Press, January 2014

⋆ ⋆ ⋆

In this chapter we extend the sequential sampling method from N-dimensional hyperrectangular design space regions, described in the earlier chapter, to simplicial partitions to be used in the macromodeling of parameterized system responses. The chapter is divided into two main parts.

In the first part, the new refinement strategy using the well-conditioned local path-simplex refinement is discussed. A numerical example is also provided to validate the approach.

In the second part, a more elaborate discussion on the pros and cons of the grid-based and the path-simplex based refinement strategies are discussed. A hybrid scheme is later proposed to take the good points of both the schemes to make the sampling and the modeling strategy more efficient. Several numerical examples are provided to validate the method.

All the methods discussed here are tailored towards the local parametric macromodeling schemes on scattered grids.

6.1 Parametric Macromodeling of Microwave Systems Responses using Sequential Sampling with Path-Simplexes

6.1.1 Introduction

Efficient design of electromagnetic (EM) systems often requires expensive simulations using EM solvers which normally provide high accuracy at a significant cost in terms of memory storage and computing time. Alternatively, parametric macromodels can be used, which approximate the complex behavior of EM systems, characterized by frequency and additional design parameters, such as geometrical or substrate features. Parametric macromodeling of EM systems has attracted a lot of attention during recent years [1–3]. However, one of the key issues in these modeling approaches is that they select the number of modeling samples *a priori* which might result in under sampling or over sampling at the cost of computational resources.

In this chapter the state-of-the art parametric macromodeling schemes are automated with the help of a scattered sampling scheme which works on local refinement of well-conditioned simplexes such that optimum number of data samples are selected [4]. The refinement on the simplexes can be done in many ways such as dividing along in-center. However, this might lead to the creation of ill-conditioned simplexes called *slivers*. Generation of slivers can be avoided by refining either locally [5, 6] or globally [7, 8]. The local refinement scheme [5, 6] starts from the corner points of an N-cube and then refines it into smaller simplexes in a tree fashion like the sequential sampling method of [4], whereas the global refinement

schemes [7, 8] work on a primary Delaunay tessellation and then refine it to improve the conditioning of simplexes. Hence the local path-simplex method [5, 6] assures good partition from the beginning of the sampling process and is suitable for the application of different passivity-preserving parametric macromodeling algorithms on scattered grids [1–3]. In case if the global refinement schemes [7, 8] were used, the existing mesh has to undergo global refinement indicating that the local interpolated models change significantly with a consequent computational burden. Moreover, the tessellation generated by the method of path-simplexes is proved to be Delaunay by construction [6], and for the above mentioned reasons it is used in this chapter.

6.1.2 Passivity Preserving Parametric Macromodeling

In this chapter, we use one of the local parametric macromodeling schemes which use the Vector Fitting (VF) technique [9] to build frequency-dependent rational models called *root macromodels* at the selected design space samples and then parameterize them, see [1–3]. These methods preserve stability and passivity over the complete design space, and therefore are suitable for time-domain simulations. The parametric macromodeling process starts with a set of multivariate data samples $\{(s, \vec{g})_k, \mathbf{H}(s, \vec{g})_k\}_{k=1}^{K_{tot}}$ which depends on frequency and additional design parameters. From these data samples, a set of *root macromodels* in pole-residue form are built for a set of design space samples $\vec{g}_k$ by means of VF yielding a set of *root macromodels* $\mathbf{R}(s, \vec{g}_k)$. Stability and passivity are enforced using robust standard techniques [9, 10], resulting in a set of stable and passive *root macromodels*. The next step of these parametric macromodeling algorithms is the parameterization of the set of *root macromodels* $\mathbf{R}(s, \vec{g}_k)$. In [1], a parametric macromodel is built by interpolating a set of *root macromodels* at an input-output level, while in [2, 3], a novel enhanced interpolation of *root macromodels* is described, which results in high modeling capability and robustness in comparison to [1].

6.1.3 Refinement using Well Conditioned Path-Simplexes

A path-simplex in $\mathbf{R}^N$ is defined as an N-Simplex having N mutually orthogonal edges which, in the sense of graph theory, form a path [6]. Two edges are mutually orthogonal if they are perpendicular to each other. The property of a path-simplex which makes it attractive for the proposed sequential sampling is the fact that it is a *non-obtuse* simplex. To understand what a non-obtuse simplex is, it is necessary to define the *facets* and the *dihedral angles* of an N-simplex. A facet is defined as the $N + 1$ convex hulls of N distinct vertices or corner points of the N-simplex (for example, the face of a tetrahedron in 3D; facet is the generalization in higher dimensional spaces). The dihedral angle between two facets are the angles made between the inward normals of these facets. A non-obtuse simplex is then defined

as an N-simplex with all of its $\frac{1}{2}N(N+1)$ dihedral angles greater than $\frac{\pi}{2}$. This ensures that, slivers are never created during the local refinement of a simplex, ensuring convergence of the algorithm. The proposed sequential sampling algorithm starts from a single N-box region of the design space which is then normalized to a N-cube and divided into $N!$ path-simplexes using the result of [6].

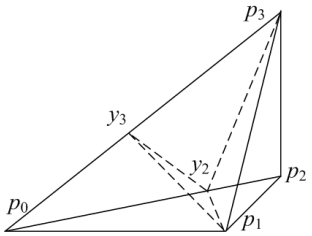

Figure 6.1: Coexter's trisection of the path-simplex in $\mathbf{R}^3$ (as in [5]).

In [5], Brandts et al. prove that given a path-simplex in $\mathbf{R}^N$, it can be divided into N path-subsimplexes using Coxeter's trisection method generating $N-1$ new sample points. Fig. 6.1 shows such a division for a path-simplex in $\mathbf{R}^3$. The corners of the path-simplex are represented by the position vectors p_0, p_1, p_2, and p_3 with respect to any arbitrary origin, and the edges $p_0 - p_1$, $p_1 - p_2$, and $p_2 - p_3$ are the three orthogonal edges which form a path. Three new path-simplexes are formed using the points y_2 and y_3 calculated as

$$y_j = p_j(\|p_1\|^2/\|p_j\|^2),\ j = 2, 3, \ldots, N, \tag{6.1}$$

where, $\|.\|$ is the Euclidean norm [5]. In (6.1), y_j, $j = 2, 3, \ldots, N$ are position vectors with respect to the same origin as for p_j. Generation of slivers during the local refinement can be monitored by calculating the aspect ratio,

$$R_{\text{asp}} = N\frac{d}{D}. \tag{6.2}$$

in (6.2), d and D are the diameters of the inscribing and circumscribing N-spheres of the N-simplex respectively. *Root macromodels* are created at the corner points of these simplexes and using the parametric macromodeling method of [2], passive interpolated models are created for the parameterized frequency responses.

6.1.4 Proposed Sequential Sampling Algorithm

The sequential sampling algorithm consists of the following steps:

I) *Initialization*: Define a N-box design space with N design parameters $\vec{g} = (g^{(1)}, \ldots, g^{(N)})$ and generate $Q = N!$ path-simplexes.

II) Update the parametric macromodel $\mathbf{R}(s, \vec{g})$ for the entire design space with Q path-simplexes using the method of [2].

III) For each path-simplex $q = 1, \ldots, Q$, check the error criteria at its in-center,

1. IF: ($Err_q > \Delta$): Divide q^{th}path-simplex into N path-subsimplexes [5], update $Q = Q + N - 1$, $q = q + 1$ and go to Step II.
2. ELSE: increment $q = q + 1$.
   i. IF ($q \leq Q$): Not all subspaces are checked for the error criteria, go to Step III;
   ii. ELSE: Go to Step IV.

IV) Terminate the algorithm. The final parametric macromodel with the required accuracy target is found.

6.1.5 Numerical Example

The S-Parameter response of a Hairpin bandpass filter generated with the help of ADS Momentum[1] on a substrate with relative permittivity $\epsilon_r = 9.9$ and a thickness of 0.0635 mm is modeled (Fig. 6.2). Two spacings S_1 and S_2 and two lengths L_1 and L_2 are chosen as design parameters (see Fig. 6.2) in addition to frequency. Their corresponding ranges are shown in Table 6.1. The parametric behavior of the filter is shown in Fig. 6.3.

Parameter	Min	Max
Frequency ($freq$)	1.5 GHz	3.5 GHz
Spacing 1 (S_1)	0.25 mm	0.35 mm
Spacing 2 (S_2)	0.65 mm	0.75 mm
Length 1 (L_1)	12.0 mm	12.5 mm
Length 2 (L_2)	2.75 mm	3.25 mm

Table 6.1: Design parameters of hairpin bandpass filter.

For the sequential sampling the Mean Absolute Error (MAE) is used

$$E^{\text{MAE}}(\vec{g}) = \sum_{i=1}^{P_{in}} \sum_{j=1}^{P_{out}} \sum_{k=1}^{N_s} \frac{|R_{i,j}(s_k, \vec{g}) - H_{i,j}(s_k, \vec{g})|}{P_{in} P_{out} N_s}, \tag{6.3}$$

[1] Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

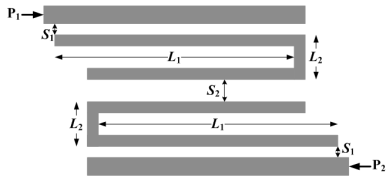

Figure 6.2: Layout of the microwave hairpin bandpass filter.

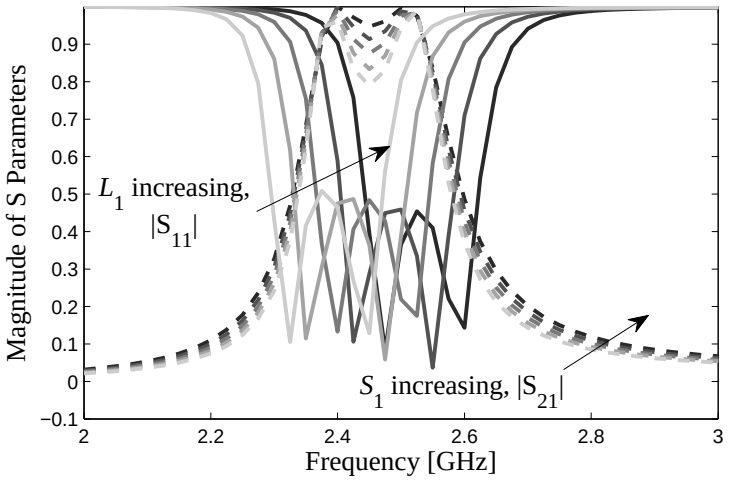

Figure 6.3: Parameterization: $|\mathbf{S}_{11}|$ *and* $|\mathbf{S}_{21}|$ *as a function of* L_1 *and* S_1 *resp.*

to assess the accuracy of the model at the in-center of each simplex. The method compares the actual EM simulation response $H_{i,j}(s, \vec{g})$ to the parametric macromodel response $R_{i,j}(s, \vec{g})$, with P_{in} input ports, P_{out} output ports and N_s frequency samples. The proposed sequential sampling algorithm is used along with the parametric macromodeling method of [2, 3]. The MAE measure of (6.3) is used to assess the accuracy of the models generated with a target accuracy of -50 dB. This resulted in the selection of 68 design space points, with an achieved accuracy of -50.21 dB.

Fig. 6.4 shows the normalized values of the 68 design space points selected, using a parallel coordinate plot [11]. In Fig. 6.4, the black dots represent the sample points selected for each design parameters with the gray lines representing different samples points in four dimension. The minimum aspect ratio was found to be equal to 0.0514, meaning no slivers were created. Fig. 6.5 compares the magnitude of S-parameter matrix entry $\mathbf{S}_{21}$ between the actual momentum simulation with the macromodel for three random validation points in the design space and the responses overlap showing the accuracy. In order to check the passivity,

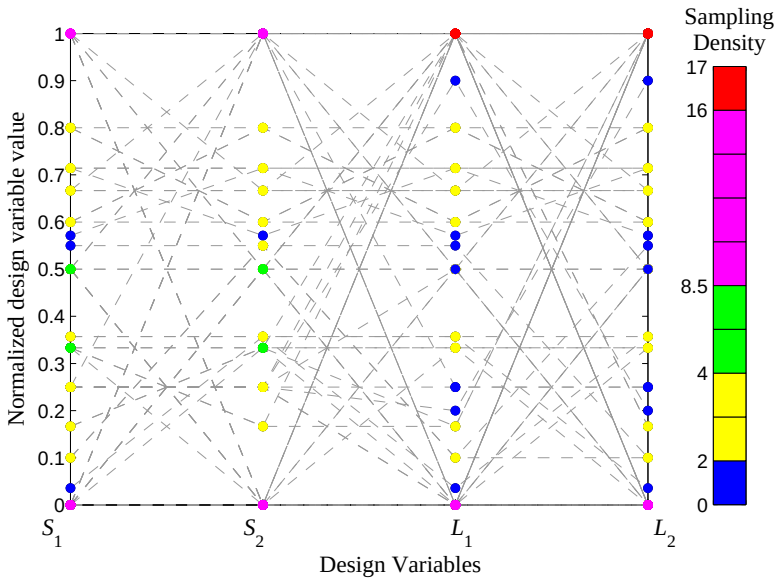

Figure 6.4: Design space generated for Hairpin Filter

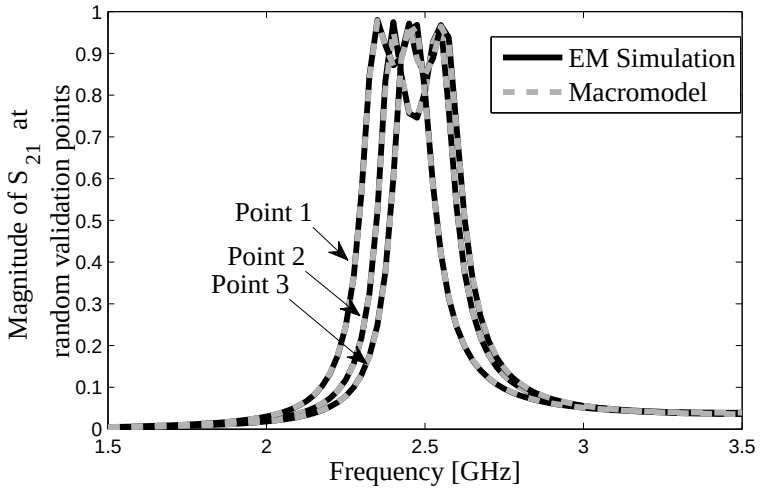

Figure 6.5: Magnitude of $\mathbf{S}_{21}$ at three random validation points.

the $\mathbf{H}_\infty$ norm $||\mathbf{R}(s, S_1, S_2, L_1, L_2)||_\infty$ of the parametric macromodel for a dense grid of $5 \times 5 \times 5 \times 5$ (S_1, S_2, L_1, L_2) was calculated and was found to be within the passivity bound, $||\mathbf{R}(s, S_1, S_2, L_1, L_2)||_\infty \leq 1$.

6.2 Parametric macromodeling using sequential sampling with a combination of grid-based and path-simplex based sampling

6.2.1 Introduction

Efficient design of electromagnetic (EM) systems using accurate parametric macromodels is an active field of research [1–3, 12–16]. These parametric macromodels are computationally cheap and they act as a replacement model for the expensive EM solvers thereby reducing the overall computational burden. The parametric macromodeling schemes are also able to preserve system properties such as stability and passivity and hence can be used in time-domain simulations [1–3, 12–16]. However, these state-of-the-art macromodeling schemes suffer from the fact that the sample distribution over the design parameter space should be known a priori based on rules of thumb [1–3, 12–16].

Several sequential sampling algorithms have been suggested in the literature for automatically building parametric macromodels for the EM systems [17–22]. All of these sampling schemes are global but often fail to guarantee stability and passivity. Preserving system properties is very important especially if the model thus generated is employed in time-domain simulations [23]. Also, for relatively high dimensions the memory requirement for these methods can be relatively high since big matrices has to be solved, limiting their applicability [17–19, 22]. Recently a local tree-based sequential sampling has been proposed in [4] which uses interpolation-based local parametric macromodeling method to build accurate parameterized macromodels. The method is able to preserve system properties and can also build multi-fidelity models. This means that the designer can already begin the design process once sufficient accuracy is reached for the intermediate model while the model is still being refined. Also the method is implemented as a tree with independent branches for different regions of the design space making it easily expandable and portable to parallel computing platforms.

In this chapter, we improve the method of [4], which we refer throughout this chapter as *grid method*, on the following aspects:

1 The grid method deals with hyperrectangular regions of the design space which are called *subspaces* in this chapter. In the previous method, after performing an edge-based division the algorithm finally divides along the center of the subspace. This is an exploratory step which tends to generate a considerable number of samples per division. This becomes more critical with higher dimension. In this work, the final refinement after the edge-based division is performed using a scattered division with well-conditioned simplicial partitions called path-simplexes [5, 6] reducing the overall complexity of the problem.

2 The grid method requires validation samples to access the accuracy of the terminal subspaces which need not be used further in the final model. In this chapter, the validation samples are altogether eliminated by using a level-based check wherein two subsequent levels of models are compared for convergence. This results in a considerable reduction in the overall number of points required.

The refinement on the simplexes is done using the well-conditioned path-simplex division (see Section 6.1.3). However, since the path-simplex method gives more importance to conditioning of the simplexes, it is more a space-filling strategy. So, in this work a hybrid scheme which combines all the benefits of the grid-based refinement and the path-simplex refinement is used to get an efficient sequential sampling strategy requiring less computational resources.

6.2.2 Preliminaries

This section briefly explains a robust parametric macromodeling method used in the chapter and also recapitulate the grid-based sequential sampling method of [4].

6.2.2.1 Passivity Preserving Parametric Macromodeling

As discussed in Section 6.1.2, the novel enhanced interpolation method of [2, 3] is used here. To understand the macromodeling method of [2, 3], let us consider a 2 parameter design space region Ω_l, $l = 1, \ldots, L$ given in Fig. 6.6. A two parameter description is presented here for clarity and ease of notation, even though the method is general for any dimension N of the design space. The rational *root macromodels* $\mathbf{R}^{\Omega_l}(s, \vec{g}_i^{\,\Omega_l})$, $i = 1, \ldots, 2^N$ contained in the N-box region Ω_l are represented in a pole-residue form:

$$\mathbf{R}^{\Omega_l}(s, \vec{g}_i^{\,\Omega_l}) = \sum_{p=1}^{P_i^{\Omega_l}} \frac{C_{p,i}^{\Omega_l}}{s - a_{p,i}^{\Omega_l}} + D_i^{\Omega_l} \quad ; p = 1, \ldots, P_i^{\Omega_l} \tag{6.4}$$

where $C_{p,i}^{\Omega_l}$ represents the residue matrices, $a_{p,i}^{\Omega_l}$ the pole $P_i^{\Omega_l}$. $D_i^{\Omega_l}$ is the direct-term matrix.

Later, the parametric macromodeling is applied on these N-box regions Ω_l. The design space region Ω_l is defined by four bounding corners $\vec{g}_1^{\,\Omega_l} = (g_1^1, g_2^1)$, $\vec{g}_2^{\,\Omega_l} = (g_1^2, g_2^1)$, $\vec{g}_3^{\,\Omega_l} = (g_1^1, g_2^2)$, and $\vec{g}_4^{\,\Omega_l} = (g_1^2, g_2^2)$ as in Fig. 6.6. Each corner possesses a different *root macromodel* $\mathbf{R}^{\Omega_l}(s, \vec{g}_i)$, $i = 1, \ldots, 4$. We will discuss the interpolation of the *root macromodels* next. For simplicity and ease of notation we omit the superscript Ω_l. In [2, 3], one amplitude scaling and one frequency

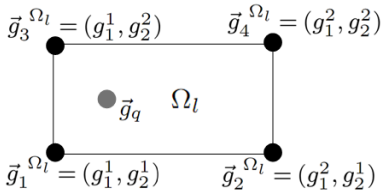

Figure 6.6: A two dimensional design space with four root macromodels.

scaling coefficient (α_1, α_2) are calculated using the optimization

$$(\alpha^*_{1,ij}, \alpha^*_{2,ij}) = \underset{(\alpha_{1,ij}, \alpha_{2,ij})}{\operatorname{argmin}} \left[\mathrm{Err}(\tilde{\mathbf{R}}^j(s, \vec{g}_i), \mathbf{R}(s, \vec{g}_j))\right] . \; i = 1, \ldots, 4; \; j = 1, \ldots, 4; \tag{6.5}$$

In (6.5), $\tilde{\mathbf{R}}^j(s, \vec{g}_i) = \alpha_{1,ij}\mathbf{R}(s\alpha_{2,ij}, \vec{g}_i)$, is the interpolated response of $\mathbf{R}(s, \vec{g}_i)$ obtained to match $\mathbf{R}(s, \vec{g}_j)$ and $\mathrm{Err}(\cdot)$ is a suitable error measure between the two responses [2, 3]. Note that, $\alpha^*_{1,ij} = \alpha^*_{2,ij} = 1$ when $i = j$.

The evaluation of the model taken at a generic point $\vec{g}_q$ in the design space (Fig. 6.6) is done similarly to [2, 3] as:

i. For each *root macromodel* $\mathbf{R}(s, \vec{g}_i)$, $i = 1, \ldots, 4$, the amplitude scaling coefficient $\alpha_{1,ij}$ and frequency scaling coefficient $\alpha_{2,ij}$ are interpolated using a multilinear interpolation [24] over $\vec{g}$ at the point $\vec{g}_q$ to find $\alpha_{1,iq}$ and $\alpha_{2,iq}$. This results in the modified *root macromodels*, $\tilde{\mathbf{R}}_q(s, \vec{g}_i) = \alpha_{1,iq} \times \sum_{p=1}^{P_i} \frac{C_{p,i}}{s\alpha_{2,iq} - a_{p,i}} + D_i$ at $\vec{g}_q$,

ii. Then the models $\tilde{\mathbf{R}}_q(s, \vec{g}_i)$, are interpolated using the multilinear interpolation [24] over $\vec{g}$ to get the final model interpolated at the points $\vec{g}_q$, $\mathbf{R}(s, \vec{g}_q)$.

This parametric macromodeling approach is performed for each region Ω_l which can either be a N-dimensional hyperrectangle or a N-simplex to cover the complete design space.

6.2.2.2 Sequential sampling using Grid-Based Refinement

The grid-based refinement scheme of [4] works on hyperrectangular grids and generates local parametric macromodel for each and every *subspace*. The grid-based sequential sampling algorithm begins from a single subspace with 2^N corners defined by the design parameter ranges [4]. Then it finds the maximum sensitive edge by checking difference between two responses of every edge and selects the edge with maximum difference. Later, a $(N-1)$-Hyperplane perpendicular to the selected edge is used to divide the subspace into two child subspaces if the accuracy is not satisfactory. This procedure is repeated until all the subspaces are

accurate and then finally a center refinement is used to complete the process as clearly explained in [4].

The idea of selecting the maximum sensitive edge is slightly modified in this work to make use of the available information generated by [2, 3]. The idea here is to find the most *difficult-to-model* edge in terms of the macromodeling method of [2, 3] as explained in detail below.

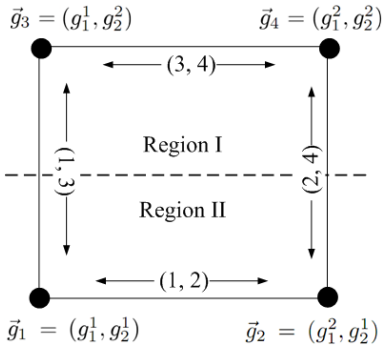

Figure 6.7: Subspace division along the most difficult-to-model edge.

As in Section 6.2.2.1, let us consider a two parameter design space $\vec{g} \in (g_1, g_2)$ defined by four corners $\vec{g}_1 = (g_1^1, g_2^1)$, $\vec{g}_2 = (g_1^2, g_2^1)$, $\vec{g}_3 = (g_1^1, g_2^2)$, and $\vec{g}_4 = (g_1^2, g_2^2)$ of a rectangular region (two dimensional subspace) as in Fig. 6.7. In the parametric macromodeling method of [2, 3], one amplitude scaling and another frequency scaling coefficients $(\alpha^*_{1,ij}, \alpha^*_{2,ij})$ are calculated as in (6.5). The error information obtained from (6.5) can be used as a measure of the modeling difficulty of each and every edge $1 \leq (i,j)_{\text{edge}} \leq 4$ (see Fig. 6.7), and the most difficult-to-model edge is the edge with the worst-case error given by the formula,

$$(i,j)^{\max}_{\text{edge}} = \underset{(i,j)_{\text{edge}}}{\text{argmax}} \left(\min_{(\alpha_{1,ij}, \alpha_{2,ij})} \left[\text{Err}(\tilde{\mathbf{R}}(s, \vec{g}_i), \mathbf{R}(s, \vec{g}_j)) \right] \right). \tag{6.6}$$

Then, a hyperplane perpendicular to the edge $(i,j)^{\max}_{\text{edge}}$ is used to divide the subspace into two halves. In Fig. 6.7, the pair $(2,4)$ was selected as the most difficult-to-model edge and a line perpendicular to that edge is used to divide the subspace into two.

However, the grid-based scheme suffers from the following:

1. The final refinement after the edge-based refinement of each subspace is performed at the center [4], and to keep the hyperrectangular nature of the grid all the lower dimensional hyperplanes such as edges, faces etc., of the subspace are divided generating a lot of points.

2. The local parametric macromodel is build by linearly interpolating 2^N *root macromodels* for every subspace [4]. Since the *root macromodel* transfer

functions are appended as in [2, 3], the order of the final parametric macromodel inside a subspace can have high values (in proportion to 2^N). This increases the evaluation time for the macromodel especially for time-domain simulations.

Both of the above mentioned issues become even more troublesome with higher dimensions. Therefore, a scattered refinement using well-conditioned simplexes becomes necessary to overcome these issues.

6.2.3 Path-Simplexes

The scattered refinement as discussed in Section 6.1.3 is done with the help of well-conditioned simplicial partition using path-simplexes. In this section we give a much more elaborate discussion on this local refinement strategy. A path-simplex has the following properties which makes it useful for the proposed sequential sampling [6]:

i. a path-simplex in $\mathbb{R}^N$ is a non-obtuse simplex and all its $(N-i)$-simplexes, $i = 1, 2, ..(N-1)$ are also path-simplexes,

ii. it contains its circumcenter ensuring good conditioning,

iii. path-simplex refinement is assured to be Delaunay by construction,

iv. every alternate division performed on path-simplex generates geometrically similar simplexes.

The above mentioned points ensures that slivers are never created during the local refinement of a simplex, ensuring convergence of the algorithm [25]. The proposed sequential sampling algorithm starts from an edge-based refinement scheme described above and then uses the result of [6] to refine a N-box region of the design space into $N!$ path-simplexes. Then these path-simplexes are further divided as in the Coxeter's trisection method [5] which is described in Section 6.1.3.

To see the advantage of using a path-simplex division, it is compared with respect to a division at the incenter of the simplex and the minimum aspect ratio (6.2) for each level of division is plotted in Fig. 6.8. The comparison is made with respect to the number of times the simplex is divided as well as the dimension N of the simplex. Both the division techniques start from a path-simplex of unit orthonormal edges. As seen in Fig. 6.8, the path-simplex division preserves the aspect ratio of the simplexes thereby assuring a good conditioning. This is much better as compared to the incenter division. As stated in [5], a path-simplex if divided twice, one of the sub-sub-simplexes is similar to the original simplex keeping its aspect ratio. This can also be seen in Fig. 6.8, where the aspect ratio shows oscillatory behavior showing each alternate division levels are similar.

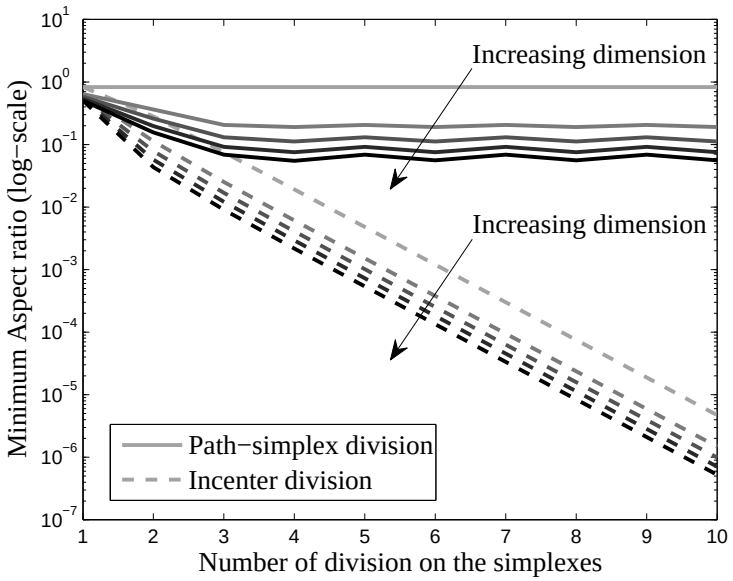

Figure 6.8: Comparison of Coxeter's trisection method versus incenter-based division.

Next, to show why a scattered sampling is required, some important parameters for the edge-based and center-based subspace division of [4] is compared with the path-simplex division and is tabulated in Table 6.2. By using a simplex-based division, the number of expensive samples generated per refinement can be brought down to $(N-1)$. Note that the number of points created per center-based division is calculated by summing all the possible hyperplanes such as edges, faces, etc., which are divided. Also, the order of the final macromodel can be reduced from that proportional to a factor of 2^N for the grid-based scheme to a factor of $(N+1)$ for the scattered scheme. This creates considerable speed-ups in macromodel evaluations both in frequency and time-domain.

Refinement Method	# points per refinement	# regions per refinement	model order proportional to
Edge-based	2^{N-1}	2	2^N
Center-based	$1+\sum_{m=1}^{N-1}\frac{2^{N-m}N!}{m!(N-m)!}$	2^N	2^N
Path-simplex	$N-1$	N	$N+1$

Table 6.2: Comparison of different refinement strategies.

However, one of the issues with the path-simplex based division is that the method acts as a space-filling strategy by generating well-conditional simplexes with less emphasis on sequential sampling. Therefore a hybrid method is proposed in this chapter which brings the advantages of both schemes to get a better sequential sampling strategy as will be explained in Section 6.2.5.

6.2.4 Error estimation without validation points

In [4], the accuracy of the model is calculated by comparing the parametric macromodel with the actual EM simulations at each and every terminal subspaces. When a subspace is found to be accurate, it is not further divided and the expensive validation points from the EM solver may not used [4]. This can be tackled to a certain extend by performing an estimation of the accuracy or the error at each and every subspace as described here.

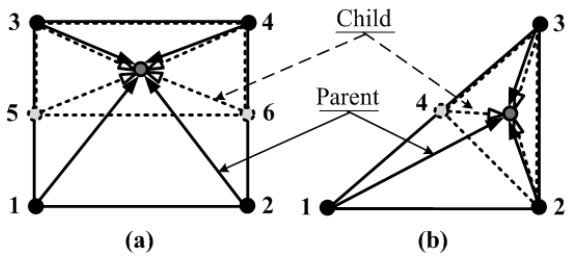

Figure 6.9: Error estimation with two subsequent model levels: (a) Rectangular region and (b) Simplicial region.

In this chapter, a comparison between two subsequent levels of parametric macromodels is proposed to assess the convergence of the sequential sampling as shown in Fig. 6.9.a for a 2D case on a rectangular grid. Two different parametric macromodels are compared, one from the parent subspace (region 1-2-3-4, solid line arrows) and the other from the child subspace (region 5-6-3-4, dashed line arrows) at the center of the child subspace region. On the other hand, if the simplexes are divided, a similar strategy is used, wherein the two macromodels are compared at the incenter of the child simplex as shown in Fig. 6.9.b. When a convergence is observed between the two levels, the algorithm is terminated.

6.2.5 Proposed sequential sampling algorithm

Fig. 6.10 shows the flowchart of the proposed sequential sampling algorithm. The algorithm is divided into different stages as explained below.

Stage 1:*Initialization* is done by defining the boundaries of the design space and then generating the 2^N corner *root macromodels*. Then an initial parametric

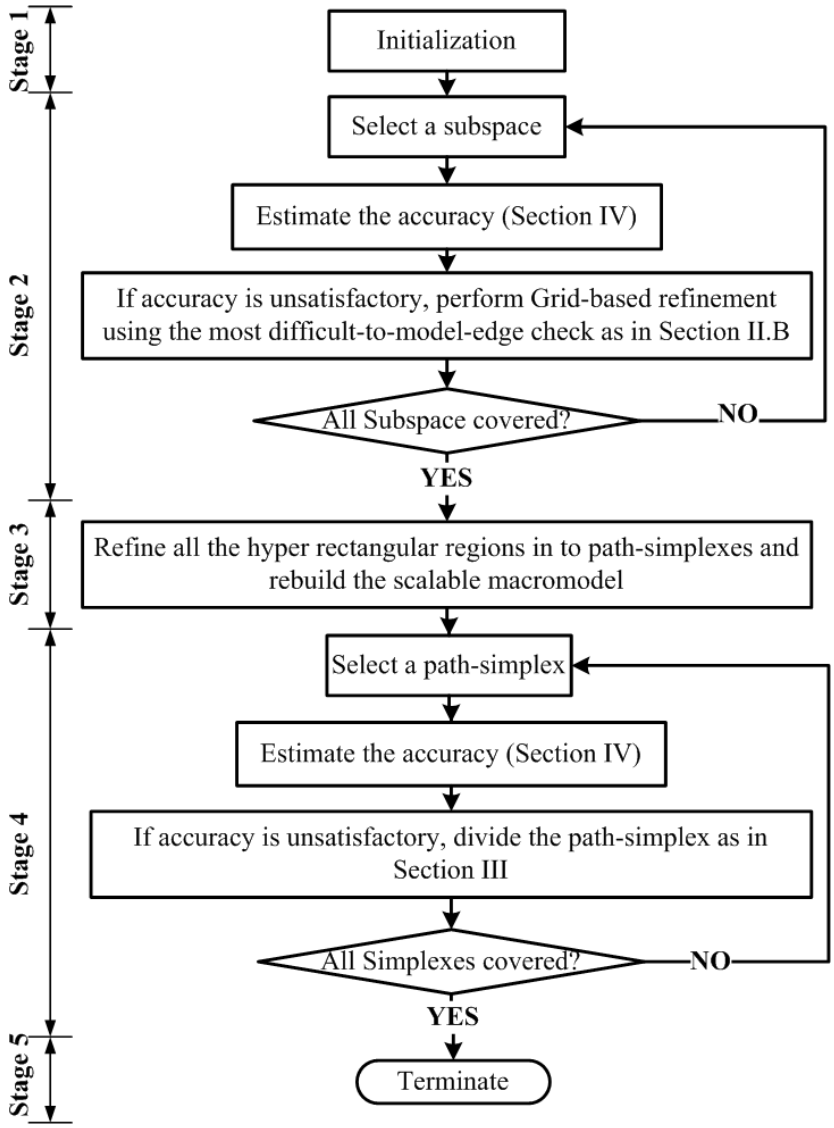

Figure 6.10: Flow chart of the proposed sequential sampling.

macromodel is built for this subspace by using [2, 3] and this is the starting point of the tree-based sequential sampling. The number of EM simulations at this stage is only the 2^N corner points of the design space.

Stage 2:Then, the *grid-based refinement* is performed to refine the initial parametric macromodel using the edge refinement method as in Section 6.2.2.2. At every iteration a subspace is selected and the modeling error is estimated using the

parent-child response comparison as explained in Section 6.2.4. If the subspace is found to be inaccurate, the subspace is divided using a hyperplane perpendicular to the most difficult-to-model edge as per Section 6.2.2.2 similar to Fig. 6.7. This step is continued till all the subspaces are sufficiently accurate.

This initial accuracy target can only be set depending on the problem at hand. For example, considering microwave filter, if a passband requirement of -30 dB is required, the parametric macromodel should be able to describe the filter characteristics up to an accuracy of -30 dB. So, the initial modeling accuracy can be set to the bare minimum accuracy required by the designer (-30 dB in the example stated here) such that this low fidelity model can already be used the design process. The number of EM simulations at this stage depends on the grid-based refinement.

Stage 3:In the next step, the initial grid generated is rearranged into path-simplexes regions. No additional EM simulations are required at this stage. Here, the model generated using the grid-based refinement is used as the starting point and then each and every terminal subspaces are refined into path-simplexes [6]. It should be noted that, during this process only hyperrectangular regions are converted into path-simplex regions and no further calculation of amplitude and frequency scaling parameters for the parametric macromodeling [2, 3] is needed since the *root-macromodels* stays the same.

One important point to note here is that, the refined hyperrectangular regions can already be long (meaning some edges are significantly longer than the others) according to the relative importance of different design parameters. So, if these hyperrectangular regions are converted directly into path-simplexes, the generated path-simplexes can have poor aspect ratio. Therefore, the hyperrectangular regions are first normalized to unit cubes before being refined into simplicial regions to avoid such a scenario.

Stage 4: Finally, the *Scattered refinement* is done using the method of path-simplexes by which the final target accuracy is to be achieved. After this conversion, the simplex regions are selected and error is estimated at their incenters similar to Section 6.2.4. A higher accuracy can be selected here compared to Stage 2, but since the error is estimated between two different models (as in Setcion IV), the accuracy target at Stage 2 and Stage 4 can very well be equal and the convergence is checked by estimating the error between the two models. If the accuracy is not satisfied for some simplicial regions, they are further divided using the path-simplex refinement procedure of Section 6.1.3 until all the simplexes are accurate. As in the case of grid-based refinement, the number of EM simulation is decided by the scattered refinement process.

Stage 5: Once all the simplicial regions are accurate, the sequential sampling algorithm is terminated. No further EM simulations are required at this stage.

It is important to note that, when the range of the design space is increased, the algorithm takes care of the change by exploring the design space. This is done by

generating additional samples (or *root macromodels*) in the newly added regions. Thus, by generating additional samples and refining the bigger regions into smaller domains, the algorithm will ensure that the parametric macromodeling scheme is able to build accurate models over the complete design space.

6.2.6 Numerical results

In this section three numerical examples are presented which demonstrate the capability of the proposed sequential sampling method for efficiently building the parametric macromodels for EM systems. For comparison purposes in terms of the computational time, all the numerical simulations have been performed on a Linux platform on Intel(R) Xeon(R) CPU E5504 @ 2.00 GHz machine with 6 GB RAM.

6.2.6.1 Example I: Microstrip bandpass filter

A microstrip bandpass filter on a substrate with relative permittivity $\epsilon_r = 9.0$ and a thickness of 0.660 mm is modeled in this example. The S-Parameter response of the filter is generated with the help of ADS Momentum[2]. The ADS Momentum simulation engine is used in full-wave mode. All ports are defined as single mode ports, with $50\ \Omega$ characteristic impedance. The automatic meshing (with edge mesh) uses 20 cells per wavelength, at a mesh frequency of 6 GHz. The layout of this filter is shown in Fig. 6.11. Two lengths L_1 and L_2 and the spacing S are chosen as design parameters (see Fig. 6.11) in addition to frequency whose ranges are $L_1 \in [6.0, 7.0]$ mm, $L_2 \in [4.0, 5.0]$ mm, $S \in [0.05, 0.10]$ mm, and frequency $\in [4.0, 6.0]$ GHz. The Mean Absolute Error (MAE) measure or the $L1$-norm per port is used to assess the accuracy of the model in every N-box region of the design space:

$$E^{\mathrm{MAE}}(\vec{g}) = \max_{\substack{u=1,\ldots,P \\ v=1,\ldots,P}} \frac{1}{N_s} \left(\sum_{n=1}^{N_s} |R_{u,v}(s_n, \vec{g}) - H_{u,v}(s_n, \vec{g})| \right). \tag{6.7}$$

The method compares the EM simulation response $H_{u,v}(s, \vec{g})$ with the parametric macromodel response $R_{u,v}(s, \vec{g})$, where P is the number of system ports. The MAE error measure or the $L1$-norm gives a global view on the error between the two frequency responses and hence it is preferred. The target accuracy was kept at -45 dB and the initial refinement accuracy for the proposed method also kept at -45 dB.

Fig. 6.12 shows the parametric behavior of the magnitude of S_{11} as a function of L_1 and frequency, other values being kept at the mean value of the design

[2]Momentum EEsof EDA, Agilent Technologies, Santa Rosa, CA.

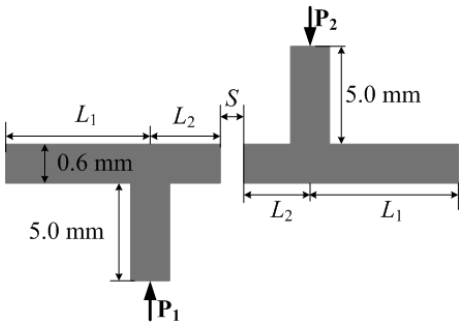

Figure 6.11: Example I: Layout of the microstrip bandpass filter.

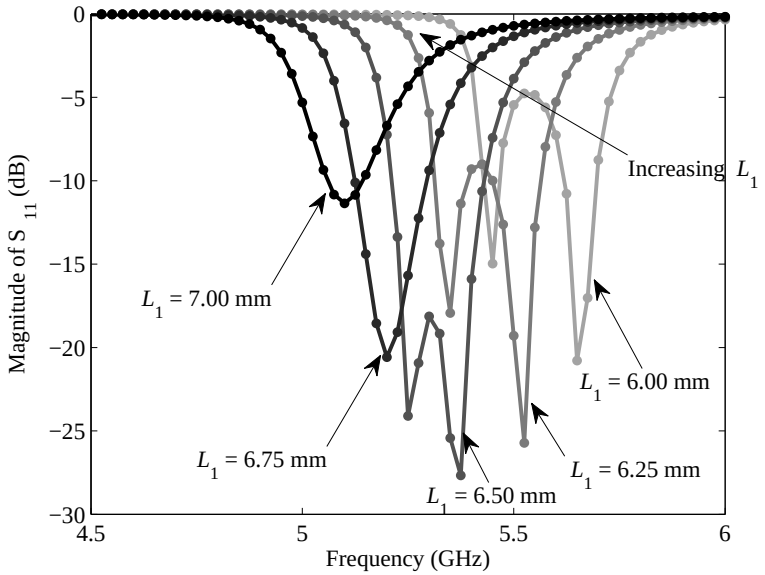

Figure 6.12: Example I: Magnitude of S_{11} as a function of L_1.

space. Similarly, Fig. 6.13 shows the magnitude of S_{21} as a function of S and frequency. The proposed algorithm and the grid method [4] have been implemented in Matlab R2012a[3] and used to drive the ADS Momentum simulations to generate S-responses with 31 frequency points at selected samples. The dotted curves in Figs. 6.12 and 6.13 represent the response of the parametric macromodel obtained from the proposed method. As seen, a good agreement can be observed.

Fig. 6.14 shows the distribution of the design sample points with the proposed

[3]The Mathworks, Inc., Natick, MA, USA

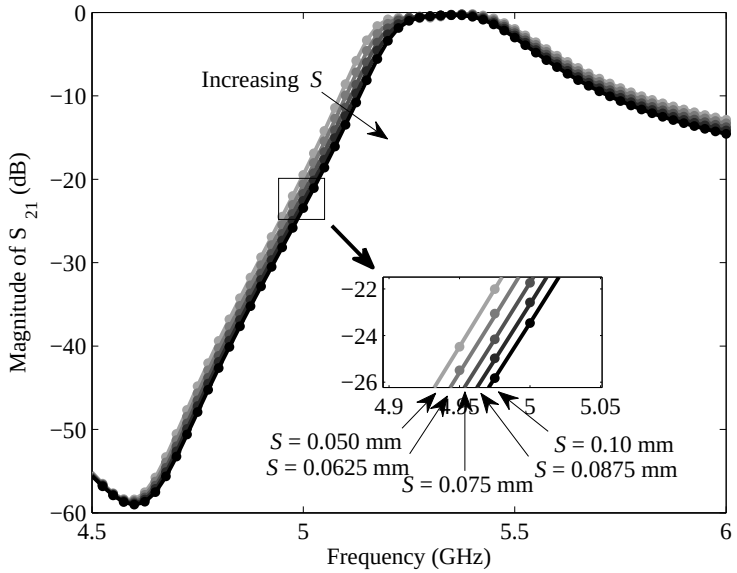

Figure 6.13: Example I: Magnitude of S_{21} as a function of S.

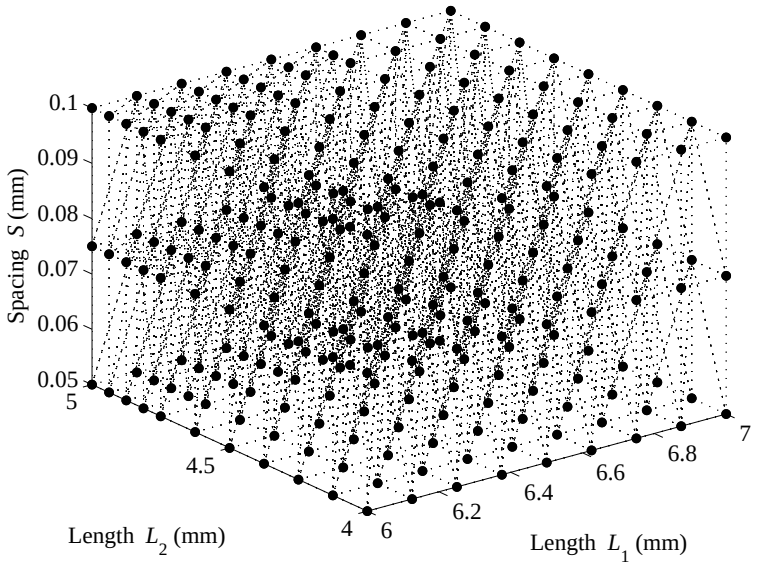

Figure 6.14: Example I: Sample distribution with the proposed algorithm.

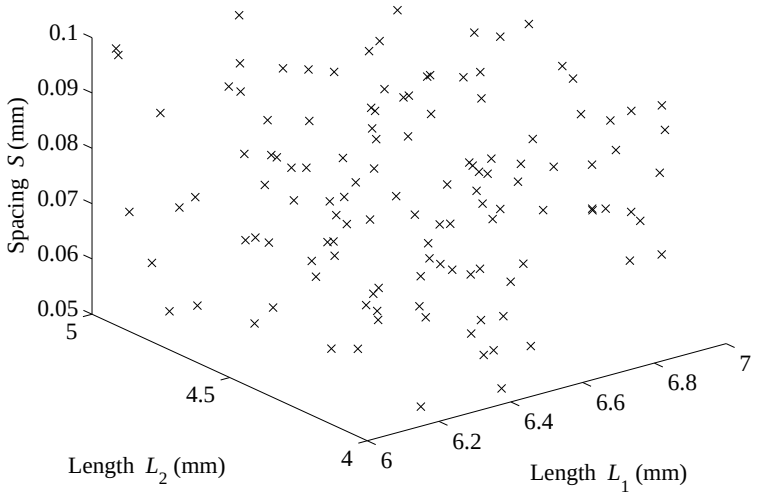

Figure 6.15: Example I: Verification sample distribution.

hybrid sequential sampling algorithm. The tessellation with the path-simplexes are also shown here. It can be observed from Fig. 6.14 that along the design parameter L_2, the maximum number of samples are taken, which means that this parameter is highly influential on the output S-parameters of the filter. The design parameter S is the least influential and hence it is sparsely sampled. Also, it can be noticed that the higher values of the design parameter $L_2 \in (4.5, 5.0)$ mm is more densely sampled than the other parts of the design space. This indicates a high sensitivity of the output S-parameter response of the filter to the changes in $L_2 \in (4.5, 5.0)$ mm in comparison with the lower values, $L_2 \in (4.0, 4.5)$ mm. In order to show the capability of the proposed algorithm, it is compared with the grid method on 125 verification points spread across the design space as shown in Fig. 6.15 using a Latin hypercube space filling.

Fig. 6.16 shows the mean absolute error distribution for both sequential sampling methods over the final verification points. As seen in the figure, a comparable accuracy is achieved for the proposed sequential sampling scheme without having to use any validation points during the sampling process like the grid method. Next, to check the advantage gained by performing a hybrid algorithm, the parametric macromodeling is performed directly on the scattered sampling. That is, the Stage 2 of the algorithm in Fig. 6.10 is not performed and the algorithm is checked for its convergence. It was observed that the algorithm, even after generating 492

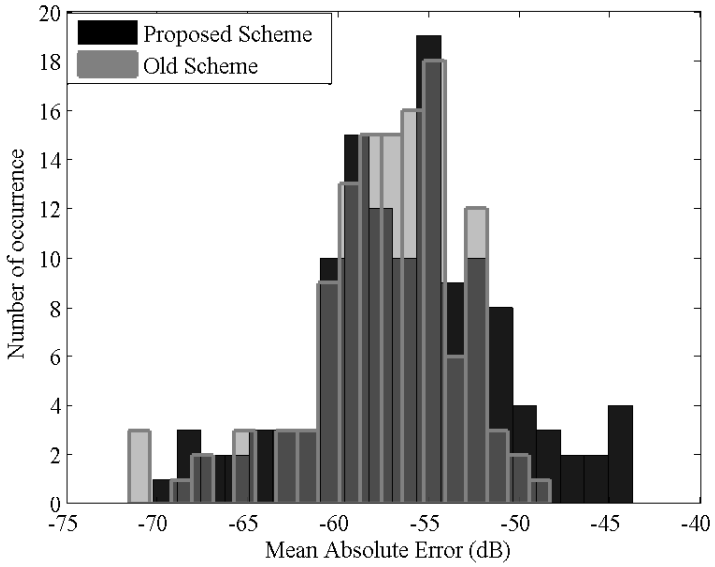

Figure 6.16: Example I: Mean absolute error distribution for the final verification samples.

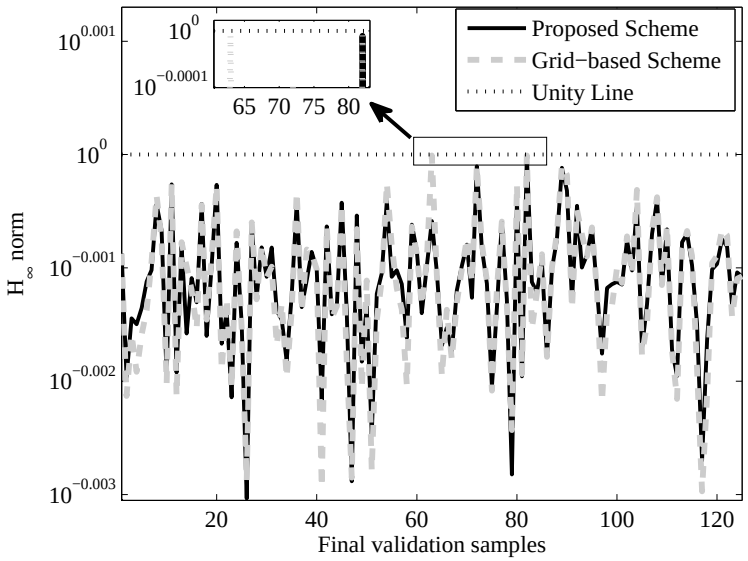

Figure 6.17: Example I: H_∞ norm for the final verification samples.

Sampling Method	# Samples		Max. Order	Error (dB)	Max. $\|\|H\|\|_\infty$	Evaluation Time (s)	Speed-up
	Gen.	Val.					
Proposed	275	-	86	-43.8	0.999	0.136	199 ×
grid [4]	438	248	164	-48.3	0.999	0.256	105 ×
CPU time for EM simulator to calculate a single frequency response = 27 s.							

Table 6.3: Example I: Comparison of different sampling strategies.

samples could converge only to an accuracy level of -25.11 dB. As explained in Section 6.1.3, the path-simplex division, even though very well-conditioned, is highly exploratory in nature and hence slow in converging to a good modeling accuracy. Thus combining the advantages of both grid-based refinement and the simplicial refinement becomes necessary to have an efficient and automated sampling strategy.

In order to check the passivity, the $\mathbf{H}_\infty$ norm $||\mathbf{R}(s, L_1, L_2, S)||_\infty$ of the parametric macromodel was calculated for the final 125 points and is plotted in Fig. 6.17. From the figure, it is clear that maximum norm is bounded by unity as expected. Table 6.3 compares the two sampling schemes over some important parameters. As seen in the table, the maximum order of the parametric macromodel is relatively high for the grid method in comparison to the proposed scheme. This is because of the fact that all the 2^N corner *root macromodels* of a subspace is augmented in the grid method whereas there are only $N + 1$ corner *root macromodels* defining a simplex for the proposed scheme. This becomes more severe with higher dimensions as will be seen in a later example. The table also shows the worst case mean absolute error for the two schemes and as expected, the proposed scheme has a slightly lower accuracy level, but only requires 40% of the samples compared to grid method (please note that the target accuracy level was -45dB). This is because of the fact that the error is estimated at each level during the sampling whereas for the grid method it is validated using expensive EM simulations. This is the tradeoff between the two sampling schemes.

Finally, to see the speed-up gained by using a parametric macromodel in the design process, the time required for one single frequency response evaluation of the parametric macromodels are compared with the corresponding EM simulator time and the speed-up is tabulated in Table 6.3. The speed-up for each case is calculated by comparing the evaluation time for one frequency sweep using the macromodel (column 7 of Table 6.3) with the CPU time for the EM simulator. However, it should be noted that the generation of the parametric macromodel requires some initial EM simulations, but once the model is built, it can be used in multiple design optimization scenarios such as changing specifications, to make the overall design cycle very efficient.

So, in conclusion the following points can be observed from Table 6.3:

1. The number of samples required for the proposed scheme is reduced in comparison with the methods of [4, 25].

2. The validation samples are completely removed by estimating error by comparing two different levels of the model as in Section 6.2.4.

3. Estimating the error means that the accuracy of the proposed method might not be as good as the grid method of [4]. This is the tradeoff between the validation points and error estimation.

4. The modeling complexity is reduced in comparison with grid method of [4] as explained before in Table 6.2. This also gains in the speed-up in terms of the parametric macromodel evaluation as clear from the last column of the table.

5. It should be noted here that the order of rational models for both methods are significantly huge in comparison with the *root macromodels*. This is because of the fact that the parametric macromodeling method described in [2, 3] performs the weighted averaging of different *root macromodels* which increases the interpolated rational model order.

6.2.6.2 Example II:Microstrip with two coupled vias

A microstrip with two coupled vias on a substrate with relative permittivity $\epsilon_r = 9.0$ and a thickness of 500 μm is modeled in this example. The S-parameter response of the structure is generated with ADS Momentum. The ADS Momentum simulation engine is used in full-wave mode. All ports are defined as single mode ports, with 50 Ω characteristic impedance. The automatic meshing (with edge mesh) uses 30 cells per wavelength, at a mesh frequency of 5 GHz. Fig. 6.18 shows the top and cross sectional view of the structure. The length of the two vias L, the distance between the two vias D and the radius of the vias r are chosen as design parameters (see Fig. 6.18) in addition to frequency whose ranges are $L \in [150, 250]$ μm, $D \in [1.5, 2.5]$ mm, $r \in [0.1, 0.4]$ mm, and frequency $\in [0.1, 10.0]$ GHz. The target accuracy (6.7) was kept at -45 dB and the initial refinement accuracy for the proposed method also kept at -45 dB. Fig. 6.19 shows the parametric behavior of the magnitude of S_{11} as a function of L and frequency, other values being kept at the mean value of the design space. Similarly, Fig. 6.20 shows the magnitude of S_{21} as a function of r and frequency. As in Example I, the proposed algorithm and the grid method [4] have been implemented in Matlab R2012a and used to drive the ADS Momentum simulations to generate S-responses at selected samples. The dotted curves in Figs. 6.19 and 6.20 represent the response of the parametric macromodel obtained from the proposed method. As seen a good agreement can be observed.

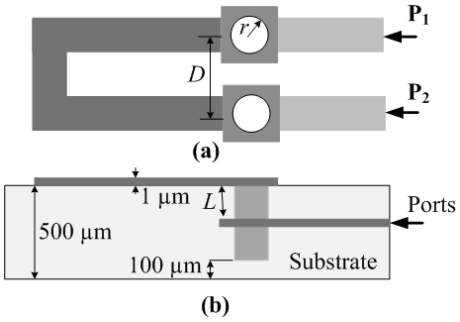

Figure 6.18: Example II: Layout of Microstrip with two coupled vias: (a) Top view, (b) cross sectional view.

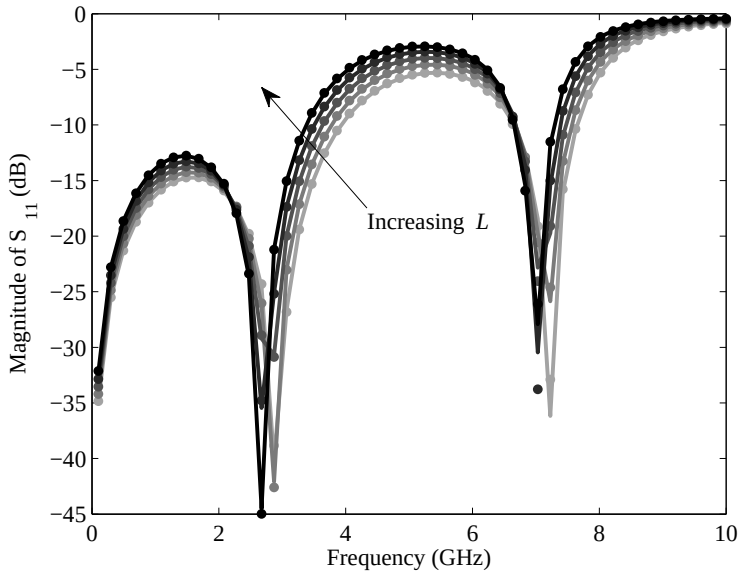

Figure 6.19: Example II: Magnitude of S_{11} as a function of L.

As in the previous example, the proposed algorithm is compared with the grid method on 125 verification points spread across the design space using a Latin hypercube space filling. Fig.6.21 shows the mean absolute error distribution for the sequential sampling methods over the final verification points. A comparable accuracy is achieved for the proposed sequential sampling scheme as it was observed for Example I. The $\mathbf{H}_\infty$ norm $||\mathbf{R}(s, L, D, r)||_\infty$ of the parametric macromodel was calculated for the final 125 points and is plotted in Fig.6.22 and it is observed to be passive.

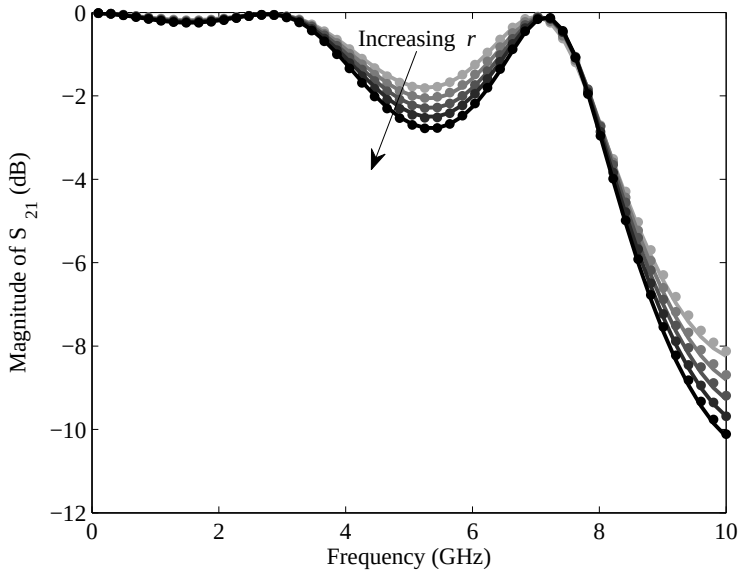

Figure 6.20: Example II: Magnitude of S_{21} as a function of r.

Sampling Method	# Samples		Max. Order	Error (dB)	Max. $\|\|H\|\|_\infty$	Evaluation Time [s]	Speed-up
	Gen.	Val.					
Proposed	30	-	64	-40.72	0.999	0.1180	1008 ×
Grid [4]	92	38	128	-45.41	0.999	0.2017	590 ×
CPU time for EM simulator to calculate a single frequency response = 119 s.							

Table 6.4: Example II: Comparison of different sampling strategies.

Table 6.4 compares the two sampling schemes over some important parameters similar to Table 6.3 for the first example. As for the Example I, a similar conclusion can be derived for the different quantities such as maximum model order, mean error etc. Note that the target accuracy level was -40 dB. The tradeoff between the two sampling schemes in terms of the error versus number of modeling samples required is also clear in the Table 6.4.

As in the first example, to show the speed-up gained by using a parametric macromodel in the design process, the time required for one single frequency response evaluation of the parametric macromodels are compared with the corresponding EM simulator time and the speed-up is tabulated in Table 6.4.

So from Table 6.4 a similar conclusion can be derived as in the case of Example I

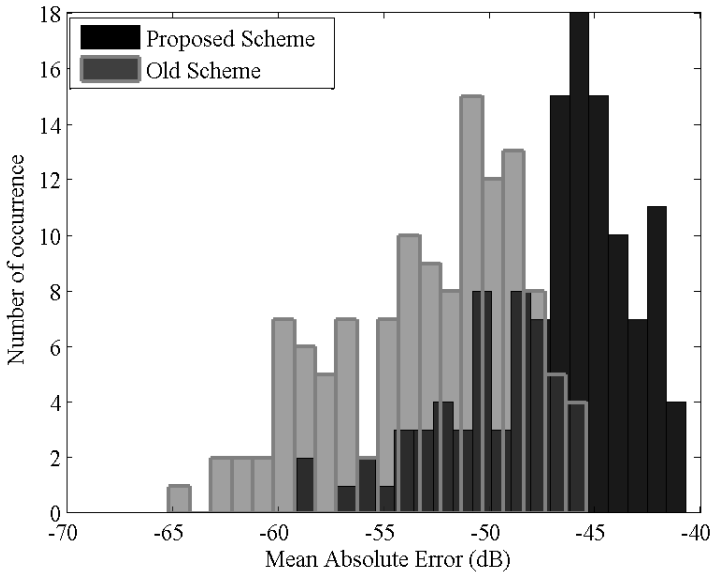

Figure 6.21: Example II: Mean absolute error distribution for the final verification samples.

6.2.6.3 Example III: Ring resonator filter

A Ring resonator bandpass filter on a substrate with relative permittivity $\epsilon_r = 4.32$ and a thickness of 152 μm is modeled in this example. The layout of this filter is shown in Fig. 6.23. The S-parameter response of the structure is generated with ADS Momentum. The ADS Momentum simulation engine is used in full-wave mode. All ports are defined as single mode ports, with 50 Ω characteristic impedance. The automatic meshing (with edge mesh) uses 20 cells per wavelength, at a mesh frequency of 4 GHz. Two spacings S_1 and S_2 and three lengths L_1, L_2 and L_3 are chosen as design parameters (see Fig. 6.23) in addition to frequency whose ranges are $S_1 \in [0.20, 0.30]$ mm, $S_2 \in [0.04, 0.06]$ mm, $L_1 \in [20.0, 24.0]$ mm, $L_2 \in [19.0, 21.0]$ mm, $L_3 \in [26.0, 27.0]$ mm and frequency $\in [1.0, 3.0]$ GHz. The MEA measure (6.7) was used to estimate the modeling accuracy. The target accuracy was kept at -40 dB and the initial refinement accuracy for the proposed method also kept at -40 dB.

Parametric behavior of some of the S-Parameter matrix entries of the filter are shown in Figs. 6.24, 6.25 and 6.26 with respect to some design parameters. In Fig. 6.24, the magnitude of S_{11} and the S_{21} of the filter are shown for five different values of L_1 keeping the other parameters constant at their mid values of range. Similar plot can be observed in Figs. 6.25 and 6.26 for the parameters

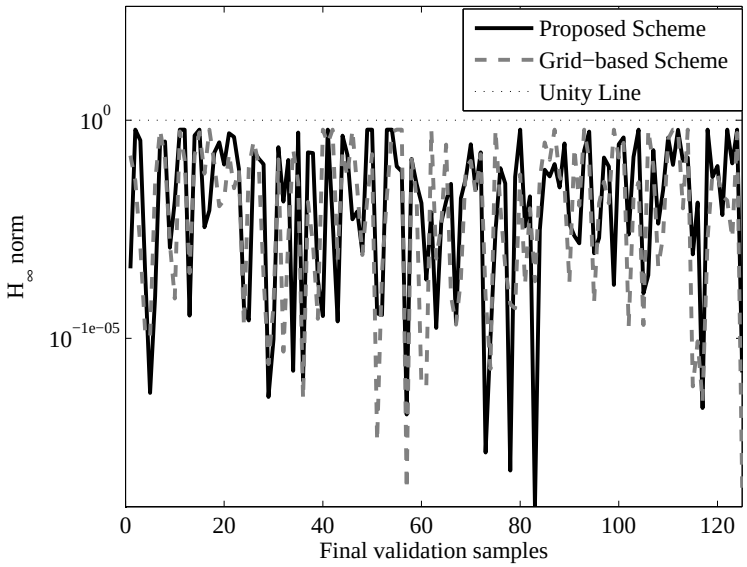

Figure 6.22: Example II: H_∞ norm for the final verification samples.

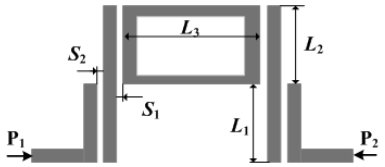

Figure 6.23: Example III: Layout of ring resonator bandpass filter.

L_2 and S_1 respectively. The dotted curves in Figs. 6.24, 6.25 and 6.26 represent the response of the parametric macromodel obtained from the proposed method. As seen a good agreement can be observed. For this example, first a parametric macromodeling is performed using the proposed and the grid method generating two different macromodels and then as a next step, a global optimization is performed on these two macromodels generated. Finally, the modeling part as well as the optimization part for the two sequential sampling algorithms are compared in terms of important parameters.

Macromodeling of the ring resonator filter

As in Example I and II, The proposed algorithm and the grid method [4] have been implemented in Matlab R2012a and used to drive the ADS Momentum simulations to generate S-responses at selected samples. As in the previous examples, the

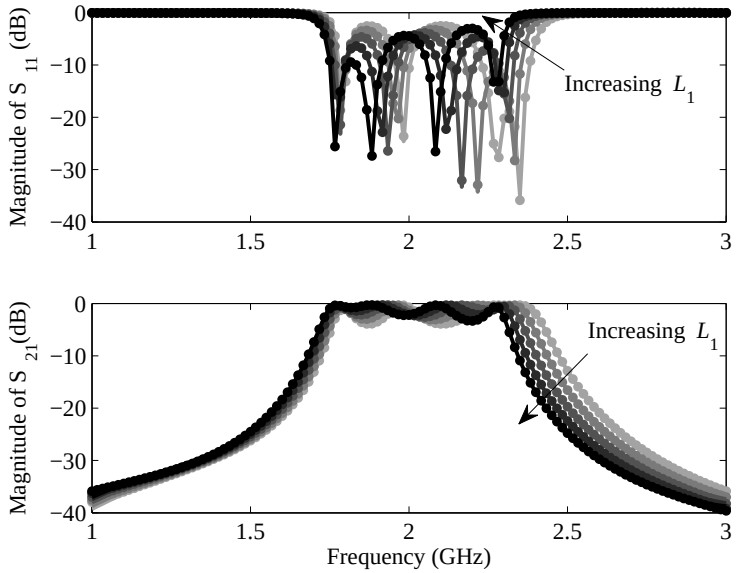

Figure 6.24: Example III: Magnitude of S_{11} and S_{21} as a function of L_1.

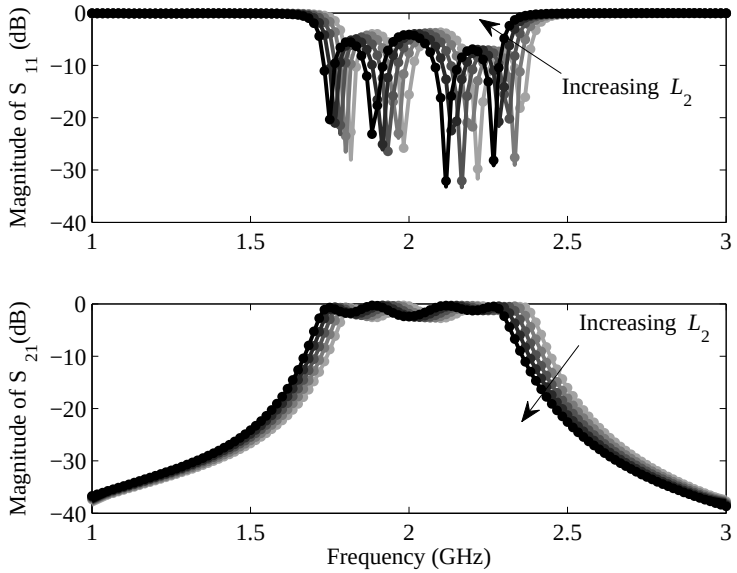

Figure 6.25: Example III: Magnitude of S_{11} and S_{21} as a function of L_2.

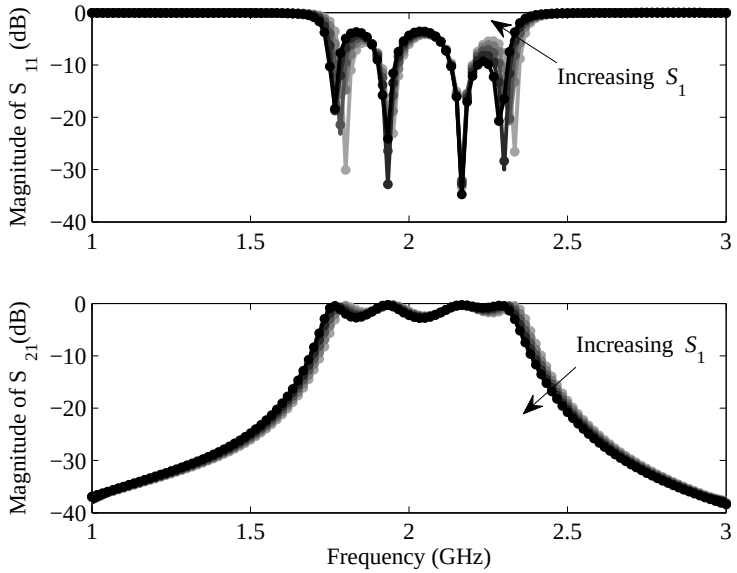

Figure 6.26: Example III: Magnitude of S_{11} and S_{21} as a function of S_1.

proposed algorithm is compared with the grid method on 250 verification points spread across the design space using a Latin hypercube space filling. The MEA measure (6.7) was used to estimate the modeling accuracy.

Table 6.5 compares the two sequential sampling schemes over some important parameters such as the number of samples required, the worst-case accuracy over the 250 verification samples, the maximum macromodel order and the speed-up.

Sampling Method	# Samples		Max. Order	Error (dB)	Evaluation Time [s]	Speed-up
	Gen.	Val.				
Proposed	306	-	156	-42.92	0.3156	434 ×
Grid [4]	405	66	832	-42.03	1.9584	70 ×
CPU time for EM simulator to calculate a single frequency response = 137 s.						

Table 6.5: Example III: Comparison of different sampling strategies.

So, as in the case of Example I and II, the following can be observed from the table:

i. considerable reduction in the number of samples and consequently in overall complexity for the proposed scheme over the grid method,

ii. computational complexity of the parametric macromodel generated using

the grid method is higher than the proposed scheme as explained in Section 6.1.3 in Table 6.2, and finally

iii. Comparable accuracy for the proposed method and the grid method is observed.

Optimization of the ring resonator filter

The two parametric macromodels generated using the grid method and the proposed method were used in a design optimization scenario. The design specifications of the filter are given in terms of the scattering parameters S_{21}:

$$|S_{21}| > -2.0\,\mathrm{dB} \;\text{ for } 1.75\,\mathrm{GHz} \leq freq \leq 2.25\,\mathrm{GHz} \tag{6.8a}$$

$$|S_{21}| < -25\,\mathrm{dB} \;\text{ for } freq < 1.5\,\mathrm{GHz}\,,\; freq > 2.5\,\mathrm{GHz} \tag{6.8b}$$

From the design specifications (6.8), a cost function is formulated in terms of S_{21} and frequency. A global optimization method based on the DIviding RECTangle (DIRECT) strategy [26] is used to minimize the cost function. The method [26] balances between a global and local search and finds an optimization solution $\vec{g}^* = (L_1^*, L_2^*, L_3^*, S_1^*, S_2^*)$. Since the cost functions are generated with the help of the parametric macromodels generated with the grid method [4] and the proposed sampling scheme, the evaluation time is much less in comparison with the actual ADS simulations.

Sampling method	Proposed	Grid method [4]
# model evaluations	802	802
Optimization time (s)	311.6	2099.5
Final solution $\vec{g}^*$ (mm)	[22.96, 21.00, 26.67, 0.30, 0.06]	[22.96, 21.00, 26.67, 0.30, 0.06]
Optimum cost	-0.0018	-0.0018

Table 6.6: Example III: Optimization results.

It can be observed from Table 6.6 that the average macromodel evaluation time per sample for the grid method is relatively higher in comparison with the proposed scheme. This is indeed because of the increasing complexity of the macromodel generated using the grid-based interpolation in comparison with the scattered interpolation as described before (Table. 6.5). However, both the macromodels finds a single optimum and is also verified with ADS Momentum simulation and is shown in Fig. 6.27. In the figure the requirements (6.8) are shown by the dotted lines. The two responses generated by the ADS Momentum at the solutions given in Table 6.6 are also shown in Fig. 6.27 which satisfy the requirements.

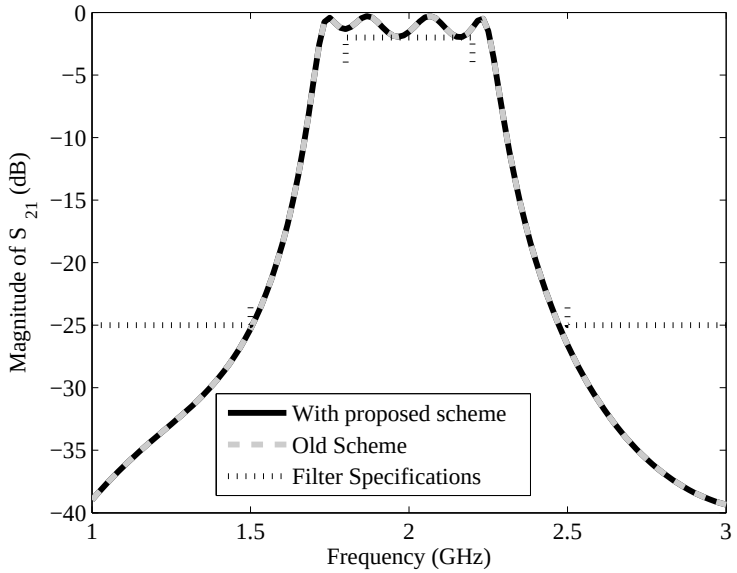

Figure 6.27: Example III: ADS Momentum response at the optimal solutions generated with the two macromodels.

6.2.7 Discussions

In this section, a brief discussion about the relative pros and cons of the hybrid method with respect to the grid-based sampling scheme of [4] is given.

The **advantages** of using such an approach are:

i. Preserving the structure of grid is no more needed as in [4] which helps in refining the grids into finer regions.

ii. The interpolated parametric macromodels have a smaller rational order in comparison with the grid-based scheme as it is evident from Table 6.2 and also from the three numerical examples of Section 6.2.6. This generate much compact parametric macromodels.

iii. The model-based refinement strategy can be used here since the comparison is also done between a local parametric macromodel which is on a hyperrectangular grid and the one on a simplicial grid. This ensures that two different models are compared which gives more credibility to the estimated error. A model-based refinement strategy reduces the number of EM simulations considerably.

Unfortunately, the proposed approach also has some **limitations** which are:

i. It is difficult to go for an output-based validation as in [4] because each hyperrectangular grids generate $N!$ path-simplexes. The validation of the generated simplexes (at their incenters) with EM simulations becomes expensive. This is also a reason for model-based refinement strategy.

ii. The model-based refinement strategy gives good estimation of the error, but is often optimistic which is a limitation of this method. It is also difficult to quantify the estimated error with respect to the true error. Some work is needed in improving such an algorithm so as to get rid of the validation samples and at the same time getting a good estimate of the modeling error.

iii. In the grid-based scheme, the advantage is that along the edges only one parameter is varied at a time which is an advantage for the local parametric macromodeling. This is not possible with scattered scheme which can put additional modeling challenge on the simplicial refinement schemes.

In conclusion, it is true that the hybrid scheme also has some limitations with respect to the grid-based scheme. Nevertheless, the scheme is able to give good results despite these limitations as it is evident from the three numerical examples. It is to be noted here that this is the state-of-the-art for the scattered sampling scheme which automate the property-preserving parametric macromodeling process for EM systems and needs further investigation.

6.3 Conclusions

This chapter is divided into two parts: in the first part, a scattered sampling algorithm for the automatic construction of stable and passive macromodels of parameterized system responses. The proposed method avoids the generation of slivers by using path-simplex based refinement. The proposed technique is validated on a pertinent numerical example.

A hybrid sequential sampling scheme on scattered grids for automatic construction of parametric macromodels for microwave systems is presented in the second part of this chapter. The method uses a passivity-preserving parametric macromodeling method along with a sampling based on well-conditioned simplicial refinement strategy to build the parametric macromodels. The method is also compared with previous hyperrectangular sampling method in terms of several important parameters. Three pertinent numerical examples show the modeling capability of the proposed sequential sampling method with less computational resource requirements. One of the example also demonstrates a design optimization scenario using the generated macromodels. These numerical examples show the advantages of using the proposed method in generating accurate parametric macromodels automatically from the EM simulation data, over the design space of interest, with minimum computational resources.

6.4 Acknowledgements

This work was supported by the Research Foundation Flanders (FWO) and by the Interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office.

References

[1] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving interpolation-based parameterized macromodeling of scattered s-data," *IEEE Microwave and Wireless Components Letters,*, vol. 20, no. 3, pp. 133 –135, March 2010.

[2] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139–151, 2012.

[3] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2394 –2403, Oct. 2011.

[4] K. Chemmangat, F. Ferranti, T. Dhaene, and L. Knockaert, "Scalable models of microwave system responses using sequential sampling on unstructured grids," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 27, no. 1, pp. 122–137, 2014.

[5] J. Brandts, S. Korotov, and M. Krizek, "Dissection of the path-simplex in $\mathbb{R}^n$ into n path-subsimplices," *Linear Algebra and its Applications*, vol. 421, no. 23, pp. 382 – 393, 2007.

[6] J. Brandts, S. Korotov, M. Krizek, and J. Solc, "On nonobtuse simplicial partitions," *SIAM Review*, vol. 51, no. 2, pp. 317–335, 2009.

[7] X. Li, "Generating well-shaped d-dimensional Delaunay meshes," *Computing and Combinatorics*, pp. 91–100, 2001.

[8] A. Constantiniu, P. Steinmann, T. Bobach, G. Farin, and G. Umlauf, "The adaptive delaunay tessellation: a neighborhood covering meshing technique," *Computational Mechanics*, vol. 42, no. 5, pp. 655–669, 2008.

[9] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," vol. 14, no. 3, pp. 1052–1061, July 1999.

[10] B. Gustavsen, "Fast passivity enforcement for S-parameter models by perturbation of residue matrix eigenvalues," vol. 33, no. 1, pp. 257–265, Feb. 2010.

[11] A. Inselberg, "FT-1 the plane with parallel coordinates," in *Parallel Coordinates*. Springer New York, 2009, pp. 49–61.

[12] D. Deschrijver and T. Dhaene, "Stability and passivity enforcement of parametric macromodels in time and frequency domain," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 56, no. 11, pp. 2435–2441, 2008.

[13] F. Ferranti, L. Knockaert, and T. Dhaene, "Parameterized S-parameter based macromodeling with guaranteed passivity," *IEEE Microwave and Wireless Component Letters*, vol. 19, no. 10, pp. 608–610, Oct. 2009.

[14] ——, "Guaranteed passive parameterized admittance-based macromodeling," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 3, pp. 623–629, Aug. 2010.

[15] P. Triverio, M. Nakhla, and S. Grivet-Talocia, "Passive parametric macromodeling from sampled frequency data," *IEEE International Conference on Signal Propagation and Interconnects*, pp. 117 –120, May 2010.

[16] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on Lur'e equations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 12, pp. 3688 –3696, Dec. 2010.

[17] S. Peik, R. Mansour, and Y. Chow, "Multidimensional Cauchy method and adaptive sampling for an accurate microwave circuit modeling," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 46, no. 12, pp. 2364 –2371, Dec. 1998.

[18] A. Lamecki, P. Kozakowski, and M. Mrozowski, "Efficient implementation of the Cauchy method for automated CAD-model construction," *IEEE Microwave and Wireless Components Letters,*, vol. 13, no. 7, pp. 268 –270, July 2003.

[19] A. Cuyt, R. Lenin, S. Becuwe, and B. Verdonk, "Adaptive multivariate rational data fitting with applications in electromagnetics," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 54, no. 5, pp. 2265 – 2274, May 2006.

[20] V. Devabhaktuni, B. Chattaraj, M. Yagoub, and Q.-J. Zhang, "Advanced microwave modeling framework exploiting automatic model generation, knowledge neural networks, and space mapping," *IEEE Transactions on Microwave Theory and Techniques,*, vol. 51, no. 7, pp. 1822 – 1833, July 2003.

[21] A. Lamecki, L. Balewski, and M. Mrozowski, "Adaptive CAD-model construction schemes," *IEEE Transactions on Magnetics,*, vol. 45, no. 3, pp. 1538 –1541, March 2009.

[22] P. Basl, R. Gohary, M. Bakr, and R. Mansour, "Modelling of electromagnetic responses using a robust multi-dimensional Cauchy interpolation technique," *IET Microwaves, Antennas Propagation,*, vol. 4, no. 11, pp. 1955 –1964, Nov. 2010.

[23] R. Rohrer and H. Nosrati, "Passivity considerations in stability studies of numerical integration algorithms," *IEEE Transactions on Circuits and Systems,*, vol. 28, no. 9, pp. 857 – 866, Sep. 1981.

[24] A. Weiser and S. E. Zarantonello, "A note on piecewise linear and multilinear table interpolation in many dimensions," *Mathematics of Computation*, vol. 50, no. 181, pp. 189–196, Jan. 1988.

[25] K. Chemmangat, T. Dhaene, and L. Knockaert, "Scalable macromodelling of microwave system responses using sequential sampling with path-simplexes," *Electronics Letters*, vol. 49, no. 15, pp. 950–952, 2013.

[26] Y. D. Sergeyev and D. E. Kvasov, "Global search based on efficient diagonal partitions and a set of lipschitz constants," *SIAM Journal on Optimization*, vol. 16, no. 3, pp. 910–937, 2006.

7 Conclusions

7.1 General Conclusions

In this PhD thesis, the work was focused towards developing robust data-driven parametric macromodeling strategies for high-frequency linear time-invariant systems using automated sequential sampling strategies and parametric macromodels. Developing modeling tools for generating accurate parametric macromodels for such systems is crucial for an efficient and successful design. These parametric macromodels replace the expensive simulators in the design cycle and hence reduce the computational cost. They additionally give much more insight for the designer to understand the system. They also can be used in different design activities such as design space optimization, sensitivity analysis, robustness studies, etc., which is a very significant added advantage.

In Chapter 2, an introduction and overview of the several state-of-the-art parametric macromodeling methods is given. Their relative merits and limitations have also been addressed here. The need for having an automated parametric macromodeling scheme which can be efficiently used in the design process were also discussed here. The general flow of the parametric macromodeling starting from the rational identification of the frequency responses and different parameterization methods for the rational models over the design space were introduced. In Chapter 3, an extension of the state-of-the-art parametric macromodel using interpolation of state-space matrices is performed to generate parametric sensitivity macromodels. These parametric sensitivity macromodels can generate analytic sensitivities

or gradients of the output responses with respect to the design parameters which can be used in gradient-based design optimizations or sensitivity analysis. Chapter 4 is concerned with bringing automation into the state-of-the-art parametric macromodeling techniques. The sampling schemes developed in this PhD thesis are tailored towards the local parametric macromodeling schemes which refines local regions of the design space. In Chapter 4, two different grid-based sampling schemes have been developed to sample the design space adaptively. Chapter 5 uses the fully automated parametric macromodeling process with sequential sampling to build a real-life example of a microwave filter. In this chapter, the complete design flow of the filter is described which is assisted with the help of the parametric macromodel, thereby gaining insight into the design with very little computational cost. The chapter also shows the amount of design cycle time gained by using a parametric macromodel instead of the direct EM simulations in a robust global design optimization and a Monte-Carlo variability analysis. Additionally, Chapter 5 also proposes an enhanced parametric macromodeling strategy which parameterize the poles of the frequency response transfer function to have a better parametric macromodeling strategy. In Chapter 6, the grid-based refinement scheme for sequential sampling had been generalized to scattered refinement using well-conditioned simplicial refinements.

7.2 Major Research Contributions

The work done in this PhD thesis made the following contributions in the field of efficient and automated parametric macromodeling of LTI EM systems:

1. **Bringing automation** into the parametric macromodeling of system frequency responses was the most important contribution of this PhD work. Several sequential sampling strategies to optimally sample the design space were proposed such as the grid-based refinement schemes, the simplicial or scattered schemes with well-conditioned simplex refinement, and the hybrid scheme which combines both grid-based and simplicial refinement methods.

2. **Selecting suitable parametric macromodeling methods** from existing state-of-the-art methods for the efficient sequential sampling process. **Improving existing parametric macromodeling methods** to increase their modeling power and robustness so as to reduce the number of expensive EM simulations for modeling.

3. **Choosing appropriate error measures** from several possibilities for assessing the model accuracy, and using them in the parametric macromodeling process. The error measure has a significant influence on the number of expensive EM simulations required for modeling and an appropriate one

should be selected to ensure the generation of an efficient replacement model for the EM simulator with as little expensive EM simulations as possible.

4. **Validating such an automated modeling procedure** by incorporating it in the design flow of a real-life microwave filter example. The automated modeling process is applied to the microwave filter and the parametric macromodel generated is used in the complete design cycle with several steps such as design space optimization and variability analysis. The optimal design solution is then fabricated and measured.

7.3 Possible Improvements and Future Directions

7.3.1 Scalability to Higher dimensions

Even though the parametric macromodeling techniques developed are very efficient, these methods, like any other modeling method, are limited by the *curse of dimensionality* which refers to the difficulties associated with analysis in higher dimensional spaces [1]. Building interpolation models for parameterized frequency response systems which are accurate over the complete design space of interest can be quite challenging as the dimensionality increases. Considering the corner frequency response simulations itself become computationally very expensive as it is exponentially proportional to the number of design parameters.

Several screening methods such as Morris method, Sobol method and Fast Amplitude Screening Test (FAST) may be investigated in the future as a possible solution. These methods screen the important parameters by prioritizing them based on their influence on the system response to be modeled [2]. Once the less important parameters are screened out the parametric macromodeling can be performed on the reduced set of design parameters. Also, another possibility, which could be investigated is to use state-of-the-art dimensionality reduction techniques such as Principle Component Analysis (PCA), Kernel PCA and Local Linear Embedding (LLE) [3].

7.3.2 Parallelization of the Sequential Sampling Algorithm

Parallel computing capability is nowadays very common with computers which use multiple processors to carry out several independent tasks. For computationally expensive tasks, parallel computing can be a way to significantly reduce the corresponding computational cost. As explained in the introduction of this PhD thesis, the sequential sampling algorithms developed during this work are in a tree-fashion with independent branches modeling different parts of the design space. This can thus speed-up the parametric macromodeling process considerably. As

an improvement, standard computational tools such as MATLAB Parallel Computing Toolbox [4] can be used to perform the parallelization of the sequential sampling algorithms developed in this PhD thesis.

7.3.3 Applications to Other Engineering Fields

The work done in this PhD thesis is focused towards developing parametric macromodels for the system frequency responses of electromagnetic systems. These systems fall under the framework of LTI systems which finds it application in other disciplines such as mechanical engineering, chemical engineering and signal processing. For example, a parametric macromodel for the system frequency response can be built for the 3D cantilever system of [5]. Design parameters such as length of the beam L, the height of the beam h and thickness t can be considered. Thus as a possible future work, the state-of-the-art parametric macromodeling method can be extended to other research areas and applications.

References

[1] D. Deschrijver, A. Narbudowicz, E. Laermans, and T. Dhaene, "On the application of dimensional analysis to parametric macromodeling," *Microwave and Wireless Components Letters, IEEE*, vol. 20, no. 4, pp. 190–192, 2010.

[2] A. Boukouvalas and D. Cornford, "Screening strategies for high dimensional input spaces," *Technical report, Neural Computing Research Group, Aston University*, 2007.

[3] L. Van der Maaten, E. Postma, and H. Van Den Herik, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, pp. 1–41, 2009.

[4] G. Sharma and J. Martin, "Matlab®: a language for parallel computing," *International Journal of Parallel Programming*, vol. 37, no. 1, pp. 3–36, 2009.

[5] H. Panzer, J. Hubele, R. Eid, and B. Lohmann, "Generating a parametric finite element model of a 3d cantilever timoshenko beam using matlab," *Technical Reports on Automatic Control*, vol. TRAC-4, 2009.

A

Implementation Details of the Sequential Sampling Algorithms

In this Appendix, the implementation details of both grid-based and scattered sequential sampling schemes discussed in Chapter 4 and 6 respectively will be explained. The factors such as the memory requirement, the storage structure of the parametric macromodeling coefficients and automatic selection of the order of the root macromodel will be described in more detail.

A.1 Tree-based algorithm

As explained in Chapter 4, the sequential sampling algorithm is implemented in a tree-like fashion. The word "tree" in the discussion here is mapped to the hyperrectangular space formed by the design parameters of interest which is referred to as the design space. The sequential sampling algorithm whether it is scattered or grid-based, starts from a single hyperrectangular design space. This can be considered as the main stem of the tree which is also referred to as a node. Then based on the refinement criteria, the node or the main stem is branched to different subregions. The refinement can be grid-based or scattered and it is usually modeling error driven. After finishing all the error-based refinement, the terminal nodes are called the leaves of the tree and they represent the accurate subregions of the initial hyperrectangular space.

This kind of implementation has certain advantages over the global sampling.

The algorithm divide different regions of the design space independently. Thus, parallelization is very much possible here. If a local parametric macromodeling scheme which builds different models for different sub-regions of the design space is used, parallelization will be very effective. This will increase the computational efficiency by making use of the multiple core processing power.

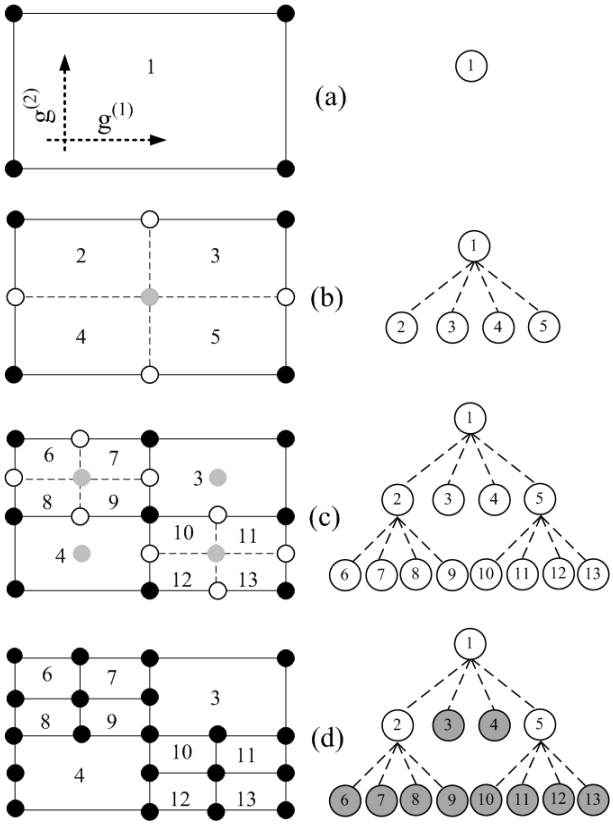

Figure A.1: Evolution of the design space during sampling.

To understand the tree-based sequential sampling implementation, lets consider a 2 parameter design space with two design parameters $(g^{(1)}, g^{(2)})$, for which the design space is defined by four corners of a rectangle as shown in Fig. A.1.a. Here the grid-based sequential sampling scheme of Chapter 4 is used to explain the concept. For the local parametric macromodeling methods discussed in this PhD thesis, frequency is treated as a special parameter and does not belong to the sampling space. Here the frequency responses are approximated as rational functions of complex Laplace variable s identified with the help of the well-known Vector Fitting (VF) technique [1]. Thus each of the corner points in Fig. A.1.a represent

a frequency dependent rational model or a state-space model called *root macromodel*. Parametric macromodels are generated by interpolating the *root macromodels* and the subspace is refined using some error criteria which compares the EM simulation response with the parametric macromodel at some validation point (for example, light gray circle in Fig. A.1.b). This refinement procedure is repeated until the modeling error satisfies a predefined error threshold Δ_{err}. Note that the same procedure can be applied for scattered sequential sampling wherein only the refinement strategy changes from hyperrectangular refinement to simplicial refinement (Chapter 6). As shown in Fig.A.1, the algorithm can be represented in a tree structure with each circle representing a particular subspace.

Following points can be observed from the tree-based implementation:

1. as stated before, each of the points in the hyperrectangular region represent a frequency-dependent *root macromodels*. The sub-regions irrespective of scattered or grid-based refinement strategies, share these *root macromodels* along their boundaries. Thus the memory requirement of storing the *root macromodel* coefficients is proportional to the number of sample points.

2. The local parametric macromodels are different for different sub-regions of the design space. This means that unlike the global modeling schemes, the parameterization variables such as interpolation coefficients are not shared among the sub-regions. Consequently, with the number of sub-regions generated by the sampling scheme, the memory requirement for storing the parameterization variables increases proportionally.

In the next section, the parametric macromodeling is explained in detail with the implementation details such as how the code is organized and the model parameters are stored.

A.2 Parametric macromodeling

This section gives a detailed description of the implementation of parametric macromodeling with the tree-based sequential sampling. The local parametric macromodeling method of [2, 3] which uses a robust interpolation scheme is used in this section for description purpose. Other local parametric macromodeling schemes can also be used. Also, the 2D design space considered in Section A.1 and the associated Fig. A.1 are used to explain the concept.

A.2.1 Generation of *root macromodels* and their storage

The first step in these macromodeling methods is to generate a set of *root macromodels* which form the interpolation nodes for the modeling. The decision has to be made about the order of the rational model to fit the frequency response using

the VF method. In this PhD thesis, to ensure complete automation in the modeling process, a bottom-up approach is used. That is, the order of the rational approximation is gradually increased and the VF accuracy is calculated. Once a desirable accuracy is reached with the minimum order of the rational function, the iteration is stopped and the order is fixed at the found out value. This is illustrated in Fig. A.2.

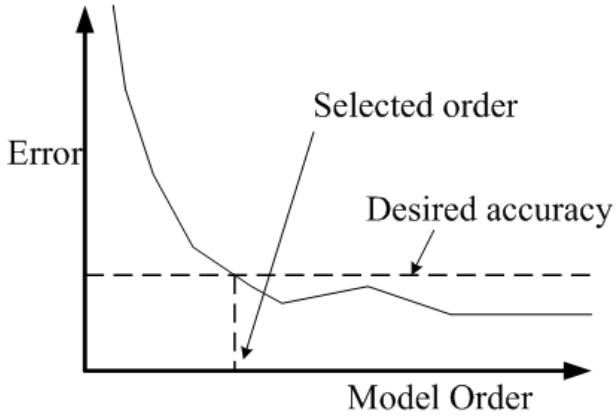

Figure A.2: Model order selection of a root macromodel.

Once the rational model is calculated, it can be stored in the modeling database in the form of either rational functions or state-space models. The rational *root macromodels* $\mathbf{R}^{\Omega}(s, \vec{g}_{rm}^{\,\Omega})$, $rm = a, b \ldots, s$ contained in the N-box region Ω_l are represented in a pole-residue form:

$$\mathbf{R}^{\Omega}(s, \vec{g}_{rm}^{\,\Omega}) = \sum_{p=1}^{P_{rm}^{\Omega}} \frac{C_{p,rm}^{\Omega}}{s - a_{p,rm}^{\Omega}} + D_{rm}^{\Omega} \;\; ; p = 1, \ldots, P_{rm}^{\Omega} \tag{A.1}$$

where $C_{p,rm}^{\Omega}$ represents the residue matrices, $a_{p,rm}^{\Omega}$ the pole P_{rm}^{Ω}. This can either be a real pole or one of the members of a complex-conjugate pole pair. D_{rm}^{Ω} is the direct-term matrix.

It should be noted that, there will be one *root macromodel* per design space sample point. The *root macromodels* are shared between different sub-regions of the design space and they are mapped from the sub-space to the *root macromodel* array using indexing.

Fig. A.3 illustrates this concept. Fig. A.1.d is reproduced here for the purpose. In Fig. A.3, the *root macromodels* are named from alphabet a to s. For example, it is seen that the *root macromodel* j is shared between four sub-regions of the design space namely 10, 11, 12 and 13. And for example, for sub-region 4, the

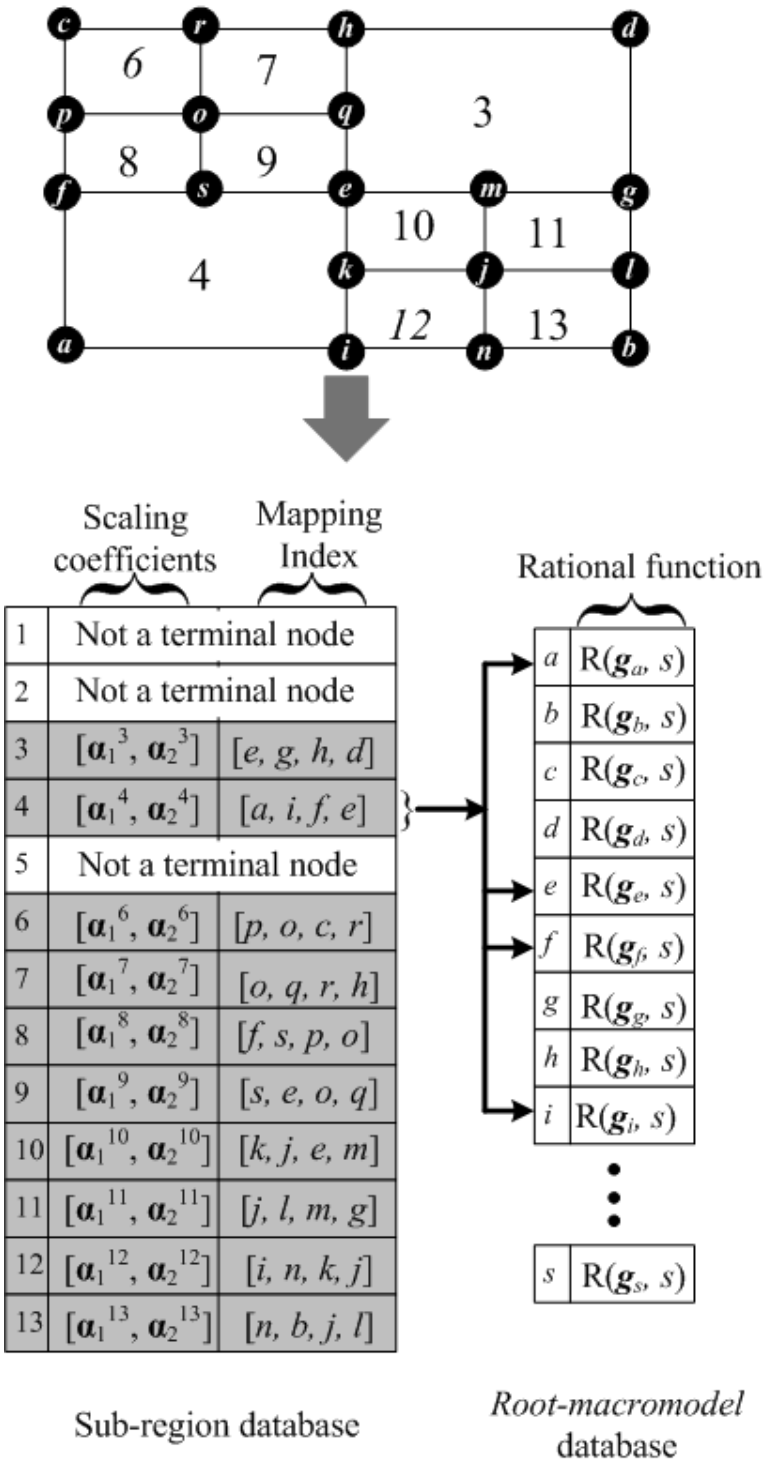

Figure A.3: Sample distribution and the corresponding modeling database.

index which maps the sub-region database to the *root macromodel* database is give as

$$\text{Mapping Index}(4) = [a, i, f, e], \tag{A.2}$$

which is also shown in Fig. A.3.

A.2.2 Calculation of model parameters and storing them

This subsection describes briefly the amplitude and frequency scaling coefficient-based parametric macromodeling scheme of [2, 3] and describes how these coefficients are stored in the sub-region database (Fig. A.3).

Lets consider the design space region $\Omega = 4$ which is defined by four bounding corner *root macromodels* with indexes $[a, i, f, e]$ as in Fig. A.3. For any design space region Ω, several amplitude scaling and frequency scaling coefficients $(\vec{\alpha}_1^{\Omega}, \vec{\alpha}_2^{\Omega})$ are calculated [2]. For example, the pair (a, i) of the region $\Omega = 4$ the coefficients $(\alpha_{1,ai}^4, \alpha_{2,ai}^4)$ are calculated using the optimization

$$(\alpha_{1,ai}^{\Omega}, \alpha_{2,ai}^{\Omega}) = \underset{(\alpha_1,\alpha_2)}{\text{argmin}} \left[\text{Err}(\tilde{\mathbf{R}}_i(s, a), \mathbf{H}(s, i))\right]. \tag{A.3}$$

In (A.3), $\tilde{\mathbf{R}}^i(s, a) = \alpha_1 \mathbf{R}(s\alpha_2, \vec{a})$, is the interpolated response of $\mathbf{R}(s, a)$ obtained to match $\mathbf{R}(s, i)$ and $\text{Err}(\cdot)$ is a suitable error measure between the two responses [2].

From (A.3), for the region $\Omega = 4$, the amplitude and frequency scaling coefficients can be expressed in the form of row vectors as

$$\begin{aligned}\vec{\alpha}_x^4 = [&\alpha_{x,ai}^4, \alpha_{x,af}^4, \alpha_{x,ae}^4, \alpha_{x,ia}^4, \alpha_{x,if}^4, \alpha_{x,ie}^4,\\ &\alpha_{x,fa}^4, \alpha_{x,fi}^4, \alpha_{x,fe}^4, \alpha_{x,ea}^4, \alpha_{x,ei}^4, \alpha_{x,ef}^4].\end{aligned} \tag{A.4}$$

In case of the parametric macromodeling method of [2, 3], we have for N-dimensional design space,

$$\text{Number of } 2-\text{variable optimizations} = \frac{1}{2}\frac{N_c!}{(N_c-2)!}, \tag{A.5}$$

$$\text{Number of elements in } (\vec{\alpha}_1^{\Omega}, \vec{\alpha}_2^{\Omega}) = 2\frac{N_c!}{(N_c-2)!}, \tag{A.6}$$

where, N_c is the number of corner points which is equal to 2^N and $N + 1$ for grid-based and scattered schemes respectively. Similarly, for multiple frequency scaling based parametric macromodeling method of Chapter 5, for N-dimensional

design space we have

$$\text{Number of multi} - \text{variable optimizations} = \frac{N_c!}{(N_c-2)!}, \tag{A.7}$$

$$\text{Number of elements in } \vec{\alpha}_1^{\Omega}, = \frac{N_c!}{(N_c-2)!}, \tag{A.8}$$

$$\text{Number of elements in } \vec{\alpha}_2^{\Omega} = \frac{N_c!}{(N_c-2)!} \sum_{n=1}^{N_c} \left(P_n^{\text{real}} + P_n^{\text{comp}} \right), \tag{A.9}$$

with P_n^{real}, real and P_n^{comp} complex poles of $n = 1, \ldots N_c$ corner *root macromodels*.

References

[1] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," vol. 14, no. 3, pp. 1052–1061, July 1999.

[2] F. Ferranti, L. Knockaert, and T. Dhaene, "Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2394 –2403, Oct. 2011.

[3] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini, "Parametric macromodeling based on amplitude and frequency scaled systems with guaranteed passivity," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 25, no. 2, pp. 139–151, 2012.