Parameterized Modeling of Multiport Passive Circuit Blocks

by

Zohaib Mahmood

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2010

Author ....
Department of Electrical Engineering and Computer Science
August 22, 2010

Certified by ....
Luca Daniel
Associate Professor
Thesis Supervisor

Accepted by ....
Terry Orlando
Chairman, Department Committee on Graduate Theses

Parameterized Modeling of Multiport Passive Circuit Blocks

by

Zohaib Mahmood

Submitted to the Department of Electrical Engineering and Computer Science
on August 22, 2010, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

System level design optimization has recently started drawing the attention of circuit designers. A system level optimizer would search over the entire design space, adjusting the parameters of interest, for optimal performance metrics. These optimizers demand for the availability of parameterized compact dynamical models of all individual modules. The parameters may include geometrical parameters, such as width and spacing for an inductor or design parameters such as center frequency or characteristic impedance in case of distributed transmission line structures. The parameterized models of individual blocks need to be compact and passive since the optimizer would be solving differential equations (time domain integration or periodic steady state methods) to compute the performance metrics. Additionally, these parameterized models would be able to facilitate the job of the designer who could instantiate the models with different parameter value during manual optimization.

In this thesis, we have designed and implemented various highly efficient algorithms for the identification of individual and parameterized models for multiport passive structures. The algorithms are based on convex relaxations of the original non-convex problem consisting of modeling multiport devices from frequency response data. Passivity is enforced in the final models by constrained fitting, where the constraints are either Linear Matrix Inequalities or semidefinite constraints. These individual non-parameterized models can be used for system level simulations for fixed parameter values or for building up a parameterized model. In the first algorithm, we identify a collection of first and second order networks to model individual non-parameterized passive blocks. Passivity of the overall model is guaranteed by enforcing passivity on the individual building blocks. In the second algorithm we exploit the property of causal and stable systems for which the real and imaginary parts of the frequency response are related by the Hilbert transform, by minimizing only the mismatch between real parts. Passivity is enforced in the identified model using semidefinte constraints.

In this thesis we also propose an algorithm for generating parameterized multiport models of linear systems that the user will be able to instantiate for any parameter value, always obtaining a stable and passive model. Our approach uses constrained optimization to construct a parameterized model that optimally fits a set of given non-parameterized models using polynomial or rational basis. By using optimization, as opposed to interpolation as in the available parameterized modeling techniques, we are capable of guaranteeing global passivity with respect to the parameters, while simultaneously keeping the number of terms describing the model small.

The proposed algorithms are supported by various modeling examples including Wilkinson combiners, power and ground distribution grid, on-chip coupled inductors, microstrip patch antenna and parameterized attenuator. The identified models are verified for passivity using the Hamiltonian matrix based eigenvalue test. Several comparisons with existing techniques are also provided, which demonstrate a promising speed up of 40× in some cases and an amazing efficiency, by generating a highly accurate model in the cases where alternative techniques even failed to generate the model.

Thesis Supervisor: Luca Daniel
Title: Associate Professor

Acknowledgments

I would like to thank Prof Luca Daniel for his constant support and guidance over the last several years. I am grateful to him for all the creative discussions we've had which inspired several new ideas presented in this research. It would not have been possible for me to complete this research without his advice and constant mentoring.

I have had been working very closely with other faculty members including Prof. Alex Megretski, Prof. Vladamir Stojanovic and Prof. Joel Dawson. I am obliged to all of them for their time and feedback on my research.

I am greatly indebted to Brad Bond for getting me started with this new field. Our first interaction started when he was TAing 6.336. I never missed office hours so that I could get his words of wisdom on numerical simulations. I am thankful to him for making 6.336 such an exciting experience. Later we continued to collaborate very closely on different topics including nonlinear system identification and the research presented in this thesis. I am grateful to him for always having a time for technical discussions. He has been a great source of learning and inspiration for me.

Working with Computational Prototyping Group has been a great pleasure. I would like to thank Tarek Moselhy and Yu-Chung Hsiao for sharing the office with me and having some very useful discussions. I am also grateful to Omar Mysore with whom I've been collaborating on different projects.

I am extremely grateful to Faisal Kashif, who has always acted like a big brother even before I started my career at MIT. He has always been there to answer my mostly random questions and helping me to develop a better view of life. He has been a great source of inspiration to me. Although being an international student I've never felt home sick mainly because of the great companionship of Faisal Kashif, Zia Rizvi, Hassan Bukhari and Nabeel Ahmad who have always made me feel at home.

I would like to thank the MIT Office of the Provost for awarding me the Irwin Mark Jacobs and Joan Klein Jacobs Presidential Fellowship, which supported my research during my first year at MIT.

Last, but definitely not the least, I am wholeheartedly thankful to my family for their

love, care and encouragement throughout my life. Without their support, help and confidence in me, this journey could not even have started.

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

1.1 Motivation

Generation of accurate and passive dynamical models for linear multiport analog circuit blocks is a crucial part of the design and optimization process for complex integrated circuit systems. Quite often, these models also need to capture dependence of the system on design and geometrical parameters while providing apriori passivity and stability certificates for the entire parameter range of interest. These identified models are interfaced with commercial circuit simulators where they are used to perform transient simulations being interconnected with other circuit blocks. If any of these building blocks violate essential physical properties such as stability and passivity, then the overall interconnected system might turn out to be unstable, and hence causing the transient simulations to blow up.

Let us consider the typical design flow for an analog circuit block, say a distributed power amplifier, containing on-chip multiport passive interconnect structures, such as multi-primary transformers for power combining. The full system design is completed in two steps. As the first step, the interconnect passive structures are laid out in an electromagnetic field solver and then simulated for frequency response in the desired frequency band. The second step consists of developing a reduced model based on the frequency samples or system matrices extracted by the solver which can be incorporated into a circuit simulator (e.g. Spice or Spectre).

Once the models are generated, they are interfaced with the circuit simulators using

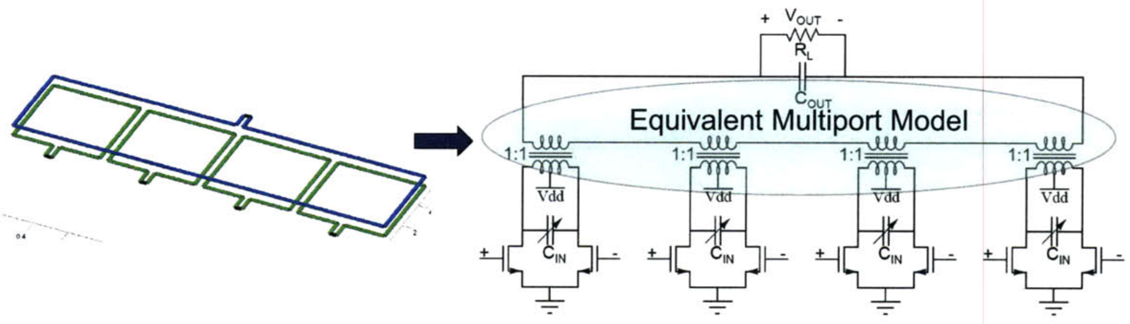


Figure 1-1: The model of a multiprimary transformer interfaced with circuit simulator to perform full distributed power amplifier simulations

either equivalent netlists or behavioral description. The final network in Figure 1-1 shows the interconnection of the generated model with other building blocks of the amplifier such as transistors and capacitors. Inside of the circuit simulator, transient simulations are performed in order to get performance metrics of the complete nonlinear power amplifier. If the generated model encounters a violation of any basic physical property of the structure, such as passivity, it can cause huge errors in the response of the overall system, and the results may become completely nonphysical.

In addition, the designers would greatly benefit from these models if these models are parameterized in some of the design parameters, such as width and spacing for an inductor or characteristic impedance in the case of distributed transmission line structures. These parameterized models will greatly reduce the design cycle by allowing the designer to instantiate the structure with various design parameters without performing another full-wave electromagnetic simulation. However these models are useful only if the final parameterized models come with apriori passivity certificates in the entire parameter range of interest.

1.2 Overview and Contributions of this Thesis

In this thesis we propose various highly efficient algorithms to automatically generate both individual non-parameterized and parameterized passive multiport models. In particular the main contributions of this thesis are summarized as follows:

- We have proposed a new algorithm to identify individual passive dynamical models.

The proposed algorithm identifies the unknown system in two steps. The first step is to identify a common set of stable poles for the multiport structure. The second step is to identify residue matrices which conform to passivity conditions. These passivity conditions are enforced as Linear Matrix Inequalities (LMIs).

- We have proposed another algorithm to identify individual passive dynamical model. This algorithm generates the passive model in a single step. Since the poles and residues are identified simultaneously, the identified model in this algorithm is near optimal. The passivity constraints in this algorithm require polynomial positivity. These constraints are enforced by using a semidefinite relaxation.

- We have proposed a new approach to develop globally stable and passive parameterized models. In this approach a collection of individual passive models is approximated by a single closed form model which conforms to passivity conditions in a continuous parameter range of interest.

The matlab implementations of various algorithms proposed in this thesis will be posted on public domain as free open source software at [1].

1.3 Organization of This Thesis

This thesis is organized as follows: Chapter 2 covers the relevant background on linear systems and passivity. Chapter 3 describes a brief overview of the existing techniques along with their advantages and shortcomings. Chapter 4 explains our convex formulation to identify passive system models in pole residue form. Chapter 5 describes our algorithm to identify passive linear system models as a ratio of complex polynomials. Chapter 6 covers the details how our passive models can be interconnected. Chapter 7 details how the proposed algorithm can be extended to identify globally passive parameterized models with apriori passivity certificates. Chapter 8 summarizes the thesis. Appendix A describes how some of the convex problems can be cast as semidefinte programs. All the theoretical developments are supported by examples which are provided at the end of relevant chapters.

1.4 Notation

In this thesis we use the following notations: Any letter with hat, such as $\hat{H}$, is used for the unknown or identified variables. The given samples and data points are indicated by using a subscripted letter such as H_i where i is the index. λ indicates the parameters, such as design or geometrical parameters, λ is also occasionally used as a symbol for eigen values, it is always specified in the context to which quantity λ is refering to. Real and imaginary parts of the complex quantities are denoted by prefixing $\Re$ and $\Im$ respectively.

Chapter 2

Background

2.1 Linear Time Invariant (LTI) systems

Dynamical systems are an extremely useful tool for the time-domain analysis of physical systems, as they provide a relationship between input signals $u(t)$ to an output signal $y(t)$ for the system. In state-space models, the evolution of the system is described by a state vector $x(t)$, which is controlled by input $u(t)$ and from which output $y(t)$ is determined. In the general form, a Linear Time Invariant (LTI) state-space model can be expressed as

$$\begin{aligned} \frac{d(x)}{dt} &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{2.1}$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{p \times n}$, and $D \in \mathbb{R}^{p \times p}$. Here n and p are the number of states and ports respectively. Linear systems can also be expressed in terms of a transfer matrix, by taking Laplace transform of (2.1), as

$$H(s) = \frac{Y(s)}{U(s)} \triangleq C(sI - A)^{-1}B + D = \begin{bmatrix} H_{11}(s) & \cdots & H_{1p}(s) \\ \vdots & \ddots & \vdots \\ H_{p1}(s) & \cdots & H_{pp}(s) \end{bmatrix} \tag{2.2}$$

Here $H_{ij}(s)$ indicates the transfer function from *Port j* to *Port i*.

2.2 Passivity

Passivity is one of the most important properties of dynamical systems. It describes the dissipative nature of the system which implies stability and causality. Passive systems (or models) are incapable of generating energy. Passivity is an essential property if the model, being interconnected with other systems, is to be used for time domain simulations, since arbitrary connections of passive systems are guaranteed to be passive. While it may be possible for a non-passive model to provide high accuracy in the frequency domain, the same model when used in time domain simulation could produce extremely inaccurate results resulting from passivity violations.

Consider a system described by input (say current) u and output (say voltage) y. Then $\langle u, y \rangle_\tau$ describes the total energy of the system upto time τ , where

$$\langle u, y \rangle_\tau = \int_{-\infty}^{\tau} y(t)^T u(t)\, dt. \tag{2.3}$$

Then the system is passive if

$$\langle u, y \rangle_\tau \geq 0, \forall \tau \in R^+ \tag{2.4}$$

Linear systems are usually described as a transfer matrix or as a state space model. Let us consider the passivity of impedance or admittance system specified by a transfer matrix $H(s)$. Passivity for an impedance or admittance system corresponds to *'positive realness'* of the transfer matrix. To be *positive real*, the transfer matrix $\hat{H}(s)$ must satisfy the following constraints

$$\overline{\hat{H}(\bar{s})} = \hat{H}(s) \tag{2.5a}$$

$$\hat{H}(s) \text{ is analytic in } \Re\{s\} > 0 \tag{2.5b}$$

$$\hat{H}(j\omega) + \hat{H}(j\omega)^{\dagger} \succeq 0 \ \forall\omega \tag{2.5c}$$

Where $\Re\{\ \}$ denotes the real part and $\dagger$ indicates the hermitian transpose.

The first condition (2.5a), commonly known as *conjugate symmetry*, ensures that the impulse response corresponding to $\hat{H}(s)$ is real. The second condition (2.5b) implies stability of the transfer function. A causal linear system in the transfer matrix form is stable if all of its poles are in the left half of the complex plane, i.e. all the poles have negative real part. The system is marginally stable if it has simple poles (i.e. poles with multiplicity one) on imaginary axis. The third and final condition (2.5c), which is *positivity condition*, implies positive realness of the symmetric part of the transfer matrix on the $j\omega$ axis.

2.2.1 Manifestation of Passivity for a Simple RLC Network

We consider a simple RLC network as shown in Figure 2-1. In this simple schematic we can analytically compute the equivalent of passivity conditions as follows

$$\begin{aligned} Z_{eq}(\omega) &= R + jX_{eq}(\omega) \\ &= R + j\frac{\omega L}{1-\omega^2 LC} \end{aligned} \tag{2.6}$$

Here $\Re Z_{eq} = R$. The passivity condition translates into R being non-negative, i.e. $R \geq 0$ in addition to L and C being non-negative.

2.3 Tests for Certifying Passivity

There are several tests, all based on positive real lemma, by which a model can be certified to be passive. Section 2.3.1 and 2.3.2 describe conditions which are both necessary and sufficient for passivity [8]. Section 2.3.3 describes a necessary condition for passivity.

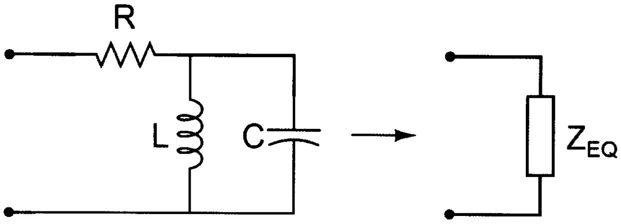


Figure 2-1: Manifestation of passivity for a simple RLC network where $Z_{eq}(\omega) = R + jX_{eq}(\omega)$. Passivity implies that $R, L, C \geq 0$

2.3.1 Tests Based on Solving a Feasibility Problem

When the system is represented in general state space form,

$$E\frac{d(x)}{dt} = Ax + Bu$$
$$y = Cx + Du$$

with minimal realization (i.e. every eigenvalue of A is a pole of $H(s)$), passivity is implied by the positive real lemma. The positive real lemma states that if there exists a positive definite matrix $P = P^T \succ 0$ and $P \in \mathbb{R}^{n \times n}$ such that the following matrix is negative semidefinite

$$\begin{bmatrix} E^TPA + A^TPE & E^TPB - C^T \\ B^TPE - C & -D - D^T \end{bmatrix} \preceq 0 \tag{2.7}$$

then $H(s)$ is positive real and hence the system is passive. Hence to certify if a system is passive, the feasibility problem for the existence of a positive definite matrix $P = P^T \succ 0$ can be solved. However such a formulation cannot be used efficiently to identify a passive system since it would then contain the product of unknowns, and is non-convex.

2.3.2 Tests Based on Hamiltonian Matrix

We can solve a condition equivalent to (2.7) based on Riccati equations and Hamiltonian Matrices. If we assume that $D+D^T \succ 0$, then the inequality (2.7) is feasible if and only if there exists a real matrix $P = P^T \succ 0$ satisfying the Algebraic Riccati Equation

$$A^T P + PA + (PB - C^T)(D + D^T)^{-1}(PB - C^T)^T = 0 \tag{2.8}$$

In order to solve the Algebraic Riccati Equation 2.8 we first form the associated Hamiltonian matrix M as follows

$$\begin{bmatrix} A - B(D+D^T)^{-1}C & B(D+D^T)^{-1}B^T \\ -C^T(D+D^T)^{-1}C & -A^T + C^T(D+D^T)^{-1}B^T \end{bmatrix}. \tag{2.9}$$

Then the system (2.1) is passive, or equivalently, the LMI (2.7) is feasible, if and only if M has no pure imaginary eigenvalues [8].

2.3.3 Sampling Based Tests - Only Necessary

Another class of passivity tests are based on checking the passivity conditions (2.5) at discrete frequency samples. Since passivity requires condition (2.5) to hold for all ω, a reduced model is non-passive if $\lambda_{\text{min}}(\Re\{\hat{H}(j\omega_i)\}) < 0$ for some ω_i. Here λ denotes the eigen values. Note that this test is based on a necessary condition and can only be used to check data samples or samples from the model for passivity violations and cannot be used to certify the passivity of a model.

2.4 Convex Optimization Problems

In this section we describe general convex optimization problems. Before discussing the optimization problems, we first describe the notion of convexity and convex functions in the following section. For detailed description we refer the readers to [9].

2.4.1 Convex and Non-Convex Functions

A function $f(x)$ is convex if the domain of $f(x)$ is convex and if for all $x, y \in domain\ f(x)$ and $\theta \in [0,1]$, it satisfies

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y) \tag{2.10}$$

Geometrically, this inequality means that the line segment between $(x, f(x))$ and $(y, f(y))$ lies above the graph of $f(x)$ as shown in Figure 2-2.

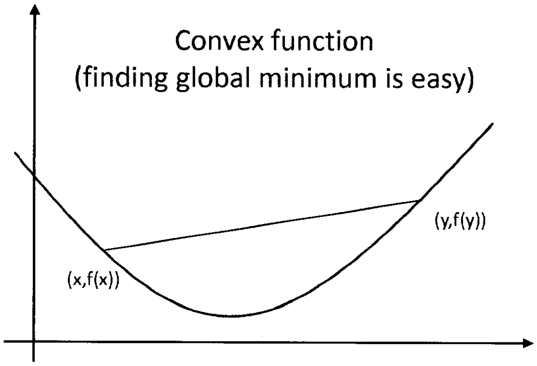


Figure 2-2: Shows a convex function. In general finding global minimum for convex functions is easy

One of the nice properties of a convex function is that they have only global minimum which is relatively easier to compute compared to non-convex functions, as shown in Figure 2-3, which may have local minimas and hence making the computation of global minimum an extremely difficult task.

2.4.2 Convex Optimization Problems

Using the notation of [9], a convex optimization problem is of the form

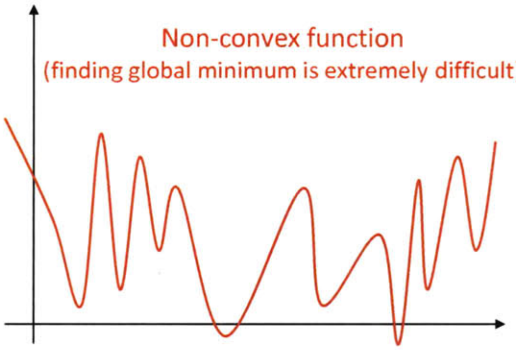


Figure 2-3: Shows a convex function. In general finding global minimum for convex functions is extremely difficult

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & a_i^T(x) = b_i, \quad i = 1, \ldots, p \end{aligned} \tag{2.11}$$

where the objective function $f_0(x)$ is convex, the inequality constraint functions $f_i(x)$ are convex and the equality constraint functions $h_i(x) = a_i^T x - b_i$ are affine. Hence in a convex optimization problem we minimize a convex objective or cost function over a convex set. Convex optimization problems are particularly attractive since finding global minimum for the cost function is a relatively easy.

2.4.3 Semidefinite Programs

Semidefinte Programs or simply SDPs belong to a special type of convex optimization problems where a linear cost function is minimized subject to linear matrix inequalities.

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & F_1 x_1 + F_2 x_2 + \cdots + F_n x_n - F_0 \succeq 0 \end{aligned} \tag{2.12}$$

where all of the matrices $F_0, F_1, \ldots F_n \in S^k$, here S^k indicates set of symmetric matrices of order $k \times k$. Semidefinite Programs are particularly important since most of the convex optimization problems can be cast as an SDP (described in Appendix A) and can be solved efficiently using public domain solvers such as [2,28].

Chapter 3

Existing Techniques

In this chapter we summarize various existing techniques for non-parameterized and parameterized model identification of multiport passive structures.

3.1 Traditional Approaches

Traditionally, the critical task of generating a model is completed manually where the circuit designer or system architect approximates the unknown system with emprirical or semi empirical formulas. These models rely on the designers' experience and intuition accumulated after a lifetime of simulations with electromagnetic field solvers and circuit simulators. In these approaches, normally the designer would either approximate the structure with lumped RC and RL networks characterized at the operating frequency as shown in Figure 3-1 or generate a simple schematic from intuition consisting of RLC elements having frequency response 'close' to the original structure as shown in Figure 3-1.

Unfortunately, these intuitive approaches in addition to being extremely limited, are also prone to generate erroneous results. Additionally, in order to generate the complete multiport transfer matrix, quite often the individual transfer functions are approximated separately. Hence, these models can be completely nonphysical, violating important physical properties of the original system such as passivity, since passivity is a property of the entire transfer matrix and cannot be enforced if transfer functions are identified individually. Also, these techniques are not scalable, hence generating a model for a system with

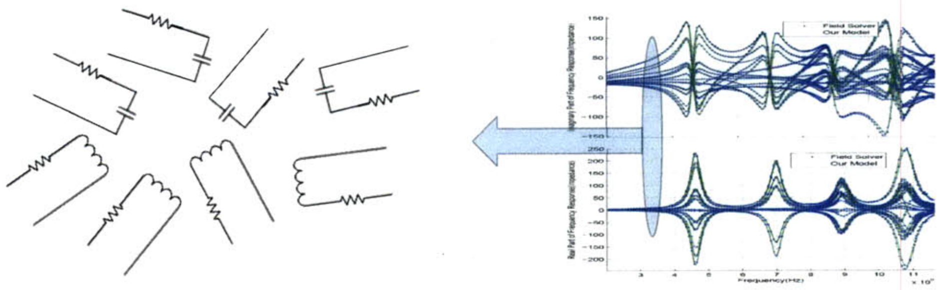


Figure 3-1: Approximation at operating frequency

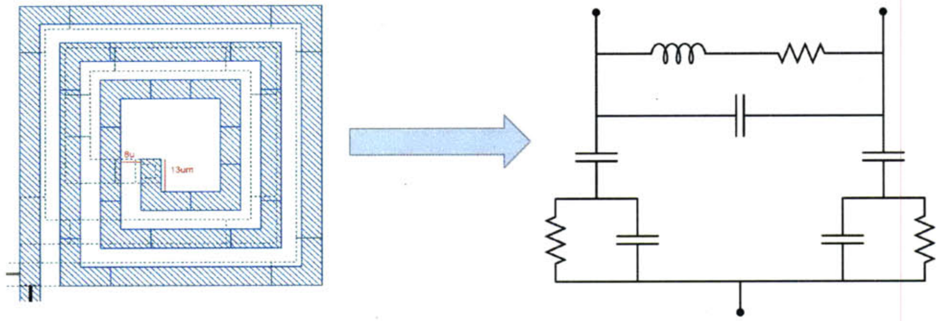


Figure 3-2: Approximation from intuition or basic physics

larger than a few ports become extremely challenging. Furthermore, the modeling task becomes even more difficult when attempting to generate by hand closed form compact models of the frequency response parameterized by design or geometrical parameters.

In order to make this process more efficient and robust, it is desirable to replace hand-generated macromodels by automatically-generated compact dynamical models that come with guarantees of accuracy, and passivity. In the following sections we discuss some of the commonly used techniques for the identification of compact dynamical models.

3.2 Automated Approaches

In the recent years, considerable effort has been put in automating the procedure to generate compact parameterized models. There are two commonly used available techniques to generate models for linear structures. The first ones are projection based approaches and the second ones are rational fitting approaches. A detailed survey of these approaches is presented in [7]. We describe these techniques one by one in the following sections.

3.3 Projection Based Approaches

3.3.1 The Traditional Projection Framework

Most of the model order reduction techniques can be interpreted within a projection framework. In such a framework the solution to a given large linear multiport system

$$E\dot{x} = Ax + Bu, \qquad y = C^T x, \tag{3.1}$$

is approximated in a low-dimensional space $x \approx V\hat{x}$, where $V \in \mathbb{R}^{N \times q}$ is the right projection matrix, $\hat{x}$ is the reduced state vector, and $N >> q$. A reduced set of equations is then obtained by forcing the residual, $r(V\hat{x}) = EV\dot{\hat{x}} - AV\hat{x} - Bu$, to be orthogonal to the subspace defined by a left projection matrix U, i.e. $U^T r(V\hat{x}) = 0$. The resulting state space model has the form

$$\hat{E}\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u, \qquad y = \hat{C}^T \hat{x}, \tag{3.2}$$

where $\hat{E} = U^T EV, \hat{A} = U^T AV, \hat{B} = U^T B$, and $\hat{C} = V^T C$. The accuracy of the reduced model created via projection is completely determined by the choice of projection matrices U and V. The most common approaches for selecting the vectors are methods based on balanced truncation, moment matching, and singular value decomposition (e.g. proper orthogonal decomposition, principle components analysis, or Karhunen-Loeve expansion). For more details on generating projection vectors see [5, 18].

3.3.2 Stable Projection for Linear Systems

Traditionally it is assumed that the original large system (3.1) possesses an extremely special structure viz. $E = E^T \succeq 0, A \preceq 0$, and $B = C$. In such cases selecting $U = V$ (known as congruence transform or Galerkin projection) will preserve stability and passivity in the reduced system for any choice of V. While all digital RLC type interconnect networks possess the required semidefinite structure, for analog modeling it is, unfortunately, completely unrealistic to restrict consideration to only semidefinite systems. Therefore for the vast majority of analog systems, the congruence transform cannot guarantee stability and passivity of the generated model. One possible computationally cheap solution is to use as a first step any of the available traditional projection based methods (including congruence transforms) and then attempt to perturb the generated model to enforce stability and passivity. One semidefinite formulation of this problem is

$$\begin{aligned} \underset{\Delta\hat{E},\Delta\hat{A},\Delta\hat{C}}{\text{minimize}} \quad & ||\Delta\hat{E}|| + ||\Delta\hat{A}|| + ||\Delta\hat{C}|| \\ \text{subject to} \quad & \hat{E} \succeq 0, \\ & \hat{A} + \hat{A}^T \preceq 0, \\ & \hat{B} = \hat{C} \end{aligned} \tag{3.3}$$

where $\hat{E} = U^T EV + \Delta\hat{E}, \hat{A} = U^T AV + \Delta\hat{A}, \hat{B} = U^T B$, and $\hat{C} = V^T C + \Delta\hat{C}$. Here stability and passivity are enforced in the reduced model by forcing it to be described by semidefinite system matrices, which introduced no loss of generality even if the original system (3.1) is not described by semidefinite matrices [6].

Unfortunately, in most cases any such perturbation could completely destroy the accuracy of the reduced model. Instead of perturbing the reduced model , a better approach that can guarantee accuracy in the reduced model is to perturb one of the projection matrices. That is, given U and V, search for a 'small' ΔU such that the system (3.2), defined by reduced matrices $\hat{E} = (U + \Delta U)^T EV, \hat{A} = (U + \Delta U)^T AV, \hat{B} = (U + \Delta U)^T B$, and $\hat{C} = V^T C + \Delta\hat{C}$ is passive. This problem can similarly be formulated as a semidefinite program

$$\begin{aligned} \underset{\Delta U}{\text{minimize}} \quad & ||\Delta U|| \\ \text{subject to} \quad & \hat{E} \succeq 0, \\ & \hat{A} + \hat{A}^T \preceq 0, \\ & \hat{B} = \hat{C} \end{aligned} \tag{3.4}$$

It can be shown that if the original model (3.1) is stable and passive, then for any projection matrix V there exist projection matrices U such that the resulting reduced model is stable and passive [6].

3.3.3 Parameterization of Projection Methods

Generating a parameterized reduced model, such as

$$\hat{E}(\lambda)\dot{\hat{x}} = \hat{A}(\lambda)\hat{x} + \hat{B}(\lambda)u \tag{3.5}$$

for a linear system where λ is a vector of design parameters, is of critical importance if the models are to be used for system level design trade-off explorations. Two modifications to the previously described projection procedures must be made when constructing parameterized models. First the subspace defined by V must capture the solution response to changes in parameter values. Expanding the subspace is typically achieved for linear systems by generating projection vectors that match the frequency response derivatives with respect to the parameters λ in addition to the frequency [11, 33]. Alternative approaches for handling variability resulting from a large number of parameters are based on sampling and statistical analysis [14, 34].

The second issue involves specifically the case of nonlinear parameter dependence, where the system matrix or vector field must be able to cheaply capture changes in λ. One way to make a parameterized system matrix $A(\lambda)$ projectable with respect to the parameters is to represent them as a sum on non-parameterized functions that are linear in scalar functions of the original parameters. For instance, for the parameterized linear system

$$\dot{x} = A(\lambda) \tag{3.6}$$

we seek to approximate and project as follows:

$$A(\lambda) \approx \sum_{i=0}^{\kappa} \tilde{A}_i g_i(\lambda) \rightarrow \hat{A}(\lambda) \approx \sum_{i=0}^{\kappa} (U^T A_i V) g_i(\lambda) \quad (3.7)$$

such that $\hat{A}_i = U^T A_i V$ are constant matrices and can be precomputed. Here $g_i(\lambda)$ are scalar functions of the original parameter set λ. The matrix approximation in (3.7) can be achieved using a polynomial expansion if $A(p)$ is known analytically, or via fitting in the case when only samples of the matrix $A_k = A(\lambda_k)$ are known [11].

3.4 Rational Fitting of Transfer Functions

Projection methods have been successful for certain classes of linear systems, but in many applications, such as when modeling analog passive components affected by full-wave effects or substrate effects, the resulting system matrices include delays, or frequency dependency. To capture such effects in a finite-order state-space model, one must approximate this frequency dependence, using for instance polynomial fitting of matrices [12], making preservation of passivity through projection even more challenging. Furthermore, often times only transfer function samples are available, obtained possibly from measurements of a fabricated device or from a commercial electromagnetic field solver. In such a scenario, since original matrices are not available, projection based approaches cannot be used.

An alternative class of methods are based on transfer matrix fitting. There exist different approaches to generate rational transfer function matrices from frequency response data. The problem of finding a passive multiport model from complex frequency response data is highly nonlinear and non convex. Given a set of frequency response samples $\{H_i, \omega_i\}$, where $H_i = H(j\omega_i)$ are the transfer matrix samples of some unknown multiport linear system, the compact modeling task is to construct a low-order rational transfer matrix $\hat{H}(s)$ such that $\hat{H}(j\omega_i) \approx H_i$. Formulated as an L_2 minimization problem of the sum of squared errors, it can be written as

$$\begin{aligned} \underset{\hat{H}}{\text{minimize}} \quad & \sum_i \left| H_i - \hat{H}(j\omega_i) \right|^2 \\ \text{subject to} \quad & \hat{H}(j\omega) \quad \textit{passive} \end{aligned} \tag{3.8}$$

Even after ignoring the passivity constraint in (3.8), the unconstrained minimization problem is non-convex and is therefore very difficult to solve. Direct solution using nonlinear least squares have been proposed, such as Levenberg-Marquardt [25]. However, there is no guarantee that such approach will converge to the global minimum, and quite often the algorithm will yield only a locally optimal result. Rather than solving non-convex minimization problem, many methods apply a relaxation to the objective function in (3.8) resulting in an optimization problem that can be solved efficiently. These schemes can be broadly classified into two categories: those which use unconstrained minimization combined with post processing perturbation to enforce passivity; and those that simultaneously enforce passivity during the fitting process by formulating a convex optimization problem.

3.4.1 Passivity During Fitting

Over the past years considerable effort has been put into finding a convex relaxation to the original problem including the passivity constraint (3.8) such as [10, 29]. Although these techniques provide an analytical formulation, they are often criticized as being still computationally quite expensive. Most of these techniques rely on enforcing the positive real lemma by constraining the real part of the impedance matrix to be positive definite over *all frequencies*. Although such a constraint can be certifiably enforced by using a Sum-Of-Squares (*SOS*) relaxation, it is normally a costly operation, specially when the constraints are defined on frequency dependent matrices such as in [29].

3.4.2 Passivity via Post-Processing

Some iterative techniques also exist, such as [16, 19]. In these techniques a stable but non-passive model is first identified. This non-passive model is then checked for passivity violations by examining if there exist pure imaginary eigen values of the corresponding Hamiltonian matrix. During the post processing step, pole locations are kept fixed and passivity is obtained by altering only the residues. This is achieved by transforming from pole-residue form to state space form and perturbing only the C matrix. A generic formulation of the positivity-enforcing minimal perturbation can be stated as

$$\min_{\Delta C} ||\Delta C||_{\mathrm{x}}, \quad \text{subject to } \mathcal{P}(\Omega) \tag{3.9}$$

where $\mathcal{P}(\Omega)$ is a positivity constraint for the transfer matrix over the set of positivity violations Ω, and $||.||_{\mathrm{x}}$ denotes the norm used for quantifying the effects of the perturbation on the accuracy of the model. In [16] it was proposed to select the norm that produces the minimal perturbation to the impulse response of the original system, defined as $||\Delta C||_{\mathrm{x}} = ||\Delta C K^T||_2$ where $K^T K = W$ is the controllability Grammian satisfying $EWA^T + AWE^T + BB^T = 0$. Other possible choices for the objective function in (3.9) are presented in [19]. The positivity constraint $\mathcal{P}(\Omega)$ is enforced using first-order perturbation to the eigen values of the Hamiltonian matrix, and can be expressed as a linear matrix equality or matrix inequality in terms of the unknown perturbation vector ΔC.

These techniques are computationally efficient, however since perturbing the system is an ill-posed problem, there is no guarantee that the final passivated model is optimal for accuracy, specially in the case where the initial non-passive model had significant passivity violations.

3.4.3 Passivity via Passive-Subsections

A model is passive if all of its building blocks are passive. There are approaches, such as [24, 26], where the individual building blocks of a non-passive model are checked for passivity individually. However, since such a condition is only a sufficient condition, many passive models will fail the test. Also in these approaches [24, 26] no efficient method or algorithm was presented in order to rectify for passivity violations. For example in [26] it was proposed that the pole-residue pairs violating passivity conditions should be discarded, this is highly restrictive and can significantly deteriorate the accuracy. We instead propose that the identified residue matrices should conform to passivity conditions during the identification such that there are no passivity violation in the final model.

3.4.4 Parameterized Rational Fitting

There are two possible approaches to generating a parameterized transfer matrix $\hat{H}(s,\lambda)$ from a given set of frequency response and parameter values $\{H_i,\omega_i,\lambda_i\}$. The first approach is to fit simultaneously to the frequency response data and parameter values, i.e. minimizing $\sum|\hat{H}(j\omega_i,\lambda_i)-H_i|^2$. This approach was first proposed in [29] along with the simultaneously enforcement of stability passivity. However, simultaneous frequency and parameter fitting can become quite expensive for a large number of parameters. Alternative fitting approaches rely on interpolating between a collection of non-parameterized models, each generated by a stable and passive fitting procedure. The main challenge for such approaches based on interpolation is to guarantee passivity in the final interpolated model, since one may produce very trivial stable systems where simple interpolation will not preserve stability or passivity. All the current existing interpolation based algorithms [13, 15, 32] only provide a *test* to simply check stability after a particular instance of the parameterized model has been instantiated. The downside of such methods is that the user (i.e. a circuit designer, or a system level optimization routine) would need to run such test *every single time* a new model is instantiated. Furthermore, if the instantiated model does not pass the stability test, the user would either be stuck with an unstable model, or would need to basically rerun the fitting algorithm to perturb the unstable instantiation until stability is achieved. In other words none of the available interpolation based approaches can *guarantee* that *any* instantiation of their identified parameterized models will be *a priori* stable and passive for *any* value of the parameters in a predefined range.

In this thesis we present a method for generating parameterized models of linear systems that the user will be able to instantiate for any parameter value either within a limited given range, or for an unlimited range, and be sure *a priori* to obtain a passive model. Given a collection of systems swept over design and geometrical parameters of interest, we identify a closed form parameterized dynamical model using constrained fitting. The details can be found in Chapter 7. Our algorithm is completely independent from the type of initial non-parameterized identification procedure used for the individual systems, if *only* stability is sought in the final parameterized model. In other words, the individual (non-parameterized) models may be generated by any stability preserving modeling scheme such as convex optimization based approaches [10,22,29], vector fitting based approaches [16, 17, 19, 20] or Loewner matrix based approaches [21]. However, in order to enforce global passivity, the individual non-parameterized models need to have the structure described in Chapter 4.

Chapter 4

Passive Fitting for Multiport Systems - Method I

4.1 Rational Transfer Matrix Fitting in Pole Residue Form

The problem of constructing a rational approximation of multi port systems in pole residue form consists of finding residue matrices $\mathbf{R_k}$, poles a_k and the matrices $\mathbf{D}$ & $\mathbf{F}$ such that the identified model, defined by the transfer function $\hat{H}(s)$ in (4.1), minimizes the mismatch with the frequency response samples from the original system as described in (3.8).

$$\hat{H}(s) = \sum_{i=1}^{\kappa} \frac{\mathbf{R_k}}{s - a_k} + \mathbf{D} + s\mathbf{F} \tag{4.1}$$

here $\mathbf{R_k}, \mathbf{D}$ and $\mathbf{F}$ are $T \times T$ residue matrices (assuming the system has T ports) and a_k are poles. Since most of the passive structures have a symmetric response, we consider the case when $\mathbf{R_k}, \mathbf{D}$ and $\mathbf{F}$ are symmetric matrices. In the case when the matrices are non-symmetric, we can apply the same formulation to the symmetric part of the matrices.

4.2 Passive Fitting for Multiport LTI Systems

4.2.1 Problem Formulation

To formulate the problem, we expand the summation for $\hat{H}(s)$ in (4.1) in terms of the purely real and complex poles. Also, since we are mainly interested in the properties of $H(s)$ on the imaginary axis, we replace s with $j\omega$.

$$\hat{H}(j\omega) = \sum_{k=1}^{\kappa_r} \frac{\mathbf{R_k^r}}{j\omega - a_k^r} + \sum_{k=1}^{\kappa_c} \frac{\mathbf{R_k^c}}{j\omega - a_k^c} + \mathbf{D} + j\omega\mathbf{F} \tag{4.2}$$

Where κ_r and κ_c denote the number of purely real and the number of complex poles, respectively. Also, $\mathbf{R_k^r} \in \mathbb{R}^{T\times T}$, $\mathbf{R_k^c} \in \mathbb{C}^{T\times T}$, $a_k^r \in \mathbb{R}$, $a_k^c \in \mathbb{C}\ \forall k$, and $\mathbf{D},\mathbf{F} \in \mathbb{R}^{T\times T}$, where T is the number of ports.

In the following sections, we consider one by one the implications of each passivity condition in (2.5) on the structure of (4.2).

4.2.2 Conjugate Symmetry

Let us consider the implications of first condition of passivity on the structure of our proposed model in (4.2). The terms in (4.2) corresponding to the matrices **D** and **F**, and to the summation over purely real poles satisfy automatically the property of conjugate symmetry in (2.5a). On the other hand such condition *requires* that the complex-poles a_k^c and complex residue matrices $\mathbf{R_k^c}$ always come in *complex-conjugate-pairs*

$$\hat{H}(j\omega) = \sum_{k=1}^{\kappa_r} \frac{\mathbf{R_k^r}}{j\omega - a_k^r} + \sum_{k=1}^{\kappa_c/2} \left\{ \frac{\Re\mathbf{R_k^c} + j\Im\mathbf{R_k^c}}{j\omega - \Re a_k^c - j\Im a_k^c} + \frac{\Re\mathbf{R_k^c} - j\Im\mathbf{R_k^c}}{j\omega - \Re a_k^c + j\Im a_k^c} \right\} + \mathbf{D} + j\omega\mathbf{F} \tag{4.3}$$

In (4.3) $\Re$ and $\Im$ indicate the real and imaginary parts respectively. Note that the summation for complex poles now extends only upto $\kappa_c/2$.

Proof The condition requires $\overline{\hat{H}(\overline{j\omega})} = \hat{H}(j\omega)$. We show that the $\hat{H}(j\omega)$, as in (4.3) satisfies this constraint.

$$\begin{aligned}
\hat{H}(\overline{j\omega}) &= \sum_{k=1}^{\kappa_r} \frac{\mathbf{R_k^r}}{-j\omega - a_k^r} + \sum_{k=1}^{\kappa_c/2} \left\{ \frac{\Re \mathbf{R_k^c} + j\Im \mathbf{R_k^c}}{-j\omega - \Re a_k^c - j\Im a_k^c} + \frac{\Re \mathbf{R_k^c} - j\Im \mathbf{R_k^c}}{-j\omega - \Re a_k^c + j\Im a_k^c} \right\} + \mathbf{D} - j\omega\mathbf{F} \\
\Rightarrow \overline{\hat{H}(\overline{j\omega})} &= \sum_{k=1}^{\kappa_r} \frac{\mathbf{R_k^r}}{j\omega - a_k^r} + \sum_{k=1}^{\kappa_c/2} \left\{ \frac{\Re \mathbf{R_k^c} - j\Im \mathbf{R_k^c}}{j\omega - \Re a_k^c + j\Im a_k^c} + \frac{\Re \mathbf{R_k^c} + j\Im \mathbf{R_k^c}}{j\omega - \Re a_k^c - j\Im a_k^c} \right\} + \mathbf{D} + j\omega\mathbf{F} \\
&= \hat{H}(j\omega)
\end{aligned}$$

∎

Rewriting (4.3) compactly we get:

$$\hat{H}(j\omega) = \sum_{k=1}^{\kappa_r} \hat{H}_k^r(j\omega) + \sum_{k=1}^{\kappa_c/2} \hat{H}_k^c(j\omega) + \mathbf{D} + j\omega\mathbf{F} \tag{4.4}$$

$$\text{where: } \hat{H}_k^r(j\omega) = \frac{\mathbf{R_k^r}}{j\omega - a_k^r} \tag{4.5}$$

$$\hat{H}_k^c(j\omega) = \frac{\Re \mathbf{R_k^c} + j\Im \mathbf{R_k^c}}{j\omega - \Re a_k^c - j\Im a_k^c} + \frac{\Re \mathbf{R_k^c} - j\Im \mathbf{R_k^c}}{j\omega - \Re a_k^c + j\Im a_k^c} \tag{4.6}$$

4.2.3 Stability

The second condition (2.5b), which requires analyticity of $\hat{H}(s)$ in $\Re\{s\} > 0$, implies stability. For a linear causal system in pole-residue form (4.1), the system is strictly stable if all of its poles a_k are in the left half of complex plane i.e. they have negative real part ($\Re\{a_k\} < 0$). Note that the system is marginally stable if conjugate pair poles with multiplicity one are present on the imaginary axis.

4.2.4 Positivity

The positivity condition for passivity (2.5c) is the most difficult condition to enforce analytically. We present here an extremely efficient condition which implies (2.5c). We consider the case when all the building blocks in the summation (4.4), namely: purely real poles/residues $\hat{H}_k^r(j\omega)$, complex-conjugate pairs of poles/residues $\hat{H}_k^c(j\omega)$, and the direct term matrix D are individually positive real. Please note that the $j\omega\mathbf{F}$ term has purely imaginary response and therefore does not affect positivity condition.

Lemma 4.2.1 (Positive Real Summation Lemma) *Let $\hat{H}(j\omega)$ be a stable and conjugate symmetric transfer matrix given by* (4.4)*, then $\hat{H}(j\omega)$ is positive-real if $\hat{H}_k^r(j\omega)$, $\hat{H}_k^c(j\omega)$ and D are positive-real $\forall k$. i.e.*

$$\Re\hat{H}_k^r(j\omega) \succeq 0, \Re\hat{H}_k^c(j\omega) \succeq 0 \forall k \;\&\; D \succeq 0 \implies \hat{H}(j\omega) \succeq 0 \qquad (4.7)$$

Proof The sum of positive-real, complex matrices is positive real. ∎

Lemma 4.2.1 describes a *sufficient*, but *not-necessary*, condition for (2.5c). However, as it will be shown in the examples, this condition is not restrictive.

In the following sections we derive the equivalent conditions of positive realness on each term separately.

Purely Real Pole-Residues

In this section we derive the condition for the purely real pole/residue term $\hat{H}_k^r(j\omega)$ in the summation (4.4) to be positive real. Such a condition can be obtained by rationalizing $\hat{H}_k^r(j\omega)$ defined in (4.5), as following:

$$\begin{aligned}
\hat{H}_k^r(j\omega) &= \frac{\mathbf{R_k^r}}{j\omega - a_k^r} && (4.8)\\
&= \frac{\mathbf{R_k^r}}{j\omega - a_k^r} \times \frac{-j\omega - a_k^r}{-j\omega - a_k^r} &&\\
&= -\frac{a_k^r\mathbf{R_k^r}}{\omega^2 + a_k^{r2}} - j\frac{\omega\mathbf{R_k^r}}{\omega^2 + a_k^{r2}} && (4.9)\\
\Re\hat{H}_k^r(j\omega) \succeq 0 &\implies -\frac{a_k^r\mathbf{R_k^r}}{\omega^2 + a_k^{r2}} \succeq 0 \; \forall \omega, k = 1, \ldots, \kappa_r && (4.10)
\end{aligned}$$

Complex Conjugate Pole-Residues

In this section we derive the positive realness condition for the complex pole/residue term $\hat{H}_k^c(j\omega)$ in the summation (4.4). Since complex terms always appear conjugate pairs, we first add the two terms for $\hat{H}_k^c(j\omega)$ in (4.6) resulting into:

$$\begin{aligned}\hat{H}_k^c(j\omega) &= \frac{\Re\mathbf{R_k^c} + j\Im\mathbf{R_k^c}}{j\omega - \Re a_k^c - j\Im a_k^c} + \frac{\Re\mathbf{R_k^c} - j\Im\mathbf{R_k^c}}{j\omega - \Re a_k^c + j\Im a_k^c} && (4.11)\\ &= \frac{-2(\Re a_k^c)(\Re\mathbf{R_k^c}) - 2(\Im a_k^c)(\Im\mathbf{R_k^c}) + j2\omega(\Re\mathbf{R_k^c})}{(\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2 - j2\omega\Re a_k^c} && (4.12)\end{aligned}$$

In order to obtain positive realness condition on $\hat{H}_k^c(j\omega)$ we rationalize (4.12) to form (4.13). The resulting condition for $\Re\hat{H}_k^c(j\omega) \succeq 0$ is given in (4.14)

$$\begin{aligned}\hat{H}_k^c(j\omega) &= \frac{-2(\Re a_k^c)(\Re\mathbf{R_k^c}) - 2(\Im a_k^c)(\Im\mathbf{R_k^c}) + j2\omega(\Re\mathbf{R_k^c})}{(\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2 - j2\omega\Re a_k^c} \times \frac{(\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2 + j2\omega\Re a_k^c}{(\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2 + j2\omega\Re a_k^c}\\ &= \frac{-2\{(\Re a_k^c)^2 + (\Im a_k^c)^2\}\{(\Re a_k^c)(\Re\mathbf{R_k^c}) + (\Im a_k^c)(\Im\mathbf{R_k^c})\} - 2\omega^2\{(\Re a_k^c)(\Re\mathbf{R_k^c}) - (\Im a_k^c)(\Im\mathbf{R_k^c})\}}{((\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2)^2 + (2\omega\Re a_k^c)^2}\\ &\quad + j\frac{-2\omega\{(\Re a_k^c)^2 - (\Im a_k^c)^2 + \omega^2\}\Re\mathbf{R_k^c} - 4\omega(\Re a_k^c)(\Im a_k^c)\Im\mathbf{R_k^c}}{((\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2)^2 + (2\omega\Re a_k^c)^2} && (4.13)\end{aligned}$$

$$\begin{aligned}&\Re\hat{H}_k^c(j\omega) \succeq 0 \implies\\ &\quad \frac{-2\{(\Re a_k^c)^2 + (\Im a_k^c)^2\}\{(\Re a_k^c)(\Re\mathbf{R_k^c}) + (\Im a_k^c)(\Im\mathbf{R_k^c})\} - 2\omega^2\{(\Re a_k^c)(\Re\mathbf{R_k^c}) - (\Im a_k^c)(\Im\mathbf{R_k^c})\}}{((\Re a_k^c)^2 + (\Im a_k^c)^2 - \omega^2)^2 + (2\omega\Re a_k^c)^2} \succeq 0\\ &\qquad\qquad \forall\omega, k = 1, \ldots, \kappa_c && (4.14)\end{aligned}$$

Direct Term Matrix

Since $\mathbf{D}$ is a constant real symmetric matrix, we require $\mathbf{D}$ to be a positive semidefinite matrix, i.e.

$$\mathbf{D} \succeq 0$$

4.2.5 The Constrained Minimization Problem

We combine all the constraints derived earlier and formulate a constrained minimization problem as follows:

$$\begin{array}{ll}
\underset{\hat{H}\equiv\{\mathbf{R_k},a_k,\mathbf{D},\mathbf{F}\}}{\text{minimize}} & \sum_i \left| H_i - \hat{H}(j\omega_i) \right|^2 \\
\text{subject to} & a_k^r < 0 \; \forall k = 1,\ldots,\kappa_r \\
 & \Re a_k^c < 0 \; \forall k = 1,\ldots,\kappa_c \\
 & \Re \hat{H}_k^r(j\omega) \succeq 0 \; \forall \omega, k = 1,\ldots,\kappa_r \\
 & \Re \hat{H}_k^c(j\omega) \succeq 0 \; \forall \omega, k = 1,\ldots,\kappa_c \\
 & \mathbf{D} \succeq 0 \\
\text{where} & \hat{H}(j\omega) = \sum_{k=1}^{\kappa_r} \hat{H}_k^r(j\omega) + \sum_{k=1}^{\kappa_c/2} \hat{H}_k^c(j\omega) + \mathbf{D} + j\omega\mathbf{F}
\end{array} \qquad (4.15)$$

Here H_i are the given frequency response samples at frequencies ω_i; $\hat{H}_k^r$ and $\hat{H}_k^c$ are defined in (4.5) and (4.6) respectively; a_k^r and a_k^c denotes the real and complex poles respectively. The detailed expressions for $\Re \hat{H}_k^r(j\omega) \succeq 0$ and $\Re \hat{H}_k^c(j\omega) \succeq 0$ are described in (4.10) and (4.14) respectively. The optimization problem described above in (4.15) is non convex. In the following section, we shall see how we can implement the relaxed version of (4.15) as a convex problem in terms of linear matrix inequalities.

4.3 Implementation

In this section we describe in detail the implementation of our passive multiport model identification procedure based on solving the constrained minimization framework developed in Section 4.2.

The optimization problem in (4.15) is non-convex because both the objective function and the constraints are non-convex. The non-convexity in (4.15) arises mainly because of the terms containing *products* and *ratios* between decision variables such as ratio of residue matrices, R_k, and poles, a_k, in the objective function, and product terms and ratios of R_k

and a_k in the constraints.

Since the main cause of non-convexity in (4.15) is the coupling between R_k and a_k, it is natural to uncouple the identification of unknowns, namely R_k and a_k in order to convexify (4.15). We propose to solve the optimization problem in (4.15) in two steps. The first step consists of finding a set of stable poles a_k for the system. The second step consists of finding a passive multiport dynamical model for the system, given stable poles from step 1. In the following sections we describe how to solve the two steps.

4.3.1 Step 1: Identification of stable poles

Several efficient algorithms already exist for the identification of stable poles for multiport systems. Some of the stable pole identification approaches use optimization based techniques such as in [29]. Some schemes such as [4, 20] find the location of stable poles iteratively. Any one of these algorithms can be used as the first step of our algorithm, where we identify a common set of stable poles for all the transfer functions in the transfer matrix. As mentioned before, to enforce conjugate symmetry, the stable poles can either be real or be in the form of complex-conjugate pairs. We employ a binary search based algorithm to automatically find the minimum number of poles required to achieve a user defined error bound on the mismatch between given frequency response samples and the frequency response of identified stable model.

4.3.2 Step 2: Identification of Residue Matrices

In this section we formulate the convex optimization problem for the identification of residue matrices using the stable poles from step 1. We first revisit the conditions for passivity (4.10) and (4.14), and later we shall develop the convex objective function.

Purely Real Pole-Residues

Let us consider the positive realness condition on the purely real pole residue term $H_k^r(j\omega)$ as in (4.10). The constraint (4.10) requires frequency dependent matrices to be positive semidefinite for all frequencies. This is in general very expensive to enforce. However, a careful observation of (4.10) reveals that the denominator, which is the only frequency dependent part of (4.10) is a positive real number for all frequency. This allows us to ignore the positive denominator which leaves us enforcing $-a_k^r\mathbf{R_k^r} \succeq 0$. Since we are already given stable poles (i.e. $a_k^r < 0$), the constraint in (4.10) reduces to enforcing positive semidefiniteness on $\mathbf{R_k^r}$.

$$\Re \hat{H}_k^r(j\omega) \succeq 0 \implies \mathbf{R_k^r} \succeq 0\, \forall k = 1, ..., \kappa_r \tag{4.16}$$

Such a constraint is convex and can be enforced extremely efficiently using SDP solvers [2, 28].

Complex Conjugate Pole-Residues

In this section we reconsider the positive realness condition on the complex conjugate pole residue pair term $H_k^c(j\omega)$ as in (4.14). As before, a closer examination of the frequency dependent denominator in (4.14) reveals the fact that it is positive for all frequencies. Given that we have a fixed set of stable poles, and the denominator is always positive, we rewrite the constraint (4.14) only in terms of the variables i.e. ω and $\mathbf{R_k^c}$. Also, we replace the constant expressions of $\Re a_k^c$ and $\Im a_k^c$ in (4.14) with generic constants c_i. We finally obtain the following equivalent condition

$$\begin{aligned}&\Re \hat{H}_k^c(j\omega) \succeq 0 \implies \\ &(c_1 \Re \mathbf{R_k^c} + c_2 \Im \mathbf{R_k^c}) + \omega^2 (c_3 \Re \mathbf{R_k^c} + c_4 \Im \mathbf{R_k^c}) \} \succeq 0\, \forall \omega, k = 1, \ldots, \kappa_c\end{aligned} \tag{4.17}$$

The problem is however still not solved since the condition in (4.17) is frequency dependent.

Lemma 4.3.1 *Let $\mathcal{X}_1, \mathcal{X}_2 \in \mathbb{S}^T$ and $\omega \in [0, \infty)$, where $\mathbb{S}^T$ is the set of symmetric $T \times T$*

matrices, then

$$\mathcal{X}_1 + \omega^2 \mathcal{X}_2 \succeq 0 \forall \omega \Leftrightarrow \mathcal{X}_1 \succeq 0, \mathcal{X}_2 \succeq 0 \tag{4.18}$$

Proof Direction $\Rightarrow$

Given $\mathcal{X}_1 + \omega^2 \mathcal{X}_2 \succeq 0$ we consider the following limits:

$$\begin{aligned} \lim_{\omega \longrightarrow 0} (\mathcal{X}_1 + \omega^2 \mathcal{X}_2) \succeq 0 &\implies \mathcal{X}_1 \succeq 0 \\ \lim_{\omega \longrightarrow \infty} (\mathcal{X}_1 + \omega^2 \mathcal{X}_2) \succeq 0 &\implies \mathcal{X}_2 \succeq 0 \end{aligned} \tag{4.19}$$

Direction $\Leftarrow$ follows from the fact that a non-negative weighted sum of positive semidefinite matrices is positive semidefinite. ∎

We define

$$\begin{aligned} \mathcal{X}_k^1 &= c_1 \Re \mathbf{R_k^c} + c_2 \Im \mathbf{R_k^c} \\ \mathcal{X}_k^2 &= c_3 \Re \mathbf{R_k^c} + c_4 \Im \mathbf{R_k^c}, \end{aligned} \tag{4.20}$$

and apply Lemma 4.3.1 to the constraint defined in (4.17) which results into

$$\Re \hat{H}_k^c(j\omega) \succeq 0 \implies \mathcal{X}_k^1 \succeq 0, \mathcal{X}_k^2 \succeq 0 \; \forall k = 1, ..., \kappa_c \tag{4.21}$$

Since $\mathcal{X}_k^1, \mathcal{X}_k^2$ are linear combinations of the unknown matrices, $\Im \mathbf{R_k^c}$ & $\Im \mathbf{R_k^c}$ the constraint (4.21) is a semidefinite convex constraint and thus can be enforced very efficiently.

Convex Optimization to Find Residue Matrices

In this section we summarize the final convex optimization identifying the residue matrices which correspond to to passive $H(j\omega)$, given stable poles a_k.

$$
\begin{aligned}
\underset{\mathbf{R}^{\mathbf{r}}_{\mathbf{k}},\mathbf{R}^{\mathbf{c}}_{\mathbf{k}},\mathbf{D},\mathbf{F}}{\text{minimize}} \quad & \sum_i \left|\Re H_i - \Re \hat{H}(j\omega_i)\right|^2 + \sum_i \left|\Im H_i - \Im \hat{H}(j\omega_i)\right|^2 \\
\text{subject to} \quad & \mathbf{R}^{\mathbf{r}}_{\mathbf{k}} \succeq 0 \; \forall k = 1,\ldots,\kappa_r \\
& -\Re a^c_k \Re \mathbf{R}^{\mathbf{c}}_{\mathbf{k}} + \Im a^c_k \Im \mathbf{R}^{\mathbf{c}}_{\mathbf{k}} \succeq 0 \; \forall k = 1,\ldots,\kappa_c \\
& -\Re a^c_k \Re \mathbf{R}^{\mathbf{c}}_{\mathbf{k}} - \Im a^c_k \Im \mathbf{R}^{\mathbf{c}}_{\mathbf{k}} \succeq 0 \; \forall k = 1,\ldots,\kappa_c \\
& \mathbf{D} \succeq 0 \\
\text{where} \quad & \hat{H}(j\omega) = \sum_{k=1}^{\kappa_r} \hat{H}^r_k(j\omega) + \sum_{k=1}^{\kappa_c/2} \hat{H}^c_k(j\omega) + \mathbf{D} + j\omega\mathbf{F}
\end{aligned}
\tag{4.22}
$$

This final problem (4.22) is convex, since the objective function is a summation of L_2 norms. All the constraints in (4.22) are linear matrix inequalities. This convex optimization problem is a special case of semidefinite programming, requiring only few frequency independent matrices to be positive semidefinite. This problem formulation is extremely fast to solve, compared to other convex formulations [22, 29] where the unknown matrices are frequency dependent.

Complexity

In the problem formulation (4.22), all the matrices are symmetric, allowing us to search only for the upper triangular part. Also since complex residues are enforced by the construction to appear in conjugate pairs, we account for only half of the terms in the complex conjugate pair.

4.3.3 Equivalent Circuit Synthesis

From the circuits perspective, the algorithm identifies a collection of low-pass, band-pass, high-pass and all-pass passive filter networks. These passive blocks can be readily synthesized into an equivalent passive circuit networks, and can be interfaced with commercial circuit simulators by either generating a spice-like netlist, or by using Verilog-A. Alternatively, we can develop equivalent state space realizations for our passive multiport models, for example a Jordan-canonical form can be obtained as described in [4] and then diagonalized.

4.3.4 The Complete Algorithm

In this section we present the description of the complete framework in Algorithm 1.

Algorithm 1 Complete Passive Multiport Model Identification

```
Input: The set of frequency response samples ${H_i, \omega_i}$, either the number of poles $N$ or
    the rms error bound $\varepsilon$
Output: Passive model $\hat{H}(j\omega)$
Find stable poles $a_k$ for the system
if $\nexists N$ then
  $N_L \Leftarrow 1, N_U \Leftarrow N_{MAX}$
  repeat
    $t \Leftarrow (N_L + N_U)/2$
    Find the stable system $\hat{H}_t$ with $t$ poles $a_k$
    Compute the rms error $\varepsilon_t = \sqrt{\sum_i |H_i - \hat{H}_t(j\omega_i)|^2}$
    if $\varepsilon_t < \varepsilon$ then
      $N_U \Leftarrow t$
    else
      $N_L \Leftarrow t$
    end if
  until $N_U = N_L$
  $N \Leftarrow N_U$
end if
Find the stable system with $N$ poles $a_k$
Solve the optimization problem (4.22) for $\mathbf{R}_k$
Construct the model in pole/residue form as in (4.2)
Synthesize the equivalent passive circuit and generate the corresponding netlist or ver-
    ilogA model file
```

This algorithm minimizes a cost function based on L_2 norm subject to linear matrix inequalities. Such a formulation can be solved very efficiently and is guaranteed to converge to the global minimum. However, the fact that this algorithm provides analytical expressions to enforce passivity in a highly efficient manner has an enormous potential such as in extensions to parameterized passive multiport models (discussed in Chapter 7); or to include designers specific constraints such as ensuring a good match for qualify factors in RF inductor dynamical models.

4.4 Results

In this section we shall present modeling examples of various multiport passive structures. All examples are implemented in Matlab and run on a laptop with Intel Core2Duo processor with 2.1GHz clock, 3GB of main memory, and running windows 7. We have also posted on public domain free open source software implementing this algorithm [1].

4.4.1 Wilkinson Combiner in a LINC Amplifier

In this section we shall present an example illustrating the usefulness of our proposed methodology for modeling and simulating a LINC (*LI*near amplification with *N*onlinear *C*omponents) power amplifier [30]. The architecture, as described in Figure 4-1, consists of a signal splitter, two power amplifiers, and a Wilkinson type power combiner. This architecture is designed to operate at $40GHz$. $PA1$ and $PA2$ are class B amplifiers designed in $130nm$ SiGe process using BJTs. The Wilkinson combiner is designed on alumina substrate with characteristic impedance of 50Ω and operating frequency of $40GHz$.

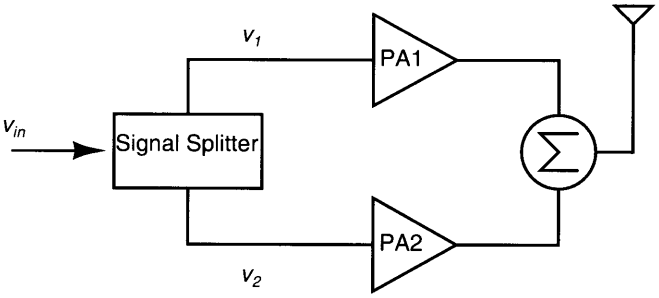


Figure 4-1: Block diagram of the LINC power amplifier architecture

Input, v_{in}, to this architecture is a $64-QAM$ signal. The signal splitter decomposes the input QAM signal into two phase modulated fixed amplitude signals. Let $v_{in} = V_{in}\angle\phi$ be the input signal; $v_1 = V_0\angle\phi_1$ and $v_2 = V_0\angle\phi_2$ be the two signals generated by the splitter then,

$$v_{in} = v_1 + v_2, \qquad V_{in}\angle\phi = V_0\angle\phi_1 + V_0\angle\phi_2 \tag{4.23}$$

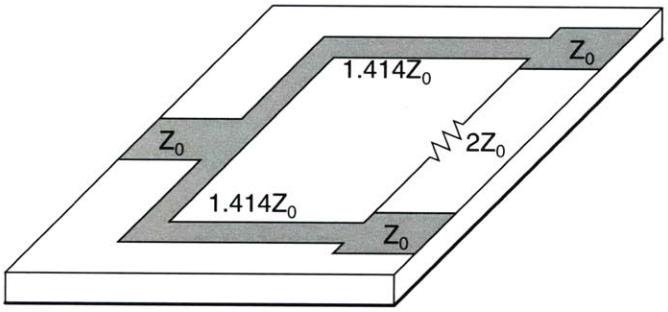


Figure 4-2: Layout of the wilkinson combiner

The splitted signals are amplified by individual nonlinear power amplifiers. The outputs of these two power amplifiers are added using a Wilkinson type power combiner. This 3-port Wilkinson combiner, as shown in Figure 4-2, is simulated inside a full wave public domain field solver [27] available at [3]. Using the frequency response samples generated by the field solver, a closed form state space model of order $m = 30$ is identified using our passive modeling algorithm. To demonstrate the accuracy of this model in frequency domain Figure 4-3 compares the impedance parameters from field solver (dots) and frequency response of our identified passive model (solid lines). The modeling error $e_{i,k}(\omega)$, defined by (4.24), was less than 0.7% for all i,k in the bandwidth of interest between $2GHz - 60GHz$

$$e_{i,k}(\omega) = \frac{|H_{i,k}(j\omega) - \hat{H}_{i,k}(j\omega)|}{\max\limits_{i,k,\omega} |H_{i,k}(j\omega)|} \tag{4.24}$$

The algorithm took only $2 seconds$ to generate the entire model, whereas for the same order and simular accuracy the algorithm described in [29] took $83 seconds$ giving us a speed-up of 40×.

A model is passive if there are no purely imaginary eigen values of the associated Hamiltonian matrix. Figure 4-4 is a zoomed-in plot of the eigen values of the associated hamiltonian matrix for the identified model. It is clear that the model passes the passivity

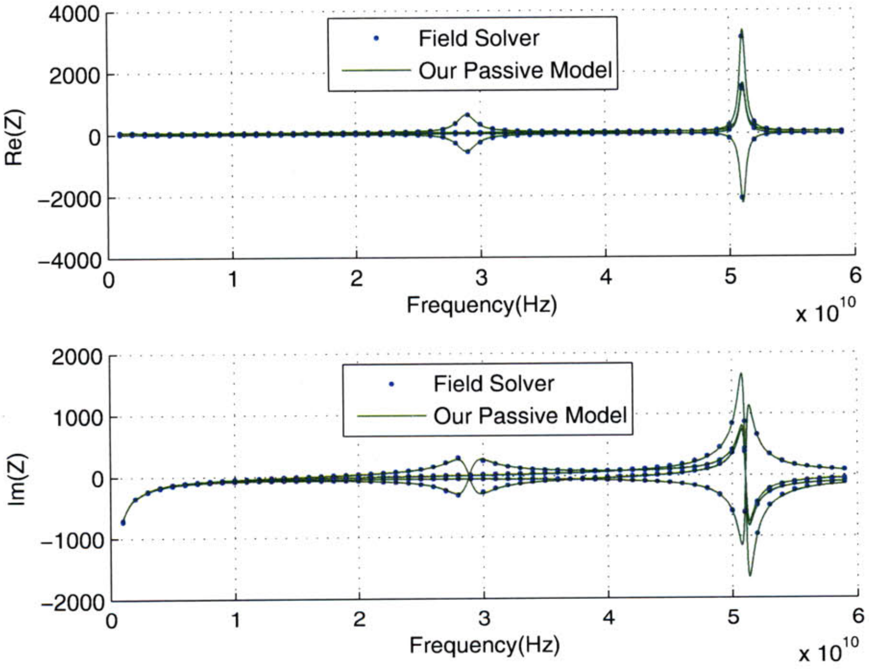


Figure 4-3: Comparing real and imaginary part of the impedance parameters from field solver (dots) and our passive model (solid lines). The mismatch, defined by (4.24), is $e_{i,k}(\omega) < 0.7\% \ \forall i,k,\omega \in [2,60]GHz$

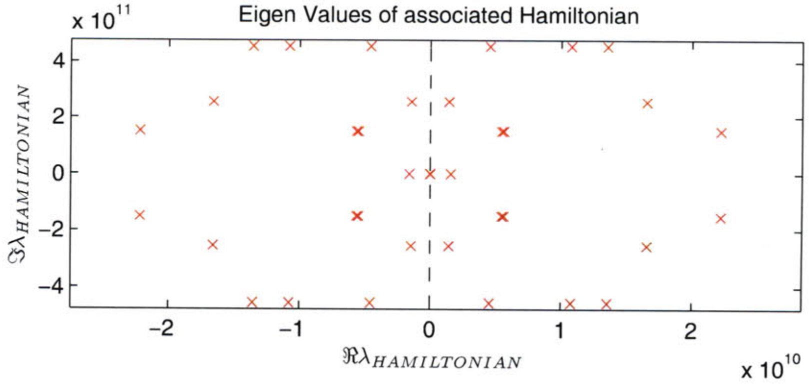


Figure 4-4: Plotting the zoomed-in eigen values of the associated hamiltonian matrix for the identified model of Wilkinson combiner

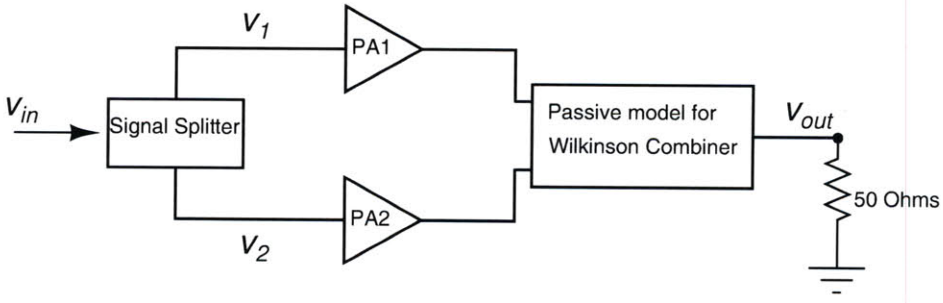


Figure 4-5: Block diagram of the LINC power amplifier architecture as simulated inside the circuit

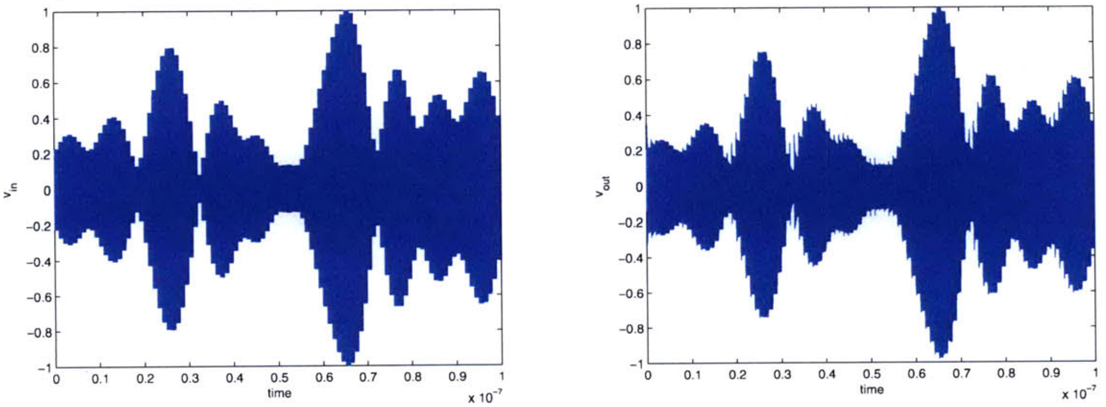
test since there are no purely imaginary eigen values.

Finally, the overall amplifier architecture is simulated inside a commercial circuit simulator after connecting the linear model for the combiner with the rest of the circuit components including the nonlinear amplifiers, as shown in Figure 4-5.

Figures 4-6(a) and 4-6(b) plots the *normalized* input (v_{in}) and output (v_{out}) voltages respectively. Practically speaking, as verified in Figures 4-6(a) and 4-6(b), the passive nature of the identified model for the Wilkinson combiner guarantees that transient simulations for the overall architecture converge, and the final output signal v_{out} is also a $64-QAM$ signal similar to the input v_{in}.

4.4.2 Power & Ground Distribution Grid

The second example we present is a power & ground distribution grid used in systems on chip or on package. The 3D layout for this power grid is shown in Figure 4-7, and is com-



(a) Ideal normalized 64-QAM input voltage v_{in}

(b) Normalized output voltage v_{out} generated by transient simulation of the overall architecture in Figure 4-5

Figure 4-6: Normalized input and output 64-QAM signals

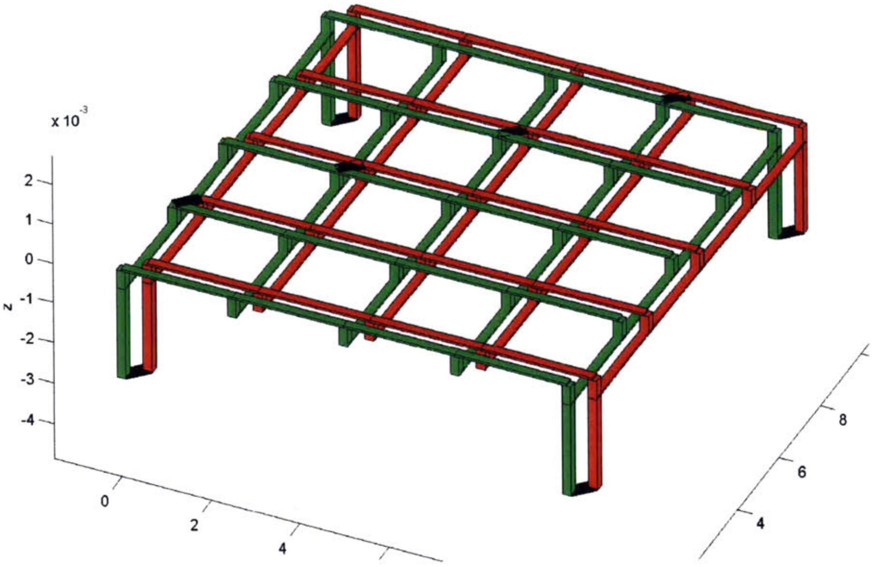


Figure 4-7: 3D layout of the distribution grid (not to scale) showing Vdd (red or dark grey) and Gnd (green or light grey) lines. Black strips represent location of ports

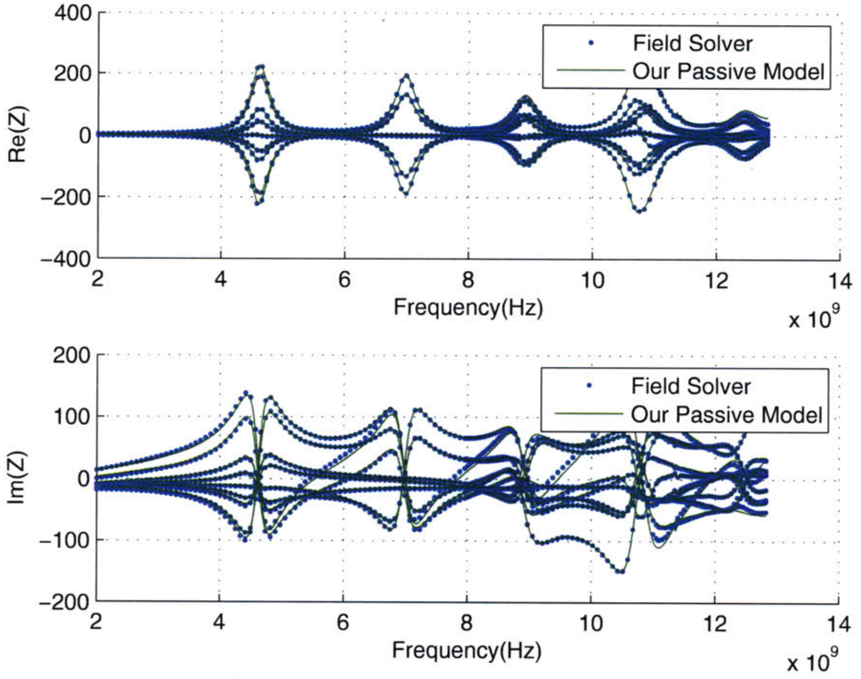


Figure 4-8: Comparing real and imaginary parts of the impedance from our passive model (solid line) and from the field solver (dots) for a power distribution grid

posed of five Vdd (red or dark grey) and Gnd (green or light grey) segments placed along both x and y axes. External connections, given by solder balls in a flip chip technology, are modeled with bond wires running vertically. Important parameters of this power grid are as follows: die size= $10mm \times 10mm$, wire width= $20\mu m$, wire height= $5\mu m$, vertical separation= $4\mu m$, gnd-vdd separation= $20\mu m$, bond-wire lengths= $500\mu m$ and solder ball radius= $20\mu m$. This structure was simulated using 52390 unknowns in the full wave mixed potential integral equation (MPIE) solver, FastMaxwell [27], to obtain frequency response samples up to 12 GHz. The multiport simulation was arranged by placing eight ports: four at the grid corners and four inside the grid. Ports are illustrated in Figure 4-7 as black strips.

For this example our proposed algorithm identified an 8×8 passive transfer matrix of order $m = 160$ in $74seconds$, whereas the algorithm in [29] ran out of memory and did not generate the model. To demonstrate the accuracy, Figure 4-8 compares the real and imaginary impedance respectively of our reduced model with the field solver data.

Although the models are passive by construction, the passive nature was verified by the absence of purely imaginary eigen values of the associated hamiltonian matrix. A

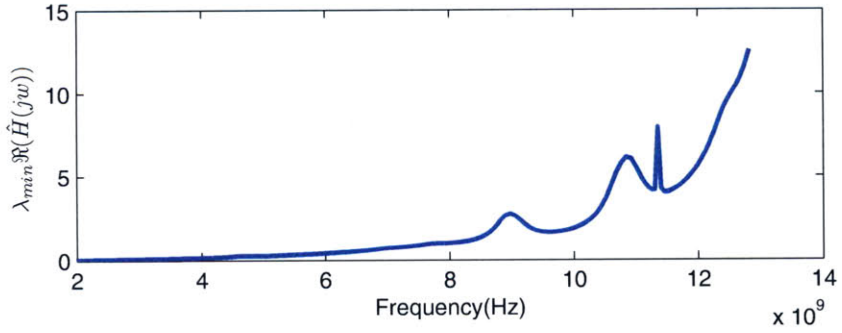


Figure 4-9: $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\})$

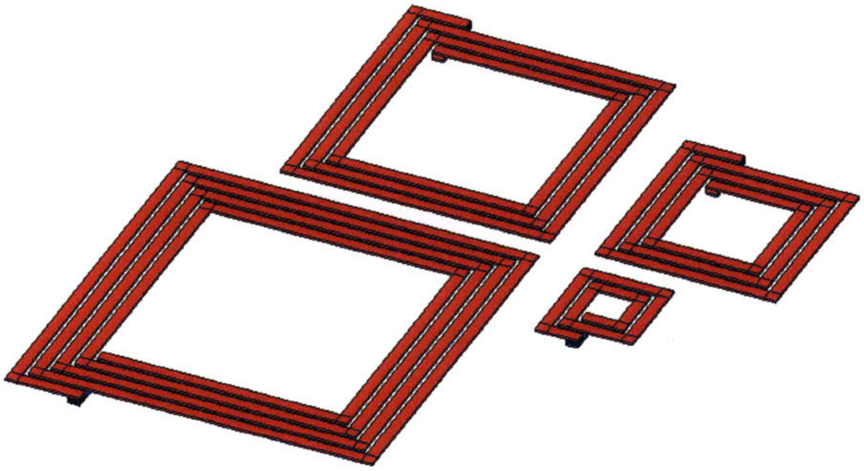


Figure 4-10: 3D layout of the RF inductors (wire widths not to scale)

necessary condition for a model to be passive requires $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\}) > 0$. We plot $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\})$ for our identified model within the bandwidth of interest in Figure 4-9.

4.4.3 On-Chip RF Inductors

The third example is a collection of 4 RF inductors on the same chip or package that are used in the design of multichannel receivers. The layout is shown in Figure 4-10. The array is comprised of four inductors laid out in the form of a 2x2 matrix. Important dimensions of this array are as follows: wire width$= 10\mu m$, wire height$= 4\mu m$, height of inductors above substrate$= 20\mu m$, horizontal separation between sides of two adjacent inductors$= 400\mu m$, length of sides of each inductor$= 800\mu m, 600\mu m, 400\mu m, 200\mu m$, and

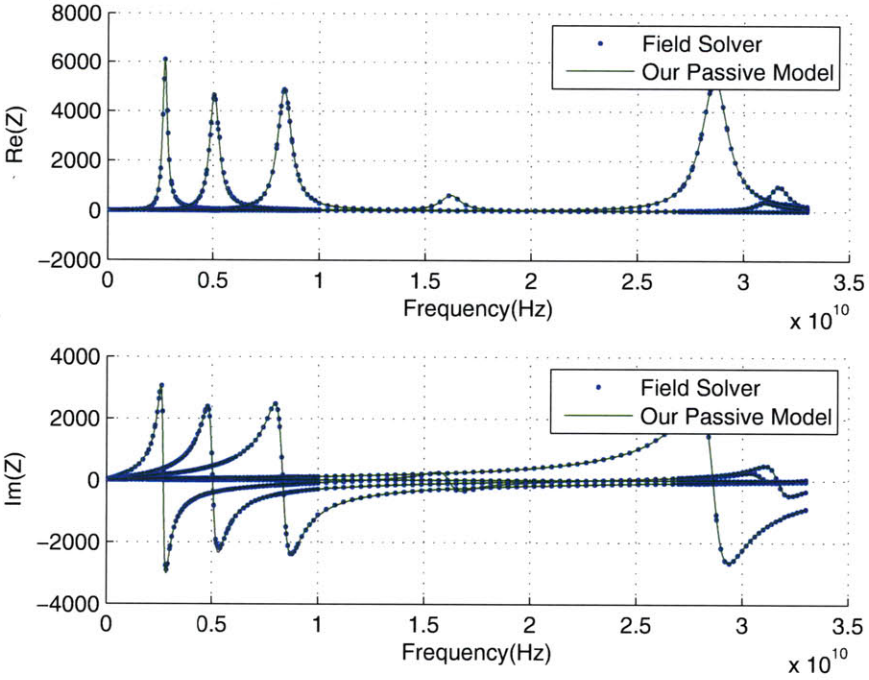


Figure 4-11: Comparing real part and imaginary of impedance from our passive model (solid line) and from field solver (dots) for the RF inductors

having $4,3,3,2$ turns respectively. The structure has four ports in total, configured at the input of each inductor. This structure was simulated using 10356 unknowns in the full wave field solver, FastMaxwell [27] which captures substrate using a Green function complex image method.

For this example a 4×4 passive transfer matrix of order $m = 92$ was identified. The algorithm took $72 seconds$ to identify the passive model, compared to the algorithm in [29] which ran out of memory and did not generate the model.

Figure 4-11 shows impedance parameters both from the field solver and from our identified model. The passive nature of this model was verified by the absence of pure imaginary eigen values of the associated hamiltonian matrix.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

Passive Fitting for Multiport Systems - Method II

In this chapter we describe an algorithm which identifies the unknown passive system as a ratio of matrix polynomial, $P(s)$ and scalar polynomial $q(s)$. In this case our identified model is represented by $\hat{H}(s = j\omega) = P(j\omega)/q(j\omega)$. The problem can be set as to minimize the mismatch between given frequency response samples in either L_2 or L_∞ sense.

$$L_2: \min_{\hat{H}} \sum_i \left| H_i - \frac{P(j\omega_i)}{q(j\omega_i)} \right|^2 \quad OR \quad L_\infty: \min_{\hat{H}} \max \left| H_i - \frac{P(j\omega_i)}{q(j\omega_i)} \right| \qquad (5.1)$$
$$\text{subject to } \hat{H}(j\omega) \quad \textit{passive}$$

where $H_i = H(j\omega_i)$ are given transfer function samples at frequencies $\omega_i \in \mathbb{R}$.

5.1 Semidefinite Formulation of Rational Fitting

In this algorithm we search for reduced models in the following form:

$$\hat{H}(s) = \hat{H}_+(s) + \hat{H}_0(s),$$
$$\hat{H}_+(s) = P(s)/q(s), \text{ and } \hat{H}_0(s) = P_0(s)/q_0(s),$$
$$P, P_0 : \mathbb{C} \mapsto \mathbb{C}^{n\times n}, \text{ are symmetric matrix-valued polynomials}$$
$$q, q_0 : \mathbb{C} \mapsto \mathbb{C}, \text{ are scalar polynomials}$$
$$q : \text{all roots of } q \text{ are in the open left half plane}$$
$$q_0 : \text{all roots of } q_0 \text{ are on the imaginary axis}$$

Here $\hat{H}_+(s)$ accounts for the dissipative part whereas $\hat{H}_0(s)$ accounts for the marginally stable part of the transfer matrix. The marginally stable part of the transfer matrix, $\hat{H}_0$, may be needed to capture effects in the data resulting from non-physical behavior outside the frequency range of interest. Such effects are often numerical artifacts introduced by the field solvers. Since this term is purely imaginary, it does not affect passivity and can therefore be fit using a simple least squares fit.

The transfer matrix of a stable and causal system is completely defined by its real part on the $j\omega$ axis, hence for $\hat{H}_+(j\omega)$ we shall identify $\Re\{\hat{H}_+(j\omega)\}$, where we define matrix polynomial $B = B(\lambda)$ and a scalar polynomial $a = a(\lambda)$ $a = a(\lambda)$ such that the real part, where

$$\hat{H}_+(j\omega) = \frac{P(j\omega)}{a(j\omega)} \quad \Longrightarrow \quad \Re\{\hat{H}_+(j\omega)\} = \frac{B(\omega^2)}{a(\omega^2)}$$

Here $B(\omega^2)$ and $a(\omega^2)$ are real valued matrix and scalar polynomials respectively. Also $B(\omega^2)$ and $a(\omega^2)$ are functions of ω^2 because of the rationalization of $\hat{H}_+(j\omega) = \frac{P(j\omega)}{a(j\omega)}$. Once $B(\omega^2)$ and $a(\omega^2)$ are known, $P(s)$ and $q(s)$ can be uniquely constructed from $B(\omega^2)$ and $a(\omega^2)$ using inverse Hilbert transform. To enforce the passivity conditions given in (2.5), we require $B(\omega^2) = B(\omega^2)^T \succ \forall\omega$, and $a(\omega^2) > 0\ \forall\omega$. The resulting optimization problem can be written as:

$$\begin{aligned} \underset{B,a}{\text{minimize}} \quad & \max_i \left|\Re\{H_i\} - \frac{B(\omega_i^2)}{a(\omega_i^2)}\right|^2 \\ \text{subject to} \quad & B(\omega^2) = B(\omega^2)^T \succ 0 \quad \forall\omega \\ & a(\omega^2) > 0 \quad \forall\omega \end{aligned} \tag{5.2}$$

The objective function in (5.2) is non-convex. However it can be relaxed and formulated as a second order cone program, as described in (5.3).

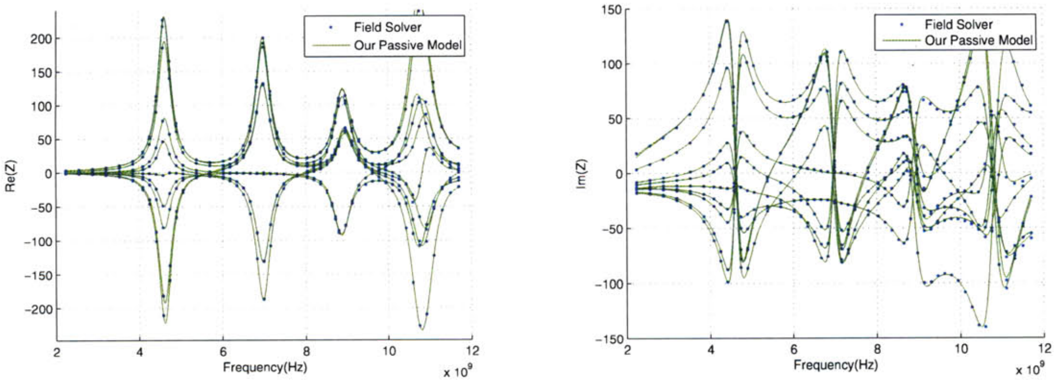
$$\begin{aligned} \max_i \left|\Re\{H_i\} - \frac{B(\omega_i^2)}{a(\omega_i^2)}\right|^2 &\geq \sum_i \left|\Re\{H_i\} - \frac{B(\omega_i^2)}{a(\omega_i^2)}\right|^2 a(\omega_i^2) \quad \text{if} \quad \sum_i a(\omega_i^2) = 1 \\ &= \sum_i \frac{|a(\omega_i^2)\Re\{H_i\} - B(\omega_i^2)|^2}{a^2(\omega_i^2)} a(\omega_i^2) \\ &= \sum_i \frac{|a(\omega_i^2)\Re\{H_i\} - B(\omega_i^2)|^2}{a(\omega_i^2)} \end{aligned} \tag{5.3}$$

The new objective function in (5.3) can be interpreted as a weighted version of the original objective function in (5.2) with normalized weights. We formulate the relaxed optimization problem in (5.4) The optimal solution to (5.4) provides a lower bound for the uniformly optimal solution

$$\begin{aligned} \underset{B,a}{\text{minimize}} \quad & \max_i \frac{\left|a(\omega_i^2)\Re\{H_i\} - B(\omega_i^2)\right|^2}{a(\omega_i^2)} \\ \text{subject to} \quad & B(\omega^2) = B(\omega^2)^T \succ 0 \quad \forall\omega \\ & a(\omega^2) > 0 \quad \forall\omega \\ & \sum_i a(\omega_i^2) = 1 \end{aligned} \tag{5.4}$$

The constraints in (5.4) are enforced as Sum of Squares (SOS) relaxation. Hence the convex program (5.4) is a special case of semidefinite programming and can thus be solved very efficiently using public domain solvers such as SeDuMi [2] or SDPT3 [28].

Once the transfer matrix is identified, it can be transformed into a state-space model and interfaced with commercial circuit simulators using VerilogA.



(a) Comparing real part of impedance from our passive model (solid line) and from field solver (dots)

(b) Comparing imaginary part of impedance from our passive model (solid line) and from field solver (dots)

Figure 5-1: Power Grid: Impedance Parameters

5.2 Results

In this section we shall present modeling examples of various multiport passive structures we previously modeled with the algorithm presented in Chapter 4. All examples are implemented in Matlab and run on a laptop with Intel Core2Duo processor with 2.1GHz clock, 3GB of main memory, and running windows 7.

5.2.1 Power & Ground Distribution Grid

In this example we shall model the same power and ground distribution grid as presented in Section 4.4.2. The 3D layout for this power grid is shown in Figure 4-7. For this example our proposed algorithm identified an 8×8 passive transfer matrix of order $m = 400$. Figures 5-1(a) and 5-1(b) compare the real and imaginary impedance respectively of our reduced model with the field solver data. Figure 5-2 plots the error $e_{i,k}(\omega)$ for each entry of the transfer matrix of the identified model, defined by (4.24).

We have compared our algorithm with standard rational fitting [20] and stable rational fitting algorithms [29] on individual transfer functions. While both alternative methods produce accurate fits to all elements of the transfer function matrix with order $m = 640$, the resulting models are not passive.

We confirmed the passivity of our identified model using Hamiltonian matrix test as

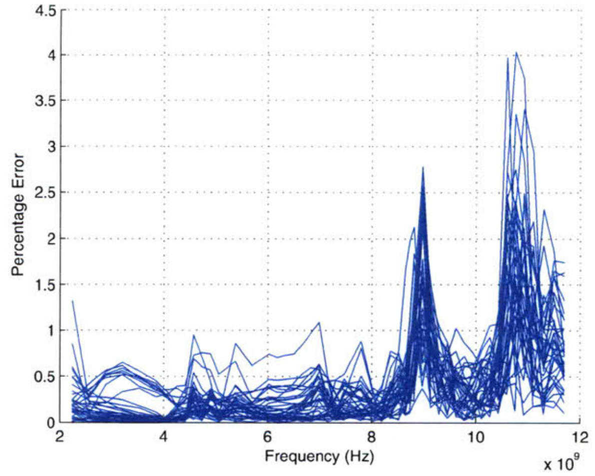


Figure 5-2: Percentage error between the identified model and given samples, defined by (4.24)

described in the Section 2.3.2. Since passivity requires condition (2.5) to hold for all ω, a reduced model is non-passive if $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\}) < 0$ for some ω_i. Figure 5-3 plots $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\})$ for the three reduced models. It is clear from Figure 5-3 that both alternative methods generate models which are non-passive.

5.2.2 On-chip RF Inductors

In this example we model on-chip RF inductors. The structure is described in Section 4.4.3. The layout for this array is shown in Figure 4-10. For this example a 4 × 4 passive transfer matrix of order $m = 96$ was identified.

Figure 5-4 shows impedance parameters both from the field solver and from our identified model. Figure 5-5 plots error $e_{i,k}(\omega)$ of the identified model as defined in (4.24), which attains a maximum of 4.5% error.

To emphasize the importance of preserving passivity during model identification, we identified two additional models for this structure using the standard rational fit [20] and stable rational fit [29] approaches. Although for the same model-complexity ($m = 96$) the rational fits identified quite accurate models, passivity was still not preserved, as is evident from the negative eigenvalues plotted in Figure 5-6 corresponding to the two alternative models.

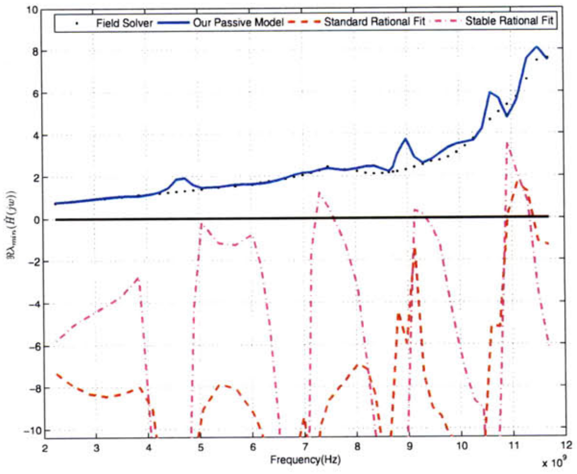
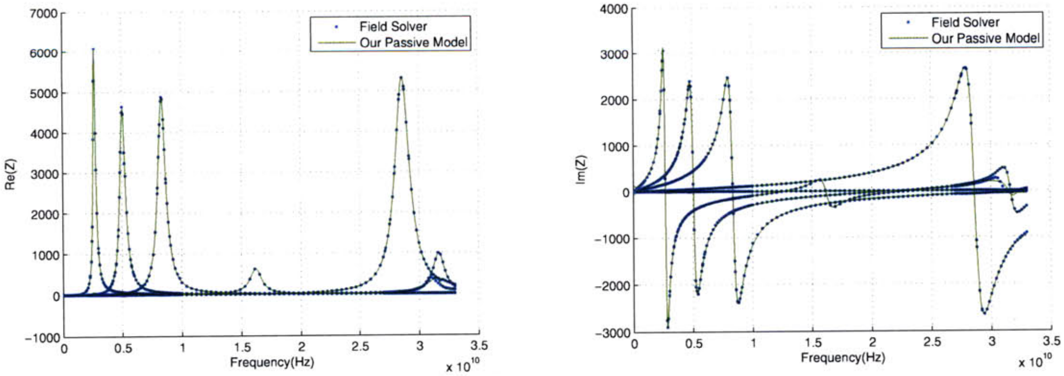


Figure 5-3: $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\})$



(a) Comparing real part of impedance from our passive model (solid line) and from field solver (dots)

(b) Comparing imaginary part of impedance from our passive model (solid line) and from field solver (dots)

Figure 5-4: Inductor Array: Impedance Parameters

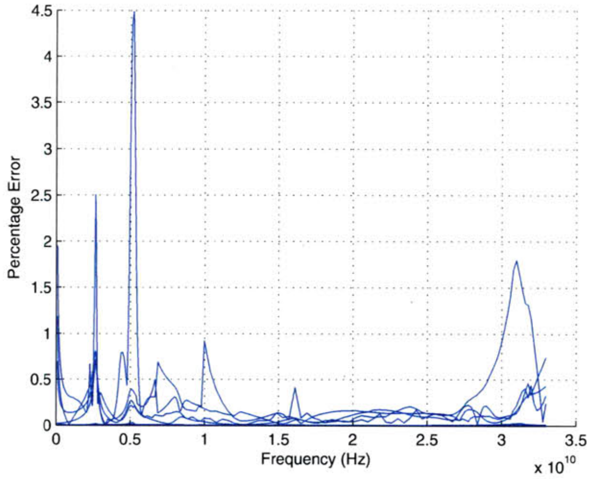


Figure 5-5: Percentage error between the identified model and given samples, defined by (4.24)

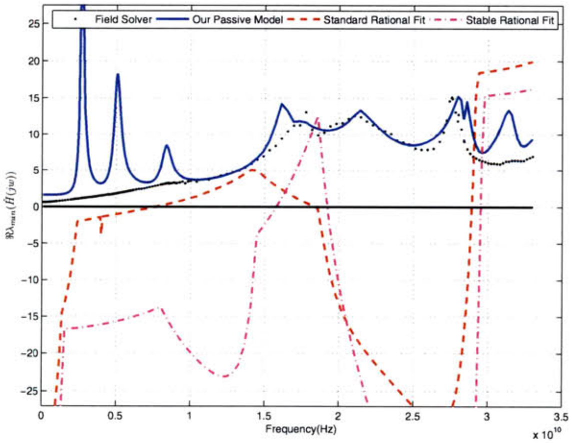


Figure 5-6: $\lambda_{\min}(\Re\{\hat{H}(j\omega_i)\})$

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 6

Interconnection of Passive Identified Models

In this chapter we present a framework for system-level modeling and simulation of complex analog systems which are composed of several linear sub-systems. In the proposed framework stable and passive models are first developed for individual linear building blocks using semidefinite programming based dynamical modeling techniques as described in Chapter 4 and 5. The individual models are interconnected using an automated stamping procedure to generate the representation for the complete linear block.

6.1 Motivation

Typically during analog system level designs, one decomposes a larger system into multiple more manageable sub-system blocks. These smaller blocks may represent linear systems such as passive interconnect structures, power combiners, filters, and distribution grids, and nonlinear systems such as amplifiers, mixers, MEMS structures and non-traditional devices. These sub-systems are simulated in different simulation environments. Nonlinear devices are simulated for time domain response inside spice-like simulators. On the other hand, linear passive interconnect structures are laid out and simulated inside a full wave electromagnetic field solver, which generates frequency response data in the form of S-parameters or Z-parameters. Quite often these passive structures are first fabricated and frequency response data is then collected after physical measurements. To simulate the complete analog architecture, first compact dynamical models are developed from frequency response samples for linear structures. These models are then interfaced with the circuits simulators. Inside the circuit simulator these blocks are interconnected to other sub-systems containing possibly nonlinear devices such as transistors and diodes.

One challenge with such block-level modeling approaches is that individually accurate and stable models connected together could produce an unstable system, such as in the case of non-passive models. In order to compute the response of the complete analog system, the circuit simulator needs to solve differential equations, by either time domain integration or periodic steady state methods. Simulating the overall system accurately requires the final differential equations to be *stable*, which can only be guaranteed if the linear identified models are *passive*. To avoid such instability problems, we use convex optimization techniques described in Chapter 4 and 5 to guarantee that individual linear multiport subsystem blocks are passive [22, 23].

Section 6.2 we discuss an automated stamping procedure to interconnect impedance or admittance type linear state space models

6.2 Interconnection by Automatic Stamping

When the analog system architecture consists of a collection of interconnected linear structures, it is highly desirable to interconnect all the individual passive linear models and generate an equivalent model describing the complete linear block. This section describes an algorithm that constructs the coupled interconnected system using an automated stamping procedure.

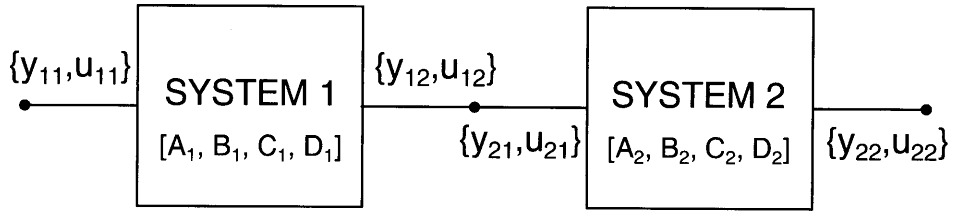


Figure 6-1: Two linear systems interconnected

Consider two linear state space systems described by the following equations:

$$System1: \quad \dot{x}_1 = A_1 x_1 + \begin{bmatrix} B_{11} & B_{12} \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{12} \end{bmatrix}$$

$$\begin{bmatrix} y_{11} \\ y_{12} \end{bmatrix} = C_1 x_1 + \begin{bmatrix} D_{11} & D_{12} \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{12} \end{bmatrix}$$

$$System2: \quad \dot{x}_2 = A_2 x_2 + \begin{bmatrix} B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} u_{21} \\ u_{22} \end{bmatrix}$$

$$\begin{bmatrix} y_{21} \\ y_{22} \end{bmatrix} = C_2 x_2 + \begin{bmatrix} D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} u_{21} \\ u_{22} \end{bmatrix}$$

Here x_1 and x_2 are vectors representing the states of the individual systems. Let these systems represent the impedance of a network. In this case the inputs $u'_{ij}s$ represent the currents flowing into the ports while the outputs $y'_{ij}s$ represent the voltages at the ports. These two systems are interconnected to each other as shown in Figure 6-1. In the model of the overall interconnected electrical network, we need to enforce the conservation laws at the interconnected node, which implies $y_{12} = y_{21}$ and $u_{12} = -u_{21}$.

$$\begin{array}{l} Row\,1: \\ Row\,2: \\ Row\,3: \\ Row\,4: \\ Row\,5: \\ Row\,6: \end{array} \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_{u_{11}} \\ \dot{x}_{u_{12}} \\ \dot{x}_{u_{21}} \\ \dot{x}_{u_{22}} \end{bmatrix} = \begin{bmatrix} A_1 & 0 & B_{11} & B_{12} & 0 & 0 \\ 0 & A_2 & 0 & 0 & B_{21} & B_{22} \\ 0 & 0 & 0 & 1 & 1 & 0 \\ C_1 & -C_2 & D_{11} & D_{12} & -D_{21} & -D_{22} \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_{u_{11}} \\ x_{u_{12}} \\ x_{u_{21}} \\ x_{u_{22}} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{22} \end{bmatrix} \tag{6.1}$$

$$\begin{bmatrix} y_{11} \\ y_{22} \end{bmatrix} = \begin{bmatrix} C_1 & 0 & 0 & D_{12} & 0 & 0 \\ 0 & C_2 & 0 & 0 & D_{21} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_{u_{11}} \\ x_{u_{12}} \\ x_{u_{21}} \\ x_{u_{22}} \end{bmatrix} + \begin{bmatrix} D_{11} & 0 \\ 0 & D_{22} \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{22} \end{bmatrix} \tag{6.2}$$

We introduce extra states $x_{u_{ij}}$ for the inputs in order to enforce conservation laws in the descriptor type state space model for the overall interconnected system, as described in (6.1). Row 1 and Row 2 in (6.1) describe the original system1 and system 2 respectively. Row 3 enforces *KCL* $u_{12} = -u_{21}$, while Row 4 equates the voltage at the connection node $y_{12} = y_{21}$. Row 5 and Row 6 relates the input of interconnected system to the internal states corresponding to these inputs i.e. $x_{u_{11}} = u_{11}$ and $x_{u_{22}} = u_{22}$. This algorithm only requires the description of port interconnections in order to stamp in the connections. Therefore we can connect large numbers of models in any arbitrary configuration at no additional cost.

Chapter 7

Globally Passive Parameterized Model Identification

7.1 Motivation

Globally passive parameterized models are essential if one wishes to explore the design space of a complex system containing interconnected linear and non-linear components. In these models, passivity is required for the whole continuous parameter range, since the user or the optimizer can instantiate the models with any parameter value. The ability to generate passive parameterized models would greatly facilitate the circuit designers. As an example consider a multi-primary transformer which is used for power combining in distributed power amplifiers, as shown in Figure 7-1. The transformer design variables are length, width and spacing for the windings. With a parameterized modeling tool, the designer would be able to create an equivalent circuit block which approximates the dependence of frequency response on design parameters with high fidelity. Such a parameterized modeling tool should also give apriori guarantees of stability and passivity if the final model is to be used in an interconnected environment. The user can interface these equivalent circuit blocks with circuit simulators and run full system simulations, where s/he has the control over design parameters including length, width and spacing. These parameter values can be fine-tuned for optimal power amplifier performance.

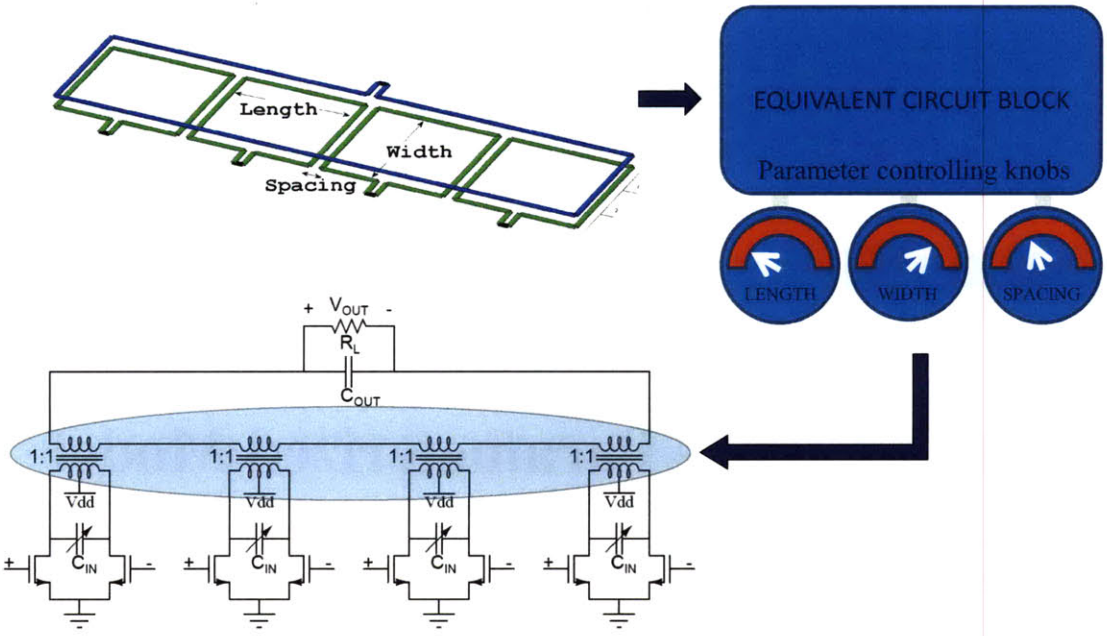


Figure 7-1: A multiprimary transformer parameterized in length, width and spacing. The equivalent circuit block with parameter controlling knobs interfaced with circuit simulator is used for design space exploration of a complete distributed power amplifier design

7.2 Background

7.2.1 Interpolatory Transfer Matrix Parameterization

A common approach to constructing parameterized transfer matrix models is to interpolate between a collection of non-parameterized models, each generated at a different parameter value. Given a set of individual models and parameter values $\{H_i, \lambda_i\}$, where each model is described in pole-residue form (4.1), meaning $H_i = \{R_i, A_i, D_i\}$ where

$$A_i = \begin{bmatrix} a_{1,i} \\ \vdots \\ a_{\kappa,i} \end{bmatrix}, \qquad R_i = \begin{bmatrix} \mathbf{R_{1,i}} \\ \vdots \\ \mathbf{R_{\kappa,i}} \end{bmatrix} \tag{7.1}$$

are a collection of the poles and residues respectively, then the goal is to construct functions $\hat{R}(\lambda)$, $\hat{a}_k(\lambda)$, and $\hat{D}(\lambda)$ that interpolate exactly the given pole-residue models,

$$\hat{R}(\lambda_i) = R_i \qquad \hat{D}(\lambda_i) = D_i \qquad \hat{A}(\lambda_i) = A_i, \tag{7.2}$$

resulting in a parameterized transfer matrix in pole-reside form

$$\hat{H}(s,\lambda) = \sum_{k=1}^{\kappa} \frac{\hat{\mathbf{R}}_{\mathbf{k}}(\lambda)}{s - \hat{a}_k(\lambda)} + \hat{D}(\lambda), \tag{7.3}$$

We $\hat{A}(\lambda)$ and $\hat{R}(\lambda)$ as follows

$$\hat{A}(\lambda) = \begin{bmatrix} \hat{a}_1(\lambda) \\ \vdots \\ \hat{a}_\kappa(\lambda) \end{bmatrix}, \qquad \hat{R}(\lambda) = \begin{bmatrix} \hat{\mathbf{R}}_1(\lambda) \\ \vdots \\ \hat{\mathbf{R}}_\kappa(\lambda) \end{bmatrix}. \tag{7.4}$$

For instance, in [32] piecewise linear functions were used to interpolate exactly the given models, while in [13] interpolation was achieved using Lagrange polynomials. Although such approaches guarantee exact matching at the given set of parameter values used for interpolation, they do so at the expense of being able to ensure reasonable behavior between the points, and they require an extremely large number of coefficients to describe a model interpolating many points. Piecewise linear fits are not smooth, and therefore preclude the use of models for sensitivity analysis where derivatives with respect to the parameter are necessary. Methods based on polynomial interpolation are smooth, but are likely to produce non-monotonic oscillatory curves, resulting in non-physical behavior of the model between the interpolation points.

Additionally, while interpolation approaches can guarantee that the resulting parameterized model $H(s,\lambda)$ is passive (or stable) when evaluated at the set of parameter values λ_i used for interpolation (provided the original set of non-parameterized models are all stable), no guarantees can be made about the passivity (or stability) of the model when evaluated at any parameter value differing from the small set used for interpolation.

7.2.2 Positivity of Functions

Enforcing passivity in a system inevitably relies on ensuring positivity (or non-negativity) of some quantity in the system, such as functions in (7.4). A positivity requirement on an arbitrary function is in general a non-convex constraint making this a difficult task. However, one way to ensure that functions are globally positive is to require them to be expressible as a quadratic form of a positive semidefinite (PSD) matrix. That is, if we want to ensure that a scalar multivariate function is globally non-negative, i.e. $f(x) \geq 0$ is satisfied for all possible x, then we can choose to construct $f(x)$ such that it can be expressed as $\phi(x)^T M \phi(x)$ for some PSD matrix M. Here $\phi(x)$ can be any vector of nonlinear functions of the argument x. If such a construction is used to describe f, then it is guaranteed that $f(x) = \phi(x)^T M \phi(x) \geq 0$ for all possible x. If instead we wish $f(x)$ to be globally negative, we simply require that M_k is a *negative* semidefinite matrix.

Positivity of Polynomials

Let us consider the special case when the function $f(x)$ is a polynomial. A sufficient condition (necessary for univariate case) for the polynomial $p(x) = f(x)$ to be non-negative is that it can be written as a sum of squares (SOS). i.e.

$$\text{If} \quad p(x) = \sum_i g_i^2(x) = SOS \quad \Longrightarrow \quad p(x) \geq 0 \;\; \forall x \tag{7.5}$$

A sum of squares polynomial can be represented as a quadratic form of a positive semidefinite matrix. i.e.

$$p(x) = \sum_i g_i^2(x) = SOS \quad \Leftrightarrow \quad p(x) = \phi^T M \phi, \quad M \succeq 0 \tag{7.6}$$

here ϕ is a vector of monomials. As an example, suppose we want to certify that

$$p(s) = 1 + 4x + 5x^2 \geq 0 \;\; \forall x \tag{7.7}$$

. We can certify (7.7) if we can find an $M \succeq 0$ such that it satisfies (7.6). Equation (7.8) shows that indeed we can such M

$$p(x) = \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix}, \quad M = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \succeq 0 \tag{7.8}$$

Also by performing Cholesky decomposition of M we can express $p(x)$ as SOS explicitly.

$$\begin{aligned} p(x) &= 1 + 4x + 5x^2 \\ &= \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix} \\ &= \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix} \\ &= (1 + 2x)^2 + x^2 \quad \geq 0 \;\; \forall x \end{aligned} \tag{7.9}$$

7.3 Optimal Parameterized Fitting

7.3.1 Problem Formulation

To circumvent the issues resulting from interpolatory fitting approaches described in Section 7.2.1, we formulate the parameterized modeling problem as an approximate fitting problem using optimization. Given a set of Ω non-parameterized models $\{H_i, \lambda_i\}$, we wish to construct a parameterized model $\hat{H}(s,\lambda)$ in the form of (7.3) such that $\hat{H}(s,\lambda_i) \approx H_i(s)$. We assume the models given to us are described in pole residue form (4.1), meaning $H_i = \{A_i, R_i, D_i\}$, as defined in (7.1). This assumption is not restrictive because if models are only available in numerator-denominator form or in state-space form, it is a trivial task to transform them into pole-residue from. It is advantageous to fit models in pole-reside form because the parameters of such model (e.g. the dominant poles) have a physical meaning and can be expected to vary smoothly as the parameters in the system are varied. A state-space model, on the other hand, is a non-unique representation of the system (e.g.

any rotation of the coordinate system would produce completely different coefficients in the system matrices) and therefore there is no reason to think that interpolating between such systems is a reasonable task.

Our goal is to fit parameterized functions describing the poles, residues, and feed through matrix such that

$$\hat{A}(\lambda_i) \approx A_i, \qquad \hat{R}(\lambda_i) \approx R_i, \qquad \hat{D}(\lambda_i) \approx D_i. \tag{7.10}$$

By using optimization-based fitting, as opposed to exact interpolation, we can greatly reduce the number of terms necessary to fit a model to a large number of points. Furthermore, we eliminate the non-physical oscillatory behavior that may arise between interpolation points when fitting high order polynomials to relatively well-behaved curves.

We generally assume that the poles and residues are both complex. Therefore we will identify separately the real and imaginary parts of each

$$\hat{a}_k(\lambda) = \Re\hat{a}_k(\lambda) + j\Im\hat{a}_k(\lambda) \tag{7.11}$$

$$\hat{\mathbf{R}}_\mathbf{k}(\lambda) = \Re\hat{\mathbf{R}}_\mathbf{k}(\lambda) + j\Im\hat{\mathbf{R}}_\mathbf{k}(\lambda) \tag{7.12}$$

The result of this formulation is a set of five minimization problems, that solve for the real and imaginary parts of the poles, residue matrices, and direct matrix as a function of the parameters

$$\begin{aligned} &\min_{\Re\hat{A}} \sum_{i=1}^{\Omega} ||\Re A_i - \Re\hat{A}(\lambda_i)||^2, && \min_{\Im\hat{A}} \sum_{i=1}^{\Omega} ||\Im A_i - \Im\hat{A}(\lambda_i)||^2, \\ &\min_{\Re\hat{R}} \sum_{i=1}^{\Omega} ||\Re R_i - \Re\hat{R}(\lambda_i)||^2, && \min_{\Im\hat{R}} \sum_{i=1}^{\Omega} ||\Im R_i - \Im\hat{R}(\lambda_i)||^2, \\ &\min_{\hat{D}} \sum_{i=1}^{\Omega} ||D_i - \hat{D}(\lambda_i)||^2. \end{aligned} \tag{7.13}$$

The accuracy of the resulting model, and the difficulty of solving the optimization problems, depends on how we choose to describe the unknown functions $\hat{A}(\lambda), \hat{R}(\lambda), \hat{D}(\lambda)$. In the following section we propose a convenient formulation resulting in a semidefinite optimization problem, which can be easily solved using standard freely available software.

7.3.2 Rational Least Squares Fitting

Consider the case where the unknown functions to be identified are described as the ratio of two unknown functions, each of which is expressed as a linear combination of basis functions

$$g(\lambda) = \frac{\alpha(\lambda)}{\beta(\lambda)} = \frac{\sum_{n=1}^{N} \alpha_n \phi_n(\lambda)}{\sum_{m=1}^{M} \beta_m \psi_m(\lambda)}. \tag{7.14}$$

Here ψ and ϕ are predetermined basis functions, and although we use the term 'rational', we are not forcing ϕ and ψ to be polynomials, and it is not even necessary that ϕ and ψ be the same class of functions.

Given samples g_i, the optimization task consists of solving for coefficients α_n and β_m in order to minimize

$$\sum_i \left|\left| g_i - \frac{\alpha(\lambda_i)}{\beta(\lambda_i)} \right|\right|^2.$$

Unfortunately, attempting to minimize directly such quantity is a difficult (i.e. non-convex) task due to the nature of the nonlinear dependence on the unknown coefficients α_n, β_m. Instead, a useful relaxation [22] of this objective transforms it into the following convex problem

$$\min_{\alpha,\beta} \sum_{i=1}^{\Omega} \frac{||\beta(\lambda)g_i - \alpha(\lambda_i)||^2}{\beta(\lambda)}. \tag{7.15}$$

Along with the normalization constraint that $\sum_m \beta_m = 1$, this problem can be viewed as a weighted least squares minimization of the desired objective $||g_i - \alpha(\lambda_i)/\beta(\lambda_i)||$. Although the problem is still nonlinear in the unknowns, the formulation is convex (specifically, minimization over a 'rotated Lorentz cone') and thus can be solved efficiently using freely available software [31].

In order to utilize such a description to solve (7.13), one must select the basis functions describing the five unknown quantities $(\Re\hat{a}, \Im\hat{a}, \Re\hat{R}, \Im\hat{R}, \hat{D})$, and solve a set of five optimization problems in the form of (7.17). Note that the basis functions used to describe the five different quantities need not be the same.

7.3.3 Linear Least Squares

If we restrict the denominator in rational formulation (7.14) to be constant unity (i.e. $\beta(\lambda) = 1$), then we can consider functions defined as a linear combination of basis functions

$$g(\lambda) = \sum_{n=1}^{N} \alpha_n \phi_n(\lambda), \tag{7.16}$$

which simplifies the optimization problem to the standard 'linear least squares' problem

$$\min_{\alpha} \sum_{i=1}^{\Omega} ||g_i - g(\lambda_i)||^2. \tag{7.17}$$

This optimization problem is an unconstrained linear least squares minimization that is convex and can be solved without any relaxation using freely available software [31], or even solved analytically.

7.3.4 Polynomial Basis Example

As an illustrative example of how one uses optimization problem (7.15) to identify the parameterized model, suppose we wish to fit the real part of the poles, $\Re\hat{A}(\lambda)$, to a function of two parameters λ_1, λ_2 using a polynomial basis. If we choose a second order polynomial basis for the denominator ψ, meaning that $\Psi = [1, \lambda_1, \lambda_2, \lambda_1^2, \lambda_1\lambda_2, \lambda_2^2]$, and a first order polynomial basis for the numerator ϕ, meaning $\Phi = [1, \lambda_1, \lambda_2]$, then the resulting function expression for each individual pole $\Re\hat{a}_k(\lambda)$ would be

$$\Re\hat{a}_k(\lambda) = \frac{\alpha_0 + \alpha_1\lambda_1 + \alpha_2\lambda_2}{\beta_0 + \beta_1\lambda_1 + \beta_2\lambda_2 + \beta_3\lambda_1^2 + \beta_4\lambda_1\lambda_2 + \beta_5\lambda_2^2}.$$

This expression is then used in optimization problem (7.15) to solve for the unknown coefficients α, β for each of the κ poles.

If instead we wish to use a polynomial function description to solve the linear least squares problem (7.17), then a second order polynomial basis yields the function

$$\Re \hat{a}_k(\lambda) = \alpha_0 + \alpha_1\lambda_1 + \alpha_2\lambda_2 + \alpha_3\lambda_1^2 + \alpha_4\lambda_1\lambda_2 + \alpha_5\lambda_2^2.$$

It is important to point out here that exact polynomial interpolation, such as in [13], can be thought of as a very special case of our framework. Specifically, if we choose polynomials as basis functions, select the basis ϕ to allow as many degrees of freedom as data points (i.e. $N = \Omega$), and choose the linear least squares formulation, then the unconstrained optimization problem (7.17) is *equivalent to exact polynomial interpolation*. However, unlike interpolation formulations, our optimization approach is neither confined to using a polynomial basis nor to using as many coefficients as data points, and therefore the complexity of the resulting model does not scale poorly with the number of points used for fitting.

7.3.5 Complexity of Identification

The cost of identifying parameterized model (7.3) using the previously described optimization procedure is extremely cheap because many of the unknown quantities are uncoupled, and additionally there are multiple redundancies within the system. As previously mentioned, the real and imaginary parts of the poles can be solved for separately, as described in (7.13). If the model is described by κ poles, then solving for $\Re\hat{A}(\lambda)$ and $\Im\hat{A}(\lambda)$ can each be separated into κ different optimization problems, because the poles do not depend upon one another. Additionally, since the residue matrices and direct term are symmetric, for a system with T ports, there will only be $T(T+1)/2$ unique elements to fit (corresponding to the upper triangular part of the symmetric matrices). Lastly, since the complex poles and residues occur in conjugate pairs, it is only necessary to fit half of the non-zero imaginary pole parts in $\Im\hat{A}(\lambda)$, and only necessary to fit half of the residue matrices $\Im\hat{R}(\lambda)$.

7.4 Constrained Fitting for Stability

The previously posed unconstrained optimization problems (7.13) enforce optimal accuracy of the parameterized transfer function with respect to the individual models, but provides no guarantees on global properties, such as stability, for the system. Although pointwise interpolation as described in Section 7.2.1 will yield a model that is stable at the parameter values used for fitting (assuming the original non-parameterized models are each stable), it provides no stability guarantee when evaluating the model at any other parameter value. In this section we formulate additional constraints for the previously defined optimization problems that allow us to generate guaranteed stable parameterized transfer function models.

7.4.1 Stable Pole Fitting

Stability of a model in pole-residue from (4.1) depends *only* on the real part of the poles $\Re\hat{A}(\lambda)$. We say the parameterized model $H(s,\lambda)$ is stable at a particular parameter value λ if all of the poles have negative real part at that parameter value, i.e. $\Re\hat{A}(\lambda) < 0$. Thus, enforcing guaranteed stability of $H(s,\lambda)$ for all parameter values λ can be achieved by enforcing negativity of $\Re\hat{A}(\lambda)$ *for all* such λ. Note that since we are modeling the residues and direct term separately, enforcing stability only affects one of the five optimization problems in (7.13).

Enforcing negativity when fitting the real part of the poles $\Re\hat{A}(\lambda)$ requires adding a constraint to the previously unconstrained minimization problems. If solving the linear least squares problem (7.16), the constrained problem becomes

$$\begin{aligned}&\min_{\Re\hat{A}}\sum_i ||\Re A_i - \Re\hat{A}(\lambda_i)||^2 \text{ subject to} \\ &\Re\hat{a}_k(\lambda) < 0 \quad \forall\, k.\end{aligned} \tag{7.18}$$

If we are instead fitting to rational functions as described in (7.14), then the pole function is negative if the numerator $\Re a_k(\lambda)$ is negative and the denominator $\Re\beta_k(\lambda)$ is positive, resulting in the constrained optimization problem

$$\min_{\Re\hat{A}} \sum_i ||\Re A_i - \Re\hat{A}(\lambda_i)||^2 \text{ subject to} \tag{7.19}$$
$$\Re\alpha_k(\lambda) < 0, \quad \Re\beta_k(\lambda) > 0 \;\; \forall\, k$$

To enforce these positivity and negativity constraints, we will require the functions to be describable as positive definite quadratic forms in some nonlinear basis. That is, define $\Re a_k(\lambda) = \Lambda(\lambda)^T M_k \Lambda(\lambda)$ where $\Lambda(\lambda)$ is a vector of functions of λ, and if M_k is a positive definite matrix, then $\Re a_k(\lambda) > 0$ is a positive function for all λ. This is a standard technique for enforcing positivity of functions as described in Section 7.2.2, and can similarly be used to enforce negativity by requiring M_k to be a negative definite matrix. Problems of the form (7.19) with semidefinite matrix constraints can be solved using semidefinite optimization solvers [31].

For example, if fitting to a quadratic function of two parameters, we would select

$$\Lambda(\lambda) = \begin{bmatrix} 1 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} \tag{7.20}$$

and would define $\Re a_k(\lambda_i) = \Lambda(\lambda_i)^T M_k \Lambda(\lambda_i)$, and M_k would contain the unknown coefficients α_n.

7.5 Constrained Fitting For Passivity

In additional to stability, it is desirable that the identified models also be passive. In this section we extend the optimization framework presented in Section 4.3.2 of Chapter 4 to include parameterization while preserving passivity.

7.5.1 Parameterized Residue Matrices

To enforce global passivity, we require that the individual non-parameterized models conform to the passivity conditions described in Chapter 4. Hence in our final parameterized model we want the constraints in 4.22 to be satisfied for all values of the parameter. Given stable approximation of poles $a(\lambda)$ we compute passive residue matrices by solving the following convex optimization problem

$$
\begin{array}{ll}
\underset{\mathbf{R}^{\mathbf{r}}_{\mathbf{k}}(\lambda),\mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda)}{\text{minimize}} & \sum_{i=1}^{\Omega} ||\Re R_i - \Re \hat{R}(\lambda_i)||^2 + \sum_{i=1}^{\Omega} ||\Im R_i - \Im \hat{R}(\lambda_i)||^2 \\
\text{subject to} & \mathbf{R}^{\mathbf{r}}_{\mathbf{k}}(\lambda) \succeq 0 \; \forall k = 1,\ldots,\kappa_r \; \forall \lambda \\
& -\Re a^c_k(\lambda) \Re \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) + \Im a^c_k(\lambda) \Im \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) \succeq 0 \; \forall k = 1,\ldots,\kappa_c \; \forall \lambda \\
& -\Re a^c_k(\lambda) \Re \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) - \Im a^c_k(\lambda) \Im \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) \succeq 0 \; \forall k = 1,\ldots,\kappa_c \; \forall \lambda \\
\text{where} & \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) = \Re \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda) + j \Im \mathbf{R}^{\mathbf{c}}_{\mathbf{k}}(\lambda)
\end{array}
$$

R_i and $\hat{R}(\lambda_i)$ are defined in (7.1) and (7.4) respectively. Note that the constraints in (7.21) require matrix valued functions to be positive definite, such a constraint can be enforced using SOS relaxation as described in Section 7.2.

7.5.2 Positive Definite Direct Matrix

One necessary condition for passivity of a model described in pole-reside form 4.1 is that the direct term D be a positive semidefinite matrix.

$$
\begin{array}{ll}
\underset{D(\lambda)}{\text{minimize}} & \sum_{i=1}^{\Omega} ||D_i - D(\lambda_i)||^2 \\
\text{subject to} & D(\lambda) \succeq 0
\end{array}
$$

We notice that since each of the given individual models is passive, then $D_i \succeq 0$ and can therefore be factored into a matrix V_i such that $D_i = V_i^T V_i$. Therefore, in order to enforce positive definiteness in $\hat{D}(\lambda)$, we will instead fit $\hat{V}(\lambda)$ to factors of the given direct terms V_i, and then define $\hat{D}(\lambda) = \hat{V}(\lambda)^T \hat{V}(\lambda)$. Since we are enforcing positivity after identification, the result is an unconstrained minimization problem

$$\min_{\hat{V}} \sum_i ||V_i - \hat{V}(\lambda_i)||^2 \tag{7.21}$$

that can be solved using either linear least squares formulation (7.16) or the rational formulation (7.14) to describe $\hat{V}(\lambda)$.

7.6 Implementation

In this section we describe in detail our parameterized model identification procedure based on solving the optimization problems derived in Section 7.4 and 7.5. We also discuss methods for properly generating the individual non-parameterized models and methods for transforming the identified parameterized transfer matrix into a circuit usable in commercial simulators.

7.6.1 Individual Model Identification and Preprocessing

The ability to obtain a reasonable functional approximation between models depends crucially on the assumption that the poles trace out some nice smooth curve as the parameters vary. For the dominant poles of the system this is a safe assumption, because the dominant poles in the individual linear models have a physical connection to the original large system. However, for the other less significant poles in the individual identified models, there may be many possible pole configurations that all produce a good match when originally creating those individual models. Therefore, when performing the original identification of the individual non-parameterized models, it may be beneficial to aid the pole placement of the identification procedure as shown in Algorithm 2. To perform this task, we identify poles for the first or nominal parameter value. We then use the poles identified for the previous parameter value as the initial guess to identify poles for neighboring parameter values. This way we ensure the smoothness in the path of the poles. Also if we want to enforce global passivity in the final parameterized model, the initial non-parameterized models must be identified using the algorithm described in Chapter 4.

Algorithm 2 Successive Pole Placement Algorithm

```
Given $\Omega$ frequency response data sets
Identify first non-parameterized model $H_1(s)$ described in pole-residue form (4.1)
for i=2:$\Omega$ do
    Use pole set $A_{i-1}$ as initial guess for pole set $A_i$, and identify model $H_i(s)$
end for
```

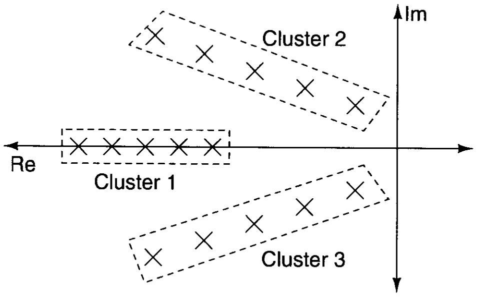


Figure 7-2: Sample clustering

Once the collection of models H_i in pole-residue form, each containing κ poles, has been identified using Algorithm 2, the next task is to construct κ functions, fitting each pole trajectory. Therefore, it is crucial that we 'order' the κ poles such that they are 'clustered' accordingly. We employ a recursive Euclidean distance based clustering approach as described in Algorithm 3. A robust implementation of this algorithm is required since we may encounter complicated scenarios, such as poles crossings and bifurcations. Figure 7-2 shows sample clusters for a system with three poles.

Algorithm 3 Pole Ordering Procedure

```
Given collection of models each having $\kappa$ poles
Convert all models to pole-residue form if necessary
for i=2:$\Omega$ do
        Compute pairwise distance matrix $D_{dist}$ between $A_{i-1}$ and $A_i$
        Group $a_{j,i}$ with $a_{k,i-1}$ based on $\underset{j,k}{\text{minimum}}\, D_{dist}(j,k)$, $\forall j,k \in \{1,2,...,\kappa\}$
end for
```

7.6.2 Parameterized Identification Procedure

The main part of our complete modeling approach, presented in Algorithm 4, is to solve for parameterized functions describing the real and imaginary parts of the poles, residues, and direct term. These optimization problems were derived in Section 7.4 and 7.5. There are two important details we wish to emphasize in this part. First, to enforce positive definiteness on the direct term $D(\lambda)$, we fit instead to the factor $V(\lambda)$ and then define $D(\lambda) = V(\lambda)^T V(\lambda)$, as described in Section 7.5. Second, We want to emphasize that when solving for the real part of the poles $\Re A(\lambda)$ we must enforce negativity to ensure stability of the resulting model, as was described in Section 7.4.1. The remaining components of the model can be solved for using either the unconstrained minimization described in Section 7.3 if passivity is not required, or the constrained minimization described in Section 7.5 if apriori global passivity is required.

Algorithm 4 Complete Parameterized Stable model Fitting

```
Given collection of models generated using Algorithm 2 and ordered using Algorithm 3
if ENFORCE GLOBAL PASSIVITY then
    Select basis functions for direct term factor $\hat{V}(\lambda)$ and solve unconstrained optimization problem (7.21) for $\hat{V}(\lambda)$
    Select basis for real part of the poles $\Re\hat{A}(\lambda)$ and solve constrained optimization problem (7.18) for $\Re\hat{A}(\lambda)$
    Select basis for imaginary parts of poles $\Im\hat{A}(\lambda)$and solve unconstrained optimization problem (7.13) for $\Im\hat{A}(\lambda)$
    Select basis for real and imaginary parts of the residues $\Re\hat{R}(\lambda)$ and $\Im\hat{R}(\lambda)$ and solve the constrained optimization problem (7.21) for $\Re\hat{R}(\lambda)$ and $\Im\hat{R}(\lambda)$
else
    Select basis functions for direct term matrix $\hat{D}(\lambda)$ and solve unconstrained optimization problem (7.13) for $\hat{D}(\lambda)$
    Select basis for real part of the poles $\Re\hat{A}(\lambda)$ and solve constrained optimization problem (7.18) for $\Re\hat{A}(\lambda)$
    Select basis for real part of the residues $\Re\hat{R}(\lambda)$ and solve unconstrained optimization problem (7.13) for $\Re\hat{R}(\lambda)$
    Select basis for imaginary parts of poles $\Im\hat{A}(\lambda)$ and residues $\Im\hat{R}(\lambda)$ and solve unconstrained optimization problem (7.13) separately for $\Im\hat{R}(\lambda)$ and $\Im\hat{A}(\lambda)$
end if
Define final parameterized model
        $H(s,\lambda) = \sum_k \frac{\hat{\mathbf{R}}_{\mathbf{k}}(\lambda)}{s - \hat{a}_k(\lambda)} + \hat{D}(\lambda)$
    where
        $\hat{a}_k(\lambda) = \Re\hat{a}_k(\lambda) + j\Im\hat{a}_k(\lambda)$
        $\hat{\mathbf{R}}_{\mathbf{k}}(\lambda) = \Re\hat{\mathbf{R}}_{\mathbf{k}}(\lambda) + j\Im\hat{\mathbf{R}}_{\mathbf{k}}(\lambda)$
```

7.6.3 Post-processing Realization

The parameterized models generated by our approach can be readily converted into equivalent circuit blocks using VerilogA. Since our models are guaranteed to be stable in the parameter space, we use voltage sources as the 'controlling knobs'. These voltage sources are used to change parameter value after the model is instantiated inside the circuit simulator.

Several equivalent state space realizations for our parameterized models can be achieved. For example, a Jordan-canonical form can be obtained as described in [4]. However in order to have a better performance, such a realization needs to be diagonalized before interfacing with the circuit simulators.

7.7 EXAMPLES

In this section we shall present three examples highlighting different aspects of our proposed methodology in modeling guaranteed stable models. All examples in this section are implemented in Matlab and run on a laptop having Intel Core2Duo processor with 2.1GHz clock, 4GB of main memory, and running windows vista. We have also posted free open source software implementing these procedures online [1].

7.7.1 Single port - Single parameter: Microstrip Patch Antenna

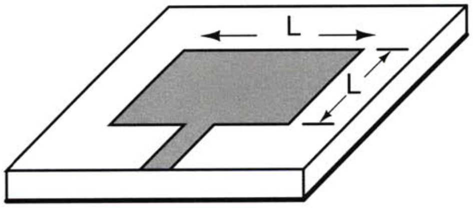


Figure 7-3: Layout of microstrip square patch antenna

The first example considered is a microstrip square patch antenna, shown in Figure 7-3. It is designed on *Rogers RT*5800 substrate having thickness of 25*mils* (1*mil* = 0.001*inch*), relative permitivity $\varepsilon_r = 2.2$ and loss tangent $tan\delta = 0.0009$. The layout is shown in Fig-

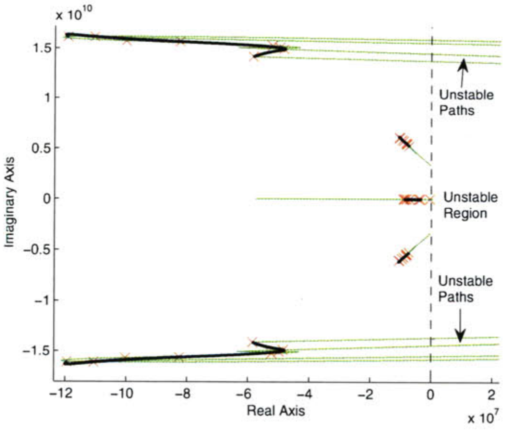


Figure 7-4: Plot showing the trajectory of poles with parameter variation. Thick black lines trace the poles' location from our stable parameterized model, while thin grey (or green) lines trace the poles' location from the unconstrained fit (which clearly becomes unstable)

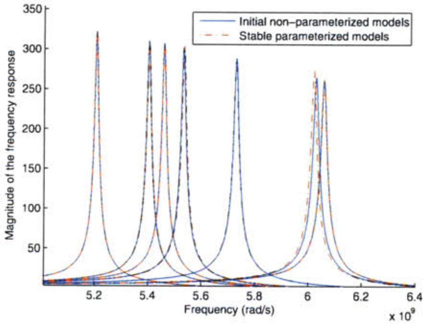


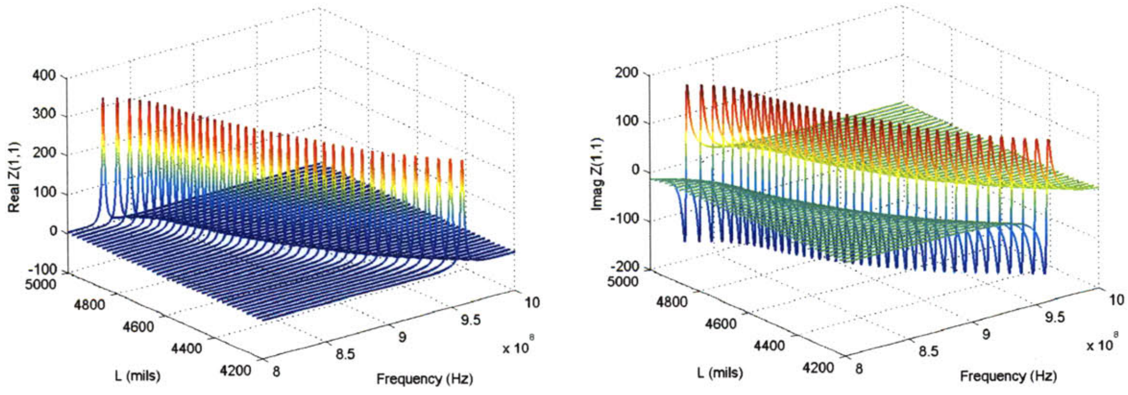
Figure 7-5: Comparison of magnitude of frequency responses of patch antenna parameterized model (dashed lines) with the initial non-parameterized models (solid lines-almost overlapping) for different parameter values. Some traces are from parameter values not used for fitting

ure 7-3. In order to control the resonant frequency, we select the side length of the square 'L' as the model parameter. A collection of individual non-parameterized models were generated by simulating the structure for S-parameter samples from $0.5GHz$ to $2GHz$ using SONNET Lite, where L was varied from $4000mils$ to $5000mils$ with an increment of $100mils$. The resulting models each have order $\kappa = 5$ and were generated and preprocessed for each value of L as described in Algorithm 2 and Algorithm 3 of Section 7.6.1.

A stable parameterized model was then identified using Algorithm 1 along with a polynomial basis of degree $N = 8$ for each of the model components. For this example the entire fitting procedure was completed in just 2.48 *seconds*.

To illustrate the guaranteed stability of the resulting parameterized model, Figure 7-4 plots the trajectory of poles in response to changes in parameter L variation. The thick black line, corresponding to our stable parameterized model, is always in the left half plane, meaning the model is stable at all parameter values. On the other hand, the thin green line, corresponding to a parameterized model generated using polynomial interpolation without stability constraints, crosses into the right half plane and is unstable for many parameter values within the shown range.

To verify the accuracy of our parameterized model, Figure 7-5 compares the frequency response magnitude of our model (dashed red line) to the response of individual non-parameterized models at a set of different parameter values, some of which were not used for fitting. Furthermore, in order to show that our model response smoothly to changes in the parameter over the entire range of interest, Figure 7-6 plots the frequency response of our stable parameterized model as a function of densely sampled parameter values.



(a) Real part of frequency response

(b) Imaginary part of frequency response

Figure 7-6: Surface traced by frequency response of parameterized model of patch antenna over parameter sweep

7.7.2 Multi port - Single parameter: Wilkinson Power Divider

In this example we consider a multiport wilkinson divider, shown in Figure 7-7. The standard wilkinson divider is designed on alumina substrate with the following specifications: characteristic impedance $Z_0 = 75\Omega$, substrate dielectric constant $\varepsilon_r = 9.8$, substrate height

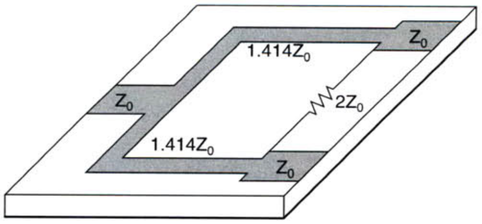


Figure 7-7: Layout of Wilkinson Divider

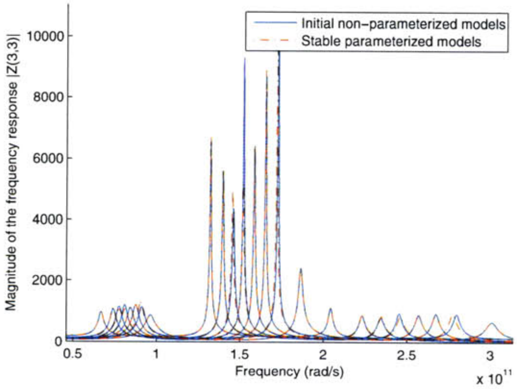


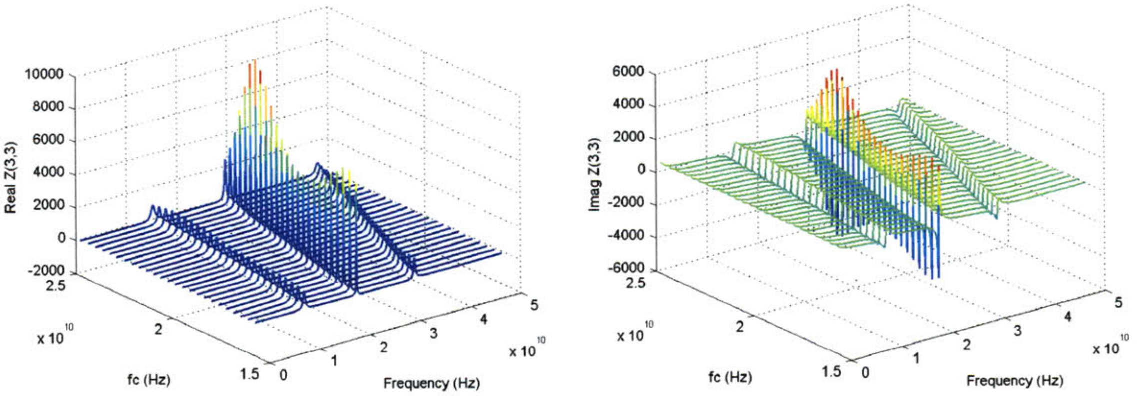
Figure 7-8: Comparison of magnitude of frequency responses, $|Z(3,3)|$, of wilkinson divider parameterized model (dotted lines) with the initial non-parameterized models (solid lines) for different parameter values. Some traces are from parameter values not used for fitting

$h = 125\mu m$, and metal thickness of $t = 4\mu m$. A natural parameter choice for this example is the center frequency f_c. A collection of non-parameterized models are generated by simulating the structure using a full wave field solver [3, 27] while varying the parameter from $15GHz$ to $25GHz$, and using Algorithms 2 and 3 for fitting. The resulting models each have order 33 and are described by 11 poles.

A stable parameterized model with coefficients described by 4^{th} order polynomials is generated using Algorithm 1 to fit to the previously generated individual models. The fitting procedure required just 1.86 *seconds* to solve all optimization problems in Algorithm 1.

To show the accuracy of our parameterized model, one component of the frequency response ($|Z(3,3)|$) is plotted in Figure 7-8 (dashed red line) and compared to the response of individually fitted non-parameterized models, some of which were not used for fitting.

Lastly, Figure 7-9 plots the frequency response of our stable parameterized model as a function of densely sampled parameter values to show the smoothness of our final models.



(a) Real part of frequency response, $\Re Z(3,3)$

(b) Imaginary part of frequency response, $\Im Z(3,3)$

Figure 7-9: Surface traced by frequency response $Z(3,3)$ of parameterized model of wilkinson divider over parameter sweep

7.7.3 Multi port - Multi parameter: T-Type Attenuator

As our third example we consider a T-type attenuator. The purpose of this example is to show the full flexibility of our algorithm. This is a multiport, *multivariate* example where we consider *two* design parameters. We chose *rational basis* of *different degrees* to approximate different elements of the model. The frequency response samples were obtained by simulating the schematic in matlab. Individual non-parameterized models of order 14 and described by 7 poles each are generated and preprocessed for each value of the parameters, controlling attenuation λ_1 and resonant frequency λ_2, as described in Algorithm 2 and Algorithm 3.

A stable *multivariate parameterized model* is generated using Algorithm 1 using the rational function description in (7.15). In this example the numerator of the residues $\hat{R}(\lambda_1, \lambda_2)$, direct term factor $\hat{D}(\lambda_1, \lambda_2)$, and poles $\hat{A}(\lambda_1, \lambda_2)$ are described by polynomials of degree 5, 4, and 2 respectively, while the denominators are chosen as 4^{th} order polynomials for each term. The identification required 6.6 *seconds* to solve all optimization problems in Algorithm 1.

To show that our parameterized model is stable with respect to both parameters, Figure 7-10 plots the densely sampled surface traced by real part of one of the dominant poles from our parameterized model as a function of λ_1 and λ_2.

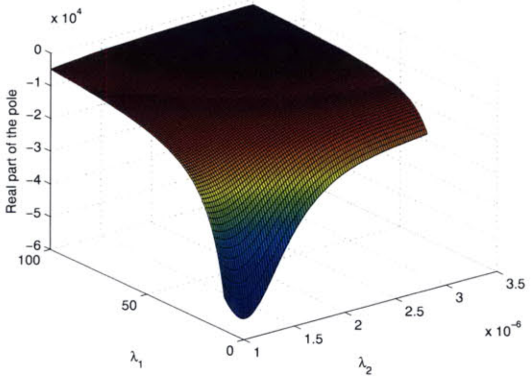


Figure 7-10: Surface traced by real part of one of the dominant poles from our *stable* multivariate parameterized model as a function of λ_1 and λ_2

An excellent match between initial non-parameterized models and final parameterized models can be observed in Figures 7-11 & 7-12.

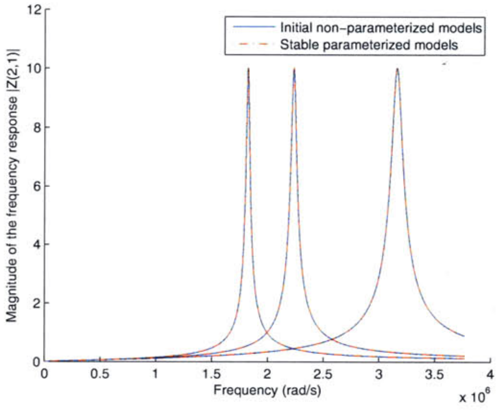


Figure 7-11: Comparison of magnitude of frequency responses, $|Z(2,1)|$, of attenuator multivariate parameterized model (dotted lines) with the initial non-parameterized models (solid lines). Fixed λ_1 varying λ_2

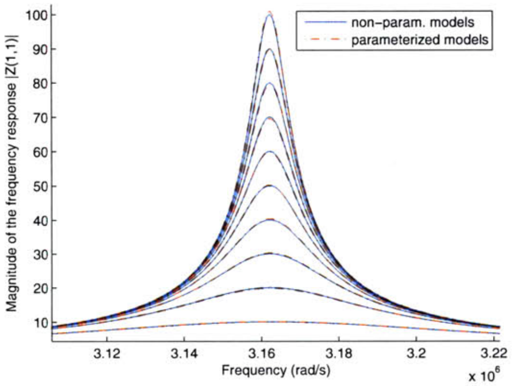


Figure 7-12: Comparison of magnitude of frequency responses, $|Z(1,1)|$, of attenuator multivariate parameterized model (dotted lines) with the initial non-parameterized models (solid lines). Fixed λ_2 varying λ_1

Chapter 8

Conclusion

In this thesis we have presented various highly efficient algorithms to identify individual and parameterized multiport passive models from frequency domain transfer matrix data samples. The algorithms are based on convex relaxations of the original non-convex problems. In the first algorithm, we identify a collection of first and second order networks to model individual non-parameterized passive blocks. Passivity of the overall model is guaranteed by enforcing passivity on the building blocks. The problem is solved in two steps. In the first step we identify a set of common poles for the transfer matrix. In the second step we use the common set of stable poles from step one and identify residue matrices which minimize the mismatch between the model and the given data, and simultaneously conforming to passivity conditions. Several examples are presented which advocate the speed and efficiency of the proposed algorithm. In these examples we have tested passivity of the identified models by verifying the absence of purely imaginary eigenvalues of the associated Hamiltonian matrix. The identified models are interfaced with commercial circuit simulators and used for time domain simulations of complete architectures including a LINC power amplifier where multiport passive model was identified for the passive combining network. The proposed algorithm is compared with existing algorithms based on optimization framework. The comparisons show that our algorithm achieved a speed-up of 40× for some examples while for other examples we generated a highly accurate model in decent amount of time whereas the alternative algorithm ran out of memory and failed to generate a model.

In the second algorithm for the identification of individual non-parameterized passive models, we identify the poles and residues or equivalently numerator and denominator polynomials for the transfer matrix in a single step. Since the complete passive model is identified in a single step, the final model will be near-optimal. In this algorithm we exploit the property of causal and stable systems for which the dispersion relations hold and the real and imaginary parts of the frequency response are related by hilbert transform. We formulate the minimization problem as a convex optimization problem where we simultaneously enforce passivity using semidefinite constraints. Several examples are presented which support the algorithm. The identified models are verified for passivity using hamiltonian matrix based eigenvalue test. We also present an efficient automated stamping based algorithm to interconnect these passive models.

Finally we present an algorithm to identify globally stable and passive multiport models. In this algorithm we combine individual stable and passive non-parameterized models to develop a closed form parameterized model. The final parameterized model conforms to passivity conditions during identification and comes with apriori global passivity certificates in the continuous parameter range of interest. In several examples we have verified that the models generated by our approach can be safely instantiated for any parameter value and always result in a stable and passive system, as opposed to all existing interpolation approaches. We have also shown that our fitting approach only requires few seconds to identify practical passive circuit components, having formulated the problem as an efficient convex optimization program. Finally, a smooth model behavior in between original parameter data points has been enforced in the model construction procedure, and has been observed in all examples.

Appendix A

Semidefintie Programming

In the standard form of a semidefinte program a linear cost function is minimized subject to linear matrix inequalities.

$$\begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \quad F_1 x_1 + F_2 x_2 + \cdots + F_n x_n - F_0 \succeq 0 \end{aligned} \tag{A.1}$$

describes the standard form of a semidefinite program, here all of the matrices $F_0, F_1, \ldots F_n \in S^k$, here S^k indicates set of symmetric matrices of order $k \times k$. The problems that we have described in this thesis are not in the standard form, however they can easily be transformed into the standard representation. Some of the relevant transformations are described in the following sections.

A.1 Minimizing Quadratic Function

Suppose we wish to minimize a quadratic function of the form $||Ax - b||^2$. We can cast this minimization problem into an equivalent semidefinite program as

$$\begin{aligned}
\underset{x}{\text{minimize}}||Ax-b||^2 \quad & \equiv \underset{t,x}{\text{minimize}} \quad t \\
& \quad \text{subject to} \quad (Ax-b)^T(Ax-b) \le t^2 \\
& \equiv \underset{t,x}{\text{minimize}} \quad t \\
& \quad \text{subject to} \quad t^2-(Ax-b)^T(Ax-b) \ge 0 \\
& \equiv \underset{t,x}{\text{minimize}} \quad t \\
& \quad \text{subject to} \quad t-(Ax-b)^T(tI)^{-1}(Ax-b) \ge 0 \\
& \equiv \underset{t,x}{\text{minimize}} \quad t \\
& \quad \text{subject to} \quad \begin{bmatrix} tI & (Ax-b) \\ (Ax-b)^T & t \end{bmatrix} \succeq 0
\end{aligned} \tag{A.2}$$

Here the last step resulted by applying Schur Complement. The final constraint (A.2) can be transformed into standard SDP constraint as:

$$\begin{bmatrix} tI & (Ax-b) \\ (Ax-b)^T & t \end{bmatrix} \succeq 0 \implies \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} t + \sum_{i=1}^{n} \begin{bmatrix} 0 & A_i \\ A_i^T & 0 \end{bmatrix} x_i - \begin{bmatrix} 0 & b \\ b^T & 0 \end{bmatrix} \succeq 0 \tag{A.3}$$

here A_i indicates the ith column of matrix A. The final equivalent semidefinite program in standard form , described in (A.4) can be solved efficiently using any SDP solver such as [2,28].

$$\begin{aligned}
\underset{x}{\text{minimize}}||Ax-b||^2 \quad & \equiv \underset{t,x}{\text{minimize}} \quad t \\
& \quad \text{subject to} \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} t + \sum_{i=1}^{n} \begin{bmatrix} 0 & A_i \\ A_i^T & 0 \end{bmatrix} x_i - \begin{bmatrix} 0 & b \\ b^T & 0 \end{bmatrix} \succeq 0
\end{aligned} \tag{A.4}$$

A.2 Implementing Linear Matrix Inequalities

Suppose we wish to enforce the following linear matrix inequality as and SDP constraint

$$c_1 \begin{bmatrix} x_1 & x_2 \\ x_2 & x_3 \end{bmatrix} + c_2 \begin{bmatrix} x_4 & x_5 \\ x_5 & x_6 \end{bmatrix} \succeq 0 \tag{A.5}$$

such a constraint can be enforced as a standard SDP constraint as follows

$$\begin{bmatrix} c_1 & 0 \\ 0 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 & c_1 \\ c_1 & 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 & 0 \\ 0 & c_1 \end{bmatrix} x_3 + \begin{bmatrix} c_2 & 0 \\ 0 & 0 \end{bmatrix} x_4 + \begin{bmatrix} 0 & c_2 \\ c_2 & 0 \end{bmatrix} x_5 + \begin{bmatrix} 0 & 0 \\ 0 & c_2 \end{bmatrix} x_6 \succeq 0 \tag{A.6}$$

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] http://http://web.mit.edu/zohaib/www/ .

[2] http://sedumi.ie.lehigh.edu/ .

[3] http://www.rle.mit.edu/cpg/research_codes.htm .

[4] R. Achar and M.S. Nakhla. Simulation of high-speed interconnects. *Proceedings of the IEEE*, 89(5):693 –728, may 2001.

[5] A. C. Antoulas, D. C. Sorensen, and S. Gugercin. A survey of model reduction methods for large-scale systems. In *Structured Matrices in Operator Theory, Numerical Analysis, Control Signal and Image Processing, 2001.*

[6] B.N. Bond and L. Daniel. Guaranteed stable projection-based model reduction for indefinite and unstable linear systems. In *Computer-Aided Design, 2008. ICCAD 2008. IEEE/ACM International Conference on*, pages 728 –735, 10-13 2008.

[7] Bradley N. Bond and Luca Daniel. Automated compact dynamical modeling: An enabling tool for analog designers. In *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, pages 415 –420, 13-18 2010.

[8] Stephen Boyd, Laurent El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[9] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[10] Carlos P. Coelho, Joel R. Phillips, and L. Miguel Silveira. A convex programming approach to positive real rational approximation. In *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, pages 245–251, San Jose, CA, November 2001.

[11] L. Daniel, C. S. Ong, S. C. Low, K. H. Lee, and J. K. White. A multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 23(5):678–93, May 2004.

[12] L. Daniel and J. R. Phillips. Model order reduction for strictly passive and causal distributed systems. In *Proc. of the IEEE/ACM Design Automation Conference*, New Orleans, LA, June 2002.

[13] D. Deschrijver and T. Dhaene. Stability and passivity enforcement of parametric macromodels in time and frequency domain. *Microwave Theory and Techniques, IEEE Transactions on*, 56(11):2435 –2441, nov. 2008.

[14] T. El-Moselhy and L. Daniel. Variation-aware interconnect extraction using statistical moment preserving model order reduction. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pages 453 –458, 8-12 2010.

[15] F. Ferranti, L. Knockaert, and T. Dhaene. Guaranteed passive parameterized admittance-based macromodeling. *Advanced Packaging, IEEE Transactions on*, PP(99):1 –1, 2009.

[16] S. Grivet-Talocia. Passivity enforcement via perturbation of hamiltonian matrices. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 51(9):1755–1769, Sept. 2004.

[17] S. Grivet-Talocia and A. Ubolli. A comparative study of passivity enforcement schemes for linear lumped macromodels. *IEEE Trans. on Advanced Packaging*, 31(4), Nov. 2008.

[18] S. Gugercin and A. C. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748 – 766, May 2004.

[19] B. Gustavsen. Fast passivity enforcement for pole-residue models by perturbation of residue matrix eigenvalues. *IEEE Trans. on Power Delivery*, 23(4), Oct. 2008.

[20] B. Gustavsen and A. Semlyen. Rational approximation of frequency domain responses by vector fitting. *IEEE Trans. on Power Delivery*, 14(3), Jul 1999.

[21] S. Lefteriu and A. C. Antoulas. A new approach to modeling multiport systems from frequency-domain data. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 29(1):14 –27, jan. 2010.

[22] Z. Mahmood, B. Bond, T. Moselhy, A. Megretski, and L. Daniel. Passive reduced order modeling of multiport interconnects via semidefinite programming. In *Proc. of the Design, Automation and Test in Europe (DATE)*, Dresden, Germany, March 2010.

[23] Z. Mahmood and L. Daniel. Circuit synthesizable guaranteed passive modeling for multiport structures. In *IEEE Intern. Workshop on Behavioral Modeling and Simulation*, San Jose, California, USA, September 2010.

[24] Sung-Hwan Min and M. Swaminathan. Construction of broadband passive macromodels from frequency data for simulation of distributed interconnect networks. *Electromagnetic Compatibility, IEEE Transactions on*, 46(4):544 – 558, nov. 2004.

[25] Jorge J. Mor. The levenberg-marquardt algorithm: Implementation and theory. In *Numerical Analysis*, Lecture Notes in Mathematics, pages 105–116. Springer Berlin / Heidelberg, 1978.

[26] J. Morsey and A.C. Cangellaris. Prime: passive realization of interconnect models from measured data. In *Electrical Performance of Electronic Packaging, 2001*, pages 47 -50, 2001.

[27] T. Moselhy, Xin Hu, and L. Daniel. pfft in fastmaxwell: A fast impedance extraction solver for 3d conductor structures over substrate. In *Proc. of Design, Automation and Test in Europe Conference, 2007. DATE '07*, pages 1–6, April 2007.

[28] K. C. Toh R. H. Ttnc and M. J. Todd. Solving semidefinite-quadratic-linear programs using sdpt3. *Mathematical Programming*, 95(2):189–217, 2003.

[29] K. C. Sou, A. Megretski, and L. Daniel. A quasi-convex optimization approach to parameterized model order reduction. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 27(3), Mach 2008.

[30] B. Stengel and W.R. Eisenstadt. Linc power amplifier combiner method efficiency optimization. *IEEE Transactions on Vehicular Technology*, 49(1):229 -234, jan 2000.

[31] Jos F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones, 1998.

[32] P. Triverio, S. Grivet-Talocia, and M.S. Nakhla. A parameterized macromodeling strategy with uniform stability test. *Advanced Packaging, IEEE Transactions on*, 32(1):205 -215, feb. 2009.

[33] D. S. Weile, E. Michielssen, Eric Grimme, and K. Gallivan. A method for generating rational interpolant reduced order models of two-parameter linear systems. *Applied Mathematics Letters*, 12:93–102, 1999.

[34] Zhenhai Zhu and J. Phillips. Random sampling of moment graph: A stochastic krylov-reduction algorithm. In *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, pages 1 -6, 16-20 2007.