BROADBAND MACROMODELING TECHNIQUES FOR INTERCONNECTS AND ELECTRONIC PACKAGES

BY

YIDNEKACHEW SINTAYEHU MEKONNEN

B.S., Addis Ababa University, 2000
M.S., University of Illinois at Urbana-Champaign, 2004

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2007

Urbana, Illinois

Doctoral Committee:

Professor José E. Schutt-Ainé, Chair
Professor Andreas C. Cangellaris
Professor Martin D.F. Wong
Assistant Professor Deming Chen

ABSTRACT

As wiring density in high-performance packaging increases, the interconnect geometry becomes nonuniform and the cross-sectional dimensions become smaller. As operating frequency increases, the interconnect and dielectric losses, dispersions, and discontinuities have to be considered in the analysis of an electronic package. These frequency-dependent phenomena of an interconnect are more accurately characterized by measured or simulated frequency response data than by closed-form functions. The transient analysis of the systems described in tabular forms cannot directly be performed using conventional simulators or methods based on order-reduction techniques. Consequently, the most efficient method of analyzing frequency-dependent components is using circuit simulation technique by generating rational approximation in order to express the interconnect governing equations using ordinary differential equations.

In this dissertation, new techniques are proposed to generate an accurate broadband macromodel for passive interconnect and packages. The physical characteristics of the structures such as stability, causality, and passivity are preserved during the model construction. Also, a new method is formulated based on vector-fitting method in z-domain. It has the advantage of faster convergence and better numerical stability compared to the s-domain vector-fitting method. The fast convergence of the method reduces the overall macromodel construction time. These macromodels are then combined with other linear blocks and nonlinear components in a circuit simulation environment.

Dedicated to my Mother
Beletu Demissie (Amiyee)
for her unconditional love

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my research adviser, Professor José E. Schutt-Ainé, for his expert guidance, much needed encouragement and support throughout this research. His patience, support, and confidence have been the driving force of my work.

I would like to thank Professor Andreas Cangellaris, Professor Martin Wong, Professor Deming Chen, and Professor Naresh Shanbhag for serving on my preliminary and final examination committees, and for their valuable time and suggestions during my research.

I would like to thank Dr. Wendemagegnehu T. Beyene of Rambus Inc. for his step-by-step guidance and encouragement along my graduate studies. His motivation in good and difficult times, and his patience and support throughout this work are unforgettable.

I would also like to acknowledge my colleagues in Everitt Laboratory for creating an enjoyable work atmosphere. I would like to acknowledge Dr. Rong Gao for her valuable discussions. I would like to thank Cadence Design Systems for supporting this work financially. Special thanks go to Jilin Tan, C. Kumar, and Feras Al-Hawari at Cadence for their valuable suggestions.

Last, but certainly not least, I express my deepest gratitude and love to my mother, Beletu Demissie, my sister Mahlet Solomon, and my brother Kirubel Solomon for their never-ending support and encouragement throughout the whole graduate school process. I am also grateful to my extended family both here in the U.S. and back home. I believe your patience finally paid off.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

As wiring density in high-performance packaging increases, the interconnect geometry becomes nonuniform and the cross-sectional dimensions become smaller. As operating frequency increases, the interconnect and dielectric losses, dispersions, and discontinuities have to be considered in the analysis of an electronic package. These frequency-dependent phenomena of an interconnect are most accurately characterized by measured or simulated frequency response data than by closed-form functional. For example, skin effect of an interconnect is best characterized by frequency-domain measurements or full-wave electromagnetic simulation. The transient analysis of the systems described in tabular forms cannot directly be performed using conventional simulators such as Simulation Program With Integrated Circuit Emphasis (SPICE), or using methods based on order-reduction techniques, such as Asymptotic Waveform Evaluation (AWE), Complex Frequency Hopping (CFH), and Padé via Lanczos (PVL).

The incorporation of interconnects characterized by frequency-domain data obtained from measurements or full-wave electromagnetic solvers into circuit simulators is complex and computationally expensive. Because nearly all interconnects are driven or terminated by nonlinear devices, the method must combine the time-domain and frequency-domain descriptions [1]. The most straight forward approaches are to calculate the impulse response of the measured network using the inverse fast Fourier transform (IFFT), and to apply the discrete time solution to solve the entire system for a given input waveform.

Such an approach requires that at every simulation time step, the impulse response to be convoluted with the entire computed input waveform. The IFFT used to transform frequency-domain data into time-domain data requires special attention to avoid aliasing. Extrapolation and low-pass filtering of the frequency-domain data are required to reduce the time-domain ripple associated with taking the IFFT. A very large number of samples is required for an adequate representation of the impulse functions, which increases the convolution computation time of the transient response. Although this method gives a reliable solution, it is not fast enough to provide results in an acceptable time interval.

Although order-reduction techniques, such as AWE, CFH, and PVL are efficient methods to analyze linear systems, they are not quite suitable to handle networks characterized by tabular data, as the derivatives at selected expansions cannot accurately be calculated to find the moments. In [2], a hybrid technique is used by first approximating the function by calculating rational functions over partitioned frequency ranges, and then using the derivative of the rational functions to obtain the moments. However, the moments obtained using this method are order-dependent, violating the definition of Taylor series.

An accurate simulation of interconnects requires a transient analysis of nonlinear driver and receiver, and frequency-dependent transmission-line systems. Consequently, the most efficient method of analyzing frequency-dependent component is using circuit simulation technique by generating rational approximation in order to express the interconnect governing equations using ordinary differential equations. Circuit level simulation of interconnects with their nonlinear drivers and receivers can be costly in terms of central processor unit (CPU) time due primarily to the size of the linear interconnect portion of the model. Considering the number of interactions between interconnect and drivers/receivers (ports) will be small compared to the number of interconnect elements,

it is often appropriate to partition the linear interconnect portions from the drivers and receivers. The macromodel of the interconnect can then be modeled accurately using an efficient order reduction technique. These macromodels are then combined with other linear blocks and nonlinear components in a circuit simulation environment.

1.1 Objective

Several methods have been proposed to generate rational representations of a distributed system. These methods include the asymptotic waveform evaluation (AWE) and CFH methods as well as their variants that are based on moment matching at single or multiple points [3]. Lanczos and Arnoldi methods are other reduced order modeling methods that are based on Krylov subspace techniques [4]. Robust rational interpolations that are based on the Hilbert transform of electrical parameters of passive networks or Kramers-Kröning relations between real and imaginary parts of material properties have also been used to generate rational approximation of frequency-dependent parameters [5]. In [6], a rational interpolation approach is used to approximate the transfer function of linear systems characterized by sampled data by using the three most common orthogonal polynomials: Legendre, Chebyshev of the first kind, and the second kind. The ill-conditioned Vandermonde-like interpolation matrix associated with the ordinary power series is avoided by using these orthogonal polynomials. Clenshaw's recurrence algorithm is applied in transforming the coefficients of the orthogonal polynomials to the ordinary power series. However, the efficiency, accuracy, or stability of these approximations techniques has been compromised in an effort to generate high-order approximations of frequency-dependent parameters over a wide frequency range.

In this dissertation, new techniques are proposed to generate an accurate broadband macromodel for passive interconnect and packages. The physical characteristic of the system such as stability, causality, and passivity are preserved during the model construction. Also, a new method is formulated based on vector-fitting method in the z-domain. It has an advantage of faster convergence and better numerical stability compared to the s-domain vector-fitting method. The fast convergence of the method reduces the overall macromodel construction time.

1.2 Organization

In this dissertation, a robust rational approximation technique for macromodel construction is studied. The dissertation develops the methodology for the construction of pole-zero models to represent complex effects of distributed systems or system responses obtained from field solvers and measurements. The models are compatible to conventional simulators and can be readily integrated into time-domain circuit simulators to estimate the response of a system.

In Chapter 2, a review of model-order reduction techniques is presented. Padé approximations using the moment-matching technique and Krylov subspace-based methods, such as Arnoldi and Lanczos methods, are studied.

In Chapter 3, the basic requirement of broadband macromodels are discussed. The stability, causality and passivity properties of a macromodel is presented.

In Chapter 4, a survey of rational macromodel generation methods for passive subnetworks characterized by tabulated data is presented. Some of the common passivity enforcement techniques are discussed. Pole-clustering techniques are discussed to further

reduce the model of the intermediate, large-order system obtained through rational approximation of the original system. The time-domain simulation based on state-space formulation in s-domain is presented.

In Chapter 5, macromodeling technique in s-domain based on vector-fitting method is discussed in detail. The vector-fitting method has emerged as a method of choice in the generation of high-order rational functions over wide frequency ranges [7], [8] to approximate the frequency-dependent parameters of interconnects and packages. The step-by-step implementation of the method is presented. Some of the convergence issues of the traditional vector-fitting method are discussed. A new hybrid method is proposed by combining the rational interpolation method with the vector-fitting method [9]. Order and error estimation of this method are also discussed.

In Chapter 6, a new technique of macromodeling in the z-domain is introduced. This method is based on the vector-fitting in the z-domain (ZDVF). The method is the reformulation of Sanathanan-Koerner (SK) iteration technique. ZDVF has the advantage of faster convergence and better numerical stability compared to the s-domain vector-fitting method. The fast convergence behavior of this method reduces the overall macromodel generation time significantly. The convergence of ZDVF is illustrated by using various examples and by studying the movement of the poles with iteration. Different test cases are simulated to validate the robustness of the method. Also, applying ZDVF method for time-domain system identification of linear system is illustrated.

In Chapter 7, a method is proposed to reduce the order of approximation by extracting delay from the response. The method is based on decomposition of the original linear response into minimum-phase, all-pass response, and delay term.

Finally, the conclusion and discussion of the future work to further improve the proposed methods are presented in Chapter 8.

CHAPTER 2

REVIEW OF MODEL-ORDER REDUCTION TECHNIQUES

2.1 Introduction

In today's electronic technology development, circuit simulators are important tools in the design and testing of electronic systems. These simulators provide an inexpensive and fast means of verifying and optimizing designs without building prototypes. Circuit simulation provides the timing details of the circuit, as well as signal levels. In addition, circuit simulation is advantageous over actual measurement because one can probe any node in a network and observe the response in time or frequency domain without overloading the circuit.

Conventional circuit simulators such as SPICE are general purpose simulators that can provide time-domain transient analysis, dc solutions, frequency-domain ac analysis, sensitivity analysis, and noise and distortion analysis. Such simulators are primarily discrete or lumped-element simulators. The transient simulation of high-speed analog and digital integrated circuits requires the analysis of distributed components. The interconnect structures such as wirebonds, pins, connectors and pads, have to be represented as distributed components to determine the performance of the systems. The interconnect discontinuities, dispersion, conductor and dielectric losses can be accurately represented by transmission line systems rather than by equivalent circuits. The accurate determina-

tion of losses, delay, and noise on interconnects in high-speed integrated circuits requires transmission line models. Model-order reduction techniques have been addressed in the literature to address the interconnect problem [3]-[17]. These techniques are based on the fact that the main behavior of a large linear system is governed mostly by the dominant poles of the system. These dominant poles are poles close to the imaginary axis and significantly influence the time as well as the frequency characteristics of the system.

Methods that are based on Padé synthesis have been applied to efficiently simulate distributed networks [4]. AWE [3]-[13], CFH [14], and PVL [4], [16], [17] have been used successfully to analyze interconnect systems. These methods are based on moment-matching techniques in which, irrespective of the presence of a large number of poles in a system, only the dominant poles are sufficient to accurately characterize a given system

.

2.2 Moment Matching Method

Moment-matching methods depends on the fact that only the dominant poles are sufficient to accurately characterize a system. The description of interconnect systems involves a large number of eigenvalues. However, this requires the determination of all the system poles and residues, which can become computationally extremely expensive for large systems. Instead, the approximate representation and hence an adequate characterization of the system can be achieved by determining its dominant poles only. The moment-matching methods seek to capture the approximate time and frequency behavior by obtaining the dominant poles of the system. These dominant poles are obtained by matching the leading moments of the system.

2.2.1 Moment computation

Consider a single input and single output system and let $H(s)$ is the transfer function and can be represented in a rational function of the form as

$$H(s) = \frac{P(s)}{Q(s)}, \tag{2.1}$$

where $P(s)$ and $Q(s)$ are polynomials in s. The above equation can be written in partial fraction form as

$$H(s) = d + \sum_{i=1}^{N} \frac{c_i}{s - p_i}, \tag{2.2}$$

where p_i and c_i are the poles and the residues, N is the total number of system poles, and d is the direct coupling constant. The time-domain impulse response can be computed in a closed form using the inverse Laplace transform as

$$h(t) = d\delta(t) + \sum_{i=1}^{N} c_i e^{p_i t}. \tag{2.3}$$

For a large linear network such as interconnects, the total number of poles can be very large. Computing all the system poles N will be highly computational time consuming. Model-order reduction techniques address this issue by constructing a reduced-order approximation in terms of the dominant poles.

The linear system can be also expressed in a state-space form as

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t) \tag{2.4}$$

$$y(t) = Cx(t) + Du(t), \tag{2.5}$$

where x is the n-dimensional state vector, u is the excitation sources, y is the output, and D is the direct coupling between the input and output. A, B, C, and D are the state-space matrices. The transfer function for the above state-space representation is given as

$$H(s) = D + C(sI - A)^{-1}B. \tag{2.6}$$

Consider the Taylor series expansion of the transfer function $H(s)$ around $s = 0$,

$$H(s) \approx H(0) + s\frac{H(s)^{(1)}}{1!} + s^2\frac{H(s)^{(2)}}{2!} + \cdots \tag{2.7}$$

where (i) denotes the i^{th} derivative. Applying Equation (2.7) on Equation (2.6), $H(s)$ can be expressed as

$$H(s) = \sum_{i=0}^{\infty} m_i s^i, \quad m_i = \frac{H(0)^{(i)}}{i!}. \tag{2.8}$$

The m_i terms in the transfer function expansion are called the frequency domain moments, and the moments are given by

$$m_i = -C^T A^{-i-1} B. \tag{2.9}$$

From Equation (2.9), successive moments are explicitly calculated as the power of A. With the increase of order, the moment calculation quickly converges to an eigenvector corresponding to an eigenvalue of the largest magnitude of matrix A [18]. As a result,

additional calculation of moments will not add any extra information to the moments. This condition creates ill-conditioning problem associated with Padé approximation.

It has been shown that the moments provide an estimation of delay and rise times. Elmore delay, which approximates the midpoint of the monotonic step response waveform by the mean of the impulse response, essentially matches the first moment of the response.

2.2.2 Moment matching

After the moments are computed, the next step is to match the moments of the original system to those of a reduced-order approximate system. If the original system moments are taken as the first $p+q$ terms of the Maclaurin series and equated to the reduced-order system's rational transfer function with unknown coefficients as

$$m_0 + m_1 s + m_2 s^2 + \cdots + m_{p+q} s^{p+q} = \frac{a_0 + a_1 s + a_2 s^2 + \cdots + a_p s^p}{1 + b_1 s + b_2 s^2 + \cdots + b_q s^q}, \tag{2.10}$$

then the unknown coefficients in the approximate rational transfer function can be found from

$$a_0 + a_1 s + a_2 s^2 + \cdots + a_p s^p = (m_0 + m_1 s + m_2 s^2 + \cdots + m_{p+q} s^{p+q})(1 + b_1 s + b_2 s^2 + \cdots + b_q s^q). \tag{2.11}$$

Matching equal powers of s up to $p+q$, the following set of equations is obtained:

$$\begin{aligned}
a_0 &= m_0 \\
a_1 &= m_1 + m_0 b_1 \\
a_2 &= m_2 + m_1 b_1 + m_0 b_2 \\
&\vdots \\
a_p &= m_p + m_{p-1} b_1 + m_{p-2} b_2 + \cdots + m_1 b_{p-1} + m_0 b_p \\
0 &= m_{p+1} + m_p b_1 + m_{p-1} b_2 + \cdots + m_{p-q+2} b_{q-1} + m_{p-q+1} b_q \\
0 &= m_{p+2} + m_{p+1} b_1 + m_p b_2 + \cdots + m_{p-q+3} b_{q-1} + m_{p-q+2} b_q \\
&\vdots \\
0 &= m_{p+q} + m_{p+q-1} b_1 + m_{p+q-2} b_2 + \cdots + m_{p+1} b_{q-1} + m_p b_q. \qquad (2.12)
\end{aligned}$$

The solution to the last q equations, rewritten in matrix form for convenience in Equation (2.13), will give the denominator polynomial coefficients:

$$\begin{bmatrix} m_p & m_{p-1} & \cdots & m_{p-q+1} \\ m_{p+1} & m_p & \cdots & m_{p-q+2} \\ \vdots & \vdots & \ddots & \vdots \\ m_{p+q-1} & m_{p+q-2} & \cdots & m_p \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_q \end{bmatrix} = - \begin{bmatrix} m_{p+1} \\ m_{p+2} \\ \vdots \\ m_{p+q} \end{bmatrix} \qquad (2.13)$$

The roots of the denominator polynomial are found from the polynomial

$$1 + b_1 p^{-1} + b_2 p^{-2} + \cdots + b_{q-1} p^{q-1} + b_q p^{-q} = 0, \qquad (2.14)$$

and hence the roots are the reciprocals of the desired poles. After the coefficients b_i are found, the numerator polynomial coefficients can be solved from the first p equations in Equation (2.12). However, the residues are determined from the partial fraction expansion of the q poles in Equation (2.15):

$$H(s) = d + \sum_{i=1}^{q} \frac{c_i}{s - p_i}. \tag{2.15}$$

Note that the moments are the Maclaurin series expansion coefficients of the transfer function, thus the relation between the moments and the pole-residue representation from which the residues are computed, is given by

$$m_r = \frac{\partial^r}{\partial s^r} \left[d + \sum_{i=1}^{q} \frac{c_i}{s - p_i} \right] \Bigg|_{s=0} \tag{2.16}$$

The residues are found as a solution to this equation and can be given in matrix form as

$$\begin{bmatrix} -1 & p_1^{-1} & \cdots & p_{q-1}^{-1} & p_q^{-1} \\ 0 & p_1^{-2} & \cdots & p_{q-1}^{-2} & p_q^{-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & p_1^{-q} & \cdots & p_{q-1}^{-q} & p_q^{-q} \\ 0 & p_1^{-q-1} & \cdots & p_{q-1}^{-q-1} & p_q^{-q-1} \end{bmatrix} \begin{bmatrix} d \\ c_1 \\ \vdots \\ c_{q-1} \\ c_q \end{bmatrix} = - \begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_{q-1} \\ m_q \end{bmatrix} \tag{2.17}$$

The procedure described above to determine the poles and the residues is known as Padé approximation [3]-[17].

2.3 Complex Frequency Hopping

CFH extends the process of moment matching to multiple expansion points in the complex plane near or on the imaginary axis using a binary search algorithm [14], [15]. With a minimized number of frequency points expansions, enough information is obtained to enable the generation of an approximate transfer function that matches the original function up to a predefined highest frequency of interest. Using the information from all the expansion points, CFH extracts a dominant poles. In addition, CFH provides an error criterion for the selection of accurate poles and transfer functions.

Padé approximation is accurate only near the point of expansion and its accuracy decreases as one move away from the point of expansion. In order to validate the accuracy of such an approximation, at least two expansion points are necessary. Accuracies of these two expansions can be verified by matching the poles generated at these two hopes. Alternatively, the two hops can be verified for their accuracy by comparing the value of the transfer functions due to both these hops at a point intermediate to them. CFH relies on a binary search algorithm to determine the expansion points and to minimize the number of expansions.

2.4 Krylov Subspace-Based Methods

Methods based on Krylov projectors can be used to generate reduced-order models of large systems. The Krylov subspace methods are more stable than moment-matching methods because the solution of an ill-conditioned moment matrix is not needed to calculate the dominant poles. Two variants of these methods, Arnoldi and Lanczos processes, are discussed here. Both methods construct the reduced-order models in a computation-

ally efficient manner by recursively operating on the original system. The methods are applied successfully for time- and frequency-response approximations of large-linear networks. The Arnoldi and Lanczos methods compare favorably to the moment-matching technique in their numerical stability and accuracy.

2.4.1 Krylov subspace method

The numerical calculations of all eigenvalues of a matrix, even for a matrix of moderate size, are prohibitively expensive. Originally, the method of Krylov was used to compute the coefficients of the characteristic polynomial of a matrix [19]. The Krylov method allows us to estimate some or all of the eigenvalues of a matrix without solving a large eigenvalue problem. All Krylov subspace methods depend on a sequence of vectors called the Krylov sequence, which is generated by a recursive formula.

Let the characteristic equation of an $n \times n$ square matrix A be given as

$$\lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_0 = 0. \tag{2.18}$$

It is known from the Cayley-Hamilton theorem, that every square matrix satisfies its own characteristic equation,

$$A^n + p_{n-1}A^{n-1} + \cdots + p_0 = 0. \tag{2.19}$$

Assuming a nonzero vector b_0 of size $n \times 1$,

$$(A^n + p_{n-1}A^{n-1} + \cdots + p_0)b_0 = 0, \tag{2.20}$$

is valid. Then, the Krylov vectors b_n defined by recursive formula

$$b_n = A^n b_0 = A b_{n-1}, \quad n = 0, 1, 2, \cdots \tag{2.21}$$

will satisfy the equation

$$A^n b_0 + p_{n-1} A^{n-1} b_0 + \cdots + p_0 b_0 = b_n + p_{n-1} b_{n-1} + \cdots + p_0 b_0 = 0. \tag{2.22}$$

This equation may be rewritten as

$$-b_n = p_{n-1} b_{n-1} + \cdots + p_0 b_0, \tag{2.23}$$

which gives a nonsingular set of linear equations from which the characteristic equation coefficients of A, p_i, $i = 0, \cdots, n-1$, can be solved. The vector space spanned by vectors $b_0, \cdots, b_{n-1}$ is called the Krylov vector space $\mathbf{K}$ and will be denoted as

$$\mathbf{K}(A, b_0, n) = span\{b_0, Ab_0, A^2 b_0, \cdots, A^{n-1} b_0\} = span\{b_0, b_1, b_2, \cdots, b_{n-1}\} \tag{2.24}$$

In the following sections, the two most common Krylov subspace based methods, namely, the methods of Arnoldi and Lanczos, are discussed.

2.4.2 Arnoldi method

To overcome the ill-conditioning problem of Padé approximation, Arnoldi method based on Krylov subspace projection was introduced. The method is based on the orthogonal projection of a system onto a subspace spanned by $\mathbf{K}$. In other words, any

vector that is a linear combination of the moments of the system, can be expressed as a linear combination of the columns of $\mathbf{K}$. The matrix generated using this subspace by itself is likely to be ill-conditioned since the columns are formed based on the sequence $A^i b_0$, which quickly converges to the eigenvector of the largest eigenvalue.

To alleviate ill-conditioning problem, the Arnoldi method generates the Krylov sequence of vector b_i using the Gram-Schmidt process. The vectors b_i are mutually orthonormal and have the property as in Equation (2.24). The method results a reduced upper Hessenberg matrix H and a set of Krylov vectors such that

$$\begin{aligned} AB &= BH \\ B^T B &= I \end{aligned} \tag{2.25}$$

where B is the generated orthonormal basis.

In the general case, given a starting nonzero vector b_0,

$$k_n b_n = A b_{n-1} - \sum_{i=0}^{n-1} h_{i,n-1} b_i = p_n. \tag{2.26}$$

From the orthogonality

$$h_{i,n-1} = \frac{b_i^T A b_{n-1}}{b_i^T b_i} = b_i^T A b_{n-1} \tag{2.27}$$

and the normalization factor is specified as $k_n = \|p_n\|_2$. Then the next Krylov vector is found from $b_n = b_{n-1}/k_n$. The eigenvalues of H_n can be used as approximations for the eigenvalues of A. If A is symmetric, then H_n is a tridiagonal matrix. In this case, the Arnoldi method reduces to the Lanczos method described in the next sections.

2.4.3 Lanczos method

The Lanczos method reduces an arbitrary large nonsymmetric matrix A to a tridiagonal matrix T_k using two biorthogonal Krylov spaces whose columns are recursively filled using the matrix A and two initial vectors. Given two nonzero initial vectors, the generated biorthogonal basis vectors

$$\begin{aligned} V_n &= (v_0, v_1, \cdots, v_{n-1}) \\ W_n &= (w_0, w_1, \cdots, w_{n-1}), \end{aligned} \tag{2.28}$$

satisfy $W_n^H V_n = I$, and they are related to A-invariant and A^H-invariant Krylov subspace, $\mathbf{K}$, in that

$$\begin{aligned} \mathbf{K}(A, v_0, n) &= span(V_n) = span\{v_0, Av_0, A^2v_0, \cdots, A^{n-1}v_0\} \\ \mathbf{K}(A, w_0, n) &= span(W_n) = span\{w_0, A^Hw_0, \cdots, (A^H)^{n-1}w_0\} \end{aligned} \tag{2.29}$$

The V_n and W_n are constructed based on three-term recurrences given by

$$\begin{aligned} v_{n+1} &= Av_n - \alpha_n v_n - \beta_n v_{n-1} \\ w_{n+1} &= A^Hw_n - \alpha_n w_n - \gamma_n w_{n-1} \end{aligned} \tag{2.30}$$

where

$$\begin{aligned} \alpha_n &= \frac{w_n^T A v_n}{\delta_n} \\ \beta_n &= \frac{\delta_n}{\delta_{n-1}} \eta_n \\ \gamma_n &= \frac{\delta_n}{\delta_{n-1}} \rho_n \end{aligned} \tag{2.31}$$

where $\delta_n = w_n^T w_n$ and $\eta_{n+1} = \|v_n\|_2$, $\rho_{n+1} = \|w_n\|_2$. And α_n, β_n and γ_n are the diagonal, upper, and lower off-diagonal elements of the reduced tridiagonal matrix T, respectively. The results V, W, and T matrices satisfy

$$\begin{aligned} AV &= W^H T \\ W^H V &= I. \end{aligned} \tag{2.32}$$

The best $n \times n$ approximation to A that can be obtained using the information from the two Krylov subspace is matrix T_n. These two bases can be built only with three-term recurrences, thus requiring only inner-products and matrix-vector multiplication. The methods also take advantage of the sparsity of the A matrix to generate the reduced triangular matrix quite efficiently. The eigenvalues of T_n can be used as approximations of the eigenvalues of A. Typically, the spectrum of T_n offers good approximations to some of the eigenvalues of A after relatively few iterations.

2.5 Conclusion

Methods based on Padé synthesis, such as AWE, CFH, and PVL techniques are the most efficient analysis methods for system with a large number of passive elements. These

techniques give a two- to three- order of saving in simulation time over the traditional methods when applied to a large system.

Both moment-matching and Krylov subspace methods have shown computationally efficient. The Krylov subspace-based methods, Arnoldi and Lanczos methods, compare favorably to the moment-matching methods when analyzing linear networks. The moment-matching method is direct and more efficient than the Krylov subspace-based method when analyzing distributed systems. Although the methods based on Padé approximations reduce the simulation time for interconnect networks, the stability of the reduced model still remains and issue of concern. In the next chapter, a guaranteed stable reduced-order macromodeling technique is discussed.

CHAPTER 3

BROADBAND MACROMODELS

3.1 Introduction

As the complexity of interconnects and packages increases and the rise and fall time of the signal decreases, the previously neglected electromagnetic effects of distributed passive devices, which lead to signal and power integrity issuses, become important factors in determining the performance of the system. The electromagnetic behavior of interconnects and passive devices can be extracted using an electromagnetic simulation or from high-frequency measurements. The frequency response of interconnect system can be represented by scattering, admittance, or impedance parameters. The behavior of the passive system can be represented as a black box at the input-output ports. This kind of port representation is known as macromodel.

Most of macromodel construction methods are based on network partitioning, robust rational approximation and recursive convolution to generate macromodels that can be used in time- and frequency-domain simulations. First, the network is partitioned into subnetworks. Then, accurate rational approximation method are applied for complex systems to get the poles and residues. Then, the pole-residue level representations of lumped-model based and measured subnetworks are combined to form a global network. The pole-residue models are incorporated as submatrices into the system matrix of the global network using a recursive convolution formula, and then the entire system, in-

cluding the nonlinear devices, is solved in the time domain. Thus, the size of the overall circuit to be simulated is reduced. These methods do not require a numerical transform treatment or the smoothing and band-limiting filtering of a large number of points, which is often required in conventional methods.

In order to obtain an accurate rational approximation over a wide frequency range, frequency normalization and shift are used to optimally condition the Vandermonde-like approximation matrices and the Householder orthogonal triangularization to solve them. By utilizing the analytic properties of the network functions, the approximation algorithm efficiently generates stable rational functions for high-order systems. The method can be integrated with reduced-order modeling techniques such as AWE, CFH, and PVL, or conventional simulators such as SPICE for transient simulation of high-speed interconnect networks.

Based on application, macromodels can be categorized as scalable macromodels for implementing design libraries, passive macromodels for time-domain simulation of circuits, broadband macromodels for capturing the dynamic response of structures over a wide frequency range, and non-linear macromodels for the representation of active devices. In this dissertation, passive and broadband macromodels are discussed in detail.

For proper transient time-domain analysis, a macromodel needs to satisfy three conditions:

(1) The macromodel has to have real coefficients in its rational approximation function.

(2) The macromodel must have stable poles.

(3) The macromodel must be causal and passive.

3.2 Real Coefficients

A macromodel needs to have a real coefficients of rational function. Since any time-domain response $h(t)$ is real, constraints need to be enforced on the coefficients of the rational function. If the Fourier transform of $h(t)$ is defined as $H(s)$, the Fourier transform of $h^*(t)$ is $H^*(-s)$, where $*$ is the complex conjugate operator. Since $h(t)$ is real, $H(s) = H^*(-s)$. For proper time-domain simulation, the rational function therefore has to satisfy the constraint $H(s) = H^*(-s)$, which requires real coefficients for the rational function of the macromodel [5], [6].

3.3 Stability

Stability is one characteristic that the macromodel needs to satisfy. A stable system means that the system has bounded output for all bounded input. The conditions for stability are $\int_{-\infty}^{\infty} |h(t)|dt < \infty$ for continuous systems and $\sum_{-\infty}^{\infty} |h[k]| < \infty$ for discrete systems, where $h(t)$ or $h[k]$ is the time-domain response of the system [20]. The rational function representing a stable system needs to satisfy the following stability criteria:

(1) All poles of the rational function must be in the left-hand side of the s-plane for continuous system or inside the unit-circle for discrete systems to ensure the stability of the macromodel.

(2) The rational function also does not contain multiple poles along the imaginary axis of the s-plane or on the unit-circle in the z-plane.

(3) The difference between the order of the numerator and the denominator of the rational function does not exceed unity.

3.4 Causality

A system is causal if the output at time $n = n_0$ depends only on the inputs at $n \leq n_0$. This implies that if $x_1[n] = x_2[n]$ for $n \leq n_0$ are inputs for the system, then the corresponding outputs $y_1[n] = y_2[n]$ for $n \leq n_0$. This type of system is know as nonanticipative. The causality criterion requires that the system impulse response $h[n]$ is null for time $n < 0$ [20]. Since the system responses in frequency- and time-domain are related by the Laplace transform, the macromodel $H(s)$, which approximate the real system in the frequency domain, is also subject to the causality requirements. As a result of the causality, the real and imaginary part of the frequency domain response satisfy the Hilbert transform. If the causality condition is violated, a spurious time-domain waveform can result at the output prior to the finite travel time of the signal. As the rise and fall times of the signal decrease, the electrical length of interconnects becomes comparable to the wavelength at the maximum operating frequency of the device. Therefore, delay and causality become important during time-domain simulation since spurious glitches prior to signal arrival can cause the false transition of the circuit.

Any linear time-domain sequence $h[n]$ can be expressed as the sum of an even sequence and an odd sequence:

$$h[n] = h_e[n] + h_o[n] \tag{3.1}$$

where $h_e[n] = \frac{1}{2}[h[n] + h[-n]]$ and $h_o[n] = \frac{1}{2}[h[n] - h[-n]]$ are the even and odd sequences $h[n]$, respectively. The above equations can be applied to an arbitrary sequence, whether causal or real or neither. However, if $h[n]$ is causal, then it is possible to recover $h[n]$ from $h_e[n]$ or to recover $h[n]$ for $n \neq 0$ from $h_o[n]$. Thus, $h[n]$ can be expressed by only

the even or odd sequence

$$\begin{aligned} h[n] &= 2h_e[n]u[n] - h_e[0]\delta[n] \\ h[n] &= 2h_o[n]u[n] + h_o[0]\delta[n], \end{aligned} \tag{3.2}$$

where $u[n]$ and $\delta[n]$ are the unit step and unit impulse sequence. Note that $h[n]$ is completely determined by $h_e[n]$. On the other hand, if $h_o[0] = 0$, then $h[n]$ can be recovered from $h_o[n]$ only for $n \neq 0$. For the stable system, its Fourier transform exists and is defined as

$$H(j\omega) = H_R(j\omega) + jH_I(j\omega), \tag{3.3}$$

where $H(j\omega)$ is the system response in the frequency domain, $H_R(j\omega)$and $H_I(j\omega)$ are the corresponding real and imaginary parts of the frequency response. When the system response is real, $H_R(j\omega)$ is the Fourier transform of $h_e[n]$ and $H_I(j\omega)$ is the Fourier transform of $h_o[n]$ [20].

For causal systems, $h[-n]$ is zero, and the system $h[n]$ can be completely determined by $h_e[n]$. Similarly, $H_R(j\omega)$ will determine the frequency response $H(j\omega)$. $H_R(j\omega)$ and $H_I(j\omega)$ are related by Hilbert relationships as

$$H_R(j\omega) = h[0] + \frac{1}{2\pi} P \int_{-\pi}^{\pi} H_I(j\theta) cot\left(\frac{\omega - \theta}{2}\right) d\theta \tag{3.4}$$

$$H_I(j\omega) = -\frac{1}{2\pi} P \int_{-\pi}^{\pi} H_R(j\theta) cot\left(\frac{\omega - \theta}{2}\right) d\theta, \tag{3.5}$$

where P is the Cauchy principal value.

3.5 Passivity

The passivity condition requires that a passive circuit does not create energy. Since interconnects networks are passive in nature, it is necessary that the macromodel representing the behavior must be passive. Stable but nonpassive models may lead to unstable systems when connected to other passive systems leading to oscillation in the transient response. Passive macromodels, when terminated with any arbitrary passive load, always guarantee the stability of the overall network. The passivity conditions for a multiport macromodel $H(s)$ are

- $H(s^*) = H^*(s)$ for all s, where $*$ is the complex-conjugate operator.
- $H(s)$ is a positive real matrix. Thus, the product $z^* \left[H^T(s^*) + H(s)\right] z$ for all s with $\Re(s) > 0$ and any arbitrary vector z for admittance and impedance response is positive real. For scattering parameter, $z^* \left[I - H^T(s^*)H(s)\right] z$ must be positive real.

These conditions translate into the following constraints:

(1) $H(s)$ does not contain poles on the right half of the s-plane.

(2) $H(s)$ does not have multiple poles on the imaginary axis of the s-plane.

(3) The coefficients of $H(s)$ are real.

(4) The real part of $H(s)$ must be positive semi-definite for all frequencies. This implies that the eigenvalues of the real part of $H(s)$ are positive or zero for all frequencies.

The first and the second constraints are included as part of stability condition while the third condition is equivalent to simply satisfying the real coefficients of the system.

Therefore, the passivity condition needs to ensure real coefficients and satisfy the stability condition. Furthermore, all the eigenvalues of the real part of the rational function matrix must be semipositive. The next important step is how to satisfy these conditions and construct a passive and stable macromodel. The constraint for real coefficients can be satisfied by construction. In the construction of the passive macromodel, if all poles and residues are constructed in such a way that they appear as a real or complex-conjugate form, the whole rational function matrix will have real coefficients. Therefore, poles and residues must appear as real or as conjugate pairs. The stability conditions can be satisfied by ensuring the poles of the rational function matrix lie on the left half of the complex s-plane and eradicating those poles that do not satisfy this condition.

The last passivity constraint must be satisfied and guaranteed over the infinite frequency bandwidth. Therefore, the last constraint should be developed independent of frequency bandwidth. Since poles are characteristic frequencies of the system, common poles should be used to represent the multiport passive system when approximating by rational function matrix $H(s)$. Different approaches are available to enforce passivity. Some of these approaches are frequency range dependent. In [21], an eigenvalue approach has been discussed, which enforces passivity of the macromodel by directly compensating the poles and residues of the rational function using linearization and constrained minimization through quadratic programming. However, this method that is based on searching the frequency band of violation, is computationally expensive and uses discrete, band-limited frequency samples for enforcing passivity of the macromodel. Hence, the generated macromodel can still violate passivity over a continuous frequency and outside the band-limited frequency response since the macromodel is tested at discrete frequency samples. Recently, the rational function matrix $H(s)$ without common poles has been

used for representing the passive macromodel for multiport circuits, and a compensation method has been proposed [22]. The compensation method can detect the frequency band violating passivity using the associated Hamiltonian matrix. In [23] and [24], a different approach is used to assure passive macromodel by constraining the residues for a fixed set of poles.

A detail review of passivity enforcement techniques will be discussed next. The next chapter review some of the existing macromodel construction techniques and passivity enforcement techniques of sampled frequency data.

CHAPTER 4

REVIEW OF MACROMODELING TECHNIQUES OF TABULATED DATA

A macromodel representation of a large, lumped, and distributed network can be obtained by finding an optimum approximation, in some sense, of the original system without the constraints of matching the eigenvalues or the moments. The method involves a constrained curve-fitting that minimizes the error between the original system and the approximation. The approach is more general and can give better approximations than a single or multipoint moment-matching methods. An efficient method of incorporating frequency response data into the circuit simulator is by representing the data by rational functions. The computational procedure to determine a rational approximation requires the solutions of ill-conditioned systems. The accurate calculation of poles and zeros often involves costly optimization.

4.1 Rational Function Interpolation

One of the most common ways of representing frequency-domain tabulated data of frequency responses is approximating it by rational functions in complex variable, s. A network function $H(s)$ of a linear system can be approximated by a rational function that interpolates the given frequency response data at a given frequency points. Rational function approximation in pole-residue form is an auxiliary step in order to make the

analytical calculation of inverse Laplace or Fourier transform possible. Suppose the network function is approximated by rational function of $H(s)$

$$H(s) = \frac{a_0 + a_1 s^1 + a_2 s^2 + \cdots + a_M s^M}{1 + b_1 s^1 + b_2 s^2 + \cdots + b_N s^N}, \tag{4.1}$$

where b_0 normalized to unity. Equation (4.1) contains $n = M + N + 1$ free coefficients, hence, at most n independent parameters. The coefficients are determined so that the approximating function, evaluated at the same frequency points, gives close approximations to the function $H(s)$. Equation (4.1) is a nonlinear function of the unknown coefficients, but can converted into standard linear least-square formulation by multiplying both sides by denominator polynomial:

$$a_0 + a_1 s^1 + a_2 s^2 + \cdots + a_M s^M = H(s)(1 + b_1 s^1 + b_2 s^2 + \cdots + b_N s^N). \tag{4.2}$$

When the above equation is written at each frequency points, it results the following matrix equation:

$$\begin{bmatrix} 1 & s_1 & \cdots & s_1^M & -s_1 H(s_1) & -s_1^2 H(s_1) & \cdots & -s_1^N H(s_1) \\ 1 & s_2 & \cdots & s_2^M & -s_2 H(s_2) & -s_2^2 H(s_2) & \cdots & -s_2^N H(s_2) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 1 & s_k & \cdots & s_k^M & -s_k H(s_k) & -s_k^2 H(s_k) & \cdots & -s_k^N H(s_k) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_M \\ b_1 \\ \vdots \\ b_N \end{bmatrix} = \begin{bmatrix} H(s_1) \\ H(s_2) \\ \vdots \\ \vdots \\ H(s_k) \end{bmatrix}. \tag{4.3}$$

However, the resulting formulation is badly conditioned and nearly singular because of the large difference between the maximum and minimum frequencies raised to the order of approximation. The approximation method can be made more efficient and accurate by utilizing the special properties of the network functions. The coefficients of the rational function must be real, and all roots of the denominator must have negative or zero real parts. In addition, network functions are analytic functions of a complex variable; hence, their real and imaginary parts are related by Cauchy-Riemann equations. In an electrical network, some constraints remain between the frequency variations of the resistance, reactance, conductance and susceptance, just as in the Kramers-Kröning relations between real and imaginary part of permittivity. The location of poles of a passive system are constrained in the left-half of complex plane due to the analytical property of the network function. Thus, the response of a passive network can only decay in time from any transient state.

To enforce this property, the real part of the sampled frequency data is approximated by the rational function, $H(s)$, to evaluate the poles of the system [5]. The real part of a network function in Equation (4.1) can be specified as the even part of $H(s)$ replacing $-s^2$ with ω^2. The real part of the original function is fitted with the real rational polynomial function of squared variable as,

$$\Re(H(s)) = \frac{a_0 + a_1 s^2 + a_2 s^4 + \cdots + a_M s^{2M}}{b_0 + b_1 s^2 + b_2 s^4 + \cdots + b_N s^{2N}}. \tag{4.4}$$

The poles of the even function of $H(s)$ are those of both $H(s)$ and $H(-s)$, those belonging to $H(s)$ lie in the left half-plane in s-domain. Thus the denominator coefficients of $H(s)$ in Equation (4.1) can be obtained from Equation (4.4). The following system equations

results from matching the real parts of the original function with Equation (4.4) at the set of frequencies, as in

$$\begin{bmatrix} 1 & \omega_1^2 & \cdots & \omega_1^{2M} & -\omega_1^2 H_1^r & -\omega_1^4 H_1^r & \cdots & -\omega_1^{2N} H_1^r \\ 1 & \omega_2^2 & \cdots & \omega_2^{2M} & -\omega_2^2 H_2^r & -\omega_2^4 H_2^r & \cdots & -\omega_2^{2N} H_2^r \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega_k^2 & \cdots & \omega_k^{2M} & -\omega_k^2 H_k^r & -\omega_k^4 H_k^r & \cdots & -\omega_k^{2N} H_k^r \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_M \\ b_1 \\ \vdots \\ b_N \end{bmatrix} = \begin{bmatrix} H_1^r \\ H_2^r \\ \vdots \\ \vdots \\ H_k^r \end{bmatrix} \tag{4.5}$$

where the subscript "r" indicates the real part of the complex data given $H(s)$, and k is the number of frequency data points given. Once the coefficients of the denominator polynomial are obtained from Equation (4.5), the roots of the denominator are calculated from

$$1 + b_1\omega^2 + b_2\omega^4 + \cdots + b_N\omega^{2N} = 0. \tag{4.6}$$

The poles are assumed to be distinct. Repeated poles are not likely because of the nature of the problem and the fact that the poles are computed numerically. If repeated poles are obtained, they can be modified slightly to make them distinct. Factoring the denominator and taking only the left half-plane roots, the partial fraction expansion of the transfer function is constructed. Stability of the poles is guaranteed, because the polynomial roots are determined in terms of the square poles. The purely imaginary single poles on the imaginary axis are rejected as spurious because the rational function has double roots.

Once the system poles are identified, the unknown residues and the constant term can be directly calculated from

$$H(s) = \sum_{n=1}^{N} \frac{c_n}{s - p_n} + d + sh. \tag{4.7}$$

Now, the system poles p_n are already determined from the pole identification process. The only remaining unknowns are c_n, d, and h. The system poles are either real or complex-conjugate form. Therefore, the corresponding residue values must be the same form as the system poles. In matrix form, it becomes

$$\begin{bmatrix} \frac{1}{s_1-p_1} & \cdots & \frac{1}{s_1-p_N} & 1 & s_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{1}{s_k-p_1} & \cdots & \frac{1}{s_k-p_N} & 1 & s_k \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \\ d \\ h \end{bmatrix} = \begin{bmatrix} H(s_1) \\ \vdots \\ \\ \vdots \\ H(s_k) \end{bmatrix}. \tag{4.8}$$

Solving Equation (4.8), all the unknown residues and constant terms are obtained. The residues obtained must guarantee complex-conjugate form for complex-conjugate poles. However, since Equation (4.5) is in ordinary power series basis, it becomes seriously ill-conditioned or even singular for a wide frequency range and high order approximation [5]. For a higher frequencies, the approximation will not be able to capture the original frequency response very well due to the ordinary power series basis have a very large dynamic range [25], [26] . In the following section, an alternative approach is provided using orthogonal polynomials to solve the ill-conditioning problem of rational interpolation.

4.2 Orthogonal Polynomial Approach

In the previous section, rational interpolation method was discussed to determine the rational function which approximate the original frequency response data. Instead of working on the rational function directly, the even function, which corresponds to the real part of the frequency response is applied. However, direct formulation by ordinary power series will result in an ill-conditioned matrix equation. Equations (4.5) and (4.8) are Vandermonde-like matrices, which become ill-conditioned or even singular over a wide frequency range or high order approximation.

In [5], several techniques have been suggested to improve the accuracy of the solution for this ill-conditioned system equations. Also, Chebyshev of the first kind, a special case of the orthogonal polynomials, is employed in [23], [26]-[28] to solve ill-conditioned problem. Since it changes the matrix itself, accurate approximation can be obtained.

4.2.1 Orthogonal polynomials

To overcome the aforementioned problem, orthogonal polynomials are used in the approximation. The Legendre and Cheybshev of the first and second kind polynomials are suitable for this problem because they have remarkable properties in the interval $[-1, 1]$, such as easy generation from recurrence, orthogonality, and distinct roots in the interval $[-1, 1]$. These polynomials are special cases of the more general Jacobi or Gegenbauer polynomials [29]. Jacobi polynomials are defined as

$$P_n^{(\alpha,\beta)}(x) = 2^{-n} \sum_{k=0}^{n} \binom{n+\alpha}{k} \binom{n+\beta}{n-k} (x-1)^{n-k}(x+1)^k. \tag{4.9}$$

The Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$ are generated by applying the orthogonalization step of the Gram-Schmidt process to the ordinary power series basis $\{1, x, x^2, x^3, \cdots\}$ of $P[-1, 1]$ with respect to the weight function given by the following continuous beta distribution of the interval $[-1, 1]$:

$$w^{(\alpha,\beta)}(x) = (1-x)^{\alpha}(1+x)^{\beta}, \quad for \;\; \alpha > -1, \beta > -1. \tag{4.10}$$

The Jacobi polynomials reduce to Gegenbauer polynomials, $C_n^{\lambda}(x)$ when $\alpha = \beta = \lambda - 1/2$, in Equation (4.9) and (4.10). The Jacobi also reduces to Chebyshev of the first and second kinds as well as Lengendre series when the parameters α and β are set to $-1/2$, $1/2$, and 0, respectively.

For simplicity, the interval $[\omega_{min}, \omega_{max}]$ can be transformed into the interval $[-1, 1]$ using the one-to-one continuous mapping given by

$$\varpi = \frac{2\,(\omega - \omega_{min})}{\omega_{max} - \omega_{min}} - 1 \tag{4.11}$$

where ω_{min} and ω_{max} are the lowest and highest frequencies, and ϖ is the normalized frequency. Table 4.1 shows the important properties of the special orthogonal functions [25] where $T_n(x), U_n(x)$, and $L_n(x)$ are defined as:

$$\begin{aligned} T_n(x) &= \cos(n \cos^{-1} x) \\ U_n(x) &= \frac{\sin((n+1)\cos^{-1} x)}{\sin(\cos^{-1} x)} \\ L_n(x) &= 2^{-n} \sum_{k=0}^{n/2} (-1)^k \binom{n}{k} \binom{2n-2k}{n} x^{n-2k}. \end{aligned} \tag{4.12}$$

Table 4.1 Special orthogonal polynomials

Function	Expression	Weight	Recurrence Relation
Chebyshev 1^{st}kind	$T_n(x)$	$(1-x^2)^{-1/2}$	$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$ $T_0(x) = 1, T_1(x) = x$
Chebyshev 2^{nd}kind	$U_n(x)$	$(1-x^2)^{1/2}$	$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x)$ $U_0(x) = 1, U_1(x) = 2x$
Legendre	$L_n(x)$	1	$(n+1)L_{n+1}(x) = (2n+1)xL_n(x) - nL_{n-1}(x)$ $L_0(x) = 1, L_1(x) = x$

For higher order, the power series become very similar or parallel over the most of interval. On the other hand, the terms of Legendre and Chebyshev of first and second kind show quite different shapes and are not parallel over the entire intervals. They also have small dynamic range of the entire interval. Therefore, the orthogonal polynomials are well suited for higher order interpolations.

Therefore, Equation (4.5), which has ordinary power series basis, can be rewritten using one of the orthogonal polynomials, $F_k(x)$ as in Equation (4.13). The frequency must be normalized to the desired range:

$$\begin{bmatrix} 1 & F_2(\varpi_1) & \cdots & F_{2M}(\varpi_1) & -F_2(\varpi_1)H_1^r & \cdots & -F_{2N}(\varpi_1)H_1^r \\ 1 & F_2(\varpi_2) & \cdots & F_{2M}(\varpi_2) & -F_2(\varpi_2)H_2^r & \cdots & -F_{2N}(\varpi_2)H_2^r \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 1 & F_2(\varpi_k) & \cdots & F_{2M}(\varpi_k) & -F_2(\varpi_k)H_k^r & \cdots & -F_{2N}(\varpi_k)H_k^r \end{bmatrix} \begin{bmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \vdots \\ \hat{a}_M \\ \hat{b}_1 \\ \vdots \\ \hat{b}_N \end{bmatrix} = \begin{bmatrix} H_1^r \\ H_2^r \\ \vdots \\ \vdots \\ H_k^r \end{bmatrix}. \tag{4.13}$$

Equation (4.13) results a well-conditioned matrix and can be solved efficiently. Once the coefficients $\hat{a}_i$ and $\hat{b}_i$ of the orthogonal polynomial are determined, a Clenshaw's recurrence method is used to determine the coefficients of the corresponding power series. A detail formulation of Clenshaw's recurrence method can be found in [25], [30]. Then the poles can be extracted from the power series denominator polynomials and Equation (4.8) is applied to determine the residues and constant terms.

4.3 Review of Passivity Enforcement Techniques

An accurate macromodel of an interconnect can be constructed by using the methods discussed in the previous sections from the measured frequency response data or from full-wave simulations of the physical structure. The macromodel constructed needs to satisfy all the requirements of a physical real system. The macromodel need to satisfy realness, stability, causality, and passivity. The constraint for real coefficients can be satisfied by construction. In the construction of the macromodel, if all poles and residues are constructed in such a way that they appear as a real or complex-conjugate form, the whole rational function matrix will have real coefficients. Therefore, poles and residues must appear as real or as conjugate pairs. The stability conditions can be satisfied by ensuring the poles of the rational function matrix lie on the left half of the complex s-plane or inside the unit circle in z-plane and eradicating those poles that do not satisfy this condition. The passivity constraint must be satisfied and guaranteed over the infinite frequency bandwidth. None of the curve-fitting methods can assure passivity of the macromodel even if the original frequency data is passive.

The conditions for passivity was discussed in Chapter 3. All existing passivity enforcement techniques for curve-fitting methods are a posteriori approaches based on first-order perturbation theory. In these methods, the transfer matrix is obtained from the curve-fitting methods accurately approximating the original frequency response over the entire frequency range. Then, the macromodel is checked for passivity violation frequency regions and perturbation methods is applied to enforce passivity, assuming only weak violation of passivity occur in the constructed macromodel.

Since poles are characteristic frequencies of the system, common poles have been used to represent the multiport passive system when approximating by rational function matrix $H(s)$. Different approaches are available to enforce passivity. Some of these approaches are frequency range dependent. In [21], an eigenvalue approach has been discussed, which enforces passivity of the macromodel by directly compensating the poles and residues of the rational function using linearization and constrained minimization through quadratic programming. However, this method that is based on searching the frequency band of violation, is computationally expensive and uses discrete, band-limited frequency samples for enforcing passivity of the macromodel. Hence, the generated macromodel can still violate passivity over a continuous frequency and outside the band-limited frequency response since the macromodel is tested at discrete frequency samples.

Recently, the rational function matrix $H(s)$ without common poles has been used to represent the passive macromodel for multiport circuits, and a compensation method has been proposed [22]. The compensation method can detect the frequency band violating passivity using the associated Hamiltonian matrix. In [23] and [24], a different approach is used to assure passive macromodel.

In the following section, some of the existing methods which guarantee passivity of the macromodel of a passive interconnect network.

4.3.1 Passive filter approach

In this method [23], using filter theory, the rational function of the system can be represented as the summation of low-pass, band-pass and all-pass filters. The main idea for this sufficient condition for constructing a passive macromodel is based on the fact that the summation of passive sub-macromodel also results in passive macromodel. Thus, $H(s)$ can be expressed in pole-residue form as the summation of filter.

$$H(s) = \sum_{n=1}^{BP} \left(\frac{\alpha_n + j\beta_n}{s - p_{nr} - jp_{ni}} + \frac{\alpha_n - j\beta_n}{s - p_{nr} + jp_{ni}} \right) + \sum_{m=1}^{LP} \frac{\gamma_m}{s - p_{mr}} + d + sh, \tag{4.14}$$

where the coefficients $p_{mr}, \gamma_m, p_{nr}, p_{ni}, \alpha_n, \beta_n, d$, and h are real values. BP is the number of complex-conjugate pole-pairs for band-pass filter and LP is the number of real poles for the low-pass filters. By combining the complex-conjugate poles, Equation (4.14) can be written as

$$H(s) = \sum_{n=1}^{BP} \frac{2\alpha_n(s - p_{nr}) - 2\beta_n p_{ni}}{(s - p_{nr})^2 + p_{ni}^2} + \sum_{m=1}^{LP} \frac{\gamma_m}{s - p_{mr}} + d + sh. \tag{4.15}$$

For a distributed multiport network with common system poles, Equation (4.15) can be generalized as

$$H(s) = \sum_{n=1}^{BP} \frac{2[\alpha_n](s - p_{nr}) - 2[\beta_n]p_{ni}}{(s - p_{nr})^2 + p_{ni}^2} + \sum_{m=1}^{LP} \frac{[\gamma_m]}{s - p_{mr}} + [d] + s[h], \tag{4.16}$$

where the matrices $[\gamma_m], [\alpha_n], [\beta_n], [d]$ and $[h]$ are $P \times P$ residue and constant matrices for a P-port macromodel network. The stability criterion requires that all poles must lie in the left half of the s-plane. This can be achieved easily during the construction of the macromodel by applying $p_{nr} < 0$ and $p_{mr} < 0$ in Equation (4.16). The passivity criterion requires that a passive circuit does not create energy. Nonpassive macromodels combined with a stable circuit can generate an unstable time-domain response. Based on the maximum modulus theorem [31], the passivity condition for a one-port network can be written as

$$\Re\{H(s = j\omega)\} \geq 0 \quad \forall\omega. \tag{4.17}$$

The one-port network rational function $H(s)$ can be separated into real $H_r(s)$ and imaginary $H_i(s)$ parts and results as follows:

$$H(s = j\omega) = H_r(j\omega) + jH_i(j\omega). \tag{4.18}$$

$$H_r(j\omega) = \sum_{n=1}^{BP} \frac{2\{\omega^2(-\alpha_n p_{nr} + \beta_n p_{ni}) + (p_{nr}^2 + p_{ni}^2)(-\alpha_n p_{nr} - \beta_n p_{ni})\}}{(p_{nr}^2 + p_{ni}^2 - \omega^2)^2 + (2p_{nr}\omega)^2} + \sum_{m=1}^{LP} \frac{-\gamma_m p_{mr}}{p_{mr}^2 + \omega^2} + d \tag{4.19}$$

$$H_i(j\omega) = \sum_{n=1}^{BP} \frac{2\omega\{\alpha_n(-p_{nr}^2 + p_{ni}^2 - \omega^2) - 2\beta_n p_{ni} p_{nr}\}}{(p_{nr}^2 + p_{ni}^2 - \omega^2)^2 + (2p_{nr}\omega)^2} + \sum_{m=1}^{LP} \frac{-\gamma_m \omega}{p_{mr}^2 + \omega^2} + \omega h. \tag{4.20}$$

For one-port network, the passivity of each subnetwork in Equation (4.16) can be satisfied using the analytical formulas for the passivity. Applying the passive constraint for a one-port network, Equation (4.17), the following constraints can be derived from Equations (4.19) and (4.20):

$$\begin{aligned} -\alpha_n p_{nr} \pm \beta_n p_{ni} &\geq 0 \\ \gamma_m &\geq 0 \\ d &\geq 0. \end{aligned} \tag{4.21}$$

Note that due to the stability constraint, $p_{nr} < 0$ and $p_{mr} < 0$. It is important to note that the summation of a passive subnetwork is always results in a passive network. The rational function $H(s)$ is regarded as a summation of subnetworks consisting of complex-conjugate poles and real poles with corresponding residues. Therefore, if every subnetwork in Equation (4.16) satisfies the passivity condition, the rational function $H(s)$ satisfies the passivity condition as well.

For a multiport network, the rational function $H(s)$ in Equation (4.16) has to be positive semidefinite at all frequencies according to the passivity constraints discussed before. Using the property of positive semidefiniteness, the one-port passivity formulae in Equation (4.17) can be extended to multiport passivity formulae as

$$\begin{aligned} eigenvalues\ of\ [\gamma_m] &\geq 0 \\ eigenvalues\ of\ [-\alpha_n p_{nr} \pm \beta_n p_{ni}] &\geq 0 \\ eigenvalues\ of\ [d] &\geq 0. \end{aligned} \tag{4.22}$$

From Equation (4.22), the multiport passivity formulae only depend on the poles and residues matrices, which are independent of frequency. Hence, the passivity of the macro-

model is satisfied over an infinite frequency. The multiport passive formulae are only enforced on each subnetwork of $H(s)$, and there is no relationship for passivity between subnetworks except that they contribute to the overall frequency response of the macromodel. This adds simplicity to the computation of the passive macromodel. For compensating negative eigenvalues in Equation (4.22), there are two free matrix variables $[\alpha_n]$ and $[\beta_n]$ related to two free variables of complex-conjugate poles $p_{nr} \pm jp_{ni}$, a free matrix variable $[\gamma_m]$ related to a real pole p_{mr}, and a free matrix variable $[d]$ that can be changed.

This method has an advantage that the coefficient matrices are frequency independent and once passivity criterion is met, the macromodel is guaranteed to be globally passive. But it has a disadvantage in which the conditions required are sufficient but not necessary.

4.3.2 Convex programming approach

A convex programming approach to the positive-real approximation of tabulated data has been suggested in [32], [33]. In this method, passivity is enforced by modifying the coefficients of an existing rational transfer function while the error between the modified transfer function and the target curve is minimized. It is assumed that a stable state-space model with no poles in the right-half plane has already been generated and pole estimates are sufficiently good for the level of approximation accuracy desired. This state-space model can be generated from the exiting rational function approximation techniques such vector-fitting method.

A linear time-invariant system is approximated by rational function and can be described in the state-space form as

$$\begin{aligned} \frac{d}{dt}x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{4.23}$$

The transfer function corresponding to the above equation can be written as

$$H(s) = C(sI - A)^{-1}B + D \tag{4.24}$$

For a system with m inputs and m outputs, the poles of the system constitute the state matrix $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ represent the input matrix. $C \in \mathbb{R}^{m\times n}$ represents the output matrix and contains the residues of the system. $D \in \mathbb{R}^{m\times m}$ consists of the direct coupling terms for n order of approximation. The passivity of the system represented by Equation (4.23) and positive realness of the transfer function $H(s)$ is related by the positive-real lemma [34].

Lemma 4.1 (positive-real lemma): The state-space system A, B, C, and D describe a state-space model whose transfer function is given by $H(s) = C(sI - A)^{-1}B + D$ and assume the state-space system described is controllable. Further, suppose that $H(s)$ has all its poles either in the left-hand plane or on the imaginary axis, in which case they are simple. $H(s)$ is passive if and only if the following linear matrix inequality (LMI) are

satisfied [33]:

$$\begin{bmatrix} -A^TP - PA & -PB + C^T \\ -B^TP + C & D + D^T \end{bmatrix} \geq 0 \qquad (4.25)$$
$$P \geq 0$$

The methods assumes that a suitable rational function approximation of the data with desired accuracy has be obtained. The main idea is to keep the poles constant and perturb the residues and direct coupling constants to obtain a passive macromodel which satisfies a given error criteria for approximation and the LMIs.

Assume that the admittance or impedance parameter matrices of a system $\hat{H}(s)$ is given for a set of N frequency points, s_1 through s_N. Also, a stable matrix rational function macromodel already exists. Let $H_{p,q}(s)$ and $\hat{H}_{p,q}(s)$ represent the transfer function and the data matrix between port p and q respectively. The problem at hand is to determine C and D such the the new model is passive and at the same time the error in the approximation is minimized. In other words, the following error cost function is minimized:

$$\sum_{k=1}^{N} \parallel H_{p,q}(s) - \hat{H}_{p,q}(s) \parallel_2^2 < t. \qquad (4.26)$$

Let B_q represents the q^{th} column of B and C_p represents the p^{th} row of C. Let e_p represents the p^{th} column of the identity matrix of size $m \times m$. Next $F_{p,q} \in \mathbb{R}^{2N \times (n+m)}$ and $G_{p,q} \in \mathbb{R}^{2N}$ are defined as (for the k_{th} row),

$$F_{p,q}(k,:) = \begin{cases} \Re(J(s_k)), \cdots k \leq N \\ \Im(J(s_{k-N})), \cdots k > N \end{cases} \qquad (4.27)$$

$$G_{p,q}(k,:) = \begin{cases} \Re(\hat{H}_{p,q}(s_k)), \cdots k \leq N \\ \Im(\hat{H}_{p,q}(s_{k-N}), \cdots k > N \end{cases} \tag{4.28}$$

where

$$J(s) = \left[B_q^T (s_k I - A^T)^{-1} e_q^T\right]. \tag{4.29}$$

With X_p denoting the p^{th} column of

$$X = \begin{bmatrix} C^T \\ \\ D^T \end{bmatrix}. \tag{4.30}$$

Therefore, Equation (4.26) can be written as

$$\sum_{k=1}^{N} \parallel H_{p,q}(s) - \hat{H}_{p,q}(s) \parallel_2^2 = \parallel F_{p,q} X_p - G_{p,q} \parallel_2^2 . \tag{4.31}$$

Performing the QR decomposition on $F_{p,q}$ as

$$F_{p,q} = Q_{p,q} R_{p,q}. \tag{4.32}$$

Equation (4.31) can be expressed as

$$\sum_{k=1}^{N} \parallel H_{p,q}(s) - \hat{H}_{p,q}(s) \parallel_2^2 = E_{p,q}^T E_{p,q} + \delta_{p,q}^2, \tag{4.33}$$

where

$$E_{p,q} = (R_{p,q} X_p - Q_{p,q}^T G_{p,q}), \tag{4.34}$$

and

$$\delta_{p,q}^2 = G_{p,q}^T (I - Q_{p,q}^T Q_{p,q}) G_{p,q}. \tag{4.35}$$

With the above definitions the constraint, Equation (4.31) becomes

$$E_{p,q}^T E_{p,q} + \delta_{p,q}^2 < t_{p,q}. \tag{4.36}$$

The optimization problem for obtaining the passive macromodel can thus be stated as

minimize $\quad t(C, P)$

subject to:

$$\begin{bmatrix} -A^T P - PA & -PB + C^T \\ -B^T P + C & D + D^T \end{bmatrix} \geq 0 \tag{4.37}$$

$$P \geq 0$$

$$E_{p,q}^T E_{p,q} + \delta_{p,q}^2 < t_{p,q}$$

$$t \geq 0$$

$$t_{p,q} \leq t \quad for \quad 1 \leq p, q \leq m. \tag{4.38}$$

The objective function and the constraints in Equation (4.37) are convex functions of $t_{p,q}$, P, and C. Therefore, it is possible to find a global minimum by using convex optimization algorithms [32].

However, with this approach, the computational cost exhibits rapid growth with the order n of the state-space system. The cost of forming the matrix needed to compute the step direction during each iteration of the optimization is roughly $O(n^6)$ [33]. Some improvements have been reported in the formulation of the above problem which can

bring down this cost to $O(mn^4)$ [33]. However, the total cost of this technique still remains high. One of the reasons for this can be attributed to the fact that this algorithm, while calculating the matrix C conforming to passivity conditions, also generates the redundant matrix $P \in \mathbb{R}^{n\times n}$. Hence, the resources are unnecessarily used to evaluate the P-matrix which is of no use for the desired purpose of obtaining passive model.

4.3.3 Nevanlinna-Pick interpolation approach

This method is based on multivariable Nevanlinna-Pick interpolation (NPI) [35], [36]. Given the sampled scattering matrix of a passive system, the method finds a guaranteed bounded-real matrix rational function approximation, while at the same time matching the frequency response of the original network. For the case of scattering parameters, a system is passive if its scattering matrix is bounded-real. For a linear system described by state-space form (A, B, C, D), the system is bounded-real if and only if the Hamiltonian matrix M,

$$M = \begin{bmatrix} A + BPC & BPB^T \\ -C^TPC & -A^T - C^TPB^T \end{bmatrix}, \tag{4.39}$$

has no purely imaginary eigenvalues, where $P = (I - D^TD)^{-1}$ exists [37].

Given set of scattering matrix, $H(s_k)$, the matrix rational function $F(s_k)$ interpolates the data set if

$$F(s_k) \cong H(s_k), \quad k = 1, 2, ..., N. \tag{4.40}$$

The Nevanlinna-Pick problem is to find an interpolating function $F(s)$, that is analytic in the open right-half plane, interpolates the data $H(s)$ and satisfies

$$\| F(s_k) \|_2 < 1 \tag{4.41}$$

Theorem 4.1 (Nevanlinna-Pick matrix): There exists a matrix function $F(s)$ that interpolates the given data set $H(s)$, that is analytic in the open right-half plane and satisfies $\| F(s_k) \|_2 < 1$, if and only if the Pick matrix,

$$\Lambda_{p,q} = \left[\frac{I - H_p^H H_q}{s_p^* + s_q} \right]_{1 \leq p,q \leq N} \tag{4.42}$$

is positive definite, where H is hermitian operator.

In Nevanlinna-Pick matrix theorem (Theorem 4.1), the data points are considered to lie in the interior of the open right-half plane, but the frequency sampled data is available on the imaginary axis. Therefore, the frequency identification problem becomes the boundary interpolation problem. This boundary interpolation problem is called a Loewner interpolation problem [36].

4.4 Review of Pole-Clustering Techniques

Once an intermediate pole-zero or pole-residue model of a system function is determined, a low-order model can be obtained. A simple method, such as the pole-clustering technique, can be used to further reduce the order of the intermediate model obtained through the approximation process. The basic idea of the approach is that poles of the intermediate higher order system that are clustered in certain areas can be grouped into several clusters. Then, these clusters are replaced by respective cluster centers that are calculated, using criteria that give good approximation to the original system.

4.4.1 Pole-Clustering method based on inverse-distance-measure

In [38], inverse distance-measure (IDM) criterion is used for obtaining the cluster centers of poles and zeros. The IDM gives larger weights to the poles near the imaginary axis in determining the cluster centers, because these poles have dominant effects on the system behavior. After the cluster centers are determined, the residues are calculated from the frequency of the original system.

The poles are first partitioned into clusters by identifying the larger jumps in the real and imaginary values of the poles. These jumps are identified by the relative difference between the poles defined as [39]

$$d = \frac{\parallel \tilde{p}_i - \tilde{p}_j \parallel}{\min(\parallel \tilde{p}_i \parallel, \parallel \tilde{p}_j \parallel)}, \quad for \;\; i \neq j \tag{4.43}$$

where $\tilde{p}_i$'s are the poles in each cluster. The cluster center for each partition is calculated as

$$p_j = \left\{ \frac{1}{r} \sum_{i=1}^{r} \frac{1}{Re(\tilde{p}_i)} \right\}^{-1} + j \left\{ \frac{1}{r} \sum_{i=1}^{r} \frac{1}{Im(\tilde{p}_i)} \right\}^{-1} \tag{4.44}$$

where p_j is the cluster center. Separate partitions are formed for real poles and complex conjugate poles. Only the complex poles in one quadrant of the complex plane need to considered. The pole-clustering algorithm is a partitioning problem that contains two restrictions to guarantee the accuracy of the low-order model. The number of partitions appears as a constraint in the partition problem. If a large number of partitions is used, the system may still be too complex to handled efficiently. At the same time, if the partitions are few, the model can be too simple to capture some of the important features of the system.

The pole-clustering method using IDM is summarized as follows [39] to calculate cluster centers:

(1) Collect the real and complex poles in separate groups.

(2) Calculate the distance between the poles using Equation (4.43).

(3) Partition the pole in to clusters.

(4) Using Equation (4.44), calculate the cluster centers.

4.4.2 Pole-Clustering method based on neural-network

Neural network-based clustering techniques are well known and commonly used in statistical and other fields [40], [41]. The clustering capabilities of self-organizing maps (SOM) have been successfully verified in data analysis and data mining to organize and to identify small number of groups in a large collection based on their similarities to each other, Neural networks are able to learn hidden patterns in very large data and identify the clusters. Similarly, a small number of pole clusters can be extracted from the large number of poles representing distributed system using the SOM neural network.

The SOM is a special type of artificial neural networks first described by Teuvo Kohonen for visualization of high dimensional data [42]. It is trained using unsupervised learning to produce low dimensional representation of the training samples while preserving the topological properties of the input space. SOM os mainly used for dimensionality reduction. This makes SOM attractive for clustering purposes of a large number of poles.

The SOM is a single layer of feed-forward network where the inputs are the poles of the intermediate model and the outputs are the cluster centers. Each input is connected to all output neurons. Attached to every neuron there is a weight vector with the same

dimensionality as the input vectors. The number of the input dimensions is a lot higher than the output cluster centers.

The SOM-based pole-clustering algorithm assumes fixed number of cluster centers and try to find optimum locations for the centers. For each cluster, the cluster center is calculated using IDM. The weight of each cluster is calculated using Equation (4.44). In the SOM algorithm, the topological closeness is learned through a neighborhood update. First, for each pole $p_i, i = 1, 2, \cdots, N$, find the best matching node c_j:

$$c_j = \min_{i,j} \parallel p_j - w_j \parallel . \tag{4.45}$$

where $w_j(n)$ is the weight associated with every node. Then, the weight of the cluster centers are updated using

$$w_j^{new}(n) = w_j^{old}(n) + \eta(n)(x^T(n) - w_j^{old}(n)) \tag{4.46}$$

where η is the learning rate.

The SOM-based pole-clustering method is summarized as follows:

(1) Given N original poles and their locations, p, initialize randomly the locations of m cluster centers.

(2) For each cluster, determine the wining neurons for which the weight vector is the closest to input space using Equation (4.45).

(3) Update the weight of the wining neuron using Equation (4.46).

(4) Repeat the training epoch if the weights do not converge to the centers of the clusters.

4.5 Time-Domain Simulation Using State-Space Form

Consider the linear time-invariant (LTI) state-space

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t) \tag{4.47}$$

$$y(t) = Cx(t) + Du(t) \tag{4.48}$$

where $x \in \mathbb{R}^n$ represents the state variables of the system, $y \in \mathbb{R}^q$ and $u \in \mathbb{R}^p$ are the q-output and p-input of the system respectively. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{q\times n}$ and $D \in \mathbb{R}^{q\times p}$ are constant matrices for the linear time-invariant system. The state matrix A contains the poles in diagonal and B is a single column matrix that contains all ones. The C matrix contains the residues in a row matrix, and D contains the direct terms. The problem is to find the transient response of the system excited by an initial state $x(0)$ and input $u(t)$. The solution hinges on the exponential function of A.

$$\frac{d}{dt}e^{At} = Ae^{At} = e^{At}A$$

Premultiplying e^{-At} on both sides of Equation (4.47) yields

$$e^{-At}\frac{d}{dt}x(t) - e^{-At}Ax(t) = e^{-At}Bu(t),$$

which implies

$$\frac{d}{dt}(e^{-At}x(t)) = e^{-At}Bu(t).$$

Its integration from 0 to t yields

$$e^{-A}x(\tau)\Big|_{\tau=0}^{t} = \int_0^t e^{-A\tau}Bu(\tau)d\tau.$$

Thus, we have

$$e^{-At}x(t) - e^{0}x(0) = \int_0^t e^{-A\tau}Bu(\tau)d\tau.$$

Finally, the state solution is

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau. \tag{4.49}$$

Substituting Equation (4.49) into Equation (4.48) to find the output response for the given initial condition and input results in

$$y(t) = Ce^{At}x(0) + C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau + Du(t). \tag{4.50}$$

Discretization: Assume the input $u(t)$ is piecewise constant. Let the input be

$$u(t) = u(kT) =: u[k] \quad for \quad hT \le t \le (k+1)T, \quad k = 0, 1, 2, \cdots \tag{4.51}$$

This input changes values only at discrete time instants. For this input, the solution of Equation (4.47) still equals to Equation (4.49). Computing Equation (4.49) at $t = kT$ and $t = (k+1)T$ yields

$$\begin{aligned} x[k] &:= x(kT) = e^{AkT}x(0) + \int_0^{kT} e^{A(kT-\tau)}Bu(\tau)d\tau \qquad (4.52)\\ x[k+1] &:= x((k+1)T) = e^{A(k+1)T}x(0) + \int_0^{(k+1)T} e^{A((k+1)T-\tau)}Bu(\tau)d\tau. \end{aligned}$$

Equation (4.52) can be rewritten as

$$\begin{aligned} x[k+1] &= e^{AT}[e^{AkT}x(0) + \int_0^{kT} e^{A(kT-\tau)}Bu(\tau)d\tau] + \int_{kT}^{(k+1)T} e^{A(kT+T-\tau)}Bu(\tau)d\tau \\ x[k+1] &= e^{AT}x[k] + \int_{kT}^{(k+1)T} e^{A(kT+T-\tau)}Bu(\tau)d\tau. \end{aligned} \tag{4.53}$$

Let $\alpha := kT + T - \tau$,

$$x[k+1] = e^{AT}x[k] + \left(\int_0^T e^{A\alpha}d\alpha\right)Bu[k]. \tag{4.54}$$

Therefore, if the input changes values only at discrete-time instants kT and if we compute only the response at $t = kT$, then the state space equations in Equation (4.47) becomes

$$x[k+1] = A_d x[k] + B_d u[k] \tag{4.55}$$

$$y[k] = C_d x[k] + D_d u[k], \tag{4.56}$$

with

$$A_d = e^{AT}, \quad B_d = \left(\int_0^T e^{A\alpha}d\alpha\right)B, \quad C_d = C, \quad D_d = D.$$

This is the discrete-time state-space equation. There is no approximation involved in this derivation and yields the exact solution at $t = kT$ if the input is piecewise constant.

The computation of B_d in Equation (4.55) is as follows:

$$\begin{aligned} \int_0^T e^{A\alpha}d\alpha &= \int_0^T \left(I + A\alpha + A^2\frac{\alpha^2}{2!} + \cdots\right)d\alpha \\ &= TI + \frac{T^2}{2!}A^2 + \frac{T^3}{3!}A^3 + \frac{T^4}{4!}A^4 + \cdots \end{aligned}$$

This power series can be computed recursively. If A is nonsingular, then the series can be written as

$$A^{-1}\left(TA + \frac{T^2}{2!}A^2 + \frac{T^3}{3!}A^3 + \frac{T^4}{4!}A^4 + \cdots + I - I\right) = A^{-1}(e^{AT} - I).$$

Thus, we have

$$B_d = A^{-1}(A_d - I)B. \tag{4.57}$$

This way of computing avoids computing of an infinite series.

In order to discuss the general behavior of discrete-time state equations, it can be computed recursively as

$$\begin{aligned} x[1] &= A_d x[0] + B_d u[0] \\ x[2] &= A_d x[1] + B_d u[1] = A_d^2 x[0] + A_d B_d u[0] + B_d u[1]. \end{aligned}$$

Proceeding forward [43], we can readily obtain, for $k > 0$:

$$x[k] = A_d^k + \sum_{m=0}^{k-1} A_d^{k-1-m} B_d u[m], \tag{4.58}$$

$$y[k] = C_d A_d^k x[0] + \sum_{m=0}^{k-1} C_d A_d^{k-1-m} B_d u[m] + D_d u[k]. \tag{4.59}$$

Therefore, Equations (4.58) and (4.59) can be computed recursively for a given input and initial state condition.

4.5.1 Handling of complex-conjugate residues and poles

When there are complex poles in the rational function, the computation involves evaluation of complex numbers. Since complex numbers cost twice as much in storage, they should be avoided whenever possible. In this method, we use a similarity transformation to overcome this problem. Since the complex poles appear in complex-conjugate pairs, the rational function will have the following form.

$$H(s) = \frac{k}{s-p} + \frac{\bar{k}}{s-\bar{p}}, \tag{4.60}$$

where $p = a + jb$ and $k = c + jd$ with the corresponding complex conjugate.

A state-space realization of the above system has the matrices of the following form

$$\begin{aligned} A &= \begin{bmatrix} a+jb & 0 \\ 0 & a-jb \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ C &= [c+jd \quad c-jd]\,. \end{aligned} \tag{4.61}$$

If two state-space realizations are equivalent, then their corresponding frequency domain models are identical. The two realizations shown below are equivalent as long as V is nonsingular:

$$\begin{aligned} \frac{d}{dt}x(t) &= VAV^{-1}x(t) + VBu(t) \\ &= A_1x(t) + B_1u(t) \\ y(t) &= CV^{-1}x(t) + Du(t) \\ &= C_1x(t) + Du(t). \end{aligned}$$

For the realization above, one can use the following V:

$$V = \begin{bmatrix} \frac{-1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ j\frac{1}{\sqrt{2}} & j\frac{-1}{\sqrt{2}} \end{bmatrix}, \tag{4.62}$$

which transform the matrices as follows:

$$\begin{aligned} A_1 &= \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \quad B_1 = \begin{bmatrix} -\sqrt{2} \\ 0 \end{bmatrix} \\ C_1 &= \begin{bmatrix} -\sqrt{2}c & \sqrt{2}d \end{bmatrix}. \end{aligned}$$

Therefore, for each complex-conjugate pole and residue pairs, the corresponding row and column of the state-space matrices are modified as in above equation. The matrices will have the following general form:

$$A = \begin{bmatrix} p_1 & & & & 0 \\ & a & -b & & \\ & b & a & & \\ & & & \ddots & \\ 0 & & & & p_N \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -\sqrt{2} \\ 0 \\ \vdots \\ 1 \end{bmatrix} \quad C = \begin{bmatrix} k_1 & -\sqrt{2}c & -\sqrt{2}d & \cdots & k_N \end{bmatrix}. \tag{4.63}$$

Note that all the matrices formulated this way are real matrices.

CHAPTER 5

RATIONAL MACROMODELING TECHNIQUE IN *s*-DOMAIN

5.1 Introduction

The vector-fitting algorithm has emerged as a method of choice in the generation of high-order rational functions over wide frequency ranges [7]. The method is an iterative technique based on pole relocation technique. It is shown that the iterative pole-relocation technique is the reformulation of SK iteration using partial fractions basis [44]. The method has two main advantages, namely, numerical stability and convergence. It does not suffer from numerical problem when approximating high-order system over a wide frequency range [9]. It arrives at the optimal solution by solving two linear solutions with few iterations.

However, the accuracy and convergence of the vector fitting method depends strongly on the number and the location of the starting poles. The standard method for selecting the orders and initial poles are in the most part heuristic [7], [8]. Orthogonal vector fitting (OVF) has been introduced in [45] to reduce the numerical sensitivity of the system equations to the choice of starting poles.

5.2 Mathematical Formulation

The basic idea of macromodeling is to construct a rational function that represents the system. The vector fitting method accurately generates high-order rational functions over a wide frequency range. The method is an iterative technique based on pole-zero relocation technique. It has two main advantages, namely, numerical stability and convergence. It does not suffer from numerical problems that occur when approximating a high-order system over a wide frequency range. It arrives at the optimal solution by solving two linear equations within a few iterations. The frequency response $H(s)$ of any linear time-invariant passive network can be represented using rational function. For an N^{th}-order system response, the rational function $H(s)$ can be written as

$$H(s) \approx \frac{\sum_{m=0}^{M} a_m s^m}{\sum_{n=0}^{N} b_m s^n} = \frac{a_0 + a_1 s + a_2 s^2 + \cdots + a_M s^M}{b_0 + b_1 s + b_2 s^2 + \cdots + b_N s^N}, \tag{5.1}$$

where $s = j\omega$ is the angular frequency in radians per second. M and N are the orders of the numerator and denominator, respectively. For stable real systems, the difference between the numerator order M and the denominator order N of the rational function does not exceed unity. $H(s)$ can be scattering, admittance, or impedance parameters generated from an electromagnetic simulation or from measurements. All coefficients in Equation (5.1) are unknown, but this nonlinear equation can be written as a linear matrix system equation. Equation (5.1) can be expressed as modal expression of the form

$$H(s) \approx \sum_{n=1}^{N} \frac{c_n}{s - a_n} + d + sh, \tag{5.2}$$

where c_n, a_n, d and h are residues, poles, real direct proportional constant, and real proportional constant, respectively. The residues and poles are either real quantities

or in complex-conjugate pairs. In order to determine the above nonlinear equation, the problem can be solved by decomposing into two linear problems. The first linear problem is a pole identification problem in which the poles of the system are identified from the given frequency sample data and an initial starting poles. The second linear problem is to determine the residues and constant terms based on the frequency sample data and the new system poles determined from the first linear problem.

5.2.1 Pole identification

In this stage, the system poles are fully identified from the given frequency sample data and initial starting poles in the frequency range. The methodology of determining the poles is discussed as follows. First, multiply the given frequency response $H(s)$ with an unknown augmented function $\sigma(s)$. The augmented function $\sigma(s)$ is defined strictly as

$$\sigma(s) = \sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \tilde{a}_n} + 1, \tag{5.3}$$

where $\tilde{c}_n$ and $\tilde{a}_n$ are unknown constants. The function $\sigma(s)$ is intentionally forced to approach unity at a very high frequencies. The whole equation under consideration can be expressed as

$$\begin{bmatrix} (\sigma H)(s) \\ \sigma(s) \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} \frac{c_n}{s-\tilde{a}_n} + d + sh \\ \sum_{n=1}^{N} \frac{\tilde{c}_n}{s-\tilde{a}_n} + 1 \end{bmatrix}. \tag{5.4}$$

In Equation (5.4) the rational approximation for $(\sigma H)(s)$ has the same poles as the approximation for $\sigma(s)$. These poles are the initial starting poles. Detailed information about how to choose these starting poles is given in later in this chapter. Multiplying

the second row in Equation (5.4) with $H(s)$ yields

$$\left(\sum_{n=1}^{N}\frac{c_n}{s-\tilde{a}_n}+d+sh\right) \approx \left(\sum_{n=1}^{N}\frac{\tilde{c}_n}{s-\tilde{a}_n}+1\right)H(s) \tag{5.5}$$

$$(\sigma H)_{fit}(s) \approx \sigma_{fit}(s)H(s). \tag{5.6}$$

Equation (5.6) is the same representation as Equation (5.5). Equation (5.5) reveals that a linear equation results with unknown $\tilde{c}_n, c_n, d$, and h. This equation can be rewritten for each frequency sample points as

$$\left(\sum_{n=1}^{N}\frac{c_n}{s-\tilde{a}_n}+d+sh\right)-\left(\sum_{n=1}^{N}\frac{\tilde{c}_n}{s-\tilde{a}_n}\right)H(s)\approx H(s). \tag{5.7}$$

In matrix form, it becomes

$$\begin{bmatrix} \frac{1}{s_1-\tilde{a}_1} & \cdots & \frac{1}{s_1-\tilde{a}_N} & \frac{-H(s_1)}{s_1-\tilde{a}_1} & \cdots & \frac{-H(s_1)}{s_1-\tilde{a}_N} & 1 & s_1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{1}{s_k-\tilde{a}_1} & \cdots & \frac{1}{s_k-\tilde{a}_N} & \frac{-H(s_k)}{s_k-\tilde{a}_1} & \cdots & \frac{-H(s_k)}{s_k-\tilde{a}_N} & 1 & s_k \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \\ \tilde{c}_1 \\ \vdots \\ \tilde{c}_N \\ d \\ h \end{bmatrix} = \begin{bmatrix} H(s_1) \\ \vdots \\ \\ \vdots \\ H(s_k) \end{bmatrix}. \tag{5.8}$$

where k is the number of frequency sample points. Since Equation (5.8) often results in an over-determined linear equation, the least square solution can easily be obtained using one of the standard methods.

Since the system considered is real, all poles must appear as real or complex-conjugate pair poles in order to insure stability of the macromodel. In addition, for every complex-conjugate pair pole, the residue must appear in complex-conjugate form to insure that the approximate rational function represents a real system because it results a real coefficients. Similarly, a real pole will result a real residue. But solving for the unknown in Equation (5.8) does not guarantee this condition. The initial starting poles must also satisfy real or complex-conjugate pair form. The above complex matrix is modified to ensure that the residues will be in complex-conjugate form for complex-conjugate as

$$\begin{aligned} a_n &= a'_n + ja''_n, \quad a^*_n = a'_n - ja''_n \\ c_n &= c'_n + jc''_n, \quad c^*_n = c'_n - jc''_n, \end{aligned} \tag{5.9}$$

where a'_n is a real part of the complex pole, a''_n is the imaginary part of the complex pole, and a^*_n is the complex-conjugate of a_n. Similarly, c'_n is the real part of the residue, c''_n is the imaginary part of the residue, and c^*_n is the complex-conjugate of c_n.

The governing modal equation can be rewritten in more detail form as having Q real poles and L complex-conjugate pole pairs as follows:

$$\begin{aligned} \sum_{r=1}^{Q} \frac{c_r}{s - \tilde{a}_r} &+ \sum_{n=1}^{L} \left(\frac{c_n}{s - \tilde{a}_n} + \frac{c^*_n}{s - \tilde{a}^*_n} \right) + d + sh \\ &- \left(\sum_{r=1}^{Q} \frac{c_r}{s - \tilde{a}_r} + \sum_{n=1}^{L} \left(\frac{\tilde{c}_n}{s - \tilde{a}_n} + \frac{\tilde{c}^*_n}{s - \tilde{a}^*_n} \right) \right) H(s) \approx H(s). \end{aligned} \tag{5.10}$$

For a real pole, two elements in the matrix have the following form:

$$A_{k,i} = \frac{1}{s_k - \tilde{a}_{i,r}}, \qquad A_{k,i+2N} = \frac{-H(s_k)}{s_k - \tilde{a}_{i,r}}. \tag{5.11}$$

For a complex-conjugate pole pair, four elements of the matrix have the following form:

$$\begin{aligned} A_{k,n} &= \frac{1}{s_k - \tilde{a}_{k,n}} + \frac{1}{s_k - \tilde{a}^*_{k,n}}, \qquad A_{k,n+2N} = \frac{-H(s_k)}{s_k - \tilde{a}_{k,n}} + \frac{-H(s_k)}{s_k - \tilde{a}^*_{k,n}} \\ A_{k,n+1} &= \frac{j}{s_k - \tilde{a}_{k,n}} - \frac{j}{s_k - \tilde{a}^*_{k,n}}, \qquad A_{k,n+2N+1} = \frac{-jH(s_k)}{s_k - \tilde{a}_{k,n}} + \frac{-jH(s_k)}{s_k - \tilde{a}^*_{k,n}}. \end{aligned} \tag{5.12}$$

In matrix form, it becomes

$$\begin{bmatrix} & & & & 1 & s_1 \\ & & & & \vdots & \vdots \\ R & C & G & F & & \\ & & & & \vdots & \vdots \\ & & & & 1 & s_k \end{bmatrix} x = \begin{bmatrix} H(s_1) \\ \vdots \\ \vdots \\ H(s_k) \end{bmatrix} \tag{5.13}$$

$$Ax = B,$$

where x is the unknowns, and R, C, G, and F are matrices of the following form:

$$R = \begin{bmatrix} \frac{1}{s_1 - \tilde{a}_{1r}} & \cdots & \frac{1}{s_1 - \tilde{a}_{Qr}} \\ \vdots & \ddots & \vdots \\ \frac{1}{s_k - \tilde{a}_{1r}} & \cdots & \frac{1}{s_k - \tilde{a}_{Qr}} \end{bmatrix} \tag{5.14}$$

$$C = \begin{bmatrix} \frac{1}{s_1-\tilde{a}_1} + \frac{1}{s_1-\tilde{a}_1^*} & \frac{j}{s_1-\tilde{a}_1} - \frac{j}{s_1-\tilde{a}_1^*} & \cdots & \cdots & \frac{1}{s_1-\tilde{a}_L} + \frac{1}{s_1-\tilde{a}_L^*} & \frac{j}{s_1-\tilde{a}_L} - \frac{j}{s_1-\tilde{a}_L^*} \\ \vdots & & \ddots & & \vdots & \vdots \\ \vdots & & & \ddots & \vdots & \vdots \\ \frac{1}{s_k-\tilde{a}_1} + \frac{1}{s_k-\tilde{a}_1^*} & \frac{j}{s_k-\tilde{a}_1} - \frac{j}{s_k-\tilde{a}_1^*} & \cdots & \cdots & \frac{1}{s_k-\tilde{a}_L} + \frac{1}{s_k-\tilde{a}_L^*} & \frac{j}{s_k-\tilde{a}_L} - \frac{j}{s_k-\tilde{a}_L^*} \end{bmatrix} \tag{5.15}$$

$$\begin{aligned} G &= -H(s)R \\ F &= -H(s)C, \end{aligned} \tag{5.16}$$

and the unknown x has the form:

$$x = (c_{1r} \ \cdots \ c_{Qr} \quad c_1^r \quad c_1^i \ \cdots \quad c_L^r \quad c_L^i \quad \tilde{c}_{1r} \ \cdots \ \tilde{c}_{Qr} \quad \tilde{c}_1^r \quad \tilde{c}_1^i \ \cdots \ \tilde{c}_L^r \quad \tilde{c}_L^i \quad d \quad h)^T. \tag{5.17}$$

The given data have only positive frequencies, and the complex matrix A can be modified in terms of real quantities by separating the real and imaginary part of the complex matrix as

$$\begin{bmatrix} A' & -A'' \\ A'' & A' \end{bmatrix} x = \begin{bmatrix} b' \\ b'' \end{bmatrix}, \tag{5.18}$$

where A' and A'' are the real and imaginary part of matrix A, respectively, with the corresponding b' and b'' vectors for the real and imaginary parts of the frequency response

data, respectively. Finally, the complex-conjugate residues are formed from x as

$$c_m = c_m^r \pm j c_m^i, \tag{5.19}$$

$$\tilde{c}_m = \tilde{c}_m^r \pm \tilde{c}_m^i. \tag{5.20}$$

Now, the augmented function $\sigma_{fit}(s)$ and the product of the augmented function and frequency response $(\sigma H)_{fit}(s)$ can be changed from modal form to fractions of numerator and denominator and rewritten as

$$(\sigma H)_{fit}(s) = \frac{\prod_{n=1}^{N+1}(s - z_n)}{\prod_{n=1}^{N}(s - \tilde{a}_n)}, \qquad \sigma_{fit}(s) = \frac{\prod_{n=1}^{N}(s - \tilde{z}_n)}{\prod_{n=1}^{N}(s - \tilde{a}_n)}. \tag{5.21}$$

From Equations (5.21), $H(s)$ can be obtained by dividing one from the other as,

$$H(s) \approx \frac{(\sigma H)_{fit}(s)}{\sigma_{fit}(s)} = \frac{\prod_{n=1}^{N+1}(s - z_n)}{\prod_{n=1}^{N}(s - \tilde{z}_n)}. \tag{5.22}$$

From Equation (5.22), the poles of $H(s)$ become the zeros of $\sigma_{fit}(s)$. The initial starting poles are canceled out in this division process because both arguments have developed to have the same poles. Therefore, if the zeros of $\sigma_{fit}(s)$ are determined, the system poles can be directly found. The next section describes how to determine the system zeros to systems described in state space representation.

5.2.2 Determination of zeros of $\sigma_{fit}(s)$

A linear time invariant system can be described by set of equations of the form

$$\begin{aligned} \frac{d}{dt}x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{5.23}$$

where $x(t) \in \mathbb{R}^n$ represents the state variables of the system, $y(t) \in \mathbb{R}^q$ and $u(t) \in \mathbb{R}^p$ are the q-output and p-input of the system respectively. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{q\times n}$ and $D \in \mathbb{R}^{q\times p}$ are constant matrices for the linear time-invariant system. Taking the Laplace transform of Equation (5.23) and $x(0)$ as initial condition of the state,

$$\begin{aligned} sX(s) - x(0) &= AX(s) + BU(s) \\ Y(s) &= CX(s) + DU(s), \end{aligned} \tag{5.24}$$

where $X(s), U(s)$, and $Y(s)$ are the Laplace transform of $x(t), u(t)$, and $y(t)$, respectively. Equation (5.24) can be rewritten as

$$\begin{aligned} (sI - A)X(s) - BU(s) &= x(0) \\ CX(s) + DU(s) &= Y(s), \end{aligned} \tag{5.25}$$

where I is an identity matrix of the same size as A. Writing the equation in matrix form,

$$\begin{bmatrix} (sI - A) & -B \\ C & D \end{bmatrix} \begin{bmatrix} X(s) \\ U(s) \end{bmatrix} = \begin{bmatrix} x(0) \\ Y(s) \end{bmatrix} \tag{5.26}$$

$$P(s)\begin{bmatrix} X(s) \\ U(s) \end{bmatrix} = \begin{bmatrix} x(0) \\ Y(s) \end{bmatrix}, \tag{5.27}$$

where $P(s)$ is called *system matrix*. The transfer function of the system, $H(s)$ is described as

$$H(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D. \tag{5.28}$$

The transfer function response representation in Equation (5.28) will be used to drive the zeros of the system defined by state space representation. For a minimal realization of a system defined by Equation (5.23), a system zero is defined as a number that makes the output of the system zero for a time greater than or equal to zero. For such system, a necessary and sufficient condition for an arbitrary input of the form $u(t) = ge^{\lambda t}l(t)$ where $l(t)$ is step input, g is arbitrary real number and output $y(t) = 0$ for $t \geq 0$, is when Equation (5.27) is determined at $t = 0$ with the input $u(t)$ and gives

$$P(\lambda)\begin{bmatrix} x_0 \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{5.29}$$

assuming that the initial condition of the state $x(0)$ is zero. Therefore, a number $\lambda \in \mathbb{C}$ is a zero of system in Equation (5.23) if and only if there exists $x_0 \neq 0$ where $x_0 \in \mathbb{C}^n$ and $g \in \mathbb{C}^p$ such that the triple λ, x_0, g satisfies Equation (5.29).

Theorem 5.1: A number $\lambda \in \mathbb{C}$ is a zero of a minimal system of Equation (5.23) if

- λ is the eigenvalue of $(A - BD^{+}C)$ and there exists an eigenvector x_0 of $(A - BD^{+}C)$ corresponding to λ.

- x_0 must be in the null space of $(I_r - DD^+)C$, then λ, x_0, g must satisfy Equation (5.29).

The matrix D^+ is called the pseudo-inverse or the Moore-Penrose generalized inverse of D [46]. D^+ is defined as $D^+ = (D^T D)^{-1} D^T$, where D^T is the transpose of D. I_r is an identity matrix. The proof of Theorem 5.1 can be found in [47]-[49].

The next step is to construct a state space model for the augmented function $\sigma_{fit}(s)$. The poles of the system response $H(s)$ are the zeros of the augmented function $\sigma_{fit}(s)$. Therefore, the modal canonical form of $\sigma_{fit}(s)$ can be written as

$$\begin{aligned} \frac{d}{dt}x(t) &= \begin{bmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_N \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} u(t) = Ax(t) + Bu(t) \\ y(t) &= [\tilde{c}_1 \quad \cdots \quad \tilde{c}_N]x(t) + [d]u(t) = Cx(t) + Du(t), \end{aligned} \tag{5.30}$$

where A is a diagonal matrix whose elements are the poles of $\sigma_{fit}(s)$, which are the initial poles, B is a column vector whose elements are 1, C is a row vector whose elements are the residues of $\sigma_{fit}(s)$ and D is 1. In order to find the zeros of $\sigma_{fit}(s)$, according to the theorem, the eigenvalue of the matrix $(A - BD^+C)$ must be determined. Since $D = 1$,

$$D^+ = (D^T D)^{-1} D^T = 1 \tag{5.31}$$

Therefore, the zeros of $\sigma_{fit}(s)$ are the eigenvalue of the matrix $(A - BC)$, which are equally the poles of the frequency response $H(s)$.

By using this method, the poles of the system are identified. At this point, unstable poles, poles with a positive real part, can be deleted or modified to be stable poles with negative real part. This insures stability of the approximation process. In this pole identification process, initial poles are assigned at first and iteration can be used to refine the pole locations. These initial poles must be stable poles. One can set a maximum root-mean square (RMS) error to end up the iteration process.

5.2.3 Residues identification

Once the system poles are identified, the unknown residues and the constant term can be directly calculated from

$$H(s) = \sum_{n=1}^{N} \frac{c_n}{s - a_n} + d + sh. \tag{5.32}$$

Now, the system poles a_n are already determined from the pole identification process. The only remaining unknowns are c_n, d and h. The system poles are either real or complex-conjugate form. Therefore, the corresponding residue values must be the same form as the system poles. In matrix form, it becomes

$$\begin{bmatrix} \frac{1}{s_1-a_1} & \cdots & \frac{1}{s_1-a_N} & 1 & s_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{1}{s_k-a_1} & \cdots & \frac{1}{s_k-a_N} & 1 & s_k \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \\ d \\ h \end{bmatrix} = \begin{bmatrix} H(s_1) \\ \vdots \\ \\ \vdots \\ H(s_k) \end{bmatrix}. \tag{5.33}$$

Therefore, by following the above method, all the unknown residues and constant terms are obtained. The residues obtained must guarantee complex-conjugate form for complex-conjugate poles.

5.3 Selection of Starting Poles

The vector fitting method needs initial poles to approximate the given data in rational function. However, the accuracy and convergence of the vector fitting method depends strongly on the number and the location of the starting poles. The standard methods for selecting the orders and initial poles are in most part heuristics [7], [8]. So far, there has not been a study on the effects of the initial poles on the convergence of the vector fitting method. In [7], poles are selected in logarithmic or linear scale in the frequency range. But this way of selecting poles does not guarantee capturing the behavior of the data at the maximums and minimums. The number of the starting poles determines the order of approximation. The initial pole assignment proposed in [7] is to start with poles covering the approximation frequency range with the relationship between the imaginary and real poles as

$$p_i = \alpha_i \pm j\beta_i, \quad where \quad \alpha = \beta/100. \tag{5.34}$$

5.3.1 Starting poles selection using min-max of data

One possible way of selecting the initial poles is performed by automatically parsing the data within the frequency range. This method is based on finding frequency points at which the local minimums or maximums of the given data occur. These frequency points are used to create the starting poles. The advantage of this method is that it does

not require guessing the initial poles by users. One of the disadvantages of this method is that it generates a high-order approximation and can easily be affected by noise spikes in the given data. The fitting curves obtained from such a method of selecting the initial pole do not often give a smooth curve. The resulting starting poles from this method are all in conjugate complex form.

The procedure is summarized as follows:

(1) Normalize the whole frequency range using the maximum frequency.

(2) Starting from the beginning of the frequency range to the end, find the local minimum and maximum points of the data.

(3) Store the frequency point at which these extreme occurs $freq_{ex}$. Include the starting and the ending points of the frequency range.

(4) Set the imaginary part of the pole, $\beta = 2\pi \times freq_{ex}$.

(5) Create the initial poles, $p_i = \alpha_i \pm j\beta_i$, where $\alpha = \beta/100$.

In this dissertation, a new combined method of selecting initial poles for the vector fitting method is proposed. The set of initial poles is obtained by performing rational interpolation over partitioned frequency ranges. The method takes the advantage of the robustness of the rational interpolation in the extracting stable poles by partitioning the frequency range. These poles improve the accuracy and convergence of the vector fitting method. The new proposed method use the set of poles obtained by performing rational interpolation over a partitioned frequency range using the method developed in [5] and described below.

5.3.2 Starting poles selection using rational interpolation method

The starting poles can be obtained by partitioning the frequency range and by finding the poles of each partition region by rational interpolation using the real part of the partition as discussed in Chapter 4. The whole frequency range can be partitioned equally or in certain other criteria.

The resulting poles are used to formulate the stable initial poles for the vector-fitting method. The transposed Vandermonde-like interpolation matrix in Equation (4.5) can be artificially ill-conditioned; the condition number of the matrix can be improved by normalizing the frequency range to unity. Therefore, the whole frequency range under consideration is normalized by the maximum frequency under consideration. The transposed Vandermonde-like matrix in Equation (4.5), even for a moderate order, is notoriously ill-conditioned in the sense that the entries along each row are simple powers of the corresponding frequency values. If the span of the frequencies being considered is large, then the magnitudes of the entries on some of the rows will be much larger than those in rows corresponding to low frequency values. For this reason, the frequency range under consideration is partitioned into smaller frequency range, and by doing this the risk of solving an ill-conditioned matrix is minimized. The frequency range can be divided equally. A sufficient order of approximation should be set for this method to work. These poles are then used as starting poles for the vector-fitting algorithm. The resulting estimate poles at each iteration are used as the starting pole for the next iteration.

5.4 Combined Method

The proposed method [50] combines the robustness and efficiency of the rational interpolation and the numerical stability and fast convergence of the vector-fitting method. The rational interpolation method is first used to generate accurate poles of the partition frequency range. Then the poles are used in the vector-fitting method to generate a high-order rational function valid over wide frequency range taking advantage of numerical stability and fast convergence properties of the method.

The procedure is summarized as follows:

(1) Normalize the entire frequency range with the maximum frequency.

(2) Partition the frequency range into subsections and select the order of approximation to each partition.

(3) For each partition perform rational interpolation in Equation (4.5).

   (a) Construct the real part of the matrix and solve for denominator coefficients in Equation (4.6).

   (b) Factor the denominator polynomial and filter the stable poles.

(4) Collect all the poles from each partition and construct the initial guess poles, and perform vector fitting.

   (a) Set the least square solution of the augmented functions at the frequency points using the initial poles.

   (b) Solve for new set of poles.

   (c) Find the residues and constants using the new set of poles.

(d) Calculate RMS error. If the error is not acceptable, repeat step 4 using the new set of poles as initial guesses. If the error is acceptable, go to passivity checking and enforcement.

The proposed combined method provides a more accurate and fast convergence rational approximation.

5.5 Order Selection and Error Estimation

An important step in any order reduction scheme is order selection. Unfortunately, in most methods predicting the order of approximation is very difficult. The approach is usually to start with a low order and increase the order until a convergence criterion is satisfied. Ideally the approximate response is compared to the exact response at the desired frequency points. If the error is not acceptable, the order can be increased until the desired RMS error is achieved. Many frequency-sampling points should be taken in order to get a better approximation in continuous frequency.

5.6 Simulation Results

5.6.1 Example 1

Measured scattering parameters of interconnect are approximated over a frequency range 50 MHz-10 GHz using the proposed methods. First, the measured data is approximated by using the initial poles selection method proposed in [7]. Next, the measured data is approximated using automatic starting pole selection. In this method, the order depends on the number of extremes in the frequency range under consideration. The

order of approximation is 162. High order is expected due to the noisy nature of the data. The same data is approximated using the proposed combined method. Figures 5.1 and 5.2 show the result of the magnitude and phase of S_{21} of the given interconnect, respectively.

Table 5.1 Comparison between different starting poles selection methods

	VF	Automatic Selection	Combined Method
Number of Iteration	8	5	8
Order of Approximation	110	162	84
RMS error	0.00707	0.00029941	0.0009

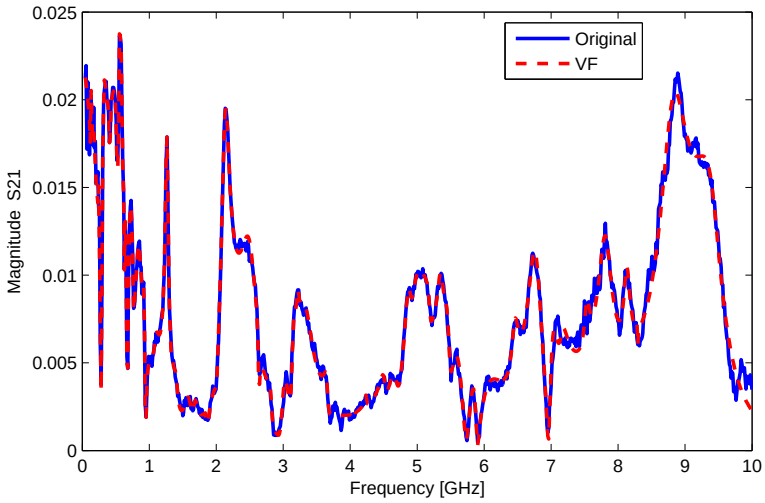


Figure 5.1 Magnitude comparison of S_{21} of the measured data and combined method.

Table 5.1 shows the comparison between these methods. The starting poles generated using the suggested rational interpolation method give better results with a lower order approximation. Even though the combined method needs more iterations than the standard vector fitting method for this example, the comparisons over many other examples show that the two methods require about the same number of iterations. The

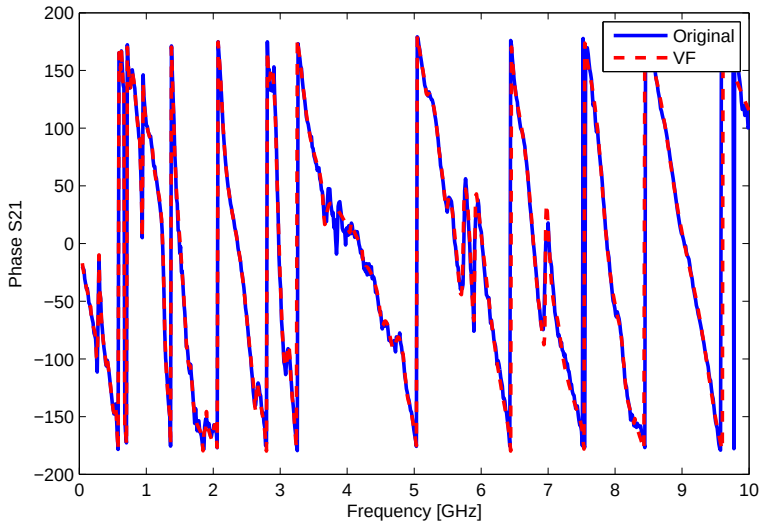


Figure 5.2 Phase comparison of S_{21} of the measured data and combined method.

RMS error is calculated at each iteration and run for 8 iterations using the combined method. Figure 5.3 shows the RMS error versus the number of iterations. The error drops very fast in the first few iterations and stagnates at high iterations. For example, the RMS error at the second iteration is 0.0029 while at the eighth iteration the RMS error is 0.0009 and stays constant for higher numbers of iterations.

5.6.2 Example 2

The scattering parameters of an interconnect system with V-shaped cross section measured over 10 GHz are used as an example to demonstrate the accuracy and validity of time-domain simulation. The reflection coefficient of the interconnect is approximated using the combined method. Figures 5.4 and 5.5 show the magnitude and phase comparison between the original response of S_{11} and the approximation with order 60, respectively.

Next, a time-domain simulation is performed with the input port terminated with match and the output port left open.

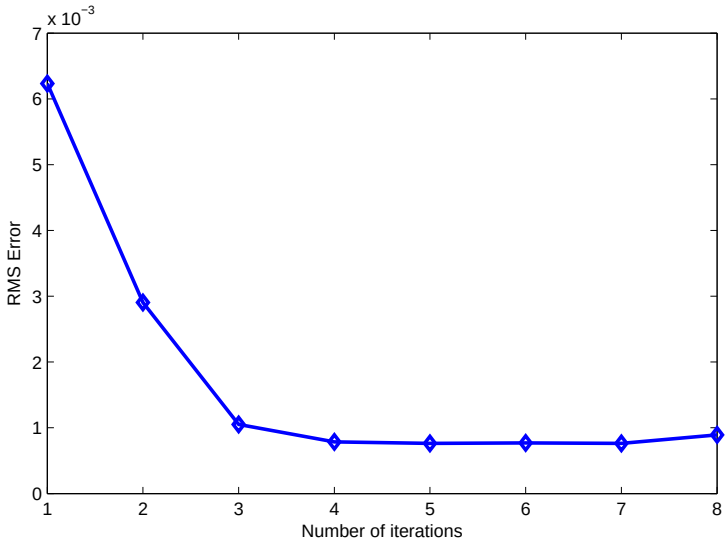


Figure 5.3 RMS error vs. iteration of the combined method.

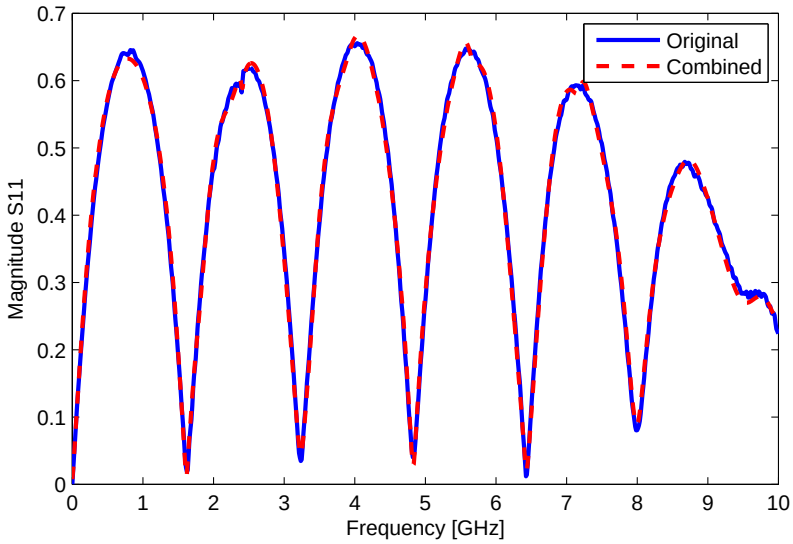


Figure 5.4 Magnitude comparison of S_{11} of the measured data and combined method.

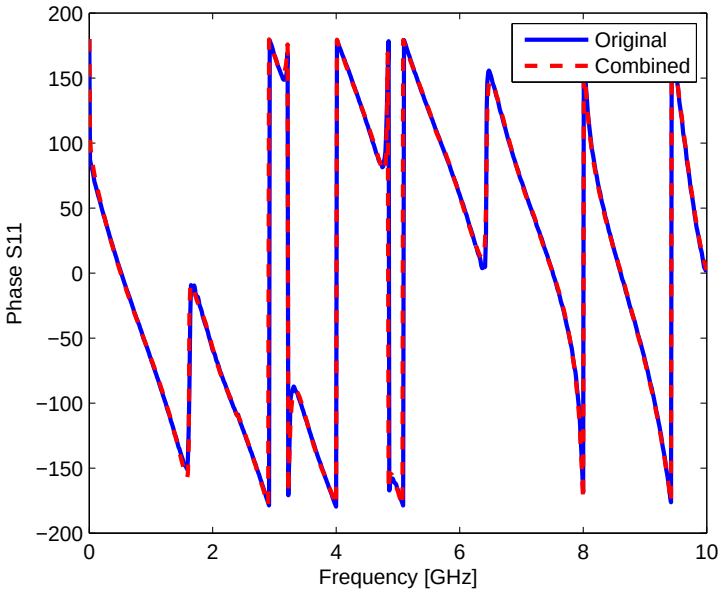


Figure 5.5 Phase comparison of S_{11} of the measured data and combined method.

Figure 5.6 shows the comparison between the recursive convolution using the state-space formulation and the direct convolution method. Since the frequency domain approximation is accurate, the time-domain response agrees very well.

Once the high-order approximation of the data is obtained, pole-clustering technique is applied on the generated poles. With the same level of accuracy, the order is reduced to 36, and cluster-centers are shown in Figure 5.7. The poles closer to the imaginary axis are retained and as the poles are located further from the axes, more and more of them are clustered into one center.

5.7 Conclusion

A combined method that uses rational interpolation is proposed to improve the initial poles and convergence of the vector-fitting method. The procedure takes advantage

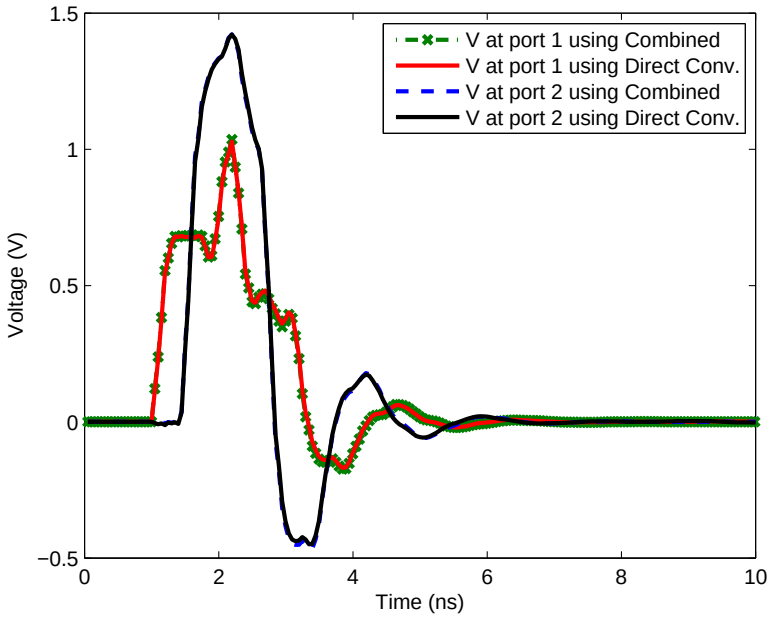


Figure 5.6 Time-domain comparison between the combined method and direct convolution.

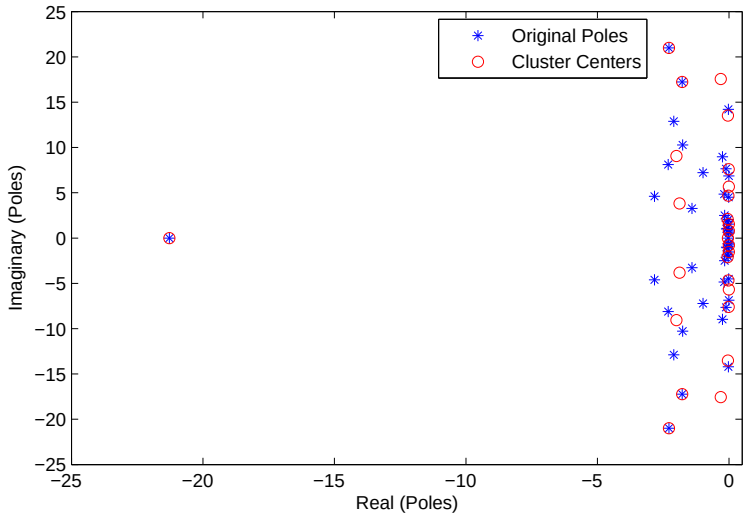


Figure 5.7 Original poles and cluster centers.

of the robustness of the rational interpolation method in generating stable poles over partitioned frequency range and the numerical stability and fast convergence of the vector fitting method to construct high-order approximation over wide frequency range. The convergence of the standard vector-fitting and proposed methods are compared. It is shown that the error decreases very fast and stagnates at high iterations.

CHAPTER 6

RATIONAL MACROMODELING TECHNIQUE IN z-DOMAIN

In this chapter, a new method of approximating a rational function is proposed using ZDVF [51], [52]. This has the advantage of faster convergence and better numerical stability compared to the s-domain vector-fitting method (VF). The fast convergence behavior of this method reduces the overall macromodel generation time. This chapter is organized as follows. First, a brief discussion on how to generate the z-domain data using bilinear transformation is presented. Next, the mathematical formulation of the proposed ZDVF method is presented. Starting pole selection is discussed next. Then, the state-space formulation for the time-domain simulation is presented. A numerical example is presented to illustrate the advantage of the ZDVF method on fast convergence and numerical stability as compared to the VF method. Finally, a conclusion is given.

6.1 s-to-z System Transformation

The z-domain response data can be generated using either by using bilinear transformation from s-domain to z-domain or using IFFT to time-domain and generate z-domain response data from time-domain response data. The z-transform of a sequence $x[n]$ is defined as

$$X(z) = \sum_{n=0}^{K} x[n] z^{-n} \tag{6.1}$$

where $z = e^{j\omega}$ and K is the number of samples.

6.1.1 The bilinear transformation

The bilinear transformation is a linear fractional transformation given by $\beta : \mathbb{C} \to \mathbb{C}$ defined by

$$\beta : s \to z = \frac{\alpha + s}{\alpha - s}, \tag{6.2}$$

where $\alpha \in \mathbb{R}$ is a constant equal to twice the sampling rate. This mapping has the property that it transforms the $j\omega$-axis in the s-plane onto the unit circle ($z = e^{j\omega}$) in the z-plane. Moreover, the left-hand side of s-plane ($\Re(s) < 0$) is mapped inside the unit circle in the z-plane and the right-hand side of s-plane ($\Re(s) > 0$) is mapped outside of the unit circle in the z-plane, thus preserves the system stability. The inverse bilinear transformation β^{-1} is given by $\beta(-\alpha/z)$.

The main advantage of using bilinear transformation over other transformations such as the impulse- and step-invariant transformations, is that it preserves the magnitude characteristics of the transfer function. This follows from the fact that the parameter α provides a one degree of freedom that can be used to make the frequency response characteristic of the z-domain system function $H(z)$ approximate those of the s-domain system function $H(s)$.

6.2 Mathematical Formulation of ZDVF

The z-domain response of a dynamic system can be constructed from the the time domain response or by using s-to-z bilinear transformation. The z-domain response $H(z)$

of any linear time-invariant passive network can be represented using rational function. For an N^{th}-order system response, the rational function can be written as

$$H(z) \approx \frac{N(z)}{D(z)} = \frac{\sum_{n=0}^{M} b_n z^{-n}}{\sum_{n=0}^{N} d_n z^{-n}}, \tag{6.3}$$

where $d_0 = 1$, M and N are the order of approximation of the numerator and denominator, respectively.

The approximation can be expressed in terms of the sum of a partial fraction expansion form shown in Equation (6.4), where c_n and a_n are the residues and poles, respectively.

$$H(z) \approx \sum_{n=1}^{N} \frac{c_n}{1 - z^{-1} a_n} \tag{6.4}$$

$$\hat{H}(z) = \frac{H(z)}{z} \approx \sum_{n=1}^{N} \frac{c_n}{z - a_n}. \tag{6.5}$$

Equation (6.5) is the response normalized by discrete frequency and is used to find the poles and the residues without losing generality. To determine the poles and the residues of Equation (6.5), a rational least-squares approximation is constructed to minimizing the following cost function:

$$\begin{aligned} & \arg\min_{b_n, d_n} \sum_{k=0}^{K} \left| H(z_k) - \frac{N(z_k)}{D(z_k)} \right|^2 \\ = \; & \arg\min_{b_n, d_n} \sum_{k=0}^{K} \frac{1}{|D(z_k)|^2} |D(z_k) H(z_k) - N(z_k)|^2. \end{aligned} \tag{6.6}$$

The above equation is nonlinear and it can be solved using SK iteration technique using the poles from the previous iteration step. At beginning, starting poles are selected in the stable region and Equation (6.6) can be reduced from nonlinear problem to a

linear problem since the denominator parameters are assumed to be known. Given the estimate poles from the previous iteration step $(i-1)$, a linear cost function is formulated to minimize and solve the poles at iteration i.

$$\arg \min_{b_n^{(i)}, d_n^{(i)}} \sum_{k=0}^{K} \frac{1}{|D^{(i-1)}(z_k)|^2} |D^{(i)}(z_k)H(z_k) - N^{(i)}(z_k)|^2 \tag{6.7}$$

Using poles from the previous iteration, Equation (6.7) can be written as partial fractions format at iteration i

$$N^{(i)}(z) = \sum_{p=1}^{P} \frac{c_p^{(i)}}{1 - z^{-1}\tilde{a}_p} = \frac{\prod_{p=1}^{P-1}(1 - z^{-1}\upsilon_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{a}_p)} \tag{6.8}$$

$$D^{(i)}(z) = \sum_{p=1}^{P} \frac{\tilde{c}_p^{(i)}}{1 - z^{-1}\tilde{a}_p} + 1 = \frac{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{a}_p)} \tag{6.9}$$

where $\tilde{a}_p$ are the estimated poles from the previous iteration. The denominator $D(z)$ has an additional basis function, which is equals to the constant value 1, since numerator and denominator can be divided by the same complex value without loss of generality. The error function can be expressed as

$$\arg \min_{b_n^{(i)}, d_n^{(i)}} \left(\sum_{k=0}^{K} \left| \frac{1}{D^{(i-1)}(z_k)} \right|^2 \left| D^{(i)}(z_k)H(z_k) - N^{(i)}(z_k) \right|^2 \right)$$

$$= \arg \min_{\upsilon_p^{(i)}, \tilde{\upsilon}_p^{(i)}} \left(\sum_{k=0}^{K} \left| \frac{\prod_{p=1}^{P}(1 - z^{-1}\tilde{a}_p)}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i-1)})} \right|^2 \left| \left(\frac{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{a}_p)} \right) H(z_k) - \frac{\prod_{p=1}^{P-1}(1 - z^{-1}\upsilon_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{a}_p)} \right|^2 \right) \tag{6.10}$$

$$= \arg \min_{\upsilon_p^{(i)}, \tilde{\upsilon}_p^{(i)}} \left(\sum_{k=0}^{K} \left| \left(\frac{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i-1)})} \right) H(z_k) - \frac{\prod_{p=1}^{P-1}(1 - z^{-1}\upsilon_p^{(i)})}{\prod_{p=1}^{P}(1 - z^{-1}\tilde{\upsilon}_p^{(i-1)})} \right|^2 \right) \tag{6.11}$$

$$= \arg \min_{c_p^{(i)}, \tilde{c}_p^{(i)}} \left(\sum_{k=0}^{K} \left| \left(\sum_{p=1}^{P} \frac{\tilde{c}_p^{(i)}}{(1 - z^{-1}\tilde{\upsilon}_p^{(i-1)})} + 1 \right) H(z_k) - \sum_{p=1}^{P} \frac{c_p^{(i)}}{(1 - z^{-1}\tilde{\upsilon}_p^{(i-1)})} \right|^2 \right) \tag{6.12}$$

The zeros of $D^{(i-1)}(z)$ from the previous iteration are used to calculate the poles. The poles of $N^{(i)}(z)$ and $D^{(i)}(z)$ remain unchanged, and cancel out in each iteration. Equation (6.12) is rewritten for the normalized response in Equation (6.5) using partial fractions at iteration i as

$$\arg \min_{\tilde{c}_p^{(i)}, c_p^{(i)}} \sum_{k=0}^{K} \left| \left(\sum_{p=1}^{P} \frac{\tilde{c}_p^{(i)}}{z_k - \tilde{\upsilon}_p^{(i-1)}} + 1 \right) \hat{H}(z_k) - \sum_{p=1}^{P} \frac{c_p^{(i)}}{z_k - \tilde{\upsilon}_p^{(i-1)}} \right|^2 . \tag{6.13}$$

Again, this reduces to solving the following set of least-squares equations.

$$\sum_{p=1}^{P} \frac{c_p^{(i)}}{(z - \tilde{\upsilon}_p^{(i-1)})} - \left(\sum_{p=1}^{P} \frac{\tilde{c}_p^{(i)}}{(z - \tilde{\upsilon}_p^{(i-1)})} \right) \hat{H}(z) = \hat{H}(z). \tag{6.14}$$

In matrix form at iteration i,

$$\begin{bmatrix} \frac{1}{z_1 - \tilde{a}_1} & \cdots & \cdots & \frac{1}{z_1 - \tilde{a}_N} & \frac{-\hat{H}(z_1)}{z_1 - \tilde{a}_1} & \cdots & \cdots & \frac{-\hat{H}(z_1)}{z_1 - \tilde{a}_N} \\ \vdots & \ddots & & \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots \\ \frac{1}{z_K - \tilde{a}_1} & \cdots & \cdots & \frac{1}{z_K - \tilde{a}_N} & \frac{-\hat{H}(z_K)}{z_K - \tilde{a}_1} & \cdots & \cdots & \frac{-\hat{H}(z_K)}{z_K - \tilde{a}_N} \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \\ \tilde{c}_1 \\ \vdots \\ \tilde{c}_N \end{bmatrix} = \begin{bmatrix} \hat{H}(z_1) \\ \vdots \\ \vdots \\ \hat{H}(z_K) \end{bmatrix}, \tag{6.15}$$

where K is the number of frequency sample points and $\tilde{a}_p = \tilde{\upsilon}_p^{(i-1)}$. Since Equation (6.15) often results in an over-determined linear equation, the least square solution can be easily be obtained using one of the standard methods.

Since the system considered is real, all poles must appear as real or complex-conjugate pair poles in order to insure stability of the macromodel. In addition, for every complex-conjugate pair pole, the residues must appear in complex-conjugate form to insure that the approximate rational function represents a real system because it results a real coefficients. Similarly, a real pole will result a real residue. The initial starting poles must also satisfy real or complex-conjugate pair form.

Once the parameter $c_p^{(i)}$ and $\tilde{c}_p^{(i)}$ are calculated, $N^{(i)}(z)$ and $D^{(i)}(z)$ are known. It is straight forward to calculated $\upsilon_p^{(i)}$ and $\upsilon_p^{(i)}$ in robust way, by solving the eigenvalue problem of Equation (6.16):

$$\{a_n\} = eig(A - B.C^T) \tag{6.16}$$

In Equation (6.16), A is a diagonal matrix holding the initial poles $\tilde{a}_n$, B is a column vector of ones, and C^T is a row vector holding the residues $\tilde{c}_n$. The stable poles are the poles inside the unit circle. If unstable poles are found outside of the unit circle, these poles can either be forced to be stable (inside the unit circle) or rejected. This procedure, known as *pole relocation*, is iterated using the estimated poles as starting poles. The solution procedure is based on least squares approximation. Using this method, convergence is often reached within a few iterations.

The residues can be easily solved using Equation (6.5), since the poles are now identified. To make sure that the transfer function has real-valued coefficients, the linear problem is constructed in such a way that a complex-conjugate pole pair results in a complex-conjugate residue pair.

The main reason for the fast convergence of the ZDVF is that the pole relocation occurs inside the stable unit circle and the initial starting pole relocate a very small distance inside the unit circle as opposed to the s-domain vector-fitting scheme in which the poles travel long distances within the stable left hand side of the s-plane.

6.3 Selection of Starting Poles

A possible way of selecting a starting pole was introduced in [7] and [8] for s-domain vector-fitting. In [7], the poles are selected in a logarithmic or linear scale in the frequency range and complex conjugate pole selection. In ZDVF, the starting poles are selected as complex-conjugate pairs in a linear scale and in the proximity of the unit circle in the stable region. The number of starting poles determines the order of the approximation. The poles are formulated as

$$\{Startingpoles\} = \alpha e^{\pm j\pi\theta_i}. \tag{6.17}$$

For example, α can be chosen 0.9 and θ_i is an angle chosen in equally spaced scale where $0 \leq \theta_i \leq 2\pi$ in complex-conjugate pairs. Figure 6.1 shows a possible starting poles distribution.

6.4 Time-Domain Simulation Using State-Space Form

For a system characterized by scattering matrix $S(j\omega)$, the input is a forward wave to the ports $\bar{a}(t)$ and the output is a reflected wave from the ports $\bar{b}(t)$ in time domain.

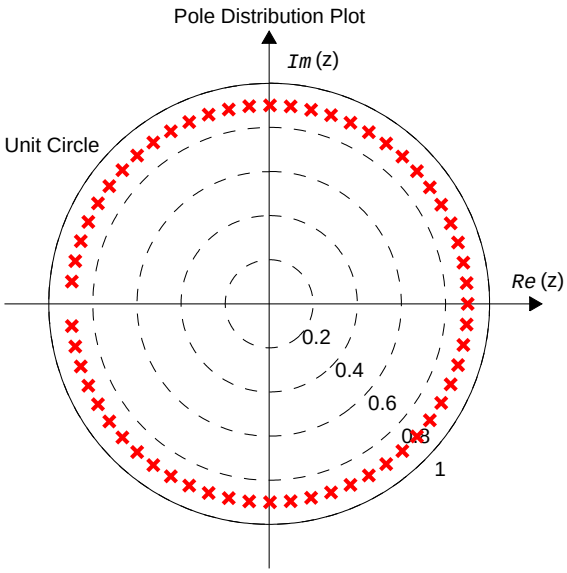


Figure 6.1 Starting Poles distribution.

Thus, the N-port S-parameter equation is given by

$$\bar{b}(t) = s(t) * \bar{a}(t), \tag{6.18}$$

where $*$ is direct convolution where $s(t)$ is the time-domain scattering-parameters. The termination condition is given by

$$\bar{b}(t) = \Gamma\bar{a}(t) + TV_g(t), \tag{6.19}$$

where Γ is the reflection coefficient matrix, T is the voltage division matrix, and $V_g(t)$ is the source voltage vector:

$$\Gamma = -[I + Z_T Z_0^{-1}]^{-1}[I - Z_T Z_0^{-1}] \tag{6.20}$$

$$T = [I + Z_T Z_0^{-1}]^{-1}. \tag{6.21}$$

Z_T is the termination impedance matrix and Z_0 is the reference impedance matrix. Once the poles and the residues of $S(z)$ are obtained, the equivalent state-space model can be formulated for time-domain simulation. Since, the system is assumed to be a linear time-invariant discrete-time system, it can be expressed in a state-space form as

$$x[k+1] = Ax[k] + B\bar{a}[k] \tag{6.22}$$

$$\bar{b}[k] = Cx[k] + D\bar{a}[k],$$

where $x \in \mathbb{R}^n$ represents the state variables of the system, $\bar{b} \in \mathbb{R}^q$ and $\bar{a} \in \mathbb{R}^p$ are the q-output reflected wave and p-input forward wave of the system respectively. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{q\times n}$ and $D \in \mathbb{R}^{q\times p}$ are constant matrices for the linear time-invariant system. The state matrix Acontains the poles in diagonal and B is a single column matrix that contains all ones. The C matrix contains the residues in a single-row matrix, and D is zero in this formulation. The general solution for the above equation can be achieved by recursive manner. It is obvious that for a given initial condition x[0],

$$x[1] = Ax[0] + B\bar{a}[0]$$

$$x[2] = Ax[1] + B\bar{a}[1] = A^2x[0] + AB\bar{a}[0] + B\bar{a}[1].$$

Let $\Phi[k] = A^k$. Therefore, the general solution becomes [43]

$$x[k] = \Phi[k]x[0] + \sum_{j=0}^{k-1} \Phi[k-1-j]B\bar{a}[j] \tag{6.23}$$

$$\bar{b}[k] = C\Phi[k]x[0] + \sum_{j=0}^{k-1} C\Phi[k-1-j]B\bar{a}[j]. \tag{6.24}$$

Equation (6.24) gives the output reflected wave at each port at each time step for the given excitation input. The terminal voltage and current at each port can be easily calculated from the forward and reflected waves.

6.5 Numerical Examples

The scattering-parameters of a two-port interconnect structure is measured in the frequency range 50 MHz-20 GHz and the data is used as an example to demonstrate the accuracy and fast convergence of ZDVF. For comparison purposes, the data is approximated using both VF and ZDVF. The initial poles for VF is chosen as proposed in [7] while the initial poles for ZDVF are chosen as discussed in this chapter. Table 6.1 shows the comparison between the two methods for the given data. It shows that ZDVF converge within two iterations with a relatively lower order of approximation and better accuracy compared to the VF method. Since ZDVF converges faster than VF, the overall macromodeling time reduces significantly.

Table 6.1 Comparison between ZDVF and VF method

	VF	ZDVF
Number of Iteration	5	2
Order of Approximation	96	80
RMS error	0.0291	0.0059

Figures 6.2-6.5 show results for the magnitude and phase of S_{11} and S_{21} for the measured and the approximated values using ZDVF, respectively.

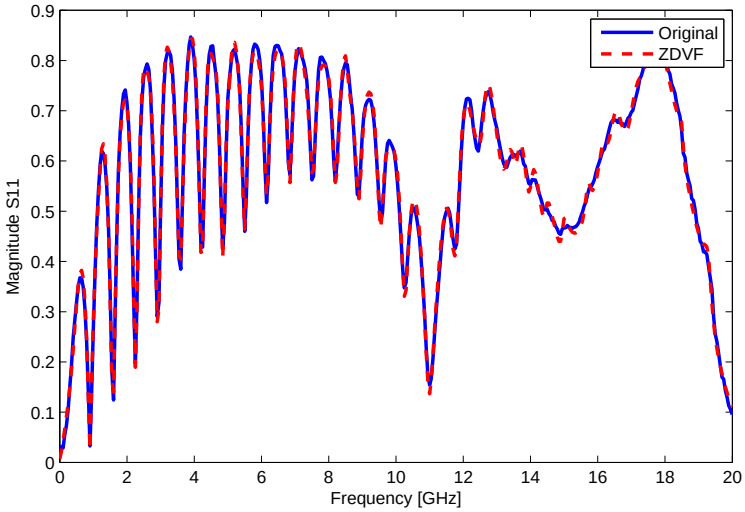


Figure 6.2 Magnitude comparison of S_{11} of the measured data and ZDVF.

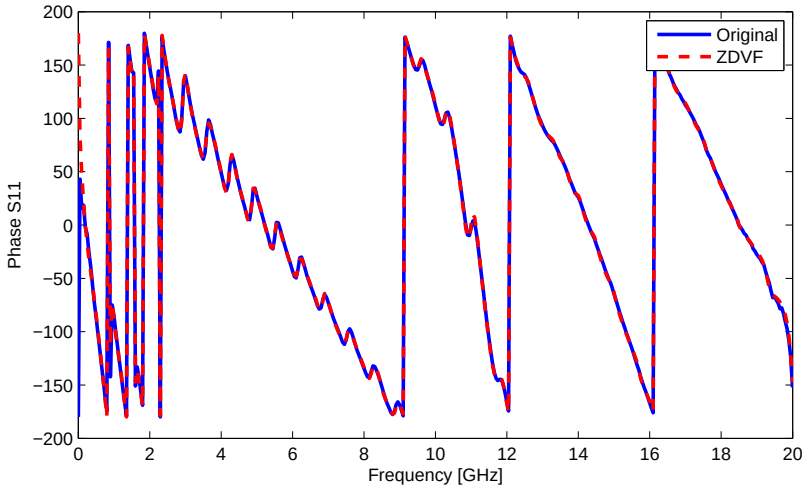


Figure 6.3 Phase comparison of S_{11} of the measured data and ZDVF.

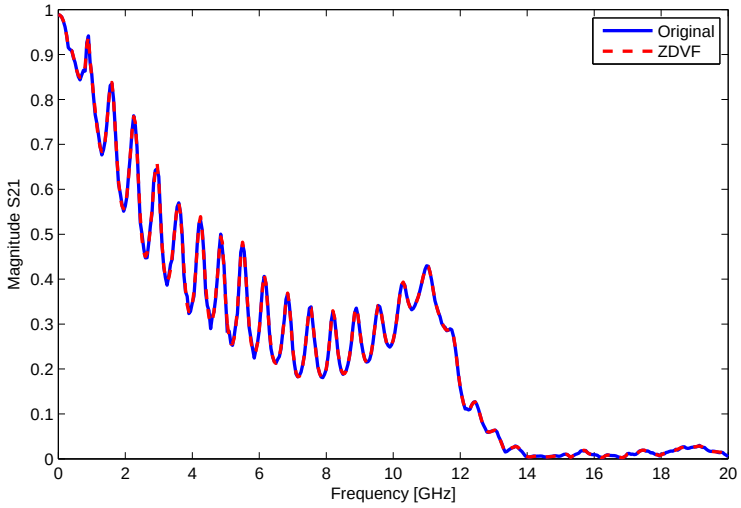


Figure 6.4 Magnitude comparison of S_{21} of the measured data and ZDVF.

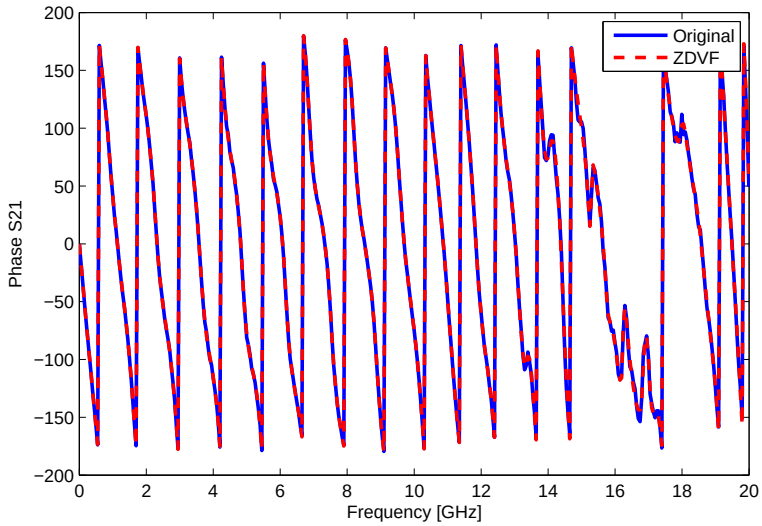


Figure 6.5 Phase comparison of S_{21} of the measured data and ZDVF.

Figure 6.6 shows the pole locations in z-plane obtained using ZDVF after two iterations. From the pole plot, the optimal poles are located close to the unit circle in the stable region. This shows choosing starting poles close to the unit circle as proposed gives a faster convergence.

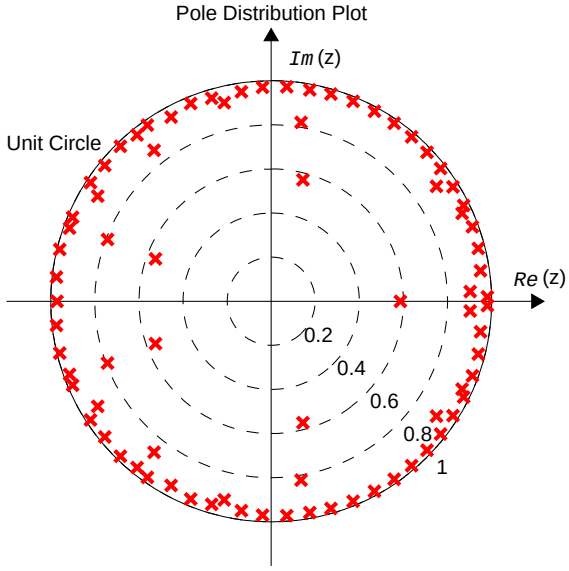


Figure 6.6 Plot of poles location in z-plane.

Figure 6.7 shows a comparison of the condition number of the pole identification between the VF and ZDVF methods. It shows that ZDVF constructs a well-conditioned matrix to identify the poles of the system at each iteration. Figure 6.8 shows a comparison of the RMS error between VF and ZDVF over 15 iterations. For ZDVF, the error drops rapidly at the first few iterations and remains constant at high iterations. It shows also ZDVF gives better accuracy even for a lower order of approximation.

Figure 6.9 shows the transient simulation results for a pulse input excitation with a matched termination condition at port 1 and open at port 2. It shows the transient

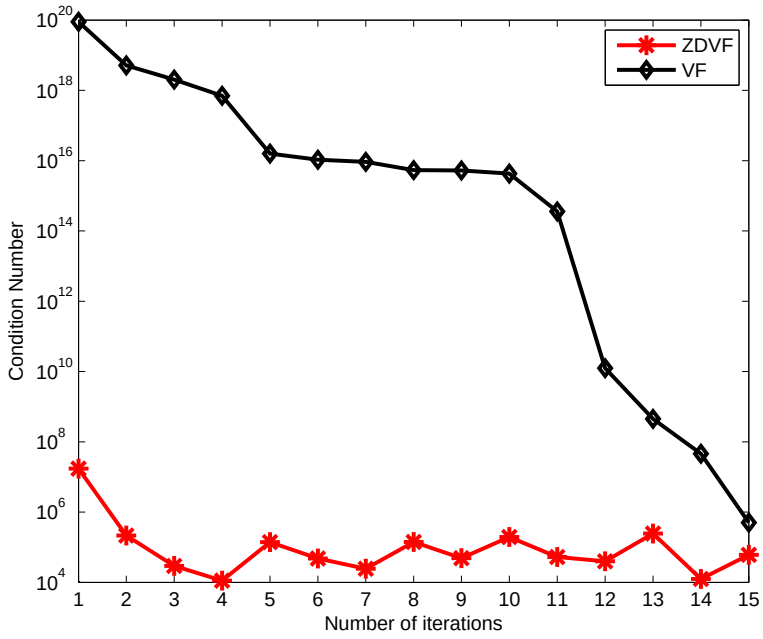


Figure 6.7 Condition number of VF versus ZDVF.

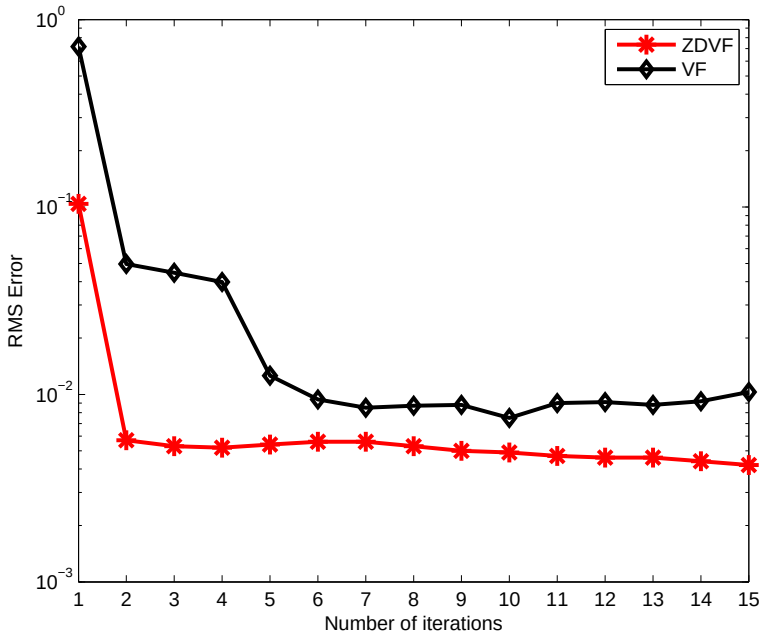


Figure 6.8 RMS Error VF versus ZDVF.

response at each port using a direct convolution technique [53] and ZDVF. Clearly, the result matches very well.

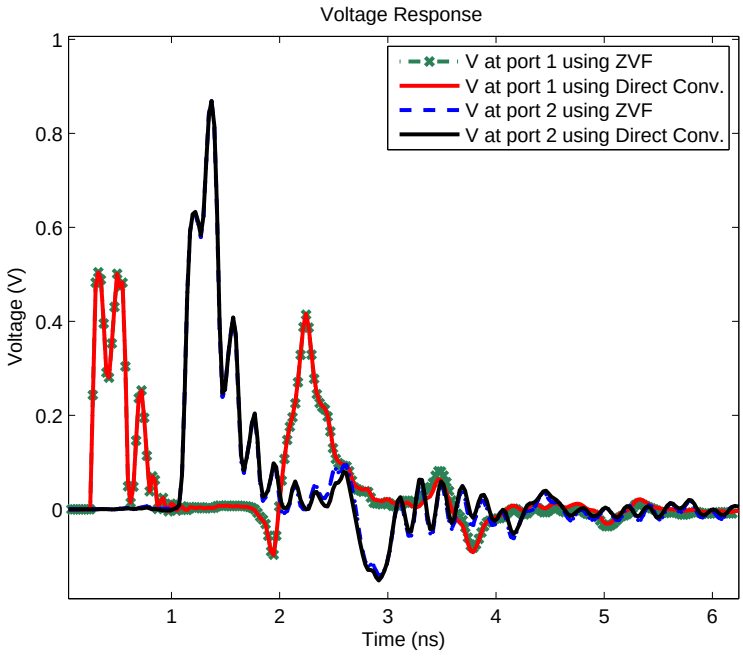


Figure 6.9 Time-domain comparison between direct-convolution and ZDVF.

The fast convergence of ZDVF is dependent on the starting poles location inside the unit circle. To illustrate, ZDVF simulation is performed by changing the starting pole locations α. Figure 6.10 shows how the condition number of a pole identification matrix at each iteration is changing for different values of α. For a small value of α near to the center of the unit circle, the condition number is large at each iterations which results a stability problem. Figure 6.11 shows how the RMS error at each iteration is changing for different values of α. Starting poles selected near the unit circle has a faster convergence compared to those poles selected near the center.

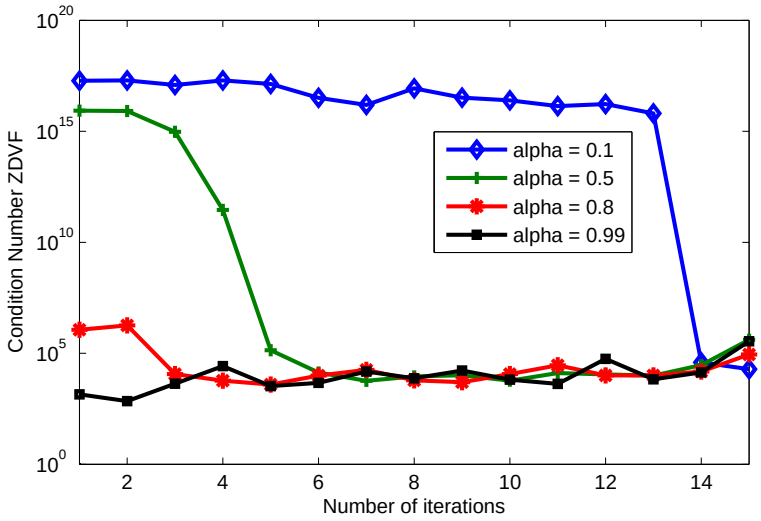


Figure 6.10 Condition Number of ZDVF as α change.

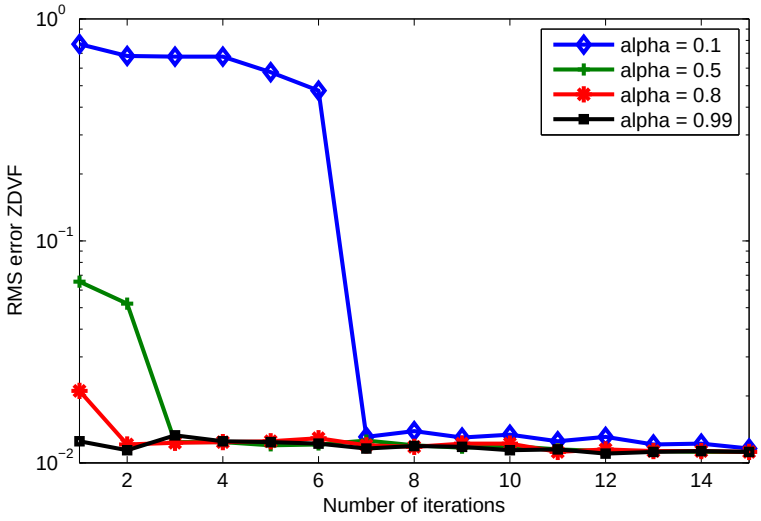


Figure 6.11 RMS Error of ZDVF as α change.

6.6 Test Cases

6.6.1 Test Case 1: Package via

The circuit model of the package via with power planes, shown in Figure 6.12, is extracted [53]-[57]. This 3D structure has a small loss, delay, and reflection. First, the two-port scattering parameters of the structure are obtained from full-wave electromagnetic solver. Then, the net consisting of two package via connected by a 6-mm trace is analyzed in the frequency and time domain. The scattering parameters of the net are measured using a vector network analyzer (VNA) over 20 GHz. Figures 6.13 and 6.14

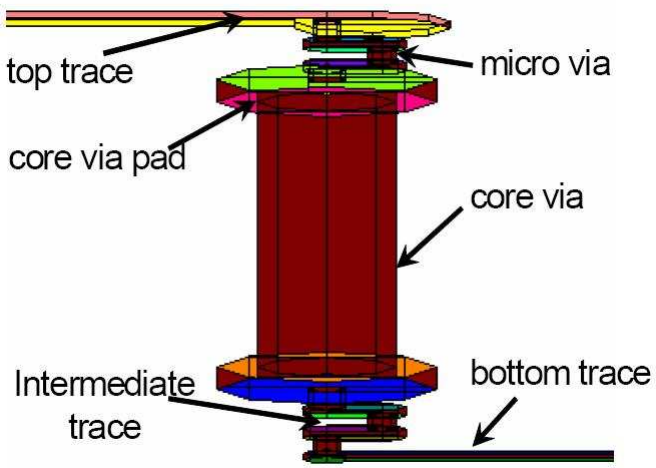
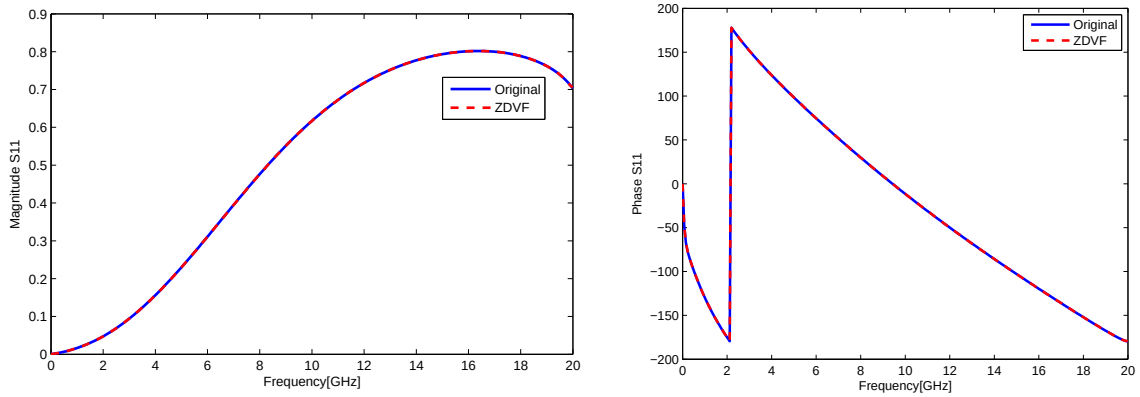


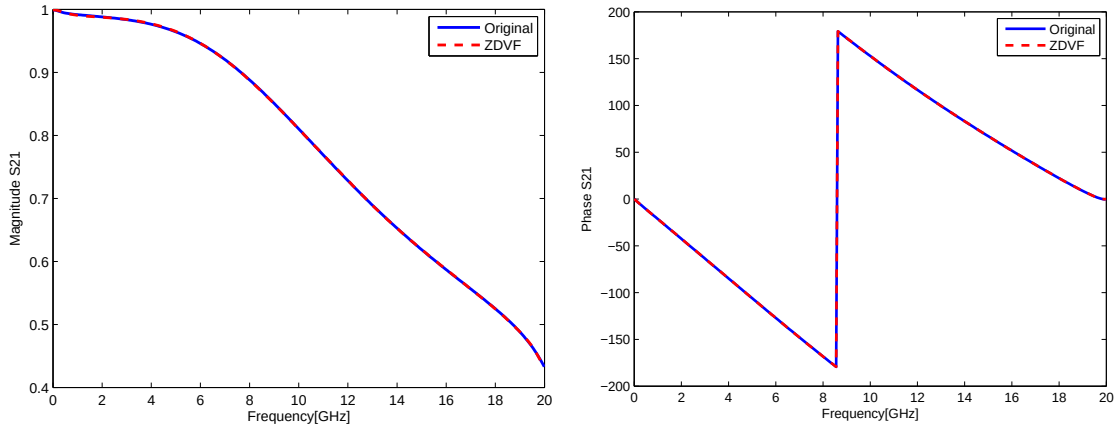
Figure 6.12 Package via.

show the comparisons of the measured return and insertion losses with those obtained from the approximation of the S-parameters using ZDVF, respectively. Table 6.2 shows selected parameters of the behavior of the system and parameters related to ZDVF.



(a) Magnitude (b) Phase

Figure 6.13 Comparison of reflection coefficient S_{11} of the original data and the approximation of ZDVF of the package via.



(a) Magnitude (b) Phase

Figure 6.14 Comparison of insertion loss S_{21} of the original data and the approximation of ZDVF of the package via.

Table 6.2 Comparisons of the circuit-based, convolution-based and ZDVF methods

Parameters	A Package Via	Chip-to-Chip interconnect	Rambus memory channel
Delay	Very small	Large (1ns)	Large (2ns)
Reflection	Some	Small	Large
Loss	Small	Small	Large
Coupling	None	Large	Large
# Circuit Elements	136	1500	1400
Circuit-based CPU Time	10 sec.	287 min.	184 min.
Convolution CPU Time	1.6 sec.	2.1 sec.	5.0 sec.
ZDVF # Order	6	84	92
ZDVF # Iterations	3	3	3
ZDVF max. RMS Error	0.000941	0.076	0.08
ZDVF CPU Time	2 sec.	21 sec.	27 sec.

6.6.2 Test Case 2: Chip-to-chip interconnect system

The chip-to-chip interconnect system operating at 6.4 Gbps is shown in Figure 6.15 [53]-[57]. It has 6 to 12 in long FR4 PCB traces, packages with substrate traces up to 20-mm, package and PCB vias, solder balls to attach the packages on the board, and devices along the signal path, all of which contribute to signal attenuation. The interconnect is measured from 50 MHz to 20 GHz. The S-parameters are approximated by using ZDVF. Figures 6.16 and 6.17 show the magnitude and phase of the original data and the

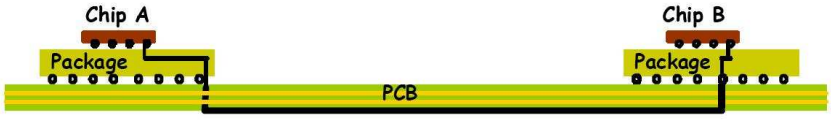
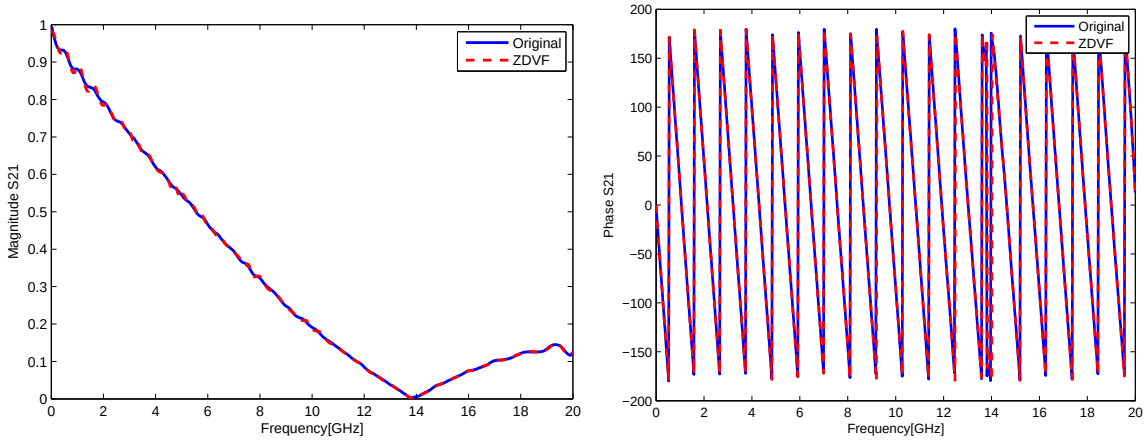


Figure 6.15 Chip-to-chip interconnect system that has up to 12-in long FR4 PCB traces and two packages with up to 20-mm long traces.

approximation using ZDVF of S_{21} and S_{31} of the chip-to-chip interconnect, respectively.

Table 6.2 shows selected parameters of the behavior of the system and parameters related to ZDVF.

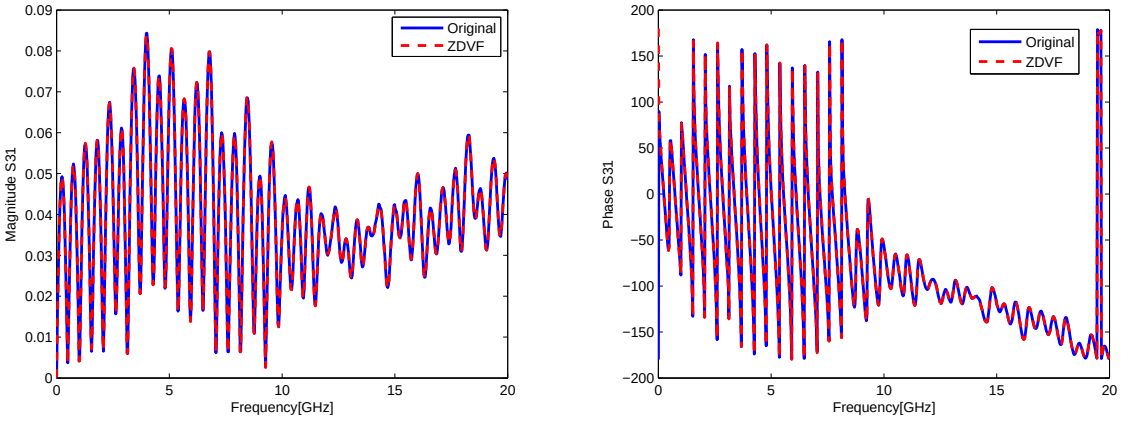


(a) Magnitude (b) Phase

Figure 6.16 Comparison of S_{21} of the original data and the approximation of ZDVF of the chip-to-chip interconnect.

6.6.3 Test Case 3: Rambus memory channel

The memory system consisting of two Rambus inline memory modules (RIMM), two connects, a controller package, and a termination network located on the motherboard [53]-[57]. The controller chip is usually placed in a wirebond package and the Rambus DRAM (RDRAM) memory devices are mounted on chip-scale package (CSP) that are located on the RIMMs, which contain up to 16 devices. Figure 6.18 shows the physical structure of the memory channel. The high-speed signal is transmitted between the chips at the controller and the memory module through packages, PCB traces, and connectors. The channel is a bus network that supports a single master and multiple slave devices, and is a uniform controlled-impedance path. This memory channel has significant loss, delay, reflection, and coupling. First, the four-port scattering parameters are calculated



(a) Magnitude (b) Phase

Figure 6.17 Comparison of S_{31} of the original data and the approximation of ZDVF of the chip-to-chip interconnect.

from the frequency-domain simulation. Next, the S-parameters are approximated by ZDVF. Figures 6.19 and 6.20 show the magnitude and phase of the original data and

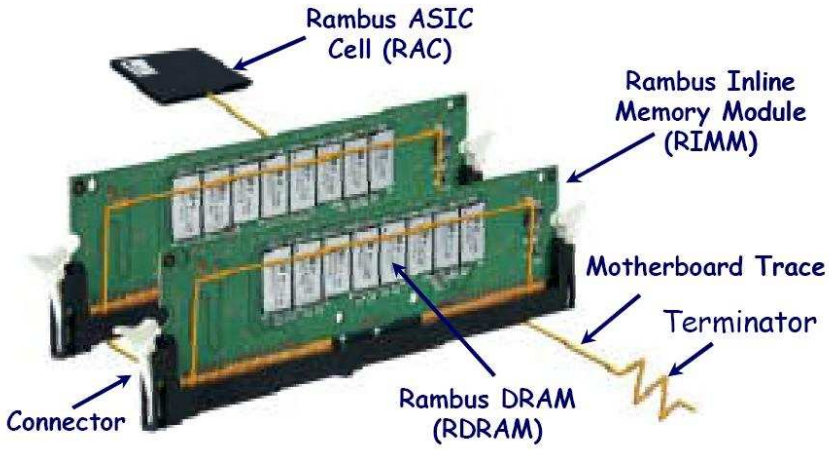
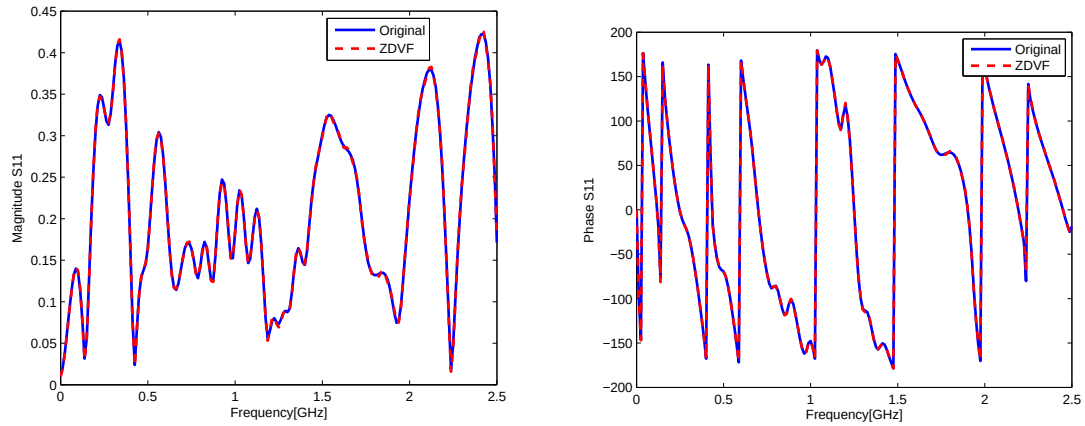


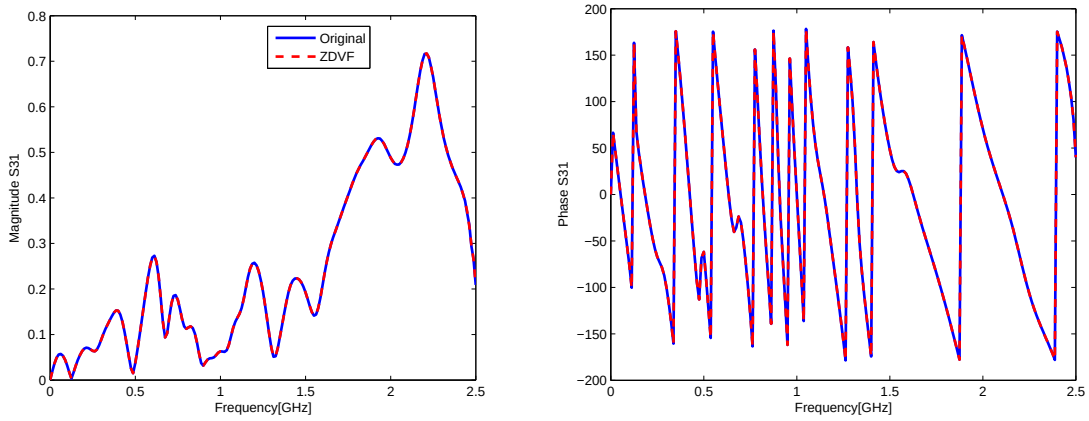
Figure 6.18 System configuration of direct Rambus memory channel.

the approximation using ZDVF of the reflection coefficient S_{11} and S_{31} of the Rambus memory channel, respectively. Table 6.2 shows selected parameters of the behavior of the system and parameters related to ZDVF.



(a) Magnitude (b) Phase

Figure 6.19 Comparison of the reflection coefficient S_{11} of the original data and the approximation of ZDVF of the Rambus memory channel.



(a) Magnitude (b) Phase

Figure 6.20 Comparison of S_{31} of the original data and the approximation of ZDVF of the Rambus memory channel.

6.7 Time-Domain System Identification Using ZDVF

In this section, an example is presented to apply the proposed ZDVF macromodeling technique for time-domain system identification and compare the accuracy and robustness of ZDVF to existing Prony's and Steiglitz-McBride (STMC) methods [58] in signal processing. Consider a 10^{th}-order Chebyshev-II filter with pass band from 200 Hz-400 Hz. Once the impulse response of the filter is created, the impulse response is approximated using Prony's, STMC, and ZDVF methods and compare the approximation with the original impulse response [58]. Figure 6.21 shows the magnitude comparison between the methods and the original response. Figure 6.22 shows the phase comparison between the methods and the original response. The order of approximation of these methods is equal to the order of the filter. There is a good agreement in the magnitude of the response. But the phase approximation of Prony's and STMC methods do not agree with the original response. On the other hand, the ZDVF approximate the original response accurately both in magnitude and in phase.

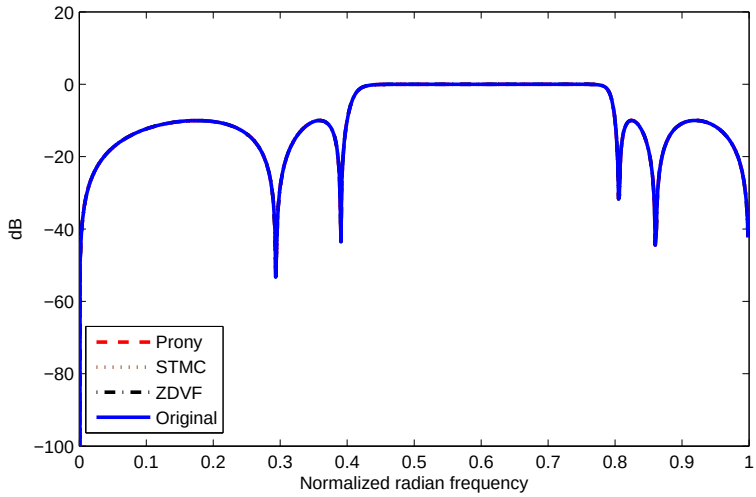


Figure 6.21 Magnitude (dB) comparison between Prony's, STMC, ZDVF, and the original 10^{th}-order Chevyshev-II filter.

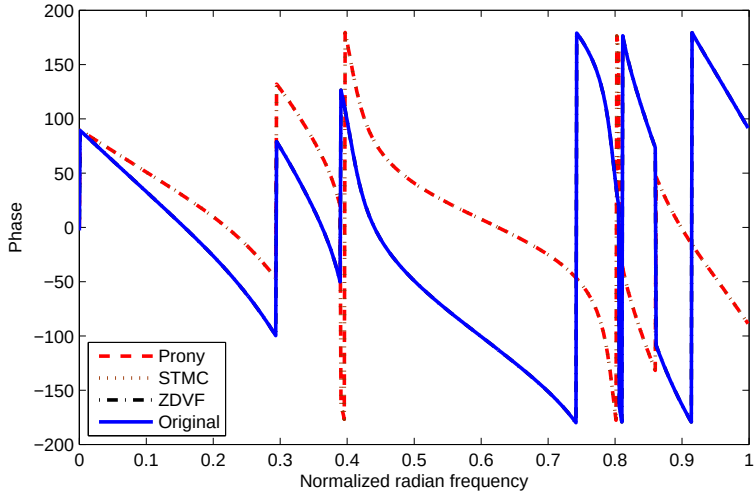


Figure 6.22 Phase comparison between Prony's, STMC, ZDVF, and the original 10^{th}-order Chevyshev-II filter.

6.8 Conclusion

The ZDVF method has been proposed and verified to improve the convergence and numerical stability of vector-fitting method to construct broadband macromodel over wide frequency range. The pole-relocation is confined within a unit-circle in the z-plane and therefore a faster convergence is achieved. In addition, selecting initial poles close to the unit-circle provides an even faster convergence. The convergence rates of the vector fitting method and the proposed ZDVF method are compared. The ZDVF method exhibits a rapid decrease in the error and stabilizes at high iterations.

CHAPTER 7

DELAY EXTRACTION SCHEME

7.1 Introduction

Macromodeling of a broadband network involves development of a rational function representation of the network which approximates its port-to-port behavior as discussed in the previous chapters. Such a representation is generated by approximating the frequency-domain or the time-domain response of the network using complex poles and residues of $H(s)$ in s-domain, $H(z)$ in the z-domain. Once the poles and residues are known, they can be easily integrated to SPICE for time-domain simulation. $H(z)$ generated this way is stable if all the poles lie inside the unit-circle in z-plane. Macromodels developed using such techniques satisfy the stability and passivity criteria using passivity checking and enforcement techniques, but not causality. This is because distributed passive systems like transmission lines have infinite poles, and rational function approximate the response using only a finite number of poles, obtained using bandlimited frequency response data. This prevents $H(z)$ from accurately capturing the delay in the network, since capturing delay using a rational function in the pole-residue form would require an infinite number of poles. The time-domain delay takes the form of complex exponential shift, usually represented by $e^{-j\omega\tau}$ in the frequency domain or $z^{-\tau}$ in z-domain. The exponential term, if represented by rational function, will need an infinite number of poles and therefore cannot be exactly modeled by any rational function with

finite order. Hence, when the rational function is used to model a system with embedded delay, a much higher order is often necessary to maintain the modeling accuracy within the frequency band of interest.

To reduce the order of approximation and to guarantee causality, it is essential to first extract the delay of the system. The approach of combining the rational function and complex exponential shift term to model systems with delay has been investigated in several papers [59]. In this chapter, a delay extraction scheme is presented using a minimum-phase extraction of the response data in z-domain. The response data is decomposed in to a minimum-phase rational function, an all-pass function and a delay term.

7.2 Minimum-Phase Extraction

A system is a minimum-phase system if all the poles and the zeros of the system is in the left half of the complex s-plane or inside the unit-circle in z-plane. This property constrains the phase response of the system such that $-\pi < \angle H(z) < \pi$. The phase response of such functions does not show any phase transition. From linear system theory [20], [60], any stable linear time-invariant system function can be represented as a product of a minimum phase function and an all-pass function, where an all-pass function is one whose magnitude is unity over the entire frequency range and the phase contains the phase-shift associated with delay:

$$H(z) = H_{min}(z) \times H_a(z). \tag{7.1}$$

One can further express the phase of all-pass function into a constant delay term. From Equation (7.1),

$$H(z) = H_{min}(z) \times H_b(z) \times z^{-\tau} \tag{7.2}$$

where $H_a(z) = H_b(z) \times z^{-\tau}$. Since $|H_b(z)| = |z^{-\tau}| = 1$, then

$$|H(z)| = |H_{min}(z)|. \tag{7.3}$$

From Equation (7.3), $H(z)$ is separated into a product of a minimum phase function and an all-pass function, the all-pass function will represent the delay. This separation can be performed using the Hilbert transform [20]. The Hilbert transform relates the magnitude and phase of a minimum phase function $H_{min}(z)$ through the following equation, where $z = e^{j\omega}$

$$\arg[H_{min}(j\omega)] = -\frac{1}{2\pi} P \int_{-\pi}^{\pi} \log|H_{min}(j\theta)| cot\left(\frac{\omega - \theta}{2}\right) d\theta, \tag{7.4}$$

where P is the Cauchy principal value. Since an all-pass function has unity magnitude, the magnitude response of the minimum phase function $H_{min}(z)$ is the same as that of $H(z)$.

7.3 Delay Extraction

Once the minimum phase part of the system is separated, the rest of the system can be obtained from Equation (7.2) as

$$H_a(z) = \frac{H(z)}{H_{min}(z)} = H_b(z) \times z^{-\tau}. \tag{7.5}$$

From Equation (7.5), the phase relationship can be expressed:

$$\arg[z^{-\tau}] = \arg[H(z)] - \arg[H_{min}(z)] - \arg[H_b(z)]. \tag{7.6}$$

From Equation (7.6), the average delay τ can be easily extracted by taking the derivative of Equation (7.6) with respect to the discrete frequency. Therefore, the response without delay is expressed as,

$$H_d(z) = \frac{H(z)}{z^{-\tau}}. \tag{7.7}$$

After the response without the delay $H_d(z)$ is determined, the ZDVF method is used to construct the transfer function representation. $H_d(z)$ should contain the least delay among all rational functions with the same spectral magnitude. Therefore, $H_d(z)$ can be accurately constructed with a relatively low-order.

7.4 Numerical Example

To illustrate the advantage of extracting the delay to reduce the order of approximation of ZDVF, a four-port long channel scattering parameters are measured in frequency range 10 MHz - 20 GHz. Due to the delay, it needs a very high order of approximation to approximate the S-parameters. Therefore, the delay is first extracted from the S-parameters by the methods discussed above. Figure 7.1 shows the time-domain response of $S_{21}(t)$ by applying IFFT on $S_{21}(s)$. There is a significant delay in this response which needs to be extracted. Figure 7.2 shows the unwrapped phase of the original system, the minimum-phase system and the response without delay. The phase of the minimum-phase system has variation inside $[-\pi, \pi]$ as expected. The response without the delay

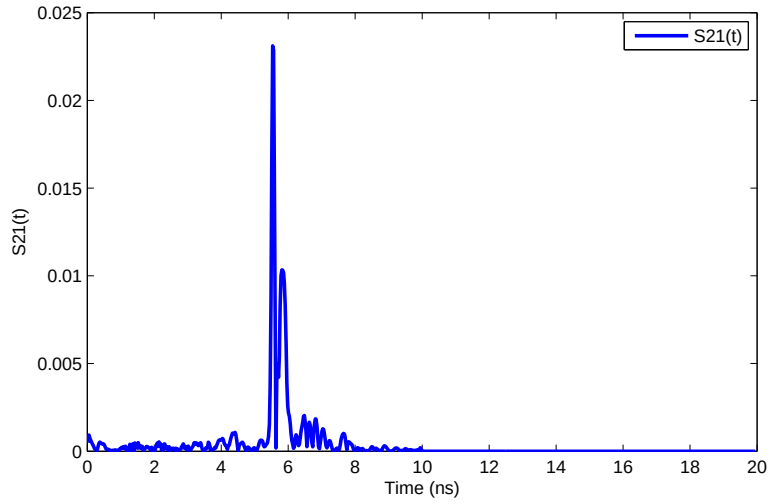


Figure 7.1 Time-domain response of S_{21}.

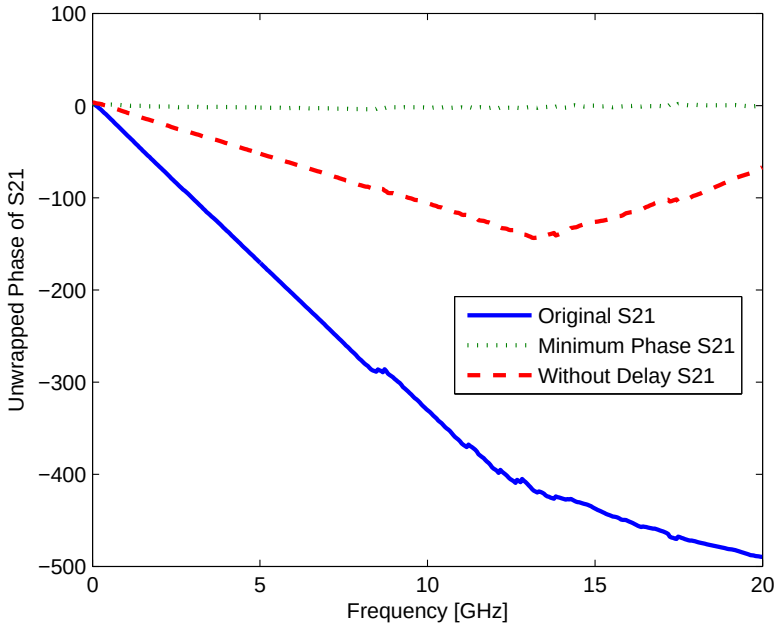


Figure 7.2 Unwrapped phase of the original, minimum-phase and without the delay of S_{21}.

is approximated using ZDVF with order of 72. Figures 7.3 and 7.4 show the comparison of magnitude and the unwrapped phase of the response without the delay and the ZDVF approximation. It shows a good agreement in the approximation after delay is extracted. For this specific example, the magnitude is very small at higher frequency range. Therefore, there is no need to assign starting poles in the higher frequency range. Figure 7.5 shows the initial poles used by ZDVF. Figure 7.6 shows the final pole distribution after two iterations.

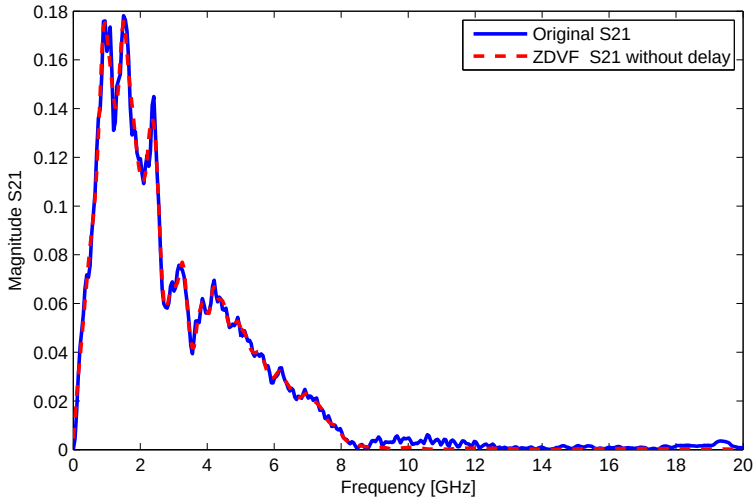


Figure 7.3 Magnitude response of the original and ZDVF approximation of without delay S_{21}.

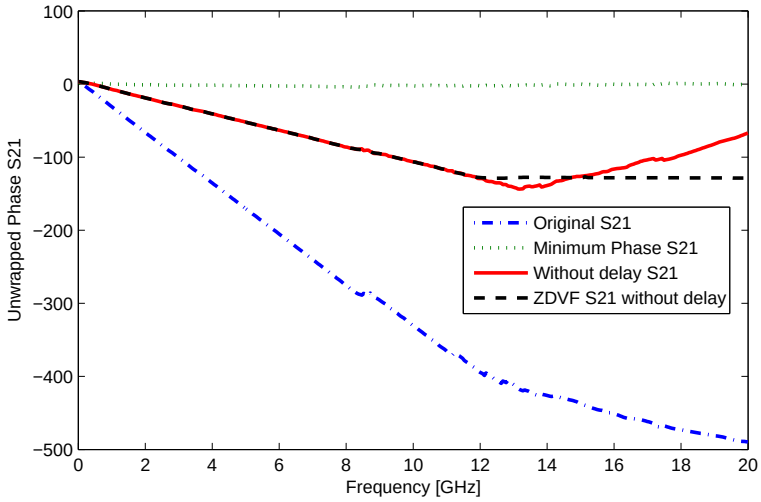


Figure 7.4 Unwrapped phase of the original, minimum-phase, response without delay and ZDVF approximation of without the delay S_{21}.

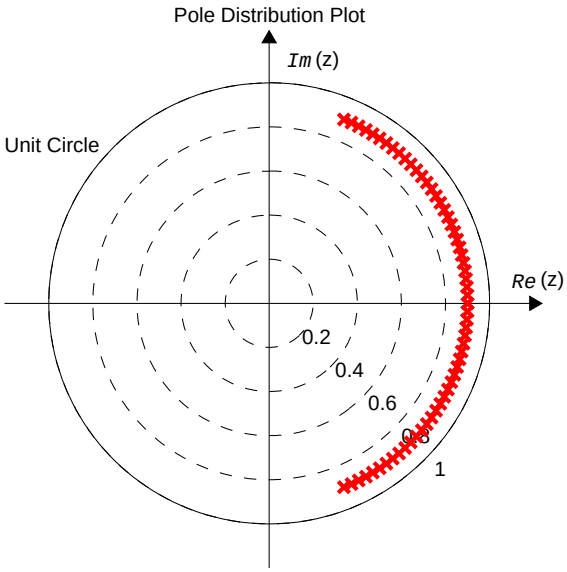


Figure 7.5 Starting poles distribution.

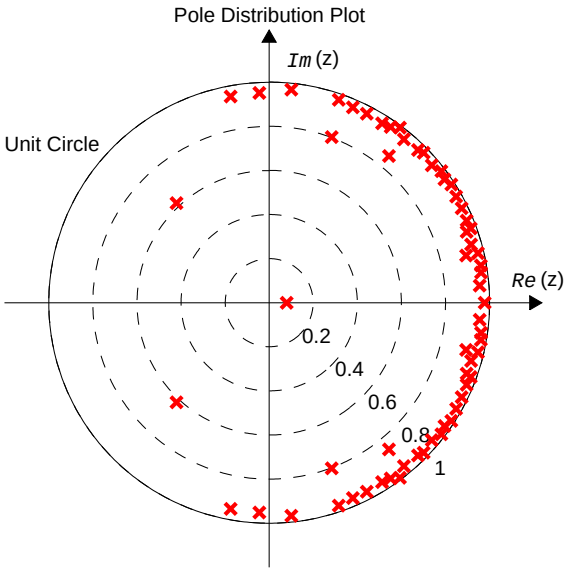


Figure 7.6 Final iteration poles distribution.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

The goal of this work is to construct accurate low-order macromodels of passive systems from frequency data for simulation of multiport networks. The macromodels need to be stable and passive to be used to high-speed time-domain circuit simulation of large networks. Using the constructed macromodels, the electromagnetic behavior of distributed networks can be successfully integrated into circuit simulators for design and analysis of a more complex system.

In Chapter 2, a review of model-order reduction techniques is presented. Padé approximations using the moment-matching technique and Krylov subspace-based methods, such as the Arnoldi method and the Lanczos method, are studied.

In Chapter 3, the basic requirement of broadband macromodels are discussed. The stability, causality and passivity properties of a macromodel is presented.

In Chapter 4, a survey of rational macromodel generation methods for passive subnetworks characterized by tabulated data is presented. Some of the common passivity enforcement techniques are discussed.

In Chapter 5, a combined method that uses rational interpolation is proposed to improve the initial poles and convergence of the vector fitting method in s-domain. The procedure takes advantage of the robustness of the rational interpolation method in

generating stable poles over partitioned frequency range and the numerical stability and fast convergence of the vector fitting method to construct high-order approximation over wide frequency range. The convergence of the standard vector fitting and proposed methods are compared. It is shown that the error decreases very fast and stagnates at high iterations.

In Chapter 6, a new method of approximating a rational function is proposed using a vector-fitting method in the z-domain (ZDVF). The method is an iterative technique based on pole relocation technique. It is shown that the iterative pole-relocation technique is the reformulation of Sanathanan-Koerner (SK) iteration using partial fractions basis. This has the advantage of faster convergence and better numerical stability compared to the s-domain vector-fitting method. The fast convergence behavior of this method reduces the overall macromodel generation time significantly. The convergence of ZDVF is illustrated by using various examples and by studying the movement of the poles with iteration. Also, an example is provided how ZDVF can be applied to system identification problem to identify a linear time-invariant system.

In Chapter 7, a delay extraction scheme is presented to reduce the order approximation of ZDVF. The methods is based on decomposition of LTI system response into minimum-phase and an all-pass system. An example is provided to show that the order reduction after the delay is removed.

8.2 Future Work

Even though the macrodeling techniques discussed results an accurate approximation of the response date, the order of approximation can be even reduced if preprocessing of

data is applied before approximation. These preprocessing of data includes dc extraction, enforcing causality and passivity of the response. In addition, the passivity enforcement in z-domain need to be studied. The current passivity enforcement method is limited by the nature of perturbation theory and therefore requires the passivity violation be relatively small. A better approach should be incorporated into the construction of the macromodel.

ZDVF shows a promising result to identify arbitrary LTI system. More work need to be done to apply ZDVF for infinite impulse response (IIR) filter design. In addition, ZDVF can be used for channel identification, inter symbol interference (ISI) prediction and jitter prediction of channels.

REFERENCES

[1] S. B. Goldberg, M. B. Steer, P. D. Franson, and J. S. Kasten, "Experimental electrical characterization of interconnects and discontinuities in high-speed digital systems," *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, vol. 14, no. 4, pp. 761-765, December 1991.

[2] M. Celik and A. C. Cangellaris, "Efficient transient simulation of lossy packaging interconnects using moment-matching techniques," *IEEE Transactions on Component, Hybrids, and Manufacturing Technology*, vol. 19, no. 1, pp. 64-73, February 1996.

[3] E. Chiprout and M. S. Nakhla, *Asymptotic Waveform Evaluation and Moment Matching of Interconnect Analysis*. Boston, MA: Kluwer Academic Publishers, 1994.

[4] P. Feldmann and R. W. Freund, "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, no. 5, pp. 639-649, May 1995.

[5] W. T. Beyene and J. E. Schutt-Ainé, "Efficient transient simulation of high-speed interconnects characterized by sample data," *IEEE Transactions on Component, Packaging and Manufacturing Technology*, vol. 21, pp. 105-114, February 1998.

[6] R. Gao, Y. S. Mekonnen, W. T. Beyene, and J. E. Schutt-Ainé, "Black-box modeling by rational function approximation," in *Proceedings of IEEE Workshop on Signal Propagation in Interconnects*, SPI- 2004, pp. 203-210.

[7] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, pp. 1052-1061, July 1999.

[8] B. Gustavsen, "Computer code for rational approximation of frequency dependent admittance matrices," *IEEE Transactions on Power Delivery*, vol. 17, pp. 1093-1098, Oct 2002.

[9] Y. S. Mekonnen, W. T. Beyene, and J. E. Schutt-Ainé, "Improved high-order approximation by combining rational interpolation with the vector fitting method," in 20^{th} *Annual Review of Progress in Applied Computational Electromagnetic*, Syracuse, NY, April 19-23, 2004, pp. 104-109.

[10] M. Celik and A. C. Cangellaris, "Simulation of dispersive multiconductor transmission lines by Padé approximation via the Lanczos process," *IEEE Transactions on Microwave Theory and Technology*, vol. 44, no. 12, pp. 2525-2535, December 1996.

[11] L. Pillage and R. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Transactions on Computer-Aided Analysis*, vol. 9, no. 4, pp. 352-366, April 1990.

[12] J. Bracken, V. Raghavan, and R. Rohrer, "Interconnect simulation with asymptotic waveform evaluation (AWE)," *IEEE Transactions on Circuits and Systems*, vol. 39, no. 11, pp. 869-878, November 1992.

[13] T. Tang and M. Nakhla, "Analysis of lossy multiconductor transmission lines using the asymptotic waveform evaluation technique," *IEEE Transactions on Microwave Theory and Technology*, vol. 39, no.12, pp. 2107-2116, December 1991.

[14] E. Chiprout and M. S. Nakhla, "Transient waveform estimation of high-speed MCM networks using complex frequency hopping," in *Proceedings of Multi-Chip Module Conference (MCMC)*, March 1993, pp. 134-139.

[15] E. Chiprout and M. S. Nakhla, "Analysis of interconnect networks using complex frequency hopping (CFH)," *IEEE Transactions on Computer-Aided Design*, vol. 14, no. 2, pp. 186-200, February 1995.

[16] K. Gallivan, E. Grimme, and P. Van Dooren, "Asymptotic waveform evaluation via Lanczos methods," *Applied Mathematics Letters*, vol. 7, pp. 75-80, 1994.

[17] K. Gallivan, E. Grimme, and P. Van Dooren, "Padé approximation of large-scale dynamic systems with Lanczos methods," in *Proceedings of* 33^{rd} *Conference on Decision and Control*, Lake Buena Vista, FL, December 1994, pp. 443-448.

[18] P. Gunupudi, M. Nakhla, and R. Achar, "Simulation of high-speed distributed interconnects using Krylov-space technique," *IEEE Transactions on Computer-Aided Design*, vol. 19, no. 7, pp. 799-808, July 2000.

[19] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*. Oxford, UK: Clarendon Press, 1965.

[20] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Upper Saddle River, NJ: Prentice Hall, 1999.

[21] B. Gustavsen and A. Semlyen, "Enforcing passivity for admittance matrices approximated by rational functions," *IEEE Transactions on Power Systems*, vol. 16, pp. 97-104, Feb. 2001.

[22] S. Grivet-Talocia, "Enforcing passivity of macromodels via spectral perturbation of the Hamiltonian matrices," in *Proceedings of IEEE Workshop on Signal Propagation in Interconnects*, SPI-2003, pp. 287-290.

[23] S. H. Min and M. Swaminathan, "Efficient construction of two-port passive macromodels for resonant networks," in *Proceedings of IEEE* 10^{th} *Topical Meeting on Electrical Performance of Electrical Packaging*, Oct. 2001, pp. 229-232.

[24] D. Saraswat, R. Achar, and M. Nakhla, "Enforcing passivity for rational function based macromodels of tabulated data," in *Proceedings of IEEE* 12^{th} *Topical Meeting on Electrical Performance of Electrical Packaging*, Oct. 2003, pp. 295-298.

[25] R. Gao, Y. S. Mekonnen, W. T. Beyene, and J. E. Schutt-Ainé, "Black-box modeling of passive systems by rational function approximation," *IEEE Transactions on Advanced Packaging*, vol. 28, no. 2, pp. 209-215, May 2005.

[26] W. T. Beyene, "Improving time-domain measurements with a network analyzer using a robust rational interpolation technique," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 3, pp. 500-508, 2001

[27] S. H. Min, "Automated construction of macromodels from frequency data for simulation of distributed interconnect networks," Ph.D dissertation, Georgia Institute of Technology, April 2004.

[28] K. L. Choi and M. Swaminathan, "Development of model libraries for embedded passives using network synthesis," *IEEE Transactions on Analog Digital Signal Processing*, vol. 47, no. 4, pp. 249-260, April 2000.

[29] J. Vlach, *Computerized Approximation and Synthesis of Linear Networks*. New York: John Wiley & Sons, Inc., 1969.

[30] W. H. Press, S. A. Teukolsky, V. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C++*. Cambridge, Cambridge University. Press, pp. 187-189, 2002.

[31] S. Kami, *Network Theory: Analysis and Synthesis*. Boston: Allyn and Bacon, 1966.

[32] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "A Convex programming approach to positive real rational approximation," in *Proceedings of IEEE/ACM International Conference on Computer Aided Design*, November 2001, pp. 245-251.

[33] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "A Convex programming approach for generating guaranteed passive approximations to tabulated frequency-data," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23, no. 2, pp. 293-301, Feb 2004.

[34] B. D. O. Anderson, "A system theory criterion for positive real matrices," *SIAM J. Control*, vol. 5, pp. 171-182, 1967.

[35] P. Delsarte, Y. Genin, and Y. Kamp, " On the role of the Nevanlinna-Pick problem in circuits and system theory," *Circuit Theory and Applications*, vol. 9, pp. 177-187, 1981.

[36] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "Passive constrained rational approximation algorithm using Nevanlinna-Pick interpolation," in *Proceedings of Design, Automation and Test in Europe Conference and Exibition*, March 2002, pp. 923-930.

[37] S. Boyd, V. Balakrishnan, and P. Kabamba, "A bisection method for computing the h_∞ norm of a transfer matrix and related problems," *Mathematics of Control, Signals and Systems*, vol 2, pp. 207-219, 1989.

[38] A. K. Sinha and J. Pal, "Simulation based reduced order modelling using clustering technique," *Computers and Electrical Engineering*, vol. 16, no. 3, pp. 159-169, 1990.

[39] W. T. Beyene, "Pole-clustering and rational interpolation for simplifying distributed systems," *IEEE Transactions on Circuits and Systems I*, vol. 46, no. 12, pp. 468-1472, Dec. 1999.

[40] W. T. Beyene, "Low-order rational approximation of interconnects using neural-network based pole-clustering techniques," in *IEEE International Symposium on Circuits and Systems*, New Orleans, May 27-30, 2007, pp. 1501-1504.

[41] A. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1999.

[42] T. Kohonen, "Self organized formualtion of topologically correct feature maps," *Biological Cybermatics*, vol. 43, pp. 59-69, 1982.

[43] C. T. Chen, *Linear System Theory and Design*, 3rd ed. New York: Oxford, 1999.

[44] Dirk Deschrijver, Bart Haegeman, and Tom Dhaene, "Orthogonal Vector Fitting: A Robust macromodeling tool for rational approximation of frequency domain responses," *IEEE Transactions on Advanced Packaging*, pp. 247-250, Feb. 2007.

[45] D. Deschrijver and T. Dhaene,"Broadband macromodelling of passive components using orthonormal vector fitting," *Electronic Letter*, vol. 41, no. 21, pp. 1160-1161, 2005.

[46] G. W. Stewart, *Introduction to Matrix Computations*. New York: Academic Press, 1973.

[47] J. Tokarzewski, "System zeros analysis via the Moore-Penrose pseudo inverse and the SVD of the first nonzero Markov parameter," *IEEE Transactions on Automatic Control*, vol. 43, no. 9, pp. 1285-1291, Sept. 1998.

[48] J. Tokarzewski, "Zeros and Kalman canonical form of MIMO LTI systems," in *Proceedings of European Control Conf.*, ECC, Brussels, Belgium, July 1-4, 1997, pp. 114-119.

[49] J. Tokarzewski, "A note on determination of zeros and zero directions by the Moore-Penrose pseudo inverse of the first nonzero Markov parameter," *UKACC International Conference on Control*, vol. 1, pp. 42-47, September 2-5, 1996.

[50] Y. S. Mekonnen, J. E. Schutt-Ainé, Jilin Tan, C. Kumar, and D. Milosevic, "Combining rational interpolation with the vector fitting method," in *Proceedings of IEEE* 14^{th} *Topical Meeting on Electrical Performance of Electrical Packaging*, Oct. 2005, pp. 51-54.

[51] Y. S. Mekonnen and J. E. Schutt-Ainé, "Broadband macromodeling of sampled frequency data using z-domain vector-fitting method," in *Proceedings of the* 11^{th} *IEEE Workshop on Signal Propagation on Interconnects*, Genova, Italy, May 13-16, 2007, pp. 45-48.

[52] Y. S. Mekonnen and J. E. Schutt-Ainé, "Fast macromodeling technique of sampled time/frequency data using z-domain vector-fitting method," in *Proceedings of IEEE* 16^{th} *Topical Meeting on Electrical Performance of Electrical Packaging*, Oct. 2007, pp. 47-50.

[53] W. T. Beyene and Chuck Yuan,"An accurate transient analysis of high-speed package interconnects using convolution technique," *Analog Integrated Circuits and Signal Processing*, Kluwer Academic Publishers, vol 34, pp. 107-120, 2003.

[54] W. T. Beyene,"Efficient simulation of chip-to-chip inteconnect system by combining waveform relaxation with reduced-order modeling methods," in *Proceedings of the* 52^{th} *Electronic Components and Technology Conf.*, San Diego, CA, May 27-29, 2002, pp. 1045-1050.

[55] W. T. Beyene, H. Shi, J. Feng, and C. Yuan,"Design and analysis methodologies of a 6.4 Gb/s memory interconnect system using conventional packaging and board technologies," in *Proceedings of* 54^{th} *Electronic Components and Technology Conf.*, Las Vegas, NV, June 1-4, 2004, pp. 1406-1411.

[56] W. T. Beyene, "Performance analysis and model-to-hardware correlation of multigigahertz parallel bus with transmit pre-emphasis equalization," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 11, pp. 3568-3577, Nov. 2005.

[57] W. T. Beyene, Hao Shi, J. Feng, and C. Yuan,"Electromagnetic modeling methodologies and desgin challenges of packages for 6.4 - 12.8 Gbps chip-to-chip interconnects," *IEEE Antennas and Propagation Society International Symposium*, pp. 3325-3328, June 20-25, 2004.

[58] MathWorks Inc., *MATLAB*^{TM} *Help Guide*, Release 2006.

[59] R. Mandrekar and M. Swaminathan, "Causality enforcement in transient simulation of passive networks through delay extraction," in *Proceedings of the* 9^{th} *IEEE Workshop on Signal Propagation on Interconnects*, Germany, May 10-13, 2005, pp 25-28.

[60] E. Ruehli, "Equivalent circuit models for three dimensional multiconductor systems," *IEEE Transactions on Microwave Theory Tech.*, vol. 22, pp. 216-221, March 1974.

AUTHOR'S BIOGRAPHY

Yidnekachew Sintayehu Mekonnen received the B.S. degree in electrical and computer engineering from the Addis Ababa University, Addis Ababa, Ethiopia, in 2000, and the M.S. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC), in 2004. From August 2002 to December 2007, he worked as a research assistant at the Center for Computational Electromagnetics and Electromagnetics Laboratory at UIUC. He worked as summer intern at Cadence Design System, Chelmsford, Massachusetts, in 2005, and at T.J. Watson Research Center IBM, Yorktown Heights, New York, in 2006. His areas of interest include signal integrity analysis, electrical package design, high-speed digital circuit design, and model-order reduction methods. He is a student member of IEEE and a member of Eta Kappa Nu.