Passive Macromodeling of Linear Subnetworks Characterized by Measured/ Simulated Data

by

Dharmendra Saraswat, B. E.,

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the the degree of
Master of Applied Sciences

Ottawa-Carleton Institute for Electrical Engineering
Department of Electronics
Faculty of Engineering
Carleton University
Ottawa, Ontario, Canada

© Dharmendra Saraswat, 2003

National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services

Acquisisitons et
services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-83480-8

Our file Notre référence

ISBN: 0-612-83480-8

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Canada

The undersigned recommend to the Faculty of Graduate Studies and Research

acceptance of this thesis,

“Passive Macromodeling of Linear Subnetworks

Characterized by Measured/Simulated Data”

submitted by Dharmendra Saraswat (B.E.) in partial fulfillment of the

requirements for the degree of Master of Applied Sciences

A handwritten signature in black ink, appearing to be 'M. Nakhla', written over a horizontal line.

Chairman, Department of Electronics

Professor Michel Nakhla

A handwritten signature in black ink, appearing to be 'R. Achar', written over a horizontal line.

Thesis Supervisor

Professor Ram Achar

A handwritten signature in black ink, appearing to be 'M. Nakhla', written over a horizontal line.

Thesis Co-Supervisor

Professor Michel Nakhla

Carleton University, April 2003

Abstract

As the rise times of digital signals drop into subnanosecond range, the effects of off-chip circuitry become important. It is becoming increasingly essential to model frequency-dependent signal integrity effects that can have considerable impact on the performance and functionality of a high-speed design. However, with the increasing frequency and complexity, it is not always possible to find an analytical model for such high-frequency passive components such as high-speed packages, vias, connectors, nonuniform transmission lines and also on-chip passive components, such as inductors and transformers. As a result, these passive components are generally characterized in a practical environment either through measurements or from the physical layout using rigorous full-wave electromagnetic simulations. In both cases, tabulated data based on S , Y , Z or T parameters is used to represent the given passive components. However, transient simulation of such frequency-dependent tabulated data in the presence of nonlinear devices to obtain a global electrical assessment is a CPU expensive process due to the mixed frequency/time problem. Prominent approaches to solve this difficulty are based on approximating the tabulated data through rational-functions and subsequently synthesizing a SPICE compatible macromodel from such an approximation. However, these approaches suffer from two major problems. Firstly, they suffer from numerical ill-conditioning for higher orders of approximations. Secondly, they do not guarantee the passivity of the macromodel. Passivity is an important property, because stable but non-passive models may lead to unstable systems when connected to other passive components.

This thesis proposes an algorithm to approximate the tabulated data accurately with higher order approximations over a wide frequency range, without suffering from the numerical ill-conditioning problem. The algorithm presents an efficient way of computing a common pole set for the network and obtaining the state space representation, suitable for linking to HSPICE. In addition, an efficient algorithm is presented to guarantee the passivity of the macromodel. Numerical examples are presented to demonstrate the validity and efficiency of the proposed algorithm.

Acknowledgments

I would like to express my gratitude and appreciation to Professor Ram Achar and Professor Michel Nakhla for their guidance, helpful suggestions and much needed encouragement. Their ever friendly manner, motivation in difficult moments, patience and good judgement made the journey through this thesis an enjoyable and unforgettable experience. I feel privileged to have them as my thesis supervisors.

I would like to acknowledge and thank several of my colleagues (some of them are now Professors): A. Dounavis for all the useful discussions on synthesis and passivity, as well as proof reading my thesis, Dr. R. Khazaka and E. Gad for answering my linear algebra and FRAME MAKER questions, Dr. P. Gunupudi, K. Harb and P. Agarwal for their support when I needed. I extend my thanks to my brother-in-law Sanjay, Deepak and Kanika for their support and making my stay at Ottawa pleasant. I would also like to express my gratitude to the professors, staff and students at the department of electronics for making my work an enjoyable experience.

I leave a special note of appreciation for my wife, Preeti (Dolly). Without her loving support, encouragement and patience, working on this thesis would have been hard. My final thoughts are with my parents, my sister late Smt. Pushplata, my brother Jitendra, sister-in-law Vineeta and niece Shipi, for patient backing in everything, I am grateful to you. I believe your patience finally paid off.

Table of Contents

<i>Abstract</i>	<i>i</i>
<i>Acknowledgments</i>	<i>ii</i>
<i>Glossary of Terms</i>	<i>ix</i>
 CHAPTER 1 Introduction	 1
1.1 Background and Motivation	1
1.2 Contributions	6
1.3 Organization of the Thesis	7
 CHAPTER 2 Literature Survey	 8
2.1 Convolution Based Methods	8
2.1.1 Analysis of Linear Passive Network with Arbitrary Nonlinear Terminal Networks	9
2.2 Rational Function Based Approaches	12
2.2.1 Pole-Residue Formulation for Transient Simulation	12
2.2.2 Interconnect Simulation Using Order Reduction and S Parameters	20
2.2.3 Rational Function Approximation Via Nonlinear Formulation	27
2.2.4 Rational Function Approximation Using Nevanlinna- Pick Interpolation	33
 CHAPTER 3 Proposed Technique For Pole Computation of Multiport Networks	 37
3.1 Vector Fitting by Pole Relocation	38
3.1.1 Evaluation of Poles of $f(s)$	44
3.1.2 Selection of Initial Set of Poles	46

3.2	Difficulties with Pole Computation for Multiport Tabulated Subnetworks	47
3.3	Proposed Multiport-Pole Computation Algorithm.....	47
3.4	Advantages of Proposed Multiport Pole-Computation over Conventional Techniques	54
CHAPTER 4	Passive Macromodeling of Linear Subnetworks Characterized by Tabulated Data.....	55
4.1	Review of Passivity Properties	55
4.2	Problem Formulation.....	57
4.2.1	Passivity Issues Related to Residue Computation	58
4.3	Proposed Passive Macromodeling Algorithm for Tabulated Data	59
4.4	Macromodel Synthesis.....	65
4.5	Proposed Passivity Check.....	66
CHAPTER 5	Time-Domain Macromodel Synthesis and Unified Transient Analysis.....	68
5.1	Macromodel Synthesis.....	68
5.1.1	Including Complex Poles in the State Matrix	71
5.2	General Formulation of Network Equations	73
5.3	Unified Transient Simulation	77
5.3.1	Conversion of Macromodels to Equivalent Subcircuits. .	78
5.3.2	Transient Simulation of Rational Functions Using HSPICE Laplace elements	80

CHAPTER 6	Computational Results	82
CHAPTER 7	Conclusions and Future Work	110
7.1	Summary	110
7.2	Future Research	112
<i>References</i>		<i>113</i>
<i>Appendix A</i>		
<i>Appendix B</i>		

List of Figures

Fig. 1.1.	High-speed interconnect effects.	2
Fig. 1.2.	Transient response of a stable but non-passive macromodel with passive terminations.	4
Fig. 1.3.	Illustration of significance of passivity.	5
Fig. 2.1.	M-port network.	9
Fig. 2.2.	M-port network.	21
Fig. 4.1.	Illustration of the general behavior of $z^*[Re(Y(j\omega))]z$ when using the proposed set of constraints.	65
Fig. 5.1.	Illustration of time-domain macromodel synthesis from pole-residue formulation.	69
Fig. 5.2.	Lumped network ϕ containing tabulated subnetwork π	73
Fig. 5.3.	Example circuit with lumped linear, nonlinear & measured devices.	75
Fig. 5.4.	Inclusion of tabulated network in overall transient simulation. .	78
Fig. 5.5.	Illustration of equivalent subcircuit generation from macro-models.	80
Fig. 5.6.	Example (Laplace element).	81
Fig. 6.1.	Nonlinear circuit containing RLC lumped network.	83
Fig. 6.2.	Frequency responses for Example 1.	85
Fig. 6.3.	Eigenvalues of $real(Y(s))$ (Example 1).	86
Fig. 6.4.	Time-domain responses for Example 1.	87
Fig. 6.5.	2-Port transmission line circuit (Example 2).	89
Fig. 6.6.	Frequency responses (Example 2).	90
Fig. 6.7.	Eigenvalues of $real(Y(s))$ (Example 2).	91
Fig. 6.8.	Time-domain responses (Example 2).	92
Fig. 6.9.	3-port transmission line circuit (Example 3).	95
Fig. 6.10.	Frequency responses (magnitude) for Example 3.	96

Fig. 6.11.	Frequency responses (real part) for Example 3	97
Fig. 6.12.	Illustration of passivity violation with unconstrained fitting . . .	98
Fig. 6.13.	Passivity preservation using the proposed algorithm	99
Fig. 6.14.	Passivity proof for the proposed algorithm	100
Fig. 6.15.	Transient results for Example 3.	101
Fig. 6.16.	Transient results for Example 3.	102
Fig. 6.17.	Frequency responses (magnitude) for Example 3.	103
Fig. 6.18.	Frequency responses (real part) for Example 3	104
Fig. 6.19.	Illustration of passivity violation with unconstrained fitting. . .	105
Fig. 6.20.	Passivity preservation using the proposed algorithm	106
Fig. 6.21.	Passivity proof for the proposed algorithm	107
Fig. 6.22.	Transient results.	108
Fig. 6.23.	Transient results.	109

List of Tables

Table. 6.1.	Poles, residues and direct coupling constants obtained using proposed method (Example 1)	84
-------------	---	----

Glossary of Terms

AWE	Asymptotic Waveform Evaluation
CAD	Computer Aided Design
CPU	Central Processing Unit
FD	Frequency Dependent
LMS	Least-Mean Squares
LU	Lower-Upper
KCL	Kirchoff's Current Law
MCM	Multichip Module
MM	Macromodel
MNA	Modified Nodal Analysis
PCB	Printed Circuit Board
PR	Positive Real
SPICE	Simulation Program with Integrated Circuits Emphasis
TF	Transfer Function

1.1 Background and Motivation

With the rapid developments in the VLSI technology, design and CAD techniques, at both the chip and package level, the central processor cycle times are reaching the vicinity of 1ns and communication switches are being designed to transmit data that have bit rates faster than 1Gb/s. The ever increasing quest for high-speed applications is placing higher demands on the performance of interconnect networks and has highlighted the previously negligible effects of interconnects (Fig. 1.1), such as ringing, signal delay, distortion, reflections and crosstalk [1] - [7], [55] - [86]. In addition, the trend in the VLSI industry towards miniature designs, low power consumption and increased integration of analog circuits with digital blocks has further complicated the issue of signal integrity analysis. If not considered during the design stage, these interconnect effects can cause logic glitches which render a fabricated digital circuit inoperable, or they can distort an analog signal such that it fails to meet specifications. Since extra iterations in the design cycle are costly, accurate prediction of these effects is a necessity in high-speed designs. Hence it becomes extremely important for designers to simulate the entire design along with interconnect subcircuits as efficiently as possible while retaining the accuracy of simulation [8] - [19].

Fig. 1.1. High-speed interconnect effects

Interconnections can be from various levels of design hierarchy, such as on-chip, packaging structures, multichip modules, printed circuit boards and backplanes. Depending on the operating frequency and nature of the structure, these interconnects are typically modeled as *lumped components* (RC or RLC) [1] - [3], *distributed elements* (frequency independent/dependent RLCG parameters, lossy, coupled) [1] - [3], [19].

However, with the increasing complexity and operating frequency, it is not always possible to find an analytical model for the interconnect network [30], [42]. For instance, interconnects in chip packages are usually nonuniform due to high circuit density, complex shapes and geometrical constraints. In such cases, it may not be possible to accurately model high-frequency effects using analytical models and they are generally characterized by their terminal behaviours described in terms of frequency sampled S, Y, Z or T parameters. The sampled data can be obtained either from measurements or full-wave simulation (*henceforth referred to as tabulated data in this thesis*). This has led to an intense

research in the area of characterization and simulation of interconnect circuitry described by S, Y, Z or T parameters, within a nonlinear environment. However, transient simulation of such frequency-dependent tabulated parameters in the presence of nonlinear devices has two major bottlenecks. The first one is, how to include the above parameters (which are described in the frequency-domain) in an overall simulation with other nonlinear devices (which are represented in the time-domain). This is typically called the mixed frequency/time problem. In addition, discrete measured data does not have adequate representation for simulation purposes.

There have been several attempts in the literature to address the above issue. These approaches can be broadly classified into two categories. In the first category, transient simulation is performed based on the traditional convolution process [20] - [22]. Here the frequency sampled data is first converted to time-domain using the inverse Fourier transform (IFFT) and then convoluted with the transient responses of both the nonlinear load and the input excitation. However, 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 IFFT. Although this method gives a reliable solution, it is not fast enough to provide results for a large multiport system in a reasonable period of time. Approaches in the second category are based on obtaining a reduced-order model [23] - [25], [28], [29] for the frequency sampled data and performing the transient analysis either by using recursive convolution [13], [34] or converting the frequency-domain model into a set of ordinary differential equations [30], [42]. Derivation of differential equations from reduced-order frequency-domain models is referred to as *macromodel synthesis*. However, there are two main difficulties associated with this category of approaches: 1) they suffer from the

problem of ill-conditioning in capturing the broad-band frequency spectrum of interest with higher order of approximations (this may lead to inaccuracy in computed models), and 2) there is no guarantee of the passivity of the resulting macromodel.

However, it is demonstrated in the literature [35] - [41], that in addition to macromodel stability, macromodel passivity is required to guarantee the stability of the overall circuit during unified transient simulation of the macromodel and rest of the circuitry. Passivity implies that a network cannot generate more energy than it absorbs, and no passive termination of the network will cause the system to go unstable. The loss of passivity can be a serious problem because transient simulations of the macromodel may encounter artificial oscillations. This is illustrated in Fig. 1.2, which represents the transient response of a stable but non-passive macromodel, when connected to an external load of 50Ω .

Fig. 1.2. Transient response of a stable but non-passive macromodel with passive terminations

This is because macromodels that are only stable but not passive, can produce unstable networks when connected to other passive loads. On the other hand, a passive macromodel, when terminated with any arbitrary passive load always guarantees the stability of the overall resulting network. To illustrate the point, consider a simple, single-port second-order macromodel shown in Fig. 1.3(a). The macromodel is stable but not passive. When this macromodel is terminated with the passive load (Fig. 1.3(b)), the overall network becomes unstable.

STABLE (Macromodel Alone)

UNSTABLE (Macromodel + Passive Load)

Fig. 1.3. Illustration of Significance of Passivity

1.2 Contributions

The subject of this thesis is the modeling and simulation of multiport high-speed interconnect networks characterized by frequency-domain sampled data, in the presence of nonlinear elements. The work presented here provides an easy means to handle high-speed interconnect circuits described by tabulated data, in industrial grade simulators such as SPICE. The main contributions of this thesis are as follows.

- 1) An efficient multiport algorithm is developed for macromodeling of high-speed interconnect subnetworks (chapter 4). The proposed algorithm generates time-domain macromodels, which facilitates fast and accurate simulation of interconnect subnetworks in the presence of nonlinear elements. In addition, the new algorithm provides a scheme to preserve the broadband frequency spectrum of the original data up to the maximum frequency of interest [42], [45], [47], [48].
 - 2) A new residue computation algorithm has been developed which helps to preserve the passivity of the macromodel. For this purpose, a new set of linear passivity conforming constraints are developed [42], [47].
 - 3) An efficient passivity verification algorithm has been developed to check for the passivity of the resulting macromodel (chapter 4) [43], [45], [48].
 - 4) An efficient technique for macromodel synthesis from frequency-domain reduced-order models is presented (chapter 5). The method also discusses the inclusion of complex poles during state-space synthesis [47].
-

1.3 Organization of the Thesis

This thesis is organized as follows. In chapter 2, a literature survey is done, providing a concise explanation of leading algorithms in the area of macromodeling of passive subnetworks characterized by tabulated data. This chapter also provides some background on the issue of passivity of the macromodel, as handled by some of the current approaches, and their limitations. Chapter 3 presents an efficient pole computation algorithm for identifying the poles of the matrix-transfer function. Chapter 4 presents the new residue computation algorithm and also the associated linear constraints to preserve the passivity of the multiport macromodel. In addition, an efficient method for passivity verification is presented. Chapter 5 outlines the method for converting rational approximation models into ordinary differential equations and subsequently linking them to SPICE for a unified transient simulation. Numerical results are presented in chapter 6, while conclusions and proposed future research are presented in chapter 7.

CHAPTER 2 *Literature Survey*

As discussed in chapter 1, with the ever increasing operating frequency and complexity of interconnect structures, it is not always possible to find their analytical model. In such cases, they are generally described by frequency-domain tabulated data obtained from terminal measurements or full-wave analysis. This chapter reviews some of the prominent techniques to include the frequency sampled data in an overall circuit simulation.

2.1 *Convolution Based Methods*

In this category of approaches, transient simulation is performed based on the traditional convolution process [20]-[22], wherein the frequency sampled data is first converted to time-domain using inverse Fourier transform (IFFT) and then convoluted with the input excitation. To understand the idea behind the approach, consider the problem formulation [21] given below.

2.1.1 Analysis of Linear Passive Network with Arbitrary Nonlinear Terminal Networks

Problem Formulation

Consider a linear, passive M-port network of Fig. 2.1. Suppose that an ideal voltage source $v_j(t)$ is connected at port j , while other ports are short circuited, then the currents at all the ports can be represented as

$$I_k(\omega) = Y_{kj}(\omega)V_j(\omega), \quad k = 1, 2, \dots, M \quad (2.1)$$

Fig. 2.1. M-Port Network

where, $V_j(\omega)$ is the Fourier transform of $v_j(t)$ and $Y_{kj}(\omega)$ are the network Y parameters relating port j to the port k , while ω is the angular frequency. For a specific case of $v_j(t)$ being a unit delta function, we will have $V_j(\omega) = 1$, which is independent of frequency. In such a case, the port currents in time-domain can be obtained as

$$i_k(t) = i_{gkj}(t) = F^{-1}\{Y_{kj}(\omega)\} \quad (2.2)$$

where F^{-1} denotes the inverse Fourier transform. These currents $i_{gkj}(t)$ are referred to as the “Green’s functions” of the network. When $v_j(t)$ is an arbitrary function, the port currents can be obtained as

$$i_k(t) = F^{-1}\{Y_{kj}(\omega)V_j(\omega)\} = i_{gkj}(t) * v_j(t) = \int_0^t i_{gkj}(t-\tau)v_j(\tau)d\tau \quad (2.3)$$

where ‘*’ denotes the convolution, and is represented by its integral form assuming that all the excitations begin after $t = 0$. By the superposition principle for linear networks, the current at port k in the presence of applied voltages at all the ports can be written as

$$i_k(t) = \sum_{j=1}^M \int_0^t i_{gkj}(t-\tau)v_j(\tau)d\tau \quad (2.4)$$

The voltages at ports with non-linear terminations may be evaluated by substituting $i_k(t)$ in the specific voltage current relationship defined by the terminating nonlinear network. For instance, for a diode connected at port k , the voltage-current relationship can be expressed as

$$i_k(t) = I_s \left(\exp\left(\frac{v_k(t)}{V_T}\right) - 1 \right) \quad (2.5)$$

Notice that the network Y -parameters $Y_{kj}(\omega)$ are computed at discrete frequency points, by connecting an ideal impulse (delta) function at a port, short-circuiting all other ports and then finding the currents at each port. Later, the inverse Fourier transform of $Y_{kj}(\omega)$ is evaluated as indicated in (2.2).

Transient Simulation

In order to prepare (2.4) for practical implementation, the integrations are replaced by summations. Thus (2.4) can be written as

$$i_k(t_q) = \sum_{j=1}^M \sum_{p=0}^q i_{gkj}(t_q - t_p) v_j(t_p) \Delta t, \quad k = 1, 2, \dots, M \quad (2.6)$$

where Δt represents the time step and t_q, t_p denote the time instances $q\Delta t, p\Delta t$ respectively.

As pointed out in [20], [22], the frequency-domain analysis is done numerically, at a finite number of discrete frequencies. In the time-domain, the green's function i_{gkj} in (2.2) must also be discretized and limited to finite duration. In addition, this function must be convolved with port voltages, which also has to be done numerically. Convolution being an expensive process, it is desired that the number of samples of i_{gkj} is kept as low as possible. This can be a problem if the response of the network with terminations in place, spans a time interval greater than a few transit times of the network (transit time of a network is the time delay associated with the signal propagation in the network). To overcome this problem, for the case of transmission lines, it has been suggested in [20], [21], to terminate the lines with well-matched networks while evaluating the transfer functions $Y_{kj}(\omega)$. This reduces the duration of function i_{gkj} to only a few line

transit times. As a result, the convolution in (2.6) can be made less expensive. The details of this can be found in [21], [22].

2.2 *Rational Function Based Approaches*

The following subsections describe the modeling approaches based on rational-function approximation of tabulated data. It is to be noted that the approximation based on just the polynomials do not accurately represent the behaviour (of electrical networks) in the vicinity of poles. In contrast, rational functions do capture the response of networks very well, including the region around the poles.

2.2.1 Pole-Residue Formulation for Transient Simulation

The problem formulation of this approach consists of two steps [24]:

STEP1: Rational-Function Approximation

The method starts with approximating the tabulated data of the admittance matrix Y over a specified frequency range f_{max} . In this step, a rational-function approximation is obtained for each of the entries of admittance matrix Y . For instance, a specific $(k, l)^{th}$ admittance matrix element can be written as $Y_{kl}(s) = A(s)/B(s)$, where $s = j\omega$. Here, $A(s)$ is the numerator polynomial while $B(s)$ is the denominator polynomial. The form of the approximation is essentially as follows

$$Y_{kl}(s) = (Y_R + jY_I)_{kl} = \left\{ \frac{a_0 + a_1s + a_2s^2 + a_3s^3 + \dots + a_{n-1}s^{n-1} + a_ns^n}{b_0 + b_1s + b_2s^2 + b_3s^3 + \dots + b_{n-1}s^{n-1} + b_ns^n} \right\}_{kl} \quad (2.7)$$

where n is the order of the approximation, Y_R and Y_I represent the real and imaginary part of $Y_{kl}(s)$ respectively. The coefficients are normalized so that b_n is equal to 1.

Both the sides of (2.7) are multiplied by $B(s)$ and the result is separated into real and imaginary parts. If n is odd, the result is

$$a_0 + a_2s^2 + \dots + a_{n-1}s^{n-1} = Y_Rb_0 + jY_Ib_1 + Y_Rb_2s^2 + \dots + Y_Rb_{n-1}s^{n-1} + jY_Ib_ns^n \quad (2.8)$$

for real part and

$$a_1s + a_3s^3 + \dots + a_ns^n = jY_Ib_0 + Y_Rb_1 + jY_Ib_2s^2 + \dots + jY_Ib_{n-1}s^{n-1} + Y_Rb_ns^n \quad (2.9)$$

for the imaginary part. It is assumed in these relations that the coefficients $(a_0, a_1, \dots, a_n, b_0, b_1, \dots, b_n)$ are purely real. If n is even, then the result is

$$a_0 + a_2s^2 + \dots + a_ns^n = Y_Rb_0 + jY_Ib_1 + Y_Rb_2s^2 + \dots + jY_Ib_{n-1}s^{n-1} + Y_Rb_ns^n \quad (2.10)$$

for the real part and

$$a_1s + a_3s^3 + \dots + a_{n-1}s^{n-1} = jY_Ib_0 + Y_Rb_1 + jY_Ib_2s^2 + \dots + Y_Rb_{n-1}s^{n-1} + jY_Ib_ns^n \quad (2.11)$$

for the imaginary part. In the next step, depending on whether n is odd or even, (2.8)-(2.9) or (2.10)-(2.11) are formulated as a set of matrix equations of the form $AX = b$, over a specified frequency range. For instance, when n is even, (2.10) and (2.11) can be written as

$$\begin{bmatrix}
 1 & -\omega_1^2 & \omega_1^4 & \dots & \omega_1^n (-1)^{\frac{n}{2}} & -Y_{1R} & Y_{1I} \omega_1 & Y_{1R} \omega_1^2 & \dots & (-1)^{\frac{n}{2}+1} Y_{1I} \omega_1^{n-1} \\
 1 & -\omega_2^2 & \omega_2^4 & \dots & \omega_2^n (-1)^{\frac{n}{2}} & -Y_{2R} & Y_{2I} \omega_2 & Y_{2R} \omega_2^2 & \dots & (-1)^{\frac{n}{2}+1} Y_{2I} \omega_2^{n-1} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 1 & -\omega_D^2 & \omega_D^4 & \dots & \omega_D^n (-1)^{\frac{n}{2}} & -Y_{DR} & Y_{DI} \omega_D & Y_{DR} \omega_D^2 & \dots & (-1)^{\frac{n}{2}+1} Y_{DI} \omega_D^{n-1}
 \end{bmatrix}
 \begin{bmatrix}
 a_0 \\
 a_2 \\
 a_4 \\
 \vdots \\
 a_n \\
 b_0 \\
 b_1 \\
 b_2 \\
 \vdots \\
 b_{n-1}
 \end{bmatrix}
 =
 \begin{bmatrix}
 (-1)^{\frac{n}{2}} Y_{1R} \omega_1^n \\
 (-1)^{\frac{n}{2}} Y_{2R} \omega_2^n \\
 \vdots \\
 (-1)^{\frac{n}{2}} Y_{DR} \omega_D^n
 \end{bmatrix}
 \quad (2.12)$$

and

$$\begin{bmatrix}
 \omega_1 & -\omega_1^3 & \dots & \omega_1^{n-1} (-1)^{\frac{n}{2}+1} & -Y_{1I} & -Y_{1R} \omega_1 & Y_{1I} \omega_1^2 & \dots & Y_{1R} \omega_1^{n-1} (-1)^{\frac{n}{2}} \\
 \omega_2 & -\omega_2^3 & \dots & \omega_2^{n-1} (-1)^{\frac{n}{2}+1} & -Y_{2I} & -Y_{2R} \omega_2 & Y_{2I} \omega_2^2 & \dots & Y_{2R} \omega_2^{n-1} (-1)^{\frac{n}{2}} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 \omega_D & -\omega_D^3 & \dots & \omega_D^{n-1} (-1)^{\frac{n}{2}+1} & -Y_{DI} & -Y_{DR} \omega_D & Y_{DI} \omega_D^2 & \dots & Y_{DR} \omega_D^{n-1} (-1)^{\frac{n}{2}}
 \end{bmatrix}
 \begin{bmatrix}
 a_1 \\
 a_3 \\
 \vdots \\
 a_{n-1} \\
 b_0 \\
 b_1 \\
 b_2 \\
 \vdots \\
 b_{n-1}
 \end{bmatrix}
 =
 \begin{bmatrix}
 (-1)^{\frac{n}{2}} Y_{1I} \omega_1^n \\
 (-1)^{\frac{n}{2}} Y_{2I} \omega_2^n \\
 \vdots \\
 (-1)^{\frac{n}{2}} Y_{DI} \omega_D^n
 \end{bmatrix}
 \quad (2.13)$$

respectively. Here Y_{qR} is the real part of Y_{kl} and Y_{qI} is the imaginary part of Y_{kl} at the radian frequency ω_q , and D is the number of frequency samples. Linear system of equations (2.12) and (2.13) can be written in the form $\mathbf{AX} = \mathbf{b}$ and solved for unknown coefficients $(a_0, a_1, \dots, a_n, b_0, b_1, \dots, b_{n-1})$. It is to be noted that the number of frequency samples D must be greater than or equal to the number of columns of $\mathbf{A}$. In [24] only (2.12) is used while formulating the least square problem of $\mathbf{AX} = \mathbf{b}$ and the Householder algorithm is used to solve this

overdetermined problem. As it is evident, the matrix in (2.12) becomes very ill-conditioned as the order of approximation n increases. To address this issue, the radian frequency ω is normalized before formulating (2.12). The normalization factor ω_{norm} is calculated as

$$\omega_{norm} = \sqrt{\omega_{min} \cdot \omega_{max}} \quad (2.14)$$

With the solution thus obtained for X , the coefficients of the numerator and the denominator polynomials are known. At this stage of the algorithm, only the denominator coefficients are needed. The denominator polynomial is factored to obtain the poles of the transfer function. Any poles found in the right half side of complex plane 's' are ignored. Only the stable poles, i.e. the poles on the left hand side are considered. These poles are now de-normalized by multiplying all of them by normalization factor ω_{norm} from (2.14).

STEP2: Pole-Residue Formulation

In this step, the transfer function Y_{kl} is expressed in terms of pole-residue formulation

$$Y_{kl}(s) = c + \sum_{i=1}^n \frac{r_i}{s - p_i} \quad (2.15)$$

where, n is the total number of poles, c is a constant, p_i are the poles and r_i are the corresponding residues. With the poles p_i calculated from Step 1, (2.15) can be written at various frequency points to form a matrix least square problem $AX = b$, where X is the vector of unknowns. The unknowns here are

$(c, r_i, i \in 1, 2, \dots, n)$. The exact form of the matrix equation will depend on whether the poles are real or complex. The real part of (2.15) can be expressed as

$$Re\{Y_{kl}(s)\} = c + \sum_{i=1}^{n_R} r_i Re\left\{\frac{1}{s-p_i}\right\} + \sum_{i=n_R+1}^{n_R+n_C} Re\left\{\frac{r_i}{\{s-p_i\}} + \frac{r_i^*}{\{s-p_i^*\}}\right\} \quad (2.16)$$

and the imaginary part as

$$Im\{Y_{kl}(s)\} = \sum_{i=1}^{n_R} r_i Im\left\{\frac{1}{s-p_i}\right\} + \sum_{i=n_R+1}^{n_R+n_C} Im\left\{\frac{r_i}{\{s-p_i\}} + \frac{r_i^*}{\{s-p_i^*\}}\right\} \quad (2.17)$$

where n_R is the total number of real poles, and n_C is the total number of pairs of complex conjugate poles. By utilizing the fact that complex conjugate poles have complex conjugate residues, (2.16) can be further expressed as

$$\begin{aligned} Re\{Y_{kl}(s)\} = c + \sum_{i=1}^{n_R} r_i Re\left\{\frac{1}{s-p_i}\right\} + \sum_{i=n_R+1}^{n_R+n_C} Re\{r_i\} Re\left\{\frac{1}{s-p_i} + \frac{1}{s-p_i^*}\right\} \\ - \sum_{i=n_R+1}^{n_R+n_C} Im\{r_i\} Im\left\{\frac{1}{s-p_i} - \frac{1}{s-p_i^*}\right\} \end{aligned} \quad (2.18)$$

Similarly, (2.17) can be expressed as

$$\begin{aligned} Im\{Y_{kl}(s)\} = \sum_{i=1}^{n_R} r_i Im\left\{\frac{1}{s-p_i}\right\} + \sum_{i=n_R+1}^{n_R+n_C} Re\{r_i\} Im\left\{\frac{1}{s-p_i} + \frac{1}{s-p_i^*}\right\} \\ + \sum_{i=n_R+1}^{n_R+n_C} Im\{r_i\} Re\left\{\frac{1}{s-p_i} - \frac{1}{s-p_i^*}\right\} \end{aligned} \quad (2.19)$$

Thus, by recasting (2.15) as (2.18) and (2.19) and then combining them into a single system of matrix equations of the form $\mathbf{AX} = \mathbf{b}$, the unknown residues r_i

and the constant c are evaluated. The form of the above matrix equation varies depending on the number of real poles and the number of complex conjugate poles. An example of the matrix equation that results from this procedure for a third order system with one real pole and one complex conjugate pair, is as follows:

$$\begin{bmatrix}
 1 & \operatorname{Re}\left\{\frac{1}{s_1-p_1}\right\} & \operatorname{Re}\left\{\frac{1}{s_1-p_2} + \frac{1}{s_1-p_2^*}\right\} & -\operatorname{Im}\left\{\frac{1}{s_1-p_2} - \frac{1}{s_1-p_2^*}\right\} \\
 1 & \operatorname{Re}\left\{\frac{1}{s_2-p_1}\right\} & \operatorname{Re}\left\{\frac{1}{s_2-p_2} + \frac{1}{s_2-p_2^*}\right\} & -\operatorname{Im}\left\{\frac{1}{s_2-p_2} - \frac{1}{s_2-p_2^*}\right\} \\
 \vdots & \vdots & \vdots & \vdots \\
 1 & \operatorname{Re}\left\{\frac{1}{s_D-p_1}\right\} & \operatorname{Re}\left\{\frac{1}{s_D-p_2} + \frac{1}{s_D-p_2^*}\right\} & -\operatorname{Im}\left\{\frac{1}{s_D-p_2} - \frac{1}{s_D-p_2^*}\right\} \\
 0 & \operatorname{Im}\left\{\frac{1}{s_1-p_1}\right\} & \operatorname{Im}\left\{\frac{1}{s_1-p_2} + \frac{1}{s_1-p_2^*}\right\} & \operatorname{Re}\left\{\frac{1}{s_1-p_2} - \frac{1}{s_1-p_2^*}\right\} \\
 0 & \operatorname{Im}\left\{\frac{1}{s_2-p_1}\right\} & \operatorname{Im}\left\{\frac{1}{s_2-p_2} + \frac{1}{s_2-p_2^*}\right\} & \operatorname{Re}\left\{\frac{1}{s_2-p_2} - \frac{1}{s_2-p_2^*}\right\} \\
 \vdots & \vdots & \vdots & \vdots \\
 0 & \operatorname{Im}\left\{\frac{1}{s_D-p_1}\right\} & \operatorname{Im}\left\{\frac{1}{s_D-p_2} + \frac{1}{s_D-p_2^*}\right\} & \operatorname{Re}\left\{\frac{1}{s_D-p_2} - \frac{1}{s_D-p_2^*}\right\}
 \end{bmatrix}
 \begin{bmatrix}
 c \\
 r_1 \\
 \operatorname{Re}\{r_2\} \\
 \operatorname{Im}\{r_2\}
 \end{bmatrix}
 =
 \begin{bmatrix}
 \operatorname{Re}\{Y_1\} \\
 \operatorname{Re}\{Y_2\} \\
 \vdots \\
 \operatorname{Re}\{Y_D\} \\
 \operatorname{Im}\{Y_1\} \\
 \operatorname{Im}\{Y_2\} \\
 \vdots \\
 \operatorname{Im}\{Y_D\}
 \end{bmatrix}
 \quad (2.20)$$

Here D is the total number of frequency points. Once again, Householder least square solver is used to solve the above matrix equation. It is to be noted that no frequency normalization is required in this case. After evaluating (2.20), the rational approximation of the transfer function (2.15) is known. This entire process is repeated for each element of admittance matrix Y . The above algorithm was later modified [25] by including a mechanism of handling lossless networks,

and also ensuring common pole-set for the admittance matrix. Once the model is generated in the form of the pole-residue pairs, a special modified version of the SPICE is used to perform transient simulations as outlined in [26].

Passivity Check

Several passivity tests are proposed depending on the nature of the network, lossy or lossless. For lossy networks, the passivity is checked by performing the Pivot Test [27]. This test involves successfully pivoting on the diagonal elements of matrix $\mathbf{G} = \text{Real}(\mathbf{Y}(s)) = ([\mathbf{Y}'(s^*) + \mathbf{Y}(s)]/2)$, first by pivoting on the first diagonal element G_{11} , then proceeding by pivoting on the second diagonal element G_{22} , and so on. $\mathbf{G} = \text{Real}(\mathbf{Y}(s))$ will be positive definite if after the successive pivoting process, the resulting matrix $\mathbf{G}$ has non-negative diagonal elements. For lossless networks a different check is proposed, since all the poles are imaginary. This passivity test involves checking that the residues corresponding to each imaginary pole is positive and real, for all the elements of the admittance matrix $\mathbf{Y}(s)$. The concept is then extended by checking that the matrix representing residues of the elements of $\mathbf{Y}(s)$ for a pole is positive semi-definite. The check is carried out for the residue matrices corresponding to all the imaginary poles.

The limitation of the above test for lossy networks is that, the pivot test needs to be done for all $\omega \in (0, \infty)$. It is difficult to determine how fine the frequency sweep should be while carrying out the pivot test, as well as up to what maximum frequency the test should be carried out. Hence, this approach does not provide an adequate passivity test.

The limitation of the proposed test for the lossless network is that, there is no guarantee that all the poles will conform to this criteria. This is because this condition is sufficient but not necessary for a network to be passive and all passive networks need not necessarily satisfy this criteria. Hence, this test is too constraining on the residues and may not be practical.

In [51], a similar approach based on approximation in terms of pole-residue pair is presented. However, in order to make the macromodel passive, each pole-residue pair is constrained to be positive-real. To understand the passivity checking method in [51], consider a M -port admittance matrix approximated by pole-residue formulation

$$Y(s) = A^{(0)} + sA^{(\infty)} + \sum_{n=1}^N \left(\frac{A_0^{(n)}}{s - \alpha_n} + \frac{A_0^{(n)*}}{s - \alpha_n^*} \right) A^{(n)} \quad (2.21)$$

where the notation z^* denotes the complex conjugate of z . The above expression is recognised as Foster's canonical representation. Extending Foster's canonical representation to the case of lossy, reciprocal multi-ports, and using the result that the admittance matrix for a passive multi-port network is a positive real matrix, the matrices $A^{(n)}$ in (2.21) must be real, symmetric and positive semi-definite [52].

In addition, the method enforces certain conditions on residues $A_0^{(n)}$ corresponding to each pole, for them to be realized by real and positive RLCG components. To understand these conditions consider the n^{th} pole-pair from (2.21)

$$\frac{A_0^{(n)}}{s - \alpha_n} + \frac{A_0^{(n)*}}{s - \alpha_n^*} \quad (2.22)$$

As shown in [52], the term in (2.22) can be synthesized through an RLCG equivalent circuit, with element values given by the following relationships,

$$L^{-1} = 2\text{Re}\{A_0^{(n)}\}, \quad GC^{-1} = \frac{-\text{Re}\{A_0^{(n)}\alpha_n^*\}}{\text{Re}\{A_0^{(n)}\}}, \quad RL^{-1} = \frac{\text{Re}\{A_0^{(n)}\alpha_n^*\}}{\text{Re}\{A_0^{(n)}\}} - 2\text{Re}\{\alpha_n\} \quad (2.23)$$

Considering that the real part of the poles α_n are negative, R, L, C, G in (2.23) will be real and positive, if the following conditions are met

- $\text{Re}\{A_0^{(n)}\} > 0$
- $0 \leq \text{Im}\{A_0^{(n)}\}\text{Im}\{\alpha_n\} \leq |\text{Re}\{A_0^{(n)}\}\text{Re}\{\alpha_n\}|$

The above conditions are checked for each pole-residue pairs.

The limitation of the above methodology is that the conditions are imposed on each pole-residue pair. This criteria for passivity is sufficient but not necessary. This is because, every pole-residue pair may not by itself conform to such conditions. As a result, enforcing such conditions on each individual pole-residue pair of the synthesized model may be too constraining and might fail to converge.

2.2.2 Interconnect Simulation Using Order Reduction and ‘S’ Parameters [28], [29]

This technique starts by approximating the tabulated scattering parameters as against the admittance parameters in the technique in Sec. 2.2.1. This is because S-parameters in addition to being stable, also have a physical meaning as they define the relationship between incident and reflected signals. In addition, they are readily available from electromagnetic analyses or from measurements.

The network shown in the Fig. 2.2 can be represented by $M \times M$ scattering matrix. The frequency-domain scattering matrix relates the incident-wave vectors and the reflected-wave vectors as

$$\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & \dots & S_{1M} \\ S_{21} & S_{22} & \dots & S_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ S_{M1} & S_{M2} & \dots & S_{MM} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix} \quad (2.24)$$

or

$$\mathbf{B} = \mathbf{S}\mathbf{A} \quad (2.25)$$

Fig. 2.2. M-Port network

where $\mathbf{S}$ is the $M \times M$ scattering matrix describing the measured network, $\mathbf{B}$ is the vector of reflected waves and $\mathbf{A}$ is the vector of incident waves.

Rational Approximation of Scattering parameters

Each element of the scattering matrix of (2.24) is approximated by the rational function

$$S_{ij}(s) = \frac{Q(s)}{P(s)} = \frac{q_0 + q_1s + q_2s^2 + \dots + q_ns^n}{1 + p_1s + p_2s^2 + \dots + p_ms^m}; \quad i, j = 1, 2, \dots, M \quad (2.26)$$

with p_0 normalized to unity. Equation (2.26) has $T_u = n + m + 1$ unknown coefficients. To calculate these coefficients, (2.26) is written for at least T_u frequency points. Given $S_{ij}(s)$ at k frequency points (2.26) can be expressed as

$$\frac{Q(s_l)}{P(s_l)} - S_{ij}(s_l) = 0, \quad \{l = 1, 2, \dots, k\} \quad (2.27)$$

By cancelling the denominators in (2.27) we get

$$Q(s_l) - S_{ij}(s_l)P(s_l) = 0, \quad \{l = 1, 2, \dots, k\} \quad (2.28)$$

Equation (2.28) can be written in the matrix form as

$$\begin{bmatrix} 1 & s_1 & s_1^2 & \dots & s_1^n & -s_1 S_{ij}(s_1) & -s_1^2 S_{ij}(s_1) & \dots & -s_1^m S_{ij}(s_1) \\ 1 & s_2 & s_2^2 & \dots & s_2^n & -s_2 S_{ij}(s_2) & -s_2^2 S_{ij}(s_2) & \dots & -s_2^m S_{ij}(s_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & s_k & s_k^2 & \dots & s_k^n & -s_k S_{ij}(s_k) & -s_k^2 S_{ij}(s_k) & \dots & -s_k^m S_{ij}(s_k) \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_n \\ p_1 \\ p_2 \\ \vdots \\ p_m \end{bmatrix} = \begin{bmatrix} S_{ij}(s_1) \\ S_{ij}(s_2) \\ \vdots \\ S_{ij}(s_k) \end{bmatrix} \quad (2.29)$$

or,

$$VX = Y \quad (2.30)$$

where $V \in C^{k \times T_u}$, $X \in \Re^{T_u \times 1}$ and $Y \in C^{k \times 1}$.

Equation (2.30) involves real as well as the imaginary parts of $S_{ij}(s)$. In practice the real part of $S_{ij}(s)$ is approximated by a rational function to evaluate the poles of the system. The real part of network function $S_{ij}(s)$ can be approximated with the real rational polynomial function of the squared variable as follows

$$Re\{S_{ij}(s)\} = \frac{c_0 + c_1s^2 + c_2s^4 + \dots + c_ns^n}{1 + p_1s^2 + p_2s^4 + \dots + p_ms^m} \quad (2.31)$$

Since the poles of the even function of $F(s)$ are those of both $F(s)$ and $F(-s)$, those belonging to $F(s)$ lie in the left-half plane [28]. Thus the denominator of (2.26) can be found from (2.31). The following system of equations result by writing the real part of $S_{ij}(s)$ in (2.31) at k number of frequency points

$$\begin{bmatrix} 1 & -\omega_1^2 & \omega_1^4 & \dots & (-1)^{\frac{n}{2}} \omega_1^n & \omega_1^2 S_1^r & -\omega_1^4 S_1^r & \dots & (-1)^{\frac{m}{2}+1} \omega_1^m S_1^r \\ 1 & -\omega_2^2 & \omega_2^4 & \dots & (-1)^{\frac{n}{2}} \omega_2^n & \omega_2^2 S_2^r & -\omega_2^4 S_2^r & \dots & (-1)^{\frac{m}{2}+1} \omega_2^m S_2^r \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & -\omega_k^2 & \omega_k^4 & \dots & (-1)^{\frac{n}{2}} \omega_k^n & \omega_k^2 S_k^r & -\omega_k^4 S_k^r & \dots & (-1)^{\frac{m}{2}+1} \omega_k^m S_k^r \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \\ p_1 \\ p_2 \\ \vdots \\ p_m \end{bmatrix} = \begin{bmatrix} Re\{S_{ij}(s_1)\} \\ Re\{S_{ij}(s_2)\} \\ \vdots \\ Re\{S_{ij}(s_k)\} \end{bmatrix} \quad (2.32)$$

or

$$WX = Z \quad (2.33)$$

where $W \in \mathbb{R}^{k \times a}$, $X \in \mathbb{R}^{a \times 1}$, $Z \in \mathbb{R}^{k \times 1}$ and $a = n + m + 1$. Equation (2.33) is solved by using the standard Householder algorithm, i.e. $RX = Q'Z$, where $W = QR$, Q is an orthogonal matrix and R is an upper triangular matrix. Once we have the solution of X , we know the denominator coefficients of $S_{ij}(s)$. The denominator polynomial is factorized to obtain the poles. Any unstable pole or a purely imaginary pole is ignored. Hence the order of the approximating rational function should be set equal to or greater than the actual order sought. Once the poles of the system are known, (2.26) is written in the pole-residue form as

$$S_{ij}(s) = c + \sum_{i=1}^{m'} \frac{r_i}{s - p_i} \quad (2.34)$$

where c is the direct coupling constant, r_i are residues and p_i are the poles. Notice that $m' \leq m$ and $m - m'$ is the number of rejected unstable and purely imaginary poles. The unknowns to be solved in (2.34) are the residues r_i 's and direct coupling constant c . This is done by writing (2.34) at various frequency points and separately equating the real and imaginary parts. This results in the matrix equation as illustrated in (2.20). The resulting matrix equation is solved by the Householder algorithm. After the residues and direct coupling constants are computed, (2.34) represents the elements of scattering matrix. In the next section we will see how this partial fraction representation is used to carry out transient simulation.

Transient Simulation

Once the partial fraction expansion of the scattering parameters is obtained, the relation between the incident and reflected waves can be rewritten as

$$B(s) = \underbrace{\left(c + \sum_{i=1}^{m'} \frac{r_i}{s-p_i} \right)}_{S(s)} A(s) \quad (2.35)$$

where $A(s)$ and $B(s)$ are the Laplace domain incident and reflected signals respectively, while $S(s)$ is the scattering parameter approximation.

The time-domain response is obtained by calculating the convolution integral given as

$$b(t) = \int_0^t s(\tau) a(t-\tau) d\tau \quad (2.36)$$

where $a(t)$, $b(t)$ are the time-domain incident and reflected signals respectively, while $s(t)$ is the time-domain scattering parameter. The convolution integral in (2.36) becomes progressively more expensive as the simulation time increases. Since the scattering parameters are expressed as a sum of partial expansion (2.35), the time for the numerical convolution can be greatly reduced by taking advantage of recursive convolution.

In [13], recursive convolution and indirect numerical integration are used for fast transient simulation of transmission line networks. It is necessary to make assumptions about the nature of $a(t)$ that, its values at a discrete set of points suffice to specify $b(t)$ uniquely. As in the backward and forward Euler formulas, the excitation is assumed to be piecewise constant, i.e. $a(t) = \text{constant}$, where $t_{n-1} \leq t \leq t_n$. By using recursive convolution (2.36) can be written as (proof in Appendix 'A')

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} \tilde{b}_i(t_n) \quad (2.37)$$

at a time point t_n , where,

$$\tilde{b}_i(t_n) = \frac{r_i}{p_i}(1 - e^{-p_i \delta t_n})a(t_n - \delta t_n) + e^{-p_i \delta t_n} \tilde{b}_i(t_n - \delta t_n), \quad (2.38)$$

and

$$\delta t_n = t_n - t_{n-1} \quad (2.39)$$

Using the recursive convolution formula (2.37), the convolution in (2.36) is reduced into a linear operation that involves a simple update of parameters at each time point. For a 2-port system, the relationship between a_i 's and b_i 's can be described as

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix}_n = \underbrace{\begin{bmatrix} c^{11} & c^{12} \\ c^{21} & c^{22} \end{bmatrix}}_{\mathbf{C}} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}_n + \underbrace{\begin{bmatrix} \sum_{i=1}^{m'} \tilde{b}_i^{11} + \sum_{i=1}^{m'} \tilde{b}_i^{12} \\ \sum_{i=1}^{m'} \tilde{b}_i^{21} + \sum_{i=1}^{m'} \tilde{b}_i^{22} \end{bmatrix}}_{\tilde{\mathbf{b}}(t_n)} \quad (2.40)$$

at a time point t_n . The terminal currents and voltages at the ports are obtained by writing the incident and reflected waves as

$$a_j = \frac{1}{2} Re \left(Z_0^{-\frac{1}{2}} \right) (v_j + Z_0 i_j) \quad (2.41)$$

$$b_j = \frac{1}{2} Re \left(Z_0^{-\frac{1}{2}} \right) (v_j - Z_0^* i_j) \quad (2.42)$$

where v_j , i_j are the terminal voltage and current respectively, and Z_0 is the reference impedance. Substituting (2.41) and (2.42) into (2.40) at time point t_n , gives the relationship between the terminal voltages and currents as

$$\mathbf{i}(t_n) = (\mathbf{Z}_0^* \mathbf{I} + \mathbf{CZ}_0)^{-1} (\mathbf{I} - \mathbf{C}) \mathbf{v}(t_n) - 2(\mathbf{Z}_0^* \mathbf{I} + \mathbf{CZ}_0)^{-1} \text{Re} \left(\mathbf{Z}_0^{\frac{1}{2}} \right) \tilde{\mathbf{b}}(t_n) \quad (2.43)$$

Equation (2.43) can be implemented in a circuit simulation in terms of conductances and time-dependent current sources, given as

$$\begin{aligned} \mathbf{G} &= (\mathbf{Z}_0^* \mathbf{I} + \mathbf{CZ}_0)^{-1} (\mathbf{I} - \mathbf{C}) \\ \mathbf{J}_n &= 2(\mathbf{Z}_0^* \mathbf{I} + \mathbf{CZ}_0)^{-1} \text{Re} \left(\mathbf{Z}_0^{\frac{1}{2}} \right) \tilde{\mathbf{b}}(t_n) \end{aligned} \quad (2.44)$$

This algorithm although benefits from the recursive convolution in terms of CPU times, but does not address the key issue of passivity.

2.2.3 Rational Function Approximation via Nonlinear Formulation [30]

The method in [30] begins by approximating the tabulated admittance function $Y(s)$ by the rational function

$$Y(\mathbf{a}, \mathbf{b}, s) = \frac{b_0 + b_1 s + b_2 s^2 + b_3 s^3 + \dots + b_{m-1} s^{m-1} + b_m s^m}{1 + a_1 s + a_2 s^2 + a_3 s^3 + \dots + a_{n-1} s^{n-1} + a_n s^n} \quad (2.45)$$

or

$$Y(\mathbf{a}, \mathbf{b}, s) = \frac{\sum_{i=0}^m b_i s^i}{\sum_{i=0}^n a_i s^i} \quad (2.46)$$

where $\mathbf{a} = [a_0, a_1, \dots, a_n]$, $a_0 = 1$, $\mathbf{b} = [b_0, \dots, b_m]$. Given a sampled transfer function $H(s_l)$, at N discrete frequency points $l = 1, 2, \dots, N$, the rational approximation problem amounts to minimizing the error

$$E(s_l) = Y(\mathbf{a}, \mathbf{b}, s_l) - H(s_l); \quad (l = 1, 2, \dots, N) \quad (2.47)$$

in some appropriate norm.

When minimizing 2-norm, problem in (2.47) can be expressed as

$$\|E(\mathbf{a}, \mathbf{b}, s)\|_2 = \sqrt{\sum_{l=1}^N (\|Y(\mathbf{a}, \mathbf{b}, s_l) - H(s_l)\|_2)^2} \quad (2.48)$$

In turn (2.48) is equivalent to minimizing

$$f(x) = \frac{1}{2}(\|E(\mathbf{a}, \mathbf{b}, s)\|_2)^2 = \frac{1}{2}\hat{\mathbf{E}}(\mathbf{a}, \mathbf{b}, s)^t \hat{\mathbf{E}}(\mathbf{a}, \mathbf{b}, s) \quad (2.49)$$

with

$$\hat{\mathbf{E}}(\mathbf{a}, \mathbf{b}, s)^t = [Re(E(s_1)) \dots Re(E(s_N)), Im(E(s_1)) \dots Im(E(s_N))] \quad (2.50)$$

and

$$\mathbf{x} = [\mathbf{a} \quad \mathbf{b}] \quad (2.51)$$

The problem in (2.49) is solved by Levenberg-Marquardt method [31], which uses a linearized model $\hat{\mathbf{M}}$ of $\hat{\mathbf{E}}(\mathbf{x})$ around $\mathbf{x}_c$ (based on Newton-Raphson iterations)

$$\hat{M}(x_c) = \hat{E}(x_c) + J(x_c)(x - x_c) \quad (2.52)$$

where $J(x_c)$ is the Jacobian of $\hat{E}$ at x_c . The method minimizes, at each step, the model error, subject to the following constraint

$$\min \|\hat{E}(x_c) + J(x_c)\delta x\|_2; \quad \delta x \in \Re^{m+n+1}; \quad \|\delta x\|_2 \leq \delta_c \quad (2.53)$$

The problem in (2.53) is solved by Gauss-Newton's method (δ_c is the desired tolerance). However, the Jacobian $J(x_c)$ in (2.53) loses numerical rank rapidly, as the order increases and therefore the solution of (2.53) becomes ill-conditioned. To understand the phenomenon, it is evident from (2.47) that the Jacobian $J(x_{k-1})$ is simply the Jacobian of $\begin{bmatrix} \text{Re}\{Y_{k-1}(s_1)\} \dots \text{Re}\{Y_{k-1}(s_N)\}, \text{Im}\{Y_{k-1}(s_1)\} \dots \text{Im}\{Y_{k-1}(s_N)\} \end{bmatrix}'$. Consequently (2.52) can be written as

$$-\underbrace{\begin{bmatrix} \text{Re}\{D_{k-1}^{-1}U_{0;m}\} & \dots & \text{Re}\{-D_{k-1}^{-1}Y_{k-1}U_{1;n}\} \\ \text{Im}\{D_{k-1}^{-1}U_{0;m}\} & \dots & \text{Im}\{-D_{k-1}^{-1}Y_{k-1}U_{1;n}\} \end{bmatrix}}_{J(x_{k-1})} \begin{bmatrix} \Delta b \\ \Delta a \end{bmatrix} = \begin{bmatrix} \text{Re}\{Y_{k-1}(s_1) - H(s_1)\} \\ \vdots \\ \text{Re}\{Y_{k-1}(s_N) - H(s_N)\} \\ \text{Im}\{Y_{k-1}(s_1) - H(s_1)\} \\ \vdots \\ \text{Im}\{Y_{k-1}(s_N) - H(s_N)\} \end{bmatrix} \quad (2.54)$$

where, $U_{0;m}$ is a Vandermonde matrix, Δa , Δb are updates of a , b respectively.

$$U_{0;m} = \begin{bmatrix} 1 & s_1 & s_1^2 & \dots & s_1^m \\ 1 & s_2 & s_2^2 & \dots & s_2^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & s_N & s_N^2 & \dots & s_N^m \end{bmatrix}, \quad (2.55)$$

$D_{k-1}(s)$ is a diagonal matrix of the denominator of $Y(s)$ at iteration number $k-1$

$$D_{k-1}(s) = \begin{bmatrix} \sum_{i=0}^n a_i s_1^i & 0 & \dots & 0 \\ 0 & \sum_{i=0}^n a_i s_2^i & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sum_{i=0}^n a_i s_N^i \end{bmatrix}, \quad (2.56)$$

$Y_{k-1}(s)$ is the diagonal matrix of approximation of $Y(s)$ at iteration number $k-1$

$$Y_{k-1}(s) = \begin{bmatrix} Y_{k-1}(s_1) & 0 & \dots & 0 \\ & Y_{k-1}(s_2) & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & Y_{k-1}(s_N) \end{bmatrix}, \quad (2.57)$$

and N is the number of frequency points at which tabulated data is provided. It can be easily seen that the Vandermonde matrix of (2.55) becomes highly ill-conditioned as the order of approximation increases, thus rendering the Jacobian $J(\mathbf{x}_{k-1})$ also ill-conditioned. This is likely to stall the Gauss-Newton procedure in (2.54). To deal with this ill-conditioning problem, it has been proposed in [30] to generate an orthonormal polynomial basis that spans the same space as the columns of Vandermonde matrix of (2.55). From (2.55) it is evident that $U_{0;m}$ can be expressed as

$$\begin{bmatrix} 1 & s1 & s^2 1 & \dots & s^m 1 \end{bmatrix} \quad (2.58)$$

where,

$\mathbf{S} = \text{diag}([s_1 \ s_2 \ \dots \ s_N]^T)$ and $\mathbf{I} \in \Re^N$ is a column vector with all the entries equal to 1. An orthonormal basis for $\mathbf{U}_{0;m}$ can be easily obtained using an Arnoldi Process [32]. Using the Arnoldi process we can write

$$\mathbf{S}\hat{\mathbf{V}} = \hat{\mathbf{V}}\hat{\mathbf{H}} \quad (2.59)$$

where, $\hat{\mathbf{V}}$ has orthonormal columns, i.e. $\hat{\mathbf{V}}^T \hat{\mathbf{V}} = \mathbf{I}$ and $\hat{\mathbf{H}}$ is an upper Hessenberg matrix [32]. The column space of $\hat{\mathbf{V}}$ spans the same space as that of the columns of $\mathbf{U}_{0;m}$. Notice that since $\mathbf{S}$ is skew-hermitian so is $\hat{\mathbf{H}}$. But $\hat{\mathbf{H}}$ is also upper Hessenberg by definition. Therefore it has to be a tridiagonal matrix. In this case, the Arnoldi process becomes a Lanczos process. In other words this is a process of constructing a polynomial basis defined by the three term recursion

$$sP_{j-1}(s) = \sum_{i=j-1}^{j+1} h_{i,j} P_{i-1}(s) \quad (2.60)$$

where, $P_j(s)$ is a polynomial of order j , ($j=1$ to m) and $h_{i,j}$ are the elements of $\hat{\mathbf{H}}$.

Replacing Vandermonde matrices in (2.54) by the new basis matrices we get

$$\begin{bmatrix} \text{Re}\{\mathbf{D}_{k-1}^{-1} \hat{\mathbf{V}}_{0;m}\} & \dots & \text{Re}\{-\mathbf{D}_{k-1}^{-1} \mathbf{Y}_{k-1} \hat{\mathbf{V}}_{1;n}\} \\ \text{Im}\{\mathbf{D}_{k-1}^{-1} \hat{\mathbf{V}}_{0;m}\} & \dots & \text{Im}\{-\mathbf{D}_{k-1}^{-1} \mathbf{Y}_{k-1} \hat{\mathbf{V}}_{1;n}\} \end{bmatrix} \begin{bmatrix} \Delta \tilde{\mathbf{b}} \\ \Delta \tilde{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} \text{Re}\{Y_{k-1}(s_1) - H(s_1)\} \\ \text{Re}\{Y_{k-1}(s_N) - H(s_N)\} \\ \text{Im}\{Y_{k-1}(s_1) - H(s_1)\} \\ \text{Im}\{Y_{k-1}(s_N) - H(s_N)\} \end{bmatrix} \quad (2.61)$$

which can be solved without the problem of ill-conditioning associated with the higher order of approximations.

By changing from the monomial basis to the new polynomial basis, (2.46) can be written as

$$Y(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}, s) = \frac{\sum_{i=0}^m \tilde{b}_i P_i(s)}{\sum_{i=0}^n \tilde{a}_i P_i(s)} \quad (2.62)$$

where $P_i(s)$ are described in (2.60).

Since the method described above is a non-linear problem, a good solution will depend on the starting guess of the coefficients $\tilde{a}_i's$ and $\tilde{b}_i's$. To obtain a good initial guess it has been suggested [30] to solve (2.62) by multiplying its both sides by the denominator and solving the overdetermined linear least square problem. However, this process of obtaining the initial guess might suffer from ill-conditioning problem for high order of approximations.

Finally, to carry out the transient simulation, (2.62) is converted into the state space formulation

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \end{aligned} \quad (2.63)$$

Details about the formulation of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ can be found in [30]. The ordinary differential equation (2.63) is linked to the circuit simulator such as HSPICE.

The above algorithm does not address the key issue of the passivity of the macro model represented by (2.63).

2.2.4 Rational Function Approximation Using Nevanlinna-Pick Interpolation

Recently an algorithm based on Nevanlinna-Pick interpolation [33] has been suggested to obtain rational function approximation for frequency-sampled data. Given the sampled transfer function matrix of a passive system, the method finds a guaranteed positive real matrix rational function approximation, while at the same time matching the frequency response of the original network. In [33], the passivity condition has been defined in terms of the Scattering parameter matrix: A system is passive if its scattering matrix is bounded real. A matrix $\mathbf{H}(s)$ is bounded real if the following conditions are satisfied:

$$(i) \mathbf{H}^*(s) = \mathbf{H}(s^*)$$

$$(ii) \mathbf{H}(s) \text{ is analytic in } \operatorname{Re}(s) > 0$$

$$(iii) \|\mathbf{H}(j\omega)\| \leq 1 \text{ for } \omega \in \Re$$

The problem formulation of the method for the frequency sampled scattering parameters proceeds as follows:

Let Δ represent the open right-half plane. Given a set of scattering matrices $\mathbf{H}(s_k) \in \zeta^{n_o \times n_i}$ ($k = 1, 2, \dots, N$) at N frequency points, n_o, n_i are the number of output and input ports respectively, ζ represents the complex domain. The $n_o \times n_i$ matrix rational function $\mathbf{F}(s)$ interpolates the data set $(s_k, \mathbf{H}(s_k))$ with $k = 1, 2, \dots, N$, if

$$\mathbf{F}(s_k) = \mathbf{H}(s_k), \quad k = 1, 2, \dots, N \quad (2.64)$$

The Nevanlinna-Pick problem is to find an interpolating function $F(s)$, that is analytic in Δ , interpolates the data set $(s_k, H(s_k))$ as in (2.64), and satisfies

$$\|F(s_k)\| < 1; \quad (k = 1, 2 \dots N) \quad (2.65)$$

Now, let

$$C_- = [I_{n_1} \dots I_{n_i}], \quad (2.66)$$

a $n_i \times Nn_i$ matrix, (I_{n_i} is a $n_i \times n_i$ identity matrix)

$$C_+ = [H_1 \dots H_N], \quad (2.67)$$

a $n_o \times Nn_i$ matrix, (here H_i stands for $(H(s_i))$) and

$$A_\pi = \begin{bmatrix} s_1 I_{n_i} & 0 \\ & \ddots \\ 0 & s_N I_{n_i} \end{bmatrix}, \quad (2.68)$$

a $Nn_i \times Nn_i$ block diagonal matrix

Theorem (Matrix Nevanlinna-Pick): There exists a matrix rational function $F(s)$ that interpolates the given data set (s, H) , that is analytic in Δ and satisfies $\|F(s_k)\| < 1; \quad (k = 1, 2 \dots N)$, if and only if the Pick matrix,

$$A_{p,q} = \left[\frac{I - H_p^H H_q}{s_p^* + s_q} \right]_{1 \leq p, q \leq N} \quad (2.69)$$

is positive definite (the operator ' H ' indicates the hermitian operator). In this case, there is a 2×2 block matrix function

$$\Theta(s) = I_{n_0+n_i} + \begin{bmatrix} C_+ \\ C_- \end{bmatrix} (sI_{Nn_i} - A_\pi)^{-1} A^{-1} \begin{bmatrix} C_+^H & C_-^H \end{bmatrix} = \begin{bmatrix} \Theta_{1,1}(s) & \Theta_{1,2}(s) \\ \Theta_{2,1}(s) & \Theta_{2,2}(s) \end{bmatrix} \quad (2.70)$$

such that the solution of the Nevanlinna-Pick interpolation becomes

$$F(s) = [\Theta_{1,1}(s)G(s) + \Theta_{1,2}(s)][\Theta_{2,1}(s)G(s) + \Theta_{2,2}(s)]^{-1} \quad (2.71)$$

where $G(s)$ is an arbitrary $n_o \times n_i$ rational matrix function that is analytic on Δ and satisfies $\|G(s_k)\| < 1$, $(k = 1, 2 \dots N)$.

In the above theorem, the data points are assumed to be in the interior of the Δ , i.e. the open right-half plane. But the frequency sampled data is essentially available on the imaginary axis. The identification then becomes the boundary interpolation problem, also called a Loewner interpolation problem [33], in which (2.69)-(2.71) are evaluated by applying a small shift σ to s .

From the first glance, this approach may seem to be straightforward, but there are several critical issues, which need to be addressed. They are as follows:

- From (2.71), the Nevanlinna-Pick interpolant will have at least Nn_i poles (e.g. a 3-port system with 200 data points will have 600 poles). A systematic approach is required to reduce the number of poles
- The choice of $G(s)$ is critical to the accuracy of the fitting process. The appropriate choice of $G(s)$ often leads to the non-linear optimization problem [33], which can be expensive and may not always converge
- The choice of σ is critical to the positive definiteness of the pick matrix in (2.69). On one hand, too small a value can lead to the pure imaginary poles,

leading to oscillatory response, while on the other hand too large a value may lead to pick matrix in (2.69) not being positive definite.

In summary, this chapter has reviewed some of the prominent techniques to include the frequency sampled data in an overall circuit simulation. Techniques based on convolution and rational function approximation have been described. In addition, the problems such as illconditioning and ensuring passivity, encountered by these techniques are discussed. The following chapters (chapter 3 and 4), describe the proposed passive macromodeling algorithm for tabulated data, which help to overcome the problem of illconditioning and passivity.

In chapter 2 we reviewed some of the conventional techniques used to macromodel interconnect networks, which are characterized by measured/simulated data. One of the major tasks in these techniques is to determine the accurate poles of the system. These techniques start by approximating the tabulated data by a rational function of the form

$$f(s) = \frac{a_0 + a_1s + a_2s^2 + \dots + a_Ns^N}{b_0 + b_1s + b_2s^2 + \dots + b_Ns^N} \quad (3.1)$$

However, when the above function is converted into a linear set of equations, the resulting matrix becomes highly ill-conditioned. Although, there have been some techniques proposed to overcome the problem of ill-conditioning, they do not yield good results for high orders of approximation [24], [25], [28], [29]. A technique known as Vector-Fitting is described in [49] to overcome the problem of ill-conditioning encountered while solving (3.1) for obtaining the poles of the system. In this thesis, a new algorithm is developed for accurate, high-order rational-function approximation of multiport tabulated data, which employs the principles outlined in [49]. Prior to describing the proposed algorithm, a brief review of Vector-fitting approach is given in the next section.

3.1 *Vector Fitting by Pole Relocation*

Consider the pole-residue approximation of a function $f(s)$ as follows

$$f(s) = \sum_{n=1}^N \frac{k_n}{s-p_n} + c + sh \quad (3.2)$$

where the residues (k_n) and poles (p_n) can be real or complex conjugate pairs, N is the number of poles and residues, c & h are real. The problem is to estimate all the coefficients in (3.2) so that a least-squares approximation of $f(s)$ is obtained over a given frequency interval. Notice that this is a nonlinear problem in terms of unknowns, as p_n appear in the denominator of (3.2).

The problem in (3.2) is solved as a linear problem in two steps, as follows.

Step 1: Pole Calculation

An initial guess of poles $\bar{p}_n$ is specified and a scaling function $\sigma(s)$ of the following rational form is considered

$$\sigma(s) = \sum_{n=1}^N \frac{\tilde{k}_n}{s-\bar{p}_n} + 1 \quad (3.3)$$

Multiplying (3.3) with $f(s)$ of (3.2) we get a scaled function $\sigma(s)f(s)$. Next, this scaled function is approximated (represented by $(\sigma f)(s)$) using the same set of poles as of $\sigma(s)$ as shown below

$$(\sigma f)(s) = \sum_{n=1}^N \frac{\hat{k}_n}{s-\bar{p}_n} + \hat{c} + s\hat{h} \quad (3.4)$$

Noting that $(\sigma f)(s)$ is an approximation of $\sigma(s)f(s)$, we can write

$$\sigma(s)f(s) \approx (\sigma f)(s) \quad (3.5)$$

or (using (3.3) and (3.4))

$$\left[\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} + 1 \right] f(s) \approx \left[\sum_{n=1}^N \frac{\hat{k}_n}{s - \bar{p}_n} + \hat{c} + s\hat{h} \right] \quad (3.6)$$

In (3.6), $\bar{p}_n$; $n = 1, 2, \dots, N$ is the initial guess of poles and the number of unknowns are $N_t = 2N + 2$. Equation (3.6) when written at k different frequency points ($k > N_t$) results in an overdetermined linear problem of the form

$$AX = b \quad (3.7)$$

where,

$$b = [f(s_1), f(s_2) \dots f(s_k)]^t \quad (3.8)$$

The form of the A and X depend on whether the poles are real or complex, as follows:

Case1: Real Poles

In case of real poles, writing (3.6) at several frequency points results in the following set of matrix equations:

$$\underbrace{\begin{bmatrix} \frac{1}{s_1 - \bar{p}_1} & \dots & \frac{1}{s_1 - \bar{p}_N} & 1 & s_1 & \frac{-f(s_1)}{s_1 - \bar{p}_1} & \dots & \frac{-f(s_1)}{s_1 - \bar{p}_N} \\ \frac{1}{s_2 - \bar{p}_1} & \dots & \frac{1}{s_2 - \bar{p}_N} & 1 & s_2 & \frac{-f(s_2)}{s_2 - \bar{p}_1} & \dots & \frac{-f(s_2)}{s_2 - \bar{p}_N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{s_k - \bar{p}_1} & \dots & \frac{1}{s_k - \bar{p}_N} & 1 & s_k & \frac{-f(s_k)}{s_k - \bar{p}_1} & \dots & \frac{-f(s_k)}{s_k - \bar{p}_N} \end{bmatrix}}_A \underbrace{\begin{bmatrix} \hat{k}_1 \\ \vdots \\ \hat{k}_N \\ \hat{c} \\ \hat{h} \\ \tilde{k}_1 \\ \vdots \\ \tilde{k}_N \end{bmatrix}}_X = \underbrace{\begin{bmatrix} f(s_1) \\ f(s_2) \\ \vdots \\ f(s_k) \end{bmatrix}}_b \quad (3.9)$$

or

$$AX = b \quad (3.10)$$

Notice that for real poles, the unknowns $X = [\hat{k}_1, \dots, \hat{k}_N, \hat{c}, \hat{h}, \tilde{k}_1, \dots, \tilde{k}_N]^t$ are real.

Case2: Complex Poles

Complex poles exist in complex conjugate pairs. As a result, their residues also exist in complex-conjugate pairs. Consider a complex pole along with its conjugate in (3.6)

$$\bar{p}_i = \text{Re}(\bar{p}_i) + j\text{Im}(\bar{p}_i), \quad \bar{p}_i^* = \text{Re}(\bar{p}_i) - j\text{Im}(\bar{p}_i) \quad (3.11)$$

where $*$ is the complex-conjugate operator. The corresponding residues of the scaled function $(\sigma f)(s)$ are

$$\hat{k}_i = \text{Re}(\hat{k}_i) + j\text{Im}(\hat{k}_i), \quad \hat{k}_i^* = \text{Re}(\hat{k}_i) - j\text{Im}(\hat{k}_i) \quad (3.12)$$

while the corresponding residues of scaling function $\sigma(s)$ are

$$\tilde{k}_i = \text{Re}(\tilde{k}_i) + j\text{Im}(\tilde{k}_i), \quad \tilde{k}_i^* = \text{Re}(\tilde{k}_i) - j\text{Im}(\tilde{k}_i) \quad (3.13)$$

For the above pole pair (3.6) becomes

$$\frac{\hat{k}_i}{s - \bar{p}_i} + \frac{\hat{k}_i^*}{s - \bar{p}_i^*} + \hat{c} + \hat{h}s - f(s)\frac{\tilde{k}_i}{s - \bar{p}_i} - f(s)\frac{\tilde{k}_i^*}{s - \bar{p}_i^*} = f(s) \quad (3.14)$$

Substituting (3.12) and (3.13) in (3.14) and after certain mathematical manipulations, we get

$$\begin{aligned} & \text{Re}(\hat{k}_i) \left(\frac{1}{s - \bar{p}_i} + \frac{1}{s - \bar{p}_i^*} \right) + \text{Im}(\hat{k}_i) \left(\frac{j}{s - \bar{p}_i} - \frac{j}{s - \bar{p}_i^*} \right) + \hat{c} + \hat{h}s \\ & - \text{Re}(\tilde{k}_i) f(s) \left(\frac{1}{s - \bar{p}_i} + \frac{1}{s - \bar{p}_i^*} \right) - \text{Im}(\tilde{k}_i) f(s) \left(\frac{j}{s - \bar{p}_i} - \frac{j}{s - \bar{p}_i^*} \right) = f(s) \end{aligned} \quad (3.15)$$

or

$$\left[\left(\frac{1}{s - \bar{p}_i} + \frac{1}{s - \bar{p}_i^*} \right) \left(\frac{j}{s - \bar{p}_i} - \frac{j}{s - \bar{p}_i^*} \right) \quad 1 \quad s \quad -f(s) \left(\frac{1}{s - \bar{p}_i} + \frac{1}{s - \bar{p}_i^*} \right) \quad -f(s) \left(\frac{j}{s - \bar{p}_i} - \frac{j}{s - \bar{p}_i^*} \right) \right] \begin{bmatrix} \text{Re}(\hat{k}_i) \\ \text{Im}(\hat{k}_i) \\ \hat{c} \\ \hat{h} \\ \text{Re}(\tilde{k}_i) \\ \text{Im}(\tilde{k}_i) \end{bmatrix} = f(s) \quad (3.16)$$

Equation (3.16) can be written at several frequencies for N complex poles resulting in the linear matrix equation $AX = b$, where

$$\begin{aligned}
 \mathbf{A} &= [\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_i, \dots, \mathbf{A}_N, \mathbf{I}, \mathbf{S}] \\
 \mathbf{A}_i &= \begin{bmatrix} \left(\frac{1}{s_1 - \bar{p}_i} + \frac{1}{s_1 - \bar{p}_i^*}\right) & \left(\frac{j}{s_1 - \bar{p}_i} - \frac{j}{s_1 - \bar{p}_i^*}\right) & -f(s_1)\left(\frac{1}{s_1 - \bar{p}_i} + \frac{1}{s_1 - \bar{p}_i^*}\right) & -f(s_1)\left(\frac{j}{s_1 - \bar{p}_i} - \frac{j}{s_1 - \bar{p}_i^*}\right) \\ \left(\frac{1}{s_2 - \bar{p}_i} + \frac{1}{s_2 - \bar{p}_i^*}\right) & \left(\frac{j}{s_2 - \bar{p}_i} - \frac{j}{s_2 - \bar{p}_i^*}\right) & -f(s_2)\left(\frac{1}{s_2 - \bar{p}_i} + \frac{1}{s_2 - \bar{p}_i^*}\right) & -f(s_2)\left(\frac{j}{s_2 - \bar{p}_i} - \frac{j}{s_2 - \bar{p}_i^*}\right) \\ \vdots & \vdots & \vdots & \vdots \\ \left(\frac{1}{s_k - \bar{p}_i} + \frac{1}{s_k - \bar{p}_i^*}\right) & \left(\frac{j}{s_k - \bar{p}_i} - \frac{j}{s_k - \bar{p}_i^*}\right) & -f(s_k)\left(\frac{1}{s_k - \bar{p}_i} + \frac{1}{s_k - \bar{p}_i^*}\right) & -f(s_k)\left(\frac{j}{s_k - \bar{p}_i} - \frac{j}{s_k - \bar{p}_i^*}\right) \end{bmatrix}; \\
 \mathbf{I} &= [1, 1, \dots, 1]_{1 \times k}^t; \\
 \mathbf{S} &= [s_1, s_2, \dots, s_k]^t
 \end{aligned} \tag{3.17}$$

$$\mathbf{X} = [\text{Re}(\hat{k}_1), \text{Im}(\hat{k}_1), \text{Re}(\tilde{k}_1), \text{Im}(\tilde{k}_1), \dots, \text{Re}(\hat{k}_N), \text{Im}(\hat{k}_N), \text{Re}(\tilde{k}_N), \text{Im}(\tilde{k}_N), c, h]^t$$

For both cases, the linear formulation $\mathbf{X} = \mathbf{b}$ is separated into real and imaginary parts, and subsequently, the following augmented problem is solved

$$\underbrace{\begin{bmatrix} \text{Re}(\mathbf{A}) \\ \text{Im}(\mathbf{A}) \end{bmatrix}}_{\hat{\mathbf{A}}} \mathbf{X} = \underbrace{\begin{bmatrix} \text{Re}(\mathbf{b}) \\ \text{Im}(\mathbf{b}) \end{bmatrix}}_{\hat{\mathbf{b}}} \tag{3.18}$$

or

$$\hat{\mathbf{A}}\mathbf{X} = \hat{\mathbf{b}} \tag{3.19}$$

The over-determined problem of (3.19) is solved by least-square technique or *QR* factorization: In the case of least-square method, pre-multiplying both sides of (3.19) by $\hat{\mathbf{A}}^t$, we get,

$$\hat{\mathbf{A}}^t \hat{\mathbf{A}} \mathbf{X} = \hat{\mathbf{A}}^t \hat{\mathbf{b}} \tag{3.20}$$

or

$$X = (\hat{A}^t \hat{A})^{-1} \hat{A}^t \hat{b} \quad (3.21)$$

While, in the case of using QR factorization, writing $\hat{A}$ in terms of QR factors,

$$\hat{A} = QR \quad (3.22)$$

where Q is an orthogonal matrix, i.e. $Q^t Q = I$, and R is an upper triangular matrix. Substituting (3.22) into (3.19) and multiplying both sides by Q^t , we get

$$RX = Q^t \hat{b} \quad (3.23)$$

or substituting $Q^t \hat{b} = \widehat{b}$, the following equation is solved to evaluate X

$$RX = \widehat{b} \quad (3.24)$$

Next, a rational-function approximation for the function $f(s)$ can be obtained by writing it in the fractional form. For this purpose, we first express the scaling (3.3) and scaled (3.4) functions in fractional form as:

$$\sigma(s) = \frac{\prod_{n=1}^N (s - \tilde{z}_n)}{\prod_{n=1}^N (s - \tilde{p}_n)} \quad (3.25)$$

and

$$(\sigma f)(s) = h \frac{\prod_{n=1}^{N+1} (s - z_n)}{\prod_{n=1}^N (s - \tilde{p}_n)} \quad (3.26)$$

From (3.5) it is evident that

$$f(s) = \frac{(\sigma f)(s)}{\sigma(s)} \quad (3.27)$$

or, substituting (3.25) and (3.26) in (3.27), $f(s)$ can be represented as

$$f(s) = \frac{(\sigma f)(s)}{\sigma(s)} = h \frac{\prod_{n=1}^{N+1} (s - z_n)}{\prod_{n=1}^N (s - \tilde{z}_n)} \quad (3.28)$$

From (3.28) and (3.25), it is clear that the zeros of $\sigma(s)$ are the poles of $f(s)$. (Note that the initial poles cancel in the division process as we started with the same initial poles for $\sigma(s)$ and $(\sigma f)(s)$). Thus, by calculating the zeros of $\sigma(s)$ we get a better set of poles than the initial set of poles p_n , for fitting the original function $f(s)$.

3.1.1 Evaluation of Poles of $f(s)$

Once the unknowns $\hat{k}_1, \dots, \hat{k}_N, \hat{c}, \hat{h}, \tilde{k}_1, \dots, \tilde{k}_N$ are known (by solving (3.21) or (3.24)), zeros of $\sigma(s)$ can be calculated, depending on whether the initial guess for the poles is real or complex, as follows:

Case 1: Real Poles

It is to be noted that zeros of $\sigma(s)$ correspond to the eigenvalues of the following matrix [49]:

$$\mathbf{H} = \mathbf{\Psi} - \mathbf{g}\phi^t \quad (3.29)$$

where,

$$\mathbf{\Psi} = \begin{bmatrix} \bar{p}_1 & & & \\ & \bar{p}_2 & & \\ & & \ddots & \\ & & & \bar{p}_N \end{bmatrix}; \quad \mathbf{g} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{N \times 1}; \quad \phi^t = \begin{bmatrix} \tilde{k}_1 & \tilde{k}_2 & \dots & \tilde{k}_N \end{bmatrix}_{1 \times N} \quad (3.30)$$

Notice that $\mathbf{\Psi}$ is the diagonal matrix of starting (initial guess) poles, $\mathbf{g}$ is a unity column vector. ϕ is the column vector of residues of $\sigma(s)$, corresponding to the starting poles (computed using (3.21) or (3.24)).

Case 2: Complex Poles

In this case too, zeros of $\sigma(s)$ correspond to the eigenvalues of the matrix $\mathbf{H}$ of (3.29), however, with $\mathbf{\Psi}$, $\mathbf{g}$ and ϕ modified as,

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{\Psi}_1 & & & \\ & \mathbf{\Psi}_2 & & \\ & & \ddots & \\ & & & \mathbf{\Psi}_N \end{bmatrix}; \quad \mathbf{\Psi}_i = \begin{bmatrix} \text{Re}(\bar{p}_i) & \text{Im}(\bar{p}_i) \\ -\text{Im}(\bar{p}_i) & \text{Re}(\bar{p}_i) \end{bmatrix}; \quad i = 1, 2, \dots, N$$

$$\phi^t = \begin{bmatrix} \text{Re}(\tilde{k}_1) & \text{Im}(\tilde{k}_1) & \text{Re}(\tilde{k}_2) & \text{Im}(\tilde{k}_2) & \dots & \text{Re}(\tilde{k}_N) & \text{Im}(\tilde{k}_N) \end{bmatrix}_{1 \times 2N}$$

$$\mathbf{g} = \begin{bmatrix} 2 & 0 & 2 & 0 & \dots & 2 & 0 \end{bmatrix}_{1 \times 2N}^t; \quad (3.31)$$

Once the zeros of $\sigma(s)$ are calculated, they are used as the new set of starting poles in the next iteration while solving (3.6). The next iteration provides a new set of zeros of $\sigma(s)$ which are again used in the next iteration as starting poles.

The iterative process is continued until the poles converge. It is to be noted that during this iterative process, some of the calculated poles may become unstable, i.e. the real part of some of the poles may be positive. This problem is overcome by inverting the sign of their real parts [49]. Successful application of vector fitting requires judicious selection of the initial set of poles.

3.1.2 Selection of Initial Set of Poles

The following guidelines should be used while selecting the initial set of poles [49].

a. Functions with distinct resonance peaks

The starting poles should be complex conjugate with the imaginary parts $\beta_1, \beta_2, \dots, \beta_n$ linearly distributed over the frequency range of interest $\omega_{min} \sim \omega_{max}$. Each pole pair is selected as follows

$$p = -\alpha \pm j\beta \quad (3.32)$$

with

$$\alpha = \beta/100 \quad (3.33)$$

b. Smooth functions

Real poles, linearly or logarithmically spaced over the frequency of interest, should be used for approximating the smooth functions.

3.2 *Difficulties with Pole Computation for Multiport Tabulated Subnetworks*

Consider the admittance matrix of the M -port measured network

$$Y(s) = \begin{bmatrix} Y_{11}(s) & Y_{12}(s) & \dots & Y_{1M}(s) \\ Y_{21}(s) & Y_{22}(s) & \dots & Y_{2M}(s) \\ \vdots & \vdots & \ddots & \vdots \\ Y_{M1}(s) & Y_{M2}(s) & \dots & Y_{MM}(s) \end{bmatrix} \quad (3.34)$$

Conventionally, each element of the above admittance matrix is approximated by a rational function. To find a common pole-set, the poles corresponding to individual elements of (3.34) are collected together. However, this approach can lead to large number of redundant poles in the macromodel resulting in inefficient transient simulation. In addition, identifying a common multi-port pole set from pole-sets of individual elements of the admittance matrix is generally heuristic and can lead to errors.

3.3 *Proposed Multiport-Pole Computation Algorithm*

In order to overcome the above problem and to minimize the number of poles in the matrix-transfer function, a method is proposed in this section to efficiently compute the common pole-set of the system. The following two propositions are used in this regard [41]:

- 1) In general, the pole set corresponding to any individual element of the admittance matrix is a subset of the union of all driving point admittances [91].

- 2) Generally, in a system with a large number of dominant poles, pole sets belonging to different driving point admittances contain mostly identical poles and only a very small percentage of poles differ among these sets.

Using the above two propositions, a common accurate pole-set for the multiport pole-residue model is obtained by performing the *pole identification algorithm* of the vector fitting approach (Sec. 3.1) [49] on the driving point admittances. This is done as follows.

The admittance matrix of a M -port electrical network can be written as (3.34). To obtain a common pole-set, the vector-fitting algorithm is carried out on the vector of the diagonal entries of the admittance matrix (proposition 1). Specifically, for the case of M -port network of (3.34), the vector

$$\mathbf{Y}_{diag}(s) = \begin{bmatrix} Y_{11}(s) \\ Y_{22}(s) \\ \vdots \\ Y_{MM}(s) \end{bmatrix} \quad (3.35)$$

is fitted. Each element of (3.35) is approximated with the following pole-residue formulation:

$$Y_{ii}(s) = \sum_{n=1}^N \frac{k_n^{i,i}}{s-p_n} + c^{i,i} + sh^{i,i} \quad (3.36)$$

where the residues (k_n) and poles (p_n) can be real or complex conjugate pairs, N is the number of poles and residues, and $c^{i,i}$ and $h^{i,i}$ are real constants. Next, an initial guess of the pole-set $\bar{\mathbf{P}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_N)$ is specified and a scaling function $\sigma(s)$ of the following rational form is considered:

$$\sigma(s) = \sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} + 1 \quad (3.37)$$

Multiplying $Y_{diag}(s)$ of (3.35) by the above scaling function we get the scaled function $\sigma(s)Y_{diag}(s)$. Next this scaled function is approximated (represented by $(\sigma Y_{diag})(s)$) using the same set of poles as of $\sigma(s)$, as below

$$(\sigma Y_{diag})(s) = \begin{bmatrix} (\sigma Y_{11})(s) \\ (\sigma Y_{22})(s) \\ \vdots \\ (\sigma Y_{MM})(s) \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^N \frac{\hat{k}_{11}}{s - \bar{p}_n} + \hat{c}^{11} + s\hat{h}^{11} \\ \sum_{n=1}^N \frac{\hat{k}_{22}}{s - \bar{p}_n} + \hat{c}^{22} + s\hat{h}^{22} \\ \vdots \\ \sum_{n=1}^N \frac{\hat{k}_{MM}}{s - \bar{p}_n} + \hat{c}^{MM} + s\hat{h}^{MM} \end{bmatrix} \quad (3.38)$$

Noting that $(\sigma Y_{diag})(s)$ is an approximation of $\sigma(s)Y_{diag}(s)$, we can write

$$\sigma(s)Y_{diag}(s) \approx (\sigma Y_{diag})(s) \quad (3.39)$$

or, from (3.35)

$$\begin{bmatrix} \sigma(s)Y_{11}(s) \\ \sigma(s)Y_{22}(s) \\ \vdots \\ \sigma(s)Y_{MM}(s) \end{bmatrix} \approx \begin{bmatrix} (\sigma Y_{11})(s) \\ (\sigma Y_{22})(s) \\ \vdots \\ (\sigma Y_{MM})(s) \end{bmatrix} \quad (3.40)$$

Using (3.37) and (3.38), we can write (3.40) as:

$$\begin{bmatrix} \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} + 1 \right) Y_{11}(s) \\ \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} + 1 \right) Y_{22}(s) \\ \vdots \\ \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} + 1 \right) Y_{MM}(s) \end{bmatrix} \approx \begin{bmatrix} \sum_{n=1}^N \frac{\hat{k}_n^{11}}{s - \bar{p}_n} + \hat{c}^{11} + s\hat{h}^{11} \\ \sum_{n=1}^N \frac{\hat{k}_n^{22}}{s - \bar{p}_n} + \hat{c}^{22} + s\hat{h}^{22} \\ \vdots \\ \sum_{n=1}^N \frac{\hat{k}_n^{MM}}{s - \bar{p}_n} + \hat{c}^{MM} + s\hat{h}^{MM} \end{bmatrix} \quad (3.41)$$

or

$$\begin{bmatrix} \sum_{n=1}^N \frac{\hat{k}_n^{11}}{s - \bar{p}_n} + \hat{c}^{11} + s\hat{h}^{11} \\ \sum_{n=1}^N \frac{\hat{k}_n^{22}}{s - \bar{p}_n} + \hat{c}^{22} + s\hat{h}^{22} \\ \vdots \\ \sum_{n=1}^N \frac{\hat{k}_n^{MM}}{s - \bar{p}_n} + \hat{c}^{MM} + s\hat{h}^{MM} \end{bmatrix} - \begin{bmatrix} \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} \right) Y_{11}(s) \\ \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} \right) Y_{22}(s) \\ \vdots \\ \left(\sum_{n=1}^N \frac{\tilde{k}_n}{s - \bar{p}_n} \right) Y_{MM}(s) \end{bmatrix} = \begin{bmatrix} Y_{11}(s) \\ Y_{22}(s) \\ \vdots \\ Y_{MM}(s) \end{bmatrix} \quad (3.42)$$

Note that in (3.42), the poles of the scaling function $\sigma(s)$ are considered to be same as the poles of the approximated scaled functions $(\sigma Y_{11})(s), (\sigma Y_{22})(s), \dots, (\sigma Y_{MM})(s)$. Also note here that $\bar{p}_n$ represents the initial guess of poles and number of unknowns are $N_t = 2M + (M+1)N$. Writing each row of (3.42) at k different frequency points ($k > 2N + 2$), we get an over-determined problem:

$$AX = b \quad (3.43)$$

where,

$$X = \begin{bmatrix} \hat{k}_1^{11} & \dots & \hat{k}_N^{11} & \hat{c}^{11} & \hat{h}^{11} & \hat{k}_1^{22} & \dots & \hat{k}_N^{22} & \hat{c}^{22} & \hat{h}^{22} & \dots & \hat{k}_1^{MM} & \dots & \hat{k}_N^{MM} & \hat{c}^{MM} & \hat{h}^{MM} & \tilde{k}_1 & \dots & \tilde{k}_N \end{bmatrix}^t \quad (3.44)$$

$$b = \begin{bmatrix} Y_{11}(s_1) & \dots & Y_{11}(s_k) & Y_{22}(s_1) & \dots & Y_{22}(s_k) & \dots & Y_{MM}(s_1) & \dots & Y_{MM}(s_k) \end{bmatrix}^t \quad (3.45)$$

$A =$

$$\begin{bmatrix} \frac{1}{s_1 - \bar{p}_1} & \dots & \frac{1}{s_1 - \bar{p}_N} & 1 & s_1 & 0 & \dots & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \frac{-Y_{11}(s_1)}{s_1 - \bar{p}_1} & \dots & \frac{-Y_{11}(s_1)}{s_1 - \bar{p}_N} \\ \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \vdots & & \vdots \\ \frac{1}{s_k - \bar{p}_1} & \dots & \frac{1}{s_k - \bar{p}_N} & 1 & s_k & 0 & \dots & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \frac{-Y_{11}(s_k)}{s_k - \bar{p}_1} & \dots & \frac{-Y_{11}(s_k)}{s_k - \bar{p}_N} \\ 0 & \dots & 0 & 0 & 0 & \frac{1}{s_1 - \bar{p}_1} & \dots & \frac{1}{s_1 - \bar{p}_N} & 1 & s_1 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \frac{-Y_{22}(s_1)}{s_1 - \bar{p}_1} & \dots & \frac{-Y_{22}(s_1)}{s_1 - \bar{p}_N} \\ \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & 0 & \frac{1}{s_k - \bar{p}_1} & \dots & \frac{1}{s_k - \bar{p}_N} & 1 & s_k & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \frac{-Y_{22}(s_k)}{s_k - \bar{p}_1} & \dots & \frac{-Y_{22}(s_k)}{s_k - \bar{p}_N} \\ \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & & & & & & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & \frac{1}{s_1 - \bar{p}_1} & \dots & \frac{1}{s_1 - \bar{p}_N} & 1 & s_1 & \frac{-Y_{MM}(s_1)}{s_1 - \bar{p}_1} & \dots & \frac{-Y_{MM}(s_1)}{s_1 - \bar{p}_N} \\ \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & \frac{1}{s_k - \bar{p}_1} & \dots & \frac{1}{s_k - \bar{p}_N} & 1 & s_k & \frac{-Y_{MM}(s_k)}{s_k - \bar{p}_1} & \dots & \frac{-Y_{MM}(s_k)}{s_k - \bar{p}_N} \end{bmatrix} \quad (3.46)$$

The over-determined problem of (3.43) is solved by least-square technique or QR factorization. In the case of least-square method, pre-multiplying both sides of (3.43) by A^t , we get,

$$A^t A X = A^t b \quad (3.47)$$

or,

$$X = (A^t A)^{-1} A^t b \quad (3.48)$$

On the other hand, in the case of using QR factorization, writing A in terms of QR factors, we get

$$A = QR \quad (3.49)$$

Here Q is an orthogonal matrix, i.e. $Q'Q = I$, and R is an upper triangular matrix. Substituting (3.49) into (3.43) and multiplying both sides by Q' , we get

$$RX = Q'b \quad (3.50)$$

or substituting $Q'b = \widehat{b}$, the following equation is solved to evaluate X

$$RX = \widehat{b} \quad (3.51)$$

Next, a rational-function approximation for the function $Y_{diag}(s)$ can be easily obtained by writing it in the fractional form. For this purpose, we first express the scaling (3.37) and scaled (3.38) functions in fractional form as:

$$\sigma(s) = \frac{\prod_{n=1}^N (s - \tilde{z}_n)}{\prod_{n=1}^N (s - \bar{p}_n)} \quad (3.52)$$

and

$$(\sigma Y_{diag})(s) = \frac{1}{\prod_{n=1}^N (s - \bar{p}_n)} \left[\begin{array}{c} N+1 \\ \hat{h}^{11} \prod_{n=1}^{N+1} (s - \hat{z}_n^{11}) \\ N+1 \\ \hat{h}^{22} \prod_{n=1}^{N+1} (s - \hat{z}_n^{22}) \\ \vdots \\ N+1 \\ \hat{h}^{MM} \prod_{n=1}^{N+1} (s - \hat{z}_n^{MM}) \end{array} \right] \quad (3.53)$$

From (3.39) it is evident that

$$Y_{diag}(s) = \frac{(\sigma Y_{diag})(s)}{\sigma(s)} \quad (3.54)$$

Substituting (3.52) and (3.53) in (3.54), $Y_{diag}(s)$ becomes

$$Y_{diag}(s) = \frac{(\sigma Y_{diag})(s)}{\sigma(s)} = \frac{1}{\prod_{n=1}^N (s - \tilde{z}_n)} \left[\begin{array}{c} N+1 \\ \hat{h}^{11} \prod_{n=1}^{N+1} (s - \hat{z}_n^{11}) \\ N+1 \\ \hat{h}^{22} \prod_{n=1}^{N+1} (s - \hat{z}_n^{22}) \\ \vdots \\ N+1 \\ \hat{h}^{MM} \prod_{n=1}^{N+1} (s - \hat{z}_n^{MM}) \end{array} \right] \quad (3.55)$$

From (3.52) and (3.55), it is clear that the zeros of $\sigma(s)$ are the poles of $Y_{diag}(s)$. *(Note that the initial poles cancel in the division process as we started with the same initial poles for $\sigma(s)$ and $(\sigma Y_{diag})(s)$).* Thus, by calculating the zeros of $\sigma(s)$ we get a better set of poles than the starting set of poles $\bar{p}_n$, for fitting the original

function $Y_{diag}(s)$. Next, using the approach outlined in Sec. 3.1.1, zeros of $\sigma(s)$ and subsequently poles of $Y_{diag}(s)$ can be computed.

The above approach helps in reducing the ill-conditioning problem and also ensures that the poles of the admittance matrix are common.

3.4 Advantages of the Proposed Multiport Pole-Computation over Conventional Techniques

The proposed method of multiport vector-fitting described above has the following advantages over the conventional curve-fitting techniques:

1. The method does not suffer from the problem of ill-conditioning as in the case of conventional techniques. This is because the entries of the matrix A of the overdetermined problem $AX = b$ are raised to single power of s , as against powers equal to the order of approximation, in conventional techniques.
2. The method can easily achieve higher orders of approximations. This enables the method to fit wideband tabulated data with many poles.
3. With a judicious selection of initial guesses of poles, the method is able to capture the response of the system.
4. The method efficiently calculates the common pole-set of the admittance matrix, by fitting the diagonal elements together, as against the conventional method of independently fitting individual admittance parameters and then collecting the common set of poles.
5. The method helps to keep the macromodel compact thereby resulting in efficient transient simulation.

CHAPTER 4 *Passive Macromodeling of Linear Subnetworks Characterized by Tabulated Data*

This chapter describes the proposed method to obtain passive macromodels of linear circuits, which are characterized by tabulated data. The method uses a pole-residue formulation to approximate the tabulated data. The poles are obtained using the technique described in Sec. 3.3. Once the poles are computed, the next step is to evaluate the corresponding residues. A new algorithm is presented to compute the residues by imposing certain linear constraints, which help in preserving the passivity of the network. This chapter also provides an efficient method to verify the passivity of the macromodel. The chapter begins by reviewing the passivity properties and its significance in macromodeling the measured/simulated data.

4.1 *Review of Passivity Properties*

Passivity implies that a network cannot generate more energy than it absorbs, and no passive termination of the network will cause the system to go unstable [88] - [90]. The loss of passivity can be a serious problem because transient simulations of the network may encounter artificial oscillations in the presence of passive terminations. This has already been illustrated in the Fig. 1.2. In addition, the loss

of passivity and its significance is illustrated in the Fig. 1.3. The passivity criteria in the frequency-domain and the time-domain are described below.

Frequency-Domain Passivity Criteria

A network with admittance matrix represented by $Y(s)$ is passive iff [88] - [90],

- 1) $Y(s^*) = Y^*(s)$ where ' $*$ ' is the complex conjugate operator.
- 2) $Y(s)$ is analytic in $Re(s) > 0$.
- 3) $Y(s)$ is a positive real (PR) matrix, that is the product $z^{*t}[Y'(s^*) + Y(s)]z \geq 0$ for all complex values of s with $Re(s) > 0$ and any arbitrary vector z (*this implies that in the practical case of networks with symmetric admittance matrices, $Real(Y(s)) = [Y'(s^*) + Y(s)]/2$ must be a nonnegative definite matrix for all s with $Re(s) > 0$*).

From [33], if the matrix rational function represented by $Y(s)$ has no poles on the closed right-half plane, condition (3) implies that

$$[Y'(j\omega^*) + Y(j\omega)] \geq 0, \text{ for } \omega \in \mathbb{R} \cup \infty \quad (4.1)$$

Time-Domain Passivity Criteria

Consider a multiport (m -input, m -output port) linear time-invariant system, with the minimum state-space representation as:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned}\tag{4.2}$$

where $\mathbf{x}(t) \in \Re^n$, $\mathbf{u}(t) \in \Re^m$, $\mathbf{y}(t) \in \Re^m$, $\mathbf{A} \in \Re^{n \times n}$, $\mathbf{B} \in \Re^{n \times m}$, $\mathbf{C} \in \Re^{m \times n}$, $\mathbf{D} \in \Re^{m \times m}$. The above system is passive iff [50], there exists a positive definite matrix $\mathbf{W} > 0$, with $\mathbf{W} \in \Re^{n \times n}$, $\mathbf{L} \in \Re^{m \times m}$, and $\mathbf{Q} \in \Re^{m \times n}$ such that

$$\mathbf{W}\mathbf{A} + \mathbf{A}'\mathbf{W} = -\mathbf{Q}'\mathbf{Q}\tag{4.3}$$

$$\mathbf{C} = \mathbf{B}'\mathbf{W} + \mathbf{L}'\mathbf{Q}\tag{4.4}$$

$$\mathbf{L}'\mathbf{L} = \mathbf{D}' + \mathbf{D}\tag{4.5}$$

The above set of equations (4.3)-(4.5) are called the Lure equations.

The following sections describe the proposed algorithm for macromodeling of linear subnetworks characterized by frequency-sampled data. The algorithm involves three steps. In the first step, the poles of the system are computed using the proposed multiport pole computation technique of Sec. 3.3. In the second step, the residues are computed by enforcing certain linear passivity conforming constraints, which help in preserving the passivity of the resulting macromodel. In the last step (i.e. third step), an efficient algorithm is presented to verify the passivity of the macromodel. We begin by the problem formulation, followed by the details of the above identified steps.

4.2 Problem Formulation

The tabulated data can be multi-port scattering ($\mathbf{S}$), admittance ($\mathbf{Y}$), impedance ($\mathbf{Z}$), transmission ($\mathbf{T}$) or hybrid ($\mathbf{H}$) parameters. Consider that the

frequency-domain Y -parameter data is given for a M -port measured network as below

$$\mathbf{Y}(s) = \begin{bmatrix} Y_{11}(s) & Y_{12}(s) & \dots & Y_{1M}(s) \\ Y_{21}(s) & Y_{22}(s) & \dots & Y_{2M}(s) \\ \vdots & \vdots & \ddots & \vdots \\ Y_{M1}(s) & Y_{M2}(s) & \dots & Y_{MM}(s) \end{bmatrix} \quad (4.6)$$

Each element of the above matrix is approximated by the pole-residue formulation

$$Y_{ij}(s) = \sum_{n=1}^q \frac{k_n^{i,j}}{s-p_n} + c^{i,j}; \quad i, j = 1, 2, \dots, M \quad (4.7)$$

where the residues (k_n) and poles (p_n) can be real or complex conjugate pairs, q is the number of poles and residues, c is the real constant. The objective is to determine these unknowns such that right hand side of (4.7) approximates the given data $Y_{ij}(s)$ (on the left hand side of (4.7)), while maintaining the accuracy and passivity.

4.2.1 Passivity Issues Related to Residue Computation

Once the poles have been calculated in Step 1 (as per Sec. 3.3), the next step is to calculate residues of each element of the admittance matrix in (4.6). Straightforward computation of residues as described in [41], does not guarantee the passivity of the resulting macromodel. To understand the proposed residue computation algorithm, let us revisit the passivity criteria for a passive network.

From Sec. 4.1 we know that for practical networks with symmetric admittance matrix, the passivity criteria can be expressed as

$$\text{Real}(Y(j\omega)) = [Y'(j\omega)^* + Y(j\omega)]/2 \geq 0 \quad (4.8)$$

for all $\omega \in \Re \cup \infty$. The relation operator $\geq$ in (4.8) means that $\text{Real}(Y(j\omega))$ must be positive semidefinite. This in turn means that the eigenvalues of $\text{Real}(Y(j\omega))$, which is real and symmetric, should be equal to or greater than zero [41].

However, guaranteeing the passivity of a macromodel is a difficult task [42], [51], as direct enforcement of (4.8) may lead to non-linear optimization and hence is likely to suffer from the problem of non-convergence and high CPU cost. The drawbacks of the existing techniques to address this issue has been discussed in detail in chapter 2.

4.3 *Proposed Passive Macromodeling Algorithm for Tabulated Data*

Each element of $Y(s)$ can be expressed in the pole-residue formulation as shown in (4.7) as:

$$Y^{i,j}(s) = c^{i,j} + \frac{k_1^{i,j}}{s-p_1} + \frac{k_2^{i,j}}{s-p_2} + \dots + \frac{k_q^{i,j}}{s-p_q}; \quad (i, j = 1, 2, \dots, M) \quad (4.9)$$

where, $p_1, p_2, \dots, p_q$ are the poles of the system, $k_1, k_2, \dots, k_q$ are the corresponding residues and c is the direct coupling constant. In order to address the difficulties pointed out in the previous section about passivity, we propose the following algorithm with linear constraints, which help to retain the passivity of the macromodel.

Let the tabulated data be given up to a maximum frequency ω_{max} . The common pole-set $\mathbf{P}$ obtained from Step 1, written in ascending order of their imaginary parts, can be expressed as:

$$\begin{aligned} \mathbf{P} &= [p_1, p_2 \dots p_{max0}, p_{max1}, \dots, p_q]; \\ \text{imag}(p_1) &< \dots < \dots < \text{imag}(p_q); \\ \text{imag}(p_{max0}) &< \omega_{max} < \text{imag}(p_{max1}) \end{aligned} \quad (4.10)$$

Consequently, the admittance matrix of (4.6) has the following form

$$\mathbf{Y}(s) = \begin{bmatrix} c^{1,1} + \sum_{n=1}^q \frac{k_n^{1,1}}{s-p_n} & c^{1,2} + \sum_{n=1}^q \frac{k_n^{1,2}}{s-p_n} & \dots & c^{1,M} + \sum_{n=1}^q \frac{k_n^{1,M}}{s-p_n} \\ \vdots & \vdots & & \vdots \\ c^{M,1} + \sum_{n=1}^q \frac{k_n^{M,1}}{s-p_n} & c^{M,2} + \sum_{n=1}^q \frac{k_n^{M,2}}{s-p_n} & \dots & c^{M,M} + \sum_{n=1}^q \frac{k_n^{M,M}}{s-p_n} \end{bmatrix} \quad (4.11)$$

The unknown residues $k_n^{i,j}$ and the direct coupling constants $c^{i,j}$ are computed by writing each entry of (4.11) at several frequency points, $\omega_1, \omega_2, \dots, \omega_k$. When $k > q + 1$, this results in an overdetermined problem of the form,

$$\mathbf{A}\mathbf{X}^{i,j} = \mathbf{b}^{i,j} \quad (4.12)$$

where, for the case of real poles, $\mathbf{A}, \mathbf{X}^{i,j}, \mathbf{b}^{i,j}$ are as follows

$$A = \begin{bmatrix} 1 & \frac{1}{s_1 - p_1} & \frac{1}{s_1 - p_2} & \dots & \frac{1}{s_1 - p_q} \\ 1 & \frac{1}{s_2 - p_1} & \frac{1}{s_2 - p_2} & \dots & \frac{1}{s_2 - p_q} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \frac{1}{s_k - p_1} & \frac{1}{s_k - p_2} & \dots & \frac{1}{s_k - p_q} \end{bmatrix} \quad (4.13)$$

$$X^{i,j} = \begin{bmatrix} c^{i,j} & k_1^{i,j} & k_2^{i,j} & \dots & k_q^{i,j} \end{bmatrix}^t \quad (4.14)$$

$$b^{i,j} = \begin{bmatrix} Y^{i,j}(s_1) \\ Y^{i,j}(s_2) \\ \vdots \\ Y^{i,j}(s_k) \end{bmatrix} \quad (4.15)$$

Notice in (4.14) that the residues corresponding to real poles will also be real. When the poles are complex, they come in complex-conjugate pairs and so do their corresponding residues. For complex poles (4.9) can be written as (with superscripts i,j ignored for the ease of presentation)

$$Y_{ij}(s) = c + \frac{k_{1R} + jk_{1I}}{s - p_1} + \frac{k_{1R} - jk_{1I}}{s - p_1^*} + \frac{k_{2R} + jk_{2I}}{s - p_2} + \frac{k_{2R} - jk_{2I}}{s - p_2^*} + \dots + \frac{k_{qR} + jk_{qI}}{s - p_q} + \frac{k_{qR} - jk_{qI}}{s - p_q^*} \quad (4.16)$$

where, the subscript 'R' and 'I' stand for the real and imaginary part of the residues, respectively, and '*' stands for the conjugate operator. The residues and the coupling constant can be computed by writing (4.16) at several frequency points. This again results in an overdetermined problem of (4.12). But in this case A , $X^{i,j}$ and $b^{i,j}$ are as follows:

$$A = \begin{bmatrix} 1 \frac{1}{s_1 - p_1} + \frac{1}{s_1 - p_1^*} & \frac{j}{s_1 - p_1} - \frac{j}{s_1 - p_1^*} & \cdots & \frac{1}{s_1 - p_q} + \frac{1}{s_1 - p_q^*} & \frac{j}{s_1 - p_q} - \frac{j}{s_1 - p_q^*} \\ 1 \frac{1}{s_2 - p_1} + \frac{1}{s_2 - p_1^*} & \frac{j}{s_2 - p_1} - \frac{j}{s_2 - p_1^*} & \cdots & \frac{1}{s_2 - p_q} + \frac{1}{s_2 - p_q^*} & \frac{j}{s_2 - p_q} - \frac{j}{s_2 - p_q^*} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 \frac{1}{s_k - p_1} + \frac{1}{s_k - p_1^*} & \frac{j}{s_k - p_1} - \frac{j}{s_k - p_1^*} & \cdots & \frac{1}{s_k - p_q} + \frac{1}{s_k - p_q^*} & \frac{j}{s_k - p_q} - \frac{j}{s_k - p_q^*} \end{bmatrix} \quad (4.17)$$

$$X^{i,j} = [c \ k_{1R} \ k_{1I} \ k_{2R} \ k_{2I} \ \dots \ k_{qR} \ k_{qI}]^t \quad (4.18)$$

$$b^{i,j} = \begin{bmatrix} Y^{i,j}(s_1) \\ Y^{i,j}(s_2) \\ \vdots \\ Y^{i,j}(s_k) \end{bmatrix} \quad (4.19)$$

Separating real and imaginary parts of the left and right hand sides, we can express (4.12) as;

$$[Re(A) + jIm(A)]X^{i,j} = Re(b^{i,j}) + jIm(b^{i,j}) \quad (4.20)$$

or, by equating the real and imaginary parts in (4.20) separately,

$$\begin{bmatrix} Re(A) \\ Im(A) \end{bmatrix} X^{i,j} = \begin{bmatrix} Re(b^{i,j}) \\ Im(b^{i,j}) \end{bmatrix} \quad (4.21)$$

Equation (4.21) is an overdetermined problem, which is solved by least squares method, subjected to the proposed passivity confirming linear constraints:

Solve

$$\begin{bmatrix} \text{Re}(A) \\ \text{Im}(A) \end{bmatrix}^t \begin{bmatrix} \text{Re}(A) \\ \text{Im}(A) \end{bmatrix} X^{i,j} = \begin{bmatrix} \text{Re}(A) \\ \text{Im}(A) \end{bmatrix}^t \begin{bmatrix} \text{Re}(b^{i,j}) \\ \text{Im}(b^{i,j}) \end{bmatrix} \quad (4.22)$$

such that,

a) For diagonal elements, i.e. $Y_{i,j}(s)$, $i = j$

- *The direct coupling constant:*

$$c^{i,i} \geq 0 \quad (C1)$$

- *The real part of the computed function is equal to or greater than the real part of the given tabulated data, i.e.*

$$\text{Re}(AX^{i,i}) \geq \text{Re}(b^{i,i}) \geq 0 \quad (C2)$$

b) For off-diagonal elements, i.e. $Y_{i,j}(s)$, $i \neq j$

- *The direct coupling constant:*

$$c^{i,j} = 0 \quad (C3)$$

- *Any pole beyond the maximum frequency of interest ω_{max} is ignored, i.e. the following poles are only considered from the common pole set in (4.10)*

$$\mathbf{P} = [p_1, p_2 \dots p_{max0}] \quad (C4)$$

The above linear constraints (C1)-(C4) help in preserving the passivity of the macromodel, by satisfying the passivity condition (4.8) (i.e., $\text{Real}(Y(j\omega)) = [Y^t(j\omega^*) + Y(j\omega)]/2 \geq 0$) in the region $0 \leq \omega \leq \infty$. In the

remainder of this section, a discussion of how this objective is achieved using the above constraints, is given.

a) Region $\omega = \infty$:

It is obvious from the constraints (C1) and (C3), that the condition (4.8) is satisfied at $\omega = \infty$.

b) Region $0 \leq \omega \leq \omega_{max}$:

Constraint (C2) achieves two objectives. The first is that, if in the given data, the real part of the diagonal element is very close to zero, this constraint ensures that the real part of the computed diagonal elements does not become negative due to numerical errors during the fitting process, i.e. it always keeps the real part greater than zero for diagonal elements. It is important to note that diagonally-dominant real symmetric matrices with positive diagonal entries are guaranteed to be positive-definite [96]. However, the reverse is not necessary. Constraint (C2) helps to maintain the diagonal dominance of the computed model either the same as or more than that of the given data. This helps to avoid the violation of the passivity condition (4.8) in the regions where the data is marginally passive. Detailed discussion of this constraint and its implication is given in Appendix-B. Also, its importance is numerically illustrated in Example 3 of Chapter 6.

c) Region $\omega_{max} \leq \omega \leq \infty$:

Constraint (C4), in conjunction with constraints (C1) and (C3), helps to satisfy the passivity condition (4.8) in this region. This is because, since any pole beyond ω_{max} is neglected for the non-diagonal elements, and also since $c^{i,j} = 0$ for non-diagonal elements, the response of non-diagonal elements beyond ω_{max} quickly tends to zero. In other words, these constraints avoid

any large variations (which could happen if poles beyond ω_{max} are considered for the non-diagonal elements) in non-diagonal element responses, thereby minimizing the possibility of violation of the passivity condition (4.8) in this region.

A graphical description of the impact of these constraints while satisfying passivity condition (4.8), at various frequency regions is given in Fig. 4.1.

Fig. 4.1. Illustration of the general behavior of $z^{*t}[Re(Y(j\omega))]z$ when using the proposed set of constraints

The advantage of the above constraints is that they are linear in nature and hence help in obtaining a passive macromodel from measured data in CPU efficient manner.

4.4 Macromodel Synthesis

Once the matrix rational function (4.11) is obtained for the admittance matrix $Y(s)$, it can be easily converted into the time-domain macromodel:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned}\tag{4.23}$$

such that,

$$\mathbf{Y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}\tag{4.24}$$

Details about obtaining time-domain macromodel of (4.23) from (4.11), is explained in Sec. 5.1.

4.5 *Proposed Passivity Check*

Although the constraints described in Sec. 4.3 help in obtaining passive macomodels, they are not strictly passivity enforcing. Hence, the macromodel obtained in (4.23) is checked for passivity. This section describes the proposed Passivity check of the macromodel.

Consider the state-space representation

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned}\tag{4.25}$$

such that all eigenvalues of $\mathbf{A}$ have negative real part, $(\mathbf{A}, \mathbf{B})$ is controllable and $\mathbf{D} + \mathbf{D}^t > 0$. The state-space system in (4.25) is passive iff [53], there exists a real positive definite matrix $\mathbf{W} = \mathbf{W}^t$ satisfying the Algebraic Riccati Equation

$$\mathbf{A}^t\mathbf{W} + \mathbf{W}\mathbf{A} + (\mathbf{W}\mathbf{B} - \mathbf{C}^t)(\mathbf{D} + \mathbf{D}^t)^{-1}(\mathbf{W}\mathbf{B} - \mathbf{C}^t)^t = 0\tag{4.26}$$

Equation (4.26) is solved by *care.m* function of MATLAB [95].

The above method of passivity check is efficient than the ones used in conventional techniques as it does not involve any frequency sweeping and puts no restrictions on the individual pole-residue pair.

CHAPTER 5 *Time-Domain Macromodel Synthesis and Unified Transient Analysis*

In chapters 3 and 4, we looked at the proposed passive macromodeling of subnetworks characterised by tabulated data. The first part of this chapter presents the time-domain macromodel synthesis from the frequency-domain pole-residue model. The second part presents a technique to extend the modified nodal analysis (MNA) [87] to interconnect subnetworks described by tabulated data, for the purpose of transient analysis. The third part presents the unified transient simulation of the entire nonlinear network including measured subnetworks.

5.1 *Macromodel Synthesis*

In chapter 4, a new algorithm was presented to represent the transfer function $Y_m(s)$ of measured subnetwork by pole-residue formulation ((4.11) of Sec. 4.3). However, it is to be noted that this formulation is in the frequency-domain, while the non-linear terminations are best described in the time-domain. This is a well known mixed frequency/time problem. In order to overcome this difficulty, the transfer function described by pole-residue model is translated into a time-domain form described by ordinary differential equations (ODEs). Derivation of differential equations from reduced-order frequency-domain models of linear

subnetworks is referred to as *macromodel synthesis*. This process is illustrated in Fig. 5.1.

Fig. 5.1. Illustration of time-domain macromodel synthesis from pole-residue formulation

In general, a set of first-order differential equations in the state-space domain can be described as

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned}\tag{5.1}$$

where $\mathbf{x}(t)$ is the state vector of length n , $\mathbf{u}(t)$ is the input vector of length n_π (where n_π is the number of ports). $\mathbf{A} \in \mathfrak{R}^{n \times n}$ is the state-matrix, $\mathbf{B} \in \mathfrak{R}^{n \times n_\pi}$ is the matrix that relates the inputs to state-variables, $\mathbf{C} \in \mathfrak{R}^{n_\pi \times n}$ is the matrix relating state variables to the outputs ($\mathbf{y}(t)$) and $\mathbf{D} \in \mathfrak{R}^{n_\pi \times n_\pi}$ is a matrix relating inputs directly to the output.

Given a matrix-transfer function described by (4.11) of chapter 4, several forms of time-domain realization can be obtained. For the purpose of illustration, macromodel synthesis using Jordan-canonical [92] - [94] form of realization is presented below.

Example: Macromodel Synthesis

Consider a two-port network containing two common poles, whose transfer function is described by

$$\begin{bmatrix} c^{1,1} + \sum_{a=1}^2 \frac{k_a^{1,1}}{s-p_a} & c^{1,2} + \sum_{a=1}^2 \frac{k_a^{1,2}}{s-p_a} \\ c^{2,1} + \sum_{a=1}^2 \frac{k_a^{2,1}}{s-p_a} & c^{2,2} + \sum_{a=1}^2 \frac{k_a^{2,2}}{s-p_a} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}$$

A Jordan-canonical [92], [94] form of realization for this case would need four state variables and it can be represented as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} p_1 & 0 & 0 & 0 \\ 0 & p_1 & 0 & 0 \\ 0 & 0 & p_2 & 0 \\ 0 & 0 & 0 & p_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$y = \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} k_1^{1,1} & k_1^{1,2} & k_2^{1,1} & k_2^{1,2} \\ k_1^{2,1} & k_1^{2,2} & k_2^{2,1} & k_2^{2,2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} c^{1,1} & c^{1,2} \\ c^{2,1} & c^{2,2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

5.1.1 Including Complex Poles in the State Matrix

In case the matrix-transfer function $Y_m(s)$ has complex poles, then they need to be treated differently as they do not have a direct meaning in the time-domain. However, since all the complex poles and residues of $Y_m(s)$ (obtained in chapter 4) come in pairs with their complex conjugate, a Jordan-form of state-space realization for a complex pole-pair would be [94]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ 0 & A_1^* \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_1^* \end{bmatrix} u \quad (5.2)$$

$$y = \begin{bmatrix} C_1 & C_1^* \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (5.3)$$

where A_1 is a diagonal matrix constructed using the poles and A_1^* is the complex conjugate of A_1 . Next, introducing an equivalence transformation defined by

$$\begin{bmatrix} \hat{x} \end{bmatrix} = \begin{bmatrix} T \end{bmatrix} \begin{bmatrix} x \end{bmatrix}; \quad T = \begin{bmatrix} I & I \\ iI & -iI \end{bmatrix} \quad (5.4)$$

Equations represented by (5.2) and (5.3) can now be easily transformed into

$$\begin{bmatrix} \hat{\dot{x}}_1 \\ \hat{\dot{x}}_2 \end{bmatrix} = \begin{bmatrix} Re(A_1) & Im(A_1) \\ -Im(A_1) & Re(A_1) \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} 2Re(b_1) \\ -2Im(b_1) \end{bmatrix} u \quad (5.5)$$

$$y = \begin{bmatrix} Re(C_1) & Im(C_1) \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} \quad (5.6)$$

An illustrative example of the above steps is given below.

Example: Inclusion of Complex Poles in the State Matrix

Consider a two port network containing one pair of complex poles $\lambda_{1,2} = w \pm iz$. Let the corresponding residues at different ports be $c_{i,j} = (r \pm iv)_{i,j}$. Original Gilbert's realization will yield

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} w + iz & 0 & 0 & 0 \\ 0 & w + iz & 0 & 0 \\ 0 & 0 & w - iz & 0 \\ 0 & 0 & 0 & w - iz \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$y = \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} (r + iv)_{11} & (r + iv)_{12} & (r - iv)_{11} & (r - iv)_{12} \\ (r + iv)_{21} & (r + iv)_{22} & (r - iv)_{21} & (r - iv)_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

By defining the transformation matrix $T = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ i & 0 & -i & 0 \\ 0 & i & 0 & -i \end{bmatrix}$ of (5.4), the realization

represented by (5.5) and (5.6) can be obtained as

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix} = \begin{bmatrix} w & 0 & z & 0 \\ 0 & w & 0 & z \\ -z & 0 & w & 0 \\ 0 & -z & 0 & w \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$y = \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & v_{11} & v_{12} \\ r_{21} & r_{22} & v_{21} & v_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix}$$

5.2 General Formulation of Network Equations

Consider a general network containing an arbitrary number of lumped linear and nonlinear components and a linear tabulated subnetwork (Fig. 5.2). The linear lumped components may be described by equations in either the time-domain or the frequency-domain, whereas the nonlinear components are generally described by time-domain equations. The linear tabulated components are assumed to be described in the frequency-domain. For simplicity, let all the lumped linear and non-linear components be grouped into a single network ϕ and the tabulated component be represented by subnetwork π .

Fig. 5.2. Lumped network ϕ containing tabulated subnetwork π

Without loss of generality, the MNA matrix [54] for the network ϕ can be written as

$$W_{\phi} \frac{d}{dt} \mathbf{v}_{\phi}(t) + \mathbf{G}_{\phi} \mathbf{v}_{\phi}(t) + \mathbf{L}_{\pi} \mathbf{i}_{\pi}(t) + \mathbf{F}(\mathbf{v}_{\phi}(t)) - \mathbf{b}_{\phi}(t) = 0, \quad t \in [0, T] \quad (5.7)$$

where

- $\mathbf{v}_{\phi}(t) \in \mathcal{R}^{N_{\phi}}$ is the vector of node voltage waveforms appended by independent voltage source current, linear inductor current, nonlinear capacitor charge and nonlinear inductor flux waveforms,

-
- $W_\phi \in \mathbb{R}^{N_\phi \times N_\phi}$ and $G_\phi \in \mathbb{R}^{N_\phi \times N_\phi}$ are constant matrices describing the lumped memory and memoryless elements of network ϕ , respectively,
 - $b_\phi \in \mathbb{R}^{N_\phi}$ is a constant vector with entries determined by the independent voltage and current sources,
 - $F(v_\phi)$ is a function describing nonlinear elements of the circuit,
 - $L_\pi = [l_{i,j}]$ with elements $l_{i,j} \in \{0, 1\}$ where $i \in \{1, \dots, N_\phi\}, j \in \{1, \dots, n_\pi\}$ with a maximum of one nonzero in each row or column, is a selector matrix that maps $i_\pi(t) \in \mathbb{R}^{n_\pi}$ the vector of currents entering the linear subnetwork π , into the node space $\mathbb{R}^{N_\phi}$ of the network ϕ ,
 - N_ϕ is the total number of variables in the MNA formulation and n_π is the total number of ports associated with the linear subnetwork π .

The first, second and fourth terms in (5.7) cover the network's lumped components, while the fifth term covers independent sources. The third term describes currents at linear subnetwork π (network characterised by tabulated data) terminals and then maps them into the rest of the network through matrix L_π . The linear multi-terminal subnetwork π can be characterized in the frequency-domain by its terminal behavior. Without loss of generality, the terminal relations for subnetwork π can be represented by frequency-domain equations in the form

$$Y_\pi(s)V_\pi(s) = I_\pi(s) \quad (5.8)$$

where

- $Y_\pi(s)$ is the complex frequency-domain admittance representation of subnet-
-

work π

- $V_\pi(s)$ is the vector of terminal voltages (at nodes that connect the subnetwork to the network ϕ)
- $I_\pi(s) = L(i_\pi(t))$; L denotes the Laplace transform, s is the complex frequency.

Example

To illustrate the formulation scheme described above, an example is presented below. Consider the circuit shown in Fig. 5.3. The network equations for this circuit can be represented by (5.7) and (5.8) where

Fig. 5.3. Example circuit with lumped linear, nonlinear & measured devices

$$G_\phi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & G_1 & -G_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -G_1 & G_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & G_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}; \quad W_\phi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C & -C & 0 & 0 \\ 0 & 0 & 0 & 0 & -C & C & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$F(v_\phi(t)) = \begin{bmatrix} -f_1(v_2 - v_1) \\ f_1(v_2 - v_1) \\ 0 \\ 0 \\ f_2(v_5) \\ 0 \\ 0 \\ -f_3(v_6) \end{bmatrix}; \quad v_\phi = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ i_e \\ q \end{bmatrix}; \quad b_\phi = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ e \\ 0 \end{bmatrix};$$

$$L_{\pi a} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \quad L_{\pi b} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \quad i_{\pi a}(t) = \begin{bmatrix} I_{a1}(t) \\ I_{a2}(t) \end{bmatrix}; \quad i_{\pi b}(t) = \begin{bmatrix} I_{b1}(t) \\ I_{b2}(t) \end{bmatrix}$$

$$\mathbf{I}_{\pi a}(s) = \begin{bmatrix} I_{a1}(s) \\ I_{a2}(s) \end{bmatrix}; \quad \mathbf{I}_{\pi b}(s) = \begin{bmatrix} I_{b1}(s) \\ I_{b2}(s) \end{bmatrix}; \quad \mathbf{V}_{\pi a}(s) = \begin{bmatrix} V_3(s) \\ V_4(s) \end{bmatrix}; \quad \mathbf{V}_{\pi b}(s) = \begin{bmatrix} V_4(s) \\ V_5(s) \end{bmatrix}$$

$$\mathbf{Y}_{\pi a}(s) = \begin{bmatrix} Y_{11a}(s) & Y_{12a}(s) \\ Y_{21a}(s) & Y_{22a}(s) \end{bmatrix}; \quad \mathbf{Y}_{\pi b}(s) = \begin{bmatrix} Y_{11b}(s) & Y_{12b}(s) \\ Y_{21b}(s) & Y_{22b}(s) \end{bmatrix}$$

5.3 Unified Transient Simulation

Once a matrix-transfer function describing the multiport measured network is obtained, a time-domain realization in the form of state-space equations can be obtained using the techniques described in Sec. 5.1 as

$$\begin{bmatrix} \frac{d}{dt} \mathbf{z}_{\pi}(t) - \mathbf{A}_{\pi} \mathbf{z}_{\pi}(t) - \mathbf{B}_{\pi} \mathbf{v}_{\pi}(t) = \mathbf{0} \\ \mathbf{i}_{\pi}(t) - \mathbf{C}_{\pi} \mathbf{z}_{\pi}(t) - \mathbf{D}_{\pi} \mathbf{v}_{\pi}(t) = \mathbf{0} \end{bmatrix} \quad (5.9)$$

where $\mathbf{i}_{\pi}$ and $\mathbf{v}_{\pi}$ are the vector of terminal currents and voltages of the linear subnetwork π (described by (5.8)). The differential equations represented by the macromodel (5.9) can be combined with (5.7) using the relation $\mathbf{v}_{\pi} = (\mathbf{L}_{\pi})^t \mathbf{v}_{\phi}$ as

$$\begin{aligned} \frac{d}{dt} \mathbf{z}_{\pi}(t) - \mathbf{A}_{\pi} \mathbf{z}_{\pi}(t) - \mathbf{B}_{\pi} (\mathbf{L}_{\pi})^t \mathbf{v}_{\phi}(t) &= \mathbf{0} \\ \mathbf{i}_{\pi}(t) - \mathbf{C}_{\pi} \mathbf{z}_{\pi}(t) - \mathbf{D}_{\pi} (\mathbf{L}_{\pi})^t \mathbf{v}_{\phi}(t) &= \mathbf{0} \\ \mathbf{W}_{\phi} \frac{d}{dt} \mathbf{v}_{\phi}(t) + \mathbf{G}_{\phi} \mathbf{v}_{\phi}(t) + \mathbf{L}_{\pi} \mathbf{i}_{\pi}(t) + \mathbf{F}(\mathbf{v}_{\phi}(t)) - \mathbf{b}_{\phi}(t) &= \mathbf{0} \end{aligned} \quad (5.10)$$

Using standard nonlinear solvers or any of the general-purpose circuit simulators the unified set of differential equations represented by (5.10) can be solved to

yield global transient solutions for the entire nonlinear circuit consisting of high-frequency measured subnetworks. Fig. 5.4 shows the process of unified transient simulation involving tabulated network in the presence of nonlinear circuit elements.

Fig. 5.4. Inclusion of tabulated network in overall transient simulation

For those simulators (such as HSPICE) which do not directly accept the differential equations as input, the macromodel represented by (5.9) can be converted to an equivalent subcircuit, and is described in the next section.

5.3.1 Conversion of Macromodels to Equivalent Subcircuits

Conversion of differential equations to equivalent subcircuits can be accomplished in several ways. For this purpose a new technique which is easy to understand as well as to implement, is developed. For the purpose of illustration,

consider a simple case of two port network with two states represented in the form of (5.9)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (5.11)$$

$$\begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (5.12)$$

Next, (5.11) and (5.12) can be rearranged as

$$\dot{x}_1 = a_{11}x_1 + a_{12}x_2 + b_{11}v_1 + b_{12}v_2 \quad (5.13)$$

$$\dot{x}_2 = a_{21}x_1 + a_{22}x_2 + b_{21}v_1 + b_{22}v_2 \quad (5.14)$$

$$i_1 = c_{11}x_1 + c_{12}x_2 + d_{11}v_1 + d_{12}v_2 \quad (5.15)$$

$$i_2 = c_{21}x_1 + c_{22}x_2 + d_{21}v_1 + d_{22}v_2 \quad (5.16)$$

In the above equations, the port voltages and currents are represented by v_1, v_2 and i_1, i_2 , respectively. An equivalent network representing (5.13) - (5.16) can be constructed as shown in Fig. 5.5. Each state in the macromodel requires a separate node in the equivalent circuit and are represented by nodes n_1, n_2 . The state variables x_1, x_2 can be represented by the capacitor voltages. These capacitors are denoted by C_{n1}, C_{n2} and the corresponding voltages by v_{n1}, v_{n2} . Next, the terms such as $a_{ij}x_j$ and $c_{ij}x_j$ in (5.13) - (5.16) can be represented by voltage controlled current sources. Equations (5.13) and (5.14) are fully represented by Fig. 5.5(c) and Fig. 5.5(d). Output equations represented by (5.15) and (5.16) are realized through

equivalent circuits shown in Fig. 5.5(a) and Fig. 5.5(b). Generalization of the above discussion in the presence of more number of states or ports is straightforward.

(a): Realization of (5.15)

(b): Realization of (5.16)

(c): Realization of (5.13)

(d): Realization of (5.14)

Fig. 5.5. Illustration of equivalent subcircuit generation from macromodels

5.3.2 Transient Simulation of Rational Functions Using HSPICE Laplace elements

The pole-residue frequency-domain macromodel can be included in the overall transient simulation using a feature of HSPICE (a widely used industrial simulator [97]), called Laplace elements. By using this feature, the pole-residue formulation is converted into either poles-zeros or polynomial rational function.

The poles-zeros or the coefficients of the polynomials of rational functions are used to represent the macromodel. An example is given below, which explains the procedure of using Laplace element feature.

Consider a one-port network in Fig. 5.6. Assume that this network is represented by the polynomial rational function (admittance form)

$$Y(s) = \frac{a_0 + a_1s + a_2s^2}{b_0 + b_1s + b_2s^2} \quad (5.17)$$

Fig. 5.6. Example (Laplace element)

A sample HSPICE netlist for the above circuit consisting of one-port network described by rational function can be written as

```
*Laplace element example
.tran 0 0.02ns 5ns
.option accurate
lin in 0 1A
R in 1 10
Gy 1 0 LAPLACE 1 0 a0 a1 a2 / b0 b1 b2
.end
```

CHAPTER 6 *Computational Results*

This chapter presents computational results to validate the accuracy and efficiency of the proposed passive macromodeling algorithm described in chapter 4. Here, three examples are presented. The first example presents a simple case, where the original data contained only a few poles. The second example consists of macromodeling of 2-port tabulated subnetwork, where the data corresponds to that of a high-speed distributed interconnect network. The third example considers a 3-port subnetwork and also discusses various macromodeling issues that may arise due to the non-availability of the dominant information in the given data. Transient analysis results in the presence of non-linear devices are presented for all the examples.

Example 1- Two Port R,L,C Circuit: In this example, a relatively small RLC network (Fig. 6.1) is analyzed using the proposed technique. The linear part of the network is simulated using HSPICE to obtain admittance parameters at 1000 frequency points up to 6GHz, and is considered as tabulated data input to the proposed algorithm (referred to as original data henceforth). Using the proposed method on the multiport data, a common pole set (with two real poles and three complex poles) and subsequently residues are computed (given in Table 6.1; values are scaled by $1e9$). Fig. 6.2 shows the comparison of frequency responses obtained using the proposed method and original data, and they match accurately. Next, the time-domain macromodel is obtained in the form of state-space equations using the steps described in Sec. 5.1. The macromodel in this case

consists of $8 \times 2 = 16$ states, with $A \in R^{16 \times 16}$, $B \in R^{16 \times 2}$, $C \in R^{2 \times 16}$, $D \in R^{2 \times 2}$. Next, the macromodel is checked for passivity by solving the Algebraic Riccati equation (4.26). In this case a symmetric positive definite matrix W was obtained satisfying (4.26) (i.e., the macromodel is passive). This is also demonstrated using the conventional approach, by plotting the eigenvalues of the $Re(Y(j\omega))$ against frequency up to 50GHz (Fig. 6.3). Since the eigenvalues are positive even up to 50GHz and also, that the direct coupling constants are positive/zero for diagonal/non-diagonal elements, it can be interpreted that the macromodel is passive. Next, the macromodel obtained from the proposed method is linked to HSPICE and a nonlinear analysis is performed. The network is excited with a trapezoidal pulse having a rise/fall time of 0.1ns and a pulse width of 4ns. Responses at nodes P_1 , P_2 and V_{out} are compared with the HSPICE simulation of the original network in Fig. 6.4. As seen, the results from both the methods match accurately

Fig. 6.1. Nonlinear circuit containing RLC lumped network

Table 6.1: Poles, residues and direct coupling constants obtained using the proposed method (Example 1)

Poles	R^{11}	R^{12} & R^{21}	R^{22}
-3.3385	1.2184e-001	-1.2723e-001	1.3286e-001
-6.2928	-7.3081e-009	-2.9387e-010	8.1838e-010
-1.9886e+000+1.7803e+001i	1.2620e+000-5.3718e-002i	1.9255e-001+3.8040e-002i	2.7689e-002+1.2787e-002i
-1.6406e+000 +2.4315e+001i	2.2657e-001+9.3384e-001i	-1.5918e-001+-4.1527e-002i	2.7124e-002+7.5762e-003i
-4.0573e-001 +3.2744e+001i	2.4330e-001+1.1006e-001i	3.0242e-002+1.2548e-002i	3.7552e-003+1.4209e-003i
Direct Coupling Constant	$D^{11} = 1.9291e-06$	$D^{12} = D^{21} = 0$	$D^{22} = 6.8651e-08$

Fig. 6.2. Frequency responses for Example 1

Fig. 6.3. Eigenvalues of $\text{Real}(Y(s))$ (Example 1)

Fig. 6.4. Time-domain responses for Example 1

Example 2: Two Port Coupled Transmission Line Circuit

In this example, circuit consisting of lumped components along with the coupled transmission lines and nonlinear terminations is considered. The circuit is shown in the Fig. 6.5. As can be seen, the linear part of the circuit consists of two ports. In order to show the accuracy and efficiency of the proposed method, the frequency-domain simulated data (admittance parameters) obtained by analyzing the circuit by HSPICE, is considered as the tabulated data (referred to as original data henceforth). By using the proposed passive macromodeling algorithm, the multiport data is approximated accurately, with 6 complex poles and one real pole. Fig. 6.6 shows the comparison of frequency responses obtained by the proposed method and the original data, and they match accurately. The state-space representation of the computed macromodel has the following dimensions: Number of states used = $13 \times 2 = 26$ states, $A \in R^{26 \times 26}$, $B \in R^{26 \times 2}$, $C \in R^{2 \times 26}$ and $D \in R^{2 \times 2}$. The macromodel is checked for passivity by solving the Algebraic Riccati equation in (4.26). A symmetric positive definite matrix W is obtained satisfying the Algebraic Riccati equation. This means that the macromodel is passive. This is also demonstrated using the conventional approach, by plotting the eigenvalues of the $Re(Y(j\omega))$ against frequency up to 50GHz (Fig. 6.7). Since the eigenvalues are positive even up to 50GHz and also, that the direct coupling constants are positive/zero for diagonal/non-diagonal elements, it can be interpreted that the macromodel is passive. Next, the macromodel is linked to HSPICE and transient analysis of the entire nonlinear circuit is performed. The network is excited with a trapezoidal pulse having a rise/fall time of 0.1ns. Responses at nodes P_1 , P_2 and d_2 are compared with the HSPICE simulation of the original network in Fig. 6.8. As seen, the results from both the methods match accurately.

Fig. 6.6. Frequency responses (Example 2)

Fig. 6.7. Eigenvalues of $\text{Real}(Y(s))$ (Example 2)

Fig. 6.8. Time-domain responses (Example 2)

Example 3: Three Port Subnetwork Characterised by Tabulated Data

In this example a 3-port subnetwork consisting of seven lossy transmission lines (Fig. 6.9) is considered. As in the previous two examples, to validate the accuracy of the proposed method, a HSPICE simulation of the circuit is performed to

obtain the admittance parameters of the subnetwork. This data (referred to as original data) is considered as the input to the proposed passive macromodeling algorithm. In order to validate the efficiency and accuracy of the proposed algorithm, this network is analysed for two different cases of maximum frequencies of interest. This is because depending on the location of the f_{max} (the point up to which the data is given), the dominant information in the response may be missing from the data being macromodeled. This kind of situation poses significant difficulties to the existing macromodeling algorithms in the literature, both with respect to passivity and accuracy issues. For example, referring to Fig. 6.10, if the data for Y_{33} is given only up to 5GHz then the resulting macromodel may not capture the most dominant pole (at approximately 5.25GHz), which can lead to inaccuracies at lower frequencies. This in turn can lead to passivity problems within the frequency region up to f_{max} . On the other hand if that pole is picked, the resulting fit may be such that it leads to passivity violation beyond f_{max} .

Case 1: Data up to 6GHz

In this case the data (Y-parameter) is given up to 6GHz, at 1000 sample points. and is fitted using the proposed method (20 complex poles and 4 real poles were required). Fig. 6.10 and Fig. 6.11 show the comparison between the original data and the response of the proposed model and they match accurately. Fig. 6.12 and Fig. 6.13 illustrate the usefulness of constraint (C2) of Sec. 4.3. Fig. 6.12(a) shows (using enlarged vertical axis) the comparison of original $real(Y_{33}(s))$ with the response of the computed model obtained without using constraint (C2) of Sec. 4.3. As seen, at certain frequencies, the fitted response is slightly smaller than the original data and the corresponding eigenvalue spectrum (Fig. 6.12(b))

contains several negative values. Fig. 6.13 shows the improved results when the constraint (C2) is enforced and the eigenvalue spectrum in this case always remained positive (Fig. 6.13(b)).

The model is checked for passivity by solving the Algebraic Riccati equation (4.26) which resulted in a symmetric positive definite matrix W satisfying (4.26) (i.e., the macromodel is passive). This is also demonstrated using the conventional approach, by plotting the eigenvalues of $Re(Y(j\omega))$ against frequency up to 50GHz (Fig. 6.14). Since the eigenvalues are positive even up to 50GHz and also, that the direct coupling constants are positive/zero for diagonal/non-diagonal elements, it can be interpreted that the macromodel is passive. Next the macromodel is linked to HSPICE and a non-linear analysis is performed. The network is excited by a pulse having a rise and fall time of 0.1 ns. The results at node P_1 , P_2 , P_3 are shown in Fig. 6.15, while results at V_{out2} and V_{out3} are shown in Fig. 6.16, in comparison to the HSPICE simulation of the original circuit (from which the measured data was obtained) and they match accurately.

Fig. 6.10. Frequency responses (magnitude) for Example 3

Fig. 6.11. Frequency responses (real part) for Example 3

Regions showing real part of fitted data less than the original data

Fig. 6.12. Illustration of passivity violation with unconstrained fitting

(a)

(b)

Fig. 6.13. Passivity preservation using the proposed algorithm

Fig. 6.14. Passivity proof for the proposed algorithm (conventional approach)

Fig. 6.15. Transient results

Fig. 6.16. Transient results

Case 2: Data up to 3GHz

In this case the original data is considered only up to 3GHz. Similar analysis as in the previous case is carried out and the corresponding results are shown in Fig. 6.17 to Fig. 6.23. The proposed method could obtain a passive model even when the data is truncated at 3GHz (note that the given circuit contains the most dominant information just after 3GHz).

Fig. 6.17. Frequency responses (magnitude) for Example 3

Fig. 6.18. Frequency responses (real part) for Example 3

Regions showing real part of fitted data less than the original data

Fig. 6.19. Illustration of passivity violation with unconstrained fitting

Fig. 6.20. Passivity preservation using the proposed algorithm

Fig. 6.21. Passivity proof for the proposed algorithm (conventional approach)

Fig. 6.22. Transient results

Fig. 6.23. Transient results

This chapter contains a summary of the work that was presented in this thesis. In addition, the directions for future work are discussed.

7.1 Summary

In this thesis an efficient multiport algorithm is presented for the modeling and simulation of interconnect networks characterized by tabulated (measured/simulated) data in the presence of nonlinear elements. A new methodology is presented which allows the computation of common multiport poles in an efficient manner. In addition, a new algorithm is presented to compute the residues while imposing certain linear passivity conforming constraints. Also, a new algorithm is presented to efficiently verify the macromodel passivity.

The algorithm presented here provides an easy means to handle high-speed interconnect circuits described by measured/simulated data, in industrial grade simulators such as SPICE. The major advantages of the proposed algorithm are as follows.

- 1) The algorithm does not suffer from the problem of ill-conditioning while capturing the wide-band frequency response of the original network. Hence, the algorithm puts no restrictions on the highest frequency range and pro-

vides a scheme to preserve the frequency-spectrum of the original network up to the maximum frequency of interest.

- 2) The method extracts the poles of the system by using a multiport rational approximation approach, thereby ensuring that a common pole-set is extracted for the entire multiport system. This in turn helps to keep the order of the macromodel to the minimum and enables efficient transient simulation.
 - 3) A new multiport residue computation algorithm has been developed which helps to preserve the passivity of the macromodel. For this purpose, a new set of linear passivity conforming constraints are developed. Since these constraints are linear, they enable fast computation of macromodels (unlike the traditional approaches which require nonlinear optimization)
 - 4) An efficient passivity check algorithm is proposed to check for the passivity of the resulting macromodel. This algorithm is able to detect any passivity violation without resorting to frequency sweep as in some conventional approaches.
 - 5) An efficient technique for the synthesis of macromodel from frequency-domain reduced-order models is presented. The method also discusses the inclusion of complex poles during state-space synthesis. Also, the proposed method overcomes the problem of inclusion of multiport measured/simulated subnetworks in the overall transient simulation in the presence of other nonlinear devices.
-

7.2 *Future Research*

The following future research directions are proposed:

- 1) **Detection of location of passivity violation:** The passivity checking algorithm presented can only tell whether the macromodel is passive or not. However, in case of passivity violation, the key information which is required to carry out compensation, is the exact location where the violation occurs. Further work is required in this direction to detect the locations of passivity violation.
 - 2) **Passivity compensation:** Since the proposed linear passivity conforming constraints are not strictly passivity enforcing, there may be minor chances of passivity violation (as no constraints were imposed on the behaviour of diagonal element after ω_{max}). Hence, further work may be required to come up with efficient mechanism of passivity compensation, for cases with minor passivity violaton.
 - 3) **Passivity by formulation:** In order to avoid the intermediate steps of passivity check and compensation, it will be ideal and desirable to enforce the passivity criteria while generating the macromodel. However, care will have to be taken that the passivity imposing constraints are efficient, to enable the generation of macromodel in a reasonable amount of time. This remains an open research problem.
-

References

- [1] H. B. Bakoglu, *Circuits, Interconnections and packaging for VLSI*, Addison-Wesley, Reading MA, 1990.
- [2] W. W. M. Dai (Guest Editor), "Special issue on simulation, modeling, and electrical design of high-speed and high-density interconnects," *IEEE Transactions on Circuits and Systems*, vol. 39, no. 11, pp. 857-982, Nov. 1992.
- [3] M. Nakhla, R. Achar and R. Khazaka, *Modeling and Simulation of High-Speed VLSI Interconnects*, CHAPTER IV: *Circuits And Systems In The Information Age*, New Jersey: IEEE Publishers, pp. 187-215, 1997.
- [4] A. Deustsch, "Electrical characteristics of interconnections for high-performance systems," *Proceedings of the IEEE*, vol. 86, no. 2, pp. 315-355, Feb. 1998.
- [5] C. Paul, *Analysis of Multiconductor Transmission Lines*, New York: John Wiley and Sons Inc., 1994.
- [6] S. S. Gao, A. Y. Yang and S. M. Kang, "Modeling and simulation of interconnection delays and crosstalks in high-speed integrated circuits," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 1-9, Jan. 1990.
- [7] H. Hasegawa and S. Seki, "Analysis of interconnection delay on very high-speed LSI/VLSI chips using a microstrip line model," *IEEE Trans. Electron Devices*, vol. ED-31, pp. 1954-1960, Dec 1984.
- [8] F. Y. Chang, "The generalized method of characteristics for waveform relaxation analysis of lossy coupled transmission lines," *IEEE Trans. Microwave Theory Tech.*, vol. 37, pp. 2028-2038, Dec. 1989.
- [9] N. Orhanovic, P. Wang, and V. K. Tripathi, "Generalized method of characteristics for time domain simulation of multiconductor lossy transmission lines," *Proceedings. IEEE Symposium on Circuits and Systems*, May 1990.
- [10] E. C. Chang and S. M. Kang, "Computationally efficient simulation of a lossy transmission line with skin effect by using numerical inversion of Laplace transform," *IEEE Transactions on Circuits and Systems*, vol. 39, pp.

861-868, July 1992.

- [11] R. Griffith and M. Nakhla, "Mixed frequency/time domain analysis on non-linear circuits," *IEEE Trans. Computer-Aided Design*, vol. 10, no. 8, pp. 1032-1043, Aug. 1992.
 - [12] R. Wang and O. Wing, "Transient analysis of dispersive VLSI interconnects terminated in nonlinear loads," *IEEE Trans. Computer-Aided Design*, vol. 11, pp. 1258-1277, Oct. 1992.
 - [13] S. Lin and E. S. Kuh, "Transient simulation of lossy interconnects based on the recursive convolution formulation," *IEEE Transactions on Circuits and Systems*, vol. 39, no. 11, pp. 879-892, Nov. 1992.
 - [14] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 352-366, Apr. 1990.
 - [15] E. Chiprout and M. Nakhla, *Asymptotic Waveform Evaluation and Moment Matching for Interconnect Analysis*, Boston: Kluwer Academic Publishers, 1993.
 - [16] S. Kumashiro, R. A. Rohrer, A. J. Strojwas, "Asymptotic waveform evaluation for transient analysis of 3-D interconnect structures," *IEEE Trans. Computer-Aided Design*, vol. 12, no. 7, pp. 988-996, 1993.
 - [17] X. Huang, "Padé Approximation of linear(ized) circuit responses," Ph.D. dissertation, Carnegie Mellon Univ., Nov 1990.
 - [18] T. Tang and M. Nakhla, "Analysis of high-speed VLSI interconnect using the asymptotic waveform evaluation technique," *IEEE Trans. Computer-Aided Design*, vol 11, pp. 2107-2116 Mar. 1992.
 - [19] R. Khazaka, J. Poltz, M. Nakhla, and Q. J. Zhang, "A fast method for the simulation of lossy interconnects with frequency dependent parameters," *Proceedings IEEE Multi-Chip Module Conference*, (Santa Cruz, CA), pp. 95-98, February 1996.
 - [20] J. E. Schutt-Aine and R. Mittra, "Scattering parameter transient analysis of transmission lines loaded with nonlinear terminations," *IEEE Trans. Micro-*
-

wave Theory Tech., vol. 36, pp. 529-536, 1988.

- [21] A. R. Djordjević and T. K. Sarkar, "Analysis of lossy transmission lines with arbitrary nonlinear terminal networks", *IEEE Trans. Microwave Theory Tech.*, vol. MTT-34, pp. 660-666, June 1986.
 - [22] D. Winklestein, M. B. Steer, and R. Pomeerleau, "Simulation of arbitrary transmission line networks with nonlinear terminations," *IEEE Transactions on Circuits Systems*, vol. 38, pp. 418-422, April 1991.
 - [23] R. Achar and M. Nakhla, "Efficient transient Simulation of embedded sub-networks characterized by S-parameters in the presence of nonlinear elements", *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-46, pp. 2356-2363, Dec. 1998.
 - [24] M. Elzinga, K. L. Virga, L. Zhao and J. L. Prince, "Pole-residue formulation for transient simulation of high-frequency interconnects using householder LS curve-fitting techniques," *IEEE Transactions on Advanced Packaging*, vol. 25, pp. 142-147, May 2000.
 - [25] M. Elzinga, K. L. Virga and J. L. Prince, "Improved Global Rational Approximation Macromodeling Algorithm for Networks Characterized by Frequency-Sampled Data," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, no. 9, Sept. 2000.
 - [26] M. Celik, "AZSPICE Circuit Simulator Using Macromodels Modified from Berkeley Spice3F4", University Arizona, Tucson, 1996.
 - [27] H. W. Brinkmann and E. A. Klotz, *Linear Algebra and Analytic Geometry*. Reading, MA: Addison-Wesley, 1971, pp. 445-446.
 - [28] W. T. Beyene and J. E. Schutt-Aine, "Efficient Transient Simulation of High-Speed Interconnects Characterized by Sampled Data," *IEEE Transactions on Components, Packaging, and Manufacturing Technology, Part B*, vol. 21, no. 1, pp. 105-114, Feb. 1998.
 - [29] W. T. Beyene and J. E. Schutt-Aine, "Interconnect Simulation Using Order Reduction and Scattering Parameters," *IEEE Electronic Components and Tech-*
-

-
- nology Conference*, pp. 627-631, 1998.
- [30] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "Robust Rational Function Approximation Algorithm for Model Generation," *Proceedings of 36th Design Automation Conference*, pp. 207-212, New Orleans, June 1999.
 - [31] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Series in Computational Mathematics, Prentice Hall, 1983
 - [32] Yousef Saad, *Iterative Methods for Sparse Linear Systems*, Pws Publishing Co., 1996.
 - [33] C. P. Coelho, J. R. Phillips, and L. M. Silveira, "Passive Constrained Rational Approximation Algorithm using Nevanlinna-Pick Interpolation," *Proceedings of 2002 Design, Automation and Test in Europe Conference and Exhibition*.
 - [34] L. M. Silveria, I. M. Elfadel, J. K. White, M. Chilukuri and K. S. Kundert, "An efficient approach to transmission line simulation using measured or tabulated S-parameter data," in *Proc. ACM/IEEE Design Automation conf.*, pp. 634-639, June 1994.
 - [35] K. J. Kerns and A. T. Yang, "Stable and efficient reduction of large, multiport RC networks by pole analysis via congruence transformations," in *Proc. ACM/IEEE DAC.*, pp. 280-285, June 1995.
 - [36] K. J. Kerns and A. T. Yang, "Preservation of passivity during RLC network reduction via split congruence transformations," *IEEE Transactions on Computer-Aided Design*, vol. 17, pp. 582-591, July 1998.
 - [37] A. Odabasioglu, M. Celik and L. T. Pillage, "PRIMA: Passive Reduced - Order Interconnect Macromodeling Algorithm", *Proceedings IEEE ICCAD*, pp. 58-65, Nov. 1997.
 - [38] I. M. Elfadel and D. D. Ling, "A block rational Arnoldi algorithm for multiport passive model-order reduction of multiport RLC networks," *Proc. of ICCAD-97*, pp. 66-71, Nov. 1997.
 - [39] Q. Yu, J. M. L. Wang and E. S. Kuh, "Multipoint multiport algorithm for passive
-

reduced-order model of interconnect networks," *Proc. of ISCAS-98*, vol. 6, pp. 74-77, Jun. 1998.

- [40] R. Achar and M. Nakhla, "Addressing passivity issues in CFH", *Proceedings 2nd IEEE International Workshop on Signal Propagation on Interconnects*, pp 33-34, May 1998, Hannover, Germany.
 - [41] R. Achar, P. Gunupudi, M. Nakhla and E. Chiprout, "Passive Interconnect Reduction Algorithm for Distributed/Measured Networks", *TCAS-II*, pp. 287-301, April 2000.
 - [42] D. Saraswat, R. Achar and M. Nakhla, "A Fast Algorithm and Practical Considerations for Passive Macromodeling of Measured/Simulated Data," *IEEE 11th Topical Meeting on Electrical Performance of Electronic Packaging*, Monterey, CA, pp. 297-300, October 2002 (best student paper award).
 - [43] D. Saraswat, R. Achar and M. Nakhla, "Passive Macromodels of Microwave Subnetworks Characterised by Measured/Simulated Data", accepted for publication in *IEEE International Microwave Symposium*, Philadelphia, June 03 (nominated for the best student paper award).
 - [44] D. Saraswat, R. Achar and M. Nakhla, "Passive Macromodeling of Subnetworks Characterised by Measured Data", accepted for publication in *IEEE International Symposium on Circuits and Systems*, Bangkok, May 03.
 - [45] D. Saraswat, R. Achar and M. Nakhla, "On Passivity Check and Compensation of Macromodels from Tabulated Data", accepted for publication in *IEEE Intl. Workshop on Signal Propagation on Interconnects*, Siena, Italy, May 03.
 - [46] D. Saraswat, R. Achar and M. Nakhla, "Passive Macromodeling Algorithm for S-parameter based Interconnect Networks", submitted to *IEEE International Conference on Computer Aided Design (ICCAD)*, 2003.
 - [47] D. Saraswat, R. Achar and M. Nakhla, "A Fast Algorithm and Practical Considerations for Passive Macromodeling of Measured/Simulated Data," submitted to *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2003.
-

-
- [48] D. Saraswat, R. Achar and M. Nakhla, "Macromodel Passivity Verification and Compensation for Microwave Subnetworks Characterized by Tabulated Data," submitted to *IEEE Transactions on Microwave Theory and Techniques*, 2003.
 - [49] B. Gustavsen and A. Semlyen, "Rational Approximation of Frequency Domain Responses by Vector Fitting", *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 1052-1061, July 1999.
 - [50] R. Lozano-Leal and S. M. Joshi, "Strictly Positive Real Transfer Functions Revisited", *IEEE Transactions on Automatic Control*, vol. 35, no. 11, pp. 1243-1245, November 1990.
 - [51] J. Morsey and A. C. Cangellaris, "PRIME: Passive Realization of Interconnect Models from Measured Data", *IEEE 10th Topical Meeting on Electrical Performance of Electronic Packaging*, Cambridge, Massachusetts, pp. 47 - 50, October 2001.
 - [52] T. Mangold and P. Russer, "Full-wave Modeling and Automatic Equivalent-circuit Generation of Millimeter-wave Planar and Multilayer Structures", *IEEE Trans. Microwave Theory Tech.*, vol. 47, no. 6, pp. 851-858, June 1999.
 - [53] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM Studies in Applied Mathematics, vol. 15, 1994.
 - [54] C. W. Ho, A. E. Ruehli and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. Circuits and Systems*, vol. CAS-22, pp. 504-509, June 1975.
 - [55] R. Griffith and M. Nakhla, "Mixed frequency/time domain analysis on nonlinear circuits," *IEEE Trans. Computer-Aided Design*, vol. 10, no. 8, pp. 1032-1043, Aug. 1992.
 - [56] R. Wang and O. Wing, "Transient analysis of dispersive VLSI interconnects terminated in nonlinear loads," *IEEE Trans. Computer-Aided Design*, vol. 11, pp. 1258-1277, Oct. 1992.
-

-
- [57] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 352-366, Apr. 1990.
 - [58] E. Chiprout and M. Nakhla, *Asymptotic Waveform Evaluation and Moment Matching for Interconnect Analysis*, Boston: Kluwer Academic Publishers, 1993.
 - [59] S. Kumashiro, R. A. Rohrer, A. J. Strojwas, "Asymptotic waveform evaluation for transient analysis of 3-D interconnect structures," *IEEE Trans. Computer-Aided Design*, vol. 12, no. 7, pp. 988-996, 1993.
 - [60] X. Huang, "Padé Approximation of linear(ized) circuit responses," Ph.D. dissertation, Carnegie Mellon Univ., Nov 1990.
 - [61] T. Tang and M. Nakhla, "Analysis of high-speed VLSI interconnect using the asymptotic waveform evaluation technique," *IEEE Trans. Computer-Aided Design*, vol 11, pp. 2107-2116 Mar. 1992.
 - [62] D. Xie and M. Nakhla, "Delay and crosstalk simulation of high speed VLSI interconnects with nonlinear terminations," *IEEE Trans. Computer-Aided Design*, vol. 12, no. 11, pp. 1798-1811, Nov. 1993.
 - [63] S. Y. Kim, N. Gopal, and L. T. Pillage, "Time-domain macromodels for VLSI interconnect analysis," *IEEE Trans. Computer-Aided Design*, pp. 1257-1270, Oct. 1994.
 - [64] R. Achar and M. Nakhla, *Minimum Realization of Reduced-Order Models of High-Speed Interconnect Macromodels*, To appear as a chapter in the book *Signal Propagation on Interconnects*, Boston: Kluwer Academic Publishers, 1998.
 - [65] R. Achar and M. Nakhla, "A novel technique for minimum-order macromodel synthesis of high-speed interconnect subnetworks", *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 70 - 73. June, 1998, Monterey, CA.
 - [66] R. Achar, M. Nakhla and E. Chiprout, "Block CFH: A model-reduction technique for distributed interconnect networks", *Proc. IEEE European Conference on Circuits Theory and Design (ECCTD)*, pp. 396-401, Sept. 1997, Budapest,
-

Hungary.

- [67] R. Achar and M. Nakhla, "Addressing passivity issues in CFH", *Proceedings 2nd IEEE International Workshop on Signal Propagation on Interconnects*, pp 33-34, May 1998, Hannover, Germany.
 - [68] R. Achar and M. Nakhla "A fast method for nonlinear simulation of transmission line networks", *Proc. IEEE International Symposium on Nonlinear Theory and Its Applications (NOLTA)*, pp. 837-840, Nov 1997, Hawaii, USA.
 - [69] E. Chiprout and M. Nakhla, "Fast nonlinear waveform estimation of large distributed networks," in *Proc. IEEE MTT-S Int. Microwave Symp.*, 1992, pp. 1341-1344.
 - [70] R. Achar, M. Nakhla and Q. J. Zhang, "Addressing high-frequency issues in VLSI interconnects with full-wave models and CFH", *Proc. IEEE/ACM Int. conference on Computer Aided Design*, pp. 53-57, Nov. 1995, San Jose, CA, USA
 - [71] R. Achar, M. Nakhla and Q. J. Zhang, "Moment-matching techniques for transient analysis of high-speed VLSI interconnects with full-wave models", *Proc. IEEE 25th European Microwave Conference, Sept. 1995*, pp. 820-823, Bologna, Italy.
 - [72] M. Nakhla, E. Chiprout, R. Achar and R. Khazaka, "Recent progress in simulation of high speed interconnects using moment matching techniques", *Proc. IEEE 4th Topical Meeting on Electrical Performance of Electronic Packaging*, Oct 1995 (Portland, Oregon, USA).
 - [73] M. Nakhla, R. Khazaka and R. Achar, "A universal macromodeling technique for high-speed VLSI interconnects", *1st IEEE International Workshop on Signal Propagation on Interconnects*, Hannover, Germany, May 1997.
 - [74] M. Nakhla, R. Achar and M. A. Kolbehdari, "Model-reduction techniques for EM and mixed EM/circuit simulation", *1997 IEEE URSI North American Radio Science Meeting*, July 1997, Montreal, PQ, pp. 151.
 - [75] R. Achar, M. A. Kolbehdari and M. Nakhla, "A unified approach for mixed EM/circuit simulation using model-reduction techniques", *IEEE International*
-

al Microwave Symposium Digest, June 1997, pp. 1017-1021, Denver, USA.

- [76] M. A. Kolbehdari, M. Nakhla, R. Achar and M. Srinivasan, "Solution of EM problems using reduced-order models by complex frequency hopping", *13th IEEE Annual Review of Progress in Applied Computational Electromagnetics*, Monterey, CA, March, 1997.
 - [77] S. D. Corey and A. T. Yang, "Interconnect characterization using time-domain reflectometry," *IEEE Trans. Microwave Theory Tech.*, vol. 43, pp. 2151-2156, Sep. 95.
 - [78] B. J. Cooke, J. L. prince and A. C. Cangellaris "S-parameter analysis of multi-conductor integrated circuit interconnect systems," *IEEE Trans. Computer-Aided Design*, vol. 11, No. 3, pp. 353-360, Mar. 1992.
 - [79] P. C. Cherry and M. F. Iskander, "FDTD analysis of high frequency electronic interconnection effects," *IEEE Trans. Microwave Theory Tech.*, vol. 43, no. 10, pp. 2445-2451, Oct. 1995.
 - [80] R. Sanaie, E. Chiprout, M. S. Nakhla, and Q. J. Zhang, "Integrating subnetworks characterized by measured data into moment-matching simulations," *Proc. Multi-Chip Module Conf.*, pp. 114-119, Mar. 1994.
 - [81] W. T. Beyene, and J. E. Schutt-Aine, "Efficient transient simulation of high-speed interconnects characterized by sampled data," *IEEE Transactions on CPMT*, Part B, Vol. 21, pp. 105 - 113, Feb. 1998.
 - [82] G. Zheng, Q. J. Zhang, M. Nakhla and R. Achar "An efficient approach for simulation of measured subnetworks with complex frequency hopping", *Proceedings IEEE/ACM In. Conf. Computer Aided Design*, pp. 23-26, Nov 1996, San Jose, CA.
 - [83] R. Achar, M. Nakhla and E. Ahmed "Nonlinear transient simulation of embedded subnetworks characterized by s-parameters using complex frequency hopping", *Proceedings 1998 IEEE MTT-S International Microwave Symposium*, pp. 1219 - 1223 June 1998, Baltimore, USA.
 - [84] W. B. Jones and A. Magnus, "Computation of poles of two-point Padé ap-
-

-
- proximants and their limits," *J. Computational App. Math.*, vol. 6, no. 2, pp. 105-119, 1980.
- [85] G. A. Baker Jr., *Essential of Padé Approximants*. New York: Academic, 1975.
- [86] J. H. McCabe, "A formal extension of the Padé table to include two point Padé quotients," *J. Inst. Math. Applic.*, vol. 15, pp. 363-372, 1975.
- [87] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, New York: Van Nostrand Reinhold, 1983.
- [88] L. Weinberg, *Network Analysis and Synthesis*, New York: McGraw-Hill Book Company Inc., 1962.
- [89] E. S. Kuh and R. Rohrer, *Theory of Active Linear Networks*, San Francisco: Holden-day Inc., 1967.
- [90] E. A. Guillemin, *Synthesis of Passive Networks*, New York: John Wiley and Sons Inc., 1957.
- [91] M. E. V. Valkenburg, *Introduction to Modern Network Synthesis*. New York: John Wiley and Sons Inc., 1960.
- [92] T. Kailath, *Linear Systems*. Toronto: Printce-Hall Inc., 1980.
- [93] E. G. Gilbert, "Controllability and observability in multivariable control systems", *SIAM J. Control*, pp. 128-151, 1963.
- [94] C. T. Chen, *Linear system theory and design*. New York: Holt, Rinehart and Winston, 1984
- [95] MATLAB, The MathWorks Inc., Version 6.0, Natick, MA.
- [96] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Boston: PWS Publishing, 2000.
- [97] Star-HSPICE Manual, Release 2001.2
-

APPENDIX ‘A’

Consider the convolution integral of (2.36) as shown below:

$$b(t) = \int_0^t s(\tau) a(t - \tau) d\tau \quad (\text{A.1})$$

By using the commutative property of convolution integral, (A.1) can be written as,

$$b(t) = \int_0^t a(\tau) s(t - \tau) d\tau \quad (\text{A.2})$$

At time point t_n , we can write (A.2) as,

$$b(t_n) = \int_0^{t_n} a(\tau) s(t_n - \tau) d\tau \quad (\text{A.3})$$

From (2.35), $s(t)$ can be written as

$$s(t) = L^{-1}(S(s)) = L^{-1}\left(c + \sum_{i=1}^{m'} \frac{r_i}{s + p_i}\right) = c\delta(t) + \sum_{i=1}^{m'} r_i e^{-p_i t} \quad (\text{A.4})$$

where L^{-1} is the inverse Laplace transform. Next, substituting (A.4) into (A.3) we obtain,

$$b(t_n) = \int_0^{t_n} a(\tau) \left(c\delta(t_n - \tau) + \sum_{i=1}^{m'} r_i e^{-p_i(t_n - \tau)} \right) d\tau \quad (\text{A.5})$$

or,

$$b(t_n) = \int_0^{t_n} a(\tau) c\delta(t_n - \tau) d\tau + \sum_{i=1}^{m'} \int_0^{t_n} a(\tau) r_i e^{-p_i(t_n - \tau)} d\tau \quad (\text{A.6})$$

Noticing that $\int_0^{t_n} a(\tau) c\delta(t_n - \tau) d\tau = ca(t_n)$, we can write (A.6) as,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} \int_0^{t_n} a(\tau) r_i e^{-p_i(t_n - \tau)} d\tau \quad (\text{A.7})$$

Next, defining $\delta t_n = t_n - t_{n-1}$ and splitting the integral in (A.7) into two parts (one from 0 to t_{n-1} and other from t_{n-1} to t_n) we obtain,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} \left\{ e^{-p_i \delta t_n} \int_0^{t_{n-1}} a(\tau) r_i e^{-p_i(t_{n-1} - \tau)} d\tau \right\} + \sum_{i=1}^{m'} \int_{t_{n-1}}^{t_n} a(\tau) r_i e^{-p_i(t_n - \tau)} d\tau \quad (\text{A.8})$$

Now, substituting $\int_0^{t_{n-1}} a(\tau) r_i e^{-p_i(t_{n-1} - \tau)} d\tau = \tilde{b}_i(t_{n-1})$ in (A.8) we get,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} e^{-p_i \delta t_n} \tilde{b}_i(t_{n-1}) + \sum_{i=1}^{m'} \int_{t_{n-1}}^{t_n} a(\tau) r_i e^{-p_i(t_n - \tau)} d\tau \quad (\text{A.9})$$

Next consider the integral term in (A.9). When $\delta t_n = t_n - t_{n-1}$ is very small $a(\tau)$ may be considered constant (i.e. $a(\tau) = a(t_{n-1})$) within the interval t_{n-1} to t_n . With this assumption (A.9) can be written as,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} e^{-p_i \delta t_n} \tilde{b}_i(t_{n-1}) + \sum_{i=1}^{m'} r_i a(t_{n-1}) \int_{t_{n-1}}^{t_n} e^{-p_i(t_n - \tau)} d\tau \quad (\text{A.10})$$

Now defining a new variable

$$m = t_n - \tau \quad (\text{A.11})$$

in (A.10), we get

$$d\tau = -dm \quad (\text{A.12})$$

Substituting (A.11) and (A.12) in (A.10), we get

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} e^{-p_i \delta t_n} \tilde{b}_i(t_{n-1}) + \sum_{i=1}^{m'} r_i a(t_{n-1}) \int_0^{\delta t_n} e^{-p_i m} dm \quad (\text{A.13})$$

or,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} e^{-p_i \delta t_n} \tilde{b}_i(t_n - \delta t_n) + \sum_{i=1}^{m'} \frac{r_i a(t_n - \delta t_n)}{p_i} (1 - e^{-p_i \delta t_n}) \quad (\text{A.14})$$

Thus,

$$b(t_n) = ca(t_n) + \sum_{i=1}^{m'} \tilde{b}_i(t_n) \quad (\text{A.15})$$

where,

$$\tilde{b}_i(t_n) = \frac{r_i}{p_i}(1 - e^{-p_i \delta t_n})a(t_n - \delta t_n) + e^{-p_i \delta t_n} \tilde{b}_i(t_n - \delta t_n), \quad (\text{A.16})$$

APPENDIX ‘B’

As described in the Sec. 4.3, constraint (C2) is introduced so that the real part of the approximated data of the diagonal elements of $Y(s)$ is either equal to or greater than the given data while solving the linear least square problem

$$\begin{bmatrix} Re(A) \\ Im(A) \end{bmatrix} X = \begin{bmatrix} Re(b) \\ Im(b) \end{bmatrix}.$$

For symmetric admittance matrices, ensuring the criteria for passivity in (4.8) can be a challenging task at the frequency points where the given admittance data matrix is marginally passive and not strictly diagonally-dominant. To understand this issue, the following section discusses the diagonal dominance in detail, and its implication on achieving the objective set by (4.8).

Diagonally-dominant matrix:

A matrix is said to be diagonally-dominant [96], if the diagonal element in each row is greater than or equal to the sum of the absolute values of the off-diagonal terms in the respective rows. i.e. for a matrix $A = [a_{ij}]$,

$$a_{ii} \geq \sum_{j \neq i} |a_{ij}| \tag{B.1}$$

As an example, the following matrix

$$\begin{bmatrix} 2 & 1 & 0.5 \\ 1 & 3 & 0 \\ 0.5 & 0 & 1 \end{bmatrix} \tag{B.2}$$

is diagonally-dominant, because each diagonal element satisfies the criteria (B.1). On the other hand the matrix

$$\begin{bmatrix} 1 & 1 & 0.5 \\ 1 & 3 & 0 \\ 0.5 & 0 & 1 \end{bmatrix} \quad (\text{B.3})$$

is not diagonally-dominant because the diagonal element of the first row violates the criteria (B.1). It can be proved by using Gershgorin's theorem [96], that a real diagonally-dominant matrix, with positive diagonal entries, is positive definite (i.e. all its eigenvalues are positive).

Theorem 1 (Gershgorin's theorem) :

Let $A = [a_{ij}]$ be a square matrix. All of the eigenvalues of A lie in the union of the discs centered at its diagonal entries a_{ii} , with radii equal to the sum of the absolute values of the off-diagonal entries in the respective row. i.e. if λ is an eigenvalue of A then:

$$|a_{ii} - \lambda| \leq \sum_{j \neq i} |a_{ij}| \quad (\text{B.4})$$

for any i . As an example, the Gershgorin's discs of matrix (B.2) are as depicted in the figure below:

Fig. B.1. Gershgorin's discs for the matrix of (B.2)

Since matrix (B.2) is 3 by 3, it has three discs as shown in Fig. B.1.(a), with the centres at 2, 3 and 1, and the corresponding radii as 1.5, 1 and 0.5 respectively. As per the Gershgorin's theorem, eigenvalues of the matrix (B.2) lie in the union of the three discs, shown by dotted curve, in Fig. B.1.(b). As verified by MATLAB, the eigenvalues (0.70, 1.64, 3.64) indicated by '*' in Fig. B.1.(b), lie inside the union of the corresponding Gershgorin's discs.

Theorem 2 : A real symmetric diagonally-dominant matrix, with positive diagonal entries, is positive-definite (i.e. all its eigenvalues are positive) [96]:

Proof :

Rearranging (B.4), we get,

$$|a_{ii}| - \sum_{j \neq i} |a_{ij}| \leq |\lambda| \quad (\text{B.5})$$

From (B.1) we know that for diagonally-dominant matrix with positive diagonal elements, $a_{ii} \geq \sum_{j \neq i} |a_{ij}|$. With this fact, it is clear from (B.5) that the eigenvalues of a diagonally-dominant matrix, with positive diagonal entries are greater than zero. This in turn means that a diagonally-dominant matrix with positive diagonal entries is positive-definite.

It is to be noted that the condition specified in Theorem 2 is sufficient, but not necessary for a matrix to be positive-definite. A simple example of a positive-definite matrix which is not diagonally-dominant is as follows:

$$\begin{bmatrix} 2 & 3 & 1 \\ 3 & 5 & 1.5 \\ 1 & 1.5 & 1 \end{bmatrix} \quad (\text{B.6})$$

In practice, the real part of circuit admittance matrices obtained at discrete frequency points may or may not be diagonally-dominant, however, still passive. When such a data is approximated, then any minor inaccuracy in the fitting, where the data is not diagonally-dominant, may result in loss of passivity. To illustrate the point, consider the following real and symmetric matrix (real part of admittance matrix), obtained at a certain frequency point :

$$\begin{bmatrix} 7 & 17 & 1 \\ 17 & 46 & 3 \\ 1 & 3 & 0.2122 \end{bmatrix} \quad (\text{B.7})$$

It is evident that the matrix is not diagonally-dominant and is marginally positive-definite as its eigenvalues are

$$\begin{bmatrix} 7.59e^{-5} \\ 6.51e^{-1} \\ 5.25e^1 \end{bmatrix} \quad (\text{B.8})$$

Next, suppose that this data is fitted and the approximated matrix be obtained as :

$$\begin{bmatrix} 6.87 & 16.95 & 0.99 \\ 16.95 & 46.1 & 2.98 \\ 0.99 & 2.98 & 0.21 \end{bmatrix} \quad (\text{B.9})$$

Note, that the matrix is fitted fairly accurately, however, it can be easily verified that the matrix (B.9) is not positive-definite as its eigenvalues are

$$\begin{bmatrix} -1.38e^{-4} \\ 5.83e^{-1} \\ 5.26e^1 \end{bmatrix} \quad (\text{B.10})$$

By inspecting (B.9) it can be easily verified that, if the diagonal elements are approximated with a constraint, that they are either equal to or greater than the given data in (B.7), the resulting approximation will be positive-definite. Constraint (C2) takes care of the situations outlined above, by ensuring that the real part of the diagonal entries of the approximated data is equal to or greater than the given data, so that the response satisfies the passivity criteria at the corresponding frequency point. This constraint has been found sufficient to obtain a passive macromodel for numerous examples tried during this thesis. In the unlikely event, that the passive macromodel could not be achieved even after using constraint (C2), the off-diagonal elements can be fitted subjected to the constraint that, their absolute values is less than or equal to the given data.