INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI®
ELECTROMAGNETIC SYSTEM FREQUENCY-DOMAIN REDUCED-ORDER MODELING AND TIME-DOMAIN SIMULATION

by

Tingdong Zhou

A Dissertation Submitted to the Faculty of the
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
In Partial Fulfillment of the Requirements
For the Degree of
DOCTOR OF PHILOSOPHY

In the Graduate College
THE UNIVERSITY OF ARIZONA

2002
THE UNIVERSITY OF ARIZONA • GRADUATE COLLEGE

As members of the Final Examination Committee, we certify that we have read the dissertation prepared by Tingdong Zhou entitled Electromagnetic System Frequency-Domain Reduced-Order Modeling and Time-Domain Simulation and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

John Prince III, Ph.D.  Date  6/21/02

Olgierd A. Palusinski, Ph.D.  Date  6/21/02

John F. O’Hanlon, Ph.D.  Date  6/21/02

Steven L. Dvorak, Ph.D.  Date  6/21/02

Kathleen Virga, Ph.D.  Date  6/21/02

Final approval and acceptance of this dissertation is contingent upon the candidate’s submission of the final copy of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

Dissertation Director John Prince III, Ph.D.  Date  6/21/02
STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Tingdong
ACKNOWLEDGEMENTS

I would like to thank my advisors, Dr. John Prince, Dr. Steven Dvorak, and Dr. Kathleen Virga, for their guidance and support in my work.

I would like also to thank Dr. Andrew Cangellaris and Dr. Li Zhao for their previous work and helpful discussions.

I am very grateful to Dr. Zhaoqing Chen and Dr. Wiren Becker for their mentoring when I worked as a co-op on transmission line time-domain modeling and simulation project in IBM.

I am indebted to the Semiconductor Research Corporation (SRC) who provided the funding for this project and made possible my graduate studies at the University of Arizona.
# TABLE OF CONTENTS

LIST OF FIGURES ................................................................. 7

LIST OF TABLES ................................................................. 10

ABSTRACT ............................................................................ 11

1. INTRODUCTION ............................................................. 12

2. THE FINITE-DIFFERENCE METHOD .................................. 20
   2.1 The Semi-Discrete Electromagnetic Model .................. 21
   2.2 Implementation of PMLs in Model Order Reduction .... 27
   2.3 Passivity of the Discrete Model ............................... 31

3. MODEL ORDER REDUCTION ALGORITHMS .................... 36
   3.1 Model Order Reduction of Linear Systems ................. 37
   3.2 Reduced-Order Modeling Algorithms ....................... 45
   3.3 Development of Passive Model Order Reduction ....... 56

4. REDUCED-ORDER MODELING CODE IMPLEMENTATION AND
   NUMERICAL EXAMPLES .................................................... 58
   4.1 Code Implementation ............................................... 58
   4.2 Examples of Reduced-Order Modeling ....................... 63

5. A TIME DOMAIN CONVOLUTION METHOD .................... 92
   5.1 Introduction to Time Domain Simulation .................... 92
   5.2 A Fast Simulation Method-Fastline ......................... 95

6. CLOSED-FORM TRIANGLE IMPULSE RESPONSES .......... 100
   6.1 Frequency Domain Formulation .............................. 101
      6.1.1 Single Lossy Line ............................................. 104
      6.1.2 Two-Line System ............................................. 104
   6.2 Analytic Inverse Fourier Transformation ................. 106
      6.2.1 Single Lossy Line ............................................. 107
      6.2.2 Two-Line System ............................................. 111
   6.3 Examples of Closed-Form TIR Application ............... 112

7. TIR CALCULATION FOR LOSSY TRANSMISSION LINE
   SIMULATION ..................................................................... 123
   7.1 Causal Models for the Line Parameters .................... 124
   7.2 TIR Calculations Using Inverse Fourier & Laplace Transforms ............................ 129
   7.3 Transmission Line Simulation Examples .................... 136
TABLE OF CONTENTS - continued

8. CONCLUSIONS AND FUTURE WORK ............................................ 146

APPENDIX A: NAMELIST COMMANDS IN M3.IN ............................ 150

APPENDIX B: NAMELIST COMMANDS IN ROMES.IN ...................... 153

REFERENCES ................................................................................. 158
LIST OF FIGURES

Fig. 2.1 Yee's finite-difference discrete lattice ......................................................... 22
Fig. 4.1 The flow diagram of ROMES .......................................................................... 59
Fig. 4.2 A parallel-plate waveguide with dielectric discontinuity ............................. 64
Fig. 4.3 Electrical field amplitudes in the excitation plane calculated using the PVL algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 64
Fig. 4.4 Electrical field phases in the excitation plane calculated using the PVL algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 65
Fig. 4.5 Electrical field amplitudes in the excitation plane calculated using the PVL algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 65
Fig. 4.6 Electrical field phases in the excitation plane calculated using the PVL algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 66
Fig. 4.7 Electrical field amplitudes in the excitation plane calculated using the Krylov algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 69
Fig. 4.8 Electrical field phases in the excitation plane calculated using the Krylov algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 70
Fig. 4.9 Electrical field amplitudes in the excitation plane calculated using the Krylov algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 70
Fig. 4.10 Electrical field phases in the excitation plane calculated using the Krylov algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 71
Fig. 4.11 The amplitudes of E field in the excitation plane calculated using the rational Krylov algorithm with expansion at 1MHz, 5GHz, 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 71
Fig. 4.12 The phases of E field in the excitation plane calculated using the rational Krylov algorithm with expansion at 1MHz, 5GHz, 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity ............................................. 72
Fig. 4.13 A parallel-plate waveguide with a dielectric discontinuity .......................... 73
Fig. 4.14 Scattering parameters for the parallel-plate waveguide with discontinuity ...... 74
Fig. 4.15 Scattering parameters for the parallel-plate waveguide with discontinuity ...... 75
Fig. 4.16 Dielectric post inside WR90 waveguide (εr=8.2, d=6mm, c=12mm) .......... 78
Fig. 4.17 Amplitude of the electric field on the excitation plane calculated using the PVL algorithms with expansions at a finite value and infinity ............................................. 81
LIST OF FIGURES - continued

Fig. 4.18 Phase of the electric field on the excitation plane calculated using the PVL algorithms with expansions at a finite value and infinity ................. 82
Fig. 4.19 Return loss from a dielectric post inside WR90 waveguide ..................... 83
Fig. 4.20 Visualization of the biorthogonality of matrices V and W ......................... 83
Fig. 4.21 Amplitude of the electric field on the excitation plane calculated using the PVL algorithm with expansion at a single frequency point, multiple points and infinity ............................................. 85
Fig. 4.22 A probe-fed patch antenna (a=41.275mm, h=3.175mm, s=14.224mm, \( \varepsilon_r=2.5 \)) .................................................. 85
Fig. 4.23 Amplitude of the input voltage for the probe-fed patch antenna .................. 88
Fig. 4.24 Phase of the input voltage for the probe-fed patch antenna ....................... 88
Fig. 4.25 Amplitude of the input current for the probe-fed patch antenna .................. 89
Fig. 4.26 Phase of the input current for the probe-fed patch antenna ....................... 90
Fig. 4.27 Magnitude of the reflection coefficient of the patch antenna in Fig. 4.22 ........ 91
Fig. 5.1 Input triangle impulse signal ............................................. 96
Fig. 5.2 A waveform composed by triangle basis waveforms .................................... 96
Fig. 5.3 N-coupled transmission line section ............................................. 97
Fig. 5.4 Equivalent circuit of N-coupled transmission line section ............................ 97
Fig. 5.5 Circuit configuration for TIR calculations. (\( \Delta t=10 \) ps) .............................. 98
Fig. 6.1 The schematic of a system containing N coupled interconnects .................... 102
Fig. 6.2 Input triangle impulse signal ............................................. 107
Fig. 6.3 The cross section of an one conductor microstrip transmission line ................ 112
Fig. 6.4 Far-end TIR voltages of 5cm single transmission line calculated with frequency dependent R, L and R, L values at 2GHz ................................. 114
Fig. 6.5 Transmission line circuit schematic ............................................. 115
Fig. 6.6 The near-end voltages simulated by Fastline on the transmission line ........... 116
Fig. 6.7 The far-end voltages simulated by Fastline on the transmission line ............ 116
Fig. 6.8 Transmission line circuit schematic ............................................. 117
Fig. 6.9 The near-end voltage simulated by Fastline on the transmission line ............ 118
Fig. 6.10 The far-end voltage simulated by Fastline on transmission line ................. 119
Fig. 6.11 Coupled MCM transmission line circuit schematic .................................. 120
Fig. 6.12 Near-end Voltages on the coupled MCM transmission lines ....................... 121
Fig. 6.13 Far-end Voltages on the coupled MCM transmission lines ........................ 121
Fig. 7.1 The cross-section of a single line ............................................. 133
Fig. 7.2 Transmission line circuit Schematic ............................................. 133
Fig. 7.3 Far-end voltage of TIR for 20 cm line ............................................. 134
Fig. 7.4 Far-end voltage on the transmission line shown in Fig. 7.2 ......................... 137
Fig. 7.5 Partly magnified far-end voltage shown in Fig. 7.4 ................................ 137
Fig. 7.6 Partly magnified far-end voltage shown in Fig. 7.4 ................................ 138
Fig. 7.7 Near-end voltage on the transmission line shown in Fig. 7.2 ....................... 138
Fig. 7.8 Diagram of a node-to-node bus ............................................. 139
LIST OF FIGURES - continued

Fig. 7.9 Schematic of the node-to-node bus ............................................................... 139
Fig. 7.10 Near-end and far-end voltage waveform on the node-to-node bus
simulated by Fastline using the TIRs calculated with the IFFT .......................... 143
Fig. 7.11 Far-end voltage waveform on the node-to-node bus simulated by Fastline
using the TIRs calculated with the IFFT for cases A, B, and C in Table 7.5 .......... 144
Fig. 7.12 Far-end voltage waveform on the node-to-node bus simulated by Fastline
using TIRs calculated with the IFFT and AILT algorithms for case B in
Table 7.5 .................................................................................................................. 144
**LIST OF TABLES**

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>Frequency dependent R and L for microstrip line shown in Fig. 6.3 calculated using UAPDSE.</td>
</tr>
<tr>
<td>6.2</td>
<td>Frequency dependent R and L for microstrip line shown in Fig. 6.3 calculated using UAPDSE.</td>
</tr>
<tr>
<td>6.3</td>
<td>Frequency dependent line parameters for the MCM coupled transmission lines.</td>
</tr>
<tr>
<td>7.1</td>
<td>Experimental data of dielectric constant $\varepsilon_r$ and loss tangent for FR-4.</td>
</tr>
<tr>
<td>7.2</td>
<td>Frequency dependent R and L for transmission line shown in Fig. 7.1 calculated using UAPDSE. (C=1.0713 pF/cm).</td>
</tr>
<tr>
<td>7.3</td>
<td>Line parameters for GC_top_via and GC_bottom_via.</td>
</tr>
<tr>
<td>7.4</td>
<td>Line parameters for GC_single_trace. (C=1.4394 pF/cm, G=0)</td>
</tr>
<tr>
<td>7.5</td>
<td>Line parameters for PCB_line. (Case A: C=1.6528 pF/cm, $G(\omega) = \omega C \tan \delta$, $\tan \delta = 0.02$)</td>
</tr>
</tbody>
</table>
ABSTRACT

Model order reduction methodologies are presented for semi-discrete electromagnetic systems obtained from the spatial discretization of the hyperbolic system of Maxwell's equations. Different reduced-order modeling algorithms, i.e., Padé via Lanczos (PVL), multiple point PVL, Krylov, rational Krylov, PVL with expansion at infinity, are presented and applied for model order reduction and the properties of these algorithms are discussed. The implementation of the model order reduction methodologies to a full-wave frequency domain electromagnetic system simulator (ROMES) is discussed in detail. Scattering parameters are calculated for several electromagnetic systems with discontinuities.

A time domain simulation framework is also introduced for transmission line embedded systems described by the Telegrapher's equations. The time domain convolution approach is selected to perform the transmission line embedded circuit simulations. Derivations for Closed-form triangle impulse responses (TIR) are discussed and numerical examples are presented. The developed triangle impulse responses are used to perform time-domain circuit simulations. The effects of frequency-dependent lossy transmission lines on signal integrity and causality issues associated with the transmission line parameters ($R$, $L$, $C$, and $G$) in Telegrapher's equation are discussed. The presented research provides an accurate and efficient way to characterize electromagnetic systems for high-speed circuit applications in the frequency domain and methods to simulate these circuits in the time domain.
1. INTRODUCTION

The requirements for fast switching speeds, low voltage, low power, highly integrated, and mixed analog-digital circuits, have brought the need for electromagnetic computer-aided design (EM-CAD) capability to integrated circuit design, especially for packaging and interconnect design. For clock frequencies in the GHz regime and rise times of a few tens of picoseconds, transmission line effects need be taken into account for proper timing analysis. Accurate prediction of displacement currents due to unbalanced interconnects as well as quantification of direct radiation from drivers are necessary for module electromagnetic compatibility assessment. The number of inputs/outputs (I/Os) per chip is climbing to several hundreds. The prediction of the disturbance of ground and power plane reference voltages and their impact on false switching is becoming a formidable task as several tens or even hundreds of the I/Os switch simultaneously at the aforementioned speeds. Accurate noise prediction and containment is becoming a major issue for the low-voltage designs that the electronics industry is trying to resolve at a consumer market-driven low cost. Such low-voltage designs (< 1.5 V supplies) require a very tight noise budget, which needs to be effected without over-hardening of the design in order to meet the low-cost demand. Furthermore, market competitiveness implies low cost and very short design cycles. Consequently, rapid prototyping is essential, and can be effected only by means of a sophisticated electronic CAD environment equipped with rapid full wave electromagnetic modeling capability.
Despite significant advances in electromagnetic modeling and simulation methodologies and computer tool sophistication, EM-CAD tools do not exhibit yet the efficiency needed for module or system design and subsequent design optimization. A brief assessment of up-to-date, commercially available EM-CAD tools reveals that their capabilities are currently limited to individual component design and circuits of rather low complexity. However, the compact integration that characterizes the new generation of electronic products requires a fairly rigorous modeling of the interactions between several components for proper design.

Even for very simple systems, the distributed linear and nonlinear electromagnetic analyses required for the design and optimization of up-to-date and future high-performance systems, which operate at high frequencies and fast rise times, are hindered by the large number of degrees of freedom involved in the electromagnetic models. Thus, the direct use of time-domain nonlinear differential-equation integration, as implemented in SPICE-like simulators enhanced with transmission-line modeling or even combined with rigorous FDTD-based Maxwell's equations solvers [1, 2], is prohibitive for circuits of the complexity encountered in realistic integrated electronic systems. Even standard SPICE-like nonlinear circuit simulators for lumped nonlinear circuit analysis can not handle the complexity of the circuits resulting from the modeling of interconnects and power distribution networks using lumped inductors, capacitors, and resistors.

The circuit simulation community has responding to this simulation challenge with the development of model order reduction techniques that allow the replacement of large linear portions of the circuit with a substantially smaller model that approximates
sufficiently its external behavior [3]-[5]. This way, the size of the problem left for the nonlinear simulator to solve becomes substantially smaller, and thus problem sizes out of the reach of conventional simulators can now be tackled.

More recently, model order reduction has been applied to distributional electromagnetic systems [6]-[9]. More specifically, Padé approximations to the responses of electromagnetic systems are used to perform the so-called fast frequency sweep, where the system response over a broad range of frequencies is obtained at approximately the cost of factoring the approximating matrix at only a few frequency points. For example, it was demonstrated in [9] that the matrix resulting from the finite difference approximation of Maxwell's equations, based on Yee's rectangular discrete lattice [10], needs to be factored only at a single frequency when the Padé via Lanczos (PVL) algorithm was used for the generation of the Padé approximation.

In addition to rapid broadband calculation of the frequency response of an individual electromagnetic component or a specific functional block, the development of macromodels of electromagnetic multi-ports is essential for design-driven simulation of large electromagnetic systems. The final goal is to achieve time domain signal integrity verification using the developed macromodels. It is important to keep in mind the fact that the resulting electromagnetic macromodel is required to be passive for stable time domain simulation. Different approaches have been presented in [11]-[15] to carry out the system characterization for time domain simulation from calculated or experimental frequency domain responses, e.g., scattering parameters. For example, the Telegrapher's equation transmission line parameters (R, L, C, and G) can be extracted from
scattering parameters [11]. These transmission line parameters are usually used to perform the time domain simulations.

The fundamental difficulty encountered in integrating transmission line simulations into a transient circuit simulator arises because circuits containing nonlinear devices or time-dependent characteristics must be characterized in the time domain, while transmission lines with loss, dispersion, or discontinuities are best characterized in the frequency domain. To cope with this difficulty, five types of approaches have been proposed in previous works. One uses a network of lumped elements and segments of ideal transmission lines to approximate the frequency response [16] or time domain response [17] of each lossy transmission line or each lossy multiconductor line. The approximated circuit models are suitable for existing general-purpose circuit simulators such as PSPICE.

The second type of approach adopts the convolution technique. For each integration, the outputs of each linear, lossy, multiconductor line is represented as a convolution of the input waveform with the impulse responses (Green's functions) of the multiconductor lines. The multiconductor lines are treated as a linear N-port system. One difficulty of this type of approach lies in how to determine the impulse responses of an arbitrary multiconductor line system. People have used the inverse fast Fourier transformation technique [18], the numerical inverse Laplace transformation technique [19], inverse Fourier transformation of frequency domain scattering parameters [20, 21], and an explicit analytic approach [22] to determine the impulse responses. In addition, a semi-analytical recursive convolution procedure was developed in [23] to characterize the
current transfer matrix and admittance matrix. To avoid the time-consuming convolution integrations, the state-based approach [24] and the waveform relaxation based approach [25] have been proposed. However, the efficiency of the approach will degrade for simulations with many sharp edges in the waveform or with a very high frequency signal propagating along the transmission line.

Finally, reduced-order modeling techniques, including Asymptotic Waveform Evaluation (AWE) type approaches [3, 26], Pade approximation [27] and passive reduced-order interconnect macromodeling algorithm (PRIMA) [28, 29] etc. have been used to solve transmission line problems. These types of approaches are usually very efficient. However, they sometimes suffer from numerical difficulties, such as ill-conditioned matrix multiplications, and recursive iteration breakdown.

A fast simulation method was proposed for circuits having single and coupled transmission lines [30]. This method, which belongs to the second type as summarized above, is based on a triangle impulse response (TIR) database of lossy transmission lines that are matched at both the near and far ends. Triangle waveforms are used as time-domain basis functions to represent the input waveforms. The time-domain response can then be calculated by superposition of triangle impulse responses. Other simulation tools are used to calculate the lossy transmission line triangle impulse responses numerically. The numerical results of the triangle impulse responses are then used to carry out the transient circuit simulations.

In the following chapters, model order reduction methodologies and a time domain simulation framework will be presented for semi-discrete electromagnetic
systems obtained from the spatial discretization of the hyperbolic system of Maxwell's equations and transmission line embedded system described by the Telegrapher's equations. It will be shown how the properties of the original semi-discrete system impact the passivity of the developed reduced-order model, and specific criteria for developing passive reduced-order models are established. Different reduced-order modeling algorithms, i.e., Krylov subspace projection algorithms [31, 32] will be presented and applied for model order reduction. The properties of those algorithms will be discussed. The implementation of the model order reduction methodologies to a full-wave frequency domain electromagnetic system simulator (ROMES) will be detailed. Scattering parameters will be calculated for several electromagnetic systems with discontinuities. For the time domain simulation framework, the time domain convolution approach will be selected to perform the transmission line embedded circuit simulations. Our work will focus on closed-form triangle impulse responses (Green's functions) derivation [33] and numerical calculation of the TIR [34]. The developed triangle impulse responses will be used to perform several example circuit simulations. The effect of frequency dependent lossy transmission lines to signal integrity issues and causality of Telegrapher's equation transmission line parameters (R, L, C, and G) will be discussed.

The background and formulation of reduced-order modeling of electromagnetic field interactions are discussed in Chapter 2. This discussion details the development of the semi-discrete electromagnetic model starting with Maxwell's equations and ending with a state-space system of first-order differential equations. Perfect electric conductor
PEC), perfect magnetic conductor (PMC), and perfectly matched layers (PML) are used to truncate the electromagnetic field interactions. Implementation of PML in model order reduction and passivity of the discrete model will be discussed.

Chapter 3 gives an overview of model order reduction of linear system using Krylov subspace projection method. Subspace projection approaches, including PVL [4, 5], Krylov [31, 35, 36] and rational Krylov [31, 35, 36], have been compared to each other in efficiency and accuracy for reduced-order modeling of high order linear systems which are derived in Chapter 2. For PVL, both expansion at finite frequency values [31] and expansion at infinity [32] will be presented. The passivity of the resulting reduced-order models and alternative approach that leads directly to a passive, reduced-order model will be discussed.

The program implementation of the semi-discrete electromagnetic model and reduced-order modeling are described in Chapter 4. Input data files and numerical examples will also be included in this chapter. The accuracy and efficiency comparisons of different reduced-order modeling algorithms will also be presented.

A transmission line embedded linear circuit simulator called FASTLINE [30] will be introduced in Chapter 5. The simulator uses the triangle impulse response database to perform circuit simulation based on a time domain convolution method. We will define the triangle impulse responses, outline the method, and discuss the properties of this method in this chapter.

The Modal analysis approach [37, 38] will be introduced in Chapter 6 to first develop the frequency domain formulas for the triangle impulse responses for single and
coupled frequency independent lossy transmission lines. We will start from Telegrapher’s equations and end with a set of compact formulas for the TIR in frequency domain. Closed-form expressions for the TIR in the time domain are given in the form of incomplete Lipschitz-Hankel integrals (ILHIs) of the first kind [33, 39]. The derived closed-form TIR will also be used with FASTLINE to perform transmission line circuit simulations.

Numerical calculation of the TIR will be detailed in Chapter 7 where frequency dependent transmission lines are introduced. An accelerated inverse Laplace transform (AILT) algorithm (Luisa D’amore) [40, 41] is used to calculate the TIR for frequency-dependent line parameters. A classic inverse fast Fourier transformation (IFFT) algorithm is also used to calculate the TIR. The calculated TIR will be further used with FASTLINE to perform frequency dependent lossy transmission line circuit simulations [34]. The causality requirements for the transmission line parameters $R$, $L$, $C$, and $G$ will also be discussed in this chapter.

Lastly, conclusions and future work are discussed. The contribution of this work is assessed and unresolved issues are examined.
2. THE FINITE-DIFFERENCE METHOD

Within the discipline of computational electromagnetics, there exist two distinct avenues through which solutions are developed. These are integral equation and differential equation based formulations. Integral equation techniques can be formulated using electric and magnetic vector potential functions, which are defined using the assumption of radiation into a locally homogeneous space or layered regions. This is a great advantage for problems whose boundaries are enclosed by perfect electric or magnetic conductors. For example, if it is desired to calculate the response of a two conductor transmission line system in homogeneous space, then the unknowns are assigned only over the surfaces of the transmission lines. On the other hand, if it is desired to calculate the response of a two conductor transmission lines in an enclosed box with perfectly conducting walls, the unknowns are assigned to not only the transmission lines, but also to the enclosure. For those geometries that include dielectrics, additional unknowns are used to describe the resulting polarization currents. This requirement can also be fulfilled by finding the Green’s function for inhomogeneous, specifically, layered media [42]. In this case additional unknowns are not needed any more. Methods for finding a numerical solution for integral equation based techniques include the Method of Moments (MoM) [43] and Partial Element Equivalent Circuit method (PEEC) [44].

On the other hand, differential equation methods are formulated through the numerical approximation of the differential form of Maxwell’s equations. This results in a sparse system of equations that are constructed in a very straightforward manner. In fact, the ease at which solutions are constructed has been a contributing factor for the
popularity of this method. An additional factor enabling the use of this technique is the development of much more powerful computers. Examples of differential equation based techniques include the Finite-Difference Time-Domain method (FDTD) [10] and the Transmission Line Matrix method (TLM) [44]. With these methods, the solution is developed by discretizing those regions of space where electromagnetic phenomena is of interest. In this dissertation, the Finite-Differential Frequency-Domain (FDFD) technique will be employed.

2.1 The Semi-Discrete Electromagnetic Model

While a variety of approaches exist for the spatial discretization of Maxwell’s equations, the semi-discrete approximation effected through the use of Yee’s lattice [10] as shown in Fig. 2.1 is chosen due to the considerations of the passivity of the semi-discrete model and the requirement of the first-order state-space equations.
A uniform, rectangular lattice is assumed, defined by equally spaced nodes along the three axes of a Cartesian coordinate system: \( I \) along \( x \), \( J \) along \( y \), and \( K \) along \( z \). The total number of nodes in the grid is \( N = I \times J \times K \). With the definitions \( U = 1 \), \( V = I \), \( W = I \times J \), the \( n \)th electric node, corresponding to node \((i,j,k)\) in the grid, is given by

\[
  n = 1 + (i-1)U + (j-1)V + (k-1)W
\]

(2.1)

where \( i = 1,2,\ldots,I \), \( j = 1,2,\ldots,J \), \( k = 1,2,\ldots,K \), and \( n = 1,2,\ldots,N \).

It is assumed that the media are linear, passive and time-independent. Thus, Maxwell's curl equations in the Laplace domain have the form

\[
  \nabla \times \mathbf{H} = s\varepsilon \mathbf{E} + \sigma \mathbf{E} + \mathbf{J},
\]

\[
  \nabla \times \mathbf{E} = -s\mu \mathbf{H}.
\]

(2.2)
where the electric permittivity, \( \varepsilon \), electric conductivity \( \sigma \), and magnetic permeability, \( \mu \), are position dependent. The functional dependencies of the electric and magnetic field vectors \( \mathbf{E} \) and \( \mathbf{H} \), as well as the imposed source current density \( \mathbf{J}_s \), on the spatial variables and the Laplace variable \( s \) are suppressed for simplicity.

In order to cast the semi-discrete form of Maxwell’s equations in a matrix form, we begin with the definition of the following two vectors of discrete unknowns

\[
\mathbf{E} = [E_x, E_y, E_z]^T
\]

\[
\mathbf{H} = [H_x, H_y, H_z]^T,
\]

where \( E_x \) is a vector of length \( N \), containing the \( N \) \( E_x \) values on the grid. Similar definitions hold for the remaining five vectors. Using the definitions in (2.3) and writing the curl operator in its matrix form,

\[
\nabla \times = \begin{bmatrix}
0 & -\partial/\partial z & \partial/\partial y \\
\partial/\partial z & 0 & -\partial/\partial x \\
-\partial/\partial y & \partial/\partial x & 0
\end{bmatrix},
\]

it is straightforward to show that the semi-discrete form of (2.2) can be written as

\[
\begin{bmatrix}
0 & \mathbf{A}_w^T & -\mathbf{A}_v^T \\
-\mathbf{A}_w^T & 0 & \mathbf{A}_u^T \\
\mathbf{A}_v^T & -\mathbf{A}_u^T & 0
\end{bmatrix} \cdot \mathbf{E} = -s\mathbf{D}_h \cdot \mathbf{H},
\]

\[
\begin{bmatrix}
0 & -\mathbf{A}_w & \mathbf{A}_v \\
\mathbf{A}_w & 0 & -\mathbf{A}_u \\
-\mathbf{A}_v & \mathbf{A}_u & 0
\end{bmatrix} \cdot \mathbf{H} = s\mathbf{D}_e \cdot \mathbf{E} + \mathbf{D}_\sigma \cdot \mathbf{E} + \mathbf{J}_s.
\]

In the above equations, the matrices \( \mathbf{A}_w \), \( \mathbf{A}_v \) and \( \mathbf{A}_w \) are sparse with only two bands having nonzero elements. One band is along the diagonal with all values equal to 1, and
the second band at a distance of $U, V, W$ respectively, to the left of the diagonal with all values equal to $-1$. The matrices $D_e, D_h$ and $D_\sigma$ are diagonal matrices with elements which depend on the electromagnetic properties of the media and the grid size.

The system of (2.5) may be cast in a compact form by defining the vector of state variables $X = [H, E]^T$, and the $3N \times 3N$ matrix $P$,

\[
P = \begin{bmatrix}
0 & A_w & -A_v \\
-A_w & 0 & A_u \\
A_v & -A_u & 0
\end{bmatrix}
\] (2.6)

In addition, the source notation is slightly modified to include the imposed currents used for the excitation of the ports. For this purpose, it is assumed that the electromagnetic system under consideration has $p$ ports. Each of them coincides with one electric field node. A constant matrix $b$ of dimension $6N \times p$ is introduced, with nonzero elements only in its bottom $3N$ rows associated with the electric field nodes in the state vector $X$. The specific values of these elements will depend, in general, on the source distribution and numerical grid characteristics. Using $U(s)$ to denote the Laplace transforms of the current source waveforms at the $p$ ports, the discrete source term may be cast in the form $bU(s)$. With these definitions, the resulting compact form of (2.5) is

\[
\begin{bmatrix}
0 & P^T \\
-P & 0
\end{bmatrix} X(s) = -s \begin{bmatrix}
D_h & 0 \\
0 & D_e
\end{bmatrix} X(s) - \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix} X(s) + bU(s),
\] (2.7)

or in a yet more compact form

\[
(G + sC)X(s) = bU(s),
\] (2.8)

where
Because of the assumed passivity of the media, the matrices $D_e$ and $D_h$ are symmetric, positive definite, and $D_\sigma$ is symmetric, non-positive definite. Consequently, $C$ is also symmetric, positive definite.

Next we introduce a frequency shift, $s = s_0 + \sigma$. Through a straightforward matrix manipulation, (2.8) is recast in the form of

$$(I - \sigma A)X(s) = rU(s),$$

where

$$A = -(G + s_0 C)^{-1}C, \quad r = (G + s_0 C)^{-1}b.$$  (2.11)

Defining a desired output vector as $Y(s) = z^TX(s)$, where $z^T$ is a selector matrix, we have

$$Y(s) = z^T(I - \sigma A)^{-1}rU(s).$$  (2.12)

For the case of the multi-port, the number of outputs is the same as the number of inputs. More specifically, with the selection of the current density as the excitation at each port, the electric field vector (or, equivalently, the voltage at the port, if the definition of the voltage makes sense) becomes the output quantity. For such cases, the selector matrix $z^T$ is a $p \times 6N$ matrix, with constant, non-zero entries at the right half of each of its rows, corresponding to the electric field unknowns in the state vector $X$. In particular, except for a scaling constant factor associated with the proper definition of the observed output quantity at the port, the selector matrix $z^T$ is simply the transpose of the

$$G = \begin{bmatrix} 0 & P^T \\ -P & D_e \end{bmatrix}, \quad C = \begin{bmatrix} D_h & 0 \\ 0 & D_\sigma \end{bmatrix}.\quad (2.9)$$
source matrix \( b \) defined earlier. Thus, the characterization of the multi-port electromagnetic system is effected in terms of a transfer-function vector, \( H(s) \), which, in view of (2.12), is given by

\[
H(s) = z^T (I - \sigma A)^{-1} r.
\]  

(2.13)

From (2.12) it is clear that the eigenvalues of \( A \) correspond to the poles of the transfer function elements of \( H(s) \). The objective of model order reduction techniques is to generate rational function approximations, e.g., Padé approximations, to the elements of \( H(s) \) of order much smaller than \( 6N \). At this point, it is appropriate to consider this approximation objective carefully, in view of the fact that what is of primary interest in the context of this discussion is the model order reduction of distributed electromagnetic systems. Continuous electromagnetic systems are, of course, of infinite order; hence, the number of poles required for the exact representation of their delay properties is infinite. However, when dealing with a numerical approximation of Maxwell’s equations, what is of interest is the reduced-order modeling of the resulting discrete electromagnetic system. These discrete systems have finite order, and their transfer functions have finite bandwidth that is controlled by the grid size and the order of approximation of the spatial derivatives. Consequently, the pole-residue representation of the transfer function of the approximated electromagnetic system will be of finite order.

For electromagnetic problems, the electric and magnetic fields at the excitation and output ports are the parameters that we want to calculate. The characteristic parameters, such as scattering parameters, can then be found using the calculated \( \mathbf{E} \) and \( \mathbf{H} \) values.
In essence, Maxwell's curl equations have been converted to a first-order differential state-space system after the above discretization step. The system in the Laplace domain is written as

\begin{align}
&scX(s) = -GX(s) + bU(s), \\
&Y(s) = z^T X(s).
\end{align} \tag{2.14}

Three kinds of boundary condition have been used to truncate the unbounded electromagnetic system, perfect electric conductor (PEC), perfect magnetic conductor (PMC), and perfectly matched layers (PML). It has been shown in [9, 46] that Maxwell's equations can also be converted to a state-space system in the same format as (2.14) in PML. For completeness, implementation of PML in model order reduction is also included in this dissertation in the following section.

2.2 Implementation of PMLs in Model Order Reduction

The successful extension of PVL and other Krylov space-related model order reduction algorithms to rapid electromagnetic analysis of radiation and scattering problems requires the implementation of numerical grid truncation schemes that are compatible with the PVL mathematical model and are frequency independent. It has been shown that the grid truncation scheme based on either Berenger's perfectly matched layers (PML) [47] or an alternative implementation using the generalized theory for PML (GT-PML) [46] is the most suitable for this purpose. Both of implementations have been validated in conjunction with the PVL model order reduction process.
With the assumption that all electromagnetic interactions occur in linear, anisotropic, time-independent media, the governing system of Maxwell's curl source-free equations in PML has the general form

\[
\nabla \times \mathbf{H} = s\varepsilon[\Lambda] \cdot \mathbf{E} \\
\n\nabla \times \mathbf{E} = -s\mu[\Lambda] \cdot \mathbf{H},
\]

where the elements of the diagonal matrix \([\Lambda]=\text{diag}\{\lambda_1, \lambda_2, \lambda_3\}\) are complex dimensionless constants. In the above system all material parameters are assumed to be position-dependent. For unbounded electromagnetic problems numerical grid truncation conditions that can be cast in linear form are the most convenient for PVL model order reduction purposes. Since Berenger's PML grid truncation is linear in \(s\), it is implemented directly from the Maxwellian system by introducing an appropriate splitting of the fields. For example, Berenger's equations in the PML for the split electric field and magnetic field components are

\[
\begin{align*}
-\mu s H_{xy} - \sigma_x \gamma y H_{xy} &= \frac{\partial}{\partial y} (E_{yx} + E_{yx}) \\
-\mu s H_{xz} - \sigma_x \gamma z H_{xz} &= -\frac{\partial}{\partial z} (E_{zx} + E_{zx}) \\
\varepsilon s E_{xy} + \sigma_x \gamma y E_{xy} &= \frac{\partial}{\partial y} (H_{yx} + H_{yx}) \\
\varepsilon s E_{xz} + \sigma_x \gamma z E_{xz} &= -\frac{\partial}{\partial z} (H_{zx} + H_{zx})
\end{align*}
\]

(2.16)

respectively, where the state variable vector is taken as

\[
\mathbf{X} = \begin{bmatrix} E_{xy}, E_{xz}, E_{yx}, E_{zx}, E_{yy}, E_{yz}, H_{yx}, H_{zx}, H_{yw}, H_{zy}, H_{xy}, H_{xz}, H_{yx}, H_{zx}, H_{yw}, H_{zy} \end{bmatrix}^T
\]

(2.17)

inside the PML regions. The remaining equations can be obtained simply by permutation of the component indices \(x, y, z\). The semi-discrete form of Berenger's equations is
similar in form to (2.8), except for the fact that the dimensionality of the matrices has increased by a factor of two. Changes occur only in the structure of sub-matrix $P$ of matrix $G$ in (2.8) due to the splitting of the fields, while $D_\sigma$, $C$ in (2.9) and (2.8) remains diagonal. Thus, the semi-discrete form of Berenger's PML equations is compatible with PVL.

An alternative to Berenger's PML is the use of uniaxial perfectly matched absorbers for grid truncation. As discussed in detail in [46], the key attributes of the generalized theory of perfectly matched layers (GT-PML) are: (1) the fact that it does not require field splitting but uses the physical components of the fields; (2) the number of extra unknowns used for its implementation is smaller than that of Berenger's PML; (3) it is well-posed and thus not prone to numerical instabilities.

Property (2) is of importance to model order reduction, because of the need to factor the matrix $G + s_0C$ for some model order reduction algorithms.

In order to show that the GT-PML equations can be cast in a form compatible with PVL, let us consider the simple case where GT-PML is implemented in the $z$ direction only. Following [46], Maxwell's curl equations in the Laplace domain assume the form
\[-s\mu H_x - \sigma_x^* H_x = (\nabla \times E)_x,\]
\[-s\mu H_y - \sigma_y^* H_y = (\nabla \times E)_y,\]
\[-s\mu H_z = (\nabla \times E)_z + \frac{\sigma_z^*}{s\mu} (\nabla \times E)_z,\]
\[se E_x + \sigma_x E_x = (\nabla \times H)_x,\]
\[se E_y + \sigma_y E_y = (\nabla \times H)_y,\]
\[se E_z = (\nabla \times H)_z + \frac{\sigma_z}{se} (\nabla \times H)_z,\]

where \((\nabla \times F)_q\) \((q=x,y,z)\) represents the \(q\) component of the vector \(\nabla \times F\). Using the notation introduced earlier, the semi-discrete matrix compact form of (2.18) is

\[(G + sC)X + \frac{1}{s} QX = 0,\]  
(2.19)

where \(X = [E_x, E_y, E_z, H_x, H_y, H_z]^T\), and the matrix \(Q\) results from the discretization of the operations \((\sigma_x^* / \mu)(\nabla \times E)_x\) and \((\sigma_z / \varepsilon)(\nabla \times H)_z\), respectively. In order to cast (2.19) in a linear form of \(s\), we introduce the column vector [9]

\[X_e = \frac{1}{s} QX.\]  
(2.20)

Notice that the length of \(X_e\) is equal to the sum of the length of the column vector \(H_z\) and the length of the column vector \(E_z\). Using this vector, (2.19) may be cast in the form

\[
\begin{pmatrix}
G & I \\
Q & 0
\end{pmatrix}
\begin{pmatrix}
X \\
X_e
\end{pmatrix} + s
\begin{pmatrix}
C & 0 \\
0 & -I
\end{pmatrix}
\begin{pmatrix}
X \\
X_e
\end{pmatrix} = 0.
\]  
(2.21)

Obviously, (2.21) is exactly the same as (2.8) and thus compatible with PVL. In contrast to Berenger's PML, where the number of state variables is doubled, the number of extra state variables used in (2.21) is much smaller and is dictated by the number of extra
variables introduced by GT-PML. According to [46], it is only at a corner PML region that the number of extra variables in GT-PML is equal to twice the number of field components. Equations similar to (2.21) for edge and corner PML regions can be derived in a similar fashion using the results in [46].

2.3 Passivity of the Discrete Model

Reduced-order model representations of passive electromagnetic multi-ports are extremely useful when the multi-ports constitute parts of more complex functional blocks. For the purposes of design-driven simulation at the functional block level, a network-oriented simulation approach is used. The incorporation of the reduced-order models for the electromagnetic multi-ports in the overall network-oriented circuit simulator may be effected either through recursive convolution, utilizing the pole-residue representations of the elements of the transfer function matrix [48], or through the direct incorporation of the state-space representation of the reduced system in the circuit simulator [49]. In either case, the passivity of the reduced system needs to be verified in order to avoid non-physical instabilities in the subsequent simulation of the overall circuit.

It is important to observe that it is meaningless to talk about passivity of the reduced-order model without establishing first the passivity of the original discrete model. Therefore, our discussion of model passivity will begin with the examination of the passivity of the system of (2.8) that results from the specific discretization of the system of Maxwell's equations. Our analysis will make use of the following useful results [50]:
Theorem 1: The transfer function matrix $H(s)$ of a passive network is positive-real; that is

(a) Each element of $H(s)$ is analytic for $\text{Re}(s) > 0$.

(b) $H(s^*) = H^*(s)$ for $\text{Re}(s) > 0$.

(c) $H_A(s) = H^T(s) + H(s) \geq 0$ for $\text{Re}(s) > 0$.

Theorem 2: If a matrix $A$ is positive-real, then so is its inverse $A^{-1}$, if it exists.

Theorem 3: If $A$ is positive-real and $A_h = A^T + A > 0$ for $\text{Re}(s) > 0$, then $A^{-1}$ exists.

Theorem 4: If $B$ is a real constant matrix and $A(s)$ is a positive-real matrix, then $B^T A B$ is a positive-real matrix.

With the output defined as $Y(s) = z^T X(s)$, the transfer function is

$$H(s) = z^T (G + jC)b.$$ (2.22)

According to Theorem 1, the discrete approximation to the system of Maxwell's equations will be passive if $H(s)$ is positive-real. However, the matrices $z$ and $b$ in (2.22) are real constant matrices. Thus, in view of Theorems 2, 3, and 4, to prove the passivity of the discrete system of (2.8), it suffices to show that the matrix $S = G + sC$ is positive-real and $S_h > 0$ for $\text{Re}(s) > 0$.

This takes us back to Theorem 1 in which the three requirements for a matrix to be positive-real are given. First, we note that matrices $G$ and $C$ are real. Hence, requirement (a) and (b) are automatically satisfied. To prove that requirement (c) is also satisfied, strengthened by the additional requirement $S_h > 0$, we need to show that
$z^T(S^T + S)z > 0$ for $\text{Re}(s) > 0$ and for any complex vector $z$. Setting $s = a + j\omega$, one obtains after some straightforward matrix algebra

$$
z^T(S^T + S)z = z^T[(G + G^T) + a(C + C^T)]z
= z^T[(G + G^T) + 2aC]z.
$$

(2.23)

where we have made use of the fact that $C$ is symmetric. Since $C$ is positive, it follows immediately that $2az^TCz > 0$ for $a > 0$. Furthermore, using the fact that $G$ is skew-symmetric, it is straightforward to show that

$$
G + G^T = \begin{bmatrix}
0 & 0 \\
0 & 2D_\sigma
\end{bmatrix}.
$$

(2.24)

But $D_\sigma$ is a non-negative definite matrix; hence, $z^T(G + G^T)z \geq 0$. Thus, we conclude that the product in (2.23) is positive definite for any complex vector $z$ and for $\text{Re}(s) > 0$. Hence, the discrete system of (2.8) is passive.

It is important to point out that critical to this proof of passivity of the discrete system is the fact that the matrix $G$ is skew-symmetric. Clearly, this skew-symmetry of $G$ is a direct consequence of the uniformity of the orthogonal Cartesian grid used for the discretization, as well as the staggering of the electric and magnetic field nodes. From a numerical integration point of view, it is extremely useful to be able to validate that the semi-discrete system of state equations resulted from the numerical approximations of the spatial derivatives is passive, since passivity guarantees stability and thus a stable numerical solution will always be achieved with a stable integration algorithm. Proof of passivity of the semi-discrete system when unstructured grids are used is rather
cumbersome since, in most cases, it can be performed only through an eigenvalue analysis.

Taking a closer look at the structure of the matrix $G$, it becomes apparent that the specific form of the numerical approximation of the two curl operators in Maxwell's system plays an important role on the passivity of the approximation. In order to account for the general case, let $P_m$ and $P_r$ denote the matrix approximations resulting from the discretization of the curl of the magnetic and electric field, respectively, over the entire computational domain. Then, the matrix $G$ assumes the general form

$$G = \begin{bmatrix} 0 & P_r \\ -P_m & D_\sigma \end{bmatrix}.$$  \hspace{1cm} (2.25)

Thus, the requirement for passivity of the approximation centers on the properties of the matrix

$$G_h = G + G^T = \begin{bmatrix} 0 & (P_r - P_m^T) \\ (P_r - P_m^T)^T & 2D_\sigma \end{bmatrix}.$$  \hspace{1cm} (2.26)

More specifically, considering that $D_\sigma$ is non-negative definite, the passivity of the numerical model depends solely on the properties of the matrix $Q = P_r - P_m$. To elaborate, let us expand the term $z^T (G_h) z$, where $z$ is written in the form

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}.$$  \hspace{1cm} (2.27)

It is then
where the fact that $Q$ is real has been used. The second term in the last equation is always non-negative. Considering that it is common to assume loss-free media for numerical wave simulation purposes, the passivity of the numerical model depends on whether the first term in (2.28) is non-negative for any $z$. Clearly, this is ensured when $Q = 0$, equivalently, $P_r = P_m^T$. This occurs naturally when the structured Yee's lattice is used for the discretization. For other discretization choices, the passivity of the discrete model can be evaluated by examining the matrix $P_r - P_m^T$. If this matrix is non-negative, the discrete model is certainly passive and its numerical integration (with a numerically stable integration scheme) will lead to a stable numerical solution.

It should be pointed out that the aforementioned discussion did not take into account the discrete approximation of any absorbing boundary conditions used for grid truncation purposes. In their discrete form such conditions lead to modifications in the matrices $P_r$ and $P_m$, as well as the matrix $C$ in (2.8), and thus may impact passivity.
3. MODEL ORDER REDUCTION ALGORITHMS

The methods forming the foundation for model order reduction are relatively old. The history of Pade approximation, for example, spans more than one hundred years. The Lanczos algorithm, an important Krylov-based iteration, was first introduced more than 50 years ago [51]. The early moment-matching methods form a reduced-order model from an explicit expression of the desired moments of the original system. Explicit methods such as [52] were utilized to construct Pade approximations. Recently, asymptotic waveform evaluation (AWE) methods have been applied for interconnect model reduction in the area of circuits [3]. The methods received attention for their ability to reduce RC interconnect models involving tens of thousands of unknowns. Unfortunately, all of these explicit moment-matching methods are known to exhibit numerical instabilities, particularly as the dimension of the reduced-order model, denoted as M, grows [5, 53]. Moment-matching via the Lanczos method, or more generally Krylov-based subspace projection method, is a better implementation.

The first significant mathematical connection between the Lanczos algorithm, a Krylov-based technique, and model reduction occurred in the early 1980s [54]. Adaptions of Krylov-based subspace projection were proposed in 1987 to generate Pade approximations and shifted Pade approximations [55]. Very recently, Lanczos-based model reduction has become a popular topic in the area of high-speed circuits [4, 5, 6, 9, 31, 35, 36, 49] as well as other areas.
3.1 Model Order Reduction of Linear Systems

As formulated in the last chapter, a state-space system of first-order differential equations is derived after applying a finite difference procedure to the electromagnetic field interactions bounded by PEC, PMC, and PML. This linear system can be a multiple input multiple output (MIMO) system or a single input single output (SISO) system. The formulations of reduced-order modeling algorithms will be based on a SISO system for the sake of simplicity. For MIMO systems, the formulations have a similar format. In the Laplace domain, the single input single output (SISO) state-space system is

\[ s \mathbf{C} \mathbf{X}(s) = -\mathbf{G} \mathbf{X}(s) + \mathbf{b} U(s), \]

\[ Y(s) = z^T \mathbf{X}(s), \]  

where the scale function \( Y(s) \) is the system's output of interest, the vector \( \mathbf{X} \) represents the electric and magnetic field components inside the truncated space, and the matrices \( \mathbf{G} \) and \( \mathbf{C} \) result from the discretization of the electromagnetic system. \( \mathbf{b} U(s) \) represents the excitation from an independent source. Also, the system matrix \( \mathbf{G} \) is assumed to be sparse or structured and the descriptor matrix \( \mathbf{C} \) is a diagonal matrix. Then, (3.1) can be recast as

\[ s \mathbf{C}^{-1} \mathbf{C} \mathbf{X}(s) = -\mathbf{C}^{-1} \mathbf{G} \mathbf{X}(s) + \mathbf{C}^{-1} \mathbf{b} U(s), \]

\[ Y(s) = z^T \mathbf{X}(s). \]  

If we rename \( \mathbf{C}^{-1} \mathbf{G} \) as \( \mathbf{G} \) and \( \mathbf{C}^{-1} \mathbf{b} \) as \( \mathbf{b} \), then the state-space system can be written as

\[ s \mathbf{X}(s) = -\mathbf{G} \mathbf{X}(s) + \mathbf{b} U(s), \]

\[ Y(s) = z^T \mathbf{X}(s). \]
We are interested in determining the impulse response of the linear system (3.1) with zero initial conditions, which can be used to determine the response to any excitation. The Laplace domain impulse responses are given by the frequency responses or transfer functions of the system, i.e., (3.1) and (3.3) below, respectively,

\[ H(s) = \frac{Y(s)}{U(s)} = z^T (G + sC)^{-1}b, \quad (3.4) \]

\[ H(s) = \frac{Y(s)}{U(s)} = z^T (G + sI)^{-1}b, \quad (3.5) \]

where \( I \) is the unity matrix. Let \( s_0 \) be an arbitrary, but fixed expansion point such that the matrix \( G + s_0I \) is nonsingular. With the change of variable \( s = s_0 + \sigma \), (3.5) can be rewritten as

\[ H(s_0 + \sigma) = z^T (\sigma I + G + s_0I)^{-1}b \]

\[ = z^T (I - \sigma A)^{-1}r, \quad (3.6) \]

where \( A = -(G + s_0I)^{-1} \), and \( r = (G + s_0I)^{-1}b \). Assuming that the matrix \( A \) is diagonalizable, then (3.6) can be written as

\[ H(s_0 + \sigma) = z^T (I - \sigma \Lambda S \Lambda^{-1})^{-1}r \]

\[ = z^T S(I - \sigma \Lambda)^{-1}S^{-1}r \]

\[ = f^T (I - \sigma \Lambda)^{-1}g, \quad (3.7) \]

where \( \Lambda = \text{diag} (\lambda_1, \lambda_2, \cdots, \lambda_N) \) is a diagonal matrix whose diagonal elements are the eigenvalues of matrix \( A \), and matrix \( S \) contains the corresponding eigenvectors as columns. Therefore, we have

\[ H(s_0 + \sigma) = \sum_{i=1}^{N} \frac{f_i g_i}{1 - \sigma \lambda_i} \quad (3.8) \]
where \( f_i \) and \( g_i \) are the components of the vectors \( f \) and \( g \). Obviously, it is prohibitively expensive to compute all of the eigenvalues and eigenvectors of matrix \( A \) as its size reaches a few hundreds while the degrees of freedom in electromagnetic field simulation can be a few millions. An alternative practical way to obtain the broadband frequency response of a large linear system is through the use of a lower order approximation function that can represent the response of the system accurately over a broad frequency range. It is the goal of reduced-order modeling to approximate (3.8) with a pole-residue representation that includes only the poles needed for sufficient accuracy in the temporal scale of interest.

More specifically, the fundamental motivation behind reduced-order modeling is to seek an approximation of the transfer function, \( H(s) \), of the linear system, which can describe adequately the properties of the system, instead of calculating the transfer function exactly. One of the possible approaches is the Padé approximation. For each pair of integers \( p, q \geq 0 \), the Padé approximation (of type \( (p/q) \)) to the network frequency response \( H(s_0 + \sigma) \) is the rational function

\[
H_{p,q}(s_0 + \sigma) = \frac{b_0 + b_1\sigma + \cdots + b_p\sigma^p}{1 + a_1\sigma + \cdots + a_q\sigma^q}
\]  

(3.9)

whose Taylor series about \( s_0 \) agrees with the Taylor series of \( H(s_0 + \sigma) \) in at least the first \( p + q + l \) terms, i.e.,

\[
H_{p,q}(s_0 + \sigma) = H(s_0 + \sigma) + O(\sigma^{p+q+l}).
\]  

(3.10)
If the power $p$ of the numerator is chosen as $q-1$, then the coefficients $a_1, a_2, \ldots, a_q, b_0, b_1, \ldots, b_{q-1}$ of the Pade approximation are uniquely determined by matching the first $2q$ Taylor coefficients (moments), $m_n$, of the exact transfer function with those of the approximate transfer function $H_q(s_0 + \sigma)$,

$$\frac{1}{n!} \frac{d^n H_q}{ds^n} = \frac{1}{n!} \frac{d^n H}{ds^n} = m_n, \quad n = 0, 1, \ldots, 2q - 1. \quad (3.11)$$

The roots of the denominator and numerator polynomials in (3.9) represent the dominant poles and zeros of the system, respectively. The specific form of (3.6) leads to the following Taylor series for the transfer function

$$H(s_0 + \sigma) = z^r (I + \sigma A + \sigma^2 A^2 + \cdots) r = \sum_{n=0} m_n \sigma^n. \quad (3.12)$$

Where $m_n = z^r A^n r$, and $n = 0, 1, 2, \ldots$ are the moments of the system response. By using a partial-fraction decomposition, we can write $H_q$ in the pole-residue form

$$H_q(s_0 + \sigma) = k_\omega + \sum_{j=1}^{q'} \frac{k_j}{\sigma - p_j}, \quad (q' \leq q). \quad (3.13)$$

Similarly, the transfer function can also be expanded at infinity, which results in a Markov series instead of a Taylor series,

$$H(s_0 + \sigma) = d + z^r (\sigma^{-1} A + \sigma^{-2} A^2 + \cdots) r = d + \sum_{n=1} m_{-n} \sigma^{-n}. \quad (3.14)$$

where $m_{-n} = z^r A^n r$, and $n = 1, 2, 3, \ldots$ are the Markov parameters in this case. In this equation, we have defined $A = -(G + s_0 I)$ and $r = b$. 
The Pade approximation in the asymptotic waveform evaluation (AWE) [3] can be used to calculate the moments of the system response and furthermore to find $H_q(s_0 + \sigma)$, which is the reduced-order approximation of the original transfer function. This approach is the aforementioned explicit moment-matching method. Calculation of the moments is a recursive procedure that can be performed very efficiently since the matrix $G + sj_0I$ is LU-factorized only once. Then each moment $m_n$ is obtained by re-solving the system with a different right-hand side at the cost of only one forward-backward substitution. As $q$ is increased, one would expect to obtain a more and more accurate approximation $H_q(s)$ of the exact frequency response $H(s)$. For Markov parameter matching, there is even no need to factorize the matrix $G + sj_0I$. Each Markov parameter is obtained by a matrix vector multiplication. Unfortunately, the moment matching process (the calculation of the coefficients in the Pade approximation) tends to be extremely ill-conditioned as the order of approximation is increased [5, 53]. The round-off errors will affect the accuracy so severely that an accurate Pade approximation can never be achieved by AWE for a system with a large number of significant poles. Despite its success, the AWE algorithm has inherent numerical limitations that restrict its application to systems that can be modeled accurately using only a relatively small number of poles.

Because of this limitation, the AWE algorithm is not used in ROMES. Instead, subspace projection algorithms have been introduced as preferred in [5, 53]. In (3.7), the transfer function has been written in eigenvalue-decomposition format. The objective of
reduced-order modeling is to find the dominant eigenvalues and corresponding eigenvectors of $A = -(G + s_0 I)^{-1}$, or for the frequency shifted expression $A = -(G + s_0 I)$ for expansion at infinity. The reduced-order matrix is $\hat{A} = S_q A q S_q^{-1}$, which is an approximation of $A = S A S^{-1}$. It is very easy to find the moments after calculating the reduced-order matrix $\hat{A}$.

Due to its relative numerical elegance and reliability, the nonsymmetrical Lanczos algorithm, one of the Krylov-based subspace projection methods, has become a popular choice for moment matching, model reduction methods. The nonsymmetrical Lanczos method is due to C. Lanczos and was originally proposed as a method for solving linear systems of equations and eigenvalue problems [51]. Because we are interested in nonsymmetrical matrices throughout the dissertation, the nonsymmetrical designation of Lanczos method versus the symmetric Lanczos method is dropped.

The algorithm of Lanczos computes rectangular matrices $V$ and $W ∈ \mathbb{R}^{N×q}$ that restrict a specified system matrix $A$ to a tridiagonal form,

$$
T_q = W^T A V = \begin{bmatrix}
\alpha_1 & \beta_2 & 0 & \cdots & 0 \\
\rho_2 & \alpha_2 & \beta_3 & \cdots & 0 \\
0 & \rho_3 & \ddots & \ddots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & \rho_q & \alpha_q
\end{bmatrix},
$$

(3.15)

and that satisfy

$$
\text{colsp}\{V\} ∈ \text{Kr}_q(A, v) = \text{span}(v_1, Av_1, \cdots, A^{q-1} v_1)
$$

(3.16)
43

\[ \text{colsp}(W) \in \text{Kry}_K(A^T, w_i) = \text{span}(w_i, A^T w_i, \cdots, (A^T)^{\nu-1} w_i), \]  

(3.17)

where \text{colsp} stands for column space. The vectors \( v_i \) and \( w_i \) are user-specified starting vectors that lie in the direction of the first columns of \( V \) and \( W \). Equivalently, the Lanczos algorithm can be viewed as an approach for constructing biorthogonal matrices \( V \) and \( W \), i.e., \( W^T V = I \), that satisfy the same Krylov subspace conditions (3.16) and (3.17). The columns of the projectors \( V \) and \( W \) satisfying these constraints can be iteratively computed via three-term recursion relations that will be detailed in the next section. Actual implementation of the Lanczos method may encounter numerical difficulties including a loss of biorthogonality and so-called serious breakdowns [56]. However, these breakdowns are less drastic than the breakdowns occurring in explicit moment-matching. Additionally, one remedy was discussed in [56].

For more general Krylov-based subspace projection methods, the projection extracts an approximate solution of dimension \( M \) from a subspace \( S \). In order to be precisely defined, this approximation is chosen from \( S \) so that \( M \) constraints are satisfied. The subspace \( T \) is associated with these constraints. For example, we typically require that the approximate solution is chosen from \( S \) so that its residual is orthogonal to a specified \( T \). Such constraints are known as Petrov-Galerkin conditions. If this \( T \) equals \( S \), then the projection is orthogonal. Otherwise, the projection is said to be oblique. A more detailed review about this projection technique can be found in [57].

The subspaces \( S \) and \( T \) can be represented by rectangular matrices \( V \in \mathbb{R}^{N \times M} \) and \( W \in \mathbb{R}^{N \times M} \), whose columns form bases for the respective subspaces. The reduced-
order model is determined by retaining only the leading $M \times M$ subsystem of the transformed original system (3.1). The components of the reduced-order model are

$$\hat{G} = W^T GV, \quad \hat{b} = W^T b, \quad \hat{z} = V^T z, \quad \hat{C} = W^T CV.$$  

(3.18)

The quantities in (3.18) are said to be the restrictions of the original system matrices by $W$ and $V$. Corresponding to the transform function (3.4), the approximate transfer function written in terms of reduced-order model is

$$\hat{H}(s) = \hat{z}^T (\hat{G} + s\hat{C})^{-1}\hat{b}.$$  

(3.19)

Similarly, for system (3.3) and transfer function (3.5), the components of the reduced-order model are

$$\hat{G} = W'^T GV, \quad \hat{b} = W'^T b, \quad \hat{z} = V'^T z, \quad \hat{C} = W'^T CV.$$  

(3.20)

The approximate transfer function in terms of reduced-order model is

$$\hat{H}(s) = \hat{z}^T (\hat{G} + s\hat{I})^{-1}\hat{b}.$$  

(3.21)

There are different choices of $W$ and $V$ available in [36] including $W$ and $V$ orthogonal, biorthogonal, and neither of them. For the state-space system (3.3), extremely simple choices for $W$ and $V$ are

$$\text{colsp}\{W(s)\} = \text{Ker}_M \left[\begin{bmatrix} -\left(G + sI\right)^{-1} \end{bmatrix}^T, z \right]$$

$$= \text{span}\left\{ z, \left[ -\left(G + sI\right)^{-1} \right]^T z, \cdots, \left[ -\left(G + sI\right)^{-1} \right]^{M-1} z \right\},$$  

(3.22)

and

$$\text{colsp}\{V(s)\} = \text{Ker}_M \left(-\left(G + sI\right)^{-1}, b\right)$$

$$= \text{span}\left\{ b, -\left(G + sI\right)^{-1} b, \cdots, -\left(G + sI\right)^{-1} b \right\},$$  

(3.23)
where $\text{Kr}_M([-G + sI]^{-1}, b)$ stands for the $m^{th}$ dimensional Krylov subspace corresponding to matrix $-(G + sI)^{-1}$ and vector $b$. For the Krylov and rational Krylov implementation in this dissertation, in which the columns of $W$ and $V$ span unions of Krylov subspaces through rational interpolations, the simple choices for $W$ and $V$ explained above will be used.

It is important to note that there are an infinite number of $W$ and $V$ whose columns are acceptable bases for given subspaces $S$ and $T$. It is only necessary to know that $W$ and $V$ satisfy $\text{colsp}\{W\} = T$ and $\text{colsp}\{V\} = S$, because in theory all possible choices for the bases lead to identical results up to a similarity transformation [36].

3.2 Reduced-Order Modeling Algorithms

The Padé approximation in AWE is not used by ROMES because of the ill-conditioned iteration. However, it served as the inspiration for modern reduced-order modeling research. We will include the detailed AWE implementation in this dissertation for the sake of completeness. Subspace projection algorithms, including Padé via Lanczos (PVL) with expansion at a finite single frequency, frequency segmentation technique enhanced PVL with expansion at finite frequency, Krylov, rational Krylov, and PVL with expansion at infinity have been implemented in ROMES. The new ROMES provides all the above options for reduced-order modeling rather than only allowing PVL with expansion at a single finite frequency. These algorithms are well developed by mathematicians. Therefore, we only summarize them and give their
properties for application in electromagnetic problems. The formulations correspond to the state-space system of (3.3).

Algorithm 1: Pade Approximation in AWE

In AWE, the Pade approximant \(H_q\) is obtained via explicit computation of the leading \(2q\) moments of the transfer function \(H(s)\) as

\[
m_n = z^T A^T r = z^T u_n, \quad n = 0, 1, 2, \ldots, 2q - 1.
\]  
(3.24)

A group of vectors \(u_0 = r, u_1 = A r, u_2 = A^2 r, \ldots, u_{2^q-1} = A^{2^q-1} r\) are first generated by recursive solution of the linear systems

\[
(G + s_0 I) u_k = -u_{k-1}, \quad k = 1, 2, \ldots, 2q - 1,
\]  
(3.25)

with the initial vector

\[
u_0 = (G + s_0 I)^{-1} b.
\]  
(3.26)

Observe that the recursive computation of the vectors \(u_k\) can be performed very efficiently. The matrix \(G + s_0 I\) is LU-factorized exactly once. Then each vector \(u_k\) is obtained by re-solving the system with a different right-hand side at the cost of only one forward-backward substitution. The moments are then computed as (3.24).

As the next step, AWE computes the coefficients of the denominator polynomial of the representation (3.9) of \(H_q\) via solution of the linear system

\[
\begin{bmatrix}
m_{q-1} & m_{q-2} & \cdots & m_0 \\
m_q & \ddots & \ddots & \vdots \\
m_{2q-3} & \cdots & m_{q-1} & \ddots \\
m_{2q-2} & m_{2q-3} & \cdots & m_{q-1}
\end{bmatrix}
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_q
\end{bmatrix}
= -
\begin{bmatrix}
m_q \\
m_{q+1} \\
\vdots \\
m_{2q-1}
\end{bmatrix}.
\]  
(3.27)
where the coefficient matrix is denoted as the moment matrix $M_q$. The poles $p_j$ of $H_q$ in (3.13) are then obtained as the roots of the equation

$$1 + a_0 \sigma + \cdots + a_{q-1} \sigma^{q-1} + a_q \sigma^q = 0.$$  \hspace{1cm} (3.28)

The constant $k_\omega$ and residues $k_j$ in (3.13) are computed by solving another linear system of order $q$

$$\begin{bmatrix}
1 & p_1^{-1} & \cdots & p_q^{-1} \\
0 & p_1^{-2} & \cdots & p_q^{-2} \\
\vdots & \vdots & \ddots & \vdots \\
0 & p_1^{-q-1} & \cdots & p_q^{-q-1}
\end{bmatrix}
\begin{bmatrix}
k_\omega \\
k_1 \\
\vdots \\
k_q
\end{bmatrix}=
\begin{bmatrix}
m_0 \\
m_1 \\
\vdots \\
m_q
\end{bmatrix}.$$  \hspace{1cm} (3.29)

As $q$ increases, one would expect more and more accurate approximations $H_q$ of the exact frequency response $H$. Unfortunately, this is not the case when the Pade approximant $H_q$ is generated with AWE. Indeed, as observed by Peter Feldmann [5], typically $H_q$ improves only for values of $q$ up to about $q = 10$, and after that the process stagnates. The reason for the stagnation of AWE is the particular computation of the Pade approximation used in AWE, and not the Pade approximation itself. More specifically, the explicit use of the moments results in extremely ill-conditioned numerical computations. This is especially the case for the numerical solution of the linear system (3.27). The condition number of the moment matrix $M_q$, $\text{cond}(M_q)$, is a measure of how round-off error affects the accuracy of the numerically computed solution of (3.27). Each increase of $\text{cond}(M_q)$ by a factor of 10 signals the loss of one decimal digit of accuracy in the computed solution. Scaling was proposed as a remedy
by X. Huang [58] but it can not solve the problem completely. The fundamental reason was explained in [5] that the vector \( u_k \) and also the moment \( m_k = z^T u_k \) generated using AWE contain only information corresponding to one eigenvalue of \( A \), the one with largest absolute value. On the other hand, in view of (3.8), the function \( H \) to be approximated clearly depends on all eigenvalues of \( A \). Therefore, for the accuracy of the numerically computed Pade approximant \( H_q \) to improve with increasing \( q \), an algorithm must be able to recover information about more than one eigenvalue of \( A \).

**Algorithm 2: Pade via Lanczos Algorithm (PVL)**

For the linear system (3.3), define \( A = -(G + s_0 I)^{-1} \) and \( r = (G + s_0 I)^{-1} b \). Let \( N \) denote the total number of state variables. The number of ports in the multi-port is denoted by \( p \).

The Lanczos algorithm reduces the system matrix \( A \) to a \( q \times q \) tridiagonal matrix, \( \hat{A} = T_q = \text{Lanczos}(A, r, A^T z) \) [5]. For the sake of completeness, the Lanczos algorithm is outlined below:

1. Set \( \rho_i = \| r \|_2 \), \( \eta_i = \| z \|_2 \), \( v_1 = r / \rho_1 \), and \( w_1 = z / \eta_1 \).

   Set \( v_0 = w_0 = 0 \) and \( \delta_0 = 1 \).

2. For \( n = 1, 2, \ldots, q \) do:
   - Compute \( \delta_n = w_n^T v_n \).
   - Set
     \[
     \alpha_n = \frac{w_n^T A v_n}{\delta_n}, \quad \beta_n = \frac{\delta_{n-1} - \eta_n}{\delta_n}, \quad \gamma_n = \frac{\delta_n}{\delta_{n-1} \rho_n}.
     \] (3.30)
Set

\[ v = A v_n - v_n \alpha_n - v_{n-1} \beta_n, \]
\[ w = A^T w_n - w_n \alpha_n - w_{n-1} \gamma_n. \]

(3.31)

Set \( \rho_{n+1} = \|v\|_2, \eta_{n+1} = \|w\|_2, \) and

\[ v_{n+1} = \frac{v}{\rho_{n+1}}, \quad w_{n+1} = \frac{w}{\eta_{n+1}} \]

(3.32)

The tridiagonal matrix \( T_q \) is constructed as

\[
T_q = \begin{bmatrix}
\alpha_1 & \beta_1 & 0 & \cdots & 0 \\
0 & \alpha_2 & \beta_2 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & \cdots & \cdots & \beta_q \\
0 & \cdots & \cdots & \cdots & \rho_q & \alpha_q
\end{bmatrix}.
\]

(3.33)

The Lanczos matrix has the following relation to the original matrix \( A \):

\[ AV = VT_q, \ W^T V = D. \]

The matrix \( V_{M \times q} \) forms an orthogonal basis for the Krylov subspace

\[ Kr_q (A, r) = \text{colsp}[r, Ar, A^2 r, \ldots, A^{q-1} r], \]

(3.34)

and the matrix \( W_{M \times q} \) forms another orthogonal basis for the Krylov subspace

\[ Kr_q (A^T, z) = \text{colsp}[z, A^T z, (A^T)^2 z, \ldots, (A^T)^{q-1} z]. \]

(3.35)

The two matrices \( V \) and \( W \) are generated to be bi-orthogonal. Hence, the matrix \( D \) is a diagonal matrix. Traditionally, the Lanczos algorithm assumes only matrix-vector products with easily accessible matrices. For the modeling approaches outlined above,
the matrix inverse must also be treated. The matrix inverses are actually implemented through solving systems of linear equations in (3.31).

It should be mentioned that the Lanczos method avoids the difficulties encountered in explicit moment matching. It does so by storing its modeling information in two biorthogonal matrices \( V \) and \( W \) rather than in moments. The biorthogonality of \( W \) and \( V \) ensures that new information about the original matrix is introduced into the projectors at every step.

Consider the change of variables \( X = V \hat{X} \), and substitute it into 
\[
(I - \sigma A)X = rU(s),
\]
then
\[
(I - \sigma T_q)\hat{X} = D^{-1}W^T rU(s)
\] (3.36)

Since \( q \) is rather small and \( T_q \) is a block tridiagonal matrix, the inversion of the matrix \( (I - \sigma T_q) \) is very fast.

The reduced-order system can be written in terms of a transfer function
\[
\hat{Y}(s) = z^T V (I - \sigma T_q)^{-1} D^{-1} W^T rU(s)
\]
\[
= \rho z^T V (I - \sigma T_q)^{-1} e_i U(s)
\] (3.37)
where \( e_i = [1, 0, 0, ..., 0]^T \in \mathbb{R}^n \) is the first unit vector in \( \mathbb{R}^n \). It can be shown that the expansion point is given simply as \( s_0 = 2 \pi f_{\text{max}} \) for the single expansion point, where \( f_{\text{max}} \) is the maximum frequency of interest.

Obviously, the Pade approximation using Lanczos algorithm with numerical stable generation of a large number of poles is more robust for model order reduction of large linear electromagnetic systems when compared with the Pade approximation in
AWE. The disadvantage of this algorithm is the accuracy limitation for broad-band problems. In other words, when $\sigma$ increases, the order of $T_q$, which can be used to approximate $A$, increases too. When using the same $T_q$ to approximate $A$ for higher $\sigma$, the accuracy will decrease. For some of the problems, the accuracy does not increase even as the order of $T_q$ is increased. The accuracy limitation of PVL, with the finite single frequency expansion, has also been discussed in [5] and [35]. There, rational subspace projection algorithms were proposed for accuracy improvement.

**Algorithm 3: PVL with multiple-point expansions**

In order to increase the calculation accuracy, a frequency segmentation technique has been applied to wide-band problems. For each frequency segment, PVL has been used to develop the reduced-order model for that frequency band. This technique can increase the calculation accuracy, while keeping the order of the reduced-order model low. It can also provide a criteria to judge whether the order of the reduced-order model is high enough to approximate the original system accurately. We can compare the calculated values using different reduced-order models at the frequency segment boundaries to judge whether the order is high enough. When the order of the reduced-order model is high enough, the values calculated using different models match very well. The method is robust. For increased orders of the macro-model and increased numbers of segments, the accuracy improves and no instability is observed.

**Algorithm 4: Rational Krylov algorithm**

Besides using the three term iteration approach to construct the projectors $W$ and $V$, many other different methods can also be used to create these projectors. Because of
the accuracy limitations of PVL for wide-band problems, the rational Krylov algorithm has also been implemented in ROMES. The columns of the projectors $W$ and $V$ span unions of subspace through rational interpolation. For the state-space system in (3.3), the rational Krylov algorithm used in our work is outlined below, where $K$ is the number of the expansion point and $J_k$ is the number of iterations involved for the $k^{th}$ expansion point.

- Set $m = 0$
- For $k = 1$ to $K$, do:
  - For $j_k = 1$ to $J_k$, do:
    - If $j_k = 1$,
      $$\tilde{v}_m = (\mathbf{-G - s_0^{(k)} I}^{-1}) \mathbf{b} \text{ and } \tilde{w}_m = (\mathbf{-G - s_0^{(k)} I}^{-T}) \mathbf{z}$$
    - else
      $$\tilde{v}_m = (\mathbf{-G - s_0^{(k)} I}^{-1}) \mathbf{v}_{m-1} \text{ and } \tilde{w}_m = (\mathbf{-G - s_0^{(k)} I}^{-T}) \mathbf{w}_{m-1}$$
  - end
  - $v_m = \tilde{v}_m / \|\tilde{v}_m\|_2$ and $w_m = \tilde{w}_m / \|\tilde{w}_m\|_2$
  - $m = m + 1$
  - end
- end

Assuming the order of the original system is $N$ and the order of the reduced-order model is $M$, the rational Krylov algorithm reduces the system matrix $\mathbf{G}$ to a $M \times M$ matrix $\hat{\mathbf{G}}$ such that
\[ \hat{G} = W^T GV, \hat{b} = W^T b, \hat{z} = V^T z, \hat{I} = W^T IV. \] (3.38)

The matrix \( V_{N \times M} \) forms a basis for the Krylov subspace

\[ Kr_m((-G - sI)^{-1}, b) = \text{colsp}[b, (-G - sI)^{-1}b, \cdots, ((-G - sI)^{-1}b^M b]. \] (3.39)

and the matrix \( W_{N \times M} \) forms another basis for the Krylov subspace

\[ Kr_m((-G - sI)^{-T}, c) = \text{colsp}[c, (-G - sI)^{-T}c, \cdots, ((-G - sI)^{-T}c^M c]. \] (3.40)

The two matrices \( V \) and \( W \) are generated to be bi-orthogonal. The original system in terms of transfer function is

\[ Y(s) = z^T (G + sI)^{-1} b U(s), \] (3.41)

and the reduced-order system is

\[ \hat{Y}(s) = \hat{z}^T (\hat{G} + \hat{sI})^{-1} \hat{b} U(s) \]
\[ = z^T V(sW^T IV + W^T GV)^{-1} W^T b U(s). \] (3.42)

The rational Krylov algorithm has an advantage in regards to accuracy. This can be seen from the numerical example in the next chapter. The projectors \( W \) and \( V \) are created by applying \((-G - sI)^{-1}\) and \((-G - sI)^{-T}\) to vectors \( b \) and \( z \), where \( s \) is fixed at some value. The disadvantage of the rational Krylov algorithm is instability. When the interpolation point changes from one value to another value during the rational Krylov iteration, numerical problems, such as a very small or large norm of the \( w, v \) vectors, can happen if the interpolation points are not well selected.

The resulting reduced-order model does not preserve the block tridiagonal property that the PVL model has. Therefore, the inversion time of the reduced-order
model for further frequency sweeps is not negligible when the dimension of the problem is very high. This has been shown in the numerical example in the next chapter.

Algorithm 5: Krylov algorithm

Because of the instability problem associated with the rational Krylov algorithms, we tried using a single interpolation point to create the projectors $W$ and $V$. This overcame the instability problem. It also provided better accuracy than the PVL algorithm. The cost is in the increased computational requirement when compared to the PVL algorithm.

Algorithm 6: PVL with expansion at infinity

All the above algorithms require the LU factorization of the original system matrix $(-G - s_0 I)$. When the dimension of this matrix is very high, the computational requirements are very large or even impossible. PVL with expansion at infinity has been implemented in ROMES. With this technique, the matrix factorization can be avoided and the PVL iteration steps are not solved using linear system theory but only require matrix vector multiplications. For the state-space system (3.3), define $A_\omega = -(G + s_0 I)$, $r_\omega = b$. Let $N$ denote the total number of original state variables and $M$ denote the dimension of the reduced-order model. The number of ports in the multi-port is denoted by $p$. The same Lanczos iteration steps as in the PVL algorithm can be applied as

$$\hat{A}_\omega = T_q = \text{Lanczos}(A_\omega, r, A_\omega^T, z), \quad (3.43)$$
where the Lanczos algorithm reduces the system matrix $A_n$ to a $q \times q$ tridiagonal matrix, $T_q$, such that $A_n V = V T_q$ and $W^T V = D$. The matrix $V_{n \times M}$ forms an orthogonal basis for the Krylov subspace

$$K_{r_n}(A_n, r) = \text{colsp}[r, A_n r, A_n^2 r, ..., A_n^{M-r} r],$$

(3.44)

and the matrix $W_{n \times M}$ forms another orthogonal basis for the Krylov subspace

$$K_{r_n}(A_n^T, z) = \text{colsp}[z, A_n^T z, A_n^{T^2} z, ..., A_n^{T(M-r)} z].$$

(3.45)

The transfer function of original system is

$$Y(s_0 + \sigma) = z^T (\sigma I + G + s_0 I)^{-1} b U(s)$$

$$= \frac{1}{\sigma} z^T (I - \frac{1}{\sigma} A_n)^{-1} r U(s)$$

(3.46)

The transfer function of the linear system in terms of reduced-order model is

$$\hat{Y}(s_0 + \sigma) = \frac{1}{\sigma} z^T r_n \cdot \vec{e}_i^T (I - \frac{1}{\sigma} T_q)^{-1} \vec{e} U(s)$$

(3.47)

Comparing with PVL with expansion at finite frequency, where the results are only correct near the expansion point, PVL with expansion at infinity gives better results over a wider frequency range. However PVL with expansion at infinity converges more slowly than PVL with expansion at finite frequency. The resulting macro-model also has a higher dimension than the PVL with finite frequency expansion. Because the resulting macro-model is a tridiagonal or block tridiagonal matrix, obtaining the inverse of this matrix is very efficient. The increased calculation time due to the higher order of the macro-model is negligible.
3.3 Development of Passive Model Order Reduction

In recent years, Krylov-subspace projection methods have become popular tools for reduced-order modeling of large-scale linear systems in different application areas. Despite the success of Krylov-subspace methods, there are still some important issues, such as preservation of passivity, loss of orthogonality, convergence criteria, etc.. When applied to passive systems, the Krylov-subspace based algorithms do not preserve passivity in general. The passivity of the resulting reduced-order models has to be checked before one can proceed with the incorporation of the reduced-order model in a network-oriented circuit simulator. The congruence transformations [59] or so-called one-sided projection [60] can be implemented to the Krylov-subspace based algorithms discussed above to generate a passive reduce-order model of the original discrete system.

For the state-space system in (3.1), applying the one-sided projections to \( G, \ C, \) and \( b \) with the matrix \( V \) generated by the Krylov-subspace based algorithms yields

\[
[(V^T G V) + s(V^T C V)] \hat{X} = (V^T b) U(s),
\]

where \( X = V \hat{X} \). The output vector is also given in terms of the state vector \( \hat{X} \),

\[
Y = (z^T V) \hat{X}.
\]

The above equations may be cast in the compact form

\[
(\hat{G} + s\hat{C}) \hat{X} = -\hat{b} U(s),
\]

\[
\hat{Y} = z^T \hat{X},
\]

where the reduced order matrices are given by the transformations
The transfer function, $\hat{H}(s)$, of the reduced system (53), is then given by

$$
\hat{H}(s) = \hat{z}^T (\hat{G} + s \hat{C})^{-1} \hat{b}.
$$  \hspace{1cm} (3.52)

The passivity of the reduced system can be proven [61] in a manner similar to that used in the previous chapter to prove the passivity of the original discrete model. Clearly, the congruence transformation matrix, $V$, needs to be generated and stored before the model-order reduction process is performed. Considering that the dimension of $V$ is $N \times q$, with $N$ in the order of several hundreds of thousands for realistic problems, this is an expensive proposition from a memory storage point of view. Thus, despite its success with the passive reduction of RLC circuits, the feasibility of the application of this algorithm to passive reduction of three-dimensional electromagnetic systems is questionable. The obvious exception is the case of passive model order reduction of networks containing transmission line systems [62]. In this case the distributed electromagnetic systems are one-dimensional. Hence, the number of degrees of freedom in the discrete model is on the order of several hundreds rather than hundreds of thousands.
4. REDUCED-ORDER MODELING CODE IMPLEMENTATION AND NUMERICAL EXAMPLES

In Chapter 2, we discussed how to develop the state-space linear system using a semi-discrete electromagnetic model. We also discussed one of the boundary conditions, PMLs, that can be implemented in a format ready for reduced-order modeling. The passivity of the semi-discrete electromagnetic model was also proven in Chapter 2. In Chapter 3, we discussed various Krylov-subspace based reduced-order modeling algorithms. Those algorithms can be applied to state-space linear systems resulting from the FDFD approach, thereby leading to a reduced-order model for the original semi-discrete electromagnetic system.

In this chapter, we will discuss how to realize the code implementation for the reduced order models. The input date file format and the functional namelist commands will also be presented. The code will serve as an electromagnetic field solver. Electromagnetic systems with discontinuities will be analyzed using the developed code. The electric and magnetic field values will be calculated and further used to characterize the systems in the numerical examples section. The properties of those reduced-order modeling algorithms will be discussed which focusing on the accuracy, efficiency, and stability for these examples.

4.1 Code Implementation

The flow diagram of ROMES is shown as Fig. 4.1. The flow diagram shows the relationships between several programs. The optimal reduced-order modeling algorithm
can be chosen to suit specific problems of different dimensions by selecting different program combinations.

Fig. 4.1 The flow diagram of ROMES.

The mesh generator of ROMES consists of a user-defined input data file, M3.IN, two FORTRAN77 source files, M3TOOL.F and M3.F. M3TOOL.F is the interface program for the mesh generator M3.F. The mesh generator M3.F is used to generate three-dimensional, rectangular, non-uniform grids.
The computational domain, rectangular or cylindrical objects, and rectangular or circular infinite sheets can be defined in M3.IN by the following rules:

1. The first row in M3.IN should be either blank or a short comment statement that should be less than 100 characters.

2. From the second row onward, the first column must be blank. The NAMELIST commands start from second column with the format

   $namelist name
   variable 1=..., variable 2=..., $end.

   The blank row will be neglected. There are six NAMELIST commands in M3.IN, which will be read by M3TOOL.F. The details about these commands can be found in APPENDIX A.

3. The row that begins with star * at the first column will be neglected. This comment line can be used for the explanation of the structure defined by the commands.

4. Either an end or END statement must begin from first column, i.e. no blank at the first column, to finish the last row of the M3.IN file.

   The M3TOOL.F program may only be compiled once under certain FORTRAN compiler environments. However, M3.F must be recompiled each time after running the executable code M3TOOL.

   ROMES also consists of a user self-defined input data file, ROMES.IN, seven FORTRAN77 source files, RTOOL.F, R1.F, R2.F, R3.F, R3_INF.F, RKRY.F, and R4.F. RTOOL.F is the interface program for ROMES. R1.F is the pre-processor of ROMES. R2.F generates the matrix elements that are required by the linear equations solver. The
coefficient matrix for the linear equations is factorized using an existing public-domain code UMPACK2.0 in R3.F. The PVL with expansion at finite frequency, PVL with expansion at infinity, and single expansion Krylov or rational Krylov algorithms are implemented in R3.F, R3_INF.F, and RKRY.F, respectively. The output data files are generated by R4.F.

The RTOOL.F may only be compiled once under certain FORTRAN compiler environments. Also, UMPACK2.0.F may be compiled once to output a FORTRAN object file. However, R1.F, R2.F, R3.F, R3_INF.F, RKRY.F, and R4.F must be recompiled each time after running RTOOL.F. A typical batch file may have the following statements.

```plaintext
# compile first program
f77 -O3 m3tool.f -o m3tool
# run first program
m3tool
# compile m3.f
f77 -O3 m3.f -o m3
# run m3
m3
# compile rtool
f77 -O3 rtool.f -o rtool
# run rtool
rtool
# compile r1
f77 -O3 r1.f -o r1
# run r1
r1
# compile r2
f77 -O3 r2.f -o r2
# run r2 and time it.
time r2
# setup imsl
setup imsl
# compile r3, r3_inf, and rkry.
f77 -O3 r3.f umpack.o -o r3
```
Using any text editor, users can generate a text file named ROMES.IN, in which the computational domain, absorbing boundary conditions (ABC), perfect electric conductors (PEC), perfect magnetic conductors (PMC), Pade via Lanczos (PVL) parameters, excitation sources and output parameters can be defined using the following rules:

1. The first row in ROEMS.IN should be either blank or a short comment statement that should be less than 100 characters.

2. From the second row onward, the first column must be blank. The NAMELIST commands start from second column with the format

   \$namelist name
   variable 1=..., variable 2=..., $end.

   The blank row will be neglected.

3. The row that begins with star * at the first column will be neglected. This comment line can be used for the explanation of the structure defined by the commands.

4. Either an end or END statement must begin from first column, i.e. no blank at the first column, to finish the last row of the ROMES.IN file.

   There are twelve NAMELIST commands in ROMES.IN, which are read by R1.F.

   They are included in the dissertation as APPENDIX B.
After running file R4.F, a set of output data files will be generated, which are stored as R.OUT.1, R.OUT.2, ..., until R.OUT.9999. These files, which contain the calculated electric field values, magnetic field values, voltage values, or current values, correspond to the probe points in ROMES.IN. They can be used to calculate the characteristics, i.e., scattering parameters, of electromagnetic systems with discontinuities. In the next section, we show several examples.

4.2 Examples of Reduced-Order Modeling

A parallel-plate waveguide with a dielectric discontinuity has been analyzed using the PVL, Krylov, and rational Krylov algorithms. As shown in Fig. 4.2, the dimension of dielectric post is 8mm×6mm×3.2mm along x, y, and z direction respectively. The cross section dimension of the parallel-plate waveguide is 14mm×6mm along x and y direction respectively. PEC walls are placed at \( y = 0 \) and \( y = 6mm \) while PMC walls are placed at \( x = 0 \) and \( x = 8mm \). The distance between the dielectric post and current excitation plane is 6.4mm. The distance between the post and output plane is 3.2mm. The permittivity of dielectric post is 8.2. A current dipole has been used to excite the waveguide.
Fig. 4.2 A parallel-plate waveguide with dielectric discontinuity.

The electrical field amplitudes and phases have been calculated at the excitation plane using PVL with expansion at 1MHz and 25GHz, and are shown in Fig. 4.3, Fig. 4.4, Fig. 4.5, and Fig. 4.6, respectively.

Fig. 4.3 Electrical field amplitudes in the excitation plane calculated using the PVL algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.
Fig. 4.4 Electrical field phases in the excitation plane calculated using the PVL algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.

Fig. 4.5 Electrical field amplitudes in the excitation plane calculated using the PVL algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.
Fig. 4.6 Electrical field phases in the excitation plane calculated using the PVL algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.

The solid lines in Fig. 4.3, Fig. 4.4, Fig. 4.5, or Fig. 4.6 are calculated using PVL with frequency segmentation, where the 1MHz to 25GHz frequency band is segmented into 5 sub bands, which are 1MHz to 10MHz, 10MHz to 100MHz, 100MHz to 1GHz, 1GHz to 10GHz, and 10GHz to 25GHz. The expansion points are 5MHz, 50MHz, 500MHz, 5GHz, and 20GHz for the five sub bands, respectively. The order of the original system resulting from the semi-discrete electromagnetic model is 4697. The order of the reduced-order PVL model for each sub band is 50. The results from this frequency segmentation model are taken as the baseline for comparison with other models developed using the Krylov and rational Krylov algorithms in the following figures. From Fig. 4.3 and Fig. 4.4, we can see that as the order of the PVL model increases from 20 to 50, where 1MHz is used as the expansion point, the calculated field
amplitudes and phases slowly converge to the benchmark values. For higher frequencies, the differences are large. This is due to the dependence of the accuracy on the expansion frequency value. We know that for higher frequencies, there are more eigenvalues that dominate the characteristics of the electrical field amplitude and phase curves. When using a low frequency value as the expansion frequency, such as 1MHz, more errors are expected at high frequencies due to the expansion point being far away from these high frequencies.

The obvious goal here is to find an expansion point which allows the poles generated by PVL algorithm to converge quickly to those poles near the frequency range of interest. Unfortunately, finding such an optimal expansion point is not practical, since it would require a knowledge of all poles of the transfer function $H$ itself. However, a simple heuristic for the choice of expansion point was developed by Peter Feldmann [5] based on the assumption that the system is stable, i.e., all its poles have negative real parts. The frequency range of interest can be written as $f_{\text{min}} \leq f \leq f_{\text{max}}$, where $f_{\text{min}} < f_{\text{max}}$ and $f_{\text{max}} > 0$, i.e., one is interested in the approximation $H_y(s)$ to $H(s)$ for $s = 2\pi jf$. Here $j = \sqrt{-1}$. Roughly speaking, the poles that determine the convergence behavior are the ones closest to the endpoints $2\pi jf_{\text{max}}$ and $2\pi jf_{\text{min}}$, and therefore, the expansion point $s_0$ should be chosen such that these "extremal" points converge as fast as possible. The resulting expansion point is as follow:

$$s_0 = (f_{\text{max}} - f_{\text{min}})\pi + (f_{\text{max}} + f_{\text{min}})\pi j.$$

(4.1)
Note that $s_0$ is a point in the right-half plane, and that its imaginary part is just the midpoint of the complex interval $[2\pi j f_{\min}, 2\pi j f_{\max}]$.

Often, the frequency range of interest is of the form $0 \leq f \leq f_{\text{max}}$. In view of the fact that $H(-2\pi j f) = H(2\pi j f)$ and $H_q(-2\pi j f) = H_q(2\pi j f)$, this case is equivalent to the frequency range $-f_{\text{max}} \leq f \leq f_{\text{max}}$, i.e., we can formally set $f_{\text{min}} = -f_{\text{max}}$. Thus, from (4.1), we obtain the expansion point

$$s_0 = 2\pi f_{\text{max}},$$

which is recommended for use when the frequency range is of the form $0 \leq f \leq f_{\text{max}}$.

Finally, although it was stated in [5] that the PVL algorithm is fairly insensitive to the choice of the expansion point $s_0$, as long as it is chosen as a point in the right-half plane whose distance to the imaginary axis is of the same order as the length of the frequency range of interest, we still found from many numerical examples that good convergence is usually found locally, but for frequencies far away from $s_0$, slow convergence can easily result. This can result in the missing of steep pulses or the introduction of unreal pulses to the transfer function curve. For our parallel-plate waveguide problem, the transfer function is equivalent to the electric field amplitude and phase spectra. Because the dynamics of the original system are rarely known a priori, the expansion point placement and selection are the other important factors that determine the accuracy and dimension of the reduced-order model as well as the complexity of the original system.
Comparing Figs. 4.3 and 4.4 with Figs. 4.5 and 4.6, we can see the importance of the expansion point placement and selection. When 25GHz is used as expansion point, the calculated field amplitude and phase values converge much faster to the benchmark values as the order of PVL model increases from 20 to 50 than when using 1MHz as the expansion point. The differences in the various models orders are still due to the dependence of accuracy to the expansion point selection. This time, however better convergence is ensured according to (4.2) because the expansion point is located at a high frequency. Similar results obtained using the Krylov algorithm are shown in Figs. 4.7 to 4.10, where 1MHz and 25GHz are also used as the expansion points, respectively. The rational Krylov results are shown in Fig. 4.11 and Fig. 4.12, where expansion points are chosen as 1MHz, 5GHz, and 25GHz.

![Graph showing electrical field amplitudes](image)

Fig. 4.7 Electrical field amplitudes in the excitation plane calculated using the Krylov algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.
Fig. 4.8 Electrical field phases in the excitation plane calculated using the Krylov algorithm with expansion at 1MHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.

Fig. 4.9 Electrical field amplitudes in the excitation plane calculated using the Krylov algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.
Fig. 4.10 Electrical field phases in the excitation plane calculated using the Krylov algorithm with expansion at 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.

Fig. 4.11 The amplitudes of E field in the excitation plane calculated using the rational Krylov algorithm with expansion at 1MHz, 5GHz, 25GHz. A current dipole was used to excite a parallel-plate waveguide with a dielectric discontinuity.
From these plots, we can see that the Lanczos algorithm converges slowly compared with the Krylov and rational Krylov methods. However the PVL algorithm is faster than the Krylov method. The computation times are about 11 sec for PVL, 32 sec for Krylov, and 96 sec for the rational Krylov algorithm on our SGI computer. The computation time for the rational Krylov algorithm is much higher than the others because of the required matrix factorizations at each of the three expansion points. The Lanczos approach is robust, especially when using the look-ahead technique [56] in the PVL process. The rational Krylov algorithm is the least robust method out of the above three approaches. However, it gives the best convergence and accuracy since it projects the original system to the subspace spanned over multiple expansion points. The order of the macromodel and expansion frequency for each expansion point has a pronounced effect on the performance of the rational Krylov method. The expansion points and order
of the macromodel have to be selected carefully. Otherwise, numerical difficulties occur during the process of creating the \( W \) and \( V \) projectors, i.e., the norm of \( W_i \) or \( V_i \) overflows. This limits the application of the rational Krylov algorithm to higher order linear systems resulting from electromagnetic problems.

![Current Sheet Diagram](image)

**Fig. 4.13** A parallel-plate waveguide with a dielectric discontinuity.

In order to extract the scattering parameters for another parallel-plate waveguide discontinuity. We employed a uniform current sheet excitation instead of the dipole source as shown in Fig. 4.13. The electric field values are calculated at both the excitation plane and the output plane with and without dielectric discontinuity present. We denote the electric field values as \( E_0 \) for no dielectric discontinuity, \( E_{\text{input}} \) for the electric field in the excitation plane, and \( E_{\text{output}} \) for the electric field in the output plane.

Besides a phase shift, the electric field values are the same in input and output planes when no discontinuity is employed. This was verified by looking at the numerical calculation results. The scattering parameters can be calculated from these field values using
Here, the field values at the center of the waveguide along x direction are used. The calculated amplitudes of scattering parameter are shown in Fig. 4.14 and Fig. 4.15.

![Scattering parameters for the parallel-plate waveguide with discontinuity](image)

Fig. 4.14 Scattering parameters for the parallel-plate waveguide with discontinuity.
For Fig. 4.14, two PMC boundaries are used to truncate the waveguide in the transverse (i.e., x) direction. One is located at the center of the waveguide and one is at the edge. PML boundaries are employed in the longitudinal direction (i.e., z). The other two surfaces are PEC. This yields a lossless system with a dimension of 1829. PVL with frequency segmentation is used to calculate the electric field values. The 10 MHz to 20 GHz frequency band is segmented to 5 sub bands that are 10 MHz to 60 MHz, 60 MHz to 400 MHz, 400 MHz to 2 GHz, 2 GHz to 8 GHz, and 8 GHz to 20 GHz. The expansion frequencies for the 5 sub bands are 30 MHz, 200 MHz, 1 GHz, 6 GHz, and 16 GHz. The order of the reduced-order model is 50. The input files M3.IN and ROMES.IN are:

```
Input file M3.IN for parallel-plate waveguide discontinuity
$def_box
nx1=1 ny1=1 nz1=1 nx2=7 ny2=3 nz2=44
$end
$def_block
```
$end
$cube
xl=1.0 x2=3.0 yl=0. y2=2.0 zl=40 z2=52 scale=1e-3
er=8.2
mate='media'
$end
end

Input file ROMES.IN for parallel-plate waveguide discontinuity
$def_box
nx1=1 nx2=7 ny1=1 ny2=3 nz1=1 nz2=44
$end
$def_pml
nz1=4 nz2=4
$end
$pmc
nx1=1 nx2=2 ny1=1 ny2=3 nz1=1 nz2=44
$end
$pmc
nx1=6 nx2=7 ny1=1 ny2=3 nz1=1 nz2=44
$end
$pec
nx1=1 nx2=7 ny1=1 ny2=1 nz1=1 nz2=44
$end
$pec
nx1=1 nx2=7 ny1=3 ny2=3 nz1=1 nz2=44
$end
$mate
nx1=2 nx2=4 ny1=1 ny2=3 nz1=21 nz2=27
er=8.2
$end
$pvl
nexp=5 npoint=300 npvl=50
fs(1)=1e7 fs(2)=6e7 fs(3)=4e8 fs(4)=2e9 fs(5)=8e9
fn(1)=6e7 fn(2)=4e8 fn(3)=2e9 fn(4)=8e9 fn(5)=20e9
fe(1)=3e6 fe(2)=2e8 fe(3)=1e9 fe(4)=6e9 fe(5)=16e9
$end
$source
nx1=2 nx2=6 ny1=1 ny2=3 nz1=8 nz2=8
jx=0. jy=1 jz=0.
$end
$probe_e
nx1=2 nx2=2 ny1=2 ny2=3 nz1=8 nz2=8
$end
$probe_e
nx1=2 nx2=2 ny1=2 ny2=3 nz1=40 nz2=40
$end
end
For Fig. 4.15, PML boundaries are used for all the x, y, and z directions. This results a lossy system with a much higher dimension, which is 46538. The same frequency segmentation scheme is employed. The order of the reduced-order model is 500. The input files M3.IN and ROMES.IN are:

Input file M3.IN for parallel-plate waveguide discontinuity.
$def_box
nxl=1 ny1=1 nz1=1 nx2=13 ny2=13 nz2=44
$end
$def_block
xl=0. yl=0. zl=0. x2=12.0 y2=12.0 z2=86 scale=1.e-3
nx1=1 ny1=1 nz1=1 nx2=13 ny2=13 nz2=44
$end
$cube
x1=1.0 x2=3.0 y1=5.0 y2=7.0 z1=40 z2=52 scale=1e-3
er=1.00
mate='media'
$end
end

Input file ROMES.IN for parallel-plate waveguide discontinuity.
$def_box
nx1=1 nx2=13 ny1=1 ny2=13 nz1=1 nz2=44
$end
$def_pml
nx2=4 ny1=4 ny2=4 nz1=4 nz2=4
$end
$pmc
nx1=1 nx2=2 ny1=1 ny2=13 nz1=1 nz2=44
$end
$pec
nx1=1 nx2=6 ny1=6 ny2=6 nz1=1 nz2=44
$end
$pec
nx1=1 nx2=6 ny1=8 ny2=8 nz1=1 nz2=44
$end
$mate
nxl=2 nx2=4 ny1=6 ny2=8 nz1=21 nz2=27
er=1.00
$end
$pv1
nexp=5 npoint=100 npvl=500
fs(1)=1e7 fs(2)=6e7 fs(3)=4e8 fs(4)=2e9 fs(5)=8e9
fn(1)=6e7 fn(2)=4e8 fn(3)=2e9 fn(4)=8e9 fn(5)=20e9
fe(1)=3e7 fe(2)=2e8 fe(3)=1e9 fe(4)=6e9 fe(5)=16e9
$end
$source
nx1=2 nx2=6 ny1=6 ny2=8 nz1=8 nz2=8
jx=0. jy=1 jz=0.
At low frequencies, the calculated scattering parameters using the two approaches agree with each other very well. However, at high frequencies, the effects of fringe fields are apparent in the S parameters of the system. The price for the more accurate results is much more computer processing time. The computation time for each field calculation run is about 90 minutes. This is due to the costly factorization process of the large-scale matrix. Here, the dimension of the matrix is 46538. That is why the PVL technique with expansion at infinity is employed in ROMES.

PVL with expansion at infinity has been applied to the problem shown in Fig. 4.16. In this example, a WR90 rectangular waveguide has a dielectric post inside it. The post is of dimensions 12mm × 10.16mm × 6mm and has a relative dielectric constant of 8.2. The transverse dimensions of the waveguide are 22.86mm × 10.16mm. The excitation is an electric current sheet source that generates only the dominant TE_{10}
waveguide mode. The current sheet is located a distance from the dielectric post. The
electric field on the excitation plane has been calculated using both PVL with expansion
at 10GHz and PVL with expansion at infinity. The electric field on the excitation plane
provides information about the reflected field. The original electromagnetic system gives
a linear system of order 4697, which is the number of unknowns. The input files M3.IN
and ROMES.IN for this problem are:

Input file M3.IN for WR90 rectangular waveguide.
$def_box
nxl=1 nyl=1 nzl=1 nx2=9 ny2=4 nz2=44
$end
$def_block
x1=0. y1=0. z1=0. x2=5.43 y2=10.16 z2=46. scale=1.e-3
nxl=1 nyl=1 nzl=1 nx2=4 ny2=4 nz2=24
$end
$def_block
x1=0. y1=0. z1=46. x2=5.43 y2=10.16 z2=52. scale=1.e-3
nxl=1 nyl=1 nzl=24 nx2=4 ny2=4 nz2=30
$end
$def_block
x1=0. y1=0. z1=52. x2=5.43 y2=10.16 z2=80. scale=1.e-3
nxl=1 nyl=1 nzl=30 nx2=4 ny2=4 nz2=44
$end
$def_block
x1=5.43 y1=0. z1=0. x2=12.096667 y2=10.16 z2=46. scale=1.e-3
nxl=4 nyl=1 nzl=1 nx2=9 ny2=4 nz2=24
$end
$def_block
x1=5.43 y1=0. z1=46. x2=12.096667 y2=10.16 z2=52. scale=1.e-3
nxl=4 nyl=1 nzl=24 nx2=9 ny2=4 nz2=30
$end
$def_block
x1=5.43 y1=0. z1=52. x2=12.096667 y2=10.16 z2=80. scale=1.e-3
nxl=4 nyl=1 nzl=30 nx2=9 ny2=4 nz2=44
$end
$cube
x1=5.43 x2=12.096667 y1=0. y2=10.16 z1=46. z2=52. scale=1.e-3
er=8.2
mate='media'
$end
end

Input file ROMES.IN for WR90 rectangular waveguide.
$def_box
nxl=1 nx2=9 nyl=1 ny2=4 nzl=1 nz2=44
$end
$def_pml
nz1=4 nz2=4
$end
$pmc
nx1=8 nx2=9 ny1=1 ny2=4 nz1=1 nz2=44
$end
$mate
nx1=4 nx2=9 ny1=1 ny2=4 nz1=24 nz2=30
er=8.2
$end
$pvl
nexp=1 npoint=100 npv1=50
fs(1)=8e9
fn(1)=12e9
fe(1)=10e9
$end
$source
nx1=2 nx2=2 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.2461866 jz=0.
$end
$source
nx1=3 nx2=3 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.4772191 jz=0.
$end
$source
nx1=4 nx2=4 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.6788764 jz=0.
$end
$source
nx1=5 nx2=5 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.8013018 jz=0.
$end
$source
nx1=6 nx2=6 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.8968980 jz=0.
$end
$source
nx1=7 nx2=7 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.9624646 jz=0.
$end
$source
nx1=8 nx2=8 ny1=1 ny2=4 nz1=8 nz2=8
jx=0. jy=.9958060 jz=0.
$end
$probe_e
nx1=8 nx2=8 ny1=2 ny2=3 nz1=8 nz2=8
$end
end
Fig. 4.17 Amplitude of the electric field on the excitation plane calculated using the PVL algorithms with expansions at a finite value and infinity.

Fig. 4.17 shows the magnitude of the E field plotted as a function of frequency for models of various orders. Fig. 4.18 shows the corresponding phase of the E field. The PVL method with finite value expansion yields results that agree with measurements [9] and will be used to test the PVL with expansion at infinity algorithm. We can see that as the order of the PVL expansion at infinity technique reaches 400, the results of both methods agree with each other very well, e.g. curves 1 and 5.
Using the electric fields that are calculated with the discontinuity (Figs. 4.17 and 4.18) and without the discontinuity, we can compute the return loss $|S_{11}|$ using (4.3), see Fig. 4.19. These results further prove the effectiveness of PVL with expansion at infinity.

The computational requirements of PVL with expansion at infinity are much less than those for the expansion at 10 GHz. This is especially true when the problem dimension gets higher. The memory required for PVL with finite expansion frequency is 58 Mbytes. The running time on a SGI machine is about 8 seconds for 50 finite PVL steps. Only 1.2 Mbytes are required for PVL with expansion at infinity and the running time is about 4 seconds on the same computer.
Fig. 4.19 Return loss from a dielectric post inside WR90 waveguide.

Fig. 4.20 Visualization of the biorthogonality of matrices V and W.
The biorthogonality of matrix $V$ and $W$ has also been explored. Fig. 4.20 shows the value of $|V-W|$ plotted on a logarithmic scale. We can see that as $i$ or $j$ approaches 400, the non-diagonal elements are no longer negligible compared to the diagonal elements. Ideally, we expect $V-W=\Lambda$, where $\Lambda$ is a diagonal matrix. The efficiency and accuracy of this method are limited by the loss of biorthogonality of $V$ and $W$.

Fig. 4.21 shows the electric field over a wider frequency band. Line 1 has been calculated using PVL of 100 steps with multiple point expansions at 6GHz, 10GHz and 15GHz. This provides better accuracy than is obtained with a single point expansion. Line 2 has been calculated using PVL of 400 steps with expansion at infinity. Line 3 has been calculated using PVL of 50 steps with expansion at 10GHz. From Fig. 4.21, we can see that expansion at infinity is more accurate than expansion at a single finite value for high frequencies far from the expansion frequency (10GHz).
Fig. 4.21 Amplitude of the electric field on the excitation plane calculated using the PVL algorithm with expansion at a single frequency point, multiple points and infinity.

Fig. 4.22 A probe-fed patch antenna (a=41.275mm, h=3.175mm, s=14.224mm, ε_r=2.5).

The third example deals with the modeling of the microstrip patch antenna shown in Fig. 4.22. The patch is placed on a conductor-backed dielectric substrate of thickness 3.175mm and relative dielectric constant 2.5. The size of the patch is a=41.275 mm. It is
probe-fed at a point along one of the two planes of symmetry of the patch, at a distance of 14.224 mm from the edge. The input files M3.IN and ROMES.IN are:

Input file M3.IN for patch antenna shown in Fig. 4.22.
$def_box
nxl=1 ny1=1 nz1=1 nx2=16 ny2=9 nz2=30
$end
$def_block
xl=0. yl=0. zl=0. x2=36.419118 y2=8.4666667 z2=28.448J
scale=1.e-3
nxl=1 ny1=1 nz1=1 nx2=16 ny2=9 nz2=13
$end
$def_block
xl=0. yl=0. zl=28.448 x2=36.419118 y2=8.4666667 z2=70.254091J
scale=1.e-3
nxl=1 ny1=1 nz1=13 nx2=16 ny2=9 nz2=30
$end
rectangle
xl=0. x2=21.851471 yl=3.175 y2=3.175 zl=14.224 z2=55.449J
scale=1.e-3
mate='pec'
$end
cube
xl=0. x2=36.419118 yl=0. y2=3.175 zl=0. z2=70.254091 scale=1.e-3
er=2.5
mate='media'
$end
end

Input file ROMES.IN for patch antenna shown in Fig. 4.22.
$def_box
nxl=1 nx2=16 ny1=1 ny2=9 nz1=1 nz2=30
$end
$def_pml
nzl=4 nz2=4
nx2=4 ny2=4
$end
pec
nxl=1 nx2=10 ny1=4 ny2=4 nz1=7 nz2=24
$end
pec
nxl=2 nx2=2 ny1=1 ny2=2 nz1=13 nz2=13
$end
pec
nxl=2 nx2=2 ny1=3 ny2=4 nz1=13 nz2=13
$end
pmc
nxl=1 nx2=2 ny1=1 ny2=9 nz1=1 nz2=30
$end
mate
nxl=1 nx2=16 ny1=1 ny2=4 nz1=1 nz2=30 er=2.5
The source is implemented in the finite difference grid through a perfectly conducting wire that is placed at the position of the feeding-probe and connects the patch to the ground plane. One segment of this wire, of length equal to one finite-difference cell, is removed and excited by a filamentary current. The electric field at this segment is used to calculate an effective input voltage for the antenna (Figs. 4.23 and 4.24). Likewise, the magnetic field is used to compute an effective input current for the antenna (Figs. 4.25 and Fig. 4.26). A four-layer PML is used to truncate the computational domain. It is designed for a reflection coefficient of $10^{-5}$ at normal incidence, and it is placed one cell above the patch and two cells away from its edges. After finite difference processing, a linear system with dimension of 25012 results. Both PVL at a finite frequency of 5GHz with 150 PVL steps and PVL at infinity with 1000 PVL steps have been used to calculate the input voltage and current. The amplitude and phase of the calculated input voltage are shown in Fig. 4.23 and Fig. 4.24, respectively. We can see that the results agree with each other very well.
Fig. 4.23 Amplitude of the input voltage for the probe-fed patch antenna.

Fig. 4.24 Phase of the input voltage for the probe-fed patch antenna.
Similarly, the input current has been calculated. The amplitude and phase of the calculated input current are shown in Fig. 4.25 and Fig. 4.26. We can see that the calculated currents also agree with each other very well.

Fig. 4.25 Amplitude of the input current for the probe-fed patch antenna.
From the input voltage and input current, we can calculate the input impedance of the antenna. If we take the real part of the input impedance at the first resonant point as a reference impedance $Z_0 = \text{Re}(Z_{in}(f_0))$ and use

$$S_{11} = \frac{Z_{in} - Z_0}{Z_{in} + Z_0},$$

(4.4)

then the magnitude of $S_{11}$ can be calculated and is shown in Fig. 4.27. The result obtained using PVL with expansion at 5 GHz has been compared to a Finite Difference Time Domain (FDTD) results obtained through the Fourier transform of the transient response of the antenna excited by a short Gaussian pulse [61]. Good agreement was obtained.
Fig. 4.27 Magnitude of the reflection coefficient of the patch antenna in Fig. 4.22.

The memory requirement of PVL with expansion at 5GHz is about 323 Mbytes, while PVL at infinity requires only 3.6 Mbytes. The computation time for PVL at 5GHz is about 624 seconds on our SGI computer. Only 120 seconds were needed for PVL at infinity on the same computer. From these results, we can easily see the advantage of PVL with expansion at infinity as compared with PVL with expansion at finite frequency on memory and speed.
5. A TIME DOMAIN CONVOLUTION METHOD

In the previous chapters, we have discussed a reduced-order modeling methodology for electromagnetic systems. Scattering parameters were calculated for several examples. For IC design and simulation, the time domain performances, i.e., signal integrity performances, are really important for designers. Also, circuits containing nonlinear devices or time-dependent characteristics are better characterized in the time domain. Therefore, in this chapter we present an efficient time-domain methodology.

5.1 Introduction to Time Domain Simulation

We can get scattering parameters for an electromagnetic system by using the reduced-order modeling methodology that has been discussed up until now. However, this is only half of the task. The second half is to carry out the time domain simulations using the calculated scattering parameters. For example, transmission lines with loss, dispersion, or discontinuities are best characterized in frequency domain. While on the other hand, time domain simulations are needed.

The frequency-domain telegrapher's equations are the most suitable for us to start. For a transmission line system with \( N \) conductors plus one ground conductor, Telegrapher's equations are written as:

\[
\frac{dV(x, \omega)}{dx} = -R(\omega)I(x, \omega) - j\omega L(\omega)I(x, \omega),
\]

\[
\frac{dI(x, \omega)}{dx} = -G(\omega)V(x, \omega) - j\omega C(\omega)V(x, \omega).
\]
In these frequency-domain equations the $N \times N$ matrices $R$, $L$, $C$, and $G$ contain the per unit length transmission line parameters. The entries of the $N$ dimensional vectors $V$ and $I$ are the voltages and currents on the line. The space coordinate $x$ varies from 0 up to the line length $d$ and the angular frequency $\omega$ determines the sinusoidal component for which the equation is evaluated.

The calculation of these transmission line parameters is out of scope of this dissertation. The moment method [63], the coupled circuit method, or the surface integral equation method [64] can be used to directly calculate the $R$, $L$, $G$, and $C$ parameters. Pre-calculated scattering parameter or scattering parameter measurements can also be used to extract the transmission line parameters [11-15, 65]. Scattering parameters can also be directly used for the transient analysis of lossy, nonlinearly terminated transmission line structures in high-speed microelectronic circuits [66].

As we discussed in the first chapter, five types of approaches have been proposed in previous works for transmission line embedded circuit time domain simulations. The drawback of the network of lumped elements method [16, 17] is that the amount of computation increases for the simulation because a large number of nodes and elements are introduced.

For the second type of approach, i.e., the convolution technique, one difficulty lies in how to determine the impulse responses of an arbitrary multiconductor line system. Different methods [18]-[23] have been used to determine the impulse responses. Secondly, the convolution simulations suffer from a common drawback: the convolution operation needs to extend over the entire past time history. The computation time
required at any time point $t$ is then proportional to $t$, therefore, the convolutions at large
time points will be very time-consuming. Furthermore, the inverse Fourier
transformations that are used to calculate the impulse responses will suffer from either
the aliasing effect or the fact that too many frequency points are needed for the
transformation to avoid the aliasing effect.

To avoid the time-consuming convolution integrations, the state-based approach
[24] and the waveform relaxation based approach [25] have been proposed. The state
based approach utilizes information about the internal states of a transmission line at a
given time to solve the states for the next time point. The voltage and the current at the
sample points are kept as the states of the line. The voltages and the currents are assumed
to be piecewise-linear between adjacent sample points. Based on this assumption, the
state variables can be determined by using integrations on space, hence avoiding
convolutions. The samples will be chosen densely in the regions where waveforms are
rapidly-varying. The waveform relaxation based approach solves the transmission line’s
equations in the frequency domain and uses the FFT to transform the results back and
forth between the two domains for each iteration. Hence, time-domain convolutions are
avoided by performing multiplications in the frequency domain. Again, this type of
approach is not suitable for handling rapidly-varying signals. For a simulation containing
a 100 ps rise-time pulse and having a simulation time of 100ns, each FFT needs to
process around 100000 data points in order to avoid aliasing effects. In summery, in
order to solve simulations involving rapidly-varying signals we need to focus our effort
on developing fast convolution integration algorithms.
Finally, reduced-order modeling techniques, including Asymptotic Waveform Evaluation (AWE) type approaches [3, 26], Pade approximation [27] and passive reduced-order interconnect macromodeling algorithm (PRIMA) [28, 29] etc. have been used to solve transmission line problems. Macromodels have been developed for transmission lines. Some macromodels are even SPICE compatible so that the time domain simulation functionality of SPICE can be used for circuit simulations. These types of approaches are usually very efficient. However, they sometimes suffer from numerical difficulties, such as ill-conditioned matrix multiplications, and recursive iteration breakdown.

A fast simulation method was proposed for single and coupled transmission lines that are connected to linear and non-linear circuit elements in [30]. This method, which belongs to the second type as summarized above, is based on a triangle impulse response (TIR) database of lossy transmission lines that are matched at both the near and far ends. In next section, more details will be discussed about this method.

5.2 A Fast Simulation Method-Fastline

There are many transmission lines in the package of a mainframe computer system [67]. As the system clock frequency and integration density increase, the frequency-dependent loss of transmission lines in IC packages can no longer be neglected. In mainframe computer package design and other high-speed IC package design applications, we have to simulate systems with a large number (for example, 100k) of single and coupled lossy transmission lines. In addition to accuracy, simulation time is another important criterion for a tool. The existing models for lossy transmission
lines are usually too slow to be used directly in such a simulation, especially when frequency-dependent properties such as skin-effect are taken into account. Fast simulation methodology becomes one of the key issues in accurate design of high-speed IC packages in mainframe computers.

The fast simulation method developed in [30] for circuits having single and coupled transmission lines uses triangle waveforms as time domain basis function to represent any waveform as shown in Fig. 5.1 and Fig. 5.2. The time domain response can be calculated by superposition of triangle impulse responses.

![Fig. 5.1 Input triangle impulse signal.](image)

![Fig. 5.2 A waveform composed by triangle basis waveforms.](image)
A N-coupled transmission line section and its equivalent circuit [68] for this method are illustrated in Fig. 5.3 and Fig. 5.4. The equations for calculating $i_j(t)$ and $w_j(t)$ ($j = 1, 2, \cdots, 2N$) in Fig. 5.4 are given as

$$i_j(t) = \sum_{n=1}^{\infty} (w_j(t-n\Delta t) - v_j(t-n\Delta t))i_{n\eta}((n+1)\Delta t),$$

$$j = 1, 2, \cdots, 2N$$

(5.2)
where the current source \( i_j(t) \) is caused by the imaginary part of \( Z_{0j} \), the characteristic impedance of transmission line \( j \), \( R_{0j} \) is the real part of \( Z_{0j} \), and \( i_{irj}(t) \) and \( v_{irmj}(t) \) are triangle impulse responses. For single and two coupled lines, the triangle impulse responses are defined in Fig. 5.5, where long line loads are used to eliminate the reflections at the near and far ends of the line under test. A typical value for the simulation time step \( \Delta t \) is 10 ps.

A general purpose simulation tool called Fastline, which is based on this method, has been developed in IBM. Lumped elements such as resisters, inductors, capacitors, and voltage and current sources are implemented as well as transmission lines. During
simulations, the whole circuit is divided into many local circuits separated by transmission lines which are solved individually at every time step [68].

We can see that the key feature of Fastline is the introduction of triangle impulse responses. Once the triangle impulse responses are generated and stored in a database, they can be repeatedly used to perform large numbers of time domain transmission line simulations for the design of a system like a mainframe computer. Usually other simulation tools such as IBM AS/X are used to perform the TIR calculations. More detailed TIR calculation results and applications with Fastline for transmission line embedded circuit simulations can be found in [30].
6. CLOSED-FORM TRIANGLE IMPULSE RESPONSES

A fast simulation method was introduced for circuits having single and coupled
transmission lines in the last chapter. This method is based on a triangle impulse
response database of lossy transmission lines that are matched at both the near and far
ends. Triangle waveforms are used as time-domain basis functions to represent the
waveforms. The time-domain response can then be calculated by superposition of the
triangle impulse responses. Other simulation tools are used to calculate the lossy
transmission line triangle impulse responses numerically. The numerical results of the
triangle impulse responses are then used to carry out the transient circuit simulations.

In this chapter, analytical frequency-domain expressions for single and coupled
transmission lines, which are matched at both ends and excited with a triangle impulse
input, are first developed using the modal analysis technique [37]. Here, the line
parameters (i.e., resistance $R$, inductance $L$, conductance $G$, and capacitance $C$) are
assumed to be constants with frequency. A set of compact formulas is obtained to
describe the voltage and current waveforms on the active as well as the disturbed line in
the frequency domain. The inverse Fourier transform is then used to obtain an expression
for the time-domain triangle impulse responses. The integral associated with the inverse
Fourier transform is solved analytically using a differential-equation-based technique
[69]. Closed-form expressions for the triangle impulse responses are given in the form of
incomplete Lipschitz-Hankel integrals (ILHIs) of the first kind [33]. The ILHIs can be
efficiently calculated using algorithms developed in [39]. The combination of these
closed-form expressions for the triangle impulse responses with the method discussed in
last chapter provides an accurate and efficient simulation method for transmission line embedded circuit simulations.

This chapter is organized as follows. In section 6.1, we give the frequency domain formulation based on modal analysis theory [37, 38]. The closed-form inverse Fourier transformations for the triangle impulse responses of single and coupled transmission lines are carried out in section 6.2. In section 6.3, we present numerical examples using Fastline [30] and the closed-form TIR results along with comparisons with frequency dependent line parameter simulation results. We will also give some conclusions in section 6.3.

6.1 Frequency Domain Formulation

Before working on the analytical inverse Fourier transformation, we recall the frequency domain formulation based on spectral domain modal analysis [37, 38]. Given a $N$-line system, as shown in Fig. 6.1, let $[R]$, $[L]$, $[G]$, and $[C]$ be the resistance, inductance, conductance, and capacitance matrices. Also let $E_n(t)$, $Z_{cn}$ and $Z_{ln}$ be the input signal, the source and load impedances of the $n$th ($n=1,2,\cdots,N$) line, and $l$ be the length of the coupled lines. Next we define the series impedance matrix and parallel admittance matrix as

$$[Z] = [R] + j\omega [C],$$  \hspace{1cm} (6.1)  

$$[Y] = [G] + j\omega [C].$$  \hspace{1cm} (6.2)
In this derivation, we will assume the $[R]$, $[L]$, $[G]$, and $[C]$ are independent of frequency.

![Diagram]

**Fig. 6.1** The schematic of a system containing $N$ coupled interconnects

As is well known [37], there exist $N$ propagation modes in a system of $N$ signal conductors. The $N$ complex modal propagation constants $\gamma_n$ and the voltage eigenvector matrix $[S_v]$ are solutions of the eigenvalue equation

$$ (\gamma^2 [I] + [Z][Y])[S_v] = 0, \quad (6.3) $$

where $[I]$ is the identity matrix. The current eigenvector matrix $[S_i]$ is then given as

$$ [S_i] = [Z]^{-1}[S_v][\Gamma], \quad (6.4) $$

where $[\Gamma] = \text{diag}(\gamma_1, \gamma_2, \ldots, \gamma_n)$. The characteristic impedance matrix is defined to be $[Z_c] = [S_v][S_i]^{-1}$ and the characteristic admittance matrix $[Y_c]$ is the inverse of $[Z_c]$. The line voltage and current vectors can be represented as sums of the incident and reflected waves in terms of voltage and current eigenvector matrices $[S_v]$ and $[S_i]$, i.e.,
\[ [V(x)] = [S_v]([W^*(x)] + [W^-(x)]), \]
\[ [I(x)] = [Y_c][S_v]([W^*(x)] - [W^-(x)]), \]

where
\[ [W^*(x)] = [W^*(0)\exp(\mp\gamma_n x)]. \]

\( W_n^+ \) and \( W_n^- \) are the amplitudes of the incident and the reflected components, respectively, for the \( n \)th mode. Assuming that the length of the coupled lines is \( l \), then the terminal conditions are
\[ [V(0)] = [E] - [Z_g][I(0)], \]
\[ [V(l)] = [Z_L][I(l)]. \]

where \([E] = [E_n], [Z_g] = \text{diag}(Z_{g1}, Z_{g2}, \ldots Z_{gN})\) and \([Z_L] = \text{diag}(Z_{L1}, Z_{L2}, \ldots Z_{LN})\). \( E_n \) is the input voltage applied to the \( n \)th line, \( Z_{gn} \) is the internal impedance of the \( n \)th generator, and \( Z_{ln} \) is the load impedance of the \( n \)th line. Substituting (6.5) and (6.6) into (6.7), we have
\[ \begin{bmatrix} [S_v] + [Z_g][S_1] & [S_v] - [Z_g][S_1] \\ ([S_v] - [Z_L][S_1])'[\mathbf{P}] & ([S_v] + [Z_L][S_1])[\mathbf{P}]' \end{bmatrix} \begin{bmatrix} [W^*(0)] \\ [W^-(0)] \end{bmatrix} = \begin{bmatrix} [E] \\ [0] \end{bmatrix}, \]

where \([\mathbf{P}] = \text{diag}(\exp(-\gamma_n l)).\) The voltage and current transfer functions can be obtained from (6.5) after (6.8) is solved. The transfer functions are relatively easy to obtain when both the generator and load ends are matched, which is the case for calculating triangle impulse responses.
6.1.1 Single Lossy Line

Notice that if both ends are matched for a single line, then the voltage at the near end should be half the source voltage. The voltage along the line then satisfies

\[ \frac{V(x, \omega)}{E(\omega)} = \frac{1}{2} \exp(-\gamma x), \]
\[ I(x, \omega) = \frac{V(x, \omega)}{Z_0}, \]

where

\[ \gamma = \sqrt{(R + j\omega L)(G + j\omega C)}, \]
\[ Z_0 = \sqrt{(R + j\omega L)/l(G + j\omega C)}. \] (6.10)

6.1.2 Two-Line System

The voltage transfer function for a lossless two-line system has been developed in [38], where the generator and load are not necessarily matched with the transmission line. In order to calculate the triangle impulse response, the generator and load must be matched with the transmission line to avoid reflections from both ends. For a two-line system, it is impossible to ideally match the generator and load to the transmission line with single-valued \( Z_L \) and \( Z_G \). For weak coupling, we can use \( \sqrt{(Z_1^2 + Z_2^2)/2} \) as an approximate matching impedance, which assures zero reflection on the active line and a small reflection on the quiet line, under the conditions of weak coupling. The \([R], [L], [G], \) and \([C]\) matrices for a two-line system are
The propagation constants are defined as:

\[ \gamma_1 = \sqrt{(R_{11} + R_{12} + j\omega(L_{11} + L_{12}))(G_{11} + G_{12} + j\omega(C_{11} - C_{12}))}, \]

\[ \gamma_2 = \sqrt{(R_{11} - R_{12} + j\omega(L_{11} - L_{12}))(G_{11} - G_{12} + j\omega(C_{11} + C_{12}))}, \]

and the impedances \( Z_1 \) and \( Z_2 \) are defined as:

\[ Z_1 = \frac{\sqrt{(R_{11} + R_{12} + j\omega(L_{11} + L_{12})) (G_{11} + G_{12} + j\omega(C_{11} - C_{12}))}}{\sqrt{G_{11} + G_{12} + j\omega(C_{11} + C_{12})}}, \]

\[ Z_2 = \frac{\sqrt{(R_{11} - R_{12} + j\omega(L_{11} - L_{12})) (G_{11} - G_{12} + j\omega(C_{11} + C_{12}))}}{\sqrt{G_{11} + G_{12} + j\omega(C_{11} + C_{12})}}. \]

The characteristic impedance matrix of this two-line system is:

\[ \begin{bmatrix} Z_c \end{bmatrix} = \frac{1}{2} \begin{bmatrix} Z_1 + Z_2 & Z_1 - Z_2 \\ Z_1 - Z_2 & Z_1 + Z_2 \end{bmatrix}. \]

If we let \( Z_g = Z_1 = \sqrt{(Z_1^2 + Z_2^2)}/2 \) and assume there is no reflection from either end, then the far-end voltages on the active and passive lines satisfy:

\[ \frac{V_{12}(l, \omega)}{E_1(\omega)} = \frac{1}{2} \frac{1}{p_1} \exp(-\gamma_1 l) \pm \frac{1}{p_2} \exp(-\gamma_2 l), \]

where \( p_1 = 1/(1 + \frac{Z_g}{Z_1}) \) and \( p_2 = 1/(1 + \frac{Z_g}{Z_2}) \). The "+" sign is for the active line voltage \( V_1(l, \omega) \) and the "-" sign is for the passive line voltage \( V_2(l, \omega) \). The near-end voltage on the passive line satisfies:

\[ \frac{V_{10}(0, \omega)}{E_1(\omega)} = \frac{1}{2} \left( \frac{1}{p_1} - \frac{1}{p_2} \right) \left[ 1 - \frac{1}{p_1} \exp(-2\gamma_1 l) + \frac{1}{p_2} \exp(-2\gamma_2 l) \right]. \]
while the current on the active line will satisfy

\[
\frac{I_i(x, \omega)}{E_i(\omega)} = \frac{1}{2}\left[ \frac{1}{Z_{1p_1}} \exp(-\gamma_1 x) + \frac{1}{Z_{2p_2}} \exp(-\gamma_2 x) \right].
\]  

(6.17)

Before we start the next section, let us look at the voltage and current transfer functions for single-line and two-line systems. We find that the resistance and conductance affect the signal waveforms in two ways. Both the characteristic impedances and propagation constants of the transmission lines are functions of \([R]\) and \([G]\) as well as \([L]\) and \([C]\). Note that small changes in the propagation constants will have a larger effect on the waveform than changes in the characteristic impedances, especially when the line is electrically long.

6.2 Analytic Inverse Fourier Transformation

The triangle impulse response is needed for system level circuit simulation using the technique in [30]. The input signal of the triangle impulse is shown in Fig. 6.2, where the amplitude is chosen as 2V so that the amplitude at the near end of the matched active line is 1V. In the frequency domain

\[
E_i(\omega) = \frac{8\sin^2(\omega\Delta t/2)}{\omega^2\Delta t} \exp(-j\omega\Delta t).
\]  

(6.18)
6.2.1 Single Lossy Line

In the time domain, the voltage and current along the transmission line are expressed as Fourier transforms of the frequency domain results shown in section 6.1.

\[ V(x,t) = \frac{1}{2\pi} \int V(x,\omega) \exp(j\omega t) d\omega, \]  
\[ (6.19) \]

\[ V(x,t) = \frac{1}{2\pi} \int \frac{4\sin^2(\omega\Delta t/2)}{\omega^2\Delta t} \exp(j\omega t - j\omega \Delta t - \gamma x) d\omega. \]  
\[ (6.20) \]

If we define

\[ \phi(t) = \int \frac{1}{\omega^2} \exp(j\omega t - \gamma x) d\omega = \int \frac{\exp(j(\omega t - d\sqrt{\omega^2 - \omega/\tau - \omega_x^2}))}{\omega^2} d\omega, \]

where \( d = x\sqrt{LC} \), \( \tau = \frac{LC}{LG + RC} \), and \( \omega_x = \sqrt{RG/LC} \), then the voltage along the line can be expressed as

\[ V(x,t) = \frac{1}{\pi\Delta t} \left[ \phi(t - \Delta t) - \frac{1}{2} \phi(t) - \frac{1}{2} \phi(t - 2\Delta t) \right]. \]  
\[ (6.21) \]

Before we proceed, let us look at the results in [69]. In that paper, an analytical expression in terms of Incomplete Lipschitz-Hankel Integrals (ILHIs) of the first kind
was developed for \[ \int_{-\infty}^{\infty} \frac{\exp(j(\omega - (z/c)\sqrt{\omega^2 - \omega_p^2}))}{\omega - j\alpha} \, d\omega. \] A differential-equation-based technique was used in the derivation of this integral identity. In a previous paper [70], contour integration techniques were used to develop a similar integral identity for a related application. The ILHIs can be efficiently calculated using algorithms developed in [39]. Two factorial-Neumann series expansions are derived for the ILHI and are used together with the Neumann series expansion in an algorithm that efficiently computes the ILHI \( J_{e_0}(a, z) \) to a user defined number of significant digits. We will use the results from [69] to analytically express \( \phi(t) \) in terms of ILHIs.

In order to directly employ the results in [69], we first define

\[ \phi = \int_{-\infty}^{\infty} \frac{\exp(j(\omega - d\sqrt{\omega^2 - \omega_p^2}))}{\omega - \omega_b} \, d\omega. \]  

The desired result can then be obtained through the limit \( \phi(t) = \lim_{\omega_b \to 0} \frac{\partial \phi}{\partial \omega_b} \). After changing variables,

\[ \phi = e^{-j\frac{t}{2\tau}} \int_{-\infty}^{\infty} \frac{\exp(j(\tilde{\omega} - d\sqrt{\tilde{\omega}^2 - \omega_p^2}))}{\tilde{\omega} - j\alpha} \, d\tilde{\omega}, \quad (6.22) \]

where \( \tilde{\omega} = \omega - \frac{j}{2\tau}, \ \omega_p = \sqrt{\omega_c^2 - \frac{1}{4\tau^2}} \), and \( \alpha = -\frac{1}{2\tau} - j\omega_b \). Using the results from [69], we find that

\[ \phi = e^{-j\frac{t}{2\tau}2\pi ju(t-d)} \left\{ e^{-\omega_c} \cosh(d\sqrt{\alpha^2 + \omega_p^2}) + \frac{1}{2\omega_p\sqrt{t^2 - d^2}} \right\} \]

\[ \left[ (\alpha d - t\sqrt{\alpha^2 + \omega_p^2}) e^{\alpha z} J_{e_0}(a_+, \xi') + (\alpha d + t\sqrt{\alpha^2 + \omega_p^2}) e^{-\alpha z} J_{e_0}(a_-, \xi') \right], \quad (6.23) \]

where
Finally, after taking the limit $a_0 \to 0$, we obtain the desired result

$$a_z = \frac{-\alpha t \pm d \sqrt{\alpha^2 + \omega_p^2}}{\omega_p \sqrt{t^2 - d^2}},$$

$$\zeta = \omega_p \sqrt{t^2 - d^2},$$

$$u(t) = \begin{cases} 
0 & t < 0 \\
1/2 & t = 0 \\
1 & t > 0 
\end{cases}$$

$$\phi(t) = u(t-d) \left\{-2\pi t \cosh(d \omega_c) - \frac{\pi d}{\tau \omega_c} \sinh(d \omega_c) + \pi e^{\frac{i t \tau}{2}} \left[ \left( \frac{-d - t \omega_c}{2\tau} \right) \frac{d' a_{\xi} \zeta}{a_z + 1} \right] J_0(\zeta) + \left( \frac{d \omega_c}{2\tau} \right) + \left( \frac{d}{2\tau} + t \omega_c \right) \frac{d' a_{\xi} \zeta}{a_z + 1} \right] J_1(\zeta) + \left[ \frac{t}{2\omega_c} \right] + \left( \frac{-d}{2\tau} + t \omega_c \right) \frac{d' a_{\xi} \zeta}{a_z + 1} \right] J_0(\zeta) + \left[ \frac{t}{2\omega_c} \right] + \left( \frac{-d}{2\tau} + t \omega_c \right) \frac{d' a_{\xi} \zeta}{a_z + 1} \right] J_1(\zeta) + \right\}.$$ (6.24)

where $a_z'$ is the derivative of $a_z$ with respect to $\alpha$, $a_z' = \frac{-t \mp d/(2\tau \omega_c)}{\omega_p \sqrt{t^2 - d^2}}$, $J_0(\zeta)$ and $J_1(\zeta)$ are first and second order Bessel functions, and $J_0(\alpha_+ \xi) = \int_0^\xi e^{-u} J_0(\xi) d\xi$ is the ILHI of the first kind. The voltage $V(x,t)$ can be calculated from (6.21) if the value for $x$ is given.

Likewise, the current on the transmission line can be calculated from (6.9). The time domain current is

$$i_{tr}(x,t) = \frac{1}{2\pi} \int \frac{4\sin^2(\omega \Delta t/2)}{\omega^3 \Delta t} e^{-j \omega t} e^{-\gamma x} \sqrt{\frac{G + j \omega C}{R + j \omega L}} e^{j \omega t} d\omega.$$ (6.25)
If we define \( \psi(t) = \int \frac{1}{\omega^2} e^{j\omega x} e^{-j\omega t} \sqrt{G + j\omega C} \frac{d\omega}{R + j\omega L} \), then

\[
i_{mr}(x, t) = \frac{1}{\pi \Delta t} \psi(t - \Delta t) - \frac{1}{2\pi \Delta t} \psi(t) - \frac{1}{2\pi \Delta t} \psi(t - 2\Delta t).
\] (6.26)

Following similar procedure, we find that

\[
\psi(t) = u(t - d) \pi \left\{ \frac{2\sqrt{C/L}}{\omega_c} - \frac{G}{\omega_c \sqrt{LC}} + \frac{2Gi}{\omega_c \sqrt{LC}} \right\} \sinh(d \omega_c) + \\
\frac{Gd}{\omega_c \sqrt{LC}} \cosh(d \omega_c) + \frac{Ge^{-j\tau}}{\omega_c \omega_p \sqrt{LC}} \left\{ \left( \frac{d}{2\tau} + t \omega_c \right) \frac{a' a_c \xi}{a_c + 1} \right\} \\
\left\{ J_0(\xi) - \left( \frac{d}{2\tau} + t \omega_c \right) \frac{a' a_c \xi}{a_c + 1} \right\} + \left( \frac{d}{2\tau} + t \omega_c \right) \frac{a' a_c \xi}{a_c + 1} - \left( \frac{d}{2\tau} - t \omega_c \right) a_c \xi \\
\left( \frac{d}{2\tau} + t \omega_c \right) \frac{a' a_c \xi}{a_c + 1} \right\} \\
\frac{e^{-j\tau}}{\sqrt{t^2 - d^2}} \frac{e^{a_c \xi}}{a_c \sqrt{LC}} \left( \frac{d}{2\tau} - t \omega_c \right) \frac{a' a_c \xi}{a_c + 1} + \frac{\sqrt{C/L}}{\omega_c \omega_p \sqrt{LC}} \left( \frac{d}{2\tau} - t \omega_c \right) - \\
\frac{G}{2\pi \omega_c \omega_p \sqrt{LC}} \left( \frac{d}{2\tau} - t \omega_c \right) \frac{e^{\frac{d}{2\tau} - t \omega_c}}{\sqrt{t^2 - d^2}} \frac{e^{a_c \xi}}{a_c \sqrt{LC}} \right\} \\
\right. \}
\] (6.27)

where the notation is the same as in (6.24). The accuracy of these closed-form formulas has been validated in [33] by numerical IFFT calculations.
6.2.2 Two-Line System

The voltages and currents along two coupled transmission lines can be derived by taking an inverse Fourier transformation of (6.15), (6.16), and (6.17). The voltages on the active and passive lines at the far-end are given by

\[
V_{1,2}(l,t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} 4\sin^2(\frac{\omega \Delta t}{2}) \frac{\exp(-j\omega \Delta t)}{\omega^2 \Delta t} \exp(-\gamma_1 l) \pm \exp(-\gamma_2 l) \frac{1}{p_1} \exp(-\gamma_1 l) \pm \frac{1}{p_2} \exp(-\gamma_2 l) \, d\omega. \tag{6.28}
\]

\[
V_{1,2}(l,t) \text{ can be calculated using (6.20) and is given as}
\]

\[
V_{1,2}(l,t) = \frac{1}{p_1} V(l,t)\big|_{\xi_1} \pm \frac{1}{p_2} V(l,t)\big|_{\xi_2}, \tag{6.29}
\]

where \( p_1 = 1/(1 + \frac{Z_0}{Z_1}) \), \( p_2 = 1/(1 + \frac{Z_0}{Z_2}) \), \( Z_1 = \sqrt{(L_{11} + L_{12})/(C_{11} - C_{12})} \), \( Z_2 = \sqrt{(L_{11} - L_{12})/(C_{11} + C_{12})} \), and \( Z_0 = \frac{1}{2} \sqrt{2(Z_1^2 + Z_2^2)} \). When calculating \( V(l,t)\big|_{\xi_1} \), the following parameters are used with (6.20).

\[
R = R_{11} + R_{12}
\]
\[
L = L_{11} + L_{12}
\]
\[
G = G_{11} + G_{12}
\]
\[
C = C_{11} - C_{12}
\]

Likewise, when calculating \( V(l,t)\big|_{\xi_2} \), the following parameters are used with (6.20).

\[
R = R_{11} - R_{12}
\]
\[
L = L_{11} - L_{12}
\]
\[
G = G_{11} - G_{12}
\]
\[
C = C_{11} + C_{12}
\]
Similarly, the near-end crosstalk and current expressions for the two-line system are

\[
V_{c2}(0,t) = \frac{1}{p_1} V(0,t)_{\gamma_1} - \frac{1}{p_2} V(0,t)_{\gamma_2} - \\
\left( \frac{1}{p_1} - \frac{1}{p_2} \right) \left( \frac{1}{p_1} V(2l,t)_{\gamma_1} + \frac{1}{p_2} V(2l,t)_{\gamma_2} \right),
\]

(6.32)

\[
i_1(0,t) = \frac{1}{p_1} i_{TR}(0,t)_{\gamma_1} + \frac{1}{p_2} i_{TR}(0,t)_{\gamma_2}.
\]

(6.33)

In order to obtain (6.33), we have made an approximation to the characteristic impedances when calculating \( p_1 \) and \( p_2 \) (see (6.15)). By ignoring the contributions of resistance and conductance to the characteristic impedance, so that \( p_1 \) and \( p_2 \) were constant with frequency, we were able to analytically evaluate the transient voltage and current. However, we still keep the resistance and conductance terms in the propagation constants since the propagation constants have the greatest effect on the signal waveform.

### 6.3 Examples of Closed-Form TIR Application

In this section, we evaluate the Fastline simulation results of single and double transmission lines with closed-form triangle impulse response and a numerical TIR database calculated using numerical inverse fast Fourier transformation (IFFT) and frequency dependent line parameters [34].

![Fig. 6.3 The cross section of an one conductor microstrip transmission line.](image-url)
Table 6.1 Frequency dependent R and L for microstrip line shown in Fig. 6.3 calculated using UAPDSE.

The first example is a single microstrip line. The cross section of the microstrip is shown in Fig. 6.3, where $W=5\mu m$, $t=2\mu m$, $T=10\mu m$, and $\varepsilon_r=4.4$. The half width of the triangle waveform (Fig. 6.2) is chosen as $\Delta t=10ps$ and the length of the line is 5 cm. The frequency dependent R and L parameters are calculated using the transmission line parameter extraction tool UAPDSE [63] and are shown in Table 6.1. Capacitance ($C=107.13 \text{ pf/m}$) is calculated using UAPDSE. We assume that the conductance of the dielectric is very small and negligible. The voltage at the far-end of the line can be calculated using (6.21) with constant R and L values chosen in the neighborhood of $\lambda = \frac{\pi}{2nt}$, where $t$ is the rise time of the input signal waveform in time domain, and is shown as the dashed line in Fig. 6.4. Here, we have chosen R and L values at 2 GHz. The solid line in Fig. 6.4 is calculated using a numerical inverse fast Fourier transformation with frequency dependent R and L shown in Table 6.1 [34]. We can see
that the amplitudes of the two TIRs are different because of dispersion effects associated with the frequency dependent transmission line parameters. However, the areas underneath the two TIR curves are the most important factor for the time domain convolution approach provided that the time step is small enough to recover the detail of the waveform we want to simulate.

![Graph](image)

**Fig. 6.4** Far-end TIR voltages of 5cm single transmission line calculated with frequency dependent R, L and R, L values at 2GHz.
The TIRs are then used by Fastline [30] to perform the time domain simulation of the transmission line circuit shown in Fig. 6.5. The input signal is a bit string with 150 ps rise and fall times. The bit time is 1.6 ns. The rise and fall parts are sin^2 waveforms. The transmitted pattern is 10101100, where the voltage level for “1” is 1.2 V. The near-end and far-end voltages for the circuit are displayed in Figs. 6.6 and 6.7. We can see that Fastline simulation with the closed-form TIR agrees very well with the Fastline simulation with the TIR database calculated using frequency dependent R and L values. Furthermore, the closed-form TIR formulas are much more efficient for creating the TIR database than using numerical methods such as an IFFT. It costs only 2 to 3 seconds to obtain a TIR database using the closed-form formulas instead of 2 to 3 hours using an IFFT method with 2^{21} sample points on an IBM RS/4000. Also, the closed-form formulas are free from numerical difficulties such as aliasing effects, the required large number of sample points in frequency domain, etc.
Fig. 6.6 The near-end voltages simulated by Fastline on the transmission line.

Fig. 6.7 The far-end voltages simulated by Fastline on the transmission line.
Table 6.2 Frequency dependent R and L for microstrip line shown in Fig. 6.3 calculated using UAPDSE.

<table>
<thead>
<tr>
<th>f(Hz)</th>
<th>R(Ω/m)</th>
<th>L(nH/m)</th>
<th>f(Hz)</th>
<th>R(Ω/m)</th>
<th>L(nH/m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1e1</td>
<td>68.812</td>
<td>529.52</td>
<td>2e9</td>
<td>216.48</td>
<td>495.43</td>
</tr>
<tr>
<td>1e3</td>
<td>68.812</td>
<td>529.52</td>
<td>5e9</td>
<td>310.12</td>
<td>489.83</td>
</tr>
<tr>
<td>1e5</td>
<td>68.812</td>
<td>529.52</td>
<td>7e9</td>
<td>366.94</td>
<td>488.47</td>
</tr>
<tr>
<td>1e6</td>
<td>68.812</td>
<td>529.52</td>
<td>1e10</td>
<td>438.58</td>
<td>487.37</td>
</tr>
<tr>
<td>1e7</td>
<td>68.873</td>
<td>529.48</td>
<td>1.4e10</td>
<td>518.93</td>
<td>486.63</td>
</tr>
<tr>
<td>5e7</td>
<td>70.304</td>
<td>528.69</td>
<td>2e10</td>
<td>620.24</td>
<td>486.09</td>
</tr>
<tr>
<td>1e8</td>
<td>74.252</td>
<td>526.55</td>
<td>3e10</td>
<td>759.64</td>
<td>485.73</td>
</tr>
<tr>
<td>2e8</td>
<td>85.316</td>
<td>521.09</td>
<td>5e10</td>
<td>980.69</td>
<td>485.51</td>
</tr>
<tr>
<td>3e8</td>
<td>97.67</td>
<td>516.45</td>
<td>8e10</td>
<td>1240.5</td>
<td>485.43</td>
</tr>
<tr>
<td>4e8</td>
<td>107.34</td>
<td>512.96</td>
<td>1e11</td>
<td>1386.9</td>
<td>485.41</td>
</tr>
<tr>
<td>6e8</td>
<td>126.98</td>
<td>508.05</td>
<td>1e12</td>
<td>4385.5</td>
<td>485.39</td>
</tr>
<tr>
<td>1e9</td>
<td>160.04</td>
<td>502.08</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 6.8 Transmission line circuit schematic.

The second example is also a single transmission line circuit. The relative cross sectional dimensions of the line are the same as the line shown in Fig. 6.3 (i.e., the dimensions are 5 times larger this time), where W=25μm, t=10μm, T=50μm, and εr=4.4. The line parameters R and L are again calculated using UAPDSE and are shown in Table 6.2. The capacitance (C=107.13 pf/m) is calculated using UAPDSE. We again assume that the conductance of the dielectric is very small and negligible. The TIRs are again calculated using an IFFT with frequency dependent R and L and the closed-form
formulas with R and L values at 1GHz, which is determined by $f = 1/(2\pi \tau_1)$. The calculated TIRs are then used by Fastline to perform the time domain simulation of the circuit shown in Fig. 6.8. The near-end and far-end voltages are shown in Fig. 6.9 and Fig. 6.10. We can see that the difference between the two methods is larger due to the more serious skin effect of the second transmission line with the larger cross section. These effects are also attributable to the longer length of the second transmission line.

Fig. 6.9 The near-end voltage simulated by Fastline on the transmission line.
Fig. 6.10 The far-end voltage simulated by Fastline on transmission line.

Table 6.3 Frequency dependent line parameters for the MCM coupled transmission lines.

<table>
<thead>
<tr>
<th>f(Hz)</th>
<th>$R_{11}(\Omega/m)$</th>
<th>$L_{11}(\mu H/m)$</th>
<th>$R_{12}(\Omega/m)$</th>
<th>$L_{12}(\mu H/m)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1e1$</td>
<td>49.9</td>
<td>560.91</td>
<td>1.2155e-4</td>
<td>111.69</td>
</tr>
<tr>
<td>$1e6$</td>
<td>49.901</td>
<td>560.91</td>
<td>1.2152e-4</td>
<td>111.69</td>
</tr>
<tr>
<td>$1e7$</td>
<td>49.938</td>
<td>560.83</td>
<td>1.1859e-2</td>
<td>111.69</td>
</tr>
<tr>
<td>$1e8$</td>
<td>53.615</td>
<td>546.17</td>
<td>4.3366e-1</td>
<td>111.94</td>
</tr>
<tr>
<td>$4e8$</td>
<td>101.99</td>
<td>9.0e-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$1e9$</td>
<td>157.08</td>
<td>523.14</td>
<td>2.7</td>
<td>112.30</td>
</tr>
<tr>
<td>$2e9$</td>
<td>226.19</td>
<td>523</td>
<td>3.8</td>
<td>112.30</td>
</tr>
<tr>
<td>$4e9$</td>
<td>309.13</td>
<td>5.36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$8e9$</td>
<td>434.29</td>
<td>7.55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$1.6e10$</td>
<td>612.23</td>
<td>10.65</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$3.2e10$</td>
<td>862.56</td>
<td>15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The last example is a MCM double line problem. The frequency dependent parameters $R_{11}$, $R_{12}$, $L_{11}$, and $L_{12}$ are shown in Table 6.3. $G_{11}$ and $G_{12}$ are assumed to be zero, $C_{11}$ is 50.115 pF/m, and $C_{12}$ is 13.664 pF/m. The length of the transmission line is
10 cm. The input signal is a trapezoidal pulse with 150 ps rise and fall times, and a 4 ns bit time. The TIRs are calculated using an IFFT method with frequency dependent line parameters and the closed-form formulas are calculated with the line parameters chosen at 2GHz. The calculated TIRs are then used by Fastline to perform the time domain simulation of the circuit shown in Fig. 6.11. The near-end and far-end voltages on the transmission lines are shown in Fig. 6.12 and 6.13. We can see that the active line voltages simulated using the closed-form formulas with the constant line parameters are close to the line voltages calculated using the frequency dependent line parameters. For the passive line, the differences between the two results are larger. There are two reasons for this. First, constant line parameters do not account for the skin effect. Second, when calculating $p_1$ and $p_2$ in the frequency domain, the characteristic impedances were approximated by inductance and capacitance values per unit length only. In order to overcome these limitations, modified closed-form formulas have to be derived so that they account for frequency dependent transmission line parameters.

Fig. 6.11 Coupled MCM transmission line circuit schematic.
Fig. 6.12 Near-end Voltages on the coupled MCM transmission lines.

Fig. 6.13 Far-end Voltages on the coupled MCM transmission lines.
In summery, these closed-form expressions for the triangle impulse responses provide an accurate and efficient simulation method for transmission line embedded circuit simulations. The results are very accurate when the skin depth in the conductors is comparable to or greater than the conductor thickness. For other cases, frequency dependent line parameters must be employed to obtain accurate results. When compared to using FFT techniques to evaluate the inverse Fourier transformation numerically, additional insight can be gained by solving the integral analytically. In addition, the closed-form solutions avoid the requirements of large amounts of computer memory and computation time. Furthermore, the analytical solution for this integral is free from problems inherent to numerical FFT-based algorithms (e.g. aliasing).
7. TIR CALCULATION FOR LOSSY TRANSMISSION LINE SIMULATION

A fast simulation method for single and coupled lossy lines with frequency-dependent parameters, which is based on triangle impulse responses (TIR), was introduced in chapter 5 for the purpose of time domain simulation of a large number of transmission lines, e.g. 100k, in the package of mainframe computer systems. Fastline, which is the name of the simulation tool based on this method, is capable of simulating linear circuits with lossy transmission lines. The advantage of this method is that a large number of lines can be simulated in a short time as long as an accurate TIR database is available. This tool makes it possible to simulate the properties of the transmission lines and to perform signal integrity checks, including time delay verification and signal waveform distortion evaluation, in a system as complicated as a mainframe computer.

Closed-form representations of the TIR were introduced in chapter 6 for the case of frequency-independent line parameters. Under the quasi-TEM assumption, closed-form TIR representations were given in terms of incomplete Lipschitz-Hankel integrals (ILHI). These closed-form results for the TIR can also be used to carry out time domain simulations. They are accurate when the skin effect is not too serious, i.e., the skin depth is comparable to conductor thicknesses. For interconnections that transmit high frequency digital signals, skin effect may play an important role in the shape of the signal waveform on off-chip lines. Therefore, frequency-dependent line parameters, i.e.,
resistance $R$, inductance $L$, capacitance $C$, and conductance $G$, have to be considered when calculating the TIR [34].

In this chapter, an accelerated inverse Laplace transform (AILT) algorithm [40, 41] is used to calculate the TIR for frequency-dependent line parameters. The excellent performance of the acceleration scheme proposed by D'Amore [40] makes it possible to recover 10 ps signals in the time domain, which requires calculated results for frequencies over more than 6 decades. In section 7.1, we cover the causality of the lossy transmission line model. In section 7.2, we examine the effects of frequency-dependent line parameters on the TIR. We show that frequency dependent capacitance and conductance have to be used to calculate the TIR to ensure the proper shape of the time domain results and accurate time delay. We outline the AILT algorithm that is used to calculate the TIR as well as an IFFT algorithm. In section 7.3, we present an example of an ideal transmission line circuit simulation using AILT to calculate the TIR. We also provide an example of a node-to-node bus in a realistic design simulation by using both the IFFT and AILT algorithms to calculate the TIRs. Fastline and the calculated TIRs are then used to perform the time domain simulations for both transmission line circuits. We also present some conclusions about these methods in section 7.3.

7.1 Causal Models for the Line Parameters

As is well known, at high frequencies the line resistance is proportional to the square root of frequency. Furthermore, the inductance of a transmission line decreases from the DC inductance value and approaches the external inductance of the line as the frequency increases. Also, the electrical properties can vary dramatically for different
materials. For FR-4, which is frequently used for printed circuit boards (PCB), experimental data for dielectric constant $\varepsilon_r$ and loss tangent are given by Coombs [71] as shown in Table 7.1. We can see that from 100Hz to 10GHz, $\varepsilon_r$ varies from 4.8 to 4.4 for FR-4. Likewise, the loss tangent varies from 0.009 to 0.025 over the same frequency range.

<table>
<thead>
<tr>
<th>$f$ (Hz)</th>
<th>$\varepsilon_r$</th>
<th>$\tan\delta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>4.80</td>
<td>0.009</td>
</tr>
<tr>
<td>1000</td>
<td>4.75</td>
<td>0.012</td>
</tr>
<tr>
<td>10000</td>
<td>4.70</td>
<td>0.015</td>
</tr>
<tr>
<td>100000</td>
<td>4.65</td>
<td>0.018</td>
</tr>
<tr>
<td>1MHz</td>
<td>4.60</td>
<td>0.020</td>
</tr>
<tr>
<td>10MHz</td>
<td>4.55</td>
<td>0.022</td>
</tr>
<tr>
<td>100MHz</td>
<td>4.50</td>
<td>0.024</td>
</tr>
<tr>
<td>1000MHz</td>
<td>4.45</td>
<td>0.025</td>
</tr>
<tr>
<td>10000MHz</td>
<td>4.40</td>
<td>0.025</td>
</tr>
</tbody>
</table>

Table 7.1 Experimental data of dielectric constant $\varepsilon_r$ and loss tangent for FR-4.

Changes in the dielectric constant directly affect the line capacitance. Furthermore, the loss tangent and capacitance changes affect the line conductance. It is important to account for the frequency-dependent material properties as well as skin effect in order to calculate the 10 ps triangle impulse responses on lossy transmission lines. There are two reasons for this. First, as shown by Arabi [72], the transfer function of the transmission line $H(\omega)$ must fulfill the requirements that

\[
H(-\omega) = H^*(\omega),
\]

(7.1)

\[
h(t) = 0 \quad \text{for } t < 0.
\]

(7.2)

This puts certain constraints on the frequency dependences of the R, L, G, and C values. In Arabi's paper [72], it is assumed that the resistance of the line is skin effect limited and
that it varies with the square root of frequency, i.e., $R = R_0 \sqrt{\omega}$, where $R_0$ is determined by the geometrical and electrical parameters of the line, and $\omega$ is given in radians per second. We know this square root approximation is quite accurate for very high frequencies. However, for medium frequencies, the skin effect is more complicated than a simple square root law approximation. Assuming a constant capacitance $C$ and negligible conductance loss, i.e., $G = 0$, the transfer function of the line is in this case given by

$$H(\omega) = \exp(-j\omega \sqrt{LC}(1 - jR_0/L\sqrt{\omega})l).$$

(7.3)

For small $R_0$ and particularly at high frequencies, the square root in (7.3) can be expanded whereby only the first two terms are kept. The transfer function of the line can then be written as

$$H(\omega) = \exp(-j\omega \sqrt{LC}) \exp(-R_0/2\sqrt{\omega C/L}).$$

(7.4)

It can be shown that (7.3) and (7.4) do not satisfy constraints (7.1) or (7.2) [72]. The reason is to be found in the modeling of the inductance of the line. The internal inductance is inversely proportional to the square root of $\omega$ and is very small at high frequencies. If the frequency-dependent internal inductance of the line is included in the total inductance, then the transfer function of a line can be reduced to

$$H(\omega) = \exp(-j\omega \sqrt{LC}(1 + R_0/2L\sqrt{\omega})l) \exp(-R_0/2\sqrt{\omega C/L}).$$

(7.5)

It was concluded by Arabi [72] that equation (7.5) satisfies both conditions (7.1) and (7.2). Equation (7.5) differs from (7.4) by a nonlinear phase term. This term, which results from the small variation of the line inductance with frequency, is very small and is
negligible compared with the linear part of the phase \( R_0 / 2L\sqrt{\omega} \) versus 1). Therefore, it is often neglected, particularly at high frequencies. This nonlinear phase term does not appreciably affect the time delay at the far end of the line. However, it is the only phase term that contributes to the pulse degradation and thus significantly affects the pulse distortion at the far end of the line, particularly for very short rise times. The above derivation and statements are concluded under the assumptions of square root law behavior for the per-unit-length resistance, constant capacitance per unit length, and zero conductance loss.

For more general cases, skin effect and frequency dependences of material properties are modeled through frequency dependent R, L, G, and C values. A more general conclusion is that if frequency-dependent resistance R is used for time domain simulations, then frequency-dependent inductance L must also be used to ensure that the results are causal. From the propagation constant expression

\[
\gamma = \sqrt{(R + j\omega L)(G + j\omega C)},
\]

(7.6)

R and L are symmetrical to G and C. That means if frequency-dependent conductance G is used for time domain simulations, then accurate frequency-dependent capacitance also has to be used to ensure the causality of the time domain results. R and L are often currently calculated using line parameter extraction tools that ensure their physical validity. If the same grid, which is two dimensional if a coupled circuit approach is used or one dimensional if a surface integral equation approach [64] is used, is selected when calculating R and L values at different frequencies, then causality due to R and L is ensured. However, frequency independent values of C and linearly increasing
conductance $G = \omega C \tan \delta$ are typically used for transmission line simulations. Unfortunately, this combination of parameters can lead to non-causal time-domain results [72, 73]. For non-causal waveforms, the voltage values at the far-end of a transmission line are not zero at the times prior to the time delay of the transmission line. Specifically, the noncausal voltage waveform has a very slow rise part before a pulse arriving at the far-end. This is a clearly nonphysical phenomenon. The physical result should be that the output voltage is always zero prior to the time delay of a transmission line. For a transmission line circuit, we define that the waveform is noncausal if the voltage at the far-end $V_{\text{far}}(t) \neq 0$ when $t < t_d$. Likewise, constant capacitance $C$ and linearly increasing conductance $G$ with frequency do not model the correct physical phenomenology. Therefore, these approximations can introduce errors in the time-delay calculations as well as the amplitudes of the signal waveform. For a rigorously accurate and physically feasible, i.e., causal, time domain simulation result, accurate frequency-dependent $R$, $L$, $C$, and $G$ have to be used.

For FR-4, Svensson [73] gave a derived formula to fit the complex dielectric constant data in [71]. We will use this formula, which is written as

$$
\varepsilon = \varepsilon_i + a \left( \frac{1}{2} \ln \left( \frac{\tau_2^2(1+\omega^2\tau_1^2)}{\tau_1^2(1+\omega^2\tau_2^2)} \right) - j \arctg(\omega \tau_2) + j \arctg(\omega \tau_1) \right),
$$

(7.7)

to calculate the frequency-dependent complex permittivity, which in turn is used to calculate the frequency dependent values of $C$ and $G$. For the case of FR-4, the coefficients in (7.7) are given by [73] $\varepsilon_i = 4.1$, $\tau_1 = 1.6 \, ps$, $\tau_2 = 1.6 \, ms$, and $a = 0.06$. 

7.2 TIR Calculations Using Inverse Fourier & Laplace Transforms

The frequency domain voltage and current on a single transmission line, which is matched at both the near-end and far-end, can be expressed as

\[ \frac{V(x, \omega)}{E(\omega)} = \frac{1}{2} \exp(-\gamma x), \]

\[ I(x, \omega) = \frac{V(x, \omega)}{Z_0}, \]

where

\[ \gamma = \sqrt{(R + j\omega L)(G + j\omega C)}, \]

\[ Z_0 = \sqrt{(R + j\omega L)/(G + j\omega C)}, \]

and \( E(\omega) \) denotes the spectral response of the voltage source. From the definition of the TIR [30], \( E(t) \) is a triangle impulse input with a 10 ps rise time and a 2V amplitude. In the frequency domain, it can be expressed as

\[ E(\omega) = \frac{8 \sin^3(\omega \Delta t/2)}{\omega^3 \Delta t} \exp(-j\omega \Delta t). \]

In the time domain, the voltage and current along the transmission line are expressed as Fourier transforms of the frequency domain results shown in (7.8)

\[ V(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} V(x, \omega) \exp(j\omega t) d\omega, \]

\[ V(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{4 \sin^3(\omega \Delta t/2)}{\omega^3 \Delta t} \exp(j\omega t - j\omega \Delta t - \gamma x) d\omega, \]

\[ I(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{4 \sin^3(\omega \Delta t/2)}{\omega^3 \Delta t Z_0} \exp(j\omega t - j\omega \Delta t - \gamma x) d\omega. \]
The above Fourier transforms are calculated using both inverse fast Fourier transform (IFFT) and accelerated inverse Laplace transform (AILT) algorithms [40]. For convenience, we outline both methods below.

\[ V(x,t) \text{ and } I(x,t) \text{ can be obtained through an inverse Fast Fourier transform.} \]

Thus

\[ V(x,t) = \frac{1}{2\pi} \sum_{i=1}^{N} V(x,\omega)e^{i\omega t} \Delta \omega, \quad (7.14) \]

\[ I(x,t) = \frac{1}{2\pi} \sum_{i=1}^{N} I(x,\omega)e^{i\omega t} \Delta \omega, \quad (7.15) \]

where

\[ \Delta \omega = \frac{2\omega_c}{N}, \]

\[ \omega_i = \omega_{i-1} + \Delta \omega, \]

\[ \omega_h = -\omega_c - \frac{1}{2} \Delta \omega, \]

\[ \omega_c = \frac{8\pi}{T_r}. \]

In the above equations, \( N \) is the number of frequency divisions and \( \omega_c \) is defined as the maximum angular frequency of the given pulse. For better accuracy, more high frequency harmonics should be included in the transform from the frequency domain to the time domain.

The Fortran software package, INVLTF, which is described by D'Amore [40, 41], can be used for the numerical inversion of the Laplace transform. This package obtains
approximations for the real-valued function \( f(t) \), when numerical values of its Laplace transform

\[
F(z) = \int_0^\infty e^{-zt} f(t) dt, \quad \text{Re}(z) > \sigma_0
\]  

(7.16)

are provided for complex \( z \) in a form of a user-supplied subprogram. In (7.16), \( \sigma_0 \) is the abscissa of convergence for \( F(z) \). The package is based on an accelerated inverse Laplace transform algorithm [40] that employs a Fourier series expansion and Q-D acceleration of the summation of the Fourier series. For convenience, the method is outlined below.

A discretization of the inverse Laplace transform integral,

\[
f(t) = \frac{1}{2\pi i} \int_{\sigma - i\infty}^{\sigma + i\infty} F(z)e^{zt} dz, \quad z = \sigma + iy, \quad \sigma > \sigma_0, \quad t > 0,
\]

(7.17)

can be performed using the trapezoidal rule [74] with step size \( h = \pi/T, \ t \in [0,T] \). This yields

\[
\tilde{f}_m(t) = \frac{e^{\alpha t}}{T} \text{Re} \left\{ \frac{F(\sigma)}{2} + \sum_{k=1}^{\infty} F(\sigma + \frac{ik\pi}{T}) e^{\frac{ik\pi}{T}} \right\} \]  

(7.18)

where the corresponding discretization error is denoted by \( \eta(t) \) and

\[
f(t) = \tilde{f}_m(t) + \eta(t).
\]

(7.19)

The software is designed to return an approximation \( \tilde{f}_N(t) \) of \( f(t) \)

\[
\tilde{f}_N(t) = \frac{e^{\alpha t}}{T} \text{Re} \left\{ \frac{F(\sigma)}{2} + \sum_{k=1}^{N} F(\sigma + \frac{ik\pi}{T}) e^{\frac{ik\pi}{T}} \right\},
\]

(7.20)

which satisfies
where $TOL$ is a user required accuracy.

The original Fourier series can be represented as a formal power series in $z^k$, i.e., as in De Hoog et al. [74]. We write

$$f_N(t) = \frac{2e^{it}}{st} \text{Re} \left\{ \sum_k a_k z^k \right\}, \quad (7.22)$$

where

$$a_0 = \frac{F(\bar{d})}{2}, \quad a_k = F \left( \bar{d} + i \frac{2\pi k}{st} \right), \quad z = e^{it}, \quad \bar{d} = \sigma - \sigma_0, \quad s = \frac{2\pi}{ht},$$

and calculate successive pade approximations $v(z, M)$ to the partial sums of the power series (7.22) in the form of a terminating continued fraction, i.e.,

$$v(z, M) = \frac{A_{2M}}{B_{2M}} = \frac{d_0}{1 + \frac{d_1 z}{1 + \frac{\ldots}{1 + d_{2M} z}}} = \sum_{k=0}^{2M} a_k z^k = S_{2M}, \quad M = 1, 2, 3, \ldots. \quad (7.23)$$

Here, the coefficients $d_k$ are calculated using the Q-D algorithm of Rutishauser [75].
The TIR is calculated for the single transmission line shown in Figs. 7.1 and 7.2. The frequency dependent R, L, and frequency-independent C are calculated using UAPDSE [63] with $\varepsilon_r = 4.6$ and shown in Table 7.2.

<table>
<thead>
<tr>
<th>f(Hz)</th>
<th>R(Ω/m)</th>
<th>L(nH/m)</th>
<th>f(Hz)</th>
<th>R(Ω/m)</th>
<th>L(nH/m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1e1</td>
<td>17.203</td>
<td>529.51</td>
<td>4e8</td>
<td>49.248</td>
<td>497.33</td>
</tr>
<tr>
<td>1e3</td>
<td>17.203</td>
<td>529.51</td>
<td>7e8</td>
<td>62.100</td>
<td>493.01</td>
</tr>
<tr>
<td>1e5</td>
<td>17.203</td>
<td>529.51</td>
<td>1e9</td>
<td>71.407</td>
<td>490.92</td>
</tr>
<tr>
<td>1e6</td>
<td>17.205</td>
<td>529.50</td>
<td>2e9</td>
<td>100.98</td>
<td>488.02</td>
</tr>
<tr>
<td>1e7</td>
<td>17.445</td>
<td>528.98</td>
<td>5e9</td>
<td>159.67</td>
<td>486.09</td>
</tr>
<tr>
<td>2e7</td>
<td>18.112</td>
<td>527.51</td>
<td>7e9</td>
<td>188.93</td>
<td>485.79</td>
</tr>
<tr>
<td>3e7</td>
<td>19.068</td>
<td>525.49</td>
<td>1e10</td>
<td>225.81</td>
<td>485.59</td>
</tr>
<tr>
<td>5e7</td>
<td>21.328</td>
<td>521.09</td>
<td>2e10</td>
<td>319.34</td>
<td>485.44</td>
</tr>
<tr>
<td>7e7</td>
<td>23.611</td>
<td>517.27</td>
<td>5e10</td>
<td>504.92</td>
<td>485.39</td>
</tr>
<tr>
<td>1e8</td>
<td>26.831</td>
<td>512.96</td>
<td>1e11</td>
<td>714.07</td>
<td>485.39</td>
</tr>
<tr>
<td>2e8</td>
<td>36.126</td>
<td>504.63</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.2 Frequency dependent R and L for transmission line shown in Fig. 7.1 calculated using UAPDSE. (C=1.0713 pF/cm)
R(f) and L(f) are interpolated using the smooth interpolation algorithm developed by Akima [76]. The method is devised in such a way that the resulting curve will pass through all the given data points and appear smooth and natural. It is based on a piecewise function; a portion of the curve between a pair of given points is represented by a third-degree polynomial for a single-valued function and by two third-degree polynomials for a multiple-valued function. In this method, the slope of the curve is determined at each given data point locally by the coordinates of five data points, with the data point in question as a center point and two data points on each side of it. Each piece of the function representing a portion of the curve between a pair of given data points is determined by the coordinates of and the slopes at the points.

![Graph](image)

Fig. 7.3 Far-end voltage of TIR for 20 cm line.

The far-end voltages of the TIR for a 20 cm line are shown in Fig. 7.3 and calculated using AILT. An IFFT is also used to calculate the TIR. The results are almost
identical when \( N = 2^{22} \) sample points are used. For the sake of clarity, only the AILT results are shown in Fig. 7.3. Curve 1 is calculated using frequency dependent \( C \) and \( G \), i.e., \( C(\omega) = C_{\infty} \text{real}(\varepsilon(\omega))/4.6 \) and \( G(\omega) = \omega C(\omega) \tan \delta(\omega) \), where \( \text{real}(\varepsilon(\omega)) \) and \( \tan \delta(\omega) \) are obtained from Svensson's formula (7.7). Curve 2 is calculated using Coombs' measurement data for permittivity and loss tangent as shown in Table 7.1 to calculate frequency dependent \( C \) and \( G \). This curve is assumed to be the correct result for later comparisons. Curve 3 is calculated using constant \( C \) at 1GHz, i.e., \( \varepsilon_r = 4.45 \), and linearly increasing conductance \( G \) with frequency. Curve 4 is calculated using constant \( C \) at 1MHz, i.e., \( \varepsilon_r = 4.6 \), and linearly increasing conductance \( G \) with frequency. From Fig. 7.3, we can see that the time delay and shape of the TIR are affected by the frequency dependence of \( C \) and \( G \). Compared to curve 2, curve 3 has a slightly more serious non-causal waveform. This can be concluded from the slopes of the rise edges of the pulses at the far-end. Curve 3 has a smaller slope compared to curve 2. Curve 4 has larger error in time delay. Curve 1 is calculated using Svensson's formula, which overestimates frequency dependence of the capacitance. The capacitance variation of (7.7) is about 30% from 100 Hz to 10 GHz given by (7.7), while the variation given by measurement [71] is about 10%. The overestimated capacitance reduces the dispersion effect in the TIR waveform and introduces an error in the time delay calculation. If the frequency dependences of \( R, L, C, \) and \( G \) are known over the required frequency range, then their use in simulations will give the most accurate results. However, using constant \( C \) and frequency dependent \( R, L, \) and \( G \) can give a good approximation as we see from curve 3 in Fig. 7.3. This is true only when the capacitance value at an appropriate frequency is
used. The appropriate frequency strongly depends on the rise and fall times of the signal waveform.

7.3 Transmission Line Simulation Examples

For the transmission line example shown in Fig. 7.2, the near-end and far-end voltages are calculated using Fastline based on the voltage and current TIRs, and the results are shown in Figs. 7.4, 7.5, 7.6, and 7.7. The voltage source is assumed to be a 10101111000000000000000 pattern waveform with 1.2 V for 1 and 0 V for 0. The bit-time is 1.6 ns while the rise and fall times are both 150 ps. A \( \sin^2 \) waveform is used for the rising and falling edges. Figure 7.6 shows that when compared to the results obtained using Coombs’ measurements directly, the far-end voltage amplitudes differ by about 3%, 0.5%, and 1.5% when using Svensson’s formula, constant C at 1 GHz, and constant C at 1 MHz, respectively. The time delay differences at the far end are about 20 ps, 2 ps, and 50 ps for simulations using Svensson’s formula, constant C at 1 GHz, and constant C at 1 MHz when compared to simulations using Coombs’ measurement data. Using constant C can introduce errors in both the time delay and the signal amplitude if an inappropriate value is used (e.g., 1 MHz value instead of 1 GHz value in this particular case). Using an overestimated frequency dependent C can introduce errors to the time delay and the amplitude of signal. It also reduces the dispersion effect of long transmission lines. We can see this from the rising edge of the far-end voltage. For very high frequency applications, accurate time delay estimation is critical. For rigorously accurate simulations of transmission line circuits with FR-4 as the dielectric material, accurate frequency-dependent C and G models should be used.
Fig. 7.4 Far-end voltage on the transmission line shown in Fig. 7.2.

Fig. 7.5 Partly magnified far-end voltage shown in Fig. 7.4.
Fig. 7.6 Partly magnified far-end voltage shown in Fig. 7.4.

Fig. 7.7 Near-end voltage on the transmission line shown in Fig. 7.2.
The second example is a node-to-node bus in a realistic design simulation problem. The diagram of the net is shown in Fig. 7.8. The net consists of a driver, a receiver, 2 chip to multi-chip module (MCM) lines, 2 MCM lines, 2 printed circuit board
(PCB) lines, 2 sockets, and a strait between these two sockets. The schematic of the net is shown in Fig. 7.9, where the chip to MCM lines are modeled with lumped elements, each MCM line is modeled with lossy transmission lines named as GC_top_via, GC_single_trace, and GC_bottom_via, each printed circuit board line is modeled with a frequency dependent lossy transmission line named as PCB_line, the socket is modeled with lumped elements, and the strait is modeled with lumped elements and a 8cm PCB_line. The voltage source is a bit-string waveform with 1.2 V for 1 and 0 V for 0. The bit-time is 1.6 ns while the rise and fall times are both 150 ps. A sin² waveform is used for the rising and falling edges. The internal impedance of the source is 30Ω. The values for all the lumped elements are shown in Fig. 7.9. Because of the different lengths and losses of the lossy transmission lines, the PCB_lines dominate the waveform at both ends of the net. GC_top_via and GC_bottom_via are modeled with constant R, L, C, and G shown in Table 7.3. The GC_single_trace is modeled with frequency dependent R(f), L(f), and constant C and G shown in Table 7.4. The PCB_line is modeled with frequency dependent R(f), L(f), G(f), and C(f) or constant C listed in Table 7.5. These line parameters are calculated using IBM internal line parameter extraction tools. For extremely high frequencies, such as 1.00e50 GHz, the values are obtained using the square root law for the skin effect. Since we do not have measurement data for the frequency dependent complex dielectric constant of the printed circuit board, the frequency dependent G(f) and C(f) in Table 3 were generated by Barry J. Rubin and Gerard V. Kopcsay at IBM T. J. Watson Research Center in Yorktown Heights using their unpublished algorithm that ensures the causality of the R, L, G, and C model.
Because constant permittivity is assumed in previous simulations and measured frequency dependent permittivities are not available for the dielectric material used, we match the low frequency capacitance per unit length to the constant \( C \), which is 1.6528 pF/cm, in case B and 1 GHz capacitance per unit length value to the constant \( C \) in case C.

<table>
<thead>
<tr>
<th></th>
<th>( R \text{ (} \Omega \text{/cm)} )</th>
<th>( L \text{ (} \mu \text{H/cm) } )</th>
<th>( C \text{ (} p\text{F/cm)} )</th>
<th>( G )</th>
</tr>
</thead>
<tbody>
<tr>
<td>GC_top_via</td>
<td>0.0544</td>
<td>4.457</td>
<td>1.273</td>
<td>0</td>
</tr>
<tr>
<td>GC_bottom_via</td>
<td>0.0544</td>
<td>5.51</td>
<td>1.03</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 7.3 Line parameters for GC_top_via and GC_bottom_via.

<table>
<thead>
<tr>
<th>( F \text{ (GHz)} )</th>
<th>( R(F) \text{ (} \Omega \text{/cm)} )</th>
<th>( L(F) \text{ (} \mu \text{H/cm)} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00E-9</td>
<td>0.30137</td>
<td>5.6976</td>
</tr>
<tr>
<td>1.00E-5</td>
<td>0.30137</td>
<td>5.6976</td>
</tr>
<tr>
<td>2.15E-5</td>
<td>0.30137</td>
<td>5.6976</td>
</tr>
<tr>
<td>4.64E-5</td>
<td>0.30137</td>
<td>5.6976</td>
</tr>
<tr>
<td>1.00E-4</td>
<td>0.30139</td>
<td>5.6976</td>
</tr>
<tr>
<td>2.15E-4</td>
<td>0.30144</td>
<td>5.6836</td>
</tr>
<tr>
<td>4.64E-4</td>
<td>0.30169</td>
<td>5.6730</td>
</tr>
<tr>
<td>1.00E-3</td>
<td>0.30279</td>
<td>5.6323</td>
</tr>
<tr>
<td>2.15E-3</td>
<td>0.30700</td>
<td>5.4782</td>
</tr>
<tr>
<td>4.64E-3</td>
<td>0.31727</td>
<td>5.1180</td>
</tr>
<tr>
<td>1.00E-2</td>
<td>0.32891</td>
<td>4.7739</td>
</tr>
<tr>
<td>2.15E-2</td>
<td>0.34222</td>
<td>4.6187</td>
</tr>
<tr>
<td>4.64E-2</td>
<td>0.37742</td>
<td>4.5169</td>
</tr>
<tr>
<td>1.00E-1</td>
<td>0.45910</td>
<td>4.3852</td>
</tr>
<tr>
<td>2.15E-1</td>
<td>0.59591</td>
<td>4.2649</td>
</tr>
<tr>
<td>4.64E-1</td>
<td>0.82125</td>
<td>4.2623</td>
</tr>
<tr>
<td>1.00</td>
<td>1.1758</td>
<td>4.1960</td>
</tr>
<tr>
<td>2.15</td>
<td>1.7230</td>
<td>4.1521</td>
</tr>
<tr>
<td>4.64</td>
<td>2.5355</td>
<td>4.1306</td>
</tr>
<tr>
<td>1.00E+1</td>
<td>3.6949</td>
<td>4.1306</td>
</tr>
</tbody>
</table>

Table 7.4 Line parameters for GC_single_trace. (C=1.4394 pF/cm, G=0)
| f (GHz) | R (Ω/cm) | L(μH/cm) | G (mS/cm) | C (pF/cm) | R (Ω/cm) | L(μH/cm) | G (mS/cm) | C (pF/cm) | R (Ω/cm) | L(μH/cm) | G (mS/cm) | C (pF/cm) |
|------|---------|---------|----------|----------|---------|---------|----------|----------|---------|---------|----------|----------|---------|
| 1.00e-9 | 0.30184 | 6.2405 | 0.69875e-13 | 1.6528 | 0.74069e-13 | 1.7520 |
| 1.00e-5 | 0.30184 | 6.2403 | 0.69875e-9 | 1.6528 | 0.74069e-9 | 1.7520 |
| 2.15e-5 | 0.30185 | 6.2396 | 0.323e-8 | 1.6528 | 0.34238e-8 | 1.7520 |
| 4.64e-5 | 0.30187 | 6.2361 | 0.15044e-7 | 1.6528 | 0.15947e-7 | 1.7520 |
| 1.00e-4 | 0.30197 | 6.2200 | 0.69875e-7 | 1.6528 | 0.74069e-7 | 1.7520 |
| 2.15e-4 | 0.30243 | 6.1484 | 0.32299e-6 | 1.6528 | 0.34238e-6 | 1.7520 |
| 4.64e-4 | 0.30422 | 5.8794 | 0.15043e-5 | 1.6528 | 0.15946e-5 | 1.7520 |
| 1.00e-3 | 0.30890 | 5.2360 | 0.69852e-5 | 1.6528 | 0.74044e-5 | 1.7520 |
| 2.15e-3 | 0.31605 | 4.4994 | 0.32250e-4 | 1.6527 | 0.34186e-4 | 1.7519 |
| 4.64e-3 | 0.32432 | 3.9838 | 0.14937e-3 | 1.6524 | 0.15833e-3 | 1.7516 |
| 1.00e-2 | 0.33239 | 3.7061 | 0.67625e-3 | 1.6509 | 0.71684e-3 | 1.7500 |
| 2.15e-2 | 0.34192 | 3.5860 | 0.28042e-2 | 1.6451 | 0.29725e-2 | 1.7438 |
| 4.64e-2 | 0.36375 | 3.5194 | 0.90225e-2 | 1.6293 | 0.95672e-2 | 1.7271 |
| 1.00e-1 | 0.41688 | 3.4464 | 0.20142e-1 | 1.6099 | 0.21350e-1 | 1.7065 |
| 2.15e-1 | 0.51870 | 3.3669 | 0.43805e-1 | 1.5955 | 0.46434e-1 | 1.6913 |
| 4.64e-1 | 0.70355 | 3.2949 | 0.10277 | 1.5775 | 0.10894 | 1.6722 |
| 1.00 | 1.0260 | 3.2313 | 0.20255 | 1.5593 | 0.21471 | 1.6528 |
| 2.15 | 1.4905 | 3.1814 | 0.42474 | 1.5466 | 0.45024 | 1.6394 |
| 4.64 | 2.1693 | 3.1472 | 0.10104e1 | 1.5298 | 1.0711 | 1.6216 |
| 1.00e+1 | 3.1726 | 3.1240 | 0.20577e1 | 1.5115 | 2.1812 | 1.6022 |
| 1.00e+5 | 1.0033e+22 | 3.0738 | 0.19523e2 | 1.4471 | 20.695e2 | 1.5340 |

Table 7.5 Line parameters for PCB_line. (Case A: C=1.6528 pF/cm,
\[ G(\omega) = \omega C \tan \delta, \tan \delta = 0.02 \])

Both the IFFT and AILT algorithms were used to calculate the TIR database for each kind of lossy transmission line in the schematic. The TIR databases calculated using the IFFT are then used by Fastline to perform the time domain simulations. First we simulated the voltage waveform at the near-end and far-end of the net shown in Fig. 7.10 for case A. In order to demonstrate the differences due to the different line models of the PCB_line, the first pulse at the far-end is shown in Fig. 7.11 for cases A, B, and C in Table 7.5. We can see that the time domain waveform for case A has a slow rise portion, which is not zero, before the pulse arrives at the far-end. We say the time
domain result for case A is not causal. This is due to the noncausality of the G and C model. However, as expected Cases B and C yield causal time domain results. The time delay differences among the three cases are on the order of 150 to 200 ps, which is about 2 to 3 percent of the total time delay. However, this time delay difference is about 10% of bit time. In this case, in order to accurately simulate transmission lines and obtain causal time domain results, accurate frequency dependent line models, i.e., $R(f)$, $L(f)$, $G(f)$, and $C(f)$ have to be used. Far-end voltage waveforms on the node-to-node bus are simulated and shown in Fig. 7.12 for case B in Table 7.5. IFFT and AILT are used to calculated TIR databases. The databases are then used by Fastline to perform the time domain simulation. We can see that the time domain results agree with each other very well when different methods are used to calculate TIRs. This also proves the accuracy of TIR database calculated using the AILT algorithm.

![Fig. 7.10 Near-end and far-end voltage waveform on the node-to-node bus simulated by Fastline using the TIRs calculated with the IFFT.](image)
Fig. 7.11 Far-end voltage waveform on the node-to-node bus simulated by Fastline using the TIRs calculated with the IFFT for cases A, B, and C in Table 7.5.

Fig. 7.12 Far-end voltage waveform on the node-to-node bus simulated by Fastline using TIRs calculated with the IFFT and AILT algorithms for case B in Table 7.5.
In summary, TIRs of frequency dependent, single, lossy transmission lines are calculated using both the inverse fast Fourier transform (IFFT) and accelerated inverse Laplace transform (AILT) approaches. The results of these two methods agree with each other very well. The computed TIR database is also used by Fastline to perform time domain simulation for transmission line embedded circuit simulation.

For very high data rate systems fabricated on FR-4, e.g., for near-GHz or higher applications, the transmission lines are best modeled with accurate frequency-dependent line parameters, R(f), L(f), C(f), and G(f). Using R(f), L(f), constant C, and G(f) can lead to non-causal results as well as errors in the time delay and waveform shape estimation if the constant C value is chosen at the wrong frequency. Using an overestimated frequency dependent C (Svensson model) can also lead to inaccurate simulation results. In this chapter, we have only discussed single lines. For coupled lines, frequency domain results developed by You [38] can be used to carry out the inverse Laplace transform for the TIRs. The calculated TIRs can be used by Fastline or similar tools to do timing and cross-talk verifications.
8. CONCLUSIONS AND FUTURE WORK

Frequency domain finite difference (FDFD) method is used to solve Maxwell's equations. Linear systems in terms of state space equations are derived from electromagnetic systems with discontinuities. Three kinds of boundary conditions, the perfect electric conductor, perfect magnetic conductor, and perfectly-matched layer, are used to truncate electromagnetic systems. PVL, Krylov, rational Krylov, and a frequency segmentation technique with PVL are used to perform the reduced-order modeling of the linear systems derived from electromagnetic systems with discontinuities. The numerical examples show that PVL and Krylov algorithms are limited in accuracy for general wide-band applications. However PVL combined with frequency segmentation is a very robust and accurate approach for reduced-order modeling of complex electromagnetic systems.

For solution of the linear systems obtained in FDFD electromagnetic solvers, PVL with expansion at infinity can avoid the LU decomposition step that is costly both in speed and memory. In each iteration step, only a matrix-vector product must be processed instead of the solution of a large-scale system of linear equations. Although it suffers from the problem of degraded biorthogonality, which makes the order of the reduced-order model higher than PVL with expansion at finite frequency, PVL with expansion at infinity makes it possible to solve many complicated electromagnetic problems efficiently. It also provides better wide frequency band results than PVL with expansion at a finite frequency, which only gives correct results near the expansion point.
However, for some problems, such as some of the microstrip discontinuity problems, we found the results calculated using PVL with expansion at infinity are not correct. The exact reason for the errors has not been determined yet and needs further investigation. There are several possible factors that may account for the shortcoming of this approach. First, the FDFD step introduces low-frequency noise to the linear system. For microstrip problems, cutoff frequency is very low. Thus low frequency components of the electrical and magnetic fields are an important part of the field values. Secondly, the degraded biorthogonality is another possible reason for the errors in this approach. More research is needed to resolve this point.

The reduced-order modeling algorithms do not assure the passivity of the macromodels derived due to the large number of degrees of freedom involved in the electromagnetic models. More research work is needed to develop efficient passive reduced-order modeling algorithms.

A time domain convolution method is introduced to perform time domain simulations of transmission line circuits. The method is based on the triangle impulse responses (TIR) of these transmission lines. Fastline is already developed in IBM based on this method. Time domain triangle impulse responses for single and two-line systems have been developed using an inverse Fourier transformation from the frequency domain modal analysis formulas. The inverse Fourier transform has been carried out analytically (assuming frequency independent line parameters) and closed-form triangle impulse responses have been derived. Combining these closed-form expressions for the triangle impulse responses with the time domain convolution method provides an accurate and
efficient simulation method for transmission line embedded circuit simulations. The results are very accurate when the skin depth in the conductors is comparable to or greater than the conductor thickness. For other cases, frequency dependent line parameters must be employed to obtain accurate results. Furthermore, when compared to using FFT techniques to evaluate the inverse Fourier transformation numerically, additional insight can be gained by solving the integral analytically. In addition, the closed-form solutions avoid the requirements of large amounts of computer memory and computation time. Furthermore, the analytical solution for this integral is free from problems inherent to numerical FFT-based algorithms (e.g. aliasing).

This general method can also be used to analyze transmission line responses with other input signals such as step function inputs. The disadvantage of this approach is that the RLCG parameters used in the derivation are frequency independent. In some cases, this will introduce errors in the calculated time domain responses if the frequency dependence of these terms must be accounted for. However, these cases are not often encountered in practice. Development of closed-form TIR including frequency dependent line parameters will be a good effort as future work in this topic as long as we are using numerical methods to calculate the TIR database.

In the cases that frequency dependent transmission line parameters have to be used for accurate results, numerical methods can be used to calculate the TIR database. TIR of frequency dependent, single, lossy transmission lines are calculated using both an inverse fast Fourier transform (IFFT) and an accelerated inverse Laplace transform (AILT) approaches. The results of these two methods agree with each other very well.
However, AILT is much better compared to the IFFT technique in running time. The computed TIR database is also used by Fastline to perform time domain simulation for transmission line embedded circuit simulation.

The simulation results have shown that for very high data rate systems fabricated on FR-4, e.g., for near-GHz or higher applications, the transmission lines are best modeled with accurate frequency-dependent line parameters, R(f), L(f), C(f), and G(f). Using R(f), L(f), constant C, and G(f) can lead to non-causal results as well as errors in the time delay and waveform shape estimation if the constant C value is chosen at the wrong frequency.

As for future work, we can focus on how to extract causal frequency dependent transmission line parameters. This will require more work to be done on dielectric material frequency dependent characterizations, e.g., extractions of frequency dependent permittivity and loss tangent of dielectric materials used in printed circuit boards, multi-chip modules (MCM) etc.
APPENDIX A: NAMELIST COMMANDS IN M3.IN

There are six NAMELIST commands in M3.IN. They are:

- **NAMELIST $def_box ($DEF_BOX)**
  
  Integer variables: nx1, ny1, nz1, nx2, ny2, nz2

  $def_box ($DEF_BOX)
  
  $end ($END)

  The NAMELIST $def_box ($DEF_BOX) is used to define the indices of discrete nodes in rectangular computational domain. The index (nx1, ny1, nz1) denotes the index of minimum node of the domain, and the index (nx2, ny2, nz2) denotes the index of maximum node of the domain. The default value of nx1, ny1, ..., and nz2 is zero. The NAMELIST $def_box ($DEF_BOX) is recommended to be used first, prior to other commands.

- **NAMELIST $def_block ($DEF_BLOCK)**
  
  Integer variables: nx1, ny1, nz1, nx2, ny2, nz2

  Real variables: x1, y1, z1, x2, y2, z2, scale

  $def_block ($DEF_BLOCK)
  
  $end ($END)

  The NAMELIST $def_block ($DEF_BLOCK) is used to define both the indices and coordinates of discrete nodes in rectangular sub-domain. Each sub-domain consists of the uniform grid. However, the size of the grid may change from the sub-domain to the sub-domain. It must be noticed that the whole rectangular computational domain must be filled by sub-domains and the sub-domains must be connected together. The index (nx1, ny1, nz1) denotes the index of minimum node of the sub-domain, and the index (nx2, ny2, nz2) denotes the index of maximum node of the sub-domain. The point (x1, y1, z1) denotes the coordinate of the minimum node of the sub-domain, and the point (x2, y2, z2) denotes the coordinate of the maximum node of the sub-domain. The default value of nx1, ny1, ..., and nz2; x1, y1, ..., and z2 is zero. The real variable scale is used to scale the coordinate variables into meter unit. Its default value is equal to 1. The NAMELIST $def_block ($DEF_BLOCK) must be used prior to the NAMELIST $cube ($CUBE), $cylinder ($CYLINDER), $rectangle ($RECTANGLE), and $circle ($CIRCLE).

- **NAMELIST $cube ($CUBE)**
Real variables: \(x_1, y_1, z_1, x_2, y_2, z_2, \text{scale}, \text{er}, \text{ur}\)
Character variable: \text{mate}

$\text{cube} (\$\text{CUBE})$

\(x_1=\text{real}\ y_1=\text{real}\ z_1=\text{real}\ x_2=\text{real}\ y_2=\text{real}\ z_2=\text{real}\ \text{scale}=\text{real}\)

\text{mate}=\text{pec}'\ ('\text{PEC}')\ or\ '\text{media}'\ ('\text{MEDIA}')
\text{er}=\text{real}, \text{ur}=\text{real}

$\text{Send} (\$\text{SEND})$

The NAMELIST $\text{cube} (\$\text{CUBE})$ defines a rectangular object whose eight sides parallel to the coordinate axes of Cartesian coordinate. The point \((x_1, y_1, z_1)\) denotes the coordinate of the minimum node of the object, and the point \((x_2, y_2, z_2)\) denotes the coordinate of the maximum node of the object. The default value of \(x_1, y_1, \ldots,\text{and } z_2\) is zero. The variable \text{scale} is the same as above. The object can be either perfect electric conductor (PEC) or other media, which is defined by character string: \text{mate}='\text{pec}'\ or\ '\text{media}'. The default character string of \text{mate} is 'media'. The \text{er}, \text{ur} stands for the permittivity and permeability of isotropic media, respectively. The default value of \text{er}, \text{ur} is one.

\vspace{1em}

\text{NAMELIST} $\text{scylinder} (\$\text{CYLINDER})$

Real variables: \(x_1, y_1, z_1, x_2, y_2, z_2, \text{radius}, \text{scale}, \text{er}, \text{ur}\)
Character variable: \text{mate}

$\text{scylinder} (\$\text{CYLINDER})$

\(x_1=\text{real}\ y_1=\text{real}\ z_1=\text{real}\ x_2=\text{real}\ y_2=\text{real}\ z_2=\text{real}\ \text{scale}=\text{real}\)
\text{radius}=\text{real}

\text{mate}=\text{pec}'\ ('\text{PEC}')\ or\ '\text{media}'\ ('\text{MEDIA}')
\text{er}=\text{real}, \text{ur}=\text{real}

$\text{Send} (\$\text{SEND})$

The NAMELIST $\text{scylinder} (\$\text{CYLINDER})$ defines a cylindrical object whose axis parallels to a coordinate axis of Cartesian coordinate. The point \((x_1, y_1, z_1)\) denotes the coordinate of the central origin at the bottom section of the cylinder. The point \((x_2, y_2, z_2)\) denotes coordinate of the central origin at the top section of the cylinder, i.e. the straight line connecting point \((x_1, y_1, z_1)\) and \((x_2, y_2, z_2)\) is the axis of the cylinder. The default value of \(x_1, y_1, \ldots,\text{and } z_2\) is zero. The \text{radius} means the radius of the cylinder. The default value of \text{radius} is zero. The \text{scale} is the same as above. The object can be either perfect electric conductor (PEC) or other media, which is defined by character string: \text{mate}='\text{pec}'\ or\ '\text{media}'. The default character string of \text{mate} is 'media'. The \text{er}, \text{ur} stands for the permittivity and permeability of isotropic media, respectively. The default value of \text{er} and \text{ur} is set to one.

\vspace{1em}

\text{NAMELIST} $\text{rectangle} (\$\text{RECTANGLE})$
Real variables: x1, y1, z1, x2, y2, z2, scale
Character variable: mate

$rectangle ($RECTANGLE)
  x1=real y1=real z1=real x2=real y2=real z2=real scale=real
  mate='pec' ('PEC')
$end ($END)

The NAMELIST $rectangle ($RECTANGLE) defines an infinite rectangular PEC sheet. The point (x1, y1, z1) denotes the coordinate of the minimum point of the rectangle. The point (x2, y2, z2) denotes the coordinate of the maximum point of the rectangle. The default value of x1, y1, ..., and z2 is zero. The scale is the same as above. The PEC sheet is defined by the character string: mate='pec' or 'media'. The default character string of mate is 'media'.

NAMELIST $circle ($CIRCLE)

Real variables: x1, y1, z1, radius, scale
Character variable: mate, axis

$circle ($CIRCLE)
  x1=real y1=real z1=real radius=real scale=real
  mate='pec' ('PEC')
  axis='x' ('X') or 'y' ('Y') or 'z' ('Z')
$end ($END)

The NAMELIST $circle ($CIRCLE) defines an infinite circular PEC sheet. The point (x1, y1, z1) denotes the original coordinate of the circle. The default value of x1, y1, and z1 is zero. The scale is the same as above. The character string axis denotes the normal direction of the circular sheet. The default value of axis is 'x'. The PEC sheet is defined by the character string: mate='pec' or 'media'. The default character string of mate is 'media'.

There are twelve namelist commands in ROMES.IN. They are:

- **NAMELIST $def_box ($DEF_BOX)**

  Integer variables: nxl, nyl, nzl, nx2, ny2, nz2
  $def_box ($DEF_BOX)
  nxl=integer nyl=integer nzl=integer nx2=integer ny2=integer nz2=integer
  Send ($SEND)

  The NAMELIST $def_box ($DEF_BOX) is used to define the indices of discrete nodes in the cubic computational domain. The index (nxl, nyl, nzl) denotes the index of minimum node of the domain, and the index (nx2, ny2, nz2) denotes the index of maximum node of the domain. The default value of nxl, nyl, ..., and nz2 is zero. The NAMELIST $def_box ($DEF_BOX) is recommended to be used first, prior to other commands.

- **NAMELIST $def_pml ($DEF_PML)**

  Integer variables: nxl, nyl, nzl, nx2, ny2, nz2
  Logical variables: wavexl, waveyl, wavezl, wavex2, wavey2, wavez2
  Real variables: rxl, ryl, rzl, rx2, ry2, rz2
  $def_pml ($DEF_PML)
  nx l=integer ny l=integer nzl=integer nx2=integer ny2=integer nz2=integer
  rx l=real ryl=real rzl=real rx2=real ry2=real rz2=real
  wavex l=.true. (.false.) wavey l=.true. (.false.) wavezl=.true. (.false.)
  wavex2=.true. (.false.) wavey2=.true. (.false.) wavez2=.true. (.false.)
  Send ($SEND)

  The NAMELIST $def_pml ($DEF_PML) defines the perfectly matched layers (PML) to absorb the outgoing wave and evanescent wave. The integer nxl, nx2, nyl, ny2, nzl, and nz2 denote the layers of PML on six faces of the rectangular truncated boundary, respectively. Their default values are zero. The logical variable wavexl, ..., and wavez2 equal to .TRUE. denotes that the wave is propagating wave in corresponding direction, while the .FALSE. means the evanescent wave. The default value of wavexl, ..., and wavez2 is .TRUE.. The real variable rxl, rx2, ryl, ry2, rzl, and rz2 indicate the absorbing rate of the six PML faces. For the propagating wave, the default value of rxl, rx2, ryl, ry2, rzl, rz2 is -100 dB, whose unit is dB for propagating wave. For the evanescent wave, the default value of rxl, rx2, ryl, ry2, rzl, rz2 is 10.

- **NAMELIST $inter ($INTER)**

  Integer variables: nxl, nyl, nzl, nx2, ny2, nz2
$sinter (SINTER)$

nx1=integer ny1=integer nz1=integer nx2=integer ny2=integer nz2=integer
$end (SEND)$

The NAMELIST $sinter (SINTER)$ defines a cubic computational domain. The index (nx1, ny1, nz1), (nx2, ny2, nz2) denotes the minimum index of the domain and the maximum index of the domain, respectively. The default value of nx1, ny1, ..., and nz2 is zero. The NAMELIST $sinter (SINTER)$ can modify the computational sub-domain defined by M3.IN.

NAMELIST $spec (SPEC)$

Integer variables: nx1, ny1, nz1, nx2, ny2, nz2
$spec (SPEC)$

nx1=integer ny1=integer nz1=integer nx2=integer ny2=integer nz2=integer
$end (SEND)$

The NAMELIST $spec (SPEC)$ defines a cubic perfect electric conductor (PEC), either a cube or an infinite sheet. The index (nx1, ny1, nz1), (nx2, ny2, nz2) denotes the minimum index of the domain and the maximum index of the domain, respectively. The default value of nx1, ny1, ..., and nz2 is zero. The NAMELIST $spec (SPEC)$ can modify the PEC domain defined by M3.IN.

NAMELIST $spmc (SPMC)$

Integer variables: nx1, ny1, nz1, nx2, ny2, nz2
$spmc (SPMC)$

nx1=integer ny1=integer nz1=integer nx2=integer ny2=integer nz2=integer
$end (SEND)$

The NAMELIST $spmc (SPMC)$ defines ONE-CELL cubic perfect magnetic conductor (PMC). The index (nx1, ny1, nz1), (nx2, ny2, nz2) denotes the minimum index of the domain and the maximum index of the domain, respectively. The default value of nx1, ny1, ..., and nz2 is zero.

NAMELIST $smate (SMATE)$

Integer variables: nx1, ny1, nz1, nx2, ny2, nz2
Logical variables: isotr_e, isotr_u, isotr_s
Real variables: er, erx, ery, erz, urx, ury, urz, sig, sigx, sigy, sigz
$smate (SMATE)$

nx1=integer ny1=integer nz1=integer nx2=integer ny2=integer nz2=integer isotr_e=.true. (.false.)
er=real (erx=real, ery=real, erz=real)
isotr_u=.true. (.false.)
ur=real (urx=real, ury=real, urz=real)
isotr_s=.true. (.false.)
\( \text{sig}=\text{real} \ (\text{sigx}=\text{real}, \ \text{sigy}=\text{real}, \ \text{sigz}=\text{real}) \)

\$\text{end} \ (\$\text{SEND}) \$

The NAMELIST $mate \ ($MATE) defines the material characteristics in a cubic region. The index \((nx1, ny1, nz1), (nx2, ny2, nz2)\) denotes the minimum index of the domain and the maximum index of the domain, respectively. The default value of \(nx1, ny1, ..., \) and \(nz2\) is zero. The logical variable \(\text{isotr}_e, \ \text{isotr}_u, \ \text{isotr}_s\) equal to \(\text{TRUE}\). denotes that the media is isotropic, while the value of \(er, ur, sig\) should be defined. When \(\text{isotr}_e, \ \text{isotr}_u, \ \text{isotr}_s\) is equal to \(\text{FALSE}\), the elements of diagonal tensor, \(erx, ery, erz, urx, ury, urz, sigx, sigy, sigz\) should be defined. The default value of \(\text{isotr}_e, \ \text{isotr}_u, \ \text{isotr}_s\) is \(\text{TRUE}\). \(\text{TRUE}\). The default value of \(er, erx, ery, erz, ur, urx, ury, urz\) is equal to one. The default value of \(sig, sigx, sigy, sigz\) is set to zero. The NAMELIST $mate \ ($MATE) can modify the material characteristics defined by M3.IN.

\( \Box \)

NAMELIST $source \ ($SOURCE)

Integer variables: \(nx1, ny1, nz1, nx2, ny2, nz2\)
Real variables: \(jx, jy, jz\)

\$source \ ($SOURCE)
\begin{align*}
\text{nx1}=\text{integer} & \ \ \text{ny1}=\text{integer} & \ \ \text{nz1}=\text{integer} & \ \ \text{nx2}=\text{integer} & \ \ \text{ny2}=\text{integer} & \ \ \text{nz2}=\text{integer} \\
\text{jx}=\text{real} & \ \ \text{jy}=\text{real} & \ \ \text{jz}=\text{real}
\end{align*}
\$\text{end} \ (\$\text{SEND}) \$

The NAMELIST $source \ ($SOURCE) defines the excitation current distribution. The index \((nx1, ny1, nz1), (nx2, ny2, nz2)\) denotes the minimum index of the domain, either a cube or an infinite sheet, and the maximum index of the domain, respectively. The default value of \(nx1, ..., nz2\) is zero. The real variable \(jx, jy, jz\) indicates the \(x, y, z\) component of excitation current, respectively. Their default values are zero.

\( \Box \)

NAMELIST $pvl \ (PVL)

Integer variables: \(nexp, npoint, npvl\)
Real variables: \(fs(l), fn(l), fe(l), fs(2), fn(2), fe(2), ..., fs(nexp), fn(nexp), fe(nexp)\)

\$pvl \ (PVL)
\begin{align*}
\text{nexp}=\text{integer} & \ \ \text{npoint}=\text{integer} & \ \ \text{npvl}=\text{integer} \\
\text{fs}(1)=\text{real (Hz)} & \ \ \text{fs}(2)=\text{real (Hz)} & \ \ ... & \ \ \text{fs}(nexp)=\text{real (Hz)} \\
\text{fn}(1)=\text{real (Hz)} & \ \ \text{fn}(2)=\text{real (Hz)} & \ \ ... & \ \ \text{fn}(nexp)=\text{real (Hz)} \\
\text{fe}(1)=\text{real (Hz)} & \ \ \text{fe}(2)=\text{real (Hz)} & \ \ ... & \ \ \text{fe}(nexp)=\text{real (Hz)}
\end{align*}
\$\text{end} \ (\$\text{SEND}) \$

The NAMELIST $pvl \ (PVL) defines the computational parameters that are required by PVL, Krylov, and rational Krylov. The parameter \(nexp\) denotes the number of segmented sub bands. The parameter \(fs, fn\) denotes the minimum frequency and maximum frequency for given sub frequency band, respectively.
The parameter $fe$ is the frequency of expansion point for the corresponding sub band. For PVL with expansion at infinity, $fe$ is the frequency shift value. The default value of $fs$, $fn$, $fe$ is zero. The integer npoint indicates the calculating points within the frequency sub band, whose default value is 100. The integer npvl defines the order of PVL, Krylov, and rational Krylov procedure. The default value of npvl is 50.

- **NAMELIST $probe_e (PROBE_E)**

  Integer variables: $nx1$, $ny1$, $nz1$, $nx2$, $ny2$, $nz2$

  $probe_e (PROBE_E)$

  \[
  nx1=integer \quad ny1=integer \quad nz1=integer \quad nx2=integer \quad ny2=integer \quad nz2=integer
  \]

  Send ($SEND$)

  The NAMELIST $probe_e (PROBE_E)$ defines the extraction of electric field components on Yee's lattice. The index ($nx1$, $ny1$, $nz1$), ($nx2$, $ny2$, $nz2$) denotes the minimum node and maximum node in the cubic domain in which the electric field components will be sampled. It should be noticed that the node ($nx1$, $ny1$, $nz1$) can not coincide with node ($nx2$, $ny2$, $nz2$).

- **NAMELIST $probe_h (PROBE_H)$**

  Integer variables: $nx1$, $ny1$, $nz1$, $nx2$, $ny2$, $nz2$

  $probe_h (PROBE_H)$

  \[
  nx1=integer \quad ny1=integer \quad nz1=integer \quad nx2=integer \quad ny2=integer \quad nz2=integer
  \]

  Send ($SEND$)

  The NAMELIST $probe_h (PROBE_H)$ defines the extraction of magnetic field components on Yee's lattice. The index ($nx1$, $ny1$, $nz1$), ($nx2$, $ny2$, $nz2$) denotes the minimum node and maximum node in the cubic domain in which the magnetic field components will be sampled. It should be noticed that the node ($nx1$, $ny1$, $nz1$) can not coincide with node ($nx2$, $ny2$, $nz2$).

- **NAMELIST $probe_v (PROBE_V)$**

  Integer variables: $nx1$, $ny1$, $nz1$, $nx2$, $ny2$, $nz2$

  $probe_v (PROBE_V)$

  \[
  nx1=integer \quad ny1=integer \quad nz1=integer \quad nx2=integer \quad ny2=integer \quad nz2=integer
  \]

  Send ($SEND$)

  The NAMELIST $probe_v (PROBE_V)$ defines the extraction of the voltage along a line routine on Yee's lattice. The index ($nx1$, $ny1$, $nz1$), ($nx2$, $ny2$, $nz2$) denotes the minimum node and maximum node in the line.

- **NAMELIST $probe_{i} (PROBE_{I})$**

  Integer variables: $nx1$, $ny1$, $nz1$, $nx2$, $ny2$, $nz2$
$probe_i \ (PROBE_1) \\
nx1=integer \
nyl=integer \
nzl=integer \
nx2=integer \
ny2=integer \
nz2=integer \\
Send \ (SEND) \\
The NAMELIST $probe_i \ (PROBE_1) \ defines \ the \ extraction \ of \ the \ current \ around \ a \ loop \ routine \ on \ Yee's \ lattice. \ The \ index \ (nx1, \ ny1, \ nz1), \ (nx2, \ ny2, \ nz2) \ denotes \ the \ minimum \ node \ and \ maximum \ node \ in \ the \ loop.
REFERENCES


