

Waveform Relaxation and Transverse Partitioning Algorithms for Simulation of Massively Coupled Interconnects

by

Natalie M. Nakhla

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

Ottawa-Carleton Institute for Electrical and Computer Engineering

Department of Electronics

Carleton University

Ottawa, Ontario, Canada

November 2008

Copyright ©

2008 - Natalie M. Nakhla



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-47485-3
Our file *Notre référence*
ISBN: 978-0-494-47485-3

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

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

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

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

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Abstract

With the continually increasing operating frequencies, signal integrity and interconnect analysis in high-speed designs are becoming increasingly important. Interconnect effects such as ringing, signal delay, distortion and crosstalk can severely degrade signal integrity. As a result, these effects have become the dominant factors which limit the overall performance of VLSI systems. If not considered during the design stage, interconnect effects can render a circuit inoperable or cause it to fail in meeting the required specifications. In addition, at relatively higher frequencies, conventional lumped models are no longer adequate in describing the interconnect performance and distributed models become necessary. The major difficulty usually encountered while linking distributed transmission lines and nonlinear simulators is the problem of mixed frequency/time. This is because distributed elements are best characterized in the frequency-domain, whereas nonlinear components such as drivers and receivers are represented generally in the time-domain.

To address the above difficulties, several algorithms were proposed in the literature for macromodeling and transient analysis of high-speed circuits and interconnects. The common goal of these techniques is to transform the Telegrapher's equations describing the transmission lines, into a set of ordinary differential equations with appropriate time-delayed controlled sources that can be integrated with circuit simulators. Most of these methods use some form of decoupling algorithm to convert

the set of coupled partial differential equations describing the lines into a set of decoupled single modal equations. Subsequently, the line voltages and currents are obtained as a linear combination of modal variables. As a result, the interconnect stamp used by a time-domain circuit simulator is in the form of coupled ordinary differential equations. *This coupling is one of the major reasons for the excessive computational cost of simulating large multiconductor structures.* It has been shown that the average cost of simulating an N-coupled lines circuit is proportional to N^β , where $3 < \beta < 4$, which results in a prohibitively time-consuming simulation task compared to the simple case of simulating a single line.

In this thesis, new algorithms which address the computational complexity associated with the time-domain simulation of massively coupled interconnect circuits are developed. The new methods reduce the coupled simulation problem into a series of simulation steps, where each step is of complexity equivalent to that of simulating a single line. Advantages are that the computational cost of the new algorithms grows only *linearly* with the number of lines. In addition, the methods are highly parallelizable, thus array processors can be used to provide further significant reductions in the computational cost. Various numerical examples are provided which validate the accuracy and efficiency of all the proposed algorithms.

Acknowledgments

I would like to express my gratitude and appreciation to my supervisor, Professor Ram Achar, for his constant guidance, support and encouragement. I would also like to acknowledge my colleagues in the CAD group at Carleton University, for their continuous help and useful discussions. Lastly, I give a special thanks to my family, my husband Nino, my parents Michel and Mouna Nakhla, and my sister Meranda, for their endless support, love and strong encouragement.

Table of Contents

Abstract	iii
Acknowledgments	v
Table of Contents	vi
List of Tables	xi
List of Figures	xii
List of Abbreviations	xvii
List of Symbols	xix
1 Introduction	1
1.1 Background and Motivation	1
1.2 Contributions	5
1.3 Organization of the Thesis	7
2 Review of High-Speed Interconnect Macromodeling Algorithms	8
2.1 Problem Formulation	8
2.2 Conventional Uniform Lumped Segmentation	11
2.3 Method of Characteristics (MoC)	13
2.4 Matrix Rational Approximation (MRA)	17

2.5	MoC vs. MRA	18
2.6	DEPACT: Delay Extraction-Based Compact Macromodeling Algorithm	22
2.6.1	Overview of DEPACT	22
2.6.2	Application to Lines with Frequency-Dependent Parameters .	24
2.6.3	Estimation of the Order of Approximation	26
2.6.4	DEPACT Macromodel Realization	27
2.7	Summary	29
3	Simulation of Coupled Interconnects Using Waveform Relaxation and Transverse Partitioning	31
3.1	Introduction	31
3.2	Background and Previous Techniques	34
3.2.1	Longitudinal Partitioning	35
3.3	Basis of the Proposed Algorithm	39
3.4	Updating the Waveform Relaxation Sources	43
3.4.1	Single-Ended Representation	43
3.4.2	Distributed Representation	46
3.5	Implementation Considerations	52
3.6	Numerical Results	54
3.6.1	Example 1	55
3.6.2	Example 2	57
3.6.3	Example 3	62
3.6.4	Example 4	66
3.7	Summary	70
4	Waveform Relaxation Techniques for Simulation of Coupled Inter- connects with Frequency-Dependent Parameters	71
4.1	Introduction	71

4.2	Modeling of Frequency-Dependent Line Parameters	73
4.3	Development of the WR Algorithm for F.D. Parameters	78
4.4	Waveform Scaling Algorithm for Updating the Waveform Relaxation Sources	81
4.4.1	Waveform Scaling Approach	82
4.5	Implementation Considerations	88
4.5.1	Using Controlled Sources for Updating the Relaxation Sources	88
4.5.2	Convergence and Diagonal Dominance	91
4.6	Numerical Results	92
4.6.1	Example 1	92
4.6.2	Example 2	96
4.6.3	Example 3	98
4.6.4	Example 4	100
4.7	Summary	102
5	Simplified Delay Extraction-Based Passive Transmission Line Macromodels	103
5.1	Macromodels for Lines with Frequency-Independent Parameters	104
5.1.1	Realization of Single Resistive Sections	105
5.1.2	Realization of Coupled Resistive Sections	107
5.1.3	Alternative Realization of Lossy Coupled Sections	111
5.2	Macromodels for Lines with Frequency-Dependent $\mathbf{R}(s)$ and $\mathbf{L}(s)$ parameters	113
5.2.1	Case 1: Constant \mathbf{C} , and $\mathbf{G} = 0$	114
5.2.2	Case 2: Constant \mathbf{C} and \mathbf{G}	117
5.3	Numerical Results	121
5.3.1	Example 1	121

5.3.2	Example 2	123
5.3.3	Example 3	126
5.3.4	Example 4	130
5.3.5	Example 5	133
5.3.6	Example 6	135
5.4	Summary	139
6	A Delay Extraction-Based Algorithm for Simulation of Coupled Interconnects using Transverse Partitioning	140
6.1	Introduction	140
6.2	Transverse Partitioning of DEPACT Cells	142
6.3	Partitioning of Lossy Sections	142
6.4	Partitioning of Lossless Sections	147
6.4.1	Conventional Mode Decoupling	147
6.4.2	Closed-Form Evaluation of WR Sources for Lossless Lines	149
6.5	DEPACT+WR-TP Macromodel Realization	158
6.6	Numerical Results	159
6.6.1	Example 1	160
6.6.2	Example 2	162
6.7	Summary	163
7	A General Approach for Time-Domain Sensitivity Analysis of High-Speed Interconnects	164
7.1	Introduction	164
7.2	Sensitivity Analysis of Lumped Circuits	166
7.2.1	Sensitivity Circuits for RLC components	166
7.2.2	Sensitivity Circuits for Dependent Sources	170
7.2.3	Sensitivity Circuits for Nonlinear Components	174

7.3	Sensitivity Analysis of Transmission Lines	175
7.3.1	Companion Model for Transmission Lines	175
7.3.2	Computation of the Sensitivity Sources	178
7.3.3	Overall Sensitivity Circuit	182
7.3.4	Application to Waveform Relaxation with Transverse Partitioning	184
7.4	Computational Results	187
7.4.1	Example	187
7.5	Summary	189
8	Conclusions and Future Research	190
8.1	Summary	190
8.2	Future Research	192
	List of References	194

List of Tables

2.1	CPU Time Comparison	19
3.1	Results for Longitudinal Partitioning Example	39
3.2	CPU Time Comparison for Example 3	62
7.1	Sensitivity Analysis Computational Cost Comparison	187

List of Figures

2.1	Multiconductor transmission lines	9
2.2	Lumped transmission line model	12
2.3	MoC Interconnect model for a lossless single line	15
2.4	Single transmission line circuit [1]	19
2.5	Time-domain response (far end) for single line circuit in Figure 2.4 . .	20
2.6	Real and imaginary parts of $H(s)$ for the single TL in Figure 2.4 . . .	21
2.7	Error comparison between MLF-I and MLF-II	24
2.8	Relative approximation error versus DEFACT order m for varying bandwidths	27
2.9	DEFACT macromodel realization	29
3.1	Average CPU cost for varying number of lines using conventional cir- cuit simulators	34
3.2	Longitudinal partitioning of an interconnect circuit	36
3.3	Two-coupled line circuit	37
3.4	Two-coupled line circuit partitioned in the longitudinal direction . . .	38
3.5	Single-ended representation of decoupled network using transverse par- titioning	46
3.6	Distributed representation of decoupled network using transverse par- titioning	52
3.7	Transient response at near-end of victim line # 2 (Example 1)	56

3.8	Transient response at far-end of victim line # 3 (Example 1)	56
3.9	Coupled interconnect system with nonlinear terminations (Example 2)	58
3.10	Decoupled interconnect subcircuits for Example 2	59
3.11	Transient response of voltage at active line far-end after 2 iterations (Example 2)	60
3.12	Transient response of voltage at active line far-end after 3 iterations (Example 2)	60
3.13	Transient response of voltage at victim line far-end after 2 iterations (Example 2)	61
3.14	Transient response of voltage at victim line far-end after 3 iterations (Example 2)	61
3.15	Coupled line circuit for Example 3	63
3.16	Comparison between IFFT and HSPICE for far-end voltage of line #1	63
3.17	Comparison between IFFT and HSPICE for near-end voltage of line #2	64
3.18	Comparison between IFFT and proposed WR-TP for far-end voltage of line #1	64
3.19	Comparison between IFFT and proposed WR-TP for near-end voltage of line #2	65
3.20	Transient response at far-end of line #1	67
3.21	Transient response at near-end of line #2	68
3.22	CPU cost for varying number of lines using WR-TP	68
3.23	Comparison between CPU cost of proposed WR-TP method vs. con- ventional simulation techniques	69
4.1	Equivalent circuit representation for $Z_{j,k}(s)$	76
4.2	Equivalent circuit representation for $Y_{j,k}(s)$	77
4.3	Equivalent circuit representation for $\mathbf{Z}(s)$ using Laplace Elements	78

4.4	Distributed representation of decoupled network using transverse partitioning	81
4.5	Equivalent circuit demonstrating the computation of $\tilde{e}_j^{(r)}(x_p, t)$	86
4.6	Equivalent circuit demonstrating the computation of $\tilde{q}_j^{(r)}(x_p, t)$	87
4.7	Calculation of relaxation sources at iteration r using Laplace Elements	90
4.8	Nonlinear circuit with four on-chip coupled lines (Example 1)	93
4.9	Comparison of fitted results vs. original data for $Z_{1,1}(s)$	94
4.10	Transient response at far-end of active line (line #1) (Example 1) . .	94
4.11	Transient response at near-end of victim line (line #2) (Example 1) .	95
4.12	Linear circuit consisting of twelve on-chip coupled lines (Example 2) .	96
4.13	Transient response at near-end of victim line (line #3) (Example 2) .	97
4.14	Transient response at far-end of victim line (line #12) (Example 2) .	97
4.15	Four coupled lines circuit (Example 3)	98
4.16	Transient response at far-end of active line (line #3) (Example 3) . .	99
4.17	Transient response at near-end of victim line (line #4) (Example 3) .	99
4.18	Coupled lines circuit with varying N (Example 4)	101
4.19	Computational cost as a function of the number of lines N (Example 4)	101
5.1	Macromodel realization for a single line with frequency-independent parameters	107
5.2	Equivalent circuit representation for the analysis of resistive coupled sections	110
5.3	Equivalent circuit representation of (5.15) for $N = 2$	112
5.4	General algorithm for circuit realization of lossy sections described by (5.15)	113
5.5	Macromodel realization of the k th DEFACT cell for a single line with frequency-dependent $R(s)$, $L(s)$, constant C and $G = 0$	116

5.6	DEPACT cell with macromodel for coupled lossy sections described by (5.23)	117
5.7	Circuit realization for the DEPACT cell with constant G	120
5.8	Circuit realization for the coupled DEPACT cell with constant \mathbf{G}	120
5.9	Long (10m) cable transmission line circuit for Example 1	122
5.10	Transient response at far-end of cable line for Example 1	122
5.11	Multiconductor transmission line network with nonlinear terminations considered in Example 2	124
5.12	Transient response at near-end of victim line of MTL subnetwork #1	125
5.13	Transient response at far-end of active line of MTL subnetwork #2	125
5.14	Fitted p.u.l. resistance $R(s)$ as a function of frequency (Example 3)	127
5.15	Fitted p.u.l. inductance $L(s)$ as a function of frequency (Example 3)	127
5.16	Fitting results for $Z(s)$ for Example 3	128
5.17	Transient response at near-end of active line for Example 3	128
5.18	Transient response at far-end of active line for Example 3	129
5.19	Fitting results for $Z_{1,1}(s)$ for Example 4	131
5.20	Output transient waveforms for Example 4	132
5.21	Transient response at near-end of active line (Example 5)	134
5.22	Transient response at far-end of active line (Example 5)	134
5.23	Real and imaginary parts of the voltage at the near-end of active line	136
5.24	Real and imaginary parts of the voltage at the far-end of victim line	137
5.25	Transient response at far-end of active line (Example 6)	138
5.26	Transient response at near-end of victim line (Example 6)	138
6.1	Two coupled lossy lines	144
6.2	Equivalent circuit for the j th line of the decoupled lossy section	147
6.3	Equivalent circuit for the j th line of the decoupled lossless section	158
6.4	Overall decoupled DEPACT cell for line j	159

6.5	Voltage at far-end of active line (Example 1)	160
6.6	Voltage at near-end of victim line (Example 1)	161
6.7	Voltage at near-end of active line (Example 2)	162
6.8	Voltage at far-end of victim line (Example 2)	163
7.1	Companion model for resistors	167
7.2	Companion model for inductors	168
7.3	Companion model for capacitors	169
7.4	Companion model for voltage controlled current source (VCCS)	172
7.5	Companion model for voltage controlled voltage source (VCVS)	172
7.6	Companion model for current controlled current source (CCCS)	173
7.7	Companion model for current controlled voltage source (CCVS)	173
7.8	Companion model for nonlinear resistors	175
7.9	Equivalent sensitivity TL circuit	178
7.10	Example TL network and corresponding sensitivity circuit	183
7.11	Application of the WR-TP distributed representation to the sensitivity circuit in Figure 7.9	185
7.12	Application of the delay extraction based WR-TP algorithm to the sensitivity circuit in Figure 7.9	186
7.13	Sensitivity of voltage at far-end of victim line with respect to L_{11}	188

List of Abbreviations

<i>MCM</i>	Multi Chip Module	1
<i>PCB</i>	Printed Circuit Board	1
<i>TL</i>	Transmission Line	1
<i>MRA</i>	Matrix Rational Approximation	2
<i>MoC</i>	Method of Characteristics	2
<i>DEPACT</i>	Delay Extraction-based Passive Compact Macromodeling Algorithm	2
<i>p.u.l.</i>	Per Unit Length	3
<i>FD</i>	Frequency Dependent	5
<i>WR</i>	Waveform Relaxation	5
<i>TP</i>	Transverse Partitioning	5
<i>WR – TP</i>	Waveform Relaxation with Transverse Partitioning	5
<i>MOR</i>	Model Order Reduction	6
<i>MLF – I</i>	Modified Lie Formula-I	23
<i>MLF – II</i>	Modified Lie Formula-II	23
<i>FFT</i>	Fast Fourier Transform	45
<i>IFFT</i>	Inverse Fast Fourier Transform	45
<i>MNA</i>	Modified Nodal Analysis	111
<i>VCCS</i>	Voltage Controlled Current Source	170
<i>VCVS</i>	Voltage Controlled Voltage Source	170
<i>CCCS</i>	Current Controlled Current Source	171

<i>CCVS</i>	Current Controlled Voltage Source	171
-------------	---	-----

List of Symbols

t	Time
N	Number of coupled transmission lines
$\mathbf{R}, \mathbf{L}, \mathbf{G}, \mathbf{C}$	Resistance, Inductance, Conductance and Capacitance Matrices of the Transmission Line (TL)
x	Position along the TL
f	Frequency
s	Complex frequency
d	Length of the TL
$\begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix}, \begin{bmatrix} \mathbf{V}(d, s) \\ \mathbf{I}(d, s) \end{bmatrix}$	Terminal voltages/currents of TL in the frequency domain, at the near-end and far-end, respectively.
Λ	Wavelength
v_p	Velocity of propagation
τ	Delay of the TL
Γ	Propagation constant of the TL
\mathbf{Z}_c	Characteristic impedance of the TL
\mathbf{Q}_k	Product terms in MLF approximations
m	DEFACT order of approximation

ε_m	DEPACT approximation error
L_i	Inductance due to internal magnetic field
L_e	Inductance due to magnetic field external to the conductor
L_∞	Inductance value at $s = \infty$
$\mathbf{R}_m, \mathbf{G}_m$	Resistance and conductance of resistive section in MLF-II approximations
$\mathbf{P}_I, \mathbf{P}_V$	Transformation matrices for lossy decoupling
d_m	Length of middle section in DEPACT cell
l_m	Length of outer section of DEPACT cell
l_0	Near-end of lossless section of DEPACT cell
T	Transpose operator
$\bar{\mathbf{Y}}$	Y-parameter matrix of TL
$\tilde{\mathbf{Y}}$	Re-arranged Y-parameter matrix of TL
$\mathbf{Z}(s), \mathbf{Y}(s)$	P.u.l. impedance and admittance parameters of the TL
Φ_i	Products in MLF approximation for lossy sub-cells
p	Number of lossy sub-cells within a single DEPACT cell
d_p	Length of lossy sub-cell
$\delta_j(d, s)$	WR sources for single-ended representation using WR-TP
n	Number of discretizations used in distributed WR-TP approach
$\tilde{e}_j(x_p, t), \tilde{q}_j(x_p, t)$	WR sources for distributed approach
$\varphi_j(t)$	Relaxation sources for lossy section of DEPACT cell in Chapter 6
$\mathbf{T}_I, \mathbf{T}_V$	Modal transformation matrices for lossless decoupling
\mathbf{U}_k	Selector matrix
$u(t)$	Unit step function
$v_{src_j}(d_m, t), i_{src_j}(d_m, t)$	Voltage/current WR sources for lossless sections in Chapter 6

Chapter 1

Introduction

1.1 Background and Motivation

The rapid increase in operating speeds, density and complexity of modern integrated circuits has made interconnect analysis a requirement for all state-of-the-art circuit simulators. Interconnect effects such as ringing, signal delay, distortion and crosstalk can severely degrade signal integrity. Interconnects are present at various levels of design hierarchy, such as on-chip, packaging structures, multi-chip modules (MCMs), printed circuit boards (PCBs), backplanes and cables. This warrants different modeling strategies based on the hierarchy and the operating frequency. At relatively lower frequencies, the interconnect can be modeled using lumped elements, such as resistors, capacitors and inductors. However, as the frequency of operation increases, the interconnect lengths become a significant fraction of the operating wavelength, and conventional lumped models become inadequate in describing the interconnect performance and transmission line (TL) models become necessary. Skin and proximity effects also become prominent at higher frequencies and distributed models with frequency-dependent parameters may be needed [2].

The major difficulty usually encountered while linking distributed transmission lines

and nonlinear simulators is the problem of mixed frequency/time [2, 3]. This is because distributed elements are best characterized in the frequency-domain, whereas nonlinear components such as drivers and receivers are represented generally in the time-domain. Macromodeling techniques attempt at solving this problem by representing the interconnect in the form of a set of ordinary differential equations, which can then be directly linked to nonlinear circuit simulators (such as HSPICE [4]). The goal of these techniques is to develop a general model (referred to here as the macromodel) suitable for nonlinear simulation, which enables the circuit simulator to view the distributed interconnect as a regular circuit component.

Several publications can be found in the literature which try to address the issues described above [2, 3, 5–9]. One approach, referred to as conventional lumped segmentation provides a brute force solution to the problem of mixed frequency/time simulation. However, these type of methods lead to large circuit matrices, rendering the simulation inefficient [9].

There are other more advanced methods in the literature developed for interconnect analysis. These algorithms include the method of characteristics (MoC) [10, 11], matrix rational approximation (MRA) [12–14] and the delay extraction-based passive compact macromodeling algorithm (DEPACT) [5, 15]. MoC based algorithms provide fast solutions for long low-loss lines, and ensure the transmission line causality [16] (which implies that an excitation signal entering one end of a transmission line segment will appear at the other end only after the time-of-flight delay). Although MoC based approaches are suitable for lossless lines, they have difficulty modeling lossy interconnects. In addition, one of the main disadvantages of MoC methods is that they do not guarantee the passivity of the resulting macromodels. Passivity

of the macromodel is essential, since non-passive (even if they are stable) models when coupled with arbitrary nonlinear elements, can lead to unstable systems [12].

On the other hand, algorithms based on MRA guarantee the macromodel passivity. The MRA technique approximates the exponential stamp of the transmission line with closed-form rational functions using pre-determined coefficients, which are computed as a function of the per-unit-length (p.u.l.) parameters of the line. However, for long lines, MRA requires high-order approximations (due to its inefficiency in capturing the flat portion corresponding to the delay of the line), which leads to inefficient transient simulation [5]. This limits its usefulness to short lines (such as on-chip interconnects).

To address the above issues, the DEPACT algorithm was developed [5, 15]. DEPACT combines the merits of both the MoC and MRA algorithms, and is based on delay extraction [5] prior to performing the MRA. This results in significantly lower-order macromodels for long lossy-coupled lines, leading to fast transient simulation for both on-chip and off-chip interconnects. One of the main advantages is that the resulting DEPACT macromodel is guaranteed to be passive by construction.

The common goal of the techniques described above is to transform the Telegrapher's equations describing the transmission lines, into a set of ordinary differential equations with appropriate time-delayed controlled sources that can be integrated with circuit simulators. Most of these methods use some form of decoupling algorithm to convert the set of coupled partial differential equations describing the lines into a set of decoupled single modal equations. Subsequently, the line voltages and currents are obtained as a linear combination of modal variables. As a result, the interconnect stamp used by the time-domain circuit simulator is in the form of coupled ordinary

differential equations. *This coupling is one of the major reasons for the excessive computational cost of simulating large multiconductor structures.* This increased computational cost occurs at two stages; the first is during the formulation of the interconnect stamp and the second, which is more costly, is during the transient simulation of the signal path including the nonlinear terminations. For example, the y -parameters of a set of 64-coupled lines require approximation of 8320 different transfer functions. These transfer functions not only have to accurately match the frequency response of the line over the desired frequency bandwidth, but they also have to satisfy the challenging constraints of preserving the passivity and causality of the macromodel [13]. In the next stage, these transfer functions have to be converted to a set of ordinary differential equations or alternatively into an equivalent circuit to be interfaced with other circuit elements, and/or nonlinear driver/receiver circuits. The complexity of the equivalent circuit grows approximately at a rate of N^2 , where N is the number of coupled lines. Moreover, the average CPU expense of simulating a circuit of size M is proportional to M^α , where $1.5 < \alpha < 2$, (depending on the sparsity of the circuit). Consequently, the average cost of simulating an N -coupled lines circuit is proportional to N^β , where $3 < \beta < 4$, which results in a prohibitively time-consuming simulation task compared to the simple case of simulating a single line.

The goal of this thesis is to develop new algorithms which address the computational complexity of the time-domain simulation of large coupled interconnect circuits. The specific contributions are listed below.

1.2 Contributions

1. A new algorithm based on waveform relaxation with transverse partitioning (WR-TP) has been developed [17, 18]. The new technique reduces the coupled simulation problem into a series of simulation steps, where each step is of complexity equivalent to that of simulating a single line. The computational cost of the new method grows only linearly with the number of lines. Various approaches for updating the WR sources have also been developed. In addition, the WR-TP algorithm is highly parallelizable. Thus, array processors can be used to effectively simulate each subcircuit in parallel, which would provide further reduction in the computational cost.
2. A new WR-TP algorithm applied to the practically important case of interconnects with frequency-dependent (FD) line parameters has been developed [19]. In this approach, the resulting rational function based approximations of the FD parameters are used to generate equivalent macromodels compatible with SPICE-like circuit simulators. An efficient waveform scaling algorithm for updating the relaxation sources has been developed. Also, several relevant implementation considerations are presented.
3. New simplified delay extraction-based macromodels have been developed for several practical cases. These new methods do not require any MRA-based approximations. Specifically, for lines with frequency-independent parameters, the final DEFACT cells are realized as exact implementations (without any approximations) in terms of lossless lines and lumped elements. For lines with FD resistance and inductance, a numerical fitting algorithm is used for modeling the line parameters and the resulting macromodels are realized using equivalent circuits and lossless lines.

4. Based on the simplified delay extraction-based macromodel, a new waveform relaxation with transverse partitioning algorithm has been developed [20]. In this approach, each section of the DEPACT macromodel is partitioned separately in the transverse direction, into single line subcircuits with independent WR sources. Advantages are that the WR sources are computed analytically in a closed-form, directly in the time-domain. In addition, since the algorithm employs an efficient delay extraction technique, it is suitable for both on and off-chip interconnects.
5. A general algorithm for the time-domain sensitivity analysis of high-speed interconnects has been developed [20]. One of the main advantages of the new sensitivity algorithm is that it is independent of the specific macromodeling technique used to represent the transmission lines. In addition, a generic and efficient approach for computing the sensitivity sources was developed.
6. A new WR-TP algorithm based on model order reduction (MOR) macromodels has been proposed for simulation of large multi-port networks [21, 22].
7. The WR-TP algorithms opened the door for further research which is collaboratively being carried out with other members of the CAD group.

Specifically:

- (a) The WR-TP algorithm has been advanced for simulation of coupled lines in the presence of external EM interference (EMI) [23, 24].
- (b) Based on WR-TP, a new parallel algorithm has been developed and implemented on two multiprocessing platforms (an Intel Xeon UMA system and an AMD Opteron ccNUMA system) [25].

1.3 Organization of the Thesis

This thesis is organized as follows. Chapter 2 presents an overview of previous high-speed interconnect macromodeling algorithms. Chapter 3 provides details of the new waveform relaxation and transverse partitioning algorithm (WR-TP). In Chapter 4, a description of the WR-TP algorithm for lines with FD parameters is presented. Chapter 5 describes the new simplified delay extraction-based macromodeling algorithm. In Chapter 6, the proposed delay extraction-based WR-TP algorithm is presented. Chapter 7 describes a new general method for time-domain sensitivity analysis of transmission lines. Finally, conclusions and future research are provided in Chapter 8.

Chapter 2

Review of High-Speed Interconnect

Macromodeling Algorithms

The simulation of high-speed interconnects requires selecting a suitable modeling and solution algorithm, which typically depend on the application of interest. There are several techniques in the literature, each offering different challenges to circuit simulators. This chapter provides an overview of some of the relevant methods currently available for interconnect analysis. The chapter is organized as follows. Section 2.1 presents the problem formulation. Sections 2.2, 2.3 and 2.4 describe three techniques: conventional lumped segmentation [9], the method of characteristics (MoC) [10, 26] and matrix rational approximation (MRA) [12, 13], respectively. Section 2.5 provides a comparison between MoC and MRA and discusses the significance of delay extraction. Finally, Section 2.6 describes the DEPACT algorithm [5, 15].

2.1 Problem Formulation

Consider the distributed transmission lines shown in Figure 2.1. The transmission lines can be represented by Telegrapher's equations as [9]

$$\begin{aligned}
\frac{\partial}{\partial x} \mathbf{v}(x, t) &= -\mathbf{R}\mathbf{i}(x, t) - \mathbf{L} \frac{\partial}{\partial t} \mathbf{i}(x, t) \\
\frac{\partial}{\partial x} \mathbf{i}(x, t) &= -\mathbf{G}\mathbf{v}(x, t) - \mathbf{C} \frac{\partial}{\partial t} \mathbf{v}(x, t)
\end{aligned} \tag{2.1}$$

where $\mathbf{R} \in \mathfrak{R}^{N \times N}$, $\mathbf{L} \in \mathfrak{R}^{N \times N}$, $\mathbf{C} \in \mathfrak{R}^{N \times N}$, and $\mathbf{G} \in \mathfrak{R}^{N \times N}$ are the per-unit-length (p.u.l.) resistance, inductance, capacitance and conductance parameter matrices of the transmission lines, $\mathbf{v}(x, t) \in \mathfrak{R}^N$ and $\mathbf{i}(x, t) \in \mathfrak{R}^N$ represent the voltage and current vectors as a function of position x and time t , and N is the number of coupled lines.

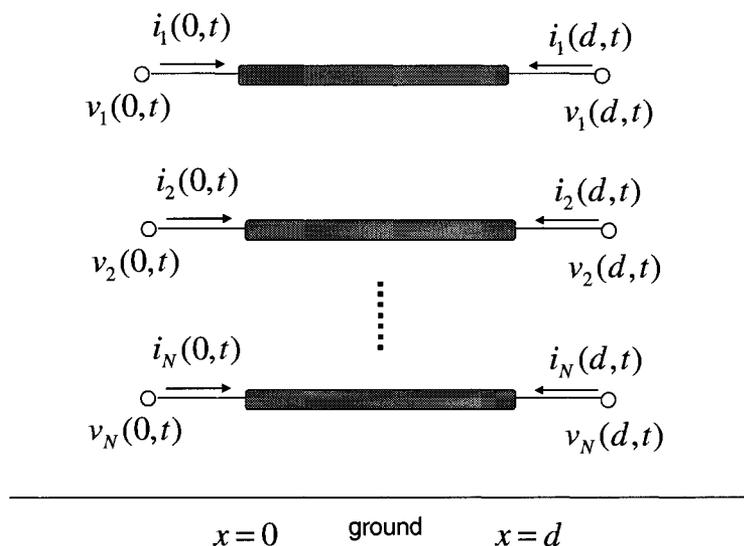


Figure 2.1: Multiconductor transmission lines

Next, the solution of (2.1) can be expressed in the matrix-exponential form, in the frequency-domain as

$$\begin{bmatrix} \mathbf{V}(d, s) \\ -\mathbf{I}(d, s) \end{bmatrix} = e^{\mathbf{A}+s\mathbf{B}} \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \quad (2.2)$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & -\mathbf{R} \\ -\mathbf{G} & \mathbf{0} \end{bmatrix} d; \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} & -\mathbf{L} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} d \quad (2.3)$$

and d is the length of the line.

In order to include transmission lines in nonlinear circuit simulators, such as HSPICE [4], (2.2) must be expressed in the time-domain. However, the exponential matrix $e^{\mathbf{A}+s\mathbf{B}}$ is best characterized in the frequency-domain, thus making it difficult to interface with nonlinear circuit simulators. In the literature, this problem is referred to as the ‘mixed frequency/time’ problem [3].

To overcome this problem, the Telegrapher’s equations, which are in the form of *partial* differential equations, must be transformed into *ordinary* differential equations, which can then be linked to nonlinear simulators [3]. Approximating the exponential stamp of the line in the form of a set of ordinary differential equations is referred to as ‘macromodeling’, and the new circuit representing the transmission line is referred to as a ‘macromodel’. Several publications can be found in the literature which try to address the problem of mixed frequency/time through macromodeling [3,5–8]. Among the most relevant techniques are conventional lumped segmentation [3], the method of characteristics (MoC) [10,26], matrix rational approximation (MRA) [12–14] and DEPACT [5,15]. The following sections provide a brief overview of these methods.

2.2 Conventional Uniform Lumped Segmentation

In this conventional approach, lumped equivalent circuits are used to represent the transmission line. In this case, the partial differential equations in (2.1) are approximated by the following difference equations [9]

$$\begin{aligned} \mathbf{v}_{k+1}(t) - \mathbf{v}_k(t) &\cong -\mathbf{R}\Delta x \mathbf{i}_k(t) - \mathbf{L}\Delta x \frac{\partial}{\partial t} \mathbf{i}_k(t) \\ \mathbf{i}_{k+1}(t) - \mathbf{i}_k(t) &\cong -\mathbf{G}\Delta x \mathbf{v}_k(t) - \mathbf{C}\Delta x \frac{\partial}{\partial t} \mathbf{v}_k(t) \end{aligned} \quad (2.4)$$

where the line is divided into M segments, each of length $\Delta x = d/M$ and $k = 1, 2, \dots, M - 1$.

If each of these segments is electrically small at the frequencies of interest, i.e. $\Delta x \ll \Lambda$, where $\Lambda = v_p/f$ is the wavelength, v_p is the velocity of propagation and f is the frequency, then each of the M segments can be replaced by lumped models. Figure 2.2 shows the general structure of a lumped model used to discretize a single transmission line, where R, L, G and C are the p.u.l. line parameters. The lumped segmentation model can also be extended to coupled transmission lines as presented in [9].

It is often of practical interest to estimate how many lumped segments (M) are required to accurately represent the distributed transmission line. A general rule of thumb is given by the following relation [3]

$$M \geq 10\tau/t_r \quad (2.5)$$

where $\tau = d\sqrt{LC}$ is the delay of the line and t_r is the rise time of the propagating

signal.

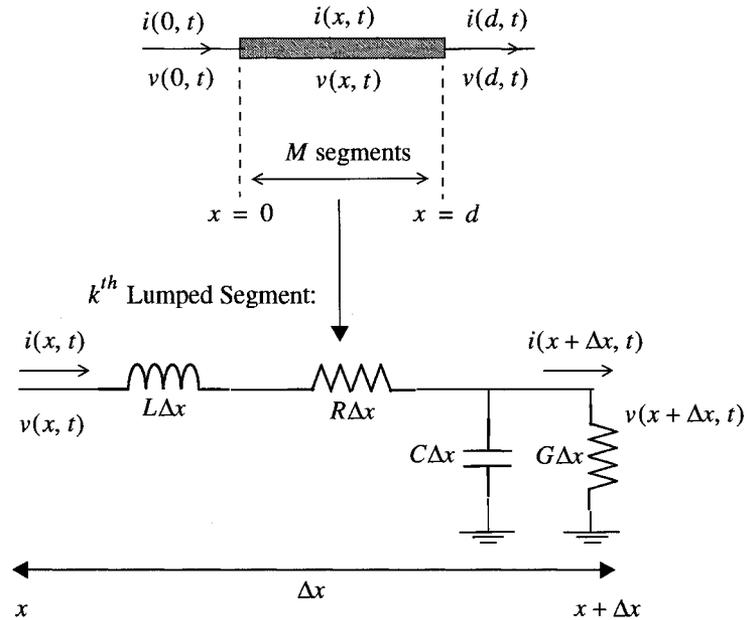


Figure 2.2: Lumped transmission line model

Although conventional lumped segmentation provides a direct approach to solving the problem of mixed frequency/time, the approximation is only valid if Δx is chosen to be a small fraction of the wavelength Λ . If the frequency of interest is high, or if the interconnect is electrically long (i.e. $d > 0.1\Lambda$), many lumped segments are required. This leads to large circuit matrices, rendering the simulation inefficient [9]. In addition, conventional lumped segmentation has difficulty when trying to handle a large number of coupled lines, or lines with frequency-dependent parameters.

2.3 Method of Characteristics (MoC)

The method of characteristics (MoC) transforms the partial differential equations of the transmission line into ordinary differential equations. Consider the Telegrapher's equations for a single transmission line in the frequency-domain as

$$\begin{aligned}\frac{\partial}{\partial x}V(x, s) &= -(R + sL)I(x, s) \\ \frac{\partial}{\partial x}I(x, s) &= -(G + sC)V(x, s)\end{aligned}\quad (2.6)$$

An analytical solution for (2.6), in terms of the y -parameters can be derived as [3]

$$\begin{bmatrix} I_0 \\ I_d \end{bmatrix} = \frac{1}{Z_c(1 - e^{-2\Gamma d})} \begin{bmatrix} 1 + e^{-2\Gamma d} & -2e^{-\Gamma d} \\ -2e^{-\Gamma d} & 1 + e^{-2\Gamma d} \end{bmatrix} \begin{bmatrix} V_0 \\ V_d \end{bmatrix}\quad (2.7)$$

where the propagation constant Γ and the characteristic impedance Z_c are given by

$$\Gamma = \sqrt{(R + sL)(G + sC)}; \quad Z_c = \sqrt{\frac{(R + sL)}{(G + sC)}}\quad (2.8)$$

and where V_0 , I_0 and V_d , I_d represent the near- and far-end voltages and currents of the line, respectively.

The y -parameters in (2.7) are a complex function of frequency, and in most cases, cannot be directly transformed into ordinary differential equations in the time-domain. However, MoC succeeds in doing such a transformation, but only for lossless lines [27]. The terms in (2.7) can be re-arranged as follows

$$\begin{aligned}
V_0 &= Z_c I_0 + e^{-\Gamma d} [2V_d - e^{-\Gamma d} (Z_c I_0 + V_0)] \\
V_d &= Z_c I_d + e^{-\Gamma d} [2V_0 - e^{-\Gamma d} (Z_c I_d + V_d)]
\end{aligned} \tag{2.9}$$

Next, (2.9) can be re-written as

$$\begin{aligned}
V_0 - Z_c I_0 &= W_1 \\
V_d - Z_c I_d &= W_2
\end{aligned} \tag{2.10}$$

where

$$\begin{aligned}
W_1 &= e^{-\Gamma d} [2V_d - e^{-\Gamma d} (Z_c I_0 + V_0)] \\
W_2 &= e^{-\Gamma d} [2V_0 - e^{-\Gamma d} (Z_c I_d + V_d)]
\end{aligned} \tag{2.11}$$

Using (2.9)-(2.11), a recursive relationship for W_1 and W_2 can be expressed as [3]

$$\begin{aligned}
W_1 &= e^{-\Gamma d} [2V_d - W_2] \\
W_2 &= e^{-\Gamma d} [2V_0 - W_1]
\end{aligned} \tag{2.12}$$

If the transmission line is lossless, the propagation constant and characteristic impedance simplify to

$$\Gamma = s\sqrt{LC}; \quad Z_c = \sqrt{\frac{L}{C}} \tag{2.13}$$

This results in Γ being purely imaginary and Z_c a real constant. Thus, (2.12) can be written in the time-domain by replacing $e^{-\Gamma d}$ with a time-delay as

$$\begin{aligned} w_1(t + \tau) &= 2v_d(t) - w_2(t) \\ w_2(t + \tau) &= 2v_0(t) - w_1(t) \end{aligned} \quad (2.14)$$

where $\tau = d\sqrt{LC}$. In this case, the lossless transmission line can be modeled using lumped impedances and time-delayed voltage controlled voltage sources. The equivalent circuit for modeling of a lossless single line using MoC is shown in Figure 2.3. Using this approach, the model for lossless lines can be directly linked to nonlinear circuit simulators.

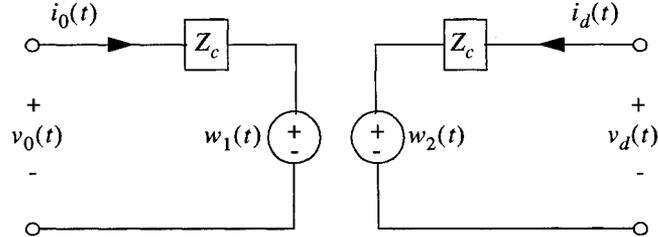


Figure 2.3: MoC Interconnect model for a lossless single line

For lossy lines, the propagation constant is not purely imaginary and thus, cannot be replaced with a time delay. In this case, analytical expressions for $w_1(t)$ and $w_2(t)$ cannot be found in the time-domain. To address this issue, several numerical techniques have been proposed in the literature to approximate Γ and Z_c in terms of rational functions in the frequency-domain [10, 16, 26]. To illustrate, consider the function $H(s) = e^{-\Gamma d}$ in (2.12). Using MoC, $H(s)$ is obtained by extracting the delay

$\tau = d\sqrt{LC}$, and approximating the remaining portion with a rational function [26], i.e.

$$\begin{aligned} H(s) &= e^{-\Gamma d} \\ &\cong e^{-s\tau} P(s) \end{aligned} \tag{2.15}$$

where $P(s)$ is a rational function corresponding to the delayless propagation operator.

In the case of lines with frequency-dependent parameters, the line delay in (2.15) is defined by the infinite frequency asymptotic values of inductance and capacitance as [26]

$$\tau = d\sqrt{L_\infty C_\infty} \tag{2.16}$$

The MoC formulations described in the above paragraphs for lossless and lossy single lines can be extended for the case of coupled transmission lines as presented in [9, 26].

Although MoC succeeds in transforming the Telegrapher's equations to ordinary differential equations, one of the main disadvantages of MoC is that the resulting macromodels are not guaranteed to be passive. Passivity is essential, since nonpassive but stable macromodels when coupled with arbitrary nonlinear elements can lead to unstable systems. These passivity violations present themselves as spurious oscillations in the transient responses or as simulation errors (by circuit simulators) [3]. In addition, the numerical fitting algorithms used by MoC can be computationally expensive and sometimes unreliable, particularly for lines with frequency-dependent parameters or when simulating a large number of coupled lines.

2.4 Matrix Rational Approximation (MRA)

In order to address the issue of passivity, passive matrix rational approximation (MRA) was suggested [6, 12–14, 28]. MRA approximates the exponential stamp of the line in (2.2) (also referred to as the hybrid parameters) with closed-form rational functions using pre-determined coefficients. Using MRA, the exponential matrix in (2.2) is expressed as

$$\mathbf{P}_{N_1, N_2}(\mathbf{Z})e^{\mathbf{Z}} \approx \mathbf{Q}_{N_1, N_2}(\mathbf{Z}) \quad (2.17)$$

where $\mathbf{P}_{N_1, N_2}(\mathbf{Z}), \mathbf{Q}_{N_1, N_2}(\mathbf{Z})$ are polynomial matrices expressed in terms of closed-form Padé rational functions defined as [28]

$$\begin{aligned} \mathbf{P}_{N_1, N_2}(\mathbf{Z}) &= \sum_{j=0}^{N_1} \frac{(N_1 + N_2 - j)!(N_1)!}{(N_1 + N_2)!j!(N_1 - j)!} (-\mathbf{Z})^j \\ \mathbf{Q}_{N_1, N_2}(\mathbf{Z}) &= \sum_{j=0}^{N_2} \frac{(N_1 + N_2 - j)!(N_2)!}{(N_1 + N_2)!j!(N_2 - j)!} (\mathbf{Z})^j \end{aligned} \quad (2.18)$$

To preserve the macromodel passivity, we let $N_1 = N_2 = n$, where n is the MRA approximation order. Using $\mathbf{Z} = (\mathbf{A} + s\mathbf{B})$, the exponential matrix in (2.2) can be expressed as

$$\mathbf{P}_{n, n}(\mathbf{Z})e^{\mathbf{Z}} \approx \mathbf{Q}_{n, n}(\mathbf{Z}) \quad (2.19)$$

where $\mathbf{P}_{n, n}(s), \mathbf{Q}_{n, n}(s)$ are polynomial matrices given by

$$\mathbf{P}_{n, n}(s) = \sum_{i=0}^n p_i s^i; \quad \mathbf{Q}_{n, n}(s) = \sum_{i=0}^n q_i s^i \quad (2.20)$$

The Padé rational functions in (2.19) can also be represented in an alternative form in terms of subsections, characterized by pole-zero pairs, as described in [6].

The above approximations are formulated analytically in terms of p.u.l. parameters and constants (i.e. p_i and q_i). These constants are obtained *a priori*, using methods such as Padé approximation or minimax optimization [12], while ensuring macromodel passivity. Using (2.19),(2.20) and the formulations in [12], (2.2) can be translated into a set of ordinary differential equations, which can be directly linked to nonlinear simulators [3,29]. One of the main advantages of MRA is that it is passive by construction. As previously mentioned, passivity is essential for macromodeling of high-speed interconnects. In addition, due to its closed-form nature, MRA avoids computationally expensive numerical fitting algorithms associated with MoC.

2.5 MoC vs. MRA

The accuracy of the MRA approximation depends on the norm of the term $\mathbf{A} + s\mathbf{B}$, which is a function of the line length d . In particular, the smaller the value of d , i.e. the shorter the line, the better the approximation. For large delay lines (e.g. long lines with small losses), high-order approximations are required, which lead to inefficient transient simulation [30]. To illustrate this point, consider the single line circuit shown in Figure 2.4.

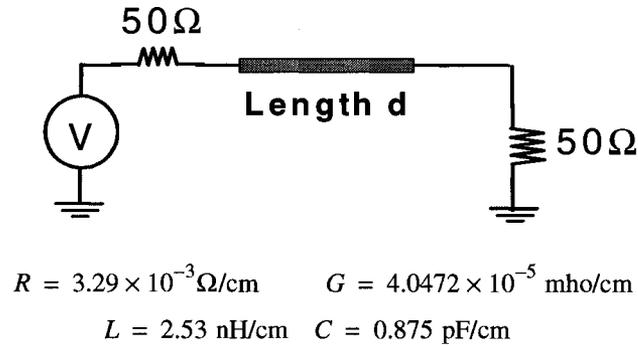


Figure 2.4: Single transmission line circuit [1]

The circuit in Figure 2.4 was simulated using both MRA and MoC. The results from both methods were in good agreement, however, the CPU time varied depending on the length of the line, as shown in Table 2.1.

Table 2.1: CPU Time Comparison

	MoC (sec)	MRA (sec)	Speed up of MRA
d=0.5cm	2.1	0.19	11x faster
d=10m	4	463	100x slower

Observing Table 2.1, we can see that, for short lines, MRA is fast and efficient. However, as the length of the line is increased, for example $d=10\text{m}$, MRA was more than 100 times slower than MoC. The reason for this can be seen in the time-domain response shown in Figure 2.5.

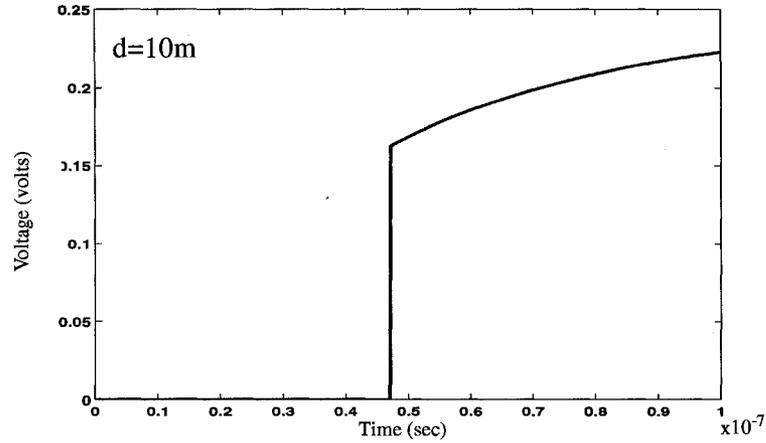
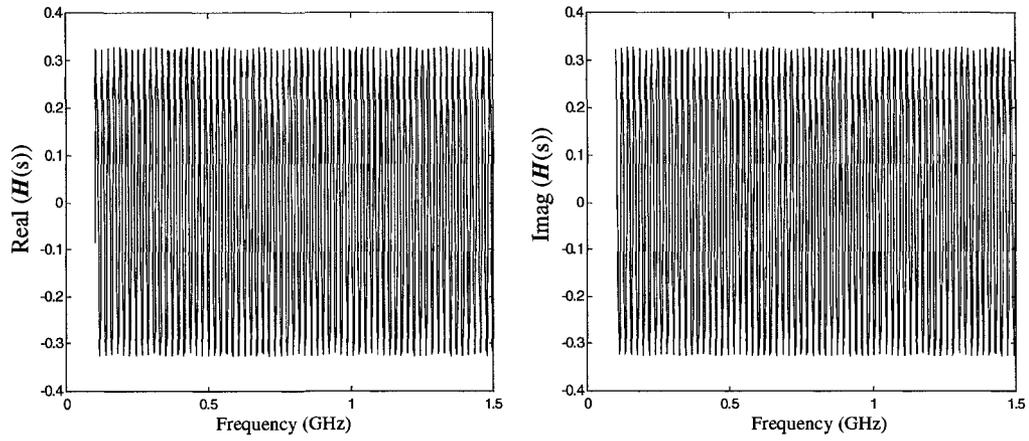


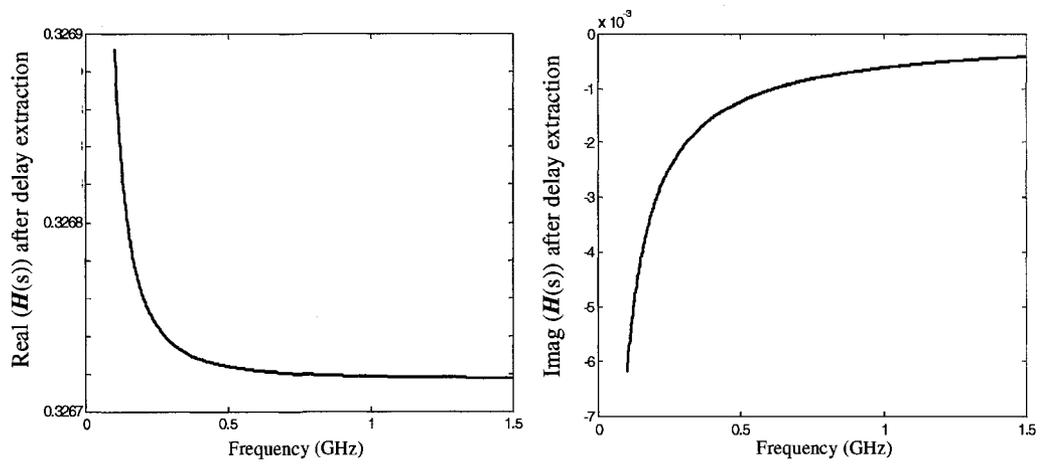
Figure 2.5: Time-domain response (far end) for single line circuit in Figure 2.4

The graph in Figure 2.5 shows that there is a long delay present in the time-domain response. Since MRA tries to capture this flat delay portion in terms of sums of exponentials, it requires a very high order rational function approximation, which in turn, results in inefficient transient simulation.

The effect of the delay can be best demonstrated in the frequency-domain. Figure 2.6(a) shows the real and imaginary parts of the propagation operator $H(s)$, (defined in (2.15)), for the line shown in Figure 2.4. It is obvious from the plot that approximating this function would require a very high order (due to the rapid oscillatory nature of the response). On the other hand, Figure 2.6(b) shows the real and imaginary parts of the propagation operator *after* delay extraction. As can be seen, approximating such a delay extracted waveform would require a rational function of a much lower order.



(a) Before delay extraction



(b) After delay extraction

Figure 2.6: Real and imaginary parts of $H(s)$ for the single TL in Figure 2.4

Therefore, it is evident that, although MRA guarantees the passivity of the macromodel and derives its accuracy and efficiency from the closed-form nature of the pre-determined coefficients, it does not employ any type of delay extraction prior to performing the rational function approximation. Thus, in the case of long lines, MRA requires high-order approximations due to the presence of the delay. This limits its usefulness to short lines (on-chip interconnects). On the other hand, MoC does employ a delay extraction mechanism, however, it does not guarantee the passivity of the resulting macromodel. In addition, the numerical fitting techniques used by MoC can be computationally expensive and in some cases unreliable. To address these issues, the delay extraction-based algorithm DEPACT was proposed in [5]. An overview of the DEPACT algorithm is provided in the next section.

2.6 DEPACT: Delay Extraction-Based Compact Macromodeling Algorithm

In order to address the issues described in the previous sections, a delay extraction-based passive macromodeling algorithm (DEPACT) was proposed [5]. DEPACT combines the merits of both the MoC and MRA algorithms. Using DEPACT, an efficient mechanism for delay extraction is employed prior to performing MRA, which results in significantly lower order macromodels. In addition, the DEPACT macromodel is guaranteed to be passive. This section presents an overview of the DEPACT algorithm and describes the structure of the resulting macromodel.

2.6.1 Overview of DEPACT

Using the DEPACT algorithm, the exponential stamp of line in (2.2) is approximated using a product of exponential terms. This is referred to as the *Modified Lie Formula-I*

(MLF-I) [5], and is given by

$$e^{A+sB} \approx \prod_{k=1}^m Q_k + \varepsilon_m \quad (2.21)$$

where

$$Q_k \equiv e^{\frac{A}{2m}} e^{\frac{sB}{m}} e^{\frac{A}{2m}} \quad (2.22)$$

and where the matrices A and B are given in (2.3) and m is the DEFACT approximation order.

In this case, the associated error (ε_m) is given by

$$\|\varepsilon_m\| = \max_{0 \leq s \leq s_{max}} \|e^{A+sB} - \prod_{k=1}^m Q_k\| \cong O\left(\frac{1}{m^2}\right) \quad (2.23)$$

Since the delay is inherent in the matrix B (defined in (2.3)), the inner exponential matrix in (2.22) represents the extracted delay portion of the transmission line.

In the case of long low-loss lines, it can be noted that $\|A\| \ll \|sB\|$. This fact is used to arrive at a more accurate relationship referred to as the *Modified Lie Formula-II* (MLF-II) [5], which is given by

$$e^{A+sB} \approx \prod_{k=1}^m Q_k + \varepsilon_m; \quad Q_k \equiv e^{\frac{sB}{2m}} e^{\frac{A}{m}} e^{\frac{sB}{2m}} \quad (2.24)$$

Similarly, the outer exponential matrices of Q_k in (2.24) represent the extracted delay portion of the line.

For illustration purposes, Figure 2.7 shows the norm of the absolute error as a function of the approximation order m , obtained using MLF-I and MLF-II, for a line with the following parameters: $R = 2 \Omega/cm$, $L = 3.36 nH/cm$, $C = 1.29 pF/cm$, $G = 5 \times 10^{-9} \mathcal{U}/cm$, and $d = 5cm$.

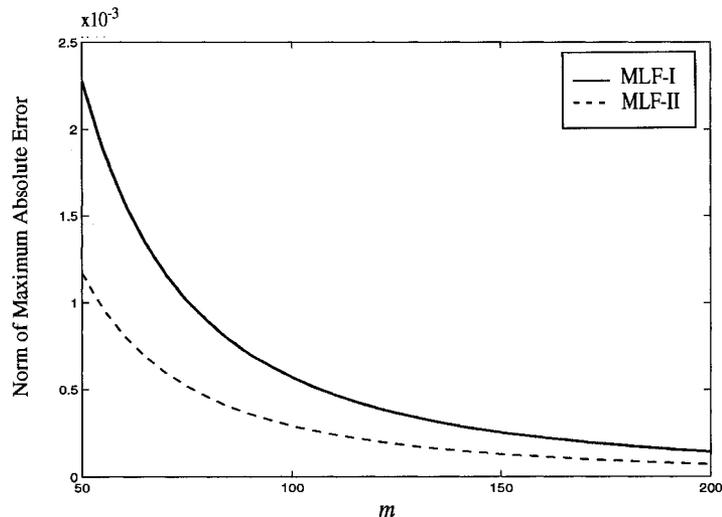


Figure 2.7: Error comparison between MLF-I and MLF-II

2.6.2 Application to Lines with Frequency-Dependent Parameters

As the frequency increases, the current distribution in conductors vary, which causes the p.u.l. resistance (\mathbf{R}) and inductance (\mathbf{L}) to change. The p.u.l. inductance matrix $\mathbf{L}(s)$ can be separated into a portion $\mathbf{L}_i(s)$, due to the internal magnetic field, and a portion $\mathbf{L}_e(s)$, due to the magnetic field external to the conductor [9]. Thus, the total inductance $\mathbf{L}(s)$ can be expressed as

$$\mathbf{L}(s) = \mathbf{L}_i(s) + \mathbf{L}_e(s) \quad (2.25)$$

At lower frequencies, the resistance and internal inductance remain almost constant, since the current may be assumed to be uniformly distributed throughout the cross-section of the conductor. On the other hand, at higher frequencies, the resistance increases approximately as \sqrt{f} , and the internal inductance $\mathbf{L}_i(s)$ decreases approximately with \sqrt{f} , since the current begins to concentrate in a thin layer at the outer edges of the conductor [9]. This is referred to as the *skin effect*. Thus, at higher frequencies, the internal inductance $\mathbf{L}_i(s)$ becomes negligible, and the remaining inductance $\mathbf{L} = \mathbf{L}_e$ essentially remains constant, representing the inductance at high frequencies. In addition, for many practical applications, the p.u.l. capacitance \mathbf{C} , and admittance \mathbf{G} , do not vary significantly with frequency [9].

Thus, in the case of lines with frequency-dependent parameters, the objective is to extract a maximum delay without affecting transmission line causality conditions. Here, the MLF relations are easily adjusted to account for the variations in frequency. For example, the relations in (2.24) are modified as

$$e^{\mathbf{A}(s)+s\mathbf{B}(s)} \approx \prod_{k=1}^m \mathbf{Q}_k + \varepsilon_m; \quad \mathbf{Q}_k \equiv e^{\frac{s\mathbf{B}_\infty}{2m}} e^{\frac{\mathbf{A}(s)+s\mathbf{B}_a(s)}{m}} e^{\frac{s\mathbf{B}_\infty}{2m}} \quad (2.26)$$

where

$$\mathbf{B}_a(s) = \mathbf{B}(s) - \mathbf{B}_\infty = \begin{bmatrix} \mathbf{0} & -\mathbf{L}_a(s) \\ -\mathbf{C}_a(s) & \mathbf{0} \end{bmatrix} \quad (2.27)$$

such that $\mathbf{L}_a(s)$, $\mathbf{C}_a(s)$ are positive definite matrices [9], and $\mathbf{B}_\infty = \mathbf{B}|_{s=\infty}$.

2.6.3 Estimation of the Order of Approximation

As shown in (2.23), the error using the DEPACT MLF formulations decreases as the approximation order $m \rightarrow \infty$. Thus, is it of practical interest to arrive at a relationship between the error and the order m . In general, the maximum error occurs at the maximum value of frequency f_{max} . Hence, we can explicitly write an expression for the relative error as

$$\|\varepsilon_m\|_{rel} = \frac{\left\| e^{(A(s_{max})+s_{max}B(s_{max}))} - \prod_{k=1}^m Q_k \right\|}{\left\| e^{(A(s_{max})+s_{max}B(s_{max}))} \right\|} \quad (2.28)$$

For most practical cases, f_{max} can be obtained from the rise/fall time of the input signal as [3]

$$f_{max} \cong \frac{0.35}{t_r} \quad (2.29)$$

For a given value of $\|\varepsilon_m\|_{rel}$, the DEPACT order of approximation m can be determined using (2.28). Figure 2.8 shows the relationship between m and $\|\varepsilon_m\|_{rel}$ using MLF-II for a line with the following parameters: $R = 1\Omega/cm$, $L = 3.36nH/cm$, $C = 1.29pF/cm$, $G = 5 \times 10^{-9}$, and $d = 5cm$. As expected, for a given maximum allowable error, the order m increases with the increase in the bandwidth of approximation. For illustration purposes, consider the rise time of an input signal to be 0.2ns. Using (2.29), the corresponding f_{max} is 1.75GHz. From Figure 2.8, it can be seen that, for example, in order to achieve a maximum relative error of $\|\varepsilon_m\|_{rel} = 0.5 \times 10^{-3}$, an approximation order of $m = 8$ or $m = 9$ should be sufficient. This was tested by approximating the exponential stamp of the line with the MLF-II in (2.24) for $m = 8$.

The resulting relative error was 0.4808×10^{-3} , which falls below $\|\varepsilon_m\|_{rel}$.

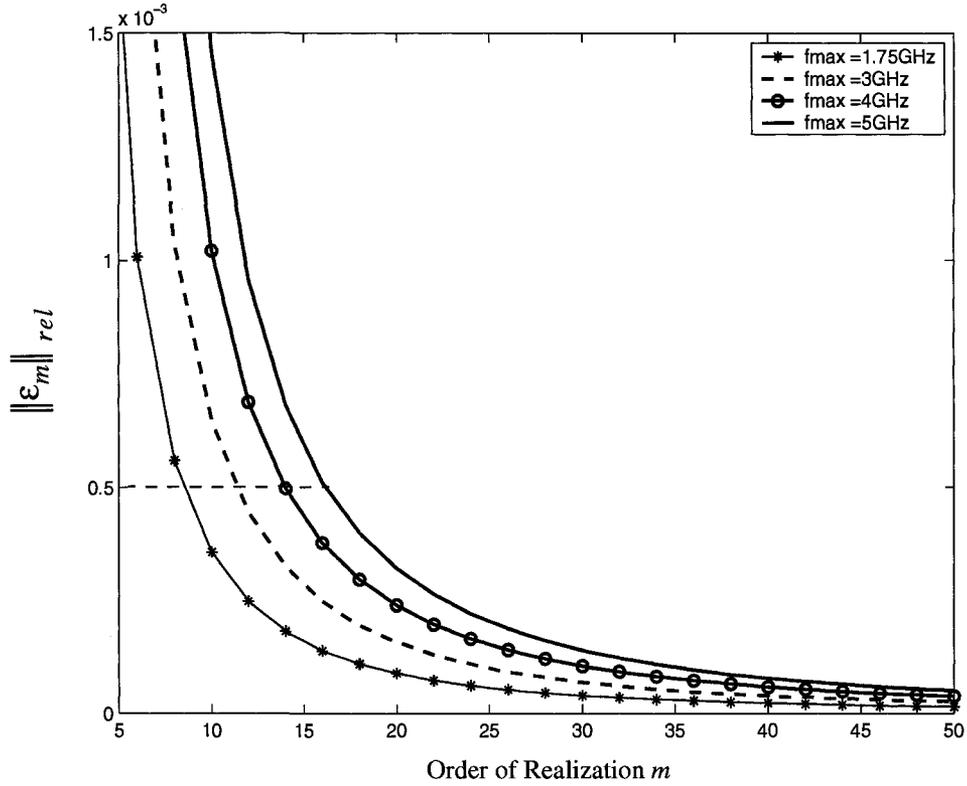


Figure 2.8: Relative approximation error versus DEPACT order m for varying bandwidths

2.6.4 DEPACT Macromodel Realization

The product terms of the MLF-II equations given in (2.26) can be viewed as a cascade of m transmission line subnetworks as shown in Figure 2.9(a). These subnetworks are referred to as DEPACT cells. The DEPACT cells, which realize each Q_k in (2.26) can be represented by a cascade of lossy and lossless transmission lines. Figure 2.9(b) depicts the configuration for the k th DEPACT cell. Here,

the lossy terms are macromodeled using the passive matrix rational approximation (MRA) [12–14] which was described in Section 2.4. In this case, the resulting macromodels are of significant lower orders, since a relevant delay portion is already extracted from these functions. The lossless sections can be macromodeled using the lossless MoC realization, along with a modal decoupling algorithm as described in [9]. The resulting macromodels for both the lossy and lossless sections are then combined to form the overall DEPACT cell [5, 15].

One of the main advantages of DEPACT is that the resulting macromodels are guaranteed to be passive. Passivity is an essential aspect since stable but non-passive macromodels, coupled with passive terminations, can lead to overall unstable systems. A proof of passivity for the DEPACT algorithm is provided in [15].

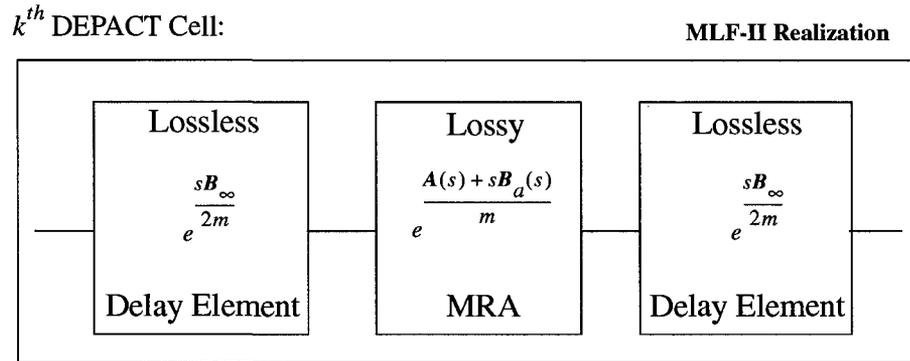
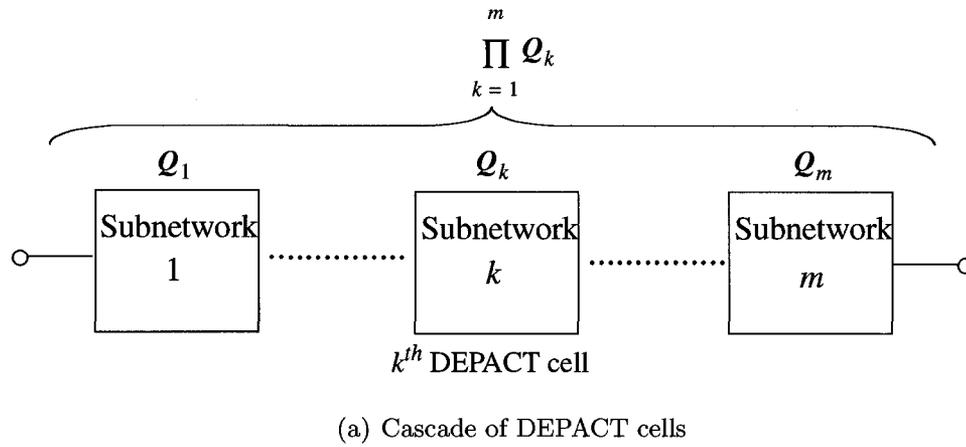


Figure 2.9: DEFACT macromodel realization

2.7 Summary

In summary, this chapter described the problem formulation for simulation of high-speed interconnects and provided an overview of some of the relevant macromodeling techniques. Conventional lumped segmentation, the method of characteristics (MoC), matrix rational approximation (MRA) and DEFACT were discussed. In addition, the significance of macromodeling using delay extraction was presented. In the next chapter, a new algorithm based on waveform relaxation and transverse partitioning

for simulation of massively coupled transmission lines is presented.

Chapter 3

Simulation of Coupled Interconnects Using Waveform Relaxation and Transverse Partitioning

In this chapter, a new waveform relaxation with transverse partitioning method for simulation of multiconductor transmission lines is described. Section 3.1 introduces the topic and provides motivation for the new method. Section 3.2 presents background information and describes previous partitioning and waveform relaxation (WR) techniques. Sections 3.3 and 3.4 provide the basis of the proposed algorithm and details regarding updating the WR sources, respectively. Implementation considerations and numerical examples are presented in Sections 3.5 and 3.6, respectively.

3.1 Introduction

One of the major difficulties in simulation of high-speed interconnect circuits is the problem of mixed frequency/time [3]. As mentioned in Chapter 2, this problem is due to that fact that distributed interconnect models do not have a direct representation in the time-domain and are best represented in the frequency-domain. However, nonlinear devices such as drivers and receivers are generally described in

the time-domain. These simultaneous formulations are difficult to handle by ordinary differential equation solvers which exist in traditional circuit simulators. Several methods have been proposed in the literature to address the mixed frequency/time representation [3, 5–8]. The common goal of these techniques is to transform the Telegrapher’s equations describing transmission lines, into a set of ordinary differential equations with appropriate time-delayed controlled sources that can be integrated with circuit simulators. Most of these techniques use some form of decoupling algorithm to convert the set of coupled partial differential equations describing the lines into a set of decoupled single modal equations. Subsequently, the line voltages and currents are obtained as a linear combination of modal variables. As a result, the interconnect stamp used by a time-domain circuit simulator is in the form of coupled ordinary differential equations. *This coupling is one of the major reasons for the excessive computational cost of simulating large multiconductor structures.* This increased computational cost occurs at two stages; the first is during the formulation of the interconnect stamp and the second, which is more costly, is during the transient simulation of the signal path including nonlinear terminations. For example, the y-parameters of a set of 64-coupled lines require approximation of 8320 different transfer functions. These transfer functions not only have to accurately match the frequency response of the line over the desired frequency bandwidth, but they also have to satisfy the challenging constraints of preserving the passivity and causality of the macromodel [13]. In the next stage, these transfer functions have to be converted to a set of ordinary differential equations or alternatively into an equivalent circuit to be interfaced with other circuit elements, and/or nonlinear driver/receiver circuits. The complexity of the equivalent circuit grows approximately at a rate of N^2 , where N is the number of coupled lines. Moreover, the average CPU expense of simulating a circuit of size M is proportional to M^α , where $1.5 < \alpha < 2$, (depending on the sparsity of the circuit). Consequently, the

average cost of simulating an N -coupled lines circuit is expected to be proportional to N^β , where $3 < \beta < 4$, which results in a prohibitively time-consuming simulation task compared to the simple case of simulating a single line. This was validated numerically by considering a simple interconnect circuit with varying number of lines. The p.u.l. parameters were generated statistically within a typical range of practical values. The generated circuits were simulated using uniform lumped segmentation in HSPICE [4]. The CPU time vs. the number of lines (N) is shown in Figure 3.1, which confirms the expected computational complexity. In addition, the method-of-characteristics [7, 10, 11], (which is also adopted as the W-element model in HSPICE) was not able, in most cases, to handle circuits with more than 12 coupled lines.

In this chapter, a new method is presented to address the computational complexity of the time-domain simulation of large coupled interconnect circuits. The proposed algorithm is based on the concept of waveform relaxation using transverse partitioning [17, 18, 31]. The new technique reduces the coupled simulation problem into a series of simulation steps. Each step is of complexity equivalent to that of simulating a single line. In addition, these steps can be performed in parallel and thus opening the door for multiprocessor implementation that has the potential of providing further significant reduction in the computational time.

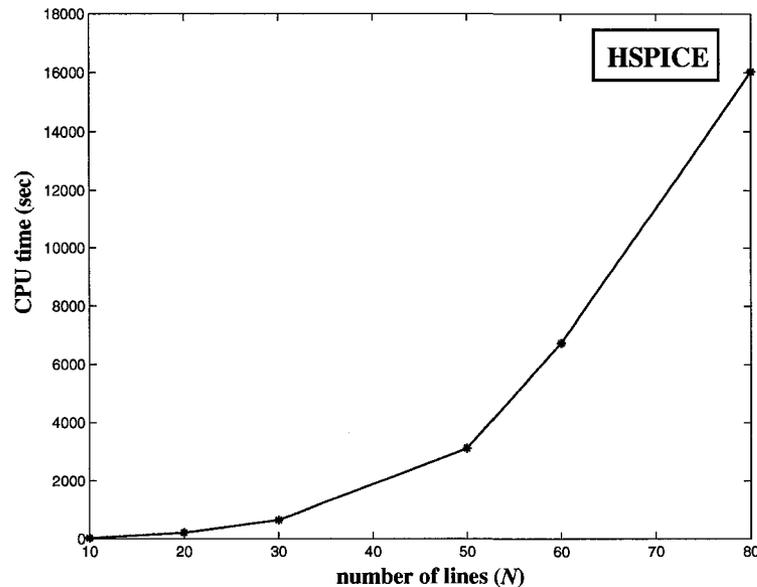


Figure 3.1: Average CPU cost for varying number of lines using conventional circuit simulators

3.2 Background and Previous Techniques

Waveform relaxation (WR) techniques [32–36], were introduced in the early 1980’s as an alternative to the conventional time-stepping algorithms used by circuit simulators for solving ordinary differential equations. The concept of WR is based on partitioning the original circuit into subcircuits that are solved independently for the entire time-interval of interest. Coupling effects among individual subcircuits are represented by time-domain sources that are initially assumed to be known. Next, the currents and voltages of the subcircuits are computed and used to update these sources. This iterative process is continued until convergence is obtained. Several techniques were proposed in the literature to improve the efficiency of WR analysis including dynamic partitioning, scheduling and time windowing. In addition, WR

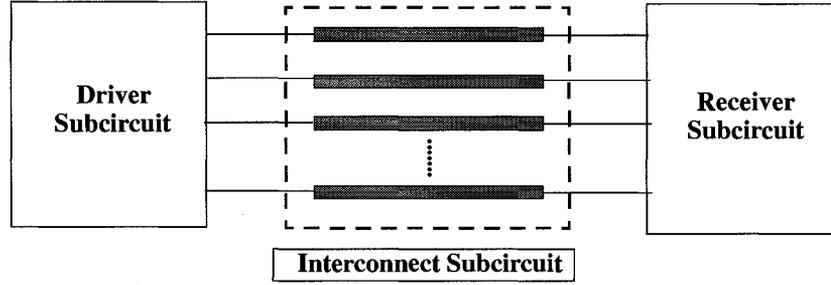
techniques are well suited for parallel computations. However, the rate of convergence remains a major challenge facing WR techniques and limiting their applicability to interconnect circuits. To obtain fast convergence, it is necessary that partitioning the original circuit is done in such a way that the coupling among the individual subcircuits is weak. The reported algorithms [10, 37, 38] that attempted application of WR to interconnect circuits used the physically natural *longitudinal* partitioning between the input circuit and the output circuit to the multiconductor transmission line system interface. A brief overview of longitudinal partitioning is given in the following section.

3.2.1 Longitudinal Partitioning

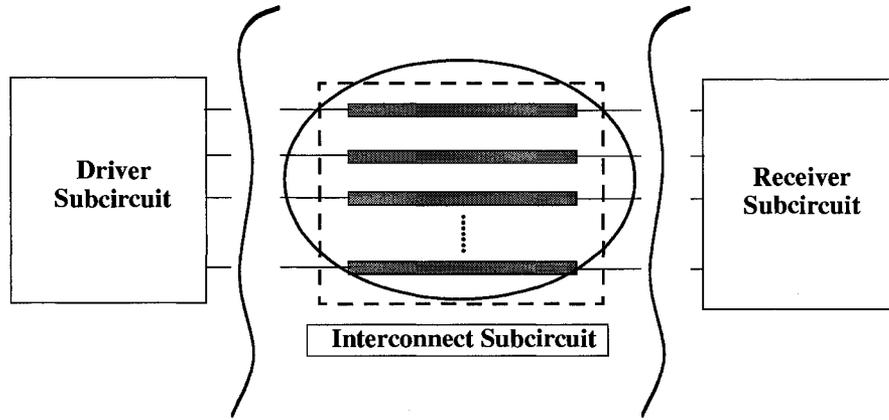
The concept of longitudinal partitioning is demonstrated in Figure 3.2. The original circuit is shown in Figure 3.2(a), and the partitioned circuit is shown in Figure 3.2(b). As seen from Figure 3.2(b), using longitudinal partitioning, the input and output circuits are partitioned from the transmission line system interface, into separate subcircuits. In order for WR to achieve convergence, it is vital that the coupling between these individual subcircuits is relatively weak. However, there is no guarantee on the strength of this coupling, since it varies from one circuit to another (depending on the terminations).

For example, consider the two-coupled line circuit shown in Figure 3.3. To apply WR with longitudinal partitioning, the terminations are separated from the transmission lines and the coupling effects are represented with current sources (as shown in Figure 3.4), where r represents the r th WR iteration.

The coupling current sources $J_1 - J_4$ can be computed at a given iteration as



(a) Original interconnect circuit



(b) Longitudinal partitioning of the original interconnect circuit in Figure 3.2(a)

Figure 3.2: Longitudinal partitioning of an interconnect circuit

$$\begin{aligned}
 J_1^{(r)} &= \frac{V_{src} - V_1^{(r)}}{R_1} \\
 J_2^{(r)} &= -V_2^{(r)} sC_1 \\
 J_3^{(r)} &= -V_3^{(r)} g_2 \\
 J_4^{(r)} &= -V_4^{(r)} sC_2
 \end{aligned} \tag{3.1}$$

and the voltages for the next iteration can be computed as

$$\mathbf{V}^{(r+1)} = \bar{\mathbf{Y}}^{-1} \mathbf{I}^{(r)} \quad (3.2)$$

where $\bar{\mathbf{Y}}$ is the y -parameter matrix of the transmission lines [3], and $\mathbf{I}^{(r)} = [J_1^{(r)} J_2^{(r)} J_3^{(r)} J_4^{(r)}]^T$.

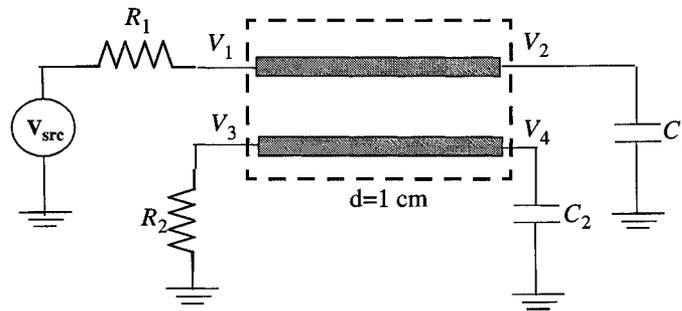


Figure 3.3: Two-coupled line circuit

Using (3.1), (3.2) and $V_{src} = 50V$, waveform relaxation was applied to the circuit in Figure 3.4. Starting with an initial guess of zero for all sources, the WR results were obtained and compared to the exact solution (using the exponential stamp of the transmission line). Table 3.1 summarizes the results for varying terminations and number of iterations. For example, for Cases 1-3, the resistor values R_1 and R_2 are relatively large (500Ω), and the capacitor values C_1 and C_2 are relatively small (1.5 pF). In these cases, the method converges, and the relative error decreases as we increase the number of iterations. This is expected behaviour for a typical WR approach. On the other hand, for Cases 4-7, the coupling between the subcircuits was increased by either decreasing the resistor values or increasing the capacitor values. As seen, in these cases the longitudinal partitioning method diverges. In

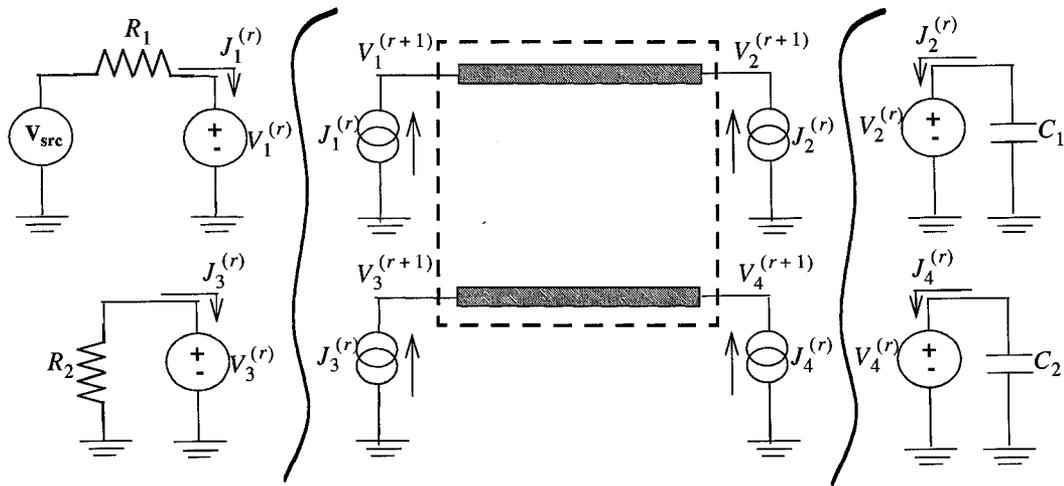


Figure 3.4: Two-coupled line circuit partitioned in the longitudinal direction

addition, it is important to note that in many practical cases, the driver and receiver circuits are much more complex, making it difficult to ensure whether the method will achieve convergence or not. Consequently, these factors limited the success of the reported algorithms in [10, 37, 38].

There is one more problem which longitudinal partitioning does not address. This is the coupling between individual transmission lines within the interconnect subcircuit. As discussed in Section 3.1, this coupling is one of the major bottlenecks when simulating a large number of coupled lines. Using longitudinal partitioning, the transmission lines in the interconnect subcircuit are still coupled, and must be simulated together, resulting in a prohibitively time-consuming simulation task as the number of lines is increased.

To address these issues, the proposed method is based on the concept of *transverse*

Table 3.1: Results for Longitudinal Partitioning Example

	R_1, R_2 (Ω)	C_1, C_2 (pF)	No. of Iter	Relative Error
Case 1	500	1.5	5	0.359
Case 2	500	1.5	20	0.002
Case 3	500	1.5	30	7.458e-5
Case 4	50	1.5	—	Diverges
Case 5	25	1.5	—	Diverges
Case 6	500	3	—	Diverges
Case 7	500	4	—	Diverges

partitioning where the multiconductor transmission line system is partitioned, exploiting the weak coupling between individual traces [17]. The number of lines in each subcircuit could be as low as one and the coupling effects due to neighboring traces are represented by voltage/current sources. A relaxation-based algorithm is used to iterate between the subcircuits until convergence is achieved. In general, very few iterations are required due to the relatively weak coupling between individual subcircuits. The computational complexity of each iteration is essentially equivalent to simulating N single lines, where N is the number of coupled lines. The remaining of this chapter describes the details of the new method.

3.3 Basis of the Proposed Algorithm

Recall from Chapter 2, the Telegrapher's equations in the following form [9]

$$\begin{aligned}
\frac{\partial}{\partial x} \mathbf{v}(x, t) &= -\mathbf{R}\mathbf{i}(x, t) - \mathbf{L} \frac{\partial}{\partial t} \mathbf{i}(x, t) \\
\frac{\partial}{\partial x} \mathbf{i}(x, t) &= -\mathbf{G}\mathbf{v}(x, t) - \mathbf{C} \frac{\partial}{\partial t} \mathbf{v}(x, t)
\end{aligned} \tag{3.3}$$

where $\mathbf{R} \in \mathfrak{R}^{N \times N}$, $\mathbf{L} \in \mathfrak{R}^{N \times N}$, $\mathbf{C} \in \mathfrak{R}^{N \times N}$, and $\mathbf{G} \in \mathfrak{R}^{N \times N}$ are the per-unit-length (p.u.l.) resistance, inductance, capacitance and conductance parameter matrices of the transmission lines, $\mathbf{v}(x, t) \in \mathfrak{R}^N$ and $\mathbf{i}(x, t) \in \mathfrak{R}^N$ represent the voltage and current vectors as a function of position x and time t , and N is the number of coupled lines.

The equations for the j th line can be written as

$$\begin{aligned}
\frac{\partial v_j}{\partial x} &= -R_{jj}i_j - L_{jj} \frac{\partial i_j}{\partial t} - \sum_{\substack{k=1 \\ k \neq j}}^N (R_{jk}i_k + L_{jk} \frac{\partial i_k}{\partial t}) \\
\frac{\partial i_j}{\partial x} &= -\hat{G}_{jj}i_j - \hat{C}_{jj} \frac{\partial v_j}{\partial t} - \sum_{\substack{k=1 \\ k \neq j}}^N (G_{jk}(v_j - v_k) + C_{jk}(\frac{\partial v_j}{\partial t} - \frac{\partial v_k}{\partial t}))
\end{aligned} \tag{3.4}$$

where $\hat{C}_{jj} = \sum_{k=1}^N C_{jk}$ and $\hat{G}_{jj} = \sum_{k=1}^N G_{jk}$ represent the *self* capacitance and conductance of the j th line, respectively.

The summation terms in (3.4) represent the coupling effects on line j due to the neighboring lines. Equation (3.4) can be rewritten as

$$\begin{aligned}
\frac{\partial v_j}{\partial x} &= -R_{jj}i_j - L_{jj}\frac{\partial i_j}{\partial t} + e_j(x, t) \\
\frac{\partial i_j}{\partial x} &= -\hat{G}_{jj}i_j - \hat{C}_{jj}\frac{\partial v_j}{\partial t} + q_j(x, t)
\end{aligned} \tag{3.5}$$

where the coupling terms $e_j(x, t)$ and $q_j(x, t)$ are defined as follows

$$\begin{aligned}
e_j(x, t) &= -\sum_{\substack{k=1 \\ k \neq j}}^N (R_{jk}i_k + L_{jk}\frac{\partial i_k}{\partial t}) \\
q_j(x, t) &= -\sum_{\substack{k=1 \\ k \neq j}}^N (G_{jk}(v_j - v_k) + C_{jk}(\frac{\partial v_j}{\partial t} - \frac{\partial v_k}{\partial t}))
\end{aligned} \tag{3.6}$$

Equation (3.5) represents a set of *coupled* differential equations. In order to *decouple* the equations, an initial guess $\mathbf{v}^{(0)}(x, t)$ and $\mathbf{i}^{(0)}(x, t)$ is assumed for the voltages and currents, leading to initial waveforms $e_j^{(0)}(x, t)$ and $q_j^{(0)}(x, t)$ for the coupling terms. An iterative scheme is then used to obtain the solution of (3.5). Applying waveform relaxation techniques [32] to (3.5), we obtain a recursive set of decoupled differential equations given by

$$\begin{aligned}
\frac{\partial v_j^{(r+1)}}{\partial x} &= -R_{jj}i_j^{(r+1)} - L_{jj}\frac{\partial i_j^{(r+1)}}{\partial t} + e_j^{(r)}(x, t) \\
\frac{\partial i_j^{(r+1)}}{\partial x} &= -\hat{G}_{jj}i_j^{(r+1)} - \hat{C}_{jj}\frac{\partial v_j^{(r+1)}}{\partial t} + q_j^{(r)}(x, t)
\end{aligned} \tag{3.7}$$

where r represents the r th iteration.

Expressing (3.7) in the frequency-domain we get

$$\frac{\partial \Phi_j^{(r+1)}}{\partial x} = \mathbf{F}_j(s) \Phi_j^{(r+1)}(x, s) + \Psi_j^{(r)}(x, s) \quad (3.8)$$

where

$$\Phi_j^{(r+1)}(x, s) = \begin{bmatrix} V_j^{(r+1)}(x, s) \\ I_j^{(r+1)}(x, s) \end{bmatrix} \quad (3.9)$$

$$\mathbf{F}_j(s) = \begin{bmatrix} 0 & -R_{jj} - sL_{jj} \\ -\hat{G}_{jj} - s\hat{C}_{jj} & 0 \end{bmatrix} \quad (3.10)$$

$$\begin{aligned} \Psi_j^{(r)}(x, s) &= \begin{bmatrix} E_j^{(r)}(x, s) \\ Q_j^{(r)}(x, s) \end{bmatrix} \\ &= \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & -R_{jk} - sL_{jk} \\ -G_{jk} - sC_{jk} & 0 \end{bmatrix} \begin{bmatrix} V_j^{(r)}(x, s) - V_k^{(r)}(x, s) \\ I_k^{(r)}(x, s) \end{bmatrix} \end{aligned} \quad (3.11)$$

and where $E_j^{(r)}(x, s) = \mathcal{F}\{e_j^{(r)}(x, t)\}$, $Q_j^{(r)}(x, s) = \mathcal{F}\{q_j^{(r)}(x, t)\}$ and $\mathcal{F}\{\}$ denotes the Fourier transform operator. The vectors $V_j^{(r+1)}(x, s)$ and $I_j^{(r+1)}(x, s)$ in (3.9) represent the voltages and currents along the j th line at iteration $r + 1$.

Equation (3.8) represents a set of inhomogeneous ordinary differential equations corresponding to the j th line, which can be solved as

$$\Phi_j^{(r+1)}(x, s) = e^{\mathbf{F}_j x} \Phi_j^{(r+1)}(0, s) + \delta_j^{(r)}(x, s) \quad (3.12)$$

The relaxation sources which represent the coupling between individual lines are given by

$$\delta_j^{(r)}(x, s) = \begin{bmatrix} V_{src_j}(x, s) \\ I_{src_j}(x, s) \end{bmatrix} = \int_0^x e^{\mathbf{F}_j(x-\eta)} \Psi_j^{(r)}(\eta, s) d\eta \quad (3.13)$$

Using the relations in (3.7)-(3.13), the N -coupled interconnect circuit is partitioned in the *transverse* direction, and can be modeled as N decoupled individual subcircuits, with independent WR sources. Equation (3.13) forms the basis for updating these sources, which is one of the main aspects of the proposed method. In the next section, the computational details of updating the WR sources will be presented.

3.4 Updating the Waveform Relaxation Sources

The coupling sources in (3.13) can be computed using two approaches. The first approach is a single-ended representation, where the sources are located at the end of the lines. The second approach is a distributed representation, where the sources are distributed throughout the length of the lines.

3.4.1 Single-Ended Representation

In this approach, (3.12) is modeled as a set of single transmission lines with WR sources located at the end of each line as shown in Figure 3.5. The relaxation sources are given by

$$\begin{bmatrix} v_{src_j}(d, t) \\ i_{src_j}(d, t) \end{bmatrix} = \mathcal{F}^{-1}\{\boldsymbol{\delta}_j^{(r)}(d, s)\} \quad (3.14)$$

where

$$\boldsymbol{\delta}_j^{(r)}(x, s) = \begin{bmatrix} V_{src_j}(d, s) \\ I_{src_j}(d, s) \end{bmatrix} = \int_0^d e^{\mathbf{F}_j(d-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta \quad (3.15)$$

and $\mathcal{F}^{-1}\{\}$ denotes the inverse Fourier transform operator, and d is the length of the line.

The integral in (3.15) can be computed recursively as follows [17]

$$\begin{aligned} \boldsymbol{\delta}_j^{(r)}(x + \Delta x, s) &= \int_0^x e^{\mathbf{F}_j(x+\Delta x-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta + \int_x^{(x+\Delta x)} e^{\mathbf{F}_j(x+\Delta x-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta \\ &= e^{\mathbf{F}_j \Delta x} \int_0^x e^{\mathbf{F}_j(x-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta + \int_x^{(x+\Delta x)} e^{\mathbf{F}_j(x+\Delta x-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta \\ &= e^{\mathbf{F}_j \Delta x} \boldsymbol{\delta}_j^{(r)}(x, s) + \int_x^{(x+\Delta x)} e^{\mathbf{F}_j(x+\Delta x-\eta)} \boldsymbol{\Psi}_j^{(r)}(\eta, s) d\eta \end{aligned} \quad (3.16)$$

Next, approximating the integral in (3.16) with a suitable integration formula (e.g. Trapezoidal rule) we get

$$\boldsymbol{\delta}_j^{(r)}(x + \Delta x, s) \cong e^{\mathbf{F}_j \Delta x} \boldsymbol{\delta}_j^{(r)}(x, s) + \frac{\Delta x}{2} \left[e^{\Delta x} \boldsymbol{\Psi}_j^{(r)}(x, s) + \boldsymbol{\Psi}_j^{(r)}(x + \Delta x, s) \right] \quad (3.17)$$

The computational steps for updating the relaxation sources can be summarized as follows:

Let $r=0$:

- *Step 1: Start with an initial guess for the relaxation sources $v_{src_j}(d, t)$, $i_{src_j}(d, t)$ for $1 \leq j \leq N$ and $0 \leq t \leq T$, where T is the time interval of interest.*
- *Step 2: Simulate each individual single line subcircuit to obtain $v_j^{(r+1)}(0, t)$ and $i_j^{(r+1)}(0, t)$.*
- *Step 3: Using the Fast Fourier Transform (FFT), evaluate $\Phi_j^{(r+1)}(0, s)$.*
- *Step 4: Calculate $\Phi_j^{(r+1)}(x, s)$ using (3.12) and (3.13).*
- *Step 5: Using $\Phi_j^{(r+1)}(x, s)$, calculate $\Psi_j^{(r+1)}(x, s)$.*
- *Step 6: Use (3.14)-(3.17) to update the relaxation sources and compute $\delta_j^{(r+1)}(x, s)$.*
- *Step 7: Calculate $v_{src_j}^{(r+1)}(d, t)$ and $i_{src_j}^{(r+1)}(d, t)$ using the Inverse Fast Fourier Transform (IFFT) as in (3.14).*
- *Step 8: Let $r = r + 1$; go to step 2 and repeat until convergence is achieved within an acceptable error tolerance.*

It is worth noting that in this approach, there are several transformations between the time-domain and the frequency-domain (using FFT and IFFT) in order to update the relaxation sources. In the next section, we present an alternative direct approach in the time-domain which avoids these transformations.

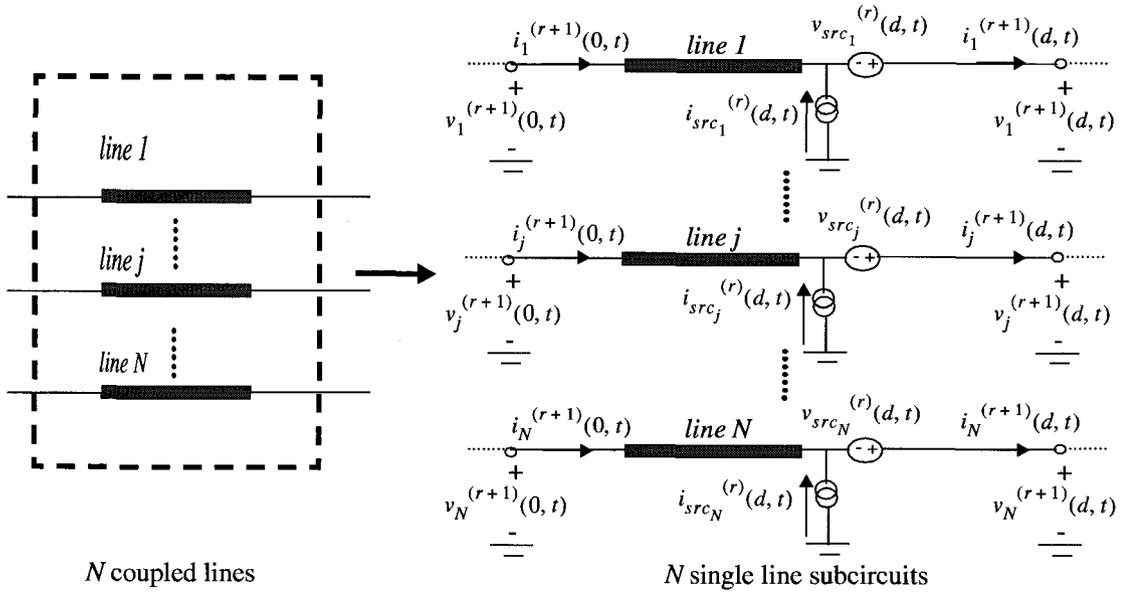


Figure 3.5: Single-ended representation of decoupled network using transverse partitioning

3.4.2 Distributed Representation

In this approach, we discretize the integral in (3.13) using a suitable integration formula [39] as follows

$$\begin{aligned}
 \delta_j^{(r)}(x, s) &= \begin{bmatrix} V_{src_j}(d, s) \\ I_{src_j}(d, s) \end{bmatrix} = \int_0^d e^{\mathbf{F}_j(d-\eta)} \Psi_j^{(r)}(\eta, s) d\eta \\
 &\cong \sum_{p=1}^n w_p e^{\mathbf{F}_j(d-x_p)} \Psi_j^{(r)}(x_p, s) \Delta x_p \quad (3.18)
 \end{aligned}$$

where w_p is a weighting factor, $\Delta x_p = x_p - x_{p-1}$, and n is the number of discretization points.

In this case, the solution of the decoupled differential equations at the $(r + 1)^{th}$ iteration can be expressed as

$$\Phi_j^{(r+1)}(d, s) = e^{\mathbf{F}_j d} \Phi_j^{(r+1)}(0, s) + \sum_{p=1}^n w_p e^{\mathbf{F}_j(d-x_p)} \Psi_j^{(r)}(x_p, s) \Delta x_p \quad (3.19)$$

PROPOSITION:

Equation (3.19) can be realized in the time-domain using the equivalent circuit shown in Figure 3.6, which consists of a cascade of n single lines with lengths $\Delta x_1, \Delta x_2, \dots, \Delta x_n$, respectively. In this case, the relaxation coupling sources $\tilde{e}_j^{(r)}(x_p, t)$ and $\tilde{q}_j^{(r)}(x_p, t)$ are distributed throughout the length of the lines and are located at the points x_1, x_2, \dots, x_n , and are defined as

$$\begin{aligned} \tilde{e}_j^{(r)}(x_p, t) &= -w_p \sum_{\substack{k=1 \\ k \neq j}}^N \left(R_{jk} \Delta x_p i_k^{(r)}(x_p, t) + L_{jk} \Delta x_p \frac{\partial i_k^{(r)}(x_p, t)}{\partial t} \right) \\ \tilde{q}_j^{(r)}(x_p, t) &= -w_p \sum_{\substack{k=1 \\ k \neq j}}^N \left(G_{jk} \Delta x_p (v_j^{(r)}(x_p, t) - v_k^{(r)}(x_p, t)) \right. \\ &\quad \left. + C_{jk} \Delta x_p \left(\frac{\partial v_j^{(r)}(x_p, t)}{\partial t} - \frac{\partial v_k^{(r)}(x_p, t)}{\partial t} \right) \right) \end{aligned} \quad (3.20)$$

where $1 \leq p \leq n$.

PROOF:

The relationship between the terminal voltages and currents can be written as

$$\begin{aligned} \begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &= \begin{bmatrix} V_j^{(r+1)}(x_n, s) \\ I_j^{(r+1)}(x_n, s) \end{bmatrix} \\ &= e^{\mathbf{F}_j \Delta x_n} \begin{bmatrix} V_j^{(r+1)}(x_{n-1}, s) \\ I_j^{(r+1)}(x_{n-1}, s) \end{bmatrix} + w_n \Delta x_n \Psi_j^{(r)}(x_n, s) \end{aligned} \quad (3.21)$$

Writing $\begin{bmatrix} V_j^{(r+1)}(x_{n-1}, s) \\ I_j^{(r+1)}(x_{n-1}, s) \end{bmatrix}$ in terms of $\Psi_j^{(r)}(x_{n-1}, s)$ and $\begin{bmatrix} V_j^{(r+1)}(x_{n-2}, s) \\ I_j^{(r+1)}(x_{n-2}, s) \end{bmatrix}$, we get

$$\begin{aligned} \begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &= e^{\mathbf{F}_j \Delta x_n} \left[e^{\mathbf{F}_j \Delta x_{n-1}} \begin{bmatrix} V_j^{(r+1)}(x_{n-2}, s) \\ I_j^{(r+1)}(x_{n-2}, s) \end{bmatrix} + w_{n-1} \Delta x_{n-1} \Psi_j^{(r)}(x_{n-1}, s) \right] \\ &\quad + w_n \Delta x_n \Psi_j^{(r)}(x_n, s) \end{aligned} \quad (3.22)$$

$$\begin{aligned}
\begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &= e^{\mathbf{F}_j(\Delta x_n + \Delta x_{n-1})} \begin{bmatrix} V_j^{(r+1)}(x_{n-2}, s) \\ I_j^{(r+1)}(x_{n-2}, s) \end{bmatrix} + e^{\mathbf{F}_j \Delta x_n} w_{n-1} \Delta x_{n-1} \Psi_j^{(r)}(x_{n-1}, s) \\
&\quad + w_n \Delta x_n \Psi_j^{(r)}(x_n, s)
\end{aligned} \tag{3.23}$$

and

$$\begin{aligned}
\begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &= e^{\mathbf{F}_j(\Delta x_n + \Delta x_{n-1})} \left[e^{\mathbf{F}_j \Delta x_{n-2}} \begin{bmatrix} V_j^{(r+1)}(x_{n-3}, s) \\ I_j^{(r+1)}(x_{n-3}, s) \end{bmatrix} + w_{n-2} \Delta x_{n-2} \Psi_j^{(r)}(x_{n-2}, s) \right] \\
&\quad + e^{\mathbf{F}_j \Delta x_n} w_{n-1} \Delta x_{n-1} \Psi_j^{(r)}(x_{n-1}, s) + w_n \Delta x_n \Psi_j^{(r)}(x_n, s)
\end{aligned} \tag{3.24}$$

In general, we can write

$$\begin{aligned}
\begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &\cong e^{\mathbf{F}_j \left(\sum_{k=0}^p \Delta x_{n-k} \right)} \begin{bmatrix} V_j^{(r+1)}(x_{n-p-1}, s) \\ I_j^{(r+1)}(x_{n-p-1}, s) \end{bmatrix} \\
&\quad + \sum_{k=0}^p e^{\mathbf{F}_j \left(d - \sum_{i=0}^{p-k+1} \Delta x_i \right)} w_{n-k} \Delta x_{n-k} \Psi_j^{(r)}(x_{n-k}, s)
\end{aligned} \tag{3.25}$$

where $1 \leq p \leq n - 1$.

Next, letting $p = n - 1$, we obtain

$$\begin{aligned}
\begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} &\cong e^{\mathbf{F}_j \left(\sum_{k=0}^{n-1} \Delta x_{n-k} \right)} \begin{bmatrix} V_j^{(r+1)}(0, s) \\ I_j^{(r+1)}(0, s) \end{bmatrix} \\
&+ \sum_{k=0}^{n-1} e^{\mathbf{F}_j \left(d - \sum_{i=0}^{n-k} \Delta x_i \right)} w_{n-k} \Delta x_{n-k} \Psi_j^{(r)}(x_{n-k}, s)
\end{aligned} \tag{3.26}$$

Noting that $d = \Delta x_1 + \Delta x_2 + \dots + \Delta x_n$ and after some mathematical manipulation, (3.26) can be re-written as

$$\begin{bmatrix} V_j^{(r+1)}(d, s) \\ I_j^{(r+1)}(d, s) \end{bmatrix} \cong e^{\mathbf{F}_j d} \begin{bmatrix} V_j^{(r+1)}(0, s) \\ I_j^{(r+1)}(0, s) \end{bmatrix} + \sum_{p=1}^n w_p e^{\mathbf{F}_j(d-x_p)} \Psi_j^{(r)}(x_p, s) \Delta x_p \tag{3.27}$$

Finally, replacing $\begin{bmatrix} V_j^{(r+1)}(x, s) \\ I_j^{(r+1)}(x, s) \end{bmatrix}$ with $\Phi_j^{(r+1)}(x, s)$, we obtain the following relationship

$$\Phi_j^{(r+1)}(d, s) = e^{\mathbf{F}_j d} \Phi_j^{(r+1)}(0, s) + \sum_{p=1}^n w_p e^{\mathbf{F}_j(d-x_p)} \Psi_j^{(r)}(x_p, s) \Delta x_p \tag{3.28}$$

which is equivalent to the relationship in (3.19). \square

Based on the above proposition, the computational steps for updating the relaxation sources can be summarized as follows

Let $r=0$:

- *Step 1: Start with an initial waveform for $\tilde{e}_j^{(r)}(x_p, t)$ and $\tilde{q}_j^{(r)}(x_p, t)$ for $1 \leq j \leq N$ and $1 \leq p \leq n$.*
- *Step 2: Simulate each individual single line subcircuit to obtain $v_j^{(r+1)}(x, t)$ and $i_j^{(r+1)}(x, t)$.*
- *Step 3: Based on the solution obtained from step 2, update the relaxation sources $\tilde{e}_j^{(r+1)}(x_p, t)$ and $\tilde{q}_j^{(r+1)}(x_p, t)$ using (3.20).*
- *Step 4: Let $r=r+1$; go to step 2 and repeat until convergence is achieved within an acceptable error tolerance.*

Obviously, the numerical accuracy in approximating the integral in (3.18) depends on the number of discretization points n and the specific numerical integration technique. Most of the widely-used numerical integration algorithms have an error estimate associated with them [39]. In our tested cases, n varied between 20 and 50, depending on the specific example. However, underestimating the value of n would only lead to a slight increase in the number of iterations required, but it would not affect the accuracy of the final solution.

It is to be noted that in this approach, the sources are calculated directly in the time-domain. Conceptually, this is equivalent to the single-ended representation. However, in the distributed approach, the circuit simulator is used to implicitly compute the integral in (3.18). This avoids the explicit time/frequency transformations encountered in the single-ended approach.

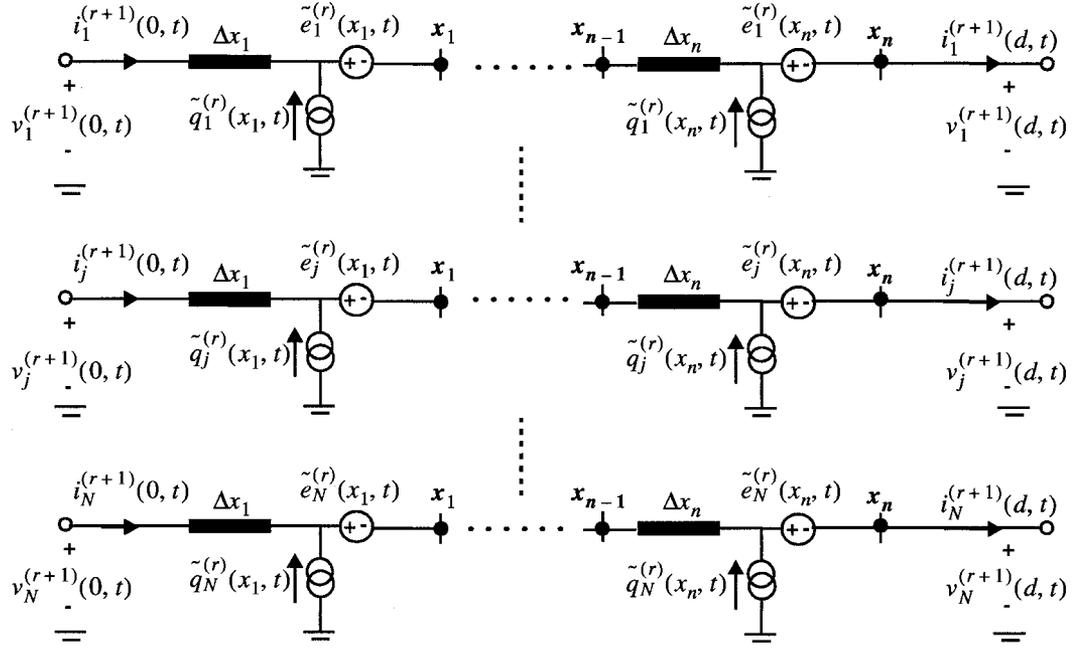


Figure 3.6: Distributed representation of decoupled network using transverse partitioning

3.5 Implementation Considerations

Several relevant points regarding the implementation of the proposed waveform relaxation scheme are worth noting.

1. There are many possibilities for the discretization of the integral in (3.18), such as Trapezoidal rule, quadrature formulas [39], etc.. Based on the specific technique, the weights w_p and the abscissas x_p can be determined.
2. One of the key features of the proposed algorithm is its generality with respect to the specific macromodel used for the individual transmission line subcircuits. For example, in our implementation in this chapter, we used macromodels based

on uniform lumped segmentation [9]. However, a variety of other macromodeling algorithms could be used, such as MRA [13] and DEPACT [5]. Specifically, a new WR with transverse partitioning algorithm applied to DEPACT macromodels will be described in Chapter 6 of this thesis.

3. *Convergence of the WR method:* The proposed WR method is based on the conventional Gauss-Seidel (GS) and Gauss-Jacobi (GJ) relaxation techniques. The convergence of the GS/GJ has been previously investigated in the literature [40]. Convergence properties of WR as applied to general circuit simulation can be found in [41]. The specific application of WR to PEEC models and convergence properties has been recently presented in [42]. The rate of convergence is dependent on a contraction factor γ , which given by

$$\gamma = K^2 = \max_{\substack{i,j \\ i \neq j}} \frac{l_{i,j}^2}{l_{ii}l_{jj}} \quad (3.29)$$

where l_{ij} is the $(i, j)^{th}$ component of the \mathbf{L} matrix. A similar relationship can be derived in terms of the self and mutual capacitances. Although there is no theoretical proof of guaranteeing the convergence of the WR method, in our numerical experiments, convergence was obtained in 2-4 iterations even for coupling coefficients K as high as 0.75. This fast convergence can be easily explained from the fact that the capacitance matrix \mathbf{C} is diagonally dominant and the largest elements of the inductance matrix \mathbf{L} are located on the diagonal, with the magnitude of the off-diagonal elements rapidly decreasing as the distance between the lines increases [9]. For stronger coupling between two adjacent lines, it is recommended to group these two lines within the same subcircuit.

4. An important advantage of the proposed algorithm is that it is highly parallelizable. This is because each subcircuit is solved independently, and array processors can be used to effectively simulate each subcircuit in parallel. In order to update the relaxation sources, the processors only have to communicate after the simulation of the individual subcircuits is complete. This reduces the CPU overhead associated with interprocessor communication. This work has been collaboratively developed on a multiprocessor platform and submitted for publication [25].
5. *Integration with circuit simulators:* It is relatively easy to integrate the proposed algorithm with SPICE-like circuit simulators. The individual subcircuits can be simulated using suitable macromodeling techniques. The relaxation sources can be implemented as embedded data-driven sources in the circuit simulator netlist. After each iteration, the data driving the relaxation sources is updated based on the voltages and currents of the individual subcircuits. The updated values for the relaxation sources are then fed back to the circuit simulator for the next iteration.

3.6 Numerical Results

In this section, we consider four case studies which validate the accuracy and efficiency of the proposed WR with transverse partitioning (WR-TP) algorithm. The first two examples demonstrate the accuracy and convergence properties of the WR-TP algorithm as applied to various types of circuits. In the third example, the robustness of the proposed algorithm, as well as the speed of convergence are demonstrated for specific cases which conventional methods cannot handle. Lastly, the fourth example considers several cases of a large number of coupled lines and illustrates numerically the linear relationship between the CPU cost and the number of lines. Comparison

of the CPU time for all examples was performed using an INTEL P4 2GHz CPU.

3.6.1 Example 1

A distributed interconnect circuit consisting of nine lossless coupled lines with a length of 1 cm is considered. The parameters of the line are given in [13] with 50Ω resistor and $1pF$ capacitor terminations at the near and far-ends of each line, respectively. The circuit was partitioned into nine subcircuits, where each subcircuit consists of a single line with its corresponding terminations and equivalent relaxation sources. The individual single lines were macromodeled using the uniform lumped segmentation [9]. To start the waveform relaxation iterations, the voltage and current waveforms across all lines were initialized to zero values. Figures 3.7 and 3.8 show a sample of the output waveforms (after 3 iterations) compared to the W-element in HSPICE (based on the method of characteristics), corresponding to an input pulse on line #1 with rise/fall times of 0.5ns and a pulse width of 5ns. As seen, the results are in very good agreement. It is to be noted that, in contrast to the proposed transverse partitioning scheme, the classical longitudinal partitioning waveform relaxation technique [10, 37, 38] failed to converge even when using a relatively accurate initial guess for the output waveforms.

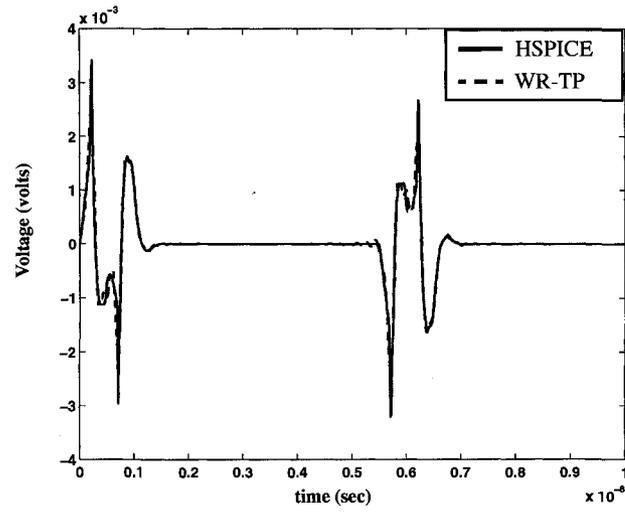


Figure 3.7: Transient response at near-end of victim line # 2 (Example 1)

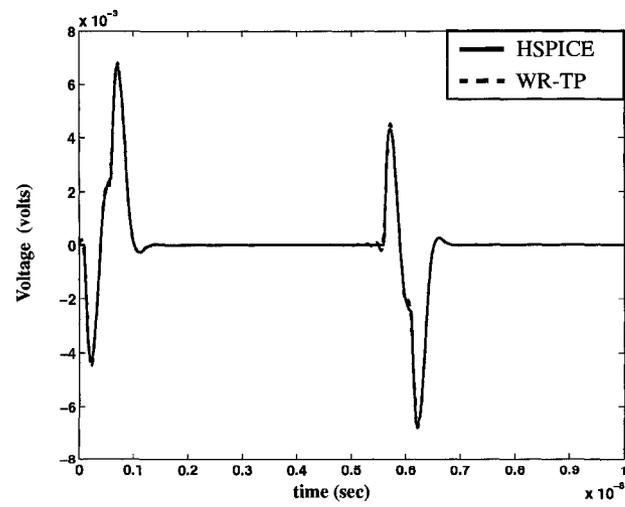


Figure 3.8: Transient response at far-end of victim line # 3 (Example 1)

3.6.2 Example 2

The purpose of this example is to illustrate the applicability of the WR-TP technique to networks consisting of multiple sets of coupled interconnects. We consider here three coupled interconnect circuits [5] with nonlinear terminations as shown in Figure 3.9. The input voltage is a unit step with a rise time of 0.1ns. Using the WR-TP algorithm, the circuit is partitioned in the transverse direction along the dotted line shown in Figure 3.9. The coupling effects between the decoupled subcircuits are represented by relaxation sources as shown in Figure 3.10. To compare the accuracy of the computed results, the circuit in Figure 3.9 was simulated using its *coupled* lumped equivalent circuit in HSPICE. Figures 3.11 - 3.14 show a sample of the transient response of the far-end voltages of subnetwork #2 (labeled active and victim line in Figure 3.9), respectively, after 2 and 3 WR iterations. As seen from the plots, the results are in very good agreement. In addition, the graphs demonstrate the fast convergence properties of the proposed WR-TP algorithm.

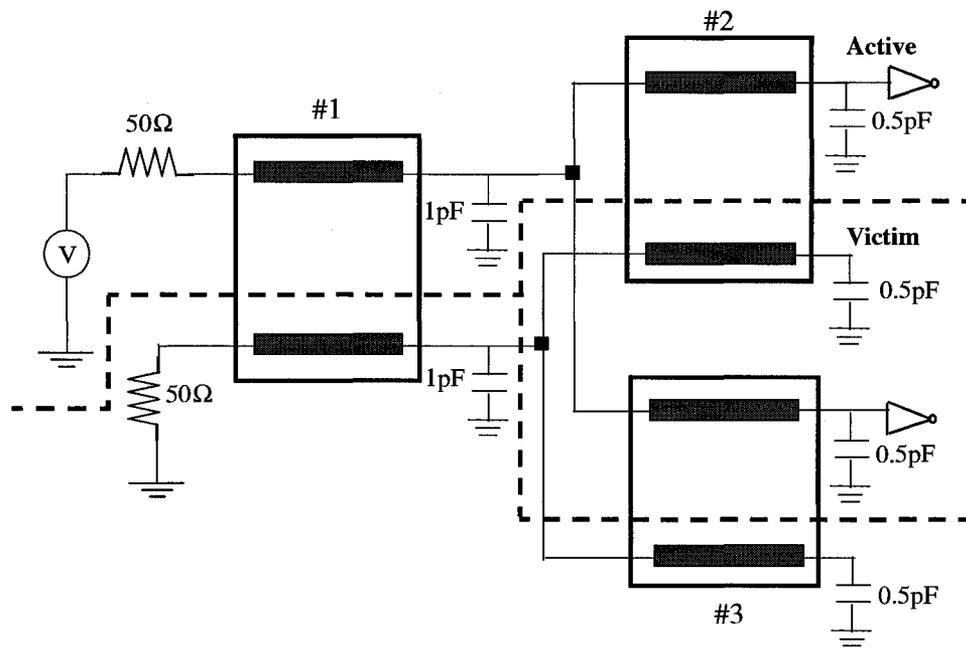


Figure 3.9: Coupled interconnect system with nonlinear terminations (Example 2)

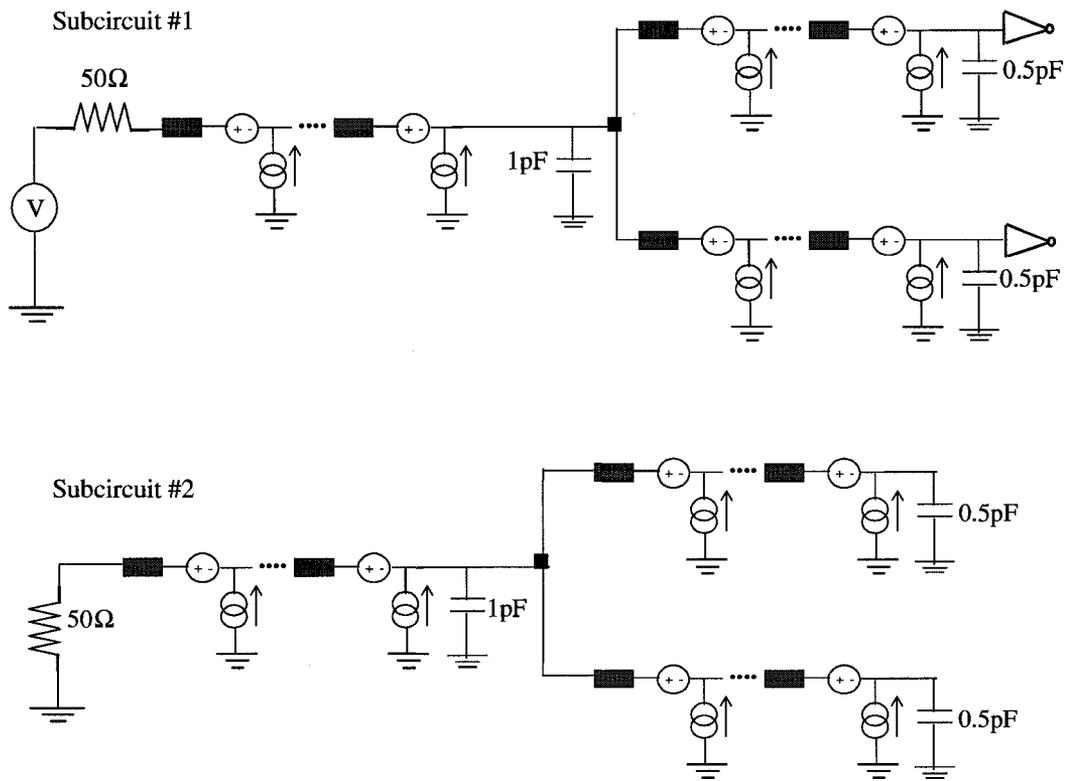


Figure 3.10: Decoupled interconnect subcircuits for Example 2

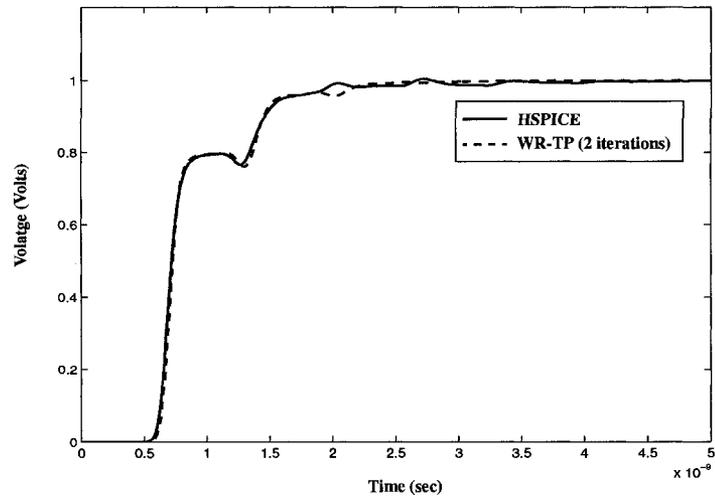


Figure 3.11: Transient response of voltage at active line far-end after 2 iterations (Example 2)

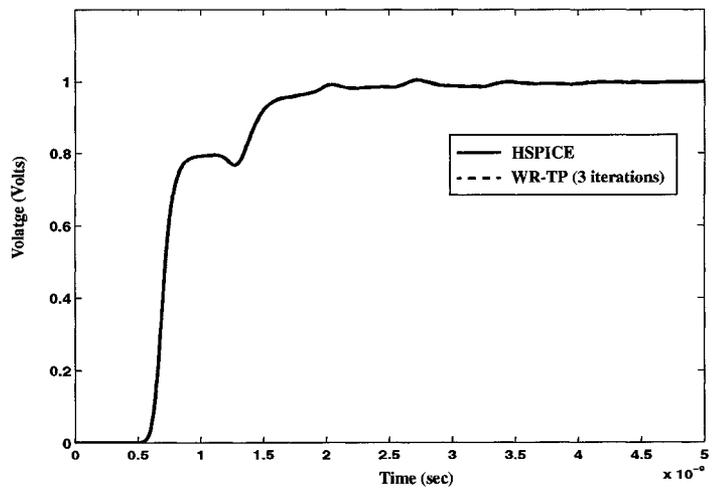


Figure 3.12: Transient response of voltage at active line far-end after 3 iterations (Example 2)

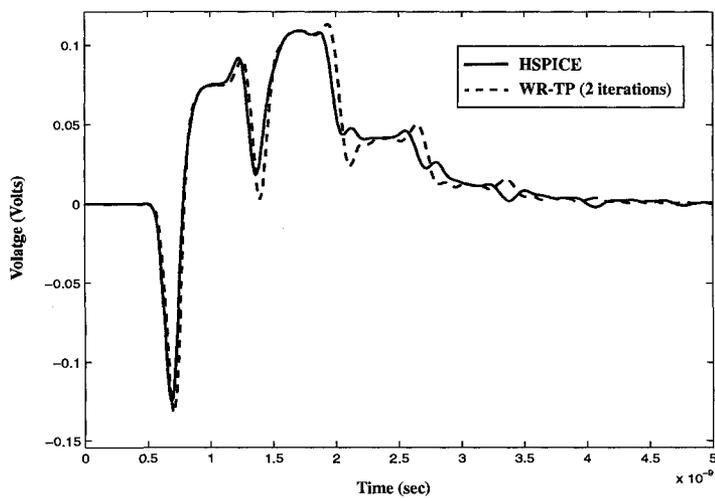


Figure 3.13: Transient response of voltage at victim line far-end after 2 iterations (Example 2)

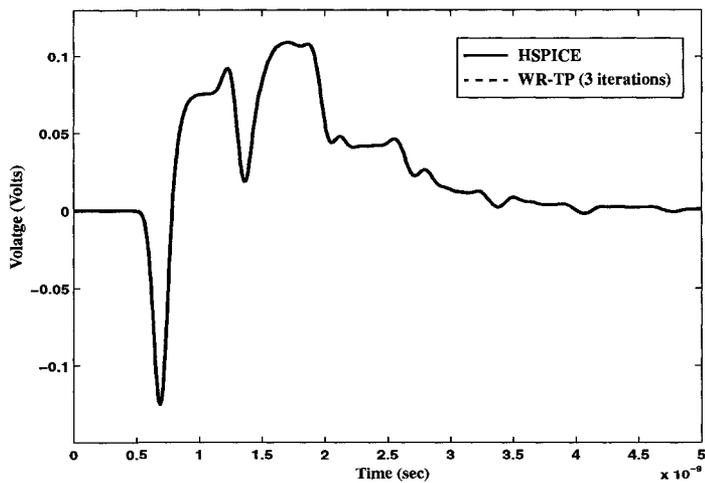


Figure 3.14: Transient response of voltage at victim line far-end after 3 iterations (Example 2)

3.6.3 Example 3

In this example, we consider highly-resistive low-inductive lines. The circuit under consideration is shown in Figure 3.15. The input voltage on line #1 is a pulse with rise/fall times of 0.1ns and a width of 5ns. Three cases were considered, $N=12$, 15 and 24. Table 3.2 shows the CPU time comparison between the proposed method and the W-element in HSPICE. Figures 3.16 - 3.17 compare a sample of the output waveforms for $N=12$ using IFFT and the W-element in HSPICE. As seen from the figures, the results from the W-element do not match those from IFFT. In contrast to this, Figures 3.18 - 3.19 compare the output waveforms using IFFT and the proposed WR-TP algorithm, which show excellent agreement after only 3 iterations.

Table 3.2: CPU Time Comparison for Example 3

Number of Lines (N)	HSPICE (W-element) (sec)	WR-TP (sec)	Speed-up
12	455	13.66	33
15	1319.61	17.10	77
24	2986.22	28	109

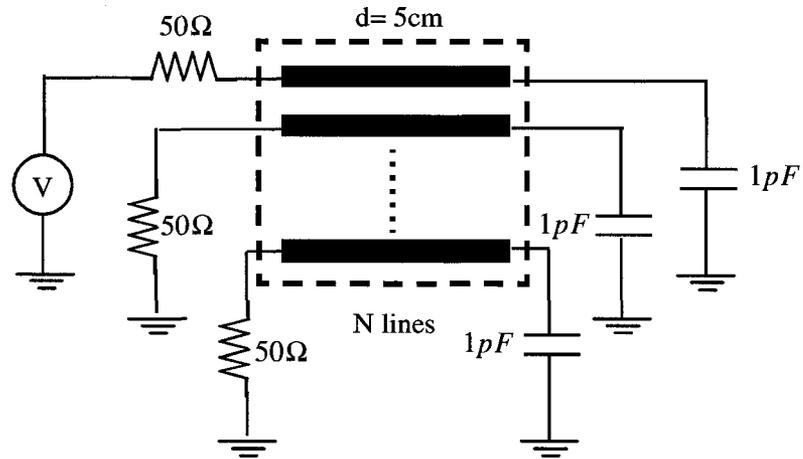


Figure 3.15: Coupled line circuit for Example 3

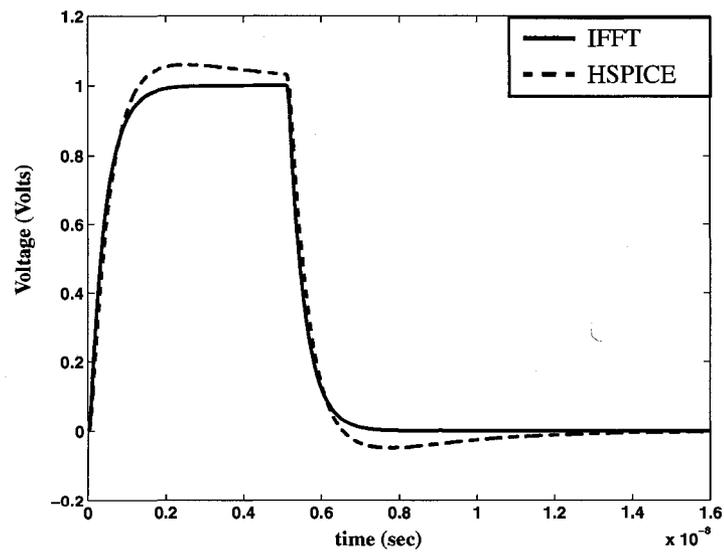


Figure 3.16: Comparison between IFFT and HSPICE for far-end voltage of line #1

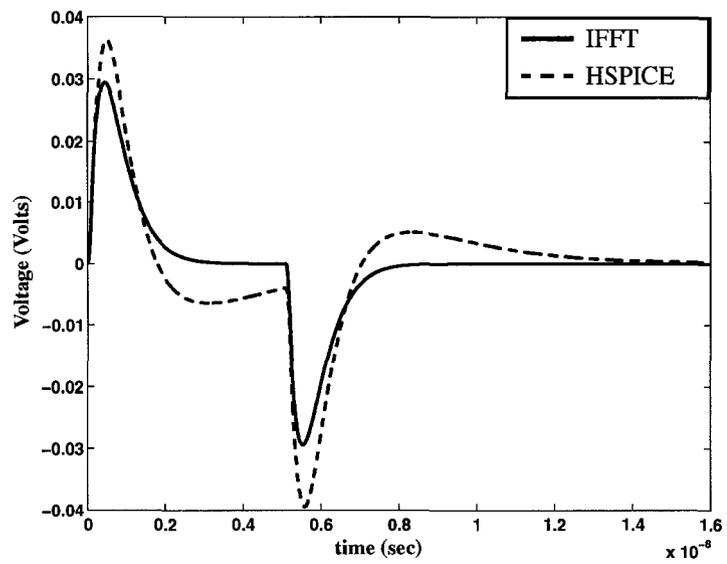


Figure 3.17: Comparison between IFFT and HSPICE for near-end voltage of line #2

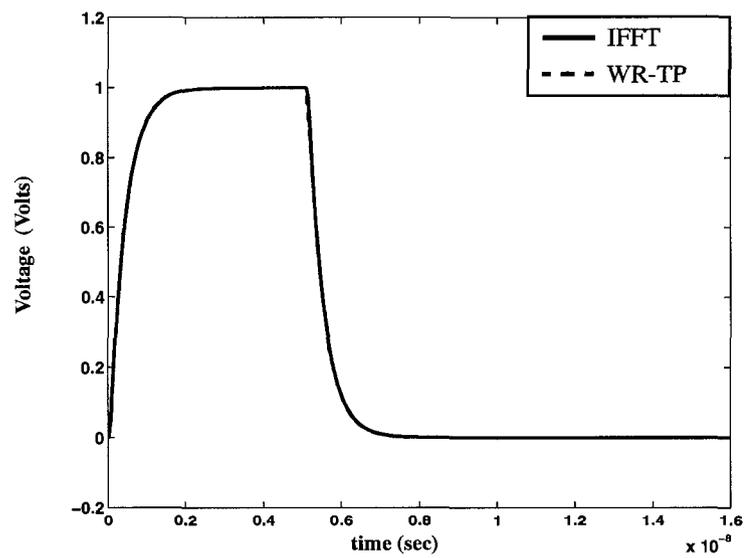


Figure 3.18: Comparison between IFFT and proposed WR-TP for far-end voltage of line #1

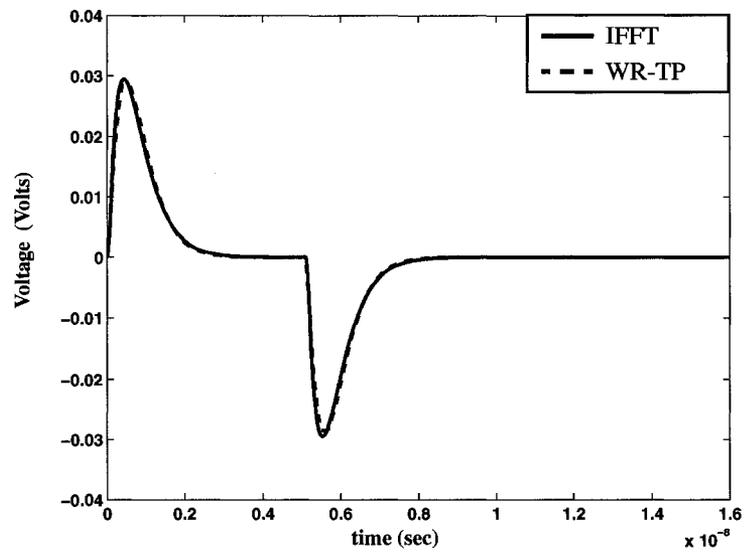


Figure 3.19: Comparison between IFFT and proposed WR-TP for near-end voltage of line #2

3.6.4 Example 4

This example illustrates the computational complexity of the proposed WR-TP method as a function of the number of coupled lines. Here we consider a lossy N -coupled on-chip interconnect circuit similar to that in Figure 3.15, except with $d = 1\text{cm}$. The number of lines was varied between $2 \leq N \leq 80$, and the lines were macromodeled using uniform lumped segmentation. The p.u.l. parameters of the coupled lines were randomly generated with the coupling coefficients ranging from 0.01 to 0.60. Zero initial guess was used for the voltage and current waveforms on all lines. In all tested cases, convergence was achieved within 5 iterations with a maximum tolerance $\leq 10^{-4}$. In contrast to this, HSPICE (W-element) could not handle or gave inaccurate results for lossy structures with more than 9 lines. Due to this deficiency of HSPICE, we compare here the results of the proposed algorithm with the IFFT output obtained from the frequency-domain solution of the coupled circuit. Figure 3.20-3.21 present a sample of the output waveforms after 3 iterations for $N = 4$, corresponding to an input pulse on line #1 with rise/fall times of 0.5ns and a width of 5ns.

In addition, Figure 3.22 shows the CPU cost in seconds as a function of the number of lines using WR-TP. As can be seen, in contrast to Figure 3.1 (which depicts the computational cost of conventional techniques), the CPU cost of the proposed algorithm increases almost *linearly* with the number of lines. To further illustrate the difference in the computational complexity, the data in Figure 3.1 and Figure 3.22 is plotted in a log scale (shown in Figure 3.23). Figure 3.23 shows that for this example, the computational complexity of conventional simulation techniques is of $O(N^{3.1})$, whereas the computational complexity of the proposed method is of $O(N)$. It is to be noted, that the linear behaviour of the proposed method emphasizes that

the CPU time required to compute the relaxation sources is in general, relatively small compared to the CPU time required to simulate the individual subcircuits. Also, it was observed that the slope of this linear behaviour depends on the specific macromodeling technique used to simulate each subcircuit.

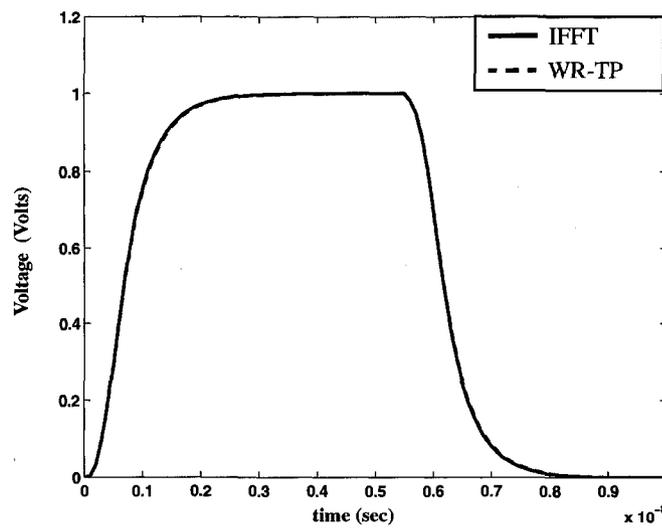


Figure 3.20: Transient response at far-end of line #1

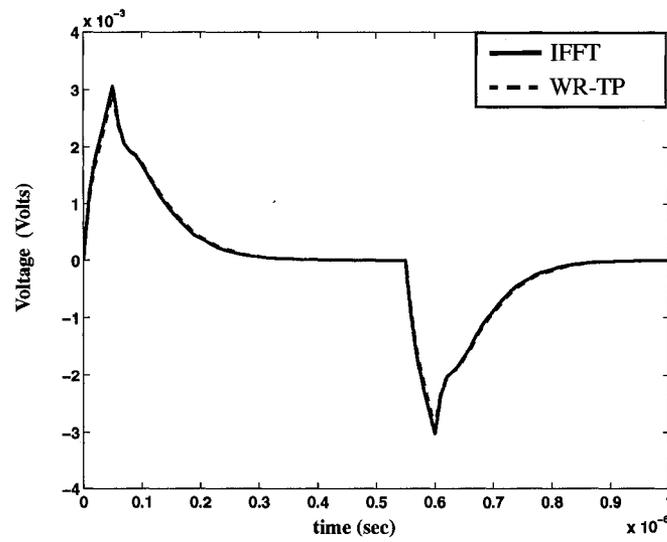


Figure 3.21: Transient response at near-end of line #2

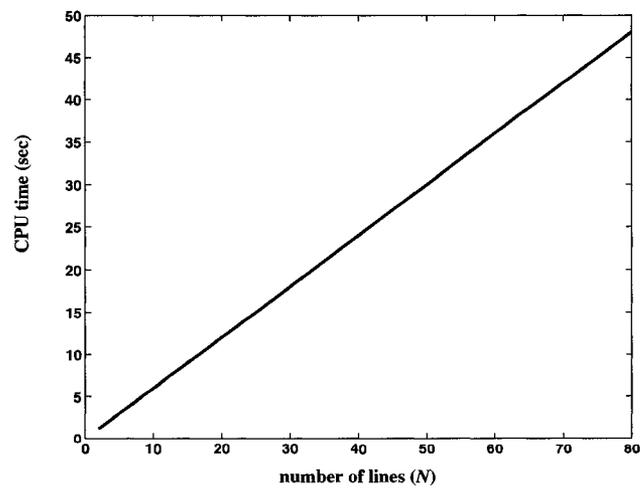


Figure 3.22: CPU cost for varying number of lines using WR-TP

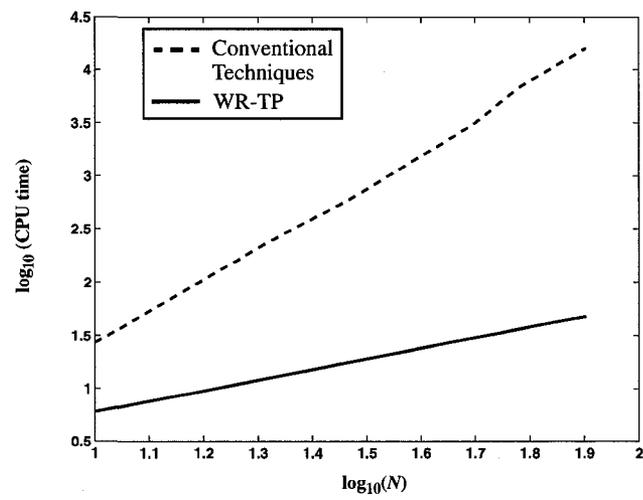


Figure 3.23: Comparison between CPU cost of proposed WR-TP method vs. conventional simulation techniques

3.7 Summary

In this chapter, a new method for macromodeling of coupled transmission lines has been presented. The proposed waveform relaxation with transverse partitioning (WR-TP) algorithm partitions the large coupled structure into smaller subcircuits, where each subcircuit can consist of as low as one single line. The coupling effects are represented by equivalent relaxation sources and an iterative algorithm has been presented for efficient evaluation of these sources. The proposed WR-TP method reduces the coupled simulation problem into a series of simulation steps, where each step is of complexity equivalent to that of simulating a single line. Using WR-TP, the CPU cost increases almost linearly with the number of lines, providing a significant speed-up as compared to conventional methods. In addition, the algorithm is highly parallelizable, which makes it suitable for efficient implementation using array processors, which has the potential to significantly further reduce the computational cost. Validity and efficiency of the proposed algorithm was demonstrated using several benchmark examples. In the next chapter, a new WR-TP algorithm for lines with frequency-dependent parameters is presented.

Chapter 4

Waveform Relaxation Techniques for Simulation of Coupled Interconnects with Frequency-Dependent Parameters

In this chapter, a new algorithm based on waveform relaxation and transverse partitioning (WR-TP) for the important and practical case of lines with frequency-dependent parameters is presented. Section 4.1 introduces the new method and Section 4.2 describes the rational function approximation and equivalent circuit representation for the frequency-dependent parameters. Section 4.3 provides details of the basis for the WR-TP algorithm as applied to lines with frequency-dependent parameters. An efficient technique for updating the WR sources is presented in Section 4.4. Sections 4.5 and 4.6 provide relevant implementation considerations and numerical examples which validate the accuracy and efficiency of the new method, respectively.

4.1 Introduction

In the previous chapter, a new method was presented to address the computational complexity of time-domain simulation of large coupled interconnects. The proposed

algorithm is based on WR using transverse partitioning (TP), where the multiconductor transmission line system is partitioned in the transverse direction, exploiting the relatively weak coupling between individual traces [17, 18]. In this case, the N coupled line circuit is partitioned into N or less independent subcircuits, where the number of lines in each subcircuit could be as low as one. The coupling effects due to the neighboring lines are represented by voltages/current sources within the subcircuits. A relaxation-based algorithm is used to iterate between the subcircuits until convergence is achieved. In general, very few iterations are required due to the relatively weak coupling between individual subcircuits. The WR-TP algorithm reduces the coupled simulation problem into a series of simulation steps, where the complexity of each step is approximately equivalent to that of simulating a single line. It was also demonstrated that using the WR-TP technique, the CPU cost increases almost linearly with the number of lines, providing a significant speed-up as compared to conventional methods. In addition, the algorithm is highly suitable for parallel implementation since the simulation of individual subcircuits can be performed simultaneously.

The WR-TP algorithm in the previous chapter considers coupled interconnects with frequency-independent per-unit-length (p.u.l) parameters. However, at relatively high frequencies, these parameters vary with frequency due to skin, edge and proximity effects [9]. It has been shown in the literature that neglecting these frequency-dependency effects leads to significant errors in transient simulation results [9, 43–45]. In this chapter, the WR-TP algorithm is extended to this practically significant case of coupled interconnects with frequency-dependent p.u.l parameters [19]. A numerical fitting approach is used to realize the FD parameters. The resulting rational function approximations are used to generate equivalent macromodels compatible with SPICE-like circuit simulators. An efficient algorithm for updating the relaxation sources is

developed. Using the proposed method, the CPU cost increases almost linearly with the number of lines, providing significant computational cost advantages.

4.2 Modeling of Frequency-Dependent Line Parameters

The per-unit-length (p.u.l.) parameters used to describe the transmission line are typically obtained from measurements, empirical formulas, or electromagnetic simulations. In general, the line parameters become frequency-dependent due to skin, edge, and proximity effects, and lossy dielectrics [9]. The p.u.l. impedance $\mathbf{Z}(s)$ and admittance $\mathbf{Y}(s)$ parameter matrices can be written as

$$\mathbf{Z}(s) = \mathbf{R}(s) + s\mathbf{L}(s); \quad \mathbf{Y}(s) = \mathbf{G}(s) + s\mathbf{C}(s) \quad (4.1)$$

where $\mathbf{R}(s) \in \mathfrak{R}^{N \times N}$, $\mathbf{L}(s) \in \mathfrak{R}^{N \times N}$, $\mathbf{C}(s) \in \mathfrak{R}^{N \times N}$, and $\mathbf{G}(s) \in \mathfrak{R}^{N \times N}$ are the FD p.u.l. resistance, inductance, capacitance and conductance matrices, respectively, and N is the number of lines.

One popular approach in modeling the FD line parameters is based on approximating (4.1) in terms of rational functions. In order to ensure the passivity of the resulting transmission line macromodel, $\mathbf{Z}(s)$ and $\mathbf{Y}(s)$ must be positive-real (PR) matrices. Several techniques have been proposed in the literature for rational approximation of tabulated data [46–50]. For our purpose here, we use the vector-fit method [49] followed by a passivity check and compensation algorithm based on the Hamiltonian matrix approach [46]. For most practical cases, the behavior of the p.u.l. parameters is a monotonic function of frequency. As a result, the poles of the rational functions are located on the negative real axis, which improves the efficiency of the passivity

check and compensation scheme.

The monotonicity of the line parameters $\mathbf{Z}(s)$ and $\mathbf{Y}(s)$ also implies that they can be easily characterized by RL and RC subnetworks, respectively. For efficient realization, the following properties of 2-element circuits were effectively utilized:

Lemma 1 *A real rational function is a driving-point impedance of an RC (RL) network if and only if all the poles and zeros are simple, lie on the nonpositive real axis, and alternate with each other, with the first critical frequency being a pole (zero) [51].*

Lemma 2 *For multiport RC (RL) network, the mutual impedance $Z_{j,k}$ contains only those poles present in $Z_{j,j}$ and $Z_{k,k}$ simultaneously. The poles must lie on the negative real axis [51].*

Based on Lemmas 1 and 2, the real poles of $\mathbf{Z}(s)$ and $\mathbf{Y}(s)$ are obtained using the vector-fit method [49]. Next, using the resulting poles, the off-diagonal rational functions are approximated using a least-squares fitting algorithm.

Equation (4.1) can be re-written in the following form

$$\begin{aligned}
\mathbf{Z}(s) &= \begin{bmatrix} Z_{1,1}(s) & Z_{1,2}(s) & \dots & Z_{1,N}(s) \\ \vdots & \vdots & \ddots & \vdots \\ Z_{N,1}(s) & Z_{N,2}(s) & \dots & Z_{N,N}(s) \end{bmatrix} \\
\mathbf{Y}(s) &= \begin{bmatrix} \sum_{k=1}^N Y_{1,k}(s) & -Y_{1,2}(s) & \dots & -Y_{1,N}(s) \\ \vdots & \vdots & \ddots & \vdots \\ -Y_{N,1}(s) & -Y_{N,2}(s) & \dots & \sum_{k=1}^N Y_{N,k}(s) \end{bmatrix} \quad (4.2)
\end{aligned}$$

where $Z_{j,k}(s)$ and $Y_{j,k}(s)$ represent the mutual impedance and admittance between lines j and k , respectively, and where $Z_{j,j}(s)$ and $Y_{j,j}(s)$ represent the *self* impedance and admittance of line j , respectively.

The elements in (4.2) can be approximated as

$$Z_{j,k}(s) \approx \Omega_{j,k} + sl_{j,k} + \sum_{\beta=1}^{n_z} \frac{sK_{j,k,\beta}}{s - p_\beta} \quad (4.3)$$

$$Y_{j,k}(s) \approx g_{j,k} + sc_{j,k} + \sum_{\beta=1}^{n_y} \frac{sB_{j,k,\beta}}{s - \lambda_\beta} \quad (4.4)$$

where n_z and n_y are the number of poles used in the approximation for the elements of $\mathbf{Z}(s)$ and $\mathbf{Y}(s)$, respectively. The approximations in (4.3) and (4.4) can be realized using equivalent RL/RC circuits, as shown in Figure 4.1 and Figure 4.2, respectively.

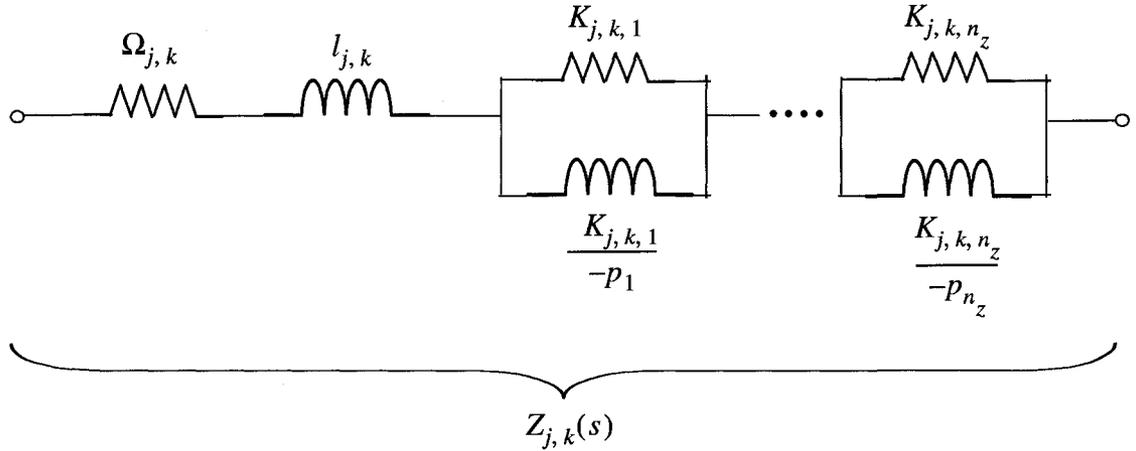


Figure 4.1: Equivalent circuit representation for $Z_{j,k}(s)$

Alternative Realization of FD line parameters using Laplace Elements

In certain commercial simulators, such as HSPICE [4], controlled voltage and current sources which have frequency-dependent proportionality factors are allowed to be used within the time-domain simulation, (in HSPICE, this feature is referred to as the Laplace Element). Specifically, the elements of $\mathbf{Z}(s)$ can be written in a rational function form as

$$Z_{j,k}(s) = \frac{\sum_{i=1}^{\alpha} a_i^{(j,k)} s^i}{\sum_{i=1}^{\beta} b_i^{(j,k)} s^i} \quad (4.5)$$

where superscript (j, k) denotes the coefficient for element $Z_{j,k}(s)$.

Exploiting this feature, the elements of $\mathbf{Z}(s)$ can be represented by controlled voltage

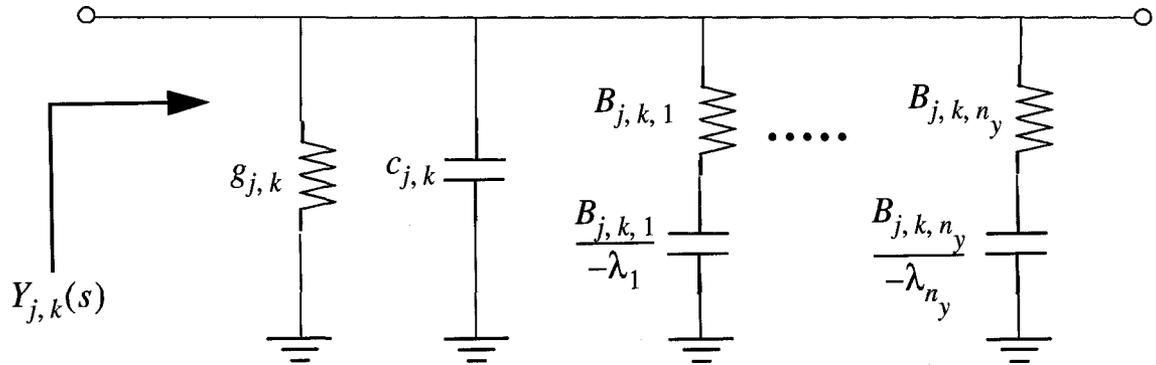


Figure 4.2: Equivalent circuit representation for $Y_{j,k}(s)$

and current sources and therefore circuit synthesis is not required. Figure 4.3 shows the equivalent circuit representation for $\mathbf{Z}(s)$ using the Laplace element. Here, the self impedances $Z_{j,j}(s)$ are realized using voltage controlled current sources (G-elements in HSPICE), and the coupling impedances are realized using voltage controlled voltage sources (E-elements in HSPICE), all with Laplace proportionality factors. For example, to realize the self impedance $Z_{1,1}(s)$, a voltage controlled current source is used, where the controlling voltage is the voltage across the element itself, and the Laplace gain is the rational function for $Z_{1,1}(s)$ as defined in (4.5). The FD admittance parameters $\mathbf{Y}(s)$ can be also realized in a form similar to that shown in Figure 4.3 using voltage controlled current sources. This capability can be used to simplify the implementation of the frequency-dependent parameters, since circuit synthesis is not required. This is especially useful since circuit synthesis can sometimes result in unrealistic component values, such as extremely large capacitor and inductor values, very small resistor values or negative elements.

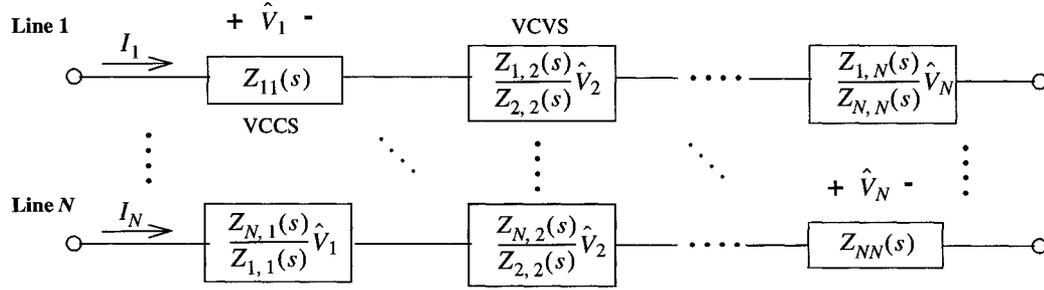


Figure 4.3: Equivalent circuit representation for $\mathbf{Z}(s)$ using Laplace Elements

4.3 Development of the WR Algorithm for F.D. Parameters

Consider the Telegrapher's equations of a transmission line in the following form [9]

$$\begin{bmatrix} \frac{\partial \mathbf{V}(x, s)}{\partial x} \\ \frac{\partial \mathbf{I}(x, s)}{\partial x} \end{bmatrix} = - \begin{bmatrix} \mathbf{0} & \mathbf{Z}(s) \\ \mathbf{Y}(s) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}(x, s) \\ \mathbf{I}(x, s) \end{bmatrix} \quad (4.6)$$

where $\mathbf{V}(x, s) \in \mathcal{C}^{N \times 1}$ and $\mathbf{I}(x, s) \in \mathcal{C}^{N \times 1}$ are complex vectors representing the voltages and currents as a function of position x and complex frequency s , $\mathbf{Z}(s)$, $\mathbf{Y}(s)$ are the p.u.l. impedance and admittance parameter matrices, respectively (given in (4.1)), and N is the number of coupled lines.

Applying the Waveform Relaxation with Transverse Partitioning (WR-TP) algorithm, which separates the coupling between the lines in (4.6), results in a recursive set of decoupled differential equations as

$$\frac{\partial \Phi_j^{(r+1)}}{\partial x} = \mathbf{F}_j(s) \Phi_j^{(r+1)}(x, s) + \Psi_j^{(r)}(x, s) \quad j = 1, 2, \dots, N \quad (4.7)$$

where

$$\Phi_j^{(r+1)}(x, s) = \begin{bmatrix} V_j^{(r+1)}(x, s) \\ I_j^{(r+1)}(x, s) \end{bmatrix} \quad (4.8)$$

$$\mathbf{F}_j(s) = \begin{bmatrix} 0 & -Z_{j,j}(s) \\ -Y_{j,j}(s) & 0 \end{bmatrix} \quad (4.9)$$

$$\begin{aligned} \Psi_j^{(r)}(x, s) &= \begin{bmatrix} E_j^{(r)}(x, s) \\ Q_j^{(r)}(x, s) \end{bmatrix} \\ &= \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & -Z_{j,k}(s) \\ -Y_{j,k}(s) & 0 \end{bmatrix} \begin{bmatrix} V_j^{(r)}(x, s) - V_k^{(r)}(x, s) \\ I_k^{(r)}(x, s) \end{bmatrix} \end{aligned} \quad (4.10)$$

where $Z_{j,j}(s)$ and $Y_{j,j}(s)$ represent the *self* impedance and admittance of line j , respectively, which can be approximated by (4.3)-(4.4), and r represents the r th iteration.

Equation (4.7) represents a single transmission line with frequency-dependent parameters and embedded relaxation sources $E_j^{(r)}(x, s)$ and $Q_j^{(r)}(x, s)$, which correspond to the coupling effects on line j due to the neighboring lines.

Recall from the previous chapter, a time-domain equivalent circuit can be realized using the distributed representation as shown in Figure 4.4. In this case, the waveform relaxation sources are approximated using a suitable numerical integration technique [39]. As a result, each line consists of a cascade of n single lines with lengths $\Delta x_1, \Delta x_2, \dots, \Delta x_n$. The waveform relaxation voltage and current sources $\tilde{e}_j^{(r)}(x_p, t)$ and $\tilde{q}_j^{(r)}(x_p, t)$ are distributed throughout the length of the lines, located at the points x_1, x_2, \dots, x_n , and are defined as

$$\tilde{e}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N z_{j,k}(t) * i_k^{(r)}(x_p, t)$$

$$\tilde{q}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N y_{j,k}(t) * [v_j^{(r)}(x_p, t) - v_k^{(r)}(x_p, t)] \quad (4.11)$$

where

$$1 \leq p \leq n$$

$$z_{j,k}(t) = F^{-1}\{Z_{j,k}(s)\} \quad (4.12)$$

$$y_{j,k}(t) = F^{-1}\{Y_{j,k}(s)\}$$

and where $F^{-1}\{ \}$ and “*” denote the inverse Fourier transform and convolution operators, respectively.

In the next section, an algorithm for updating the relaxation sources in (4.11) will be presented. Although the associated steps for the proposed method are general with respect to the macromodel used for representing the lines, for the

purpose of illustration, in the rest of this chapter we will focus on uniform lumped segmentation [9]. In this case, each line in Figure 4.4 is represented by a cascade of n lumped sections; where each of the sections on line j consist of a series impedance proportional to $Z_{j,j}$ and a parallel admittance proportional to $Y_{j,j}$.

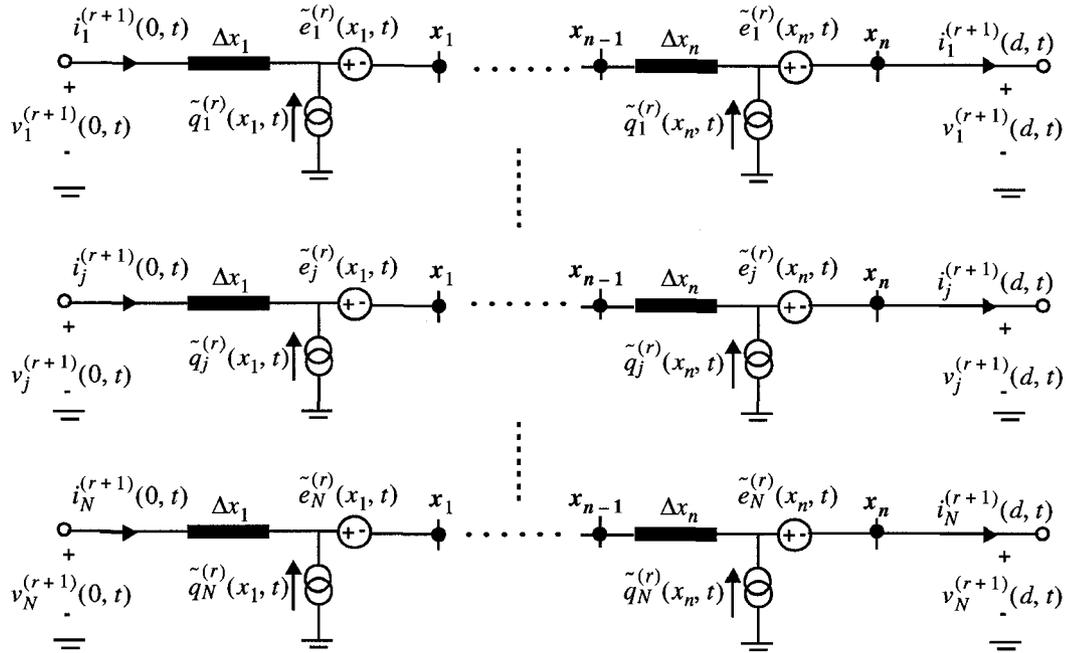


Figure 4.4: Distributed representation of decoupled network using transverse partitioning

4.4 Waveform Scaling Algorithm for Updating the Waveform Relaxation Sources

In order to obtain the coupling sources $\tilde{e}_j^{(r)}(x, t)$ and $\tilde{q}_j^{(r)}(x, t)$ in Figure 4.4, the convolution in (4.11) must be computed. Given that the poles and residues of

$Z_{j,k}(s)$ and $Y_{j,k}(s)$ are known from the parameter-fitting process, (4.11) can be easily computed using recursive convolution. However, this approach may increase the computational cost associated with the relaxation sources, since it is required for all sections ($1 \leq p \leq n$), for all lines ($1, \dots, N$), and at every iteration.

In this section, to address the above mentioned difficulties, a method based on waveform scaling that completely avoids numerical convolution is presented. The new method leads to a significant reduction in the computational cost of updating the relaxation sources.

4.4.1 Waveform Scaling Approach

For illustration purposes, we will consider here the p^{th} section of an N coupled line circuit, as shown in Figure 4.5.

The relaxation voltage source $\tilde{e}_j^{(r)}(x_p, t)$ for line j , which represents the coupling effects on line j due to the neighboring lines, can be written at a given iteration (r) as follows

$$\tilde{e}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N z_{j,k}(t) * i_k^{(r)}(x_{p-1}, t) \Delta x_p \quad (4.13)$$

where $i_k^{(r)}(x_{p-1}, t)$ is the current through line k , as shown in Figure 4.5.

Substituting (4.3) in (4.13) yields

$$\begin{aligned} \tilde{e}_j^{(r)}(x_p, t) = -\Delta x_p \sum_{\substack{k=1 \\ k \neq j}}^N \left[\Omega_{j,k} i_k^{(r)}(x_{p-1}, t) + l_{j,k} \frac{di_k^{(r)}(x_{p-1}, t)}{dt} \right. \\ \left. + \sum_{\beta=1}^{n_z} i_k^{(r)}(x_{p-1}, t) * F^{-1} \left\{ \frac{sK_{j,k,\beta}}{s - p_\beta} \right\} \right] \quad (4.14) \end{aligned}$$

Referring to Figure 4.5, it can be seen that the voltage across the resistor $\Omega_{k,k}\Delta x_p$ on line k , is simply

$$v_{R_k} = \Omega_{k,k}\Delta x_p i_k^{(r)}(x_{p-1}, t) \quad (4.15)$$

and the voltage across the inductor $l_{k,k}\Delta x_p$ is

$$v_{L_k} = l_{k,k}\Delta x_p \frac{di_k^{(r)}(x_{p-1}, t)}{dt} \quad (4.16)$$

Thus, the first two terms inside the summation in (4.14) are given by

$$\begin{aligned} \Omega_{j,k}\Delta x_p i_k^{(r)}(x_{p-1}, t) &= \frac{\Omega_{j,k}}{\Omega_{k,k}} \left(\Omega_{k,k}\Delta x_p i_k^{(r)}(x_{p-1}, t) \right) = \frac{\Omega_{j,k}}{\Omega_{k,k}} v_{R_k} \\ l_{j,k}\Delta x_p \frac{di_k^{(r)}(x_{p-1}, t)}{dt} &= \frac{l_{j,k}}{l_{k,k}} \left(l_{k,k}\Delta x_p \frac{di_k^{(r)}(x_{p-1}, t)}{dt} \right) = \frac{l_{j,k}}{l_{k,k}} v_{L_k} \end{aligned} \quad (4.17)$$

Next, in order to use the same approach for the third term in (4.14), we utilize Lemma 2 (Section 4.2), *which states that the poles of $Z_{j,k}(s)$ are a subset of the poles of $Z_{j,j}(s)$ and $Z_{k,k}(s)$* . Consider the voltage across the β^{th} RL tank circuit on line k , which can be written as

$$v_{p_\beta}^{(k,k)} = i_k^{(r)}(x_{p-1}, t) * F^{-1} \left\{ \frac{sK_{k,k,\beta}\Delta x_p}{s - p_\beta} \right\} \quad (4.18)$$

Using (4.18), we can write

$$\begin{aligned}
& i_k^{(r)}(x_{p-1}, t) * F^{-1} \left\{ \frac{sK_{j,k,\beta} \Delta x_p}{s - p_\beta} \right\} \\
&= \frac{K_{j,k,\beta}}{K_{k,k,\beta}} \left(i_k^{(r)}(x_{p-1}, t) * F^{-1} \left\{ \frac{sK_{k,k,\beta} \Delta x_p}{s - p_\beta} \right\} \right) \\
&= \frac{K_{j,k,\beta}}{K_{k,k,\beta}} v_{p_\beta}^{(k,k)}
\end{aligned} \tag{4.19}$$

Substituting (4.17) and (4.19) into (4.14), $\tilde{e}_j^{(r)}(x_p, t)$ can be re-written as

$$\tilde{e}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N \left(\frac{\Omega_{j,k}}{\Omega_{k,k}} v_{R_k} + \frac{l_{j,k}}{l_{k,k}} v_{L_k} + \sum_{\beta=1}^{n_z} \frac{K_{j,k,\beta}}{K_{k,k,\beta}} v_{p_\beta}^{(k,k)} \right) \tag{4.20}$$

Thus, using the technique presented above, the relaxation source $\tilde{e}_j^{(r)}(x_p, t)$ can be obtained as a linear combination of the voltages on the neighboring lines.

Similarly, the same method can be used to calculate the relaxation source $\tilde{q}_j^{(r)}(x_p, t)$. The current source $\tilde{q}_j^{(r)}(x_p, t)$ for line j (see Figure 4.6) is given by

$$\tilde{q}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N y_{j,k}(t) \Delta x_p * \left[v_j^{(r)}(x_p, t) - v_k^{(r)}(x_p, t) \right] \tag{4.21}$$

Using (4.4), (4.21) can be expressed as

$$\begin{aligned}
& \tilde{q}_j^{(r)}(x_p, t) \\
&= \Delta x_p \sum_{\substack{k=1 \\ k \neq j}}^N \left[g_{j,k} v_k^{(r)}(x_p, t) + c_{j,k} \frac{dv_k^{(r)}(x_p, t)}{dt} + \sum_{\beta=1}^{n_y} v_k^{(r)}(x_p, t) * F^{-1} \left\{ \frac{s B_{j,k,\beta}}{s - \lambda_\beta} \right\} \right. \\
&\quad \left. - g_{j,k} v_j^{(r)}(x_p, t) - c_{j,k} \frac{dv_j^{(r)}(x_p, t)}{dt} - \sum_{\beta=1}^{n_y} v_j^{(r)}(x_p, t) * F^{-1} \left\{ \frac{s B_{j,k,\beta}}{s - \lambda_\beta} \right\} \right] \quad (4.22)
\end{aligned}$$

Similar to the approach used to derive (4.20) and with some mathematical manipulation, (4.22) can be re-written as

$$\tilde{q}_j^{(r)}(x_p, t) = \sum_{\substack{k=1 \\ k \neq j}}^N \left[\frac{g_{j,k}}{g_{k,k}} i_{G_k} + \frac{c_{j,k}}{c_{k,k}} i_{C_k} + \sum_{\beta=1}^{n_y} \frac{B_{j,k,\beta}}{B_{k,k,\beta}} i_{\lambda_\beta}^{(k,k)} - \frac{g_{j,k}}{g_{j,j}} i_{G_j} - \frac{c_{j,k}}{c_{j,j}} i_{C_j} - \sum_{\beta=1}^{n_y} \frac{B_{j,k,\beta}}{B_{j,j,\beta}} i_{\lambda_\beta}^{(j,j)} \right] \quad (4.23)$$

where the branch currents (Figure 4.6) in (4.23) are given by

$$\begin{aligned}
i_{G_j} &= g_{j,j} \Delta x_p v_j^{(r)}(x_p, t) \\
i_{G_k} &= g_{k,k} \Delta x_p v_k^{(r)}(x_p, t) \\
i_{C_j} &= c_{j,j} \Delta x_p \frac{dv_j^{(r)}(x_p, t)}{dt} \\
i_{C_k} &= c_{k,k} \Delta x_p \frac{dv_k^{(r)}(x_p, t)}{dt} \\
i_{\lambda_\beta}^{(j,j)} &= v_j^{(r)}(x_p, t) * F^{-1} \left\{ \frac{s B_{j,j,\beta} \Delta x_p}{s - \lambda_\beta} \right\} \\
i_{\lambda_\beta}^{(k,k)} &= v_k^{(r)}(x_p, t) * F^{-1} \left\{ \frac{s B_{k,k,\beta} \Delta x_p}{s - \lambda_\beta} \right\}
\end{aligned} \quad (4.24)$$

Therefore, using (4.20) and (4.23), performing the numerical convolution is no longer required. The relaxation sources for a given line can be computed by scaling the voltages and currents on the neighboring lines. As a result, this provides a significant reduction in the computational cost associated with updating the relaxation sources.

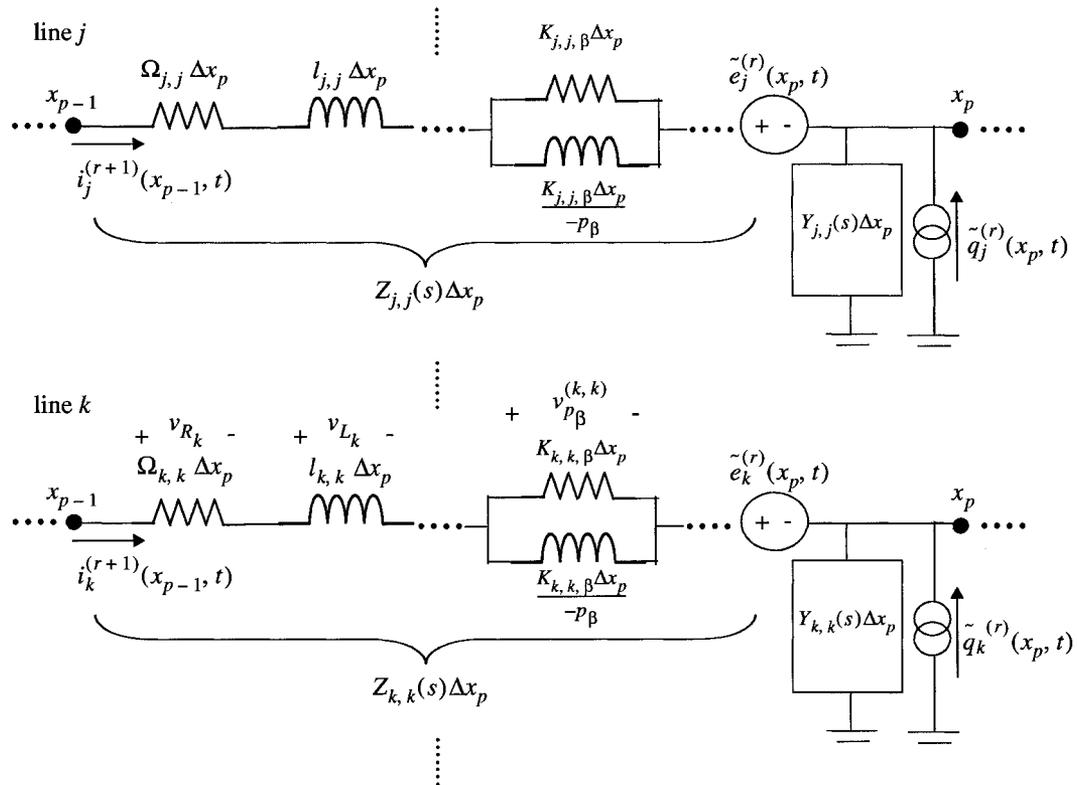


Figure 4.5: Equivalent circuit demonstrating the computation of $\tilde{e}_j^{(r)}(x_p, t)$

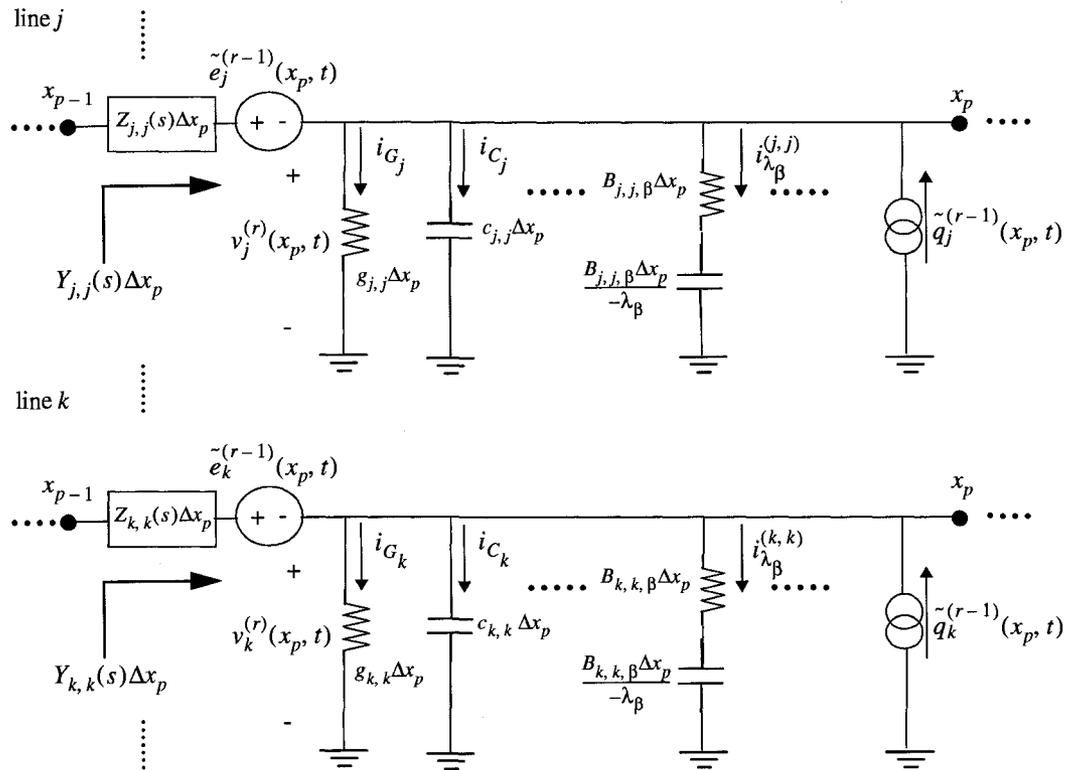


Figure 4.6: Equivalent circuit demonstrating the computation of $\tilde{q}_j^{(r)}(x_p, t)$

4.5 Implementation Considerations

Several relevant aspects regarding the implementation of the proposed waveform relaxation scheme are worth mentioning.

4.5.1 Using Controlled Sources for Updating the Relaxation Sources

As discussed in Section 4.2, certain commercial simulators such as HSPICE [4] allow controlled voltage and current sources which have frequency-dependent proportionality factors (Laplace Elements). Exploiting this feature, the diagonal elements of $\mathbf{Z}(s)$ and $\mathbf{Y}(s)$ can be represented by controlled voltage and current sources and circuit synthesis is no longer required (see Figure 4.3). This capability can also be used to simplify the implementation of updating the relaxation sources. To illustrate, consider the relaxation source $\tilde{e}_j^{(r)}(x_p, t)$ for line j , which can be expressed in the time-domain as

$$\tilde{e}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N F^{-1}\{H_{j,k}(s)\} * \hat{v}_k^{(r)}(x_{p-1}, t) \quad (4.25)$$

where $H_{j,k}(s)$ is the ratio of the mutual impedance to the self impedance, given by

$$H_{j,k}(s) = \frac{Z_{j,k}(s)}{Z_{k,k}(s)} = \frac{\sum_{i=1}^{\alpha} a_i s^i}{\sum_{i=1}^{\phi} b_i s^i} \quad (4.26)$$

and the voltage $\hat{v}_k^{(r)}(x_{p-1}, t) = F^{-1}\{Z_{k,k}(s)\Delta x_p\} * i_k^{(r)}(x_{p-1}, t)$ is the voltage across the circuit used to realize $Z_{k,k}(s)\Delta x_p$, as shown in Figure 4.7.

The convolution in (4.25) can be represented by the use of controlled sources. Instead of computing the relaxation source $\tilde{e}_j^{(r)}(x_p, t)$ using the waveform scaling approach, a

series of voltage controlled voltage sources can be included in the circuit (see Figure 4.7), where the controlling voltage is the waveform $\hat{v}_k(x_{p-1}, t)$ from the previous iteration, and where the proportionality factor is $H_{j,k}(s)$.

The same approach can be used to compute the relaxation source $\tilde{q}_j^{(r)}(x_p, t)$. In this case, $\tilde{q}_j^{(r)}(x_p, t)$ in (4.11) can be re-written as

$$\tilde{q}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N \left[F^{-1}\{W_{j,k}^j(s)\} * \hat{i}_j^{(r)}(x_p, t) - F^{-1}\{W_{j,k}^k(s)\} * \hat{i}_k^{(r)}(x_m, t) \right] \quad (4.27)$$

where

$$W_{j,k}^j(s) = \frac{Y_{j,k}(s)}{Y_{j,j}(s)}, \quad W_{j,k}^k(s) = \frac{Y_{j,k}(s)}{Y_{k,k}(s)} \quad (4.28)$$

and the current $\hat{i}_k^{(r)}(x_p, t) = F^{-1}\{Y_{k,k}(s)\Delta x_p\} * v_k^{(r)}(x_p, t)$ is the branch current through the circuit used to realize $Y_{k,k}(s)\Delta x_p$.

Similarly, performing the convolution in (4.27) is equivalent to including a parallel combination of current controlled current sources in the circuit netlist, where the controlling currents $\hat{i}_k(x_p, t)$ are obtained directly from the previous iteration, and the proportionality factors are given in (4.28). Note that the coefficients of the proportionality factors for $\tilde{e}_j^{(r)}(x_p, t)$ and $\tilde{q}_j^{(r)}(x_p, t)$ (given by (4.26) and (4.28)) are known *a priori*, from the rational function approximation for the line parameters.

It is important to mention that both approaches presented for updating the relaxation sources provide significant computational cost advantages in comparison to performing the numerical convolution. Moreover, the difference in the computational

cost between the two schemes is negligible. However, the availability of the ‘‘Laplace Element’’ [4] feature merely simplifies the implementation of the WR-TP algorithm.

It is also to be noted that, using the WR-TP method allows for each individual subcircuit to be solved completely independently. Thus, in the circuit simulator, multirate simulation may be used, where each subcircuit is solved at its own speed and time step. At a given iteration, the simulator accesses the coupling waveforms obtained from the previous iteration (which are in the form of tabulated data). In the case that the simulator needs a value of the coupling waveform at an intermediate time point, a suitable interpolation scheme (such as SPLINE fitting [52]) may be used.

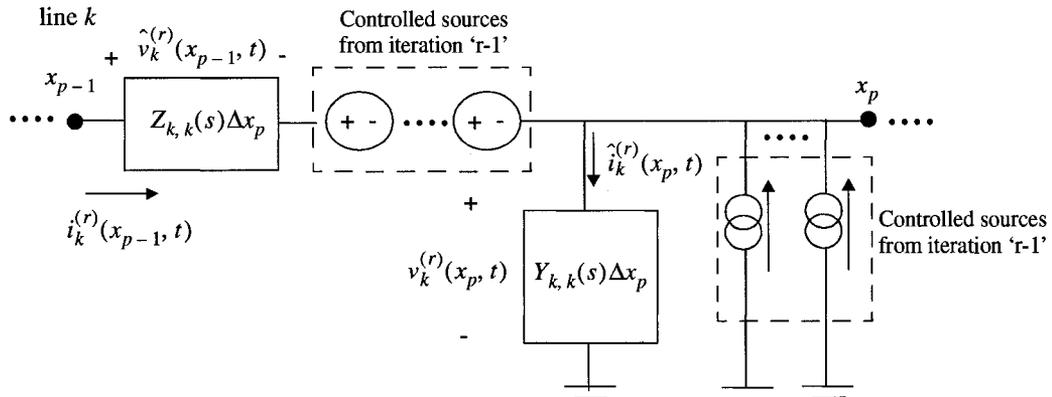


Figure 4.7: Calculation of relaxation sources at iteration r using Laplace Elements

4.5.2 Convergence and Diagonal Dominance

Since the p.u.l. admittance matrix $\mathbf{Y}(s)$ is strictly diagonally dominant [53], it follows from (4.2) that

$$\left| \sum_{k=1}^N Y_{j,k} \right| > \sum_{\substack{k=1 \\ k \neq j}}^N |Y_{j,k}| \quad j, k = 1, 2, \dots, N \quad (4.29)$$

This fact is used to accelerate the convergence of the overall WR-TP algorithm by re-writing (4.7)-(4.10) in the following form

$$\frac{\partial \Phi_j^{(r+1)}}{\partial x} = \hat{\mathbf{F}}_j \Phi_j^{(r+1)}(x, s) + \hat{\Psi}_j^{(r)}(x, s) \quad j = 1, 2, \dots, N \quad (4.30)$$

where

$$\Phi_j^{(r+1)}(x, s) = \begin{bmatrix} V_j^{(r+1)}(x, s) \\ I_j^{(r+1)}(x, s) \end{bmatrix} \quad \hat{\mathbf{F}}_j = \begin{bmatrix} 0 & -Z_{j,j}(s) \\ -\sum_{k=1}^N Y_{j,k}(s) & 0 \end{bmatrix} \quad (4.31)$$

$$\hat{\Psi}_j^{(r)}(x, s) = \begin{bmatrix} \hat{E}_j^{(r)}(x, s) \\ \hat{Q}_j^{(r)}(x, s) \end{bmatrix} = \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & -Z_{j,k}(s) \\ -Y_{j,k}(s) & 0 \end{bmatrix} \begin{bmatrix} V_k^{(r)}(x, s) \\ I_k^{(r)}(x, s) \end{bmatrix} \quad (4.32)$$

$$\begin{aligned} Z_{j,k}(s) &\approx \Omega_{j,k} + sl_{j,k} + \sum_{\beta=1}^{n_z} \frac{sK_{j,k,\beta}}{s - p_\beta} \\ Y_{j,k}(s) &\approx g_{j,k} + sc_{j,k} + \sum_{\beta=1}^{n_y} \frac{sB_{j,k,\beta}}{s - \lambda_\beta} \end{aligned} \quad (4.33)$$

In this case, the admittance $Y_{j,j}$ in Figure 4.5 is replaced by $\sum_{k=1}^N Y_{j,k}(s)$ (which is just the element in position (j, j) of the overall \mathbf{Y} matrix), and the relaxation current source $\tilde{q}_j^{(r)}(x_p, t)$ is modified to

$$\tilde{q}_j^{(r)}(x_p, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N y_{j,k}(t) * \left[v_k^{(r)}(x_p, t) \right] \quad (4.34)$$

Although the equations in (4.30)-(4.34) are mathematically equivalent to those in (4.7)-(4.12), they offer better convergence properties [53, 54], since they guarantee the diagonal dominance of the subcircuit equations in (4.30).

4.6 Numerical Results

In this section, four case studies are presented. The first three examples demonstrate the validity, accuracy and convergence properties of the proposed WR-TP algorithm and the fourth example considers several cases of a large number of coupled lines, and numerically illustrates the linear relationship between the CPU cost and the number of lines.

4.6.1 Example 1

In this example, we consider a nonlinear circuit (Figure 4.8) consisting of four on-chip coupled lines with frequency-dependent p.u.l. resistance ($\mathbf{R}(s)$) and inductance ($\mathbf{L}(s)$), constant \mathbf{C} and $\mathbf{G} = 0$. The extracted line parameters are based on data supplied by IBM [55]. The frequency-dependent impedance was fitted using the approach described in Section 4.2. Figure 4.9 shows a sample of the fitting results as compared to the original extracted data. To start the waveform relaxation iterations, the terminal current and voltage waveforms across all the lines were initialized to

zero values, and convergence was achieved after 3 iterations. In order to validate the accuracy of the WR-TP method, the coupled transmission line in Figure 4.8 was represented by its *coupled* lumped equivalent circuit which was generated using the technique described in Section 4.2 and the resulting circuit was simulated in HSPICE [4]. Figures 4.10 and 4.11 show a sample of the output waveforms for an input pulse with rise/fall times of $0.5ns$ and a pulse width of $5ns$. As seen, the results are in very good agreement.

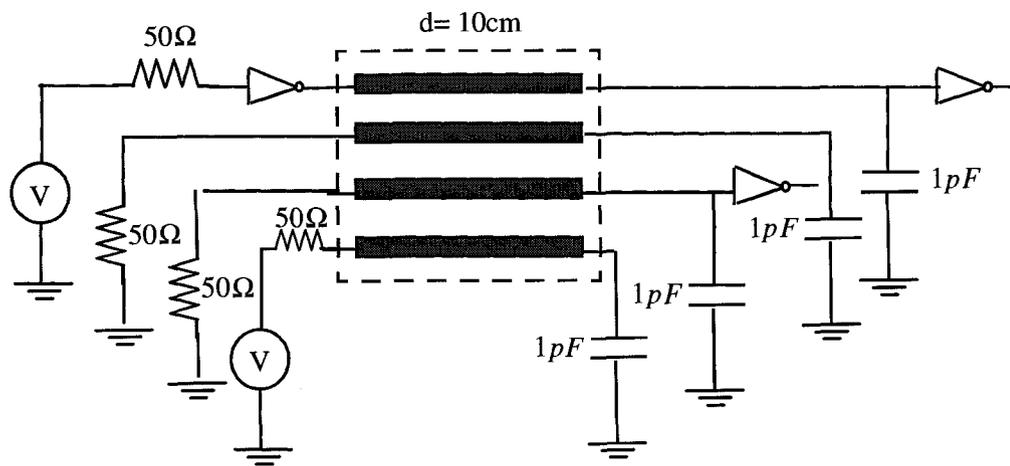


Figure 4.8: Nonlinear circuit with four on-chip coupled lines (Example 1)

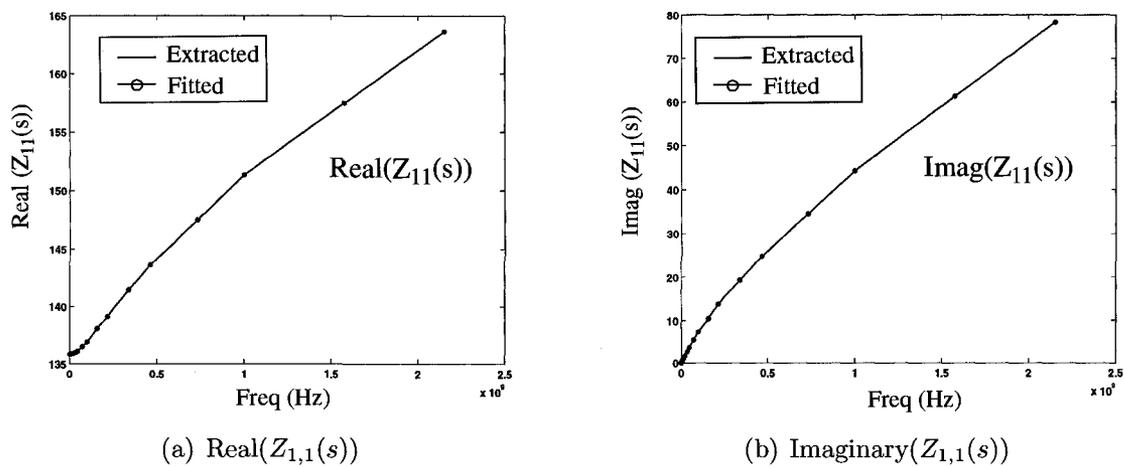


Figure 4.9: Comparison of fitted results vs. original data for $Z_{1,1}(s)$

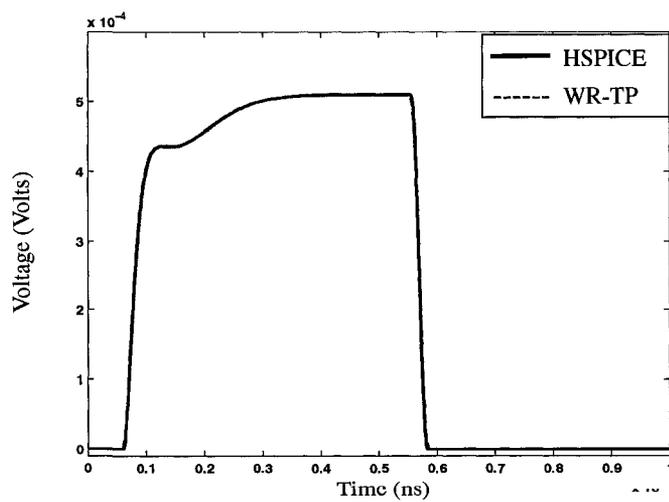


Figure 4.10: Transient response at far-end of active line (line #1) (Example 1)

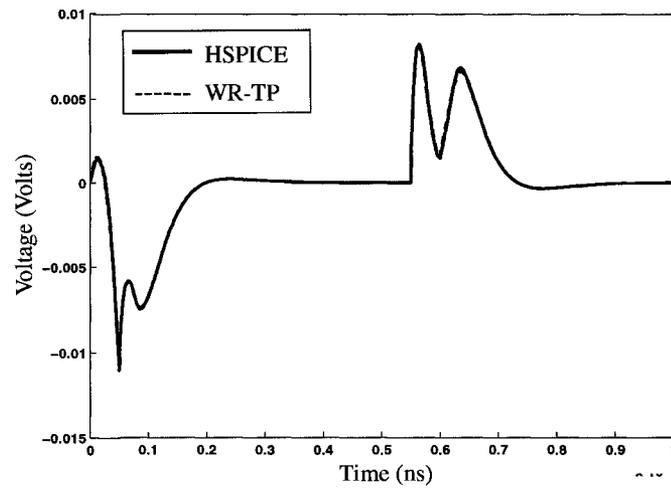


Figure 4.11: Transient response at near-end of victim line (line #2) (Example 1)

4.6.2 Example 2

In this example, the number of lines used in the previous example was increased to twelve. The input voltage sources (on lines 1, 5 and 10) are input pulses with rise/fall times of $0.5ns$ and pulse widths of $5ns$. To validate the accuracy of the WR-TP algorithm, the circuit in Figure 4.12 was simulated in the frequency-domain using the exact stamp [9] of the coupled lines, and the time-domain response was obtained using the IFFT. Figures 4.13 and 4.14 show a sample of the output waveforms (after 3 iterations) compared to the IFFT of the frequency-domain solution. As seen from the plots, the results are in excellent agreement.

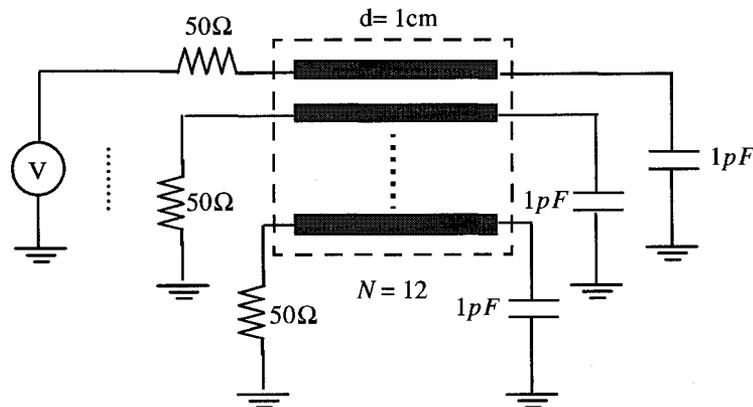


Figure 4.12: Linear circuit consisting of twelve on-chip coupled lines (Example 2)

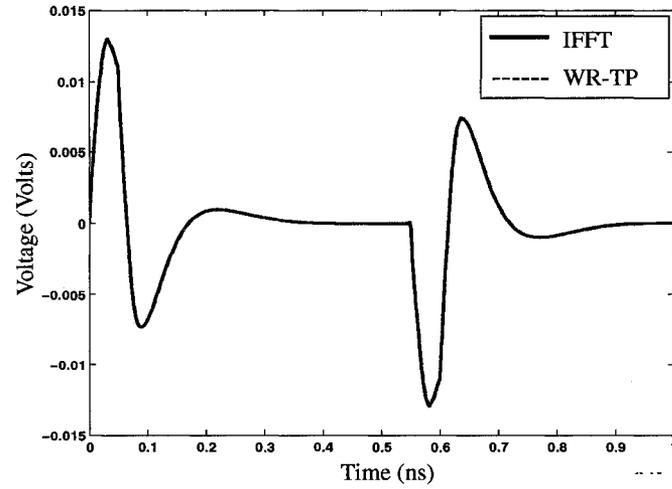


Figure 4.13: Transient response at near-end of victim line (line #3) (Example 2)

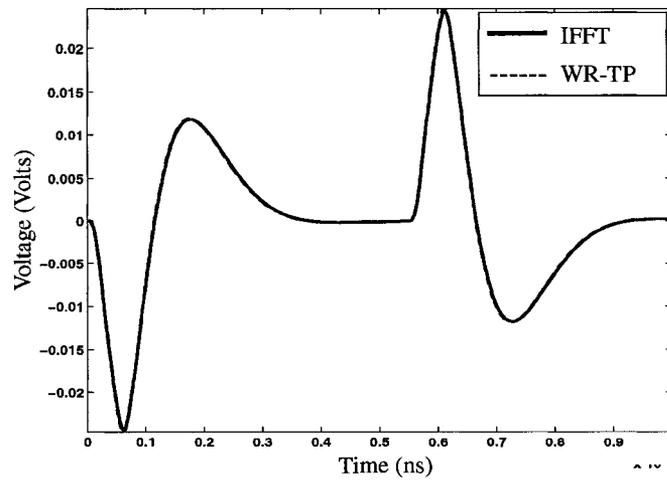


Figure 4.14: Transient response at far-end of victim line (line #12) (Example 2)

4.6.3 Example 3

In this example, we consider four coupled multichip module (MCM) lines of length=10cm. In this case, $\mathbf{R}(s)$, $\mathbf{L}(s)$, $\mathbf{G}(s)$ and $\mathbf{C}(s)$ are all frequency-dependent and based on the parameters used in [1]. The circuit in Figure 4.15 was simulated using the proposed WR-TP algorithm with uniform lumped segmentation, starting with initial values of zero for the voltage and current waveforms. Figures 4.16 and 4.17 present a sample of the output waveforms after 3 iterations, compared to the IFFT of the frequency-domain solution, corresponding to an input pulse with rise/fall times of 0.5ns and a width of 5ns. As seen from the figures, the results from the WR-TP method are in very good agreement with the results from the IFFT.

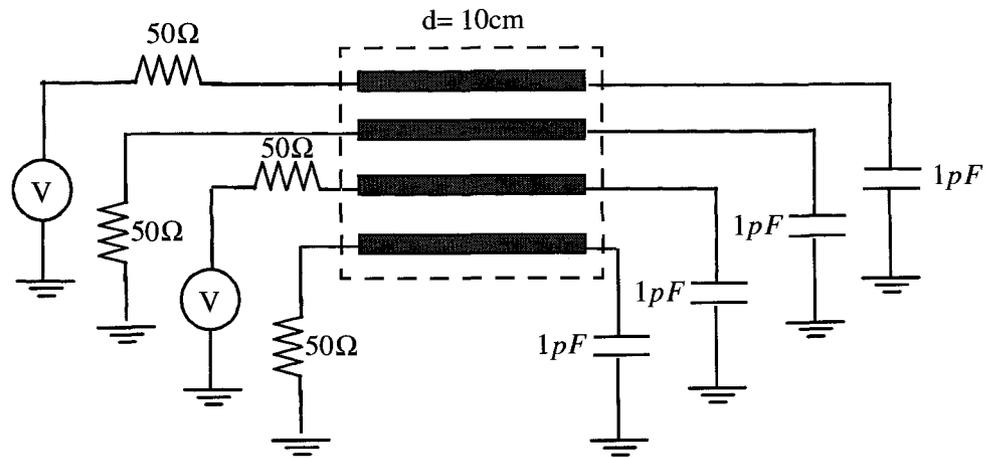


Figure 4.15: Four coupled lines circuit (Example 3)

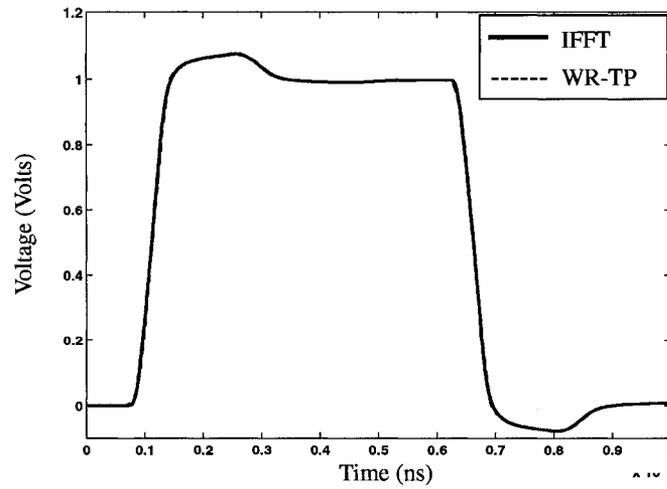


Figure 4.16: Transient response at far-end of active line (line #3) (Example 3)

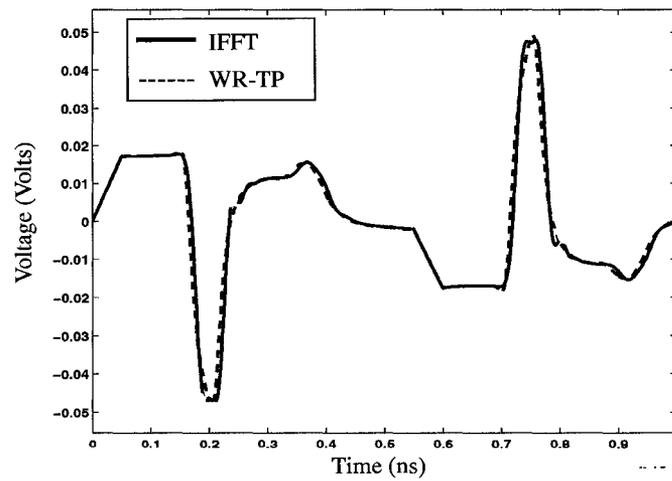


Figure 4.17: Transient response at near-end of victim line (line #4) (Example 3)

4.6.4 Example 4

This example provides an analysis of the computational complexity of the proposed WR-TP algorithm as a function of the number of lines (N). In this numerical experiment, the number of frequency-dependent on-chip coupled lines (used in Examples 1 and 2) was varied in the range of $2 \leq N \leq 24$ (Figure 4.18). Zero initial guess was used for the voltage and current waveforms on all lines. In all cases, convergence was achieved in less than 5 iterations. Figure 4.19 shows the computational cost as a function of the number of lines. As expected, the computational cost of the proposed WR-TP algorithm increases almost linearly with the number of lines. In contrast to this, as was illustrated in [17], the CPU cost of conventional circuit simulators even for the simpler case of frequency-independent p.u.l. parameters grows with N^α , where $3 \leq \alpha \leq 4$. Also, it is worth mentioning that in the case of lines with frequency-dependent parameters, for this example, HSPICE [4] failed for more than eight lines. It is to be noted, that the linear behaviour of the computational complexity of the WR-TP method emphasizes that the CPU time required for updating the relaxation sources is in general, relatively small compared to the CPU time required to simulate the individual subcircuits. In addition, the slope of this linear behaviour depends on the specific macromodeling technique used to simulate each circuit.

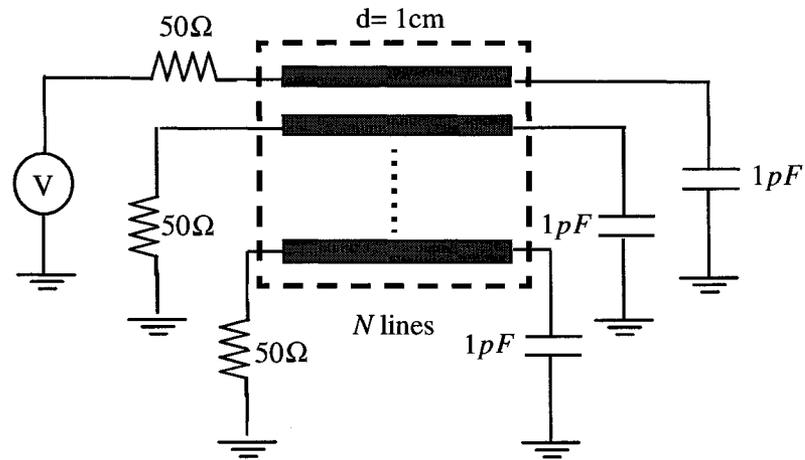


Figure 4.18: Coupled lines circuit with varying N (Example 4)

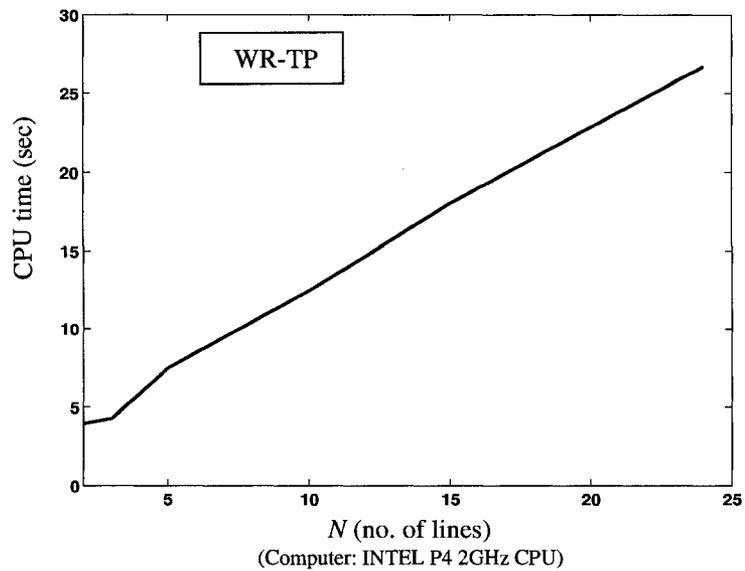


Figure 4.19: Computational cost as a function of the number of lines N (Example 4)

4.7 Summary

In this chapter, a new method for macromodeling coupled transmission lines with frequency-dependent parameters has been presented. The proposed algorithm partitions the N -coupled line circuit into N or less individual subcircuits. The coupling effects between subcircuits is represented by voltage and current relaxation sources, and an efficient method for calculation of these sources has been presented. The frequency-dependent parameters of the lines are approximated using an efficient fitting algorithm. The resulting rational function approximations are used to generate equivalent macromodels compatible with SPICE-like circuit simulators. Using the proposed WR-TP algorithm, the CPU cost increases only *linearly* with the number of lines, providing a significant speed-up compared to conventional techniques. The accuracy and efficiency of the WR-TP algorithm was demonstrated using various benchmark examples. In the next chapter, simplified delay extraction-based macromodels for several special cases are described.

Chapter 5

Simplified Delay Extraction-Based Passive Transmission Line Macromodels

In Chapter 2, the delay extraction-based passive transmission line macromodeling algorithm (DEPACT) was discussed. In practice, there are several special cases where the DEPACT macromodel can be further reduced. This chapter presents the details of a new simplified delay extraction-based macromodeling algorithm which handles these special cases. The proposed simplifications are of particular importance for advanced transverse partitioning with delay extraction which will be described in the next chapter.

The chapter is organized as follows. Section 5.1 describes the macromodeling algorithm for the case when the line parameters are frequency-independent. Section 5.2 presents the macromodeling algorithm for several practical cases when only $\mathbf{R}(s)$ and $\mathbf{L}(s)$ are frequency-dependent. Section 5.3 provides numerical examples which validate the accuracy and efficiency of the new method.

5.1 Macromodels for Lines with Frequency-Independent Parameters

As the line lengths and frequencies increase, lumped interconnect models may no longer be valid. As a result, the interconnects are treated as distributed transmission lines, with either frequency-independent or frequency-dependent **RLGC** parameters. In this section, simplifications to the DEPACT macromodeling algorithm are presented for this case of frequency-independent parameters. One of the important advantages of the new approach is that MRA-based approximations are no longer required in the DEPACT macromodel. The final DEPACT cells are realized as exact implementations (without any approximations) in terms of lossless transmission lines and lumped elements, which are computed analytically based on the p.u.l. parameters. This aspect of the new macromodel leads to a relatively easier and more accurate implementation in SPICE-like circuit simulators.

For lines with frequency-independent parameters, the exponential stamp of the line can be approximated as

$$e^{\mathbf{A}+s\mathbf{B}} \approx \prod_{k=1}^m \mathbf{Q}_k + \varepsilon_m; \quad \mathbf{Q}_k \equiv e^{\frac{s\mathbf{B}}{2m}} e^{\frac{\mathbf{A}}{m}} e^{\frac{s\mathbf{B}}{2m}} \quad (5.1)$$

In this case, the DEPACT macromodel can be realized using a cascade of lossless lines and purely resistive networks. As mentioned in Chapter 2, the lossless lines can be macromodeled using the MoC and a decoupling algorithm. This decoupling algorithm separates the modes using modal transformation matrices, which are computed as a function of the p.u.l. \mathbf{L} and \mathbf{C} line parameters. The resulting lossless circuit consists of dependent voltage and current sources (representing the coupling between modes)

and single delay lines. Details of this approach are provided in [9]. The rest of this section will focus on the circuit realization of the resistive networks.

5.1.1 Realization of Single Resistive Sections

The Telegrapher's equations for a single resistive section of the DEPACT cell (represented by $e^{\frac{A}{m}}$ in (5.1)) can be written in the frequency-domain as

$$\begin{aligned}\frac{\partial}{\partial x}V(x, s) &= -R_m I(x, s) \\ \frac{\partial}{\partial x}I(x, s) &= -G_m V(x, s)\end{aligned}\quad (5.2)$$

where $R_m = \frac{R}{m}$, $G_m = \frac{G}{m}$, and m is the DEPACT order of approximation.

An analytical solution, in terms of y -parameters for (5.2) can be derived as [3]

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \frac{1}{Z_m(1 - e^{-2\Gamma_m d})} \begin{bmatrix} 1 + e^{-2\Gamma_m d} & -2e^{-\Gamma_m d} \\ -2e^{-\Gamma_m d} & 1 + e^{-2\Gamma_m d} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}\quad (5.3)$$

where I_1 , V_1 , I_2 , and V_2 represent the near- and far-end currents and voltages of the resistive section, respectively, and where

$$\Gamma_m = \sqrt{R_m G_m}; \quad Z_m = \sqrt{\frac{R_m}{G_m}}\quad (5.4)$$

Equation (5.3) can be re-written as

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} g_a + g_b & -g_b \\ -g_b & g_a + g_b \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \quad (5.5)$$

where the admittances g_a and g_b are given by [3]

$$g_a = \frac{(1 - e^{-\Gamma_m d})^2}{Z_m(1 - e^{-2\Gamma_m d})}; \quad g_b = \frac{(2e^{-\Gamma_m d})^2}{Z_m(1 - e^{-2\Gamma_m d})} \quad (5.6)$$

Note that since $\Gamma_m > 0$, this implies that $e^{-2\Gamma_m d} < 1$, which results in all resistor values being positive.

Next, using (5.5) and (5.6), the middle term $e^{\frac{A}{m}}$ in (5.1) can be realized in terms of a purely resistive network as shown in Figure 5.1, where $r_a = 1/g_a$, $r_b = 1/g_b$ and $d_m = d/m$.

It is important to note here that (5.3)-(5.6) represent an exact implementation of (5.2) without any numerical approximations. In addition, the macromodel is realized in terms of purely resistive elements, whose values are computed analytically in terms of the p.u.l. line parameters, R and G . This aspect of the proposed method greatly simplifies its implementation in SPICE-like simulators

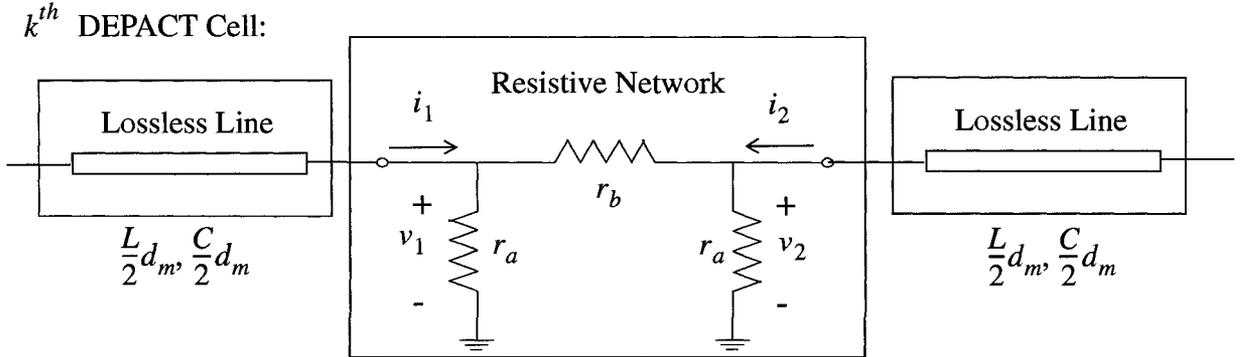


Figure 5.1: Macromodel realization for a single line with frequency-independent parameters

5.1.2 Realization of Coupled Resistive Sections

For coupled lossy sections, a decoupling method is employed in which the lossy lines are separated into single line subcircuits represented by the resistive network shown in Figure 5.1. The distributed voltages and currents of the coupled lossy section are related by the Telegrapher's equations as in (5.2), except $\mathbf{R}_m \in \mathfrak{R}^{N \times N}$, $\mathbf{G}_m \in \mathfrak{R}^{N \times N}$, $\mathbf{i}(x, t) \in \mathfrak{R}^{N \times 1}$, $\mathbf{v}(x, t) \in \mathfrak{R}^{N \times 1}$.

Next, the following lemma is introduced:

Lemma 3 *There exist transformation matrices \mathbf{P}_I and \mathbf{P}_V which simultaneously diagonalize both \mathbf{R}_m and \mathbf{G}_m as*

$$\begin{aligned} \mathbf{P}_V^{-1} \mathbf{R}_m \mathbf{P}_I &= \hat{\mathbf{R}} = \text{diag}\{\hat{r}_1, \hat{r}_2, \dots, \hat{r}_N\} \\ \mathbf{P}_I^{-1} \mathbf{G}_m \mathbf{P}_V &= \hat{\mathbf{G}} = \text{diag}\{\hat{g}_1, \hat{g}_2, \dots, \hat{g}_N\} \end{aligned} \quad (5.7)$$

The resulting circuit consists of decoupled resistive networks as shown in Figure 5.2 with auxiliary voltages $\hat{v}(x, t)$ and currents $\hat{i}(x, t)$ which are related by

$$\begin{aligned}\frac{\partial}{\partial x}\hat{v}(x, t) &= -\hat{\mathbf{R}}\hat{\mathbf{i}}(x, t) \\ \frac{\partial}{\partial x}\hat{\mathbf{i}}(x, t) &= -\hat{\mathbf{G}}\hat{v}(x, t)\end{aligned}\quad (5.8)$$

PROOF:

Following the approach used in [9] for lossless lines, we can decouple the purely resistive network in a similar manner.

The voltages and currents of the resistive subnetwork can be expressed in terms of auxiliary voltages $\hat{v}(x, t)$ and currents $\hat{i}(x, t)$ as [9]

$$\mathbf{v}(x, t) = \mathbf{P}_V\hat{v}(x, t); \quad \mathbf{i}(x, t) = \mathbf{P}_I\hat{\mathbf{i}}(x, t) \quad (5.9)$$

Since \mathbf{R} and \mathbf{G} are real, symmetric and positive-definite matrices, we can compute \mathbf{P}_V and \mathbf{P}_I as follows [9]

$$\begin{aligned}\mathbf{P}_I &= \mathbf{E}_m\mathbf{F}_m\mathbf{S}_m\boldsymbol{\alpha} \\ \mathbf{P}_V &= \mathbf{E}_m\mathbf{F}_m^{-1}\mathbf{S}_m\boldsymbol{\alpha}^{-1} = (\mathbf{P}_I^T)^{-1}\end{aligned}\quad (5.10)$$

where

- \mathbf{E}_m is a matrix of the right eigenvectors of \mathbf{G}_m , and \mathbf{F}_m^2 is a diagonal matrix of the eigenvalues of \mathbf{G}_m , i.e. $\mathbf{E}_m^T\mathbf{G}_m\mathbf{E}_m = \mathbf{F}_m^2$.

- The matrix \mathbf{S}_m consists of the right eigenvectors of the product $\mathbf{F}_m \mathbf{E}_m^T \mathbf{R}_m \mathbf{E}_m \mathbf{F}_m$, i.e. $\mathbf{S}_m^T (\mathbf{F}_m \mathbf{E}_m^T \mathbf{R}_m \mathbf{E}_m \mathbf{F}_m) \mathbf{S}_m = \mathbf{\Lambda}_m^2$.
- The diagonal matrix $\boldsymbol{\alpha}$ is a normalizing factor such that the Euclidean norm of the columns of \mathbf{P}_I is unity.

Substituting (5.10) into (5.7) results in

$$\begin{aligned}\hat{\mathbf{R}} &= \boldsymbol{\alpha} \mathbf{\Lambda}_m^2 \boldsymbol{\alpha} \\ \hat{\mathbf{G}} &= \boldsymbol{\alpha}^{-2}\end{aligned}\tag{5.11}$$

Since $\hat{\mathbf{R}}$ and $\hat{\mathbf{G}}$ are diagonal matrices, (5.8) represents a set of decoupled equations. This implies that the solutions for the auxiliary voltages and currents have the same general form as those for the single line case. Figure 5.2 depicts the circuit realization for the lossy coupled sections, where the resistor values $r_{a_i} = 1/g_{a_i}$ and $r_{b_i} = 1/g_{b_i}$ for the i th resistive network are computed as

$$g_{a_i} = \frac{(1 - e^{-\Gamma_{m_i} d})^2}{Z_{m_i} (1 - e^{-2\Gamma_{m_i} d})}; \quad g_{b_i} = \frac{(2e^{-\Gamma_{m_i} d})^2}{Z_{m_i} (1 - e^{-2\Gamma_{m_i} d})}\tag{5.12}$$

and where

$$\Gamma_{m_i} = \sqrt{\hat{r}_i \hat{g}_i}; \quad Z_{m_i} = \sqrt{\frac{\hat{r}_i}{\hat{g}_i}}\tag{5.13}$$

□

Note that the coupling effects between the actual voltages and currents ($\mathbf{v}(x, t)$, $\mathbf{i}(x, t)$) and the auxiliary voltages and currents ($\hat{\mathbf{v}}(x, t)$, $\hat{\mathbf{i}}(x, t)$) via \mathbf{P}_V and \mathbf{P}_I can

be realized using dependent voltage and current sources in conventional SPICE-like circuit simulators [9]. The overall DEPACT cell is similar to the one shown in Figure 5.1, except the middle resistive network is replaced with the coupled resistive network shown in Figure 5.2, and the single lossless lines are replaced with coupled lossless line circuits.

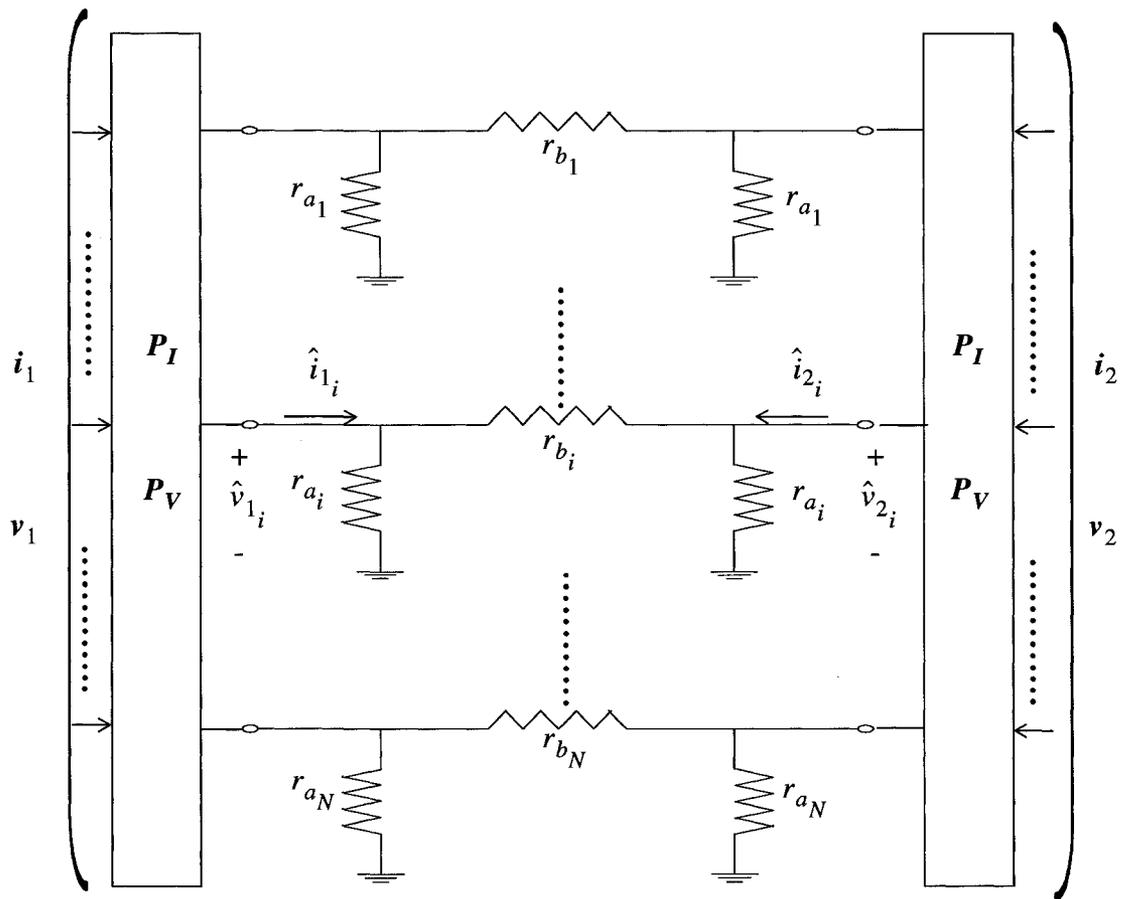


Figure 5.2: Equivalent circuit representation for the analysis of resistive coupled sections

5.1.3 Alternative Realization of Lossy Coupled Sections

In the previous section, an equivalent circuit was derived which decouples the coupled lossy lines into single resistive networks using transformation matrices. An alternate approach is presented in this section which does not require any decoupling algorithm. The resulting macromodel is a resistive network derived from the exponential stamp of the lossy coupled section.

The exponential stamp of the lossy section of the DEPACT cell can be written in terms of hybrid parameters as [3]

$$\begin{bmatrix} \mathbf{v}_2 \\ -\mathbf{i}_2 \end{bmatrix} = e^{\frac{\mathbf{A}}{m}} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{i}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{T}_{21} & \mathbf{T}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{i}_1 \end{bmatrix} \quad (5.14)$$

As described in [3], (5.14) can be expressed in y -parameter form as

$$\mathbf{I} = \bar{\mathbf{Y}}\mathbf{V}; \quad \bar{\mathbf{Y}} = \{y_{j,k}; j, k = 1, 2, \dots, 2N\} = \begin{bmatrix} \bar{\mathbf{Y}}_{11} & \bar{\mathbf{Y}}_{12} \\ \bar{\mathbf{Y}}_{21} & \bar{\mathbf{Y}}_{22} \end{bmatrix} \quad (5.15)$$

where

$$\begin{bmatrix} \bar{\mathbf{Y}}_{11} & \bar{\mathbf{Y}}_{12} \\ \bar{\mathbf{Y}}_{21} & \bar{\mathbf{Y}}_{22} \end{bmatrix} = \begin{bmatrix} -\mathbf{T}_{12}^{-1}\mathbf{T}_{11} & \mathbf{T}_{12}^{-1} \\ \mathbf{T}_{21} - \mathbf{T}_{22}\mathbf{T}_{12}^{-1}\mathbf{T}_{11} & \mathbf{T}_{22}\mathbf{T}_{12}^{-1} \end{bmatrix} \quad (5.16)$$

Using the Modified Nodal Analysis (MNA [3]), an equivalent circuit realization for (5.15) can be derived in terms of purely resistive elements. For the purpose of

illustration, Figure 5.3 shows the equivalent circuit for two coupled lines ($N = 2$). The algorithm in Figure 5.4 summarizes the steps for general circuit realization for N lines. It is important to note that, in general, the off-diagonal elements ($y_{j,k}$) of the y -parameter matrix $\bar{\mathbf{Y}}$ are negative. This results in a resistive network with all positive resistor values.

Using this approach, the overall DEPACT cell consists of coupled lossless line circuits surrounding a resistive network with the same structure as that shown in Figure 5.3. In addition, it can be shown that the realization described in Section 5.1.1 and Figure 5.1 is just a special case of the algorithm presented here when $N = 1$.

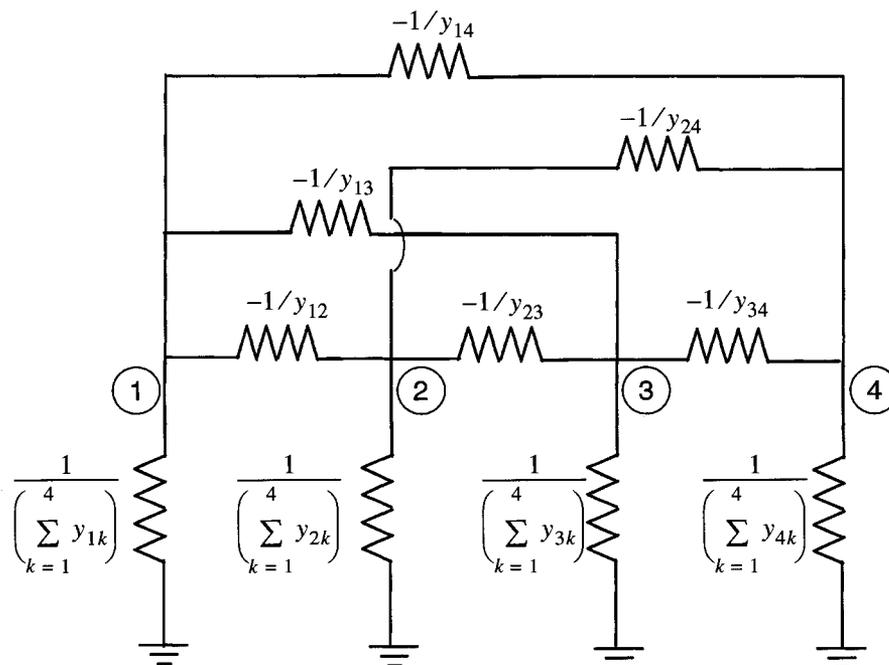


Figure 5.3: Equivalent circuit representation of (5.15) for $N = 2$

To represent diagonal elements: y_{jj} ($j=1,2,\dots,2N$)

Between node j and ground

$$r_{jj} = \frac{1}{\left(\sum_{k=1}^{2N} y_{jk} \right)}$$

To represent off-diagonal elements: $y_{jk} = y_{kj}$ ($j=1,2,\dots,2N; k=1,2,\dots, 2N$)

Between nodes j and k

$$r_{jk} = \frac{-1}{y_{jk}} \quad j \neq k$$

Figure 5.4: General algorithm for circuit realization of lossy sections described by (5.15)

5.2 Macromodels for Lines with Frequency-Dependent $R(s)$ and $L(s)$ parameters

As frequencies advance, the resistance R and inductance L become increasingly frequency-dependent. It has been shown in the literature that neglecting these frequency-dependent effects can lead to errors in the transient simulation results [9, 43–45]. For several practical cases, such as on-chip interconnects [55], setting $G = 0$ or G and C as constant matrices does not significantly affect the time-domain solutions [9]. Simplification of the DEPACT macromodeling algorithm for such special cases is described in this section.

5.2.1 Case 1: Constant \mathbf{C} , and $\mathbf{G} = 0$

For frequency-dependent $\mathbf{R}(s)$, $\mathbf{L}(s)$, constant \mathbf{C} , and $\mathbf{G} = 0$, the exponential stamp representing the overall transmission line can be written as

$$e^{\mathbf{A}(s)+s\mathbf{B}(s)} \cong \prod_{k=1}^m \mathbf{Q}_k; \quad \mathbf{Q}_k = e^{\frac{s\mathbf{B}_\infty}{2m}} e^{\mathbf{A}(s)+s\mathbf{B}_a(s)} e^{\frac{s\mathbf{B}_\infty}{2m}} \quad (5.17)$$

where

$$\mathbf{A}(s) = - \begin{bmatrix} \mathbf{0} & \mathbf{R}(s) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} d_m; \quad \mathbf{B}_a(s) = - \begin{bmatrix} \mathbf{0} & \mathbf{L}_a(s) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} d_m \quad (5.18)$$

and

$$\mathbf{L}_a(s) = \mathbf{L}(s) - \mathbf{L}(\infty); \quad d_m = d/m \quad (5.19)$$

Expanding $e^{\mathbf{A}(s)+s\mathbf{B}(s)}$ in terms of the exponential Taylor series, we get

$$e^{\mathbf{A}(s)+s\mathbf{B}(s)} = \mathbf{I} + (\mathbf{A}(s) + s\mathbf{B}(s)) + \frac{(\mathbf{A}(s) + s\mathbf{B}(s))^2}{2!} + \dots \quad (5.20)$$

where

$$\mathbf{A}(s) + s\mathbf{B}(s) = \begin{bmatrix} \mathbf{0} & -(\mathbf{R}(s) + s\mathbf{L}_a(s))d_m \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (5.21)$$

$$(\mathbf{A}(s) + s\mathbf{B}(s))^p = \mathbf{0}, \quad p \geq 2 \quad (5.22)$$

and \mathbf{I} is the $N \times N$ identity matrix.

Using (5.20)-(5.22), we can write

$$e^{A(s)+sB(s)} = \begin{bmatrix} \mathbf{I} & -(\mathbf{R}(s) + sL_a(s))d_m \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (5.23)$$

For the case of single lines, the terminal voltages and currents of the DEFACT lossy section are related by

$$\begin{bmatrix} V_2 \\ -I_2 \end{bmatrix} = \begin{bmatrix} 1 & -(R(s) + sL_a(s))d_m \\ 0 & \mathbf{1} \end{bmatrix} \begin{bmatrix} V_1 \\ I_1 \end{bmatrix} \quad (5.24)$$

Expanding (5.24) yields the following

$$\begin{aligned} V_2 &= V_1 - [d_m(R(s) - sL_a(s))I_1] \\ I_1 &= -I_2 \end{aligned} \quad (5.25)$$

Figure 5.5 shows the equivalent circuit representation for the equations in (5.25), where the middle section consists of a frequency-dependent resistor and inductor in series, represented by

$$\tilde{Z}(s) = (R(s) + sL_a(s))d_m \quad (5.26)$$

The realization of the FD impedance $\tilde{Z}(s)$ can be achieved using the modeling approach described in Section 4.2.

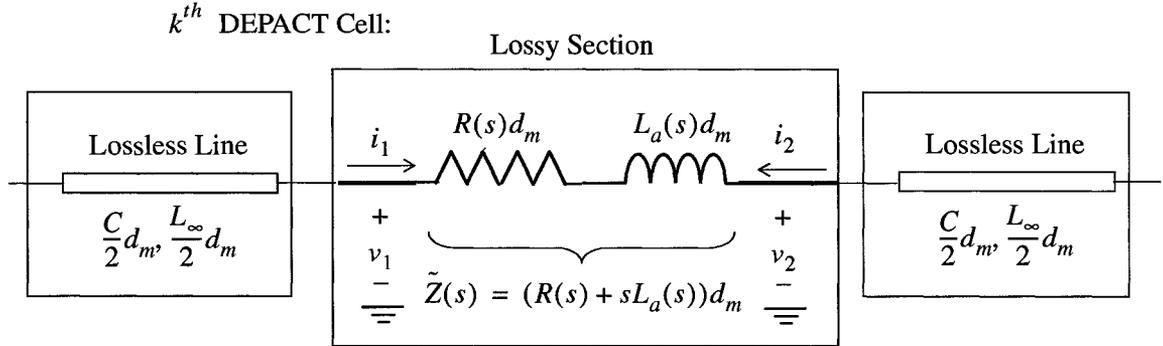


Figure 5.5: Macromodel realization of the k^{th} DEPACT cell for a single line with frequency-dependent $R(s)$, $L(s)$, constant C and $G = 0$

Similarly, for coupled lines, (5.23) can be represented in terms of mutually coupled inductances and resistors, as shown in Figure 5.6, where

$$\tilde{\mathbf{R}}(s) = \mathbf{R}(s)d_m = \{\tilde{R}_{j,k}(s); j, k = 1, 2, \dots, N\} \quad (5.27)$$

$$\tilde{\mathbf{L}}(s) = \mathbf{L}_a(s)d_m = \{\tilde{L}_{j,k}(s); j, k = 1, 2, \dots, N\} \quad (5.28)$$

In this case, the DEPACT cell will consist of a coupled circuit representing $\tilde{\mathbf{Z}}(s)$ (which can be modeled as described in Section 4.2 and in Figure 4.3), with lossless coupled lines along the outer portion of the cell.

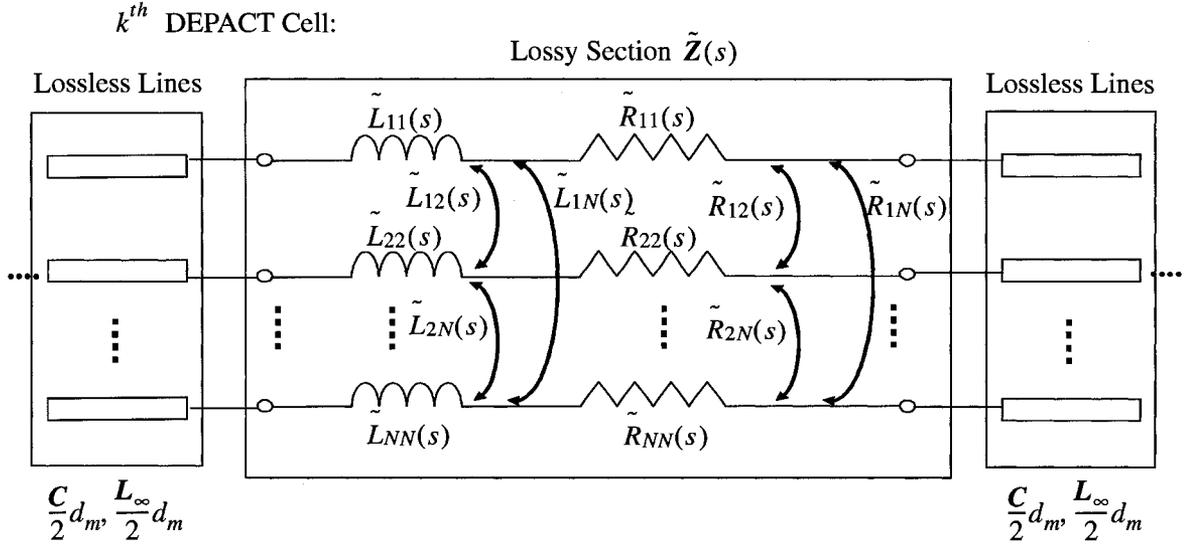


Figure 5.6: DEFACT cell with macromodel for coupled lossy sections described by (5.23)

5.2.2 Case 2: Constant C and G

In this section, we present an equivalent macromodel when the resistance $\mathbf{R}(s)$ and inductance $\mathbf{L}(s)$ are frequency-dependent, and the capacitance C and conductance G are constant.

The delay-extracted exponential matrix of the middle section of the DEFACT cell is given by

$$e^{\mathbf{Z}(s)} = e^{\mathbf{A}(s) + s\mathbf{B}_a(s)} \quad (5.29)$$

where for this case,

$$\mathbf{A}(s) = - \begin{bmatrix} \mathbf{0} & \mathbf{R}(s) \\ \mathbf{G} & \mathbf{0} \end{bmatrix} d_m; \quad \mathbf{B}_a(s) = - \begin{bmatrix} \mathbf{0} & \mathbf{L}_a(s) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} d_m \quad (5.30)$$

Equations (5.29) and (5.30) can be re-written as

$$e^{\mathbf{Z}(s)} = e^{\mathbf{E}(s)+\mathbf{D}} \quad (5.31)$$

where

$$\mathbf{E}(s) = \begin{bmatrix} \mathbf{0} & -(\mathbf{R}(s) + s\mathbf{L}_a(s)) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} d_m; \quad \mathbf{D} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{G} & \mathbf{0} \end{bmatrix} d_m \quad (5.32)$$

Using the Modified Lie Formula [15], the exponential matrix $e^{\mathbf{Z}(s)}$ in (5.31) can be approximated using a product of p terms as follows

$$e^{\mathbf{Z}(s)} = e^{\mathbf{E}(s)+\mathbf{D}} \cong \prod_{i=1}^p \Phi_i \quad (5.33)$$

where Φ_i is given by

$$\Phi_i = e^{\frac{\mathbf{D}}{2p}} e^{\frac{\mathbf{E}(s)}{p}} e^{\frac{\mathbf{D}}{2p}} \quad (5.34)$$

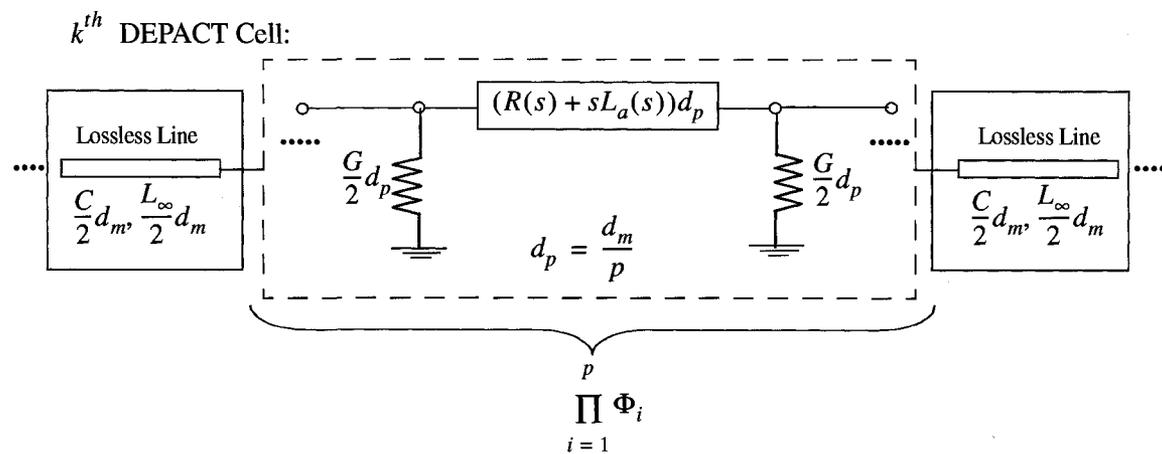
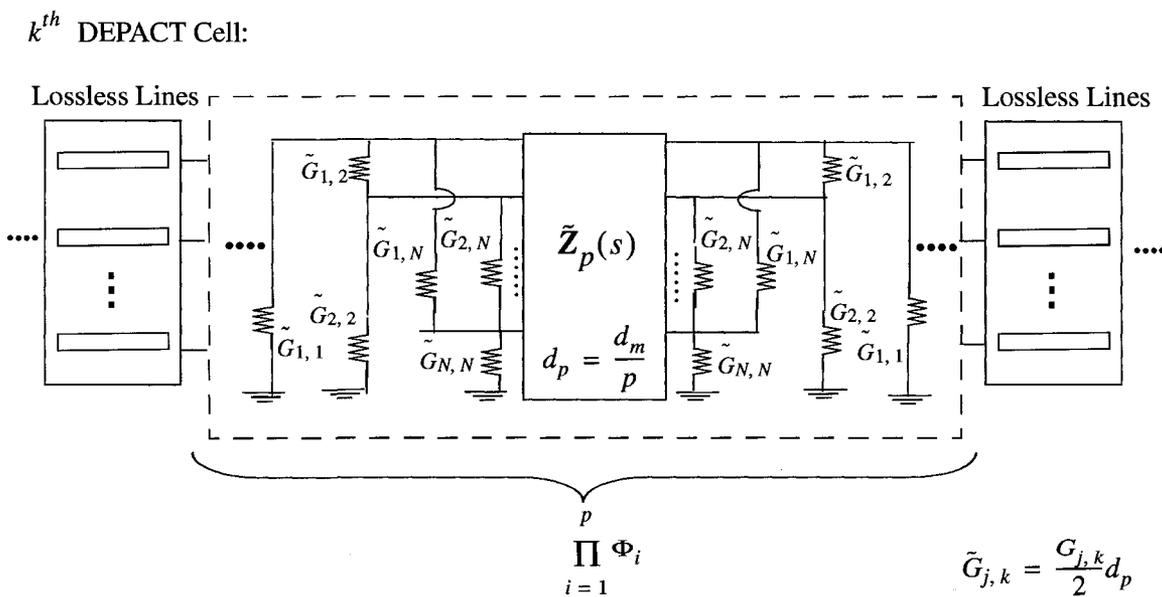
Using the approximation in (5.33) and (5.34), the exponential stamp representing the *overall* transmission line can be expressed in a compact form as

$$e^{\mathbf{A}(s)+s\mathbf{B}(s)} \cong \prod_{k=1}^m e^{\frac{s\mathbf{B}_\infty}{2m}} \left(\prod_{i=1}^p \Phi_i \right) e^{\frac{s\mathbf{B}_\infty}{2m}} \quad (5.35)$$

Equivalent Macromodel Realization

For the macromodel realization, the middle term $e^{\frac{\mathbf{E}(s)}{p}}$ of Φ_i can be represented by the frequency-dependent impedance as shown in Figures 5.5 and 5.6, with a length of $d_p = d_m/p$. The outer terms $e^{\frac{\mathbf{D}}{2p}}$ of Φ_i represent purely resistive networks, and can be realized using the approach described in Sections 5.1.1- 5.1.3. In this case, the resistive networks will be located at the terminals of each of the p circuit realizations for the FD impedance.

For the overall macromodel, (5.35) implies that each DEFACT cell can be viewed as a cascade of p lossy sub-cells (represented by $\prod_{i=1}^p \Phi_i$) with lossless lines on the outer portions. In addition, each sub-cell (Φ_i) can be realized as a cascade of frequency-dependent impedances terminated by resistors. Figures 5.7 and 5.8 show the circuit realization of the resulting DEFACT cells for single and coupled lines, respectively. In Figure 5.8, $\tilde{\mathbf{Z}}_p(s)$ represents the FD impedance of the i th lossy sub-cell, given by $\tilde{\mathbf{Z}}_p(s) = (\mathbf{R}(s) + s\mathbf{L}_a(s))d_p$.

Figure 5.7: Circuit realization for the DEFACT cell with constant G Figure 5.8: Circuit realization for the coupled DEFACT cell with constant G

5.3 Numerical Results

This section presents several numerical examples which validate the accuracy and efficiency of the proposed simplified delay extraction-based macromodels described in this chapter. Six examples are considered. Examples 1 and 2 deal with transmission lines with frequency-independent parameters. The special case of lines with FD $\mathbf{R}(s)$, $\mathbf{L}(s)$, constant \mathbf{C} , and $\mathbf{G} = 0$ is studied in Examples 3 and 4. Examples 5 and 6 consider the second special case when we have FD $\mathbf{R}(s)$, $\mathbf{L}(s)$, and constant \mathbf{C} and \mathbf{G} .

5.3.1 Example 1

In this example, we consider a long 10 m cable line with frequency-independent parameters. This example is based on one of three benchmark problems provided in [1]. The circuit and line parameters are shown in Figure 5.9. The input voltage is a step input with rise time of 0.1ns. Figure 5.10 shows the transient response at the far-end of the transmission line using the simplified delay-extraction macromodel with $m = 15$. Note that since we are dealing with frequency-independent parameters, the lossy sections are modeled using purely resistive networks. The results are compared with uniform lumped segmentation in HSPICE [4]. As seen, the results from both methods are in good agreement. However, the CPU time for the proposed model was only 1.12 s, while lumped segmentation required 1272 s, resulting in a speed-up of 1100.

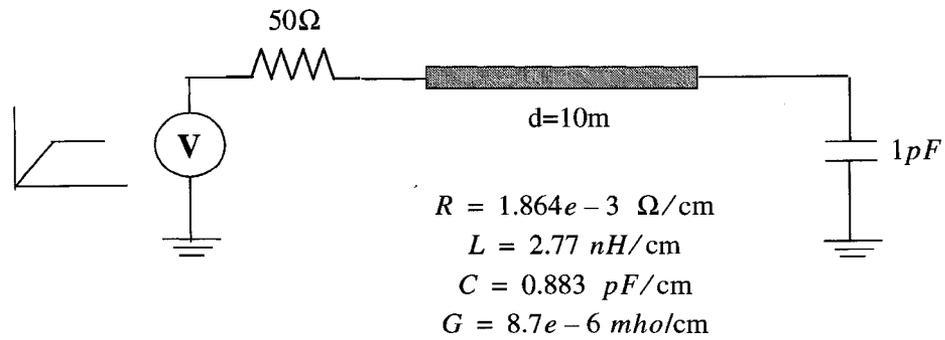


Figure 5.9: Long (10m) cable transmission line circuit for Example 1

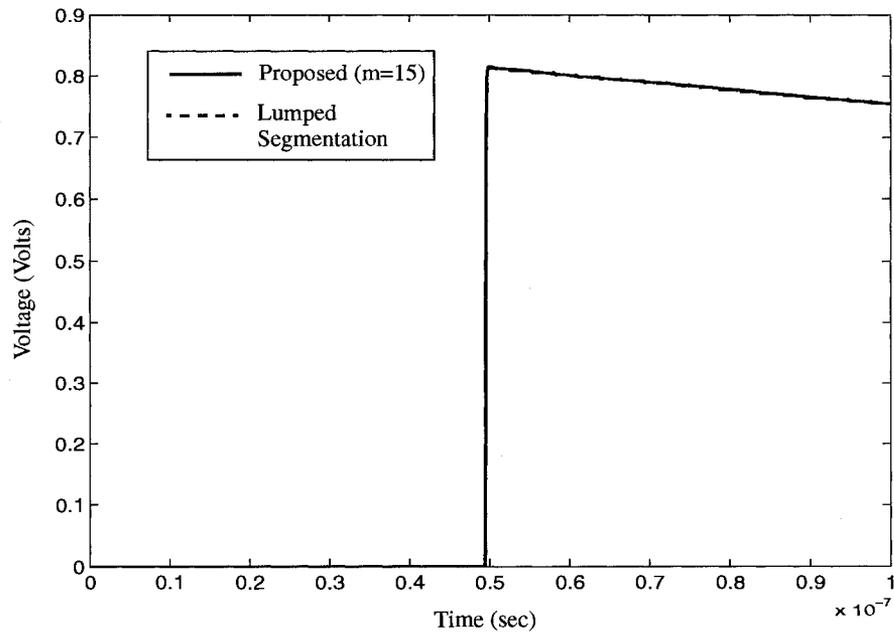


Figure 5.10: Transient response at far-end of cable line for Example 1

5.3.2 Example 2

In this example, we consider a nonlinear multiconductor transmission line circuit as shown in Figure 5.11. The line parameters for MTL subnetwork #1 are given by [5]

$$\begin{aligned}
 \mathbf{R} &= \begin{bmatrix} 0.252 & 0 \\ 0 & 0.252 \end{bmatrix} \Omega/cm; & \mathbf{L} &= \begin{bmatrix} 3.36 & 0.865 \\ 0.865 & 3.36 \end{bmatrix} nH/cm \\
 \mathbf{C} &= \begin{bmatrix} 1.29 & -0.197 \\ -0.197 & 1.29 \end{bmatrix} pF/cm & \mathbf{G} &= 0 \\
 & & d &= 5cm
 \end{aligned} \tag{5.36}$$

and the parameters for MTL subnetworks #2 and #3 are

$$\begin{aligned}
 \mathbf{R} &= \begin{bmatrix} 0.34 & 0 \\ 0 & 0.34 \end{bmatrix} \Omega/cm; & \mathbf{L} &= \begin{bmatrix} 4.76 & 0.974 \\ 0.974 & 4.76 \end{bmatrix} nH/cm \\
 \mathbf{C} &= \begin{bmatrix} 1.02 & -0.178 \\ -0.178 & 1.02 \end{bmatrix} pF/cm & \mathbf{G} &= 0 \\
 & & d &= 4cm
 \end{aligned} \tag{5.37}$$

The input voltage is a unit step function with a rise time of 0.1ns. The circuit was simulated using the simplified delay extraction macromodel for frequency-independent parameters, for each MTL subnetwork. Figures 5.12 and 5.13 show a sample of the transient responses obtained using an approximation order of $m = 6$. As seen from

the graphs, the results of the proposed method are in very good agreement with those from uniform lumped segmentation. The CPU time for lumped segmentation was 23.4 s, while the proposed method only required 0.45 s, thus providing a speed-up of 52.

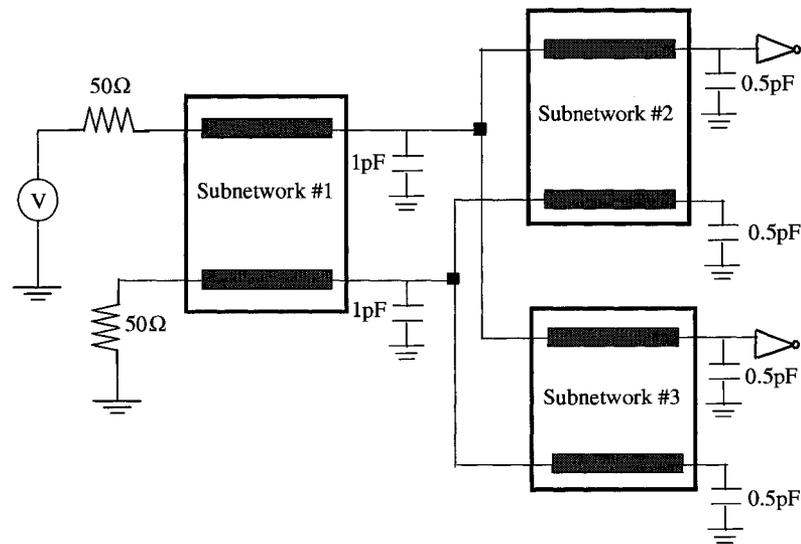


Figure 5.11: Multiconductor transmission line network with nonlinear terminations considered in Example 2

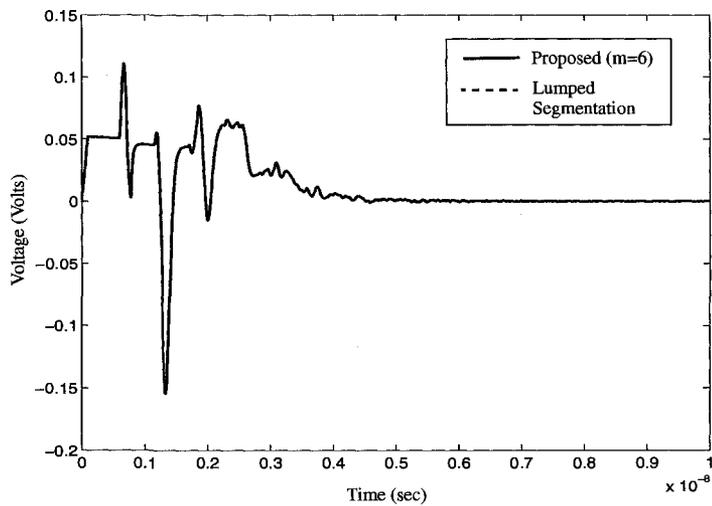


Figure 5.12: Transient response at near-end of victim line of MTL subnetwork #1

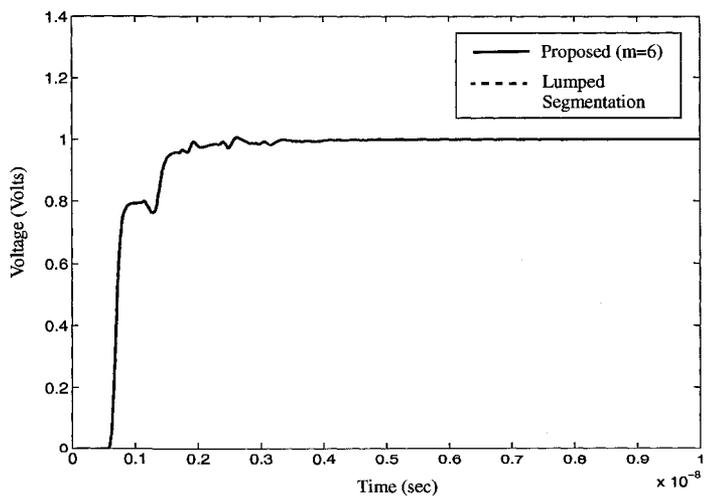


Figure 5.13: Transient response at far-end of active line of MTL subnetwork #2

5.3.3 Example 3

In this example, a single PCB (Printed Circuit Board) line with frequency-dependent $R(s)$ and $L(s)$, constant C , and $G = 0$ is considered. The FD line parameters were fitted using the modeling approach described in Section 4.2. Figures 5.14 - 5.16 show the fitting results for the p.u.l. $R(s)$, $L(s)$ and the impedance $Z(s) = R(s) + sL(s)$, respectively. As seen from the figures, the fitting results match very closely to the original parameters. For simulation, the single line was terminated with 50Ω resistors, and the input to the circuit was a trapezoidal pulse with rise/fall times of 0.04ns. The length of the line is $d = 0.0508$ m, (2 inches). The line was macromodeled using the algorithm presented in Section 5.2.1. Figures 5.17 and 5.18 show the transient responses for the near and far-end of the single PCB line using the proposed method with $m = 11$, compared to the IFFT (Inverse Fast Fourier Transform) of the exact response using the original data. As seen from the plots, the results are in excellent agreement. Using the proposed method, the CPU time for approximating the line parameters was 0.032 s (performed only once), while the CPU cost for simulation was 0.45 s. In contrast to this, the circuit was also simulated using the equivalent lumped segmentation model obtained from the FD parameter fitting algorithm. The resulting CPU time was 7.05 s, which is more than 15 times slower than the proposed method.

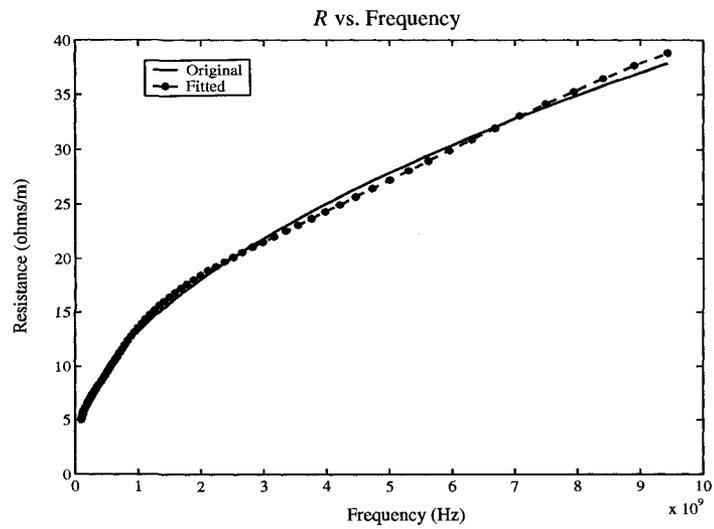


Figure 5.14: Fitted p.u.l. resistance $R(s)$ as a function of frequency (Example 3)

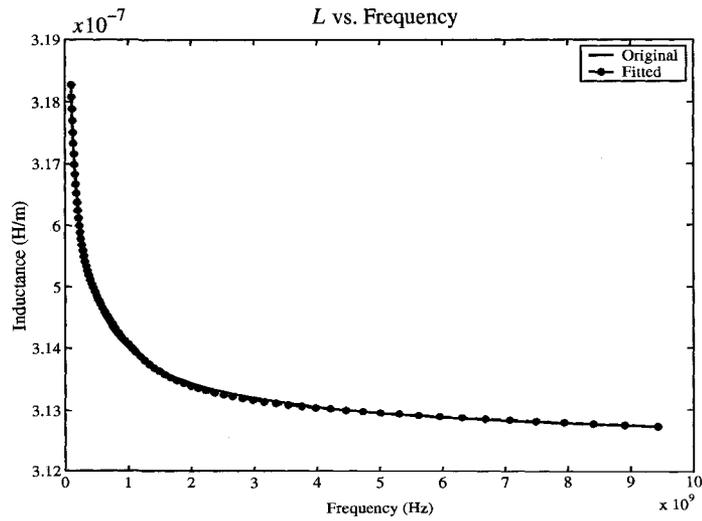


Figure 5.15: Fitted p.u.l. inductance $L(s)$ as a function of frequency (Example 3)

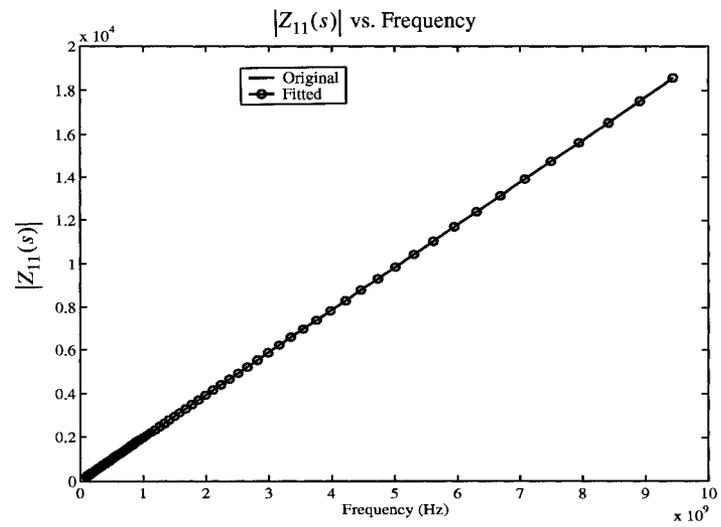


Figure 5.16: Fitting results for $Z(s)$ for Example 3

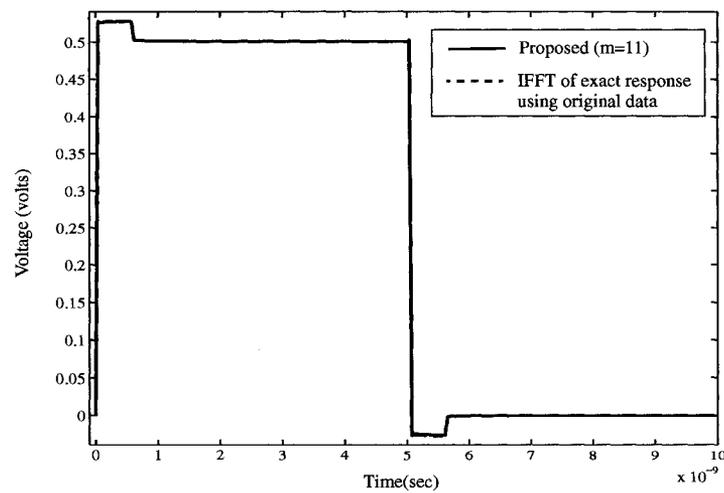


Figure 5.17: Transient response at near-end of active line for Example 3

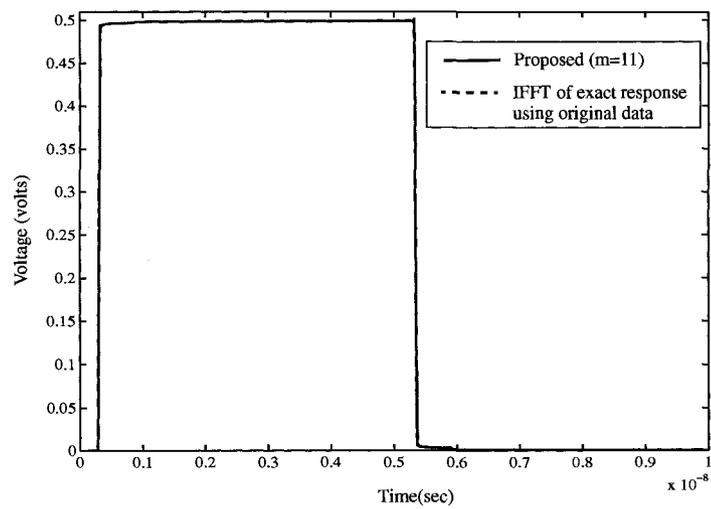


Figure 5.18: Transient response at far-end of active line for Example 3

5.3.4 Example 4

In this example, we consider a longer PCB ($d = 0.254$ m) with two coupled lines. Similar to the previous example, we also have frequency-dependent $\mathbf{R}(s)$ and $\mathbf{L}(s)$, constant \mathbf{C} , and $\mathbf{G} = 0$. Figure 5.19 shows a sample of the fitting results for $Z_{1,1}(s) = R_{1,1}(s) + sL_{1,1}(s)$. As seen from the figure, the fitting results match very closely to the original $Z_{1,1}(s)$. For simulation, the coupled lines were terminated with 50Ω resistors and a trapezoidal pulse with rise/fall times of 0.2ns was used as the input to the circuit. The lines were macromodeled using the coupled approach described in Section 5.2.1 and Figure 5.6. Figure 5.20 shows a sample of the transient responses obtained using the proposed method ($m=25$) vs. the IFFT of the exact response using the original data. As seen, the results from the new delay extraction macromodeling algorithm match very closely to the exact response. In addition, the equivalent lumped segmentation model was generated using the FD parameter fitting algorithm. The CPU time using the lumped circuit was 45 s, while the CPU time for the proposed method was only 1.05 s, thus providing a speed-up of 43.

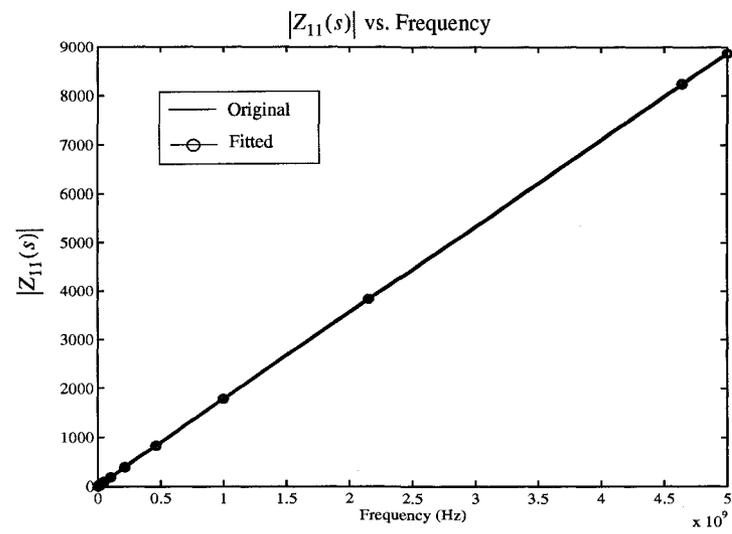
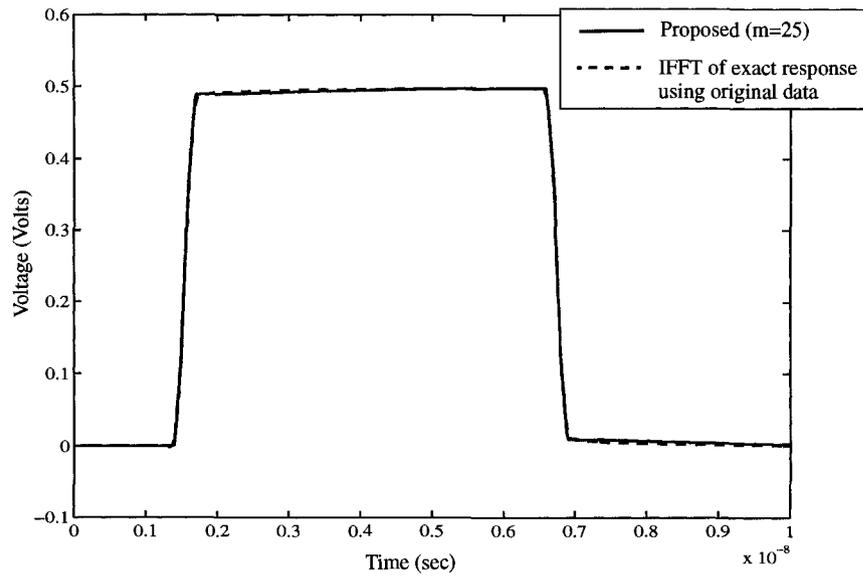
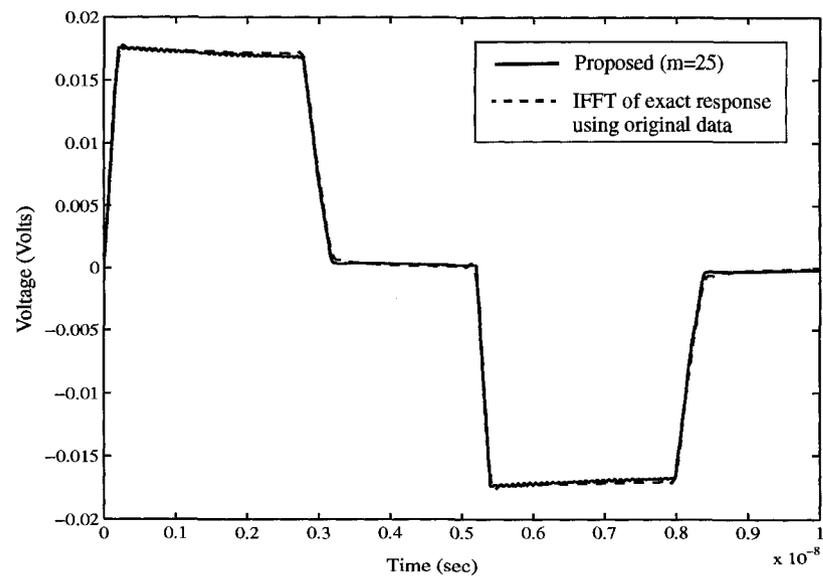


Figure 5.19: Fitting results for $Z_{1,1}(s)$ for Example 4



(a) Voltage at far-end of active line



(b) Voltage at near-end of victim line

Figure 5.20: Output transient waveforms for Example 4

5.3.5 Example 5

In this example, we consider a similar PCB line to that used in Example 3. The p.u.l. $R(s)$ and $L(s)$ are frequency-dependent, C is constant, but in this case, we have a constant conductance value of 0.2888 U/m . The same approximations for the line parameters used in Example 3 were used in this example (since $R(s)$ and $L(s)$ did not change). However, the resulting macromodel is of the form shown in Figure 5.7, where the lossy middle section is realized using a cascade of p lossy sub-cells and terminating resistors. In this case, the length of the line is $d = 0.0808 \text{ m}$, $p = 10$ and $m = 8$. The line was terminated with $50 \text{ } \Omega$ resistors and the input to the circuit was a trapezoidal pulse with rise/fall times of 0.1 ns . Figures 5.21 and 5.22 show the transient responses at the near and far-end of the line, respectively, compared to the IFFT of the exact response using the original data. As seen from the plots, the results from both methods are in very close agreement. Using the proposed method, the CPU time for simulation was only 0.6 s , while the lumped segmentation model required 6.7 s .

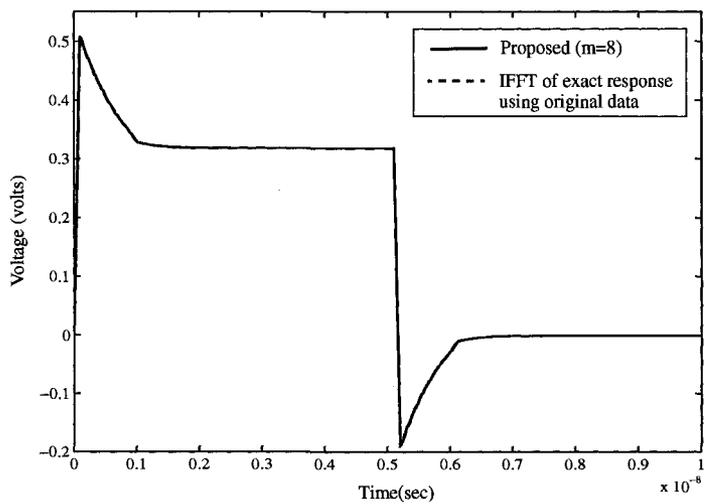


Figure 5.21: Transient response at near-end of active line (Example 5)

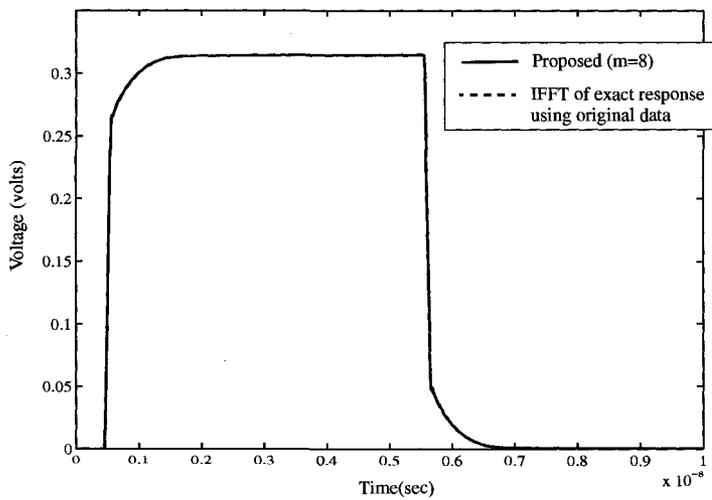
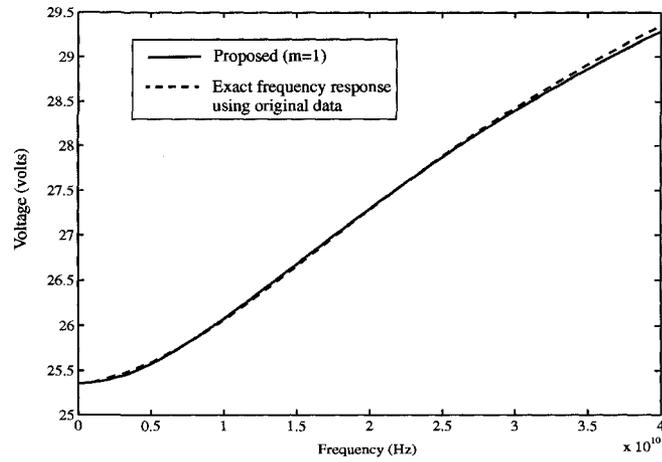


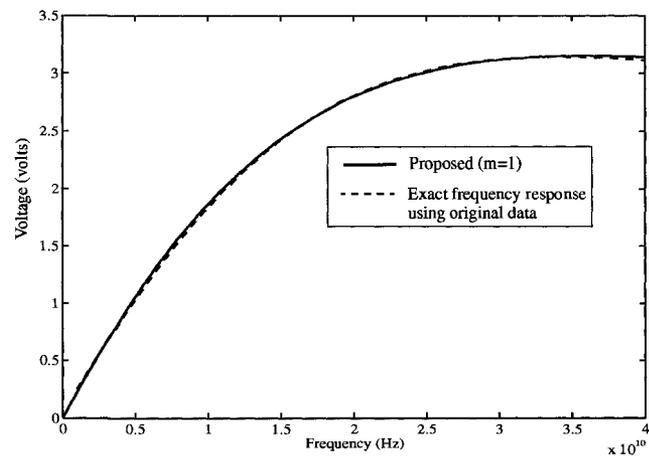
Figure 5.22: Transient response at far-end of active line (Example 5)

5.3.6 Example 6

In this example, we consider an on-chip five-coupled-line circuit with frequency-dependent $\mathbf{R}(s)$ and $\mathbf{L}(s)$, constant \mathbf{C} and constant conductance \mathbf{G} . The lines were macromodeled using the coupled approach described in Section 5.2.2 and Figure 5.8. The length of the line is $d = 200\mu\text{m}$. The coupled lines were terminated by $50\ \Omega$ resistors. Figures 5.23 and 5.24 show a sample of the frequency-domain responses obtained using the proposed method for $m = 1$ and $p = 1$, vs. the responses obtained from the exponential stamp of the line (exact solution using original data). As seen from the plots, the results match very closely, which confirms that the approximation orders $m = 1$ and $p = 1$ are sufficient. A trapezoidal pulse with rise/fall times of 0.08 ns was used as input to the circuit. A sample of the transient responses are presented in Figures 5.25 and 5.26 using the proposed method vs. the IFFT of the exact response using the original data. As seen from the plots, the results from both methods are in very good agreement. In addition, the CPU time for fitting the coupled line parameters was 0.08 s (performed only once), and the transient simulation time was only 2.10 s. In contrast, the simulation using the equivalent lumped segmentation model required 26 s.

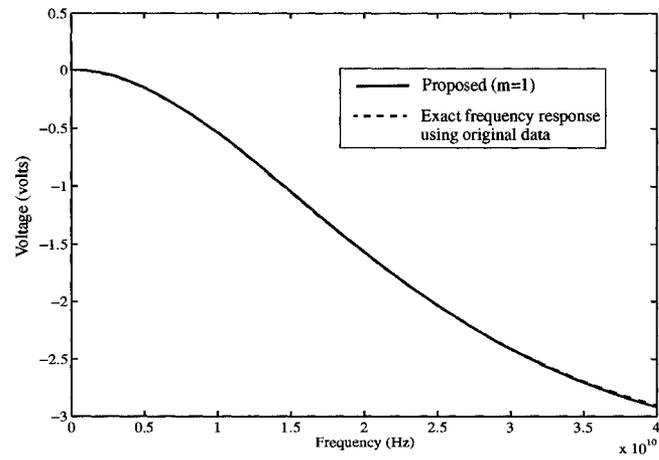


(a) Real part of voltage at active line near-end

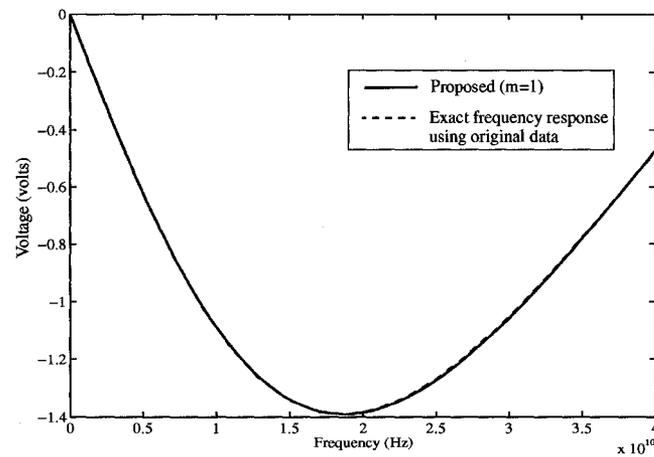


(b) Imaginary part of voltage at active line near-end

Figure 5.23: Real and imaginary parts of the voltage at the near-end of active line



(a) Real part of voltage at victim line far-end



(b) Imaginary part of voltage at victim line far-end

Figure 5.24: Real and imaginary parts of the voltage at the far-end of victim line

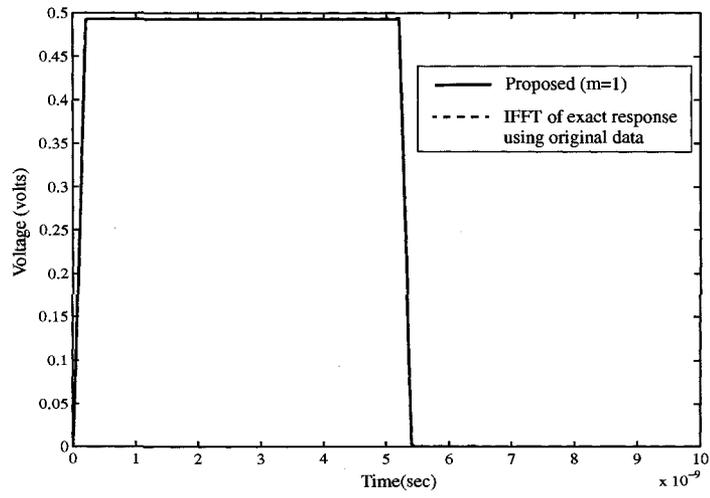


Figure 5.25: Transient response at far-end of active line (Example 6)

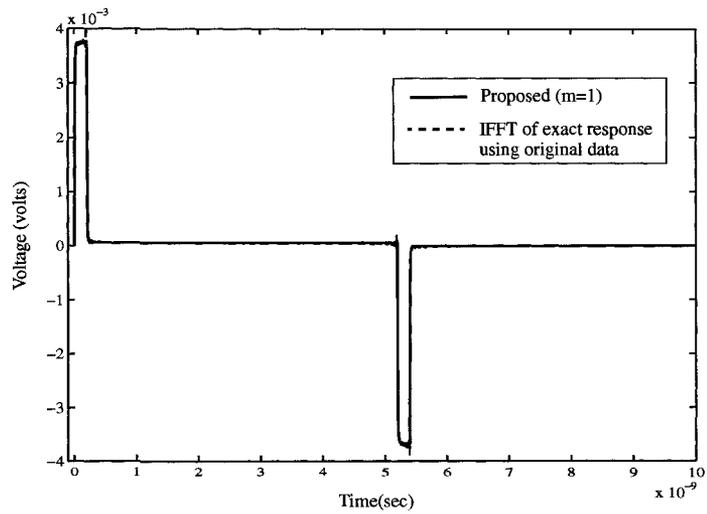


Figure 5.26: Transient response at near-end of victim line (Example 6)

5.4 Summary

In summary, this chapter presented a new simplified delay extraction-based macromodeling algorithm for several special cases. One of the main advantages of the new method is that MRA-based approximations are no longer required for the special cases discussed in this chapter. Specifically, for lines with frequency-independent parameters, the final DEPACT cells are realized as exact implementations (without any approximations) in terms of lossless lines and lumped elements, which are computed analytically based on the p.u.l. parameters. New macromodeling algorithms for the special cases of lines with frequency-dependent $\mathbf{R}(s)$, $\mathbf{L}(s)$, constant \mathbf{C} , with either $\mathbf{G} = 0$, or constant \mathbf{G} were also described. With these methods, an efficient modeling technique is used for rational function approximations of the FD line parameters and the resulting equivalent circuits form the lossy section of the DEPACT cell. Several numerical examples were provided to validate the accuracy and efficiency of the proposed method. In addition, the algorithms described in this chapter are relevant for the new delay extraction-based waveform relaxation with transverse partitioning method which will be described in the next chapter.

Chapter 6

A Delay Extraction-Based Algorithm for Simulation of Coupled Interconnects using Transverse Partitioning

In this chapter, a new algorithm for simulation of large coupled interconnects based on delay extraction and transverse partitioning is presented. Section 6.1 introduces the new method and Section 6.2 describes transverse partitioning of the DEFACT cells. Sections 6.3 and 6.4 provide formulations for partitioning of the lossy and lossless cells, respectively, and Section 6.5 describes the overall resulting macromodel. Numerical examples are provided in Section 6.6.

6.1 Introduction

In chapters 3 and 4, new waveform relaxation and transverse partitioning (WR-TP) algorithms for simulation of coupled interconnects were presented. Using these algorithms, the N coupled interconnect circuit is partitioned into N or less independent subcircuits with relaxation sources. It was demonstrated that, using these approaches, the CPU cost increases almost linearly with the number of lines, providing a significant speed-up as compared to conventional methods. Moreover, one of the main

advantages of these algorithms is their generality with respect to the specific macro-model used for simulation of the single line subcircuits. In Chapter 3, two different approaches for updating the WR sources were described: single-ended and distributed representations. In the single-ended approach, the WR sources are located at the end of the transmission lines, but their calculation requires the evaluation of a convolution integral and several transformations between frequency and time. On the other hand, using the distributed approach, the convolution integral is discretized using a suitable integration formula [39]. In this case, the WR sources are distributed throughout the length of the lines, and have a direct time-domain representation (thus avoiding expensive FFT/IFFT operations). Specifically, formulations for distributed approaches using uniform lumped segmentation models were presented, along with implementation details regarding updating the sources. However, in the lumped segmentation approach, the number of lumped segments depends on the length of the line and the maximum frequency of interest [3]. Consequently, the usefulness of the distributed approach using lumped models is limited to relatively short lines. To address this, in this chapter, the WR-TP algorithm is extended to include more sophisticated transmission line macromodels, specifically delay extraction-based models (DEPACT) [5] (described in the previous chapter). In this case, the algorithm is suitable for both long and short lines and the WR sources are computed directly in the time-domain. In addition, fewer WR sources are required since discretization of the convolution integral is not involved. Moreover, it is important to note that the proposed method is highly parallelizable (just like the other WR-TP algorithms described in Chapters 3 and 4), since the coupled line circuit is partitioned into *independent* subcircuits. Thus, array processors can be used to simulate the individual subcircuits in parallel, which would provide further computational cost advantages.

6.2 Transverse Partitioning of DEPACT Cells

DEPACT is a robust and compact algorithm, and since it employs an efficient delay-extraction mechanism, it is suitable for both long and short lines (off-chip and on-chip applications). In addition, the resulting macromodel is guaranteed to be passive [5]. However, as the number of lines is increased, we are still faced with the challenging problem of simulating a large number of coupled lines. To address this issue, the WR-TP concept is extended to include delay extraction-based macromodels.

One alternative is to use the single-ended representation for the WR-TP algorithm described in equations (3.7)-(3.13). In this case, the decoupled transmission lines can be realized using the DEPACT macromodel (cascade of lossy and lossless lines as shown in Figure 2.9), with independent WR sources located at the end of each line [17]. However, using this approach would require the evaluation of the convolution integral in (3.13) and numerous frequency/time transformations of the voltages and currents along the lines. This can lead to an increase in computational cost. Thus, to avoid this, the solution is to apply the WR-TP concept on the DEPACT cells themselves, as opposed to the Telegrapher's equations. Using this approach, each of the coupled DEPACT lossy and lossless transmission line subnetworks are partitioned independently in the transverse direction, and WR is applied on the lossy and lossless lines separately. It will be shown that, in this case, the WR sources have a direct time-domain representation and are computed analytically in a closed-form.

6.3 Partitioning of Lossy Sections

For the case of transmission lines with frequency-independent *RLGC* parameters, the stamp of the lossy section of the DEPACT cell can be written as

$$\begin{bmatrix} \mathbf{v}(l_m, t) \\ -\mathbf{i}(l_m, t) \end{bmatrix} = e^{\frac{\mathbf{A}}{2m}} \begin{bmatrix} \mathbf{v}(0, t) \\ \mathbf{i}(0, t) \end{bmatrix} \quad (6.1)$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & -\mathbf{R} \\ -\mathbf{G} & \mathbf{0} \end{bmatrix} d \quad (6.2)$$

and where \mathbf{R} and \mathbf{G} are the p.u.l. resistance and conductance parameter matrices, respectively and $l_m = d/2m$ is the length of the lossy section.

With simple manipulation, (6.2) can be written in the following form

$$\Theta = \tilde{\mathbf{Y}} \Omega \quad (6.3)$$

$$\Theta \equiv \begin{bmatrix} \Theta_1 \\ \vdots \\ \Theta_N \end{bmatrix}; \quad \Omega \equiv \begin{bmatrix} \Omega_1 \\ \vdots \\ \Omega_N \end{bmatrix} \quad (6.4)$$

$$\tilde{\mathbf{Y}} \equiv \{\tilde{\mathbf{Y}}_{jk}; \quad j, k = 1, \dots, N\} \quad (6.5)$$

where the matrix $\tilde{\mathbf{Y}}$ denotes the re-arranged y -parameter matrix, and $\tilde{\mathbf{Y}}_{jk}$ are 2×2 submatrices representing the coupling between lines j and k . The vectors $\Theta_j(t)$ and $\Omega_j(t)$ represent the terminal currents and voltages for the j th line of the lossy section, respectively, and are defined as

$$\begin{aligned}\Theta_j(t) &= [i_j(l_0, t) \quad i_j(l_m, t)]^T \\ \Omega_j(t) &= [v_j(l_0, t) \quad v_j(l_m, t)]^T\end{aligned}\tag{6.6}$$

where l_0 and l_m represent the near and far ends of the lossy section, respectively, and superscript ‘ T ’ denotes the transpose operator.

Example

To illustrate the formulations in (6.3)-(6.6), consider two coupled lossy lines as shown in Figure 6.1.

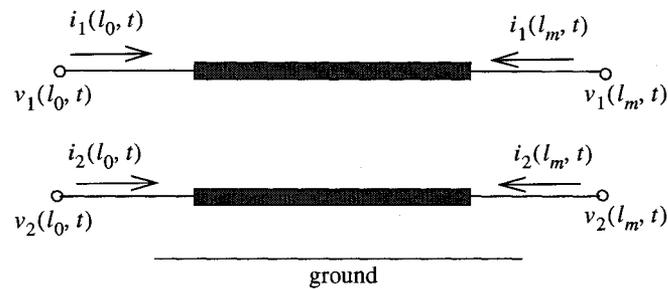


Figure 6.1: Two coupled lossy lines

The exponential stamp of the lines can be written as

$$\begin{bmatrix} v_1(l_m, t) \\ v_2(l_m, t) \\ -i_1(l_m, t) \\ -i_2(l_m, t) \end{bmatrix} = e^{\frac{A}{2m}} \begin{bmatrix} v_1(l_0, t) \\ v_2(l_0, t) \\ i_1(l_0, t) \\ i_2(l_0, t) \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{T}_{21} & \mathbf{T}_{22} \end{bmatrix} \begin{bmatrix} v_1(l_0, t) \\ v_2(l_0, t) \\ i_1(l_0, t) \\ i_2(l_0, t) \end{bmatrix} \quad (6.7)$$

Using the approach in [3], the terminal relations of the two coupled lines can be written in y -parameter form as

$$\begin{bmatrix} i_1(l_0, t) \\ i_2(l_0, t) \\ i_1(l_m, t) \\ i_2(l_m, t) \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{Y}}_{11} & \bar{\mathbf{Y}}_{12} \\ \bar{\mathbf{Y}}_{21} & \bar{\mathbf{Y}}_{22} \end{bmatrix} \begin{bmatrix} v_1(l_0, t) \\ v_2(l_0, t) \\ v_1(l_m, t) \\ v_2(l_m, t) \end{bmatrix} \quad (6.8)$$

where

$$\begin{bmatrix} \bar{\mathbf{Y}}_{11} & \bar{\mathbf{Y}}_{12} \\ \bar{\mathbf{Y}}_{21} & \bar{\mathbf{Y}}_{22} \end{bmatrix} = \begin{bmatrix} -\mathbf{T}_{12}^{-1}\mathbf{T}_{11} & \mathbf{T}_{12}^{-1} \\ \mathbf{T}_{21} - \mathbf{T}_{22}\mathbf{T}_{12}^{-1}\mathbf{T}_{11} & \mathbf{T}_{22}\mathbf{T}_{12}^{-1} \end{bmatrix} \quad (6.9)$$

Next, in order to group together the y -parameters for each individual line, the matrix $\bar{\mathbf{Y}}$ is re-arranged by swapping rows/columns 2 and 3 in (6.8). With simple mathematical manipulation, we can write

$$\begin{bmatrix} i_1(l_0, t) \\ i_1(l_m, t) \\ i_2(l_0, t) \\ i_2(l_m, t) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{Y}}_{11} & \tilde{\mathbf{Y}}_{12} \\ \tilde{\mathbf{Y}}_{21} & \tilde{\mathbf{Y}}_{22} \end{bmatrix} \begin{bmatrix} v_1(l_0, t) \\ v_1(l_m, t) \\ v_2(l_0, t) \\ v_2(l_m, t) \end{bmatrix} \quad (6.10)$$

where the 2×2 submatrices $\tilde{\mathbf{Y}}_{11}$ and $\tilde{\mathbf{Y}}_{22}$ represent the admittance parameters for lines 1 and 2, respectively, and where $\tilde{\mathbf{Y}}_{12}$ and $\tilde{\mathbf{Y}}_{21}$ represent the coupling between the lines. \square

Note that since \mathbf{A} is frequency-independent, the y -parameter matrix $\tilde{\mathbf{Y}}$ in (6.10) is real, and the exponential stamp in (6.2) can be realized using a purely resistive network, with all positive resistor values (further details are provided in Section 5.1.2).

Next, applying waveform relaxation techniques to (6.3), we obtain the following recursive equations

$$\Theta_j^{(r+1)} = \tilde{\mathbf{Y}}_{jj} \Omega_j^{(r+1)} + \varphi_j^{(r)}(t); \quad j = 1, 2, \dots, N \quad (6.11)$$

where $\varphi_j^{(r)}(t)$ represents the coupling effects on line j due to the neighboring lines and is given by

$$\varphi_j^{(r)}(t) = \begin{bmatrix} \varphi_j(l_0, t) \\ \varphi_j(l_m, t) \end{bmatrix}^{(r)} = \sum_{\substack{k=1 \\ k \neq j}}^N \tilde{\mathbf{Y}}_{jk} \Omega_k^{(r)} \quad (6.12)$$

The WR source $\varphi_j^{(r)}(t)$ in (6.12) is represented by independent current sources at the

terminals of each line within the lossy section. The advantage here is that the sources are computed directly in the time-domain as a linear combination of the voltages on the neighboring lines. Figure 6.2 shows the equivalent circuit for the j th line of the partitioned lossy sections, where the resistor values are obtained using (5.6).

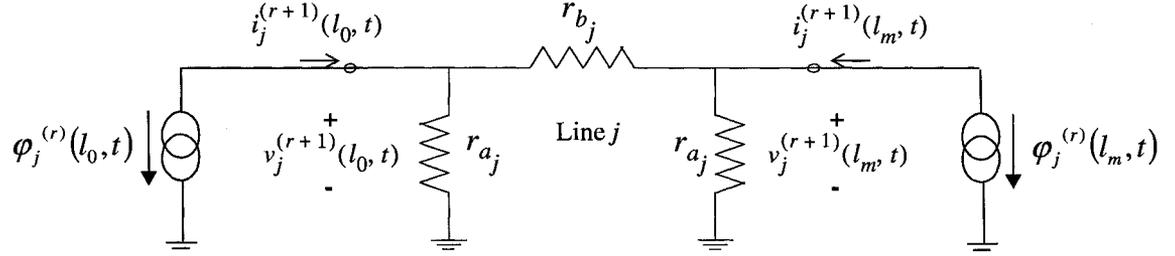


Figure 6.2: Equivalent circuit for the j th line of the decoupled lossy section

6.4 Partitioning of Lossless Sections

6.4.1 Conventional Mode Decoupling

One common approach used for decoupling of lossless lines is accomplished via mode decoupling [9]. In this approach, the actual voltages and currents $(\mathbf{v}(x, t), \mathbf{i}(x, t))$ of the lossless lines are expressed in terms of propagating *modal* voltages and currents $(\hat{\mathbf{v}}(x, t), \hat{\mathbf{i}}(x, t))$ as

$$\begin{aligned}\mathbf{v}(x, t) &= \mathbf{T}_V \hat{\mathbf{v}}(x, t) \\ \mathbf{i}(x, t) &= \mathbf{T}_I \hat{\mathbf{i}}(x, t)\end{aligned}\tag{6.13}$$

where \mathbf{T}_V and \mathbf{T}_I are modal transformation matrices, which are obtained as a function of the eigenvalues and eigenvectors of the \mathbf{L} and \mathbf{C} line parameter matrices [9].

Using (6.13), the decoupled terminal voltages and currents of the lossless section are related by

$$\begin{aligned}\frac{\partial}{\partial x}\hat{\mathbf{v}}(x,t) &= -\hat{\mathbf{L}}\frac{\partial}{\partial t}\hat{\mathbf{i}}(x,t) \\ \frac{\partial}{\partial x}\hat{\mathbf{i}}(x,t) &= -\hat{\mathbf{C}}\frac{\partial}{\partial t}\hat{\mathbf{v}}(x,t)\end{aligned}\quad (6.14)$$

where $\hat{\mathbf{L}}$ and $\hat{\mathbf{C}}$ are diagonal matrices defined as

$$\begin{aligned}\hat{\mathbf{L}} &= \text{diag}\{\hat{l}_1, \dots, \hat{l}_N\} = \mathbf{T}_V^{-1}\mathbf{L}_m\mathbf{T}_I \\ \hat{\mathbf{C}} &= \text{diag}\{\hat{c}_1, \dots, \hat{c}_N\} = \mathbf{T}_I^{-1}\mathbf{C}_m\mathbf{T}_V\end{aligned}\quad (6.15)$$

and $\mathbf{L}_m = \frac{\mathbf{L}}{m}$ and $\mathbf{C}_m = \frac{\mathbf{C}}{m}$.

Using equations (6.13)-(6.15), the N -coupled lossless line network can be decoupled into N single lossless lines with parameters l_j and c_j for $j = 1, \dots, N$. The coupled relationship between the actual voltages and currents ($\mathbf{v}(x,t), \mathbf{i}(x,t)$), and the modal voltages and currents ($\hat{\mathbf{v}}(x,t), \hat{\mathbf{i}}(x,t)$) is realized using dependent sources at the terminals of each lossless line [9]. Although this approach is commonly used for simulation and analysis, if waveform relaxation was applied to this formulation, it is not guaranteed to converge. This is due to the fact that the modal transformation matrices \mathbf{T}_V and \mathbf{T}_I are *not diagonally dominant*. Thus, the coupling between the actual voltages/currents and the modal voltages/currents is relatively strong, and

is not the ideal scenario to apply waveform relaxation. To address this, we must decouple the *actual* voltages/currents between the lines, since those are relatively weakly coupled.

6.4.2 Closed-Form Evaluation of WR Sources for Lossless Lines

In this new algorithm, the actual voltages and currents between neighboring lossless lines are decoupled and the waveform relaxation sources are computed analytically in a closed-form. Using this approach, the coupling sources are located at the far-end of the lossless sections and are obtained as delayed time-domain waveforms of the voltages and currents on the neighboring lines.

Consider again the integral form of the WR sources for the j th line which is given by

$$\delta_j^{(r)}(d, s) = \begin{bmatrix} V_{src_j}(d, s) \\ I_{src_j}(d, s) \end{bmatrix}^{(r)} = \int_0^d e^{\mathbf{F}_j(d-\eta)} \Psi_j^{(r)}(\eta, s) d\eta \quad (6.16)$$

Since we are applying WR-TP to the *lossless* section of the DEFACT cell, (6.16) can be simplified as

$$\begin{bmatrix} V_{src_j}(d_m, s) \\ I_{src_j}(d_m, s) \end{bmatrix}^{(r)} = \int_0^{d_m} e^{\tilde{\mathbf{F}}_j(d_m-\eta)} \tilde{\Psi}_j^{(r)}(\eta, s) d\eta \quad (6.17)$$

where

$$\tilde{\mathbf{F}}_j = \begin{bmatrix} 0 & -sL_{jj} \\ -sC_{jj} & 0 \end{bmatrix} \quad (6.18)$$

$$\tilde{\Psi}_j^{(r)}(\eta, s) = \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & -sL_{jk} \\ -sC_{jk} & 0 \end{bmatrix} \begin{bmatrix} V_k(\eta, s) \\ I_k(\eta, s) \end{bmatrix}^{(r)} \quad (6.19)$$

and $d_m = d/m$ is the length of the lossless section.

In (6.19), the voltages and currents of the k th line can be expressed as

$$\begin{bmatrix} V_k(\eta, s) \\ I_k(\eta, s) \end{bmatrix} = \mathbf{U}_k \mathbf{W}_\eta(s) \quad (6.20)$$

where $\mathbf{U}_k \in [0, 1]$ is a selector matrix which selects the voltages and currents on the k th line and $\mathbf{W}_\eta(s)$ is a $2N \times 1$ vector of the voltages and currents on all N lines at a distance η , which is given by

$$\mathbf{W}_\eta(s) = \begin{bmatrix} V_1(\eta, s) \\ V_2(\eta, s) \\ \vdots \\ V_N(\eta, s) \\ I_1(\eta, s) \\ I_2(\eta, s) \\ \vdots \\ I_N(\eta, s) \end{bmatrix} \quad (6.21)$$

Next, using the exponential stamp of the overall line, $\mathbf{W}_\eta(s)$ can be expressed in terms of the voltages and currents at the near end of the lossless line as follows [3]

$$\mathbf{W}_\eta(s) = e^{s\tilde{\mathbf{B}}\eta} \mathbf{W}_0(s) \quad (6.22)$$

where

$$\mathbf{W}_0(s) = \begin{bmatrix} V_1(d_0, s) \\ V_2(d_0, s) \\ \vdots \\ V_N(d_0, s) \\ I_1(d_0, s) \\ I_2(d_0, s) \\ \vdots \\ I_N(d_0, s) \end{bmatrix}; \quad \tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{0} & -\mathbf{L} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \quad (6.23)$$

and d_0 denotes the near end of the lossless line.

Using (6.20)-(6.23), and with some mathematical manipulation, the lossless WR sources in (6.17) can be re-written as

$$\begin{aligned} \text{Srcs}_j^{(r)}(s) &= \begin{bmatrix} V_{src_j}(d_m, s) \\ I_{src_j}(d_m, s) \end{bmatrix}^{(r)} \\ &= -s \int_0^{d_m} e^{\tilde{\mathbf{F}}_j(d_m-\eta)} \hat{\mathbf{B}}_j e^{s\tilde{\mathbf{B}}\eta} \mathbf{W}_0(s)^{(r)} d\eta \end{aligned} \quad (6.24)$$

where $\hat{\mathbf{B}}_j$ represents the coupling parameters for line j and is given by

$$\hat{\mathbf{B}}_j = \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & L_{jk} \\ C_{jk} & 0 \end{bmatrix} \mathbf{U}_k \quad (6.25)$$

Next, the term $e^{\tilde{\mathbf{F}}_j(d_m-\eta)}$ in (6.24) can be diagonalized as

$$e^{\tilde{\mathbf{F}}_j(d_m-\eta)} = \beta_j \begin{bmatrix} e^{s\tau_j(d_m-\eta)} & 0 \\ 0 & e^{-s\tau_j(d_m-\eta)} \end{bmatrix} \beta_j^{-1} \quad (6.26)$$

where β_j are the eigenvectors of $\tilde{\mathbf{F}}_j/s$ and $\tau_j = \sqrt{L_{jj}C_{jj}}$.

Similarly, $e^{s\tilde{\mathbf{B}}\eta}$ can be expressed as follows

$$e^{s\tilde{\mathbf{B}}\eta} = \tilde{\Gamma} \text{diag}\{e^{s\tilde{\gamma}_1\eta}, \dots, e^{s\tilde{\gamma}_N\eta}, e^{-s\tilde{\gamma}_1\eta}, \dots, e^{-s\tilde{\gamma}_N\eta}\} \tilde{\Gamma}^{-1} \quad (6.27)$$

where $\tilde{\gamma}_1, \dots, \tilde{\gamma}_N$ and $\tilde{\Gamma}$ are the eigenvalues and eigenvectors of the matrix $\tilde{\mathbf{B}}$ (in (6.23)), respectively.

Next, we define the the following $2 \times 2N$ matrix

$$\mathbf{K}_j = \beta_j^{-1} \hat{\mathbf{B}}_j \tilde{\Gamma} \quad (6.28)$$

Using (6.26)-(6.28), and with some mathematical manipulation, (6.24) can be simplified as

$$\text{Srcs}_j^{(r)}(s) = -s \int_0^{d_m} \beta_j \Lambda_j \tilde{\Gamma}^{-1} \mathbf{W}_0(s)^{(r)} d\eta \quad (6.29)$$

where Λ_j is a matrix of exponential terms defined as

$$\mathbf{\Lambda}_j = \begin{bmatrix} \mathbf{K}_j^{(1,1)} e^{s\alpha_j^{(1,1)}} & \dots & \mathbf{K}_j^{(1,2N)} e^{s\alpha_j^{(1,2N)}} \\ \mathbf{K}_j^{(2,1)} e^{-s\alpha_j^{(2,1)}} & \dots & \mathbf{K}_j^{(2,2N)} e^{-s\alpha_j^{(2,2N)}} \end{bmatrix} \quad (6.30)$$

where $\mathbf{K}_j^{(n,p)}$ denotes the element in location (n, p) of the \mathbf{K}_j matrix, and α_j is given by

$$\alpha_j^{(1,p)} = \begin{cases} \tau_j(d_m - \eta) + \tilde{\gamma}_p \eta, & p = 1, \dots, N \\ \tau_j(d_m - \eta) - \tilde{\gamma}_{(p-N)} \eta, & p = N + 1, \dots, 2N \end{cases} \quad (6.31)$$

$$\alpha_j^{(2,p)} = \begin{cases} \tau_j(d_m - \eta) - \tilde{\gamma}_p \eta, & p = 1, \dots, N \\ \tau_j(d_m - \eta) + \tilde{\gamma}_{(p-N)} \eta, & p = N + 1, \dots, 2N \end{cases} \quad (6.32)$$

and N is the number of lines.

Next, since $\boldsymbol{\beta}_j$, $\tilde{\Gamma}^{-1}$ and $\mathbf{W}_0(s)^{(r)}$ are all independent of η , the frequency-domain WR sources in (6.29) can be expressed in a compact form as follows

$$\text{Srcs}_j^{(r)}(s) = -\boldsymbol{\beta}_j \mathbf{H}_j \tilde{\Gamma}^{-1} \mathbf{W}_0(s)^{(r)} \quad (6.33)$$

where \mathbf{H}_j is a $2 \times 2N$ matrix given by

$$\mathbf{H}_j = s \int_0^{d_m} \mathbf{\Lambda}_j d\eta \quad (6.34)$$

whose elements are defined as

$$H_j^{(1,p)} = \frac{\mathbf{K}_j^{(1,p)}}{\tilde{\gamma}_p - \tau_j} \left[e^{s\tilde{\gamma}_p d_m} - e^{s\tau_j d_m} \right], \quad p = 1, \dots, N$$

$$H_j^{(1,p)} = \frac{\mathbf{K}_j^{(1,p)}}{-\tilde{\gamma}_{(p-N)} - \tau_j} \left[e^{-s\tilde{\gamma}_{(p-N)}} d_m - e^{s\tau_j} d_m \right], \quad p = N + 1, \dots, 2N \quad (6.35)$$

$$H_j^{(2,p)} = \frac{-\mathbf{K}_j^{(2,p)}}{-\tilde{\gamma}_p - \tau_j} \left[e^{s\tilde{\gamma}_p} d_m - e^{-s\tau_j} d_m \right], \quad p = 1, \dots, N$$

$$H_j^{(2,p)} = \frac{-\mathbf{K}_j^{(2,p)}}{\tilde{\gamma}_{(p-N)} - \tau_j} \left[e^{-s\tilde{\gamma}_{(p-N)}} d_m - e^{-s\tau_j} d_m \right], \quad p = N + 1, \dots, 2N \quad (6.36)$$

In order to link to nonlinear circuit simulators, it is required to obtain a time-domain representation for the sources given in (6.33). This is straightforward since the exponential terms in the \mathbf{H}_j matrix represent delay operations in the time-domain.

Next, we define the following relation

$$\hat{\Phi}(t)^{(r)} = \begin{bmatrix} \hat{\phi}_1(t) \\ \vdots \\ \hat{\phi}_{2N}(t) \end{bmatrix}^{(r)} = \tilde{\Gamma}^{-1} \mathbf{W}_0(t)^{(r)} \quad (6.37)$$

where

$$\mathbf{W}_0(t)^{(r)} = \begin{bmatrix} v_1(d_0, t) \\ \vdots \\ v_N(d_0, t) \\ i_1(d_0, t) \\ \vdots \\ i_N(d_0, t) \end{bmatrix}^{(r)} \quad (6.38)$$

represents the time-domain voltage and current waveforms at the near end of the lossless lines.

Lastly, using (6.34)-(6.37), the time-domain WR sources can be expressed in a closed-form as follows

$$\begin{aligned} \mathit{Srcs}_j^{(r)}(t) &= \begin{bmatrix} v_{src_j}(d_m, t) \\ i_{src_j}(d_m, t) \end{bmatrix}^{(r)} \\ &= -\boldsymbol{\beta}_j \begin{bmatrix} f_{j_1}(t) \\ f_{j_2}(t) \end{bmatrix}^{(r)} \end{aligned} \quad (6.39)$$

where $f_{j_1}(t)$ and $f_{j_2}(t)$ are defined as

$$\begin{aligned}
f_{j1}(t)^{(r)} &= \sum_{p=1}^N \frac{\mathbf{K}_j^{(1,p)}}{\tilde{\gamma}_p - \tau_j} \left[\hat{\phi}_p^{(r)}(t + \tilde{\gamma}_p d_m) u(t + \tilde{\gamma}_p d_m) - \hat{\phi}_p^{(r)}(t + \tau_j d_m) u(t + \tau_j d_m) \right] \\
&+ \sum_{p=N+1}^{2N} \frac{\mathbf{K}_j^{(1,p)}}{-\tilde{\gamma}_{(p-N)} - \tau_j} \left[\hat{\phi}_p^{(r)}(t - \tilde{\gamma}_{(p-N)} d_m) u(t - \tilde{\gamma}_{(p-N)} d_m) - \hat{\phi}_p^{(r)}(t + \tau_j d_m) u(t + \tau_j d_m) \right]
\end{aligned} \tag{6.40}$$

$$\begin{aligned}
f_{j2}(t)^{(r)} &= \sum_{p=1}^N \frac{-\mathbf{K}_j^{(2,p)}}{-\tilde{\gamma}_p - \tau_j} \left[\hat{\phi}_p^{(r)}(t + \tilde{\gamma}_p d_m) u(t + \tilde{\gamma}_p d_m) - \hat{\phi}_p^{(r)}(t - \tau_j d_m) u(t - \tau_j d_m) \right] \\
&+ \sum_{p=N+1}^{2N} \frac{-\mathbf{K}_j^{(2,p)}}{\tilde{\gamma}_{(p-N)} - \tau_j} \left[\hat{\phi}_p^{(r)}(t - \tilde{\gamma}_{(p-N)} d_m) u(t - \tilde{\gamma}_{(p-N)} d_m) - \hat{\phi}_p^{(r)}(t - \tau_j d_m) u(t - \tau_j d_m) \right]
\end{aligned} \tag{6.41}$$

and $u(t)$ is the unit step function given by

$$u(t - a) = \begin{cases} 0 & t < a \\ 1 & t \geq a \end{cases} \tag{6.42}$$

Using (6.39)-(6.42), the WR sources for the lossless sections of the DEFACT cell can be computed in a closed-form. This is achievable due to the fact that we are dealing with purely lossless lines, and the convolution integral in (6.16) may be evaluated analytically. As a result, the sources are obtained as a combination of delayed time-domain waveforms of the voltages and currents at the near end of the lines. The equivalent circuit for the j th decoupled lossless line is shown in Figure 6.3.

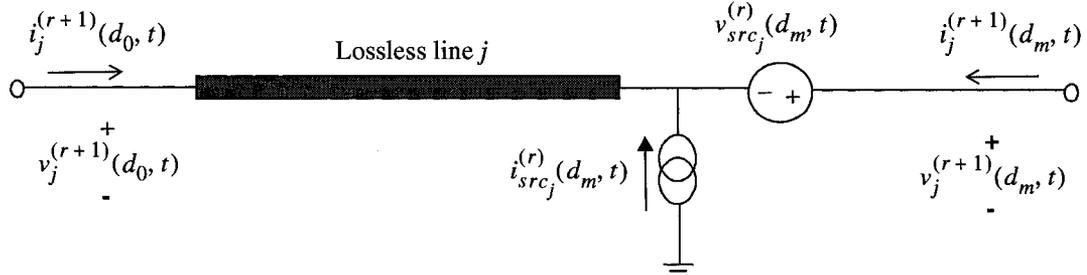


Figure 6.3: Equivalent circuit for the j th line of the decoupled lossless section

6.5 DEPACT+WR-TP Macromodel Realization

The resulting partitioned DEPACT cells are constructed as a cascade of the decoupled lossy and lossless sections. Figure 6.4 shows the equivalent circuit for the decoupled DEPACT cell with WR sources. The overall line is realized using a cascade of m DEPACT cells. Using this approach, each individual line is macromodeled using a cascade of single lossy and lossless lines with voltage and current WR sources. It is also important to note that, in the proposed method, there are fewer WR sources (compared to the distributed approach described in Chapter 3), since there is no discretization of the convolution integral.

The computational steps of the proposed algorithm can be summarized as follows:

- *Step 1: Using (2.28), select a suitable DEPACT approximation order m .*

Let $r=0$.

- *Step 2: Start with an initial waveform for the lossy and lossless WR sources, $\varphi_j^{(r)}(t)$ and $Srcs_j^{(r)}(t)$, respectively, for $1 \leq j \leq N$ and for all DEPACT cells*

$(1 \leq k \leq m)$.

- *Step 3: Simulate each individual line independently and obtain $\mathbf{\Omega}_j(t)^{(r+1)}$ (in (6.6)), and $\mathbf{W}_0(t)^{(r+1)}$ (in (6.38)).*
- *Step 4: Based on the solution obtained in Step 3, update the WR sources using (6.12) and (6.39).*
- *Step 5: Let $r=r+1$; go to Step 3 and repeat until convergence is achieved within an acceptable error tolerance.*

k^{th} DEFACT cell:

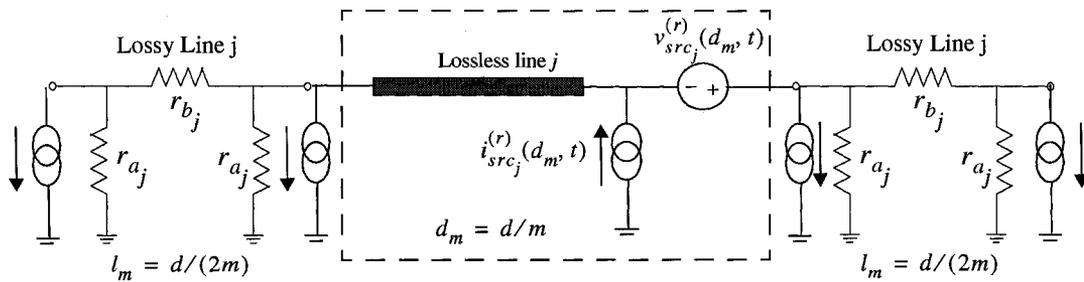


Figure 6.4: Overall decoupled DEFACT cell for line j

6.6 Numerical Results

In this section we present two case studies. The first example validates the accuracy of the proposed method, and the WR source formulations. In the second example, the new algorithm is applied to a relatively long transmission line circuit. The examples validate the fact that the new method is suitable for both short and long lines (on and off-chip interconnects), which is one of its key features.

6.6.1 Example 1

In this example, a two coupled line circuit is considered. The length of the line is $d = 2\text{cm}$. The line parameters were extracted from [14]. The input voltage is a pulse with rise/fall times of 0.5ns and a pulse width of 5ns. The terminations were 50Ω resistors at the near end and 1 pF capacitors at the far-end. Figures 6.5 - 6.6 show a sample of the transient response after 2 WR iterations, for a DEPACT order of $m = 1$. The results are compared to the coupled uniform lumped segmentation in HSPICE. As seen, the results are in excellent agreement. In addition, the CPU time for the proposed WR-TP+DEPACT algorithm was 0.28 s, while the lumped segmentation model required 1.8 s.

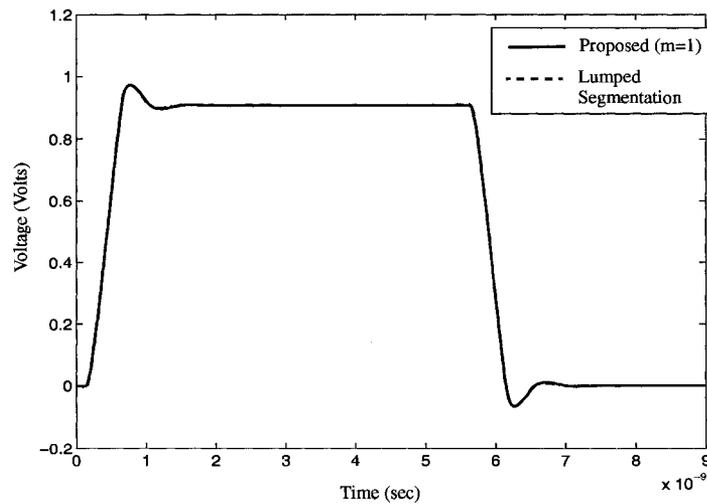


Figure 6.5: Voltage at far-end of active line (Example 1)

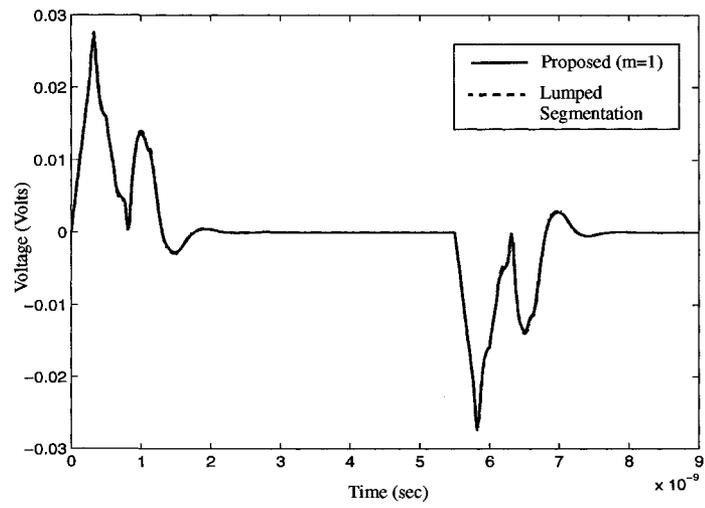


Figure 6.6: Voltage at near-end of victim line (Example 1)

6.6.2 Example 2

In this example, we consider a three-coupled-line circuit with $d = 25$ cm. The input voltage is a pulse with rise/fall times of 0.5ns and a pulse width of 5ns, and the terminations are the same as in Example 1. Figures 6.7 - 6.8 show a sample of the results after 3 WR iterations using $m = 8$, vs. the coupled uniform lumped segmentation in HSPICE. As seen from the plots, the results are in excellent agreement. In addition, the CPU time for the proposed algorithm was only 0.96 s, compared to 7.0 s using uniform lumped segmentation.

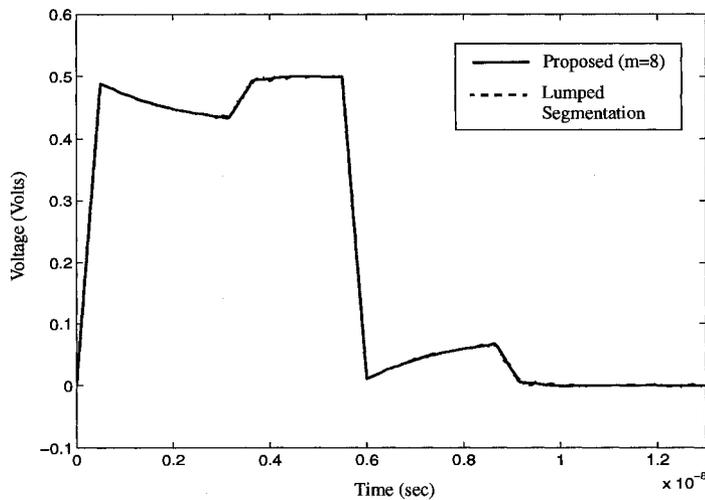


Figure 6.7: Voltage at near-end of active line (Example 2)

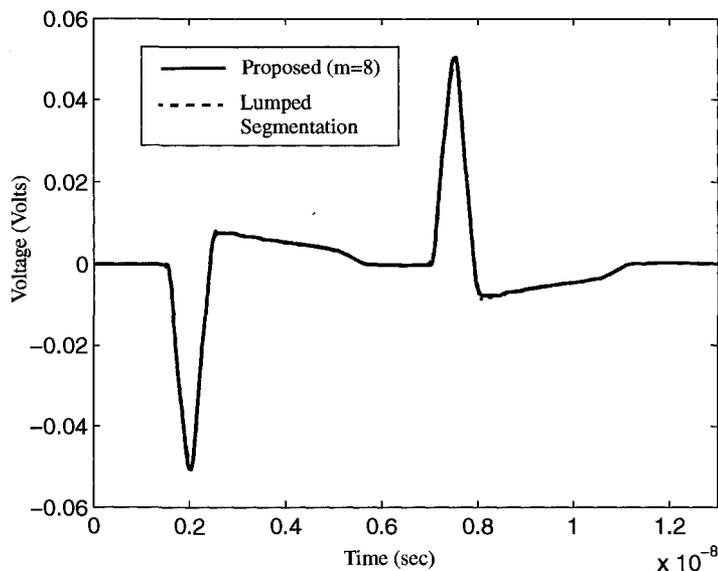


Figure 6.8: Voltage at far-end of victim line (Example 2)

6.7 Summary

In summary, this chapter presented a new delay extraction-based waveform relaxation with transverse partitioning (WR-TP) algorithm. In this method, each of the lossy and lossless portions of the DEFACT cell are partitioned separately in the transverse direction. Advantages of the proposed algorithm are that it is suitable for both long and short lines, and the WR sources are computed analytically in a closed-form, directly in the time-domain. Several examples were presented which validate the accuracy and efficiency of the new method. In the next chapter, a new general algorithm for time-domain sensitivity analysis of transmission lines is described.

Chapter 7

A General Approach for Time-Domain Sensitivity Analysis of High-Speed Interconnects

In this chapter, a new general method for time-domain sensitivity analysis of transmission lines is presented. Section 7.1 introduces the topic, and Section 7.2 presents an overview of the direct sensitivity method. Details of the new transmission line sensitivity analysis algorithm are presented in Section 7.3, and Section 7.4 provides computational results.

7.1 Introduction

Sensitivity analysis plays a major role in the design and optimization of high-speed circuits and interconnects. It provides designers with the ability to make the proper trade-offs, often between conflicting designer requirements, to obtain the best possible performance. In addition, when using optimization tools as part of the design cycle, the sensitivity vector in the form of the gradient of the cost function is crucial to obtain fast convergence. Moreover, sensitivity information is of significant importance in identifying critical design components and in tolerance

assignments [56, 57].

Several techniques have been presented in the literature for sensitivity analysis of distributed interconnects [56–60]. The commonly used approach is based on two steps. In the first step, the distributed interconnect represented by the Telegrapher's equations is replaced by a suitable lumped circuit macromodel. In general, the macromodel consists of RLC components and delay dependent sources. In the second step, the sensitivity of these lumped components due to variations in the interconnect parameters are evaluated and are subsequently related to the sensitivity of the overall circuit response. This approach has a major drawback: in general, it is not a straight forward process to obtain relationships between lumped components and interconnect parameters. For example, using the MoC transmission line macromodeling algorithm [10], the computation of the sensitivity of the eigenvalues and eigenvectors of the propagation matrix are usually required, and for the case of the MRA algorithm [12], the sensitivity of the coefficients of the approximating rational functions are needed. In addition, this approach for sensitivity analysis is restricted to the specific macromodel used. In other words, *for every new macromodeling technique, a new sensitivity algorithm has to be developed.*

In this chapter, a novel and generic method for sensitivity analysis of distributed interconnects is developed [20]. The new algorithm has two major advantages over the current techniques:

1. It is general in the sense that it is independent of the specific macromodeling approach used.
2. It does not require the computation of the sensitivity of the macromodel components with respect to the interconnect parameter of interest.

The new approach significantly simplifies the sensitivity computation process. In contrast to the current techniques in the literature, we start by differentiating the Telegrapher's equations with respect to the interconnect parameter of interest. This results in an equivalent "sensitivity circuit" which consists of lumped and distributed components, as well as independent sources. The sensitivity information is obtained by simulating the original and sensitivity circuits using any suitable macromodeling technique. In addition, the method is highly compatible with the WR-TP algorithms described in the previous chapters, opening the door for fast and efficient time-domain sensitivity analysis of massively coupled interconnect networks.

7.2 Sensitivity Analysis of Lumped Circuits

Sensitivity analysis of lumped circuits has been extensively described in the literature. One common approach is the direct sensitivity method [61]. This method involves the construction of a new sensitivity circuit with the same topology as the original circuit, with the voltages and currents being replaced by the *sensitivities* of the voltages and currents, respectively. The solution of the sensitivity circuit yields the required sensitivity information. A review of the direct sensitivity method is presented in the remainder of this section.

7.2.1 Sensitivity Circuits for RLC components

Consider a linear resistor (Figure 7.1(a)) with the terminal relationship given by

$$v = iR \tag{7.1}$$

Differentiating (7.1) with respect to a parameter of interest λ , yields

$$\frac{\partial v}{\partial \lambda} = \frac{\partial R}{\partial \lambda} i + R \frac{\partial i}{\partial \lambda} \quad (7.2)$$

Therefore, the resulting sensitivity circuit (Figure 7.1(b)) consists of a resistor with resistance R , in series with an independent voltage source with a value of $\frac{\partial R}{\partial \lambda} i$, where the current i is obtained from the solution of the original circuit. Note that the variables in the sensitivity circuit are the derivatives of the voltages and currents with respect to the parameter λ .

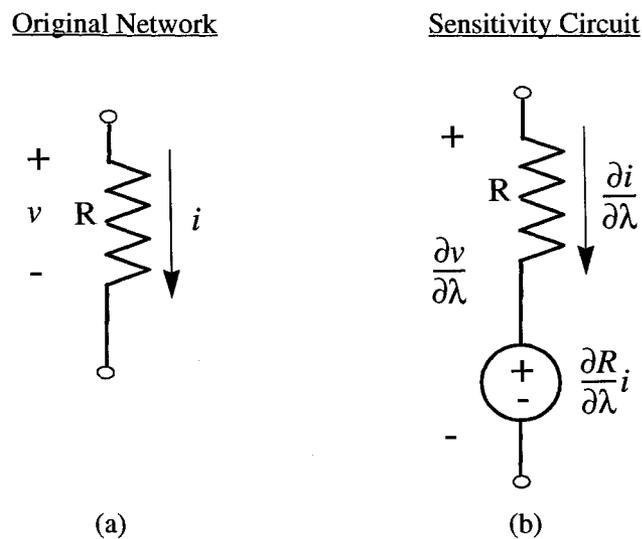


Figure 7.1: Companion model for resistors

Similarly, for inductors, the relationship between the voltage and current can be written as

$$v = L \frac{di}{dt} \quad (7.3)$$

Differentiating (7.3) with respect to λ , we get

$$\begin{aligned}\frac{\partial v}{\partial \lambda} &= \frac{\partial L}{\partial \lambda} \frac{di}{dt} + L \frac{\partial}{\partial \lambda} \frac{\partial i}{\partial t} \\ &= \frac{\partial L}{\partial \lambda} \frac{v}{L} + L \frac{\partial}{\partial \lambda} \frac{\partial i}{\partial t}\end{aligned}\quad (7.4)$$

From the above equation, each inductor in the original circuit is replaced by an inductor of the same value, in series with an independent voltage source equal to $\frac{\partial L}{\partial \lambda} \frac{v}{L}$, as shown in Figure 7.2(b), where the voltage v is obtained from the solution of the original circuit.

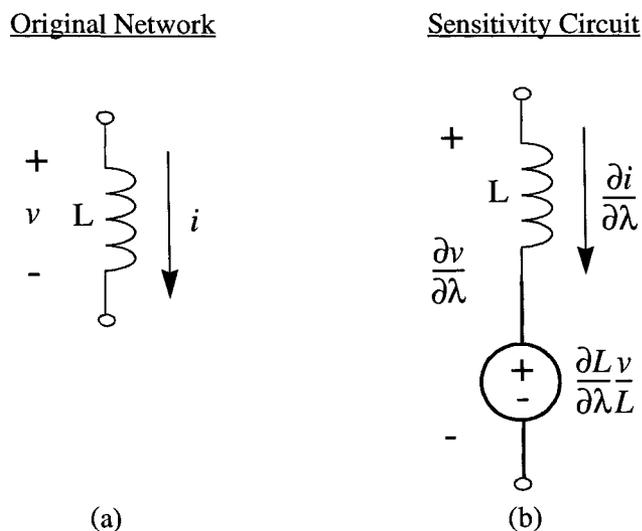


Figure 7.2: Companion model for inductors

Next, for capacitors (Figure 7.3(a)), we can write

$$i = C \frac{dv}{dt} \quad (7.5)$$

Differentiating (7.5) yields the following

$$\begin{aligned} \frac{\partial i}{\partial \lambda} &= \frac{\partial C}{\partial \lambda} \frac{dv}{dt} + C \frac{\partial}{\partial \lambda} \frac{\partial v}{\partial t} \\ &= \frac{\partial C}{\partial \lambda} \frac{i}{C} + C \frac{\partial}{\partial \lambda} \frac{\partial v}{\partial t} \end{aligned} \quad (7.6)$$

In this case, the sensitivity circuit consists of a capacitor with capacitance C , in parallel with an independent current source which is given by $\frac{\partial C}{\partial \lambda} \frac{i}{C}$, as shown in Figure 7.3(b), where the current i is acquired from the solution of the original circuit.

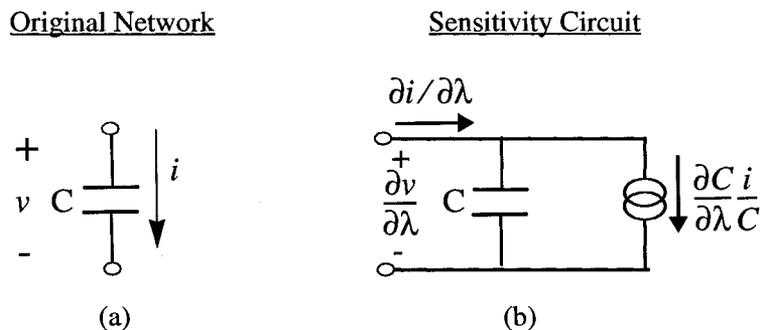


Figure 7.3: Companion model for capacitors

For independent voltage and current input sources, v_s and i_s , in the sensitivity circuit they are replaced with $\frac{\partial v_s}{\partial \lambda}$ and $\frac{\partial i_s}{\partial \lambda}$. However, usually the independent sources are not functions of λ , and thus can be replaced with closed and open circuits, respectively [61].

7.2.2 Sensitivity Circuits for Dependent Sources

- Voltage Controlled Current Sources (VCCS) :

Consider the VCCS shown in Figure 7.4(a) defined by the following terminal relationship

$$i_b = g_m v_a \quad (7.7)$$

where g_m is the transconductance of the device.

(Note: In Figures 7.4-7.7, the diamond shaped source represents a controlled source).

Differentiating (7.7) with respect to λ yields

$$\frac{\partial i_b}{\partial \lambda} = g_m \frac{\partial v_a}{\partial \lambda} + \frac{\partial g_m}{\partial \lambda} v_a \quad (7.8)$$

Therefore, the sensitivity circuit for a VCCS (Figure 7.4(b)) consists of the original controlled current source in parallel with an independent current source, where the voltage v_a is known from the solution of the original circuit.

- Voltage Controlled Voltage Sources (VCVS):

The terminal relationship for the VCVS shown in Figure 7.5(a) is given by

$$v_b = \mu v_a \quad (7.9)$$

Differentiating (7.9) we get

$$\frac{\partial v_b}{\partial \lambda} = \mu \frac{\partial v_a}{\partial \lambda} + \frac{\partial \mu}{\partial \lambda} v_a \quad (7.10)$$

Figure 7.5(b) shows the sensitivity circuit for the VCVS, which consists of the same controlled voltage source in series with an independent voltage source, where v_a is

obtained from the solution of the original network.

- Current Controlled Current Sources (CCCS):

Consider the CCCS shown in Figure 7.6(a) with the following terminal relationship

$$i_b = \alpha i_a \quad (7.11)$$

Differentiating (7.11) yields

$$\frac{\partial i_b}{\partial \lambda} = \alpha \frac{\partial i_a}{\partial \lambda} + \frac{\partial \alpha}{\partial \lambda} i_a \quad (7.12)$$

Therefore, the sensitivity circuit for the CCCS (Figure 7.6(b)) is composed of an identical controlled source in parallel with an independent current source, where the current i_a is known from the solution of the original circuit.

- Current Controlled Voltage Sources (CCVS):

The terminal relationship for the CCVS shown in Figure 7.7(a) is given by

$$v_b = r_m i_a \quad (7.13)$$

Differentiating (7.13) we get

$$\frac{\partial v_b}{\partial \lambda} = r_m \frac{\partial i_a}{\partial \lambda} + \frac{\partial r_m}{\partial \lambda} i_a \quad (7.14)$$

Figure 7.7(b) shows the sensitivity circuit for the CCVS which consists of the original controlled source in series with a known independent voltage source, and i_a is obtained from the solution of the original circuit.

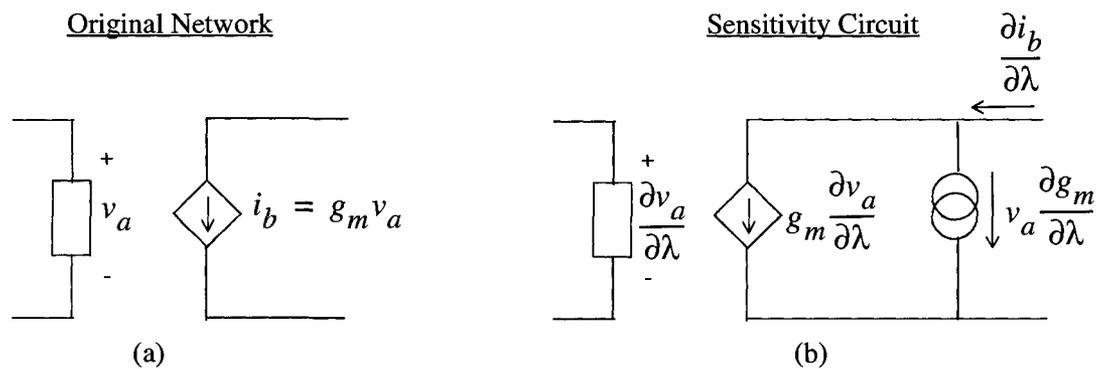


Figure 7.4: Companion model for voltage controlled current source (VCCS)

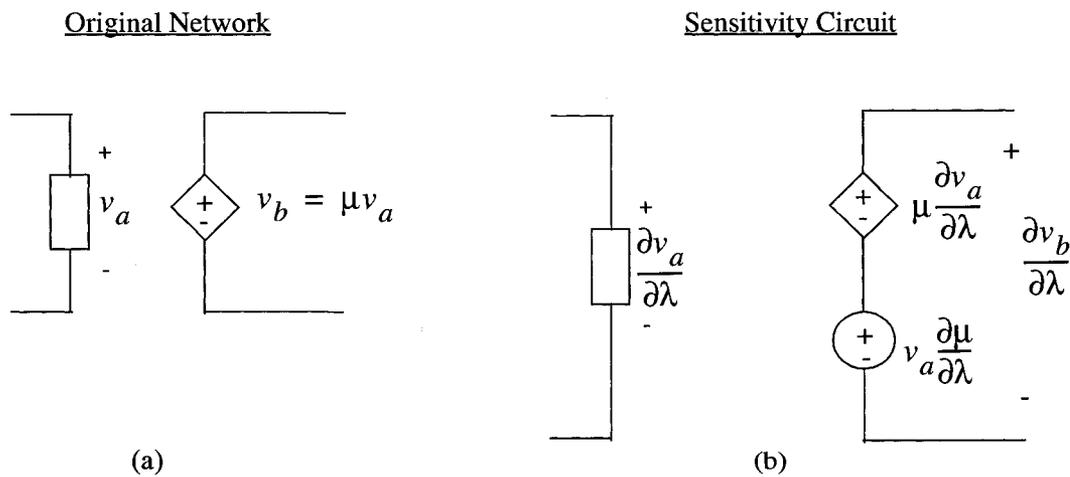


Figure 7.5: Companion model for voltage controlled voltage source (VCVS)

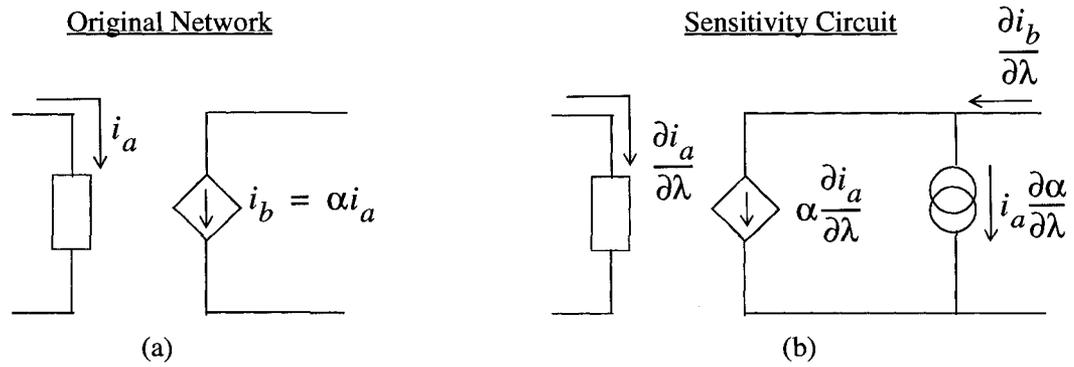


Figure 7.6: Companion model for current controlled current source (CCCS)

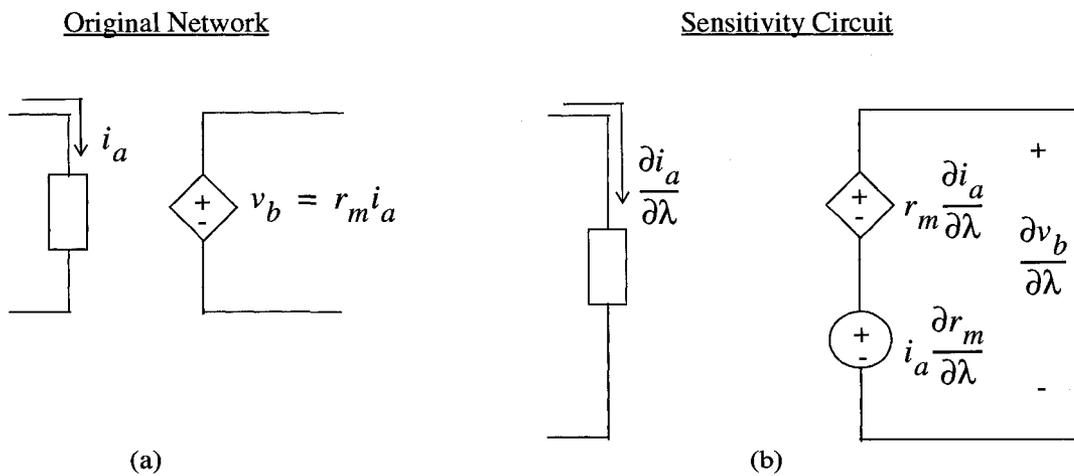


Figure 7.7: Companion model for current controlled voltage source (CCVS)

7.2.3 Sensitivity Circuits for Nonlinear Components

Consider a nonlinear component (shown in Figure 7.8(a)), with the terminal relationship given by

$$i = f(v, \lambda) \quad (7.15)$$

Differentiating (7.15), yields

$$\frac{\partial i}{\partial \lambda} = \frac{\partial f}{\partial v} \frac{\partial v}{\partial \lambda} + \frac{\partial f}{\partial \lambda} \quad (7.16)$$

The resulting companion model for (7.15) is shown in Figure 7.8(b), which consists of a time-variant conductance G_t , in parallel with an independent current source $\frac{\partial f}{\partial \lambda}$, where

$$G_t = \left. \frac{\partial f}{\partial v} \right|_{v(t)} \quad (7.17)$$

and where $v(t)$ is acquired from the solution of the original circuit [61].

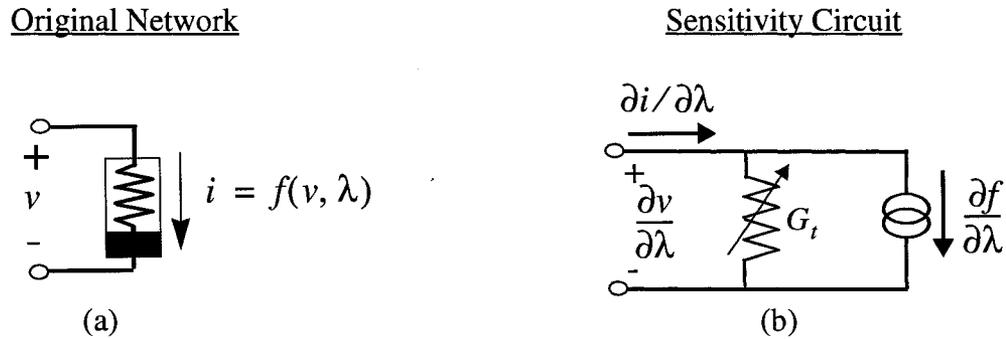


Figure 7.8: Companion model for nonlinear resistors

7.3 Sensitivity Analysis of Transmission Lines

In this section, the direct sensitivity approach is extended to transmission line circuits. In the new method, the equivalent sensitivity circuit is derived directly from the Telegrapher's equations. Advantages are that it is independent of the specific macro-modeling technique used and it does not require the computation of the sensitivity of the specific macromodel components with respect to the interconnect parameter of interest.

7.3.1 Companion Model for Transmission Lines

Recall the Telegrapher's equations describing the transmission lines in the following form [9]

$$\begin{aligned}
 \frac{\partial}{\partial x} \mathbf{v}(x, t) &= -\mathbf{R} \mathbf{i}(x, t) - \mathbf{L} \frac{\partial}{\partial t} \mathbf{i}(x, t) \\
 \frac{\partial}{\partial x} \mathbf{i}(x, t) &= -\mathbf{G} \mathbf{v}(x, t) - \mathbf{C} \frac{\partial}{\partial t} \mathbf{v}(x, t)
 \end{aligned} \tag{7.18}$$

where $\mathbf{R} \in \mathfrak{R}^{N \times N}$, $\mathbf{L} \in \mathfrak{R}^{N \times N}$, $\mathbf{C} \in \mathfrak{R}^{N \times N}$, and $\mathbf{G} \in \mathfrak{R}^{N \times N}$ are the p.u.l. parameters of the line, $\mathbf{v}(x, t) \in \mathfrak{R}^N$ and $\mathbf{i}(x, t) \in \mathfrak{R}^N$ represent the voltage and current vectors as a function of position x and time t , and N is the number of coupled lines.

Differentiating (7.18) with respect to a parameter λ (where λ represents any electrical or physical interconnect parameter of interest) yields the following relationship

$$\begin{aligned} \frac{\partial}{\partial x} \tilde{\mathbf{v}}(x, t) &= -\mathbf{R} \tilde{\mathbf{i}}(x, t) - \mathbf{L} \frac{\partial}{\partial t} \tilde{\mathbf{i}}(x, t) - \left(\frac{\partial \mathbf{R}}{\partial \lambda} \mathbf{i}(x, t) + \frac{\partial \mathbf{L}}{\partial \lambda} \frac{\partial}{\partial t} \mathbf{i}(x, t) \right) \\ \frac{\partial}{\partial x} \tilde{\mathbf{i}}(x, t) &= -\mathbf{G} \tilde{\mathbf{v}}(x, t) - \mathbf{C} \frac{\partial}{\partial t} \tilde{\mathbf{v}}(x, t) - \left(\frac{\partial \mathbf{G}}{\partial \lambda} \mathbf{v}(x, t) + \frac{\partial \mathbf{C}}{\partial \lambda} \frac{\partial}{\partial t} \mathbf{v}(x, t) \right) \end{aligned} \quad (7.19)$$

where the sensitivity variables in (7.19) are defined as

$$\tilde{\mathbf{v}}(x, t) = \frac{\partial}{\partial \lambda} \mathbf{v}(x, t); \quad \tilde{\mathbf{i}}(x, t) = \frac{\partial}{\partial \lambda} \mathbf{i}(x, t) \quad (7.20)$$

Next, the solution of (7.19) can be written in the frequency-domain as

$$\begin{bmatrix} \tilde{\mathbf{V}}(d, s) \\ \tilde{\mathbf{I}}(d, s) \end{bmatrix} = e^{(\tilde{\mathbf{A}} + s\tilde{\mathbf{B}})d} \begin{bmatrix} \tilde{\mathbf{V}}(0, s) \\ \tilde{\mathbf{I}}(0, s) \end{bmatrix} + \int_0^d e^{(\tilde{\mathbf{A}} + s\tilde{\mathbf{B}})(d-\eta)} \Phi(\eta, s) d\eta \quad (7.21)$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{0} & -\mathbf{R} \\ -\mathbf{G} & \mathbf{0} \end{bmatrix}; \quad \tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{0} & -\mathbf{L} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} \quad (7.22)$$

$$\Phi(\eta, s) = - \begin{bmatrix} \mathbf{0} & \frac{\partial \mathbf{R}}{\partial \lambda} + s \frac{\partial \mathbf{L}}{\partial \lambda} \\ \frac{\partial \mathbf{G}}{\partial \lambda} + s \frac{\partial \mathbf{C}}{\partial \lambda} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}(\eta, s) \\ \mathbf{I}(\eta, s) \end{bmatrix} \quad (7.23)$$

and d is the length of the line.

The sensitivity equations in (7.21) can be modeled as a transmission line circuit with independent sources at the end of the lines, as shown in Figure 7.9. In this case, the variables are the *sensitivities* of the circuit voltages and currents, with respect to the parameter λ . The sensitivity sources are obtained from the solution of the original circuit and can be expressed in the time-domain as

$$\begin{bmatrix} \mathbf{u}(d, t) \\ \mathbf{p}(d, t) \end{bmatrix} = \mathcal{F}^{-1} \left\{ \begin{bmatrix} \mathbf{U}(d, s) \\ \mathbf{P}(d, s) \end{bmatrix} \right\} = \mathcal{F}^{-1} \left\{ \int_0^d e^{(\tilde{\mathbf{A}} + s\tilde{\mathbf{B}})(d-\eta)} \Phi(\eta, s) d\eta \right\} \quad (7.24)$$

where $\mathcal{F}^{-1}\{\}$ denotes the inverse Fourier transform operator.

The equations in (7.21)-(7.24) are general in nature and thus any macromodel can be used to realize the transmission line portion of these equations.

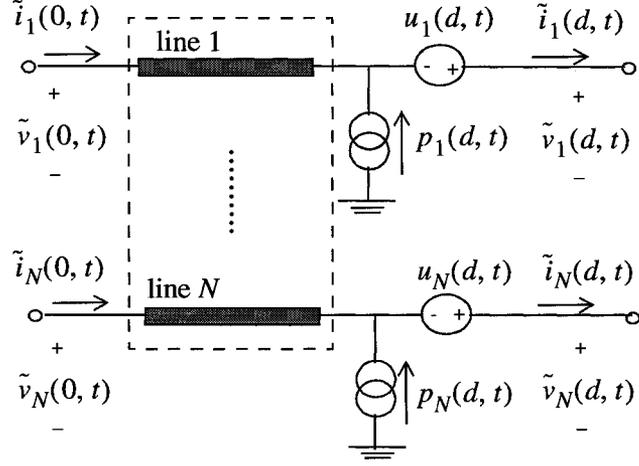


Figure 7.9: Equivalent sensitivity TL circuit

7.3.2 Computation of the Sensitivity Sources

The direct calculation of sensitivity sources in (7.24) can, in general, be computationally expensive. This is because we need to convert the time-domain voltages and currents at all points along the line (at every η) to the frequency-domain, using the FFT, followed by the numerical evaluation of the integral. In this section, a new and more efficient eigenvalue based approach for obtaining the sensitivity sources is presented. In this approach, only the frequency spectrum of the voltages and currents of the near-end of the transmission lines are required. In addition, a closed-form solution for the integral in (7.24) is obtained.

From (7.24), the sources can be written in the frequency-domain as

$$\boldsymbol{\delta}_{sens} = \begin{bmatrix} \mathbf{U}(d, s) \\ \mathbf{P}(d, s) \end{bmatrix} = \int_0^d e^{(\tilde{\mathbf{A}}+s\tilde{\mathbf{B}})(d-\eta)} \boldsymbol{\Phi}(\eta, s) d\eta \quad (7.25)$$

Using the exponential stamp of the line [3],

$$\begin{bmatrix} \mathbf{V}(\eta, s) \\ \mathbf{I}(\eta, s) \end{bmatrix} = e^{(\tilde{\mathbf{A}} + s\tilde{\mathbf{B}})\eta} \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \quad (7.26)$$

and moving the terms which are independent of η outside the integral, (7.25) can be re-written as

$$\delta_{sens} = e^{\Upsilon d} \left[\int_0^d e^{-\Upsilon\eta} \tilde{\mathbf{H}}(s) e^{\Upsilon\eta} d\eta \right] \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \quad (7.27)$$

where

$$\Upsilon = \tilde{\mathbf{A}} + s\tilde{\mathbf{B}} \quad (7.28)$$

and $\tilde{\mathbf{H}}(s)$ is given by

$$\tilde{\mathbf{H}}(s) = - \begin{bmatrix} \mathbf{0} & \frac{\partial \mathbf{R}}{\partial \lambda} + s \frac{\partial \mathbf{L}}{\partial \lambda} \\ \frac{\partial \mathbf{G}}{\partial \lambda} + s \frac{\partial \mathbf{C}}{\partial \lambda} & \mathbf{0} \end{bmatrix} \quad (7.29)$$

Defining the following

$$\mathbf{J}_s = \int_0^d e^{-\Upsilon\eta} \tilde{\mathbf{H}}(s) e^{\Upsilon\eta} d\eta \quad (7.30)$$

we get

$$\boldsymbol{\delta}_{sens} = e^{\boldsymbol{\Upsilon}d} \mathbf{J}_s \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \quad (7.31)$$

where $\mathbf{V}(0, s)$ and $\mathbf{I}(0, s)$ are the voltage and current vectors at the beginning of the transmission lines, respectively.

Next, $e^{\boldsymbol{\Upsilon}\eta}$ can be diagonalized as

$$e^{\boldsymbol{\Upsilon}\eta} = \mathbf{V}_s \text{diag}\{e^{\gamma_1\eta}, e^{\gamma_2\eta}, \dots, e^{\gamma_{2N}\eta}\} \mathbf{V}_s^{-1} \quad (7.32)$$

where $\gamma_1, \gamma_2, \dots, \gamma_N$ and \mathbf{V}_s are the eigenvalues and eigenvectors of $\boldsymbol{\Upsilon}$, respectively, and $\gamma_i = -\gamma_j$, for $i = N + 1, \dots, 2N$ and $j = 1, \dots, N$.

Using (7.32), \mathbf{J}_s in (7.30) can be re-written as

$$\mathbf{J}_s = \mathbf{V}_s \left(\int_0^d \text{diag}\{e^{-\gamma_1\eta}, \dots, e^{-\gamma_{2N}\eta}\} \mathbf{K}_s \text{diag}\{e^{\gamma_1\eta}, \dots, e^{\gamma_{2N}\eta}\} d\eta \right) \mathbf{V}_s^{-1} \quad (7.33)$$

where the matrix \mathbf{K}_s is given by

$$\mathbf{K}_s = \mathbf{V}_s^{-1} \tilde{\mathbf{H}}(s) \mathbf{V}_s \quad (7.34)$$

The term inside the integral in (7.33) can be expressed in a general form as follows

$$\begin{aligned} \mathbf{G}_s &= \text{diag}\{e^{-\gamma_1\eta}, \dots, e^{-\gamma_{2N}\eta}\} \mathbf{K}_s \text{diag}\{e^{\gamma_1\eta}, \dots, e^{\gamma_{2N}\eta}\} \\ &= \{\mathbf{G}_s(i, j); i, j = 1, \dots, 2N\} \end{aligned} \quad (7.35)$$

where

$$\mathbf{G}_s(i, j) = \begin{cases} \mathbf{K}_s(i, j), & i = j \\ \mathbf{K}_s(i, j)e^{(\gamma_j - \gamma_i)\eta}, & i \neq j \end{cases} \quad (7.36)$$

Integrating \mathbf{G}_s with respect to η , we get

$$\begin{aligned} \tilde{\mathbf{G}}_s &= \int_0^d \mathbf{G}_s d\eta \\ &= \{\tilde{\mathbf{G}}_s(i, j); i, j = 1, \dots, 2N\} \end{aligned} \quad (7.37)$$

where the elements in $\tilde{\mathbf{G}}_s$ are given by

$$\tilde{\mathbf{G}}_s(i, j) = \begin{cases} \mathbf{K}_s(i, j)d, & i = j \\ \frac{\mathbf{K}_s(i, j)}{\gamma_j - \gamma_i} e^{(\gamma_j - \gamma_i)d} - \frac{\mathbf{K}_s(i, j)}{\gamma_j - \gamma_i}, & i \neq j \end{cases} \quad (7.38)$$

Using (7.37)-(7.38), \mathbf{J}_s can be expressed as

$$\mathbf{J}_s = \mathbf{V}_s \tilde{\mathbf{G}}_s \mathbf{V}_s^{-1} \quad (7.39)$$

Finally, the sensitivity voltage and current sources, $\mathbf{u}(d, t)$ and $\mathbf{p}(d, t)$ can be obtained using (7.39), and

$$\begin{bmatrix} \mathbf{u}(d, t) \\ \mathbf{p}(d, t) \end{bmatrix} = \mathcal{F}^{-1} \left\{ e^{\mathbf{r}d} \mathbf{J}_s \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \right\} \quad (7.40)$$

7.3.3 Overall Sensitivity Circuit

Following the approaches described in the previous sections, the overall sensitivity circuit can be obtained by using the derived companion models for the transmission lines as well as the lumped components. To illustrate, consider the transmission line network shown in Figure 7.10(a). The corresponding overall sensitivity circuit is shown in Figure 7.10(b), which has the same topology as the original network. It is important to note, that the transmission lines in both the original and sensitivity circuits can be macromodeled using any desired algorithm.

In the case where λ represents an interconnect physical parameter, the sensitivity of an output node v_{out} can be obtained as follows

$$\frac{\partial v_{out}}{\partial \lambda} = \sum_{i=1}^N \sum_{j=1}^N \left(\frac{\partial v_{out}}{\partial R_{i,j}} \frac{\partial R_{i,j}}{\partial \lambda} + \frac{\partial v_{out}}{\partial L_{i,j}} \frac{\partial L_{i,j}}{\partial \lambda} + \frac{\partial v_{out}}{\partial G_{i,j}} \frac{\partial G_{i,j}}{\partial \lambda} + \frac{\partial v_{out}}{\partial C_{i,j}} \frac{\partial C_{i,j}}{\partial \lambda} \right) \quad (7.41)$$

where the subscript i, j represents the element in position (i, j) of the corresponding p.u.l. parameter matrices.

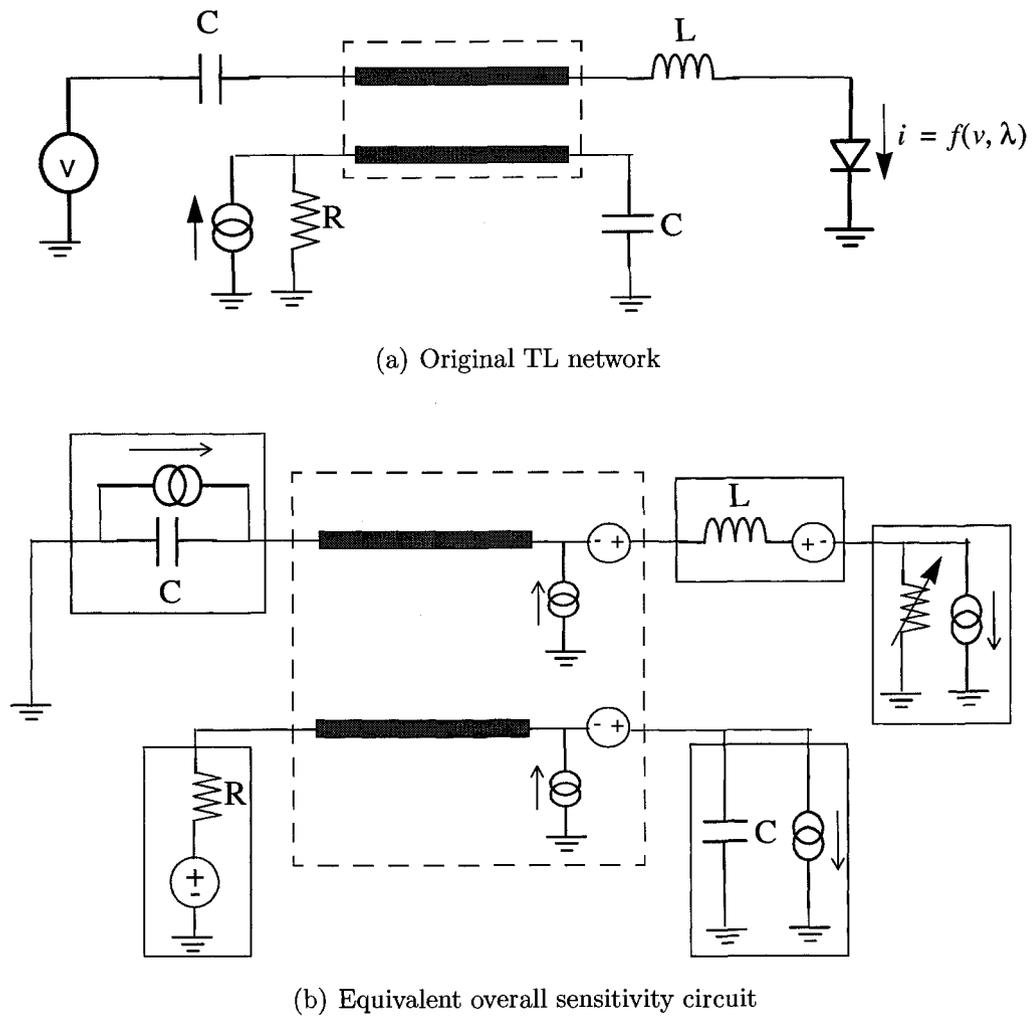


Figure 7.10: Example TL network and corresponding sensitivity circuit

7.3.4 Application to Waveform Relaxation with Transverse Partitioning

Due to the general nature of the new sensitivity analysis method, the transmission lines can be represented using any suitable macromodeling technique. Specifically, the sensitivity method can be applied to the WR-TP algorithms which were developed in the previous chapters of this thesis. To illustrate, consider the WR-TP method using the distributed representation which was described in Chapter 3. Applying the WR-TP algorithm to the coupled sensitivity circuit in Figure 7.9, results in the *decoupled* sensitivity circuit presented in Figure 7.11. In Figure 7.11, the variables are the derivatives of the voltages and currents with respect to λ , and the independent sensitivity sources $u_j(d, t)$, and $p_j(d, t)$, are obtained from the solution of the original decoupled subcircuits.

Next, consider the delay extraction-based WR-TP algorithm which was presented in Chapter 6. Applying this method to the coupled sensitivity circuit in Figure 7.9, results in the decoupled sensitivity circuit shown in Figure 7.12. Here, the sensitivity circuit consists of a cascade of decoupled DEFACT cells and independent WR sources, and sensitivity voltage and current sources located at the end of the line.

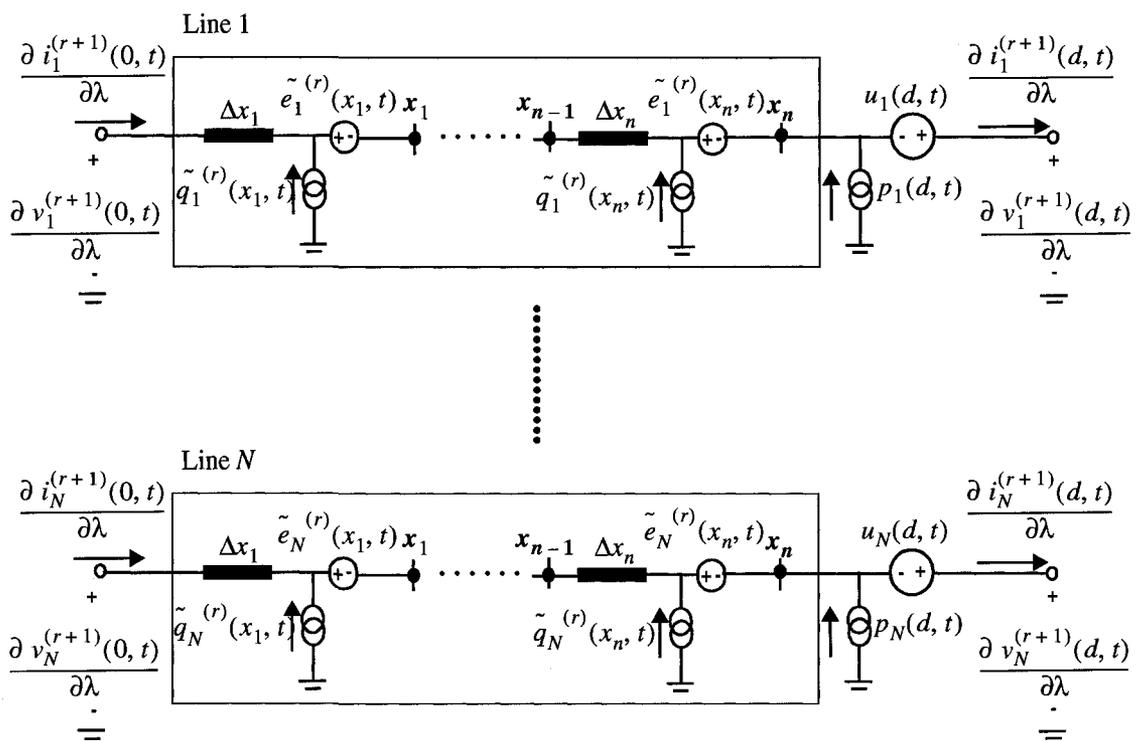


Figure 7.11: Application of the WR-TP distributed representation to the sensitivity circuit in Figure 7.9

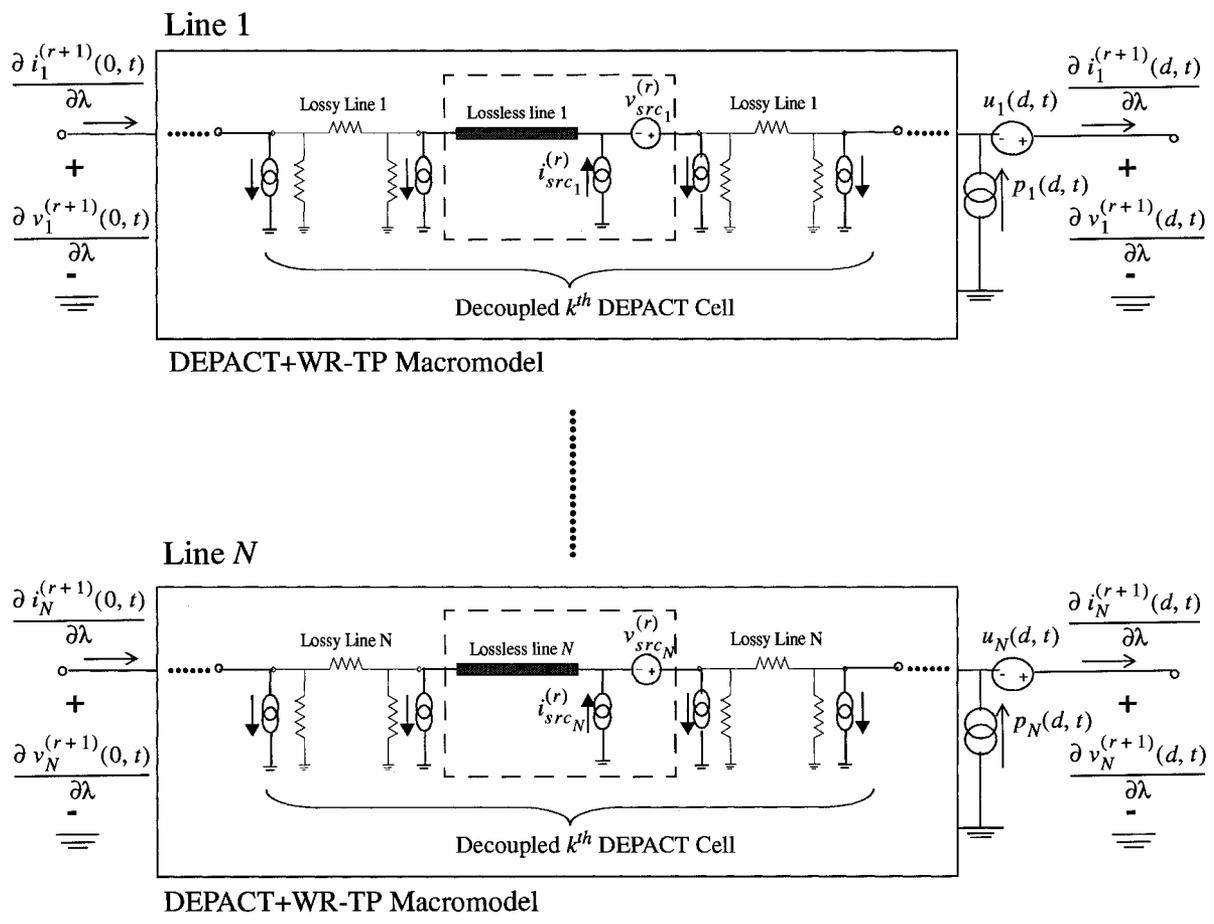


Figure 7.12: Application of the delay extraction based WR-TP algorithm to the sensitivity circuit in Figure 7.9

7.4 Computational Results

7.4.1 Example

In this example, we consider a network which consists of eight coupled transmission lines of length 10 cm. The p.u.l. parameters are based on the data in [13]. The input source is a 3V trapezoidal pulse with rise/fall times of 0.5ns and a pulse width of 5ns. The coupled interconnects in both the original and sensitivity circuits were macromodeled using the DEPACT algorithm [5] with an approximation order $m = 6$. A sample of the results is shown in Figure 7.13, which depicts the sensitivity of the voltage at the victim line far-end (line #7) with respect to L_{11} using the proposed method vs. the perturbation approach. As seen from the figure, the results from both methods are in excellent agreement. Table 7.1 shows a comparison of savings in the main computational cost (in terms of LU decompositions) using the new algorithm as compared to the perturbation technique. It is to be noted that the advantage in computational cost provided by the new method increases with the increase in the number of parameters.

Table 7.1: Sensitivity Analysis Computational Cost Comparison

# of Parameters	Perturbation (# of LU Decompositions)	Proposed Method (# of LU Decompositions)
18	23902	2516

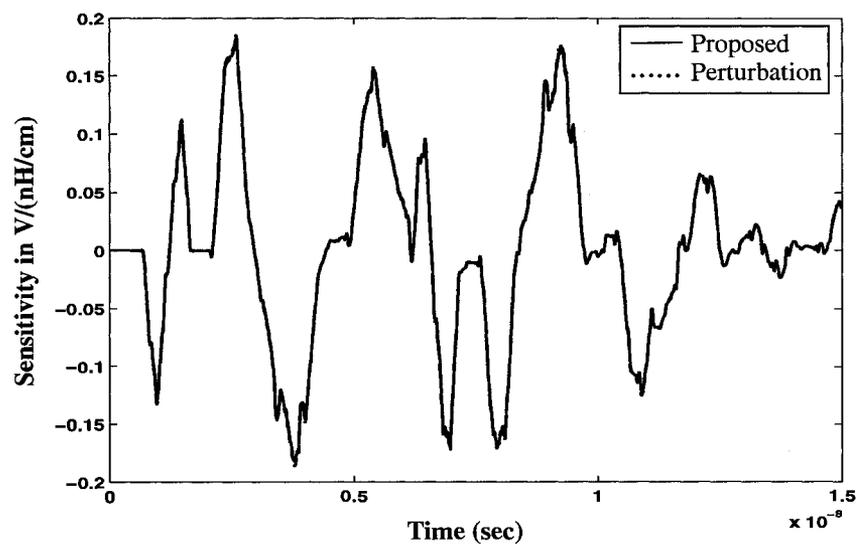


Figure 7.13: Sensitivity of voltage at far-end of victim line with respect to L_{11}

7.5 Summary

In this chapter, a new general and efficient algorithm for the time-domain sensitivity analysis of high-speed interconnects was presented. The new method extends the direct sensitivity approach to transmission lines. The equivalent sensitivity circuit is derived directly from the Telegrapher's equations. Advantages of the new sensitivity algorithm are that, it is independent of the specific macromodeling technique used and it does not require the computation of the sensitivity of the specific macromodel components with respect to the interconnect parameter of interest. In addition, a generic and efficient approach for computing the sensitivity sources was developed.

Chapter 8

Conclusions and Future Research

8.1 Summary

The coupling between transmission lines is one of the major reasons for the excessive computational cost of simulating large multiconductor structures. New interconnect macromodeling algorithms were developed in this thesis which address the computational complexity of the time-domain simulation of large coupled interconnect networks. A new waveform relaxation algorithm based on transverse partitioning (WR-TP) was developed. The new technique reduces the coupled simulation problem into a series of simulation steps, where each step is of complexity equivalent to that of simulating a single line. The computational cost of the new method grows only *linearly* with the number of lines (as opposed to N^α , where $3 \leq \alpha \leq 4$ for conventional techniques). Various approaches for updating the WR sources have also been developed. In addition, the WR-TP algorithm is highly parallelizable. Thus, array processors can be used to effectively simulate each subcircuit in parallel, which would provide further significant reduction in the computational cost.

The WR-TP algorithm was extended to the practically important case of lines with frequency-dependent parameters. In this approach, a numerical fitting algorithm is

employed to model the frequency-dependent line parameters. The resulting rational function approximations are used to generate equivalent macromodels compatible with SPICE-like circuit simulators. An efficient method for updating the relaxation sources was developed. Also, several relevant implementation considerations were presented.

A new simplified delay extraction-based macromodeling algorithm was developed. In this approach, MRA based approximations are no longer required for several special cases. For lines with frequency independent parameters, the resulting macromodels can be obtained as exact implementations (without any approximations) in terms of lossless lines and lumped elements. For lines with frequency-dependent parameters, an efficient modeling technique is used, and the resulting macromodels are obtained using equivalent circuit realizations and lossless lines.

Based on the simplified delay extraction-based macromodel, a new waveform relaxation with transverse partitioning algorithm was developed. In this approach, each of the lossy and lossless sections of the macromodel are partitioned separately in the transverse direction, into single line subcircuits with independent WR sources. Advantages are that the WR sources are computed analytically in a closed-form, directly in the time domain. In addition, since the algorithm employs an efficient delay extraction technique, it is suitable for both on and off-chip interconnects.

A new general algorithm for the time-domain sensitivity analysis of high-speed interconnects was developed. In this method, the equivalent sensitivity circuit is derived directly from the Telegrapher's equations. Advantages are that it is independent of the specific macromodeling technique used and it does not require the computation

of the sensitivity of the specific macromodel components with respect to the interconnect parameter of interest. A generic and efficient approach for computing the sensitivity sources was also developed. Moreover, the new sensitivity analysis method is highly compatible with the WR-TP algorithms described in the thesis, opening the door for fast and efficient time-domain sensitivity analysis of massively coupled interconnects. Various numerical examples were also presented to validate the accuracy and efficiency of all the proposed algorithms.

8.2 Future Research

1. The implementation of the transverse partitioning algorithm in this thesis assumed that the interconnect circuit is partitioned such that each subcircuit contains a single transmission line. However, the theory can be extended to the case where each subcircuit may contain two or more coupled lines. This will be beneficial if there is relatively strong coupling between adjacent lines. The simulation algorithm in this case is based on the concept of “sliding partitioning” where the first subcircuit contains lines 1 and 2; the second subcircuit contains lines 2 and 3; etc. Preliminary results show an improvement in convergence for the case of strongly coupled adjacent lines particularly in the high-frequency range.
2. Parallel Implementation: one of the main features of the proposed WR-TP algorithms is that they are all highly parallelizable. The implementation of the WR-TP using lumped segmentation on a multiprocessor platform has already been collaboratively developed [25]. Future work would involve:
 - (a) Parallel implementation on a multiprocessor platform of other WR-TP algorithms, namely, the delay extraction based WR-TP algorithm (Chapter

6).

(b) **Parallel Implementation of Sensitivity Analysis (Chapter 7).** Due to its parallel nature, the new sensitivity analysis algorithm combined with WRTP can be implemented on a multiprocessor platform. In this case, each of the decoupled sensitivity circuits would be simulated in parallel, with the processors only communicating after each iteration is complete. This new parallel algorithm would result in further reductions in the computational cost associated with the time-domain sensitivity analysis of massively coupled interconnect circuits.

3. **Hybrid Partitioning:** In the case of general large circuits, a hybrid partitioning algorithm combining both longitudinal and transverse partitioning schemes is needed. An example of such a circuit is one which consists of several interconnected sets of coupled transmission lines that may also include coupled terminations. A dynamic partitioning strategy in the longitudinal direction has to be developed such that there is minimum interaction between the individual subcircuits in a given time-interval.

List of References

- [1] A. Ruehli, A. Cangellaris, and H.-M. Huang, "Three test problems for the comparison of lossy transmission line algorithms," in *Proceedings of IEEE 11th Topical Meeting on Electrical Performance of Electronic Packaging*, (Monterey, California), pp. 347–350, Oct. 2002.
- [2] A. Deutsch, "Electrical characteristics of interconnections for high performance systems," *Proceedings of IEEE*, vol. 86, pp. 315–355, Feb. 1998.
- [3] R. Achar and M. Nakhla, "Simulation of high-speed interconnects," *Proceedings of IEEE*, vol. 89, pp. 693–728, May 2001.
- [4] *HSPICE: Star-Hspice Manual*. Synopsis.
- [5] N. Nakhla, A. Dounavis, R. Achar, and M. Nakhla, "DEPACT: Delay extraction-based passive compact transmission line macromodelling algorithm," *IEEE Transactions on Advanced Packaging*, vol. 28, pp. 13–23, Feb. 2005.
- [6] A. Dounavis, R. Achar, and M. Nakhla, "Efficient passive circuit models for distributed networks with frequency-dependent parameters," *IEEE Transactions on Advanced Packaging*, vol. 23, pp. 382–392, Aug. 2000.
- [7] D. Kuznetsov and J. Schutt-Aine, "Optimal transient simulation of transmission lines," *IEEE Transactions on Circuits and Systems*, vol. 43, pp. 110–121, Feb. 1996.
- [8] Q. Chu, Y. Lau, and F. Chang, "Transient analysis of microwave active circuits based on time domain characteristic models," *IEEE Transactions on Microwave Theory and Techniques*, vol. 46, pp. 1097–1104, Aug. 1998.
- [9] C. R. Paul, *Analysis of Multiconductor Transmission Lines*. New York: John Wiley and Sons, 1994.

- [10] F. Y. Chang, "The generalized method of characteristics for waveform relaxation analysis of lossy coupled transmission lines," *IEEE Transactions on Microwave Theory and Techniques*, vol. 37, pp. 2028–2038, Dec. 1989.
- [11] F. Y. Chang, "Transient analysis of lossless coupled transmission lines in a nonhomogeneous medium," *IEEE Transactions on Microwave Theory and Techniques*, vol. 18, pp. 1097–1104, Aug. 1998.
- [12] A. Dounavis, R. Achar, and M. Nakhla, "Addressing transient errors in passive macromodels of distributed transmission line networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, pp. 2759–2768, Dec. 2002.
- [13] A. Dounavis, R. Achar, and M. Nakhla, "A general class of passive macromodels for lossy multiconductor transmission lines," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, pp. 1686–1696, Oct. 2001.
- [14] A. Dounavis, *PhD Thesis: Passive Time-Domain Macromodels of High Speed Interconnect Networks*. Canada: Carleton University, Dec. 2003.
- [15] N. Nakhla, *Masters Thesis: Analytical Algorithms for Macromodeling and Sensitivity Analysis of High-Speed Interconnects*. Canada: Carleton University, Sept. 2005.
- [16] S. Grivet-Talocia and F. Canavero, "Topline: a delay pole-residue method for simulation of dispersive interconnects," in *Proceedings of IEEE 11th Topical Meeting on Electrical Performance of Electronic Packaging*, (Monterey, CA), pp. 359–362, Nov. 2002.
- [17] N. Nakhla, A. E. Ruehli, M. Nakhla, and R. Achar, "Simulation of coupled interconnects using waveform relaxation and transverse partitioning," *IEEE Transactions on Advanced Packaging*, vol. 29, pp. 78–87, Feb. 2006.
- [18] N. Nakhla, A. E. Ruehli, M. Nakhla, and R. Achar, "Simulation of coupled interconnects using waveform relaxation and transverse partitioning," in *Proceedings of IEEE 13th Topical Meeting on Electrical Performance of Electronic Packaging*, (Portland, OR), pp. 25–28, Oct. 2004.
- [19] N. Nakhla, A. E. Ruehli, M. Nakhla, R. Achar, and C. Chen, "Waveform relaxation techniques for simulation of coupled interconnects with frequency-dependent parameters," *IEEE Transactions on Advanced Packaging*, vol. 30, pp. 257–269, May 2007.

- [20] N. Nakhla, M. Nakhla, and R. Achar, "A general approach for time-domain sensitivity analysis of high-speed interconnects," in *Proceedings of IEEE 15th Topical Meeting on Electrical Performance of Electronic Packaging*, (Scottsdale, AZ), pp. 189–192, Oct. 2006.
- [21] N. Nakhla, M. Nakhla, and R. Achar, "Model order reduction of large multiport interconnect structures using waveform relaxation techniques," in *Proceedings of IEEE International Conference on Computer Aided Design, ICCAD'07*, (Camogli, Italy), pp. 69–70, May 2007.
- [22] N. Nakhla, M. Nakhla, and R. Achar, "Sparse and passive reduction of massively coupled large multiport interconnects," in *Proceedings of IEEE Workshop on Signal Propagation on Interconnects, SPI'07*, (San Jose, CA), pp. 622–626, Nov. 2007.
- [23] A. Sridar, N. Nakhla, R. Achar, and M. Nakhla, "Fast EMC analysis of high-speed interconnects via waveform relaxation and transverse partitioning," in *Proceedings of IEEE 16th Topical Meeting on Electrical Performance of Electronic Packaging*, (Atlanta, GA), pp. 329–392, Oct. 2007.
- [24] A. Sridar, N. Nakhla, R. Achar, M. Nakhla, A. Ruehli, and I. Erdin, "Fast EMC analysis of high-speed interconnects via waveform relaxation and transverse partitioning," *IEEE Transactions on Electromagnetic Compatibility*, Aug. 2008. Accepted, in press.
- [25] D. Paul, N. Nakhla, R. Achar, and M. Nakhla, "Parallel simulation of massively coupled interconnect networks," *Submitted to IEEE Transactions on Advanced Packaging*, July 2008.
- [26] S. Grivet-Talocia, H. Huang, A. Ruehli, F. Canavero, and I. Elfadel, "Transient analysis of lossy transmission lines: an efficient approach based on the method of characteristics," *IEEE Transactions on Advanced Packaging*, vol. 27, pp. 45–56, Feb. 2004.
- [27] F. H. Branin, "Transient analysis of lossless transmission lines," *Proceedings of IEEE*, vol. 55, pp. 2012–2013, Nov. 1967.
- [28] A. Dounavis, X. Li, M. Nakhla, and R. Achar, "Passive closed-loop transmission line model for general purpose circuit simulators," *IEEE Transactions on Microwave Theory and Techniques*, vol. 47, pp. 2450–2459, Dec. 1999.

- [29] T. Palenius and J. Roos, "Comparison of reduced-order interconnect macromodels for time-domain simulation," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, pp. 2240–2250, Sept. 2004.
- [30] I. Elfadel, H. Huang, A. Ruehli, A. Dounavis, and M. Nakhla, "A comparative study of two transient analysis algorithms for lossy transmission lines with frequency-dependent data," in *Proceedings of IEEE 10th Topical Meeting on Electrical Performance of Electronic Packaging*, (Cambridge, MA), pp. 255–258, Oct. 2001.
- [31] A. Ruehli, P. Feldmann, A. Deutsch, and H. Smith, "System and method efficient analysis of transmission lines," 2004. Docket YOR920040041US1, Filed with U.S. Patent Office.
- [32] A. Ruehli and T. Johnson, *Circuit Analysis Computing by Waveform Relaxation*, in *Encyclopedia of Electrical and Electronics Engineering*. New York: John Webster, Wiley, 1999.
- [33] E. Lelarasmee, A. Ruehli, and A. Sangiovanni-Vincentelli, "The waveform relaxation method for the time-domain analysis of large scale integrated circuits," *IEEE Transactions on Computer-Aided Design*, vol. 1, pp. 131–145, July 1982.
- [34] J. White and A. Sangiovanni-Vincentelli, "Relax 2: A modified waveform relaxation approach to the simulation of mos digital logic," in *Proceedings of IEEE International Symposium on Circuits and Systems*, (Newport Beach, CA), May 1983.
- [35] U. Miekkala and O. Nevanlinna, "Convergence of waveform relaxation method," in *Proceedings of IEEE International Symposium on Circuits and Systems*, (Espoo, Finland), pp. 1643–1646, June 1998.
- [36] J. White and A. Sangiovanni-Vincentelli, *Relaxation Techniques for the Simulation of VLSI Circuits*. Norwell, MA: Kluwer, 1987.
- [37] R. Wang and O. Wing, "Transient analysis of dispersive VLSI interconnects terminated in nonlinear loads," *IEEE Transactions on Computer-Aided Design*, vol. 11, pp. 1258–1277, Oct. 1992.
- [38] F. Y. Chang, "Waveform relaxation analysis of RLCG transmission lines," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 1394–1415, Nov. 1990.
- [39] P. Davis and P. Rabinowitz, *Methods of Numerical Integration*. New York: Academic Press, 1984.

- [40] R. Bagnara, "A unified proof for the convergence of jacobi and gauss- seidel methods," *Society for Industrial and Applied Mathematics Review*, vol. 37, pp. 93–97, March 1995.
- [41] G. Gristede, A. E. Ruehli, and C. Zukowski, "Convergence properties of waveform relaxation circuit simulation methods," *IEEE Transactions on Circuits and Systems*, vol. 45, pp. 726–738, July 1995.
- [42] G. Antonini, J. Ekman, and A. E. Ruehli, "Waveform relaxation for the parallel solution of large PEEC model problems," *To appear in IEEE Transactions on Electromagnetic Compatibility*.
- [43] S. Mei and Y. Ismail, "Modeling skin and proximity effects with reduced realizable RL circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, pp. 437–447, April 2004.
- [44] T. Wuyts and D. Zutter, "Circuit model for plane-wave incidence on multiconductor transmission lines," *IEEE Transactions on Electromagnetic Compatibility*, vol. 36, pp. 206–212, Aug. 1994.
- [45] T. V. Dinh, B. Cabon, and J. Chilo, "Time domain analysis of skin effect on lossy interconnections," *Electronics Letters*, vol. 26, pp. 2057–2058, Dec. 1990.
- [46] D. Saraswat, R. Achar, and M. Nakhla, "A fast algorithm and practical considerations for passive macromodeling of measured/simulated data," *IEEE Transactions on Advanced Packaging*, vol. 27, pp. 57–70, Feb. 2004.
- [47] S. Grivet-Talocia, "Passivity enforcement via perturbation of hamiltonian matrices," *IEEE Transactions on Circuits and Systems I*, vol. 51, pp. 1755–1769, Sept. 2004.
- [48] G. Antonini, A. Ruehli, and C. Yang, "PEEC modelling of dispersive and lossy dielectrics," in *Proceedings of IEEE 12th Topical Meeting on Electrical Performance of Electronic Packaging*, (Princeton, NJ), pp. 349–352, Oct. 2003.
- [49] B. Gustavsen and A. Semlyen, "Rational approximation of frequency responses by vector fitting," *IEEE Transactions on Power Delivery*, vol. 14, pp. 1052–1061, July 1999.
- [50] K. C. Branch, J. Morsey, A. Cangellaris, and A. Ruehli, "Physically consistent transmission line models for high-speed interconnects in lossy dielectrics," *IEEE Transactions on Advanced Packaging*, vol. 25, pp. 129–135, May 2002.

- [51] J. Vlach, *Computerized Approximation and Synthesis of Linear Networks*. New York: John Wiley and Sons, 1969.
- [52] H. Spath, *One Dimensional Spline Interpolation Algorithms*. MA: A.K. Peters, 1995.
- [53] A. Ruehli, *Advances in CAD for VLSI: Circuit Analysis, Simulation and Design*. 1987.
- [54] J. White and A. Sangiovanni-Vincentelli, "Partitioning algorithms and parallel implementations of waveform relaxation algorithms for circuit simulation," in *Proceedings of IEEE International Symposium on Circuits and Systems*, (Kyoto, Japan), pp. 1069–1072, 1985.
- [55] Alina Deutsch, Private Communications, 2006.
- [56] Q. Zhang, S. Lum, and M. Nakhla, "Minimization of delay and crosstalk in high-speed VLSI interconnects," *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, pp. 1555–1563, July 1992.
- [57] T. Arabi, R. Suarez-Gartner, and R. Pomerleau, "Optimization and sensitivity analysis of multiconductor transmission line networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, pp. 1827–1836, Sept. 1994.
- [58] S. Lum, M. Nakhla, and Q. Zhang, "Sensitivity analysis of lossy coupled transmission lines with nonlinear terminations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, pp. 607–615, April 1994.
- [59] C. Jiao, A. Cangellaris, A. Yaghmour, and J. Prince, "Sensitivity analysis of multiconductor transmission lines and optimization for high-speed interconnect circuit design," *IEEE Transactions on Advanced Packaging*, vol. 23, pp. 132–141, May 2000.
- [60] N. Nakhla, A. Dounavis, M. Nakhla, and R. Achar, "Delay-extraction based sensitivity analysis of multiconductor transmission lines with nonlinear terminations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, pp. 3520 – 3530, Nov. 2005.
- [61] T. Pillage, R. Rohrer, and C. Visweswariah, *Electronic Circuit and System Simulation Methods*. New York: McGraw Hill, 1995.