MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file - Votre référence

Our file - Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

Mixed Frequency/Time Domain Analysis of High-Speed Interconnects

by

Richard Griffith

**A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfilment of the requirements
for the degree of
Master of Engineering**

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

July 13, 1993

© Richard Griffith 1993

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

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

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

ISBN 0-315-89867-4

Canada

Name

Richard Griffith

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

0544

U·M·I

SUBJECT TERM

SUBJECT CODE

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS

Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION

General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523

Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS

Language 0679
 General 0289
 Ancient 0290
 Linguistics 0291
 Modern 0401
Literature 0294
 General 0295
 Classical 0297
 Comparative 0298
 Medieval 0316
 Modern 0591
 African 0305
 American 0352
 Asian 0355
 Canadian (English) 0593
 Canadian (French) 0311
 English 0312
 Germanic 0315
 Latin American 0313
 Middle Eastern 0314
 Romance 0314
 Slavic and East European

PHILOSOPHY, RELIGION AND THEOLOGY

Philosophy 0422
Religion 0318
 General 0321
 Biblical Studies 0319
 Clergy 0320
 History of 0322
 Philosophy of 0469
Theology

SOCIAL SCIENCES

American Studies 0323
Anthropology 0324
 Archaeology 0326
 Cultural 0327
 Physical 0310
Business Administration 0272
 General 0770
 Accounting 0454
 Banking 0338
 Management 0385
Canadian Studies 0501
Economics 0503
 General 0505
 Agricultural 0508
 Commerce-Business 0509
 Finance 0510
 History 0511
 Labor 0358
 Theory 0366
Folklore 0351
Geography 0578
Gerontology 0578
History 0578
 General

Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0337
United States 0337
History of Science 0585
Law 0398
Political Science 0615
 General 0616
 International Law and Relations 0617
 Public Administration 0814
Recreation 0452
Social Work 0426
Sociology 0627
 General 0938
 Criminology and Penology 0631
 Demography 0628
 Ethnic and Racial Studies 0629
 Individual and Family Studies 0630
 Industrial and Labor Relations 0700
 Public and Social Welfare 0344
 Social Structure and Development 0709
 Theory and Methods 0999
Transportation 0453
Urban and Regional Planning 0453
Women's Studies

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture 0473
 General 0285
 Agronomy 0475
 Animal Culture and Nutrition 0476
 Animal Pathology 0359
 Food Science and Technology 0478
 Forestry and Wildlife 0479
 Plant Culture 0480
 Plant Pathology 0817
 Plant Physiology 0777
 Range Management 0746
 Wood Technology

Biology

General 0306
Anatomy 0287
Biostatistics 0308
Botany 0309
Cell 0379
Ecology 0329
Entomology 0353
Genetics 0369
Limnology 0793
Microbiology 0410
Molecular 0307
Neuroscience 0317
Oceanography 0416
Physiology 0433
Radiation 0821
Veterinary Science 0778
Zoology 0472

Biophysics

General 0786
Medical 0760

EARTH SCIENCES

Biogeochemistry 0425
Geochemistry 0996

Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Polynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES

Environmental Sciences 0768
Health Sciences 0566
 General 0300
 Audiology 0992
 Chemotherapy 0567
 Dentistry 0350
 Education 0769
 Hospital Management 0758
 Human Development 0982
 Immunology 0564
 Medicine and Surgery 0347
 Mental Health 0569
 Nursing 0570
 Nutrition 0380
 Obstetrics and Gynecology 0354
 Occupational Health and Therapy 0381
 Ophthalmology 0571
 Pathology 0419
 Pharmacology 0572
 Pharmacy 0382
 Physical Therapy 0573
 Public Health 0574
 Radiology 0575
 Recreation

Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences 0485
Chemistry 0749
 General 0486
 Agricultural 0487
 Analytical 0488
 Biochemistry 0738
 Inorganic 0490
 Nuclear 0491
 Organic 0494
 Pharmaceutical 0495
 Physical 0754
 Polymer 0405
Mathematics 0605
Physics 0986
 General 0606
 Acoustics 0608
 Astronomy and Astrophysics 0748
 Atmospheric Science 0607
 Atomic 0798
 Electronics and Electricity 0759
 Elementary Particles and High Energy 0609
 Fluid and Plasma 0610
 Molecular 0752
 Nuclear 0756
 Optics 0611
 Radiation 0463
 Solid State 0346
Statistics 0984
Applied Sciences 0346
Applied Mechanics 0984
Computer Science

Engineering 0537
 General 0538
 Aerospace 0539
 Agricultural 0540
 Automotive 0541
 Biomedical 0542
 Chemical 0543
 Civil 0544
 Electronics and Electrical 0348
 Heat and Thermodynamics 0545
 Hydraulic 0546
 Industrial 0547
 Marine 0794
 Materials Science 0548
 Mechanical 0743
 Metallurgy 0551
 Mining 0552
 Nuclear 0549
 Packaging 0765
 Petroleum 0554
 Sanitary and Municipal 0790
 System Science 0428
Geotechnology 0796
Operations Research 0795
Plastics Technology 0994
Textile Technology

PSYCHOLOGY

General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

The undersigned recommend to the Faculty of Graduate
Studies and Research acceptance of the thesis

Mixed Frequency/Time Domain Analysis of High-Speed Interconnects

submitted by Richard Griffith
in partial fulfilment of the requirements for
the degree of Master of Engineering.

Thesis Supervisor

Thesis Co-supervisor

Chairman,
Department of Electronics

Ottawa-Carleton Institute for Electrical Engineering,
Department of Electronics
Faculty of Engineering
Carleton University
July 13, 1993

Abstract

A new method is described for transient analysis of nonlinear circuits with frequency dependent parameters. The method does not require numerical integration of differential equations. Applications include delay and crosstalk simulation of high speed VLSI interconnects and simulation of nonlinear microwave circuits with parameters calculated or measured as a function of frequency. The new method is integrated into a CAD framework with optimization and monte-carlo simulation capabilities. The new work includes a parallel processing extension to the CAD framework. Examples and comparisons with previously published techniques are presented.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Contributions	2
1.3	Organization	3
2	High-Speed Interconnect Analysis	5
2.1	Introduction	5
2.2	High-Speed Interconnect	5
2.2.1	Performance Measures Associated with High-Speed Interconnect Analysis	6
2.2.2	Modeling High-Speed Interconnect	7
2.3	High Speed Interconnect Analysis	8
2.3.1	Current Methods	9
2.4	SysCAD: A CAD Framework for Interconnect Analysis	10
2.5	Summary	12
3	Calculation of Linear Circuit Response using Numerical Inversion of the Laplace Transform	14
3.1	Introduction	14
3.2	Numerical Inversion of the Laplace Transform	15
3.2.1	Time Domain Response Evaluation	15

3.2.2	Implementation Considerations and Extensions to NILT . . .	16
3.2.3	Time Domain Superposition	17
3.2.4	Higher Order Formulas	18
3.2.5	Reinitialization of the Analysis	20
3.3	Linear Circuit Analysis using NILT	21
3.3.1	Lumped Component Analysis	21
3.3.2	Distributed Component Analysis	22
3.3.3	Sensitivity	26
3.4	Example Applications of NILT	26
3.4.1	Example 1: Equation Formulation	27
3.4.2	Example 2: Mismatched Circuit	31
3.4.3	Example 3: Searching for a Threshold Crossing	35
3.5	Summary	37
4	Parallel Implementation of Numerical Inversion of the Laplace Transform	39
4.1	Introduction	39
4.2	Partitioning Numerical Inversion of the Laplace Transform for Parallel Implementation	40
4.3	Parallel Implementation	41
4.3.1	Support Routines	41
4.3.2	Standard Interface	43
4.4	Example Framework	43
4.5	Parallelism and Efficiency	46
4.6	Optimization Example	48
4.7	Parallel Performance	52
4.8	Summary	59
5	The Piecewise Decomposition Technique	61

5.1	Introduction	61
5.2	Formulation of the Network Equations	62
5.2.1	Overall Network Equations	62
5.2.2	Separating the Time and Frequency Domain Representations	65
5.3	Solution of the Network Equations	67
5.3.1	Solution using Contraction Mapping	67
5.3.2	The Piecewise Linear Time Domain Representation	68
5.3.3	Solution using Newton-Raphson Iterations	72
5.4	Summary of the Computation Algorithm	75
5.5	Error Analysis	76
5.6	Numerical Examples	76
5.6.1	Example 1: Comparision with HSpice	77
5.6.2	Example 2: Simple Transmission Line Circuit	77
5.6.3	Example 3: Comparision with Linear Analysis	79
5.6.4	Example 4: Transmission Line Network with Multiple Nonlinear Terminations	84
5.7	Summary	88
6	Conclusions	89
6.1	Summary	89
6.2	Additional Research	90
A	Numerical Calculation of the Poles and Residues required for NILT	98

List of Figures

2.1	Process levels in SysCAD and their interconnection.	11
2.2	A variation of Figure 2.1.	11
2.3	Block diagram of the analysis methods available under SysCAD. . . .	13
3.1	A pulse waveform calculated directly by NILT and with improved accuracy using time domain superposition.	19
3.2	Reference signals and directions for a transmission line.	23
3.3	Simple transmission line circuit.	28
3.4	Waveforms for the circuit in Figure 3.3 with a lossless transmission line.	29
3.5	Waveforms for the circuit in Figure 3.3 with a lossy transmission line.	30
3.6	A more complex interconnect circuit.	32
3.7	Response of the interconnect circuit shown in Figure 3.6	33
3.8	Accuracy comparison of different NILT calculations using the response at node 8 of the circuit shown in Figure 3.6	34
3.9	Interconnect circuit for the threshold search example.	35
3.10	Response of the circuit shown in Figure 3.9 showing the threshold crossings.	36
4.1	SysCAD optimization configuration with parallel NILT.	44
4.2	Example circuit for parallel optimization.	50
4.3	Optimization circuit waveform V_a before and after optimization. . . .	53
4.4	Optimization circuit waveform V_b before and after optimization. . . .	54

4.5	Optimization circuit waveform V_c before and after optimization. . . .	55
4.6	Optimization circuit waveform V_d before and after optimization. . . .	56
5.1	Example circuit used to demonstrate formulation of the circuit equations.	64
5.2	Example circuit with nonlinear elements replaced by independent sources.	67
5.3	A piecewise linear waveform.	70
5.4	Excitation of the example circuit with the waveform $h(t)$	72
5.5	A nonlinear circuit with frequency dependent lumped elements. . . .	77
5.6	The voltages across the nonlinear elements of the circuit shown in Figure 5.5.	78
5.7	A circuit with a single transmission line.	79
5.8	Response of the circuit shown in Figure 5.7. A comparison with HSpice results is shown for the case of a lossless transmission line.	80
5.9	A linear circuit with coupled transmission lines.	81
5.10	Response of the circuit shown in Figure 5.9.	83
5.11	A transmission line interconnect network with multiple nonlinear ter- minations.	85
5.12	Response at V_3 of the circuit shown in Figure 5.11.	86

List of Symbols

Boldface is used to indicate a vector or matrix, while italics are used for scalar values. Lower case is used for time domain functions while upper case is used for frequency domain functions. Function domains are lower case. Lower case is used for matrices of integer valued constants while upper case is used for matrices of real valued constants. Lower case is used for indices into sets while upper case is used for limits on the indices.

The notation A' is used to indicate the transpose of the matrix A and A^{-1} indicates the inverse of matrix A .

A_k The slopes of a set of piecewise linear waveforms.

B_k The intercepts of a set of piecewise linear waveforms.

C A capacitance matrix, possibly augmented to MNA form.

C Capacitance per unit length of a transmission line structure.

$f(t)$ Network time domain response to independent inputs.

G An admittance matrix, possibly augmented to MNA form.

G Admittance per unit length of a transmission line structure.

$g(t)$ Network time domain response to a sawtooth pulse.

$h(t)$ Network time domain response to a rectangular pulse.

- L** Inductance per unit length of a transmission line structure.
- M** Order of a numerical inversion of the Laplace transform.
- M'** Computational order of a numerical inversion of the Laplace transform.
- R** Resistance per unit length of a transmission line structure.
- R_r** Piecewise linear response component due to the the waveform values at the current time point.
- $\mathfrak{R}^{m \times n}$** A real valued matrix of size m by n .
- s** Complex valued frequency. The Laplace operator.
- S_r** Piecewise linear response component due to previous time point terms.
- t** Time.
- $u(t)$** A unit step time function.
- $u_g(t)$** A sawtooth pulse.
- $u_h(t)$** A rectangular pulse.
- U** The unity matrix.
- $v(t)$** A time domain signal.
- $\hat{v}(t)$** An approximation to a time domain signal.
- $V(s)$** A vector of frequency domain signal responses.
- $x(t)$** A vector of piecewise linear time domain signal responses.
- $Y(s)$** A frequency domain admittance matrix.

Chapter 1

Introduction

1.1 Motivation

An important problem in the computer-aided design of modern high-speed systems is the time domain analysis of nonlinear circuits with frequency dependent parameters. A primary application is the analysis of interconnections in high speed VLSI chips and printed circuit boards [1]-[5] where improperly designed interconnects can result in an increase of signal delay because of losses, inadvertent switching and noise because of crosstalk, and false switching and ringing due to reflections. These phenomena can be observed at both chip and system levels, and interconnected blocks can be analog, digital or a mix of both. With higher signal speeds the electrical length of interconnects can become a significant fraction of a wavelength. Consequently, the conventional lumped impedance interconnect model is not adequate. Instead, a distributed transmission line model should be used. The difficulty in the analysis in this case is because the quasi-transverse electromagnetic propagation modes in lossy transmission lines are defined in the frequency domain only. In contrast to this, the nonlinear terminations can be in the general case analyzed in the time domain only. Other applications can be found in the area of nonlinear microwave circuits [6, 7], where it is common for these circuits to have components whose models include loss,

dispersion or coupling parameters calculated or measured as a function of frequency.

An intuitive solution to the problem of delay and reflections is to decrease the length of the interconnections by increasing the system density. The trend toward greater density, however, fosters another problem, that of crosstalk between adjacent lines. A designer must make proper trade-offs between various conflicting factors. For large circuits the relation between circuit parameters and design criteria becomes extremely complicated. There is an increasing need for a design optimization approach that uses interconnect parameters explicitly in overall network design. A prerequisite for circuit optimization is circuit analysis which for modern high-speed circuits must provide for both the frequency domain analysis of transmission line structures and the time domain analysis of nonlinear elements.

This thesis is directed at the analysis of circuits with both nonlinear elements and distributed interconnect. Several new techniques are developed.

1.2 Contributions

Specific contributions applicable to the analysis of high-speed interconnect presented in this thesis are:

1. A method for the time domain analysis of linear circuits with distributed components using numerical inversion of the Laplace transform is developed. This method is used to simulate high-speed interconnect circuits where interconnects are modelled as lossy coupled transmission lines. (Section 3.3.2).
2. The notion of time domain superposition is introduced as a method for increasing the accuracy of numerical inversion of the Laplace transform. This increased accuracy applies to piecewise linear time domain waveforms and linear functions of the piecewise linear waveforms. (Section 3.2.3).
3. The time point independence property of numerical inversion of the Laplace transform is used to demonstrate an efficient method for the evaluation of circuit

delays and other time domain performance criteria. (Section 3.4.3).

4. Time domain analysis using numerical inversion of the Laplace transform was integrated into the SysCAD high-speed interconnect analysis system. This integration provides optimization and statistical analysis of high-speed interconnect circuits based on evaluations performed by the time domain analysis method. (Section 4.4).
5. Time domain analysis using numerical inversion of the Laplace transform was implemented in parallel on a network of workstations. This implementation was integrated into SysCAD and increased performance of the overall interconnect analysis system. (Section 4.3).
6. The parallel implementation of numerical inversion of the Laplace transform is used to reduce the optimization of an example interconnect circuit from a period of eight hours to less than one hour. (Section 4.7).
7. A decomposition method for the analysis of circuits with nonlinear components. Methods such as this are essential for mixed frequency/time domain analysis. This new method does not require iterated solutions to the linear part of the circuit, reducing the computational requirements when compared to other methods. (Chapter 5).

1.3 Organization

Chapter 2 introduces the problems associated with interconnect in high-speed circuits. Current computational methods for evaluating and reducing these problems are briefly reviewed. SysCAD a computer aided design framework for interconnecting various tools for interconnect analysis is introduced. The analysis methods developed later are introduced as functional blocks of the overall SysCAD system.

Chapter 3 presents numerical inversion of the Laplace transform as a method for analysis of high-speed interconnect circuits. The inversion method is reviewed along with some recent methods for increasing accuracy. The superposition method for increasing accuracy when piecewise linear waveforms are present is described. A transmission line, or distributed element, model for interconnect is developed and incorporated into frequency domain modified nodal analysis. Numerical inversion of the Laplace transform is then used for time domain analysis of circuits including the transmission line element. A number of examples are presented demonstrating advantages of the new method.

Chapter 4 describes a parallel implementation of numerical inversion of the Laplace transform. A brief discussion of the time point independence of the inversion that allows the parallelism is followed by some details of the implementation. A look at the efficiency of the implementation is followed by an example that illustrates the performance of the method.

Chapter 5 presents a piecewise linear decomposition technique for the analysis of nonlinear circuits. As the theory of the technique is developed an example is used to illustrate the steps of decomposition, equation formulation and application of the piecewise linear waveforms. Simultaneous solution of the linear and nonlinear equations of the system is compared to commonly used relaxation methods. Examples are presented to demonstrate the performance of the method.

Chapter 6 contains a summary of the thesis and suggestions for future research.

Chapter 2

High-Speed Interconnect Analysis

2.1 Introduction

This chapter discusses some of the aspects of high-speed interconnect analysis providing a setting for the work presented later. A definition is given for the term “high-speed interconnect” and the main properties of interconnects to which this term applies are presented. Interconnect design problems related to the properties of high-speed interconnects are discussed and some of the methods for analyzing and solving these problems are presented. A CAD framework which integrates a number of these analysis methods into a single system is described. The work presented later in this thesis details new methods added into the framework as well as providing significant enhancements to the framework itself.

2.2 High-Speed Interconnect

High-speed interconnect analysis refers to the analysis of circuits where the interconnect between discrete circuit components cannot be considered as an ideal conductive medium but must instead be considered as a circuit component that influences circuit performance. This is a contrast to standard analog or digital circuit design where

the interconnect between components is ignored and circuit performance is considered only as a function of the discrete components. Effectively the interconnect in standard analog or digital design is considered to be an ideal conductive media.

The difference between the ideal performance and actual performance is very important in circuit design. It is possible for interconnect effects to stop a circuit from meeting performance criteria. For this reason it is important to give circuit designers information about circuit performance including the effects of interconnect.

The non-ideal characteristics of interconnect typically increase with increases in circuit speeds. High-speed interconnect analysis is required whenever circuit signal speeds are such that the interconnect medium has significant impact on the circuit performance and can not be considered ideal. Any interconnect used in an electrical system may meet this condition if the signal speed is high enough relative to the electrical characteristics of the interconnect. The term "high-speed" is then relative to the electrical characteristics of interconnect when compared to the signal or functional speed of the circuit.

2.2.1 Performance Measures Associated with High-Speed Interconnect Analysis

Under high-speed interconnect conditions the circuit interconnect will cause unwanted signal degradation. High-speed interconnect analysis is required to correctly model the effects of the interconnect on the circuit signals so that the circuit designer can make allowances for the signal degradation. Circuit phenomena related to high-speed interconnect include

1. signal delay,
2. slew rate reductions,
3. signal ringing,

4. crosstalk.

Signal delay is a direct function of the distance a signal has to travel. With physical separation between discrete components the delay introduced by interconnect will always be present. The propagation velocity of the signal is a function of the characteristics of the interconnect. Changes to the interconnect can increase or decrease the propagation velocity, however the speed of light cannot be exceeded.

Slew rate reductions are due to the fact that the charge transported through the interconnect experiences some resistance. As a function of this resistance the charge transport rate is reduced from the ideal situation.

Signal ringing is a function of both the signal delay and mismatches between the impedances at either end of an interconnect segment and the interconnect itself. The mismatches in impedance cause some energy from a signal propagating along the interconnect to be reflected back into the interconnect instead of exiting the ends. This reflected energy bounces back and forth on the interconnect until it either exits the line or is absorbed in losses. This ringing results in the dispersion of the signal energy over time, distorting the signal shape.

Crosstalk results when signals from one section of interconnect are coupled into another section of interconnect. This functions as an extra signal path where the signals from one conductor appear as interference in the signals of the other conductor.

2.2.2 Modeling High-Speed Interconnect

High-speed interconnect analysis requires modeling the electrical characteristics of the circuit interconnect. The complexity of this model depends on the relative “speed” of the circuit signals to the characteristics of the interconnect. Model complexity ranges from lumped element models at relatively slow speeds[8] to full electro-magnetic field

models at very high speeds[9]. Full electro-magnetic modeling is very computationally expensive and is usually done on single interconnect structures rather than on entire circuits.

Simple yet robust interconnect models that can be used in full circuit analysis are required. A common approach is to use distributed transmission line models parameterized over a particular physical configuration. A more general approach is to model the interconnect using tabulated impedance data such as S-parameters. Both methods are available in Microwave Spice[10].

The tabulated impedance data approach, such as that available in Microwave spice, is very general and can be applied to structures analyzed using full electro-magnetic modeling or physical structures measured in a lab. This table approach cannot be fully parameterized over the physical parameters of the system without many analyses or measurements. For example data would not be available for the performance of the interconnect for small changes in the physical parameters without actual mathematical analyses being performed. In contrast to this a formula based transmission line model can be parameterized over variations in physical parameters or fitted to tabulated data. This allows circuit analysis to be performed directly from circuit layout information where these physical parameters are established.

The transmission line model or other form of parameterization based on the physical parameters is preferred because of the direct association with the physical parameters. Analysis can then easily be performed for variation in the physical parameters and the circuit designer will then have results associated directly with parameters that can be manipulated in the design.

2.3 High Speed Interconnect Analysis

Even with the appropriate interconnect models high-speed interconnect analysis is not a simple task. The most compact and efficient interconnect models are in the frequency domain while the descriptions of nonlinear circuit terminations such as

diodes and transistors are in the time domain. A method for simultaneously solving a system of time and frequency domain equations is required.

2.3.1 Current Methods

In [11], a piecewise harmonic balance technique was presented for the steady-state analysis of nonlinear circuits. The method can be applied to the analysis of nonlinear circuits with frequency dependent parameters [12]. However, it is limited to the steady-state case with periodic or quasi-periodic [13] inputs and cannot be extended to transient analysis.

In the harmonic balance technique the circuit waveforms are iterated between the time and frequency domain using an FFT and relaxation techniques to converge to a solution. The use of the FFT to convert between the two domains imposes a periodic requirement on the signals being considered and is not a fully general transient analysis method. Multiple uses of the FFT and multiple solutions to the circuit equations in both the time and frequency domain are required.

Several methods have been proposed for the analysis of circuits which contain coupled transmission lines[14, 15, 16, 17, 18, 2]. In most cases, the lossy transmission line has been analyzed either by using a series of lumped models in the time domain [17], or by using the Fourier transformation [18, 2]. Recently [5], a time domain model has been proposed where coupling effects are assumed to be significant only between immediately adjacent lines, and neighboring lines are assumed to be separated with equal space. In addition, this model can not handle the case where the line parameters are frequency dependent.

An overview of techniques for the simultaneous solving of frequency domain and time domain equations is given by Rissoli and Neri[6]. A historical review of circuit simulators is presented by Pederson[19]. With current trends in increased circuit speed and increased circuit density there are great pressures on simulators to incorporate more elaborate element models such as transmission lines while simultaneously

allowing for larger and more complex circuits.

As a result of these pressures new techniques such as asymptotic waveform evaluation[20] and the methods presented here are being developed. The analysis of nonlinear circuits using asymptotic waveform evaluation is addressed in [21].

2.4 SysCAD: A CAD Framework for Interconnect Analysis

SysCAD is a framework for interconnecting various computer aided design and analysis tools. The framework and associated tools are under development at Carleton University. SysCAD is based on concepts present in OSA90[22]. As a framework SysCAD is a standard programming interface and design methodology allowing independently developed analysis methods to be combined in a hierarchical system in an arbitrary manner.

SysCAD is organized around independent functions implemented as separate computer processes. These processes use other independent functions through a standard interface mechanism. Three distinct process types can be identified in the SysCAD framework. These three process types are:

1. Top level processes which provide no functionality to higher levels;
2. Mid level processes which provide functionality to top or mid level processes and in turn require functionality from lower level processes;
3. Bottom level process which provide functionality to top or mid level processes but do not require functionality from any other blocks.

As shown in Figure 2.1 the three process types can be interconnected like building blocks to form a large system. The logical and hierarchical connection of the modules is performed at execution time providing a flexible system. Interchangeable optimizers, simulators, and other process modules have been developed.

Figure 2.1: Process levels in SysCAD and their interconnection.

Figure 2.2: A variation of Figure 2.1.

Figure 2.2 shows a variation of the system in Figure 2.1. In both figures the optimizer manipulates a set of variables x relative to a set of goals y . The specification process passes the variables to the appropriate analysis program and evaluates the goals for the optimizer. Each interface level is a functional interface a set of arguments x are passed to the lower level function which returns a result y . This interface is common to all process in SysCAD and allows the processes to be interconnected in an arbitrary manner.

Figure 2.3 shows the current functionality of the SysCAD processes for interconnect analysis. The shaded boxes mark the functions addressed by this thesis. The optimization and sweep functions are top level processes. The analysis methods are bottom level processes. The evaluation process is a mid level process. The functions addressed in this thesis are

1. Numerical inversion of the Laplace transform (NILT) analysis of linear circuits at arbitrary time points.
2. A piecewise decomposition method for the analysis of nonlinear circuits with linear subcircuits.
3. Parallel evaluation of analysis functions.

These functions are detailed in chapters 3, 5, and 4 respectively.

2.5 Summary

This chapter has reviewed the high-speed interconnect analysis problem. The SysCAD framework was briefly described. The analysis techniques developed later in this thesis have been introduced as parts of an interconnect analysis system for SysCAD.

Figure 2.3: Block diagram of the analysis methods available under SysCAD.

Chapter 3

Calculation of Linear Circuit Response using Numerical Inversion of the Laplace Transform

3.1 Introduction

This chapter describes how numerical inversion of the Laplace transform (NILT) can be used to calculate the time domain response of linear circuits with arbitrary frequency dependent elements. The chapter starts by presenting a formula for the numerical inversion of the Laplace transform. The basic properties of the inversion method are reviewed. The application of this inversion to circuit analysis is discussed. The major contribution presented in this chapter is the extension of the method to generalized distributed elements such as transmission lines. Additional extensions to the analysis method such as sensitivity calculations and higher order formulas with more accuracy are also reviewed. Several examples are presented to demonstrate the method.

3.2 Numerical Inversion of the Laplace Transform

NILT is a method for calculating an approximate time domain waveform corresponding to a given complex frequency domain function. Time domain waveform points are calculated directly from evaluation of the frequency domain function. The derivation of the method presented here follows that of [23].

3.2.1 Time Domain Response Evaluation

Given a complex frequency domain function $V(s)$ the corresponding time domain waveform, $v(t)$, is obtained using the inverse Laplace transformation as

$$v(t) = \frac{1}{2\pi j} \int_{c-j\infty}^{c+j\infty} V(s) e^{st} ds, \quad (3.1)$$

where c is an arbitrary positive constant such that $\text{Re}\{s_i\} < c$ where s_i are the poles of $V(s)$. Using the substitution $z = st$ and approximating e^z by the predetermined Padé rational function

$$R_{N,M}(z) = \frac{P_N(z)}{Q_M(z)} = \frac{\sum_{i=0}^N (M+N-i)! \binom{N}{i} z^i}{\sum_{i=0}^M (-1)^i (M+N-i)! \binom{M}{i} z^i} \approx e^z, \quad (3.2)$$

where M and N determine the order of the Padé approximation $R_{N,M}(z)$ the inversion integral (3.1) becomes

$$\hat{v}(t) = \frac{1}{2\pi j} \int_{c-j\infty}^{c+j\infty} V(z/t) R_{N,M}(z) \frac{dz}{t} \quad (3.3)$$

where $\hat{v}(t) \approx v(t)$ due to the replacement of e^z by the rational approximation. Taking M, N such that

$$F(z) = V(z/t) R_{N,M}(z) \quad (3.4)$$

has at least two more finite poles than zeros the integral can be performed in the right half plane about the roots of the exponential approximation $R_{N,M}(z)$. Completing

this integration yields

$$\hat{v}(t) = \frac{-1}{t} \sum_{i=1}^M k_i V(z_i/t) \quad (3.5)$$

where k_i and z_i are the poles and residues of $R_{N,M}(z)$. When $N < M$ the poles and residues may be expressed using

$$R_{N,M}(z) = \sum_{i=1}^M \frac{k_i}{z - z_i} . \quad (3.6)$$

The calculation of k_i and z_i given M and N is presented in Appendix A. Typically M and N are predetermined based on the desired application, thus k_i and z_i can be calculated ahead of time and stored. Tables of k_i and z_i for various values of M and N can be found in [24].

NILT is defined by (3.5) where $\hat{v}(t) \approx v(t)$ is the calculated time domain equivalent to the complex frequency domain function $V(s)$. M evaluations of $V(s)$ are required to calculate each $\hat{v}(t)$.

3.2.2 Implementation Considerations and Extensions to NILT

For a real time function $v(t)$ the conjugate poles z_i will contribute conjugate components to the sum of (3.5). With M even (3.5) can then be shortened to

$$\hat{v}(t) = \frac{-1}{t} \sum_{i=1}^{M'} k'_i V(z'_i/t) \quad (3.7)$$

where $M' = M/2$, $k'_i = 2k_i$, and z'_i are the poles in the upper half plane. When M is odd $M' = (M + 1)/2$ and $k'_i = k_i$ for the residue corresponding to the real pole. This shortened form is used in all implementations, however all references to NILT will still refer to (3.5).

In addition several general properties of (3.5) should be commented on

1. Exactly $M + N + 1$ terms of the Taylor series expansion of $v(t)$ are preserved in the calculation of $\hat{v}(t)$ [23]. Thus $\hat{v}(t)$ is a good approximation to $v(t)$ over the portion of $v(t)$ approximated by these initial terms of its Taylor expansion.

An expression for the truncation error was derived in [25]. The truncation error rapidly decreases with increasing M , and N . The method is equivalent to a numerical integration technique with order $M + N - 2$, however numerical inversion of the Laplace transform is always stable.

2. The function $V(s) = s^{-m}$ is inverted exactly for $N + 2 - M \leq m \leq M + N + 1$.
In this case

$$\hat{v}(t) = v(t) = \frac{t^{m-1}}{(m-1)!} \quad (3.8)$$

3. The presence of the term z_i/t in (3.5) makes it difficult to evaluate $\hat{v}(t)$ for $t = 0$. Limit methods yield the correct result however these are difficult to evaluate on a computer. A more practical method is to evaluate the function at as small a value of t as possible.

From these properties it can be determined that the realization of a function $v(t)$ is a function of the values used for M and N . Selecting $N = M - 2$ will cover all realizable circuit functions, including impulsive functions, where $V(s) = \sum a_m s^{-m}$, $m \geq 0$. When $V(s)$ includes the derivative of an impulse, $m = -1$, selecting $N = M - 3$ yields correct results for $t > 0$.

Numerical value ranges limit typical computer applications to $M < 28$. The values $M = 10$ and $M = 22$ have been used extensively using values of k_i and z_i published in tables[24].

As an approximation method there are some cases where the accuracy of NILT can be improved. Two methods for increasing the accuracy of NILT are described.

3.2.3 Time Domain Superposition

Careful implementation of NILT can lead to improved accuracy when the properties of the method are taken into account. Consider the waveforms shown in Figure 3.1. The dashed waveform is calculated using NILT with $M = 22$ and $N = 20$ from the

frequency domain description

$$V_e(s) = \frac{10}{s^2} (1 - e^{-.1s} - e^{-2s} + e^{-2.1s}) . \quad (3.9)$$

of a time domain pulse with finite rise and fall times. The error in the inversion, due to the incomplete approximation of e^{-As} is visible in the figure. The correct time domain waveform also shown in the figure was calculated using time domain subtraction of the inverse of the ramp function

$$V'_e(s) = \frac{10}{s^2} \quad (3.10)$$

which is exactly inverted by NILT. Combination in the time domain yields

$$v_e(t) = v'_e(t) - v'_e(t - 0.1) - v'_e(t - 2) + v'_e(t - 2.1). \quad (3.11)$$

which is exactly the time domain equivalent of the frequency domain function $V_e(s)$.

As illustrated in Figure 3.1 time domain decomposition of a signal and reconstruction after inversion can greatly increase the accuracy of the method and is to be preferred in all calculations. This decomposition and reconstruction, a form of superposition, is a valid method because this is a linear time invariant system.

In circuit analysis the input waveform is typically composed of a series of ramps and steps. The circuit response is calculated for a single step and a single ramp. The responses are then combined according to the composition of the input using superposition. Superposition increases accuracy by reducing the complexity of the waveform calculated by NILT.

3.2.4 Higher Order Formulas

More accurate forms of the NILT equation (3.5) have recently been developed[26]. These new forms use the frequency domain derivative $\partial V(s)/\partial s$ of the frequency domain functions $V(s)$ to provide increased accuracy.

Figure 3.1: A pulse waveform calculated directly by NILT and with improved accuracy using time domain superposition.

The original NILT equation (3.5) is denoted the order $p = 0$ NILT, where p is the order of derivatives used. The NILT expression for $p = 1$ is

$$v(t) = \frac{-2}{t} \sum_{i=1}^M \left(\frac{k_i^2}{t} \frac{\partial V(2z_i/t)}{\partial s} + 2k_i A_i V(2z_i/t) \right) \quad (3.12)$$

where

$$A_i = \sum_{j=1}^M \frac{k_j}{z_i - z_j}, \quad j \neq i.$$

The NILT expression based on second order derivatives, $p = 2$ is

$$v(t) = \frac{-9}{t} \sum_{i=1}^M \left(\frac{3k_i^3}{8t^2} \frac{\partial^2 V(3z_i/t)}{\partial s^2} + \frac{3k_i^2}{2t} A_i \frac{\partial V(3z_i/t)}{\partial s} + D_i V(3z_i/t) \right) \quad (3.13)$$

where

$$\begin{aligned} D_i &= C_i k_i - k_i^2 B_i / 2 \\ B_i &= \sum_{j=1}^M \frac{L_j}{(z_i - z_j)^2}, \quad j \neq i \\ C_i &= \sum_{j=1}^M \sum_{v=1}^{M'} \frac{k_j k_v}{(z_i - z_j)(z_i - z_v)}, \quad j \neq i, v \neq i. \end{aligned}$$

These formulas may be used when increased accuracy is desired. The calculation of the required derivatives may be nontrivial and require significant extra computation. In addition the numerical size of derivatives may introduce additional numerical problems to the method.

3.2.5 Reinitialization of the Analysis

The accuracy of NILT may also be extended by reinitializing the system and shifting the time reference[27]. This method works because NILT is more accurate close to the time reference point. Stepping forward and reinitializing the system approximates forward integration. Recently[28] the reinitialization method has been extended to distributed circuit elements.

3.3 Linear Circuit Analysis using NILT

The equations describing a linear circuit can be written in the Laplace domain. Solution to the circuit equations in the Laplace domain produces a Laplace domain function describing the response of the circuit. NILT may then be applied to directly calculate the time domain response. In addition to time domain responses NILT can be used to calculate time domain sensitivities directly from the frequency domain sensitivities.

3.3.1 Lumped Component Analysis

Using modified nodal analysis[24] a linear network may be represented in the Laplace domain as

$$\mathbf{Y}(s)\mathbf{V}(s) = \mathbf{I}(s) \quad (3.14)$$

where $\mathbf{Y}(s)$ is a matrix of admittance parameters, $\mathbf{I}(s)$ is a vector of input sources, and $\mathbf{V}(s)$ is network response vector. The admittance matrix is formed as

$$\mathbf{Y}(s) = \mathbf{G} + s\mathbf{C} \quad (3.15)$$

where $\mathbf{G}$ describes the structure and passive elements of the circuit, and $\mathbf{C}$ describes the energy storing elements of the circuit. Solution of (3.14) yields

$$\mathbf{V}(s) = \mathbf{Y}^{-1}(s)\mathbf{I}(s) \quad (3.16)$$

which can be directly substituted into (3.5). The time domain circuit response is then

$$\hat{\mathbf{v}}(t) = \frac{-1}{t} \sum_{i=1}^M k_i \mathbf{Y}^{-1}(z_i/t) \mathbf{I}(z_i/t) . \quad (3.17)$$

If higher order formulas are to be used the required derivatives can be calculated by differentiating (3.14) and solving for $\mathbf{V}(s)$. Derivatives of $\mathbf{Y}(s)$ will be required and can be obtained from (3.15) as

$$\frac{\partial \mathbf{Y}(s)}{\partial s} = \mathbf{C} \quad (3.18)$$

and

$$\frac{\partial^2 Y(s)}{\partial s^2} = 0 \quad (3.19)$$

Lumped frequency dependent components are handled directly using their admittance matrix $Y_i(s)$.

3.3.2 Distributed Component Analysis

Distributed components may be directly included in (3.15) as

$$Y(s) = G + sC + \sum_i Y_i(s) \quad (3.20)$$

where $Y_i(s)$ is the admittance representation of the i th distributed or complex frequency dependent element. For elements such as frequency dependent resistances the admittance matrix is described by the element model. More complex element models have more involved admittance representations. The admittance matrix, $Y_i(s)$, for a transmission line operating in quasi-TEM mode is derived here. This derivation was published as [29].

Assume a transmission line uniform along its length with an arbitrary cross section. The reference directions for the voltage and current signals on a transmission line are shown in Figure 3.2. The cross section, with N signal conductors and a reference, can be represented by the following $N \times N$ matrices of line parameters: the inductance per unit length L , the resistance per unit length R , the capacitance per unit length C , and the conductance per unit length G .

Let x represent the length along the transmission line, with $x = 0$ corresponding to the near end, and $x = D$ corresponding to the far end as shown in Figure 3.2. In the frequency domain the transmission line can be described by the differential equations

$$\frac{\partial V(x)}{\partial x} = -Z_p I(x), \quad \frac{\partial I(x)}{\partial x} = -Y_p V(x) \quad (3.21)$$

where the vectors $V(x)$, $I(x)$ are the line voltages and currents,

$$Z_p = R + sL, \quad Y_p = G + sC \quad (3.22)$$

Figure 3.2: Reference signals and directions for a transmission line.

and s is the Laplace operator. The differential equations can be combined into the single wave equation

$$\frac{\partial^2 \mathbf{V}(x)}{\partial x^2} = \mathbf{Z}_p \mathbf{Y}_p \mathbf{V}(x) \quad (3.23)$$

which will have solutions of the form

$$\mathbf{V}_m(x) = \mathbf{V}_m(0) \exp(\pm \gamma_m x) \quad (3.24)$$

$$\mathbf{I}_m(x) = \mathbf{I}_m(0) \exp(\pm \gamma_m x) \quad (3.25)$$

where γ_m describes the wave propagating along the conductors with the amplitude and phase on a particular conductor the set by the the complex vectors $\mathbf{V}_m(0)$ and $\mathbf{I}_m(0)$.

Substituting the solution form into the wave equation (3.23) yields

$$\{\gamma_m^2 \mathbf{U} - \mathbf{Z}_p \mathbf{Y}_p\} \mathbf{V}_m(0) = 0 \quad (3.26)$$

where $\mathbf{U}$ is an identity matrix. This equation has nontrivial solutions when γ_m^2 satisfies the eigenvalue problem

$$\det \{\gamma_m^2 \mathbf{U} - \mathbf{Z}_p \mathbf{Y}_p\} = 0 \quad (3.27)$$

There will be N solutions γ_m^2 , $m = 1, N$ to the eigenvalue problem with associated eigenvectors S_m , $m = 1, N$. Let Γ be a diagonal matrix $\{\gamma_1, \gamma_2, \dots, \gamma_N\}$ obtained by taking the square roots of the eigenvalues, and S_v be a matrix with the eigenvectors S_m in the columns. Solutions for the voltages can now be written as

$$V(x) = S_v E(x) C_1 + S_v E(x)^{-1} C_2 \quad (3.28)$$

where $E(x)$ is the diagonal matrix $\{\exp(-\gamma_1 x), \exp(-\gamma_2 x), \dots, \exp(-\gamma_N x)\}$, C_1 and C_2 constant vectors. By substituting $V(x)$ into (3.21) the currents are obtained as

$$I(x) = S_i E(x) C_1 - S_i E(x)^{-1} C_2 \quad (3.29)$$

where $S_i = Z_p^{-1} S_v \Gamma$.

From (3.28) the relationship between the near and far end voltages can be written as

$$\begin{bmatrix} V(0) \\ V(D) \end{bmatrix} = \begin{bmatrix} S_v & S_v \\ S_v E(D) & S_v E(D)^{-1} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \quad (3.30)$$

Similarly, from (3.29) the relationship between the near and far end currents can be written as

$$\begin{bmatrix} I(0) \\ I(D) \end{bmatrix} = \begin{bmatrix} S_i & -S_i \\ S_i E(D) & -S_i E(D)^{-1} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \quad (3.31)$$

Solving (3.30) for the constants and substituting into (3.31) gives

$$\begin{bmatrix} I(0) \\ I(D) \end{bmatrix} = \begin{bmatrix} S_i & -S_i \\ S_i E(D) & -S_i E(D)^{-1} \end{bmatrix} \times \begin{bmatrix} S_v & S_v \\ S_v E(D) & S_v E(D)^{-1} \end{bmatrix}^{-1} \begin{bmatrix} V(0) \\ V(D) \end{bmatrix} \quad (3.32)$$

This equation now represents the transmission line in admittance form $Y(s)V(s) = I(s)$ where

$$I = \begin{bmatrix} I(0) \\ -I(D) \end{bmatrix} = \begin{bmatrix} I_{near} \\ I_{far} \end{bmatrix} \quad (3.33)$$

is a vector of the terminal currents into the transmission line,

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}(0) \\ \mathbf{V}(D) \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{near} \\ \mathbf{V}_{far} \end{bmatrix} \quad (3.34)$$

is a vector of the terminal voltages of the transmission line, and

$$\mathbf{Y}_i = \begin{bmatrix} \mathbf{S}_i & -\mathbf{S}_i \\ -\mathbf{S}_i \mathbf{E}(D) & \mathbf{S}_i \mathbf{E}(D)^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{S}_v & \mathbf{S}_v \\ \mathbf{S}_v \mathbf{E}(D) & \mathbf{S}_v \mathbf{E}(D)^{-1} \end{bmatrix}^{-1} \quad (3.35)$$

$$= \begin{bmatrix} \mathbf{S}_i \mathbf{E}_1 \mathbf{S}_v^{-1} & \mathbf{S}_i \mathbf{E}_2 \mathbf{S}_v^{-1} \\ \mathbf{S}_i \mathbf{E}_2 \mathbf{S}_v^{-1} & \mathbf{S}_i \mathbf{E}_1 \mathbf{S}_v^{-1} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_{i1} & \mathbf{Y}_{i2} \\ \mathbf{Y}_{i2} & \mathbf{Y}_{i1} \end{bmatrix} \quad (3.36)$$

is an admittance representation for the transmission line, where $\mathbf{E}_1$ and $\mathbf{E}_2$ are diagonal matrices

$$\mathbf{E}_1 = \text{diagonal} \left\{ \frac{\exp(-2\gamma_m D) + 1}{1 - \exp(-2\gamma_m D)}, m = 1, N \right\} \quad (3.37)$$

$$\mathbf{E}_2 = \text{diagonal} \left\{ \frac{2}{\exp(-\gamma_m D) - \exp(\gamma_m D)}, m = 1, N \right\} \quad (3.38)$$

The matrix $\mathbf{Y}_i$ is a function of the eigenvalues γ_m^2 and the eigenvectors, $\mathbf{S}_m$ of (3.27). These eigenvalues and eigenvectors will be a function of the transmission line parameters $\mathbf{R}, \mathbf{L}, \mathbf{G}, \mathbf{C}$ and the complex frequency s . Note that the transmission line parameters $\mathbf{R}, \mathbf{L}, \mathbf{G}, \mathbf{C}$ are often themselves a function of the complex frequency.

Note that calculation of the eigenvalues and eigenvectors requires finding the roots of the determinant of (3.27). For large coupled line systems these roots cannot be calculated directly and root searching techniques must be used. These search techniques greatly increase the computational cost of evaluating the admittance matrix.

Higher orders of NILT are much more difficult to evaluate when distributed elements are involved as the required derivatives are now

$$\frac{\partial \mathbf{Y}(s)}{\partial s} = \mathbf{C} + \sum_i \frac{\partial \mathbf{Y}_i(s)}{\partial s}. \quad (3.39)$$

$$\frac{\partial^2 \mathbf{Y}(s)}{\partial s^2} = \sum_i \frac{\partial^2 \mathbf{Y}_i(s)}{\partial s^2}. \quad (3.40)$$

3.3.3 Sensitivity

The adjoint sensitivity method[30] has been used with NILT to compute sensitivities for use in optimization of system performance over a circuit parameter. Adjoint methods may be used when the sensitivity of the full vector $\mathbf{v}(t)$, from (3.5), is not required. Let the frequency domain response of the required signals be

$$\mathbf{V}_o(s) = \mathbf{u}^T \mathbf{V}(s) \quad (3.41)$$

where $\mathbf{u}$ is a selector matrix, $u_{ij} \in \{0, 1\}$ with one nonzero entry per row. By the adjoint sensitivity method the desired sensitivities with respect to a circuit parameter x are obtained in the frequency domain from (3.14) using (3.41) as

$$\frac{\partial \mathbf{V}_o}{\partial x} = (\mathbf{V}^*)^T \left[\frac{\partial \mathbf{Y}}{\partial x} \mathbf{V} - \frac{\partial \mathbf{I}}{\partial x} \right] \quad (3.42)$$

where $\mathbf{V}^*$ is solved from the adjoint equation

$$\mathbf{Y}^T \mathbf{V}^* = -\mathbf{u}. \quad (3.43)$$

Note that for a circuit parameter x , it is normally the case that $\partial \mathbf{I} / \partial x = 0$. The sensitivity of the time domain response $\mathbf{v}(t)$ with respect to x is obtained from the frequency domain sensitivity $\partial \mathbf{V}(s) / \partial x$ as

$$\frac{\partial \mathbf{v}_o(t)}{\partial x} = \frac{-1}{t} \sum_{i=1}^M k_i \frac{\partial \mathbf{V}_o(z_i/t)}{\partial x} \quad (3.44)$$

with all values as described previously. The required derivatives are detailed in [31].

Note that sensitivity calculations may also be performed with the higher order, $p > 0$, NILT expressions described in the previous section. Additional derivatives are required when this is done.

3.4 Example Applications of NILT

Three examples are presented. The first example illustrates the application of the method and provides a point of comparison with results published for another method.

The second example applies the method to a more complex circuit illustrating that the full circuit response need not be calculated. The third example uses time point calculations to find where circuit responses cross a threshold.

The analysis program uses superposition. Higher order formulas and reinitialization are not used. Reinitialization of transmission lines removes the time point independence property of further applications of NILT.

3.4.1 Example 1: Equation Formulation

Consider the circuit of Figure 3.3. A Laplace domain modified nodal analysis representation for this circuit in the form $\mathbf{YV} = \mathbf{I}$, where $\mathbf{Y}$ is given by (3.20) is

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & \frac{1}{50} & -\frac{1}{50} & 0 & 0 & 0 \\ 0 & -\frac{1}{50} & (\frac{1}{50} + Y_{11}) & Y_{12} & Y_{13} & Y_{14} \\ 0 & 0 & Y_{21} & (\frac{1}{100} + Y_{22}) & Y_{23} & Y_{24} \\ 0 & 0 & Y_{31} & Y_{32} & (\frac{1}{100} + Y_{33}) & Y_{34} \\ 0 & 0 & Y_{41} & Y_{42} & Y_{43} & (\frac{1}{100} + Y_{44}) \end{bmatrix} \begin{bmatrix} I_E \\ V_1 \\ V_2 \\ V_4 \\ V_3 \\ V_5 \end{bmatrix} = \begin{bmatrix} V_E(s) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.45)$$

where the circuit input is $V_E(s)$ and the admittance matrix for the two conductor transmission line is

$$\mathbf{Y}_i = \begin{bmatrix} \mathbf{Y}_{i1} & \mathbf{Y}_{i2} \\ \mathbf{Y}_{i2} & \mathbf{Y}_{i1} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \\ Y_{31} & Y_{32} \\ Y_{41} & Y_{42} \end{bmatrix} & \begin{bmatrix} Y_{13} & Y_{14} \\ Y_{23} & Y_{24} \\ Y_{33} & Y_{34} \\ Y_{43} & Y_{44} \end{bmatrix} \end{bmatrix}, \quad (3.46)$$

as described in the previous section. The circuit response is calculated by solving (3.45) for different values of s . The response is the vector $[I_E V_1 V_2 V_4 V_3 V_5]^T$. The frequency response is obtained with $s = j\omega$. The time domain response is obtained directly using numerical inversion of the Laplace transform as given by (3.5).

Figure 3.3: Simple transmission line circuit.

This circuit was presented and analyzed in [32]. The transmission line is 0.3048 meters long and is described by the parameters

$$\mathbf{L} = \begin{bmatrix} 494.6 & 63.3 \\ 63.3 & 494.6 \end{bmatrix} nH/m \quad (3.17)$$

$$\mathbf{C} = \begin{bmatrix} 62.8 & -4.9 \\ -4.9 & 62.8 \end{bmatrix} pF/m \quad (3.48)$$

The applied voltage is a 1 volt, 6 nanosecond pulse with 1.5 nanosecond rise and fall times. Numerical inversion of the Laplace transform with $M' = 11$ was used to calculate the circuit response, superposition was used to increase the accuracy. Waveforms for the lossless case where $\mathbf{R} = 0$ and $\mathbf{G} = 0$ are shown in Figure 3.4.

For the lossy case the line resistance was assumed to vary in proportion to the square root of the frequency, while the conductance matrix was assumed to be independent of frequency. The resistance matrix at 1 MHz and the conductance matrix are given by

$$\mathbf{R} = \begin{bmatrix} 0.1 & 0.02 \\ 0.02 & 0.1 \end{bmatrix} \Omega/m \quad (3.49)$$

$$\mathbf{G} = \begin{bmatrix} 0.1 & -0.01 \\ -0.01 & 0.1 \end{bmatrix} S/m \quad (3.50)$$

Figure 3.4: Waveforms for the circuit in Figure 3.3 with a lossless transmission line.

The calculated response shown in Figure 3.5 used $M' = 11$ and superposition. All results compare well with those presented in [32].

Figure 3.5: Waveforms for the circuit in Figure 3.3 with a lossy transmission line.

3.4.2 Example 2: Mismatched Circuit

The circuit shown in Figure 3.6 contains two coupled lines highly interconnected by a terminal network. The interconnect is not well matched to the terminations. The first transmission line has four conductors, is 0.3 meters long, and is described by the following parameters

$$\mathbf{L} = \begin{bmatrix} 1.5 & 0.18 & 0 & 0 \\ 0.18 & 1.5 & 0.18 & 0 \\ 0 & 0.18 & 1.5 & 0.18 \\ 0 & 0 & 0.18 & 1.5 \end{bmatrix} \mu H/m \quad (3.51)$$

$$\mathbf{C} = \begin{bmatrix} 0.266 & -0.02 & 0 & 0 \\ -0.02 & 0.266 & -0.02 & 0 \\ 0 & -0.02 & 0.266 & -0.02 \\ 0 & 0 & -0.02 & 0.266 \end{bmatrix} nF/m \quad (3.52)$$

and $\mathbf{R} = 0$, $\mathbf{G} = 0$. The second transmission line has two conductors, is 0.4 meters long and is described by the following parameters

$$\mathbf{L} = \begin{bmatrix} 750 & 95 \\ 95 & 750 \end{bmatrix} nH/m \quad (3.53)$$

$$\mathbf{C} = \begin{bmatrix} 0.133 & -0.009 \\ -0.009 & 0.133 \end{bmatrix} nF/m \quad (3.54)$$

and $\mathbf{R} = 0$, $\mathbf{G} = 0$. Circuit response for an input identical to that used in Example 1 is shown in Figure 3.7. The response was calculated using $M' = 11$ and superposition. Figure 3.8 compares the responses calculated at node 3 using $M' = 11$ with and without superposition and using $M' = 5'$ using superposition. All three of the waveforms are similar. As expected $M' = 5$ shows the least detail and $M' = 11$ with superposition shows the most detail.

As most of this network is mismatched reflected signals will exist on the lines for a long period of time. If an FFT were used to obtain the time domain response from a

Figure 3.6: A more complex interconnect circuit.

Figure 3.7: Response of the interconnect circuit shown in Figure 3.6.

Figure 3.8: Accuracy comparison of different NILT calculations using the response at node 8 of the circuit shown in Figure 3.6.

Figure 3.9: Interconnect circuit for the threshold search example.

frequency domain analysis it would not be possible to obtain the section of response shown independent of the rest of the response. Knowledge of the time required for the reflected signals to die out is required in order to establish the time period base for the FFT. No such knowledge is required for the NILT analysis as each time point is obtained independently. With NILT it is possible to calculate the response at an point or range of points independently from response calculations at other points. There are no problems with aliasing or periodicity when using NILT.

3.4.3 Example 3: Searching for a Threshold Crossing

The advantages of the inverse Laplace method can be exploited when the entire response of the circuit is not desired. An important application in interconnect analysis is the evaluation of waveform propagation delay. Consider the circuit of Figure 3.9 with response as shown in Figure 3.10. The time delay until each waveform crosses a 0.3 volt threshold can be found using a search algorithm that evaluates the time response only at selected points. The threshold search algorithm starts by evaluating the response at intervals spaced linearly along the time axis. When a waveform crosses the threshold the algorithm backs up and uses bi-section to locate the point to the desired accuracy.

Figure 3.10: Response of the circuit shown in Figure 3.9 showing the threshold crossings.

Threshold at 0.3 volts										
	11 Pole Approximation					5 Pole Approximation				
Node	Time Points			Time ns	Value volts	Time Points			Time ns	Value volts
	T	L	B			T	L	B		
v2	10	8	2	7.880	.30003	9	8	1	7.976	.30036
v3	5	3	2	2.749	.29997	5	3	2	2.743	.29998
v4	9	7	2	6.142	.29965	9	7	2	6.151	.30089
v5	7	5	2	4.106	.30001	7	5	2	4.105	.29993
v6	7	5	2	4.552	.29926	8	5	3	4.519	.30024

Table 3.1: Results of the Threshold Search; T:Total, L:Linear,B:Bisection.

For the circuit of Figure 3.9 this algorithm was used to evaluate the threshold points for all the waveforms to an accuracy of 0.001 volts. Results are summarized in Table 3.1 for the number of poles, M' , in (3.7) equal to 11 and 5. The voltage criteria is overly tight but reveals the differences in the two methods.

This algorithm found all of the threshold points after evaluating the response at only 19 time points. This is equivalent to 95 frequency domain evaluations of the network response when the 5 pole approximation is used. To achieve the same accuracy using FFT at least 2048 frequency points would be required. The 2048 points would cover a 20.48 nanosecond time range in increments of .01 nanoseconds.

3.5 Summary

The calculation of linear circuit response using NILT has been reviewed. Extensions to the basic NILT formula increasing the accuracy of the method have also been presented. In addition formulas for the inclusion of distributed circuit elements in the linear circuit model have been presented. These formulas are one of the contributions of this thesis and have been published as [29]. Several examples were presented

demonstrating NILT analysis of interconnect circuits.

Chapter 4

Parallel Implementation of Numerical Inversion of the Laplace Transform

4.1 Introduction

This chapter describes a parallel implementation of numerical inversion of the Laplace transform that distributes the computation over a network of workstations. This parallel implementation is based on the properties of the numerical inversion method as presented in the previous chapter, and scales linearly with the number of processors used for large problems. The chapter starts with a theoretical examination of the numerical inversion of the Laplace transform that exposes the independence properties that allow parallelization. The practical work required to implement the method using the SysCAD framework on a network of workstations is then presented. Performance measures for the implementation are discussed. An example circuit optimization is used to demonstrate the performance of the implementation within the SysCAD framework.

4.2 Partitioning Numerical Inversion of the Laplace Transform for Parallel Implementation

As presented in Section 3.2.1 numerical inversion of the Laplace Transform is accomplished by evaluating the expression

$$\hat{v}(t) = \frac{-1}{t} \sum_{i=1}^M k_i V(z_i/t) . \quad (4.1)$$

A NILT simulator developed from (4.1) provides a crude parallel processing ability. Running the simulator on two separate machines, one calculating the results for $t_a \in \{t_1, t_2, \dots, t_{n/2}\}$ the other calculating the results for $t_b \in \{t_{n/2+1}, t_{n/2+2}, \dots, t_n\}$, and concatenating the results provides a simple parallel processing system.

Parallelism is achieved at the time point level and the calculation can easily be split over up to n machines, where n is the number of time points to evaluate. A slightly more elaborate system programmed to evaluate (4.1) from collected values of $V(s)$ would allow the parallelism to be done at the z_i/t level allowing $n * M$ machines to be used.

The calculation of $V(s)$ requires solution to a set of linear equations and is the major computational expense of NILT method. The ability to perform this calculation in parallel provides the performance gain. In addition there is no reason NILT time point parallelism would not benefit from additional techniques applied to the calculation of $V(s)$ such as vectorization and partitioning of the solution.

An examination of the extensions to NILT, such as higher order functions and sensitivity calculations, shows that the calculations at one time point are linked by the calculation of derivatives. It is efficient to solve both the circuit function and its derivatives in the same processes as the same $Y^{-1}(s)$ is used in all calculations. This still allows parallelism at the time point level where separate machines calculate values at different time points.

Unlike the other extensions the decomposition method links the calculations at different time points together. The connection between time points is determined by

the decomposition and the required superposition to construct the overall response. The time domain response to the inputs can be calculated in parallel but the final response requires results from different time points to be combined. A final time point can be calculated in parallel with other final time point calculations. A loss in parallel efficiency results as separate final time point calculations involve calculation at identical decomposed time points.

4.3 Parallel Implementation

Parallel implementation of NILT was incorporated into the SysCAD framework. Each simulation process was identical to the single simulation process used previously. Parallelism was achieved using a mid level in the SysCAD structure that controlled multiple simulation processes. Implementation required the development of methods for controlling and communicating with separate parallel processes.

4.3.1 Support Routines

To support process control and data communication three levels of support interfaces were developed.

1. The lowest level of support provides asynchronous read and write operations on input output streams. Both the read and write are implemented so that if the operation cannot be completed immediately the operation is buffered and completed, asynchronously, at some later time without any further interaction with the users code. A number of read and write operations may be waiting for completion at any point in time. A wait operation is used for synchronization.
2. The middle level provides process control support for starting and stopping process on different machines as well as establishing the communication between the processes. This middle level manages the input output streams associated

with a particular subprocess. Error messages are passed to parent processes and signals are passed to child processes. This level makes sure that once a parent dies the child processes, possibly located on other machines, also die.

3. The top level is a standard interface that provides a uniform structure for data passing. This level implements the main receive-perform-send loop used by subprocesses as well as the send-wait-receive functions used by the controlling functions.

This system was developed in C under SunOS 4.1, an implementation of the UNIX operating system.

Asynchronous Read and Write

The main functions at this support level are

```
int apipe_read(ape ap, char *ptr, int len, int (*func) (), void *parm)
int apipe_write(ape ap, char *ptr, int len, int (*func) (), void *parm)
int apipe_wait(ape ap, int flag)
void func(ape ap, char *ptr, int len, void *parm)
```

The read and write operations are similar to the standard system functions with the additional provision that `func()` will be called once the operation is completed. The wait operation is used for synchronization.

This level is implemented using the signalling and asynchronous stream support associated with the SunOS system calls `poll(2)` and `ioctl(2)`.

Process Control

The process control level starts and stops the subprocesses. Along with access to the lower level asynchronous operations this level provides the operations.

```
int pipe_open(char *hostname, char *command)
int pipe_close(int rpd)
```


Error messages are automatically propagated to the controlling process and signals are automatically propagated to the subprocess. The SunOS library call *rexec(3N)* is used to implement this level.

4.3.2 Standard Interface

The SysCAD standard interface was used by all levels of the parallel system. Additional interface levels were added over the process control, read and write functions to make the interface match the SysCAD standard interface.

4.4 Example Framework

Figure 4.1 illustrates a SysCAD circuit optimization configuration of the framework using three process levels, a top level optimizer, a second level specification process that computes error functions, and a third level of simulator processes. Yield analysis and parameter sweeps can be performed using a different framework configuration with two process levels, a top level specification process sets the parameter values, and a second level of simulation processes.

Previous work with this system [33] was performed with the optimization process, the specification process and a single simulation process all running sequentially on one machine. This paper describes the parallel processing structure of the system. Moving from the sequential to the parallel system required only a redesign of the specification/simulation interface into the parallel version described later. This redesign resulted in the addition of a new parallel specification process module to the system. The previously existing simulation and optimization processes were linked with the new parallel processing libraries and then run with the new parallel specification process. Yield analysis and parameter sweeps may also be performed in parallel.

For the specific example configuration, shown in Figure 4.1 the optimizer is responsible for managing the optimization variable set x . At each optimization step

Figure 4.1: SysCAD optimization configuration with parallel NILT.

the optimizer passes the current variable set to the specification process and then blocks waiting the return of the error function values $e(x)$ and their derivatives $e'(x)$. When the error function values are returned the optimizer updates x and continues or exits. The optimizer minimizes the error function $e(x)$ subject to constraints on the variable set x .

The specification process manages the simulation processes and evaluates the error functions for the optimizer. The specification process translates the optimization variables from the optimizer into element values y for the simulators. The function values $f(y)$ and sensitivities $f'(y)$ returned from the simulator are used to calculate the error function values and derivatives for the optimizer.

The overall simulation consists of several simulation processes, running in parallel, completely independent from each other. Each simulation process reads the circuit description file and internally models the circuit. Once initialized a simulation process operates as driven by the specification process. The variable set is received to start the analysis for a time point. When the time point analysis is completed the output

functions and their derivatives are sent back to the specification process. Using NILT, parallel time domain analysis is possible.

The structure of each simulation process is given in pseudo code as

```

Initialize simulation from source deck.
Repeat {
  Receive simulation parameters.
  Perform simulation and sensitivity analysis.
  Send results.
}

```

This simple form is used by all processes except the top level optimization process. In the specification process the **Perform simulation and sensitivity analysis** function is replaced by the calculation of the error functions and their sensitivities.

Parallel performance is obtained at the specification/simulation interface. The basic algorithm used at this interface is

```

Receive variable set from Optimizer.
While there are error functions to evaluate {
  For each simulation process {
    If there is more data for this process
      Send data to this process activating it
  }A
  For each active simulation process {
    Wait for this process to finish.
    Receive circuit solutions and their sensitivities.
    Calculate new error function values and their sensitivitiesB.
  }
}
Send error function values and their sensitivities to the
Optimizer.

```

This receive-calculate-send operation is contained within a Repeat loop identical to that used in the simulation processes.

The problem formulation specifies the number of simulation processes to run and the machine on which each is to be located. This information is read in during the initialization process. Each simulation process is assigned a number of time points to

evaluate. One or more error functions will be formed based on the response function values at each time point. At the point marked ^A the simulation processes will be running in parallel, each calculating the response at a separate time point. The calculations at the point marked ^B can occur in parallel with simulation processes, running at ^A, that have not yet been waited on.

In each iteration through the `while` loop the simulation processes evaluate the circuit response at a single time point. The time required to complete the `while` loop depends on the time required for the processes to evaluate all of their time points. Total processing time is approximately the time required by the the longest of all the simulation processes.

4.5 Parallelism and Efficiency

The efficiency of the system is given by the ratio of execution time with N parallel processors, $t_{parallel}$, to the execution time for a single processor t_{serial} .

$$\frac{t_{parallel}}{t_{serial}} = F_{serial} + \text{MAX}(F_{shared}, \frac{F_{shared} + F_{parallel}}{N}) \quad (4.2)$$

where F_{serial} , F_{shared} and $F_{parallel}$ are the fractions of time spent executing serial, shared or parallel code respectively. For the simple case above F_{serial} refers to the time spent concatenating the results, F_{shared} is the common simulator initialization times and $F_{parallel}$ is the time spend calculating the results for the required time points. Where $F_{parallel} \gg \text{MAX}(F_{serial}, F_{shared})$ parallel execution will provide good results.

With the optimization system the same efficiency principle applies. In the example presented the specification process assigned a fixed set of time points to each simulation process. This does not provide the optimal efficiency that would be obtained by dividing the process evenly over N processors. An examination of the efficiency of the optimization system follows.

For reasons presented in Section 4.2 the parallel grain level of the system is a single time point. Computational considerations result in uneven processing requirements for each time point. It is generally not possible to evenly divide the simulation into N_p parts. An examination of the realizable efficiency of the optimization system follows. N_{ideal} will be used to represent the ideal speed up factor for the actual distribution of the computation. Given the ability to evenly divide the computation task N_{ideal} would equal N_p .

Suppose $t_i, i = 1, 2, \dots$, are the time points selected in optimization. Divide these time points into N_g groups, each group being handled by an independent simulation process. Let index set I_k contain time indices in the k th group. Suppose the circuit is excited by a trapezoidal signal which starts and stops rising (falling) at times T_1 and T_2 (T_3 and T_4), respectively. The relative computation effort for simulation and sensitivity analysis of group k is approximately

$$E_k = \sum_{i \in I_k} W(t_i) \quad (4.3)$$

where

$$W(t) = \begin{cases} 1, & \text{if } T_1 < t < T_2 \\ 2, & \text{if } T_2 < t < T_3 \\ 3, & \text{if } T_3 < t < T_4 \\ 4, & \text{if } T_4 < t \end{cases} \quad (4.4)$$

Slight variations in $W(t)$ exist due to the formulation requirements.

The computational effort expressed in $W(t)$ is a result of additional time point evaluations being used to compute a more accurate response through subtraction. As an example the response $v(t_a)$, $T_2 < t_a < T_3$, is calculated from the ramp response of the circuit $v(t)_{ramp}$ as

$$v(t_a) = v(t_a - T_1)_{ramp} - v(t_a - T_2)_{ramp} \quad (4.5)$$

and thus actually requires 2 time point evaluations. This formulation enhances mathematical accuracy with NILT and transmission lines.

Let T_{opt} , T_{spec} and T_{sim} represent the CPU time required by optimizer process, specification process and the complete simulator process, respectively. The total CPU time for serial mode optimization is

$$T_{serial} = T_{opt} + T_{spec} + T_{sim} \quad (4.6)$$

When running in parallel mode, the total CPU time is

$$T_{parallel} = T_{opt} + T_{spec} + T_{sim} \times \frac{\text{MAX}(E_k)}{\sum_{k=1}^{N_g} E_k} + T_{loss} \quad (4.7)$$

where T_{loss} includes system overhead for starting up various processes and for data communication between the processes, and the processes E_k , $k = 1, \dots, N_g$, are distributed to N_g different machines.

In high-speed VLSI interconnect design with many distributed transmission lines, the dominant CPU time is spent in circuit simulation. Therefore the real efficiency N_{real} of parallel processing approaches the ideal efficiency N_{ideal} , i.e.,

$$N_{real} = \frac{T_{serial}}{T_{parallel}} \approx N_{ideal} = \frac{\sum_{k=1}^{N_g} E_k}{\text{MAX}(E_k)} \quad (4.8)$$

4.6 Optimization Example

Design optimization of the time response for the circuit shown in Figure 4.2 originally required eight hours without parallel processing. This length of time made the circuit a good example for testing the speed up to be obtained using the new parallel processing technique. Results related to the speed up are presented in the next section. This section describes the optimization problem.

The circuit of Figure 4.2 represents a 4-bit bus structure in which the excitation signals propagate through the bus lines to various circuit blocks. There are 13 4-conductor transmission lines and 12 single conductor transmission lines. The applied voltage of both excitations is a 5 volt pulse with 1 nanosecond rise and fall times. The 5 volt level is held for 5 nanoseconds. The output responses of interest are the

waveforms of V_a , V_b , V_c and V_d . These responses before optimization are shown in Figure 4.3-Figure 4.6, respectively. The delay time for the signal responses V_b and V_d are approximately 3.1 and 4.1ns, respectively, based on a 3V threshold criterion. The crosstalk voltages V_a and V_c have peak voltages being 0.7V and 0.5V, respectively.

The design specification for optimization is a 2ns upper specification on the delay time of V_b and a 4ns upper specification on the delay time of V_d . Upper specifications were placed on the magnitude of crosstalk voltages V_a and V_c at 0.2V. A specification of 0.2 is also placed on the magnitude of V_d between the time range of 12.5ns and 14.5ns to reduce the ringing in the signal waveform. The weighted difference between the specification and the response voltage at a time point is an error function to be minimized by the optimization. This weighted difference is negative if the corresponding specification is exceeded. The total number of error functions is 61. The specifications are marked in Figure 4.3-Figure 4.6 as ∇ and $\wedge$. A total of 22 different time point evaluations are required in each iteration of optimization.

The vector of optimization variables x consists of the lengths of 13 4-conductor transmission lines ($l_1, l_2, \dots, l_{13}$), the distance between the 4-conductors (d_1, d_2, d_3), the terminating resistances ($R_2, R_4, R_7, R_{10}, R_{14}, R_{18}, R_{20}$), and the terminating capacitances (C_1, C_2, C_3, C_5). The distance variables d_1, d_2 and d_3 are common variables among the 13 4-conductor transmission lines. These values of these variables are given in Table 4.1. The total number of variables is 27.

The minimum distance separating the conductors was set at 0.8mm while the maximum total separation between conductors 1 and 4 was set at 5mm. The total length of the 13 transmission lines was fixed at 40cm. These distances are the constraints on the optimization variables x .

A combined first- and second-order minimax optimization program[34, 35] is used to minimize the maximum of all the error functions. After 16 iterations of optimization, all the specifications were satisfied. The parameters before and after optimization are listed in Table 4.1. The output responses V_a , V_b , V_c and V_d after optimization

Figure 4.2: Example circuit for parallel optimization.

Table 4.1: Variables for the Optimization Example

Variable	Description	Before Optimization	After Optimization
l_1	length of transmission line 1	0.025m	0.03834m
l_2	length of transmission line 2	0.035m	0.03686m
l_3	length of transmission line 3	0.025m	0.04052m
l_4	length of transmission line 4	0.04m	0.02266m
l_5	length of transmission line 5	0.04m	0.01m
l_6	length of transmission line 6	0.025m	0.01m
l_7	length of transmission line 7	0.035m	0.04008m
l_8	length of transmission line 8	0.035m	0.03222m
l_9	length of transmission line 9	0.025m	0.02222m
l_{10}	length of transmission line 10	0.04m	0.05707m
l_{11}	length of transmission line 11	0.025m	0.01108m
l_{12}	length of transmission line 12	0.025m	0.02486m
l_{13}	length of transmission line 13	0.025m	0.05408m
d_1	distance between conductors 1 & 2	1.0mm	3.29mm
d_2	distance between conductors 2 & 3	1.0mm	0.80mm
d_3	distance between conductors 3 & 4	1.0mm	0.91mm
R_2	resistor	25 Ω	12.53 Ω
R_4	resistor	25 Ω	19.00 Ω
R_7	resistor	200 Ω	144.17 Ω
R_{10}	resistor	200 Ω	37.28 Ω
R_{14}	resistor	200 Ω	200.78 Ω
R_{18}	resistor	200 Ω	57.93 Ω
R_{20}	resistor	100 Ω	75.66 Ω
C_1	capacitor	5pF	0.10pF
C_2	capacitor	5pF	1.50pF
C_3	capacitor	5pF	0.59pF
C_5	capacitor	10pF	6.53pF

are plotted against those before optimization in Figure 4.3-Figure 4.6, respectively. The propagation delay of V_b and V_d are reduced to 2ns and 3.5ns, respectively. The peak of crosstalk voltages V_a and V_c are reduced to 0.07 and 0.16V, respectively. In addition, the ringing in the output response V_d was significantly smaller than before optimization. The rise/fall times of signal V_b were also improved.

4.7 Parallel Performance

The example optimization was performed using a number of different parallel distributions of the problem. NILT order $p = 0$ with sensitivity calculations was used. Performance results for the distributions are summarized in Table 4.2. For tests 1 through 5 all processes were run on SUN IPC machines under SunOS 4.1 with the machines otherwise idle.

Efficiency calculations are based on the difference between the overall elapsed time and the time attributed to each process. Individual process times and weighting factors P_k for the simulations are given in Table 4.3. The total time shown in Table 4.3 is the total active process time excluding system overhead. This total process time was used with the elapsed real time to calculate, using (4.8), the efficiency values shown in Table 4.2. Ideal efficiencies are based on the P_k values while the real efficiencies are based on the measured process and elapsed times.

Time measurements were performed using the UNIX system call *gettimeofday()* for the elapsed time, and the call *getrusage()* for the process time. Process time was calculated as the sum of system and user times.

In different test runs of the example variations in overall time were noted. These variations were less than 10 seconds in the overall time of approximately 30,000 seconds. The variations, a function of the operating system performance and other network activity, were not characterized or included in the results. Results are reported for a typical run. No irregular results were observed.

Figure 4.3: Optimization circuit waveform V_a before and after optimization.

Figure 4.4: Optimization circuit waveform V_b before and after optimization.

Figure 4.5: Optimization circuit waveform V_c before and after optimization.

Figure 4.6: Optimization circuit waveform V_d before and after optimization.

Table 4.2: Distributed Process Efficiency Results for the Optimization Example

Test	Real Time	Efficiency		Process Distribution					
		Real	Ideal	Opti.	Spec.	Sim.1	Sim.2	Sim.3	Sim.4
1	30320	0.988	1.00	A	A	A			
2	30507	0.982	1.00	A	A	A	A	A	A
3	30464	0.984	1.00	A	A	B	B	B	B
4	17190	1.743	1.77	A	A	B	B	C	C
5	8811	3.401	3.45	A	A	B	C	D	E
6	3668	8.170	8.625	A	A	22 proc. on 17 mach.			
where A, B, C, D, E represent 5 different machines.									

Test 1-3 The first test was performed with all processes on a single machine. One simulation process was used to include all simulation points. The second test was also performed on a single machine. In this case the simulation task was divided over four processes. The elapsed time is slightly longer than that measured in the first run. The extra elapsed time was probably due to overhead in the operating system required to swap between the processes and the startup time required for the initialization of each process. The third test was divided over two machines. The four simulation processes were placed on one machine with the specification and optimization processes on the other machine. Note the speed up over the second run. Communication between machines produces better results than communication and swapping between the simulators and the specification program on a single machine.

Elapsed time differences for the first three tests are less than 1 percent. These differences are not significant factors in the overall elapsed time and indicate that the inter-process and inter-machine communication are relatively efficient. The high overall efficiency shows that there is no penalty for the strategy of separate optimizer, specification and simulation.

Test 4 In the fourth test the simulation process were divided between two machines with the controlling programs on a third machine. If the simulation loads had been balanced the ideal efficiency would have been close to 2.00 instead of the actual ideal efficiency calculated as

$$N_{ideal} = \frac{20 + 19 + 10 + 20}{\text{MAX}(20 + 19, 10 + 20)} = 1.77 \quad (4.9)$$

A measured efficiency of

$$N_{real} = \frac{29969}{17190} = 1.743 \quad (4.10)$$

was obtained.

Test 5 In the fifth test each simulation process was run on a separate machine with the controlling processes on a fifth machine. The ideal efficiency was $N_{ideal} = 69/20 = 3.45$ while the actual efficiency was $N_{real} = 29969/8811 = 3.401$.

Test 6 This test was run on an active network with competing processes from other users. The 17 machines used include Sun IPC's as used in the other tests, Sun SPARC 2's, a Sun SPARC 1+, a Sun 4/110 and a Sun 4/390. The simulation was divided into 22 different simulation process, with a maximum P_k of 4 while the total was 69 as before. The placement of two $P_k = 4$ process on one IPC machine was used as a predictable processing limit to make the results comparable to the earlier tests. The achievable ideal efficiency under these conditions was $N_{ideal} = 69/4/2 = 8.625$. The actual efficiency was $N_{real} = 8.170$.

A more equitable division of process load considering cpu speed and competing processes would have resulted in a higher N_{real} . An optimal division is difficult to achieve on a heterogeneous network with dynamic competing loads and was not attempted.

4.8 Summary

A parallel implementation of NILT has been demonstrated. This parallel implementation provides a near linear speed up with the number of workstations applied to a NILT problem. The implementation has been incorporated into the SysCAD framework allowing optimization and monte-carlo analysis to benefit from the technique. These results have been presented in [36].

Table 4.3: Active Process Times for the Processes in Table 4.2

	Opti.	Spec.	Sim.1	Sim.2	Sim.3	Sim.4	Total
Time	10	6	8686	8246	4350	8671	29969
P_k			20	19	10	20	69

Chapter 5

The Piecewise Decomposition Technique

5.1 Introduction

This chapter describes a piecewise linear decomposition technique for the simultaneous solution to a set of time and frequency domain equations. The technique is important as it allows the analysis of circuits composed of interconnect requiring frequency domain analysis and nonlinearities requiring time domain analysis. The previously described numerical inversion of the Laplace transform technique applies only to linear circuits described in the frequency domain, the piecewise decomposition technique described combines such linear solution methods with the time domain solution of nonlinear circuits.

The chapter starts with a set of network equations describing the entire circuit. The equations are then separated into frequency domain and time domain equations and a piecewise linear waveform is used as the solution at the boundary between the two domains. An efficient iterative technique for simultaneously solving the time domain and frequency domain equations is developed. A number of examples are presented to illustrate the method.

5.2 Formulation of the Network Equations

The network equations for a circuit containing both nonlinear elements and distributed linear elements are described here. The nonlinear elements require a time domain representation relating values at the current time point, while the distributed linear elements are best described using a frequency domain expression. The network equations are required to relate the time and frequency domains so that a single solution to the network can be obtained.

5.2.1 Overall Network Equations

Consider a nonlinear circuit π which contains linear lumped components, nonlinear components and arbitrary linear subcircuits. The linear lumped components may be described by equations in either the time or frequency domain while the nonlinear components may only be described by time domain equations. The arbitrary linear subcircuits may contain frequency dependent or distributed elements that are best described in the frequency domain. Let the linear subcircuits be grouped into a single subcircuit ψ . The subcircuit ψ is not constrained to be connected.

Without loss of generality the modified nodal admittance matrix equation[37] of the circuit π can be written as

$$\mathbf{C}_\pi \frac{\partial \mathbf{v}_\pi(t)}{\partial t} + \mathbf{G}_\pi \mathbf{v}_\pi(t) + \mathbf{d}_\psi \mathbf{i}_\psi(t) - \mathbf{f}(\mathbf{v}_\pi(t)) - \mathbf{e}_\pi(t) = \mathbf{0}, \quad t \in [0, T] \quad (5.1)$$

where

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

$\mathbf{e}_\pi(t) \in \mathbb{R}^{N_\pi}$ is the vector of source waveforms,

$\mathbf{C}_\pi \in \mathbb{R}^{N_\pi \times N_\pi}, \mathbf{G}_\pi \in \mathbb{R}^{N_\pi \times N_\pi}$ are constant matrices,

$f(\mathbf{v}) : \mathbb{R}^{N_\pi} \rightarrow \mathbb{R}^{N_\pi}$ is a function describing the nonlinear elements of the circuit and, $\mathbf{d}_\psi = [d_{i,j}]$, $d_{i,j} \in \{0, 1\}$, $i \in \{1, 2, \dots, N_\pi\}$, $j \in \{1, 2, \dots, N_\psi\}$ with a maximum of one nonzero in each row or column is a selector matrix that maps, $\mathbf{i}_\psi(t) \in \mathbb{R}^{N_\psi}$ the vector of currents entering the linear subcircuit nodes, into the node space $\mathbb{R}^{N_\pi}$ of the nonlinear circuit π .

The first, second and fourth terms in (5.1) cover the linear lumped components and independent sources. The fifth term covers the nonlinear components. The third term connects the terminal currents of the linear subcircuits to the rest of the circuit through the mapping $\mathbf{d}_\psi$. Generalization of (5.1) to include nonlinear elements with terminal characteristics dependent on the derivatives or integrals of the voltages or currents of the circuit is possible and will not change the basic principle of the method.

The subcircuit ψ can be represented in the complex frequency domain by its modified nodal admittance matrix equation

$$\mathbf{Y}_\psi(s)\mathbf{V}_\psi(s) = \mathbf{I}_\psi(s) \quad (5.2)$$

where s is the complex frequency, $\mathbf{Y}_\psi(s)$ is the complex frequency domain modified nodal matrix of the subcircuit ψ , $\mathbf{V}_\psi(s)$ is the vector of node voltages appended by linear inductor currents, and $\mathbf{I}_\psi(s) = \mathcal{L}[\mathbf{i}_\psi(t)]$, where $\mathcal{L}$ denotes the Laplace transform. In the special case of multiconductor transmission lines the matrix $\mathbf{Y}_\psi(s)$ can be efficiently derived as a function of the transmission line parameters as described in Section 3.3.2.

Note that the modified nodal formulation is used in (5.2) in order to allow for currents through inductors, short circuits and elements which do not have explicit admittance representations. However, other appropriate terminal characterizations such as Z or S parameters may also be used.

The node voltages of the nonlinear circuit π are related to the node voltages of the linear subcircuit ψ by the equation

$$\mathbf{V}_\psi(s) = \mathbf{d}_\psi^t \mathcal{L}[\mathbf{v}_\pi(t)] \quad (5.3)$$

Figure 5.1: Example circuit used to demonstrate formulation of the circuit equations.

where $'$ indicates transpose, and d_ψ is as defined in (5.1).

Example Circuit: Equation Formulation The circuit shown in Figure 5.1 will be used to illustrate the presented concepts. The circuit consists of a linear subcircuit with frequency dependent parameters and two nonlinear terminations. The circuit can be described by (5.1) and (5.2) where

$$\mathbf{v}_\pi = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ i_E \\ q \end{bmatrix}, \quad \mathbf{f}(\mathbf{v}_\pi(t)) = \begin{bmatrix} 0 \\ 0 \\ f_1(v_3, v_5) \\ 0 \\ 0 \\ 0 \\ f_2(v_4) \end{bmatrix}, \quad \mathbf{C}_\pi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & C & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{e}_\pi = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ E \\ 0 \end{bmatrix}, \quad \mathbf{i}_\psi(t) = \begin{bmatrix} i_2 \\ i_5 \\ i_3 \\ i_4 \end{bmatrix}, \quad \mathbf{G}_\pi = \begin{bmatrix} G & -G & 0 & 0 & 0 & 1 & 0 \\ -G & G & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{d}_\psi = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The linear subcircuit is a lossy coupled transmission line that can be represented in the frequency domain by (see Appendix A)

$$\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} \end{bmatrix} \begin{bmatrix} V_2 \\ V_5 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} I_2 \\ I_5 \\ I_3 \\ I_4 \end{bmatrix}$$

5.2.2 Separating the Time and Frequency Domain Representations

The difficulty in solving (5.1) and (5.2) simultaneously is due to the fact that they implicitly contain a mixture of frequency and time domain representations. To overcome this difficulty the nonzero entries in $\mathbf{f}(\mathbf{v}(t))$ are replaced by a set of independent waveform sources $\mathbf{x}(t)$.

Let

$$\mathbf{f}(\mathbf{v}_\pi(t)) = \mathbf{d}_f \mathbf{x}(t) \quad (5.4)$$

where $\mathbf{x} \in \mathbb{R}^{N_f}$,

$$\mathbf{d}_f[d_{i,j}], \quad d_{i,j} \in \{0, 1\}, \quad i \in \{1, 2, \dots, N_\pi\}, \quad j \in \{1, 2, \dots, N_f\}$$

is a selector matrix, and N_f is the number of nonzero entries in $\mathbf{f}(\mathbf{v}_\pi(t))$.

Using (5.4), (5.1) is reduced to a set of linear differential equations in the form

$$\mathbf{C}_\pi \frac{\partial \mathbf{v}_\pi(t)}{\partial t} + \mathbf{G}_\pi \mathbf{v}_\pi(t) + \mathbf{d}_\psi \mathbf{i}_\psi(t) = \mathbf{d}_f \mathbf{x}(t) + \mathbf{e}_\pi(t), \quad t \in [0, T] \quad (5.5)$$

which is represented in the complex frequency domain as

$$[s\mathbf{C}_\pi + \mathbf{G}_\pi] \mathbf{V}_\pi + \mathbf{d}_\psi \mathbf{I}_\psi(s) = \mathcal{L}[\mathbf{d}_f \mathbf{x}(t) + \mathbf{e}_\pi(t)] + \mathbf{C}_\pi \mathbf{v}_\pi(0) \quad (5.6)$$

Using (5.2) and (5.3)

$$\mathbf{Y}_\pi(s) \mathbf{V}_\pi = \mathcal{L}[\mathbf{d}_f \mathbf{x}(t) + \mathbf{e}_\pi(t)] + \mathbf{C}_\pi \mathbf{v}_\pi(0) \quad (5.7)$$

where

$$\mathbf{Y}_\pi(s) = s\mathbf{C}_\pi + \mathbf{G}_\pi + \mathbf{d}_\psi \mathbf{Y}_\psi(s) \mathbf{d}_\psi^t \quad (5.8)$$

is the complex frequency domain modified nodal analysis admittance representation for all of the linear components in the circuit π .

Solving (5.7) for the time domain node voltages

$$\mathbf{v}_\pi(t) = \mathcal{L}^{-1} \left[\mathbf{Y}_\pi^{-1}(s) \{ \mathcal{L}[\mathbf{d}_f \mathbf{x}(t) + \mathbf{e}_\pi(t)] + \mathbf{C}_\pi \mathbf{v}_\pi(0) \} \right] \quad (5.9)$$

The nonlinear circuit is effectively partitioned into two parts, represented by a nonlinear equation (5.4) and a linear equation (5.9).

A physical interpretation of (5.4) and (5.9) can be viewed as replacing the nonlinear branches in the circuit π by a set of time dependent augmenting sources represented by the waveform sources $\mathbf{x}(t)$. This reduces π to a linear circuit which can be solved in the frequency domain. The choice of the set of waveforms $\mathbf{x}(t)$ should be

Figure 5.2: Example circuit with nonlinear elements replaced by independent sources.

such that $\mathbf{v}_r(t)$ meets the constraints imposed by the constitutive relationships of the nonlinear branches given by (5.4). As an example consider again the circuit shown in Figure 5.1. The resulting circuit after replacing the nonlinear branches is shown in Figure 5.2. For this example circuit the selector matrix $\mathbf{d}_f$ is given as

$$\mathbf{d}_f^t = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

5.3 Solution of the Network Equations

5.3.1 Solution using Contraction Mapping

A possible method for solving (5.4) and (5.9) is by using the contraction mapping principle [38]. Starting with an initial guess $\mathbf{x}^{(0)}(t); t \in [0, T]$ the following set of iterations are performed:

$$\mathbf{v}_r^{(i)}(t) = \mathcal{L}^{-1} \left\{ \mathbf{Y}_r^{-1}(s) \left\{ \mathcal{L} \left[\mathbf{d}_f \mathbf{x}^{(i)}(t) + \mathbf{e}_r(t) \right] + \mathbf{C}_r \mathbf{v}_r(0) \right\} \right\} \quad (5.10)$$

$$\mathbf{x}^{(i+1)}(t) = \mathbf{d}_f^t \mathbf{f}(\mathbf{v}_r^{(i)}(t)) \quad (5.11)$$

where $i = 0, 1, 2, \dots$.

Convergence is obtained whenever

$$\int_0^T \|v_r^{(i+1)}(\tau) - v_r^{(i)}(\tau)\| d\tau < \epsilon \quad (5.12)$$

where ϵ is a prespecified accuracy measure.

Note that the contraction mapping technique as given by (5.10) and (5.11) is equivalent to the Waveform Relaxation (WR) approach [39],[40] coupled with partitioning of the circuit into linear and nonlinear subcircuits where each subcircuit is analyzed using direct incremental techniques. (5.10) and (5.11) require re-evaluation of the time domain response of the linear circuit (5.10) at each waveform iteration. An alternative approach, based on a piecewise linear time domain representation of $x(t)$ is described in the following section. This alternate approach requires the evaluation of the time domain response of the linear circuit only once instead of two to four times as is typical for the WR approach.

5.3.2 The Piecewise Linear Time Domain Representation

Let $x(t) = [x_m(t)]$; $m \in \{1, 2, \dots, N_f\}$ be a piecewise linear function (Figure 5.3) described by

$$x(t) = \sum_{k=1}^L (B_k + A_k t) [u(t - t_{k-1}) - u(t - t_k)] \quad (5.13)$$

where

L is the number of linear segments

$$A_k = [a_{1k} \ a_{2k} \ a_{3k} \dots \ a_{N_f,k}]^t$$

$$B_k = [b_{1k} \ b_{2k} \ b_{3k} \dots \ b_{N_f,k}]^t$$

$u(t)$ is the unit step function,

and a_{mk}, b_{mk} are parameters defining the segment k of waveform $x_m(t)$ in the interval

$$t_{k-1} \leq t \leq t_k \text{ such that}$$

$$a_{mk} = \frac{x_{mk} - x_{m(k-1)}}{t_k - t_{k-1}}$$

$$b_{mk} = \frac{x_{m(k-1)}t_k - x_{mk}t_{k-1}}{t_k - t_{k-1}}$$

Let the interval of analysis $t \in [0, T]$ be divided into L equal length intervals such that $t_k - t_{k-1} = \Delta$ and $L\Delta = T$. (5.13) can be re-written in the form

$$x(t) = \sum_{k=1}^L u_k(t - t_{k-1})(B_k + A_k t_{k-1}) + \sum_{k=1}^L u_g(t - t_{k-1})A_k \quad (5.14)$$

where

$$u_k(t) = \text{diagonal} \{u(t) - u(t - \Delta)\} \in \mathbb{R}^{N_f \times N_f}$$

$$u_g(t) = \text{diagonal} \{t(u(t) - u(t - \Delta))\} \in \mathbb{R}^{N_f \times N_f}$$

Using (5.14) the piecewise linear representation for $x(t)$ is substituted into (5.9) yielding

$$v_r(t) = \sum_{k=1}^L H(t - t_{k-1})(B_k + A_k t_{k-1}) + \sum_{k=1}^L G(t - t_{k-1})A_k + f(t) \quad (5.15)$$

where

$$f(t) = \mathcal{L}^{-1} \left[Y_r^{-1}(s) \{ \mathcal{L}[e_r(t)] + C_r v_r(0) \} \right] \quad (5.16)$$

$$g(t) = \mathcal{L}^{-1} \left[Y_r^{-1}(s) \mathcal{L} \{ d_f u_g(t) \} \right] \quad (5.17)$$

$$h(t) = \mathcal{L}^{-1} \left[Y_r^{-1}(s) \mathcal{L} \{ d_f u_k(t) \} \right] \quad (5.18)$$

The vector $f(t)$ represents the output $v_r(t)$ corresponding to the independent sources $e_r(t)$ when the augmenting sources are set to zero. $g(t)$ and $h(t)$ are time domain response matrices

$$g(t) = \{g_{ij}\}; i \in \{1, 2, \dots, N_r\}, j \in \{1, 2, \dots, N_f\} \quad (5.19)$$

$$g_{ij} = [v_r(t)]_i \begin{cases} x_j(t) = t[u(t) - u(t - \Delta)] \\ x_k(t) = 0; k \neq j \\ e_r(t) = 0 \end{cases} \quad (5.20)$$

Figure 5.3: A piecewise linear waveform.

$$\mathbf{h}(t) = \{h_{ij}\}; i \in \{1, 2, \dots, N_\pi\}, j \in \{1, 2, \dots, N_f\} \quad (5.21)$$

$$h_{ij} = [\mathbf{v}_\pi(t)]_i \begin{cases} x_j(t) = u(t) - u(t - \Delta) \\ x_k(t) = 0; k \neq j \\ \mathbf{e}_\pi(t) = 0 \end{cases} \quad (5.22)$$

corresponding to the response of the circuit π when a pulse of the specified form, $x_j(t)$, is applied in place of augmenting source j and all other sources are set to zero.

$\mathbf{f}(t)$, $\mathbf{g}(t)$ and $\mathbf{h}(t)$ can be obtained by analyzing the linear circuit in the frequency domain and using the inverse Fourier transform, or the inverse Laplace transform, to convert the results to the time domain. Note that $\mathbf{f}(t)$, $\mathbf{g}(t)$ and $\mathbf{h}(t)$ need to be calculated only once for a given circuit.

To solve for the response of the nonlinear circuit, $\mathbf{A}_k$, $\mathbf{B}_k$, and therefore $\mathbf{x}(t)$, must be determined so that (5.15) and (5.4) are satisfied simultaneously. Substituting for $\mathbf{A}_k$ and $\mathbf{B}_k$ in (5.15) yields

$$\mathbf{v}_\pi(t) = \sum_{k=1}^L \mathbf{H}(t - t_{k-1}) \mathbf{x}_{k-1} + \sum_{k=1}^L \mathbf{G}(t - t_{k-1}) \frac{\mathbf{x}_k - \mathbf{x}_{k-1}}{\Delta} + \mathbf{f}(t) \quad (5.23)$$

where $\mathbf{x}_k = \mathbf{x}(t_k)$.

An approximate solution to the circuit equations is obtained by solving (5.23) with (5.4) at a finite number of points L . The approximate solution may be made arbitrarily accurate by increasing L .

Example Circuit: Calculation of $\mathbf{g}$ and $\mathbf{h}$ The evaluation of the first column of $\mathbf{h}(t)$ is illustrated in Figure 5.4 for the example circuit of Figure 5.1. The pulse excitation is applied in place of the first nonlinear element which is replaced by a waveform source and the response calculated at all other nodes.

No mention has yet been made of a method for calculating $\mathbf{f}(t)$, $\mathbf{g}(t)$ and $\mathbf{h}(t)$, the responses of the linear system. NILT as described in Chapter 3 is the method used in the examples presented here. Any other method that produces the required time domain responses may also be used.

Figure 5.4: Excitation of the example circuit with the waveform $h(t)$.

In principle, the solution of (5.23) and (5.4) can be obtained using the contraction mapping approach described by (5.10) and (5.11). Note that in this case the time domain response of the linear circuit (5.11) is completely defined at each iteration by $f(t)$, $g(t)$ and $h(t)$ which are to be evaluated only once. A more robust technique for solving (5.23) and (5.4), based on Newton-Raphson iterations is described in the next section. The solution technique also takes advantage of the piecewise linear time domain functions to avoid the need for re-evaluating the time domain response of the linear circuit at each iteration.

5.3.3 Solution using Newton-Raphson Iterations

To use Newton Raphson iterations the nonlinear equations (5.4) are discretized at L time points resulting in

$$f(v_\pi(t_r^*)) - d_f x(t_r^*) = 0 \quad (5.24)$$

where $t_{r-1} < t_r^* \leq t_r$ and $r \in \{1, 2, \dots, L\}$. Let $x_r = x(t_r)$ and $v_r^* = v_\pi(t_r^*)$. At each time point, (5.24) represents a set of N_f nonlinear equations in the N_f unknowns $x_r =$

$\{x_{1r}, x_{2r}, \dots, x_{N,r}\}$. Using (5.23), $v_r(t_r^*)$ can be represented as an explicit function of x_r :

$$v_r^* = v_r(t_r^*) = R_r x_r + S_r \quad (5.25)$$

where

$$\begin{aligned} R_r &= \frac{g(t_r^* - t_{r-1})}{\Delta} \\ S_r &= \sum_{k=1}^r \left[h(t_r^* - t_{k-1}) - \frac{g(t_r^* - t_{k-1})}{\Delta} \right] x_{k-1} + \sum_{k=1}^{r-1} \left[\frac{g(t_r^* - t_{k-1})}{\Delta} \right] x_k \\ &\quad + f(t_r^*) \end{aligned}$$

Solution of (5.24) and (5.25) can be obtained using Newtons iterations [41] in the form

$$J^{(i)} [x_r^{(i+1)} - x_r^{(i)}] = - [f(v_r^{(i)}) - d_f x_r^{(i)}] \quad (5.26)$$

Using (5.25) the Jacobian J can be evaluated as

$$J = \left[\frac{df}{dv} \right] R_r - \frac{t_r^* - t_{r-1}}{\Delta} D_f \quad (5.27)$$

The iterations are repeated until $\|x_r^{(i+1)} - x_r^{(i)}\|$ is less than a predefined tolerance value. The entire time domain solution is calculated by carrying out Newtons iterations until convergence for each time point $r \in \{1, 2, \dots, L\}$.

It should be mentioned that the proposed algorithm differs from the Waveform Newton (WN) [42, 43] and Waveform Relaxation-Newton (WRN) [44]-[46] techniques in two main aspects:

1. Application of the WN algorithm to the analysis of nonlinear circuits with frequency dependent parameters will lead to a set of linear time-varying differential equations which have to be solved simultaneously with the set of frequency domain equations describing the frequency-dependent subcircuits. These simultaneous sets of equations can not be solved directly and relaxation techniques have to be used.

2. The iteration equations of the WRN algorithm are linear time-varying differential equations which are solved using WR techniques. By contrast, using the proposed piecewise linear method, a set of nonlinear algebraic equations are obtained at each time-point through the decomposition of the nonlinear circuit. These algebraic equations are solved directly using Newtons iterations.

Example Circuit: The Newton-Raphson Iterations For the example circuit of Figure 5.1, (5.24) and (5.25) are in the form

$$\begin{bmatrix} 0 \\ 0 \\ f_1(v_3(t_r^*), v_5(t_r^*)) \\ 0 \\ 0 \\ 0 \\ f_2(v_4(t_r^*)) \end{bmatrix} - \mathbf{d}_f \begin{bmatrix} x_1(t_r^*) \\ x_2(t_r^*) \end{bmatrix} = 0 \quad (5.28)$$

$$\begin{bmatrix} v_1(t_r^*) \\ v_2(t_r^*) \\ v_3(t_r^*) \\ v_4(t_r^*) \\ v_5(t_r^*) \\ i_E(t_r^*) \\ q(t_r^*) \end{bmatrix} = \mathbf{R}_r \begin{bmatrix} x_1(t_r^*) \\ x_2(t_r^*) \end{bmatrix} + \mathbf{S}_r \quad (5.29)$$

and the Jacobian $\mathbf{J}$ is given by :

$$\mathbf{J} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{df_1}{dv_3} & 0 & \frac{df_1}{dv_5} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{df_2}{dv_4} & 0 & 0 & 0 \end{bmatrix}_{t=t_r^*} \quad \mathbf{R}_r - \frac{t_r^* - t_{r-1}}{\Delta} \mathbf{D}_f \quad (5.30)$$

5.4 Summary of the Computation Algorithm

The following is a summary of the piecewise decomposition algorithm :

Formulate $\mathbf{Y}_r(s)$ and solve for $\mathbf{f}(t), \mathbf{g}(t), \mathbf{h}(t)$; using (5.18)-(5.16),

Calculate $\mathbf{R}_r$, and coefficients in $\mathbf{S}_r$; using (5.25),

for $i = 1$ to number of time points

begin { time point i

Calculate $\mathbf{S}_r$ as a function of $\mathbf{x}_k, k < i$; using (5.25),

Set $j = 0, \mathbf{x}_i^{(j)} = \mathbf{x}_{i-1}$

repeat {

Calculate $f(\mathbf{x}_i^{(j)}), f'(\mathbf{x}_i^{(j)})$

Calculate $\mathbf{J}(\mathbf{x}_i^{(j)})$

Use (26) to find $\mathbf{x}_i^{(j+1)}, j = j + 1$

} until $\|\mathbf{x}_i^{(j)} - \mathbf{x}_i^{(j-1)}\| \leq \epsilon$

} done time point i

Note that : $\mathbf{f}(t), \mathbf{g}(t)$ and $\mathbf{h}(t)$ need to be calculated only once for a given circuit. Consequently $\mathbf{R}_r$ and the coefficients of $\mathbf{S}_r$ (5.25) need to be calculated only once for the circuit while $\mathbf{S}_r$ is calculated once per time point. $\mathbf{h}(t), \mathbf{g}(t)$ and $\mathbf{f}(t)$ can be

calculated simultaneously using one LU decomposition of $\mathbf{Y}_r$ and different forward and backward substitutions.

5.5 Error Analysis

There are three possible sources of computational error in the algorithm. These sources are

1. **Error in the linear circuit response:** This is dependent on the method used to obtain the linear circuit response. If NILT is used there is a truncation error associated with the finite Taylor series expansion (Section 3.2.2). The more accurate the linear circuit response the more accurate the final response.
2. **The piecewise linear approximation:** The error introduced by the piecewise linear approximation is coupled tightly to the number of time points in the approximation, the larger the number of time points the smaller the error. The accuracy of the piecewise time domain representation may be estimated by increasing the number of time points and comparing the two solutions. It is to be pointed out that a variable time step size can be used. In general, the response of the linear subcircuit must be reevaluated for each time step size. However if the time step increase is a multiple of Δ the new response $g(t)$ and $h(t)$, can be calculated using superposition.
3. **Newton's iterations:** The convergence criteria for the iterations allows this error to be tightly controlled and reduced to insignificant levels[41].

5.6 Numerical Examples

The following examples are used to demonstrate the accuracy and capabilities of the presented method. Results were verified, where applicable, using other available simulators or by comparing with other published results.

Figure 5.5: A nonlinear circuit with frequency dependent lumped elements.

5.6.1 Example 1: Comparison with HSpice

The circuit shown in Figure 5.5 contains two coupled nonlinear elements and three frequency dependent elements. For the purpose of comparing the results with those obtained using other methods the frequency dependent elements have been selected so that they are equivalent to a combination of frequency independent elements. The frequency dependent elements are described by

$$L_1(\omega) = L_2(\omega) = \frac{1-\omega^2}{1-2\omega^2} \quad \text{and} \quad C_1(\omega) = \frac{3-\omega^2}{1-\omega^2}$$

The nonlinear equations are $I_1 = V_1^3 \times V_2$, $|I_2| = V_2^2$. The circuit input V_a rises from 0 to 5 volts in 0.25 seconds and then remains at 5 volts.

The circuit response is shown in Figure 5.6 as calculated by HSpice[8] and by the new piecewise decomposition method. Note that HSpice cannot handle generic frequency dependent elements in combination with nonlinear elements. NILT with $M' = 5$ and superposition was used to calculate the response of the linear subcircuit.

5.6.2 Example 2: Simple Transmission Line Circuit

As shown in Figure 5.7 this example includes a single transmission line. The transmission line is 0.5 meters long with $m/L = 160\text{nH/m}$, $C = 100\text{pF/m}$, $R = 25\Omega/\text{m}$

Figure 5.6: The voltages across the nonlinear elements of the circuit shown in Figure 5.5.

Figure 5.7: A circuit with a single transmission line.

and $G = 5\text{mS/m}$. The nonlinear element is described by $I_1 = V_1^3$. The input waveform V_a is a 6 nanosecond pulse with 1.5 nanosecond rise and fall times. When the transmission line is lossless, $R = 0$ and $G = 0$, the circuit may be simulated using HSpice[8]. A comparison between the results obtained using HSpice and those obtained using the piecewise method are shown in Figure 5.8. NILT with $M' = 11$ and superposition was used to calculate the response of the linear subcircuit. A time step of .1nanoseconds was used for a total number of 200 time points.

In the lossy case HSpice cannot be used since the lossy coupled transmission line is modeled as a frequency dependent circuit element. The piecewise method was used to simulate the circuit with a lossy transmission line as described above. The output waveform is also shown in Figure 5.8.

5.6.3 Example 3: Comparison with Linear Analysis

The circuit shown in Figure 5.9 is the same one used in Section 3.4.2. The circuit contains two coupled lines highly interconnected by a terminal network. The 200Ω resistor at V8 is treated as a nonlinear element with $V_r = I_r * 200$ as the "nonlinear" equation. The first transmission line has four conductors, is 0.3 meters long, and is

Figure 5.8: Response of the circuit shown in Figure 5.7. A comparison with HSpice results is shown for the case of a lossless transmission line.

Figure 5.9: A linear circuit with coupled transmission lines.

described by the following parameters

$$\mathbf{L} = \begin{bmatrix} 1.5 & 0.18 & 0 & 0 \\ 0.18 & 1.5 & 0.18 & 0 \\ 0 & 0.18 & 1.5 & 0.18 \\ 0 & 0 & 0.18 & 1.5 \end{bmatrix} \mu\text{H/m} \quad (5.31)$$

$$\mathbf{C} = \begin{bmatrix} 0.266 & -0.02 & 0 & 0 \\ -0.02 & 0.266 & -0.02 & 0 \\ 0 & -0.02 & 0.266 & -0.02 \\ 0 & 0 & -0.02 & 0.266 \end{bmatrix} \text{nF/m} \quad (5.32)$$

and $\mathbf{R} = 0$, $\mathbf{G} = 0$. The second transmission line has two conductors, is 0.4 meters long and is described by the following parameters

$$\mathbf{L} = \begin{bmatrix} 750 & 95 \\ 95 & 750 \end{bmatrix} \text{nH/m} \quad (5.33)$$

$$\mathbf{C} = \begin{bmatrix} 0.133 & -0.009 \\ -0.009 & 0.133 \end{bmatrix} \text{nF/m} \quad (5.34)$$

and $\mathbf{R} = 0$, $\mathbf{G} = 0$. As in Section 3.4.2 the input is a 1 volt, 6 nanosecond pulse with 1.5 nanosecond rise and fall times. `refig5res3` compares the circuit response at V8 to that shown in Figure 3.7 which was calculated without using the piecewise method. Note the improved accuracy between NILT5 and PWL5 as compared to NILT11. Considering this the increased detail on PWL11 should be improved accuracy over NILT11. PWL11 and PWL5 were calculated using $\mathbf{F}$, $\mathbf{G}$ and $\mathbf{H}$ as calculated using the indicated NILT order and superposition. The increased accuracy in the PWL waveforms is due to the convolution nature of the piecewise linear method which is an extended form of superposition.

Note that a change to the far end load, such as the introduction of a nonlinear equation in place of the resistor equation, does not require a new analysis of the linear network, instead the previous values of $\mathbf{F}$, $\mathbf{G}$ and $\mathbf{H}$ are used to calculate a new

Figure 5.10: Response of the circuit shown in Figure 5.9.

solution to the network equations. Similarly a change to the applied source requires only a new calculation for $\mathbf{F}$. As the network topology has not changed the previously calculated $\mathbf{G}$ and $\mathbf{H}$ can be used.

5.6.4 Example 4: Transmission Line Network with Multiple Nonlinear Terminations

Figure 5.11 shows an interconnect circuit. The nonlinear elements are all described by the equation $I = V^3$. The parameters of the single transmission line are $\mathbf{L} = 600\text{nH/m}$, $\mathbf{C} = 1\text{nF/m}$, $\mathbf{R} = 1\Omega/\text{m}$, $\mathbf{G} = 5\text{mS/m}$. The parameters of the two coupled lines are

$$\begin{aligned}\mathbf{L} &= \begin{bmatrix} 600 & 50 \\ 50 & 600 \end{bmatrix} \text{nH/m}, & \mathbf{C} &= \begin{bmatrix} 1.2 & -.11 \\ -.11 & 1.2 \end{bmatrix} \text{nF/m}, \\ \mathbf{G} &= \begin{bmatrix} 7.5 & 0 \\ 0 & 7.5 \end{bmatrix} \text{mS/m}, & \mathbf{R} &= \begin{bmatrix} 2.25 & .225 \\ .225 & 2.25 \end{bmatrix} \Omega/\text{m}.\end{aligned}$$

The parameters of the four coupled lines are

$$\begin{aligned}\mathbf{L} &= \begin{bmatrix} 1 & .11 & .03 & 0 \\ .11 & 1 & .11 & .03 \\ .03 & .11 & 1 & .11 \\ 0 & .03 & .11 & 1 \end{bmatrix} \mu\text{H/m}, & \mathbf{C} &= \begin{bmatrix} 1.5 & -.17 & -.03 & 0 \\ -.17 & 1.5 & -.17 & -.03 \\ -.03 & -.17 & 1.5 & -.17 \\ 0 & -.03 & -.17 & 1.5 \end{bmatrix} \text{nF/m}, \\ \mathbf{G} &= \begin{bmatrix} 10 & 1 & .1 & 0 \\ 1 & 10 & 1 & .1 \\ .1 & 1 & 10 & 1 \\ 0 & .1 & 1 & 10 \end{bmatrix} \text{mS/m}, & \mathbf{R} &= \begin{bmatrix} 3.5 & .35 & .035 & 0 \\ .35 & 3.5 & .35 & .035 \\ .035 & .35 & 3.5 & .35 \\ 0 & .035 & .35 & 3.5 \end{bmatrix} \Omega/\text{m}.\end{aligned}$$

HSpice can be used to simulate the circuit when the transmission lines are lossless and uncoupled. In the lossless uncoupled case the transmission line resistance and conductance values are zero, and the inductance and capacitance matrices are diagonal.

2 of/de 2

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

Figure 5.11: A transmission line interconnect network with multiple nonlinear terminations.

Figure 5.12: Response at V_3 of the circuit shown in Figure 5.11.

Solution Step	Computer		Ideal Performance On Eight Sun 3/60's
	Sun SS1+	Sun 3/60	
Matrix Formulation	2.4	14.0	14.0/8
Transmission Line Eigenvalues	3.3	26.4	26.4/8
LU Decomposition	2.6	16.0	16.0/8
F/B Substitution	15.7	146.6	146.6/8
Piecewise Method	0.7	10.3	10.3
<i>Total (seconds)</i>	24.7	213.3	35.7

Table 5.1: Approximate CPU times for the coupled transmission line network of Figure 5.11. Input/Ouput processing time is not included.

The output waveforms for the lossless uncoupled case as calculated by HSpice and the piecewise method are shown in Figure 5.12. Also shown in Figure 5.12 is the output waveform for the lossy coupled case obtained using the piecewise method. The small oscillations before the large pulse are numerical artifacts similar to Gibbs phenomena and result from the fast rise time of pulse.

The CPU time taken to analyze the lossy coupled version of this circuit was 24.7 seconds on a Sun SS 1+. The run time distribution is shown in Table 5.6.4. The analysis used 11 complex frequency solutions to the linear equations at each of the 140 time points. As is clearly indicated by the figure most of the CPU time was taken by the time-domain solution of the linear subcircuits and very little by the piecewise method. Since by using numerical inversion of the Laplace transform, the evaluation of the time domain response of the linear subcircuit at time t_1 , is independent of the response at time t_2 , the relatively expensive step of computing the time-domain response of the linear subcircuit can be easily implemented on a system of parallel processors. The estimated savings in solution time when eight processors are used is shown in Table 5.6.4.

5.7 Summary

A piecewise linear decomposition method for the analysis of circuits containing both nonlinearities and transmission lines was presented. This method is an alternative to other solution methods which require iteration between the time and frequency domains. This method is one of the contributions of this thesis and has been published as [47]. Several examples were presented showing the application of the piecewise linear decomposition method. This method increases the accuracy of waveforms calculated using NILT and places no constraints on the parallelization of the NILT method.

Chapter 6

Conclusions

6.1 Summary

This thesis has contributed several new techniques for the analysis of high-speed interconnect circuits. These new techniques presented respectively in chapters three, four and five are:

1. Numerical inversion of the Laplace transform for transmission line structures.
2. Parallel evaluation of transient time domain circuit responses using numerical inversion of the Laplace transform.
3. A piecewise linear decomposition technique for the analysis of nonlinear circuits with linear subnetworks.

These techniques have been implemented and incorporated into a the SysCAD high-speed interconnect analysis framework providing additional operations such as optimization and monte-carlo analysis.

In addition to the main contributions this thesis includes many examples of the new techniques. The use of time domain superposition to improve the accuracy of numerical inversion of the Laplace transform is illustrated. Where possible the

example have been compared to the industry standard simulator HSpice. The ability of numerical inversion of the Laplace transform to independently calculate circuit response at different time points has been used to provide efficient location of threshold crossings and to efficiently perform waveform optimizations. The parallel extension to SysCAD applies to more than just numerical inversion of the Laplace and can also be used to execute frequency domain simulations in parallel.

6.2 Additional Research

Further research has already been based on some of the contributions presented here.

1. The piecewise decomposition technique has been used with asymptotic waveform evaluation of the linear circuit response instead of the use of NILT[48][36].
2. The piecewise decomposition technique has been extended to include sensitivity information[33] for use in performance optimization.
3. The analysis of nonuniform transmission lines[49][50] has been performed with nonlinear elements using the piecewise decomposition technique.

Additional work could be performed to increase the accuracy and utility of the system by including some of the extension techniques such as higher order methods or reinitialization into the NILT simulator. This work would include adapting the simulator to include all standard spice models.

A generalized parallel processing capability allowing parallel simulation, decomposition, optimization and statistical design should also be developed.

References

- [1] H. Hasegawa and S. Seki, "Analysis of interconnection delay on very high-speed LSI/VLSI chips using a microstrip line model," *IEEE Trans. on Electron Devices*, vol. ED-31, pp. 1954-1960, Dec. 1984.
- [2] A. R. Djordjević, T. K. Sarkar, and R. F. Harrington, "Time-domain response of multiconductor transmission lines," *Proceedings of the IEEE*, vol. 75, pp. 743-764, June 1987.
- [3] R. Sainati and T. Moravec, "Estimating high speed circuit interconnect performance," *IEEE Trans. on Circuits and Systems*, vol. CAS-36, pp. 533-541, Apr. 1989.
- [4] D. Zhou, F. Preparata, and S. Kang, "Interconnection delay in very high-speed VLSI," In *Proc. IEEE International Conference on Computer Design*, pp. 52-56, Oct. 1988.
- [5] S. S. Gao, A. Y. Yang, and S. M. Kang, "Modelling and simulation of interconnection delays and crosstalks in high-speed integrated circuits," *IEEE Trans. on Circuits and Systems*, vol. CAS-37, pp. 1-9, Jan. 1990.
- [6] V. Rissoli and A. Neri, "State of the art and present trends in nonlinear microwave CAD techniques," *IEEE Trans. Microwave Theory and Tech.*, vol. 36, pp. 343-365, Feb. 1988.

- [7] A. Matezrka and T. Kocprzak, "Computer calculation of large-signal GaAs FET amplifier characteristics," *IEEE Trans. Microwave Theory and Tech.*, vol. 33, pp. 129-135, Feb. 1985.
- [8] Meta-Software Inc., *HSPICE User's Manual*, 1990.
- [9] A. Hill and V. K. Tripathi, "An efficient algorithm for the three-dimensional analysis of passive microstrip components and discontinuities for microwave and millimeter-wave integrated circuits," *IEEE Trans. Microwave Theory and Tech.*, vol. 39, pp. 83-91, Jan. 1991.
- [10] EEsof Inc., *Microwave Spice Users Guide*, 1988.
- [11] M. Nakhla and J. Vlach, "A piecewise harmonic balance technique for determination of periodic response on nonlinear systems," *IEEE Trans. on Circuits and Systems*, vol. CAS-23, pp. 85-91, Feb. 1976.
- [12] K. S. Kundert and A. Sangiovanni-Vincentelli, "Simulation of nonlinear circuits in the frequency domain," *IEEE Trans. Computer Aided Design*, vol. CAD-5, pp. 521-535, Oct. 1986.
- [13] K. S. Kundert, G. B. Sorkin, and A. Sangiovanni-Vincentelli, "Applying harmonic balance to almost-periodic systems," *IEEE Trans. Microwave Theory and Tech.*, vol. 36, pp. 366-378, Feb. 1988.
- [14] M. Cases and D. Quinn, "Transient response of uniformly distributed rlc transmission lines," *IEEE Trans. on Circuits and Systems*, vol. CAS-27, pp. 200-207, Mar. 1980.
- [15] Y. Chang, "Transient analysis of lossless-coupled transmission lines in a nonhomogeneous dielectric medium," *IEEE Trans. Microwave Theory and Tech.*, vol. 18, pp. 616-626, Sept. 1987.

- [16] J. E. Schutt-Aine and R. Mitra, "Nonlinear transient analysis of coupled transmission lines," *IEEE Trans. on Circuits and Systems*, vol. CAS-36, pp. 959-967, July 1989.
- [17] A. J. Groudin, "Transient analysis of uniform resistive transmission lines in a homogeneous medium," *IBM Journal of Research Development*, vol. 23, pp. 675-681, Nov. 1979.
- [18] A. J. Groudin and C. S. Chang, "Coupled lossy transmission line characterization and simulation," *IBM Journal of Research Development*, vol. 25, pp. 25-41, Jan. 1981.
- [19] D. O. Pederson, "A historical review of circuit simulation," *IEEE Trans. on Circuits and Systems*, vol. , pp. 103-111, 1984.
- [20] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer Aided Design*, vol. 9, pp. 352-366, Apr. 1990.
- [21] D. Xie and M. S. Nakhla, "Delay and crosstalk simulation of high-speed VLSI interconnects with nonlinear terminations," *IEEE Trans. Computer Aided Design*, (accepted for publication).
- [22] Optimization System Associates Inc., *OSA90 users manual*, 1992.
- [23] K. Singhal and J. Vlach, "Computation of time domain response by numerical inversion of the Laplace transform," *Journal of the Franklin Institute*, vol. 299(2), pp. 109-126, 1975.
- [24] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, Van Nostrand Reinhold, 1983.
- [25] K. Singhal, J. Vlach, and M. Nakhla, "Absolutely stable, high order method for time domain solution of networks," *Archiv fur Elektronik und Uebertragungstechnik*, vol. 30, pp. 157-166, 1976.

- [26] L. Lu and M. S. Nakhla, "Time-domain analysis of distributed networks," In *Proc. IEEE International Conference on Circuits and Systems*, pp. 888–891, June 1991.
- [27] K. Singhal and J. Vlach, "Method for computing time responses of systems described by transfer functions," *Journal of the Franklin Institute*, vol. 311(2), pp. 123–130, Feb. 1981.
- [28] L. Lu, M. S. Nakhla, and Q. J. Zhang, "A stepping algorithm for transient analysis of distributed networks," In *Proc. IEEE International Symposium on Circuits and Systems*, pp. 1957–1960, June 1992.
- [29] R. Griffith and M. S. Nakhla, "Time-domain analysis of lossy coupled transmission lines," *IEEE Trans. Microwave Theory and Tech.*, vol. 38, pp. 1480–1487, Oct. 1990.
- [30] S. W. Director and R. A. Rohrer, "Automated network design – the frequency domain case," *IEEE Trans. on Circuit Theory*, vol. CT-16, pp. 330–337, Aug. 1969.
- [31] S. Lum, M. S. Nakhla, and Q. J. Zhang, "Sensitivity analysis of lossy coupled transmission lines," *IEEE Trans. Microwave Theory and Tech.*, vol. 39, pp. 2089–2099, Dec. 1991.
- [32] A. R. Djordjević and T. K. Sarkar, "Analysis of time response of lossy multiconductor transmission line networks," *IEEE Trans. Microwave Theory and Tech.*, vol. 35(10), pp. 898–908, Dec. 1987.
- [33] Q. J. Zhang, S. Lum, and M. S. Nakhla, "Minimization of delay and crosstalk in high-speed VLSI interconnects," *IEEE Trans. Microwave Theory and Tech.*, vol. 40, pp. 1555–1563, 1992.

- [34] J. W. Bandler, W. Kellerman, and K. Madsen, "A superlinearly convergent minimax algorithm for microwave circuit design," *IEEE Trans. Microwave Theory and Tech.*, vol. 33, pp. 1519–1530, 1985.
- [35] J. W. Bandler and Q. J. Zhang, "Optimization techniques for modelling, diagnosis and tuning," In T. Ozawa, editor, *Analog Methods for Computer-Aided Circuit Analysis and Diagnosis*. Marcel Dekker, 1988.
- [36] R. Griffith, E. Chiprout, Q. J. Zhang, and M. Nakhla, "A cad framework for simulation and optimization of high-speed VLSI interconnections," *IEEE Trans. on Circuits and Systems*, vol. 39(11), Nov. 1992.
- [37] C. W. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. on Circuits and Systems*, vol. CAS-22, pp. 504–509, Jun 1975.
- [38] G. Korn and T. Korn, *Mathematical Handbook for Scientists and Engineers*, McGraw-Hill, Inc., 1968.
- [39] E. Lelarasmee, A. E. Ruehli, and A. L. Sangiovanni-Vincentelli, "The waveform relaxation method for time domain analysis of large scale integrated circuits," *IEEE Trans. Computer Aided Design*, vol. 1, pp. 131–145, July 1982.
- [40] P. Defebve, J. Beetem, W. Donath, H. Y. Hsieh, F. Odeh, A. E. Ruehli, S. P. K. Wolff, and J. White, "A large-scale Mosfet circuit analyzer based on waveform relaxation," In *Proceedings of the International Conference on Computer Design*, Rye, NY, Oct. 1984.
- [41] C. F. Gerald, *Applied Numerical Analysis*, Addison-Wesley Publishing Company, second edition, 1980.
- [42] L. V. Kantorovich and G. P. Akilov, *Functional Analysis in Normed Spaces*, Pergammon, Oxford, U.K., 1964.

- [43] J. White and A. S. Vincentelli, *Relaxation techniques for the simulation of VLSI circuits*, Kluwer Academic, Norwell, MA, 1986.
- [44] J. White, R. Saleh, A. L. Sangiovanni-Vincentelli, and A. R. Newton, "Accelerating relaxation algorithms for circuit simulation using waveform-newton, step-size refinement and parallel techniques," In *Proc. IEEE International Conference on Computer-Aided Design (ICCAD)*, pp. 5-7, Santa Clara, CA, Nov. 1985.
- [45] R. Saleh, D. Webber, E. Xia, and A. L. Sangiovanni-Vincentelli, "Parallel waveform newton algorithms for circuit simulation," In *Proceedings of the International Conference on Computer Design*, pp. 660-668, Port Chester, NY, Oct. 1987.
- [46] R. Saleh and J. White, "Accelerating relaxation algorithms for circuit simulation using Waveform-Newton and step size refinement," *IEEE Trans. Computer Aided Design*, vol. CAD-9, pp. 951-958, Sept. 1990.
- [47] R. Griffith and M. S. Nakhla, "Mixed frequency/time domain analysis of nonlinear circuits," *IEEE Trans. Computer Aided Design*, vol. 10, pp. 1032-1043, Aug. 1992.
- [48] T. Tang, M. S. Nakhla, and R. Griffith, "Analysis of lossy multiconductor transmission lines using the asymptotic waveform evaluation technique," *IEEE Trans. Microwave Theory and Tech.*, vol. 39, pp. 2107-2116, Dec. 1991.
- [49] S. Manney, *Transient Analysis of Nonuniform High-Speed Interconnects*. PhD thesis, Carleton University, 1992.
- [50] S. Manney, M. Nakhla, and Q. J. Zhang, "Time domain analysis of nonuniform frequency dependent high-speed interconnect," In *Proc. IEEE International Conference on Computer-Aided Design (ICCAD)*, pp. 449-452, Nov. 1992.
- [51] The MATH WORKS Inc., *PRO-MATLAB Users guide*, 1989.

- [52] S. Wolfram, *Mathematica, A System for Doing Mathematics by Computer*, Addison Wesley, Inc., second edition, 1991.

Appendix A

Numerical Calculation of the Poles and Residues required for NILT

The following is a Matlab[51] function for evaluating the roots and residues used in NILT of order m, n .

```
function [ki, zi, err] = ilkz(m,n)
% [ki zi err] = ilkz(m,n)
% Approximate roots of Pade' expansion  $R_{n,m}(s)$  of  $\exp(s)$ 
% These are the  $k_i, z_i$  values for NILT
%   requires a factorial function fact()

for i=0:max( [ m n] ) % precalculate required factorials
    f(i+1) = fact(m+n - i);
end

for i=0:n
    % numerator
    pn(n+1-i) = f(i+1) * choose(n,i);
```

```

end

for i=0:m                % denominator
    qm(m+1-i) = (-1)^(i) * f(i+1) * chcose(m,i);
end

[ki zi err] = residue(pn,qm);

```

A collection of the roots and residues is produced by the following function. Note that $n = m - 2$ is used.

```

function somekizi(low,high)
% Produce the files zmat.M.N and kmat.M.N containing the roots and
% residues for NILT of all orders low to high.
format long e
format compact
for i=low:high,
    [ki zi err] = ilkz(i,i-2);
    eval(sprintf('diary zmat%g.%g',i,i-2));
    disp('xmatlab = {}')
    for j=1:2:i,
        disp(zi(j));
    end
    disp('}')';
    diary off
    eval(sprintf('diary kmat%g.%g',i,i-2));
    disp('kmatlab = {}')
    for j=1:2:i,
        disp(ki(j));
    end
end

```

```

end
disp('');
diary off
end

```

Matlab uses the natural numeric size of the computer it runs on and thus has limited precision. The numbers produced by Matlab were found to be inaccurate and Mathematica[52] was used to improve the precision and accuracy of the roots and residues.

The following definitions were used.

```

Pz := Sum[(m+n-j)! Binomial[n,j]z^j, {j,0,n}]
Qz := Sum[(-1)^j (m+n-j)! Binomial[m,j]z^j, {j,0,m}]

SetOptions[FindRoot,AccuracyGoal->20,WorkingPrecision->60,
MaxIterations->9999]

IlapRoots[xorig_] :=
Table[ FindRoot[Qz, {z, xorig[[i]]} ], {i,Length[xorig]} ];

IlapResidues[roots_] :=
Table[ Residue[Pz/Qz , {z, z /. roots[[i]]} ] , {i,Length[roots]}];

```

These definitions were then used with the values calculated using Matlab as illustrated below.

```

m=26
n=24
<< zmat26.24.m
roots = IlapRoots[xmatlab];

```

```
residues = IlapResidues[roots];  
roots >> roots.26.24  
2*residues >> residues.26.24
```

By changing the values in the `SetOptions` function Mathematica can be used to produce results to arbitrary accuracy and precision.

END

2010-07-94

FIN