List of Tables

6.1	Design specifications for the multiplexing circuit.	52
6.2	Elapsed time count for Example 1.	51
6.3	Design specifications for the distributed network.	61
6.4	Design parameters in the connection circuit before and after optimization for Example 2.	67
6.5	Design parameters of 5 subcircuits before and after optimization for Example 2.	68
6.6	Design parameters after overall optimization for Example 2.	69
6.7	Elapsed time count by parallel decomposition for Example 2.	70

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

Multilevel Optimization of High Speed VLSI Interconnect Networks by Decomposition

by

Yuji Wei, B. Sc.

This thesis is submitted to the Faculty of Graduate Studies and Research in partial
fulfillment of the requirements for the degree of

Master of Engineering

Ottawa-Carleton Institute of Electrical Engineering

Dept. of Electronics

Carleton University

Ottawa, Canada, K1S 5B6

Tel: (613) 788-5754

Fax: (613) 788-5708

Email: wei@doe.carleton.ca

November 10, 1993

©1993 Yuji Wei

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 ACCORDE UNE LICENCE
IRREVOCABLE ET NON EXCLUSIVE
PERMETTANT A LA BIBLIOTHEQUE
NATIONALE DU CANADA DE
REPRODUIRE, PRETER, DISTRIBUER
OU VENDRE DES COPIES DE SA
THESE DE QUELQUE MANIERE ET
SOUS QUELQUE FORME QUE CE SOIT
POUR METTRE DES EXEMPLAIRES DE
CETTE THESE A LA DISPOSITION DES
PERSONNE INTERESSEES.

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 PROPRIETE
DU DROIT D'AUTEUR QUI PROTEGE
SA THESE. NI LA THESE NI DES
EXTRAITS SUBSTANTIELS DE CELLE-
CI NE DOIVENT ETRE IMPRIMES OU
AUTREMENT REPRODUITS SANS SON
AUTORISATION.

ISBN 0-315-98591-7

Canada

Name

Yuji Wei

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.

Electrical Engineering

SUBJECT TERM

0544

SUBJECT CODE

U-M-I

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	0277
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	
General	0679
Ancient	0289
Linguistics	0290
Modern	0291
Literature	
General	0401
Classical	0294
Comparative	0295
Medieval	0297
Modern	0298
African	0316
American	0591
Asian	0305
Canadian (English)	0352
Canadian (French)	0355
English	0593
Germanic	0311
Latin American	0312
Middle Eastern	0315
Romance	0313
Slavic and East European	0314

PHILOSOPHY, RELIGION AND THEOLOGY

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

SOCIAL SCIENCES

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

Ancient	0579
Medieval	0581
Modern	0582
Black	0328
African	0331
Asia, Australia and Oceania	0332
Canadian	0334
European	0335
Latin American	0336
Middle Eastern	0333
United States	0337
History of Science	0585
Law	0398
Political Science	
General	0615
International Law and	
Relations	0616
Public Administration	0617
Recreation	0814
Social Work	0452
Sociology	
General	0626
Criminology and Penology	0627
Demography	0938
Ethnic and Racial Studies	0631
Individual and Family	
Studies	0628
Industrial and Labor	
Relations	0629
Public and Social Welfare	0630
Social Structure and	
Development	0700
Theory and Methods	0344
Transportation	0709
Urban and Regional Planning	0999
Women's Studies	0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture	
General	0473
Agronomy	0285
Animal Culture and	
Nutrition	0475
Animal Pathology	0476
Food Science and	
Technology	0359
Forestry and Wildlife	0478
Plant Culture	0479
Plant Pathology	0480
Plant Physiology	0817
Range Management	0777
Wood Technology	0746
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
Palynology	0427
Physical Geography	0368
Physical Oceanography	0415

HEALTH AND ENVIRONMENTAL SCIENCES

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

Speech Pathology	0460
Toxicology	0383
Home Economics	0386

PHYSICAL SCIENCES

Pure Sciences

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

Applied Sciences

Applied Mechanics	0346
Computer Science	0984

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

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
the Faculty of Graduate Studies and Research
acceptance of the thesis

**“Multilevel Optimization of High Speed VLSI Interconnect Networks by
Decomposition”**

submitted by Yuji Wei, B. Sc.,
in partial fulfillment of the requirements for
the degree of Master of Engineering

Q.J. Zhang, Thesis Supervisor

M.S. Nakhlá, Thesis Co-Supervisor

J. Wight, Chair, Department of Electronics

Abstract

As signal speeds increase, the effects of VLSI interconnects become a dominant factor limiting the performance of an overall VLSI system. The interconnect effect in a VLSI system exists at several levels and the number of interconnects in such a system can be very large. Design optimization of such a large distributed network involving signal integrity criteria and a large number of transmission line parameters is extremely challenging and remains unsolved.

In this thesis, a multilevel optimization technique is developed for large-scale and hierarchical optimization of high-speed VLSI interconnects modeled by distributed transmission lines. Mathematical programming decomposition is combined with network tearing where the overall network is optimized by a set of parallel suboptimizations. The technique takes advantage of VLSI interconnect hierarchies in PCB, MCM and IC designs to speed up the optimization process.

The convergence property of the technique is analyzed through Gauss-Seidel relaxation analysis between multiple sets of nonlinear equations which are derived from optimality conditions for the high- and the low-level optimizations.

The proposed multilevel optimization algorithm is designed and implemented using SYSCAD environment and involves multiple computer processes running in a network of workstations. Several time- and frequency-domain optimization examples are tested. Results show that the proposed multilevel optimization formulation is faster than the standard optimization.

Acknowledgment

I wish to thank my supervisor Dr. Q. J. Zhang for his invaluable advice and guidance during the course of my research and studies. I am also grateful for his patience in reviewing this thesis.

I would like to thank Dr. Michel Nakhla for his guidance and encouragement.

I am grateful to my wife, Sherry Xiuhua Zhang. Her patience and support are two most valuable things in my life.

At this time, I feel deeply indebted to my parents. I would like to express my deep appreciation for the encouragement and support they have provided through the years.

I deeply appreciate the fruitful discussions with my fellow colleagues Eli Chiprout, Donghui Xie, Sanjay Manney and Roulong Liu. Research engineers in the Department of Electronics are thanked for their help provided in research and computer systems.

The financial support through the Computer-Aided Engineering Chair funded by Bell-Northern Research and the Natural Sciences and Engineering Research Council of Canada is gratefully appreciated.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Thesis Objectives	2
1.3	Thesis Outline	3
2	Literature Review and Basic Concepts	5
2.1	Problem Overview	5
2.2	Partitioning Strategies	9
2.2.1	Functional Decomposition	10
2.2.2	Structural Decomposition	10
2.3	Coordination Techniques	11
2.3.1	Model Coordination	11
2.3.2	Goal Coordination	15
2.4	Illustration of Circuit Formulations by Goal Coordination	18
3	Formulations of Original Optimization	22
3.1	Formulation of Network Equations	22
3.2	Numerical Inversion of Laplace Transform	23
3.3	Formulation of Error Functions	24
3.4	Formulation of the Original Optimization Problem	28

4	Multilevel Formulation of Circuit Optimization and Its Parallel Implementation	29
4.1	Network Partitioning for Multilevel Formulation	29
4.2	Multilevel Formulation of Minimax Optimization	32
4.3	Parallel Multilevel Optimization and Implementation	34
5	Convergence Properties	38
5.1	Generalized Multilevel Optimization Algorithm	38
5.2	Convergence of Multilevel Optimization	39
5.3	Convergence of Two-Level Minimax Optimization	41
5.4	Example: An Intuitive Explanation of the Convergence Criteria . . .	45
6	Optimization Examples	48
6.1	Example 1: Frequency Domain Optimization of A Lumped Multiplexing Network	49
6.2	Example 2: Time Domain Optimization of A Distributed Network . .	58
6.3	CPU Analysis	66
7	SYSCAD: A Computer-Aided Design System Implementation	72
7.1	Introduction to SYSCAD	72
7.2	SYSCAD Module Descriptions	73
7.3	SYSCAD Overall File Structure	75
7.4	An Optimization Example Illustrating SYSCAD	77
8	Conclusions and Further Research	81
8.1	Summary and Contributions	81
8.2	Suggestions for Further Research	82

Appendix A	91
A.1 Definition of Matrix Norm	91
A.2 Definition of G-differentiable	92
A.3 Kuhn-Tucker Conditions	92

List of Figures

2.1	Example of coupled system.	12
2.2	Multilevel solution using model coordination	14
2.3	Decoupled system	15
2.4	Multilevel solution using goal coordination.	17
2.5	i th subcircuit block after partition.	19
3.1	Physical/geometrical parameters of a 2-conductor transmission line. .	25
4.1	Network partitioning.	30
4.2	Block diagram for implementation of parallel multilevel optimization. K computers are used to run the lower level suboptimizations in parallel.	35
6.1	Example 1. A lumped multiplexing network.	50
6.2	Response from the 1st subcircuit.	55
6.3	Response from the 2nd subcircuit.	56
6.4	Response from the 3rd subcircuit.	57
6.5	A distributed network for Example 2. Each subcircuit is shown in Fig.	
6.6.		59
6.6	A typical subcircuit in Fig. 6.5, $j = 48, 69, 90, 111, 132$	60
6.7	Trapezoidal signal used as excitations for the distributed network. . .	61
6.8	Responses at the outputs of 5 subcircuits before optimization.	64

6.9	Responses at the outputs of 5 subcircuits after optimization using mul- tilevel optimization.	65
6.10	Estimation of speed up ratio in parallel computing.	71
7.1	SYSCAD file structure.	76
7.2	A LC filter circuit.	78

List of Tables

6.1	Design specifications for the multiplexing circuit.	52
6.2	Elapsed time count for Example 1.	54
6.3	Design specifications for the distributed network.	61
6.4	Design parameters in the connection circuit before and after optimization for Example 2.	67
6.5	Design parameters of 5 subcircuits before and after optimization for Example 2.	68
6.6	Design parameters after overall optimization for Example 2.	69
6.7	Elapsed time count by parallel decomposition for Example 2.	70

List of Symbols

t :	time.
n_t :	number of time points.
ω :	frequency.
n_ω :	number of frequencies.
ψ :	independent parameter, i.e. frequency, time ... etc.
K :	number of subcircuits.
N :	number of nodes of an overall circuit.
$\mathbf{f}$:	general non-linear functions.
$\mathbf{V}$:	node voltage vector of an overall circuit.
$\mathbf{V}_c$:	node voltage vector in the connection subnetwork.
$\mathbf{X}$:	vector of all optimization variables.
$\mathbf{X}_c$:	vector of all variables in the connection subnetwork.
λ, μ :	vectors of coordinating factors.
P :	objective function.
m :	number of error functions of an overall circuit.
e :	error functions.
w :	weighting factor.
N_{max} :	number of maximal decomposition loops.
c :	index for connection circuit.
IL :	insertion loss.

C :	capacitor.
L :	inductor.
R :	resistor.
g :	transconductance.
l :	length of transmission lines.
S :	specifications.
$T_{overall}$:	time for solving an overall optimization.
N_{cycle} :	total number of cycles between upper and lower level optimizations.
$R_{speedup}$:	speed up ratio.
$T_{overhead}$:	overhead time, it is the time for communications between lower level optimization and upper level optimization.
N_{update} :	number of variable updates in optimization.
N_{optvar}^c :	number of optimization variables in the connection circuit.
N_{optvar} :	total number of optimization variables.
$N_{processor}$:	number of processors for parallel computing.
N_{demp}^c :	computation effort needed for optimizing connection subcircuit.
N_{demp}^i :	computation effort needed for optimizing the subcircuit i.
N_{demp} :	computation effort for optimization by decomposition.
$N_{overall}$:	computation effort for overall optimization.

Chapter 1

Introduction

1.1 Motivation

As the signal speeds increase, the effects of VLSI interconnects become a dominant factor limiting the performance of an overall VLSI system [1]. Improperly designed interconnects can result in increased signal delay, ringing, reflection and false switching. Such signal integrity issues must be dealt with in the design stage of a high-speed VLSI system. In the past several years, there has been a thrust of research in the simulation of interconnects modeled by lossy coupled transmission lines, e.g., [1]-[5]. Design optimization of such interconnects is also recently addressed [6]-[7].

The interconnect effect in a VLSI system exists at several levels, e.g., printed circuit board (PCB) and multichip modules (MCM). The number of interconnects in such a system can be very large. Signal integrity of the overall digital system is affected by the interconnects at all the levels. However, design optimization of such a large distributed network involving signal integrity criteria and a large number of transmission line parameters is extremely challenging and remains unsolved.

A traditional mathematically based vehicle for approaching large scale problems

is decomposition, e.g., in mathematical programming [10]-[15], circuit analysis [16]-[22], control system design [23]-[26], fault diagnosis [27], state estimation [28], power systems optimization [29]-[31] and operation planning [32] [33]. An essential feature for decomposition based optimization is the coordination scheme between various suboptimizations such as model coordination, goal coordination and combined model-goal coordination [24].

Decomposition for the purpose of circuit optimization has its own special requirements due to factors, such as, the high nonlinearity and nonideal structures in objective functions, and nonideal distinctions between inputs and outputs of decomposed problems [34]. Bandler and Zhang [34] studied large scale optimization of microwave circuits using automatic decomposition based on sensitivity information. Low and Director [35] used model coordination and a functional decomposition concept for design centering of IC fabrication process.

However these techniques are not directly suitable for the VLSI interconnect problem where a large and hierarchical network topology adds to the complexity of large numbers of optimization variables and functions. A topologically oriented approach to optimization decomposition is needed [8] [9].

1.2 Thesis Objectives

The first objective of the thesis is to develop a method for solving large scale circuit optimizations taking interconnect effects into account. New formulations are to be derived based on the decomposition theory and circuit optimization theory. The optimization must efficiently handle interconnect effects at several levels, i.e., at the PCB and MCM level. The optimization should be capable of processing large number of interconnects. This technique should improve computation speed of large scale circuit optimizations substantially.

The solution of the optimization problem proceeds in an iterative fashion with the effort divided among different suboptimizations. Thus the convergence property of such an iterative process needs to be addressed. The second objective of this thesis is to obtain theoretical conditions for convergence for the proposed optimization process.

The third objective is to incorporate parallel processing concept into large-scale optimization. A computer CAD framework needs to be implemented to facilitate the parallel computation at the sub-optimization level.

Finally, the formulations are to be applied to solve high-speed interconnect optimization problems. Examples for signal integrity optimization will be tested. The CPU speed up analysis will be provided for the proposed optimization.

1.3 Thesis Outline

A literature review is presented in **Chapter 2**. The applications of decomposition to electronics engineering , mathematical programming and electrical power systems are studied. Two basic coordination techniques are described. An illustration of circuit decomposition based on goal coordination is presented.

In **Chapter 3**, the original signal integrity optimization is formulated. The design variables and error functions in high-speed interconnect networks are discussed.

A general decomposition formulation in circuit optimization is presented in **Chapter 4**. Firstly, a multilevel optimization approach based on decomposing both the network topology and optimization is proposed. Minimax optimization is formulated as a special case. Finally, a parallel implementation for the multilevel optimization problem as a result of decomposition is presented.

The convergence property is presented in **Chapter 5**. The optimization convergence problem is analyzed through Gauss-Seidel relaxation analysis between multiple

sets of non-linear equations which are derived from the optimality conditions for multiple suboptimizations. In order to derive conditions for convergence, the notion of strict diagonal dominance in relaxation analysis is extended to blockwise diagonal dominance.

In **Chapter 6**, two examples, frequency- and time-domain optimizations involving interconnect transmission lines, are presented illustrating the advantages of the proposed decomposition approach.

Chapter 7 discusses the computer implementations. SYSCAD, an open-architecture CAD system, is introduced. SYSCAD employs the UNIX interprocess communication features to facilitate the connection of different program modules. The proposed method of decomposition is implemented under the SYSCAD environment.

The conclusions and future research directions are presented in **Chapter 8**.

Chapter 2

Literature Review and Basic Concepts

2.1 Problem Overview

With the advent of new devices having clock rate, or rise/fall times, in the order of subnanoseconds and with significant increases in the level of integration found in most VLSI systems, the problem of crosstalk between neighboring lines due to electromagnetic coupling and reflections due to mismatch and incorrect line terminations have become prominent in VLSI circuit and systems. In order to properly evaluate electrical interconnect effects, it is necessary to use complex simulators that are designed for high-speed interconnect analysis. The added complexity and the fact that typical VLSI systems and multichip modules contain a large number of interconnections can result in analysis and optimization problems that require a lot of computing power.

In order to improve signal integrity in high-speed circuits, a circuit designer can use automated design techniques to optimize the time responses to meet specifications under a given set of design constraints. However, performing optimization, which

requires repeated analysis runs, requires even more computing power.

A solution to the CPU intensive problem is to divide the problem over a number of processors and complete the computation tasks in parallel [37]. Large-scale optimization has been approached by decomposing the large number of optimization variables and functions. In [34], a novel and general decomposition technique applicable to the optimization of large microwave systems has been proposed. Using sensitivity information obtained from a suitable Monte Carlo analysis, possible decomposition properties are extracted. The overall optimization problem is automatically separated into a sequence of subproblems, each being characterized by the optimization of a subset of circuit functions with respect to variables which are sensitive to the selected responses. Good starting points and savings in computation time and memory space were achieved. Low and Director [35] reformulated the yield maximization problem into a deterministic design centering problem. Problem decomposition and macromodelling were then applied to solve the design centering problem efficiently. Furthermore, by applying the decomposition technique, the yield maximization technique became efficient enough so as not to be limited to low dimensional problems. Similar hierarchical techniques have been applied in logic circuit optimization as well. Malik [38] described a novel method for applying the two-level minimization for the optimization of primitive gate networks. Instead of actually breaking up a network into smaller, disjoint subnetworks, optimizing them separately and splicing them back afterwards, Limqueco and Muroga [39] partitioned the network into overlapping regions and used a varying size window to sweep through these regions, each time making only the gates within a region visible to the logic optimizing algorithm.

In dealing with large-scale networks and systems, many partitioning algorithms have been proposed. An efficient heuristic algorithm for solving a cluster problem using the concept of a contour tableau is proposed in [16]. The problem of decomposing a network into different groups of electrical components, which are interconnected by

the minimum number of leads, is discussed in [17]. A density of a graph is defined as the maximum number of disjoint spanning trees of the graph. Using this density, an algorithm to find all the subgraphs of a graph with a given density is developed in [40].

Unfortunately, there are reasons to believe [16] that all cluster partitioning problems belong to a class of hard problems, the so-called NP-complete class, where no polynomial-bounded global solutions are likely to exist. Therefore, any algorithm in such an attempt is heuristic.

Relaxation technique, which is related to decomposition technique, have also been proposed in the simulation of large integrated circuits. A broad survey of this decomposition related technique in circuit simulation can be found in [41, 42]. In the survey, the techniques used in relaxation-based electrical simulation were presented in a rigorous and unified framework and the numerical properties of the various methods were explored. Nakano *et al* [43] described a method of implementing waveform relaxation for bipolar digital circuits. In [44], partitioning techniques for Asymptotic Waveform Evaluation (AWE) were proposed to allow AWE to handle circuits in partitioned form. A multilevel Newton algorithm based on macromodels was presented in [45].

As far as transmission line effects are concerned, a bi-level waveform relaxation method is developed to compute the transient response of a coupled, dispersive multiconductor system terminated in non-linear loads such as transistors [46]. The transient response of lossy coupled transmission lines is also simulated by iterative waveform relaxation analyses of equivalent disjoint networks constructed with congruence transformers, FFT waveform generators, and characteristic impedances synthesized by Padé approximation [47].

In solving a large electrical power network, a special decomposition technique known as diakoptics was found by Kron [18] and Happ [19, 20]. The basic idea of

diakoptics is to solve a large system by breaking or tearing it apart into smaller subsystems; solving the individual parts, and then combining and modifying the solutions of the torn parts to yield the solution of the original untorn problem. In [22], tearing is explained simply in terms of elementary concepts from linear circuit theory.

Decomposition techniques are applied in power system design and optimization [32, 33]. The objectives of power system planning are mainly to minimize cost, which can be expressed in simple mathematical functions. Multilevel optimizations are often employed to reduce the size of the problem and computation time requirement.

The concept of decomposition has been a traditional, mathematically based vehicle for approaching large-scale problems. A major inspiration in the field of mathematical programming has been the discovery of the decomposition principle by Danzig and Wolfe [10]. Decomposition techniques have found wide applications in control systems [24, 25]. Acar and Ozguner [26] discussed the optimal control method for a class of interconnected systems. More recently, an optimal control problem [12] was presented, in which the problem is decomposed by using time as an interaction variable and iterated towards a solution. Efforts are made on efficient methods for extracting derivative information on the bi-level optimization [13]. Some popular special decomposition techniques [14] are generalized [15].

Despite all these efforts in decomposition, little has been introduced in circuit optimization. Decomposition for the purpose of circuit optimization has its own special requirements due to the high nonideal structures in objective functions, and nonideal distinctions between inputs and outputs of decomposed problems [34]. The objective functions in circuit optimization are highly nonlinear and the connections between circuits are usually strongly coupled. Most techniques developed so far are not directly suitable for the VLSI interconnect problem where a large and hierarchical network topology adds to the complexity of large numbers of optimization variables

and functions. A topologically oriented approach to optimization decomposition is needed to achieve the fast and correct design of high speed VLSI interconnects.

2.2 Partitioning Strategies

In order to apply decomposition techniques, the first step is to divide a large scale system into many small ones [36]. There are many proposed methods for circuit partitioning. Theoretically, we would like to decompose a circuit into a minimally interconnected circuit. In this case the total number of connections between any two subcircuits should be minimized. The computational time required by such an algorithm increases exponentially with the number of subcircuits [16]. Consequently, Sangiovanni-Vincentelli *et al* [16, 17] adopted a modified minimal cut heuristics. An almost linear time complexity partition algorithm was described in [36]. In VLSI design and optimization it is logical that a large circuit block be designed using many relatively small and independent circuit blocks. These design blocks suggest a natural circuit partitioning. In solving a large circuit optimization problem, the natural partitioning can also be exploited.

Generally, there are two major strategies in partitioning a large circuit optimization problem. The first approach, called functional decomposition, is based on partitioning the objective and constraint functions into a number of disjoint sets. The second one, structural decomposition, involves the partitioning of the underlying physical circuit into a number of subcircuits. The two partitioning strategies can be employed in conjunction with one another. For instance, one can first apply structural decomposition to break a large circuit into a number of subcircuits and then use functional decomposition on the subproblem associated with each of the subcircuits.

2.2.1 Functional Decomposition

The central idea of functional decomposition is to partition the objective and constraint functions into a number of disjoint groups according to the dependencies on different (not necessarily disjoint) sets of variables. The purpose of performing such a partitioning is to decompose the original problem into a number of subproblems, each corresponding to a group of objectives and constraint functions. The overall solution is reached by solving a sequence of subproblems.

Suppose sets I and J are defined as

$$I \equiv \{1, 2, \dots, n\}, \quad (2.1)$$

$$J \equiv \{1, 2, \dots, m\}. \quad (2.2)$$

The overall optimization problem, e.g., a minimax optimization, is

$$\min_{X_i, i \in I} \max_{j \in J} f_j(X). \quad (2.3)$$

In a functional decomposition approach, one attempts to reach the overall solution by solving a sequence of subproblems. A subproblem is characterized by

$$\min_{X_i, i \in I_i} \max_{j \in J_i} f_j(X) \quad (2.4)$$

where I_i and J_i are subsets of I and J , respectively. The basic idea of partitioning is to decouple a variable X_i from a function f_j if the interaction between them is weak. The partitioning of X and f can be achieved either manually or automatically. The manual procedure corresponds to the manual determination of variables and function groups using *a priori* knowledge. The automatic procedure corresponds to the computerized partitioning of X and f based on sensitivity information [34].

2.2.2 Structural Decomposition

In structural decomposition, one attempts to partition the optimization problem by exploiting the structure of the underlying physical system. There are two aspects

related to such a decomposition: one is to deal with the partition of the physical circuit and the second is concerned with the decomposition of the objective and constraint functions after circuit partitioning has been achieved.

Structural decomposition is based on the observation that it is usually possible to physically partition a large circuit into a number of loosely coupled subcircuits. The partitioning may be a natural consequence of the hierarchical design practice or may be obtained through the application of a partitioning algorithm. Circuit partitioning is mathematically identical to functional decomposition above. Here, subcircuits may be individual transistors, logic gates, or functional blocks. The aim of structural decomposition is to find a partition that minimizes the interactions among the subcircuits. For large scale VLSI, it is often reasonable to simply use the partitioning which already exists in the design. Such a partitioning usually provides subcircuits with relatively loose couplings because it is a primary requirement for modular design.

2.3 Coordination Techniques

Although there are many ways to decompose a given optimization problem into a multilevel problem, they are essentially the combinations of two approaches, which are termed as *model coordination method*, and *goal coordination method* [11]. These two techniques are illustrated by means of a simple example involving two coupled subsystems.

2.3.1 Model Coordination

Consider the optimization problem associated with the system diagram in Fig. 2.1. The variables in the system are described below:

$\mathbf{x}$ = vector of manipulated variables.

Figure 2.1: Example of coupled system.

x^i = vector of manipulated variables for subsystem i .

y = vector of output variables for the system.

y^i = vector of output variables for subsystem i .

u^1 = vector of interaction variables from subsystem 1 to subsystem 2.

u^2 = vector of interaction variables from subsystem 2 to subsystem 1.

Let the generalized system equations be

$$G(x, y, u) = 0 \quad (2.5)$$

where the vector G contains the system equations for each subsystem

$$G^i(x^i, y^i, u^1, u^2) = 0. \quad (2.6)$$

Let the objective function which is to be minimized be

$$P(x, y, u) = P_1(x^1, y^1, u^1) + P_2(x^2, y^2, u^2). \quad (2.7)$$

Notice that the objective function has been selected such that the objective function of the overall system is the sum of the two subsystem objective functions. This

additive feature is a common assumption in mathematical programming. The functions $P_1(\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1)$ and $P_2(\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2)$ are different from $P(\mathbf{x}, \mathbf{y}, \mathbf{u})$.

In case of circuit optimization, the above mentioned manipulative variables and output variables correspond to the design variables and output voltages or currents, respectively. The system equation is the equation which will be described in Chapter 3.

The optimization problem is

$$\min_{\mathbf{x}, \mathbf{y}, \mathbf{u}} P(\mathbf{x}, \mathbf{y}, \mathbf{u}) \quad (2.8)$$

subject to

$$\mathbf{G}(\mathbf{x}, \mathbf{y}, \mathbf{u}) = \mathbf{0}. \quad (2.9)$$

Although the objective function may be separated into two noninteracting functions, one for each subsystem, the interaction variable u affects both subsystems. The model coordination method converts this overall optimization problem into a two-level problem by fixing the interactive variable u at some value $\mathbf{z}$. The above problem can be reformulated into the following two-level problem.

Higher level problem is

$$\min_{\mathbf{z}} H(\mathbf{z}). \quad (2.10)$$

Lower level problem is

$$H(\mathbf{z}) = \min_{\mathbf{x}, \mathbf{y}} P(\mathbf{x}, \mathbf{y}, \mathbf{z}) \quad (2.11)$$

subject to

$$\mathbf{G}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \mathbf{0}. \quad (2.12)$$

Notice in the lower level optimization

$$H(\mathbf{z}) = \min_{\mathbf{x}, \mathbf{y}} P(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \min_{\mathbf{x}, \mathbf{y}} \sum_{i=1,2} P_i(\mathbf{x}^i, \mathbf{y}^i, \mathbf{z})$$

Figure 2.2: Multilevel solution using model coordination

and

$$G(x, y, z) = 0 \Leftrightarrow G_i(x^i, y^i, z) = 0, i = 1, 2. \quad (2.13)$$

The lower level optimization is further decomposed into two small optimizations. Furthermore, as there are no interactions between the two subsystems, the lower level optimizations can be solved in parallel. The resulting multilevel solution to this optimization problem is diagrammed in Fig. 2.2.

The important generalization is that the lower level problem was created by fixing certain variables, e.g., interaction variables, in the original problem and assigning to the higher level the task for determining these variables.

Figure 2.3: Decoupled system

2.3.2 Goal Coordination

The two coupled subsystems in Fig. 2.1 are considered. The goal coordination method literally removes the interactions by “cutting” all links between subsystems. This may be done as indicated in Fig. 2.3 where the outputs of the subsystems which are the inputs to the another system are labeled u^i as before, but the corresponding inputs to the subsystem are labeled as z^i . When the coupled system is cut into subsystems, z^i and x^i need not to be equal. Moreover u^i now acts like a manipulative variable and must be selected. This decouples the two subsystems completely. Since the objective function was already decoupled, it is clear that there is no interaction in the system at all. However, in order to ensure that the independent subsystem problem yields the overall system optimality, it is necessary to ensure that the independent variables x^i and z^i are actually equal.

The addition of a penalty term is considered to ensure the interactions are properly

balanced. The objective function is

$$P(\mathbf{x}, \mathbf{y}, \mathbf{u}, \lambda) = P_1(\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1) + P_2(\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2) + \lambda^T(\mathbf{u} - \mathbf{z}). \quad (2.14)$$

The constraints are

$$\mathbf{G}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{u}) = \mathbf{0} \Leftrightarrow \begin{cases} \mathbf{G}_i(\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1, \mathbf{z}^2) = \mathbf{0} \\ \mathbf{G}_i(\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2, \mathbf{z}^1) = \mathbf{0} \end{cases} \quad (2.15)$$

where λ is a Lagrange multiplier or coordinating factor which penalizes the performance of the system if interactions do not balance.

The problem can be treated as a two level optimization problem. Higher level problem is assigned the task of balancing the interaction variables by manipulating λ . When considering the lower level optimization, the penalty term is reorganized.

Lower level problems are formulated as

subsystem 1

$$\min_{\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1, \mathbf{z}^2} P_1(\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1) + \lambda_1^T \mathbf{u}^1 - \lambda_2^T \mathbf{z}^2 \quad (2.16)$$

subject to

$$\mathbf{G}_1(\mathbf{x}^1, \mathbf{y}^1, \mathbf{u}^1, \mathbf{z}^2) = \mathbf{0}; \quad (2.17)$$

subsystem 2

$$\min_{\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2, \mathbf{z}^1} P_2(\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2) - \lambda_1^T \mathbf{z}^1 + \lambda_2^T \mathbf{u}^2 \quad (2.18)$$

subject to

$$\mathbf{G}_2(\mathbf{x}^2, \mathbf{y}^2, \mathbf{u}^2, \mathbf{z}^1) = \mathbf{0}. \quad (2.19)$$

A block diagram representing the two level problem is in Fig. 2.4.

The multilevel formulation then involves cutting the interacting variables in order to create a lower level problem which can easily be decomposed into independent

Figure 2.4: Multilevel solution using goal coordination.

subproblems. The higher level attempts to force the lower level subproblems to arrive at a solution for which the independent variables x_i and z_i are equal.

Generally, if the overall optimization is formulated by including the constraints through the use of Lagrange multipliers or penalty terms, conversion of a multilevel form is done by

1. Choosing a set of coordinating variables(actual variables, Lagrange multipliers, penalty weights, or any combination of these)and assuming these to be fixed, thereby reproducing a lower level optimization problem with certain fixed variables.
2. Assigning to the higher level the task of determining the optimal values of the coordinating variables.
3. Deriving an algorithm by which the higher level optimization may successively

improve its estimates of the optimal interaction variables.

The choice of coordinating variables is made on the basis of decomposition of the lower level problem. A set of coordinating variables must be selected so that the resulting lower level problem, which is the original problem with coordinating variables assumed fixed, can be decomposed into independent problems.

2.4 Illustration of Circuit Formulations by Goal Coordination

In this section, a goal coordination formulation for circuit optimization is illustrated.

Suppose that a circuit is decomposed into K blocks by partitioning. An overall optimization is

$$\min_{\mathbf{X}} P(\mathbf{X}) \quad (2.20)$$

subject to

$$\mathbf{YV} = \mathbf{I} \quad (2.21)$$

where $\mathbf{X}$ is the vector of all design variables and $\mathbf{Y}, \mathbf{V}, \mathbf{I}$ are the admittance matrix, voltage source vector and current source vector, respectively. Suppose the circuit is decomposed into K subcircuits, the matrices $\mathbf{Y}$, $\mathbf{V}$ and $\mathbf{I}$ can be partitioned accordingly.

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_{11} & \mathbf{Y}_{12} & \cdots & \mathbf{Y}_{1K} \\ \mathbf{Y}_{21} & \mathbf{Y}_{22} & \cdots & \mathbf{Y}_{2K} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{Y}_{K1} & \mathbf{Y}_{K2} & \cdots & \mathbf{Y}_{KK} \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_K \end{bmatrix}, \quad \mathbf{I} = \begin{bmatrix} \mathbf{I}_1 \\ \mathbf{I}_2 \\ \vdots \\ \mathbf{I}_K \end{bmatrix}. \quad (2.22)$$

Figure 2.5: i th subcircuit block after partition.

Voltages and currents in the i th circuit block are represented by $\mathbf{V}_i$ and $\mathbf{I}_i$, respectively.

An example block is shown in Fig. 2.5. The following notations are used to formulate the goal coordination. Let $\mathbf{V}_i^{(e)}$ be the external voltage vector of the subcircuit i , in the diagram $\mathbf{V}_i^{(e)} = [V_{i1}, V_{i2}]^T$. Let $\mathbf{I}_i^{(e)}$ be the external current vector. Let nc_{ij} be the number of connections between subcircuits i and j . Let $n_i^{(e)}$ be the number of external nodes in subcircuit i . Let n_i be the number of nodes in subcircuit i . Let $\mathbf{C}_{ij}$ be the selection matrix, its dimension is $nc_{ij} \times n_j^{(e)}$. Let $\mathbf{C}_{ij}(k, l) = 1$, if the l th external node of subcircuit j is connected with the k th node of the subcircuit i where $k = 1, 2, \dots, nc_{ij}$, $l = 1, 2, \dots, n_j^{(e)}$. The i th circuit block can be represented by the following equation:

$$\mathbf{Y}_i^{(e)}(\mathbf{X}_i)\mathbf{V}_i^{(e)} = \mathbf{I}_i^{(e)} \quad (2.23)$$

where $\mathbf{Y}_i^{(e)}(\mathbf{X}_i)$ is the reduced nodal admittance matrix of the i th circuit block. Variables in subcircuit i are contained in $\mathbf{X}_i$.

The higher level optimization is

$$\begin{aligned} \max_{\lambda, \mu} H(\lambda, \mu) = & \sum_{i=1}^K P_i(\mathbf{X}_i) + \\ & \sum_{i=1}^{K-1} \sum_{j=i+1}^K \lambda_{ij}^T (\mathbf{C}_{ji} \mathbf{V}_i^{(e)} - \mathbf{C}_{ij} \mathbf{V}_j^{(e)}) \end{aligned}$$

$$+ \sum_{i=1}^{K-1} \sum_{j=i+1}^K \mu_{ij}^T (\mathbf{C}_{ji} \mathbf{I}_i^{(e)} + \mathbf{C}_{ij} \mathbf{I}_j^{(e)}) \quad (2.24)$$

Where λ, μ are $K \times K$ matrices of the Lagrange multipliers. They are represented by

$$\lambda = \begin{bmatrix} 0 & \lambda_{12} & \lambda_{13} & \cdots & \lambda_{1k} \\ 0 & 0 & \lambda_{23} & \cdots & \lambda_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \lambda_{k-1k} \end{bmatrix}, \quad (2.25)$$

and

$$\mu = \begin{bmatrix} 0 & \mu_{12} & \mu_{13} & \cdots & \mu_{1k} \\ 0 & 0 & \mu_{23} & \cdots & \mu_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \mu_{k-1k} \end{bmatrix} \quad (2.26)$$

where each λ_{ij} and μ_{ij} are $nc_{ij} \times 1$ vectors. Both matrices are upper triangle type. The total number of optimization variables is $2 \times \sum_{i=1}^{n-1} \sum_{j=i+1}^n nc_{ij} = 2 \times$ (number of connections).

The lower level optimization is

$$\begin{aligned} \min_{\mathbf{X}_i} P_i(\mathbf{X}_i) + \sum_{j=i+1}^K \lambda_{ij}^T \mathbf{C}_{ji} \mathbf{V}_i^{(e)} - \sum_{j=1}^{i-1} \lambda_{ji}^T \mathbf{C}_{ji} \mathbf{V}_i^{(e)} \\ - \sum_{j=i+1}^K \mu_{ij}^T \mathbf{C}_{ji} \mathbf{I}_i^{(e)} + \sum_{j=1}^{i-1} \mu_{ij}^T \mathbf{C}_{ji} \mathbf{I}_i^{(e)} \end{aligned} \quad (2.27)$$

subject to

$$\mathbf{Y}_i^{(e)}(\mathbf{X}_i) \mathbf{V}_i^{(e)} = \mathbf{I}_i^{(e)}. \quad (2.28)$$

It can be shown that the sensitivities of H with respect to the coordination variables λ_{ij}, μ_{ij} are

$$\frac{\partial H}{\partial \lambda_{ij}} = \mathbf{C}_{ji} \mathbf{V}_i^{(e)} - \mathbf{C}_{ij} \mathbf{V}_j^{(e)}, \quad (2.29)$$

$$\frac{\partial H}{\partial \mu_{ij}} = \mathbf{C}_{ji} \mathbf{I}_i^{(e)} + \mathbf{C}_{ij} \mathbf{I}_j^{(e)}. \quad (2.30)$$

From the above the equation, it can be seen that the balance of the interactions is dependent on the upper level optimization. A logical explanation of the above equation is that the upper level adjusts the coordination variables to reach a maximal point. The maximization process reduces the differences between the interactions. At last, when the upper level optimization is finished, the sensitivities in (2.29) and (2.30) become zero, forcing the consistency of voltages and currents between subcircuits.

The original goal coordination concept was laid by Danzig and Wolf in decomposing a linear programming problem[10]. Goal coordination has been applied to power systems[32] and environmental systems [23], where the objective function is easy to decompose. In circuit optimization, the responses are much more complicated and highly nonlinear which makes it difficult to decompose a circuit and to coordinate optimizations at different levels.

Equations (2.29) and (2.30) show that the lower level optimizations should get exactly local minima, that is, their sensitivities have to be equal to zero. This is a very stringent requirement for optimization algorithms. In the later formulation, we take a different approach, which employs the concepts of model coordination. A numerically stable algorithm is achieved.

Chapter 3

Formulations of Original Optimization

3.1 Formulation of Network Equations

Suppose the network π consists of lumped elements and N_s multiconductor transmission lines. The modified nodal equations [48] for the overall network π are

$$\mathbf{C} \frac{d\mathbf{V}}{dt} + \mathbf{G}\mathbf{V}(t) + \sum_{k=1}^{N_s} \mathbf{D}_k \mathbf{i}_k(t) = \mathcal{E}(t) \quad (3.1)$$

where $\mathbf{C}$ and $\mathbf{G}$ are $N \times N$ matrices determined by lumped elements in the network. $\mathbf{V}(t)$ is the vector of node voltage waveforms appended by independent voltage source current and inductor current waveforms. $\mathbf{D}_k$ is an incidence matrix containing 1's and 0's which maps $\mathbf{i}_k(t)$, the terminal current waveforms of the k th distributed transmission line, into the N -node space of the network π . $\mathcal{E}(t)$ is the vector of source waveforms. The distributed interconnect models are represented by $\mathbf{i}_k(t)$ in the network equation. For quasi-TEM model, an eigenvalue solution is required for the transmission line elements. Techniques for solving such distributed networks are

based on, e.g., FFT [3], NILT[4] or AWE [5].

In the presence of nonlinear terminations in the interconnect network, techniques such as piecewise decomposition [5] or macromodelling [2] may be used. In this case the frequency domain models for the transmission lines are converted into time domain macromodels by NILT or AWE and then substituted into the overall network equations. Let $\mathbf{X}$ be a vector of network parameters. The general equation for a VLSI interconnect network is equivalently described by

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{f}(\mathbf{V}, \mathbf{X}, t). \quad (3.2)$$

3.2 Numerical Inversion of Laplace Transform

In the thesis, the method used for simulating transmission lines in the time-domain is based on the Numerical Inversion of Laplace Transform (NILT) [4].

The s -domain equation of the circuit is obtained by taking Laplace transform of (3.1)

$$\mathbf{Y}_\pi \mathbf{V}_\pi(s) = \mathcal{E}_\pi(s) + \mathbf{C}_\pi \mathbf{V}_\pi(0) \quad (3.3)$$

where

$$\mathbf{Y}_\pi = \mathbf{G}_\pi + s\mathbf{C}_\pi + \sum_{i=1}^{N_g} \sum_{k \in I_i} \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^T \quad (3.4)$$

where T denotes tranpose, $\mathbf{A}_k$ is the nodal admittance matrix of the k th distributed transmission line and $I_i = \{k | \text{if transmission line } k \text{ belongs to group } i, i = 1, 2, \dots, N_g\}$.

The circuit is solved in the s -domain to obtain nodal voltages $\mathbf{V}_\pi$. The transient voltages $\mathbf{v}_\pi(t)$ are obtained through numerical Laplace inversion, i.e.,

$$v_\pi(t) = -(1/t) \sum_{i=1}^{M'} \text{Real}[K_i V_\pi(z_i/t)] \quad (3.5)$$

where z_i and K_i , $i = 1, 2, \dots, M'$, are predetermined poles and residues of a Padé rationale function approximating e^z [4].

From Eq. (3.5), it can be seen that the response at time t can be obtained without calculating the responses at any other time points. This is particularly well suited to design optimization since specifications are typically imposed on time subintervals.

3.3 Formulation of Error Functions

In order to formulate optimization, optimization variables and a set of error functions need to be defined. Such a problem has been addressed by Zhang, Lun and Nakhla in [6]. A brief summary of error functions is given here in order to facilitate the description of the new multilevel optimization. Consider a VLSI interconnect network excited by a trapezoidal signal. Let $\mathbf{X}$ be a vector of design variables. Such variables can include physical/geometrical parameters of interconnects and parameters in termination/matching circuits. Fig. 3.1 shows the parameters of a 2-conductor transmission line. These parameters include the conductor length, conductor width, distance between two adjacent conductors, PCB height and dielectric constant. A detailed discussion on the selection of optimization variables is provided in [6].

Let $\mathbf{V} = [v_1, v_2, \dots, v_N]^T$, where $v_j = v_j(\mathbf{X}, t)$, $j = 1, 2, \dots, N$, represents the voltage of interest at node j and time point t . Let T be the signal duration between the time when the trapezoidal signal triggers a switch-on state and the time when it triggers a switch-off state. Let w_d, w_f, w_c and w_r denote positive weighting factors. Suppose $\mathbf{J}_a$ is an index set containing all nodes of interest at which the desired response is a signal corresponding to the excitation. Let $\mathbf{J}_b$ be an index set containing all nodes of interest at which the desired response is zero. The error functions for various signal integrity aspects are defined as follows.

Suppose the signal propagation delay is described by the time at which the tran

Figure 3.1: Physical/geometrical parameters of a 2-conductor transmission line.

sient signal reached a threshold value v_T . Let $\tau_{j,max}$ be the upper specification for the propagation delay at node j . The error function for delay minimization can be equivalently described by

$$-w_d(v_j(\mathbf{X}, \tau_{j,max}) - v_T) \quad \text{for } j \in \mathbf{J}_a. \quad (3.6)$$

Suppose τ_j is the desired delay value. To locate the signal to be exactly or almost exactly at this desired time, two error functions are defined

$$w_d(v_j(\mathbf{X}, \tau_j) - v_T) \quad (3.7)$$

$$\begin{aligned} -w_d(v_j(\mathbf{X}, \tau_j) - v_T) \\ \text{for } j \in \mathbf{J}_a. \end{aligned} \quad (3.8)$$

Due to transmission line effects, a response signal may rise/fall much more slowly than the source signal. Suppose the required rise time is t during which the signal should rise from below threshold $v_{T,low}$ to above $v_{T,high}$. The following two error functions are used

$$w_f(v_j(\mathbf{X}, \tau_j - \alpha t_r) - v_{T,low}) \quad (3.9)$$

$$\begin{aligned} -w_f(v_j(\mathbf{X}, \tau_j + \beta t_r) - v_{T,high}) \\ \text{for } j \in \mathbf{J}_a \end{aligned} \quad (3.10)$$

where α and β are positive and $\alpha + \beta = 1$. t_r is the required rise time.

Suppose the required fall time is t_f during which the signal should fall from above $v_{T,high}$ to below $v_{T,low}$. The following two error functions are used

$$w_f(v_j(\mathbf{X}, \tau_j + T + \gamma t_f) - v_{T,low}) \quad (3.11)$$

$$\begin{aligned} -w_f(v_j(\mathbf{X}, \tau_j + T - \mu t_f) - v_{T,high}) \\ \text{for } j \in \mathbf{J}_a. \end{aligned} \quad (3.12)$$

where γ and μ are positive and $\gamma + \mu = 1$.

The existence of undesired signal at node j , $j \in \mathbf{J}_2$, is due to coupling between the multiconductors. Suppose $S_{cj}(t)$ denotes the upper specification on the magnitude of crosstalk at node j and time t , $0 < t < \infty$. Several time samples t_i , $i = 1, 2, \dots$, in the interval $0 < t_j < \infty$ are selected and at each sample point two error functions are defined

$$w_c(v_j(\mathbf{X}, t_i) - S_{cj}(t_i)) \quad (3.13)$$

$$-w_c(v_j(\mathbf{X}, t_i) + S_{cj}(t_i)) \quad (3.14)$$

for $j \in \mathbf{J}_b$ and $0 < t_i < \infty$.

Since the source signal is of duration T , an ideal response should vanish after $t = T + \tau_j + \delta$, where δ is a small value not exceeding t_f . But due to the reflections and undesired ringing, the signal may continue to exist for an extended period of time. Suppose $S_{rj}(t)$ is the upper specification on the magnitude of signal reflections at node j and time t , $T + \tau_j + \delta < t < \infty$. For each time sample t_i selected in the interval $T + \tau_j + \delta < t_i < \infty$ two functions are defined

$$w_r(v_j(\mathbf{X}, t_i) - S_{rj}(t_i)) \quad (3.15)$$

$$-w_r(v_j(\mathbf{X}, t_i) + S_{rj}(t_i)) \quad (3.16)$$

for $j \in \mathbf{J}_a$ and $T + \tau_j + \delta < t_i < \infty$.

3.4 Formulation of the Original Optimization Problem

Let $e_j, j = 1, 2, \dots, m$, represent all error functions defined for a network. A general circuit optimization problem is

$$\min_{\mathbf{X}} P(\mathbf{V}, \mathbf{X}) \quad (3.17)$$

subject to

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{f}(\mathbf{V}, \mathbf{X}, t) \quad (3.18)$$

where $\mathbf{X}$ and $\mathbf{V}$ are vectors of all design variables and nodal voltages in the circuit, respectively. (3.18) is the general circuit equation and $P(\mathbf{V}, \mathbf{X})$ is the objective function. For example, $P(\mathbf{V}, \mathbf{X})$ can be a least p th function [49] of the error functions defined previously. A frequently encountered objective function in circuit design is

$$P(\mathbf{V}, \mathbf{X}) = \max_{j=1,2,\dots,m} \{e_j(\mathbf{V}, \mathbf{X})\} \quad (3.19)$$

where e_j is the j th error function and m is the total number of error functions. Equation (3.19) is a minimax optimization which can be effectively solved by a two stage combined LP and quasi-Newton method [50] [49]. The combined algorithm is computationally practical and has been implemented by Bandler *et al* [51].

Chapter 4

Multilevel Formulation of Circuit Optimization and Its Parallel Implementation

4.1 Network Partitioning for Multilevel Formulation

Let the circuit be decomposed into K subnetworks, e.g. [16]. Let $\mathbf{V}_i$ and $\mathbf{X}_i$ be the vectors of node voltages and of design variables in subnetwork i , respectively. A connection element is defined as an element that connects a node from subnetwork i to a node in subnetwork j , $i \neq j$. The subnetwork formed by all connection elements is called the connection subnetwork. Let $\mathbf{X}_c$ and $\mathbf{V}_c$ represent the design variables and the node voltages, respectively, in the connection subnetwork. The network partitioning is illustrated in Fig. 4.1.

Figure 4.1: Network partitioning.

After partitioning, the circuit equations can be represented as

$$\begin{bmatrix} \frac{\partial \mathbf{V}_1}{\partial t} \\ \frac{\partial \mathbf{V}_2}{\partial t} \\ \vdots \\ \frac{\partial \mathbf{V}_K}{\partial t} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1(\mathbf{V}_1, \mathbf{X}_1, \mathbf{V}_c, \mathbf{X}_c, t) \\ \mathbf{f}_2(\mathbf{V}_2, \mathbf{X}_2, \mathbf{V}_c, \mathbf{X}_c, t) \\ \vdots \\ \mathbf{f}_K(\mathbf{V}_K, \mathbf{X}_K, \mathbf{V}_c, \mathbf{X}_c, t) \end{bmatrix}. \quad (4.1)$$

The connection subnetwork is not further decomposed under the assumption that the circuit has been decomposed such that there are only a small number of elements in the connection subnetwork, i.e., $\mathbf{X}_c$ and $\mathbf{V}_c$ are low dimensional vectors. All the subcircuits should be sparsely connected with each other in order to take full advantage of decomposition. If subcircuits are very densely connected between each other, overall optimization should be considered instead of using decomposition.

In a least *pth* formulation of optimization [49], the objective function follows an additive structure, i.e.,

$$P(\mathbf{V}, \mathbf{X}) = \sum_{i=1}^K P_i(\mathbf{V}_i, \mathbf{X}_i) \quad (4.2)$$

where $P_i(\mathbf{V}_i, \mathbf{X}_i)$ is the objective function for the i th subcircuit. In this case, the original problem can be formulated as a two-level optimization.

The upper level optimization is

$$\min_{\mathbf{X}_c} P(\mathbf{V}, \mathbf{X}_c) \quad (4.3)$$

subject to

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{f}(\mathbf{V}, \mathbf{X}, t). \quad (4.4)$$

The lower level optimization is

$$\min_{\mathbf{X}_i} P_i(\mathbf{V}_i, \mathbf{X}_i) \quad (4.5)$$

subject to:

$$\frac{\partial \mathbf{V}_i}{\partial t} = \mathbf{f}_i(\mathbf{V}_i, \mathbf{X}_i, \mathbf{V}_c, \mathbf{X}_c, t) \quad (4.6)$$

where $i = 1, 2, \dots, K$. Notice that at lower level optimization i , the optimization variable is $\mathbf{X}_i$. $\mathbf{V}_c$ and $\mathbf{X}_c$ are constants obtained from the upper level optimization.

4.2 Multilevel Formulation of Minimax Optimization

However, in many circuit optimization problems, e.g., the signal integrity optimization of VLSI interconnects, minimax optimization is often used. In this case, the objective function is not additive. In this section, the decomposition formulations applicable to minimax optimization is described.

The original problem is

$$\min_{\mathbf{X}} \max_{j=1,2,\dots,m} e_j \quad (4.7)$$

subject to

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{f}(\mathbf{V}, \mathbf{X}, t) \quad (4.8)$$

where e_j is the j th error function and m is the total number of error functions.

The proposed approach is to decompose the original problem into a high-level problem

$$\min_{\mathbf{X}_c} \max_{j=1,2,\dots,m} e_j \quad (4.9)$$

subject to

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{f}(\mathbf{V}, \mathbf{X}, t) \quad (4.10)$$

and a low-level one which is further decomposed into several independent suboptimizations. The i th suboptimization involves only the i th subnetwork

$$\min_{\mathbf{X}_i} \max_{j \in J_i} e_j \quad (4.11)$$

subject to

$$\frac{\partial \mathbf{V}_i}{\partial t} = \mathbf{f}_i(\mathbf{V}_i, \mathbf{X}_i, \mathbf{V}_c, \mathbf{X}_c, t) \quad (4.12)$$

where $i = 1, 2, \dots, K$. J_i is the index set for error functions in the i th subcircuit, $J_i = \{j | \text{if } e_j \text{ belongs to subnetwork } i\}$. Variables that affect multiple subnetworks are used at the high-level optimization.

The coordination between high- and low-level suboptimizations is provided by model coordination [11]. The electrical relation between the connection subnetwork and other subnetworks is represented by Thevenin equivalences $\mathbf{V}_c$. Several cycles between high- and low-level optimizations are usually performed.

The above formulation fits well with a VLSI system, e.g., several MCM's on a PCB. The PCB interconnect circuitry is optimized by the high-level optimization and several MCM's are optimized at the low level. In addition, the bilevel optimization described so far can be extended to 3- (or more) level optimization by further decomposing each low-level subnetwork.

It should be noted that the multilevel optimization approach is independent of specific network topological decomposition approaches. Network decomposition can be achieved by heuristic algorithms such as [16] or obtained from the natural PCB/MCM partition in VLSI systems.

4.3 Parallel Multilevel Optimization and Implementation

An important feature of the proposed multilevel optimization is the parallel processing of the multiple suboptimizations and associated simulation of subcircuits. We have developed SYSCAD (see Chapter 7 for details), a system of programs for analysis and optimization with interchangeable simulators. The system has parallel computing capability [37].

The proposed multilevel algorithm is designed and implemented under the SYSCAD environment and involves multiple computer processes running in a network of workstations. In the decomposition algorithm, the upper level optimization is the main process. It spawns several child processes that run independently and in parallel on several workstations, each solving a suboptimization problem. Process interaction is supported by a message passing interprocess communication package providing remote process control.

Fig. 4.2 shows an implementation of the parallel multilevel optimization using SYSCAD. Each suboptimization drives its own specification module where error functions and variables for the optimization are defined. Each specification module then drives its own simulation module which simulates a subcircuit. Under this implementation both the suboptimization and the related subsimulation are performed in parallel with other suboptimizations and subsimulations. Let p be the count of cycles between the upper and lower level optimizations. Let $\mathbf{X}^p$ be the value of variable $\mathbf{X}$ at the p th cycle. The multilevel optimization proceeds in several steps:

1. Choose starting point for design variables in subcircuit $\mathbf{X}_i^0$, $i = 1, 2, \dots, K$. Set $p = 1$.
2. Upper level optimization:

Figure 4.2: Block diagram for implementation of parallel multilevel optimization. K computers are used to run the lower level suboptimizations in parallel.

Fix $\mathbf{X}_i$ at $\mathbf{X}_i^{p-1}$, $i = 1, 2, \dots, K$. Perform optimization with respect to $\mathbf{X}_c$ which has all the design variables in the connection subnetwork, i.e., solve (4.9). The solution of $\mathbf{X}_c$ at this cycle is denoted as $\mathbf{X}_c^p$.

3. Lower level suboptimizations:

Upon receiving values of design parameters and voltages $\mathbf{X}_c^p$ and $\mathbf{V}_c^p$ from upper level optimization, solve the K suboptimizations in parallel, i.e., solve (4.11) for $i = 1, 2, \dots, K$. The i th suboptimization is with respect to $\mathbf{X}_i$. Note that all optimizations at this level are independent of each other and can be processed in parallel. The solution of $\mathbf{X}_i$ is represented by $\mathbf{X}_i^p$.

4. Check the stopping condition

$$|P^p(\mathbf{V}, \mathbf{X}) - P^{p-1}(\mathbf{V}, \mathbf{X})| < \epsilon \quad (4.13)$$

and

$$p \leq N_{max} \quad (4.14)$$

where ϵ is a predefined small constant. N_{max} is a predefined maximum number of decomposition cycles between upper and lower level optimizations. P^p is the value of objective functions at the p th cycle.

If one of the conditions (4.13) and (4.14) is met, stop the decomposition and exit. Otherwise set $p = p + 1$ and go to step 2.

Computation efficiency of the algorithm w.r.t. conventional optimization can be estimated. Let $N_{overall}$ represent the computation effort of an overall optimization. Let N_{demp}^c and N_{demp}^i represent the computation efforts needed for optimizing the connection subcircuit and the i th subcircuit, respectively. The speed up ratio for parallel multilevel optimization, $R_{speedup}$, is

$$R_{speedup} = N_{cycle} * (N_{demp}^c + \max_{i=1}^K \{N_{demp}^i\}) / N_{overall} \quad (4.15)$$

where N_{cycle} is the total number of decomposition cycles between upper and lower level optimizations. A detailed computation of $R_{speedup}$ is illustrated for Example 2 in Chapter 6.

Chapter 5

Convergence Properties

5.1 Generalized Multilevel Optimization Algorithm

The two level optimization described so far can be generalized into multilevels. Assume that the network equations (3.18) is implicitly substituted into the objective function in (3.17), eliminating the intermediate variable $\mathbf{V}$. Let $U(\mathbf{X})$ be the objective function of this new optimization problem. The overall optimization is

$$\min_{\mathbf{X}} U(\mathbf{X}). \quad (5.1)$$

where $\mathbf{X}^T = [\mathbf{X}_1^T, \mathbf{X}_2^T, \dots, \mathbf{X}_K^T]$ is a vector of all design variables, and T denotes transpose. Suppose the overall optimization is partitioned into K suboptimizations such that the i th suboptimization is:

$$\min_{\mathbf{X}_i} U_i(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K) \quad (5.2)$$

where U_i is the objective function of the i th subcircuit. In the case of minimax optimization,

$$U_i(\mathbf{X}_i) = \max_{j \in J_i} e_j(\mathbf{X}_i).$$

Two types of algorithms are involved in the proposed multilevel optimization formulation.

1. Let $\mathbf{X}^0$ be the initial guess of $\mathbf{X}$. Let p be the count of cycles between upper and lower level optimizations. For each $p, p = 1, 2, \dots, N_{cycle}$, a total of K suboptimizations are performed. The i th suboptimization is:

$$\min_{\mathbf{X}_i} U_i(\mathbf{X}_1^{p-1}, \mathbf{X}_2^{p-1}, \dots, \mathbf{X}_{i-1}^{p-1}, \mathbf{X}_i, \mathbf{X}_{i+1}^{p-1}, \dots, \mathbf{X}_K^{p-1}) \quad (5.3)$$

where only $\mathbf{X}_i$ is the optimization variable. $\mathbf{X}_j, (j \neq i)$ is fixed at its value in the previous cycle. The solution of $\mathbf{X}_i$ becomes $\mathbf{X}_i^p$.

2. If $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{i-1}$ are fixed at their values solved in the present cycle, i.e., the p th cycle, we have the second algorithm. The i th suboptimization is:

$$\min_{\mathbf{X}_i} U_i(\mathbf{X}_1^p, \mathbf{X}_2^p, \dots, \mathbf{X}_{i-1}^p, \mathbf{X}_i, \mathbf{X}_{i+1}^{p-1}, \dots, \mathbf{X}_K^{p-1}) \quad (5.4)$$

The proposed multilevel optimization is decomposed into multilevels using scheme (5.4) between different levels. At the lower level, the optimization problem is further decomposed into several relatively small scale suboptimizations using scheme (5.3).

The first and second algorithms are analogous to Gauss-Jacobi and Gauss-Seidel iterative methods, respectively. The first algorithm allows solving suboptimizations in parallel. It should be noted that the iterations here refer to the iterations between the suboptimizations at different levels.

5.2 Convergence of Multilevel Optimization

Suppose that the objective functions U_i in (5.2), $i = 1, 2, \dots, K$, are G-differentiable (see Appendix A) at the first and second order. The necessary condition for optimality

is:

$$\mathbf{F}(\mathbf{X}) = \begin{bmatrix} \frac{\partial U_1}{\partial \mathbf{X}_1} \\ \frac{\partial U_2}{\partial \mathbf{X}_2} \\ \vdots \\ \frac{\partial U_K}{\partial \mathbf{X}_K} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \quad (5.5)$$

The convergence of multilevel optimization requires that the left hand side of (5.5) be a blockwise diagonal dominant function.

Ilic-Spong [53] and More [54] proved that the condition for a function $\mathbf{F}(\mathbf{X})$, $\mathbf{X}^T = [\mathbf{X}_1^T, \mathbf{X}_2^T, \dots, \mathbf{X}_K^T]$ to be blockwise diagonally dominant is that its Jacobian is a blockwise diagonally dominant matrix for every point of $\mathbf{X} \in D_0$, where D_0 is a real convex set. The above conditions were previously used to verify convergence for circuit simulation algorithms, e.g., [53]. Here the convergence criteria for the proposed optimization method are derived.

The above condition is applied to gradient vector (5.5). Since the derivative of the gradient vector is a Hessian matrix, the convergence of multilevel optimization requires that the Hessian matrix be a blockwise diagonally dominant matrix for every

point of $\mathbf{X}$. The Hessian matrix is

$$\mathbf{H}(\mathbf{X}) = \begin{bmatrix} \frac{\partial^2 U_1}{\partial \mathbf{X}_1^2} & \frac{\partial^2 U_1}{\partial \mathbf{X}_1 \partial \mathbf{X}_2} & \cdots & \frac{\partial^2 U_1}{\partial \mathbf{X}_1 \partial \mathbf{X}_K} \\ \frac{\partial^2 U_2}{\partial \mathbf{X}_2 \partial \mathbf{X}_1} & \frac{\partial^2 U_2}{\partial \mathbf{X}_2^2} & \cdots & \frac{\partial^2 U_2}{\partial \mathbf{X}_2 \partial \mathbf{X}_K} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 U_K}{\partial \mathbf{X}_K \partial \mathbf{X}_1} & \frac{\partial^2 U_K}{\partial \mathbf{X}_K \partial \mathbf{X}_2} & \cdots & \frac{\partial^2 U_K}{\partial \mathbf{X}_K^2} \end{bmatrix} \quad (5.6)$$

where each $\frac{\partial^2 U_i}{\partial \mathbf{X}_i \partial \mathbf{X}_j}$ symbolically represents a sub-Hessian matrix. The blockwise diagonal dominance of the Hessian matrix requires the following [53]:

$$\left\| \frac{\partial^2 U_i}{\partial \mathbf{X}_i^2} \right\|_\infty^{-1} \cdot \sum_{\substack{j=1,2,\dots,K \\ j \neq i}} \left\| \frac{\partial^2 U_i}{\partial \mathbf{X}_i \partial \mathbf{X}_j} \right\|_\infty \leq 1 \quad (5.7)$$

where $\| \cdot \|_\infty$ is a matrix norm defined in Appendix A.

5.3 Convergence of Two-Level Minimax Optimization

The original optimization is decomposed into 2 levels. The lower level optimization is further decomposed into K suboptimizations. The convergence formulas are first applied to the decomposition between the high- and low-levels and then to the decomposition of lower level suboptimizations.

Case 1: Decomposition of Overall Optimizations into 2-levels

Kuhn-Tucker conditions [11] [52] (Appendix A.3) are applied to upper and lower level suboptimizations (4.9) and (4.11) to obtain the optimality conditions

$$\left\{ \begin{array}{l} 1 + \sum_{j \in A} \lambda_j = 0 \\ \sum_{j \in A} \lambda_j \frac{\partial e_j}{\partial \mathbf{X}_c} = \mathbf{0} \\ e_j = e_{n_A} \quad j \in A, \quad j \neq n_A. \end{array} \right. \quad (5.8)$$

and

$$\left\{ \begin{array}{l} 1 + \sum_{j \in A'} \zeta_j = 0 \\ \sum_{j \in A'} \zeta_j \frac{\partial e_j}{\partial \mathbf{X}_1} = \mathbf{0} \\ \sum_{j \in A'} \zeta_j \frac{\partial e_j}{\partial \mathbf{X}_2} = \mathbf{0} \\ \vdots \\ \sum_{j \in A'} \zeta_j \frac{\partial e_j}{\partial \mathbf{X}_K} = \mathbf{0} \\ e_j = e_{n_{A'}} \quad j \in A', \quad j \neq n_{A'}. \end{array} \right. \quad (5.9)$$

respectively, where A, A' are the index sets of active functions [57] in the upper and lower level optimizations, respectively. $n_A, n_{A'}$ are numbers of active functions in the upper and lower levels, respectively. Let $\lambda^T = [\lambda_1, \lambda_2, \dots, \lambda_{n_A}]$ and $\zeta^T = [\zeta_1, \zeta_2, \dots, \zeta_{n_{A'}}]$. Solving upper and lower level optimizations iteratively is equivalent to solving the above two sets of equations iteratively by either Gauss-Seidel or Gauss-Jacobi relaxation method. If the iterative process is to converge, the left hand sides

of (5.8) and (5.9) are required to be blockwise diagonal functions. From [51], the gradient matrices of (5.8) and (5.9) are required to be blockwise diagonally dominant for every point of $\mathbf{X}_c, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \lambda$ and ζ . For simplicity, compact notations are used $\mathcal{F}(\mathbf{X}_c, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \lambda)$ and $\mathcal{G}(\mathbf{X}_c, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \zeta)$ to represent upper and lower level equations of (5.8) and (5.9), respectively.

Equations (5.8) and (5.9) now become

$$\begin{cases} \mathcal{F}(\mathbf{X}_c, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \lambda) = \mathbf{0} \\ \mathcal{G}(\mathbf{X}_c, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \zeta) = \mathbf{0} \end{cases} \quad (5.10)$$

Let $\mathcal{X}^T = [\mathbf{X}_c^T, \lambda^T]$, $\mathcal{Y}^T = [\mathbf{X}_1^T, \mathbf{X}_2^T, \dots, \mathbf{X}_K^T, \zeta^T]$. Let σ and τ be the dimensions of $\mathcal{X}$ and $\mathcal{Y}$, respectively. The Jacobian matrix of (5.10) is

$$\mathbf{J}(\mathcal{X}, \mathcal{Y}) = \begin{bmatrix} \frac{\partial \mathcal{F}}{\partial \mathcal{X}} & \frac{\partial \mathcal{F}}{\partial \mathcal{Y}} \\ \frac{\partial \mathcal{G}}{\partial \mathcal{X}} & \frac{\partial \mathcal{G}}{\partial \mathcal{Y}} \end{bmatrix}. \quad (5.11)$$

If the iterative solution converges, matrix (5.11) is required to be blockwise diagonally dominant, i.e.,

$$\|(\frac{\partial \mathcal{F}}{\partial \mathcal{X}})^{-1}\|_{\infty} \cdot \|\frac{\partial \mathcal{F}}{\partial \mathcal{Y}}\|_{\infty} < 1, \quad (5.12)$$

$$\|(\frac{\partial \mathcal{G}}{\partial \mathcal{Y}})^{-1}\|_{\infty} \cdot \|\frac{\partial \mathcal{G}}{\partial \mathcal{X}}\|_{\infty} < 1. \quad (5.13)$$

Case 2: Decomposition of the Lower Level Optimization into K Suboptimizations

Kuhn-Tucker conditions for i th suboptimization is

$$\begin{cases} 1 + \sum_{j \in A_i} \zeta_j = 0 \\ \sum_{j \in A_i} \zeta_j \frac{\partial c_j}{\partial \mathbf{X}_i} = \mathbf{0} \\ c_j = c_{n_{A_i}} \quad j \in A_i, \quad j \neq n_{A_i}. \end{cases} \quad (5.14)$$

where A_i is the active set of the i th suboptimization. n_{A_i} is number of active functions in the set A_i .

Let $\mathbf{Z}_i$ be a vector containing all the $\zeta_j, j \in A_i$. Let the above equation (5.14) be represented by $\mathcal{H}_i(\mathbf{X}_c, \mathbf{X}_i, \mathbf{Z}_i) = 0$. The optimality conditions of the lower level suboptimizations are

$$\begin{cases} \mathcal{H}_1(\mathbf{X}_c, \mathbf{X}_1, \mathbf{Z}_1) = 0 \\ \mathcal{H}_2(\mathbf{X}_c, \mathbf{X}_2, \mathbf{Z}_2) = 0 \\ \vdots \\ \mathcal{H}_K(\mathbf{X}_c, \mathbf{X}_K, \mathbf{Z}_K) = 0 \end{cases} \quad (5.15)$$

The ij th block of the Hessian matrix $\mathbf{H}_{ij}$ is

$$\mathbf{H}_{ij} = \frac{\partial \mathcal{H}_i}{\partial \begin{bmatrix} \mathbf{X}_j \\ \mathbf{Z}_j \end{bmatrix}} = \begin{cases} 0 & \text{if } i \neq j \\ \text{nonzero} & \text{if } i = j \end{cases}. \quad (5.16)$$

Therefore the Hessian matrix in (5.6) is block diagonal. The convergence condition (5.7) is automatically satisfied for the lower level optimization by decomposition.

5.4 Example: An Intuitive Explanation of the Convergence Criteria

Without loss of generality, index sets $A = \{1, 2, \dots, n_A\}$ is assumed. As an example, the elements in the Jacobian matrix of (5.11) are

$$\frac{\partial \mathcal{F}}{\partial \mathbf{X}} = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 \\ \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_1^2} & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_1 \partial x_2} & \dots & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_1 \partial x_\sigma} & \frac{\partial \epsilon_1}{\partial x_1} & \frac{\partial \epsilon_2}{\partial x_1} & \dots & \frac{\partial \epsilon_{n_A}}{\partial x_1} \\ \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_2 \partial x_1} & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_2^2} & \dots & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_2 \partial x_\sigma} & \frac{\partial \epsilon_1}{\partial x_2} & \frac{\partial \epsilon_2}{\partial x_2} & \dots & \frac{\partial \epsilon_{n_A}}{\partial x_2} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_\sigma \partial x_1} & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_\sigma \partial x_2} & \dots & \sum \lambda_j \frac{\partial^2 \epsilon_j}{\partial x_\sigma^2} & \frac{\partial \epsilon_1}{\partial x_\sigma} & \frac{\partial \epsilon_2}{\partial x_\sigma} & \dots & \frac{\partial \epsilon_{n_A}}{\partial x_\sigma} \\ \frac{\partial \epsilon_1}{\partial x_1} - \frac{\partial \epsilon_{n_A}}{\partial x_1} & \frac{\partial \epsilon_1}{\partial x_2} - \frac{\partial \epsilon_{n_A}}{\partial x_2} & \dots & \frac{\partial \epsilon_1}{\partial x_\sigma} - \frac{\partial \epsilon_{n_A}}{\partial x_\sigma} & 0 & 0 & \dots & 0 \\ \frac{\partial \epsilon_2}{\partial x_1} - \frac{\partial \epsilon_{n_A}}{\partial x_1} & \frac{\partial \epsilon_2}{\partial x_2} - \frac{\partial \epsilon_{n_A}}{\partial x_2} & \dots & \frac{\partial \epsilon_2}{\partial x_\sigma} - \frac{\partial \epsilon_{n_A}}{\partial x_\sigma} & 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \frac{\partial \epsilon_{n_A-1}}{\partial x_1} - \frac{\partial \epsilon_{n_A}}{\partial x_1} & \frac{\partial \epsilon_{n_A-1}}{\partial x_2} - \frac{\partial \epsilon_{n_A}}{\partial x_2} & \dots & \frac{\partial \epsilon_{n_A-1}}{\partial x_\sigma} - \frac{\partial \epsilon_{n_A}}{\partial x_\sigma} & 0 & 0 & \dots & 0 \end{bmatrix} \dots\dots (5.17)$$

and

$$\frac{\partial \mathcal{F}}{\partial \mathcal{Y}} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \sum \lambda_j \frac{\partial^2 c_j}{\partial x_1 \partial y_1} & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_1 \partial y_2} & \cdots & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_1 \partial y_\tau} & 0 & 0 & \cdots & 0 \\ \sum \lambda_j \frac{\partial^2 c_j}{\partial x_2 \partial y_1} & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_2 \partial y_2} & \cdots & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_2 \partial y_\tau} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \sum \lambda_j \frac{\partial^2 c_j}{\partial x_\sigma \partial y_1} & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_\sigma \partial y_2} & \cdots & \sum \lambda_j \frac{\partial^2 c_j}{\partial x_\sigma \partial y_\tau} & 0 & 0 & \cdots & 0 \\ \frac{\partial c_1}{\partial y_1} - \frac{\partial c_{n_A}}{\partial y_1} & \frac{\partial c_1}{\partial y_2} - \frac{\partial c_{n_A}}{\partial y_2} & \cdots & \frac{\partial c_1}{\partial y_\tau} - \frac{\partial c_{n_A}}{\partial y_\tau} & 0 & 0 & \cdots & 0 \\ \frac{\partial c_2}{\partial y_1} - \frac{\partial c_{n_A}}{\partial y_1} & \frac{\partial c_2}{\partial y_2} - \frac{\partial c_{n_A}}{\partial y_2} & \cdots & \frac{\partial c_2}{\partial y_\tau} - \frac{\partial c_{n_A}}{\partial y_\tau} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \frac{\partial c_{n_A-1}}{\partial y_1} - \frac{\partial c_{n_A}}{\partial y_1} & \frac{\partial c_{n_A-1}}{\partial y_2} - \frac{\partial c_{n_A}}{\partial y_2} & \cdots & \frac{\partial c_{n_A-1}}{\partial y_\tau} - \frac{\partial c_{n_A}}{\partial y_\tau} & 0 & 0 & \cdots & 0 \end{bmatrix}$$

..... (5.18)

where $\mathcal{X} = [x_1, x_2, \dots, x_\sigma]^T$, $\mathcal{Y} = [y_1, y_2, \dots, y_\tau]^T$ and each sum operation is performed over all the j 's, $j \in A$. The norms of above matrices are

$$\begin{aligned} \left\| \frac{\partial \mathcal{F}}{\partial \mathcal{X}} \right\|_\infty = \max \{ n_A, \max_i \left[\sum_{k=1}^{\sigma} \sum_{j=1}^{n_A} \left| \frac{\partial^2 c_j}{\partial x_i \partial x_k} \right| + \sum_{j=1}^{n_A} \left| \frac{\partial c_j}{\partial x_i} \right| \right], \right. \\ \left. \max_{i=1,2,\dots,n_A-1} \left[\sum_{j=1}^{\sigma} \left| \frac{\partial c_i}{\partial x_j} - \frac{\partial c_{n_A}}{\partial x_j} \right| \right] \right\} \end{aligned} \quad (5.19)$$

$$\begin{aligned} \left\| \frac{\partial \mathcal{F}}{\partial \mathcal{Y}} \right\|_\infty = \max \{ 0, \max_i \left[\sum_{k=1}^{\tau} \sum_{j=1}^{n_A} \left| \frac{\partial^2 c_j}{\partial x_i \partial y_k} \right| \right], \right. \\ \left. \max_{i=1,2,\dots,n_A-1} \left[\sum_{j=1}^{\tau} \left| \frac{\partial c_i}{\partial y_j} - \frac{\partial c_{n_A}}{\partial y_j} \right| \right] \right\} \end{aligned} \quad (5.20)$$

A necessary condition to satisfy conditions (5.12) and (5.13) is that the norm in (5.19) be greater than that in (5.20). Examining the contents of (5.19) and (5.20),

we have the following comparison:

$$n_A \leftrightarrow 0, \quad (5.21)$$

$$\max_i \left[\sum_{k=1}^{\sigma} \sum_{j=1}^{n_A} \left| \frac{\partial^2 e_j}{\partial x_i \partial x_k} \right| + \sum_{j=1}^{n_A} \left| \frac{\partial e_j}{\partial x_i} \right| \right] \leftrightarrow \max_i \left[\sum_{k=1}^{\tau} \sum_{j=1}^{n_A} \left| \frac{\partial^2 e_j}{\partial x_i \partial y_k} \right| \right], \quad (5.22)$$

$$\max_{i=1,2,\dots,n_A-1} \left[\sum_{j=1}^{\sigma} \left| \frac{\partial e_i}{\partial x_j} - \frac{\partial e_{n_A}}{\partial x_j} \right| \right] \leftrightarrow \max_{i=1,2,\dots,n_A-1} \left[\sum_{j=1}^{\tau} \left| \frac{\partial e_i}{\partial y_j} - \frac{\partial e_{n_A}}{\partial y_j} \right| \right]. \quad (5.23)$$

Intuitive explanation for (5.22) and (5.23) is that each subcircuit is mainly affected by its own variables and less affected by the variables in other subcircuits. The chances of $\|\frac{\partial \mathcal{F}}{\partial x}\|$ greater than $\|\frac{\partial \mathcal{F}}{\partial y}\|$ are good if the couplings between any two subnetworks are weak.

Chapter 6

Optimization Examples

The proposed formulation is tested by several time- and frequency-domain optimization examples. In the thesis, two optimization examples are presented. The first example is an optimization problem with lumped elements, the second one is an optimization problem with distributed elements. The optimization of the first example is solved in the frequency domain. The second one is optimized in the time domain. NILT(Numerical Inversion of Laplace Transform)[4] is used to calculate the responses at the required time points. The combination of NILT method and decomposition technique makes parallel computation possible. Optimization variables include lengths of transmission lines, terminating resistances and capacitances. Minimax optimizations are used for both the high- and the low-level problems. The optimizer used is MMAG in which partial perturbation is used internally to obtain sensitivity for optimization [49]. Simulators are not required to provide sensitivity output.

The optimization algorithm is a two stage minimax algorithm which combines the robustness of the first order Gauss-Newton type with the speed of the quasi-Newton method [57]. At the solution, all error functions are simultaneously minimized. All examples are tested on the Sun SPARC 2 machines. The parallel optimization is

realized under the SYSCAD environment developed at Carleton University.

6.1 Example 1: Frequency Domain Optimization of A Lumped Multiplexing Network

The multiplexing circuit shown in Fig. 6.1 is used to verify and illustrate the proposed method. Subcircuit 1 is a low pass filter; subcircuit 2 is a band pass filter; subcircuit 3 is also a low pass filter. The circuit has a total of 11 nodes. $g_i, i = 1, 2, 3$, is the transconductance of the voltage controlled current source in i th subcircuit.

Responses of interest are at nodes 7, 9 and 11 which correspond to the outputs of subcircuits 1, 2 and 3, respectively. The insertion loss at the nodes of interest are plotted in solid lines in Fig. 6.2 - Fig. 6.4. Insertion loss IL between a source and a load is defined as

$$IL(\mathbf{X}, \omega) = 20 \log_{10} \frac{R_L}{R_L + R_S} \left| \frac{V_{in}}{V_{out}} \right| \quad (6.1)$$

where R_L, R_S are load and source resistances, respectively. Voltages V_{in} and V_{out} are the source voltage and the voltage across R_L , respectively.

The angular frequency ω is in $rad/s \times 10^9$. The frequency ranges are discretized and frequency step is taken as $\Delta\omega = 0.25 rad/s \times 10^9$. Various design specifications are listed in Table 6.1. We specify $-5dB$ upper specification at frequency 0 to 1.0 and a $20dB$ lower specification at frequency 4.0 on IL_7 and to achieve low pass property. Two lower specifications and one upper specification are imposed on IL_9 to pass the signal from frequency 1.5 to 3.5 and to stop the signals elsewhere. To make subcircuit 3 a low pass filter, an upper specification of $0dB$ for IL_{11} at frequency 0 to 2.0 and a lower specification of $40dB$ at frequency 3.0 and beyond are imposed. The angular frequency ω is in $rad/s \times 10^9$. The frequency ranges are discretized and error functions are defined at each frequency point. Frequency step is taken as

Figure 6.1: Example 1. A lumped multiplexing network.

$\Delta\omega = 0.25\text{rad/s} \times 10^9$. Various design specifications are listed in Table 6.1. Error functions are defined according to (3.6) - (3.16) where all weighting factors are taken as 1.0, where $c_1(\mathbf{X}_c)$ to $c_4(\mathbf{X}_c)$ are

$$c_1(\mathbf{X}_c) = (IL_7(\mathbf{X}_c, 0.25) - (-5dB))$$

$$c_2(\mathbf{X}_c) = (IL_7(\mathbf{X}_c, 0.5) - (-5dB))$$

$$c_3(\mathbf{X}_c) = (IL_7(\mathbf{X}_c, 0.75) - (-5dB))$$

$$c_4(\mathbf{X}_c) = (IL_7(\mathbf{X}_c, 1.0) - (-5dB));$$

$c_{15}(\mathbf{X}_c)$ and $c_{18}(\mathbf{X}_c)$ are

$$c_{15}(\mathbf{X}_c) = -(IL_9(\mathbf{X}_c, 0.25) - (30dB))$$

$$c_{16}(\mathbf{X}_c) = -(IL_9(\mathbf{X}_c, 0.5) - (30dB))$$

$$c_{17}(\mathbf{X}_c) = -(IL_9(\mathbf{X}_c, 0.75) - (30dB))$$

$$c_{18}(\mathbf{X}_c) = -(IL_9(\mathbf{X}_c, 1.0) - (30dB));$$

(6.2)

$c_{36}(\mathbf{X}_c)$ and $c_{39}(\mathbf{X}_c)$ are

$$c_{36}(\mathbf{X}_c) = (IL_{11}(\mathbf{X}_c, 1.5) - (0dB))$$

$$c_{37}(\mathbf{X}_c) = (IL_{11}(\mathbf{X}_c, 1.75) - (0dB))$$

$$c_{38}(\mathbf{X}_c) = (IL_{11}(\mathbf{X}_c, 2.0) - (0dB))$$

$$c_{39}(\mathbf{X}_c) = -(IL_{11}(\mathbf{X}_c, 3.0) - (40dB)).$$

The lower level error functions are defined in the same way as the upper level error functions. The total number of error functions in subcircuits 1, 2 and 3 are 14, 15 and 10, respectively. The weighting factors are taken as 1.0. The error function index set for the three subcircuits are $J_1 = \{1, 2, \dots, 14\}$, $J_2 = \{15, 16, \dots, 29\}$ and $J_3 = \{30, 31, \dots, 39\}$. The total number of error functions is 39.

Response of Interest	Frequency Range	Design Specification
Insertion Loss at Node 7	$0 < \omega < 3.0$	$IL_7 < -5 \text{ dB}$
	$\omega > 4.0$	$IL_7 > 20 \text{ dB}$
Insertion Loss at Node 9	$0 < \omega < 1.0$	$IL_9 > 30 \text{ dB}$
	$1.5 < \omega < 3.5$	$IL_9 < 15 \text{ dB}$
	$\omega > 4.5$	$IL_9 > 30 \text{ dB}$
Insertion Loss at Node 11	$0 < \omega < 2.0$	$IL_{11} < 0 \text{ dB}$
	$\omega > 3.0$	$IL_{11} > 40 \text{ dB}$

Table 6.1: Design specifications for the multiplexing circuit.

For example, the error functions in subcircuit 3 are

$$\begin{aligned}
 e_{30}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 0.00) - (0dB)) \\
 e_{31}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 0.25) - (0dB)) \\
 e_{32}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 0.50) - (0dB)) \\
 e_{33}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 0.75) - (0dB)) \\
 e_{34}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 1.00) - (0dB)) \\
 e_{35}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 1.25) - (0dB)) \\
 e_{36}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 1.50) - (0dB)) \\
 e_{37}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 1.75) - (0dB)) \\
 e_{38}(\mathbf{X}_3) &= (IL_{11}(\mathbf{X}_3, 2.00) - (0dB)) \\
 e_{39}(\mathbf{X}_3) &= -(IL_{11}(\mathbf{X}_3, 3.0) - (40dB)).
 \end{aligned}$$

The first four error functions in subcircuit 1 are

$$\begin{aligned}
 e_1(\mathbf{X}_1) &= (IL_7(\mathbf{X}_1, 0.25) - (-5dB)) \\
 e_2(\mathbf{X}_1) &= (IL_7(\mathbf{X}_1, 0.5) - (-5dB))
 \end{aligned}$$

$$e_3(\mathbf{X}_1) = (IL_7(\mathbf{X}_1, 0.75) - (-5dB))$$

$$e_4(\mathbf{X}_1) = (IL_7(\mathbf{X}_1, 1.0) - (-5dB)).$$

The first four error functions in subcircuit 2 are

$$e_{15}(\mathbf{X}_2) = -(IL_9(\mathbf{X}_2, 0.25) - (30dB))$$

$$e_{16}(\mathbf{X}_2) = -(IL_9(\mathbf{X}_2, 0.5) - (30dB))$$

$$e_{17}(\mathbf{X}_2) = -(IL_9(\mathbf{X}_2, 0.75) - (30dB))$$

$$e_{18}(\mathbf{X}_2) = -(IL_9(\mathbf{X}_2, 1.0) - (30dB)).$$

Design parameters in the connection subcircuit and other subcircuits include all capacitors, all conductors and transconductances of the voltage controlled current sources. The design variables in the connection circuit are

$$\mathbf{X}_c = [C_1, C_2, C_3, L_1, L_2, L_3, L_4]^T.$$

The design variables in subcircuits 1, 2 and 3, are

$$\mathbf{X}_1 = [g_1, C_4, C_5, L_5]^T,$$

$$\mathbf{X}_2 = [g_2, L_6, L_7, C_6]^T,$$

and

$$\mathbf{X}_3 = [g_3, C_7, C_8, L_8]^T,$$

respectively. The insertion loss responses with respect to frequency before and after optimization are in Fig. 6.2 - Fig. 6.4. It can be seen that after optimization the multi-band filtering properties of the multiplexing circuit have been greatly improved. Elapsed time counts are listed in Table 6.2. Also the objective function values listed in the table are final values of the optimization. The actual speed up ratio by the proposed method vs. the conventional optimization can be computed as $\frac{12}{4} = 3.0$.

Number of Cycles	Proposed Approach		Conventional Optimization	
	Elapsed Time	Objective Function	Elapsed Time	Objective Function
1	4 mins	4.1693	12 mins	4.1700
2	8 mins	4.7240e-2	35 mins	6.2280e-2
3	10 mins	-5.1190e-2	42 mins	4.1370e-1

Table 6.2: Elapsed time count for Example 1.

Figure 6.2: Response from the 1st subcircuit.

Figure 6.3: Response from the 2nd subcircuit.

Figure 6.4: Response from the 3rd subcircuit.

6.2 Example 2: Time Domain Optimization of A Distributed Network

This example represents high-speed VLSI interconnect hierarchy in PCB and MCM. The circuit optimization is performed in the time domain. The overall circuit of Fig. 6.5 contains a bus line structure where signals propagate along the bus to 5 subnetworks. The overall circuit is partitioned into a connection circuit and five subcircuits. The bus line is the connection subnetwork modeled by 11 coupled transmission lines. Each of the five subnetworks contains 7 transmission lines as shown in Fig. 6.6. The circuit in this example has a total of 132 nodes, 46 transmission lines and 105 optimization variables. The excitation is a trapezoidal signal shown in Fig. 6.7. The responses of interest are the outputs from each of the five subnetworks as shown in Fig. 6.8, where the signals are distorted and the propagation delays between the five signals are different. The purpose of the optimization is to achieve exact zero skew [58] [59] and at the same time reduce distortion. An upper specification of 0.1 V at 3.5 ns is applied to ensure the proper signal rise time. Another set of specifications are the upper and lower specification at the $t = 5$ ns. This set of specifications intend to locate signal propagation delay at exactly 5 ns. In order to reduce the signal distortion, an upper specification of 2.0 V and lower specification of 1.8 V are placed during the period when signal is high. To reduce the reflections, an upper specification of 0.1 V and a lower specification of 0 V are imposed during the period when the signal should vanish. The various specifications are listed in Table 6.3, where j is the node number of interest, i.e., $j = 48, 69, 90, 111, 132$. The time intervals are discretized and error functions at each time point are defined according to (3.6) - (3.16). All weighting factors are taken as 1.0. Step size of t is taken as $\Delta t = 1.0$ ns.

Figure 6.5: A distributed network for Example 2. Each subcircuit is shown in Fig. 6.6.

Figure 6.6: A typical subcircuit in Fig. 6.5, $j = 48, 69, 90, 111, 132$.

Figure 6.7: Trapezoidal signal used as excitations for the distributed network.

Response of Interest	Time Range	Design Specification
Voltage v_j $j = 48, 69, 90,$ $111, 132.$	$t = 3.5 \text{ ns}$	$0.6 \text{ V} < v_j < 0.1 \text{ V}$
	$t = 5.0 \text{ ns}$	$1.5 \text{ V} \leq v_j \leq 1.5 \text{ V}$
	$6.0 \text{ ns} < t < 9.0 \text{ ns}$	$1.8 \text{ V} < v_j < 2.0 \text{ V}$
	$12.0 \text{ ns} < t < 14.0 \text{ ns}$	$0.0 \text{ V} < v_j < 0.1 \text{ V}$

Table 6.3: Design specifications for the distributed network.

Design variables can be the length of transmission lines, terminations and matching circuits. In the connection circuit, optimization variables are lengths of transmission lines, i.e.,

$$\mathbf{X}_c = [l_7, l_8, l_9, l_{10}, l_{11}]^T.$$

In subcircuit i , the optimization variables include lengths of transmission lines, inductors, capacitors and transconductance of the VCCS,

$$\begin{aligned} \mathbf{X}_i = & [g^i, l_4^i, L_4^i, C_7^i, L_5^i, l_5^i, \\ & l_6^i, L_7^i, C_{11}^i, L_8^i, l_7^i, l_1^i, \\ & l_2^i, l_3^i, C_6^i, C_8^i, C_{10}^i, C_{11}^i]^T, \\ & i = 1, 2, 3, 4, 5. \end{aligned} \quad (6.3)$$

Circuit parameters with their starting values are presented in Fig. 6.5 and Fig. 6.6. The electrical parameters of the 3-conductor coupled transmission lines for $l_1, l_2, \dots, l_6$ are

$$\mathbf{R} = 0, \quad (6.4)$$

$$\mathbf{G} = 0, \quad (6.5)$$

$$\mathbf{L} = \begin{bmatrix} 1500 & 180 & 0 \\ 180 & 1500 & 180 \\ 0 & 180 & 1500 \end{bmatrix} nH/m, \quad (6.6)$$

$$\mathbf{C} = \begin{bmatrix} 0.266 & -0.02 & 0.0 \\ -0.02 & 0.266 & -0.02 \\ 0.0 & -0.02 & 0.266 \end{bmatrix} nF/m. \quad (6.7)$$

The parallel multilevel optimization algorithm described in Chapter 4 is employed. Six SPARC 2 workstations were used. One is for the upper level suboptimization and

the other five for the lower level suboptimizations performed in parallel. Elapsed time counts and final values of objective function for optimization using the new and conventional approaches are listed in Table 6.7. The speed up ratio of 6.2 is achieved by two factors. The first factor is the speedup from using the decomposition algorithm. The second factor is the speedup from parallel processing. Waveforms before optimization are plotted in Fig. 6.8. The waveforms after multilevel optimization are shown in Fig. 6.9.

Figure 6.8: Responses at the outputs of 5 subcircuits before optimization.

Figure 6.9: Responses at the outputs of 5 subcircuits after optimization using multi-level optimization.

From the computation results, it can be seen that error functions are greatly reduced. The propagation delay time of the output signals of different subcircuits are all equal to 5 ns. Exact zero time skew is therefore achieved. Signal distortions are reduced and the signal level between the five subcircuits are much more consistent than before optimization. Waveforms of all the outputs are in general also improved.

Design parameters before and after optimization are listed in Table 6.4 for the connection circuit. The parameters after overall optimization are listed in Table 6.6. Design variables before and after optimization of all the different subcircuits are in Table 6.5.

6.3 CPU Analysis

To compare the efficiency of the proposed algorithm with conventional optimization technique, The major computation effort is represented by the number of multiplications and divisions. Let K be the number of subcircuits. Let N be the number of nodes in the overall circuit. Let N_i be the number of nodes in subcircuit i . Let N_{cycle} be the number of decomposition cycles between high- and low-level optimizations. Let $N_{iteration}$ be the number of iterations inside optimizations. Let n_t be the number of time samples. Let n_ω be the number of frequency points. n_ω is equal to $5 * n_t$ for the numerical inversion of Laplace transform simulator [4]. Let N_{optvar}^c be the number of design variables in the connection circuit. Let N_{optvar}^i be the number of design variables in subcircuit i . Using sparse matrix techniques, the number of multiplications and divisions for solving a set of n linear equations is proportional to n^α , where $\alpha = 1.1 \sim 1.3$ [60]. Optimizer used is MMAG [49] which is MiniMax with Approximate Gradient. Partial perturbation and Broyden Updates are used for automatically evaluating sensitivities. The number of perturbations is equal to the

Variables	Before optimization(m)	After decomposition(m)	After overall optimization(m)
l_7	0.05	4.303e-2	3.086e-2
l_8	0.04	3.579e-2	2.521e-2
l_9	0.03	8.972e-2	6.144e-2
l_{10}	0.02	8.480e-3	8.724e-2
l_{11}	0.01	8.890e-3	5.000e-3

Table 6.4: Design parameters in the connection circuit before and after optimization for Example 2.

number of design variables. The major computation effort in upper level is

$$N_{demp}^c = N^a * N_{iteration} * N_{optvar}^c * n_{\omega} + lud(N_{optvar}^c) * N_{iteration}. \quad (6.8)$$

where $lud(n) = (n^3 - n)/3 + n^2$ is the computation effort of LU factorization and Forward and Backward substitution for a dense set of $n \times n$ linear equations. In the low-level, the computation effort for the i th suboptimization is

$$N_{demp}^i = N_i^a * N_{iteration} * N_{optvar}^i * n_{\omega} + lud(N_{optvar}^i) * N_{iteration}. \quad (6.9)$$

The total computation effort for decomposition when considering parallel optimization is

$$N_{demp} = N_{cycle} * (N_{demp}^c + \max_{i=1}^K \{N_{demp}^i\}). \quad (6.10)$$

The total computation effort of conventional optimization $N_{overall}$ is

$$N_{overall} = N^a * N_{iteration} * N_{optvar} * n_{\omega} + lud(N_{optvar}) * N_{iteration} \quad (6.11)$$

where the total number of design variables N_{optvar} is

$$N_{optvar} = \sum_{i=1}^K N_{optvar}^i + N_{optvar}^c. \quad (6.12)$$

Variables	Before optimization	After optimization				
		sub01	sub02	sub03	sub04	sub05
g	1.00	7.355e-1	9.370e-1	1.087	1.046	9.588e-1
l_4	0.05m	5.104e-2	7.427e-2	1.000e-4	1.913e-3	2.460e-2
L_4	10.0nH	9.974	9.957	1.001e+1	9.829	9.8973
C_7	0.0001nF	3.853e-2	3.590e-4	1.000e-4	8.240e-3	1.000e-1
L_5	10.0nH	1.004e+1	9.997	1.001e+1	9.829	9.897
l_5	0.03m	5.616e-2	1.000e-4	3.500e-4	5.410e-3	1.000e-4
l_6	0.04m	1.000e-4	1.865e-1	1.042e-1	9.210e-4	1.000e-4
L_7	5.0nH	4.989	4.989	5.001	1.862	1.959
C_{11}	0.0005nF	4.950e-4	1.000e-4	1.000e-4	1.000e-4	1.000e-4
L_8	5.0nH	4.989	4.989	5.001	1.866	1.958
l_7	0.02m	1.144e-2	2.109e-3	3.360e-4	2.558e-3	1.000e-4
l_1	0.03m	3.000e-2	3.000e-2	2.992e-2	3.000e-2	3.000e-2
l_2	0.03m	1.000e-4	4.103e-3	2.997e-2	1.050e-3	3.000e-2
l_3	0.03m	1.000e-4	8.283e-4	2.472e-2	2.929e-3	3.000e-2
C_6	0.001nF	1.000e-4	1.398e-2	1.000e-4	1.912e-2	1.123e-2
C_8	0.0005nF	1.000e-4	1.000e-4	1.000e-4	1.000e-4	1.000e-4
L_6	5.0nH	5.010	5.005	5.007	5.082	5.015
C_9	0.002nF	4.980e-3	3.867e-3	1.000e-4	1.000e-4	1.504e-2
C_{10}	0.001nF	5.380e-3	1.872e-2	2.960e-2	1.000e-4	9.234e-3
C_{12}	0.001nF	4.180e-4	1.708e-3	1.000e-4	1.000e-4	1.000e-4

Table 6.5: Design parameters of 5 subcircuits before and after optimization for Example 2.

Variables	Before optimization	After optimization				
		sub01	sub02	sub03	sub04	sub05
g	1.00	7.982e-1	1.023	9.854e-1	1.059	1.012
L_4	0.05m	4.133e-2	2.229e-2	4.603e-2	1.000e-4	5.190e-3
L_4	10.0nH	1.000e+1	9.999	1.000e+1	1.000e+1	1.000e+1
C_7	0.0001nF	1.457e-2	1.470e-2	1.000e-4	1.409e-2	2.150e-3
L_5	10.0nH	1.000e+1	9.999	1.000e+1	1.000e+1	9.999
l_5	0.03m	3.300e-2	5.875e-3	3.585e-2	1.000e-4	6.660e-3
l_6	0.04m	4.290e-2	4.005e-2	3.956e-2	2.189e-2	4.312e-2
L_7	5.0nH	5.000	4.999	5.000	4.999	4.999
C_{11}	0.0005nF	3.990e-3	2.105e-2	1.000e-4	1.550e-4	1.0e-4
L_8	5.0nH	4.999	4.999	5.000	5.000	4.999
l_7	0.02m	7.380e-3	1.000e-4	1.749e-2	1.000e-4	1.062e-2
l_1	0.03m	2.718e-2	1.637e-2	2.575e-2	1.000e-4	3.000e-2
l_2	0.03m	3.000e-2	2.999e-2	2.999e-2	3.000e-2	2.791e-2
l_3	0.03m	3.000e-2	2.999e-2	2.999e-2	3.000e-2	2.304e-2
C_6	0.001nF	2.290e-2	1.860e-3	1.000e-4	1.640e-2	1.000e-4
C_8	0.0005nF	4.430e-3	6.392e-4	1.000e-4	1.000e-4	3.790e-2
L_6	5.0nH	5.000	5.000	5.000	5.000	5.000
C_9	0.002nF	4.341e-2	1.013e-2	1.310e-2	1.000e-4	1.000e-4
C_{10}	0.001nF	7.260e-2	1.768e-2	3.452e-2	1.000e-4	2.159e-3
C_{12}	0.001nF	6.370e-3	1.361e-2	3.374e-3	2.820e-3	1.000e-4

Table 6.6: Design parameters after overall optimization for Example 2.

Proposed Optimization		Conventional Optimization		Speed Up
Objective Function	Elapsed Time	Objective Function	Elapsed Time	Ratio
0.1823	30 mins	0.19923	3 hrs 4 mins	6.2
0.1330	60 mins	0.17390	4 hrs 17 mins	
0.0900	90 mins	0.03000	7 hrs 10 mins	

Table 6.7: Elapsed time count by parallel decomposition for Example 2.

Equations (6.8) - (6.10) define the number of operations for the proposed algorithm whereas (6.11) - (6.12) define the operation count for conventional optimization. In the distributed network example, parameters are estimated as $n_w = 30$, $N_{gate} = 3$, $N_{iteration} = 3$, $N_{optvar}^c = 5$, $N_{optvar}^i = 20$, $N_i = 21$, $i = 1, 2, \dots, 5$, $N = 132$, $\alpha = 1.0$. The speed up ratio with respect to the number of subcircuits for parallel computation are plotted in 6.10.

When $K = 5$, the speed up ratio estimated using above parameters for parallel optimization is 7.0. These results are very close to the actual speed up ratio of 6.2 obtained from elapsed time count. It is to be noted that the speed up ratio will become higher when the number of subcircuits is increased.

Figure 6.10: Estimation of speed up ratio in parallel computing.

Chapter 7

SYSCAD: A Computer-Aided Design System Implementation

7.1 Introduction to SYSCAD

SYSCAD, a system of programs for analysis and optimization with interchangeable simulators, was developed. The system has parallel computing capability [37]. The CAD framework is the system used to connect and integrate the individual function blocks or modules.

To systematically connect various modules in the CAD system, a macrofunction was defined for each module of the system

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) \quad (7.1)$$

where $\mathbf{y}$ is a vector representing the output argument of a module and $\mathbf{x}$ is a vector of input arguments of a module. For example, in an optimization, $\mathbf{y}$ can represent the optimal solution of the design variables and $\mathbf{x}$ can represent the initial design parameters and constraints.

Using such abstract macro-function interfaces, a modular system with different

simulation and optimization modules can be hierarchically and systematically connected. The communication between various modules of the system is carried out by interprocess communication in a multiworkstation network environment.

7.2 SYSCAD Module Descriptions

The entire stock of independent modules are classified into three types:

1. function: given $\mathbf{x}$, return $\mathbf{y} = \mathbf{f}(\mathbf{x})$
2. dfunction: given $\mathbf{x}$, return $\mathbf{y} = \mathbf{f}(\mathbf{x})$ and $\frac{d\mathbf{y}}{d\mathbf{x}}$.
3. main program: main programs that does not return any numerical data to other modules.

In order to use each type of module, standard definitions of input/output arguments of the three types of modules are presented.

Type 1: `function(n,m,x,y,k,s,ierror)`

Input arguments:

<code>n</code>	dimension of $\mathbf{x}$.
<code>m</code>	dimension of $\mathbf{y}$.
<code>k</code>	length of character string s .
<code>x</code>	vector of input parameters.
	The index of $\mathbf{x}$ is: $x(i)$, $i=0,1,2, \dots, n-1$.
<code>s</code>	character string of length k .

Output arguments:

y vector of outputs.
 The index of y is: $y(i)$, $i=0,1,2, \dots, m-1$.
ierror error message number. $ierror=0$ means no error.
s error message if $ierror$ is not equal to 0.

Type 2: `dfunction(n,m,x,y,dy,k,s,ierror)`

Input arguments:

same as in type 1 above.

Output arguments:

same as in type 1 above, and

dy vector of derivatives of y w.r.t x.
 Its dimension is $n * m$. $dy(1+n*j)$ is the
 derivative of $y(j)$ w.r.t. $x(i)$,
 $i=0,1, \dots, n-1$ and $j=0,1,2, \dots, m-1$.

Type 3: main program

Input argument:

none.

Output argument:

none.

Main program is to invoke all analysis and optimization process. It is the top level module.

A user can either use built in modules or write a simple user interface to integrate his own module into the system. The benefit is that a wide selection of design and analysis tools are ready to be used.

7.3 SYSCAD Overall File Structure

In its present form, SYSCAD consists of several file directories as shown in Fig. 7.1.

- SYSCAD is the top level directory.
- There are several directories under SYSCAD:
 - common: a directory containing all communication utilities. Several directories under it are:
 - * common: global common utility modules.
 - * function: communication modules for $y = f(x)$.
 - * dfunction: communication modules for $y = f(x)$ and $\frac{dy}{dx}$.
 - * lib: libraries for pipe communications.
 - simulator: simulation modules. It has several subdirectories.
 - * executable files: these files are executable programs ready to be run by their parent programs.

Figure 7.1: SYSCAD file structure.

- * **common**: communication modules for simulators only.
- * **src**: source file directories. Examples include **nilt**, Numerical Inversion of Laplace Transform; **lckt**, a linear simulator; **dem106**, a demonstration simulator.
- **dsimulator**: simulators with sensitivity analysis. This directory has several subdirectories similar to the simulator directory.
- **spec**, **dspec**: specification modules. These directories are similar to the simulator directory.
- **optimize**: optimization modules. This directory has several subdirectories similar to the simulator directory.
- **examples**: an input file directory. All analyses are invoked under this directory, which contains necessary input files for optimization, specification, simulation, etc.
- **help**: contains files for online help.

7.4 An Optimization Example Illustrating SYSCAD

In order to optimize the response of an electrical circuit, a user can use three independent modules: a circuit simulator, a problem formulator which formulates error functions for optimization, and an optimizer. SYSCAD uses inter-process communication techniques to run the three modules simultaneously.

A simple filter circuit is shown in Fig. 7.2. This is a low-pass filter with three circuit elements, L_1, C_1, L_2 . All of the three elements are design variables. The optimization is accomplished using the following modules.

1. **simulation module**. This module provides simulation for the circuit at each frequency point.

Figure 7.2: A LC filter circuit.

2. **specification module.** This module formulates the error functions for the optimization module.
3. **optimization module.** This module receives error functions from the specification module and changes design parameters systematically to accomplish the optimal design task.

The optimization module calls the specification module to get error functions and design variables; the specification module calls the simulation module to do simulation at required frequency points and at the same time sends error functions to the optimizer; the simulation module, which is at the lowest level, does simulations at the request of the specification module.

The interfaces of the three modules are formulated according to the generic function $y = f(x)$.

1. **Simulation:**

`function(n, m, x, y, k, s, ierror)`

Input arguments:

$n = 4$ is the dimension of x .
 $m = 1$ is the dimension of y .
 $k = 64$ is the length of s .
 $x = (\omega, L1, L2, C)$ are the circuit parameters.
 $y = V2$ at node 2, is the output
 of the filter at frequency ω .
 $s = \text{filename}$ is the input file for
 simulation. This file provides the
 circuit topology and element
 values for simulation purpose.

2. Specification:

`function(n, m, x, y, s, ierror)`
 $n = 3$ is the dimension of x .
 $m = 11$ is the dimension of y .
 $k = 64$ is the length of s .
 $x = (L1, L2, C1)$ is the vector
 of optimization variables.
 $y = (e1, e2, \dots, em)$ is the
 vector of error functions.
 $s = \text{filename}$ is the input file for the
 specification module.

3. Optimization:

Optimization is the main program.

Its input file contains the
starting point of design variables
and design constraints.

Output results are in the commercial software MATLAB format, so that the responses are ready to be plotted.

In case of using minimax optimizer MMLC[51], which requires sensitivity information, the above simulation module is:

`dfunction(n, m, x, y, dy, s, ierror)`
where `dy` is a `n * m` vector containing
derivative information.

dy is the vector of derivatives of responses with respect to all the circuit parameters at one frequency point.

The specification module has the same form:

`dfunction(n, m, x, y, dy, s, ierror)`

Note that in this function *dy* is different from that of the simulators. It contains the sensitivities of error functions with respect to the design variables.

Chapter 8

Conclusions and Further Research

8.1 Summary and Contributions

This research was motivated by the need for faster and more reliable automated design tools to meet the challenges of the rapid advances in the development of high-speed VLSI circuit and systems. Several contributions have been made to address the design of interconnections:

- A new multilevel optimization approach taking advantage of network topological decomposition is developed. Under this new approach, a large VLSI circuit is partitioned into a connection subcircuit and a set of subnetworks. Correspondingly, the overall minimax optimization is reformulated to yield a set of independent suboptimization problems. The suboptimizations are coordinated by a model coordination scheme.
- Parallel formulation of decomposition is developed. Formulas for estimating speed up factor by the parallel formulation are developed.
- The convergence property of the proposed technique is derived. It is analyzed through Gauss-Seidel relaxation analysis between multiple sets of nonlinear

equations which are derived from optimality conditions for the multiple suboptimizations. The two-level optimization formulation is generalized into n levels. Convergence criteria for both minimax and L_p type optimization decompositions are developed.

- The above algorithm is successfully implemented using our CAD framework featuring parallel processing. The multilevel formulation have been tested by both time and frequency domain optimization examples. Examples of interconnect optimizations demonstrated simultaneous improvement of signal integrity such as signal propagation delay and distortion at several vital connection ports. The actual CPU speed up is confirmed to be close to that from our proposed CPU estimation formulae. Results show that decomposition could be many times faster than standard optimization.

8.2 Suggestions for Further Research

1. The model coordination technique used in our formulation is applied to circuits of limited topological complexity. Further study is, therefore, necessary in order to fully exploit the potentials of decomposition. The formulations are to be tested by more complex and arbitrary topology networks.
2. Instead of using physical inspection, automatic partitioning algorithm can be applied to large scale distributed networks. Linking the partition program with the decomposition program automatically will further improve the design optimization process.
3. A more theoretically vigorous coordination scheme is the *goal coordination* scheme. As shown in Chapter 2, the circuit optimization formulated by goal

2 of/de 2

PM-1 3½"x4" PHOTOGRAPHIC MICROCOPY TARGET
NBS 1010a ANSI/ISO #2 EQUIVALENT

PRECISIONSM RESOLUTION TARGETS

coordination has clear theoretical interpretations. However, balancing the interaction variables of goal coordination can be difficult. Efficient formulations of goal coordination for large scale distributed network optimization will be an interesting and challenging topic.

4. When applying decomposition techniques to large scale optimization problems, a realistic solution to goal coordination is to use saddle point optimizer [11]. A saddle point is the point at which the objective function achieves its maximum with respect to some variables and achieves its minimum with respect to some other variables. Development and implementation of a state-of-the-art saddle point optimizer will be an important step toward applying decomposition techniques to practical large VLSI interconnection design problems.
5. The convergence properties of the optimization by decomposition are to be studied in parallel with relaxation techniques. As it has been shown in Chapter 5, the convergence of the decomposition is equivalent to solving a set of non-linear equations. Such non-linear equations are closely related to the relaxation formulations of a circuit. An interesting research direction is to combine waveform relaxation technique with optimization decomposition.

Bibliography

- [1] R. A. Sainati and T. J. Moravec, "Estimating high-speed circuit interconnect performance" *IEEE Trans. Circuits Syst.*, vol. 36, pp. 533-541, 1989.
- [2] D.H. Xie and M. S. Nakhla, "Delay and crosstalk simulation of high-speed VLSI interconnects with nonlinear terminations", *IEEE Int. Conf. Computer-Aided Design*, (Santa Clara, CA), pp. 66-69, 1991.
- [3] A. R. Djordjevic, T. K. Sarkar and R. F. Harrington, "Time-domain response of multiconductor transmission lines", *Proc. IEEE*, vol. 75, pp. 743-764, 1987.
- [4] R. Griffith and M. S. Nakhla, "Time-domain analysis of lossy coupled transmission lines", *IEEE Trans. Microwave Theory Tech.*, vol. 38, pp. 1480-1487, 1990.
- [5] T. Tang, M. S. Nakhla and R. Griffith, "Analysis of lossy multiconductor transmission lines using the asymptotic waveform evaluation technique", *IEEE Trans. Microwave Theory Tech.*, vol. 39, pp. 2107-2116, 1991.
- [6] Q. J. Zhang, S. Lum and M. Nakhla, "Minimization of delay and crosstalk in high-speed VLSI interconnects", *IEEE Trans. Microwave Theory Tech.*, vol. 40, pp. 1555-1563, 1992.

- [7] R. L. Liu, Q. J. Zhang and M. S. Nakhla, "A frequency domain approach to performance optimization of high-speed VLSI interconnects", *IEEE Trans. Microwave Theory Tech.*, vol. 40, pp. 2403-2411, 1992.
- [8] Yuji Wei, Q. J. Zhang and M. Nakhla, "Multilevel optimization of high speed VLSI interconnects using decomposition", *IEEE Custom Integrated Circuits Conference*, San Diego, CA, pp. 29.5.1-29.5.4, May, 1993.
- [9] Yuji Wei, Q. J. Zhang and M. Nakhla, "Multilevel optimization of high speed VLSI interconnect networks by decomposition", accepted by *IEEE Trans. Microwave Theory Tech.*.
- [10] G. B. Danzig and P. Wolfe, "The decomposition algorithm for linear programming", *Econometrica*, vol. 29, pp. 767 - 778, 1961.
- [11] D. A. Wismer, *Optimization Method for Large-Scale Systems with Applications*, New York: McGraw-Hill, 1971.
- [12] T. A. Edmunds and J. F. Bard, "Time-axis decomposition of large-scale optimal control problems", *Journal of Optimization Theory and Applications*, vol. 67, pp. 259-277, 1990.
- [13] C. D. Kolstad and L. S. Lasdon, "Derivative evaluation and computation experience with large bilevel mathematical programs", *Journal of Optimization Theory and Applications*, vol. 65, pp. 485-499, 1990.
- [14] L. S. Lasdon, *Optimization Theory for Large-Scale Systems*, New York: MacMillan, 1970.
- [15] A. M. Geoffrion, "Generalized benders decomposition", *Journal of Optimization Theory and Applications*, vol. 10, pp. 237-260, 1972.

- [16] A. Sangiovanni-Vincentelli, L-K Chen and L.O. Chua, "An efficient heuristic cluster algorithm for tearing large networks", *IEEE Trans. Circuits and Systems*, vol. CAS-24, pp. 709-711, 1977.
- [17] F. Luccio and M. Sami, "On the decomposition of minimally interconnected subnetworks", *IEEE Trans. Circuit Theory*, vol. CT-16, pp. 184-188, 1969.
- [18] G. Kron, *Diakoptics - A Piece-wise Solution of Large-Scale Systems*. London, England: MacDonald, 1971.
- [19] H. H. Happ, *Diakoptics and Networks*, New York: Academic Press, 1977.
- [20] H. H. Happ, "Diakoptics - the solution of system problems by tearing". *Proc. IEEE*, vol. 62, pp. 930-940, 1974.
- [21] F. F. Wu, "Solution of large-scale networks by tearing", *IEEE Trans. Circuits and Systems*, vol. CAS-23, pp. 706-713, 1976.
- [22] R. A. Rohrer, "Circuit partitioning simplified", *IEEE Trans. Circuits and Systems*, vol. CAS-35, pp. 2-5, 1988.
- [23] D. M. Himmelblau, *Decomposition of Large-Scale Problems*. Amsterdam: North-Holland, 1973.
- [24] M. G. Singh and A. Titli, *Systems: Decomposition, Optimization and Control*, New York: Pergman Press, 1978.
- [25] D. D. Siljak, *Decentralized Control Systems*, New York: Academic Press Limited, 1991.
- [26] L. Acar and U. Ozguner, "A locally computed suboptimal control strategy for a class of interconnected systems", *Journal of Optimization Theory and Applications*, vol. 62, pp. 189-210, 1989.

- [27] A. E. Salama, J. A. Starzyk and J. W. Bandler, "A unified decomposition approach for fault location in large analog circuits", *IEEE Trans. Circuits and Systems*, vol. CAS-31, pp. 609-622, 1984.
- [28] K. L. Lo and Y. M. Mahmoud, "A decoupled linear programming technique for power system state estimation", *IEEE Trans. Power Syst.*, vol. PWRS-1, pp. 154-160, 1986.
- [29] S. N. Talukdar, T. C. Giras and V. K. Kalyan, "Decomposition for optimal power flows", *IEEE Trans. Power App. Syst.*, vol. PAS-102, pp. 3877-3884, 1983.
- [30] G. C. Contaxis, C. Delkis and G. Korres, "Decoupled optimal load flow using linear or quadratic programming", *IEEE Trans. Power Syst.*, vol. PWRS 1, pp. 1-7, 1986.
- [31] R. Billinton and S. S. Sachdeva, "Real and reactive power optimization by sub optimum techniques", *IEEE Trans. Power App. Syst.*, vol. PAS-92, pp. 950-956, 1973.
- [32] H. Habibollahzadeh and J. A. Bubenko, "Application of decomposition techniques to short-term operation of hydrothermal power system", *IEEE Trans. Power Systems*, vol. PWRS-1, pp. 41-47, 1986.
- [33] R. Nadira, W. M. Lebow and P. B. Usono, "A decomposition approach to preventive planning of reactive volt ampere(var) source expansion", *IEEE Trans. Power Systems* vol. PWRS-1, pp. 72-77, 1987.
- [34] J. W. Bandler and Q. J. Zhang, "An automatic decomposition approach to optimization of large scale microwave systems", *IEEE Trans. Microwave Theory Tech.*, vol. MTT-35, pp. 1231-1238, 1987.

- [35] K. K. Low and S. W. Director, "A new methodology for the design centering of IC fabrication processes", *IEEE Trans. Computer-Aided Design*, vol. 10, pp. 895-903, 1991.
- [36] K. K. Low, "A design methodology of IC fabrication" Ph. D. dissertation, Carnegie-Mellon University, Pittsburgh, Pennsylvania, U.S.A., 1989.
- [37] R. Griffith, E. Chiprout, Q. J. Zhang and M. S. Nakhla, "A CAD framework for simulation and optimization of high-speed VLSI interconnections", *IEEE Trans. Circuit and Systems*, vol. CAS-39, pp. 906-906, 1992.
- [38] A. A. Malik, "Optimization of primitive gate networks using multiple output two-level minimizations", *IEEE Design Automation Conference*, pp. 449-453, 1992.
- [39] J. C. Limqueco and S. Muroga, "Optimizing large network by repeated local optimization using a windowing scheme", *IEEE International Symposium on Circuits and Systems*, pp. 1993 - 1996, 1992.
- [40] S. Toida, "A decomposition of a graph into dense subgraph", *IEEE Trans. Circuits and Systems*, vol. CAS-32, pp. 583-589, 1985.
- [41] A. R. Newton and A. L. Sangiovanni-Vincentelli, "Relaxation-based electrical simulation", *IEEE Trans. Electron Devices*, vol. ED-30, pp. 1184-1207, 1983.
- [42] J. K. White and A. Sangiovanni-Vincentelli, *Relaxation Techniques for the Simulation of VLSI Circuits*, Boston: Kluwer Academic, 1987.
- [43] M. Nakano, Y. Mae and I. Shirakawa, "A waveform relaxation method applicable to bipolar digital circuits", *Proceedings of IEEE*, pp. 2276 - 2279, 1991.

- [44] M. M. Alaybeyi, J. E. Bracken, J. Y. Lee, V. Raghavan, R.J. Trihy and R.A. Rohrer, "Exploiting partition in the asymptotic waveform evaluation(AWE)", *IEEE Custom Integrated Circuits and Conference* pp. 15.7.1 - 15.7.4, 1992.
- [45] N. B. G. Rabbat, A. L. Sangiovanni-Vincentelli and H. Y. Hsieh, "A multilevel newton algorithm with macromodelling and latency for the large-scale nonlinear circuits in the time domain", *IEEE Trans. Circuits and Systems*, vol. CAS 26, pp. 733 - 740, 1979.
- [46] R. Wang and O. Wing, "Waveform relaxation on tightly coupled systems", *Proceedings of IEEE*, pp. 2280 - 2283, 1991.
- [47] F-Y. Chang, "The generalized method of characteristics for waveform relaxation analysis of lossy coupled transmission lines", *IEEE Trans. Microwave Theory and Tech.*, vol. 37, pp. 2028 - 2038, 1989.
- [48] C. W. Ho, A. E. Ruehli and P. A. Brennan, "The modified nodal approach to network analysis", *IEEE Trans. Circuits and Systems*, vol. CAS-22, pp. 504-509, 1975.
- [49] J. W. Bandler and Q. J. Zhang, "Optimization techniques for modeling diagnosis and tuning", *Analog Methods for Computer-Aided Circuit Analysis and Diagnosis*, T. Ozawa Ed. New York: Marcel Dekker, 1988, Chap. 11.
- [50] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for minimax optimization", *Math. Program.*, vol. 20, pp. 49-62, 1981.
- [51] J. W. Bandler and W. M. Zuberek, " MMLC - a Fortran package for linearly constrained minimax optimization", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Rep SOS-82-5-U2, 1983.

- [52] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd Edition, New York: Addison-Wesley, 1984.
- [53] M. Ilie-Spong, "Block diagonal dominance for systems of nonlinear equations with applications to load flow calculations in power systems", *Mathematical Modelling*, vol. 5, pp. 275-297, 1984.
- [54] J. J. More, "Nonlinear generalizations of matrix diagonal dominance with application to Gauss-Siedel iterations", *SIAM J. Numerical Analysis*, vol. 9, No. 2, pp. 357-378, 1972.
- [55] H. S. Price, "A practical application of block diagonally dominant matrices", *Mathematics of Computation*, vol 19, pp. 307-313, 1965.
- [56] R. S. Varga, *Matrix Iterative Analysis*, New Jersey: Prentice-Hall, 1962.
- [57] J. W. Bandler, W. Kellermann and K. Madsen, "A superlinearly convergent min-max algorithm for microwave circuit design", *IEEE Trans. Microwave Theory Tech.*, vol. 33, pp. 1591-1530, 1985.
- [58] R. Tsay, "Exact zero skew clock routing algorithm", *IEEE. Computer Aided Design*, vol. 12, pp. 242-249, 1993.
- [59] J. P. Fishburn, "Clock skew optimization", *IEEE Trans. Computers*, vol. 39, pp. 945-951, 1990.
- [60] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, New York: Van Nostrand Reinhold, 1983.

Appendix A

A.1 Definition of Matrix Norm [53]

Let $\mathbf{x}^T = [x_1, x_2, \dots, x_n]$. The vector norm is defined as

$$\|\mathbf{x}\|_\infty = \max_i |x_i|, \quad i = 1, 2, \dots, n. \quad (\text{A.1})$$

Let $\mathbf{A}$ be a matrix with scalar element a_{ij} at row i and column j ,

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}. \quad (\text{A.2})$$

Let index set $N = \{1, 2, \dots, n\}$. The norm of a matrix is defined based on the vector norm as

$$\|\mathbf{A}\|_\infty = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} \quad (\text{A.3})$$

and

$$\|\mathbf{A}\|_\infty = \max_{i \in N} \sum_{j \in N} |a_{ij}|. \quad (\text{A.4})$$

where sup denotes the supremum of a function.

Let $\mathbf{A}$ be an $n \times n$ matrix partitioned into $K \times K$ blocks. Each block is a $n_i \times n_j$ matrix represented by $\mathbf{A}_{ij}$, $1 \leq i, j \leq K$,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1K} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2K} \\ \vdots & \vdots & & \vdots \\ \mathbf{A}_{K1} & \mathbf{A}_{K2} & \cdots & \mathbf{A}_{KK} \end{bmatrix}, \quad \sum_{i=1}^K n_i = n. \quad (\text{A.5})$$

Let index set $\mathcal{K} = \{1, 2, \dots, K\}$. The norm of matrix $\mathbf{A}$ is defined as

$$\|\mathbf{A}\|_{\infty} = \max_{i \in \mathcal{K}} \sum_{j \in \mathcal{K}} \|\mathbf{A}_{ij}\|_{\infty} \quad (\text{A.6})$$

where $\|\mathbf{A}_{ij}\|_{\infty}$ is defined by (A.4).

A.2 Definition of G-differentiable [54]

$\mathbf{F} : D \subset \mathcal{R}^n \rightarrow \mathcal{R}^m$ is G-differentiable at an interior point $\mathbf{x} \in D$ if there is an $m \times n$ matrix $\mathbf{A}$ such that for any $\mathbf{h} \in \mathcal{R}^n$,

$$\lim_{t \rightarrow 0} \frac{1}{|t|} \|\mathbf{F}(\mathbf{x} + t\mathbf{h}) - \mathbf{F}(\mathbf{x}) - t\mathbf{A}\mathbf{h}\| = 0. \quad (\text{A.7})$$

A.3 Kuhn-Tucker Conditions [11] [52]

Let the optimization problem be

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad (\text{A.8})$$

subject to

$$\mathbf{x} \in \mathbf{S}_1 \cap \mathbf{S}_2 \cap \mathbf{S}_3 \quad (\text{A.9})$$

where

$$\mathbf{S}_1 = (\mathbf{E}^n)^+, \quad (\text{A.10})$$

$$\mathbf{S}_2 = \{\mathbf{x} | w_i(\mathbf{x}) = 0, i = 1, 2, \dots, n_w\}, \quad (\text{A.11})$$

$$\mathbf{S}_3 = \{\mathbf{x} | g_i(\mathbf{x}) \leq 0, i = 1, 2, \dots, n_g\}, \quad (\text{A.12})$$

$\mathbf{x}$ is a n -vector of real variables, $f(\mathbf{x}), w_i(\mathbf{x}), g_i(\mathbf{x})$ are real valued functions defined on $\mathbf{S}_1$ and differentiable everywhere.

Let the Lagrange function be

$$L(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \sum_{i=1}^{n_w} \lambda_i w_i(\mathbf{x}) + \sum_{i=1}^{n_g} \mu_i g_i(\mathbf{x}). \quad (\text{A.13})$$

The necessary conditions for the point $(\mathbf{x}^0, \lambda^0, \mu^0)$ to be the solution to the optimization problem (A.8) are the Kuhn-Tucker conditions as the following:

$$\nabla_{\mathbf{x}} L(\mathbf{x}^0, \lambda^0, \mu^0) \geq 0, \quad (\text{A.14})$$

$$(\mathbf{x}^0)^T \nabla_{\mathbf{x}} L(\mathbf{x}^0, \lambda^0, \mu^0) = 0,$$

$$\mathbf{x}^0 \geq 0,$$

$$w_i(\mathbf{x}^0) = 0, \quad i = 1, 2, \dots, n_w, \quad (\text{A.15})$$

$$g_i(\mathbf{x}^0) \leq 0, \quad (\text{A.16})$$

$$\mu_i^0 g_i(\mathbf{x}^0) = 0,$$

$$\mu_i \geq 0 \quad i = 1, 2, \dots, n_g.$$

END

3 1 - 0 5 - 9 5

FIN