12-2016

Fast time- and frequency-domain finite-element methods for electromagnetic analysis

Woochan Lee
Purdue University

FAST TIME- AND FREQUENCY-DOMAIN FINITE-ELEMENT METHODS FOR ELECTROMAGNETIC ANALYSIS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Woochan Lee

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2016

Purdue University

West Lafayette, Indiana

Graduate School Form
30 Updated 12/26/2015

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Woochan Lee

Entitled

Fast Time- and Frequency-Domain Finite-Element Methods for Electromagnetic Analysis

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Dan Jiao
Chair

Byunghoo Jung

John A. Nyenhuis

Peide (Peter) Ye

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Dan Jiao

Approved by: V. Balakrishnan 9/22/2016

Head of the Departmental Graduate Program Date

To my family

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor and mentor Professor Dan Jiao for introducing me to the field of computational electromagnetics and for giving me a precious opportunity to join her research group at Purdue University. I have been able to reach my long-sought goal thanks to her deep insights into the unsolved nature of this field, as well as her full support of my efforts.

I also would like to thank other members of my Ph.D. committee: Professor Byunghoo Jung, Professor John Nyenhuis, and Professor Peide Ye for their consideration of and suggestions for this dissertation.

I'm grateful to all the members of On-Chip Electromagnetics Lab who spent time with me, giving me constant support and lasting friendships. These include the following: Dr. Jongwon Lee, Dr. Wenwen Chai, Dr. Duo Chen, Dr. Haixin Liu, Dr. Feng Sheng, Dr. Qing He, Dr. Saad Omar, Dr. Bangda Zhou, Dr. Md. Gaffar, Dr. Yanpu Zhao, Dr. Jin Yan, Miaomiao Ma. Dr. Gaffar and Dr. Yan, who spent the most time with me, always model how to be a good researcher.

My old friend Dr. Jongwon Lee is my mentor and he always inspires me to achieve good results. Dr. Jaeyoung Park and Dr. Seokmin Hong gave me good opinions and suggestions on how to succeed in my doctoral aspirations. Also, I have always been the recipient of good advice from the SNU alumni at Purdue University and my colleagues in Korea Military Academy and Korean Intellectual Property Office.

I am indebted to my parents and my wife. Without their support, I wouldn't have even been able to start this study. I hope I have made all of them as well as my daughters Juwon and Gawon proud.

TABLE OF CONTENTS

Page

LIST OF TABLES vii

LIST OF FIGURES viii

ABSTRACT xi

1 INTRODUCTION 1

1.1 Importance of Electromagnetic Analysis 1

1.2 Challenge and Recent Progress of Electromagnetic Analysis 1

1.3 Contributions of This Dissertation 4

2 STRUCTURE-AWARE TIME-DOMAIN FINITE-ELEMENT SOLVER . 7

2.1 Introduction 7

2.2 Formulations 8

2.3 Fast Structure-Aware **T**'s Solution Leveraging Manhattan-Type Geometry and Layered Permittivity 10

2.4 Numerical Results 14

2.5 Conclusions 18

3 FASTER STRUCTURE-AWARE DIRECT TDFEM SOLVER WITHOUT SACRIFICING TIME STEP SIZE 19

3.1 Introduction 19

3.2 Proposed Method 20

3.2.1 General Idea 20

3.2.2 Fast DC-mode Extraction from a New Problem 23

3.2.3 Synthesis of Solution of the Original Problem 26

3.3 Numerical Results 26

3.3.1 Two On-Chip Interconnect Structures 26

3.3.2 On-Chip Power Grid 29

Page

3.3.3 Rectangular Spiral Inductor 32
3.3.4 Suite of Large-Scale On-Chip Power Grids 34
3.4 Conclusions 39
4 A NEW EXPLICIT AND UNCONDITIONALLY STABLE TIME-DOMAIN FINITE-ELEMENT METHOD 40
4.1 Introduction 40
4.2 Proposed Method 41
4.3 Numerical Results 43
4.4 Conclusions 47
5 EXPLICIT AND UNCONDITIONALLY STABLE TDFEM FOR ANALYZING GENERAL LOSSY PROBLEMS 48
5.1 Introduction 48
5.2 Proposed Method 48
5.2.1 Explicit Time Marching Scheme based on Central Difference 52
5.2.2 Scaling 55
5.3 Numerical Results 56
5.3.1 Shorted On-Chip Stripline 56
5.3.2 On-Chip Power Grid 57
5.3.3 Rectangular Spiral Inductor 59
5.3.4 Lossy Multiscale Structure 59
5.4 Conclusions 62
6 STRUCTURE-AWARE TIME-DOMAIN FINITE-ELEMENT SOLVER FOR GENERAL FULL-WAVE ANALYSIS 64
6.1 Introduction 64
6.2 Proposed Method 65
6.3 Numerical Results 69
6.3.1 Stripline 69
6.3.2 A Suite of Striplines 70
6.3.3 Lossy Multiscale Structure 73

Page

6.4 Conclusions 77

7 SYMMETRIC POSITIVE-DEFINITE REPRESENTATION OF FREQUENCY-DOMAIN FINITE-ELEMENT SYSTEM MATRIX FOR EFFICIENT ELECTROMAGNETIC ANALYSIS 78

7.1 Introduction 78

7.2 Proposed Method 79

7.2.1 Transformed System 80

7.2.2 Absorbing Boundary Condition Imposition 85

7.2.3 Finding Non-Positive Definite Modes $\mathbf{V}_n$ 86

7.3 Numerical Results 87

7.3.1 Waveguide with Absorbing Boundary Condition 87

7.3.2 Demonstration of Accuracy and Efficiency 90

7.4 Conclusions 99

8 CONCLUSIONS 101

LIST OF REFERENCES 103

A Unstable Mode Determination Criterion in Forward-Difference based Time Discretization Scheme 106

VITA 108

LIST OF TABLES

Table Page

2.1 Algorithm for recovering unknowns 15

6.1 Algorithm for evaluating matrix exponential components 68

6.2 Simulation results along dielctric block extentions of the structure . . . 75

LIST OF FIGURES

Figure Page

2.1 The structure of **T** matrix. (a) Overall **T** matrix structure. (b) The structure of each diagonal block (layer) in $\mathbf{T}_{\xi\xi}$ (ξ = x, y, z), which is a block tridiagonal matrix with each tridiagonal block linearly proportional to each other. 10

2.2 Illustration of layers of x-unknowns. (y-unknowns and z-unknowns can be visualized using the same figure by switching the coordinates.) (a) 3-D view. (b) Cross-sectional view with red dots denoting unknowns perpendicular to the cross section. 11

2.3 Comparison of the matrix solution cost. 16

2.4 Accuracy validation of the proposed algorithm. 17

2.5 CPU time vs.N for simulating a suite of on-chip circuits. 17

3.1 Illustration of an on-chip interconnect. (a) 3-D view of the structure. (b) y-z plane view of the structure. 27

3.2 Simulation of an on-chip interconnect. 28

3.3 Side view of a shorted on-chip interconnect. 29

3.4 Accuracy validation of the proposed algorithm in simulating a far-end shorted on-chip interconnect. 30

3.5 Illustration of the structure of an on-chip power grid. (a) 3-D view. (b) x-z plane view. (c) y-z plane view. 31

3.6 Accuracy validation of the proposed algorithm in power grid simulation. 32

3.7 Illustration of a rectangular spiral inductor structure. 33

3.8 Accuracy validation of the proposed algorithm for the simulation of a rectangular spiral inductor with $\Delta t = 1.5 \times 10^{-13}$. 34

3.9 Illustration of a larger on-chip power grid structure. (a) x-y plane view. (b) x-z plane view. (c) y-z plane view. 35

3.10 Accuracy validation of the proposed algorithm in simulating a larger on-chip power grid. 36

Figure Page

3.11 CPU time vs. N for simulating a suite of on-chip power grids. (a) One solution time; (b) Factorization and one solution time. 38

4.1 Voltages of a parallel plate with different time steps compared with analytical solution. 44

4.2 Voltages of a mm-level parallel plate. 45

4.3 The bus structure configuration (unit: μm). 46

4.4 Accuracy validation of on-chip bus structure. 46

4.5 Total solution error plot of bus structure. 47

5.1 Accuracy validation of the proposed algorithm in simulating a shorted on-chip stripline. 57

5.2 Accuracy validation of the proposed algorithm in simulating on-chip power grid. 58

5.3 Accuracy validation of the proposed algorithm for the simulation of a rectangular spiral inductor. 60

5.4 Geometry of a lossy multiscale structure. 61

5.5 Accuracy validation of lossy multiscale structure. 62

6.1 The cross-sectional view of the stripline structure. 69

6.2 Accuracy validation of the current algorithm for a stripline case. 71

6.3 A suite of two stripline structures. 72

6.4 Cross-sectional view of the basic stripline block. 72

6.5 Accuracy validation of the proposed method. 73

6.6 Accuracy validation of the proposed algorithm with the structure of Fig. 5.4. 74

6.7 Pre-marching time comparison. 76

6.8 Marching time comparison. 76

6.9 Total elapsed time comparison. 77

7.1 Transformed eigenvalue system. 81

7.2 The controllability of condition number. 83

7.3 Dielectric loaded rectangular waveguide. 88

7.4 Reflection coefficient of dielectric loaded waveguide (b=10 mm). 89

Figure Page

7.5 Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a dielectric-loaded waveguide with 14,652 unknowns. 89

7.6 Solution time comparison. 90

7.7 Structure of parallel-plate waveguide with inhomogeneous vertical layer. 91

7.8 Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a parallel-plate waveguide with vertical inhomogeneous layer. 92

7.9 Input reactance versus frequency for a parallel-plate waveguide. 92

7.10 Iteration number in a parallel plate example for convergence comparison. 93

7.11 The structure of a cavity-backed patch antenna. 94

7.12 Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a cavity-backed patch antenna. 94

7.13 Input reactance versus frequency for a cavity-backed patch antenna. . . 95

7.14 Iteration number in a patch antenna example for convergence comparison. 96

7.15 Planar view of 6 by 6 patch antenna array. 97

7.16 Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a 6 by 6 patch array. 97

7.17 Iteration number comparison. 98

7.18 Solution time comparison. 99

ABSTRACT

Lee, Woochan Ph.D., Purdue University, December 2016. Fast Time- and Frequency-Domain Finite-Element Methods for Electromagnetic Analysis. Major Professor: Dan Jiao.

Fast electromagnetic analysis in time and frequency domain is of critical importance to the design of integrated circuits (IC) and other advanced engineering products and systems. Many IC structures constitute a very large scale problem in modeling and simulation, the size of which also continuously grows with the advancement of the processing technology. This results in numerical problems beyond the reach of existing most powerful computational resources. Different from many other engineering problems, the structure of most ICs is special in the sense that its geometry is of Manhattan type and its dielectrics are layered. Hence, it is important to develop structure-aware algorithms that take advantage of the structure specialties to speed up the computation. In addition, among existing time-domain methods, explicit methods can avoid solving a matrix equation. However, their time step is traditionally restricted by the space step for ensuring the stability of a time-domain simulation. Therefore, making explicit time-domain methods unconditionally stable is important to accelerate the computation. In addition to time-domain methods, frequency-domain methods have suffered from an indefinite system that makes an iterative solution difficult to converge fast.

The first contribution of this work is a fast time-domain finite-element algorithm for the analysis and design of very large-scale on-chip circuits. The structure specialty of on-chip circuits such as Manhattan geometry and layered permittivity is preserved in the proposed algorithm. As a result, the large-scale matrix solution encountered in the 3-D circuit analysis is turned into a simple scaling of the solution of a small 1-D

matrix, which can be obtained in linear (optimal) complexity with negligible cost. Furthermore, the time step size is not sacrificed, and the total number of time steps to be simulated is also significantly reduced, thus achieving a total cost reduction in CPU time.

The second contribution is a new method for making an explicit time-domain finite-element method (TDFEM) unconditionally stable for general electromagnetic analysis. In this method, for a given time step, we find the unstable modes that are the root cause of instability, and deduct them directly from the system matrix resulting from a TDFEM based analysis. As a result, an explicit TDFEM simulation is made stable for an arbitrarily large time step irrespective of the space step.

The third contribution is a new method for full-wave applications from low to very high frequencies in a TDFEM based on matrix exponential. In this method, we directly deduct the eigenmodes having large eigenvalues from the system matrix, thus achieving a significantly increased time step in the matrix exponential based TDFEM.

The fourth contribution is a new method for transforming the indefinite system matrix of a frequency-domain FEM to a symmetric positive definite one. We deduct non-positive definite component directly from the system matrix resulting from a frequency-domain FEM-based analysis. The resulting new representation of the finite-element operator ensures an iterative solution to converge in a small number of iterations. We then add back the non-positive definite component to synthesize the original solution with negligible cost.

1. INTRODUCTION

1.1 Importance of Electromagnetic Analysis

The past few decades have witnessed the dramatic rise of computational electromagnetics. Electromagnetic simulation is becoming an increasingly important technology and numerous methods have been developed [1–4]. The reasons include higher operating frequencies and higher complexity of the structures to be designed. A lot of efforts have been made to scale down semiconductor structures such as adopting FinFET and 3-D packaging, they make an understading of electromagetic nature much more difficult, thus more sophisticated computational electromagnetic techniques are required to address the issue. Even though computational electromagnetics is an 'invisible hand' buried in electronic design tool, the compuational electromagnetics is routinely used to maximize the performance of the real world product.

The analysis of on-chip circuits across a broad range of electromagnetic spectrum is of critical importance to the higher-performance design of integrated circuits (IC) and systems [5–15]. Many VLSI circuit structures such as a global power grid network constitute a very large scale problem in modeling and simulation, the size of which also continuously grows with the advancement of the processing technology.

1.2 Challenge and Recent Progress of Electromagnetic Analysis

A straightforward solution to the very large-scale electromagnetic problem would result in a numerical system that is beyond the capability of existing most powerful computational resources. Therefore, fast electromagnetic solvers with high capacity are called for to guide the very large-scale IC design in a fast turnaround time with uncompromised accuracy.

Different from many other engineering problems, the structure of an on-chip circuit is special in the sense that its geometry is of Manhattan type and its dielectrics are layered. Not taking advantage of these specialties would unnecessarily slow down the computation; however, preserving the structure specialties and taking advantage of them in a numerical solution is not a straightforward task either. Take the layered dielectrics as an example. If one employs a frequency-domain solver to analyze the on-chip circuits, the resultant system matrix is composed of permittivity, conductivity, and permeability related terms. Since conductivity and permeability are not layered, the layered property of the permittivity cannot be preserved and taken advantage of in the solution of the frequency-domain system matrix. The same is true for an implicit time-domain method. If one employs an explicit time-domain method, although only the weighted sum of the permittivity- and conductivity-related matrices needs to be solved, since the conductivity is not layered as the layout structure is different in different dielectric layers, the layered structure of the permittivity is also lost in the numerical solution of an explicit time-domain method.

There have been structure-preserving algorithms developed to exploit the structure specialty of on-chip circuits. In [5], a time-domain layered finite-element reduction-recovery (LAFE-RR) method [5] was developed to solve large-scale IC design problems at high frequencies. In this method, the layered property of dielectrics is employed to perform system reduction analytically from multiple layers to a single layer which is a 2-D numerical system. With Manhattan geometry taken into account, the 2-D single-layer system is further reduced to a single line that is 1-D in the hierarchical FE-RR method [12]. However, since the conductivity is not layered, in [5, 12], the conductivity-related term is moved to the right hand side of the system matrix equation to enable the analytical system reduction. This makes the time step required for a stable time-domain simulation much smaller than that permitted by a traditional explicit time-domain method, and hence slowing down the overall computation. In [6], a fast marching method is developed to address the time step issue. In this method, the conductivity-related term is kept to the left hand side of the marching equation

in time, and a preconditioner that permits a LAFE-RR solution is constructed to iteratively solve the system matrix with an expedited convergence. The preconditioner is built by replacing some metal layers with solid metal planes, the performance of which is problem dependent for accelerating the original matrix solution.

A time-domain FEM solution of the second-order vector-wave equation can be transformed to a first-order system of equation, which can then be solved analytically by matrix exponential framework [16, 17]. The usefullness of this framework is that it supports as large time step as the maximum time step soley determined by accuracy. However, the evaluation of matrix exponential generally consumes a lot of computation resources. A reliable algorithm for matrix exponential framework is still an active research area [17, 18].

The overall computational efficiency of a time-domain method is determined by not only the cost at each time step, but also the total number of time steps required to finish one simulation. Among existing time-domain methods, explicit methods can avoid solving a matrix equation. However, their time step is traditionally restricted by the space step for ensuring the stability of a time-domain simulation. Recently, an explicit and unconditionally stable TDFEM method is developed to overcome this problem [19]. In this method, for any given time step Δt , the root cause of the instability is analytically found to be the eigenmodes whose eigenvalues are higher than $4/\Delta t^2$. The method in [19] begins with a preprocessing step that finds the space of stable eigenmodes followed by an explicit time marching stable for the given time step no matter how large it is. To preserve the advantage of an explicit time-domain method in avoiding solving a matrix equation, the preprocessing step is performed by using the conventional explicit marching. Although the time window to be simulated in the preprocessing can be much shorter than the total time window to be simulated, the performance of the preprocessing step may become limited in certain applications.

Frequency-domain analysis is essential for many engineering problems such as RF engineering. However, the frequency domain analysis of large-scale electromagnetic problems is also challenging. One notable problem is a low frequency breakdown due

to unbalanced matrix norm in the problem solving. A theoretically rigorous full-wave method addressing this problem is proposed [20, 21]. Also, The system matrix for frequency domain analysis is generally indefinite, which contains both negative and positive eigenvalues. For both iterative and direct solution, the negative eigenvalue contribution or non-positive definite modes are acting as a hindrance against the fast solver.

As the feature size is scaled down, interconnect structure becomes a bottleneck and a challenge of the design of VLSI circuits [22, 23]. Also, along with higher operating frequencies, inductance and capacitance should be taken into consideration. This trend has led to a series of transitions from R to RLC models [23]. With past RLC-based interconnect extraction, significant mismatch between the experiment and the simulation was observed at multi-GHz frequencies but full-wave electromagnetics based modeling produces an accurate simulation [24]. The full-wave electromagnetics based solution captures the exact behavior of the circuits [23]. Therefore, the proposed methods in this dissertation can play an important role in circuit design and analysis in addition to general electromagnetic analysis.

1.3 Contributions of This Dissertation

In this thesis, first, an efficient structure-aware method was developed to preserve the layered permittivity and Manhattan-type geometry in an explicit time-domain finite-element method (TDFEM) for analyzing very large scale integrated circuit problems. Different from [6], the method is a direct solution that avoids the common problems of an iterative solution. However, the method requires the computation of a matrix exponential. Theoretically speaking, this matrix exponential can be computed from the sum of a finite number of terms with guaranteed convergence irrespective of the choice of time step. However, numerically, for the sum to converge fast with a fewer number of terms, the time step used, though much larger than that in [5], is still

smaller than that allowed by a traditional central-difference based explicit TDFEM method. As a consequence, the overall computational efficiency is compromised.

The aforementioned problem is then solved by a faster structure-aware time-domain finite-element algorithm described in Chapter 3. In this algorithm, the structure specialty of on-chip circuits such as Manhattan geometry and layered permittivity is equally preserved. The large-scale matrix solution encountered in the 3-D circuit analysis is turned into a simple scaling of the solution of a small 1-D matrix, which can be obtained in linear (optimal) complexity with negligible cost. Furthermore, the time step size is not sacrificed, and the total number of time steps to be simulated is also significantly reduced, thus achieving a total cost reduction in CPU time.

In addition to significantly reducing the total computational cost at each time step, contributions are also made in this thesis to reduce the total number of time steps required to finish one simulation. Specifically, a new method for making an explicit time-domain finite-element method unconditionally stable is developed for general electromagnetic analysis. In this method, for a given time step, no matter how large it is, we upfront adapt the TDFEM numerical system to exclude the source of instability. As a result, an explicit TDFEM simulation is made stable for an arbitrarily large time step irrespective of the space step. This method has also been successfully extended to analyze general lossy problems where both dielectrics and conductors can be inhomogeneous and lossy.

Also, a new method for full-wave applications from low to very high frequencies in a TDFEM is proposed based on matrix exponential. Combining the capability of full-wave application of matrix exponential platform and instability modes exclusion for making the norm of the system matrix smaller, we exclude the unstable modes having large eigenvalues from the system matrix. Thus, we can achieve a significantly increased time step with a small number of series expansion terms for matrix exponential.

Our idea extends to frequency domain analysis. We deduct non-positive definite contributions directly from the system matrix resulting from a frequency-domain

finite-element based analysis. It has a spectral radius less than 1, and a controllable condition number as well. Then the above deducted contributions added back to synthesis the total solution with negligible cost. With such a new representation of the finite-element operator, an iterative solution is ensured to converge in a small number of iterations.

2. STRUCTURE-AWARE TIME-DOMAIN FINITE-ELEMENT SOLVER

2.1 Introduction

Many engineering problems have arbitrarily shaped structures and involve inhomogeneous materials. However, there also exist a great number of engineering problems that have certain structure specialty. Not taking advantage of the structure specialty would unnecessarily slow down the computation; however, preserving the structure specialty and taking advantage of it in the numerical solution is not a straightforward task either. For example, the very large-scale integrated (VLSI) circuit is an important class of engineering problems. For this class of problems, the structure specialty lies in two aspects: Manhattan-type geometry and layered permittivity. The former can certainly be used to simplify geometrical modeling. For example, brick elements become a natural choice for discretization. However, taking advantage of the layered permittivity is not an easy task. If one employs a frequency-domain finite-element method (FEM) or an implicit time-domain FEM to solve a VLSI circuit problem, since a weighted sum of the mass, stiffness, and conductivity-related matrices need to be solved, the layered structure in the permittivity cannot be preserved and taken advantage of in the solution of the system matrix. If one employs an explicit time-domain finite-element method, although only the weighted sum of the mass and the conductivity-related matrix needs to be solved, since the layout structure is different in different dielectric layers, the layered structure of the permittivity is also ruined in the numerical solution. In this chapter, we present an efficient time-domain finite-element algorithm that preserves the layered property of the permittivity and the Manhattan-type structure in the direct solution of the underlying system matrix. As a result, the linear proportionality of the matrix blocks can be fully exploited,

and the matrix solution of a large-scale matrix becomes a simple scaling. The contents of this chapter have been extracted and revised from the following publication: Woochan Lee and Dan Jiao, "Structure-aware time-domain finite-element method for efficient simulation of VLSI circuits," 2014 IEEE Antennas and Propagation Society International Symposium (APSURSI). 2014.

2.2 Formulations

A time-domain FEM solution of the second-order vector-wave equation for an integrated circuit problem results in the following linear system of equations

$$\mathbf{T}\ddot{u}(t) + \mathbf{R}\dot{u}(t) + \mathbf{S}u(t) = \dot{I}(t) \tag{2.1}$$

in which $\mathbf{T}$ is a mass matrix, $\mathbf{R}$ is associated with conductivity, $\mathbf{S}$ is a stiffness matrix, u is the field solution vector, and I is a vector of current sources. The single dot above a letter denotes a first-order time derivative, while the double dots denote a second-order time derivative. The $\mathbf{T}$, $\mathbf{R}$, and $\mathbf{S}$ are assembled from their elemental contributions as the following:

$$\mathbf{T}^e = \mu_0 \varepsilon \langle \mathbf{N}_i, \mathbf{N}_j \rangle \tag{2.2}$$

$$\mathbf{R}^e = \mu_0 \sigma \langle \mathbf{N}_i, \mathbf{N}_j \rangle \tag{2.3}$$

$$\mathbf{S}^e = {\mu_r}^{-1} \langle \nabla \times \mathbf{N}_i, \nabla \times \mathbf{N}_j \rangle \tag{2.4}$$

where ε is permittivity, σ is conductivity, μ_0 is free-space permeability, μ_0 is relative permeability, $\mathbf{N}$ is the vector basis employed to represent electric field $\mathbf{E}$, and $\langle \cdot, \cdot \rangle$ denotes an inner product. The layered property of the permittivity manifests itself in the mass matrix $\mathbf{T}$. In view of this, in [6], the $\mathbf{R}$ matrix is disconnected from the most advanced time step, i.e. moved to the right hand side to carry out an explicit time marching of (2.1). However, the resultant time step for a stable simulation is significantly reduced to the level of $\varepsilon/\sigma \approx 10^{-19}$ s. The fast marching method in [6] mitigates the problem, but the performance is problem dependent. Here, we propose

to first transform (2.1) to the following first-order system of equation without any approximation

$$\begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{T} & \mathbf{0} \end{bmatrix} \frac{d}{dt} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} + \begin{bmatrix} \mathbf{S} & 0 \\ 0 & -\mathbf{T} \end{bmatrix} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} = \begin{Bmatrix} \dot{I} \\ 0 \end{Bmatrix} \tag{2.5}$$

which can then be analytically converted to

$$\frac{d}{dt} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} + \begin{bmatrix} 0 & -\mathbf{I} \\ \mathbf{T}^{-1}\mathbf{S} & -\mathbf{T}^{-1}\mathbf{R} \end{bmatrix} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} = \begin{Bmatrix} 0 \\ \mathbf{T}^{-1}\dot{I} \end{Bmatrix}. \tag{2.6}$$

As can be seen, the resultant new system of equation only requires the solution of T to obtain the solution of (2.1) instead of the weighted sum of the T and R matrix. However, a stability analysis reveals that an explicit marching on (2.6) would again result in a small time step. We thus propose to solve (2.6) by an analytical formula [16]. Let (2.6) be denoted in short by

$$\frac{d}{dt}\tilde{u} + \mathbf{M}\tilde{u} = \tilde{b}. \tag{2.7}$$

Its solution is analytically known as

$$\tilde{u}(t) = e^{-\mathbf{M}t} \left[\int e^{\mathbf{M}t} \tilde{b}\,(t)\, dt + C \right], \tag{2.8}$$

where C is an initial condition. Numerically, (2.8) can be evaluated as [16]

$$\tilde{u}^{n+1} = \frac{\tilde{b}^{n+1}\Delta t}{2} + e^{-\mathbf{M}\Delta t} \left[\frac{\tilde{b}^{n}\Delta t}{2} + \tilde{u}^{n} \right], \tag{2.9}$$

where the matrix-exponential term can be obtained from the sum of multiple matrix-vector multiplication $\mathbf{M}x$, which can be efficiently computed as

$$\mathbf{M} \begin{Bmatrix} x1 \\ x2 \end{Bmatrix} = \begin{Bmatrix} -x2 \\ \mathbf{T}^{-1}(\mathbf{S} \cdot x1 - \mathbf{R} \cdot x2) \end{Bmatrix} \tag{2.10}$$

Theoretically, (2.8) allows for the use of any large time step. Numerically, we choose a time step that only requires a small number of terms to obtain the matrix exponential.

2.3 Fast Structure-Aware T's Solution Leveraging Manhattan-Type Geometry and Layered Permittivity

Since on-chip circuits are Manhattan-type structures, a brick-element based discretization is ideal for use without sacrificing accuracy in geometrical modeling. The $\mathbf{T}$ matrix with a brick-element based discretization can be naturally decomposed into $\mathbf{T_{xx}}$, $\mathbf{T_{yy}}$, and $\mathbf{T_{zz}}$ diagonal blocks due to the orthogonality of x, y, and z directions, as illustrated in Fig. 2.1(a).

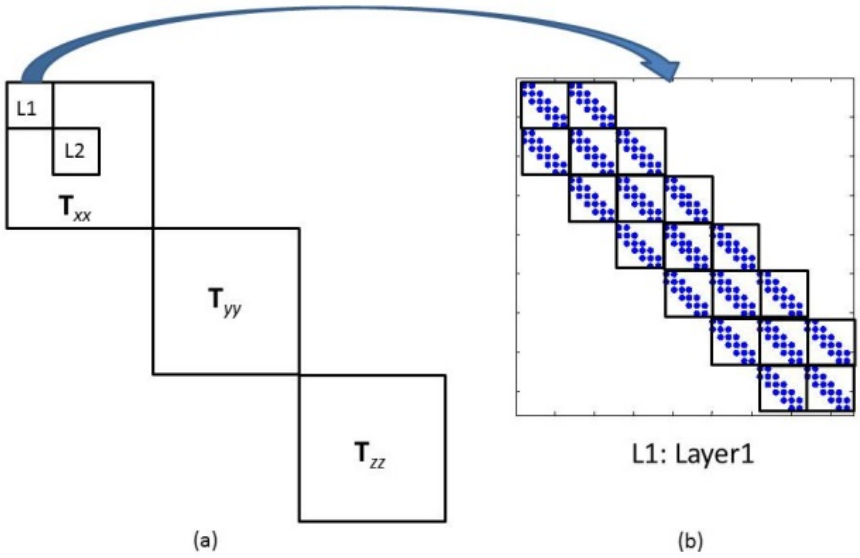


Fig. 2.1. The structure of $\mathbf{T}$ matrix. (a) Overall $\mathbf{T}$ matrix structure. (b) The structure of each diagonal block (layer) in $\mathbf{T}_{\xi\xi}$ (ξ = x, y, z), which is a block tridiagonal matrix with each tridiagonal block linearly proportional to each other.

With proper ordering of unknowns, each of $\mathbf{T_{xx}}$, $\mathbf{T_{yy}}$, and $\mathbf{T_{zz}}$ can further be structured to a block diagonal matrix consisting of L1, L2, etc., shown in Fig. 2.1(a). This is because the matrix blocks in different layers are fully decoupled, where the *layer* here refers to the region where the x-, y-, and z-orientated unknowns reside. The

x-unknown, y-unknown, and z-unknown layer is, respectively, normal to the x-, y-, and z-direction. To see this more clearly, Fig. 2.2(a) illustrates a 3-D view of the multiple layers of x-unknowns, with its cross-sectional view shown in Fig. 2.2(b), where each red dot represents one x-unknown. The layers of y-unknowns can be visualized by replacing the x, y, and z coordinates in Fig. 2.2 by y, z, and x respectively. Similarly, the layers of z-unknowns can be visualized by replacing the x, y, and z in Fig. 2.2 by z, x, and y respectively.

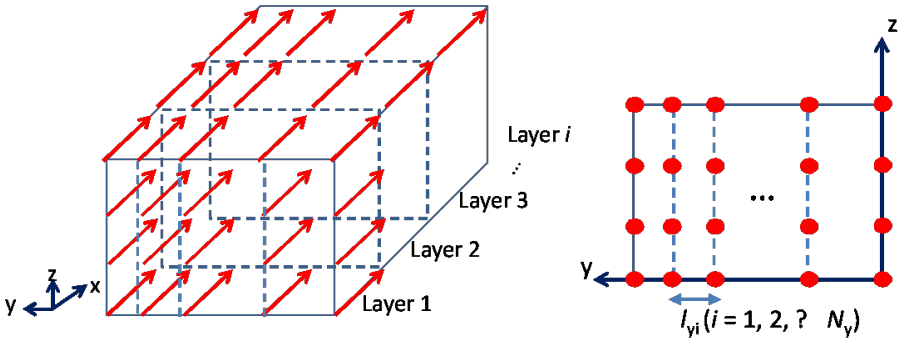


Fig. 2.2. Illustration of layers of x-unknowns. (y-unknowns and z-unknowns can be visualized using the same figure by switching the coordinates.) (a) 3-D view. (b) Cross-sectional view with red dots denoting unknowns perpendicular to the cross section.

From Fig. 2.2(a), it can be seen clearly that the matrix formed for x-unknowns in each layer is completely decoupled from that formed in another layer since the unknowns are internal to the layer. As a result, the $\mathbf{T_{xx}}$ in Fig. 2.1(a) is a block diagonal matrix, with each block representing the $\mathbf{T_{xx}}$ in one layer. The same is true for $\mathbf{T_{yy}}$ and $\mathbf{T_{zz}}$, as evident by treating the red arrows/dots in Fig. 2.2 as y-, and z-unknowns respectively.

In each layer, if we order the unknowns line by line, i.e., order all the unknowns along one vertical (or horizontal) line shown in Fig. 2.2(b), then move along the

horizontal (or vertical) direction to the unknowns on the next line, we will obtain a block tridiagonal matrix as shown in Fig. 2.1(b), which can be written as

$$\begin{bmatrix} \mathbf{A}_0 & \mathbf{B}_0 & & & & & & \\ \mathbf{B}_0 & \mathbf{A}_0+\mathbf{A}_1 & \mathbf{B}_1 & & & & & \\ & \mathbf{B}_1 & \mathbf{A}_1+\mathbf{A}_2 & \mathbf{B}_2 & & & & \\ & & \mathbf{B}_2 & \mathbf{A}_2+\mathbf{A}_3 & & & & \\ & & & \ddots & & & & \\ & & & & \ddots & & & \\ & & & & & \mathbf{A}_{L-2}+\mathbf{A}_{L-1} & \mathbf{B}_{L-1} & \\ & & & & & & \mathbf{B}_{L-1} & \mathbf{A}_{L-1} \end{bmatrix} \tag{2.11}$$

where each $\mathbf{A}_i$ (i= 0, 1, 2, , $L-1$) represents a sparse matrix formed on a single line. Due to the layered permittivity and the Manhattan geometry leveraged by the brick-element based discretization, the $\mathbf{A}_i$ for x-, y-, and z-unknowns respectively can be written as

$$\begin{aligned} \mathbf{A}_{i,x} &= \tfrac{l_{y,i}}{l_{y,0}}\mathbf{A}_{0,x}, \quad i = 0,1,2,...,N_y \\ \mathbf{A}_{i,y} &\sim \tfrac{l_{x,i}}{l_{x,0}}\mathbf{A}_{0,y}, \quad i = 0,1,2,...,N_x \\ \mathbf{A}_{i,z} &\sim \tfrac{l_{x,i}}{l_{x,0}}\mathbf{A}_{0,z}, \quad i = 0,1,2,...,N_x \end{aligned} \tag{2.12}$$

where Nx and Ny are, respectively, the number of unknowns along the x- and y-direction; $l_{\xi,i}(\xi = \text{x,y,z})$ is the i-th segment length along ξ-direction; $\mathbf{A}_{0,\xi}(\xi = \text{x,y,z})$ is the mass matrix $\mathbf{T}$ formed by unknowns along the first line in the cross section that is perpendicular to the x-, y-, and z-direction respectively. For y-orientated unknowns, the line direction for ordering unknowns is chosen as z instead of x, because in this way each region in between two lines has the same permittivity configuration. We can also choose x as the line direction. The only change is to incorporate a permittivity ratio into $\mathbf{A}_{i,y}$. Equation (2.12) gives $\mathbf{A}_i$ matrix components in each layer shown in (2.11). For different layers, $\mathbf{A}_{i,x}$and $\mathbf{A}_{i,y}$ are the same; but $\mathbf{A}_{i,z}$ needs to be scaled by a ratio of permittivity since the permittivity in each z-layer (one dielectric stack) is different. As for $\mathbf{B}_i$, it is nothing but

$$\mathbf{B}_i = 0.5\mathbf{A}_i \tag{2.13}$$

for unknowns along any direction. As a result, all the Ai and Bi blocks in (2.11) are linearly proportional to each other. Furthermore, each of the Ai and Bi is tridiagonal. Hence, the solution of $\mathbf{T}$ becomes a simple scaling of the solution of a small tridiagonal matrix as follows.

For the matrix shown in (2.11), we perform an analytical line reduction (the union of the number of unknowns in each block Ai forms a line) to a single line based on the proportionality of the blocks, from which we recover the solution anywhere else. The reduced matrix can be expressed by the following

$$\begin{aligned} \tilde{\mathbf{A}}_{i,\xi} &= \mathbf{A}_{i-1,\xi} + \mathbf{A}_{i,\xi} - \mathbf{B}_{i-1,\xi}\tilde{\mathbf{A}}^{-1}_{i-1,\xi}\mathbf{B}_{i-1,\xi} = s_\xi(i)\cdot\mathbf{A}_{0,\xi}, \quad i = 1,2,...,L-1 \\ \tilde{\mathbf{A}}_{i,\xi} &= \mathbf{A}_{i-1,\xi} - \mathbf{B}_{i-1,\xi}\tilde{\mathbf{A}}^{-1}_{i-1,\xi}\mathbf{B}_{i-1,\xi} = s_\xi(i)\cdot\mathbf{A}_{0,\xi}, \qquad i = L \end{aligned} \tag{2.14}$$

where L=Ny, Nx, and Nx respectively along the x-, y-, and z-direction. As can be seen, no inverse and matrix-matrix products are needed for the computation of $\tilde{\mathbf{A}}_{i,\xi}$, since the blocks involved are all linearly proportional to $\mathbf{A}_{0,\xi}$. We only need to calculate the scaling coefficients s_ξ based on the edge length ratio as the following

$$\begin{aligned} s_\xi(i) =& length\_ratio\_\xi[i-1] + length\_ratio\_\xi[i] \\ & - 0.25\cdot(length\_ratio\_\xi[i-1])^2\cdot[s_\xi(i-1)]^{-1}(i<L); \\ s_\xi(i) =& length\_ratio\_\xi[i-1] - \\ & 0.25\cdot(length\_ratio\_\xi[i-1])^2\cdot[s_\xi(i-1)]^{-1}(i=L) \end{aligned} \tag{2.15}$$

where the length ratio $length\_ratio\_\xi(\xi = x,y,z)$, based on (2.12), is given by

$$\begin{aligned} length\_ratio\_x[i] &== \tfrac{l_{y,i}}{l_{y,0}} \\ length\_ratio\_y[i] &= length\_ratio\_z[i] = \tfrac{l_{x,i}}{l_{x,0}}. \end{aligned} \tag{2.16}$$

The right hand side b is also analytically reduced to a single-line based right hand side as the following

$$\begin{aligned} \tilde{b}_{i,\xi} &= b_{i,\xi} - \mathbf{B}_{i-1,\xi}\tilde{\mathbf{A}}^{-1}_{i-1,\xi}\tilde{b}_{i-1,\xi} \\ &= b_{i,\xi} - 0.5\times length\_ratio\_\xi[i-1]\times[s_\xi(i-1)]^{-1}\cdot\tilde{b}_{i-1,\xi} \end{aligned} \tag{2.17}$$

in which $[i = 1,2,...,L, \xi = x,y,z$. After the unknowns in the last line is solved from the reduced single-line system

$$x_i = \tilde{\mathbf{A}}^{-1}_{i,\xi}\tilde{b}_{i,\xi} = [s_\xi(i)]^{-1}\mathbf{A}^{-1}_{0,\xi}\tilde{b}_{i,\xi}, i = L. \tag{2.18}$$

The unknowns on other lines are obtained recursively from the following:

$$\begin{aligned} x_i &= \tilde{\mathbf{A}}_{i,\xi}^{-1}(\tilde{b}_{i,\xi} - \mathbf{B}_{i,\xi}x_{i+1}) \\ &= [s_\xi(i)]^{-1}\mathbf{A}_{0,\xi}^{-1}\tilde{b}_{i,\xi} - 0.5[s_\xi(i)]^{-1}length\_ratio\_\xi[i]x_{i+1}, \end{aligned} \qquad (i = L-1, ...2, 1, 0). \tag{2.19}$$

The aforementioned algorithm for recovering the $\xi = x, y, z$ direction unknowns is shown in Algorithm 2.1. In this algorithm, Step 1 is to generate the right hand side vector shown in (2.17); Step 2 produces the result of $\mathbf{A}_{0,\xi}^{-1}\tilde{b}_{i,\xi}$ with i being the last line index. The u_ξ and v_ξ are the pre-computed UV factors of tridiagonal matrix $\mathbf{A}_{0,\xi}$, and the solution is stored in vector *temp*. Step 3 generates the solution of the last line shown in (2.18). Finally, Step 4 is to compute the solutions on all the other lines from (2.19).

The overall computation is the solution of one tridiagonal matrix, and all the other steps are simple algebraic operations whose computational cost is negligible. The tridiagonal matrix is the $\mathbf{A}_{0,\xi}(\xi = x, y, z)$ matrix, whose size is 1-D. Its solution can be accomplished by UV factorization for tridiagonal matrices in linear complexity with negligible cost [25].

2.4 Numerical Results

We first compare the matrix solution cost of the proposed method with that of the conventional brick-element based TDFEM that employs the multifrontal based direct solver. The solution time for one right hand side is shown in Fig. 2.3(a), while the factorization and one solution time is shown in Fig. 2.3(b). It is evident that the proposed method costs much less in matrix solution time, and it also has linear complexity.

Table 2.1.
Algorithm for recovering unknowns

Algorithm 2.1: Solving $\xi(\xi = x, y, z)$ -unknowns
1. for $i = 1, 2, ..., L$
1.1. $b[i] = b[i] - 0.5 \cdot length\_ratio\_\xi[i-1] \cdot (s\_\xi[i-1])^{-1} \cdot b[i-1]$
2. $inv\_uv(u\_\xi,\ v\_\xi,\ b[L],\ temp)$
3. $x[L] = (s\_\xi[L])^{-1} \cdot temp$
4. for $i = (L-1), ...1, 0$
4.1. $inv\_uv(u\_\xi, v\_\xi, b[i], temp)$
4.2. $x[i] = (s\_\xi[i])^{-1} \cdot temp$
$-0.5 \cdot (s\_\xi[i])^{-1} \cdot (length\_ratio\_\xi[i]) \cdot x[i+1]$

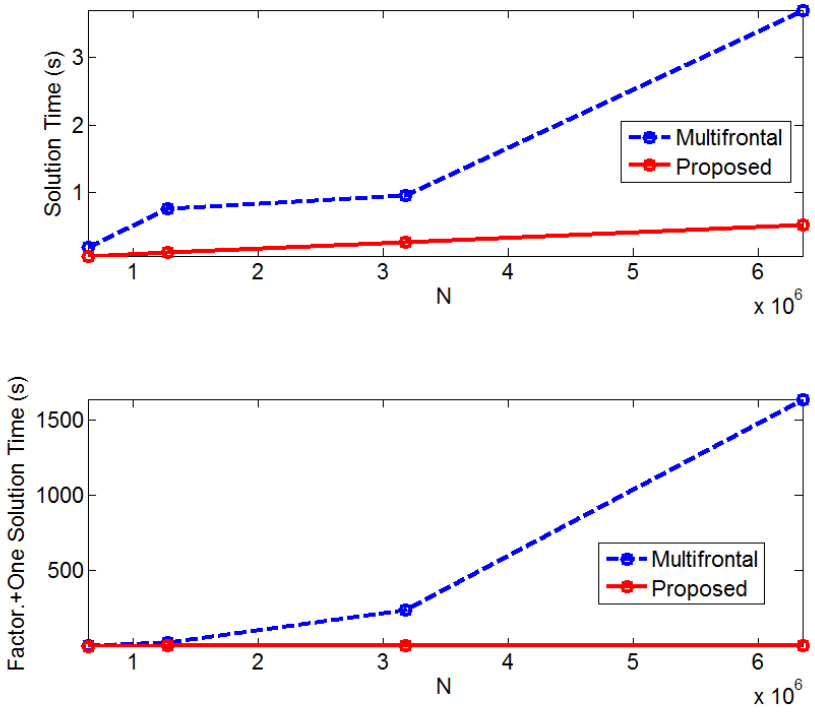


Fig. 2.3. Comparison of the matrix solution cost.

We then validate the accuracy of the entire scheme by simulating a test-chip interconnect structure. The length, width and height of the structure are 100 μm, 30 μm, and 3.192 μm respectively. The input source is a Gaussian derivative pulse with $\tau = 3 \times 10^{-12}$ s. Fig. 2.4 illustrates near/far end voltages of the proposed scheme in comparison with reference data obtained from the traditional TDFEM solution. Excellent agreement is observed. In the last example, we test the complexity of the proposed method from a small number up to 25 million unknowns for simulating a suite of interconnect structures. A clear linear complexity can be observed from Fig. 2.5.

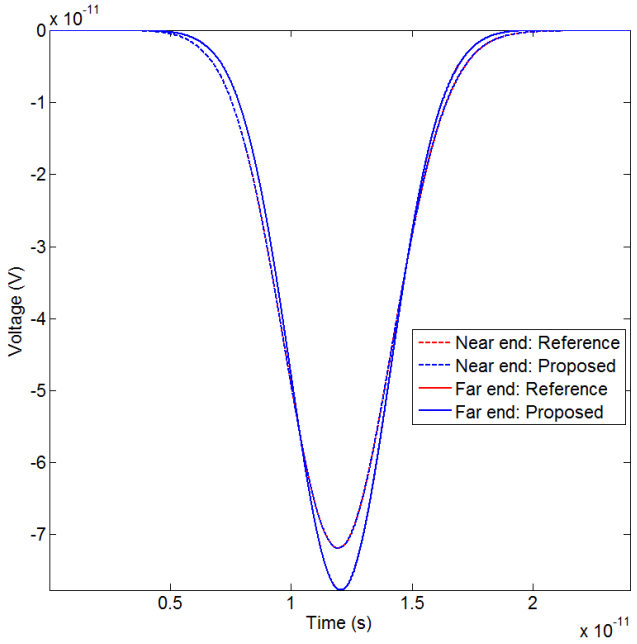


Fig. 2.4. Accuracy validation of the proposed algorithm.

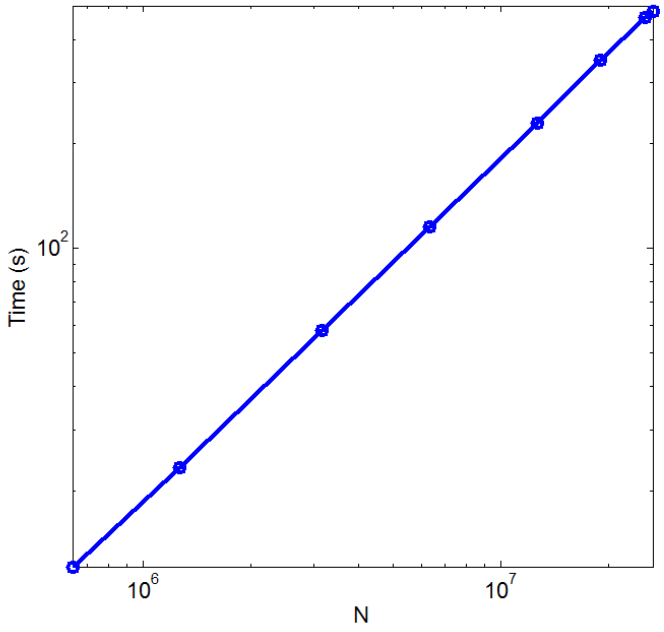


Fig. 2.5. CPU time vs.N for simulating a suite of on-chip circuits.

2.5 Conclusions

In this chapter, a fast structure-aware direct time-domain finite element solver is developed. The structure specialty of on-chip circuits such as Manhattan geometry and layered permittivity is preserved in the proposed numerical solution. As a result, the computational challenge of solving a very large-scale matrix encountered in the large-scale circuit analysis is removed since the matrix solution at each time step is converted to a simple scaling regardless of the matrix size.

3. FASTER STRUCTURE-AWARE DIRECT TDFEM SOLVER WITHOUT SACRIFICING TIME STEP SIZE

3.1 Introduction

In previous chapter, a structure-aware direct TDFEM solver was developed to simulate on-chip circuits, which has successfully addressed the computational challenge of simulating a very large scale matrix resulting from the time-domain analysis of a VLSI circuit. However, the efficiency of the solver is still limited by the small time step size required for the fast convergence of a matrix exponential term involved in the time marching. In this Chapter, we present an algorithm to overcome the small time step problem, while preserving the advantage of the algorithm in previous chapter in turning a 3-D large-scale system matrix solution to a simple scaling. In this algorithm, the time step is not reduced as compared to that of an explicit TDFEM scheme. Furthermore, the total number of time steps to be simulated is significantly reduced. As a result, a total cost reduction in CPU time is achieved. Comparisons with existing TDFEM solutions have demonstrated the obvious advantages of the proposed method in computational capacity and efficiency. The contents of this chapter have been extracted and revised from the following publication: Woochan Lee and Dan Jiao, "Fast Structure-Aware Direct Time-Domain Finite-Element Solver for the Analysis of Large-Scale On-Chip Circuits," IEEE Transactions on Components, Packaging and Manufacturing Technology, 2015.

3.2 Proposed Method

3.2.1 General Idea

A time-domain FEM solution of the second-order vector-wave equation for an integrated circuit problem results in the following linear system of equations

$$\mathbf{T}\ddot{u}(t) + \mathbf{R}\dot{u}(t) + \mathbf{S}u(t) = \dot{I}(t), \tag{3.1}$$

in which $\mathbf{T}$ is a mass matrix, $\mathbf{R}$ is associated with conductivity, $\mathbf{S}$ is a stiffness matrix, u is the field solution vector, and I is a vector of current sources. The single dot above a letter denotes a first-order time derivative, while the double dots denote a second-order time derivative. A central-difference based discretization of (3.1) in time results in the following explicit updating equation

$$\left(\mathbf{T} + \frac{\Delta t}{2}\mathbf{R}\right) u^{n+1} = 2\mathbf{T}u^n - \mathbf{T}u^{n-1} - \Delta t^2 \mathbf{S}u^n + \frac{\Delta t}{2}\mathbf{R}u^{n-1} - \Delta t^2 \dot{I}^n, \tag{3.2}$$

where the field solution at the most advanced time step, u^{n+1}, is obtained from the field solutions at the previous two time steps, u^n and u^{n-1}, step by step. Obviously, the updating of (3.2) in time requires a matrix solution of $\left(\mathbf{T} + \frac{\Delta t}{2}\mathbf{R}\right)$. This matrix does not have a layered property since $\mathbf{R}$ is related to conductivity, and conductivity is not layered. If one moves the $\mathbf{R}$-based term from the left hand side of (3.2) to the right hand side, i.e., let $\mathbf{R}$ be associated with the field value at the previous time step. The resultant time step for a stable simulation is too small to be used for an efficient simulation [2]. To fully exploit the layered property of the permittivity distribution, we propose a fast algorithm that turns a matrix solution to a simple scaling without sacrificing in the time step size as follows.

We begin by rewriting (3.2) as

$$\mathbf{K}u^{n+1} = \tilde{b}, \tag{3.3}$$

in which

$$\mathbf{K} = \mathbf{I} + \frac{\Delta t}{2}\mathbf{T}^{-1}\mathbf{R}, \tag{3.4}$$

$$\tilde{b} = 2u^n - u^{n-1} - (\Delta t)^2 \mathbf{T}^{-1} \left(\mathbf{S}u^n - \frac{1}{2\Delta t}\mathbf{R}u^{n-1} + \dot{I}^n \right). \tag{3.5}$$

To take advantage of the layered property of materials, we propose to obtain the inverse of $\mathbf{K}$ from the following series expansion

$$\mathbf{K^{-1}} = (\mathbf{I} + \mathbf{A})^{-1} = \mathbf{I} - \mathbf{A} + \mathbf{A^2} - \mathbf{A^3} + \mathbf{A^4} - \mathbf{A^5} + ..., \tag{3.6}$$

where

$$\mathbf{A} = \frac{\Delta t}{2}\mathbf{T^{-1}R}. \tag{3.7}$$

However, the series (3.6) would not converge unless the following condition is satisfied

$$\|\mathbf{A}\| < 1, \tag{3.8}$$

i.e., the norm of $\mathbf{A}$ is less than 1. Unfortunately, this condition is generally not satisfied in on-chip circuits, with a large time step Δt used in a central-difference based TDFEM scheme like (3.2). This is because the metal conductivity of on-chip circuits is high, the typical value of which is in the order of 10^7 S/m, and the $\|\mathbf{T^{-1}R}\|$ is proportional to σ/ε. With a high conductivity σ, one has to use a very small time step Δt to make $\|\mathbf{A}\| < 1$, rendering the time-domain simulation of (3.1) computationally expensive. To overcome the aforementioned difficulty, we propose to reduce the conductivity σ and increase the permittivity ε such that $\|\mathbf{T^{-1}R}\|$ is reduced to such a value that (3.8) is satisfied. Apparently, this change is not feasible because the material parameters are altered, and hence the original structure is completely changed. However, based on the fact that the on-chip circuit response is dominated by static modes in a fairly wide range of frequencies from zero to a few GHz [21], and the space distribution of static fields does not change when the permittivity and conductivity are scaled, we can modify permittivity and conductivity so that Δt can be enlarged to a desired value while (3.8) is still satisfied.

To explain, the solution of (3.1) is governed by the following quadratic eigenvalue problem,

$$(\lambda^2 \mathbf{T} + \lambda \mathbf{R} + \mathbf{S})V = 0, \tag{3.9}$$

in which λ is eigenvalue, and V is the eigenvector. The field solution of (3.1) at any time is nothing but a linear superposition of the eigenvectors of (3.9). The eigenvectors whose eigenvalues are zero are termed DC modes or static modes. They represent one kind of fundamental space variations of the fields in the given structure, which satisfies Maxwells equations as well as all the boundary conditions such as those at the material interface and on the truncation boundary. As quantitatively analyzed in [21], for relatively small electrical sizes (true for many on-chip structures), the solution of (3.1) is dominated by DC modes, whereas the contributions from full-wave modes are negligible. Since DC modes have eigenvalues $\lambda = 0$, they satisfy

$$\mathbf{S}V = 0. \tag{3.10}$$

In other words, the space distribution of the DC eigenmode V makes $\mathbf{S}V$ vanish, which also agrees with physics since the curl of static $\mathbf{E}$ fields is zero. Since $\mathbf{S}$ is only related to permeability, the field distributions of DC modes do not depend on the specific values of permittivity and conductivity. Hence, we can utilize this fact to change the material parameters of the original structure so that time step Δt can be significantly enlarged. Although a different problem is simulated, the DC mode of the new problem is the same as that in the original problem.

It also should be noted here that letting $\mathbf{R} = \mathbf{0}$, i.e., removing lossy part from the entire structure cannot produce correct DC modes of the original problem that has lossy conductors. This is because there are two kinds of DC modes [21] that satisfy (3.10). One represents the capacitance (C) effect. This mode is the nullspace eigenvector of the following generalized eigenvalue problem

$$\mathbf{S}_{oo}V = \lambda \mathbf{T}_{oo}V, \tag{3.11}$$

where $\mathbf{S}_{oo}$, $\mathbf{T}_{oo}$ respectively denotes the $\mathbf{S}$, and $\mathbf{T}$ formed by unknowns outside conductors, with the conductor surface serving as the perfect conductor boundary condition of the dielectric region. Clearly, by scaling permittivity, and hence $\mathbf{T}$, the eigenvector

of (3.11) stays the same. The other DC mode carries the resistance (R) effect. This mode is the nullspace eigenvector of the other generalized eigenvalue problem,

$$\mathbf{S}_{ii}V = \lambda \mathbf{R}_{ii}V, \tag{3.12}$$

where $\mathbf{S}_{ii}$, and $\mathbf{R}_{ii}$ respectively denotes the $\mathbf{S}$, and $\mathbf{R}$ formed by unknowns inside and on the surface of standalone conductors, without being superposed by the contribution from unknowns outside conductors. Again, by scaling conductivity, $\mathbf{R}$ is scaled by a single number, but the eigenvectors of (3.12) do not change. By setting $\mathbf{R} = 0$, the conductor is changed to a dielectric material, thus the DC modes of such a dielectric problem are not governed by (3.11) and (3.12) any more. Therefore, even though a lossless treatment has a lot of advantages in computation, to obtain a complete set of DC modes for a general lossy problem, the $\mathbf{R}$ part cannot be set as zero. However, for any conductor whose conduction current is larger than displacement current by two orders of magnitude or above, (3.11) and (3.12) would hold true for accurately representing DC modes. Therefore, we have a wide range of conductivity to choose from.

Based on the above finding, we choose σ and ε in such a way so that $\|\mathbf{A}\| < 1$ with a central-difference based time step, and hence the inverse of $\mathbf{K}$ can be obtained based on (3.6). Since (3.6) and (3.5) only require the computation of $\mathbf{T^{-1}}$, we turn the solution of $\mathbf{K}$ to the solution of $\mathbf{T}$. Since $\mathbf{T}$ is only related to permittivity, the layered property of permittivity can be fully exploited to further turn the solution of $\mathbf{T}$ to a simple scaling, which is detailed in previous chapter 2.

3.2.2 Fast DC-mode Extraction from a New Problem

We perform the time marching of (3.3) but with modified $\mathbf{T}$ and $\mathbf{R}$ matrices

$$\mathbf{T}_{new} = a_t\mathbf{T}, \mathbf{R}_{new} = a_r\mathbf{R} \tag{3.13}$$

, by scaling all the conductivity values down by a_r, and increasing all the permittivity values by a_t. By doing so, (3.3) can be performed fast with a large time step while

(3.6) can still be converged in a few terms. Since (3.3) is based on a central-difference time discretization, the time step needs to satisfy the stability condition for a central-difference based time marching. Thus, we enlarge time step through (3.13) but we do not exceed the time step allowed by the central-difference based time marching for a stable simulation. It is worth mentioning that the time step of the central-difference scheme in the proposed method is also larger than that in the original problem because permittivity is increased. To explain, from the stability analysis of a central-difference TDFEM scheme [26], it is known that the time step needs to satisfy $\Delta t < 2/\sqrt{\rho(\mathbf{T}^{-1}\mathbf{S})}$ for a stable time marching, where $\rho(\mathbf{T}^{-1}\mathbf{S}))$ denotes the spectral radius of $\mathbf{T}^{-1}\mathbf{S}$. Thus, with (3.13), the new time step can be enlarged by a factor of $\sqrt{a_t}$. In addition, by modifying both conductivity and permittivity, the number of terms used in (3.6) can be further reduced. The reasons are: 1) The Δt of $\mathbf{A}$ in (3.7) is proportional to $\sqrt{a_t}$ and 2) $\|\mathbf{T}^{-1}\mathbf{R}\|$ of $\mathbf{A}$ is proportional to $1/a_t$. Hence, overall the norm of $\mathbf{A}$ in (3.7) is reduced by $1/\sqrt{a_t}$, which results in a smaller number of terms for the convergence of (3.6), and thereby more speedup.

We only need to perform the simulation of (3.3) for a small time window to reveal the DC modes, like the preprocessing algorithm given in [19]. The detailed algorithm is as the following. We start the solution of (3.3). Every p step, we sample the solution of (3.3) and add it as one column vector in $\mathbf{X}$, which is initialized to be zero. The sample interval p is generally chosen as $\Delta t_{accuracy}/\Delta t_{stability}$, where $\Delta t_{accuracy}$ is the time step required by sampling accuracy for the input spectrum, and $\Delta t_{stability}$ is the time step determined by stability condition. When adding a new solution vector into $\mathbf{X}$, we orthogonalize it with the column vectors that have already been stored in $\mathbf{X}$.

With $\mathbf{X}$, whose column dimension is denoted by k, we transform the original large quadratic eigenvalue problem of size N in (3.9) to a small eigenvalue problem of size k

$$\mathbf{B}_r\mathbf{\Phi}_r = \lambda\mathbf{A}_r\mathbf{\Phi}_r \tag{3.14}$$

where

$$\mathbf{A}_r = \begin{bmatrix} \mathbf{R}_r & \mathbf{T}_r \\ \mathbf{T}_r & \mathbf{0} \end{bmatrix}, \mathbf{B}_r = \begin{bmatrix} -\mathbf{S}_r & 0 \\ 0 & \mathbf{T}_r \end{bmatrix} \tag{3.15}$$

in which

$$\mathbf{T}_r = \mathbf{X}^{\mathbf{H}}\mathbf{T}\mathbf{X}, \mathbf{R}_r = \mathbf{X}^{\mathbf{H}}\mathbf{R}\mathbf{X}, \mathbf{S}_r = \mathbf{X}^{\mathbf{H}}\mathbf{S}\mathbf{X}. \tag{3.16}$$

At early time, we observe eigenvalues of large magnitude from (3.14). The DC modes appear later, which can be identified by its small values as compared to other eigenvalues. Once DC modes are identified from (3.14), we can terminate the time marching of (3.3). Let the DC modes extracted from (3.14) be $\tilde{\mathbf{\Phi}}_{DC}$. The DC modes of the original problem (3.9) can be obtained as

$$\mathbf{V}_{DC} = \mathbf{X}_{N\times k}\tilde{\mathbf{\Phi}}_{DC,k\times k_{DC}}, \tag{3.17}$$

where the subscripts denote the matrix dimensions, and k_{DC} is the number of DC modes. In (3.14), if we increase permittivity too much, the electrical size of the structure will be greatly enlarged, which will enlarge the time window to be simulated to identify DC modes. This is because for an electrically larger problem, more modes are involved in the field solution. Especially, the first higher-order mode would appear at a lower frequency. The frequency at which to observe the DC mode thus becomes lower. As a result, in time domain, one has to wait for a longer time before the DC mode becomes not negligible in the field solution. Therefore, the cannot be chosen too large. In general, we choose it to be no greater than 10. Similarly, if we reduce the conductivity too much, the metal would be changed to a dielectric. The physical behavior of the DC modes, which is dominated by RC-effects, cannot be captured. In view of this, we reduce the conductivity in such a way that the resultant material is still a metal. In general, the conductivity is chosen to be no less than 1000 S/m.

3.2.3 Synthesis of Solution of the Original Problem

Since the original problem and the new modified problem share the same DC modes in common, the field solution of the original problem (3.1) can be accurately expanded into the DC modes extracted from the modified problem as the following

$$u(t) = \mathbf{V}y(t), \tag{3.18}$$

with $\mathbf{V} = \mathbf{V}_{DC}$ as shown in (3.17). We then substitute (3.18) into (3.1) and multiply the resultant by $\mathbf{V^H}$ on both sides, obtaining

$$\mathbf{T}_r\ddot{y}(t) + \mathbf{R}_r\dot{y}(t) + \mathbf{S}_r y(t) = \tilde{I}(t) \tag{3.19}$$

where $\mathbf{T}_r = \mathbf{V}^H\mathbf{T}\mathbf{V}$, $\mathbf{R}_r = \mathbf{V}^H\mathbf{R}\mathbf{V}$, $\mathbf{S}_r = \mathbf{V}^H\mathbf{S}\mathbf{V}$ and $\tilde{I}(t) = \mathbf{V}^H\dot{I}(t)$. The dimension of (3.19) is of $O(1)$, which is much smaller than the original size of (3.1). In addition, the time step used for simulating (3.19) can be solely determined by accuracy, thereby much larger than that of the conventional explicit TDFEM. This is because the modes that cannot be stably simulated by such a large time step are not included in $\mathbf{V}$, as analyzed in [4]. As a result of small size and large time step, (3.19) can be simulated with negligible time.

It is also worth mentioning that the input pulse used for the DC mode extraction can be different from the real pulse used in the final simulation. For example, a higher-frequency input pulse can be employed in the step of DC mode extraction so that the CPU time can be further reduced.

3.3 Numerical Results

3.3.1 Two On-Chip Interconnect Structures

We first simulate an on-chip interconnect structure to validate the proposed algorithms. The structure is illustrated in Fig. 3.1. The length, width, and height of the structure are 120 μm, 30 μm, and 3.192 μm respectively. The top and bottom planes are truncated by a PEC (perfect electric conductor) boundary condition, while

the front and back planes are terminated by ABC (absorbing boundary condition), and the other two boundaries are left open. The permittivity and conductivity distribution of the structure is shown in Fig. 3(b). The input current sources have a Gaussian derivative pulse of $I(t) = 2(t - t_0)\exp(-(t - t_0)^2/\tau^2)$, with $\tau = 3 \times 10^{-8}$ s, and $t_0 = 4\tau$. They are launched from the bottom metal layer to the inner conductor, and from the upper metal layer to the inner conductor as shown in Fig. 3.1(a) by the red arrows.

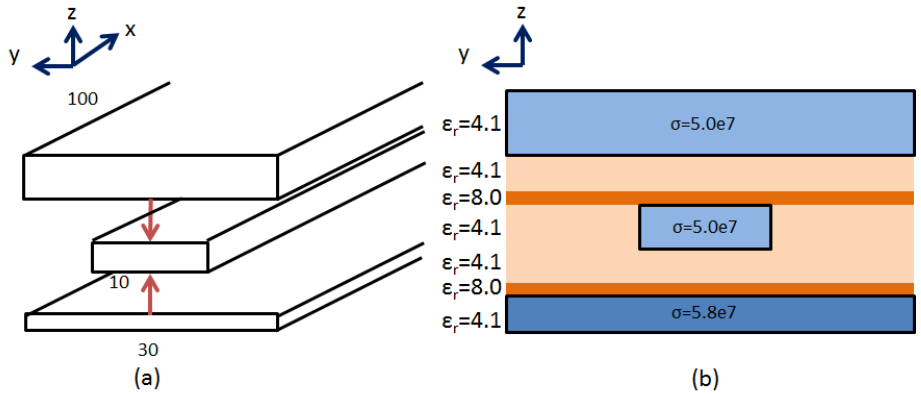


Fig. 3.1. Illustration of an on-chip interconnect. (a) 3-D view of the structure. (b) y-z plane view of the structure.

The modified problem for fast DC mode extraction has a conductivity reduced by 1×10^4, and permittivity increased by 10, which results in a time step of 3.3×10^{15} s used in the time-marching for DC mode extraction. Due to the increase of permittivity by 10, the resultant time step is $\sqrt{10}$ larger than that required by the original problem for stability. In the stage of DC mode extraction, 10 solutions are sampled with a sampling interval of p = 303,030 before the DC mode is accurately identified. The number of terms used in the series expansion (3.6) is 9. After the DC mode extraction from the modified problem, the transient solution of the original problem is synthesized. The voltage obtained at the far end of the structure is plotted in Fig. 3.2. Excellent agreement is observed between the proposed method and the conventional TDFEM scheme. With the same computer (Intel XEON E5410 2.33

GHz processor), the speedup of the proposed method over the conventional TDFEM is shown to be 4.824, which includes the CPU time at every step from the time-domain simulation of a modified problem for fast DC mode extraction to the synthesis of the solution of the original problem. In contrast, with the method in [13], although the challenge of matrix solution is also overcome, the total CPU time is still longer than that of a conventional TDFEM due to a reduced time step.

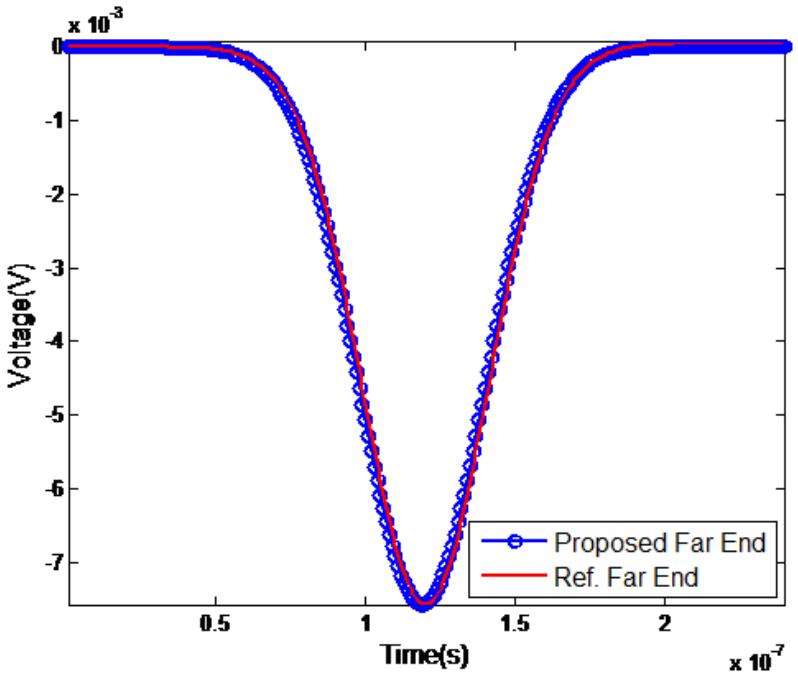


Fig. 3.2. Simulation of an on-chip interconnect.

The structure simulated in the above is dominated by capacitance effects at relatively low frequencies. To examine the accuracy of the proposed method in handling R-dominant circuits, we simulate the same structure but let the far end shorted to the bottom plane by a metal contact as shown in Fig. 3.3. Different from the setting in the previous structure, the input current source is launched from the bottom metal layer to the inner conductor only, and it has a Gaussian derivative pulse with $\tau = 3 \times 10^{-9}$ s. The modified problem has a conductivity reduced by 1×10^{-4}. The time step used in the time-marching for DC mode extraction is 1×10^{-15} s, which is

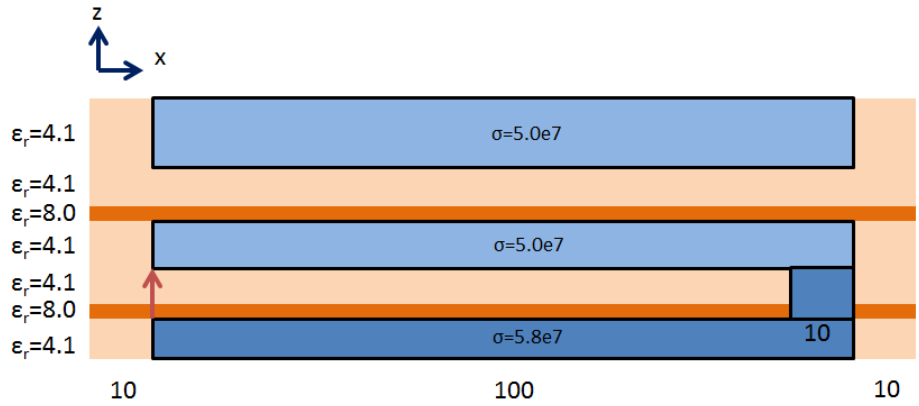


Fig. 3.3. Side view of a shorted on-chip interconnect.

the same as that permitted by a traditional TDFEM based simulation of the original problem. The sampling interval p is 100,000 and 7 terms are used in (3.6). The CPU time of the conventional TDFEM is 3.729×10^4 s, while the proposed method is 1.835 times faster. Same as that in the previous example, we can also modify permittivity to enlarge time step as well as reduce the number of terms required for the convergence of (3.6). The modified problem has a conductivity reduced by 1×10^{-4} and permittivity increased by 10. Again, due to the increase of permittivity by 10, the resultant time step is enlarged to 3.3×10^{-15} s, and hence p = 30,303. In total, 10 solutions are sampled before the DC mode is accurately identified. The number of terms is 5 for the convergence of series expansion, which is smaller than that in the previous setting where permittivity is not changed. The speedup of the proposed method over the conventional TDFEM is hence increased to 8.095. In Fig. 3.4, we plot the near end voltage of the proposed method in comparison with the reference data obtained from the traditional TDFEM solution. Excellent agreement is observed.

3.3.2 On-Chip Power Grid

An on-chip power grid structure as shown in Fig. 3.5 is simulated. The red lines denote power rails; while blue ones are ground rails. There is a vertical via connecting

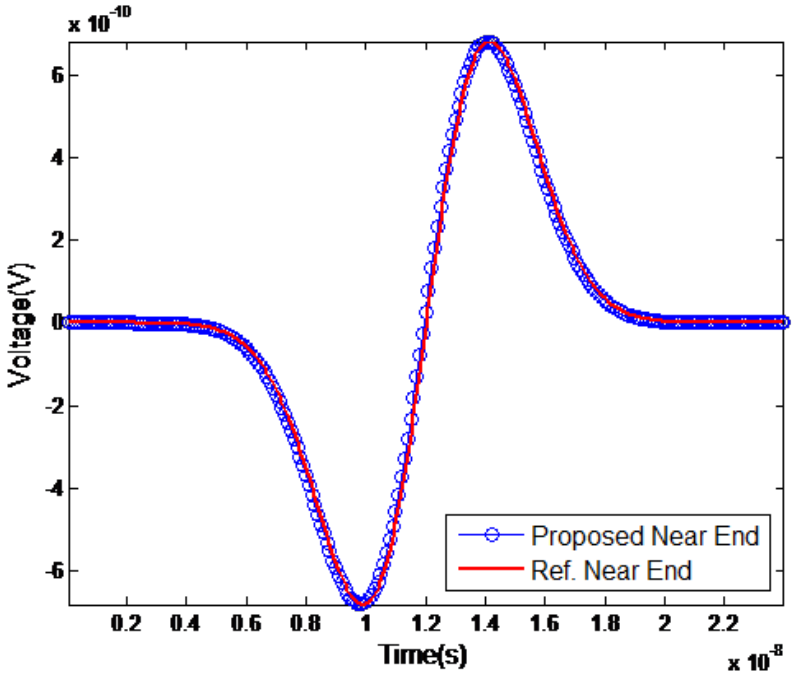


Fig. 3.4. Accuracy validation of the proposed algorithm in simulating a far-end shorted on-chip interconnect.

two metal wires wherever the two wires of same polarity cross each other in the top view. The size of the structure is 7.0 μm 7.0 μm 7.6 μm. The top and bottom planes are set to be PEC and the other 4 sides are left open. The number of unknowns in this example is 1,101. The permittivity is layered as shown in Fig. 3.5(c), and the conductivity of the metal is 5.0×10^7 S/m.

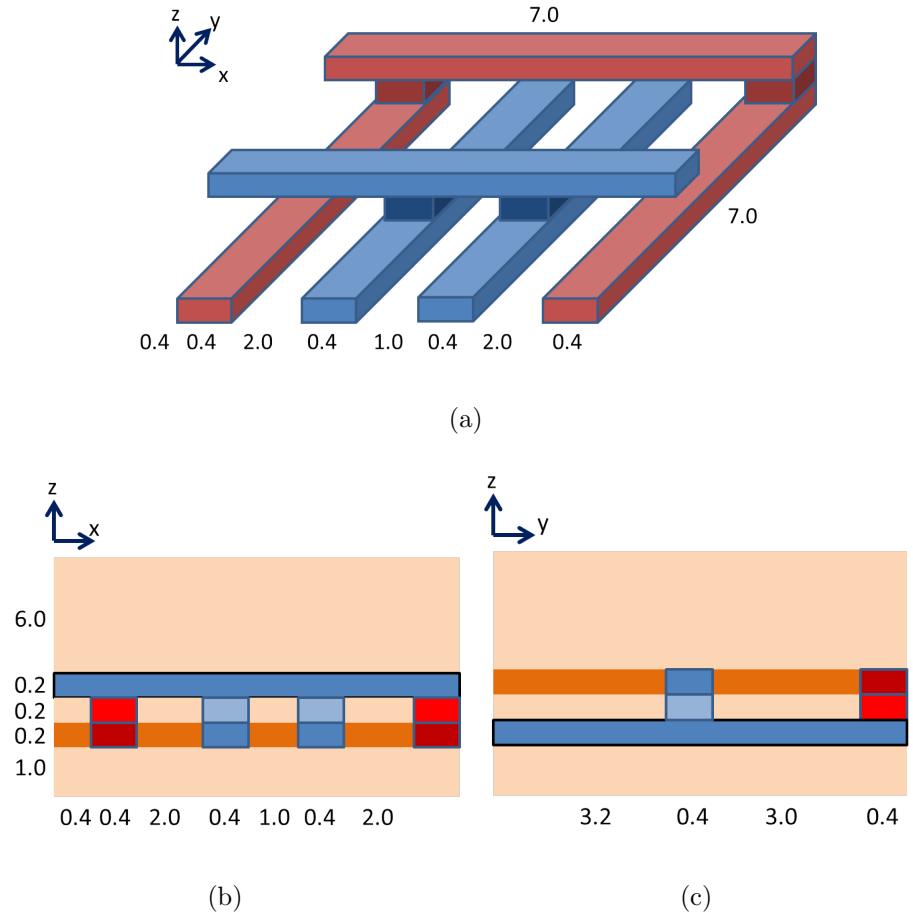


Fig. 3.5. Illustration of the structure of an on-chip power grid. (a) 3-D view. (b) x-z plane view. (c) y-z plane view.

Since the proposed method allows for the use of a different pulse in the DC-mode extraction stage as compared to the real one required in the simulation stage, we employ a higher frequency input than the original input to achieve an even better speedup. The original Gaussian derivative pulse is $I(t) = 2(t-t_0)\exp(-(t-t_0)^2/\tau^2)$, where $tau = 3 \times 10^{-9}$ s and $t_0 = 4\tau$. The pulse we use in the stage of DC-mode extraction has $\tau = 3 \times 10^{-11}$ s, and hence its maximum frequency is 100 times larger than before. The time step used in the time-marching for DC mode extraction is 5×10^{-16} s. The solutions are sampled every 2,000 steps, i.e. p = 2,000. In total, 15 solutions are sampled before the DC mode is accurately identified. The modified

problem has a conductivity reduced by 1×10^{-4}. The permittivity is kept the same as before. The number of terms is 5 for the convergence of series expansion shown in (3.6). The current source is launched between one power rail and one ground rail in the lower metal layer, and the voltage between the two rails is sampled and plotted in Fig. 3.6. Again, an excellent agreement with the reference TDFEM solution is observed. With the same computer, the CPU time of the conventional central-difference TDFEM is 8.950×4 s, whereas the CPU time of the proposed method including all steps is 3.555×10^2 s, thus a speedup of 251.7.

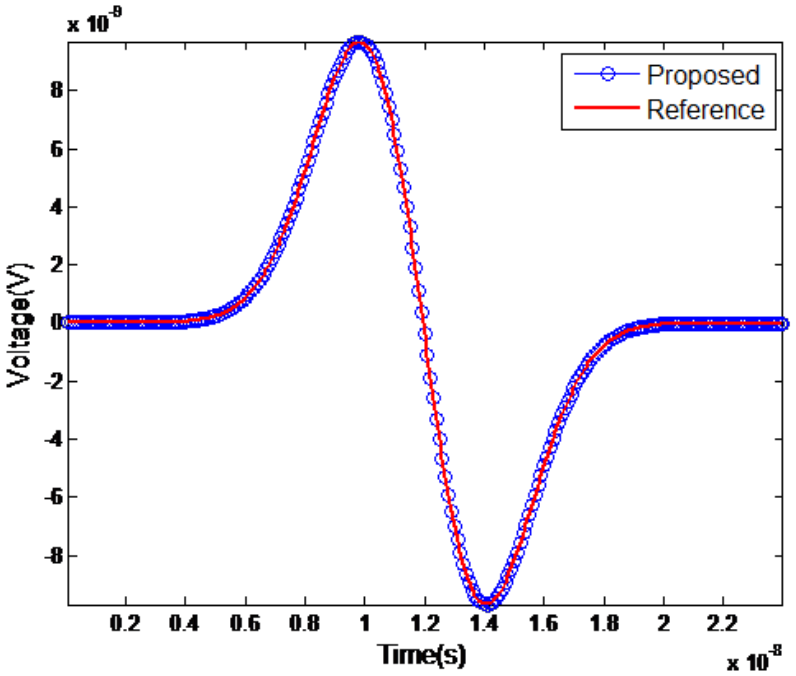


Fig. 3.6. Accuracy validation of the proposed algorithm in power grid simulation.

3.3.3 Rectangular Spiral Inductor

We then simulate an on-chip spiral inductor to validate the proposed algorithms. The structure along with its permittivity configuration is illustrated in Fig. 3.7. The entire computational domain occupies a region of 1200 μm by 1000 μm by 500 μm.

The top and bottom planes are truncated by a PEC boundary condition, while all the other boundaries are left open. The number of unknowns in this example is 14,286.

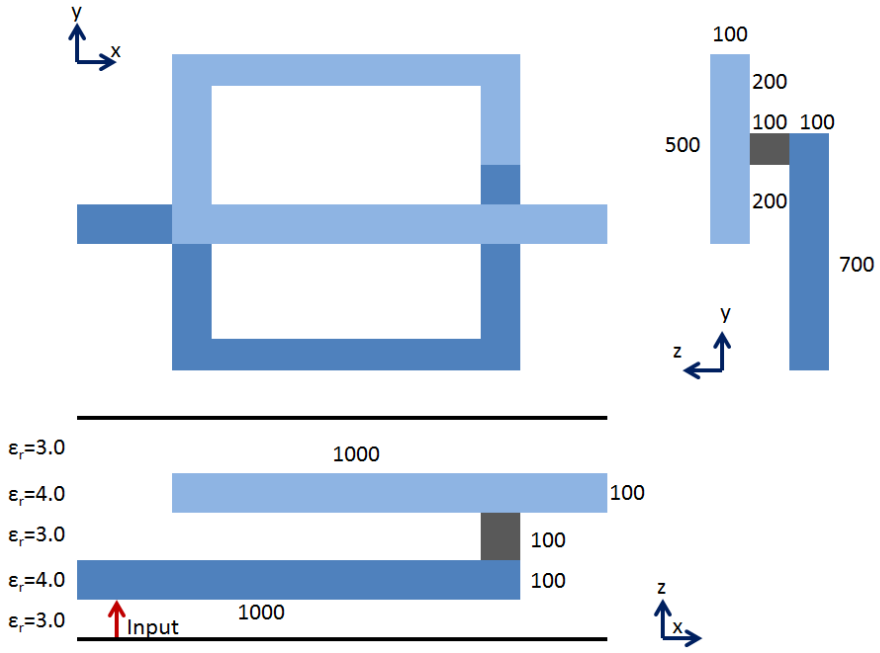


Fig. 3.7. Illustration of a rectangular spiral inductor structure.

The permittivity is layered and the conductivity of the conducting wire is 5.0×10^7 S/m. The input current source has a Gaussian derivative pulse of $I(t) = 2(t - t_0)\exp(-(t-t_0)^2/\tau^2)$, with $\tau = 3 \times 10^{-8}$ s and $t_0 = 4\tau$. It is launched from the bottom PEC plane to the left port of the inductor as shown in Fig. 3.7 by the red arrow. The time step used in the time-marching for DC mode extraction is 5×10^{-14} s. The solutions are sampled every 20,000 steps, i.e. $p = 20{,}000$, since the time step required by accuracy is 1×10^{-9} s. In total, 6 solutions are sampled before the DC mode is accurately identified. The modified problem has a conductivity reduced by 1×10^{-5}. The number of terms is 7 for the convergence of series expansion shown in (3.6). The CPU time of the conventional TDFEM is 1.734×10^5 s, whereas the proposed method is 8.874 times faster. Furthermore, we increase the permittivity by 10. As a result, the time step is enlarged by a factor of $\sqrt{10}$, yielding a time step of

1.5×10^{-13} s used in the time-marching for DC mode extraction. The solutions are sampled every 6,666 steps. In total 6 solutions are sampled before the DC mode is accurately identified. The speedup of the proposed method is 20.736. The voltage simulated at the right port of the inductor is plotted in Fig. 3.8. Excellent agreement is observed between the proposed method and the conventional central-difference based TDFEM scheme.

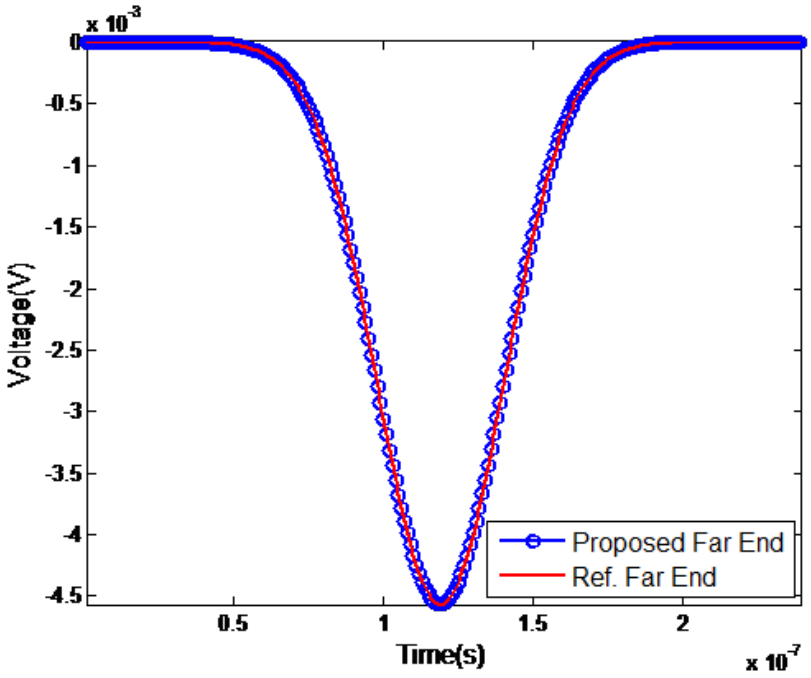


Fig. 3.8. Accuracy validation of the proposed algorithm for the simulation of a rectangular spiral inductor with $\Delta t = 1.5 \times 10^{-13}$.

3.3.4 Suite of Large-Scale On-Chip Power Grids

In the last example, we simulate a suite of large-scale on-chip power grids as shown in Fig. 3.9. The first one has a unit block size of 7.2μm $\times$ 7.2μm $\times$ 7.6μm, which is then extended 10 times along both x and y directions. The top and bottom planes are set to be PEC and the other 4 sides are left open. The number of unknowns in

this example is 118,715. The permittivity is layered as shown in Fig. 3.9(c), and the conductivity of the metal is 5.0×10^7 S/m.

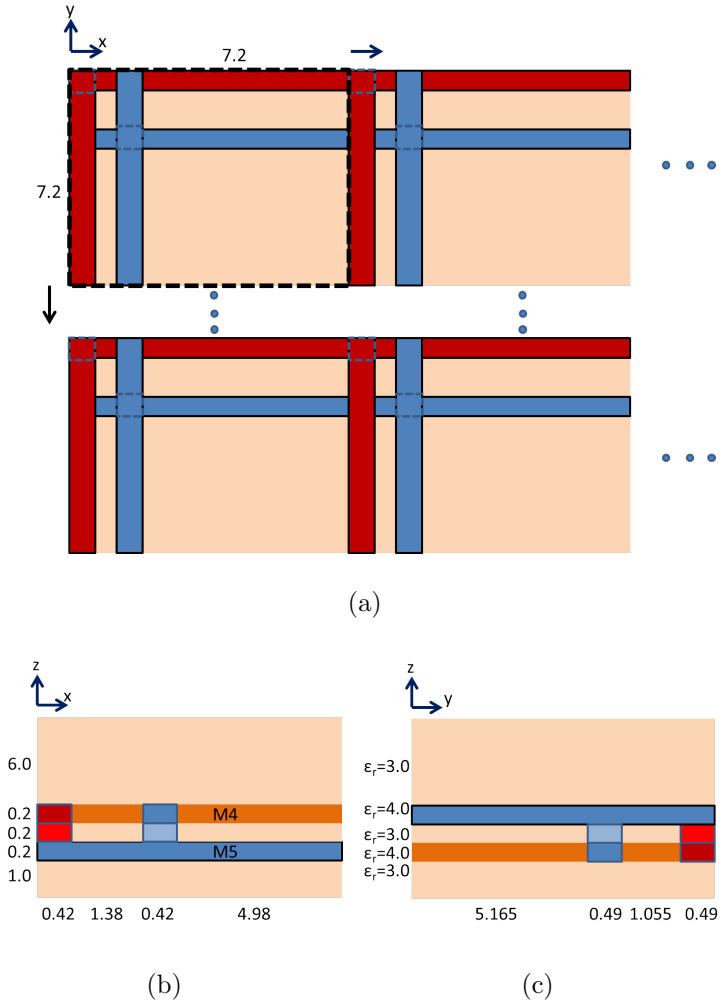


Fig. 3.9. Illustration of a larger on-chip power grid structure. (a) x-y plane view. (b) x-z plane view. (c) y-z plane view.

In the DC-mode extraction step, we employ a higher frequency input than the original input. The original Gaussian derivative pulse is $I(t) = 2(t-t_0)\exp(-(t-t_0)^2/\tau^2)$, where $tau = 3 \times 10^9$ s and $t_0 = 4\tau$. The pulse we use in the step of DC-mode extraction has $\tau = 3 \times 10^{-11}$ s, and hence a maximum frequency 100 times larger than before. The time step used in the time-marching for DC mode extraction is 5×10^{-16} s. The solutions are sampled every 2,000 steps. In total, 11 solutions are sampled

before the DC mode is accurately identified. The modified problem has a conductivity reduced by 1×10^{-4}. The permittivity is kept the same as before. The number of terms is 5 for the convergence of series expansion shown in (3.6).The current source is launched between one power rail and one ground rail in the upper metal layer, and the voltage between the two rails is sampled and plotted in Fig. 3.10 in comparison with the reference TDFEM solution. With the same computer, the CPU time of the conventional central-difference TDFEM is 4.896×10^6 s, whereas the CPU time of the proposed method including all steps is 3.248×10^4 s, thus a speedup of 150.7.

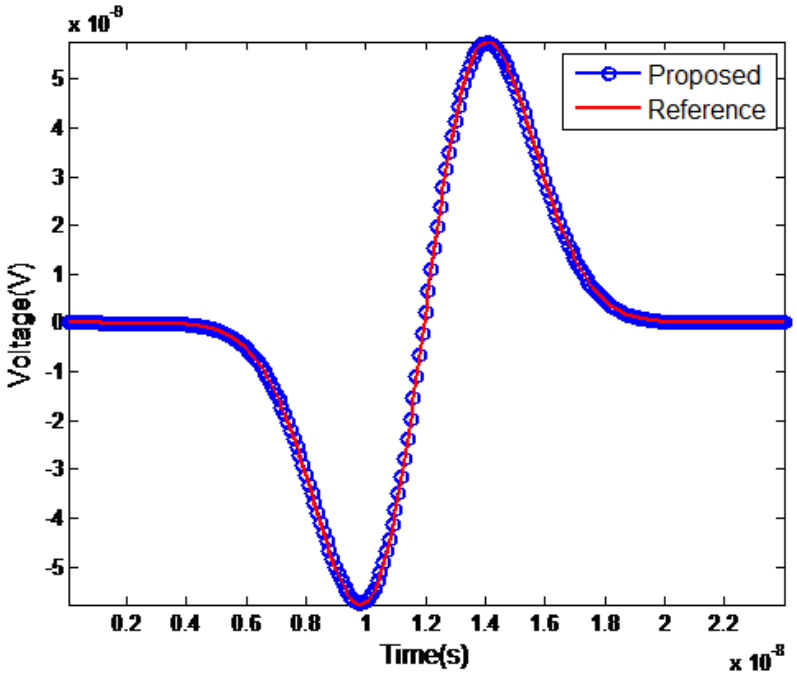
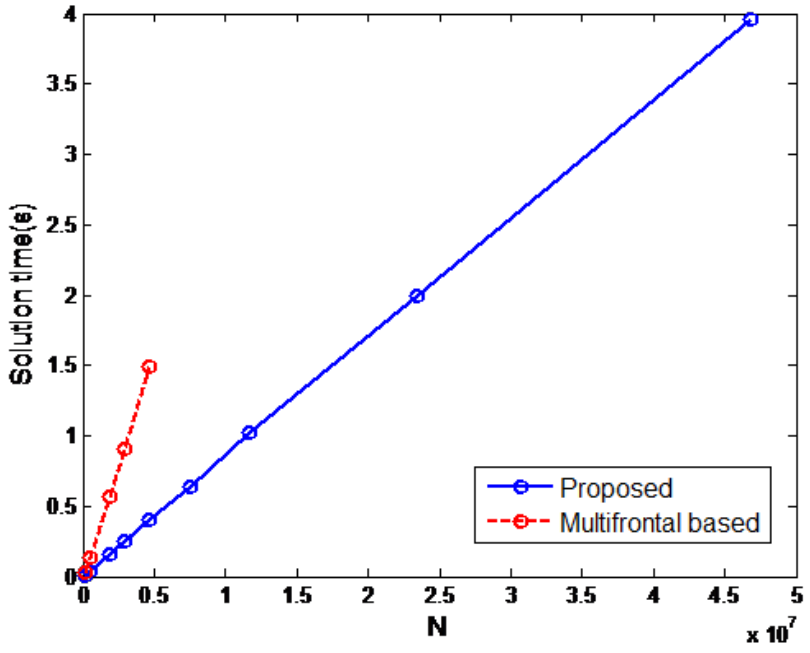


Fig. 3.10. Accuracy validation of the proposed algorithm in simulating a larger on-chip power grid.

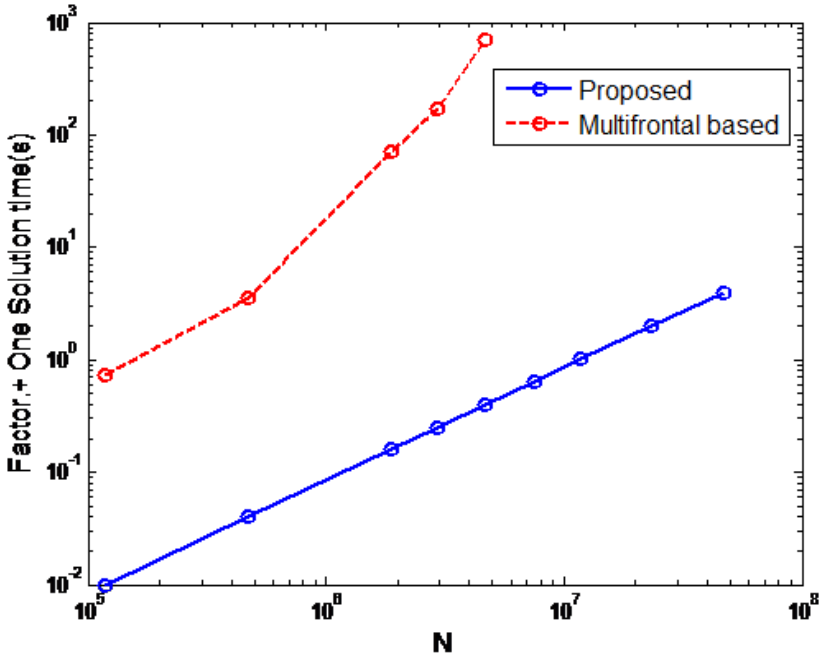
We then increase the structure simulated in the above progressively along both x and y directions. The chip area is increased from 72×72, 144×44, 288×288, 360×360, 576×360, 565×576, 720×720, 1440×720, to 1440×1440 μm^2, resulting in 118,715, 471,425, 1,878,845, 2,933,555, 4,691,255, 7,501,685, 11,717,105, 23,426,105 and 46,834,205 unknowns, respectively. Using this suite of power grid structures, we

compare the performance of the matrix solution in the proposed method with that of the conventional TDFEM which employs a multifrontal based direct solver [27]. For large cases, conventional UV factorization of tridiagonal matrices is not satisfactory because U, V coefficients grow exponentially with the number of unknowns. Thus, the UV factorization is only used for cases with a small number of unknowns, and the ratio-based DS factorization [28] is employed to solve the tridiagonal matrix with linear complexity in negligible time for large unknown cases.

The solution time for one right hand side is shown in Fig. 3.11(a), while the sum of the factorization and one solution time is shown in Fig. 3.11(b). As can be seen from Fig. 3.11, the solution time of the proposed solver is much less than that of the conventional solver. Moreover, the solution time has a clear linear scaling with the number of unknowns N. In contrast, the conventional solver has a complexity much higher than linear, and the number of unknowns the conventional solver can handle on the same computer is much fewer than that solved by the proposed method. For example, the proposed solver has no difficulty in simulating the last case in the suite of power grid structures, which has over 46 million unknowns, on a single core, whereas the conventional solver cannot go beyond a few million unknowns on the same CPU core.



(a)



(b)

Fig. 3.11. CPU time vs. N for simulating a suite of on-chip power grids. (a) One solution time; (b) Factorization and one solution time.

3.4 Conclusions

In this chapter, a fast structure-aware direct time-domain finite element solver is developed for the analysis and design of very large-scale on-chip circuits. The structure specialty of on-chip circuits such as Manhattan geometry and layered permittivity is preserved in the proposed numerical solution, and the resulting disadvantage in time step is overcome. As a result, the computational challenge of solving a very large-scale matrix encountered in the large-scale circuit analysis is removed since the matrix solution at each time step is converted to a simple scaling regardless of the matrix size, and the total number of time steps to be simulated is also significantly reduced. The proposed method can be used for not only fast transient analysis, but also IR-drop analysis, and frequency-domain analysis of the on-chip circuits in a fairly wide range of frequencies.

4. A NEW EXPLICIT AND UNCONDITIONALLY STABLE TIME-DOMAIN FINITE-ELEMENT METHOD

4.1 Introduction

The time step of a traditional explicit time-domain method is restricted by the smallest space step in order to maintain the stability of a time-domain simulation. When structures being simulated involve fine features relative to working wavelengths such as on-chip VLSI circuits, multi-scaled engineering systems that encompass a few orders of magnitude difference in geometrical scales, etc., the explicit time-domain simulation can become highly inefficient. To overcome the dependence of the time step on space step, recently, an explicit and unconditionally stable TDFEM is developed [19], which has successfully removed the constraint of the space step on the time step. This method involves a pre-processing step which identifies the stable eigenmodes for the given time step. The time step used in this step is still the same as that of a conventional TDFEM. Although the time interval simulated in the pre-processing step is much shorter than the total time interval to be simulated, the performance of the pre-processing step is still limited by the conventional time step.

In this chapter, we propose a new method for achieving unconditional stability in an explicit time-domain finite-element method. In this new method, we directly exclude the unstable modes from the numerical system. We then perform an explicit time marching on the updated numerical system that is free of unstable modes. As a result, we bypass the computational overhead of the pre-processing step, and achieve unconditional stability regardless of space step. The contents of this chapter have been extracted and revised from the following publication: Woochan Lee and Dan Jiao, "A new explicit and unconditionally stable time-domain finite-element method,"

2015 IEEE Antennas and Propagation Society International Symposium (APSURSI), 2015.

4.2 Proposed Method

A time-domain FEM solution of the second-order vector-wave equation results in the following linear system of equations

$$\mathbf{T}\bar{u}(t) + \mathbf{S}u(t) = \dot{I}(t) \tag{4.1}$$

in which $\mathbf{T}$ is the mass matrix, $\mathbf{S}$ is the stiffness matrix, u is the field solution vector, and I is the current source vector. The first- and second-order time derivative are, respectively, represented by a single, and double dots above a letter. The solution of (4.1) is governed by the following generalized eigenvalue problem

$$\mathbf{SV} = \mathbf{TV\Lambda}, \tag{4.2}$$

in which $\mathbf{\Lambda}$ denotes a diagonal matrix whose entries are eigenvalues λ, and $\mathbf{V}$ is the eigenvector matrix. Since $\mathbf{T}$ is symmetric positive definite and $\mathbf{S}$ is symmetric, the eigenvectors of (4.2) are $\mathbf{T}-$ and $\mathbf{S}-$orthogonal as the following

$$\mathbf{V}^T\mathbf{TV} = \mathbf{I}, \quad \mathbf{V}^T\mathbf{SV} = \mathbf{\Lambda} \tag{4.3}$$

where $\mathbf{I}$ denotes an identity matrix. As shown in [19], the root cause of the instability of (4.1) for any given time step is the eigenmodes of (4.2) that have the following eigenvalues:

$$\lambda > 4/\Delta t^2. \tag{4.4}$$

These eigenmodes are termed unstable modes ($\mathbf{V}_h$) for the given time step δt. They clearly have the largest eigenvalues of (4.2). More importantly, it is shown in [19] that when the time step is chosen based on accuracy, the unstable modes are also those modes that are not needed for accuracy.

Based on the aforementioned understanding, before we perform an explicit time-domain simulation, if we upfront exclude the unstable eigenmodes from the underlying

numerical system, we can ensure the stability of the simulation for the given time step, and meanwhile maintain the solution accuracy. Along this line of thought, we propose to update the original system of equations (4.1) to a new system of equations as the following

$$\mathbf{T}\ddot{u}(t) + \mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right) = \dot{I}(t), \tag{4.5}$$

where $\mathbf{S}$ is changed to $\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right)$. The above is the same as changing the original numerical system consisting of both unstable and stable modes to a system of stable modes only. To see this point clearly, first, we realize that the solution of (4.5) is now governed by the eigen-solution of a new matrix $\mathbf{T}^{-1}\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right)$ instead of the original $\mathbf{T}^{-1}\mathbf{S}$ shown in (4.2). Since $\mathbf{T}$ satisfies (4.3), its inverse can be written as

$$\mathbf{T}^{-1} = \mathbf{V}\mathbf{V}^T = \mathbf{V}_s\mathbf{V}_s^T + \mathbf{V}_h\mathbf{V}_h^T, \tag{4.6}$$

where $\mathbf{V} = [\mathbf{V}_s\ \mathbf{V}_h]$, and $\mathbf{V}_s$ has eigenmodes that do not satisfy (4.4), and hence called stable eigenmodes. Using (4.6), we have

$$\mathbf{T}^{-1}\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right) = \mathbf{T}^{-1}\mathbf{S}\left(\mathbf{T}^{-1} - \mathbf{V}_h\mathbf{V}_h^T\right)\mathbf{T} = \mathbf{T}^{-1}\mathbf{S}\left(\mathbf{V}_s\mathbf{V}_s^T\right)\mathbf{T}. \tag{4.7}$$

From (4.2), we have $\mathbf{S}\mathbf{V}_s = \mathbf{T}\mathbf{V}_s\mathbf{\Lambda}_s$. Substituting it into the above, we obtain

$$\mathbf{T}^{-1}\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right) = \left(\mathbf{V}_s\mathbf{\Lambda}_s\mathbf{V}_s^T\right)\mathbf{T}. \tag{4.8}$$

Denote $\left(\mathbf{V}_s\mathbf{\Lambda}_s\mathbf{V}_s^T\right)\mathbf{T}$ by $\mathbf{A}$. It is clear that $\mathbf{A}\mathbf{V}_s = \mathbf{V}_s\mathbf{\Lambda}_s$. Hence, $(\mathbf{\Lambda}_s\ \mathbf{V}_s)$ is the eigenvalue solution of $\mathbf{T}^{-1}\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right)$. Since the rank of (4.8) is the number of stable eigenmodes k_s, while the matrix size of (4.8) is N, the $\mathbf{T}^{-1}\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right)$ is a low-rank matrix with $(N - k_s)$ zero eigenvalues in addition to the k eigenvalues of the stable eigenmodes. As a result, the explicit marching of the updated system (4.5) is absolutely stable for the given time step, since all the eigenvalues do not satisfy (4.4).

With a central-difference based time-marching scheme, (4.5) can be discretized as

$$u^{n+1} = \left\{2 - \Delta t^2\mathbf{T}^{-1}\mathbf{S}\left[\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right]\right\}u^n - u^{n-1} + \Delta t^2\mathbf{T}^{-1}\dot{I}^n, \tag{4.9}$$

where the $\mathbf{S}\left(\mathbf{I} - \mathbf{V}_h\mathbf{V}_h^T\mathbf{T}\right)u^n$ term is efficiently evaluated by a sequence of matrix-vector multiplications from right to left.

Although (4.9) is absolutely stable for the given time step regardless of its size, to ensure accuracy, we still need to add one important step as the following

$$u^{n+1} = u^{n+1} - \mathbf{V}_h \mathbf{V}_h^T \mathbf{T} u^{n+1}. \tag{4.10}$$

This is because the solution of (4.5) and hence (4.9) is the superposition of the stable eigenmodes $\mathbf{V}_s$, and the nullspace eigenvectors of (4.8), whose eigenvalues are zero. The nullspace of (4.8) is different from the nullspace of the original system (4.2). Although these modes can be simulated stably in (4.9), they make the solution of (4.9) wrong. This is because the solution of the original problem (4.1) only resides in the space of $\mathbf{V}_s$, but with (4.9), $u = \mathbf{V}_s y_s + \mathbf{V}_{a0} y_{a0}$, where $\mathbf{V}_{a0}$ denotes the additional nullspace of (4.8). The treatment of (4.10) will hence remove these additional nullspace modes since $\mathbf{V}_{a0}$ must be in the $\mathbf{V}_h$ space if it is not in $\mathbf{V}_s$.

As for the determination of $\mathbf{V}_h$, since these modes have the largest eigenvalues of (4.2), they can be found efficiently using the k-step implicitly restarted Arnoldi algorithm, the cost of which is just $k^2 O(N)$ for finding k largest eigenpair.

4.3 Numerical Results

We first demonstrate the unconditional stability of the proposed method with a parallel-plate example that has an analytical solution. The length, width, and height of the structure are 900 μm, 6 μm, and 1 μm respectively. The input source is a Gaussian derivative pulse with $\tau = 0.2$ s. Despite the low-frequency spectrum, due to the small space step, conventional TDFEM has to use a time step as small as $\Delta t = 2.5 \times 10^{-16}$ for a stable simulation. In contrast, the proposed method permits the use of any large time step. The time step, hence, can be solely chosen based on accuracy, thus being as large as 0.01 s in this example. As shown in Fig. 4.1, the voltages simulated by the proposed method show an excellent agreement with analytical solutions. The number of removed unstable modes is 644 in this example. It is worth mentioning that since the solution at this low frequency is only dominated by

the nullspace of (4.2), we analytically vanish the S-matrix related term since $\mathbf{SV} = 0$ for nullspace modes, while numerically it is not due to finite machine precision [29].

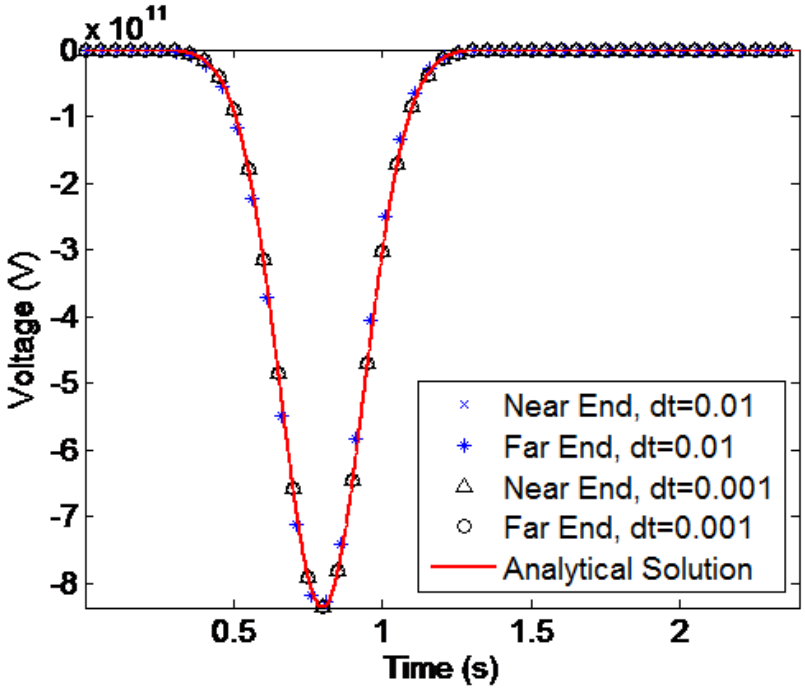


Fig. 4.1. Voltages of a parallel plate with different time steps compared with analytical solution.

The second example is a mm-level parallel plate waveguide filled by inhomogeneous materials of relative permittivity of 8.1 and 4. The length, width and height of the structure are 120 mm, 30 mm, and 3.192 mm respectively and the number of unknowns is 668. The z directional discretization is 1 mm, 1.316 mm, 1.317 mm, 2 mm and 3.192 mm. From 1.316 mm to 1.317 mm, the relative permittivity is 8, and 4.1 elsewhere. The input source is a Gaussian derivative pulse with $\tau = 8 \times 10^{-10}$ s and $t_0 = 3\tau$. Conventional TDFEM requires $\Delta t = 1 \times 10^{-13}$ s for a stable simulation. In contrast, the proposed method uses a time step $\Delta t = 1 \times 10^{-11}$ s solely determined by accuracy. The number of removed unstable modes is 291. Excellent agreement is observed between the proposed method and the reference result of a traditional

TDFEM as can be seen in Fig. 4.2. The total CPU time of current scheme is 10.95 s whereas conventional central difference based TDFEM CPU time is 33.08 s.

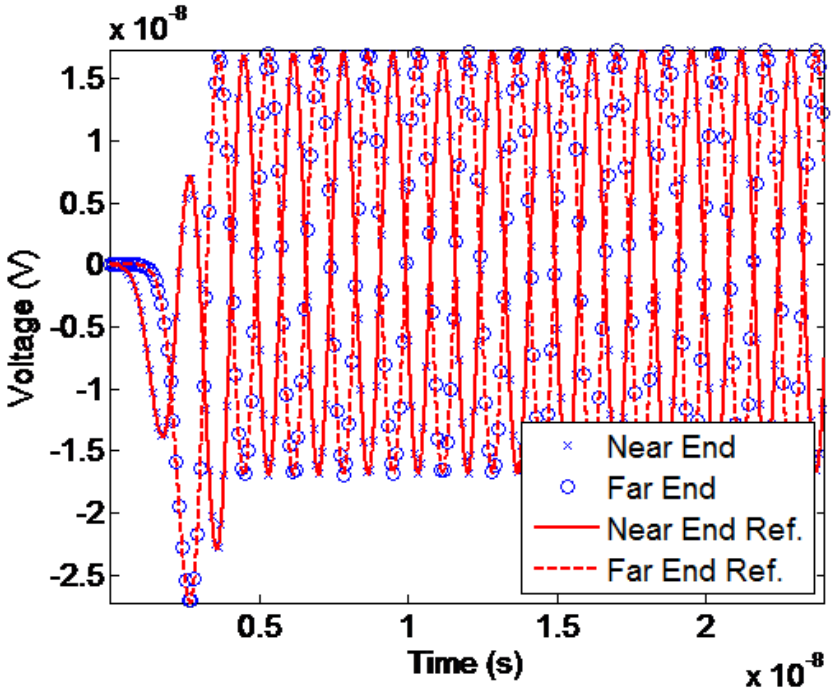


Fig. 4.2. Voltages of a mm-level parallel plate.

The last example is a lossless on-chip bus structure shown in Fig. 4.3. The width of each bus is 3 μm and so is the gap between buses. The two current sources with opposite direction are injected with Gaussian derivative pulses having $\tau = 3 \times 10^{-11}$ s and $t_0 = 4\tau$. Conventional TDFEM requires a small time step of $\Delta t = 1 \times 10^{-15}$ s. In contrast, the proposed method is able to use $\Delta t = 1 \times 10^{-12}$ s determined by accuracy. The number of removed unstable modes is 840. The speedup of the proposed method as compared to [19] is approximately 3. In Fig. 4.4 and Fig. 4.5, excellent accuracy of the proposed method is observed.

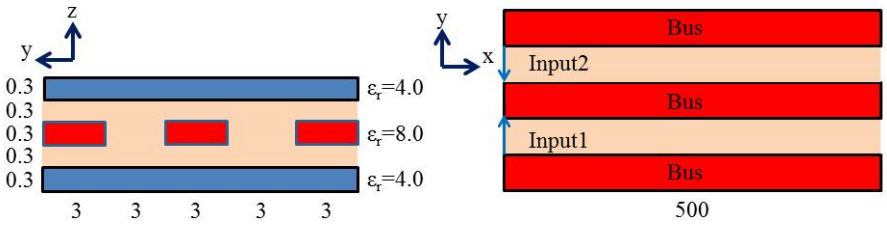


Fig. 4.3. The bus structure configuration (unit: μm).

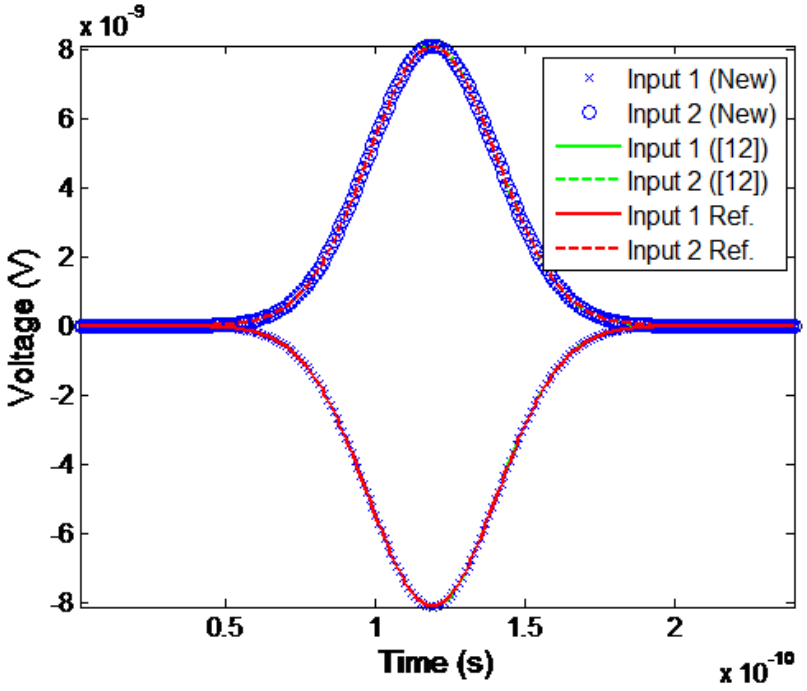


Fig. 4.4. Accuracy validation of on-chip bus structure.

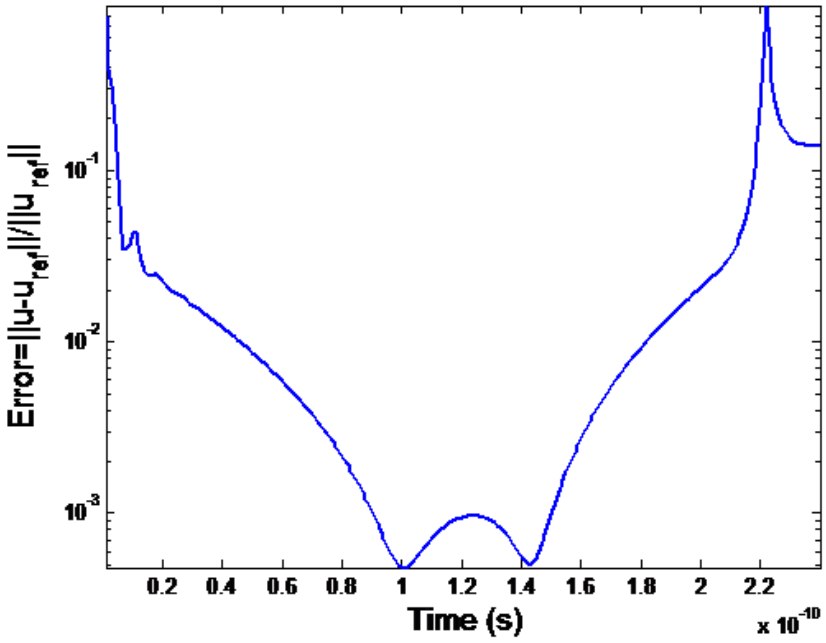


Fig. 4.5. Total solution error plot of bus structure.

4.4 Conclusions

In this chapter, we demonstrate a new explicit method for achieving unconditional stability by directly excluding unstable modes in an explicit time-domain finite-element method. The removal of the unstable modes from the numerical system does not require a preprocessing stage that may consume large computational resources. We then perform an unstable-modes-free explicit time marching over the updated system matrix. As a result, the explicit time marching is made stable for the given time step no matter how large it is. Numerical experiments have demonstrated the accuracy, efficiency, and unconditional stability of the proposed method.

5. EXPLICIT AND UNCONDITIONALLY STABLE TDFEM FOR ANALYZING GENERAL LOSSY PROBLEMS

5.1 Introduction

In this chapter, we extend the method in previous chapter to analyze general lossy problems where both dielectrics and conductors can be lossy and inhomogeneous. This class of problems is important for on-chip circuit analysis because lossy conductors are dominant in on-chip circuits. One solution to the problem is to separate the system of equations formulated inside conductors from those outside of conductors, and handle stability separately. However, this approach cannot yield a correct set of unstable eigenmodes for the given time step, due to the decoupled consideration of solutions in the dielectric region and those inside lossy conductors. In this Chapter, we present a coupled approach that finds the unstable eigenmodes for the given time step, making the time step of the proposed method solely determined by accuracy regardless of space step. The contents of this chapter have been extracted and revised from the following publication: Woochan Lee and Dan Jiao, "An Alternative Explicit and Unconditionally Stable Time-Domain Finite-Element Method for Electromagnetic Analysis," IEEE Transactions on Antennas and Propagation, *submitted*.

5.2 Proposed Method

First, we start with following linear system of equations which is also described in (3.1)

$$\mathbf{T}\ddot{u}(t) + \mathbf{R}\dot{u}(t) + \mathbf{S}u(t) = \dot{I}(t). \tag{5.1}$$

Here, we propose to first transform (1) to the following first-order double-dimension system of equation without any approximation

$$\begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{T} & \mathbf{0} \end{bmatrix} \frac{d}{dt} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} + \begin{bmatrix} \mathbf{S} & 0 \\ 0 & -\mathbf{T} \end{bmatrix} \begin{Bmatrix} u \\ \dot{u} \end{Bmatrix} = \begin{Bmatrix} \dot{I} \\ 0 \end{Bmatrix} \tag{5.2}$$

which can then be converted to

$$\begin{gathered} \tfrac{d}{dt}\tilde{u} + \mathbf{M}\tilde{u} = \tilde{b} \\ \mathbf{M} = \mathbf{A^{-1}B}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{T} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{S} & 0 \\ 0 & -\mathbf{T} \end{bmatrix}, \end{gathered} \tag{5.3}$$

where $\mathbf{M}$ is the system matrix. The solution of , whose upper part is the original field solution of (5.1), is governed by the eigenmodes of the following generalized eigenvalue problem:

$$\mathbf{MV} = \mathbf{V\Lambda} \tag{5.4}$$

in which Λ denotes a diagonal matrix whose entries are eigenvalues λ , and $\mathbf{V}$ is the eigenvector matrix where $\mathbf{V} = [\mathbf{V}_s \mathbf{V}_h]$. The stable modes ($\mathbf{V}_s$) that can be stably simulated by the time step Δt are determined by the following criterion (proof in Appendix A):

$$2real(\lambda)/(real(\lambda)^2 + imag(\lambda)^2) > \Delta t. \tag{5.5}$$

The rest of the eigenmodes that do not satisfy the above are denoted by $\mathbf{V}_h$, and termed unstable modes. They have the largest eigenvalues of (5.4).

Since $\mathbf{V}_h$ have large eigenvalues, they correspond to high-frequency modes. Since at high frequencies, skin depth of a conductor is almost zero, the fields penetrating into conductors are negligible. Therefore, the $\mathbf{V}_h$ modes found for a lossy problem have a good correlation with $\mathbf{V}_h$ modes of the lossless sub-system formulated outside conductors, i.e., with conductors acting like perfect conductors. As a result, we can deduce the following property:

$$\mathbf{V}^T \mathbf{A} \mathbf{V} = [\mathbf{V}_s \mathbf{V}_h]^T [\mathbf{A}] [\mathbf{V}_s \mathbf{V}_h] = \begin{bmatrix} \mathbf{A}_t & 0 \\ 0 & \mathbf{D}_h \end{bmatrix}, \tag{5.6}$$

where

$$\mathbf{A}_t = {\mathbf{V}_s}^T \mathbf{A} \mathbf{V}_s, \mathbf{D}_h = {\mathbf{V}_h}^T \mathbf{A} \mathbf{V}_h, {\mathbf{V}_s}^T \mathbf{A} \mathbf{V}_h = {\mathbf{V}_h}^T \mathbf{A} \mathbf{V}_s = 0, \tag{5.7}$$

and $\mathbf{D}_h$ is diagonally dominant. Also, this property can be mathematically proven as in [30].

Based on the aforementioned understanding, before we run an explicit time-domain simulation, if we are able to exclude the unstable eigenmodes from the underlying numerical system, we can ensure the stability of the simulation for the given time step as well as solution accuracy. Thus, we update the original system of equations (5.3) to a new system of equations as the following

$$\frac{d}{dt}\tilde{u} + \mathbf{M}(\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A})\tilde{u} = \tilde{b}, \tag{5.8}$$

where $\mathbf{M}$ is changed to new system matrix $\mathbf{M}(\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A})$. The above modification is the same as changing the original system involving both unstable and stable modes to a system of stable modes only. The solution of (5.8) is now governed by the eigensolution of a new system matrix $\mathbf{M}(\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A})$ instead of the original $\mathbf{M}$. Next, we prove the eigenvalues and eigenmodes of the updated matrix are the same as the stable eigenvalues and eigenmodes of the original $\mathbf{M}$, with an additional nullspace of size of $\mathbf{V}_h$.

From (5.6), we obtain

$$\mathbf{V}^{-1} = \begin{bmatrix} \mathbf{A}_t^{-1} & 0 \\ 0 & \mathbf{D}_h^{-1} \end{bmatrix} \mathbf{V}^T \mathbf{A}. \tag{5.9}$$

Hence, the original $\mathbf{M}$ can be rewritten as

$$\begin{aligned} \mathbf{M} = \mathbf{V}\Lambda\mathbf{V}^{-1} &= [\mathbf{V}_s \mathbf{V}_h] [\Lambda] \begin{bmatrix} \mathbf{A}_t^{-1} & 0 \\ 0 & \mathbf{D}_h^{-1} \end{bmatrix} [\mathbf{V}_s \mathbf{V}_h]^T \mathbf{A} \\ &= [\mathbf{V}_s \mathbf{V}_h] \begin{bmatrix} \Lambda_s & 0 \\ 0 & \Lambda_h \end{bmatrix} \begin{bmatrix} \mathbf{A}_t^{-1} & 0 \\ 0 & \mathbf{D}_h^{-1} \end{bmatrix} [\mathbf{V}_s \mathbf{V}_h]^T \mathbf{A} \\ &= \mathbf{V}_s \Lambda_s \mathbf{A}_t^{-1} \mathbf{V}_s^T \mathbf{A} + \mathbf{V}_h \Lambda_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A} \end{aligned} \tag{5.10}$$

Therefore, the modified system matrix is nothing but

$$\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A}) = \mathbf{V}_s\Lambda_s\mathbf{A}_t^{-1}\mathbf{V}_s^T\mathbf{A}. \tag{5.11}$$

Denote $\mathbf{V}_s\Lambda_s\mathbf{A}_t^{-1}\mathbf{V}_s^T\mathbf{A}$ by $\mathbf{K}$, then it is clear that $\mathbf{K}\mathbf{V}_s = \mathbf{V}_s\mathbf{\Lambda}_s$ and hence $(\mathbf{\Lambda}_s, \mathbf{V}_s)$ is the eigenpair of $\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A})$. Therefore, the stable eigenvalues and eigenmodes of the original $\mathbf{M}$ are the eigenvalues and eigenmodes of the updated matrix $\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A})$. Furthermore, since the rank of (5.11) is the number of stable eigenmodes ks, while the matrix size of (5.11) is 2N, the is a low-rank matrix with $(2N-k_s)$ additional zero eigenvalues. Thus, the marching of the updated system (5.8) is absolutely stable for the given time step.

With a forward-difference based 1st-derivative double dimension time-marching scheme, (5.8) can be discretized as

$$\tilde{u}^{n+1} = \tilde{u}^n - \Delta t\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A})\tilde{u}^n + \Delta t\tilde{b}, \tag{5.12}$$

where the $\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A})$ term is efficiently evaluated by a sequence of sparse matrix-vector multiplications from right to left as well as the structure-aware $\mathbf{T}$'s solver.

Although (5.12) is absolutely stable for the given time step, to ensure accuracy, we still need to add one more step as the following to remove the contribution of the additional nullspace of $\mathbf{M}(\mathbf{I}-\mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A})$, which is not present in the original $\mathbf{M}$,

$$\tilde{u}^{n+1} = \tilde{u}^{n+1} - \mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A}\tilde{u}^{n+1}. \tag{5.13}$$

To explain, the nullspace of (5.8) is different from the nullspace of the original system (5.3). Although these modes can be stably simulated, they make the solution of (5.12) not accurate. This is because the solution of the original problem (5.3) only resides in the space of $\mathbf{V}_s$, but with (5.8), $\tilde{u} = \mathbf{V}_s y_s + \mathbf{V}_{a0} y_{a0}$, where $\mathbf{V}_{a0}$ denotes the additional unwanted nullspace of (5.8). Hence, the treatment of (5.13) will remove these additional nullspace modes since $\mathbf{V}_{a0}$ must be in the $\mathbf{V}_h$ space if it is not in $\mathbf{V}_s$.

5.2.1 Explicit Time Marching Scheme based on Central Difference

For a general lossy problem discretized into a second-order system shown in (5.1), we can directly simulate it with a *central-difference-based* explicit time marching. The scheme described above transforms (5.1) to a first-order system, and simulates the resultant with a *forward-difference-based* explicit marching. The two schemes, i.e., central-difference based vs. forward difference based scheme, have a different requirement on time step for stability. When there is a conductor loss, the time step required by a forward-difference explicit marching can be much smaller than that of the central-difference-based explicit marching. Although in the proposed new method, we remove the unstable modes from the numerical system according to time step, and hence allowing for the forward-difference scheme to use any large time step. However, from an accuracy point of view, for simulating the same set of eigenmodes kept in the numerical system, the time step required by a forward-difference explicit marching for stably simulating these modes is smaller than that of a central-difference-based explicit marching. In this subsection, we analyze this problem and present a central-difference-based explicit marching scheme for simulating the lossy problems with unconditional stability.

Using a central-difference based explicit marching of (5.1), the time step required for stably simulating an eigenmode of λ_i eigenvalue satisfies the following condition [31], as shown in below

$$\Delta t \leq \frac{2}{\sqrt{|\lambda_{i1}\lambda_{i2}|}}. \tag{5.14}$$

However, using a forward-difference scheme, as shown in (5.5), the time step needs to satisfy

$$\Delta t \leq \frac{2|\mathrm{Re}(\lambda_i)|}{|\lambda_i|^2}. \tag{5.15}$$

For an eigenvalue pair having identical negative eigenvalues, (5.15) is the same as (5.14). For complex-conjugate eigenvalues, from (3.9), we can find

$$\lambda_i = \frac{-b_i \pm \sqrt{{b_i}^2 - 4c_i}}{2}, \tag{5.16}$$

where $b_i = V_i^H \mathbf{R} V_i / V_i^H \mathbf{T} V_i, c_i = V_i^H \mathbf{S} V_i / V_i^H \mathbf{T} V_i$ and both are greater than zero. Since for complex-conjugate eigenvalues, ${b_i}^2 < 4c_i$, and $|\lambda_i| = \sqrt{c_i}$, we have $|\mathrm{Re}(\lambda_i)| = b_i/2 < |\lambda_i|$. Hence, the time step of (5.15) is smaller than that required in (5.14), because $\Delta t \leq \frac{2|\mathrm{Re}(\lambda_i)|}{|\lambda_i|^2} < \frac{2}{|\lambda_i|}$.

Given an input spectrum, once space discretization is done, the eigenmodes that are important to the field solution in the input spectrum are known. The time step required by accuracy is hence determined by the time step that can accurately simulate these physically important eigenmodes. Based on the aforementioned analysis, for simulating the same complex-conjugate eigenmode in a lossy problem, the time step required by a forward-difference scheme can be smaller than that of a central-difference based time marching scheme. Hence, it is necessary to devise a central-difference based explicit method for simulating lossy problems in the proposed work.

We propose to perform a leap-frog-based time marching of the first-order system of (5.3). This will yield the same central-difference-based time marching of the original second-order system (5.1), and hence resulting in a time step of (5.14) for stability, which is larger than that allowed by the forward-difference-based time marching for simulating the same eigenmode. To explain, we can write (5.3) in full as

$$\frac{d}{dt}\begin{Bmatrix} u \\ w \end{Bmatrix} - \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{T}^{-1}\mathbf{S} & -\mathbf{T}^{-1}\mathbf{R} \end{bmatrix}\begin{Bmatrix} u \\ w \end{Bmatrix} = \begin{Bmatrix} 0 \\ \tilde{b}_2 \end{Bmatrix}, \tag{5.17}$$

where $w = \dot{u}$, which is also an unknown to be solved together with the field solution u, and $\tilde{b}_2$ is the lower half of vector $\tilde{b}$. Using a leap-frog based time marching, the above double-sized first-order system can be marched on in time as follows

$$\begin{aligned} &u^n - u^{n-1} = \Delta t w^{n-\frac{1}{2}} \\ &w^{n+\frac{1}{2}} - w^{n-\frac{1}{2}} + \Delta t \mathbf{T}^{-1}\mathbf{S}u^n + \Delta t \mathbf{T}^{-1}\mathbf{R}\tfrac{w^{n+\frac{1}{2}}+w^{n-\frac{1}{2}}}{2} = \Delta t \tilde{b}_2^n \end{aligned}, \tag{5.18}$$

which can be rearranged to solve u and w at the most advanced time step as:

$$u^n = u^{n-1} + \Delta t w^{n-\frac{1}{2}} \tag{5.19}$$

$$(\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})w^{n+\frac{1}{2}} = (\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})w^{n-\frac{1}{2}} - \Delta t \mathbf{T}^{-1}\mathbf{S}u^n + \Delta t \tilde{b}_2^n. \tag{5.20}$$

The above is equivalent to a central-difference based discretization of (5.1). This can be readily proved as follows. Writing (5.20) for the $(n+1)$-th step, we obtain

$$u^{n+1} = u^n + \Delta t w^{n+\frac{1}{2}}. \tag{5.21}$$

Multiplying both sides by $(\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})$, we have

$$(\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^{n+1} = (\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^n + \Delta t(\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})w^{n+\frac{1}{2}}. \tag{5.22}$$

Multiplying (5.19) by $(\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})$ on both sides, we obtain

$$(\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^n = (\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^{n-1} + \Delta t(\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R})w^{n-\frac{1}{2}}. \tag{5.23}$$

Subtracting (5.23) from (5.22), and substituting (5.20), we have

$$(\mathbf{I}+0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^{n+1} = 2u^n - (\mathbf{I}-0.5\Delta t \mathbf{T}^{-1}\mathbf{R})u^{n-1} - \Delta t^2 \mathbf{T}^{-1}\mathbf{S}u^n + \Delta t^2 \mathbf{T}^{-1}\dot{I}^n, \tag{5.24}$$

which is the same as a central-difference-based discretization of (5.1). Hence, by performing a time marching of the first-order system (5.3) in a leap-frog-based way shown in (5.19-5.20), the time step required for stably simulating an eigenmode is the same as that of a central-difference based time marching of the second-order system.

With the unstable modes $\mathbf{V}_h$ satisfying

$$\sqrt{|\lambda_{i1}\lambda_{i2}|} > \frac{2}{\Delta t}, i \in (1, N) \tag{5.25}$$

found from (5.4), to make the above leap-frog scheme shown in (5.19-5.20) stable for any time step, what we only need to do is as follows. After (5.19), we form vector $\tilde{u} = \left[u^n w^{n-\frac{1}{2}}\right]^T$, and deduct the unstable modes from it by updating it to be $\tilde{u} = (\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A})\tilde{u}$. The u^n is then taken as the upper half of $\tilde{u}$ to be free of unstable modes, and used in (5.20) to compute $w^{n+\frac{1}{2}}$. After the computation of (5.20) for obtaining $w^{n+\frac{1}{2}}$, we form $\tilde{u} = \left[u^n w^{n+\frac{1}{2}}\right]^T$, update it to be $\tilde{u} = (\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A})\tilde{u}$ so that the unstable modes are removed. The u^n is then updated to be the upper

half of $\tilde{u}$, while the $w^{n+\frac{1}{2}}$ is updated to be the lower half of $\tilde{u}$. We then continue to next time step. The entire procedure is summarized as the following.

$$
\begin{aligned}
&(1) u^n = u^{n-1} + \Delta t w^{n-\frac{1}{2}} \\
&(2) \tilde{u} = \left[u^n w^{n-\frac{1}{2}} \right]^T \\
&\quad \tilde{u} = (\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A}) \tilde{u} \\
&\quad u^n = \tilde{u}(1:N) \\
&(3) (\mathbf{I} + 0.5\Delta t \mathbf{T}^{-1}\mathbf{R}) w^{n+\frac{1}{2}} = (\mathbf{I} - 0.5\Delta t \mathbf{T}^{-1}\mathbf{R}) w^{n-\frac{1}{2}} - \Delta t \mathbf{T}^{-1}\mathbf{S} u^n + \Delta t \tilde{b}_2^n \\
&(4) \tilde{u} = \left[u^n w^{n+\frac{1}{2}} \right]^T \\
&\quad \tilde{u} = (\mathbf{I} - \mathbf{V}_h \mathbf{D}_h^{-1} \mathbf{V}_h^T \mathbf{A}) \tilde{u} \\
&\quad u^n = \tilde{u}(1:N); w^{n+\frac{1}{2}} = \tilde{u}(N+1:2N)
\end{aligned}
\tag{5.26}
$$

where steps (1) and (3) are the same as the original (5.19) and (5.20), but steps (2) and (4) are added to ensure the unstable modes are removed from the numerical system at each time step.

5.2.2 Scaling

During the study of this work, we found that when $\mathbf{T}$, $\mathbf{S}$, and $\mathbf{R}$ are very different in their norm, the solution of the standard eigenvalue problem (5.4), which is equivalent to the original quadratic eigenvalue problem (3.9), may have a poor accuracy in numerical computation. This is especially true when the problems being simulated involve conductor loss and/or multiple scales. We hence adopt an optimal scaling technique introduced in [32, 33] to achieve good accuracy in the solution of (5.4) for finding the unstable modes. Based on this optimal scaling technique, the matrix $\mathbf{T}$, $\mathbf{S}$, and $\mathbf{R}$ in (5.3) are scaled to

$$
\tilde{\mathbf{T}} = \alpha^2 \beta \mathbf{T}; \tilde{\mathbf{S}} = \beta \mathbf{S}; \tilde{\mathbf{R}} = \alpha \beta \mathbf{R}, \tag{5.27}
$$

respectively, where

$$\begin{aligned}\alpha &= \sqrt{\gamma_0/\gamma_2} \\ \beta &= 2/(\gamma_0 + \gamma_1\sqrt{\gamma_0/\gamma_2}) \\ \gamma_2 &= ||\mathbf{T}||_2, \gamma_1 = ||\mathbf{R}||_2, \gamma_0 = ||\mathbf{S}||_2\end{aligned} \tag{5.28}$$

Correspondingly, the first-order double-sized system (5.2) is updated as the following:

$$\frac{1}{\alpha}\begin{bmatrix}\tilde{\mathbf{R}} & \tilde{\mathbf{T}} \\ \tilde{\mathbf{T}} & \mathbf{0}\end{bmatrix}\frac{d}{dt}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u}\end{Bmatrix} + \begin{bmatrix}\tilde{\mathbf{S}} & 0 \\ 0 & -\tilde{\mathbf{T}}\end{bmatrix}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u}\end{Bmatrix} = \begin{Bmatrix}\beta\dot{I} \\ 0\end{Bmatrix}, \tag{5.29}$$

which can be compactly written as

$$\frac{d}{dt}\tilde{u}' + \tilde{\mathbf{M}}\tilde{u}' = \tilde{b}', \tag{5.30}$$

where

$$\begin{aligned}\tilde{u}' &= \begin{bmatrix} u \\ \alpha^{-1}\dot{u}\end{bmatrix}, \tilde{\mathbf{M}} = \tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}, \\ \tilde{\mathbf{A}} &= \tfrac{1}{\alpha}\begin{bmatrix}\tilde{\mathbf{R}} & \tilde{\mathbf{T}} \\ \tilde{\mathbf{T}} & \mathbf{0}\end{bmatrix}, \tilde{\mathbf{B}} = \begin{bmatrix}\tilde{\mathbf{S}} & 0 \\ 0 & -\tilde{\mathbf{T}}\end{bmatrix}.\end{aligned} \tag{5.31}$$

In this paper, all the lossy examples are simulated with the above scaled numerical system (5.30) instead of the original unscaled system (5.3). As can be seen from in (5.31), the upper half of the solution vector obtained from (5.30) is the same as that of (5.3). The unstable modes are hence found from the scaled system matrix $\tilde{\mathbf{M}}$, the accuracy of which is much improved.

5.3 Numerical Results

5.3.1 Shorted On-Chip Stripline

A shorted on-chip stripline as shown in Fig. 3.3 is simulated to validate the proposed method. The input current source is launched from the bottom metal layer to the inner conductor, and it has a Gaussian derivative pulse with $\tau = 3 \times 10^{-9}$ s. The time step used in the proposed method is 1×10^{-10} s solely determined by

accuracy while the time step of the central difference based conventional TDFEM is 1×10^{-15} s. Based on the required time step of 1×10^{-10} s, 1,400 over total 1,948 eigenmodes are identified as unstable modes. The CPU time of the conventional TDFEM is 3.729×10^4 s, while the time including eigenvalue analysis which identify unstable modes and marching time of the proposed method is 2.927×10^2, thus speed up is 127.3. In comparison, the speedup of algorithm shown in chapter 3 is 1.835 with the same setting. In Fig. 5.1, the near end voltage of the proposed method in comparison with the reference data obtained from the traditional TDFEM solution is shown. Excellent agreement is observed.

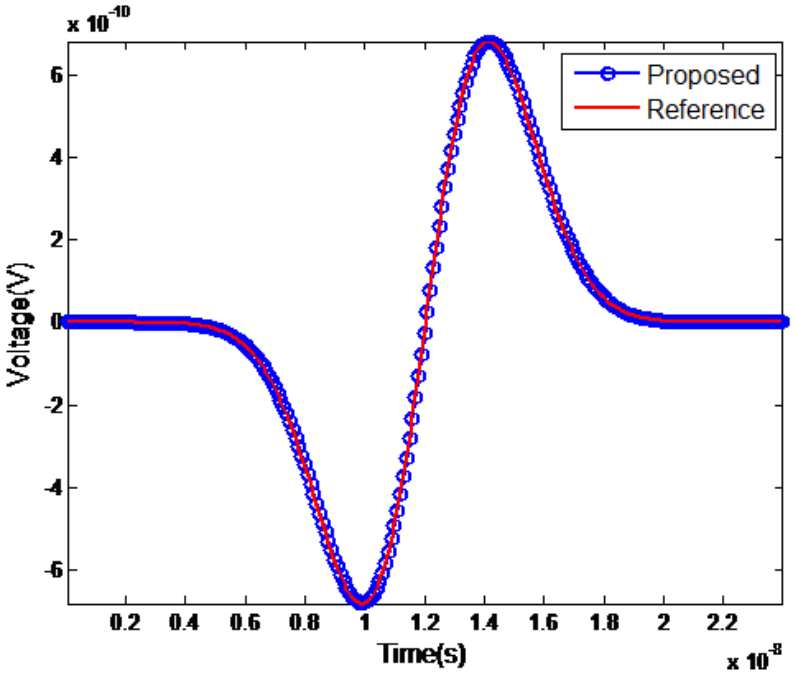


Fig. 5.1. Accuracy validation of the proposed algorithm in simulating a shorted on-chip stripline.

5.3.2 On-Chip Power Grid

The second example is an on-chip power grid structure as shown in Fig. 3.5. The size of the structure is 7.0μm $\times$ 7.0μm $\times$ 7.6μm. The top and bottom planes

are set to be PEC and the other 4 sides are left open. The number of unknowns in this example is 1,101. The permittivity is layered as shown in Fig. 3.5(c), and the conductivity of the metal is 5.0×10^7 S/m. The time step used in the proposed method is 1×10^{-10} s solely determined by accuracy. Based on the required time step of 1×10^{-10} s, 1,628 over total 2,202 eigenmodes are identified as unstable modes. The CPU time of the conventional TDFEM is 8.950×10^4 s, while the time including eigenvalue analysis which identify unstable modes and marching time of the proposed method is 2.324×10^2, thus speed up is 385.1. In comparison, the speedup of the algorithm shown in chapter 3 is 251.7 with the same setting. In Fig. 5.2, the near end voltage of the proposed method in comparison with the reference data obtained from the traditional TDFEM solution is shown. Excellent agreement is observed.

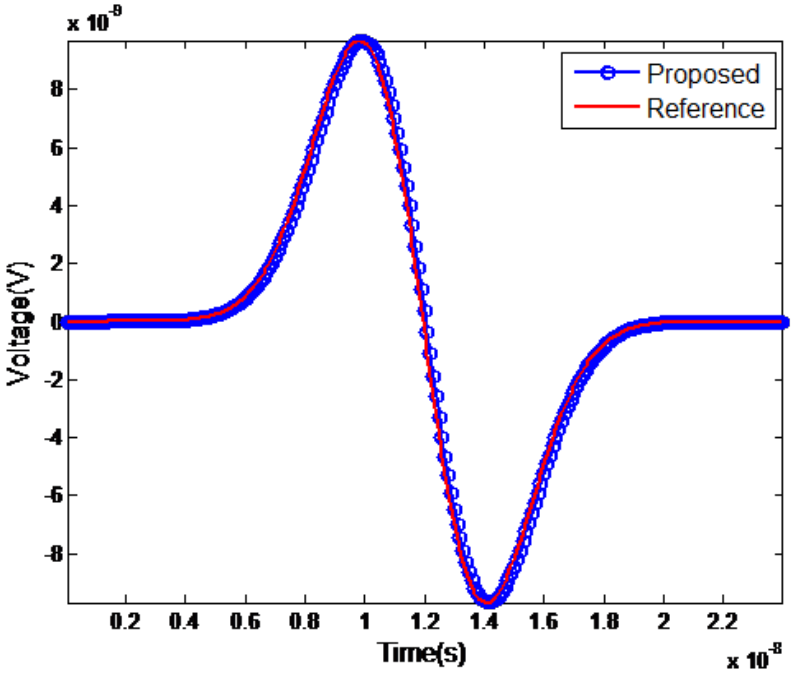


Fig. 5.2. Accuracy validation of the proposed algorithm in simulating on-chip power grid.

5.3.3 Rectangular Spiral Inductor

We then simulate an on-chip spiral inductor to validate the proposed algorithms. The structure along with its permittivity configuration is illustrated in Fig. 3.7. The entire computational domain occupies a region of 1200 μm by 1000 μm by 500 μm. The top and bottom planes are truncated by a PEC boundary condition, while all the other boundaries are left open. The number of unknowns in this example is 14,286. The input current source has a Gaussian derivative pulse with $\tau = 3 \times 10^{-8}$ s and $t_0 = 4\tau$ It is launched from the bottom PEC plane to the left port of the inductor as shown in Fig. 3.7 by the red arrow. The time step used in the conventional central difference based TDFEM is 5×10^{-14} s while the time step of current time marching scheme is 1×10^{-9} solely determined by accuracy. Based on the required time step of 1×10^{-9} s, 18,860 over total 28,572 eigenmodes are identified as unstable modes. In Fig. 5.3, the far-end voltage of the proposed method in comparison with the reference data obtained from the traditional TDFEM solution is shown. Excellent agreement is observed again.

5.3.4 Lossy Multiscale Structure

In the previous two lossy examples, the time step from the forward-difference-based explicit marching is the same as that of the central-difference-based one because the stable eigenmodes kept in the numerical system turn out to be nullspace modes only. In the last lossy example, we examine the validity of the proposed central-difference-based explicit time marching scheme described in Section 5.2.1 in a problem where the time step resulting from a forward-difference explicit marching and that of a central-difference marching is very different.

The structure is illustrated in Fig. 5.4, where there are two thin wires of width 1 nm each, and the total width of the structure is the sum of 4.5 cm, 3.5 mm, and 2 nm. This problem setup resembles a multiscale integrated structure where board-level planes co-exist with on-chip interconnects. In such a problem, regular structures

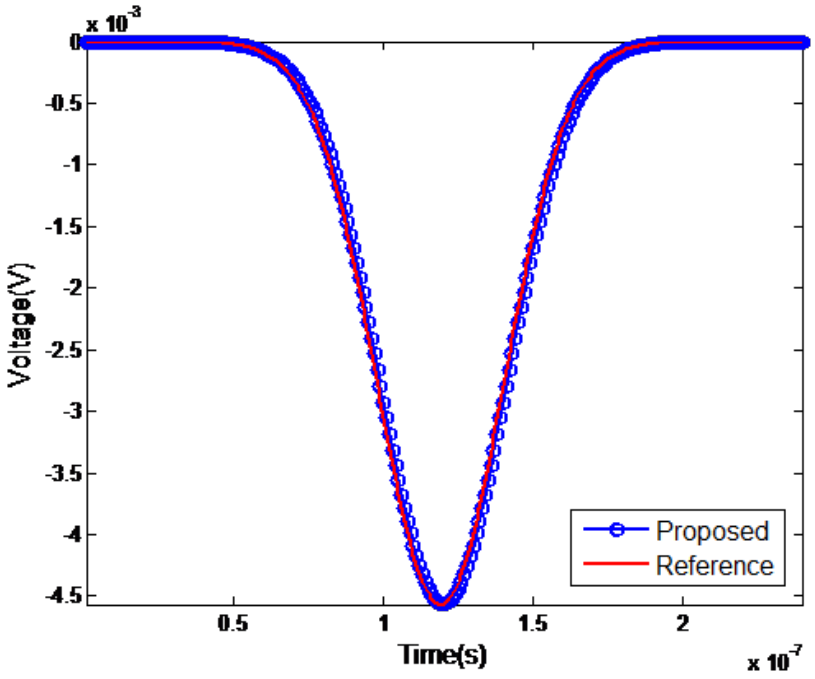


Fig. 5.3. Accuracy validation of the proposed algorithm for the simulation of a rectangular spiral inductor.

(compared to wavelength) co-exist with fine features, which is different from previous two on-chip examples where the entire structure is electrically small. For such a multiscale problem, unstable modes only occupy a portion of the entire number of modes; and the unstable mode number is proportional to the mesh elements used to discretize the fine features. There are three layers of 0.5 mm thickness each, having the permittivities shown in Fig. 5.4. The number of unknowns in this example is 3,628, and hence 7256 modes of (5.4). The input current sources are launched from the bottom and the top metal plate to the inner lossy conductor of conductivity 5.8×10^7 S/m. The sources have a Gaussian derivative pulse with $\tau = 3 \times 10^{-11}$ s and $t_0 = 4\tau$. The time step used in the proposed method is 1×10^{-12} s solely determined by accuracy while the time step of the central-difference based conventional TDFEM is 3×10^{-15} s. Based on the required time step of 1×10^{-12} s, 130 eigenmodes are identified as unstable modes and removed from the numerical system based on

(5.26). In this example, if a forward-difference explicit marching is used, the time step would have to be as small as 3×10^{-26} s for simulating the same set of stable modes kept in the numerical system. This is because many of them are complex-conjugate eigenvalues, which render the time step resulting from (5.15) much smaller than that of (5.14).

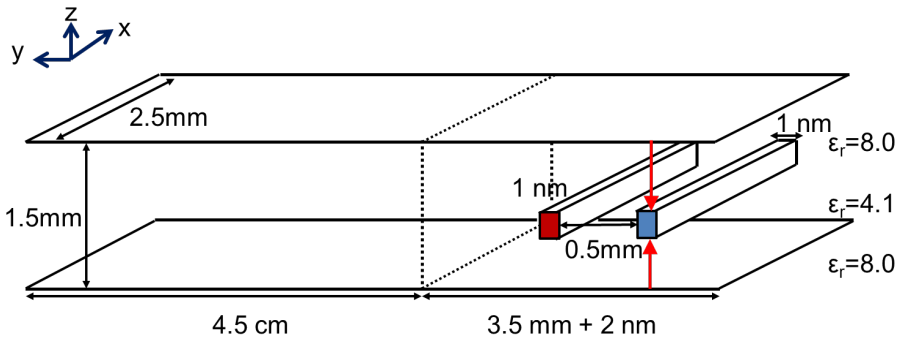


Fig. 5.4. Geometry of a lossy multiscale structure.

The marching time of the conventional TDFEM is 1.9923×10^2 s. In contrast, the marching time of the proposed leap-frog central-difference algorithm is 14.1493 s, and the CPU time spent on finding the unstable modes is only 2.1216 s. Again, very good agreement between the proposed method and the conventional TDFEM is observed as can be seen from the waveforms plotted in Fig. 5.5.

The structure is then further enlarged to result in a larger number of unknowns of 180,028, and hence 360,056 total number of modes of (5.4). To be specific, the left segment of 4.5 cm width of Fig. 5.4 is duplicated to the left to enlarge the width of the structure as well as the number of unknowns. For this large case, the conventional TDFEM takes more than 16 hours to finish the entire explicit time marching, whereas the proposed explicit method only takes 125 minutes for explicit time marching, with less than 7 minutes spent on finding the unstable modes. Out of the 360,056 total number of modes, only 130 modes are unstable. This number is also the same as that obtained from the original structure. This is because the fine features remain the same when we enlarge the structure.

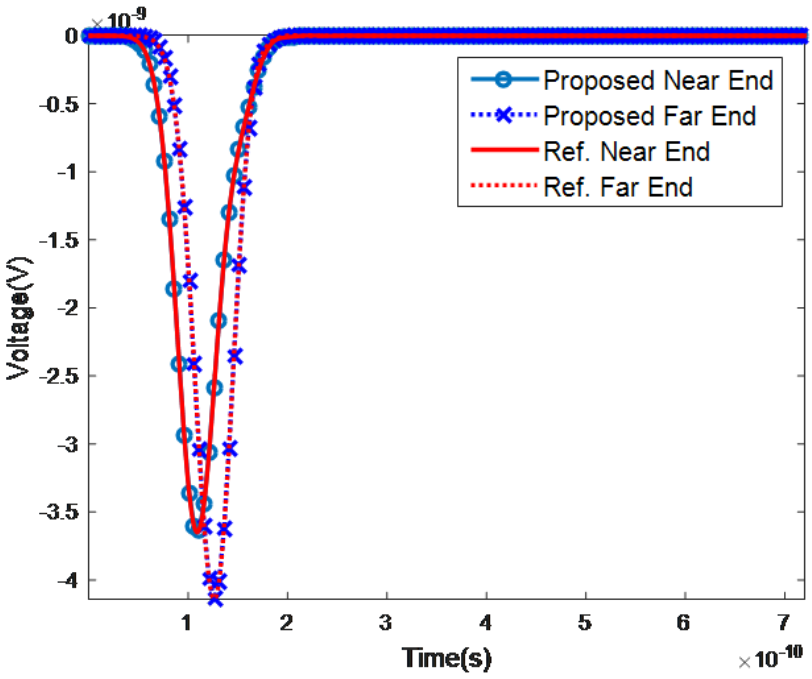


Fig. 5.5. Accuracy validation of lossy multiscale structure.

5.4 Conclusions

In this chapter, an alternative explicit and unconditionally stable TDFEM is developed for analyzing general lossy problems. In this method, the source of instability is upfront deducted from the system matrix before performing explicit time marching. As a result, the explicit time marching is made absolutely stable for the given time step no matter how large it is. The accuracy of the proposed method is also theoretically guaranteed when the time step is chosen based on accuracy. The proposed method is convenient for implementation since it only requires a minor modification of the traditional explicit TDFEM method to eradicate the source of instability. The additional computation involved in the proposed method as compared with a traditional TDFEM is mainly the cost of finding unstable modes. Since the unstable modes have the largest eigenvalues of the sparse TDFEM system matrix, they can be found efficiently in $O(k^2N)$ complexity, where k is the number of unstable modes.

In addition, these modes are frequency, time, and right hand side independent. Once found, they can be reused for different simulations.

The proposed new method complements the recently developed explicit and unconditionally stable TDFEM in [19]. When the fine features only occupy a small portion of the entire structure, the proposed method can be more advantageous to use as compared to [19], since the number of unstable modes is small whereas the number of stable modes is many. The two methods can also be combined for use to accentuate the advantages of both methods.

6. STRUCTURE-AWARE TIME-DOMAIN FINITE-ELEMENT SOLVER FOR GENERAL FULL-WAVE ANALYSIS

6.1 Introduction

The merit of the mass matrix solver preserving Manhattan-type structure and layered permittivity is manifested in chapters 2 and 3. Also, the solver with series expansion of matrix exponential described in Chapter 2 does not limit the frequency range, i.e., it supports full-wave applications from low to very high frequencies. Theoretically, the matrix exponential of this solver is able to support an arbitrarily large time step, however, numerically, the time step used should be small to make the series expansion converge within a reasonable number of expansion terms. The root cause of this performance limitation is the norm or spectral radius of matrix $\mathbf{M}$ itself in $e^{-\mathbf{M}\Delta t}$. If the norm of $\mathbf{M}$ is huge, we have to choose a small time step and a large number of expansion terms to make the series expansion of matrix exponential to converge with good accuracy. In realistic on-chip simulations, the norm of the matrix exponential term without any treatment is large because the norm of $\mathbf{T^{-1}S}$ plays a great role to determine the norm of $\mathbf{M}$, the typical value of which is at the level of 1×10^{33} for on-chip circuits. To alleviate this huge norm problem, the scaling method can be adopted, but the whole norm of new $\mathbf{M}$ is still restricted by the norm of $\mathbf{T^{-1}R}$, thus our final choice of time step is around 1×10^{17} level which is smaller than the conventional time step. The algorithm of chapter 3 does not suffer from the aforementioned problem, however, its application is limited to the frequency range where DC-modes play a dominant role in the field solution. The aforementioned problem associated with chapters 2 and 3 can be addressed based on the ideas of chapters 4 and 5. To clarify, the deduction of unstable higher modes will lead to a significant reduction of

the norm of the system matrix $\mathbf{M}$ because the norm of $\mathbf{M}$ is determined by the large eigenvalues and these are already excluded by the unstable-mode removal procedure described in chapters 4 and 5. Therefore, the algorithm in chapters 4 and 5 combined with the matrix exponential expansion framework in chapter 2 can achieve a significant enlargement of time step, while retaining the capability of chapter 2 algorithm in handling general full-wave applications that are not restricted to DC-mode dominant cases.

In this chapter, we propose a new method for full-wave applications with unconditional stability and structure-preserving capability using matrix exponential. In this new method, we directly deduct the largest eigenvalue modes from the system matrix inside the matrix exponential component. As a result, we observe much reduced number of terms for convergence of matrix exponential, hence achieving an efficient structure-aware algorithm for general full-wave analysis of on-chip circuits.

6.2 Proposed Method

Again, a time-domain FEM solution of the second-order vector-wave equation for an integrated circuit problem results in the following linear system of equations as seen in (6.1)

$$\mathbf{T}\ddot{u}(t) + \mathbf{R}\dot{u}(t) + \mathbf{S}u(t) = \dot{I}(t). \tag{6.1}$$

Here, we propose to first transform (6.1) to the following first-order system of equations with scaling as seen in (5.29)

$$\frac{1}{\alpha}\begin{bmatrix} \tilde{\mathbf{R}} & \tilde{\mathbf{T}} \\ \tilde{\mathbf{T}} & \mathbf{0} \end{bmatrix}\frac{d}{dt}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u} \end{Bmatrix} + \begin{bmatrix} \tilde{\mathbf{S}} & 0 \\ 0 & -\tilde{\mathbf{T}} \end{bmatrix}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u} \end{Bmatrix} = \begin{Bmatrix} \beta\dot{I} \\ 0 \end{Bmatrix}, \tag{6.2}$$

which can then be analytically converted to

$$\frac{d}{dt}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u} \end{Bmatrix} + \alpha\begin{bmatrix} 0 & -\mathbf{I} \\ \widetilde{\mathbf{T}}^{-1}\widetilde{\mathbf{S}} & -\widetilde{\mathbf{T}}^{-1}\widetilde{\mathbf{R}} \end{bmatrix}\begin{Bmatrix} u \\ \alpha^{-1}\dot{u} \end{Bmatrix} = \alpha\begin{Bmatrix} 0 \\ \widetilde{\mathbf{T}}^{-1}\beta\dot{I} \end{Bmatrix}. \tag{6.3}$$

The equation (6.3) is governed by generalized eigenvalue problem as seen in (6.4)

$$\mathbf{B}_{new}\mathbf{V} = \mathbf{A}_{new}\mathbf{V}\mathbf{D} \tag{6.4}$$

where $\mathbf{A}_{new} = \frac{1}{\alpha}\begin{bmatrix} \tilde{\mathbf{R}} & \tilde{\mathbf{T}} \\ \tilde{\mathbf{T}} & \mathbf{0} \end{bmatrix}$, $\mathbf{B}_{new} = \begin{bmatrix} \tilde{\mathbf{S}} & 0 \\ 0 & -\tilde{\mathbf{T}} \end{bmatrix}$ and each element of the diagonal of matrix D is the eigenvalue (λ).

Also, the solution of the 1st order equation (6.3) can be numerically evaluated as [16],

$$\tilde{u}^{n+1} = \frac{\tilde{b}_{init}^{n+1}\Delta t}{2} + e^{-\widetilde{\mathbf{M}}\Delta t}\left[\frac{\tilde{b}_{init}^{n}\Delta t}{2} + \tilde{u}^{n}\right] \tag{6.5}$$

where $\widetilde{\mathbf{M}} = \mathbf{A}_{new}^{-1}\mathbf{B}_{new} = \alpha\begin{bmatrix} 0 & -\mathbf{I} \\ \widetilde{\mathbf{T}}^{-1}\widetilde{\mathbf{S}} & -\widetilde{\mathbf{T}}^{-1}\widetilde{\mathbf{R}} \end{bmatrix}$, $\tilde{u} = \begin{Bmatrix} u \\ \alpha^{-1}\dot{u} \end{Bmatrix}$, and $\tilde{b}_{init} = \alpha\begin{Bmatrix} 0 \\ \widetilde{\mathbf{T}}^{-1}\beta\dot{I} \end{Bmatrix}$. Using a forward-difference scheme, as shown in (5.5), the time step needs to satisfy (5.15) and the eigenmodes ($\mathbf{V}_h$) that are not necessary for the stable and accurate simulation under give time step Δt are determined by the following criterion:

$$2real(\lambda)/(real(\lambda)^2 + imag(\lambda)^2) \leq \Delta t. \tag{6.6}$$

Then, we deduct above eigenmodes ($\mathbf{V}_h$) from the $\widetilde{\mathbf{M}}$ during the evaluation of the matrix exponential-related term $e^{-\widetilde{\mathbf{M}}\Delta t}\left[\frac{\tilde{b}_{init}^{n}\Delta t}{2} + \tilde{u}^{n}\right]$ in (6.5). After removing $\mathbf{V}_h$, because they are associated with largest eigenvalues, the resultant matrix $\widetilde{\mathbf{M}}_{new}$ has a significantly reduced norm, thus the series expansion of $e^{-\widetilde{\mathbf{M}}\Delta t}\left[\frac{\tilde{b}_{init}^{n}\Delta t}{2} + \tilde{u}^{n}\right]$ can be evaluated much faster than that of $\widetilde{\mathbf{M}}$. To be specific, the evaluation of $e^{-\widetilde{\mathbf{M}}\Delta t}v = v - \widetilde{\mathbf{M}}\Delta t v + \frac{1}{2!}(\widetilde{\mathbf{M}}\Delta t)^2 v - \frac{1}{3!}(\widetilde{\mathbf{M}}\Delta t)^3 v + ...$ where $v = \left[\frac{\tilde{b}_{init}^{n}\Delta t}{2} + \tilde{u}^{n}\right]$ can be accelerated by the substitution of $\widetilde{\mathbf{M}}$ to $\widetilde{\mathbf{M}}_{new}$ which is

$$\widetilde{\mathbf{M}}_{new} = \widetilde{\mathbf{M}}(\mathbf{I} - \mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A}_{new}), \tag{6.7}$$

where $\mathbf{D}_h = \mathbf{V}_h^T\mathbf{A}_{new}\mathbf{V}_h$.

However, the calculation of (6.7) is not explicitly used in the implementation since the matrix of (6.7) is dense. Rather than using (6.7), the alternative implementation is used to exploit sparse matrix-vector multiplication as well as a structure-aware **T**-solver described in chapter 2. To start, we rewrite (6.5) as below

$$\tilde{u}^{n+1} = \frac{\mathbf{A}_{new}^{-1}\tilde{b}_{new}^{n+1}\Delta t}{2} + e^{-\mathbf{A}^{-1}\mathbf{B}\Delta t}\left[\frac{\mathbf{A}_{new}^{-1}\tilde{b}_{new}^{n}\Delta t}{2} + \tilde{u}^{n}\right], \tag{6.8}$$

where $\tilde{b}_{new} = \left\{ \begin{array}{c} \beta\dot{I} \\ 0 \end{array} \right\}$. Then, the evaluation of $\mathbf{A^{-1}}\tilde{b}_{new}$ and the operation which is $\mathbf{A}_{new}^{-1}\mathbf{B}_{new}$ multiplied by a certain vector x, i.e., $\mathbf{A}_{new}^{-1}\mathbf{B}_{new}x$ turns into simple algebraic operations which exploit the structure-aware $\mathbf{T}$-solver as seen in (6.9) and (6.10) which are

$$\mathbf{A}_{new}^{-1}\tilde{b}_{new} = \alpha \left[\begin{array}{cc} \mathbf{0} & \widetilde{\mathbf{T}}^{\mathbf{-1}} \\ \widetilde{\mathbf{T}}^{\mathbf{-1}} & -\widetilde{\mathbf{T}}^{\mathbf{-1}}\widetilde{\mathbf{R}}\widetilde{\mathbf{T}}^{\mathbf{-1}} \end{array} \right] \left\{ \begin{array}{c} \beta\dot{I} \\ 0 \end{array} \right\} = \alpha \left\{ \begin{array}{c} \mathbf{0} \\ \widetilde{\mathbf{T}}^{\mathbf{-1}}\beta\dot{I} \end{array} \right\}, \tag{6.9}$$

$$\begin{aligned} \mathbf{A}_{new}^{-1}\mathbf{B}_{new}x &= \alpha \left[\begin{array}{cc} \mathbf{0} & \widetilde{\mathbf{T}}^{\mathbf{-1}} \\ \widetilde{\mathbf{T}}^{\mathbf{-1}} & -\widetilde{\mathbf{T}}^{\mathbf{-1}}\widetilde{\mathbf{R}}\widetilde{\mathbf{T}}^{\mathbf{-1}} \end{array} \right] \left[\begin{array}{cc} \widetilde{\mathbf{S}} & \mathbf{0} \\ \mathbf{0} & -\widetilde{\mathbf{T}} \end{array} \right] \left\{ \begin{array}{c} x_1 \\ x_2 \end{array} \right\} \\ &= \alpha \left\{ \begin{array}{c} -x_2 \\ \widetilde{\mathbf{T}}^{\mathbf{-1}}(\widetilde{\mathbf{S}}x_1 + \widetilde{\mathbf{R}}x_2) \end{array} \right\}. \end{aligned} \tag{6.10}$$

The matrix-exponential term can be obtained from the recursive sum of matrix-vector multiplication having a form of $\mathbf{A}_{new}^{-1}\mathbf{B}_{new}$ multiplied by x, which exploits structure-aware $\mathbf{T}$-solver. For example, the third expansion term of $e^{-\mathbf{A}_{new}^{-1}\mathbf{B}_{new}\Delta t}x = x - (\mathbf{A}_{new}^{-1}\mathbf{B}_{new}\Delta t)x + \frac{1}{2!}(\mathbf{A}_{new}^{-1}\mathbf{B}_{new}\Delta t)^2 x - ...$ can be efficiently obtained by multiplying the second term with $\mathbf{A}_{new}^{-1}\mathbf{B}_{new}$ and scalar coefficients as seen in (6.10).

The overall procedure including efficient evaluation of series expansion of matrix exponential term is summarized in Algorithm 6.1. Step 1 is to prepare the vector which is later multiplied by matrix exponential component. It is noted that structure-aware $\mathbf{T}$-solver is used throughout the steps. Step 2 sets the first expansion term. Step 3 completes the series expansion of matrix exponential multiplied by the vector in (6.8). Finally, Step 4 is to produce the most time advanced output u_2.

Higher-order mode ($\mathbf{V}_h$) removal approach described in chapter 5 may be sensitive to the choice of higher-order modes if eigenvalue distribution is quite populated. In this case, the accuracy can be affected when $\mathbf{V}_h$ modes are deducted from the system matrix even though the entire simulation is stable. In contrast, matrix exponential framework is analytical in time stepping and it is less sensitive to the selection of $\mathbf{V}_h$ modes, i.e., suppress unremoved higher-order modes effect, because the purpose

Table 6.1.
Algorithm for evaluating matrix exponential components

Algorithm 6.1: Overall procedure with scaling strategy

1. $temp\_elem = 0.5\alpha\Delta t \begin{bmatrix} 0 \\ \widetilde{\mathbf{T}}^{-\mathbf{1}}(\beta \dot{I}_1) \end{bmatrix} + u_1$
2. $temp\_sum = temp\_elem$
3. for $i = 1, 2, ..., terms$
   - 3.1. $temp\_elem = temp\_elem - \mathbf{V}_h(\mathbf{D}_h^{-1}(\mathbf{V}_h^T(\mathbf{A}_{new}\, temp\_elem)))$
   - 3.2. $b\_temp = \widetilde{\mathbf{S}}\, temp\_elem(1:N) + \widetilde{\mathbf{R}}\, temp\_elem(N+1:2N)$
   - 3.3. $temp\_elem = -\alpha\Delta t \begin{bmatrix} -temp\_elem(N+1:2N) \\ \widetilde{\mathbf{T}}^{-\mathbf{1}}\, b\_temp \end{bmatrix}$
   - 3.4. $temp\_sum = temp\_sum + \frac{1}{i!} temp\_elem$
4. $u_2 = 0.5\alpha\Delta t \begin{bmatrix} 0 \\ \widetilde{\mathbf{T}}^{-\mathbf{1}}(\beta \dot{I}_2) \end{bmatrix} + temp\_sum$

of deducting of $\mathbf{V}_h$ is not to remove individual $\mathbf{V}_h$ but to reduce an overall norm of the system matrix. Also, theoretically there is no limit of the choice of time step dt in the matrix exponential based time marching for stability. It is worth mentioning that the second cleaning process as seen in (5.13) for the newly introduced nullspace is not necessary when the nullspace effect is negligible like full-wave applications.

6.3 Numerical Results

6.3.1 Stripline

We first simulate a stripline structure to validate the proposed algorithms. The cross sectional view of the structure with its permittivity and conductivity configuration is illustrated in Fig. 6.1. The length, width, and height of the structure are 120 μm, 30 μm, and 1.596 μm respectively. The top and bottom planes are PEC, while the front and back faces are truncated by ABC, and the other two faces are PMC (left open).

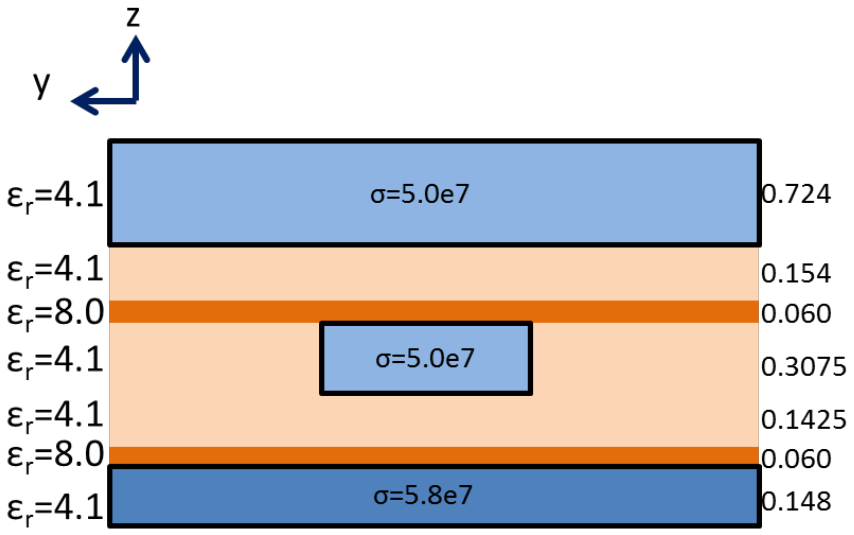


Fig. 6.1. The cross-sectional view of the stripline structure.

The number of unknowns in this example is 974. The input current source has a Gaussian derivative pulse of $I(t) = 2(t - t_0)\exp(-(t - t_0)^2/\tau^2)$, with $\tau = 3 \times 10^{-14}$ s (thus, maximum frequency approximately 34 THz) and $t_0 = 4\tau$. Based on the required time step of 1×10^{-15} s for the accuracy, 1070 over total 1948 eigenmodes are identified as higher-order modes.

For the comparison, with higher-order modes removal under matrix exponential framwork, the estimated norm of the new system matrix ($\widetilde{\mathbf{M}}_{new} = \widetilde{\mathbf{M}}(\mathbf{I} - \mathbf{V}_h\mathbf{D}_h^{-1}\mathbf{V}_h^T\mathbf{A}_{new})$) is 1.120 while the system matrix from original scheme ($\widetilde{\mathbf{M}}$) shows the norm of 103.2. Thus, approximately 1/100 of the norm value is achieved and it will lead to significant enlargement of time step and reduction of terms for series expansion of matrix exponential components. For example, previous original matrix exponential scheme requires the time step size of 1×10^{-17} s, 24000 iterations and 40 terms for the series expansion. In contrast, current scheme demonstrates the time step of 1×10^{-15} s solely determined by accuracy, 240 iteration for the simulation and 3 terms for the series expansion. In addition, the time step used in the conventional central difference based TDFEM is 5.25×10^{-17} s while the time step of current time marching scheme supports 1×10^{-15} s.

In Fig. 6.2, the near-end voltage of the current method in comparison with the reference result obtained from the traditional TDFEM solution is shown. We can observe excellent agreement again.

6.3.2 A Suite of Striplines

We then simulate a suite of stripline structure to validate the proposed algorithms. The dimensions of the basic block structure are 11 mm × 7 mm × 8.0000001 mm (inside conductor height 0.1nm). Also, the number of mesh elements along each x-, y-, z-direction is 11, 7 and 9, respectively. Domains are uniformly discretized by 1 mm except for inner conductor height 0.1 nm. A suite of 2 basic blocks of stripline structure is illustrated in Fig. 6.3 and the cross sectional view of the structure with

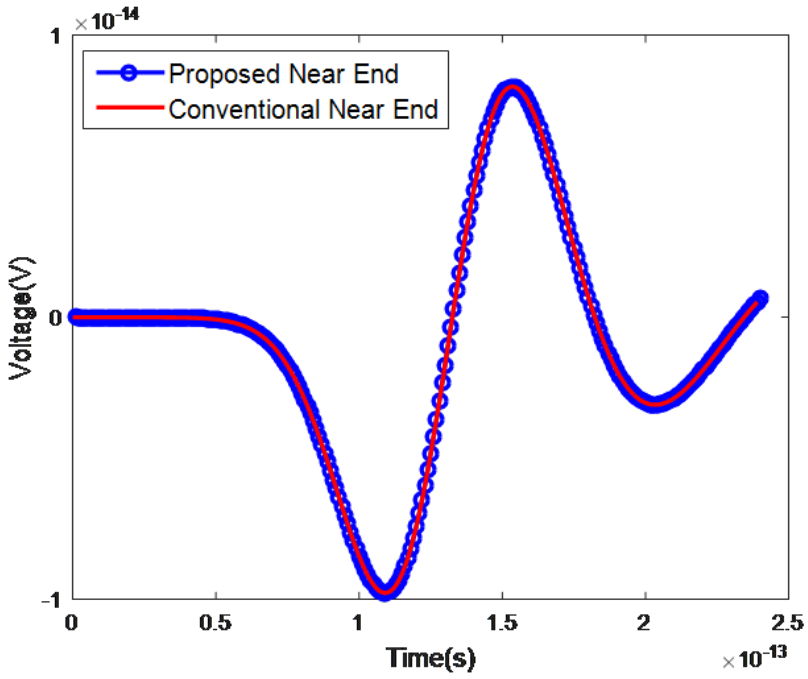


Fig. 6.2. Accuracy validation of the current algorithm for a stripline case.

its permittivity and conductivity distribution is shown in Fig. 6.4. The top and bottom planes are PEC, while the front and back faces are ABC, and the other two faces are PMC. The input current source has a Gaussian derivative pulse of $I(t) = 2(t-t_0)\exp(-(t-t_0)^2/\tau^2)$, with $\tau = 3\times10^{-11}$ s (thus, approximately 34 GHz) and $t_0 = 4\tau$. The current algorithm supports time step of 1×10^{-12} s determined by accuracy, 4 terms for the series expansion of matrix exponential related component and 20,000 steps are used to complete the simulation. In contrast, the time step used in the conventional central difference based TDFEM is 2.25×10^{-15} s.

First, for a basic block of stripline case (the left part only in Fig. 6.3), the number of unknowns of the basic block is 2240. Based on the time step of 1×10^{-12} s determined by accuracy, 364 over total 4480 eigenmodes are chosen as higher-order modes. The CPU time required for identifying the higher-order modes is 32.2999 s, and the CPU time for time marching with structure-aware $\mathbf{T}$-solver is 3.7123×10^2

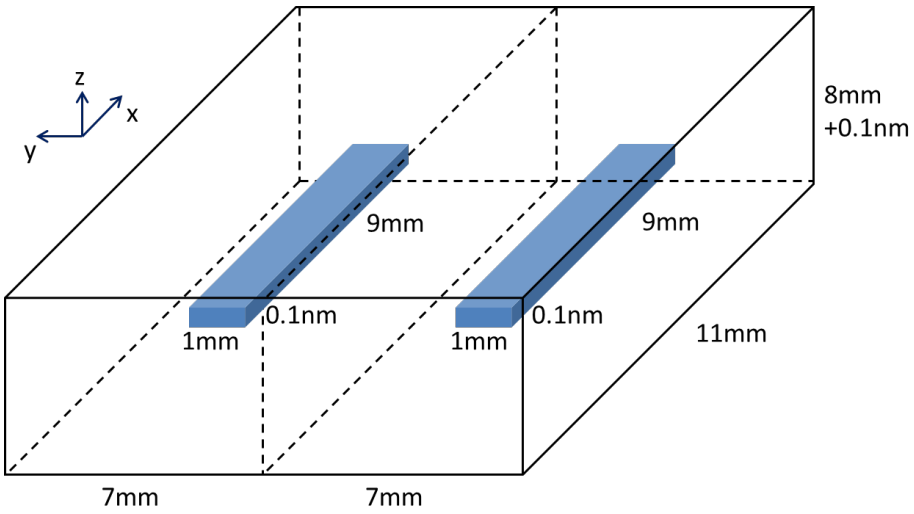


Fig. 6.3. A suite of two stripline structures.

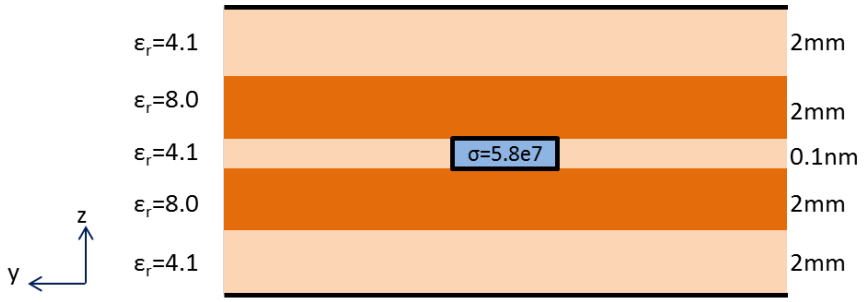


Fig. 6.4. Cross-sectional view of the basic stripline block.

s. In contrast, for the traditional method, 65.1199 s is required for LU factorization, and the marching time is 4.9888×10^2 s.

Second, for a suite of stripline cases (parallel expansion of basic blocks, Fig. 6.3), the number of unknowns of the basic block is 4284. Based on the time step of 1×10^{-12}s determined by the accuracy, 706 over total 8568 eigenmodes are chosen as higher-order modes. A time of 2.4533×10^2 s is required for identifying the higher-order modes, and the time for time marching is 9.6944×10^2 s. In contrast, for the traditional method, 9.5691×10^2 s is required for LU factorization, and the marching time is 1.6297×10^3 s.

In Fig. 6.5, the near-end/far end voltage of the current method in comparison with the reference result obtained from the conventional central-difference based scheme is shown. Excellent agreement between two methods is observed.

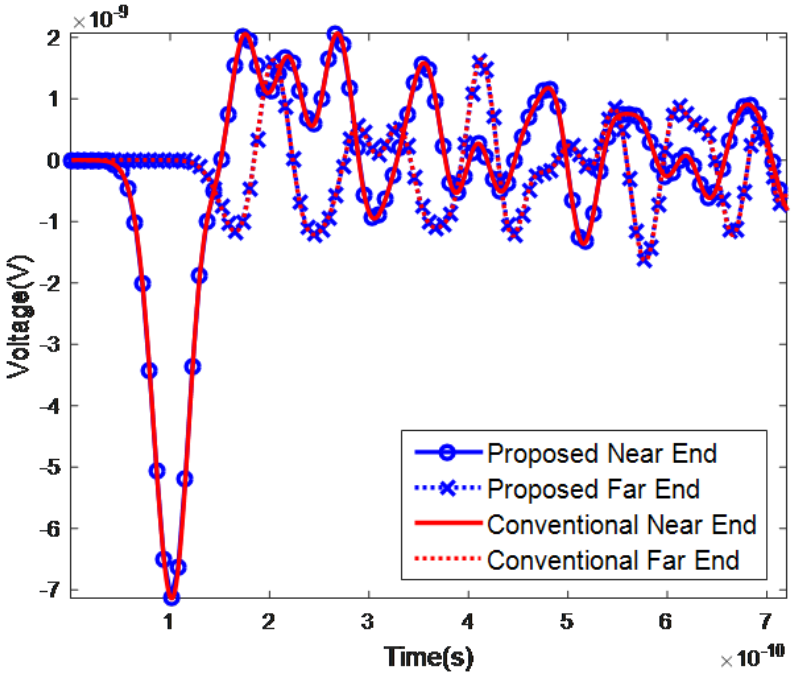


Fig. 6.5. Accuracy validation of the proposed method.

6.3.3 Lossy Multiscale Structure

The lossy multiscale structure illustrated in Fig. 5.4 again simulated to validate the proposed algorithm in chapter 6. The length and height of the structure is 2.5 mm, and 1.5 mm, respectively. The width of the structure is a sum of 4.5 cm (dielectric region), 3.5 mm, and 2 nm where the width of the inside conductor is 1 nm. The number of element along x-, y- and z-direction is 5, 9 and 3, respectively. Domains are uniformly discretized by 0.5 mm except inner conductor regions. The input is a Gaussian derivative current source of $I(t) = 2(t - t_0)\exp(-(t - t_0)^2/\tau^2)$, with $\tau = 3 \times 10^{-11}$ s and and $t_0 = 4\tau$. The time step is chosen as 5×10^{-13} s determined by the

accuracy condition and the consideration of series expansion terms for the convergence of the matrix exponential components. The required number of the convergence of the matrix exponential components in this case is 4. In contrast, the time step of the central-difference based TDFEM solution is 3×10^{-15} s.

Fig. 6.6 shows the accuracy of the proposed method in this example. Again, an excellent agreement is observed.

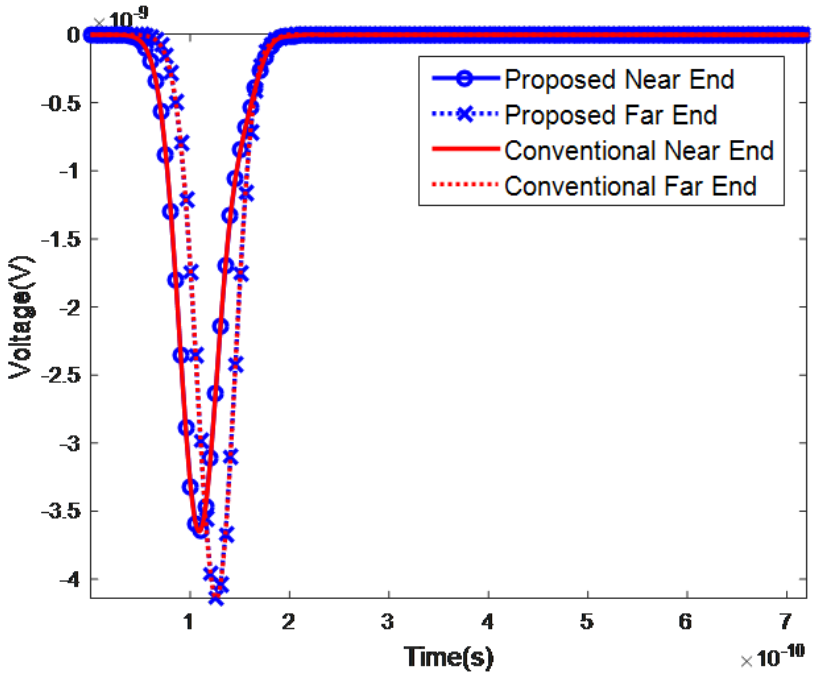


Fig. 6.6. Accuracy validation of the proposed algorithm with the structure of Fig. 5.4.

Then, the dielectric region is attached along y-direction to further enlarge the unknown size of the problems. The structure result in a larger number of unknowns up to 360,028. The simulation results and associated parameters with these extensions are listed in table 6.2.

Fig 6.7 shows the time comparison between higher-order mode identification time of the proposed algorithm and LU factorization time of conventional central-difference based TDFEM which is performed before the time marching. Fig 6.8 shows the

Table 6.2.
Simulation results along dielctric block extentions of the structure

Multiple of extension dielectric blocks	$\times 100$	$\times 500$	$\times 800$	$\times 1000$
Number of unknowns	36,028	180,028	288,028	360,028
Maximum input freq.	34GHz	34GHz	34GHz	34GHz
Time step determined by accuracy	5×10^{-13}	5×10^{-13}	5×10^{-13}	5×10^{-13}
# of higher modes/total # of eigenmodes	130/72056	130/360056	130/576056	130/720056
Elapsed time for identifying higher modes (s)	72	2.5819×10^2	5.6742×10^2	7.8582×10^2
Elapsed Time for time marching with structure-aware $\mathbf{T}$-solver	1.9689×10^4	3.6839×10^4	4.3979×10^4	5.0659×10^4
Time step required for central diff. based reference	3×10^{-15}	3×10^{-15}	3×10^{-15}	3×10^{-15}
Elapsed time for LU factorization (s)	26.9699	2.1482×10^3	8.2368×10^3	1.5825×10^4
Elapsed time for the time marching (s)	1.3068×10^4	4.2406×10^4	6.1992×10^4	7.3807×10^4

marching time comparison and Fig. 6.9 shows the total elapsed time that is the sum of pre-marching time and marching time.

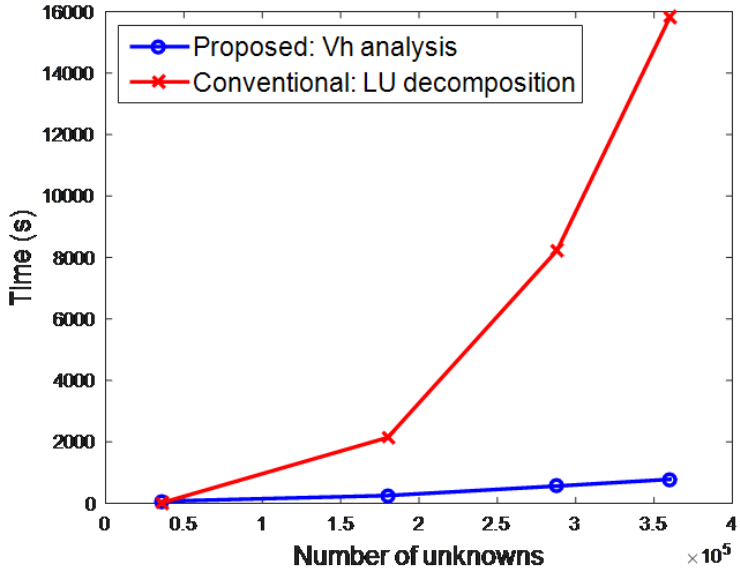


Fig. 6.7. Pre-marching time comparison.

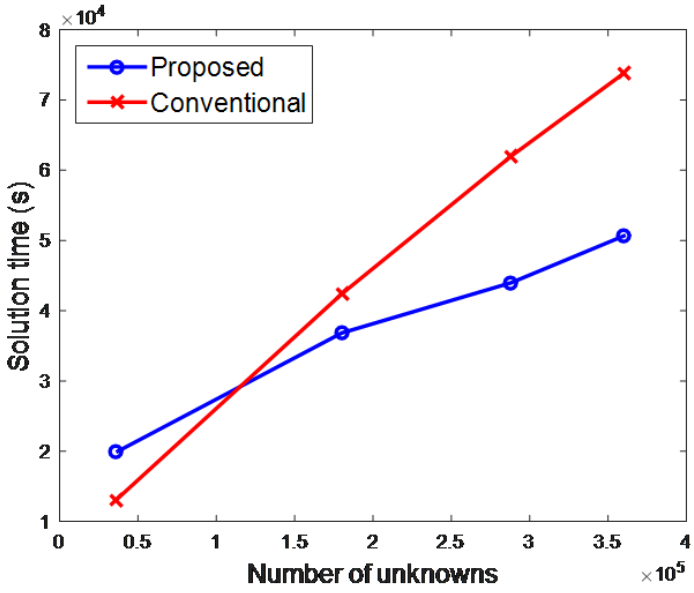


Fig. 6.8. Marching time comparison.

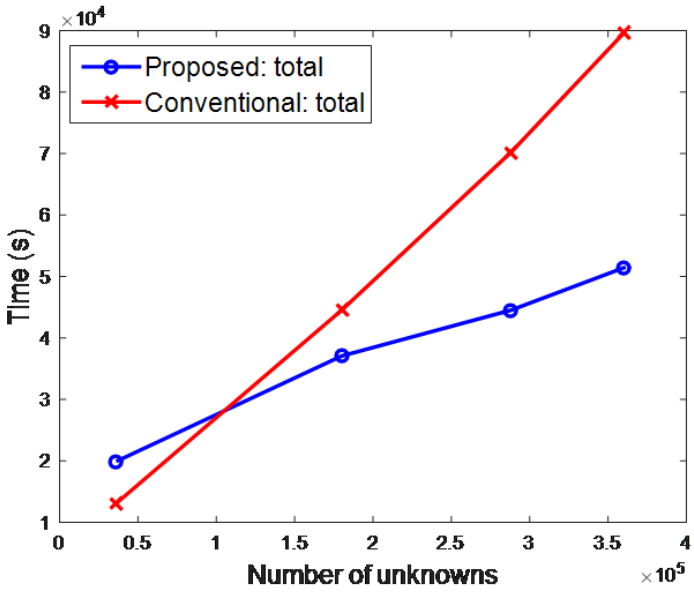


Fig. 6.9. Total elapsed time comparison.

6.4 Conclusions

In this chapter, a faster structure-aware TDFEM solver is developed which retains its original merit of being valid for general full-wave applications with support of matrix exponential framework. It is efficient in the sense that it turns a matrix solution into a simple scaling, and meanwhile allows for the use of a large time step solely determined by accuracy. The proof of the concept of this work has been completed by the successful simulation of several examples.

7. SYMMETRIC POSITIVE-DEFINITE REPRESENTATION OF FREQUENCY-DOMAIN FINITE-ELEMENT SYSTEM MATRIX FOR EFFICIENT ELECTROMAGNETIC ANALYSIS

7.1 Introduction

Frequency-domain electromagnetic analysis has been of critical importance to interpret and understand the characteristics of the electromagnetic structures. In a frequency domain method, the analysis requires simulating multiple or many frequency points to complete the analysis, thus faster and more efficient simulation algorithms are required. However, some properties of the system matrix prevent us from making the analysis efficient. The system matrix resulting from a frequency-domain finite-element method (FEM) based analysis is indefinite, containing both negative and positive eigenvalues. Its condition number is also generally large since the magnitude of the smallest eigenvalue can be close to zero. These properties have made an efficient solution of the FEM system of equations difficult in both iterative and direct solutions. Although various preconditioning techniques have been developed to change the spectrum of the FEM system matrix, the indefinite nature of the FEM operator has not been changed.

In this work, we propose to build a symmetric positive definite representation of the FEM operator by deducting the non-positive definite component from the system matrix. This is similar to removing the unstable mode contribution in a time domain method like we propose in previous chapters. However, different from the treatment in time domain, the non-positive definite contribution in frequency-domain representation should be kept for completing the frequency domain solution. To do so, in the second step, with negligible cost, we add the contribution from the non-

positive definite component back to obtain the true solution. The positive-definite representation after removing non-positive definite modes has a spectral radius less than 1. Its condition number can also be controlled to any desired value. Also, we transform the original eigenvalue system to obtain a reduced number of non-positive definite modes set. As a result, the resultant iterative solution can converge in a small number of iterations [34]. Such a representation also benefits the development of fast direct solvers. In addition, the computational overhead of the proposed method is shown to be modest. Numerical experiments associated with several different types of frequency domain examples have demonstrated the accuracy and efficiency of the proposed methods.

In this chapter, we present the proposed frequency-domain methods for analyzing lossless problems. We also provide examples to validate the accuracy and efficiency of the proposed methods. The contents of this chapter have been extracted and revised from the following publication: Woochan Lee and Dan Jiao, "Symmetric Positive-Definite Representation of Frequency-Domain Finite-Element System Matrix for Efficient Electromagnetic Analysis," 2016 IEEE Antennas and Propagation Society International Symposium (APSURSI), 2016.

7.2 Proposed Method

Consider a general lossless problem, a frequency-domain FEM-based analysis of a general problem results in the following linear system of the equations

$$(-\omega^2\mathbf{T}+\mathbf{S})u=b, \tag{7.1}$$

where ω is an angular frequency, u is the field solution vector, $\mathbf{T}$ is a mass matrix, and $\mathbf{S}$ is a stiffness matrix. The $\mathbf{T}$ and $\mathbf{S}$ are assembled from their elemental contributions as the following:

$$\begin{aligned}\mathbf{T}^e &= \varepsilon\left\langle \mathbf{N}_i, \mathbf{N}_j\right\rangle \\ \mathbf{S}^e &= \mu^{-1}\left\langle \nabla\times\mathbf{N}_i, \nabla\times\mathbf{N}_j\right\rangle\end{aligned}, \tag{7.2}$$

where μ is permeability, $\mathbf{N}$ is the vector basis function employed to expand electric field $\mathbf{E}$ in each element, and $\langle \cdot, \cdot \rangle$ denotes an inner product. The solution of (7.1) is governed by the eigenvalue problem of

$$\mathbf{S}x = \lambda \mathbf{T}x, \tag{7.3}$$

whose eigenvalues λ are real and non-negative. The smallest one is $\lambda_{\min} = 0$, and the largest one, $\lambda_{\max}$, is proportional to $(\pi^2 c^2)/{s_{\min}}^2$, where $s_{\min}$ denotes the smallest space step, and c is the speed of light. In a general full-wave analysis, the space step is chosen as no greater than half of the wavelength. This results in a relative relationship between ω^2 and the eigenvalues of (7.3) as $\lambda_{\min} < \omega^2 < \lambda_{\max}$. Therefore, some eigenvalues are less than ω^2, while the rest are greater than ω^2. Since the eigenvalues of (7.1) are $(\lambda - \omega^2)$, (7.1) is indefinite. The condition number of (7.1) can also be large since the smallest magnitude of $(\lambda - \omega^2)$ can be close to 0.

More important, for example, if we take the minimum space step as 1/10 of the wavelength, the largest eigenvalue $\lambda_{\max}$ is proportional to $(\pi^2 c^2)/{s_{\min}}^2 \sim 25\omega^2$, thus the region between $\omega^2 \leftrightarrow (\lambda_{\max} \sim 25\omega^2)$ has more eigenvalues than those of the region $\lambda_{\min} \leftrightarrow \omega^2$. Thus, if the truncation of the region is possible to alleviate indefinite problem, many number of truncated mode is still the problem because the computational overhead for removal them is still high. As a result, the method for reducing the number of truncated mode is required.

In this section, we present a transformed system that is positive definite, an iterative method for eigenanalysis, and the solution for analyzing general lossless problems.

7.2.1 Transformed System

To build a positive-definite representation of (7.1), we first rewrite (7.1) as

$$(-\omega^2 \mathbf{T} + \mathbf{S} + \omega_0^2 \mathbf{T} - \omega_0^2 \mathbf{T})u = b \tag{7.4}$$

where ω_o^2 is chosen to be larger than λ_{max} . Practically, the ω_o^2 can be obtained from eigenanalysis of the system in (7.3), and the cost for its acquisition is not high because

the analysis is just associated with the largest few eigenvalues. Then, (7.4) can then be rewritten as

$$(\mathbf{B} - \mathbf{A})u = b \tag{7.5}$$

with $\mathbf{A} = -\mathbf{S} + \omega_o^2\mathbf{T}, \mathbf{B} = (\omega_o^2 - \omega^2)\mathbf{T}$. The above now is governed by a new eigenvalue problem of

$$\mathbf{A}x = \lambda\mathbf{B}x \tag{7.6}$$

whose eigenvalues can be written as

$$\lambda_{new} = \frac{\omega_o^2 - \lambda}{\omega_o^2 - \omega^2}, \tag{7.7}$$

which is always positive as $\lambda < \omega_0^2$. More important, when $\lambda > \omega^2$, $0 < \lambda_{new} < 1$, i.e., the original largest eigenvalues of (7.3) now become the smallest eigenvalues; and vice versa as when $\lambda < \omega^2$, $\lambda_{new} > 1$. Also, as stated above, the region $\lambda_{new} > 1$ has less number of eigenvalues than the region $0 < \lambda_{new} < 1$ as they are flipped. The summary of new eigenvalue system is illustrated in Fig. 7.1.

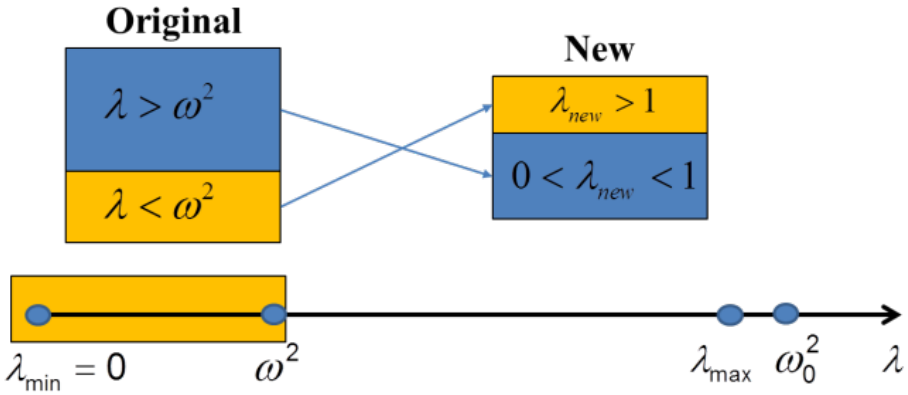


Fig. 7.1. Transformed eigenvalue system.

If we deduct the eigenmodes whose $\lambda_{new} > 1$ from $\mathbf{B}^{-1}\mathbf{A}$, then the remaining eigenvalues of $\mathbf{B}^{-1}\mathbf{A}$ would satisfy $0 < \lambda_{new} < 1$. Hence, the eigenvalues of (7.5), which is $1 - \lambda_{new}$, will be positive and no greater than 1. Thus, (7.5) becomes a

positive-definite system. Furthermore, its spectral radius is less than 1. Along this line of thought, we update (7.5) to the following new system of equations, which is

$$\left[\mathbf{B} - \mathbf{A}\left(\mathbf{I} - \mathbf{V}_n\mathbf{V}_n^T\mathbf{B}\right)\right] u_p = b, \tag{7.8}$$

where $\mathbf{V}_n$ denotes the eigenvectors of (7.6) whose eigenvalues are greater than 1. Notice the above is a symmetric system as $\mathbf{A}\mathbf{V}_n = \mathbf{B}\mathbf{V}_n\Lambda_n$. Let $\mathbf{V}_p$ and Λ_p be, respectively, the eigenvector matrix, and the diagonal eigenvalue matrix of the rest of the eigenvalues. We can analytically derive the property of the updated system matrix of (7.8), which is

$$\mathbf{Y}_{new} = \mathbf{I} - \mathbf{Y}_0 = \left[\mathbf{I} - \mathbf{B}^{-1}\mathbf{A}\left(\mathbf{I} - \mathbf{V}_n\mathbf{V}_n^T\mathbf{B}\right)\right], \tag{7.9}$$

where $\mathbf{Y}_0 = \mathbf{B}^{-1}\mathbf{A}\left(I - \mathbf{V}_n\mathbf{V}_n^T\mathbf{B}\right)$. Then, we can write

$$\mathbf{B}^{-1}\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1} = \mathbf{V}\Lambda\mathbf{V}^T\mathbf{B} = \mathbf{V}_n\Lambda_n{\mathbf{V}_n}^T\mathbf{B} + \mathbf{V}_p\Lambda_p{\mathbf{V}_p}^T\mathbf{B}. \tag{7.10}$$

Then, we have (7.11) by utilizing the property of $\mathbf{V}^T\mathbf{B}\mathbf{V} = \mathbf{I}, \mathbf{V}^T\mathbf{A}\mathbf{V} = \mathbf{\Lambda}$;

$$\mathbf{B}^{-1}\mathbf{A}\mathbf{V}_n{\mathbf{V}_n}^T\mathbf{B} = \mathbf{V}_n\Lambda_n{\mathbf{V}_n}^T\mathbf{B}. \tag{7.11}$$

By subtracting (7.11) from (7.10), we have

$$\left[\mathbf{B}^{-1}\mathbf{A}\left(\mathbf{I} - \mathbf{V}_n\mathbf{V}_n^T\mathbf{B}\right)\right] = \mathbf{V}_p\Lambda_p{\mathbf{V}_p}^T\mathbf{B}. \tag{7.12}$$

Then from (7.9) and (7.12), the following form is induced.

$$\mathbf{Y}_{new}\mathbf{V}_p = \mathbf{V}_p(\mathbf{I} - \mathbf{\Lambda}_p). \tag{7.13}$$

Hence, the eigenvalues of $\mathbf{Y}_{new}$ are the entries of diagonal matrix $(\mathbf{I} - \mathbf{\Lambda}_p)$. Since it consists of all the eigenvalues of (7.6) that are less than 1, the new system (7.8) is clearly positive definite. Its spectral radius is less than 1 as well.

In addition, its condition number can be controlled to any desired constant by the choice of $\mathbf{V}_n$. This is because the ratio of the largest to the smallest eigenvalue or the condition number of $\mathbf{Y}_{new}$ is as the following:

$$cond(\mathbf{Y}_{new}) \sim \frac{|\lambda_{\text{largest}}|}{|\lambda_{\text{smallest}}|} \sim \frac{|\lambda_{\max} - \omega^2|}{|\lambda_r - \omega^2|}, \tag{7.14}$$

where λ_r along the eigenvalue axis of the original system corresponds the largest eigenvalue of remaining $\lambda_{new} = (\omega_o^2 - \lambda) / (\omega_o^2 - \omega^2)$ after deducting $\mathbf{V}_n$, and $\lambda_{\max} = \alpha\omega^2$, where α is a constant determined by space discretization. To be specific, the eigenvalues of $\mathbf{Y}_{new}$ are $1 - \lambda_{new} = (\lambda - \omega^2) / (\omega_o^2 - \omega^2)$, thus the largest eigenvalue corresponds to $\lambda_{\max} - \omega^2$ while the smallest corresponds to $\lambda_r - \omega^2$. The position of λ_r near ω^2 in the original system is depicted in Fig. 7.2. Hence, by choosing which

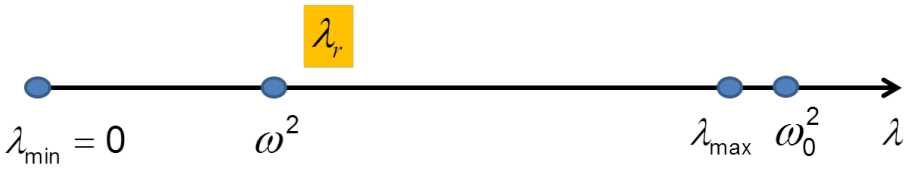


Fig. 7.2. The controllability of condition number.

set of $\mathbf{V}_n$ to obtain from (7.6) thus λ_r, the condition number can be controlled.

To summarize, in the proposed algorithm, we change the original indefinite system (7.1) to a new positive definite system (7.8) to solve, and then obtain the solution of the original problem from (7.9). The only computational overhead is to find $\mathbf{V}_n$. Since $\mathbf{V}_n$ of (7.6) is the same as the eigenvectors of the original eigenvalue problem of (7.3), whose eigenvalues are smaller than ω^2, (7.8) can be efficiently solved by an iterative solver such as GMRES in a small number of iterations, and its convergence is guaranteed. Meanwhile, (7.8) can also be directly solved as $u = (\mathbf{I}+\mathbf{Y}_0+...+\mathbf{Y}_0^k)\mathbf{B}^{-1}b$, since the spectral radius of $\mathbf{Y}_0$ is less than 1 as well. Furthermore, $\mathbf{B}$'s solution in (7.5) is the same as $\mathbf{T}$'s solution, and hence it can be efficiently computed.

Generalized minimal residual method (GMRES) is a well-known iterative method and a natural choice for finding the solutions of non-symmetric system of equations [35]. For a positive definite matrix $\mathbf{A}$, the convergence of the GMRES solver is guaranteed. In contrast, if the matrix is not positive definite, GMRES may stagnate and the convergence is not guaranteed. The convergence rate is strongly affected by the eigenvalue distribution and condition number [34, 36, 37]. The matrix $\mathbf{A}$ in the proposed method is symmetric (Hermitian) positive definite thus normal. In

this symmetric positive definite case, the convergence rate is bounded by condition number of the matrix as seen in (7.15) [35, 38]

$$\|r_n\| \leq \left(\frac{cond(A)^2 - 1}{cond(A)^2}\right)^{n/2} \|r_0\|. \tag{7.15}$$

And we can notice that smaller condition number leads to faster convergence. Also, the nullspace is clustered away from the origin and share the same eigenvalue in common, then there will also be fast convergence. Thus, regardless of the existence of nullspace in the matrix we want to solve, the performance of the GMRES will not be affected. Therefore, by the proposed transformed system and truncation of non-positive contribution, the convergence of GMRES can be accelerated.

It is worth mentioning that $\mathbf{S}$ has a nullspace whose size can grow with N. Since the convergence performance of GMRES is not affected by the nullspace as analyzed in the above, the nullspace can be bypassed in the eigenvalue analysis, where we compute a Krylov subspace that is orthogonal to the nullspace. Therefore, the number of non-positive definite modes is further reduced, so is the overhead for deducting the non-positive modes. First, in the eigenanalsysis of (7.6) using Implicitly Restarted Arnoldi (IRA) algorithm, this can be done by setting shifts as undesired eigenvalues in the standard QR step in IRA. In the QR process to approximate eigenvalues, the shifts are suppressed thus bypassing the computation to obtain the nullspace. Specifically, we can use a shift ξ as the largest eigenvalue of (7.6), which is also the zero eigenvalue of (7.3). The ξ is analytically known as $(\omega_0^2)\ /\ (\omega_0^2 - \omega^2)$, which is λ_{new} from the setting of $\lambda = 0$ in (7.7).

After solving u_p from the positive definite system (7.8), we can add back the contribution of the non-positive definite part to complete the total solution as

$$u = u_p + \mathbf{V}_n(\Lambda_n - \omega^2)^{-1}{\mathbf{V}_n}^T b. \tag{7.16}$$

The computation cost of $\mathbf{V}_n(\Lambda_n - \omega^2)^{-1}{\mathbf{V}_n}^T b$ part in (7.16) is negligible because the $\mathbf{V}_n$ set is already known and the dimension of the part is much smaller than the original dimension N.

7.2.2 Absorbing Boundary Condition Imposition

Again, a frequency-domain FEM-based analysis of waveguide discontinuities with absorbing boundary conditions results in the following matrix equation $\mathbf{A}(\omega)u(\omega) = b(\omega)$ where ω is an angular frequency, u is the frequency-domain field solution vector, and

$$\begin{aligned} \mathbf{A}(\omega) &= -\omega^2\mathbf{T} + \mathbf{S} + \frac{\gamma}{\mu}\mathbf{Q}_1 + \frac{\gamma}{\mu}\mathbf{Q}_2 \\ b(\omega) &= < \mathbf{N}_i, \mathbf{N}_j >_{S_1} u_{inc} \end{aligned} \tag{7.17}$$

in which $\mathbf{T}$ is the mass matrix, $\mathbf{S}$ is the stiffness matrix, $\mathbf{Q}$ is the matrix associated with absorbing boundary condition. The $\mathbf{Q}$ is assembled from its elemental contributions as the following:

$$\begin{aligned} \mathbf{Q}_1^e &= \langle \widehat{n} \times \mathbf{N}_i, \widehat{n} \times \mathbf{N}_j \rangle_{S_1} \\ \mathbf{Q}_2^e &= \langle \widehat{n} \times \mathbf{N}_i, \widehat{n} \times \mathbf{N}_j \rangle_{S_2} \end{aligned} . \tag{7.18}$$

In addition, for TE_{10} dominant case, $\gamma = jk_{z_{10}}$ and $k_{z_{10}} = \sqrt{k_0^2 - \left(\frac{\pi}{a}\right)^2}$ where a is the width of the structure.

The problem of the non-positive definite mode exclusion when applied to (7.1) is that these modes involve absorbing boundary condition matrix $\mathbf{Q}$ which contains angular frequency term, and thereby a need for performing a double dimensional eigenvalue analysis.

To alleviate the aforementioned difficulty, we propose to separate the boundary related matrix from the rest. We partition the entire set of unknowns u into u_i that is inside the computational domain, and u_b that is on the boundaries. Thus, the FEM matrices in (7.17) can be rewritten as

$$\begin{bmatrix} \mathbf{A}_{bb} & \mathbf{A}_{bi} \\ \mathbf{A}_{ib} & \mathbf{A}_{ii} \end{bmatrix} \begin{bmatrix} u_b \\ u_i \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} \tag{7.19}$$

where

$$\begin{aligned} \mathbf{A}_{ii} &= -\omega^2\mathbf{T}_{ii} + \mathbf{S}_{ii} \\ \mathbf{A}_{ib} &= -\omega^2\mathbf{T}_{ib} + \mathbf{S}_{ib} + \tfrac{\gamma}{\mu}\mathbf{Q}_{1ib} + \tfrac{\gamma}{\mu}\mathbf{Q}_{2ib} \\ \mathbf{A}_{ii} &= -\omega^2\mathbf{T}_{bi} + \mathbf{S}_{bi} + \tfrac{\gamma}{\mu}\mathbf{Q}_{1bi} + \tfrac{\gamma}{\mu}\mathbf{Q}_{2bi} \\ \mathbf{A}_{bb} &= -\omega^2\mathbf{T}_{bb} + \mathbf{S}_{bb} \end{aligned} . \tag{7.20}$$

Then, eliminating the first row in (7.18) yields

$$(\mathbf{A}_{bb} - \mathbf{A}_{bi}\mathbf{A}_{ii}^{-1}\mathbf{A}_{ib})u_b = b. \tag{7.21}$$

Here, $\mathbf{A}_{ii}$ is not associated with boundary matrix $\mathbf{Q}$ thus the non-positive definite mode exclusion approach can be directly applied for the solution of $\mathbf{A}_{ii}\widetilde{u} = \mathbf{A}_{ib}$ in (7.21). Thus, if we have the non-positive definite modes set $\mathbf{V}_n$, $\mathbf{A}_{ii}\widetilde{u} = \mathbf{A}_{ib}$ turns into

$$\mathbf{A}'_{ii}\widetilde{u}_p = -\omega^2(\mathbf{T}_{ii} + \frac{1}{-\omega^2}\mathbf{S}_{ii}(\mathbf{I} - \mathbf{V}_n\mathbf{V}_n^T\mathbf{T}_{ii}))\widetilde{u}_p = \mathbf{A}_{ib}. \tag{7.22}$$

Then, the non-positive definite modes contribution $\widetilde{u}_n$ is added back to obtain complete the solution $\widetilde{u}$.

After getting $\widetilde{u}$, (7.21) for u_b can be solved as a small problem whose dimension is just the number of boundary edges. Once u_b is found, u_i is recovered simply from $u_i = -\mathbf{A}_{ii}^{-1}\mathbf{A}_{ib}u_b$, which is identical to solving $\mathbf{A}_{ii}u_i = -\mathbf{A}_{ib}u_b$, thus the proposed non-positive definite mode exclusion approach can be applied again.

7.2.3 Finding Non-Positive Definite Modes $\mathbf{V}_n$

In this work, we use the implicitly restarted Arnoldi algorithm to find $\mathbf{V}_n$ efficiently since the modes being sought for have the largest eigenvalues. The implicitly restarted Arnoldi method is a method for capturing wanted eigenvalue information from shrinked m-step Krylov subspace method rather than full dimension eigenvalue analysis [39, 40]. For finding k largest eigenvalues and their eigenvectors, the computation of this algorithm is mainly sparse matrix-vector multiplications, and the orthogonalization of the obtained vectors. The overall computational complexity is $O(k^2N)$, thus it is more efficient than a traditional full eigenvalue analysis whose complexity is $O(N^3)$. In addition, by setting the eigenvalues corresponding to the nullspace as unwanted eigenvalues, nullspace originated eigenvalues can be filtered out.

Overall, as we transform the original system to have a smaller number of $\mathbf{V}_n$, the computational overhead of the $O(k^2N)$ computation for finding $\mathbf{V}_n$ is not a bottleneck.

Moreover, since $\mathbf{V}_n$ is frequency independent, after it is found, it can be reused at different frequencies.

7.3 Numerical Results

In this section, we demonstrate the accuracy and efficiency of the proposed scheme with lossless case examples. The conventional method is GMRES which is used as Matlabs built-in function '*gmres*.' The computing machine used here has an Intel i5 5300U 2.30 GHz processor, unless specified specifically.

All of these examples involve full-wave analysis in which the minimum discretization is approximately 1/10 of the wavelength of interested frequency and the nullspace modes effect is limited. Also, in these cases, the number of non-positive definite modes is much smaller than the whole system size, thus the overhead for analyzing and removing such modes is negligible.

The new method would also suffer from an increased iteration number when N increases, if we do not truncate $\mathbf{V}_n$ and do not control the condition number. For larger N cases, the smallest eigenvalue along eigenvalue axis (as shown in Fig. 7.1) greater than ω^2 will become closer to ω^2, although the largest eigenvalues does not change, then this will increase the condition number. However, if we remove $\mathbf{V}_n$ based on the criterion of keeping the condition number to be a constant, which means more truncated modes as the growth of N, the nearly constant iteration number for convergence can be achieved.

7.3.1 Waveguide with Absorbing Boundary Condition

We first demonstrate the accuracy and efficiency of the proposed method with an air-filled waveguide loaded with an internal block of relative permittivity of 6.0 as Fig. 7.3. The length, width and height of the waveguide structure are 25 mm (x-direction), 20 mm (y-direction), and 10 mm (z-direction), respectively. Also, the size parameters of inside dielectric block are 5 mm, 10 mm and 5 mm, respectively.

The whole structure is discretized with a uniform mesh size 2.5 mm. The waveguide is operating at a frequency only with the dominant mode TE_{10}.

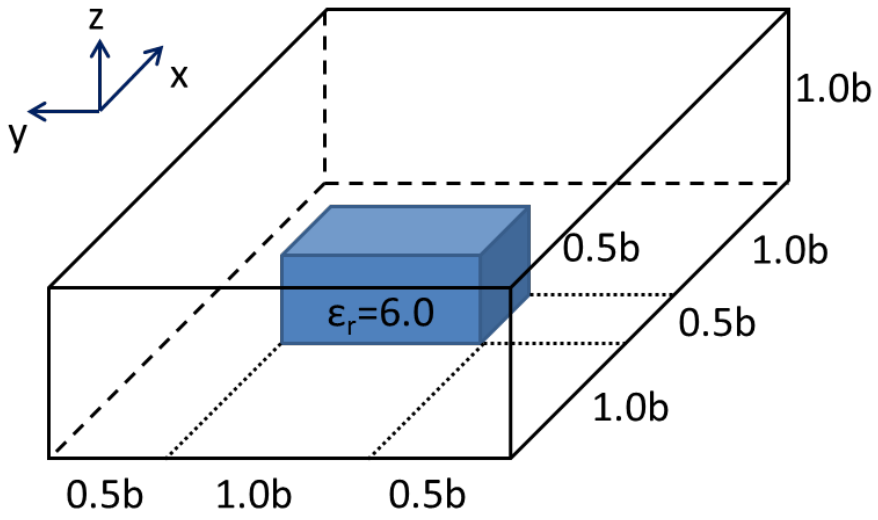


Fig. 7.3. Dielectric loaded rectangular waveguide.

Fig. 7.4 shows the reflection coefficient $|S_{11}|$ of a waveguide with a dielectric block simulated by using the proposed algorithm. Excellent agreement with the reference result from a traditional solver is observed.

The length (x-direction) of the waveguide structure is then extended to build large unknown cases and structures to study the performance of the proposed method as a function of N. Also, the computation of u_i is omitted here because reflection coefficient calculation only requires the field solution on the front surface.

When the number of unknowns N is 14,652, the number of non-positive definite modes $\mathbf{V}_n$ is identified as 1,139 over total 14,548 modes, whose absolute magnitude of the eigenvalues is over 0.95 in the transformed system. The corresponding eigenvalues and parameters along the eigenvalue axis of the original indefinite problem are illustrated in Fig. 7.5. The yellow marked region is removed in the transformed system and the highlighted red region is the remained eigenvalue region. The ω is set to be 8×10^{10} rad/s and ${w_0}^2$ is chosen as 4.5824×10^{23}. Also, the largest eigen-

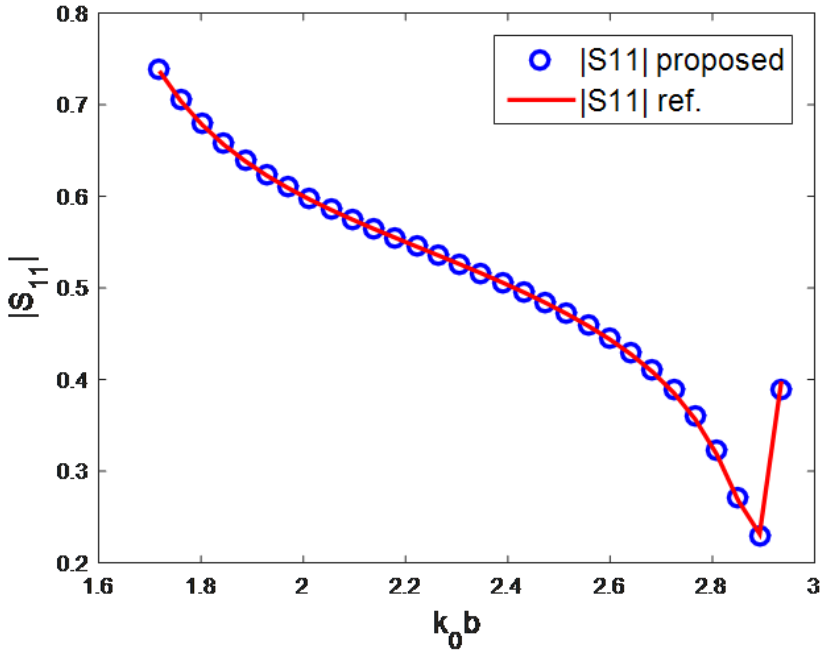


Fig. 7.4. Reflection coefficient of dielectric loaded waveguide (b=10 mm).

value λ_{max} is 4.4062×10^{23}, and the remained smallest eigenvalue λ_r is 2.9002×10^{22}. There exist 1139 eigenvalues between λ_r and the first non-zero eigenvalue λ_{min} which is 1.6153×10^{19}, and this yellow band is removed in the transformed system. The condition number defined by (7.14) of the proposed deducting method is 19.2114. In contrast, the condition number of original indefinite matrix is 1.2645×10^5.

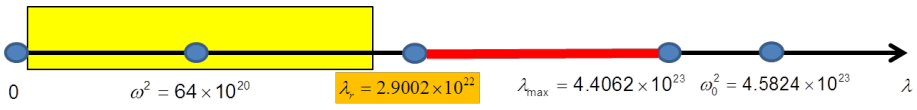


Fig. 7.5. Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a dielectric-loaded waveguide with 14,652 unknowns.

The proposed method is shown to have a constant number of iterations of ∼26 to reach an accuracy of 1×10^{-4}. In contrast, using the same GMRES solver and with the same error tolerance, the original indefinite system is shown to have a large iteration number. This number also increases with N, from 98 to 626 when N reaches

14,652. Fig. 7.6 plots the total CPU time comparison at $\omega = 8 \times 10^{10}$ rad/s for simulating the same number of right hand sides involved in the waveguide analysis. In 14,652 unknown case, the proposed method identifies 1,141 over 14,548 modes as $\mathbf{V}_n$, and shows an elapsed time of 1.0867×10^3 s with 26 iterations to simulate one frequency. In contrast, the conventional GMRES based method with original indefinite system yields an elapsed time of 1.3696×10^4 s with 626 iterations. Thus, a speed-up of 12.6032 is observed. The efficiency of the proposed new method is clearly demonstrated.

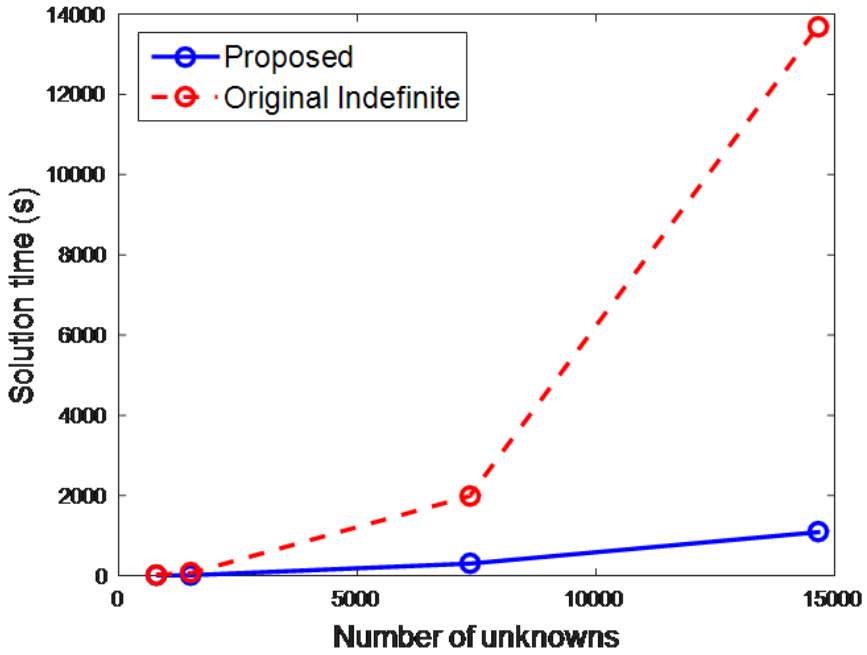


Fig. 7.6. Solution time comparison.

7.3.2 Demonstration of Accuracy and Efficiency

Millimeter-level inhomogeneous lossless waveguide

We next consider a mm-level parallel-plate with vertical inhomogeneous layer. The length, width, and height of the structure are 70 mm, 30 mm, and 20 mm respectively

as shown in Fig. 7.7. Along with the width direction, the thickness of each dielectric vertical layer is 10 mm. In addition, the discretization along height is 5 mm, 2.5 mm, 2.5 mm, 2.5 mm, 2.5 mm and 5 mm (2.5 mm is the minimum space step). The number of unknowns in this example is 1570, and the number of elements along x, y, z-axis is 14, 6, 6 respectively. The current source is launched from bottom PEC plane to top PEC plane.

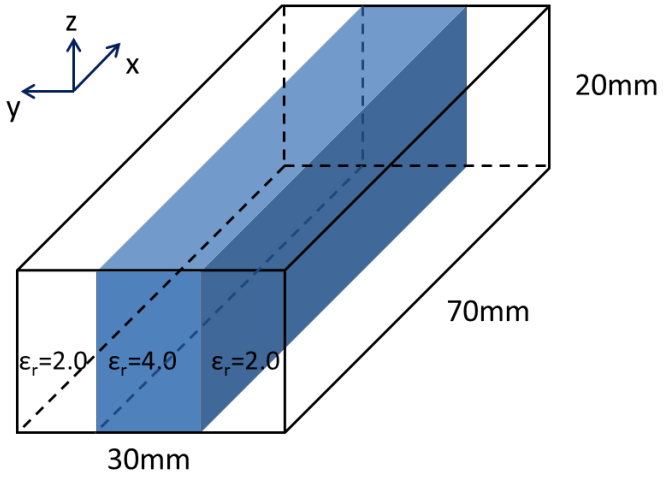


Fig. 7.7. Structure of parallel-plate waveguide with inhomogeneous vertical layer.

The number of non-positive definite modes $\mathbf{V}_n$ is identified as 78 over total 1,570 modes, whose absolute magnitude of the eigenvalues is over 0.95 in the transformed system. The corresponding eigenvalues and parameters along the eigenvalue axis of the original indefinite problem are illustrated in Fig. 7.8. The yellow marked region is removed in the transformed system and the highlighted red region is the remained eigenvalue region. The ω is set to be 6×10^9 rad/s and ${\omega_0}^2$ is chosen as 1.0663×10^{23}. Also, the largest eigenvalue λ_{max} is 1.0253×10^{23}, and the remained smallest eigenvalue λ_r is 5.3975×10^{21}. There exist 78 eigenvalues between λ_r and the first non-zero eigenvalue λ_{min} which is 2.7050×10^{20}, and this yellow band is removed in the transformed system. The condition number defined by (7.14) of the proposed

deducting method is 19.1158. In contrast, the condition number of original indefinite case matrix is 4.3410×10^2.

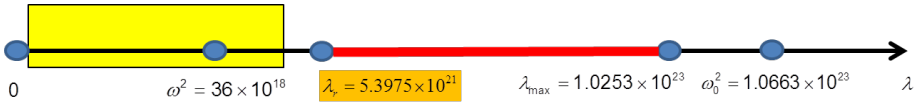


Fig. 7.8. Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a parallel-plate waveguide with vertical inhomogeneous layer.

The accuracy comparison result shown in Fig. 7.9 shows an excellent agreement. In addition, the proposed method shows 30 iterations for the iterative solver to converge with 1×10^{-5} tolerance. In contrast, the conventional GMRES requires 211 to 336 iterations to converge to achieve the same tolerance as shown in Fig. 7.10.

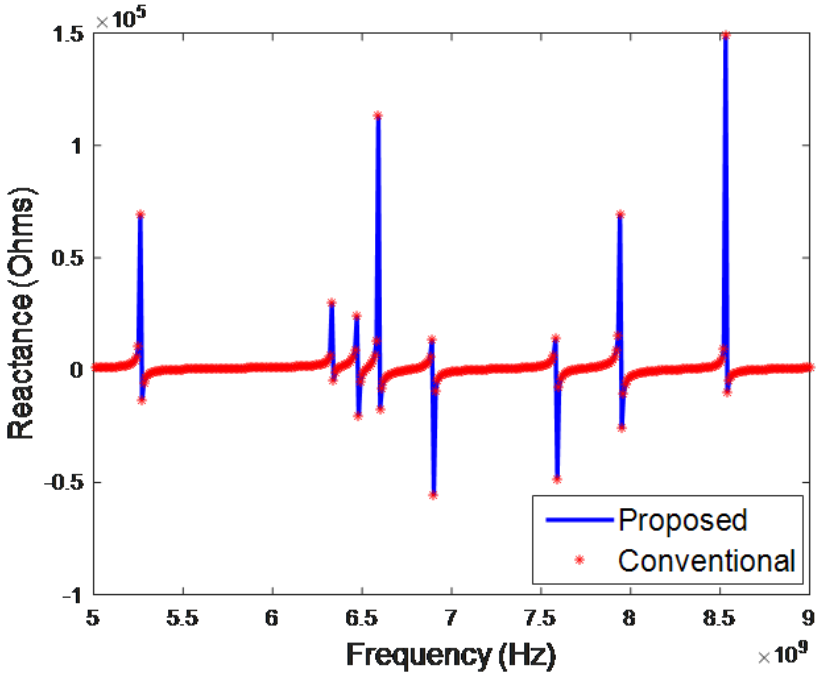


Fig. 7.9. Input reactance versus frequency for a parallel-plate waveguide.

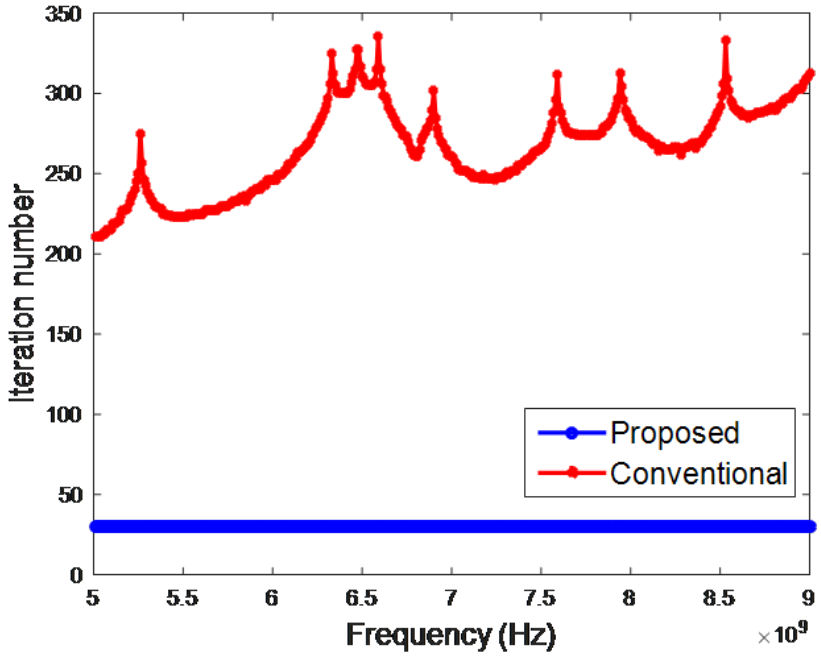


Fig. 7.10. Iteration number in a parallel plate example for convergence comparison.

The total frequency sweep CPU time of the proposed method is 1.4723×10^2 s, which includes 400 frequency points (frequency gap: 0.01 GHz) simulation while the conventional GMRES solver takes 2.6907×10^2 s to simulate the same example. Thus, the speed-up of frequency sweep time with this setting is 1.8275.

Cavity-backed patch antenna

The third example is a cavity-backed microstrip patch antenna shown in Fig. 7.11. The conductors including the patch and the ground plane are treated as perfect conductors. The overall structure size is 15 cm $\times$ 10.2 cm $\times$ 3.08779 cm including 3 cm air regions at the top of the patch antenna structure. The size of the patch immersed in 7.5 cm $\times$ 5.1 cm cavity ($\varepsilon_r = 2.17$) is 5 cm $\times$ 3.4cm. The number of element along x, y, z-axis is 12, 12, 3 respectively. In addition, the minimum discretization length is 0.08779 cm along z-direction. The number of unknowns of

this example is 1007. The current probe is launched from the bottom of the cavity to the patch, and the location of the red-arrow source shown in Fig. 11 is 1.25 cm (x-direction), 0.85 cm (y-direction) off center of the patch.

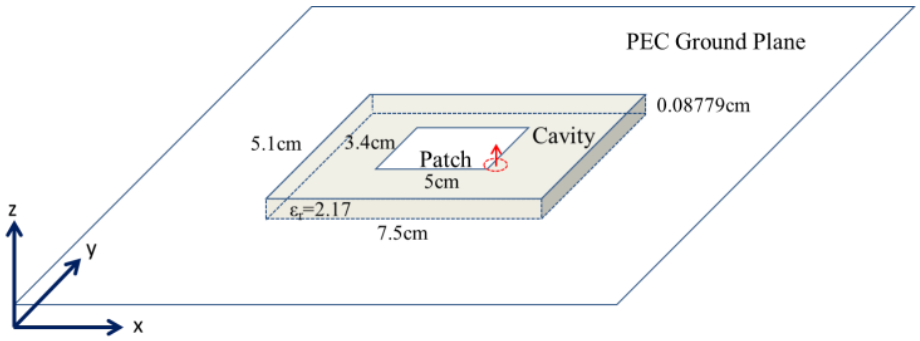


Fig. 7.11. The structure of a cavity-backed patch antenna.

The number of non-positive definite modes $\mathbf{V}_n$ is identified as 49 over total 1,007 modes, whose absolute magnitude of the eigenvalues is over 0.95 in the transformed system. The corresponding eigenvalues and parameters along the eigenvalue axis of the original indefinite problem are illustrated in Fig. 7.12. The yellow marked region is removed in the transformed system and the highlighted red region is the remained eigenvalue region. The ω is set to be 3×10^9 rad/s and ω_0^2 is chosen as 3.6763×10^{22}. Also, the largest eigenvalue $\lambda_{\max}$ is 3.5349×10^{22}, and the remained smallest eigenvalue λ_r is 1.9129×10^{21}. There exist 49 eigenvalues between λ_r and the non-zero eigenvalue $\lambda_{\min}$ which is 1.2294×10^{20}, and this yellow band is removed in the transformed system. The condition number defined by (7.14) of the proposed deducting method is 18.5623. In contrast, the condition number of original indefinite matrix is 3.0796×10^2.

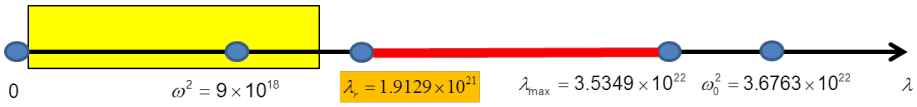


Fig. 7.12. Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a cavity-backed patch antenna.

Fig. 7.13 shows an excellent agreement between the proposed method and the reference solution. Also, the proposed method shows 50 iterations for the iterative solver to converge with 1×10^{-5} tolerance while the conventional GMRES requires 289 to 632 iterations to converge to achieve the same tolerance. Fig. 7.14 demonstrates the iteration number comparison between the proposed method and the conventional GMRES. The total frequency sweep CPU time of 120 points (frequency gap = 0.05 GHz) of the proposed method is 24.8978 s, while the conventional GMRES solver takes 1.8159×10^2 s. Thus, the speed-up of frequency sweep time with current setting is 7.2934.

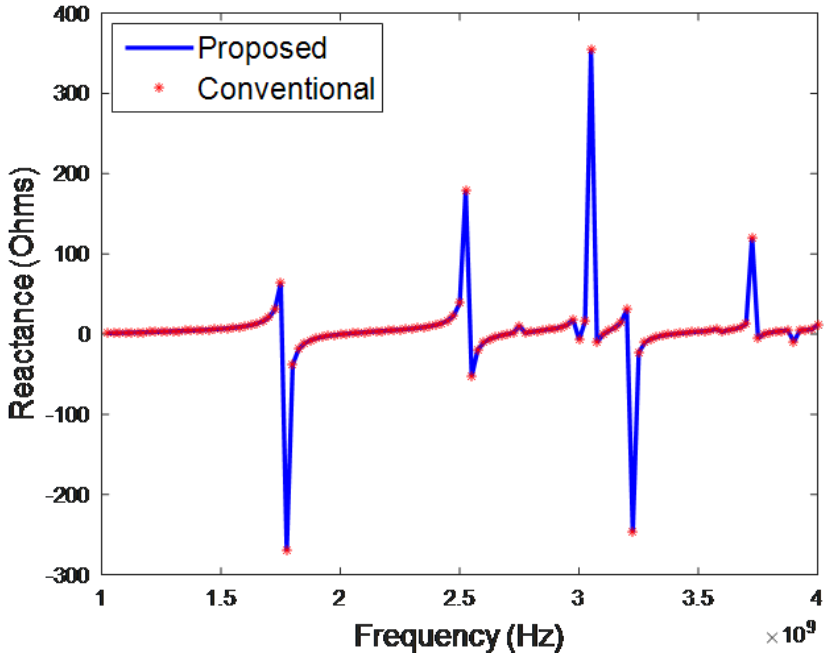


Fig. 7.13. Input reactance versus frequency for a cavity-backed patch antenna.

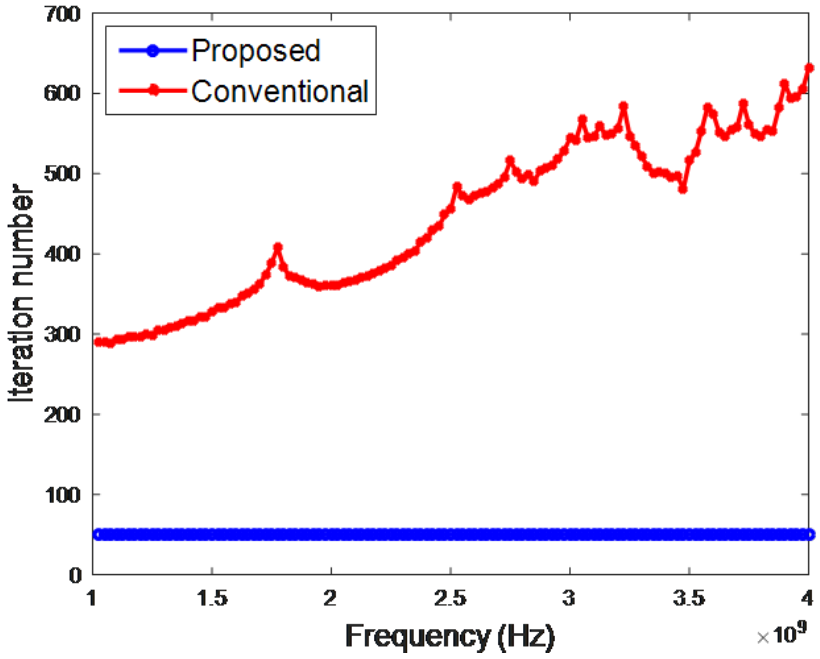


Fig. 7.14. Iteration number in a patch antenna example for convergence comparison.

Patch antenna array

We then simulated arrays of cavity-backed patch antenna of 1 by 1, 2 by 2, 3 by 3, 4 by 4, 5 by 5, and 6 by 6 elements, resulting in from 1,007 to 33,302 unknowns. The array example is shown in Fig. 7.15 and each element is the patch antenna shown in Fig. 7.11. The white region is PEC region including patches and PEC ground planes and the gray region is dielectric cavity. Also, red dots in the figure illustrate feed probes. The simulation server used here has an Intel Xeon E5-2690 3.00 GHz processor.

For the largest unknown example, in 6 by 6 array case, the proposed method identifies 605 over 33,302 modes as $\mathbf{V}_n$ whose absolute magnitude of the eigenvalues is over 0.98 in the transformed system. The corresponding eigenvalues and parameters along the eigenvalue axis of the original indefinite problem are illustrated in Fig. 7.16.

Fig. 7.15. Planar view of 6 by 6 patch antenna array.

Fig. 7.16. Illustration of the eigenvalue distribution, λ_r, ω^2, λ_{max}, and ${\omega_0}^2$ of a 6 by 6 patch array.

The yellow marked region is removed in the transformed system and the highlighted red region is the remained eigenvalue region. The ω is set to be 3×10^9 rad/s and ω_0^2 is chosen as 3.6763×10^{22}. Also, the largest eigenvalue λ_{max} is 3.5349×10^{22}, and the remained smallest eigenvalue λ_r is 7.4423×10^{20}. There exist 605 eigenvalues between λ_r and the smallest non-zero eigenvalue $\lambda_{\min}$ which is 1.6492×10^{18}, and this yellow band is removed in the transformed system. The condition number defined by (7.14) of the proposed deducting method is 48.0663. In contrast, the condition number of original indefinite matrix is 3.9278×10^3. With these condition numbers which strongly affect the convergence of GMRES method, the proposed method is

shown to have a constant number of iterations of 60 to reach an accuracy of 1×10^{-5} error tolerance. In contrast, using the conventional GMRES solver and with the same error tolerance, the original indefinite system is shown to have a large iteration number from 279 to 748 when N reaches 33,302 as shown in Fig. 7.17.

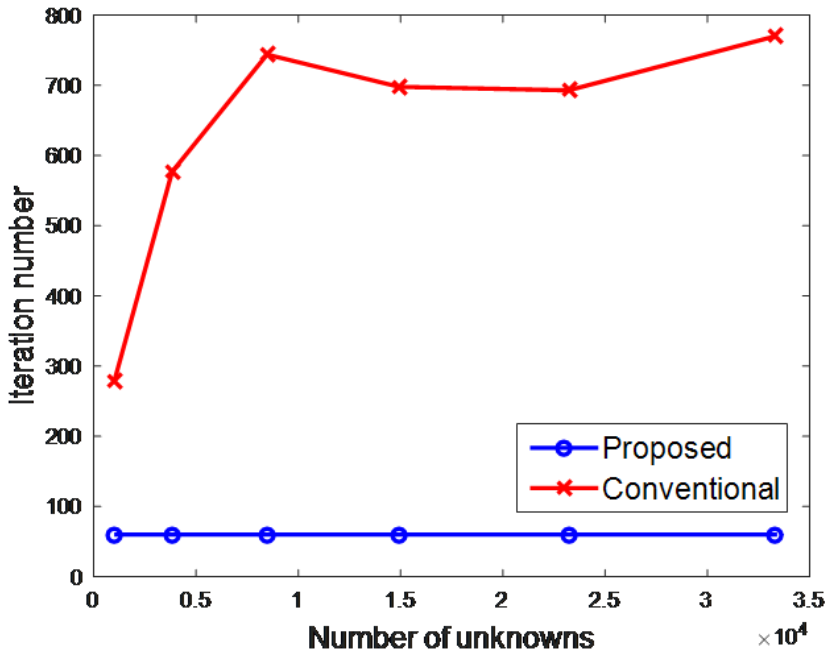


Fig. 7.17. Iteration number comparison.

Also, for example, in 6 by 6 array (33,302 unknown case), the proposed method shows an elapsed time of 6.5844×10^2 s with 60 iterations in 1 frequency sweep. In contrast, the conventional GMRES based method with original indefinite system yields an elapsed time of 3.2177×10^3 s with 770 iterations. Thus, the speed-up of 4.8869 is observed.

Fig. 7.18 plots the total CPU time versus N comparison at $\omega = 3 \times 10^9$ rad/s and the proposed method shows excellent performance compared to conventional GMRES based solution.

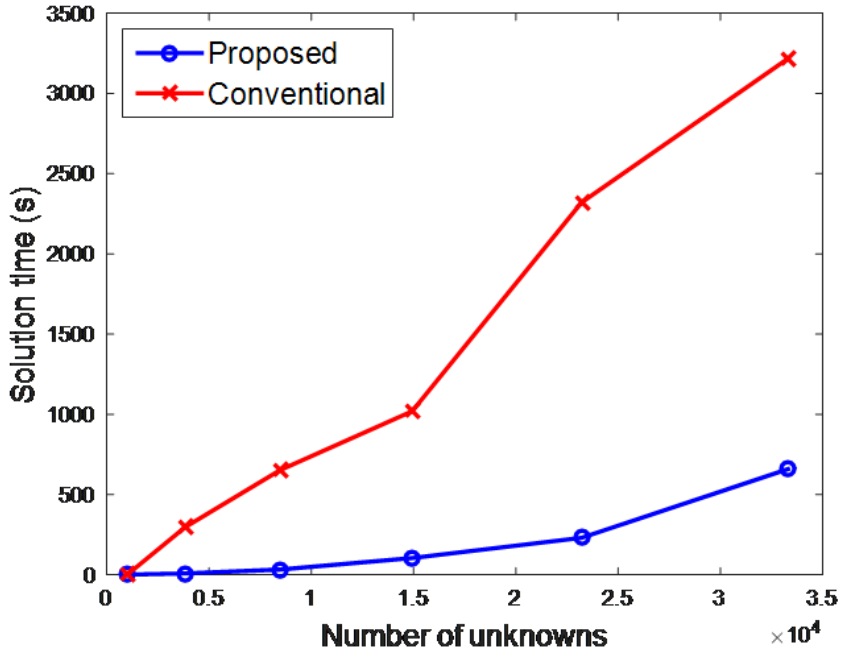


Fig. 7.18. Solution time comparison.

7.4 Conclusions

In this chapter, we develop a symmetric positive definite representation of the FEM operator by removing the non-positive definite component from the system matrix. The resultant solution in frequency domain is different from that of the original system matrix. However, in the second step, we add the contribution from the non-positive definite component with negligible cost back to obtain the true solution. The positive-definite representation after removing non-positive definite modes has a spectral radius less than 1. Its condition number can also be controlled to any desired value. Also, we transform eigenvalue system from the original to obtain a reduced number of non-positive definite modes. Thus, the non-positive definite modes can be obtained efficiently with $O(k^2N)$ complexity with implicitly restarted Arnoldi method.

As a result, the performance of the proposed method is enhanced compared to conventional frequency-domain FEM methods. Numerical experiments have demonstrated the accuracy and efficiency of the proposed method.

8. CONCLUSIONS

In this dissertation, fast time- and frequency-domain finite-element methods for electromagnetic analysis are proposed. There are mainly two directions we pursue to accelerate the time- and frequency-domain electromagnetic analysis: One is to reduce computational complexity for one simulation. The other is to reduce the number of simulations. The algorithms and implementations in this dissertation are proposed based on these strategies.

In chapter 2, the structure specialty of on-chip circuits such as Manhattan geometry and layered permittivity is preserved in the proposed algorithm. As a result, the large-scale matrix solution encountered in the 3-D circuit analysis is turned into a simple algebraic operation, which can be obtained in linear complexity with negligible cost. In chapter 3, fast structure-aware direct time domain finite element framework is proposed. Based on this framework under DC dominant condition, and utilizing **T**'s solver in previous chapter, the time step size is not sacrificed, and the total number of time steps to be simulated is also significantly reduced, thus achieving a total cost reduction in CPU time. In chapter 4 and 5, the proposed method is to update the time-domain finite-element method (TDFEM) numerical system to exclude the source of instability. As a result, an explicit TDFEM simulation is made stable for an arbitrarily large time step.

The limitation of proposed algorithm in chapter 3 is that it is applicable to only DC-dominant problems. Even though DC-dominant cases dominate many circuit applications, full-wave analysis is still required. This full-wave method is proposed in chapter 6 with the support of matrix exponential framework and structure-aware **T**'s solver in chapter 2 and 3. The ideas in chapter 4 and 5 which make the norm of the system matrix smaller by removing unstable modes accelerate the convergenc of the series expansion of matrix exponential components.

Then, we expand our understanding and knowledge into the frequency domain analysis from time domain analysis. The solution cost which is associated with an indefinite nature of the system matrix is also as high as that of time-domain cases. To reduce this high cost, in chapter 7, we show the same unstable mode removal approach we proposed in time domain can be adopted to transform an indefinite system matrix to a positive definite one in frequency domain. It is worth mentioning that in frequency domain, we have to consider the contributions from non-positive definite modes also, unlike that in time domain. In summary, the time domain algorithms developed so far have a great potential to apply to the frequency domain analysis.

The scope of this work can be extended as following future work. First, the algorithm proposed in chapter 3 requires a computation of nullspace. Fast algorithms for extracting the nullspace can be further studied. For the proposed methods from chapter 4 to 7, the number of unstable modes or non-positive definite modes is important because the determination of these modes still consumes a large part of the computation. Thus, the methods for accelerating eigenvalue analysis are still required. Also, the diagonalization perspectives of system matrix including changing basis function can be effective to fully exploit the merit of unstable mode exclusion scheme as well as to speed up the inversion process. In addition, the parallel computation of unknowns along each direction can be further studied. As the clock speed of CPU is not drastically enhanced these days, the parallel computation to Exa scale attracts many researchers' interest. By a proper arrangement of unknowns, the decoupling of the unknowns along each direction is possible and it can be naturally implemented in proposed algorithms. Therefore, taking advantage of the parallel computing can accelerate the simulation further.

LIST OF REFERENCES

LIST OF REFERENCES

[1] W. C. Chew, E. Michielssen, J. Song, and J.-M. Jin, *Fast and efficient algorithms in computational electromagnetics*. Artech House, Inc., 2001.

[2] Z. Cendes, "Keynote speech: Computational electromagnetics has changed my life," in *The 28th International Review of Progress in Applied Computational Electromagnetics*. The Applied Computational Electromagnetics Society, 2012.

[3] J. Lee, "The maturity of computational electromagnetics: Are we there yet?" in *Proceedings of the 5th European Conference on Antennas and Propagation (EUCAP)*. IEEE, 2011, pp. 3011–3014.

[4] G. A. Vandenbosch, "The future of computational electromagnetics: Science or product [euraap corner]," *IEEE Antennas and Propagation Magazine*, vol. 53, no. 3, pp. 264–269, 2011.

[5] H. Gan and D. Jiao, "A time-domain layered finite element reduction recovery (lafe-rr) method for high-frequency vlsi design," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 12, pp. 3620–3629, 2007.

[6] ——, "A fast-marching time-domain layered finite-element reduction-recovery method for high-frequency vlsi design," *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 2, pp. 577–581, 2009.

[7] M. Ha, K. Srinivasan, and M. Swaminathan, "Transient chip-package cosimulation of multiscale structures using the laguerre-fdtd scheme," *IEEE Transactions on Advanced Packaging*, vol. 32, no. 4, pp. 816–830, 2009.

[8] K. Sun, Q. Zhou, K. Mohanram, and D. C. Sorensen, "Parallel domain decomposition for simulation of large-scale power grids," in *2007 IEEE/ACM International Conference on Computer-Aided Design*. IEEE, 2007, pp. 54–59.

[9] G. Antonini and A. E. Ruehli, "Waveform relaxation time domain solver for subsystem arrays," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 4, pp. 760–768, 2010.

[10] R. Wang and J.-M. Jin, "A symmetric electromagnetic-circuit simulator based on the extended time-domain finite element method," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 12, pp. 2875–2884, 2008.

[11] A. Rong and A. C. Cangellaris, "Generalized peec models for three-dimensional interconnect structures and integrated passives of arbitrary shapes," in *Electrical Performance of Electronic Packaging, 2001*. IEEE, 2001, pp. 225–228.

[12] H. Gan and D. Jiao, "Hierarchical finite-element reduction-recovery method for large-scale transient analysis of high-speed integrated circuits," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 1, pp. 276–284, 2010.

[13] W. Lee and D. Jiao, "Structure-aware time-domain finite-element method for efficient simulation of vlsi circuits," in *2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)*, 2014.

[14] Q. He, D. Chen, J. Zhu, and D. Jiao, "Minimal-order circuit model based fast electromagnetic simulation," in *2013 IEEE 22nd Conference on Electrical Performance of Electronic Packaging and Systems*. IEEE, 2013, pp. 219–222.

[15] D. Chen and D. Jiao, "Time-domain orthogonal finite-element reduction-recovery method for electromagnetics-based analysis of large-scale integrated circuit and package problems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 8, pp. 1138–1149, 2009.

[16] H. Gan and D. Jiao, "An unconditionally stable time-domain finite element method of significantly reduced computational complexity for large-scale simulation of ic and package problems," in *2009 IEEE 18th Conference on Electrical Performance of Electronic Packaging and Systems*. IEEE, 2009, pp. 145–148.

[17] C. Moler and C. Van Loan, "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later," *SIAM review*, vol. 45, no. 1, pp. 3–49, 2003.

[18] A. J. Laub, *Computational matrix analysis*. SIAM, 2012.

[19] Q. He, H. Gan, and D. Jiao, "Explicit time-domain finite-element method stabilized for an arbitrarily large time step," *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 11, pp. 5240–5250, 2012.

[20] J. Zhu and D. Jiao, "A theoretically rigorous full-wave finite-element-based solution of maxwell's equations from dc to high frequencies," *IEEE Transactions on Advanced Packaging*, vol. 33, no. 4, pp. 1043–1050, 2010.

[21] ——, "A rigorous solution to the low-frequency breakdown in full-wave finite-element-based analysis of general problems involving inhomogeneous lossless/lossy dielectrics and nonideal conductors," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 12, pp. 3294–3306, 2011.

[22] R. Sharma and T. Chakravarty, *Compact Models and Measurement Techniques for High-Speed Interconnects*. Springer Science & Business Media, 2012.

[23] D. Jiao, J. Zhu, and S. Chakravarty, "A fast frequency-domain eigenvalue-based approach to full-wave modeling of large-scale three-dimensional on-chip interconnect structures," *IEEE transactions on advanced packaging*, vol. 31, no. 4, pp. 890–899, 2008.

[24] M. J. Kobrinsky, S. Chakravarty, D. Jiao, M. Harmes, S. List, and M. Mazumder, "Experimental validation of crosstalk simulations for on-chip interconnects at high frequencies using s-parameters," in *Electrical Performance of Electronic Packaging, 2003*. IEEE, 2003, pp. 329–332.

[25] G. Meurant, "A review on the inverse of symmetric tridiagonal and block tridiagonal matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 13, no. 3, pp. 707–728, 1992.

[26] D. Jiao and J. Jin, "Finite element analysis in time domain," *The finite element method in electromagnetics*, pp. 529–584, 2002.

[27] http://www.cise.ufl.edu/research/sparse/umfpack/UMFPACK-5.7.0.tar.gz.

[28] J. Jain, H. Li, S. Cauley, C.-K. Koh, and V. Balakrishnan, "Numerically stable algorithms for inversion of block tridiagonal and banded matrices," 2007.

[29] J. Zhu and D. Jiao, "Fast full-wave solution that eliminates the low-frequency breakdown problem in a reduced system of order one," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 2, no. 11, pp. 1871–1881, 2012.

[30] Y. Saad, *Numerical Methods for Large Eigenvalue Problems: Revised Edition*. Siam, 2011.

[31] M. Gaffar and D. Jiao, "An explicit and unconditionally stable fdtd method for the analysis of general 3-d lossy problems," *IEEE Transactions on Antennas and Propagation*, vol. 63, no. 9, pp. 4003–4015, 2015.

[32] N. J. Higham, D. S. Mackey, F. Tisseur, and S. D. Garvey, "Scaling, sensitivity and stability in the numerical solution of quadratic eigenvalue problems," *International journal for numerical methods in engineering*, vol. 73, no. 3, pp. 344–360, 2008.

[33] J. Lee, D. Chen, V. Balakrishnan, C.-K. Koh, and D. Jiao, "A quadratic eigenvalue solver of linear complexity for 3-d electromagnetics-based analysis of large-scale integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 31, no. 3, pp. 380–390, 2012.

[34] B. N. Datta, *Numerical linear algebra and applications*. Siam, 2010.

[35] Y. Saad and M. H. Schultz, "Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on scientific and statistical computing*, vol. 7, no. 3, pp. 856–869, 1986.

[36] L. N. Trefethen and D. Bau III, *Numerical linear algebra*. Siam, 1997.

[37] H. A. Van der Vorst, *Iterative Krylov methods for large linear systems*. Cambridge University Press, 2003.

[38] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, "Variational iterative methods for nonsymmetric systems of linear equations," *SIAM Journal on Numerical Analysis*, vol. 20, no. 2, pp. 345–357, 1983.

[39] D. C. Sorensen, "Implicit application of polynomial filters in ak-step arnoldi method," *Siam journal on matrix analysis and applications*, vol. 13, no. 1, pp. 357–385, 1992.

[40] R. J. Radke, "A matlab implementation of the implicitly restarted arnoldi method for solving large-scale eigenvalue problems," 1996.

APPENDIX

A. UNSTABLE MODE DETERMINATION CRITERION IN FORWARD-DIFFERENCE BASED TIME DISCRETIZATION SCHEME

The first-order double-dimension system of equation is as following

$$\begin{gathered}\mathbf{A}\frac{d}{dt}\tilde{u}+\mathbf{B}\tilde{u}=\tilde{f}\\ \mathbf{A}=\begin{bmatrix}\mathbf{R} & \mathbf{T}\\ \mathbf{T} & \mathbf{0}\end{bmatrix},\ \mathbf{B}=\begin{bmatrix}\mathbf{S} & 0\\ 0 & -\mathbf{T}\end{bmatrix}.\end{gathered} \tag{A.1}$$

The eigenvalue problem which governs the field solution is

$$\mathbf{BV}=\mathbf{AV}\mathbf{\Lambda} \tag{A.2}$$

where diagonals of $\mathbf{\Lambda}$ are eigenvalues and $\mathbf{V}$ is eigenvectors. The extended solution vector $\tilde{u}$ can be expanded using eigenspace from (A.2) as following

$$\tilde{u}=\mathbf{V}y. \tag{A.3}$$

After substituting (A.3) into (A.1) then multiplying $\mathbf{V}^T$ both sides, following relationship satisfies.

$$\mathbf{V}^T\mathbf{AV}\left(\frac{d}{dt}y+\mathbf{\Lambda}y\right)=\mathbf{V}^T\tilde{f}. \tag{A.4}$$

(A.4) can be simplified further as following

$$\frac{d}{dt}y+\mathbf{\Lambda}y=\tilde{f}'. \tag{A.5}$$

(A.5) can be divided into single line of equation, per each eigenvalue where and are the real part and the imaginary part of the eigenvalue, respectively, as following

$$\frac{d}{dt}y_i+(a_i+jb_i)y_i=\tilde{f}_i'. \tag{A.6}$$

Then, after applying forward-difference based time discretization scheme (A.6) turns into

$$y_i^{n+1}-y_i^n+\Delta t\lambda_i y_i^n=\Delta t\tilde{f}_i'. \tag{A.7}$$

By adopting z-transform for the stability analysis of (A.7),

$$z=1-\Delta t\lambda_i. \tag{A.8}$$

To satisfy stability criterion, $|z| < 1$ should satisfy. Then,

$$|z| = |1 - \Delta t\lambda_i| = |1 - \Delta t(a_i + jb_i)| = |1 - \Delta t a_i - j\Delta t b_i| < 1. \tag{A.9}$$

Also,

$$(1 - \Delta t a_i)^2 + (\Delta t b_i)^2 < 1. \tag{A.10}$$

(A.10) can be simplified as following

$$({a_i}^2 + {b_i}^2)\Delta t^2 - 2a_i\Delta t = \Delta t(({a_i}^2 + {b_i}^2)\Delta t - 2a_i) < 0. \tag{A.11}$$

It is apparent that $\Delta t > 0$, thus

$$({a_i}^2 + {b_i}^2)\Delta t - 2a_i < 0. \tag{A.12}$$

Finally, we can obtain

$$2real(\lambda)/(real(\lambda)^2 + imag(\lambda)^2) > \Delta t \tag{A.13}$$

as unstable modes which violate (A.12) stability condition.

VITA

VITA

Woochan Lee received the B.S. and M.S. degrees in electrical engineering from Seoul National University, Seoul, Korea, in 2002 and 2005, respectively. He is currently pursuing the Ph.D. degree with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA. His current research interests include computational electromagnetics for very large-scale electromagnetic analysis.

He was commissioned as a Full-time Lecturer and a First Lieutenant with Korea Military Academy, Seoul, Korea, from 2005 to 2008. He has been a Deputy Director and a Patent Examiner with Korean Intellectual Property Office, Daejeon, Korea, since 2004. He passed the National Higher Civil Service Examination, the most honorable and competitive exam given by the Korean Government, as the youngest deputy director of the year 2002 in the division of electrical engineering. He is a member of Korean Academy of Government-Supported Scholars.