Sensitivity Modeling of On-chip Capacitances

Parasitics Extraction for Manufacturing Variability

Sensitivity Modeling of On-chip Capacitances

Parasitics Extraction for Manufacturing Variability

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus Prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op vrijdag 6 juli 2012 om 12.30 uur
door

Yu BI

Master of Science in Microelectronics, Technische Universiteit Delft
geboren te Beijing, China.

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. ir. P.M. Dewilde

Copromotor:

Dr. ir. N.P. van der Meijs

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr. ir. P.M. Dewilde	Technische Universiteit Delft, promotor
Dr. ir. N.P. van der Meijs	Technische Universiteit Delft, copromotor
Prof. dr. ir. E. Charbon	Technische Universiteit Delft
Dr. ir. P.J.A. Harpe	Technische Universiteit Eindhoven
Prof. dr. K.A.A. Makinwa	Technische Universiteit Delft
Dr. ir. W. Schoenmaker	Magwel NV
Prof. dr. ir. C. Vuik	Technische Universiteit Delft
Prof. dr. ir. A.-J. van der Veen	Technische Universiteit Delft, reservelid

Printed in The Netherlands.

ISBN: 978-94-6186-039-2

Cover design by Zhenyuan Yang & Yu Bi

In memory of my grandmothers
and
dedicated to my family

Contents

1 Introduction **1**
1.1 Manufacturing Variability 3
1.2 Contributions and Outline of the Thesis 5
1.3 Notations 8
1.4 Acronyms and Abbreviations 8

2 Sensitivity Modeling of Capacitances for Manufacturing Variability **9**
2.1 Capacitance Extraction Using BEM 9
2.2 Sensitivity-based Model for Systematic and Random Variations 11
2.3 Assumptions 14

3 Capacitance Sensitivity Computation by the Adjoint Field Technique **15**
3.1 Adjoint Field Technique 15
3.1.1 Circuit Network 16
3.1.2 Electrostatic Field 16
3.1.3 Energy Conservation in the Electrostatic Field . . . 19
3.1.4 Small Variation in the Original System 20

3.2 Algorithm Development 21
3.2.1 Illustrative Example 26
3.3 Analytical Example 28

4 Algorithm Implementation and Experiments 31
4.1 Introduction 32
4.2 Procedural Algorithm 33
4.2.1 Complexity Analysis 38
4.3 Window-Scheme 39
4.4 Experiment I: Accuracy Verification 43
4.5 Experiment II: Statistical Interpretation of Sensitivity . . . 46
4.6 Experiment III: Windowing Technique 48

5 Enhanced Computation of the Capacitance Sensitivity 57
5.1 Problem Statement 57
5.2 Algorithm Extension by the Schur Complement Technique . 59
5.3 Experiment and Results 63
5.3.1 Experiment-1: Sensitivity Computation 63
5.3.2 Experiment-2: Variational Study 65
5.4 Conclusion 65

6 Efficient Sensitivity-Based Capacitance Modeling for Systematic and Random Variations 71
6.1 Copper Damascene and Random Geometric Variations . . . 72
6.2 Statistical Model of LER 74
6.3 Physical Description of LER: Random Line Pattern 75
6.4 Statistical Model of Capacitances Using Panel Sensitivities 75
6.5 Verification and Experiment 77
6.5.1 Experiment I 77
6.5.2 Experiment II 78
6.6 A Case Study 80
6.7 Sensitivity-Based Modeling for Both Systematic and Random Variations 84
6.8 Conclusion 87

7 Fast Statistical Analysis of RC Nets Subject to Manufacturing Variabilities 89
7.1 Parameterized Capacitance Extraction 91
7.2 Parameterized Conductance Extraction 92

7.2.1 FEM-based Conductance Extraction 92
7.2.2 Conductance Sensitivities 94
7.3 Order Reduction of Parameterized Systems 95
7.3.1 Parameterized Model Order Reduction 95
7.3.2 Explicit Parameter Matching 96
7.4 Statistical Analysis of RC Nets 98
7.4.1 Design Flow 98
7.4.2 Statistical Property Computation 101
7.4.3 Experiment and Result 103
7.5 Conclusion 104

8 Conclusion **107**

A Proof 1: The *adjoint* in the electrostatic field **111**

B Proof 2: Electrostatic stored energy **113**

C Sensitivity Computation Using the Domain Decomposition Method **115**

Bibliography **127**

Summary **137**

Samenvatting **141**

Acknowledgment **145**

About the Author **147**

CHAPTER 1

Introduction

"*I believe that such a large circuit can be built on a single wafer.*"

This is a one sentence paragraph in [1] published by Dr. Gordon E. Moore in 1965. By "such a large circuit", he meant "by 1975, the number of components per integrated circuit for minimum cost will be 65,000". Back then, perhaps no one, not even Dr. Moore, would imagine that in 2011, the number of transistors in the latest released Intel six-core Core i7 (Sandy Bridge-E) microprocessor would be 2,270,000,000 [2].

The integrated circuit (IC) industry has followed the famous Moore's law for about half a century: the number of transistors that can be integrated on an integrated circuit doubles approximately every two years. This has been achieved by the down-scaling of the technology node. The great motivation behind the scaling is of course profits, which means faster and more power-efficient chips, more functions on smaller-size electronics, longer battery-time on portable devices, etc. For example, the total number of smartphone sales has almost doubled in one year (from 2009 to 2010) and it takes 19.3% of the total mobile phone market share [3, 4].

However, nothing comes for free. While the scaling in the front-end process enables faster transistors, an *inverse scaling* is observed in the back-end process, which means interconnects are getting slower due to the shrinking of the cross-sectional area. Moreover, since transistors are being scaled down, more and more interconnects are needed for communication,

making the delay associated with signal propagation comparable to the gate delay.

Another by-product of the laudable scaling is the ever-increasing manufacturing variability. This has become a critical issue for circuit performance, reliability, production yield, etc. Such a problem, which has been considered as secondary for a long time, has become quite serious especially since the technology node entered the deep sub-micron era. It has, in fact, slowed down the process of scaling. The 2010 version of ITRS has predicted that after 2013, the transistor count will only double every three years instead of every two.

All relevant techniques for analyzing and solving the variability problem have been broadly labeled *Design for Manufacturability (DFM)*, and have emerged across the spectrum of the whole IC industry. For example, the use of resolution enhanced techniques (RETs), e.g. phase shift masks, in lithography has been a primary enabler of the continuing reduction in the critical dimensions (CD) [5]. Meanwhile, the optical proximity correction (OPC) technique has been used to compensate for the layout distortion induced during the lithography process due to the aggressive scaling. Apart from the techniques deployed to directly improve the process steps, techniques such as real-time feed-back control are also playing an important role in reducing environmental variabilities [6].

Analog designers, especially those working on digital-to-analog converters, reference sources, etc., were perhaps the first ones to notice the impact of process variations in terms of (mis)matching [7, 8]. Since then, various researches such as statistical designs (or optimizations) and adaptive circuits have been proposed by both analog and digital circuit designers to alleviate the effect of process variations, achieving a high performance and/or a high yield [9–13].

While the technologists have been trying to realize scaling while controlling the variability, and the designers have been investigating either adaptive or variation-insensitive circuits to mitigate the impact of the variability, the EDA community has been dedicated to developing accurate and efficient tools and methodologies to make this challenge transparent to the designers. Proposed solutions involve deterministic methods such as the worst- and best-case corner modeling and design methodology [14,15], and statistical analysis such as sampling-based simulation (e.g. Monte-Carlo and its variants) [16–18].

In fact, the technology, the circuit design and the EDA communities

can not live independently, not any more. They have to work much closer than before [19, 20]. For instance, it is proposed in [21] that design rules should become *yield-aware*. It means that instead of "hard constraints" that check yes/no, future design rules should specify "degrees of meeting various design rules with an indication of the corresponding yield penalty". The development of such yield-aware design rules definitely demands a collaboration between the three communities.

As part of the EDA community, we would also like to contribute our humble capacity to the big family of DFM. The impact of the manufacturing variability on on-chip interconnects has notably increased since the technology nodes entered the deep sub-micron era. Thus, the industry needs *a good understanding* of the effects. "A good understanding" means such effects should be captured and modeled in a way that it is understandable and useful to IC designers. Because, at the end of the day, it is the functionality and performance (or profits) of the designed chips that really matters.

Our research has been stimulated by such industry needs. Hence, the goal of our work is to study and model the effect of manufacturing variations on on-chip interconnects, to make it transparent to designers. Specifically, we focus on interconnect capacitances. This thesis includes our findings and results. Before listing our contributions, we will first give a brief introduction of the categories and sources of the manufacturing variability that relates to our research.

1.1 Manufacturing Variability

Manufacturing variability can be categorized into systematic and random variations. Systematic variations are introduced by predictable design or process procedures, and are often highly layout-dependent. Contour variations, for instance, are induced during the lithography step. Even with the constraints of design rules and the compensation technique, i.e. the optical proximity correction (OPC), the layout distortions (the difference between the actual silicon image and the original design intent), are not fully eliminated, as indicated in Figure 1.1. It contains systematic as well as random linewidth variations [22]. Another example is the thickness variations in metals and interlevel dielectrics (ILD) caused by the chemical mechanical polishing (CMP) step (Figure 1.2). Such thickness variations were at first controlled by design rules which restricted the density range for each layer.

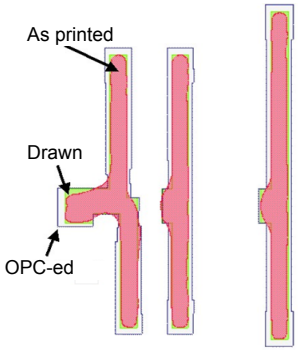

Figure 1.1: *Contour variation induced by lithography [23].*

Then as the pitches became tighter, *metal dummy fills* were introduced. Although the planarization has been greatly improved, variations still exist [24]. Theoretically, since systematic variations are layout-dependent,

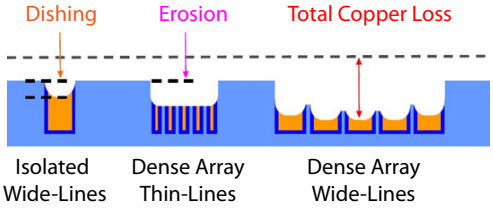

Figure 1.2: *Variation in metal thickness induced by CMP [24].*

it could be modeled deterministically at the pre-manufacturing stage by conducting a thorough analysis of the layout. However, in practice, it is better to be able to model systematic variation statistically. Because in the early design stage, the layout information may not be fully available, or layouts can be changed, adjusted and optimized (e.g. using the statistical models) [25].

Random variations, on the other hand, are caused by various unpredictable fluctuations in the manufacturing environment. In most cases, random variations are not layout dependent, but often a spatial correlation can be observed. One of the most typical random geometric variations is the so-called line-edge roughness (LER), which has been intensively stud-

ied for the critical dimensions of MOSFETs since the technology nodes reached deep sub-micron dimensions [26]. With technology shrinking, the impact of LER on interconnects and some novel designs of passive components with high-precision requirements [27] needs to be understood and modeled.

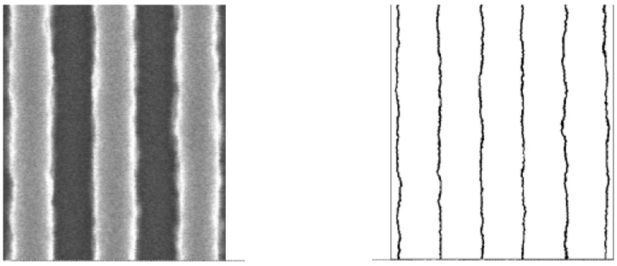

Figure 1.3: *SEM view of a rough line and its detected edge is shown [26].*

1.2 Contributions and Outline of the Thesis

The outline of this thesis is shown schematically in Figure 1.4. The contents and contributions of each chapter are summarized as follows.

In this chapter (Chapter 1), we have given a brief introduction of manufacturing variability, its ever-increasing impact on IC design and the need for advanced post-layout verification tools taking into account such effect. Our work focuses on modeling the interconnect capacitances and our goal is to capture both the systematic and the random variations with a good trade-off between efficiency and accuracy.

To achieve the goal, we propose a solution in Chapter 2, that is the sensitivity-based modeling method. In this thesis, the term *sensitivity* specifically refers to the first derivative with respect to a geometric parameter. The proposed method is applicable to Boundary Element Method (BEM) based capacitance extractors. Hence, capacitance extraction using BEM is briefly introduced, based on which we also explain the challenge associated with the sensitivity computation.

Chapter 3 proposes a fast sensitivity computation method based on the Adjoint Field Technique (AFT). It shows that the capacitance sensit-

ivities with respect to multiple parameters can be obtained together with the nominal capacitances with only one system solve. We also include an alternative idea for the sensitivity computation based on a Domain Decomposition Method (Appendix C). This approach results in the same algorithm and conclusion.

The proposed algorithm is then implemented in the SPACE layout-to-circuit parasitics extractor. Chapter 4 explains the procedural algorithm, the data structure and the complexity. A window-scheme is also introduced for applications with larger structures. Several experiments have been conducted on the SPACE platform to demonstrate the accuracy and efficiency of the algorithm.

The conducted experiments indicate certain computational errors that are unavoidable. Chapter 5 studies the cause of these errors and proposes a supplementary algorithm for improving the accuracy of the computed sensitivity. The supplementary algorithm is based on the Schur Complement Technique. A combination of the basic algorithm (proposed in Chapter 3 and Appendix C) and the supplementary algorithm leads to an enhanced algorithm which is able to achieve an extremely high accuracy at the cost of the computational time. Since the primary goal of our work is *high efficiency with good accuracy*, it is still the basic algorithm that is adopted in our application examples (Chapter 6 and 7), while the enhanced algorithm enriches the theory part of our work and can be a potential topic for future study.

Two possible applications of the proposed sensitivity-based modeling method are demonstrated in this thesis. Chapter 6 shows an efficient capacitance modeling for not only systematic but also random variations. Specifically, we study a real design case, an 8-bit binary-scaled differential charge-redistribution digital-to-analog converter (a component of a low power SAR ADC design). Measurements on test chips gives results supporting the estimate given by our model.

Chapter 7 presents a highly efficient statistical analysis methodology for RC nets subject to systematic variations. It achieves *zero* parameter sampling, based on the combination of the proposed sensitivity-based parameterized parasitics extraction technique and a structure-preserving parameterized Model Order Reduction (pMOR) technique. Given the layout and the process spreads of the technology, the statistical properties such as the mean and the standard deviation of the system response can be obtained very fast.

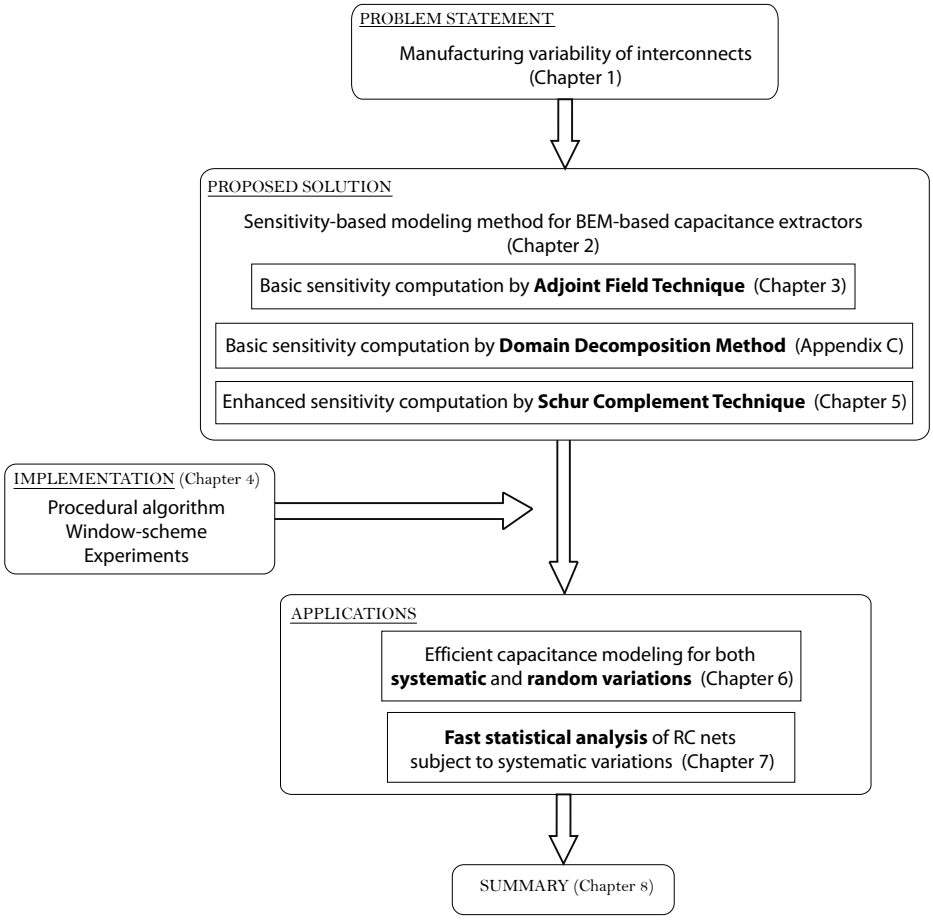

Figure 1.4: *Outline of the thesis.*

1.3 Notations

For ease of discussion, the following notations will be used throughout this thesis:

C_{ij}	Network capacitance between two conductors i and j
C_{sij}	Short-circuit capacitance between two conductors i and j
$\bar{C}_{sij}$	Partial short-circuit capacitance between two panels i and j
$\mathcal{C}^*$	Capacitance between a panel and a node/conductor
$\mathcal{G}$	Elastance matrix
$\mathbf{A}$	Incidence matrix relates $\bar{\mathbf{C}}_s$ and $\mathbf{C_s}$
$\mathbf{B}$	Reduced branch incidence matrix in circuit networks
$\mathcal{S}$	Panel sensitivity of capacitance

1.4 Acronyms and Abbreviations

We also define in the following the acronyms and abbreviations that are used in this thesis for readers' reference:

AFT	Adjoint Field Technique
ADCs	Analog-to-Digital Converters
BEM	Boundary Element Method
CD	Critical Dimension
CMP	Chemical Mechanical Polishing
DFM	Design for Manufacturability
FD	Finite Difference
ILD	Interlevel Dielectrics
LER	Line-edge Roughness
LPE	Layout Parasitics Extraction
OPC	Optical Proximity Correction
pMOR	parameterized Model Order Reduction
RETs	Resolution Enhanced Techniques
ROM	Reduced Order Model
SPEF	Standard Parasitics Exchange Format

CHAPTER 2

Sensitivity Modeling of Capacitances for Manufacturing Variability

2.1 Capacitance Extraction Using BEM

Since our proposed algorithm is intended for use with capacitance extractors using Boundary Element Method (BEM), we will first give a brief introduction on this extraction method.

Capacitance computation is an electrostatic problem. A problem in the electrostatic field usually involves the set-up and solving the Poisson equation, under a suitable boundary condition.

Consider a system with charged conductors floating in a homogeneous dielectric medium with permittivity ε. Let $\Phi(a)$ and $\rho(a)$ be, respectively, the electrostatic potential and the charge density at a point $a = (x_a, y_a, z_a)$. The Poisson equation is then expressed as

$$\nabla^2 \Phi(a) = -\frac{\rho(a)}{\varepsilon} \tag{2.1}$$

with

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}. \tag{2.2}$$

The solution of this equation can be written as

$$\Phi(a) = \int_{all\ charges} \mathcal{G}(a,b)\rho(a)db \tag{2.3}$$

where $\mathcal{G}(a,b) = \frac{1}{4\pi\varepsilon|a-b|}$ is the so-called Green's function. $|a-b|$ is the Euclidian distance between point a and point b. The Green's function $\mathcal{G}(a,b)$ can be interpreted as the potential induced at point a by a unit point charge at point b.

By using the BEM, the surfaces of conductors are discretized into small panels (or finite elements). Suppose the system has m panels after discretization, equation (2.3) can be transformed into a matrix expression:

$$\mathbf{v} = \mathcal{G}\mathbf{q} \tag{2.4}$$

where $\mathbf{v} = [v_1\ v_2\ ...\ v_m]^T$ and $\mathbf{q} = [q_1\ q_2\ ...\ q_m]^T$ are, respectively, the finite element potentials and the finite element charges on the discretized panels, and $\mathcal{G}_{ij}$ is the induced potential at panel i due to the charge at panel j.

Using the above equation (2.4), capacitances amongst panels can be derived. By reformulating the equation:

$$\mathbf{q} = \mathcal{G}^{-1}\mathbf{v} = \bar{\mathbf{C}}_s\mathbf{v}, \tag{2.5}$$

we can obtain the *partial short-circuit capacitance matrix*:

$$\bar{\mathbf{C}}_s = \mathcal{G}^{-1}, \tag{2.6}$$

the entries of which are associated with the discretized panels. And $\mathcal{G}$ is called the elastance matrix.

Finally, the conductor capacitances can be computed by collecting the associated entries of $\bar{\mathbf{C}}_s$. This is done with the help of an incidence matrix $\mathbf{A} \in \mathbb{R}^{m\times N}$, with N being the number of conductors, defined as

$$A_{ij} = \begin{cases} 1 & \text{if panel } i \text{ is on conductor } j \\ 0 & \text{otherwise} \end{cases} \tag{2.7}$$

The conductor capacitances $\mathbf{C_s}$ can then be computed:

$$\mathbf{C_s} = \mathbf{A}^T \bar{\mathbf{C}}_s \mathbf{A}. \tag{2.8}$$

The obtained conductor capacitances $\mathbf{C_s}$ are known as the *short-circuit capacitances* [28]. Its entry C_{sij} is equal to the charge on conductor i when conductor j is held at a unit potential and all other conductors are short-circuited to the ground.

These capacitances, however, are not the ones shown in an equivalent circuit which is used as an input for circuit simulators, such as SPICE [29]. Instead, it is the *two-terminal* or *network capacitance* that is used in an equivalent circuit, with a value being the ratio of the free charge and an associated voltage difference between two conductors (or between a conductor and a reference). Network capacitances, denoted $\mathbf{C}$, can be calculated from the short-circuit capacitances $\mathbf{C_s}$ by using the following simple relation:

$$C_{ij} = -C_{sij} \tag{2.9a}$$

$$C_{ii} = \sum_{j=1}^{N} C_{sij} \quad \forall \ i = 1, 2, \ldots N. \tag{2.9b}$$

Above, we have briefly introduced the capacitance computation using BEM. One can observe that it involves a matrix inversion (2.6), which is known to be computationally expensive. There are techniques for avoiding this costly matrix inversion operation, but this is not the focus of this thesis, and actually indifferent for the proposed method. Instead, we will mention two acceleration techniques adopted in the implementation platform of the proposed algorithm, namely the matrix Schur interpolation algorithm and a scan-line based window-scheme. These will be introduced in Chapter 4. With some background knowledge of the BEM-based capacitance extraction, we will then propose our modeling method of capacitances for the manufacturing variability in the next section.

2.2 Sensitivity-based Model for Systematic and Random Variations

Before proposing our method, we would like to first introduce a few studies that have been presented to account for the impact of the ever-increasing

process variations. The studies mentioned here mainly address the systematic geometric variations, while those focusing on random variations will be discussed in Chapter 6.

A parameterized interconnect model library was proposed in [30]. This was done by first generating a multidimensional table of capacitances for a layout of interest using a numerical extractor, where the simulation points were produced by varying the geometric parameters. Then, by curve fitting, a parameterized model of capacitances was produced for that layout. Another approach was described in [31] which modeled the capacitance variations induced by the pattern-dependent ILD thickness variation, with the help of a lookup table. Then, an enhanced lookup method based on analytical capacitance models was presented in [32], of which the complexity was improved over that of [31]. It computed the derivatives of capacitances with respect to the thickness and the linewidth deviations. These derivatives were later referred to as the *sensitivities*.

We have followed the idea of [32], and propose to use a sensitivity-based modeling method to capture the effects of both the systematic and the random variations on parasitic capacitances:

$$\delta C_{ij} = \frac{\partial C_{ij}}{\partial \lambda_p} \lambda_p \tag{2.10}$$

where δC_{ij} is the induced capacitance fluctuation due to a small parameter variation λ_p and $\frac{\partial C_{ij}}{\partial \lambda_p}$ is the related sensitivity. The sensitivity and the related parameter in this expression (2.10) have different meanings for the systematic and the random variations.

Systematic variability often appears in the form of variations in the structural dimensions, for example, layout expansion or shrinking. Thus, the sensitivity for systematic variation modeling is defined to be with respect to a geometric parameter variation. The parameters can be the layout dimension, the thickness of the metal layer and the height of the dielectric.

For random variations, e.g. the LER, λ_p represents the position deviation of a panel which is used to capture the roughness of the line. $\frac{\partial C}{\partial \lambda_p}$ is then the associated *panel sensitivity of capacitances*, showing the impact of the deviated panel on the capacitance. A discussion about the modeling for random variations can be found in Chapter 6.

Sensitivities can be very useful:

- It has been shown that not all variations seem to be equally implic-

ated to capacitances. For each capacitance, some variations deserve further study and modeling, while others can be simply neglected. Capacitance sensitivities with respect to these geometric parameters can be used to setup the threshold for making this distinction [33].

- Capacitance sensitivities are necessary for establishing basic formulas in various variation-aware algorithms, such as the moment-based timing analysis [34], the Hermite polynomial based statistic analysis [35] and the parametric Model Order Reduction (pMOR) proposed in [36, 37]. Also, techniques including fast-corner generation, multi-corner extraction and the variation-aware Static Timing Analysis (STA) presented in [38] are all based on sensitivity models.
- The Standard Parasitic Exchange Format (SPEF) has been extended to incorporate sensitivities for process and temperature variations. Based on the 2009 version of the SPEF standard [39], a netlist consisting of the nominal values of the parasitics and their sensitivities could be generated by Layout Parasitic Extraction (LPE) tools for subsequent analysis.

To compute the sensitivity, however, is not a trivial task. As addressed in 2.1, the (short-circuit) capacitances are obtained from the inversion of the elastance matrix (2.6), of which an entry $\mathcal{G}_{ij}$ amounts to the potential induced at panel i by a unit charge at panel j. Equation (2.11) repeats the Green's function for a uniform dielectric of infinite dimensions:

$$\mathcal{G}_{ij} = \frac{1}{4\pi\varepsilon|p_i - p_j|} \tag{2.11}$$

with $|p_i - p_j|$ being the Euclidean distance between panels p_i and p_j. Thus the capacitance is a non-straightforward function of the panel position or the dimensions of wires. Further, since capacitance is a mutual property between panels (or wires), adding or removing a panel or changing its position can theoretically induce variations in all capacitances in the system.

These two facts make the computation of capacitance sensitivities very complicated. Any technique whose computational complexity depends on the number of parameters or the number of capacitances would not be practically feasible. A major contribution of this thesis is that we present a fast algorithm for capacitance sensitivity computation, as will be introduced in the next chapter. The computational complexity of the proposed algorithm depends on neither the number of parameters nor the number of capacitances.

2.3 Assumptions

Before proposing the algorithm for capacitance sensitivity computation, we make a few constraints/assumptions as follows:

- We consider the electrostatic case only. As a matter of fact, the capacitance computation is an electrostatic problem. Hence, it is appropriate to make such assumption while computing the capacitance sensitivities. Resistance of conductors are involved in the last chapter of the thesis, and consequences of high-frequencies such as the skin-effect are beyond the scope of our research. In the electrostatic case, the full Maxwell equations are reduced to more simple and more precise descriptions such as the Poisson equation or the Laplace equation. Indeed, modern analog circuits may easily approach high frequencies in the GHz spectrum. The solution of modeling all electromagnetic behavior of on-chip features at such high frequencies need to resort to the complete Maxwell equations. Various techniques for electromagnetic field computation have been proposed in the Computational Electromagnetism community.

- We assume that the conductors are perfect and that each conductor forms an equipotential for the capacitance extraction.

- We only consider the geometric process variations of interconnects. There are naturally other aspects of manufacturing variabilities, such as the temperature gradient across the chip. This kind of variabilities needs to be handled separately and differently [40, 41], which is beyond the scope of this thesis.

CHAPTER 3

Capacitance Sensitivity Computation by the Adjoint Field Technique*

3.1 Adjoint Field Technique

The algorithm for the sensitivity computation to-be presented in this chapter is derived from the *adjoint* of a linear operator [44] used in the electrostatic field. In fact, the property of the *adjoint* has been observed in the circuit network for a long time, in the form of the well-known Tellegen's theorem [45]. Before the discussion of the *adjoint* in the electrostatic field, we would like to first review a relatively more familiar case, i.e., the *adjoint* in the circuit network.

*Part of this chapter has been published in [42]: Yu Bi, N.P. van der Meijs and D. Ioan, "Capacitance sensitivity calculation for interconnects by adjoint field technique," in *Proc. SPI*, Avignon, France, May 2008, and [43]:Yu Bi, K.J. van der Kolk, D. Ioan and N.P. van der Meijs, "Sensitivity computation of interconnect capacitances with respect to geometric parameters," in *Proc. EPEP*, San Jose, CA, pp. 209-212, October, 2008.

3.1.1 Circuit Network

Consider an arbitrary lumped network of which the graph $\mathcal{L}$ has $(n+1)$ nodes, b branches, and some certain choices of the node-to-datum potentials have been made. Let $\mathbf{B} \in \mathbb{R}^{n\times b}$ be the reduced branch incidence matrix [46] for $\mathcal{L}$. It is well known that the Kirchhoff's voltage law (KVL) and current law (KCL) hold:

$$KVL: \quad \mathbf{v} = \mathbf{B}^T\mathbf{e} \tag{3.1a}$$

$$KCL: \quad \mathbf{Bi} = 0 \tag{3.1b}$$

where $\mathbf{e} \in \mathbb{R}^n$ is the vector of node-to-datum potentials and $\mathbf{v}, \mathbf{i} \in \mathbb{R}^b$ are the vectors of the branch voltages and currents respectively.

Define $X \triangleq \mathbb{R}^b$, $Y \triangleq \mathbb{R}^n$, and let $< \cdot, \cdot >_X$ and $< \cdot, \cdot >_Y$ be the Euclidean inner products on X and Y respectively. With the Euclidean inner products, $\mathbf{B}^T$ is the *adjoint* of the linear operation $\mathbf{B}$ [44]. Hence, we have

$$< \mathbf{B}^T\mathbf{e}, \mathbf{i} >_X = < \mathbf{e}, \mathbf{Bi} >_Y = 0 \tag{3.2}$$

which is in fact the basic content of the Tellegen's theorem [45]:

$$< \mathbf{v}, \mathbf{i} >_X = 0 \tag{3.3}$$

3.1.2 Electrostatic Field

In the previous section 3.1.1, we have reviewed that in the circuit network, $\mathbf{B}^T$ is the adjoint of the linear operator $\mathbf{B}$. In this section, we will show that using the property of the *adjoint*, circuit variables can be connected with field variables, and from which we can derive an algorithm for the capacitance sensitivity computation.

It has been proved in [47] that $(-\mathbf{grad})$ is the adjoint of $(\mathbf{div})$ under an appropriate condition, which is defined as follows.

> Let P be the class of all infinitely differentiable scalar functions φ on $\mathbb{R}^3$ such that φ vanishes at infinity as fast as $O(r^{-1})$ $(r \in \mathbb{R})^\dagger$ and the derivatives of φ vanish at infinity as fast as $O(r^{-2})$. Let F be the class of all infinitely differentiable vector fields $\vec{\mathbf{u}}$ on $\mathbb{R}^3$ such that $\vec{u}_j \in P$, $j = 1, 2, 3$.

†In the original proof in [47], φ is defined to vanish at infinity as fast as $O(r^{-2})$, but in fact, being $O(r^{-1})$ is already enough. This is also explained in Footnote 3 in [47], and is proved in Appendix A.

Then, [47] proves that the operator $(-\mathbf{grad})$: $P \to F$ is the adjoint of the operator $(\mathbf{div})$: $F \to P$:

$$\langle \mathbf{div}\ \vec{\mathbf{u}}, \varphi \rangle_P = \langle \vec{\mathbf{u}}, -\mathbf{grad}\ \varphi \rangle_F \quad \forall\ (\vec{\mathbf{u}}, \varphi) \in F \times P \tag{3.4}$$

where the inner products on P and F are defined as

$$\langle \varphi, \psi \rangle_P \triangleq \int_{\mathbb{R}^3} [\varphi(x)\psi(x)] d\Omega \tag{3.5}$$

$$\langle \vec{\mathbf{u}}, \vec{\mathbf{w}} \rangle_F \triangleq \int_{\mathbb{R}^3} [\vec{\mathbf{u}}(x) \cdot \vec{\mathbf{w}}(x)] d\Omega \tag{3.6}$$

for all $\varphi, \psi \in P$, $\vec{\mathbf{u}}, \vec{\mathbf{w}} \in F$, with $\vec{\mathbf{u}} \cdot \vec{\mathbf{w}}$ the standard dot product or scalar product in $\mathbb{R}^3$.

We have included the most relevant details of the theorem and its proof in Appendix A, and for the complete theorem, readers can refer to [47].

Next, we are going to apply (3.4) to the electrostatic field in order to derive the two expressions (3.13) and (3.14) for further algorithm derivation and development.

According to the relation between electric field and the scalar potential, and the Gauss's law, the following relations hold for the electrostatic case:

$$\vec{\mathbf{E}} = -\mathbf{grad}\ \Phi \tag{3.7a}$$

$$\mathbf{div}\ \vec{\mathbf{D}} = \rho \tag{3.7b}$$

where $\vec{\mathbf{E}}$ is the electric field vector and $\vec{\mathbf{D}}$ is the electric displacement field vector, Φ is the scalar potential and ρ is the charge density.

Now, we define two electrostatic systems that have the same conductor configuration and the same medium permittivity ε. In fact, the only difference between them is the voltages and charges on conductors. To distinguish them, we name one the *original system*, using notations $\vec{\mathbf{D}}$, $\vec{\mathbf{E}}$, Φ and ρ, and the other one the *auxiliary system*, using notations $\hat{\vec{\mathbf{D}}}$, $\hat{\vec{\mathbf{E}}}$, $\hat{\Phi}$ and $\hat{\rho}$.

In free space, $\Phi, \hat{\Phi}$ vanish at infinity as fast as r^{-1} and $\vec{\mathbf{D}}, \hat{\vec{\mathbf{D}}}$ vanish at infinity as fast as r^{-2}. Thus $\Phi, \hat{\Phi} \in P$ and $\vec{\mathbf{D}}, \hat{\vec{\mathbf{D}}} \in F$, and using (3.4), we have

$$\langle \mathbf{div}\ \vec{\mathbf{D}}, \hat{\Phi} \rangle = \langle \vec{\mathbf{D}}, -\mathbf{grad}\ \hat{\Phi} \rangle \tag{3.8}$$

$$\langle \mathbf{div}\ \hat{\vec{\mathbf{D}}}, \Phi \rangle = \langle \hat{\vec{\mathbf{D}}}, -\mathbf{grad}\ \Phi \rangle \tag{3.9}$$

Next, substituting (3.7) into (3.8) gives us

$$\langle \rho, \hat{\Phi} \rangle = \langle \vec{\mathbf{D}}, \hat{\vec{\mathbf{E}}} \rangle \tag{3.10}$$

Since we only have charges on the surface of conductors and the potential on the surface of each conductor is constant, it follows that

$$\begin{aligned} \int_{S_i} \rho(\mathbf{r}) \hat{\Phi}(\mathbf{r}) dS &= \hat{\Phi}_i \int_{S_i} \rho(\mathbf{r}) dS \\ &= \hat{\Phi}_i Q_i \end{aligned} \tag{3.11}$$

where S_i is the out-surface of a conductor i and Q_i is the total charge on this conductor in the original system. $\hat{\Phi}_i$ is the potential on conductor i in the auxiliary system, and $\hat{\Phi}_i = \hat{V}_i$ with $\hat{V}_i$ being the voltage on this conductor.

Hence, for a system consisting of N conductors with $\mathbf{Q} = [Q_1\ Q_2\ ...\ Q_N]^T$ and $\hat{\mathbf{V}} = [\hat{V}_1\ \hat{V}_2\ ...\ \hat{V}_N]^T$ being their charges and voltages in the original system and the auxiliary system respectively, we have

$$\langle \rho, \hat{\Phi} \rangle = (\mathbf{Q}, \hat{\mathbf{V}}) \tag{3.12}$$

where $(\mathbf{Q}, \hat{\mathbf{V}}) = \mathbf{Q}^T \hat{\mathbf{V}} = \sum_{i=1}^{N} Q_i \hat{V}_i$.

Therefore, using (3.12), (3.8) becomes

$$(\mathbf{Q}, \hat{\mathbf{V}}) = \langle \vec{\mathbf{D}}, \hat{\vec{\mathbf{E}}} \rangle \tag{3.13}$$

Analogously, from (3.9) we can derive the following:

$$(\hat{\mathbf{Q}}, \mathbf{V}) = \langle \hat{\vec{\mathbf{D}}}, \vec{\mathbf{E}} \rangle \tag{3.14}$$

Here, we would like to emphasize that the electrical variables $\mathbf{Q}$, $\mathbf{V}$, $\vec{\mathbf{D}}$, $\vec{\mathbf{E}}$ and $\hat{\mathbf{Q}}$, $\hat{\mathbf{V}}$, $\hat{\vec{\mathbf{D}}}$, $\hat{\vec{\mathbf{E}}}$, respectively characterize the original system and the auxiliary system that have been constructed geometrically identical. Based on the above expressions (3.13) and (3.14), we can derive an algorithm for computing the capacitance sensitivities. Before doing so in Section 3.1.4 and 3.2, we would like to first present an alternative derivation of (3.13) and (3.14) in the next section, which is from an energy conservation point of view.

3.1.3 Energy Conservation in the Electrostatic Field

For a configuration of static charges, the stored energy equals the energy required to assemble the configuration with an initial condition in which the potential field is zero. Thus for a system with N conductors, if we want to know its stored electric energy, we shall study the amount of work that is needed to construct such charge configuration. The initial situation is assumed to be with all charges in a reservoir at infinity which is the zero potential reference.

It has been shown in [48][‡] that the total amount of work required to assemble a static charge configuration defined by charge density $\rho(\mathbf{r})$ in space Ω is

$$W = \frac{1}{2}\int_{\Omega_\infty} \rho(\mathbf{r})\Phi(\mathbf{r})d\Omega \tag{3.15}$$

where $\Phi(\mathbf{r})$ represents the potential induced by $\rho(\mathbf{r})$. And using the field vectors $\vec{\mathbf{D}}$ and $\vec{\mathbf{E}}$, (3.15) can also be expressed from a field-center view as

$$W = \frac{1}{2}\int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot \vec{\mathbf{E}}(\mathbf{r})d\Omega. \tag{3.16}$$

As addressed, the charges exist only in the form of surface density on conductors and the potential on each conductor is constant, thus

$$\int_{\Omega_\infty} \rho(\mathbf{r})\Phi(\mathbf{r})d\Omega = \sum_{i=1}^{N} Q_i V_i \tag{3.17}$$

where Q_i and V_i are the charge and the voltage on conductor i, $(i = 1, 2, ..., N)$ respectively. Considering (3.15), (3.16) and (3.17), we obtain the following relation:

$$W = \frac{1}{2}\sum_{i=1}^{N} Q_i V_i = \frac{1}{2}\int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot \vec{\mathbf{E}}(\mathbf{r})d\Omega \tag{3.18}$$

It can be interpreted that the stored electric energy of the configuration of N conductors with surface charges Q_i and absolute potential V_i $(i = 1, ..., N)$ is $\frac{1}{2}\sum_{i=1}^{N} Q_i V_i$, and the amount of work to build this configuration can be computed by $\frac{1}{2}\int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot \vec{\mathbf{E}}(\mathbf{r})d\Omega$ using the field vectors supported by the charge density $\rho(\mathbf{r})$.

[‡]We have included the proof from [48] in Appendix B. Readers who are interested may also refer to [49] Section 1.22 where an alternative proofing approach is stated.

Naturally, the following equation holds:

$$(\mathbf{Q}, \mathbf{V}) = \langle \vec{\mathbf{D}}, \vec{\mathbf{E}} \rangle \tag{3.19}$$

where $(\mathbf{Q}, \mathbf{V}) = \mathbf{Q}^T \cdot \mathbf{V}$ and $\langle \vec{\mathbf{D}}, \vec{\mathbf{E}} \rangle = \int_\Omega \vec{\mathbf{D}} \cdot \vec{\mathbf{E}} d\Omega$.

Now, assume there are two systems with the same geometric configuration as in Section 3.1.2, and using the same notations that $\vec{\mathbf{D}}$, $\vec{\mathbf{E}}$, $\mathbf{Q}$, $\mathbf{V}$ for the original system and $\hat{\vec{\mathbf{D}}}$, $\hat{\vec{\mathbf{E}}}$, $\hat{\mathbf{Q}}$, $\hat{\mathbf{V}}$ for the auxiliary system. Then, we can obtain the same equations as (3.13) (3.14) while using a similar derivation as for (3.19).

In fact, we may consider (3.18) as the *actual-energy conservation* and (3.13) (3.14) the *quasi-energy conservation*, which is similar to the Tellegen's *actual-power theorem* and *quasi-power theorem* for the circuit network [45].

3.1.4 Small Variation in the Original System

If there are some very small geometric variations occurring in the original system, then the variables describing the system $\mathbf{Q}$, $\mathbf{V}$ and $\vec{\mathbf{D}}$, $\vec{\mathbf{E}}$ are changed accordingly and the perturbed quantities are denoted as $\Delta\mathbf{Q}$, $\Delta\mathbf{V}$ and $\Delta\vec{\mathbf{D}}$, $\Delta\vec{\mathbf{E}}$ respectively. The configuration in the auxiliary system, on the other hand, remains the same. Thus $\hat{\mathbf{Q}}$, $\hat{\mathbf{V}}$ and $\hat{\vec{\mathbf{D}}}$, $\hat{\vec{\mathbf{E}}}$ would not be affected. Then considering the linearity of inner product, focusing on the first-order of perturbation and neglecting higher order terms, (3.13) leads to

$$(\Delta\mathbf{Q}, \hat{\mathbf{V}}) = \langle \Delta\vec{\mathbf{D}}, \hat{\vec{\mathbf{E}}} \rangle \tag{3.20}$$

and (3.14) leads to

$$(\hat{\mathbf{Q}}, \Delta\mathbf{V}) = \langle \hat{\vec{\mathbf{D}}}, \Delta\vec{\mathbf{E}} \rangle. \tag{3.21}$$

Subtracting (3.21) from (3.20) gives

$$(\Delta\mathbf{Q}, \hat{\mathbf{V}}) - (\hat{\mathbf{Q}}, \Delta\mathbf{V}) = \langle \Delta\vec{\mathbf{D}}, \hat{\vec{\mathbf{E}}} \rangle - \langle \hat{\vec{\mathbf{D}}}, \Delta\vec{\mathbf{E}} \rangle \tag{3.22}$$

Using $\mathbf{Q} = \mathbf{C_s}\mathbf{V}$, the change in $\mathbf{Q}$ can be written as

$$\Delta\mathbf{Q} = (\Delta\mathbf{C_s})\mathbf{V} + \mathbf{C_s}(\Delta\mathbf{V}) \tag{3.23}$$

Then the left-hand side of (3.22) becomes

$$(\Delta\mathbf{Q}, \hat{\mathbf{V}}) - (\hat{\mathbf{Q}}, \Delta\mathbf{V}) = ((\Delta\mathbf{C_s})\mathbf{V}, \hat{\mathbf{V}}) + (\mathbf{C_s}(\Delta\mathbf{V}), \hat{\mathbf{V}}) - (\hat{\mathbf{C}_\mathbf{s}}\hat{\mathbf{V}}, \Delta\mathbf{V}) \tag{3.24}$$

Note that by construction, the original system and the auxiliary system have the same geometric configuration and medium property. Thus $\mathbf{C_s} = \hat{\mathbf{C}}_\mathbf{s}$, as the (short-circuit) capacitances are only determined by the geometric information such as the dimensions and positions of the conductors and the medium permittivity. Hence, the final two terms in (3.24) cancel because

$$(\mathbf{C_s}(\Delta\mathbf{V}), \hat{\mathbf{V}}) = (\hat{\mathbf{C}}_\mathbf{s}\hat{\mathbf{V}}, \Delta\mathbf{V}) = \sum_{i=1}^{N}\sum_{j=1}^{N} C_{sij}\Delta V_j \hat{V}_i \tag{3.25}$$

and (3.24) becomes

$$(\Delta\mathbf{Q}, \hat{\mathbf{V}}) - (\hat{\mathbf{Q}}, \Delta\mathbf{V}) = ((\Delta\mathbf{C_s})\mathbf{V}, \hat{\mathbf{V}}) \tag{3.26}$$

Next, we study the right-hand side of (3.22). In linear isotropic media, $\vec{\mathbf{D}} = \varepsilon\vec{\mathbf{E}}$ and $\hat{\vec{\mathbf{D}}} = \hat{\varepsilon}\hat{\vec{\mathbf{E}}}$ hold for the original and the auxiliary systems with ε and $\hat{\varepsilon}$ the material permittivities respectively. As indicated, the medium of the auxiliary system is the same as that of the original system, i.e. $\hat{\varepsilon} = \varepsilon$. And $\Delta\vec{\mathbf{D}} = (\Delta\varepsilon)\vec{\mathbf{E}} + \varepsilon(\Delta\vec{\mathbf{E}})$. Hence, the right-hand side of (3.22) becomes

$$\begin{aligned}\langle\Delta\vec{\mathbf{D}}, \hat{\vec{\mathbf{E}}}\rangle - \langle\hat{\vec{\mathbf{D}}}, \Delta\vec{\mathbf{E}}\rangle &= \langle(\Delta\varepsilon)\vec{\mathbf{E}}, \hat{\vec{\mathbf{E}}}\rangle + \langle\varepsilon(\Delta\vec{\mathbf{E}}), \hat{\vec{\mathbf{E}}}\rangle - \langle\varepsilon\hat{\vec{\mathbf{E}}}, \Delta\vec{\mathbf{E}}\rangle \\ &= \langle(\Delta\varepsilon)\vec{\mathbf{E}}, \hat{\vec{\mathbf{E}}}\rangle \end{aligned} \tag{3.27}$$

Now, using (3.26) and (3.27), (3.22) turns into

$$((\Delta\mathbf{C_s})\mathbf{V}, \hat{\mathbf{V}}) = \langle(\Delta\varepsilon)\vec{\mathbf{E}}, \hat{\vec{\mathbf{E}}}\rangle \tag{3.28}$$

which is the main result for our capacitance sensitivity computation from a theoretical point of view. As it is derived from the *adjoint* of an operator in the electrostatic field, and in many publications related to the sensitivity study ([50, 51]), the term *adjoint system* is used instead of the *auxiliary system*, we will call the proposed method the *adjoint field technique (AFT)* (it follows the term in [50]). In the next section, we will show how to develop (3.28) into an applicable algorithm.

3.2 Algorithm Development

In the following, we will first derive the sensitivity of the short-circuit capacitance C_{sij}. Then the network capacitance sensitivity can be easily calculated using (2.9a).

Suppose there is a system consisting of N conductors, with $\mathbf{V} \in \mathbb{R}^{N\times 1}$ being the vector of voltages on conductors and $\mathbf{C_s} \in \mathbb{R}^{N\times N}$ being the short-circuit capacitance matrix. To compute the sensitivity of a particular capacitance with respect to a certain geometric parameter variation, e.g., $\frac{\partial C_{sij}}{\partial \lambda_p}$, we should first select the specific capacitance C_{sij}.

To locate C_{sij}, let's look at the left-hand side of (3.28), which can be expressed as

$$\begin{aligned}((\Delta\mathbf{C_s})\mathbf{V}, \hat{\mathbf{V}}) &= ((\Delta\mathbf{C_s})\mathbf{V}))^T\hat{\mathbf{V}} \\ &= \left(\begin{pmatrix} \Delta C_{s11} & \cdots & \Delta C_{s1j} & \cdots & \Delta C_{s1N} \\ \vdots & & \vdots & & \vdots \\ \Delta C_{si1} & \cdots & \Delta C_{sij} & \cdots & \Delta C_{siN} \\ \vdots & & \vdots & & \vdots \\ \Delta C_{sN1} & \cdots & \Delta C_{sNj} & \cdots & \Delta C_{sNN} \end{pmatrix}\begin{pmatrix} V_1 \\ \vdots \\ V_j \\ \vdots \\ V_N \end{pmatrix}\right)^T\begin{pmatrix} \hat{V}_1 \\ \vdots \\ \hat{V}_i \\ \vdots \\ \hat{V}_N \end{pmatrix}.\end{aligned} \tag{3.29}$$

As ΔC_{sij} is the one to be studied, the voltages on conductors in the original system are set to be

$$V_j = 1 \quad and \quad V_k = 0 \quad \forall\ k \neq j; \tag{3.30}$$

and the voltages in the adjoint system are set to be

$$\hat{V}_i = 1 \quad and \quad \hat{V}_k = 0 \quad \forall\ k \neq i. \tag{3.31}$$

Therefore,

$$((\Delta\mathbf{C_s})\mathbf{V}_j^d, \hat{\mathbf{V}}_i^d) = \Delta C_{sij} \tag{3.32}$$

where $\mathbf{V}_j^d$ is a vector whose elements are given by (3.30) and $\hat{\mathbf{V}}_i^d$ is a vector whose elements are given by (3.31).

As introduced in Chapter 2, when the BEM is applied, all surfaces of all conductors are discretized into elements (or panels). For simplicity reasons, we use constant elements and the charge distribution and the potential over the conductor surfaces are also piecewise constant. Note that the discretization is identical for the original system and the auxiliary system.

As the short-circuit capacitance matrix is symmetrical, $\Delta\mathbf{C_s} = \Delta\mathbf{C_s}^T$, and using (2.8), the short-circuit capacitance variation ΔC_{sij} can be writ-

ten as

$$\begin{aligned}\Delta C_{sij} &= (\mathbf{V}_j^d)^T(\Delta\mathbf{C_s})^T\hat{\mathbf{V}}_i^d \\ &= (\mathbf{V}_j^d)^T\mathbf{A}^T\Delta\bar{\mathbf{C}}_s\mathbf{A}\hat{\mathbf{V}}_i^d\end{aligned} \tag{3.33}$$

where $\Delta\bar{\mathbf{C}}_s$ is the perturbed partial short-circuit capacitance matrix.

Then let's look at the right-hand side of (3.28): $\langle(\Delta\varepsilon)\mathbf{E}, \hat{\mathbf{E}}\rangle$, which implies that we need to study how the geometric variation λ_p influences the inner product $\int_\Omega(\Delta\varepsilon)\vec{\mathbf{E}}\cdot\hat{\vec{\mathbf{E}}}d\Omega$.

Figure 3.1 schematically shows the cross-section of a conductor when there is a small variation in parameter p (λ_p). S_p is the influenced surface due to λ_p and we call it the *victim surface* incident to parameter p. Analogously, the panels that are located on the victim surface are called the *victim panels*. In fact, the dimensional variation λ_p is exactly the displacement of the victim surface and also that of the victim panels. It is known that in a perfect conductor, $\vec{\mathbf{E}} = 0$ and $\vec{\mathbf{D}} = 0$, thus we obtain

$$\int_\Omega \Delta\varepsilon\vec{\mathbf{E}}\cdot\hat{\vec{\mathbf{E}}}d\Omega = \int_{S_p}\varepsilon\vec{\mathbf{E}}\cdot\hat{\vec{\mathbf{E}}}(\lambda_p\vec{n}_p)\vec{n}_s dS \tag{3.34}$$

where $\vec{n}_s$ is pointing from the victim surface to the medium (characterized by ε) and $\vec{n}_p$ is the direction of the geometric parameter we defined.

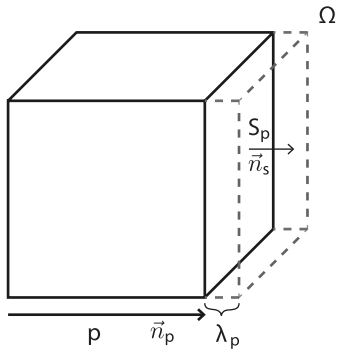

Figure 3.1: *Illustration of the parameter variation λ_p in Ω.*

Thus, (3.28) becomes

$$\begin{aligned}((\Delta\mathbf{C_s})\mathbf{V}, \hat{\mathbf{V}}) &= \langle(\Delta\varepsilon)\vec{\mathbf{E}}, \hat{\vec{\mathbf{E}}}\rangle \\ &= \int_{S_p} \Delta\varepsilon\vec{\mathbf{E}} \cdot \hat{\vec{\mathbf{E}}} d\Omega = \int_{\infty} \varepsilon\vec{\mathbf{E}} \cdot \hat{\vec{\mathbf{E}}}(\lambda_p\vec{n}_p)\vec{n}_s ds \end{aligned} \tag{3.35}$$

Since our goal is to know the induced capacitance fluctuation ΔC_{sij} due to the parameter variation λ_p, conditions (3.30) and (3.31) have to be applied. Hence, considering (3.30), (3.31) and (3.32), it follows from (3.35) that

$$\frac{\Delta C_{sij}}{\lambda_p} = \vec{n}_p\vec{n}_s \int_{S_p} \varepsilon\vec{\mathbf{E}}^d \cdot \hat{\vec{\mathbf{E}}}^d ds \tag{3.36}$$

where $\vec{\mathbf{E}}^d$ and $\hat{\vec{\mathbf{E}}}^d$ are the electric fields under condition (3.30) in the original system and condition (3.31) in the auxiliary system respectively.

While using the piecewise constant shape function for BEM, the field vector $\vec{\mathbf{E}}$ is a piecewise constant quantity on the set of panels S_p. Hence the integral over S_p becomes a summation. Using $\vec{\mathbf{D}} = \varepsilon\vec{\mathbf{E}}$, $\hat{\vec{\mathbf{D}}} = \varepsilon\hat{\vec{\mathbf{E}}}$ and the Gauss law $\nabla \cdot \mathbf{D} = \rho$, (3.36) becomes

$$\begin{aligned}\frac{\Delta C_{sij}}{\lambda_p} &= \vec{n}_p\vec{n}_s\frac{1}{\varepsilon}\sum_{k\in S_p} \vec{\mathbf{D}}\hat{\vec{\mathbf{D}}}A_k \\ &= \vec{n}_p\vec{n}_s\frac{1}{\varepsilon}\sum_{k\in S_p} \rho_k\hat{\rho}_k A_k \end{aligned} \tag{3.37}$$

where A_k is the corresponding area of victim panel k, ρ_k and $\hat{\rho}_k$ are the charge densities on panel k under the two conditions for the original system (3.30) and the adjoint system (3.31) respectively. As the piecewise constance shape function is used for the BEM, the charge density ρ_k can be related to charge q_k with the corresponding area A_k as $q_k = \rho_k A_k$. Since the discretization condition is the same for the original system and the auxiliary system, a similar relation $\hat{q}_k = \hat{\rho}_k A_k$ holds, which leads to

$$\frac{\Delta C_{sij}}{\lambda_p} = \vec{n}_p\vec{n}_s\frac{1}{\varepsilon}\sum_{k\in S_p} \frac{q_k\hat{q}_k}{A_k} \tag{3.38}$$

Then, the sensitivity of the short-circuit capacitance C_{sij} with respect to the parameter variation λ_p can be calculated using (3.38):

$$\frac{\partial C_{sij}}{\partial\lambda_p} = \lim_{\lambda_p\to 0}\frac{\Delta C_{sij}}{\lambda_p} = \vec{n}_p\vec{n}_s\frac{1}{\varepsilon}\sum_{k\in S_p} \frac{q_k\hat{q}_k}{A_k} \tag{3.39}$$

Note that q_k and $\hat{q}_k$ are the charges on panel k given by the conditions (3.30) and (3.31) respectively, which is repeated here:

$$q_k : \quad V_j = 1 \quad and \quad V_k = 0 \quad \forall \ k \neq j; \tag{3.40a}$$

$$\hat{q}_k : \quad \hat{V}_i = 1 \quad and \quad \hat{V}_k = 0 \quad \forall \ k \neq i. \tag{3.40b}$$

Using (2.5), we can calculate

$$q_k = \sum_{a \in N_j} \bar{C}_{sk,a} \tag{3.41a}$$

$$\hat{q}_k = \sum_{b \in N_i} \bar{C}_{sk,b} \tag{3.41b}$$

Subsituting (3.41) into (3.39) and also using (2.9a), we have derived the computation of the coupling capacitance sensitivity with respect to a small geometric parameter variation:

$$\frac{\partial C_{ij}}{\partial \lambda_p} = -\frac{\vec{n}_p \vec{n}_s}{\varepsilon} \sum_{k \in S_p} \frac{1}{A_k} \Big(\sum_{a \in N_j} \sum_{b \in N_i} \bar{C}_{sk,a} \bar{C}_{sk,b} \Big) \tag{3.42}$$

The sensitivity computation of the ground capacitance is very similar to that of the coupling capacitance. In the above discussion, we already calculated the sensitivity of the short-circuit capacitance C_{sij} towards the geometric parameter variation (3.39). Using (2.9b), the ground capacitance sensitivity can be easily derived:

$$\frac{\partial C_{ii}}{\partial \lambda_p} = \frac{\vec{n}_p \vec{n}_s}{\varepsilon} \sum_{k \in S_p} \frac{1}{A_k} \sum_{a \in N_i} \bar{C}_{sk,a} \Big(\sum_{j=1}^{N} \sum_{b \in N_j} \bar{C}_{sk,b} \Big) \tag{3.43}$$

where N is the number of conductors in the system.

For ease of discussion, the default condition will be set to be $\vec{n}_p \vec{n}_s = -1$ in the rest of this thesis, and we introduce a short-hand notation:

$$\mathcal{C}^*_{ki} = \sum_{a \in N_i} \bar{C}_{ska} \tag{3.44}$$

which represents the capacitance between a panel k and a conductor N_i. Hence, the sensitivities for the coupling capacitance and the ground capacitance can be rewritten as

$$\frac{\partial C_{ij}}{\partial \lambda_p} = -\frac{1}{\varepsilon} \sum_{k \in S_p} \frac{\mathcal{C}^*_{ki} \mathcal{C}^*_{kj}}{A_k} \tag{3.45}$$

and

$$\frac{\partial C_{ii}}{\partial \lambda_p} = \frac{1}{\varepsilon} \sum_{k \in S_p} \sum_{j=1}^{N} \frac{\mathcal{C}^*_{kj} \mathcal{C}^*_{ki}}{A_k}. \tag{3.46}$$

Now, we consider a special case that S_p contains only one victim panel k, then (3.45) becomes

$$\frac{\partial C_{ij}}{\partial \lambda_k} = -\frac{1}{\varepsilon} \frac{\mathcal{C}^*_{ki} \mathcal{C}^*_{kj}}{A_k} \tag{3.47}$$

where λ_k is the displacement of this victim panel. This equation (3.47) provides a way to evaluate the capacitance fluctuation (among conductors) induced by a small displacement of ONE panel. It is thus named the *panel sensitivity of capacitance*, denoted by $\mathcal{S}_k$ with k being the associated victim panel. The concept of panel sensitivity and its description (3.47) will be used for the random variation study in Chapter 6.

Until now, we have derived the capacitance sensitivity computation using the *adjoint field technique (AFT)*. It shows that the sensitivity can be expressed in terms of the partial short-circuit capacitances, which are the intermediate data of a standard capacitance extraction using the BEM. Moreover, the descriptions (3.45) and (3.46) indicate that capacitance sensitivities with respect to different parameter variations are simply incident to different sets of victim panels. All the sensitivities towards multiple parameter variations can be computed simultaneously once the associated partial short-circuit capacitances are available. In a nutshell, it shows that both the nominal capacitances and their sensitivities towards multiple parameters can be obtained together with one system solve, which makes the algorithm very efficient.

3.2.1 Illustrative Example

The following example illustrates the algorithm. As shown in Figure 3.2, there are three conductors where each has three BEM panels. Assume that there are two parameter variations, namely d_1 with two associated victim panels p_1, p_2 and d_2 with two associated victim panels p_2, p_3.

For a standard capacitance extraction, the network capacitances can be obtained from certain combinations of the partial short-circuit capacitances. For instance, the coupling capacitance C_{12} can be computed as

$$C_{12} = -\sum_{i \in N_1} \sum_{j \in N_2} \bar{C}_{sij} = -\sum_{i \in N_1} \mathcal{C}^*_{i,N_2} \tag{3.48}$$

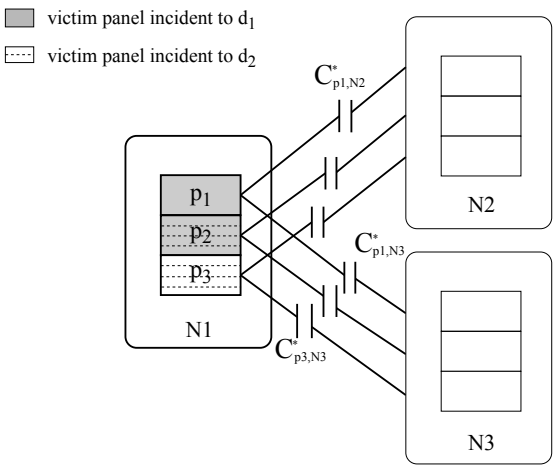

Figure 3.2: *Illustration of $\mathcal{C}^*_{ki}$, where k represents a victim panel and i represents a conductor.*

and the ground capacitance C_{11} can be computed as

$$\begin{aligned} C_{gnd1} &= \sum_{i\in N_1} \sum_{j\in N_1,N_2,N_3} \bar{C}_{sij} \\ &= \sum_{i\in N_1} \mathcal{C}^*_{i,N_1} + \sum_{i\in N_1} \mathcal{C}^*_{i,N_2} + \sum_{i\in N_1} \mathcal{C}^*_{i,N_3} \end{aligned} \tag{3.49}$$

Then, according to (3.45), we can compute the sensitivities of coupling capacitance between conductors, for instance, N_2 and N_3 towards the two parameter variations:

$$\frac{\partial C_{23}}{\partial d_1} = -\left(\frac{\mathcal{C}^*_{p_1N_2}\mathcal{C}^*_{p_1N_3}}{\varepsilon A_1} + \frac{\mathcal{C}^*_{p_2N_2}\mathcal{C}^*_{p_2N_3}}{\varepsilon A_2}\right) \tag{3.50}$$

$$\frac{\partial C_{23}}{\partial d_2} = -\left(\frac{\mathcal{C}^*_{p_2N_2}\mathcal{C}^*_{p_2N_3}}{\varepsilon A_1} + \frac{\mathcal{C}^*_{p_3N_2}\mathcal{C}^*_{p_3N_3}}{\varepsilon A_2}\right) \tag{3.51}$$

Also the sensitivities of the ground capacitances can be computed according to (3.45):

$$\frac{\partial C_{gnd2}}{\partial d_1} = \frac{\mathcal{C}^*_{p1N2}(\mathcal{C}^*_{p1N1} + \mathcal{C}^*_{p1N2} + \mathcal{C}^*_{p1N3})}{\varepsilon A_1} + \frac{\mathcal{C}^*_{p2N2}(\mathcal{C}^*_{p2N1} + \mathcal{C}^*_{p2N2} + \mathcal{C}^*_{p2N3})}{\varepsilon A_2} \tag{3.52}$$

$$\frac{\partial C_{gnd3}}{\partial d_2} = \frac{\mathcal{C}^*_{p2N3}(\mathcal{C}^*_{p2N1} + \mathcal{C}^*_{p2N2} + \mathcal{C}^*_{p2N3})}{\varepsilon A_2} + \frac{\mathcal{C}^*_{p3N3}(\mathcal{C}^*_{p3N1} + \mathcal{C}^*_{p3N2} + \mathcal{C}^*_{p3N3})}{\varepsilon A_3} \tag{3.53}$$

The example illustrates that with only one 3-D capacitance extraction using the BEM, we can obtain the nominal capacitances as well as their sensitivities with respect to multiple geometric variations.

3.3 Analytical Example

In this section, we will discuss an analytical example to support the outcome of the proposed algorithm. To distinguish the result of the analytical calculation and the result given by the proposed algorithm, the following notations will be used:

$S^{\dagger}_{c\_i}$ sensitivity of a *coupling* capacitance towards parameter i given by the *analytical* calculation

$S^{\ddagger}_{c\_i}$ sensitivity of a *coupling* capacitance towards parameter i given by the *proposed* algorithm

$S^{\dagger}_{g\_i}$ sensitivity of a *ground* capacitance towards parameter i given by the *analytical* calculation

$S^{\ddagger}_{g\_i}$ sensitivity of a *ground* capacitance towards parameter i given by the *proposed* algorithm

As shown in Figure (3.3), there are two concentric spheres. We define the inner sphere as conductor 1 and the outer sphere as conductor 2, while Q_i, V_i $(i = 1, 2)$ are the corresponding charges and voltages on them.

Analytically, the capacitance between the two spheres is

$$C_{12} = \frac{4\pi\varepsilon}{(\frac{1}{r_1} - \frac{1}{r_2})} \tag{3.54}$$

where r_1 and r_2 are the radiuses of the two spheres respectively. Thus the derivative of C_{12}, in other words the sensitivity, towards r_1 and r_2 can be easily calculated:

$$S^{\dagger}_{c\_r1} = 4\pi\varepsilon \frac{r_2^2}{(r_2 - r_1)^2} \tag{3.55}$$

$$S^{\dagger}_{c\_r2} = -4\pi\varepsilon \frac{r_1^2}{(r_2 - r_1)^2} \tag{3.56}$$

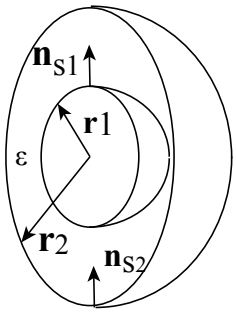

Figure 3.3: *Illustration of the concentric spheres.*

Next, we compute these two sensitivities using the proposed algorithm. Without loss of generality, we consider the inner sphere and the outer sphere to be a single panel each. Thus the area of each panel is $A_i = 4\pi r_i^2$ $(i = 1, 2)$. According to the proposed algorithm (3.39), the sensitivity of C_{12} against r_1 can be computed as

$$S^{\ddagger}_{c\_r1} = -\frac{Q_1 \hat{Q}_1}{\varepsilon A_1} \tag{3.57}$$

where Q_1 and $\hat{Q}_1$ are the charges on the inner sphere under the conditions that $V_1 = 0$, $V_2 = 1$ and $\hat{V}_1 = 1$, $\hat{V}_2 = 0$ respectively. Knowing that

$$Q_1 = -\hat{Q}_1 = -\frac{4\pi\varepsilon}{\frac{1}{r_1} - \frac{1}{r_2}}, \tag{3.58}$$

it is trivial to calculate the sensitivity of the coupling capacitance towards r_1:

$$S^{\ddagger}_{c\_r1} = 4\pi\varepsilon \frac{r_2^2}{(r_2 - r_1)^2} \tag{3.59}$$

which is the same as the analytical result (3.55).

Similarly, the sensitivity of C_{12} w.r.t. r_2 can be calculated using the proposed algorithm, resulting in

$$S^{\ddagger}_{c\_r2} = -4\pi\varepsilon \frac{r_1^2}{(r_2 - r_1)^2} \tag{3.60}$$

which is identical to (3.56).

Regarding the ground capacitance, we study a special case where there is one isolated sphere with a radius of r. Alternatively, it can be considered that the radius of the outer sphere in Figure 3.3 is infinitely large. Analytically, the capacitance to the infinity (the reference ground) is

$$C_{gnd} = 4\pi\varepsilon r \tag{3.61}$$

with a sensitivity towards its radius being

$$S^{\dagger}_{g\_r} = 4\pi\varepsilon \tag{3.62}$$

Using the proposed algorithm (3.43), the ground capacitance sensitivity can be derived

$$\begin{aligned} S^{\ddagger}_{g\_r} &= \frac{1}{\varepsilon}\frac{C^2_{gnd}}{a} \\ &= \frac{1}{\varepsilon}\frac{(4\pi\varepsilon r)^2}{4\pi r^2} = 4\pi\varepsilon \end{aligned} \tag{3.63}$$

which agrees with the analytical result.

So far, we have proposed an efficient algorithm for the capacitance sensitivity computation. It shows that the sensitivity can be obtained by manipulating the intermediate data of the standard capacitance computation using the BEM. To validate its feasibility, the algorithm should be developed further to allow an integration in an existing capacitance extrator, which brings in the next chapter.

CHAPTER 4

Algorithm Implementation and Experiments*

In this chapter, we will first develop the mathematical algorithm presented in Chapter 3 into efficient procedural algorithms. The algorithm can then e.g. be implemented in the SPACE layout-to-circuit extractor using C++ language. A windowing technique is introduced so that the proposed algorithm can be applied to larger structures. Based on such an implementation, a complexity analysis is provided. This is followed by two experiments verifying the accuracy and efficiency of the proposed algorithm. Moreover, the second experiment also demonstrates one possible application of sensitivities, for statistical analysis of capacitances. Finally, another experiment is conducted to show the application of the windowing technique. Comparisons of the accuracy and the efficiency are discussed while various window sizes are used.

*Part of this chapter has been published in [52]: Yu Bi, K.J. van der Kolk and N.P. van der Meijs, "Sensitivity computation using domain-decomposition for boundary element method based capacitance extractors," in *Proc. CICC*, San Jose, CA, pp. 423-426, September, 2009.

4.1 Introduction

As introduced in Chapter 2, the capacitance extraction using the BEM involves a computationally expensive matrix inversion (2.6): $\bar{\mathbf{C}}_s = \mathcal{G}^{-1}$. In the SPACE layout-to-circuit extractor, this operation is performed by a matrix approximation technique, namely the Matrix Schur Interpolation Algorithm. This algorithm for matrix approximation was first proposed in [53]. And the version used in SPACE was originally proposed and motivated in [54]. The work has then been intensively carried out, developed and exploited [55, 56], leading to the design tool SPACE.

The implementation of the matrix approximation in SPACE is operated in a pipeline fashion. That is, a correspondence is maintained between the entries of the elastance matrix $\mathcal{G}$ (or its inversion, $\bar{\mathbf{C}}_s$) and the pair of panels that produced the entry in $\mathcal{G}$. This is illustrated in Figure 4.1, where a second pipeline (a queue or a FIFO data structure) operates in parallel with and synchronous to the matrix inversion pipeline (i.e., the Schur algorithm). When at time t an entry of $\mathcal{G}$, e.g., $\mathcal{G}_{kl}$, is injected in the Schur pipeline, the corresponding pair of panels (p_k, p_l) is injected in the panel queue. At time $t+\delta t$ the entry $\bar{C}_{kl}$ is ejected from the Schur pipeline, and the corresponding pair of panels is ejected from the panel queue. The pair of panels ejected from the panel queue then correspond to the entry of $\mathcal{G}^{-1}$ (i.e., $\bar{C}_{kl}$) that is ejected from the Schur pipeline. Finally, based on the association between the BEM panels and the circuit nodes, the network capacitances can be updated with the partial short-circuit capacitances, which is in fact the implementation of (2.8).

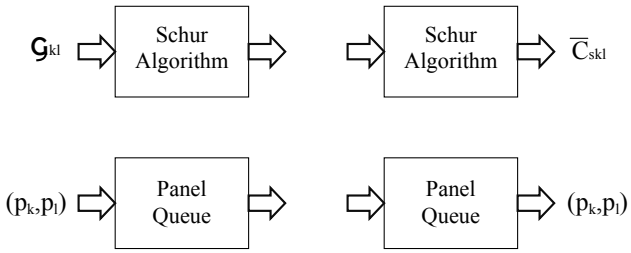

Figure 4.1: *Illustration of the pipeline fashion of the Schur algorithm and the panel queue. (Courtesy of [55])*

The matrix approximation algorithm and its implementation is beyond

the scope of this thesis[†] as the proposed algorithm for sensitivity computation only involves the partial short-circuit capacitances and some physical parameters of the relevant BEM panels. Thus the implementation of the sensitivity computation is mainly carried out after the partial short-circuit capacitances and their corresponding pairs of panels are ejected from the matrix inversion pipeline and the panel queue respectively. One thing that should be noted regarding the Schur algorithm is that it works on the upper triangular part of the elastance matrix only, and generates a low-complexity approximate of the upper triangular part of the partial short-circuit capacitance matrix. The complete matrix is then easy to obtain as it is symmetrical. In fact, only the upper triangular part of the matrix is necessary in the actual implementation, as will be indicated in the next section.

4.2 Procedural Algorithm

The BEM based capacitance extractor SPACE operates by first discretizing all conductor surfaces into panels p_i, $i = 1, ..., m$. The panels are maintained in a linked list such that they can be iterated over by using pointers. Each panel is associated to an electrical circuit node so that the network capacitances can be accumulated from the partial short-circuit capacitances. In addition, each panel is incident to a conductor surface so that the victim panels can be determined according to the interesting geometric parameter variations. These two incidences are realized via pointers, that is, the nodes between which the capacitance or the sensitivity needs to be updated and the surface where each panel is located are identified by two types of pointers, namely *node*() and *surface*().

The main algorithm for the sensitivity computation is shown in Algorithm 1, which operates following the Schur algorithm. In the algorithm, p_k ($k \in 1, ..., m$) represents a panel in the linked list and n_i represents an electrical circuit node.

The COMPUTECAPACITANCE operation computes the nominal network capacitance which is maintained in the original SPACE for the standard capacitance extraction. Note that the "2×" in Line 4, Algorithm 2, comes from the fact that only the upper-triangular part of $\bar{\mathbf{C}}_s$ is generated by the Schur module.

[†]For more information, a survey of the method can be found in [57] and details of the implementation in SPACE can be found in PhD dissertation [55]

Algorithm 1 MAIN

```
1: for (k = 1; k < m; k++) do
2:    for (l = k; l < m; l++) do
3:       computeCapacitance (p_k, p_l, C̄_sk,l)
         {for nominal capacitance computation}
4:       accumulateCstar (p_k, p_l, C̄_sk,l)
5:    end for
6:    if victim(p_k) = TRUE then
7:       for all nodes n_i do
8:          computeSensitivityGnd (p_k, n_i)
9:       end for
10:      for all pairs of nodes (n_i, n_j) do
11:         computeSensitivityCpl (p_k, n_i, n_j)
12:      end for
13:      for all nodes n_i do
14:         DEL (cstar(p_k, n_i))
            {to avoid searching}
15:      end for
16:   end if
17: end for
```

Algorithm 2 computeCapacitance

```
1: if p_k = p_l then
2:    addCapacitance(p_k, p_l, val)
      {diagonal entry of C̄_s}
3: else if node(p_k) = node(p_l) then
4:    addCapacitance (node(p_k), gndNode, 2 × val)
      {both panels belong to the same conductor}
5: else
6:    addCapacitance(node(p_k), gndNode, val)
7:    addCapacitance(node(p_l), gndNode, val)
8:    addCapacitance(node(p_k), node(p_l), val)
9: end if
```

The accumulateCstar operation is to implement Equation (3.44). Each invocation will add one term of the summation, that is one partial short-circuit capacitance $\bar{C}_{sk,l}$ associated with panel k and panel l as shown

in Line 3, Algorithm 1. This value is stored by ADDMAP using a *head*() or a *tail*() pointer as indicated in Algorithm 3 and Figure 4.2.

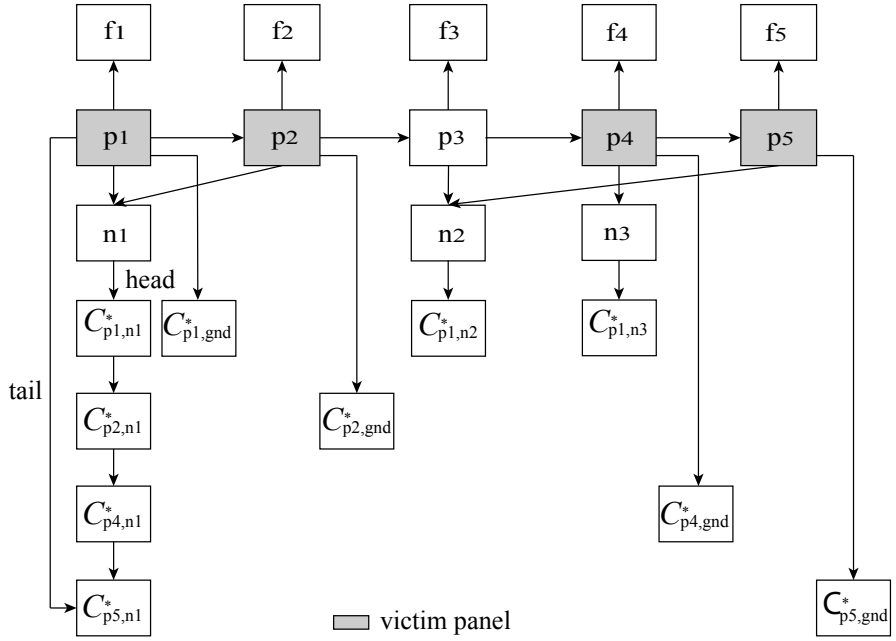

Figure 4.2: *Illustration of the data structure when $k = 1$ loop proceeds till line 12 (before DEL). There are in total 5 panels where p_1, p_2, p_4 and p_5 are the victim panels. They are associated with 5 nodes. f refers to the surface a panel is located.*

The choice between the activation of the *head*() or the *tail*() pointer depends on whether p_k or p_l is under examination. This is done by *victim*(), which tests if the panel argument refers to a victim panel on one of the victim surfaces. Hence, if p_k is a victim panel, the $\bar{C}_{sk,l}$ is added by the *head*() pointer, and if p_l is a victim, the $\bar{C}_{sk,l}$ is added by the *tail*() pointer. Nothing has to be done if neither of the above situation applies. In addition, for the ADDMAP operation, there is no need for a distinction of which victim surface is involved.

Finally, we need to implement Equations (3.45) and (3.46). The operations are executed by the COMPUTESENSITIVITYCPL(p_k, n_i, n_j)

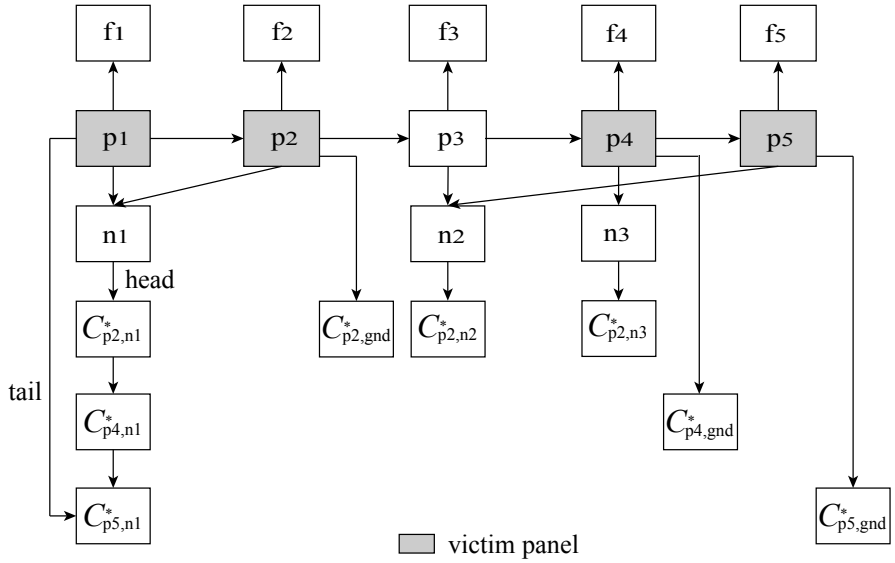

Figure 4.3: *Illustration of the data structure when $k = 2$ loop proceeds till line 12 (before DEL).*

Algorithm 3 ACCUMULATECSTAR (p_k, p_l, val)

1: **if** victim(p_k) = TRUE **then**
2: ADDMAP (p_k, $gndNode$, val)
3: ADDMAP (p_k, $node(p_l)$, val)
{p_k: $head$ pointer}
4: **end if**
5: **if** $k \neq l$ & victim(p_l) = TRUE **then**
6: ADDMAP (p_l, $gndNode$,val)
7: ADDMAP (p_l, $node(p_k)$, val)
{p_l: $tail$ pointer}
8: **end if**

and COMPUTESENSITIVITYGND(p_k, n_i) procedures described in Algorithm 4 and 5 respectively. These operations are invoked after each l-for loop (Line 2 to 5 in Algorithm 1) if the current panel p_k is on one of the victim surfaces. For each invocation, the sensitivity related to p_k is computed, either being the sensitivity of ground capacitance incident to a

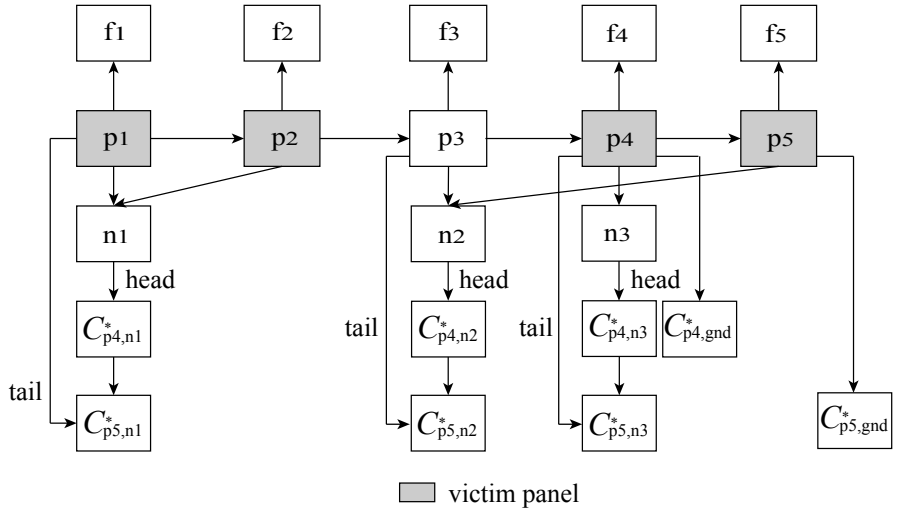

Figure 4.4: *Illustration of the data structure when $k = 4$ loop proceeds till line 12 (before DEL).*

circuit node or the sensitivity of coupling capacitance between two nodes. Special attention should be paid to the fact that since p_k is associated with certain (possibly multiple) geometric parameter variations via the victim surface that it belongs to, the sensitivity value has to be carefully placed. This is accomplished by the ADDSENSITIVITY operation with the help of *surface*().

Algorithm 4 COMPUTESENSITIVITYCPL (p_k, n_i, n_j)

1: $a :=$ area of p_k
2: $S_{ij} :=$ -$cstar(n_i) \times cstar(n_j)/\varepsilon a$
3: ADDSENSITIVITY (n_i, n_j, S_{ij})

Algorithm 5 COMPUTESENSITIVITYGND (p_k, n_i)

1: $a :=$ area of p_k
2: $gndS_i := cstarGnd(p_k) \times cstar(n_i)/\varepsilon a$
3: ADDSENSITIVITY $(n_i, gndNode, gndS_i)$

After COMPUTESENSITIVITYCPL is done for all pairs of nodes, the DEL

operation is called as shown in Algorithm 1 so as to avoid unnecessary searching during the iteration of p_k. And the *head* and *tail* pointers are reset.

Above, we have explained in detail the procedural algorithms for capacitance sensitivity computation, showing that the operations can be integrated in the nominal capacitance extraction. Finally, the sensitivities with respect to multiple geometric variations are outputted together with the nominal capacitances as part of the netlist. An example is shown in Table 4.1 where N_i and N_j present electrical circuit nodes including the ground node, m is the number of layers. l_i, d_i and t_i represent respectively the layout variation of metal layer i, the height variation of dielectric layer i and the thickness variation of metal layer i.

Table 4.1: *A partial output netlist example consisting the nominal capacitances and their sensitivities w.r.t. multiple parameter variations.*

$(N_i,\ N_j)$	C_{ij}	$\frac{\partial C_{ij}}{\partial l_0}$	$\cdots$	$\frac{\partial C_{ij}}{\partial l_m}$	$\frac{\partial C_{ij}}{\partial d_0}$	$\frac{\partial C_{ij}}{\partial t_0}$	$\cdots$	$\frac{\partial C_{ij}}{\partial d_m}$	$\frac{\partial C_{ij}}{\partial t_m}$

4.2.1 Complexity Analysis

As introduced in Chapter 2, the nominal capacitance computation *without* using any acceleration technique involves a matrix inversion:

$$\bar{\mathbf{C}}_s = A^T \mathcal{G}^{-1} A \tag{4.1}$$

where $\mathcal{G} \in \mathbb{R}^{m \times m}$ is the elastance matrix with m being the total number of BEM panels. As $\mathcal{G}$ is a full matrix, it requires $O(m^2)$ time to construct the matrix by evaluating the Green's function and $O(m^3)$ time to perform the matrix inversion.

In this case, the sensitivity computation using the proposed algorithm requires an additional computational time of $O(m^2 + nN^2)$ with n being the number of victim panels and N being the number of electrical circuit nodes. Since $n \leq m$ and normally $N \ll m$, the major computational cost for the sensitivity computation is $O(m^2)$. Compared to the computational complexity of the nominal capacitance computation $O(m^3)$, the additional cost $O(m^2)$ is negligible.

The major memory cost of the nominal capacitance extraction is for the storage of the matrix $\mathcal{G}$, being $O(m^2)$. The extra memory cost for the sensitivity computation includes:

1. The storage for the $\mathcal{C}^*_{p_k,n_i}$ is $O(nN)$;

2. The storage for the capacitance sensitivity outputs $\frac{\partial C}{\partial p_i}$, $i = 1, ..., M$, with M being the number of geometric parameter variations, is

$$O\left(M(\frac{N!}{2!(N-2)!} + N)\right) = O(MN^2) \tag{4.2}$$

Hence, the extra memory complexity for the sensitivity computation is $O(nN + MN^2)$. As we know that $n \leq m$, $N \ll m$ and in most cases $M < m$, the required storage for the sensitivity computation is thus also negligible compared to that for the nominal capacitance computation.

4.3 Window-Scheme

For large circuits, the complexities of both the time consumption and the memory cost derived above are too high to be used in practice. In SPACE, this problem is solved by using the hierarchical Schur algorithm and the window-scheme.

The window-scheme is based on the fact that when two panels are far from each other, their capacitive coupling becomes less significant. In the Matrix Schur Interpolation Method, the least significant capacitive couplings are replaced by ground capacitances in such a way that the resulting approximating matrix is as close as possible to the original matrix. The window size W is a threshold for distinguishing whether this coupling should be replaced or not. It is also a parameter to trade accuracy for efficiency. If the distance between a pair of panels is larger than $2W$, their coupling capacitance will not be counted. Note that here the *distance* W refers to the number of BEM panels rather than the real distance measured in μm. In experiments, we will use the notation w to represent the actual distance of a window in μm.

While solving a positive definite symmetrical matrix with size $\mathcal{N} \times \mathcal{N}$ that is specified on a staircase band with a width of b, the time complexity of the Schur algorithm is $O(\mathcal{N}b^2)$. Thus for each block in the elastance matrix $\mathcal{G}$ (see Figure 4.5), the Schur algorithm costs $O(W\sqrt{m}) \times O((W^2)^2) =$

$O(\sqrt{m}W^5)$ time. As there are $O(\frac{\sqrt{m}}{W})$ blocks in the layout, the total time for solving the system is $O(\sqrt{m}W^5) \times O(\frac{\sqrt{m}}{W}) = O(mW^4)$, which is linear in the size of the layout. Details of the complexity analysis of the

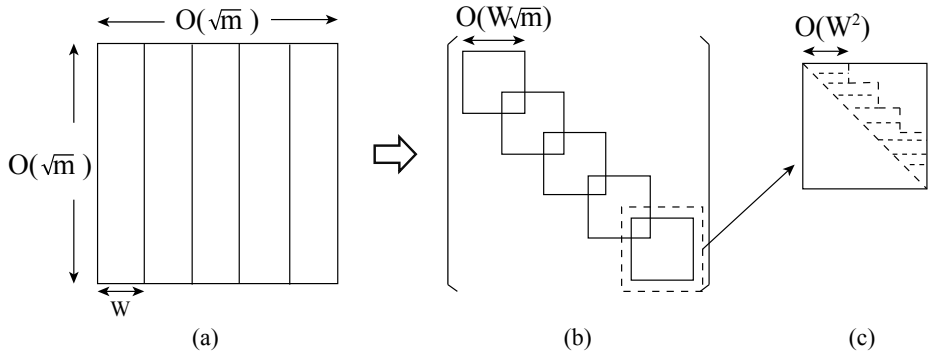

Figure 4.5: *(a) A square layout with m finite elements uniformly distributed; (b) The constructed elastance matrix $\mathcal{G}$ with blocks corresponding to the strips; (c) illustration of the staircase band with in one block.*

hierarchical Schur algorithm in SPACE can be found in [55] Section 4.7.

In Chapter 3 and Appendix C, the sensitivity computation has been derived without considering the window-scheme, or in other words, it is under an assumption that the window size is infinitely large. In the following, we will use an illustrative example to show that the algorithm is still valid when the window-scheme is applied for the capacitance extraction. The induced accuracy lost is also indicated.

As shown in 4.7, there are 4 conductors A, B, C, D, with 2 panels each. Without loss of generality, we assume 4 victim panels, namely p_1, p_3, p_5 and p_7. The space between two neighboring conductors is the same. The window size w is assumed to be a conductor width plus the space between two neighboring conductors, as shown in the figure. Thus, the capacitive coupling between a pair of panels whose distance is larger than $2w$ is considered small enough to be neglected.

First, we consider the situation where the window size is infinitely large.

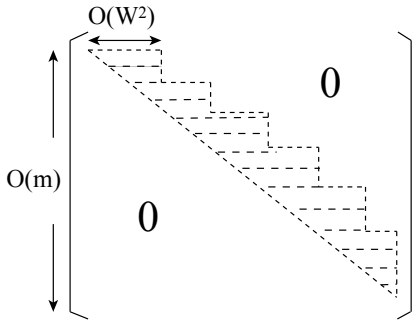

Figure 4.6: *The output of the Schur module, that is the generated low-complexity approximate of the upper triagular part of the partial short-circuit capacitance matrix* $\bar{\mathbf{C}}_s$.

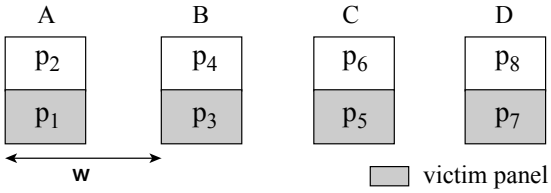

Figure 4.7: *Example to show the window-scheme.*

Hence, the partial short-circuit capacitance matrix $\bar{\mathbf{C}}_{s\_\infty}$ is

$$\bar{\mathbf{C}}_{s\infty} = \begin{bmatrix} \bar{C}_{s11} & \bar{C}_{s12} & \bar{C}_{s13} & \bar{C}_{s14} & \bar{C}_{s15} & \bar{C}_{s16} & \bar{C}_{s17} & \bar{C}_{s18} \\ \bar{C}_{s21} & \bar{C}_{s22} & \bar{C}_{s23} & \bar{C}_{s24} & \bar{C}_{s25} & \bar{C}_{s26} & \bar{C}_{s27} & \bar{C}_{s28} \\ \bar{C}_{s31} & \bar{C}_{s32} & \bar{C}_{s33} & \bar{C}_{s34} & \bar{C}_{s35} & \bar{C}_{s36} & \bar{C}_{s37} & \bar{C}_{s38} \\ \bar{C}_{s41} & \bar{C}_{s42} & \bar{C}_{s43} & \bar{C}_{s44} & \bar{C}_{s45} & \bar{C}_{s46} & \bar{C}_{s47} & \bar{C}_{s48} \\ \bar{C}_{s51} & \bar{C}_{s52} & \bar{C}_{s53} & \bar{C}_{s54} & \bar{C}_{s55} & \bar{C}_{s56} & \bar{C}_{s57} & \bar{C}_{s58} \\ \bar{C}_{s61} & \bar{C}_{s62} & \bar{C}_{s63} & \bar{C}_{s64} & \bar{C}_{s65} & \bar{C}_{s66} & \bar{C}_{s67} & \bar{C}_{s68} \\ \bar{C}_{s71} & \bar{C}_{s72} & \bar{C}_{s73} & \bar{C}_{s74} & \bar{C}_{s75} & \bar{C}_{s76} & \bar{C}_{s77} & \bar{C}_{s78} \\ \bar{C}_{s81} & \bar{C}_{s82} & \bar{C}_{s83} & \bar{C}_{s84} & \bar{C}_{s85} & \bar{C}_{s86} & \bar{C}_{s87} & \bar{C}_{s88} \end{bmatrix}$$

Taking one coupling capacitance C_{AB} and one ground capacitance C_{Agnd} for example and using (C.30) and (C.31), the sensitivities of the two capa-

citances can be calculated as

$$C'_{AB\_\infty} = -\frac{1}{\varepsilon}\left[\frac{(\bar{C}_{s11}+\bar{C}_{s12})(\bar{C}_{s13}+\bar{C}_{s14})}{A_1} + \frac{(\bar{C}_{s31}+\bar{C}_{s32})(\bar{C}_{s33}+\bar{C}_{s34})}{A_3} + \frac{(\bar{C}_{s51}+\bar{C}_{s52})(\bar{C}_{s53}+\bar{C}_{s54})}{A_5} + \frac{(\bar{C}_{s71}+\bar{C}_{s72})(\bar{C}_{s73}+\bar{C}_{s74})}{A_7}\right] \tag{4.3}$$

$$C'_{Agnd\_\infty} = \frac{1}{\varepsilon}\left[\frac{(\bar{C}_{s11}+\bar{C}_{s12})(\sum_{i=1}^{8}\bar{C}_{s1i})}{A_1} + \frac{(\bar{C}_{s31}+\bar{C}_{s32})(\sum_{i=1}^{8}\bar{C}_{s3i})}{A_3} \frac{(\bar{C}_{s51}+\bar{C}_{s52})(\sum_{i=1}^{8}\bar{C}_{s5i})}{A_5} + \frac{(\bar{C}_{s71}+\bar{C}_{s72})(\sum_{i=1}^{8}\bar{C}_{s7i})}{A_7}\right] \tag{4.4}$$

where A_i, $i = 1, 3, 5, 7$ is the area of panel i.

Then, we apply the window-scheme with a window size of w and obtain a partial short-circuit capacitance matrix $\bar{\mathbf{C}}_{s\_w}$:

$$\bar{\mathbf{C}}_{s\infty} = \begin{bmatrix} \bar{C}_{s11} & \bar{C}_{s12} & \bar{C}_{s13} & \bar{C}_{s14} & & & & \\ \bar{C}_{s21} & \bar{C}_{s22} & \bar{C}_{s23} & \bar{C}_{s24} & & & & \\ \bar{C}_{s31} & \bar{C}_{s32} & \bar{C}_{s33} & \bar{C}_{s34} & \bar{C}_{s35} & \bar{C}_{s36} & & \\ & & \bar{C}_{s43} & \bar{C}_{s44} & \bar{C}_{s45} & \bar{C}_{s46} & & \\ & & \bar{C}_{s53} & \bar{C}_{s54} & \bar{C}_{s55} & \bar{C}_{s56} & \bar{C}_{s57} & \bar{C}_{s58} \\ & & & & \bar{C}_{s65} & \bar{C}_{s66} & \bar{C}_{s67} & \bar{C}_{s68} \\ & & & & \bar{C}_{s75} & \bar{C}_{s76} & \bar{C}_{s77} & \bar{C}_{s78} \\ & & & & \bar{C}_{s85} & \bar{C}_{s86} & \bar{C}_{s87} & \bar{C}_{s88} \end{bmatrix}$$

The sensitivities of the two capacitances C_{AB} and C_{Agnd} now become

$$C'_{AB\_w} = -\frac{1}{\varepsilon}\left[\frac{(\bar{C}_{s11}+\bar{C}_{s12})(\bar{C}_{s13}+\bar{C}_{s14})}{A_1} + \frac{(\bar{C}_{s31}+\bar{C}_{s32})(\bar{C}_{s33}+\bar{C}_{s34})}{A_3}\right] \tag{4.5}$$

$$C'_{Agnd\_w} = \frac{1}{\varepsilon}\left[\frac{(\bar{C}_{s11}+\bar{C}_{s12})(\sum_{i=1}^{8}\bar{C}_{s1i})}{A_1} + \frac{(\bar{C}_{s31}+\bar{C}_{s32})(\sum_{i=1}^{8}\bar{C}_{s3i})}{A_3}\right] \tag{4.6}$$

Comparing (4.3) and (4.5), we notice that the accuracy loss of the sensitivity computation using window size w is due to the two missing terms in (4.3) related to victim panel p_5 and p_7, i.e., $\frac{(\bar{C}_{s51}+\bar{C}_{s52})(\bar{C}_{s53}+\bar{C}_{s54})}{A_5}$ and $\frac{(\bar{C}_{s71}+\bar{C}_{s72})(\bar{C}_{s73}+\bar{C}_{s74})}{A_7}$. As addressed, the window size w is used to trade accuracy for efficiency and serves as a threshold. That is, the capacitance of two panels with a distance larger than $2w$ is very small and can be neglected, compared to that of two panels within w. Therefore, the two missing terms are much smaller compared to the remaining two terms and the induced accuracy loss is well acceptable unless a very high accuracy is required. In this case, the window size has to be increased. For the sensitivity computation of the ground capacitance C_{Agnd}, similar analysis can be conducted, giving a similar conclusion.

At last, we will show the complexity of the sensitivity computation under the window-scheme. The generated partial short-circuit capacitance matrix $\bar{\mathbf{C}}_s$, being the output of the Schur module, also has a staircase band width of $O(W^2)$ as shown in Figure 4.6. Since $\bar{\mathbf{C}}_s \in \mathbb{R}^{m\times m}$, the time complexity of the sensitivity computation by the proposed algorithm is $O(mW^2)$, which is also linear in the size of the layout and is negligible compared to that of the nominal capacitance computation.

Regarding the memory cost of the sensitivity computation, the storage for the $\mathcal{C}^*_{p_k,n_i}$ is reduced to $O(n_W N_W)$, where n_W and N_W are the number of victim panels and the number of electrical nodes within a window. Obviously, we have $O(n_W N_W) \ll O(W^4)$. The storage for the sensitivity computation, on the other hand, remains the same as (4.2).

From the above discussion, we may conclude that the proposed algorithm for sensitivity computation is efficient in the sense that both the required CPU time and the memory cost are negligible compared to that of the standard capacitance computation. The high efficiency is an essential feature of the proposed algorithm and can be particularly useful for design exploration and optimization in the early design stage. This will be shown in Chapter 6 and 7 using application examples. Yet, we will first demonstrate the accuracy and the efficiency of the proposed algorithm using experiments in the subsequent sections.

4.4 Experiment I: Accuracy Verification

Experiments in Section 4.4 and 4.5 have been conducted on a 2.66GHz Intel Xeon CPU with 1GB memory. The first experiment is a 2-by-2 inter-

connect structure of which the dimensions are shown in Figure 4.8. Since

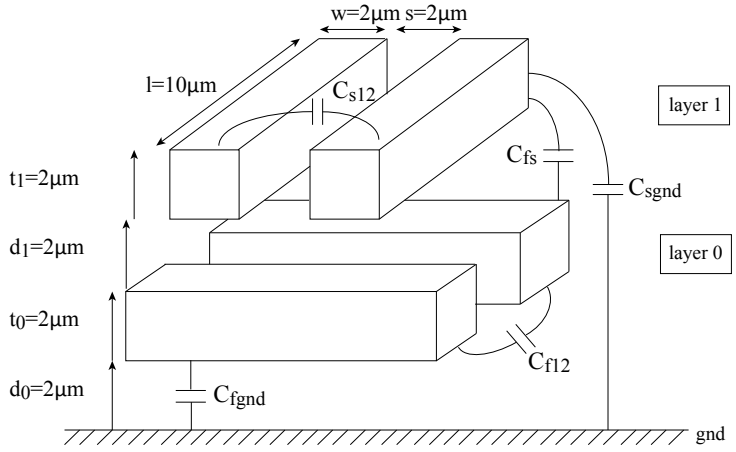

Figure 4.8: *Illustration of a 2-by-2 interconnect structure.*

the structure is symmetrical, three coupling capacitances (C_{f12}, C_{s12}, C_{fs}) and two ground capacitances (C_{fgnd}, C_{sgnd}) are studied. For each layer, we consider variations three parameters, namely the layout dimension (l_i, $i = 0, 1$), the thickness of the metal layer (t_i, $i = 0, 1$) and the height of the dielectric layer (d_i, $i = 0, 1$). Assuming a 10% variation in each parameter, we model the capacitances with 1^{st} order (i.e. linear) approximation using the sensitivities given by the proposed algorithm. Then the dimensions of the structure are changed accordingly by 10% and the extracted capacitances will serve as a reference.

Figure 4.9 shows the comparison between the 0^{th} order and the 1^{st} order approximations where the 0^{th} order is equivalent to the situation in which variability is not accounted for. Several observations can be made:

1. Process variations can not be simply neglected; some can introduce errors of capacitances exceeding 10%.

2. The 1^{st} order approximation improves much over the 0^{th} order approximation. For instance, under a 10% variation in l_0, the 0^{th} order of coupling capacitance C_{f12} gives an error of almost 15%, which drops to 3% using the 1^{st} order approximation.

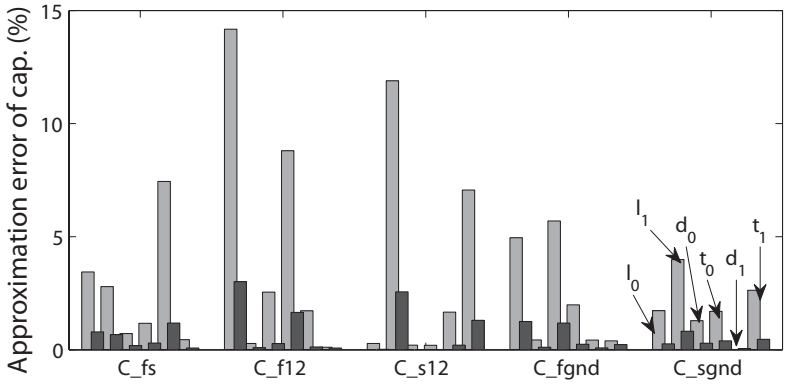

Figure 4.9: *Comparison between 0^{th} order and 1^{st} order approximations. Each group of two bars, one in light grey (0^{th} order approx.) and one in dark grey (1^{st} order approx.), represents the errors of capacitances for one parameter. The six parameters are, in sequence, l_0, l_1, d_0, t_0, d_1, t_1.*

3. The computed sensitivities have an acceptable accuracy indicated by the small errors of the 1^{st} order approximations (the maximum error is less than 3%).

4. For each capacitance, not all parameter variations are influential; some of them are even barely noticeable.

To further show the accuracy of the sensitivity computation, we construct a 2^{nd} order polynomial fit of the extracted capacitances, i.e., $C(p) = a_0 + a_1 p + a_2 p^2$ for every parameter. Then we take its derivative at the nominal dimension p_0, as the reference for sensitivities.

Here we study the sensitivities that are associated with the capacitances with 0^{th} order errors larger than 5%. The average error of these sensitivities compared to the references is 15.16%. This error comes from the fact that only the *moving* panels are considered in our algorithm (see Figure C.1), while the change of the panel size (e.g., the top and the bottom panels of N_1) is not accounted for. While detailed explanation and further improvement of the algorithm will be given in Chapter 5, we would like to give some discussions here. Considering the variations are usually very small for the sensitivity study, the error should be limited. Besides, since the sensitivity itself is a second-order effect to the capacitance, an

accuracy of better than 20% should be good enough for the sensitivity computation for most cases.

4.5 Experiment II: Statistical Interpretation of Sensitivity

In this section, we will illustrate one possible application of sensitivities in statistical analysis. Based on the sensitivities given by the proposed algorithm, we can immediately obtain the standard deviations of capacitances given the process spreads of the technology, i.e. the statistical properties of the geometric parameters. The accuracy is verified by a Monte-Carlo simulation. At last, comparisons of the time consumption are given.

We start by establishing a linear approximation of capacitance matrix $\mathbf{C}$:

$$\mathbf{C} = \mathbf{C}_0 + \sum_{i}^{Q} \lambda_i \mathbf{C}_i \tag{4.7}$$

where matrix $\mathbf{C}_0$ contains the nominal values of capacitances, and $\mathbf{C}_i$ contains the capacitance sensitivities towards the i-th parameter variation λ_i, $i = 1, ..., Q$ with Q being the number of parameters.

Since the geometric parameter variations, namely the variations in the layout, the metal and the dielectric thicknesses of each layer are induced in different process steps, they can be considered to be independent. Thus, with the process spreads of technology, we can derive the statistical property, such as the standard deviation of the capacitances once the capacitance sensitivities are computed. For instance, given the standard deviation of parameters σ_{λ_i}, the standard deviation of the capacitance can be easily computed by

$$\sigma_{C(a,b)} = \sqrt{\sum_{i=1}^{Q} (C_i(a,b)\sigma_{\lambda_i})^2} \tag{4.8}$$

where $C(a, b)$ represents an element in the capacitance matrix.

To check the accuracy of the computed standard deviation of capacitances, we perform a Monte-Carlo simulation on the same 2-by-2 interconnect structure as in the previous experiment. Parameters are assumed to be Gaussian distributed with means of their nominal values and process spreads, i.e. the 3-sigma tolerance being 10% of their nominal values.

1000 capacitance samples are generated using the standard capacitance extraction by SPACE, based on which their standard deviation is derived. Results, used as a reference, are shown in Table 4.2, in comparison to the standard deviations of capacitances given by (4.8). As shown in the table, the results given by the linear model have very good accuracies, which also implies the accuracy of the computed sensitivities. More importantly, it takes only 23 seconds to get the nominal capacitances and their standard deviations using the computed sensitivities, while the Monte-Carlo simulation consumes 21 hours and 43 minutes.

Table 4.2: *Comparison of the standard deviations given by the estimation from Monte-Carlo capacitance samples (left column) and the computation result of the linear model (middle column).*

	Monte-Carlo (F)	Proposed Model (F)	Error
$\sigma_{C_{fs}}$	$8.94e-18$	$8.19e-18$	8.40%
$\sigma_{C_{f12}}$	$25.81e-18$	$23.38e-18$	9.41%
$\sigma_{C_{s12}}$	$27.75e-18$	$25.70e-18$	7.39%
$\sigma_{C_{fgnd}}$	$29.64e-18$	$26.03e-18$	12.19%
$\sigma_{C_{sgnd}}$	$11.60e-18$	$9.89e-18$	14.70%

Next, we will show a larger example to show the induced capacitance fluctuation due to the geometric variations. The experiment is conducted on a 3-metal layer interconnect structure. There are 120 capacitances, 105 being the coupling capacitances and 15 being the ground capacitances. In this case, there are 9 dimensional parameters and in total 1080 capacitance sensitivities. Again we assume the parameters are Gaussian distributed with a 3-sigma being 10% of their mean values.

We compute the 3-sigma for every capacitance according to (4.8). To study the effect of geometric variations on capacitances from a statistical point of view, we partition the range of the 3-sigma which is expressed in percentage of the mean value of each capacitance; and plot the percentage of capacitances in each bin (see Figure 4.10).

While most of the 3-sigma values are less than 15%, we do notice that

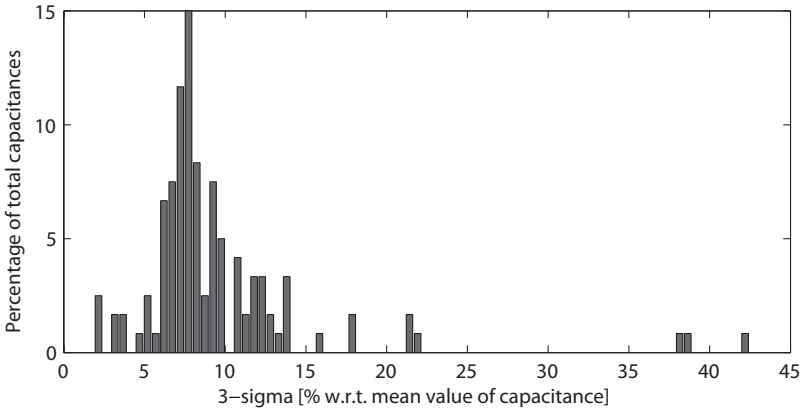

Figure 4.10: *Percentage of total 120 capacitances.*

there are a few of them being around 40%. However, the nominal values of these capacitances are in the order of 10^{-18}, which are small enough, compared to other capacitances being in the order of 10^{-16}, to be neglected.

The total CPU time for this extraction including the sensitivities is $228.6s$. Compared to the time for a standard 3-D extraction on the same configuration being $200.9s$, the additional cost for the sensitivity computation is only $27.7s$, counting for 13.94% of the standard time consumption. In comparison, Cadence uses another technique to construct capacitance sensitivity models for the fast corner generation and 10% extra time is needed to generate sensitivity models per parameter per layer [38]. Hence for their method, it would take in total 90% additional time to generate all the sensitivity models for this structure. The proposed method is much more efficient.

4.6 Experiment III: Windowing Technique

In this section, an experiment is shown to demonstrate the accuracy and efficiency while the windowing technique is applied. This experiment is conducted on a 3.00GHz Intel 2 Core CPU with 4GB memory.

As shown in Figure 4.11, the structure has 2 layers, with 6 interesting dimensional parameters, namely the layout dimensions (l_0, l_1) and the thicknesses (t_0, t_1) of the two metal layers, as well as the heights of the

two dielectric layers (d_0, d_1).

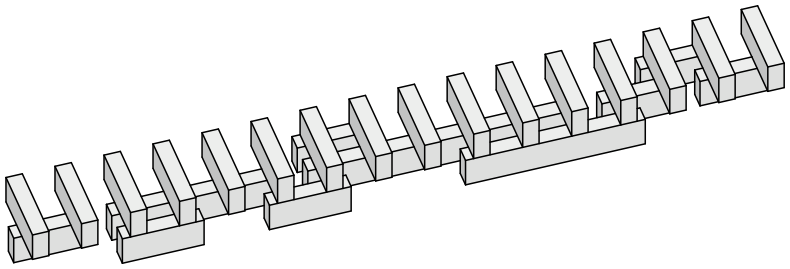

Figure 4.11: *3d Illustration of the experiment for the windowing technique.*

First of all, an extraction is performed with an infinitely large window size, i.e., $w \geq 24\mu m$, and the obtained capacitances and sensitivities are served as references. For ease of discussion, we show in Table 4.3 4 ground capacitances and 10 coupling capacitances with the largest nominal values among 194 capacitances. Their computed sensitivities w.r.t. the 6 parameters are also shown.

Next, we perform extraction on this structure with various window sizes, namely $w = 18\mu m$, $w = 12\mu m$, $w = 8\mu m$ and $w = 5\mu m$, and compare the obtained results to that given by the infinite window size. As representatives, part of the comparison results (errors in percentage) from window size of $12\mu m$ and $5\mu m$ are shown in Table 4.4 and Table 4.5 respectively. The capacitances are the ones shown in Table 4.3.

To save some space, Table 4.6 collects only the largest errors for extractions with window size $18\mu m$ (upper row) and $8\mu m$ (lower row). Clearly, further study is needed for these errors as some of them can reach 50% and even up to 99%.

As addressed, the values of sensitivities represent the impacts on the induced capacitance fluctuations due to the corresponding parameter variations. To show such impact explicitly, we assume a 10% variation for each parameter and compute the induced relative capacitance fluctuation: $\frac{\partial C_{ij}}{\partial \lambda_p} \Delta p / C_{ij}$, where $\frac{\partial C_{ij}}{\partial \lambda_p}$ is the sensitivity of capacitance C_{ij} towards parameter p, and Δp is 10% of its nominal value.

Figure 4.12 shows the induced relative variations of the capacitances with the largest computational errors of sensitivities for each parameter

(left Y-axis). As an indication, these sensitivity errors are also shown in the same figure (right Y-axis). We notice that although some computational errors of sensitivities due to the application of the windowing technique can be high, the corresponding capacitance variations are very low. This is because the corresponding sensitivities themselves are very small. It implies that the associated parameters have little impact on the capacitances of interest. Therefore, the computational errors of these sensitivities can be considered irrelevant, and the windowing technique used for nominal capacitance extraction can also be applied for the sensitivity computation.

Finally, the CPU time consumption for extraction with different window sizes is shown in Table 4.7. Naturally, the smaller the window size is, the less CPU time is needed for the extraction. Note that the CPU time shown includes the computations of both the nominal value of capacitances and their sensitivities w.r.t. 6 parameters.

Table 4.3: *Capacitances and sensitivities given by infinite window size. (unit of capacitances is fF; unit of sensitivity is fF/μm). It shows that for each capacitance (each row of the table), some sensitivities are much larger than the others and some sensitivities are relatively small. It means some parameter variations are more influential for a capacitance while some variations have little impact. Besides, different capacitances may be sensitive to different parameters. Hence, when an acceleration method such as the windowing technique is applied, we need to make sure that the induced computational error of the sensitivity associated with critical parameter variations remain low.*

	Cap.	Sens_l_0	Sens_l_1	Sens_d_0	Sens_t_0	Sens_d_1	Sens_t_1
C_{gnd1}	1.014	0.903	-0.214	-1.130	0.230	0.147	-0.018
C_{gnd2}	0.811	1.200	-0.201	-0.759	0.231	0.060	-0.024
C_{gnd3}	0.735	0.742	-0.171	-0.767	0.194	0.117	-0.014
C_{gnd4}	0.573	1.180	-0.183	-0.381	0.262	0.118	-0.016
C_{cpl1}	0.820	5.980	-0.067	0.154	0.829	0.128	-0.001
C_{cpl2}	0.527	4.000	-0.043	0.100	0.543	0.080	-0.001
C_{cpl3}	0.296	-0.035	1.480	0.009	0.009	0.056	0.299
C_{cpl4}	0.292	-0.036	1.470	0.007	0.007	0.063	0.298
C_{cpl5}	0.284	2.110	-0.027	0.061	0.299	0.048	-0.000
C_{cpl6}	0.282	2.110	-0.027	0.061	0.295	0.049	-0.000
C_{cpl7}	0.281	2.080	-0.026	0.059	0.291	0.048	-0.000
C_{cpl8}	0.281	-0.035	1.420	0.006	0.006	0.054	0.286
C_{cpl9}	0.280	2.090	-0.025	0.058	0.291	0.047	-0.000
C_{cpl10}	0.280	2.100	-0.025	0.057	0.290	0.045	-0.000

Table 4.4: *Errors (%) of capacitances and sensitivities given by window size of $12\mu m$, compared to the results given by the infinite window size. For each parameter, the largest computational error is written in bold and is collected in the bottom-row.*

	Cap.	Sens_l_0	Sens_l_1	Sens_d_0	Sens_t_0	Sens_d_1	Sens_t_1
C_{gnd1}	0.102	2.547	0	2.655	0.870	0	3.889
C_{gnd2}	0.151	1.667	1.493	3.953	2.598	0.167	2.917
C_{gnd3}	0.094	**2.830**	0.585	4.172	2.577	0	1.429
C_{gnd4}	0.118	0	1.093	0	0.382	0	4.375
C_{cpl1}	0.033	0.334	0.597	0	1.568	0	16.30
C_{cpl2}	0.011	0.500	0.930	0.100	2.578	0.125	**42.40**
C_{cpl3}	0.164	0.286	0	1.444	**4.556**	0.179	0
C_{cpl4}	0.043	0.556	0	**4.571**	1.429	0.794	0
C_{cpl5}	0.032	0	1.852	0	0.669	0.625	0
C_{cpl6}	0.150	0	1.482	0.164	0	0.612	0
C_{cpl7}	0.072	0	0.769	0	0.344	0.625	0
C_{cpl8}	0.145	0.857	0	4.000	1.500	0.185	0
C_{cpl9}	0.151	0.479	0.800	0.172	2.749	0.426	0
C_{cpl10}	0.134	0	**5.200**	0.702	0	**0.889**	0
max_val	**0.164**	**2.830**	**5.200**	**4.571**	**4.556**	**0.889**	**42.40**

Table 4.5: *Errors (%) of capacitances and sensitivities given by window size of $5\mu m$, compared to the results given by the infinite window size. For each parameter, the largest computational error is written in bold and is collected in the bottom-row.*

	Cap.	Sens_l_0	Sens_l_1	Sens_d_0	Sens_t_0	Sens_d_1	Sens_t_1
C_{gnd1}	0.153	**7.088**	6.075	4.425	0.870	4.762	21.111
C_{gnd2}	1.328	3.333	**10.448**	6.061	7.792	**7.333**	8.333
C_{gnd3}	0.137	5.256	2.339	4.563	0.516	2.564	31.071
C_{gnd4}	0.656	0.848	6.011	1.312	2.672	3.390	15.000
C_{cpl1}	0.200	1.171	0.597	1.299	6.876	0.000	1.000
C_{cpl2}	0.086	1.500	1.628	0.200	8.840	1.125	**50.100**
C_{cpl3}	0.038	0	0	5.667	8.889	0.714	0.334
C_{cpl4}	0.584	0.833	0.680	**95.000**	**98.859**	3.016	1.342
C_{cpl5}	0.088	0.474	2.593	0.984	3.679	1.042	0
C_{cpl6}	0.267	0.474	1.482	0.492	4.746	0.408	0
C_{cpl7}	0.233	0.481	2.692	2.034	3.780	1.667	0
C_{cpl8}	0.384	1.714	0.000	45.167	48.167	3.889	1.399
C_{cpl9}	0.180	0.479	0.400	0.172	4.467	0.426	0
C_{cpl10}	0.087	0.476	4.000	0.702	0.000	0.889	0
max_val	**1.328**	**7.088**	**10.448**	**95.000**	**98.859**	**7.333**	**50.100**

Table 4.6: *Largest errors (%) of capacitances and sensitivities given by window size of* $18\mu m$ *(first row of the table) and* $12\mu m$ *(second row of the table), compared to the results given by the infinite window size.*

	Cap.	Sens_l_0	Sens_l_1	Sens_d_0	Sens_t_0	Sens_d_1	Sens_t_1
win_18	0.186	1.667	5.200	4.571	4.556	0.889	47.500
win_8	0.361	4.983	5.200	5.310	6.667	2.041	45.400

Table 4.7: *CPU time comparison for various window sizes*

	$w \to \infty$	$w = 18\mu m$	$w = 12\mu m$	$w = 8\mu m$	$w = 5\mu m$
CPU time	$6'27''$	$4'54''$	$2'43''$	$1'24''$	$35''$

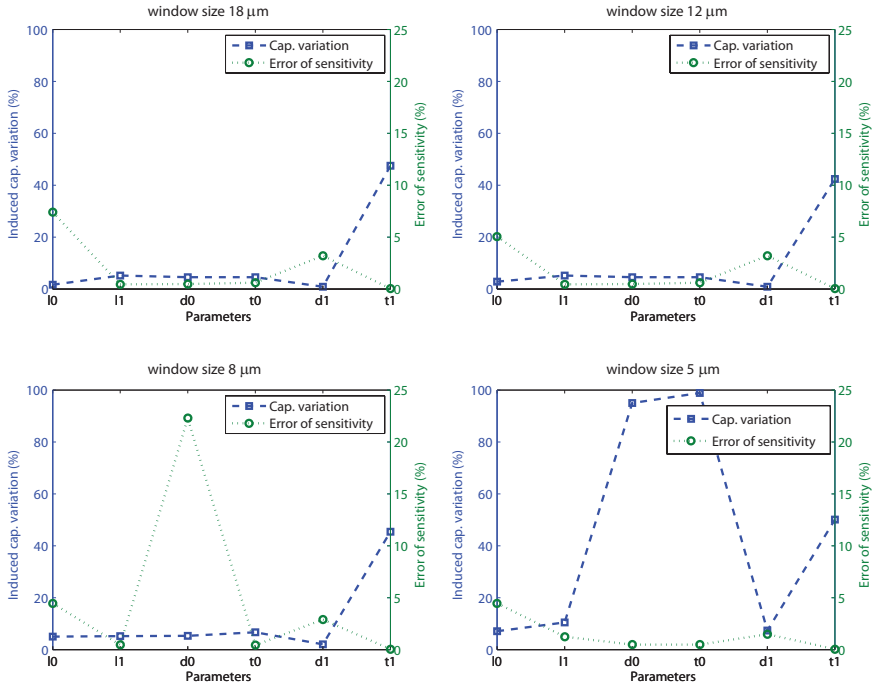

Figure 4.12: *Induced relative capacitance variations (left Y-axis) corresponding to the largest computational errors of sensitivities (right Y-axis) for 4 different window sizes. The assumed parameter variations are* 10% *of their nominal values. It shows that some computational errors of sensitivities due to the application of the windowing technique can be high, but the corresponding capacitance variations are very low, because the corresponding sensitivities themselves are very small.*

CHAPTER 5

Enhanced Computation of the Capacitance Sensitivity*

In the previous chapters, we have presented the computation of capacitance sensitivity by the adjoint field technique. While the proposed algorithm can provide a high efficiency, it has difficulties in achieving a very high accuracy. In this chapter, we will explain the source of this error and provide a method to improve the accuracy of the proposed algorithm.

5.1 Problem Statement

As mentioned, the technique proposed in the previous chapters has a drawback in terms of achieving a very high accuracy. The computational error of sensitivities by this method is mostly in the range of $5\% - 25\%$, depending on the structure of conductors and the geometric parameters of interest.

In order to study the cause of this error, we firstly review the technique briefly. Without loss of generality, we consider in the theoretical derivations only a single parameter p. It being a linear sensitivity based model,

*Part of this chapter has been published in [58]: Yu Bi, S. de Graaf and N.P. van der Meijs,"Enhanced sensitivity computation for BEM based capacitance extraction using the Schur complement technique," in *Proc. CICC*, San Jose, CA, September 2011.

extension towards more parameters is trivial.

Suppose we would like to compute the sensitivity of capacitance between two conductors N_i and N_j w.r.t. a geometric parameter p, i.e., $\frac{\partial C_{sij}}{\partial p}$. Let $a \in N_i$ and $b \in N_j$ denote any panel associated with the two conductors respectively. Also let panel k denote any panel lying on the *moving plate* s_p incident to the parameter p, with A_k being its area and ε being the material permittivity around it. The algorithm proposed in Chapter 3, which will be referred to as the basic algorithm in this chapter, states that the capacitance sensitivity can be computed as

$$\frac{\partial C_{ij}}{\partial p} = -\sum_{k \in s_p} \left(\frac{1}{\varepsilon A_k} \sum_{a \in N_i} \sum_{b \in N_j} \bar{C}_{sk,a} \bar{C}_{sk,b} \right) \tag{5.1}$$

where, as defined already, $\bar{C}_{sk,a}$ and $\bar{C}_{sk,b}$ are the partial short-circuit capacitances associated with panel k, a and panel k, b respectively.

Note that the moving plate is the surface of which the position is moved slightly due to a small variation in parameter p. For instance, assume there is a cubic conductor with a parameter of interest p as shown in Figure 5.1. The moving plate is hence the rightmost sidewall and the panels lying on it are named *moving panels* indicated as the light gray part in the figure. For clarity of discussion in the following, we give a notation of $C'_{plt}(p)$ for the sensitivity $\frac{\partial C_{ij}}{\partial p}$ computed by the basic algorithm (5.1). This description (5.1) shows that sensitivities w.r.t. different parameters are simply incident to different sets of victim panels. All sensitivities w.r.t. multiple parameters can be computed simultaneously once the associated partial short-circuit capacitances are available, i.e., once the standard BEM extraction is done. This is why such BEM-based algorithm for the sensitivity computation can be highly efficient.

However, these moving panels are not the only *victims* due to the parameter variation Δp and they are not the only cause of the capacitance fluctuation. Obviously, the edge panels connected to the moving surface are also influenced by Δp. Their sizes (widths) are either growing or shrinking depending on the direction of Δp, indicated as the hashed part in the figure.

Thus these size-changing edge panels also contribute to the capacitance variation induced by Δp. As will be shown later, neglecting the contribution of these panels is the main reason of the accuracy loss of the capacitance sensitivity computation by the technique presented in the pre-

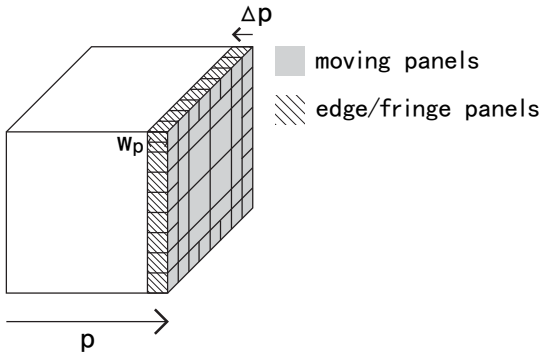

Figure 5.1: *A cubic conductor to demonstrate two parts of contributions to the capacitance fluctuation due to parameter variation Δp. Partial meshing condition is shown.*

vious chapters. In the next section, an extension of this existing technique will be proposed to achieve an improved accuracy by taking into account the influence of the size-changing panels.

5.2 Algorithm Extension by the Schur Complement Technique

As discussed, the capacitance fluctuation is a combined result of the contributions of the moving panels and the size-changing panels. Hence, using the rules of differentiation, the sensitivity w.r.t. a parameter p can be computed as

$$\mathbf{C}'_{tot}(p) = \mathbf{C}'_{plt}(p) + \mathbf{C}'_{frg}(p) \tag{5.2}$$

where $\mathbf{C}'_{tot}(p)$ is the total or the enhanced capacitance sensitivity to be derived and $\mathbf{C}'_{plt}$, given by (5.1), refers to the contribution to the sensitivity from the field lines emanating from the moving plate. $\mathbf{C}'_{frg}(p)$ refers to the contribution from the fringe field emanating from the size changing edge panels on the shortened or elongated side of the conductor. Hence, the main task now is to compute $\mathbf{C}'_{frg}(p)$.

To proceed, we first study carefully the relation between these panels and the parameter variation Δp. Note that if the variation is in the opposite direction of the positive direction of parameter p, as shown in Figure 5.1, and the size (width) of the edge panels is exactly the same

as the value of the variation Δp, these panels can thus be considered disappeared or eliminated due to such a parameter variation. They will be referred to as *fringe panels* in the following context. In other words, the effect of these fringe panels on capacitances can be captured by eliminating their associated entries from the original partial short-circuit capacitance matrix. To do so, it is necessary to let the fringe panels have an identical width (w_p), which can be done by setting appropriate parameters for the mesh generation. This is the basic idea for computing $\mathbf{C}'_{frg}(p)$, a major supplement of the existing sensitivity computation (5.1).

In the following, we will discuss how to develop such basic idea into an implementable algorithm, using the Schur complement technique. For a system not being subjected to process variations, its partial short-circuit capacitance matrix $\bar{\mathbf{C}}_{s\_o}$ is given by the inverse of the elastance matrix $\mathcal{G}_o$ for the originally designed dimensions. To distinguish the fringe panels from the rest of the panels, the $\bar{\mathbf{C}}_{s\_o}$ and the $\mathcal{G}_o$ matrices can be written as block matrices:

$$\bar{\mathbf{C}}_{s\_o} = \begin{pmatrix} \mathbf{Ac} & \mathbf{Bc} \\ \mathbf{Cc} & \mathbf{Dc} \end{pmatrix} \quad \mathcal{G}_o = \begin{pmatrix} \mathbf{Ag} & \mathbf{Bg} \\ \mathbf{Cg} & \mathbf{Dg} \end{pmatrix} \tag{5.3}$$

where $\mathbf{Ac}$, $\mathbf{Ag} \in \mathbb{R}^{n\times n}$ correspond to the n fringe panels to be eliminated, $\mathbf{Dc}$, $\mathbf{Dg} \in \mathbb{R}^{(m-n)\times(m-n)}$ correspond to the rest of the panels. $\mathbf{Bc} = \mathbf{Cc}^T \in \mathbb{R}^{n\times(m-n)}$ and $\mathbf{Bg} = \mathbf{Cg}^T \in \mathbb{R}^{n\times(m-n)}$ describe the connection between these two groups of panels.

Preserving the matrix block dimensions, the relation $\bar{\mathbf{C}}_{s\_o} = \mathcal{G}_o^{-1}$ can be expressed as

$$\begin{pmatrix} \mathbf{Ac} & \mathbf{Bc} \\ \mathbf{Cc} & \mathbf{Dc} \end{pmatrix} = \begin{pmatrix} \mathbf{Ag} & \mathbf{Bg} \\ \mathbf{Cg} & \mathbf{Dg} \end{pmatrix}^{-1} = \\ \begin{pmatrix} \mathcal{S}_{\mathbf{Dg}}{}^{-1} & -\mathcal{S}_{\mathbf{Dg}}{}^{-1}\mathbf{BgDg}^{-1} \\ -\mathbf{Dg}^{-1}\mathbf{Cg}\mathcal{S}_{\mathbf{Dg}}{}^{-1} & \mathbf{Dg}^{-1} + \mathbf{Dg}^{-1}\mathbf{Cg}\mathcal{S}_{\mathbf{Dg}}{}^{-1}\mathbf{BgDg}^{-1} \end{pmatrix} \tag{5.4}$$

where

$$\mathcal{S}_{\mathbf{Dg}} = \mathbf{Ag} - \mathbf{BgDg}^{-1}\mathbf{Cg} \tag{5.5}$$

is the Schur complement of the block $\mathbf{Dg}$ [59].

Next, we write down the Schur complement of block $\mathbf{Ac}$, using (5.4):

$$\begin{aligned}\mathcal{S}_{\mathbf{Ac}} &= \mathbf{Dc} - \mathbf{CcAc}^{-1}\mathbf{Bc} \\ &= \mathbf{Dg}^{-1} + \mathbf{Dg}^{-1}\mathbf{Cg}{\mathcal{S}_{\mathbf{Dg}}}^{-1}\mathbf{BgDg}^{-1} - \\ &\quad (-\mathbf{Dg}^{-1}\mathbf{Cg}{\mathcal{S}_{\mathbf{Dg}}}^{-1}) \cdot \mathcal{S}_{\mathbf{Dg}} \cdot (-{\mathcal{S}_{\mathbf{Dg}}}^{-1}\mathbf{BgDg}^{-1}) \\ &= \mathbf{Dg}^{-1} + \mathbf{Dg}^{-1}\mathbf{Cg}{\mathcal{S}_{\mathbf{Dg}}}^{-1}\mathbf{BgDg}^{-1} - \ \mathbf{Dg}^{-1}\mathbf{Cg}{\mathcal{S}_{\mathbf{Dg}}}^{-1}\mathbf{BgDg}^{-1} \\ &= \mathbf{Dg}^{-1} \end{aligned} \tag{5.6}$$

As addressed, $\mathbf{Dg}$ corresponds to the rest of the panels other than the fringe panels to be eliminated. In other words, it is the elastance matrix for the remaining panels after eliminating the fringe panels. Hence, the Schur complement of $\mathbf{Ac}$, being the inverse of $\mathbf{Dg}$, is the updated partial short-circuit capacitance matrix ($\bar{\mathbf{C}}_{s_{\Delta frg}}$) after the fringe panel elimination:

$$\bar{\mathbf{C}}_{s_{\Delta frg}} = \mathbf{Dc} - \mathbf{CcAc}^{-1}\mathbf{Bc} \tag{5.7}$$

It is exactly what needs to be calculated to further derive the supplement sensitivity $\mathbf{C}'_{frg}(p)$ in (5.2).

Above, we have explained that instead of solving $\mathbf{Dg}^{-1}$ to account for the impact of the fringe panels, we only need to compute the Schur complement of $\mathbf{Ac}$. In fact, from the *internal node elimination* point of view in RC networks, the derivation is quiet basic.

Again, let $\bar{\mathbf{C}}_{s\_o} = \begin{pmatrix} \mathbf{Ac} & \mathbf{Bc} \\ \mathbf{Cc} & \mathbf{Dc} \end{pmatrix}$ be the original partial short-circuit capacitance matrix, and $\mathbf{Ac}$, $\mathbf{Dc}$ correspond to the fringe panels and the rest of the panels respectively. Also let $\mathbf{U_f} \in \mathbb{R}^{n\times 1}$ be associated with the fringe panels, and $\mathbf{U_r} \in \mathbb{R}^{(m-n)\times 1}$, $\mathbf{e_r} \in \mathbb{R}^{(m-n)\times 1}$ be associated with the other panels. We may formulate the following:

$$\begin{pmatrix} \mathbf{Ac} & \mathbf{Bc} \\ \mathbf{Cc} & \mathbf{Dc} \end{pmatrix} \begin{pmatrix} \mathbf{U_f} \\ \mathbf{U_r} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{e_r} \end{pmatrix} \tag{5.8}$$

The above formula (5.8) can be considered as a simplified MNA equation. To eliminate $\mathbf{U_f}$, we rewrite (5.8), obtaining

$$\mathbf{AcU_f} + \mathbf{BcU_r} = \mathbf{0} \tag{5.9a}$$

$$\mathbf{CcU_f} + \mathbf{DcU_r} = \mathbf{e_r} \tag{5.9b}$$

Solving (5.9a) for $\mathbf{U_f}$ leads to

$$\mathbf{U_f} = -\mathbf{Ac}^{-1}\mathbf{Bc}\mathbf{U_r} \tag{5.10}$$

While it can be proved that $\mathbf{Ac}$ is nonsingular, we then substitute $\mathbf{U_f}$ into (5.9b), obtaining

$$(\mathbf{Dc} - \mathbf{CcAc}^{-1}\mathbf{Bc})\mathbf{U_r} = \mathbf{e_r} \tag{5.11}$$

which gives the updated partial short-circuit capacitance matrix after eliminating the fringe panels:

$$\bar{\mathbf{C}}_{s_{\Delta frg}} = \mathbf{Dc} - \mathbf{CcAc}^{-1}\mathbf{Bc}. \tag{5.12}$$

From the updated partial short-circuit capacitance matrix $\bar{\mathbf{C}}_{s_{\Delta frg}}$, we can now first derive the updated short-circuit capacitance matrix $\mathbf{C}_{s_{\Delta frg}}$ with an updated incidence matrix $\mathbf{A}_{\Delta frg}$ analogously to (2.8). Then using (2.9), the updated network capacitances $\mathbf{C}_{\Delta frg}$ can be computed. Finally, using the original network capacitances $\mathbf{C}_o$, the supplement sensitivity that accounts for the effect of the fringe panels incident to parameter p can be derived:

$$\mathbf{C}'_{frg}(p) = \frac{\mathbf{C}_{\Delta frg}(p) - \mathbf{C}_o}{-w_p} \tag{5.13}$$

where the minus sign of w_p comes from the fact that the variation w_p makes the corresponding parameter p smaller (shrinking). For first-order sensitivity study, it is common to assume only one side of the variation against the nominal parameter value, as the variation should be small. In fact, we can consider the above approach for calculating the sensitivity $\mathbf{C}'_{frg}$ as an enhanced FD method.

The cost of the sensitivity extraction is the sum of the cost of extracting $\mathbf{C}'_{plt}(p)$ and $\mathbf{C}'_{frg}(p)$. Note that the cost of computing $\mathbf{C}'_{plt}(p)$ was established to be negligible in section 4.2.1. We now show that the computational cost of computing $\mathbf{C}'_{frg}(p)$ is also small compared to that of the nominal extraction.

In essence, two configurations must be computed as shown in (5.13). The delta configuration is computed by a fast update of the nominal configuration, using a much smaller system of which evaluation of (5.7) forms the main cost. The cost of the update can be estimated as follows. Note that n is the number of fringe panels and m is the total number of panels. In

practical cases, $n \ll m$. Since $\mathbf{A}\mathbf{c}^{-1} \in \mathbb{R}^{n\times n}$ and $\mathbf{B}\mathbf{c} = \mathbf{C}\mathbf{c}^T \in \mathbb{R}^{n\times(m-n)}$, the evaluation of (5.7) is much faster than the evaluation of (2.6), which has a size of $m \times m$ and is the main cost of nominal extraction. In case of n_p parameters, (5.7) must be evaluated n_p times. Hence, the computational cost is linear in n_p. In the next section, a comparison of the accuracy and the efficiency between the proposed enhanced algorithm and the traditional FD method is discussed using experiments.

5.3 Experiment and Results

This section presents an experiment for verifying the accuracy and the efficiency of the enhanced algorithm. The algorithm has been implemented in C/C++ and the experiment has been conducted on a 3.00GHZ Intel Core 2 Duo CPU.

5.3.1 Experiment-1: Sensitivity Computation

As shown in Figure 5.2, the example has 4 layers with 8 cubic conductors. The cubic conductors present some kind of practical worst case situation as for the relevance of the *fringe* terms. Typically, the width of conductors is changed by process variability, and not so much the length. If the length is changed, it can be relevant only if the conductor is very short, which means an almost cubic conductor. Hence this experiment studies the effects of the width variations of cubic conductors. Since the structure is symmetrical, only the widths of the left side cubes (w_1, w_2, w_3, w_4) are studied. Sensitivities given by the traditional FD method are used as references.

Results of the sensitivity computation and its comparison to that of the FD method are shown in Table 5.1, where for each parameter, 2 out of 36 capacitances are selected to demonstrate the accuracy improvement of the enhanced algorithm. These two capacitances are selected mainly based on the effect of the parameter variation of interest, which can be indicated by the magnitude of sensitivities. As shown, the errors of the sensitivities given by the basic algorithm (C'_{plt}) may reach 26%, compared to the reference given by the FD method (C'_{ref}). While it is acceptable for many cases since sensitivity itself is a second order effect to capacitances, the proposed enhanced algorithm can be used when greater accuracy is needed. Indeed, the C'_{tot} rows of Table 5.1 show errors of less than 6%,

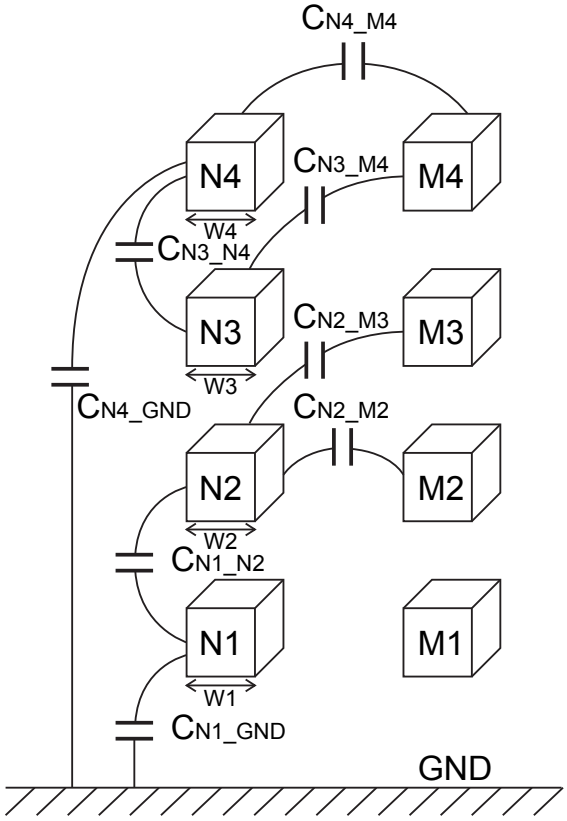

Figure 5.2: *3-D representation of 8 cubic conductors on 4 layers.*

providing a substantial accuracy improvement.

5.3.2 Experiment-2: Variational Study

Using the computed sensitivity, it is also interesting to conduct a variational study on the same structure to show how much is the effect of parameter variations on capacitances, and whether a linear model can capture such effects.

We assume ±30% variations of the nominal value of each parameter. As for the reference, the dimensions of the structure are modified manually by ±30% and a standard extraction is performed to obtain the varied capacitance (C_{var}). As shown in Table 5.2, the resulting variation in capacitance (compared to the nominal capacitance C_{nom}) goes easily beyond 15%, and even 20%. It indicates that process variations can not be simply neglected. An appropriate modeling method needs to be found and applied, for instance, a linear model. With the computed sensitivities (C'_{plt} and C'_{tot}), it is very easy to build the linear model of capacitances, obtaining C_{plt} and C_{tot} (see Table 5.2). It shows that the linear model using C'_{plt} can already capture the variational effect very well, with an error better than 5% for all cases shown. With the enhanced sensitivity C'_{tot}, the linear model is able to further decrease the error. Note that in general, the variation hardly goes up to ±30% for back-end-of-line (BEOL) processing. Thus, for realistic variations, the error of the sensitivity using the proposed algorithm can be even less, given smaller process variations.

With respect to the CPU time, the linear model is much faster than the FD method, as shown in Table 5.3. Note that the CPU time of the linear model (C_{plt} or C_{tot}) includes both the computation of the nominal capacitance (C_{nom}) and the sensitivities (C'_{plt} or C'_{tot}). It indicates that the algorithm for the basic sensitivity computation is extremely fast and results in only a little overhead. And the proposed algorithm for the enhanced sensitivity also provides a competitive efficiency, especially compared to the traditional FD method which requires n_p extra full capacitance extractions given n_p parameters of interest.

5.4 Conclusion

This chapter describes an extension for the sensitivity computation presented in the previous chapters. The extended algorithm is much faster than

the traditional FD approach while providing a similarly high accuracy. The extension serves as a useful and sometimes necessary supplement for the basic algorithm using AFT which features its high speed in generating good accuracy results. The enhanced algorithm thus is able to offer users various solutions for various requirements and applications.

For many practical applications on the other hand, the accuracy requirement for sensitivity computation is not particularly high. And the main feature of our method for capacitance modeling subject to process variations is *high efficiency*. Thus unless stated especially, in the following chapters (in particular the experiments), we will be using the basic algorithm proposed in Chapter 3.

Table 5.1: *Results of experiment-1. Comparison of the sensitivity computation by different techniques. C'_{ref} is the reference of the computed sensitivity given by the FD method. Errors of C'_{plt}, being the sensitivity given by the basic algorithm, may reach 26%. The enhanced sensitivity presented in this chapter C'_{tot} provides a substantial accuracy improvement and can be used when greater accuracy is needed.*

	parameter: w1		parameter: w2	
	N1-GND	N1-N2	N2-M2	N2-M3
C_{nom} (fF)	0.1969	0.0746	0.0666	0.0194
C'_{ref} (fF/um)	0.1665	0.0548	0.0804	0.0188
C'_{plt} (fF/um)	0.1297	0.0427	0.0717	0.0148
error	-22.08%	-22.05%	-10.48%	-21.30%
C'_{tot} (fF/um)	0.1687	0.0572	0.0849	0.0191
error	1.34%	4.35%	5.57%	1.58%
	parameter: w3		parameter: w4	
	N3-N4	N3-M4	N4-GND	N4-M4
C_{nom} (fF)	0.0798	0.0228	0.1496	0.0844
C'_{ref} (fF/um)	0.0594	0.0217	0.1207	0.0901
C'_{plt} (fF/um)	0.0440	0.0170	0.0927	0.0789
error	-25.99%	-21.63%	-23.20%	-12.44%
C'_{tot} (fF/um)	0.0603	0.0220	0.1209	0.0943
error	1.44%	1.30%	0.17%	4.64%

Table 5.2: *Results of experiment-2: Variational study of capacitances using the linear model. C_{var} is the reference of the varied capacitance induced by ±30% parameter variations. C_{plt} and C_{tot} is obtained from the linear model of capacitances using sensitivities C'_{plt} and C'_{tot} respectively. It shows that C_{plt} can already capture the variational effect pretty well and C_{tot} which uses the enhanced sensitivity is able to further decrease the error.*

	parameter variations: -30%			
	N1-N2	N2-M2	N3-N4	N4-M4
C_{nom} (fF)	0.0746	0.0666	0.0798	0.0844
C_{var} (fF)	0.0660	0.0548	0.0706	0.0712
variation in C	-11.53%	-17.72%	-11.53%	-15.64%
C_{plt} (fF)	0.0682	0.0558	0.0733	0.0726
error	3.29%	1.82%	3.81%	2.02%
C_{tot} (fF)	0.0660	0.0539	0.0708	0.0703
error	0.00%	-1.79%	0.38%	-1.22%
	parameter variations: +30%			
C_{var} (fF)	0.0822	0.0808	0.0882	0.0999
variation in C	10.19%	21.32%	10.53%	18.36%
C_{plt} (fF)	0.0810	0.0773	0.0864	0.0963
error	-1.41%	-4.29%	-1.94%	-3.63%
C_{tot} (fF)	0.0832	0.0793	0.0889	0.0986
error	1.23%	-1.84%	0.84%	-1.32%

Table 5.3: *Results of experiment-2: CPU time comparison of the capacitance variational study. The CPU time of the linear model (Cap_basic and Cap_enhanced) includes both the computation of the nominal capacitance (C_nominal) and the sensitivities ($C^{'}_{plt}$ and $C^{'}_{tot}$). The linear model, even given by the enhanced algorithm, is much faster than the traditional FD method which requires n_p extra full capacitance extractions given n_p parameters of interest.*

	Cap_nominal	Cap_basic	Cap_enhanced	Cap_FD
CPU Time	37.33″	37.69″	41.25″	186.69″
	(1×)	(1.01×)	(1.11×)	(5.00×)

CHAPTER 6

Efficient Sensitivity-Based Capacitance Modeling for Systematic and Random Variations*

In the first part of this chapter, a statistical modeling method of capacitances based on the panel sensitivities is proposed to capture the impact of the random geometric variation line-edge-roughness (LER). A real case study with measurement data is then discussed to demonstrate its application.

The second part of this chapter shows that using the panel sensitivities, the nominal parasitic capacitances and their relative standard deviations caused by both the systematic and random geometric variations can be

*Part of this chapter has been published in [60]: Yu Bi, P.J.A. Harpe and N.P. van der Meijs, "Efficient sensitivity-based capacitance modeling for systematic and random geometric variations," in *Proc. ASP-DAC*, Yokohama, Japan, pp. 61-66, January, 2011, and [61]: P.J.A. Harpe, C. Zhou, Yu Bi, N.P. van de Meijs, X. Wang, K. Philips, G. Dolmans and H. de Groot, "A 26μw 8 bit 10 MS/s Asynchronous SAR ADC for low energy radios," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1585-1595, 2011.

obtained with a single system solve.

6.1 Copper Damascene and Random Geometric Variations

In the past, feature sizes were sufficiently large so that the roughness of printed lines or the irregularity on sidewalls was a small fraction of the critical dimension error budget and the overall performance of the designed circuits was not affected. However, situations have changed since the technology nodes reached deep sub-micron dimensions. The line-edge roughness (LER), a typical random geometric variation, has been intensively studied for the critical dimensions of FETs as its impact cannot be neglected anymore [26, 62]. With technology shrinking, the impact of the roughness on interconnects and some novel designs of precision passive components also becomes important and demands to be understood and modeled.

Recently, several methods were proposed for modeling random geometric variations of on-chip interconnects. A perturbation based 3D BEM solver [63] and Hermite polynomial chaos technique based approaches [64], [65] and [66] have assessed the effects of variational surfaces on interconnect capacitances by generating quadratic models.

Unfortunately, examples demonstrated in these papers are general cases based on theoretical assumptions, which is advantageous in indicating the robustness of the methods. The obtained results, however, consequently provide less guidance for real circuit designs. The variational surfaces of interconnects, for example, are modeled using spatially correlated Gaussian distribution in 3-dimensions. This implies that the irregularity on various surfaces of interconnects, including the top, bottom surfaces and the sidewalls, are correlated. However, studies on the formation of surface roughness and measurement results suggest that roughness on different surfaces should be considered independent [62, 67–71]. In fact, the term *surface roughness* is usually referred to the roughness on the top and bottom surfaces [72, 73], while the roughness on the sidewalls is called the *sidewall roughness* [68, 71].

To understand the formation of the sidewall roughness of interconnects, we first briefly describe the copper damascene process, especially the steps related to the roughness formation [74].

The most widely adopted Cu-damascene process for today's techno-

logies is the *via-first* approach. Figure 6.1 shows the main steps of this approach.

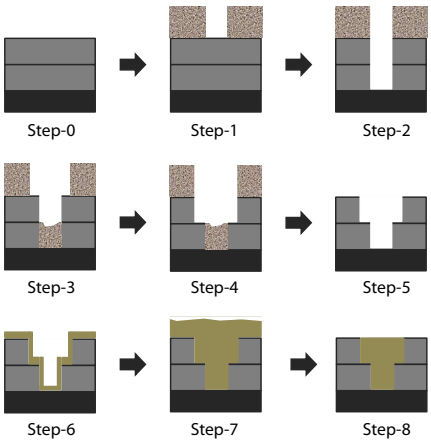

Figure 6.1: *The schematic of a typical copper damascene process after [75].*

Step-1 The wafer is coated with photoresist and patterned lithographically for the via structure.

Step-2 An *anisotropic* etch is performed to etch down through the two dielectric layers.

Step-3 This via photoresist layer is removed and another photoresist layer is placed, patterned lithographically for the trench structure. Some of the photoresist remains at the bottom of the via to prevent the lower portion of the via from being over-etched during the following trench etching.

Step-4 The trench is formed by another *anisotropic* etching which cuts through the upper dielectric layer.

Step-5 The photoresist is removed.

Step-6 The via hole and the trench are lined with a conductive barrier layer to prevent Cu diffusion. Then, a thin seed-layer of Cu is deposited. Both layers should be deposited as *conformally* as possible.

Step-7 The bulk copper is deposited

Step-8 Chemical-mechanical polishing (CMP) is performed to planarize the copper to the surface of the trenches.

As the feature size has become continuously smaller ($< 100nm$), the thickness versus width ratio, i.e. the aspect ratio, is getting much higher. Hence a very fine *anisotropic* etching is desired during the trench formation. This would ensure that the pattern transfer from the photoresist to the underlying dielectric layer is sharp and clear. Thus the line-edge roughness of the resist layer which is caused by the imperfect lithography and etching processes, becomes a template for the emerging dielectric layer. It forms, in the end, striations on the sidewall [68].

Following the etching step, the depositions of the barrier layer and the Cu-seed layer have to be conformal enough to later form a void-free Cu interconnect with a high aspect ratio. As a result, the striations on the dielectric sidewall transfer to the sidewall of the Cu-interconnect. Therefore, the problem of capturing the irregularity of the interconnect sidewall is in fact a problem of modeling the line-edge roughness.

6.2 Statistical Model of LER

The LER can be modeled with a sequence of random variables ρ, representing the fluctuation behavior along the line which in this context is defined as the y-direction (see Figure 6.2). The fluctuation itself is in the direction orthogonal to y-direction, defined as the x-direction. The random variables are often assumed to be Gaussian spatially correlated along the y-direction [26, 70], described as

$$cov(\rho_i, \rho_j) = \sigma_{LER}^2 exp(-\frac{\|\mathbf{r}_{i,y} - \mathbf{r}_{j,y}\|^2}{\eta_{LER}^2}) \tag{6.1}$$

where $\mathbf{r}_{i,y}$ and $\mathbf{r}_{j,y}$ are the y-coordinates of the positions associated with ρ_i and ρ_j respectively. σ_{LER} and η_{LER} are two parameters that characterize the LER. σ_{LER} is the standard deviation representing the absolute roughness amplitude orthogonal to the line-edge, i.e. in x-direction, and η_{LER} is the correlation length along the line-edge, i.e. in y-direction, as shown in Figure 6.2. One can see from (6.1) that the closer two variables locate, the stronger they are correlated. The correlation length is a parameter that measures the strength of the correlation between two variables: if the

distance (in y-direction) between them is much larger than η_{LER}, they can be considered independent.

6.3 Physical Description of LER: Random Line Pattern

Although the physical description of LER, i.e. the line pattern, is not needed for the capacitance modeling method to be presented in the next section, it is necessary for conducting Monte Carlo simulations for verification purposes. In this chapter, the same approach presented in [26] is used to physically capture the LER. This is done by producing random line patterns generated from the inverse Fourier transform of the power spectrum of the Gaussian autocorrelation function. Then these random line patterns can be used for approximating the LER when Monte Carlo simulations are performed as a comparison with the proposed method.

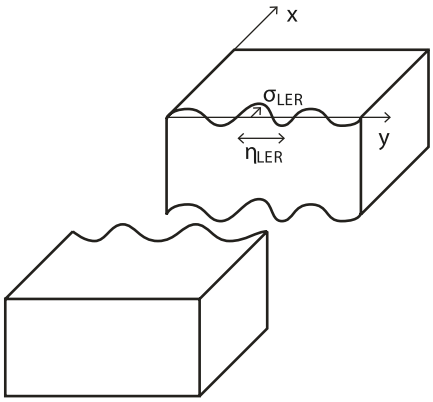

Figure 6.2: *Illustration of the LER on two parallel conductors.*

6.4 Statistical Model of Capacitances Using Panel Sensitivities

The statistical model of LER can be naturally applied to the BEM. The random variables can even be directly adopted as the displacement of the

corresponding BEM discretized panels. These panels will be called *deviation panels* throughout this chapter. Each rough line corresponds to a set of deviation panels. Different sets of deviation panels incident to different rough lines are independent.

A linear model for BEM-based extraction tools is thus proposed to capture the effect of LER on capacitances:

$$\Delta C = \sum_{l=1}^{L} \sum_{i=1}^{n_l} \mathcal{S}_i \rho_i \tag{6.2}$$

where ΔC is the capacitance variation induced by the LER; $\mathcal{S}_i$ is the panel sensitivity given by (3.47) and is associated with the panel displacement ρ_i; n_l is the number of deviation panels for rough line l while L is the number of rough lines.

Due to the linearity of (6.2), the variance of ΔC can be calculated as

$$\begin{aligned} \mathrm{var}(\Delta C)_{LER} &= \mathrm{var}(\sum_{l=1}^{L} \sum_{i=1}^{n_l} \mathcal{S}_i \rho_i) \\ &= \sum_{l=1}^{L} \left[\sum_{i=1}^{n_l} \mathcal{S}_i^2 \mathrm{var}(\rho_i) + 2 \sum_{i,j:i<j} \mathcal{S}_i \mathcal{S}_j \mathrm{cov}(\rho_i, \rho_j) \right] \end{aligned} \tag{6.3}$$

where $var(\rho_i)$ is the squared standard deviation of ρ_i in x-direction, i.e. σ^2_{LER} , and the covariance $cov(\rho_i, \rho_j)$ follows the correlation function (6.1). Note that this statistical model of capacitances (6.3) can be easily adjusted or extended if the model (6.1) of the geometric variations changes. Thus the proposed method is not limited to Gaussian distribution.

From the key Equation (6.3), it follows that the statistical property of the deviated capacitance due to the LER can be easily obtained using the panel sensitivities introduced in Chapter 3 and described in (3.47). Also note that the computational burden of (6.3) includes two parts, namely the calculation of the panel sensitivity $\mathcal{S}_k$ and the calculation of the covariance $cov(\rho_i, \rho_j)$. As has been addressed in Section 4.2.1, the panel sensitivity $\mathcal{S}_k$ is computed from $\mathcal{C}^*_k$ which is an accumulation of partial short-circuit capacitances incident to a conductor. These partial short-circuit capacitances $\bar{\mathbf{C}}_s$ are the intermediate data for calculating the nominal capacitance (2.8), thus they are already computed for the standard capacitance extraction. The time consumption for accumulating $\mathcal{C}^*_k$, i.e. the calculation of $\mathcal{S}_k$ is

negligible compared to a system solve for the nominal capacitance extraction.

As for $cov(\rho_i, \rho_j)$, the computational complexity is related to the correlation length. Within the distance of a correlation length, a certain number of panels are required in order to be able to physically approximate the roughness, which can not be avoided. However, since the correlation $cov(\rho_i, \rho_j)$ decays rapidly as the ratio of $\|\mathbf{r}_{i,y} - \mathbf{r}_{j,y}\|$ and η_{LER} increases, the computational complexity can be greatly reduced by calculating only the non-negligible elements in the double summation $\sum_{i,j:i<j} \mathcal{S}_i \mathcal{S}_j cov(\rho_i, \rho_j)$. Meanwhile, the longer the structure of interest is (in y-direction) compared to the correlation length, the more panels are needed and thus the longer the computational time will be. However, as can be seen in Section 6.5.2, once the dependence of the statistical property of a capacitance on the size of a structure is acquired, quick estimates can be made without having to simulate the complete structure.

6.5 Verification and Experiment

This section presents experiments to first verify the accuracy and the efficiency of the proposed method, and then explore its potential applications. It is common to use the relative standard deviation $\frac{\sigma_C}{C}$, which is often referred to as "mismatch" by designers, to model the effect of LER on capacitances. Hence, we will evaluate the proposed model using $\frac{\sigma_C}{C}$. The method has been implemented in the C/C++ language and the experiment has been conducted on a 3.00GHz Intel Core 2 Duo CPU.

6.5.1 Experiment I

As shown in Figure 6.2, there are two parallel conductors with the two sidewalls facing each other subject to the LER. As addressed earlier, the impact of LER depends largely on the values of σ_{LER} and η_{LER}. These values are closely related to the materials and the manufacturing process. Thus in this experiment, the two parameters have been chosen according to the measurement data of Cu wires in meander-fork structures, provided by IMEC [70]:

$$\begin{aligned} \sigma_{LER} &= 3.5nm \\ \eta_{LER} &= 16nm \end{aligned} \tag{6.4}$$

Table 6.1: *Simulation results and CPU time for modeling LER*

	$\frac{\sigma_C}{C}$	Error	CPU Time
Proposed model	0.603%	11.5%	$50''$
MC simulation	0.681%	0	$48653''$

The width/space of the conductors is $80nm/80nm$. The thickness is $100nm$ and the length is $200nm$. Using the proposed method, we can easily calculate $\sigma_C (= \sigma_{\Delta C})$ from (6.3).

To verify the result of the modeling method, a Monte Carlo simulation with 1000 samples is performed as a reference. For each sample, the random line pattern describing the LER is generated as explained in Section 6.3. The simulation results and the CPU time consumption of the proposed method and the MC simulation are shown in Table 6.1.

The table shows that the error of $\frac{\sigma_C}{C}$ given by the proposed model is around 10%. The error may come from two parts. One is the computational error of the panel sensitivities. The other one is due to the fact that the line-edge roughness is modeled using piecewise constant function, thus the variational surface is not smooth [66]. The error could be reduced by using a piecewise linear function, which is beyond the scope of this thesis. In fact, a 10% error for modeling mismatch is well acceptable in most cases since the mismatch is already small compared to the nominal value. Thus, the introduced error is a second-order effect.

With respect to the CPU time, the proposed model is almost 1000 times faster than the MC simulation. More importantly, the CPU time in the table already includes the computation for the nominal value of the capacitance. With a simple calculation, it follows that the computational time for one system solve is $48.653s$ $(= 48653s/1000)$, while the additional time for the σ_C calculation using the statistical model is only 2.77% $(= (50s - 48.653s)/48.653s)$. Hence, the proposed modeling method is extremely fast, and results in little overhead.

6.5.2 Experiment II

Using the proposed model, one can easily study:

1. the relationship between $\frac{\sigma_C}{C}$ and the conductor length;
2. the impact of parameters σ_{LER} and η_{LER} on $\frac{\sigma_C}{C}$.

Figure 6.3 shows five examples of mismatch $\frac{\sigma_C}{C}$ as a function of the conductor length using the same structure as in Experiment I. Each plot is generated with a combination of various σ_{LER} and η_{LER}.

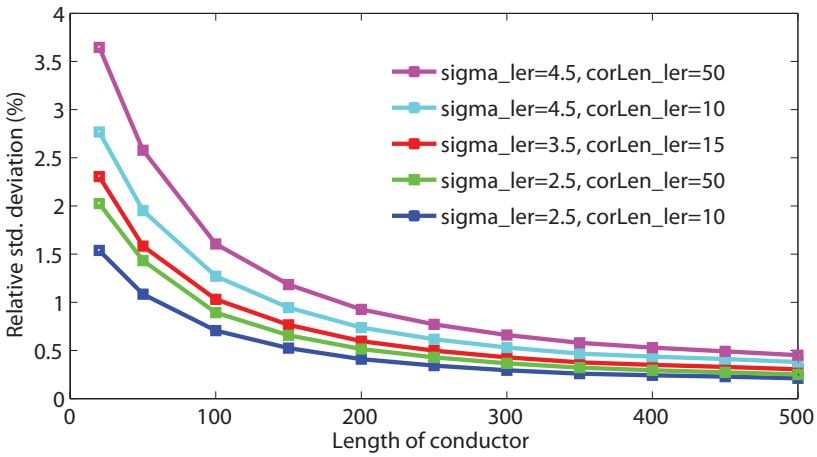

Figure 6.3: *The mismatch ($\frac{\sigma_C}{C}$) versus conductor length with various LER parameter combinations.*

All five plots indicate that the mismatch drops rapidly with the increase of the conductor length. Being able to identify such a trend is very helpful for certain designs with particular variability requirements. Sometimes, for instance, it is necessary to find a good tradeoff between high accuracy (i.e. small mismatch thus longer conductor length) and low power consumption (i.e. small layout area thus shorter conductor length). Then, the proposed modeling method provides a convenient tool to estimate the expected mismatch.

However, analyzing only five combinations of σ_{LER} and η_{LER} is far from enough for a real mismatch analysis. This is because these parameters are highly technology-dependent, thus any change in the manufacturing process in the fab could result in different estimates of them. Besides, the measurement-based estimation is normally given as a range but not a specific value. Hence the statistical modeling of mismatch should use two *sweeping* parameters (σ_{LER} and η_{LER}) instead of two particular values.

Figure 6.4 shows the simulation results of mismatch with sweeping parameters using the proposed method, which costs about an hour. For MC simulation with only a modest number of 1000 samples per data point, this would have taken 43 days.

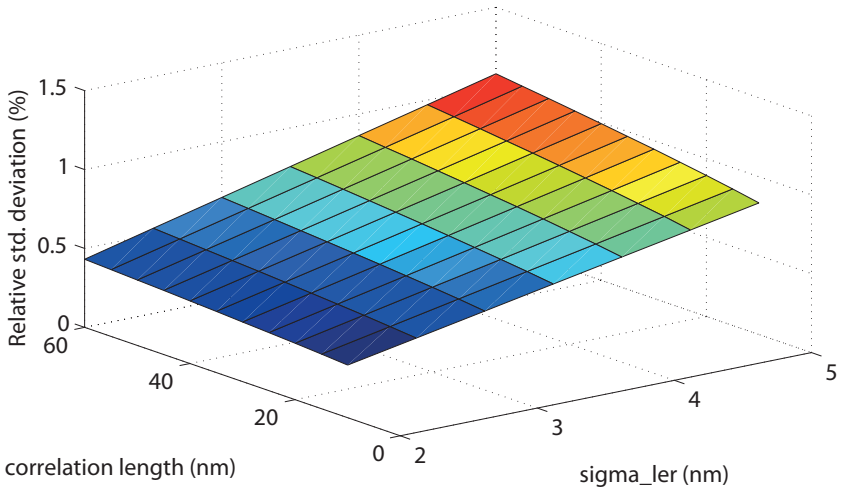

Figure 6.4: *The mismatch* ($\frac{\sigma_C}{C}$) *versus* σ_{LER} *and* η_{LER} *for a pair of* 200nm *long conductors.*

This experiment shows that the proposed modeling method provides a fast and practical tool for circuit designers to estimate mismatches and optimize dimensions of critical structures accordingly.

6.6 A Case Study

Following the experiments in the previous section, this section presents a real design case that can benefit from the proposed statistical modeling method. This is an 8-bit binary-scaled differential charge-redistribution digital-to-analog converter (DAC), a component of a low power SAR ADC design [61]. For ease and clarity of discussion in the following, we first introduce some basic concepts and notations of DA converters that relate to our study.

Least-significant bit (LSB) In a binary number, the LSB is the least weighted bit in the number.

Most-significant bit (MSB) In a binary number, the MSB is the most weighted bit in the number. Typically, binary numbers are written with the MSB in the left-most position and the LSB in the furthest-right bit.

Major-carry transition At the major-carry transition (around mid-scale), either the MSB changes from low to high and all other bits change from high to low, or the MSB changes from high to low and all other bits change from low to high. For example, 01111111 to 10000000 is a major-carry transition.

Differential nonlinearity (DNL) The deviation of two adjacent code analog values from the ideal 1-LSB step.

The designers have implemented the differential 8-bit binary-scaled capacitors $C_{7,\ldots,0}$ with a differential capacitor array consisting of 510 $(2 \times (2^8 - 1))$ identical unit capacitors. Figure 6.5 shows a partial layout of one side of the capacitor array. The value of the unit capacitor has to be minimized to reduce the analog power consumption. Thus a metal-metal unit capacitor, as illustrated in Figure 6.6, with an extremely small value of $0.5fF$ is proposed by the designers.

The major concern regarding this implementation is the mismatch of these unit capacitors due to their very small nominal values. Before evaluating the actual mismatch of this implementation, we first discuss the requirement of the capacitor matching for this design. The unit capacitors are modeled with a nominal value of C_u and a standard deviation of σ_{C_u}. This differential binary-scaled DAC is composed of 510 C_u-elements, with an LSB step of $2C_u$. Note that the factor of 2 comes from the differential implementation. The major-carry transition is usually considered as the worst case for matching, since at this transition (code transit from 01111111 to 10000000), all the capacitors in the binary-scaled design are active. It means all 510 unit capacitors are switched, leading to a DNL deviation of

$$\sigma_{DNL,mid} = \frac{\sqrt{510}\sigma_{C_u}}{2C_u} \tag{6.5}$$

The requirement of matching is evaluated using the 3σ deviation of DNL and has been set to a maximum error of 0.5LSB:

$$3\sigma_{DNL,mid} < \frac{1}{2}LSB \tag{6.6}$$

Substituting (6.5) into (6.6) results in a requirement of 1.5% for unit capacitor matching:

$$3\frac{\sqrt{510}\sigma_{C_u}}{2C_u} < \frac{1}{2} \Rightarrow \frac{\sigma_{C_u}}{C_u} < 1.5\% \tag{6.7}$$

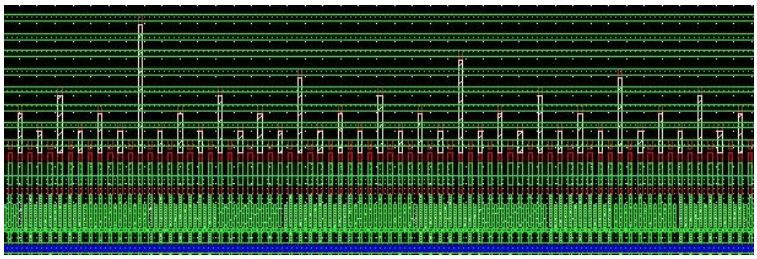

Figure 6.5: *Partial layout of the capacitor array consisting 510 unit capacitors.*

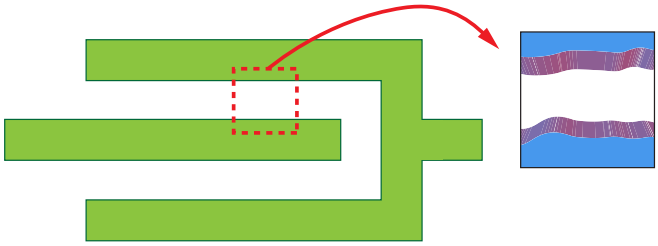

Figure 6.6: *The $0.5fF$ unit capacitor implementation, with a zoom-in window for indicating the LER effect.*

The mismatch originates from various random fluctuations during the manufacturing process. In this case of a metal-metal structure, the sidewalls of two metals facing each other are the main contributions of the intended parasitic capacitance. In other words, the distance between the sidewalls of the conductors is a critical parameter. Hence, we consider the LER effects to be the main cause of this mismatch (see Figure 6.6).

While not being able to get the estimation of σ_{LER} and η_{LER} from the technology based on which the chips have been fabricated, the estimation

from IMEC (6.4) is used for this simulation, as it is the closest measurement data available on a similar scale technology. Using the proposed modeling method, the mismatch of the unit capacitor caused by the LER is estimated to be around 0.25%. Although unavoidably, there are also other random variations during the manufacturing process, considering LER is the primary contributor, we estimate that the design should have enough margin to be fabricated.

In the end, we have conducted measurements on nine test-chips[†] to verify the matching of the unit capacitors. To distinguish the systematic and the random errors, the measured DNL is split into two parts: the systematic DNL and the random DNL. The systematic DNL is computed by averaging the nine measured samples. Then, the random DNL curves are obtained for each measured sample by subtracting the systematic DNL from the measured DNL. Figure 6.7 shows the measured systematic DNL and random DNL for all chips. As can be observed (note the different vertical scales), the systematic DNL is dominant for the overall DNL, which suggests that a systematic layout issue is causing DNL performance loss.

From the random DNL component, the random mismatch of the unit capacitors can be analyzed. Using (6.5), the $\frac{\sigma_{C_u}}{C_u}$ can be estimated based on the measured $\sigma_{DNL,mid}$ at the major-carry transition. Analogously, for each bit i (where the major-carry transition corresponds to $i = 7$), the $\frac{\sigma_{C_u}}{C_u}$ can be derived from the related code transition according to:

$$\frac{\sigma_{C_u}}{C_u} = \frac{2\sigma_{DNL_i}}{\sqrt{2 \times (2^{(i+1)} - 1)}} \tag{6.8}$$

where DNL_i is the DNL at the code transition of bit i, while the term $2 \times (2^{(i+1)} - 1)$ accounts for the number of active unit capacitors at this transition. Figure 6.8 shows, in solid line, the estimated capacitor matching for each bit transition, averaged over the 9 measured samples.

For comparison, a Matlab Monte-Carlo simulation is conducted using an ADC model. It takes into account the capacitor mismatch and the random noise during measurement. Results are also shown in Figure 6.8 in different dashed lines. Comparing to the MC simulation model, the capacitor matching is estimated to be better than 1%, most likely in the order of 0.5% or less. Since the measured mismatch includes various fluctuations

[†]Although more samples would be appreciated for statistical analysis, there are only nine test-chips available. Still, the measurement results can to some extent indicate the magnitude of the matching.

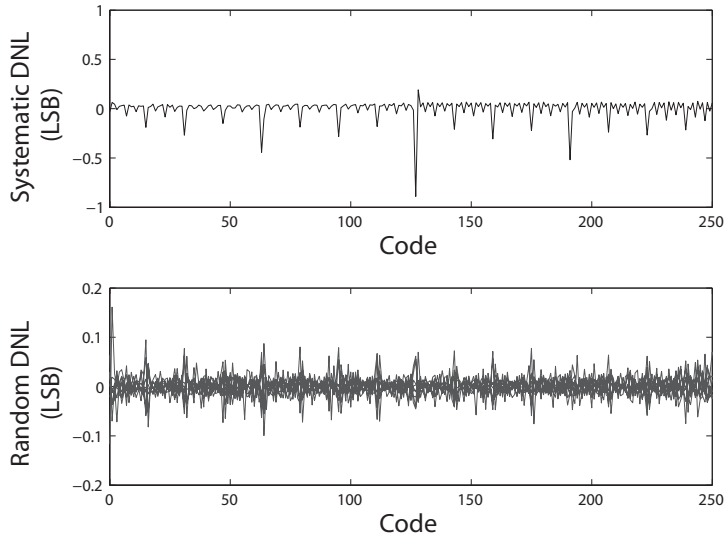

Figure 6.7: *Measured systematic and random DNL for nine test chips.*

and noises from all aspects during the fabrication and the measurement processes, it agrees with the simulation results.

Without a proper modeling tool as presented, the designers may not be confident in making such a small-size high-precision design, as the matching is a major requirement to successfully achieve the desired performance. Results suggest that the mismatch of the unit capacitors is in fact small enough for such an implementation with even higher resolution, for instance a 10-bit ADC.

6.7 Sensitivity-Based Modeling for Both Systematic and Random Variations

From derivation of the capacitance modeling methods for systematic variations (Chapter 3, Appendix C) and for random variations (Chapter 6.4), the following observations can be made:

1. Nominal capacitances are computed with the partial short-circuit capacitances $\bar{\mathbf{C}}_{\mathbf{s}}$, using an incidence matrix (2.9) and (2.8);

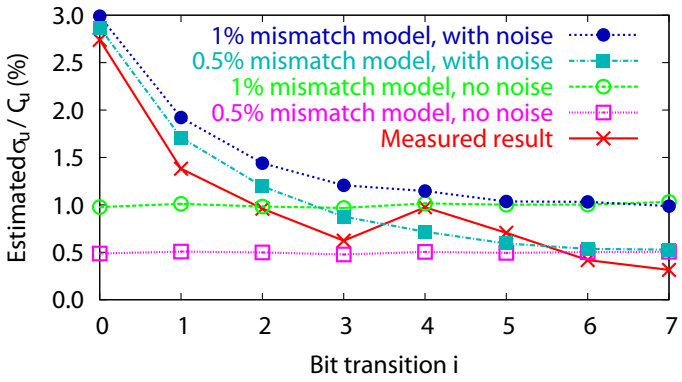

Figure 6.8: *Estimated capacitor mismatch based on DNL measurements.*

2. The dimensional capacitance sensitivity for modeling systematic variabilities is computed by assembling the associated panel sensitivities (5.1);

3. The statistical modeling for random variation LER also counts on combinations of the panel sensitivities (6.3);

4. The computation of panel sensitivities solely relies on $\bar{\mathbf{C}}_s$ (C.10).

From these observations, one can conclude that a sensitivity-based algorithm can be developed to model both the systematic and the random variabilities with only *one* system solve, integrated with the standard capacitance extraction. This is illustrated in Figure 6.9. The sensitivity-based modeling method can be integrated in BEM-based LPE tools to account for both variabilities simultaneously. Using the additional inputs of the dimensional parameters with their process spreads (σ_{sys}) as well as the rough lines of interest characterized with σ_{LER} and η_{LER}, the resulting statistical properties of the capacitance, i.e. the mismatch, can be obtained together with the nominal capacitance with one system solve.

Note that the input for the systematic variability can be multiple dimensional parameters, including the widths of all conductors, the thicknesses of all metal layers and the heights of all dielectric layers. A windowing technique can be applied to handle large structures. Similarly, the input for the random variability LER can be multiple rough lines with various characterizing parameters σ_{LER} and η_{LER}. Most importantly, this

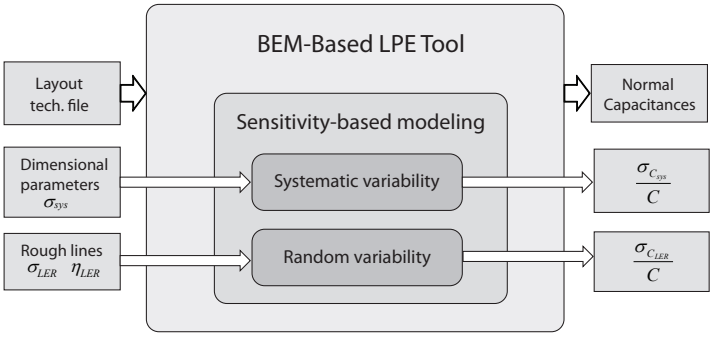

Figure 6.9: *Diagram of the proposed sensitivity-based method integrated in BEM-based LPE tools for both systematic and random variabilities.*

can all be done with one system solve with an additional computational time being a small portion of that for a standard capacitance extraction without considering any variability.

To demonstrate the proposed sensitivity-based modeling method for both systematic and random variabilities, an illustrative example is conducted. All parameters in this example are chosen based on pure assumptions.

There are two parallel conductors with the width/space being $2\mu m/2\mu m$, the thickness being $2\mu m$ and the length being $8\mu m$. It is assumed that all four edges along the length of the conductor are subject to the LER. The characterizing parameters are $\sigma_{LER} = 0.03\mu m$, $\eta_{LER} = 2.00\mu m$ for one conductor and $\sigma_{LER} = 0.04\mu m$, $\eta_{LER} = 2.88\mu m$ for the other. With this experiment, it is also interesting to compare the impacts of the systematic and the random variations on capacitances. To do so, we take the conductor width as the geometric parameter for systematic variations, with a standard deviation (σ_{sys}) being the same as the associated σ_{LER}, i.e. $0.03\mu m$ and $0.04\mu m$ for the two conductors respectively.

To verify the accuracy and the efficiency of the proposed method, three Monte-Carlo simulations with 1000 samples each, are performed for the systematic variation, the random variation and a superposition of them respectively (3000 samples in total). Since systematic and random variations are originated from different sources during the manufacturing pro-

cess, they are often considered independent. Hence, the proposed model evaluates their superposition effect using

$$\frac{\sigma_C}{C} = \sqrt{\left(\frac{\sigma_{C_{sys}}}{C}\right)^2 + \left(\frac{\sigma_{C_{LER}}}{C}\right)^2}. \tag{6.9}$$

Simulation results are shown in Table 6.2. Using MC simulations as a reference, the errors of the relative standard deviations of the capacitance given by the model are smaller than 15%. Also notice that the systematic variation is the dominant one, given the same process spreads of parameters as for the random variation. Thus, for designs that are sensitive to both variations, mainly the systematic variability should be improved. On the other hand, some designs are only vulnerable to random variations. Therefore, being able to distinguish the two variations and apply the appropriate modeling technique is essential.

Table 6.2: *Simulation results and CPU time for both systematic and random variations*

	MC simulation	Proposed model
$\frac{\sigma_{C_{sys}}}{C}$	2.22%	2.05% (7.72% error)
$\frac{\sigma_{C_{LER}}}{C}$	0.21%	0.24% (14.23% error)
$\frac{\sigma_C}{C}$	2.22%	2.06% (7.18% error)
CPU Time	$38h52'$	$58''$

6.8 Conclusion

The high efficiency of the proposed method enables the possibility to analyze, from various aspects, the mismatches of the capacitance caused by both systematic and random variations. With a good enough accuracy for most application purposes, this method provides a fast and useful tool for Design-for-Manufacturability (DFM).

CHAPTER 7

Fast Statistical Analysis of RC Nets Subject to Manufacturing Variabilities*

Much work has been done aimed at capturing the effects of process variations using parameter-aware techniques. The research focuses mainly on two aspects. One is parameterized Layout Parasitic Extraction (LPE), which models the effect of physical variations by generating linear or quadratic models of capacitances and resistances as a function of process parameters. For on-chip interconnects, most work has concentrated on the capacitances and a relatively modest amount of work has been published for parameterized resistance extraction [77,78], because it is relatively simple†. More importantly, it is the capacitance computation rather than the resistance computation which dominates the complexity of the overall LPE procedure.

*This chapter is based on [76]: Yu Bi, K.J. van der Kolk, J.F. Villena, L.M. Silveira and N.P. van der Meijs, "Fast statistical analysis of RC nets subject to manufacturing variabilities," in *Proc. DATE*, Grenoble, France, pp. 32-37, March, 2011.

†When it concerns substrate or other distributed resistance, the extraction becomes more complicated. While this is beyond the scope of our research, readers who are interested could refer to [79–81].

The capacitance and the resistance models generated by the parameterized LPE tools may be used directly for SPICE simulations, or more often, they are fed to a parameterized Model Order Reduction (pMOR) procedure to achieve an essential speedup. This pMOR procedure is the other aspect of using parameter-aware techniques for capturing variabilities.

Many pMOR methods are based on multi-dimensional moment matching. They rely on matching the moments of the parameterized system transfer function, which depends on both the frequency and the parameters [36, 82]. A sampling based approach [83] extended from the PMTBR algorithm [84] proposes to use the statistical information of the parameters as a guidance for a multidimensional sampling of the joint frequency plus parameter space. Recently, a structure-preserving pMOR technique [37] proposes a reformulation of the system to maintain an explicit parameter dependence of the transfer function. This property is very convenient for variational and statistical analysis, as will be demonstrated in this chapter.

In order to understand the impact of the physical process variations on the performance of a circuit, the above two aspects have to be fully and well integrated. This has seldom been studied yet, and therefore to draw the overall picture is an important contribution of the work presented in this chapter.

However, a simple combination of the above two aspects does not solve the real problem. The final goal of modeling manufacturing variabilities is to obtain the statistical properties of the circuit performance, for instance the system response, given the process spreads of the technology. The traditional Monte Carlo approach has a fatal drawback: the parameter sampling implies a huge computational burden for both the extraction and the reduction procedures, which is least favorable in practice. The proposed method avoids parameter samplings by using the parameterized LPE and the pMOR techniques. In particular, we only consider systematic variation in this chapter. The parameterized extraction applies the linear model of capacitances presented in Chapter 3, and the linear model of resistances which will be presented in Section 7.2. As mentioned, the structure-preserving pMOR technique [37] is used for the reduction procedure.

This chapter is organized as follows. Section 7.1 briefly reviews the description of a parameterized system and the capacitance sensitivity computation. Section 7.2 presents the parameterized extraction methods for conductances. Section 7.3 introduces the reduction methods for the para-

meterized system generated by the previous extraction, focusing on the structure-preserving pMOR technique. In Section 7.4, the proposed statistical analysis methodology for RC nets is presented.

7.1 Parameterized Capacitance Extraction

A system subject to manufacturing variabilities is often described using a parameterized representation of the conductance and the (short-circuit) capacitance matrices $\mathbf{G}(\lambda) \in \mathbb{R}^{n\times n}$ and $\mathbf{C_s}(\lambda) \in \mathbb{R}^{n\times n}$. These matrices can be approximated by a Taylor series w.r.t. multiple parameter variations, for instance, a first order approximation

$$\mathbf{C_s}(\lambda) = \mathbf{C_{s0}} + \sum_{i=1}^{Q} \lambda_i \mathbf{C_{s}}i \tag{7.1a}$$

$$\mathbf{G}(\lambda) = \mathbf{G}_0 + \sum_{i=1}^{Q} \lambda_i \mathbf{G}_i \tag{7.1b}$$

where $\mathbf{C_{s0}}$ and $\mathbf{G}_0$ are the nominal values for the matrices, $\mathbf{C_{s}}i$ and $\mathbf{G}_i$ are the sensitivities w.r.t. the i-th parameter variation λ_i, and Q is the number of parameters. Thus for such linear approximation, the most important task is to compute the sensitivities $\mathbf{C_{s}}i$ and $\mathbf{G}_i$.

First of all, a very brief review of the capacitance sensitivity computation is given below. Let $a \in N_i$ and $b \in N_j$ be any panel associated with two nodes i and j respectively. Let s_p denote the set of victim panels incident to parameter p, and $k \in s_p$ be any victim panel with A_k being its area and ε being the material permittivity around it. Then according to the algorithm proposed in Chapter 3, the sensitivity of the (short-circuit) capacitance between nodes i and j w.r.t. a parameter variation λ_p is given by

$$\frac{\partial C_{sij}}{\partial p} = \sum_{k\in s_p} \left(\frac{1}{\varepsilon A_k} \sum_{a\in N_i} \sum_{b\in N_j} \bar{C}_{sk,a} \bar{C}_{sk,b} \right) \tag{7.2}$$

where, as defined already, $\bar{C}_{sk,a}$ and $\bar{C}_{sk,b}$ are the partial short-circuit capacitances associated with panel k, a and panel k, b respectively.

The above discussion shows:

- Sensitivities between different nodes can be obtained simply by assembling different sets of associated partial short-circuit capacitances, i.e., $\sum_{a\in N_i} \bar{C}_{sk,a}$.

- Sensitivities w.r.t. different parameter variations are local for different sets of victim panels, i.e., $k \in s_{\lambda_p}$.
- The nominal capacitances are computed using the partial short-circuit capacitances $\bar{\mathbf{C}}_\mathbf{s}$.

It follows that the data needed for the sensitivity computation and for the nominal capacitance calculation is the same, i.e., the partial short-circuit capacitances $\bar{\mathbf{C}}_\mathbf{s}$. Therefore, all the sensitivities between various nodes w.r.t. multiple variations can be computed simultaneously with the nominal capacitance extraction. The representation $\mathbf{C}_\mathbf{s}(\lambda)$ in (7.1) can be obtained with a single extraction procedure. It has been shown in Section 4.2.1 that the extra computational time for the sensitivity is negligible compared to that of the standard capacitance extraction. The high efficiency of capacitance sensitivity computation is essential because during the parasitics extraction, the overall performance is ruled by the capacitance extraction instead of the conductance extraction. As can be seen in the next section, both the nominal conductance extraction and its sensitivity computation are simpler and more straightforward compared to that of the capacitances.

7.2 Parameterized Conductance Extraction

This section presents the generation of the parameterized conductances. Firstly, a brief summary of the nominal conductance extraction using the Finite Element Method (FEM) is given (see [85] for more details). Then, the computation of sensitivities w.r.t. geometric variations is presented.

7.2.1 FEM-based Conductance Extraction

To perform FEM, the interconnect polygons are firstly discretized into rectangles or triangles, which are often called *tiles*. Computation is then conducted for each tile to place a conductance element to each branch, as illustrated in Figure 7.1.

For rectangle tiles, the conductance element on a branch between two vertices (x_i, y_i) and (x_j, y_j) is computed as [85]

$$G_{ij} = \begin{cases} G_{sh}\frac{(x_i - x_j)^2}{2A} & \textit{if } y_i = y_j \\ G_{sh}\frac{(y_i - y_j)^2}{2A} & \textit{if } x_i = x_j \\ 0 & \textit{otherwise} \end{cases} \tag{7.3}$$

where A is the area of the tile, and G_{sh} is the *sheet conductance*, i.e. the inverse of the *sheet resistance*, of the material. Sheet conductances or sheet resistances of different layers are usually defined in the technology file provided by the foundry.

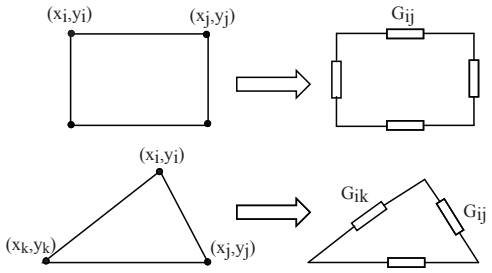

Figure 7.1: *Illustration of the FEM for conductance extraction using rectangle tiles and triangle tiles respectively.*

As to the triangle tiles defined by three vertices $(x_i,\, y_i)$, $(x_j,\, y_j)$ and $(x_k,\, y_k)$, the conductance element G_{ij} can be calculated as [85]

$$G_{ij} = G_{sh}\frac{(x_k - x_i)(x_k - x_j) + (y_k - y_i)(y_k - y_j)}{4A}. \tag{7.4}$$

It is known that the sheet conductance G_{sh} is defined as a product of the conductivity σ and the layer thickness t

$$G_{sh} = \sigma t. \tag{7.5}$$

Thus Equation (7.3) and (7.4) can be summarized in the following expression:

$$G_{ij}(t_\alpha, l_\alpha) = G_{sh}(t_\alpha)F(l_\alpha) \tag{7.6}$$

where i and j are indices of the vertices of a tile α, t_α is its corresponding layer thickness, and l_α represents the layout dimension associated with the vertices. It follows that the dependencies of G_{ij} on the layer thickness (z-dimension) and the layout (x, y-dimensions) are separate, which is very convenient for the sensitivity computation.

7.2.2 Conductance Sensitivities

Unlike capacitances, there are practically only two geometric parameters relevant to each conductance, namely the thickness of its corresponding layer t_α and its corresponding tile dimension l_α. Furthermore, as concluded from Equation (7.6), the sensitivities of a conductance w.r.t. these two parameter variations can be computed separately, i.e.,

$$\frac{\partial G_{ij}}{\partial \lambda_t} = F \frac{\partial G_{sh}(t)}{\partial \lambda_t}; \tag{7.7}$$

$$\frac{\partial G_{ij}}{\partial \lambda_l} = G_{sh} \frac{\partial F(l)}{\partial \lambda_l}. \tag{7.8}$$

Variation in the layer thickness

The calculation is simple and straightforward. Considering the sensitivity is for the nominal or designed system, the substitution of (7.5) in (7.7) leads to:

$$\frac{\partial G_{ij}}{\partial \lambda_t}|_{t_0} = F\sigma = \frac{F\sigma t_0}{t_0} = \frac{G_{ij}(t_0)}{t_0} \tag{7.9}$$

where t_0 is the nominal layer thickness and $G_{ij}(t_0)$ is the nominal conductance.

Layout variation

The perturbation method is used to calculate the sensitivities w.r.t. the layout variation.

Unlike capacitance, conductance is a *self*-property of an interconnect, depending solely on its own dimension and material property. Thus the sensitivities of all the conductances in the system w.r.t their related layout variations can be obtained using only one finite difference (FD) computation with simply one extra system solve for conductances. In this case, the perturbation method is a very appropriate choice for its high accuracy and the modest additional computational complexity, i.e., $1\times$ the nominal extraction.

Using the perturbation method, the sensitivities of conductances w.r.t. the layout variation can be calculated as

$$\frac{\partial G_{ij}}{\partial \lambda_l} = \lim_{\lambda_l \to 0} \frac{G_{ij}(l_0 + \lambda_l) - G_{ij}(l_0)}{\lambda_l} \tag{7.10}$$

where λ_l is the layout perturbation, $G_{ij}(l_0)$ is the nominal conductance and $G_{ij}(l_0 + \lambda_l)$ represents the perturbed conductance. An issue related to the computation of (7.10) in a matrix form is that the size of the conductance matrix $\mathbf{G}$ has to remain the same. To do so, the geometry of the perturbed system is generated by adjusting the vertex coordinates of the original system, illustrated in Figure 7.2. This implies that only the

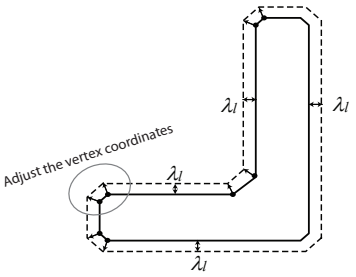

Figure 7.2: *Illustration of the layout perturbation for the sensitivity computation, where λ_l represents a small layout perturbation.*

conductances related to the boundary nodes are affected and generate non-zero sensitivities. It can also be seen from the figure that irregular shapes can be handled as well. This has a practical meaning because the corners of wires or wire-like structures are usually no longer right angles after the lithography and the etching processes.

7.3 Order Reduction of Parameterized Systems

This section summarizes the pMOR methods for the parameterized system generated in the previous sections, focusing on the structure-preserving technique to be applied in the proposed statistical analysis methodology.

7.3.1 Parameterized Model Order Reduction

In Section 7.1 and 7.2, the sensitivity-based parameterized LPE technique generates the capacitance and the conductance matrix descriptors (7.1) for a parameterized system. Such a system has an associated parameter dependent frequency response that can be modeled via the transfer function

$$\mathbf{H}(s, \lambda) = \mathbf{E}\left[s\mathbf{C_s}(\lambda) + \mathbf{G}(\lambda)\right]^{-1}\mathbf{B} \tag{7.11}$$

where $\mathbf{C_s}(\lambda), \mathbf{G}(\lambda) \in \mathbb{R}^{n\times n}$ are the generated parameter dependent capacitance and conductance matrices, and $\mathbf{B} \in \mathbb{R}^{n\times m}$ and $\mathbf{E} \in \mathbb{R}^{p\times n}$ are the input and the output incidence matrices respectively.

For a complete analysis, a linear system of dimension n has to be solved for every point of the parameter plus frequency space. Thus when the size of the system n is large, the analysis of the function in (7.11) becomes prohibitive.

To overcome this issue, Parameterized Model Order Reduction (pMOR) approaches seek to efficiently generate a reduced order approximation, usually by projecting the system into a suitable reduced q-dimensional subspace, $q \ll n$, spanned by the columns of a projector $V \in \mathbb{R}^{n\times q}$ (see [36, 82, 83] for details). The projection generates a Reduced Order Model (ROM) with an associated reduced transfer function

$$\widehat{\mathbf{H}}(s,\lambda) = \widehat{\mathbf{E}}\left[s\widehat{\mathbf{C}_\mathbf{s}}(\lambda) + \widehat{\mathbf{G}}(\lambda)\right]^{-1}\widehat{\mathbf{B}} \tag{7.12}$$

where $\widehat{\mathbf{C}_\mathbf{s}}(\lambda), \widehat{\mathbf{G}}(\lambda) \in \mathbb{R}^{q\times q}$, $\widehat{\mathbf{B}} \in \mathbb{R}^{q\times m}$, and $\widehat{\mathbf{E}} \in \mathbb{R}^{q\times p}$ define the ROM of dimension $q \ll n$, which can be handled much more efficiently.

7.3.2 Explicit Parameter Matching

Standard projection based pMOR approaches generate a ROM with an equivalent Taylor series formulation for the $\widehat{\mathbf{C}_\mathbf{s}}(\lambda)$ and $\widehat{\mathbf{G}}(\lambda)$ matrices, which is useful in terms of compatibility. This allows for a fast evaluation of the system matrix for any parameter and frequency point, but still requires solving the system in order to obtain the system response. Any modification of the frequency or parameter values implies another solve. For statistical analysis with potentially a large number of Monte Carlo (MC) samples, this could be expensive.

A different pMOR approach which is interesting in terms of statistical analysis is the one presented in [37], valid for the cases in which the output behavior w.r.t. the parameters is smooth, which as can be seen is indeed the case. It proposes a reformulation of the system as a Taylor series approximation of the transfer function w.r.t. the parameters.

This is achieved by a truncated Taylor series representation of the matrices $\mathbf{C_s}(\lambda)$ and $\mathbf{G}(\lambda)$ as in (7.1), plus an expansion of the state vector in Taylor series w.r.t. the parameters of order T, but not w.r.t. the

frequency:

$$\mathbf{x}(s,\lambda) = \mathbf{x}_0(s) + \sum_{i=1}^{Q}\sum_{j=1}^{T} \lambda_i^j \mathbf{x}_{ij}(s) \tag{7.13}$$

with $\mathbf{x}_0$ the nominal state vector, and $\mathbf{x}_{ij}$ the sensitivity of order j $(j1, ..., T)$ w.r.t. parameter λ_i $(i = 1, ..., Q)$. This formulation, presented here without cross terms, can be extended to any desired order, including cross terms. As an example, for a single parameter λ_1, using the first order sensitivities in (7.1) (i.e. $\mathbf{G_0}, \mathbf{C_{s0}}, \mathbf{G_1}$ and $\mathbf{C_{s1}}$), the states can be approximated as

$$\begin{aligned}
&\mathbf{x}(s,\lambda_1) = \mathbf{x}_0(s) + \lambda_1\mathbf{x}_1(s) + \lambda_1^2\mathbf{x}_2(s) + \dots \\
&\mathbf{x}_0(s) = (\mathbf{G_0} + s\mathbf{C_{s0}})^{-1}\mathbf{B}u \\
&\mathbf{x}_1(s) = -(\mathbf{G_0} + s\mathbf{C_{s0}})^{-1}(\mathbf{G_1} + s\mathbf{C_{s1}})\mathbf{x}_0(s) \\
&\mathbf{x}_2(s) = -(\mathbf{G_0} + s\mathbf{C_{s0}})^{-1}(\mathbf{G_1} + s\mathbf{C_{s1}})\mathbf{x}_1(s) \\
&\dots
\end{aligned} \tag{7.14}$$

This explicit parameter dependence can be shifted to the output, which generates a parameterized transfer function as [37]

$$\mathbf{H}(s,\lambda) = \mathbf{H}_0(s) + \sum_{i=1}^{Q}\sum_{j=1}^{T} \lambda_i^j \mathbf{H}_{ij}(s) \tag{7.15}$$

where $\mathbf{H}_{ij} = \mathbf{E}\mathbf{x}_{ij}$ are the frequency dependent transfer function sensitivities, each one related to one sensitivity of the states $\mathbf{x}$. In other words, the parameterized transfer function can be written as the contribution of the nominal transfer function plus the contribution of each one of the non-parameterized transfer function sensitivities w.r.t. the parameters, i.e. a linear combination of the multiple non-parameterized transfer functions weighted by the parameter variation.

The work in [37] also presents a compact state-space formulation for the complete system generating the individual transfer functions in (7.15), plus a scheme for efficiently reducing each transfer function independently (via preservation of the system structure), in order to maintain the explicit parameter dependence (see [37] for details).

A very important property of this representation and reduction is that it maintains an explicit parameter dependence on the output. Every reduced transfer function sensitivity, which only depends on the frequency,

can be solved independently. The parameterized response is obtained by a linear combination of the multiple transfer functions, as shown in (7.15).

This specific structure-preserving property enables a very fast way to evaluate the statistical properties, for instance the standard deviation of the system response, which is a main topic of this chapter presented in the next section.

7.4 Statistical Analysis of RC Nets

In this section, a statistical analysis methodology for RC nets is proposed. By combining the parameterized parasitics extraction and the structure-preserving MOR techniques, the transfer function and its standard deviation due to systematic variations can be obtained with a significant speed up, compared to the traditional Monte Carlo approach.

7.4.1 Design Flow

Traditionally, to compute the statistical properties of the transfer function, one has to perform a Monte Carlo simulation which implies a huge computational burden, as indicated in Figure 7.3.

A parameter sampling has to be conducted using the layout and the technology based on the particular process spread, obtaining k problem instances for the layout parasitics extractor. Then k standard parasitics extractions are performed to generate k non-parameterized systems, i.e., k groups of matrix descriptors ($\mathbf{G}_{var}$, $\mathbf{C}_{\mathbf{s}var}$). This is followed by k SPICE simulations or non-parameterized MOR procedures. Finally, the mean and the standard deviation of the transfer function can be calculated from the generated k transfer functions, denoted $\bar{\mathbf{H}}(s)$. Since the sampling number k has to be large enough to ensure a reliable statistical distribution, the workload of this traditional design flow is enormous due to the k-fold parasitics extraction and the k-fold SPICE simulation or standard MOR procedure.

The above two issues can be conquered by the proposed method using the parameterized LPE and the structure-preserving pMOR techniques respectively. The design flow is shown in Figure 7.4. With the input of the layout information from designers and the technology file from the foundry, parameterized RC extraction generates the first order parameterized Taylor series $\mathbf{G}(\lambda)$ and $\mathbf{C}_{\mathbf{s}}(\lambda)$ as in (7.1).

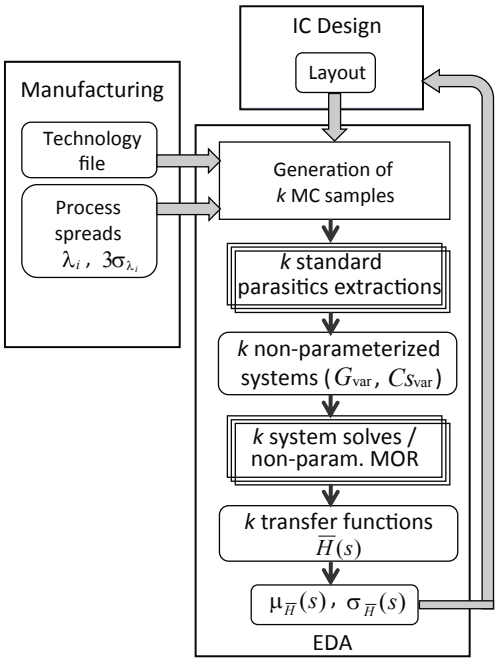

Figure 7.3: *The design flow of the Monte Carlo approach. Sampling of the parameter space is needed, which implies a huge computational burden.*

As addressed earlier, the computational complexity of setting up the parameterized conductance representation is two times the complexity of the nominal conductance $\mathbf{G}_0$ extraction and the complexity of setting up the parameterized capacitance representation is almost the same as the nominal capacitance extraction. The computation of capacitances rather than conductances dominates the overall computational burden. Therefore, the asymptotic complexity of the parameterized RC extraction, including the nominal values $\mathbf{G}_0$, $\mathbf{C_{s0}}$ and their sensitivities $\mathbf{G}_i$, $\mathbf{C_{s}}_i$ w.r.t. multiple parameters, is the same as the asymptotic complexity of the nominal capacitance extraction. The high efficiency of the parameterized extraction technique is one essential advantage of the proposed method.

Using the generated parameterized matrix descriptors (7.1), the structure-preserving pMOR technique is applied to calculate the reduced

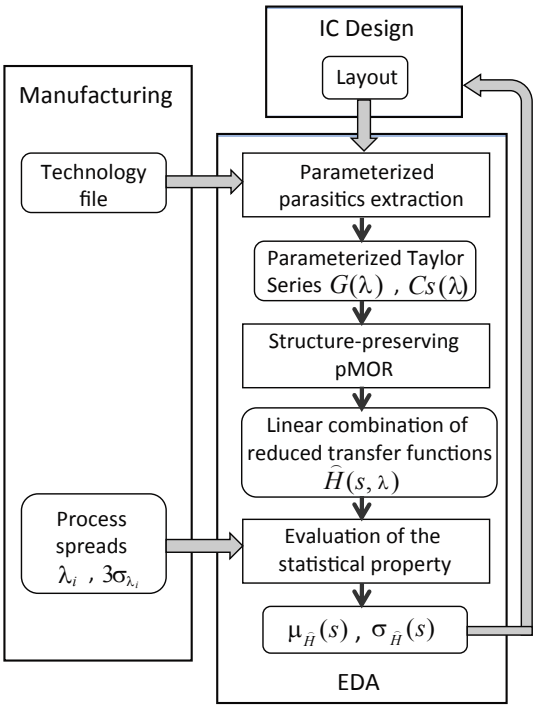

Figure 7.4: *The design flow of the proposed method. No sampling of the parameter space is needed because of the use of the parameterized parasitics extraction technique and the structure-preserving pMOR technique.*

transfer function with an explicit parameter dependence, expressed as

$$\widehat{\mathbf{H}}(s,\lambda) = \widehat{\mathbf{H}_0}(s) + \sum_{i=1}^{Q}\sum_{j=1}^{T} \lambda_i^j \widehat{\mathbf{H}_{ij}}(s) \tag{7.16}$$

where i represents the parameter index and j represents the order in the Taylor series expansion.

Applying various parameter settings, the induced variability of the transfer function can be easily obtained from (7.16). Furthermore, and unlike standard projection based pMOR approaches, any change of the parameter variations simply requires evaluating the linear combination, with no need for additional system solution. This implies a major boost in the efficiency of the variational analysis.

More importantly, given the process spreads of the parameters ($3\sigma_{\lambda_i}$)

from the manufacturer, the statistical properties such as the standard deviation of the transfer function (7.16) can be evaluated with a negligible complexity. This is the other essential advantage of the proposed method and will be discussed in Section 7.4.2.

At last, the obtained mean and the standard deviation of the transfer function $\mu_{\hat{H}}(s)$ and $\sigma_{\hat{H}}(s)$ are fed back to the designers so that adjustments or improvements can be carried out before the tape-out. Thus it is very convenient for design exploration and optimization.

As illustrated in Figure 7.4, both the parasitics extraction and the MOR procedures have avoided the parameter sampling. Therefore, the proposed method is highly efficient as a variation-aware modeling approach, especially for statistical analysis.

7.4.2 Statistical Property Computation

This section explains the computation of the mean and the standard deviation of the reduced transfer function (7.16). Both the nominal transfer function $\widehat{\mathbf{H}_0}(s)$ and the transfer function sensitivities $\widehat{\mathbf{H}_{ij}}(s)$ are vectors of the system response to frequency samples. The following computation is conducted for each frequency sample s_k. For the ease of discussion, some short-hand notations are used:

$$a_0 = \widehat{H_0}(s_k), \quad a_{ij} = \widehat{H_{ij}}(s_k), \quad F(\lambda_i) = \sum_{j=1}^{T} a_{ij}\lambda_i^j \tag{7.17}$$

The expectation of $\widehat{H}(s_k, \lambda)$ can then be expressed as

$$E[\widehat{H}(s_k, \lambda)] = E[a_0] + \sum_{i=1}^{Q} E[F(\lambda_i)] \tag{7.18}$$

where $E[a_0] = a_0$ and the expectation of $F(\lambda_i)$ can be computed as

$$\begin{aligned} E[F(\lambda_i)] &= \int_{-\infty}^{\infty} \sum_{j=1}^{T} a_{ij}\lambda_i^j f(\lambda_i) d\lambda_i \\ &= \sum_{j=1}^{T} a_{ij} \int_{-\infty}^{\infty} \lambda_i^j f(\lambda_i) d\lambda_i \end{aligned} \tag{7.19}$$

with $f(\lambda_i)$ the probability density function of λ_i. It is common to assume the distribution of the geometric parameter variations to be Gaussian with a mean of zero, i.e., $\lambda_i \sim \mathcal{N}(0, \sigma_{\lambda_i})$. Then

$$\int_{-\infty}^{\infty} \lambda_i^j f(\lambda_i) d\lambda_i = E[(\lambda_i - \mu_{\lambda_i})^j], \qquad \mu_{\lambda_i} = 0 \tag{7.20}$$

is the central moment of the Gaussian distribution with a zero mean of order j, which is calculated as

$$\mathrm{E}[(\lambda_i - \mu_{\lambda_i})^j] = \begin{cases} 0 & \textit{if } j \textit{ is odd} \\ \sigma_{\lambda_i}^j (j-1)!! & \textit{if } j \textit{ is even} \end{cases} \tag{7.21}$$

with $(j-1)!!$ denotes the double factorial, that is the product of every odd number from $j-1$ to 1. Therefore Equation (7.19) is a linear combination of the Gaussian central moments from order 1 to T, weighted by the corresponding transfer function sensitivities. The mean of the transfer function ($\mu_{\widehat{H_0}}(s_k)$) can then be computed by substituting (7.19) into (7.18).

As for the standard deviation, it is interesting to note that the variations of different parameters are usually independent since they are originated from different process steps. Assuming this is the case, the standard deviation of $\widehat{H}(s_k, \lambda)$ can be computed as

$$\sigma_{\widehat{H}}(s_k) = \sqrt{\sum_{i=1}^{Q} Var(F(\lambda_i))} \tag{7.22}$$

where the variance of $F(\lambda_i)$ is calculated as follows, using the computed expectation of $F(\lambda_i)$, i.e., $\mu_{F(\lambda_i)}$,

$$\begin{aligned} Var(F(\lambda_i)) &= E[(F(\lambda_i)) - \mu_F)^2] \\ &= -\mu_{F(\lambda_i)}^2 + E[F^2(\lambda_i)], \end{aligned} \tag{7.23}$$

with

$$E[F^2(\lambda_i)] = \sum_{j=1}^{T} a_{ij}^2 E[\lambda_i^{2j}] + 2 \sum_{j=1}^{T} \sum_{l=j+1}^{T} a_{ij} a_{il} E[\lambda_i^{j+l}]. \tag{7.24}$$

Note that $E[\lambda_i^{2j}]$ and $E[\lambda_i^{j+l}]$ are also central moments of Gaussian distributions with zero means. Thus substituting (7.24) into (7.23) solves the variance of $F(\lambda_i)$, which is a combination of the Gaussian central moments with various orders, weighted by the incident transfer function sensitivities. At last, the standard deviation of the transfer function of each frequency sample $\sigma_{\widehat{H}}(s_k)$ can be obtained by substituting (7.23) into (7.22).

7.4.3 Experiment and Result

To demonstrate the efficiency and the accuracy of the proposed method, a two-terminal *RC* structure is studied. As shown in Figure 7.5, the example has two layers consisting of a meandering poly resistor connected to terminal **A** and a fork-structure metal capacitor connected to terminal **B**. This example is initially modeled with 410 states.

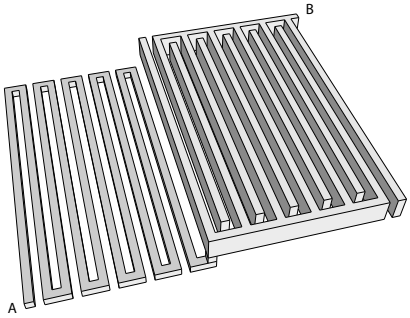

Figure 7.5: *Layout of the RC example with a poly resistor (connected to A) and a metal capacitor (connected to B).*

This structure depends on six geometric parameters, namely the poly and the metal thicknesses, the layout variations and the dielectric thicknesses of the two layers. The process spreads ($3\sigma_{\lambda_i}$) of these parameters are assumed to be 10% of their nominal values. To verify the results of the proposed method, a Monte Carlo (MC) simulation with 700 samples is performed according to the flow in Figure 7.3. The experiment is conducted on a 3.00GHz Intel 2 Core CPU.

Results are summarized in Figure 7.6 and 7.7. The mean of the transfer function computed by the proposed method shows a good agreement with the result obtained from the MC simulation (see Figure 7.6). To indicate how much is the effect of the assumed process spreads, Figure 7.6 also shows the induced variability of the transfer function, i.e., $\mu_H(s) \pm 3\sigma_H(s)$, given by the proposed method and the MC approach respectively. Figure 7.7 shows the accuracy of the proposed method. The relative error of the computed mean $\mu_{\hat{H}}(s)$ is very small, with an average over the frequency being 0.0125 and the maximum being 0.0274. The computed relative standard deviation $\frac{\sigma_{\hat{H}}}{\mu_{\hat{H}}}(s)$, known as the *mismatch* by designers, has an average relative error (absolute value) over the frequency of 0.0206 and a

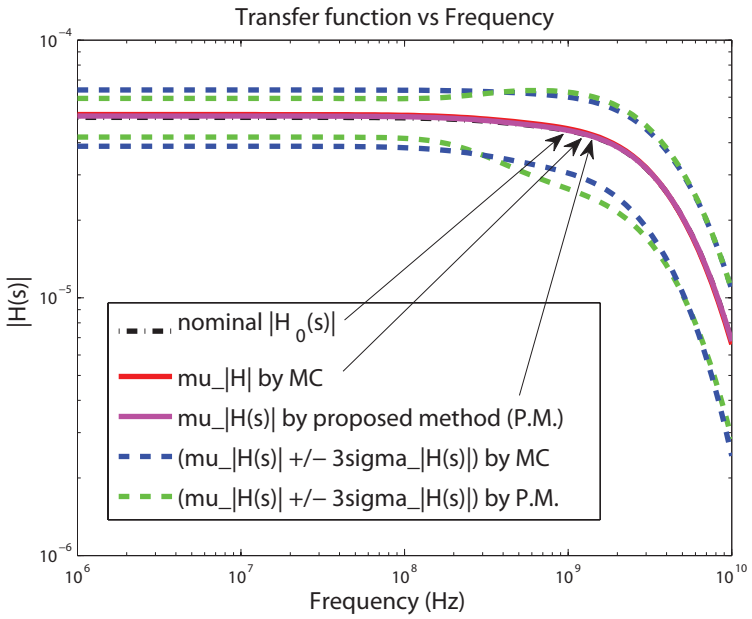

Figure 7.6: *Results of the proposed method compared to the Monte Carlo approach.*

maximum of 0.0274. Therefore, the proposed method nicely captures the effect of the physical process variations on the system response.

More importantly, the proposed method achieves a significant speed up over the traditional Monte Carlo approach. Table 7.1 shows the elapsed time and speed ups in the extraction procedure and the evaluation of the statistical properties, for the traditional MC analysis, the traditional MC analysis plus non-parameterized MOR (in this case PRIMA [86]) on each extracted system, and the proposed approach. Note that the speed up, which already achieves two orders of magnitude for a middle size example (410 states), will further increase with the increase of the size of the system and the number of samples.

7.5 Conclusion

This chapter presents a highly efficient statistical analysis methodology for RC nets subject to manufacturing variabilities. It achieves *zero* para-

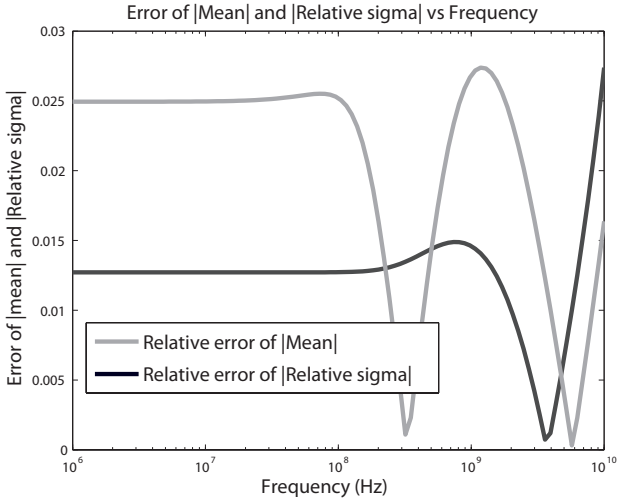

Figure 7.7: *Results of the proposed method compared to the Monte Carlo approach.*

meter sampling, based on the combination of a sensitivity-based parameterized parasitics extraction technique and a structure-preserving pMOR technique. Given the layout and the process spreads of the technology, the statistical properties such as the mean and the standard deviation of the system response can be obtained extremely fast. As such, the proposed method provides a very convenient tool for design exploration and optimization.

Table 7.1: *CPU time and speed ups in the extraction procedure and the evaluation of the statistical properties, for the traditional MC analysis, the traditional MC analysis plus non-parameterized MOR (in this case PRIMA [86]) on each extracted system, and the proposed approach.*

	Extraction	Evaluation	Total
MC	$55h49'$ (1×)	$46'12''$ (1×)	$56h35'$ (1×)
MC + MOR	$55h49'$ (1×)	$3'30''$ (13×)	$55h52'$ (1.01×)
Proposed	$4'49''$ (695×)	$0.43''$ (6446×)	$4'50''$ (701×)

CHAPTER 8

Conclusion

In this thesis, we have proposed a sensitivity-based modeling method to capture the effects of systematic and random manufacturing variations on on-chip interconnect capacitances. We are able to compute all sensitivities of important capacitances efficiently in a one pass algorithm that runs concurrently with the layout-to-circuit extraction and does only cause a small percentage increase of computational complexity. Such an algorithm forms a core of other algorithms for handling various aspects of variability-related problems in practice, such as quick estimates of capacitance mismatch induced by either systematic or random geometric variations, and fast evaluation of circuit performances, e.g. system responses. For validation, we have tested the algorithms using a layout-to-circuit extractor on concrete circuits. In this chapter, we will summarize our findings and give an overall picture of the major results presented in this thesis.

We first present a fast algorithm for the sensitivity computation, which is applicable for BEM based LPE tools. The algorithm can be derived from an adjoint method (Chapter 3) or a domain-decomposition technique (Appendix C). It is efficient in the sense that both the nominal values of capacitances and their sensitivities with respect to multiple parameters can be obtained with a single system solve. This can be done because the data needed for the sensitivity computation can be found during the standard capacitance extraction, i.e., they are the intermediate data of a

nominal capacitance computation. Therefore, no extra costly computation is required.

To validate its feasibility, we implement the proposed algorithm in a layout-to-circuit parasitics extractor SPACE (Chapter 4). The on-chip interconnect capacitances computation in SPACE is based on the BEM, thus the proposed algorithm can be applied. One feature of SPACE is the use of a window-based scheme as an acceleration technique, which remains valid with the extension of the sensitivity computation. We have subsequently conducted several experiments on the SPACE platform. The results suggest a very high efficiency as desired and a good accuracy. Although a good accuracy should be generally acceptable for most applications, the algorithm should not be limited due to the lack of knowledge of the accuracy lost.

We therefore study the cause of the error and propose a supplementary algorithm for accuracy improvement (Chapter 5). For ease of discussion, we have named the algorithm introduced in Chapter 3 and Appendix C the basic algorithm. A combination of the basic algorithm and the supplementary algorithm leads to an enhanced algorithm which provides very high accuracy results at the cost of a moderate reduce of the efficiency. The enhanced algorithm enables flexibility of the sensitivity computation as it offers optional trade-offs between accuracy and efficiency to users for different application requirements.

The algorithm for sensitivity computation forms a core of other algorithms which tackle different aspects of variation-induced problems in practice. We have developed two extension algorithms in this thesis. Specifically, we have chosen high efficiency as the primary goal of our work and thus it is still the basic algorithm that is adopted in the two extensions (Chapter 6 and Chapter 7).

Chapter 6 demonstrates a technique for computing the statistical properties of interconnect capacitances resulting from line-edge-roughness (LER). Using the proposed algorithm, the nominal values of capacitances as well as their statistical properties accounting for both the systematic and the random variations can be obtained at a negligible extra computational time, compared to the nominal capacitance computation. The fast estimate of LER effects on interconnect capacitances can be very useful for designs of passive components with high-precision requirements. In this context, a real design case is studied. Supported by the measurement results on test chips, our technique successfully estimates the mismatch of

capacitances due to LER.

Calculating the statistical properties of capacitances is, in many cases, not the eventual purpose of modeling manufacturing variabilities. Instead, it is the circuit performance, e.g. the transfer function, that designers care about. Traditionally, the statistical properties of the system response are obtained by the Monte-Carlo approach, which, however, suffers from a huge computational burden due to the need of sampling the parameter space. This problem can be solved by the second extension algorithm (Chapter 7), which achieves zero parameter sampling, by combining the proposed sensitivity-based parameterized parasitics extraction technique and a structure-preserving parameterized model order reduction technique.

Chapter 7 demonstrates a highly efficient methodology for the statistical analysis of RC nets subject to systematic variations. The sensitivity-based layout-to-circuit extraction generates first-order Taylor series approximations of resistances and capacitances with respect to multiple geometric parameter variations. This formulation becomes the input of the parameterized model order reduction, which exploits the explicit parameter dependence to produce a linear combination of multiple non-parameterized transfer functions weighted by the parameter variations. Such a formulation enables a fast computation of statistical properties such as the standard deviation of the transfer function given the process spreads of the technology. Both the extraction and the reduction techniques avoid any parameter sampling. Therefore, the proposed method achieves a significant speed up compared to the traditional Monte-Carlo approaches.

Lastly, we would like to conclude by giving a pictorial overview of the major results of this thesis. As shown in Figure 8.1, with the additional input of the process spreads, besides the layout description and the technology file, the proposed method (sensitivity-based capacitance modeling method) can generate both the nominal capacitances and their sensitivities. These sensitivities can be with respect to multiple parameter variations that could be either systematic or random, or both. Users are provided options of trade-offs between the accuracy and the efficiency so that they can have preference for one over another according to the applications.

Efforts, in this thesis, have been made mainly on exploiting the benefit of the high efficiency of the proposed algorithm. Applications can be various. We have highlighted two possible ones: a quick estimate of capacitance mismatch induced by the LER, which has been deployed in a real design case, and a fast evaluation of circuit performance on RC nets

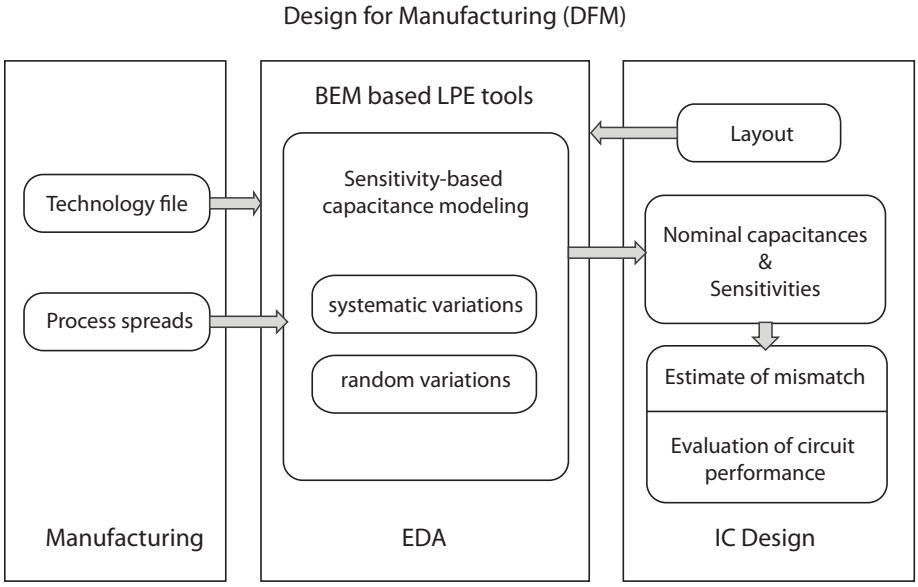

Figure 8.1: *Summary of the major results obtained in this thesis.*

subject to systematic variations, generating the statistical properties of the system response.

APPENDIX **A**

Proof 1: The *adjoint* in the electrostatic field.*

A function $f : \mathbb{R}^3 \to \mathbb{R}$ will be said to be $O(r^{-a})$ *at infinity* if it vanishes at infinity at least as fast as r^{-a}, i.e., if

$$\lim_{\|x\|\to\infty} \|x\|^b \, f(x) = 0, \qquad \forall \, b < a$$

where $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^3$.

Definition:

Let P be the class of all infinitely differentiable scalar functions φ on $\mathbb{R}^3$ such that φ is $O(r^{-1})$ and the derivative of φ is $O(r^{-2})$. Let F be the class of all infinitely differentiable vector fields $\vec{\mathbf{u}}$ on $\mathbb{R}^3$ such that $u_j \in P$, $j = 1, 2, 3$. Define inner products on P and F by

$$\langle \varphi, \psi \rangle_P \triangleq \int_{\mathbb{R}^3} [\varphi(x)\psi(x)] d\Omega \tag{A.1}$$

$$\langle \vec{\mathbf{u}}, \vec{\mathbf{w}} \rangle_F \triangleq \int_{\mathbb{R}^3} [\vec{\mathbf{u}}(x) \cdot \vec{\mathbf{w}}(x)] d\Omega \tag{A.2}$$

for all $\varphi, \psi \in P$, $\vec{\mathbf{u}}, \vec{\mathbf{w}} \in F$, with $\vec{\mathbf{u}} \cdot \vec{\mathbf{w}}$ the standard dot product or scalar product in $\mathbb{R}^3$.

*The content is based on [47]. This is a special case of the Hodge dual in Differential Geometry.

Theorem:

The operator $(-\mathbf{grad})$: $P \to F$ is the adjoint of the operator $(\mathbf{div})$: $F \to P$, i.e.,

$$\langle \mathbf{div}\ \vec{\mathbf{u}}, \varphi \rangle_P = \langle \vec{\mathbf{u}}, -\mathbf{grad}\ \varphi \rangle_F \quad \forall\ (\vec{\mathbf{u}}, \varphi) \in F \times P\ . \tag{A.3}$$

Proof:

Using (A.1), (A.2) and the vector identity

$$(\mathbf{div}\ \vec{\mathbf{u}})\varphi + \vec{\mathbf{u}} \cdot (\mathbf{grad}\ \varphi) = \mathbf{div}\ (\varphi \vec{\mathbf{u}}), \tag{A.4}$$

as well as using the divergence theorem, it follows that

$$\langle \mathbf{div}\ \vec{\mathbf{u}}, \varphi \rangle_P - \langle \vec{\mathbf{u}}, -\mathbf{grad}\ \varphi \rangle_F = \int_{\mathbb{R}^3} \mathbf{div}(\varphi \vec{\mathbf{u}}) d\Omega \tag{A.5}$$

$$= \int_{S_\infty} \varphi \vec{\mathbf{u}} dS \tag{A.6}$$

where S_∞ is the bounding surface that recedes toward infinity. As defined, $\varphi \in P$ is $O(r^{-1})$, $\vec{\mathbf{u}} \in F$ is $O(r^{-2})$, thus $\|\varphi \vec{\mathbf{u}}\|$ is $O(r^{-3})$ at infinity, while $S(r)$ only grows as $4\pi r^2$. Hence,

$$\int_{S_\infty} \varphi \vec{\mathbf{u}} dS = 0 \tag{A.7}$$

and it proves theorem (A.3).

APPENDIX **B**

Proof 2: Electrostatic stored energy*

The work required to move a point charge q from the zero potential reference to a position $\mathbf{r}$ in a potential field $\Phi(\mathbf{r})$ is

$$W = q\Phi(\mathbf{r}) \, . \tag{B.1}$$

If instead, we have a potential field Φ with a continuous charge density ρ, then the work required to increase ρ by a small quantity of charge $\delta\rho$ is

$$\delta W = \int_{\Omega_\infty} \delta\rho(\mathbf{r})\Phi(\mathbf{r}) d\Omega \, . \tag{B.2}$$

This construction could start from the initial condition that all charges are in a reservoir at infinity, and the charge density ρ is built up by adding infinitesimal spatially identical distributions

$$\delta\rho(\mathbf{r}) = \frac{\rho(\mathbf{r})}{K} \tag{B.3}$$

with K a very large number. Note that in linear medium, everytime after $\delta\rho(\mathbf{r})$ is added, the potential is increased proportionally by $\delta\Phi(\mathbf{r})$, i.e.,

*The content is based on [48].

$\delta\Phi(\mathbf{r}) = \frac{\Phi(\mathbf{r})}{K}$. Thus more work needs to be performed with the proceeding of the steps.

Hence, the total amount of work is

$$\begin{aligned} W &= \sum_{k=1}^{K} \int_{\Omega_\infty} \delta\rho(\mathbf{r}) \left[(k-1)\delta\Phi(\mathbf{r})\right] d\Omega \\ &= \left[\sum_{k=1}^{K} (k-1) \int_{\Omega_\infty} \frac{\rho(\mathbf{r})}{K} \frac{\Phi(\mathbf{r})}{K} d\Omega\right] . \end{aligned} \tag{B.4}$$

The number of steps K has to be infinitely large so that the charge added each time is infinitesimally small to avoid energy lost to radiation. Hence using

$$\sum_{k=1}^{K} (k-1) = \frac{K(K-1)}{2} , \tag{B.5}$$

and taking the limit for $K \to \infty$ (B.4) becomes

$$W = \frac{1}{2} \int_{\Omega_\infty} \rho(\mathbf{r})\Phi(\mathbf{r}) d\Omega . \tag{B.6}$$

Next, we will rewrite (B.6) from a field-centric point of view and use the field vectors $\vec{\mathbf{E}}$ and $\vec{\mathbf{D}}$. Substituting the Gauss law,

$$\rho = \nabla \cdot \vec{\mathbf{D}} , \tag{B.7}$$

into (B.6) gives us

$$\begin{aligned} W &= \frac{1}{2} \int_{\Omega_\infty} [\nabla \cdot \vec{\mathbf{D}}(\mathbf{r})]\Phi(\mathbf{r}) d\Omega \\ &= \frac{1}{2} \int_{\Omega_\infty} \nabla \cdot [\Phi(\mathbf{r})\vec{\mathbf{D}}(\mathbf{r})] d\Omega - \frac{1}{2} \int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot [\nabla\Phi(\mathbf{r})] d\Omega . \end{aligned} \tag{B.8}$$

Using the divergence theorem and $\nabla\Phi(\mathbf{r}) = -\vec{\mathbf{E}}(\mathbf{r})$, (B.8) becomes

$$\begin{aligned} W &= \frac{1}{2} \int_{S_\infty} \Phi(\mathbf{r})\vec{\mathbf{D}}(\mathbf{r}) d\mathbf{S} - \frac{1}{2} \int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r})[\nabla\Phi(\mathbf{r})] d\Omega \\ &= \frac{1}{2} \int_{S_\infty} \Phi(\mathbf{r})\vec{\mathbf{D}}(\mathbf{r}) d\mathbf{S} + \frac{1}{2} \int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot \vec{\mathbf{E}}(\mathbf{r}) d\Omega \end{aligned} \tag{B.9}$$

where S_∞ is the bounding surface towarding infinity. Finally, since $\Phi \sim 1/r$ and $\vec{\mathbf{D}} \sim 1/r^2$ as $r \to \infty$, the integral over S_∞ will vanish, thus

$$W = \frac{1}{2} \int_{\Omega_\infty} \vec{\mathbf{D}}(\mathbf{r}) \cdot \vec{\mathbf{E}}(\mathbf{r}) d\Omega . \tag{B.10}$$

APPENDIX C

Sensitivity Computation Using the Domain Decomposition Method

K.J. van der Kolk and **Yu Bi**

In this chapter, we will discuss how the domain decomposition method can be applied to determine capacitance sensitivities.

Consider Figure C.1(a), which shows a cross-section of a metal conductor N_1, with capacitive couplings to other conductors. Our goal is to derive the sensitivities of all capacitances w.r.t. a small variation of a geometric parameter. Assume that there is a negative variation in a parameter p as shown in the figure, which results in an inward displacement $-d$ of the rightmost sidewall of conductor N_1 (d is taken negative for its inward movement). For ease of discussion, we first consider that there is only one panel on this affected sidewall and name it the *victim* panel. By moving the victim panel, all capacitances in the system will change. It can certainly be treated as a new system and the updated capacitances can be computed by generating a new elastance matrix and applying a matrix inversion. However when the system is large and there are multiple parameters, this method becomes prohibitive due to the huge computational

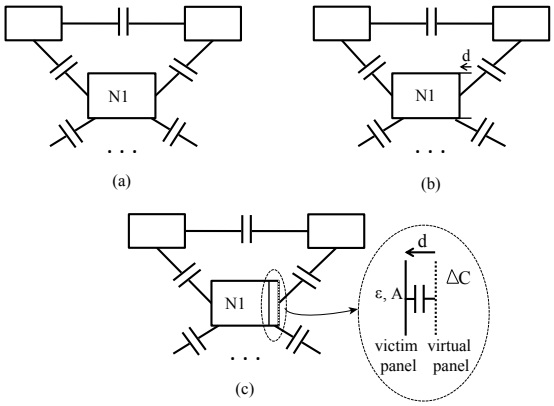

Figure C.1: *Applying domain decomposition to find capacitance sensitivities.*

burden. We thus seek for a faster solution.

What we propose to do is to place a virtual panel at the original position of the victim panel as shown in Figure C.1. This is based on the fact that an equipotential can be theoretically replaced by an infinitely thin but infinitely conductive sheet, thereby creating capacitances both sides. Thus the victim panel is now shielded from the surrounding panels and is only coupled to the virtual panel. In such a way, the capacitive couplings among the panels in the original system remains the same and the only difference is the capacitance ΔC that appears inside the original boundary of conductor N_1. Then the virtual panel is removed and the capacitance matrix is updated by eliminating the associated element. In the following, we will show how to execute this idea.

Since we are computing sensitivities, d is actually infinitesimally small. Therefore there will only be a significant capacitance between the shielded victim panel and the virtual panel. This capacitance can be computed simply from the parallel-plate formula

$$\Delta C = \frac{\varepsilon A}{d} \tag{C.1}$$

where ε is the material permittivity around the panel and A is the area of the panel.

Now, to see how the other capacitances will change, we write down the partial short-circuit capacitance matrix,

$$\bar{\mathbf{C}}_s^1 = \begin{bmatrix} \bar{C}_{s11} & 0 & \bar{\mathbf{C}}_{s1x} \\ 0 & 0 & \mathbf{0} \\ \bar{\mathbf{C}}_{s1x}^T & \mathbf{0} & \bar{\mathbf{C}}_{sxx} \end{bmatrix} + \begin{bmatrix} \Delta C & -\Delta C & \mathbf{0} \\ -\Delta C & \Delta C & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{C.2}$$

which is a block-matrix, partitioned as

$$\begin{bmatrix} 1 \times 1 & 1 \times 1 & 1 \times m \\ 1 \times 1 & 1 \times 1 & 1 \times m \\ m \times 1 & m \times 1 & m \times m \end{bmatrix} \tag{C.3}$$

where m is the number of surrounding panels. The first node and the second node in this matrix corresponds to the virtual panel and the shielded panel respectively. Thus matrix element $\bar{\mathbf{C}}_s^1(1,2) = \bar{\mathbf{C}}_s^1(2,1)$ implements the parallel plate capacitor between these two panels. The row-vector $\bar{C}_{s1x}$ represents the couplings from the virtual panel to the surrounding panels, and the matrix $\bar{\mathbf{C}}_{sxx}$ represents the couplings among the surrounding panels. Since the (shielded) victim panel has no capacitive coupling to the surrounding panels, the second row and the second column in the matrix contains only ΔC as the non-zero elements.

It is also necessary to point out that the original partial short-circuit capacitance matrix before the panel displacement is

$$\bar{\mathbf{C}}_s^0 = \begin{bmatrix} \bar{C}_{s11} & \bar{\mathbf{C}}_{s1x} \\ \bar{\mathbf{C}}_{s1x}^T & \bar{\mathbf{C}}_{sxx} \end{bmatrix} \tag{C.4}$$

where the entries are the same as the non-zero ones in the first term in (C.2).

We proceed by eliminating the first node in $\bar{\mathbf{C}}_s^{\mathbf{1}}$ by taking its Schur complement, giving

$$\bar{\mathbf{C}}_s = \begin{bmatrix} \Delta C & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{C}}_{sxx} \end{bmatrix} - \frac{1}{\bar{C}_{s11} + \Delta C} \begin{bmatrix} -\Delta C \\ \bar{\mathbf{C}}_{s1x}^T \end{bmatrix} \cdot \begin{bmatrix} -\Delta C & \bar{\mathbf{C}}_{s1x} \end{bmatrix}$$

or, collecting terms,

$$\bar{\mathbf{C}}_s = \begin{bmatrix} \dfrac{\bar{C}_{s11}\Delta C}{\bar{C}_{s11}+\Delta C} & \dfrac{\bar{\mathbf{C}}_{s1x}\Delta C}{\bar{C}_{s11}+\Delta C} \\ \dfrac{\bar{\mathbf{C}}_{s1x}^T\Delta C}{\bar{C}_{s11}+\Delta C} & \bar{\mathbf{C}}_{sxx} - \dfrac{\bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C} \end{bmatrix}. \tag{C.5}$$

Substituting the parallel-plate formula (C.1) into (C.5) and taking the derivative towards d gives us

$$\frac{\partial \bar{\mathbf{C}}_s}{\partial d} = \frac{\varepsilon A}{\left(\varepsilon A - \bar{C}_{s11}d\right)^2} \begin{bmatrix} \bar{C}_{s11}^2 & \bar{C}_{s11}\bar{\mathbf{C}}_{s1x} \\ \bar{C}_{s11}\bar{\mathbf{C}}_{s1x}^T & \bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x} \end{bmatrix}. \tag{C.6}$$

Since we are interested in the sensitivity at $d = 0$, we may simplify this as

$$\frac{\partial \bar{\mathbf{C}}_s}{\partial d} = \frac{1}{\varepsilon A} \begin{bmatrix} \bar{C}_{s11}^2 & \bar{C}_{s11}\bar{\mathbf{C}}_{s1x} \\ \bar{C}_{s11}\bar{\mathbf{C}}_{s1x}^T & \bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x} \end{bmatrix}. \tag{C.7}$$

An entry of the above sensitivity matrix (C.7) can subsequently be written as

$$\frac{\partial \bar{C}_{sab}}{\partial d} = \frac{\bar{C}_{s1a}\bar{C}_{s1b}}{\varepsilon A} \tag{C.8}$$

where a and b can be any panels in the original system including the victim panel. This expression shows that the sensitivity between two panels towards d can be computed by their couplings to the victim panel. These capacitive couplings can directly be obtained from the original partial short-circuit capacitance matrix $\bar{\mathbf{C}}_s^0$.

Note that for BEM based extractors, capacitances are also computed from the matrix $\bar{\mathbf{C}}_s^0$. Therefore the sensitivities can be computed along with the extraction of capacitances on-the-fly. This is a key result of the proposed algorithm.

Above we have derived the sensitivity of the partial short-circuit capacitance between a pair of panels (C.8). Using this, we can compute the sensitivity between conductors. This is done by summing over all panels that belong to the same conductor:

$$\frac{\partial C_{sij}}{\partial d} = \frac{1}{\varepsilon A} \sum_{a \in N_i} \sum_{b \in N_j} \bar{C}_{s1a}\bar{C}_{s1b} \tag{C.9}$$

where C_{sij} represents the short-circuit capacitance between conductors N_i and N_j.

However, until now we worked with the short-circuit capacitances and they need to be converted into network capacitances using (2.9a). Thus the coupling capacitance sensitivity with respect to the panel displacement d, becomes

$$\frac{\partial C_{ij}}{\partial d} = -\frac{1}{\varepsilon A} \sum_{a \in N_i} \sum_{b \in N_j} \bar{C}_{s1a} \bar{C}_{s1b}. \tag{C.10}$$

We notice that this equation (C.10) is in fact the same as (3.47), which is named the *panel sensitivity of capacitance*, describing the capacitance fluctuation (among conductors) induced by a small displacement of one panel. Again, using the short-hand notation $\mathcal{C}^*_{ki} = \sum_{a \in N_i} \bar{C}_{ska}$, (C.10) can be expressed as

$$\mathcal{S}_1 = \frac{\partial C_{ij}}{\partial d} = -\frac{1}{\varepsilon A} \mathcal{C}^*_{1a} \mathcal{C}^*_{1b}. \tag{C.11}$$

In fact, the basic idea of the proposed algorithm is very simple and can be illustrated by a simple example. Suppose there are two plates α and β that are capacitively coupled as shown in Figure C.2 (a). Then we move plate β by a short distance and "copy" it to its original place. The distance d between plate β and its copy β^1 is so short that the capacitance between them is significant and the plate β can be considered to be shielded from the other plate α. As indicated in Figure C.2 (b), this capacitance can be calculated as $\Delta C = \frac{\varepsilon A}{d}$, where A is the area of plate β and ε is the material permittivity around it.

Now we remove plate β^1 and the capacitance between plates α and β can be approximated, by neglecting the minor contribution from the fringe fields,

$$C = \frac{C_0 \Delta C}{C_0 + \Delta C} \tag{C.12}$$

which, by substituting the parallel plate formula of capacitor, can be written as

$$\frac{\partial C}{\partial d} = -\frac{\varepsilon A C_0^2}{(\varepsilon A + C_0 d)^2} \tag{C.13}$$

Using $d \approx 0$, we obtain the capacitance sensitivity between plates α and β towards a small displacement of plate β:

$$\frac{\partial C_0}{\partial d} = -\frac{C_0^2}{\varepsilon A}. \tag{C.14}$$

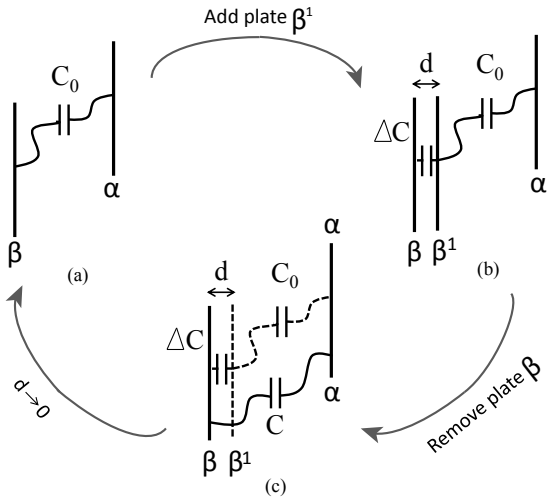

Figure C.2: *Illustrative example with a shielded victim plate and an "added-removed" virtual plate.*

Thus, once we have solved the field problem to get an accurate value of capacitance C_0, we can also calculate its sensitivity towards a small displacement of one of the plates without any extra solve. We only need the area of the plate and the local value of the permittivity. All the other aspects of the geometry and physics of the system, including the conductors, are already *encoded* in the solve of C_0.

Above, we introduced the panel sensitivity of capacitances and derived its computation. However, in most cases, more than one victim panel is needed to approximate a physical variation and its contribution to the capacitance fluctuation has to be evaluated. As addressed in Chapter 2, the sensitivities for systematic variation modeling are defined to be with respect to a geometric parameter variation, for instance the thickness variation of a metal layer or the width variation of a conductor. Such parameter variation leads to small movements of surfaces. These surfaces are named, as in Chapter 3, the *victim surfaces*, of which the displacements equal their associated parameter variations. With a BEM, each surface consist of a set of panels. The panels that belong to the victim surfaces

are thus the *victim panels*, which apparently have the same displacements as the incident victim surfaces and thus the associated parameter variations.

In the following, we will study the case with multiple victim panels. This is done by repeating the steps from (C.1) to (C.8) multiple times. For ease of discussion and without loss of generality, we will derive the capacitance sensitivities w.r.t. one parameter variation with two associated victim panels.

First of all, one of the two victim panels is moved by a small distance d and one virtual panel is added to maintain the original capacitive couplings. Then the virtual panel is eliminated, leading to a partial short-circuit capacitance matrix:

$$\bar{\mathbf{C}}_s^{10} = \begin{bmatrix} \dfrac{\bar{C}_{s11}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \dfrac{\bar{C}_{s12}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \dfrac{\bar{\mathbf{C}}_{s1x}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} \\ \dfrac{\bar{C}_{s21}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \bar{C}_{s22}-\dfrac{\bar{C}_{s12}^2}{\bar{C}_{s11}+\Delta C_1} & \bar{\mathbf{C}}_{s2x}-\dfrac{\bar{C}_{s12}\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C_1} \\ \dfrac{\bar{\mathbf{C}}_{s1x}^T\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \bar{\mathbf{C}}_{s2x}^T-\dfrac{\bar{C}_{s12}\bar{\mathbf{C}}_{s1x}^T}{\bar{C}_{s11}+\Delta C_1} & \bar{\mathbf{C}}_{sxx}-\dfrac{\bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C_1} \end{bmatrix} \tag{C.15}$$

where the first node represents the *displaced* victim panel and the second node represents the other victim panel *to be moved*. The other nodes in the matrix corresponds to the surrounding panels other than the two victims. Also,

$$\Delta C_1 = -\frac{\varepsilon A_1}{d} \tag{C.16}$$

where A_1 is the area of the moved victim panel.

Next, the same steps are repeated for the second victim panel. It is displaced by the same distance d and a virtual panel is added at its original position so that the victim panel is shielded from the surrounding panels,

which results in a partial short-circuit matrix as

$$\bar{\mathbf{C}}_s^2 = \begin{bmatrix} \frac{\bar{C}_{s11}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \frac{\bar{C}_{s12}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & 0 & \frac{\bar{\mathbf{C}}_{s1x}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} \\ \frac{\bar{C}_{s21}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \bar{C}_{s22}-\frac{\bar{C}_{s12}^2}{\bar{C}_{s11}+\Delta C_1} & 0 & \bar{\mathbf{C}}_{s2x}-\frac{\bar{C}_{s12}\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C_1} \\ 0 & 0 & 0 & \mathbf{0} \\ \frac{\bar{\mathbf{C}}_{s1x}^T\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & \bar{\mathbf{C}}_{s2x}^T-\frac{\bar{C}_{s12}\bar{\mathbf{C}}_{s1x}^T}{\bar{C}_{s11}+\Delta C_1} & \mathbf{0} & \bar{\mathbf{C}}_{sxx}-\frac{\bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C_1} \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & \mathbf{0} \\ 0 & \Delta C_2 & -\Delta C_2 & \mathbf{0} \\ 0 & -\Delta C_2 & \Delta C_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{C.17}$$

(C.18)

where this virtual panel corresponds to the third node and

$$\Delta C_2 = -\frac{\varepsilon A_2}{d} \tag{C.19}$$

with A_2 being the area of the second victim panel. Thus this time we eliminate the third node from the matrix, using again the Gaussian elim-

ination:

$$\bar{\mathbf{C}}_s = \begin{bmatrix} \frac{\bar{C}_{s11}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & 0 & \frac{\bar{\mathbf{C}}_{s1x}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} \\ 0 & \Delta C_2 & 0 \\ \frac{\bar{\mathbf{C}}_{s1x}^T\Delta C_1}{\bar{C}_{s11}+\Delta C_1} & 0 & \bar{\mathbf{C}}_{sxx} - \frac{\bar{\mathbf{C}}_{s1x}^T\bar{\mathbf{C}}_{s1x}}{\bar{C}_{s11}+\Delta C_1} \end{bmatrix} - \frac{1}{\bar{C}_{s22} - \frac{\bar{C}_{s12}^2}{\bar{C}_{s11}+\Delta C_1} + \Delta C_2} \cdot$$
$$\begin{bmatrix} \frac{\bar{C}_{s12}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} \\ -\Delta C_2 \\ \bar{\mathbf{C}}_{s2x}^T - \frac{\bar{C}_{s12}\bar{\mathbf{C}}_{s1x}^T}{\bar{C}_{s11}+\Delta C_1} \end{bmatrix} \begin{bmatrix} \frac{\bar{C}_{s21}\Delta C_1}{\bar{C}_{s11}+\Delta C_1} \\ -\Delta C_2 \\ \bar{\mathbf{C}}_{s2x}^T - \frac{\bar{C}_{s21}\bar{\mathbf{C}}_{s1x}^T}{\bar{C}_{s11}+\Delta C_1} \end{bmatrix}^T \tag{C.20}$$

Then terms are collected and (C.16) (C.19) are substituted in the matrix before the derivative of $\bar{\mathbf{C}}_s$ towards d is taken. Since the matrix is too big to be shown, some entries are presented instead.

- The coupling capacitance between the two victim panels:

$$\begin{aligned} \bar{\mathbf{C}}_s(1,2) &= \frac{\bar{C}_{s12}\Delta C_1\Delta C_2}{\bar{C}_{s11}\bar{C}_{s22} - \bar{C}s12^2 + \bar{C}_{s11}\Delta C_2 + \bar{C}_{s22}\Delta C_1 + \Delta C_1\Delta C_2} \\ &= \frac{\varepsilon^2 A_1 A_2 \bar{C}_{s12}}{\varepsilon^2 A_1 A_2 - (\bar{C}_{s11}\varepsilon A_2 + \bar{C}_{s22}\varepsilon A_1)d + (\bar{C}_{s11}\bar{C}_{22} - \bar{C}_{s12}^2)d^2} \end{aligned} \tag{C.21}$$

  Taking the derivative of $\bar{\mathbf{C}}_s(1,2)$ towards d at $d = 0$ gives us

$$\frac{\partial \bar{\mathbf{C}}_s(1,2)}{\partial d}|_{d=0} = \frac{\bar{C}_{s11}\bar{C}_{s12}}{\varepsilon A_1} + \frac{\bar{C}_{s21}\bar{C}_{s22}}{\varepsilon A_2} \tag{C.22}$$

- The second diagonal entry (associated with the second victim panel):

$$\begin{aligned}\bar{\mathbf{C}}_s(2,2) &= \frac{(\bar{C}_{s11}\bar{C}_{s22} + \bar{C}_{s22}\Delta C_1 - \bar{C}_{s12}^2)\Delta C_2}{\bar{C}_{s11}\bar{C}_{s22} + \bar{C}_{s22}\Delta C_1 - \bar{C}_{s12}^2 + \bar{C}_{s11}\Delta C_2 + \Delta C_1 \Delta C_2} \\ &= \frac{\bar{C}_{s22}\varepsilon^2 A_1 A_2 - (\bar{C}_{s11}\bar{C}_{s22} - \bar{C}_{s12}^2)\varepsilon A_2 d}{\varepsilon^2 A_1 A_2 - (\bar{C}_{s22}\varepsilon A_1 + \bar{C}_{s11}\varepsilon A_2)d + (\bar{C}_{s11}\bar{C}_{s22} - \bar{C}_{s12}^2)d^2}\end{aligned} \tag{C.23}$$

Taking the derivative of $\bar{\mathbf{C}}_s(2,2)$ towards d at $d = 0$ gives us

$$\frac{\partial \bar{\mathbf{C}}_s(2,2)}{\partial d}|_{d=0} = \frac{\bar{C}_{s12}^2}{\varepsilon A_1} + \frac{\bar{C}_{s22}^2}{\varepsilon A_2} \tag{C.24}$$

- The capacitive coupling among the surrounding panels:

$$\begin{aligned}\bar{\mathbf{C}}_s(\mathbf{x},\mathbf{x}) = \bar{\mathbf{C}}_{\mathbf{sxx}} &- \frac{\bar{\mathbf{C}}_{\mathbf{s1x}}^T \bar{\mathbf{C}}_{\mathbf{s1x}}}{\bar{C}_{s11} + \Delta C_1} - \frac{(\bar{C}_{s11} + \Delta C_1)\bar{\mathbf{C}}_{\mathbf{s2x}}^T \bar{\mathbf{C}}_{\mathbf{s2x}}}{(\bar{C}_{s22} + \Delta C_2)(\bar{C}_{s11} + \Delta C_1) - \bar{C}_{s12}^2} \\ &- \frac{\bar{C}_{s12}^2 \bar{\mathbf{C}}_{\mathbf{s1x}}^T \bar{\mathbf{C}}_{\mathbf{s1x}}}{(\bar{C}_{s22} + \Delta C_2)(\bar{C}_{s11} + \Delta C_1)^2 - \bar{C}_{s12}^2(\bar{C}_{s11} + \Delta C_1)} + \\ &+ \frac{\bar{C}_{s21}\bar{\mathbf{C}}_{\mathbf{s2x}}^T \bar{\mathbf{C}}_{\mathbf{s1x}}}{(\bar{C}_{s22} + \Delta C_2)(\bar{C}_{s11} + \Delta C_1) - \bar{C}_{s12}^2} \\ &+ \frac{\bar{C}_{s12}\bar{\mathbf{C}}_{\mathbf{s1x}}^T \bar{\mathbf{C}}_{\mathbf{s2x}}}{(\bar{C}_{s22} + \Delta C_2)(\bar{C}_{s11} + \Delta C_1) - \bar{C}_{s12}^2}\end{aligned} \tag{C.25}$$

Taking the derivative of $\bar{\mathbf{C}}_s(\mathbf{x},\mathbf{x})$ towards d at $d = 0$ gives us

$$\frac{\partial \bar{\mathbf{C}}_s(\mathbf{x},\mathbf{x})}{\partial d}|_{d=0} = \frac{\bar{\mathbf{C}}_{\mathbf{s1x}}^T \bar{\mathbf{C}}_{\mathbf{s1x}}}{\varepsilon A_1} + \frac{\bar{\mathbf{C}}_{\mathbf{s2x}}^T \bar{\mathbf{C}}_{\mathbf{s2x}}}{\varepsilon A_2} \tag{C.26}$$

Now, we have the derivative of the partial capacitance matrix towards d with two associated victim panels

$$\frac{\partial \bar{\mathbf{C}}_s}{\partial d} = \begin{bmatrix} \frac{\bar{C}_{s11}^2}{\varepsilon A_1} + \frac{\bar{C}_{s21}^2}{\varepsilon A_2} & \frac{\bar{C}_{s11}\bar{C}_{s12}}{\varepsilon A_1} + \frac{\bar{C}_{s21}\bar{C}_{s22}}{\varepsilon A_2} & \frac{\bar{C}_{s11}\bar{\mathbf{C}}_{\mathbf{s1x}}}{\varepsilon A_1} + \frac{\bar{C}_{s21}\bar{\mathbf{C}}_{\mathbf{s2x}}}{\varepsilon A_2} \\ \frac{\bar{C}_{s12}\bar{C}_{s11}}{\varepsilon A_1} + \frac{\bar{C}_{s22}\bar{C}_{s21}}{\varepsilon A_2} & \frac{\bar{C}_{s12}^2}{\varepsilon A_1} + \frac{\bar{C}_{s22}^2}{\varepsilon A_2} & \frac{\bar{C}_{s12}\bar{\mathbf{C}}_{\mathbf{s1x}}}{\varepsilon A_1} + \frac{\bar{C}_{s22}\bar{\mathbf{C}}_{\mathbf{s2x}}}{\varepsilon A_2} \\ \frac{\bar{C}_{s11}\bar{\mathbf{C}}_{\mathbf{s1x}}^T}{\varepsilon A_1} + \frac{\bar{C}_{s21}\bar{\mathbf{C}}_{\mathbf{s2x}}^T}{\varepsilon A_2} & \frac{\bar{C}_{s21}\bar{\mathbf{C}}_{\mathbf{s1x}}^T}{\varepsilon A_1} + \frac{\bar{C}_{s22}\bar{\mathbf{C}}_{\mathbf{s2x}}^T}{\varepsilon A_2} & \frac{\bar{\mathbf{C}}_{\mathbf{s1x}}^T\bar{\mathbf{C}}_{\mathbf{s1x}}}{\varepsilon A_1} + \frac{\bar{\mathbf{C}}_{\mathbf{s2x}}^T\bar{\mathbf{C}}_{\mathbf{s2x}}}{\varepsilon A_2} \end{bmatrix}$$

which shows that the sensitivity of capacitance between any pair of panels a and b is

$$\frac{\partial \bar{C}_{sab}}{\partial d} = \frac{\bar{C}_{s1a}\bar{C}_{s1b}}{\varepsilon A_1} + \frac{\bar{C}_{s2a}\bar{C}_{s2b}}{\varepsilon A_2} \tag{C.27}$$

Analogously, if there are multiple victim panels incident to a parameter variation d, the above expression (C.27) becomes

$$\frac{\partial \bar{C}_{sab}}{\partial d} = -\sum_{k \in s_p} \left(\frac{1}{\varepsilon A_k} \bar{C}_{sk,a}\bar{C}_{sk,b} \right) \tag{C.28}$$

where s_p is the set of victim panels incident to d. This description shows that capacitance sensitivities with respect to different parameter variations are simply incident to different sets of victim panels. All the sensitivities towards multiple parameter variations can be computed simultaneously once the associated partial short-circuit capacitances are available.

With (C.28), we can obtain the sensitivity of short-circuit capacitance between two conductors i and j by summing over all panels belonging to the same conductor:

$$\begin{aligned} \frac{\partial \bar{C}_{sij}}{\partial d} &= \sum_{k \in s_p} \left(\frac{1}{\varepsilon A_k} \sum_{a \in N_i} \sum_{b \in N_j} \bar{C}_{sk,a}\bar{C}_{sk,b} \right) \\ &= \sum_{k \in s_p} \frac{1}{\varepsilon A_k} \mathcal{C}^*_{ki}\mathcal{C}^*_{kj} \end{aligned} \tag{C.29}$$

Subsequently, converting the short-circuit capacitances into network capacitances (with (2.9a)) gives us the final equation for coupling capacitance sensitivity between conductors towards a parameter variation d:

$$\frac{\partial C_{ij}}{\partial d} = -\sum_{k \in s_p} \frac{1}{\varepsilon A_k} \mathcal{C}^*_{ki}\mathcal{C}^*_{kj}. \tag{C.30}$$

Using (2.9b), we also obtain the sensitivity of the ground capacitance:

$$\frac{\partial C_{gndi}}{\partial d} = \sum_{k \in s_p} \frac{1}{\varepsilon A_k} \mathcal{C}^*_{ki} \left(\sum_{j=1}^{N} \mathcal{C}^*_{kj} \right). \tag{C.31}$$

The sensitivity computation derived by the domain decomposition method is identical to the result (3.45) and (3.46) given by the adjoint field technique.

Bibliography

[1] G. E. Moore, "Cramming more components onto integrated circuits."

[2] C. Angelini, "Intel Core i7-3960X review: Sandy Bridge-E and X79 express," http://www.tomshardware.com/reviews/core-i7-3960x-x79-sandy-bridge-e,3071.html, 2011.

[3] S. Judge, http://www.mobilephonedevelopment.com/archives/1149, 2010.

[4] Gartner, Inc., http://www.gartner.com, 2010.

[5] M. D. Levenson, N. S. Viswanathan, and R. A. Simpson, "Improving resolution in photolithography with a phase-shifting mask," *IEEE Trans. Electron Dev.*, vol. 29, no. 12, pp. 1828–1836, 1982.

[6] O. D. Patterson, X. Dong, P. P. Khargonekar, and V. N. Nair, "Methodology for feedback variable selection for control of semiconductor manufacturing process-Part 1: Analytical and simulation results," *IEEE Trans. Semiconductor Manufacuring*, vol. 16, no. 4, 2003.

[7] J. B. Shyu, G. C. Temes, and K. Yao, "Random error effects in matched MOS capacitors and current sources," *IEEE J. Solid-State Circuits*, vol. 19, no. 6, pp. 948–956, 1984.

[8] M. J. M. Pelgrom and C. J.Duinmaijer, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1440, 1989.

[9] J.-S. Chen and K. T. Kornegay, "Design of a process variation tolerant CMOS Opamp in 6H-SiC technology for high-temperature operation," *IEEE Trans. circuit and systems*, vol. 45, no. 11, 1998.

[10] E. S. Fetzer, "Using adaptive circuits to mitigate process variations in a microprocessor design," *IEEE Design & Test of Computers*, pp. 476–483, 2006.

[11] G. Nabaa, N. Azizi, and F. N. Najm, "An adaptive FPGA architecture with process variation compensation and reduced leakage," *in Proc. Design Automation Conference*, pp. 624–629, 2006.

[12] K. Kang, K. Kim, and K. Roy, "Variation resilient low-power circuit design methodology using on-chip phase locked loop," *in Proc. Design Automation Conference*, pp. 934–939, 2007.

[13] S. Ghosh and K. Roy, "Parameter variation tolerance and error resiliency: new design paradigm for the nanoscale era," *in Proc. of IEEE*, vol. 98, no. 10, 2010.

[14] S. R. Nassif, "Modeling and analysis of manufacturing variations," *in Proc. Custom Integrated Circuit Conference (CICC)*, pp. 223–228, 2001.

[15] J. A. .Power, B. Donnellan, A. Mathewson, and W. A. Lane, "Relating statistical MOSFET model parameter variabilities to IC manufacturing process fluctuations enabling realistic worst case design," *IEEE Trans. Semiconductor Manufacuring*, vol. 7, no. 3, pp. 306–318, 1994.

[16] S. Tasiran and A. Demir, "Smart Monte Carlo for yield estimation," *in Proc. TAU*, 2006.

[17] A. Singhee, S. Singhal, and R. A. Rutenbar, "Practical, fast Monte Carlo statistical static timing analysis: why and how," *in Proc. International Conference on Computer-Aided Design*, pp. 190–195, 2008.

[18] V. Veetil, K. Chopra, D. Blaauw, and D. Sylvester, "Fast statistical static timing analysis using smart Monte Carlo techniques," *IEEE Trans. Computer-Aided Design*, vol. 30, no. 6, pp. 852–865, 2011.

[19] K. Shimohigashi, "DFM in perspective - a challenge and opportunity in nanometer era," *in Proc. VLSI Design, Automation and Test (VLSI-DAT)*, 2006.

[20] W. H. Joyner, S. Rawat, S. R.Nassif, V. Pitchumani, N. Rodriguez, D. Sylvester, C. Bittlestone, and R. Radojcic, "Variation-aware analysis: savior of the nanometer era?" *in Proc. Design Automation Conference*, pp. 411–412, 2006.

[21] P. Gupta and A. B. Kahng, "Manufacturing-aware physical design," *in Proc. International Conference on Computer-Aided Design*, pp. 681–687, 2003.

[22] P. Gupta and F.-L. Heng, "Toward a systematic-variation aware timing methodology," *in Proc. Design Automation Conference*, pp. 321–326, 2004.

[23] Z. Ren, W. Zhang, and J. Falbo, "Computation of parasitic capacitances of an IC cell in accounting for lithography effect," *in Proc. Computational Electromagnetics (CEM)*, 2006.

[24] C. Chiang and J. Kawa, "Three DFM challenges: random defects, thickness variation, and printability variation," *in Proc. Aisa Pacific COnference on Circuits and Systems*, pp. 1099–1102, 2006.

[25] D. Blaauw, K. Chopra, A. Srivastavaa, and L. Scheffer, "Statistical timing analysis: from basic principles to state of the art," *IEEE Trans. Computer-Aided Design*, vol. 27, no. 4, 2008.

[26] A. Asenov, S. Kaya, and A. R. Brown, "Intrinsic parameter fluctuations in decananometer mosfets introduced by gate line edge roughness," *IEEE Trans. Electron Devices*, vol. 50, no. 5, pp. 1254–1260, 2003.

[27] P. Harpe, C. Zhou, X. Wang, G. Dolmans, and H. de Groot, "A 12fJ/Conversion-Step 8bit 10MS/s asynchronous SAR ADC for low energy radios," *in Proc. European Solid-State Circuits Conference*, pp. 214–217, 2010.

[28] A. Ruehli and P. Brennan, "Capacitance models for integrated circuit metallization wires," *IEEE J. Solid-State Circuits*, vol. 10, no. 6, pp. 530–536, 1975.

[29] SPICE (Simulation Program with Integrated Circuit Emphasis), http://bwrc.eecs.berkeley.edu/classes/icbook/spice/.

[30] Z. Lin, C. J. Spanos, L. S. Milor, and Y. T. Lin, "Circuit sensitivity to interconnect variation," *IEEE Trans. Semiconductor Manufacturing*, vol. 11, no. 4, pp. 557–567, 1998.

[31] V. Mehrotra, S. Nassif, D. Boning, and J. Chung, "Modeling the effects of manufacturing variation on high-speed microprocessor interconnect performance," *in Proc. Electron Devices Meetings*, pp. 767–770, 1998.

[32] A. Labun, "Rapid method to account for process variation in full-chip capacitance extraction," *IEEE Trans. Computer-Aided Design*, vol. 23, no. 6, pp. 941–951, 2004.

[33] T. El-Moselhy, I. Elfadel, and D. Widiger, "Efficient algorithm for the computation of on-chip capacitance sensitivities with respect to a large set of parameters," *in Proc. Design Automation Conference*, pp. 906–911, 2008.

[34] K. Agarwal, M. Agarwal, D. Sylvester, and D. Blaauw, "Statistical interconnect metrics for physical-design optimization," *IEEE Trans. Computer-Aided Design*, vol. 25, no. 7, pp. 1273–1288, 2006.

[35] S. Vrudhula, J. M. Wang, and P. Ghanta, "Hermite polynomial based interconnect analysis in the presence of process variations," *IEEE Trans. Computer-Aided Design*, vol. 25, no. 10, pp. 2001–2011, 2006.

[36] Y.-T. Li, Z. Bai, Y. Su, and X. Zeng, "Model order reduction of parameterized interconnect networks via a two-directional arnoldi process," *IEEE Trans. Computer-Aided Design*, vol. 27, no. 9, pp. 1571–1582, 2008.

[37] J. F. Villena and L. M. Silveira, "Spare - a scalable algorithm for passive structure preserving, parameter-aware model order reduction," *IEEE Trans. Computer-Aided Design*, vol. 29, no. 6, pp. 925–938, 2010.

[38] H. Ji, V. Kohaal, Q. Chen, R. Salik, V. Gerousis, Z. Yao, H. Chang, and S. Sarkar, "Rc extraction and timing analysis with on-chip random variations," *CDNLive Cadence Designer Nerwork Conference*, 2006.

[39] "1481-2009 IEEE Standard for Integrated Circuit (IC) Open Library Architecture (OLA)."

[40] A. H. Ajami, K. Banerjee, and M. Pedram, "Modeling and analysis of nonuniform substrate temperature effects on global ULSI interconnects," *IEEE Trans. Computer-Aided Design*, vol. 24, no. 6, pp. 849–861.

[41] B. Wang and P. Mazumder, "An accurate interconnect thermal model using equivalent transmission line circuit," *in Proc. Design, Automation and Test in Europe*, pp. 280–283, 2009.

[42] Y. Bi, N. P. van der Meijs, and D. Ioan, "Capacitance sensitivity calculation for interconnects by adjoint field technique," *in Proc. SPI*, May 2008.

[43] Y. Bi, K. J. van der Kolk, D. Ioan, and N. P. van der Meijs, "Sensitivity computation of interconnect capacitances with respect to geometric parameters," *in Proc. EPEP*, pp. 209–212, October 2008.

[44] R. Ash, *A primer of abstract mathematics.* Cambridge University Press, 1998.

[45] P. Penfield, R. Spence, and S. Duinker, *Tellegen's theorem and electrical networks.* The M.I.T. Press, 1970.

[46] C. A. Desoer and E. S. Kuh, *Basic Circuit Theory.* New York: McGraw-Hill, 1969.

[47] J. L. Wyatt and G. Papadopoulos, "Kirchhoff's laws and continuous media," *IEEE Trans. Circuits and Systems*, vol. CAS-31, no. 7, July 1984.

[48] E. J. Rothwell and M. J. Cloud, *Electromagnetics.* CRC Press, 2001.

[49] S. Ramo, J. Whinnery, and T. van Duzer, *Fields and waves in communication electronics.* John Wiley & Sons, Inc., 1994.

[50] D. Ioan, I. Munteanu, and G. Ciuprina, "Adjoint field technique applied in optimal design of a nonlinear inductor," *IEEE Trans. Magn.*, vol. 34, no. 5, pp. 2849–2852, 1998.

[51] N. Nikolova, J. Bandler, and M. Bakr, "Adjoint techniques for sensitivity analysis in high-frequency structure cad," *IEEE Trans. Microwave Theory Tech.*, vol. 52, no. 1, pp. 403–419, 2004.

[52] Y. Bi, K. J. van der Kolk, and N. P. van der Meijs, "Sensitivity computation using domain-decomposition for boundary element method based capacitance extractors," *in Proc. Custom Integrated Circuit Conference (CICC)*, pp. 423–426, September 2009.

[53] P. Dewilde and E. Deprettere, "The generalized Schur algorithm: approximation and hierarchy," in *Operator theory: advances and applications*. Birkhauser Verlag, 1988, pp. 97–116.

[54] H. Nelis, "Sparse approximation of inverse matrix," Ph.D. dissertation, Delft University of Technology, Delft, The Netherlands, 1989.

[55] N. P. van der Meijs, "Accurate and efficient layout extraction," Ph.D. dissertation, Delft University of Technology, Delft, The Netherlands, 1992.

[56] A. J. van Genderen, "Reduced models for the behavior of VLSI circuits," Ph.D. dissertation, Delft University of Technology, Delft, The Netherlands, 1991.

[57] P. Dewilde and Z. Ning, *Models for large integrated circuits*. Kluwer, 1990.

[58] Y. Bi, S. de Graaf, and N. P. van der Meijs, "Enhanced sensitivity computation for BEM based capacitance extraction using the schur complement technique," *in Proc. Custom Integrated Circuit Conference (CICC)*, September 2011.

[59] F. Zhang, *The Schur Complement and Its Applications*. Springer, 2005.

[60] Y. Bi, P. J. A. Harp, and N. P. van der Meijs, "Efficient sensitivity-based capacitance modeling for systematic and random geometric variations," *in Proc. Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 61–66, January 2011.

[61] P. Harpe, C. Zhou, Y. Bi, N. van de Meijs, X. Wang, K. Philips, G. Dolmans, and H. de Groot, "A 26μw 8 bit 10 ms/s asynchronous

sar adc for low energy radios," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1585–1595, 2011.

[62] K. Patel, T.-J. K. Liu, and C. J. Spanos, "Gate line edge roughness model for estimation of FinFET performance variability," *IEEE Trans. Electron Dev.*, vol. 56, no. 12, pp. 3055–3063, 2009.

[63] R. Jiang, W. Fu, J. M. Wang, V. Lin, and C. C.-P. Chen, "Efficient statistical capacitance variability modeling with orthogonal principle factor analysis," *in Proc. International Conference on Computer-Aided Design*, pp. 683–690, 2005.

[64] H. Zhu, X. Zeng, W. Cai, J. Xue, and D. Zhou, "A sparse grid based spectral stochastic collocation method for variations-aware capacitance extraction of interconnects under nanometer process technology," *in Proc. Design, Automation and Test in Europe*, pp. 1–6, 2007.

[65] J. Cui, G. Chen, R. Shen, S. X.-D. Tan, W. Yu, and J. Tong, "Variational capacitance modeling using orthogonal polynomial method," *in Proc. Great Lakes Symposia on VLSI*, pp. 23–28, 2008.

[66] W. Yu, C. Hu, and W. Zhang, "Variational capacitance extraction of on-chip interconnects based on continuous surface model," *in Proc. Design Automation Conference*, pp. 758–763, 2009.

[67] J. Foucher, G. Cunge, L. Vallier, and O. Joubert, "Design of notched gate processes in high density plasmas," *J. Vac. Sci. Technol. B*, vol. 20, pp. 2024–2031, 2002.

[68] D. L. Goldfarb, A. P. Mahorowala, G. M. Gallatin, K. E. Petrillo, K. Temple, M. Angelopoulos, S. Rasgon, H. H. Sawin, S. D. Allen, M. C. Lawson, and R. W. Kwong, "Effect of thin-film imaging on line edge roughness transfer to underlayers during etch processes," *J. Vaccum Science and Technology B*, vol. 22.

[69] J. Thiault, J. Foucher, J. H. Tortai, O. Joubert, S. Landis, and S. Pauliac, "Line edge roughness characterization with a three-dimensional atomic force microscope: Transfer during gate patterning processes," *J. Vac. Sci. Technol. B*, vol. 23, pp. 3075–3079, 2005.

[70] M. Stucchi, M. Bamal, and K. Maex, "Impact of line-edge roughness on resistance and capacitance of scaled interconnects," *Microelectronic Engineering*, pp. 2733–2737, 2007.

[71] G. W. Reynolds, J. W. Taylor, and C. J. Brooks, "Direct measurement of x-ray mask sidewall roughness and its contribution to the overall sidewall roughness of chemically amplified resist features," *J. Vac. Sci. Technol. B*, vol. 17, pp. 3420–3425, 2005.

[72] Z. Zhu and J. White, "FastSies: A fast stochastic integral equation solver for modeling the rough surface effect," *in Proc. International Conference on Computer-Aided Design*, pp. 674–680, 2005.

[73] H. Marom and M. Eizenberg, "The effect of surface roughness on the resistivity increase in nanometric dimensions," *J. Appl. Phy.*, vol. 99, 2006.

[74] T. Gupta, *Copper Interconnect Technology.* Springer, 2009.

[75] J. Healey, "Current technical trends: Dual damascene & low-K dielectrics," http://www.icknowledge.com/threshold_simonton/techtrends01.pdf, 2002.

[76] Y. Bi, K. J. van der Kolk, J. F. Villena, L. M. Silveira, and N. P. van der Meijs, "Fast statistical analysis of RC nets subject to manufacturing variabilities," *in Proc. Design, Automation and Test in Europe*, pp. 32–37, March 2011.

[77] Q. Chen and N. Wong, "A statistical integral equation method for resistance extraction of conductors with random rough surfaces," *in Proc. International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 411–414, 2006.

[78] T. El-Moselhy, I. M. Elfadel, and B. Dewey, "An efficient resistance sensitivity extraction algorithm for conductors of arbitrary shapes," *in Proc. Design Automation Conference*, pp. 770–775, 2009.

[79] N. K. Verghese, D. J. Allstot, and M. A. Wolfe, "Fast parasitic extraction for substrate coupling in mixed-signal ICs," *in Proc. Custom Integrated Circuit Conference (CICC)*, pp. 121–124, 1995.

[80] T. Li, C-H-Tsai, E. Rosenboum, and S.-M. Kang, "Substrate modeling and lumped substrate resistance extraction for CMOS ESD/Latchup circuit simulation," *in Proc. Design Automation Conference*, pp. 549–554, 1999.

[81] Y. Cheng and M. Matloubian, "Parameter extraction of accurate and scaleable substrate resistance components in RF MOSFETs," *IEEE Trans. Electron Devices*, vol. 23, no. 4, pp. 221–223, 2002.

[82] L. Daniel, O. C. Siong, S. C. Low, K. H. Lee, and J. K. White, "A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models," *IEEE Trans. Computer-Aided Design*, vol. 23, pp. 678–693, 2004.

[83] J. R. Philips, "Variational interconnect analysis via PMTBR," *in Proc. International Conference on Computer-Aided Design*, pp. 872–879, 2004.

[84] J. R. Philips and L. M. Silveira, "Poor man's TBR: A simple model reduction scheme," *IEEE Trans. Computer-Aided Design*, vol. 24, pp. 43–55, 2005.

[85] A. van Genderen and N. van der Meijs, "Reduced RC models for IC interconnections with coupling capacitances," *in Proc. European Design Automation Conference*, pp. 132–136, 1992.

[86] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Computer-Aided Design*, vol. 17, pp. 645–654, 1998.

Summary

With each new generation of IC process technologies, the impact of manufacturing variability is increasing. As such, design optimality is harder and harder to achieve and effective modeling tools and methods are needed to capture the effects of variability in such a way that it is understandable and useful to IC designers.

Our research has been inspired by such needs and the goal of our work is to study and model the effect of manufacturing variations (systematic and random) on on-chip interconnects, and to make it transparent to designers. Specifically, we focus on interconnect capacitances.

We propose a sensitivity-based modeling method. The term *sensitivity* refers to the first derivative of interconnect capacitance with respect to a geometric parameter. Using multiple sensitivities, for multiple relevant geometric parameters, we can estimate the effect of manufacturing variability on the values of the interconnect capacitances.

We have developed an algorithm for sensitivity computation, which is applicable for layout parasitics extraction (LPE) tools based on a boundary element method (BEM). The algorithm consists of two parts, i.e., the basic algorithm and the supplementary algorithm, which offers optional trade-offs between accuracy and efficiency. The basic algorithm is very efficient in the sense that the sensitivities of the capacitances with respect to multiple parameters can be obtained to a smaller accuracy at a fraction of the cost for extracting the nominal capacitances. Solving extra or larger

systems of equations is not necessary. This can be achieved because all the data needed for the sensitivity computation is in fact already available during the nominal capacitance extraction. The supplementary algorithm achieves a high accuracy at a moderate cost of extra computational time. The proposed basic algorithm is, with or without the supplementary algorithm, suitable as a core of other algorithms for handling various aspects of variability-related problems in practice. Two application algorithms have subsequently been developed in this thesis.

The first application provides a technique for computing the statistical properties of interconnect capacitances resulting from line-edge-roughness (LER). Using the proposed algorithm, the nominal values of capacitances as well as their statistical properties can be obtained at a negligible extra computational time, compared to the nominal capacitance computation. The fast estimate of LER effects on interconnect capacitances can be very useful for designs of passive components with high-precision requirements. In this context, a real design case is studied. Supported by the measurement results on test chips, we can conclude that our technique successfully estimates the mismatch of capacitances due to LER.

Calculating the statistical properties of capacitances is, in many cases, not the eventual purpose of modeling manufacturing variability. Instead, designers care about the circuit performance, as e.g. expressed by the frequency response or the transfer function. Traditionally, the statistical properties of the relevant performance parameters are obtained by a Monte-Carlo method. This approach, however, suffers from a huge computational burden due to the need of sampling the parameter space. This problem can be solved by the second application algorithm, which achieves zero parameter sampling, by combining the proposed sensitivity-based parameterized parasitics extraction technique and a structure-preserving parameterized model order reduction technique. It demonstrates a highly efficient methodology to obtain the statistical properties, such as the mean and the standard deviation, of the transfer function of RC nets subject to systematic variations.

During our research, we constantly felt the necessity of a good collaboration and communication with both technologists and IC designers. In fact, we think a major challenge for developing tools serving the DFM purpose is the lack of both silicon validation and statistical foundry data. Finally, we wish to note that the modeling tools should move beyond pure analysis to enable applications that can be used by designers for design

exploration, optimization and achievement of single-pass design.

Samenvatting

De impact van variaties in productieprocessen van geïntegreerde schakelingen neemt toe met iedere volgende procesgeneratie. Het wordt daarom steeds moeilijker om een ontwerp te optimaliseren, en er ontstaat steeds meer behoefte aan doelmatige modelleringstools en –methoden om het effect van de productievariaties voor ontwerpers inzichtelijk en hanteerbaar te maken.

Ons onderzoek is geïnspireerd door bovengenoemde behoeften en het doel van ons werk is om het effect van zowel systematische als toevallige productievariaties te bestuderen, te modelleren en op een transparante manier aan de ontwerpers te presenteren. We richten ons daarbij in het bijzonder op de interconnectcapaciteiten.

Onze methode is gebaseerd op gevoeligheidsanalyse. Met het begrip gevoeligheid wordt hier bedoeld de eerste afgeleide van interconnectcapaciteit naar een geometrische parameter. Wanneer de gevoeligheid voor de verschillende relevante geometrische parameters bekend is, kan de afwijking van de capaciteit als gevolg van een afwijking in de geometrie geschat worden.

We hebben een algoritme ontwikkeld wat de betreffende gevoeligheden snel kan uitrekenen. Het algoritme is bedoeld als basis voor layout-naar-circuit extractoren welke zelf gebruik maken van de boundary element methode (BEM). Het bestaat uit twee delen, te weten het basisalgoritme en het aanvullende algoritme, wat een afweging mogelijk maakt tussen re-

kentijd en nauwkeurigheid. Het basisalgoritme is zeer efficiënt in die zin dat gevoeligheid voor meerdere parameters benaderd wordt voor een fractie van de kosten voor de extractie van de nominale capaciteitswaarden. Het is hierbij niet nodig om een extra of een groter stelsel vergelijkingen op te lossen. Dit is mogelijk omdat alle gegevens die nodig zijn voor de gevoeligheidsberekening reeds aanwezig zijn tijdens de nominale extractie. Het aanvullende algoritme verbetert de nauwkeurigheid ten koste van een beperkte extra hoeveelheid rekentijd. Het basisalgoritme is, met of zonder het aanvullende algoritme, geschikt als kern van andere algoritmen voor praktische variabiliteit-gerelateerde ontwerpproblemen. Vervolgens worden twee van zulke toepassingsalgoritmen in dit proefschrift behandeld.

De eerste toepassing betreft een techniek om de statistische eigenschappen van interconnectcapaciteit te bepalen afhangend van de line-edge-roughness (LER). Met behulp van het voorgestelde algoritme kunnen zowel de nominale capaciteitswaarden als hun statistische eigenschappen bepaald worden, en dit met een verwaarloosbare extra rekentijd. Een dergelijke snelle berekening van het effect van LER op interconnectcapaciteiten kan zeer nuttig zijn bij het ontwerp van passieve componenten met hoge eisen aan de precisie. Bevestigd door praktische meetresultaten op test chips, konden we concluderen dat onze techniek de mismatch van de capaciteiten als gevolg van LER correct kan inschatten.

In veel gevallen is het bepalen van de statistische eigenschappen van de capaciteiten niet het hoofddoel van het modelleren van de variabiliteit. Een ontwerper is meer geïnteresseerd in de prestatie van de schakeling, en daarmee in aspecten zoals de frequentie-responsie of de overdrachtsfunctie. Traditioneel worden de statistische eigenschappen hiervan bepaald met behulp van een Monte-Carlo benadering, wat vanwege de benodigde bemonstering van de parameterruimte een enorme rekenkracht vraagt. Dit probleem kan opgelost worden door een tweede toepassingsalgoritme dat bemonstering van de parameterruimte vermijdt door gebruik te maken van een combinatie van gevoeligheids-gebaseerde extractie van interconnectparameters en een structuur-behoudende techniek voor geparameteriseerde modelreductie. Het levert zo een zeer efficiënte methode om de statistische eigenschappen, zoals gemiddelde en standaardafwijking, te bepalen van prestatie-gerelateerde grootheden van RC-schakelingen die onderhevig zijn aan systematische procesvariaties.

Gedurende dit onderzoek zijn we ons steeds bewust geweest van de noodzaak tot een goede samenwerking en een goede communicatie met

zowel technologiespecialisten als IC-ontwerpers. Sterker nog, we denken dat het ontbreken van zowel siliciumvalidatie als statistische gegevens van het fabricageproces één van de grootste uitdagingen is bij het maken van gereedschap voor DFM. Tenslotte willen we opmerken dat modelleringsgereedschappen meer moeten zijn dan pure analyse-instrumenten, en dat ze toepassingen moeten krijgen in gereedschap voor exploratie en optimalisatie van het ontwerp en het reduceren van het aantal ontwerpiteraties.

Acknowledgment

I finally have a chance to thank my supervisors, collaborators, colleagues, friends and families. There is no way I could have come this far without their generous supports and encouragements.

Firstly, I thank my committee members for their time and efforts reading this thesis, giving valuable comments and attending the defense.

I would like to express my great appreciation to Dr. Nick van der Meijs, who has been my supervisor since 2006, when I started my master project. I am grateful for the freedom he gave me, the encouragements, trust and support during my research journey. I have also been permanently influenced and for the better by his attitude to work and life.

I would like to express my gratitude and respect to Prof. Patrick Dewilde, for his contributions to the STW MICES project, of which I was a member, and his careful reading and commenting of my thesis. I feel honored to be his last doctoral student.

I am indebted to Prof. Wil Schilders, who was my supervisor during my internship at NXP Semiconductors, for his mathematical advice and the care he gave to his students.

Dr. Pieter Harpe deserves my gratitude for cooperating on the topic of Chapter 7 in this thesis, his advice and his friendship.

I am particularly grateful to Prof. Miguel Silveira and Dr. Jorge Villena from INESC, Lisbon for their help and encouragement concerning the timely production of our paper for the DATE conference in 2011.

I would like to give my respect to Prof. Alle-Jan van der Veen, for the easy-going leadership he provides to our research unit. My gratitude also goes to our secretaries Minaksie Ramsoekh and Laura Bruns for their secretarial support and Antoon Frehe for his efficient and effective IT support.

I wish to thank the members of SPACE project for providing a nice platform for the algorithm implementation. Special thanks to Simon de Graaf and Kees-Jan van der Kolk for their inspiring discussions and constant help.

I also gratefully acknowledge the support of the Technology Foundation STW under the MICES project, which made the research I present possible, as well as the contributions of the STW Utilization Committee that was supervising the project for helpful advice and access to information.

In my daily work, I have been surrounded by a friendly group of colleagues and fellow students: Alexander, Amir, Claude, Dony, Geert, Hadi, Hamid, Kun, Mat, Michel, Millad, Mu, Qin, Raj, René, Rob, Rosario, Seyran, Shahzad, Shingo, Sina, Sumeet, Sundeep, Tao, Vijay, Wim, Yiyin, Yuki, Zhifeng, Zijian, etc. My Ph.D. journey would not have been so pleasant without their company, especially Yiyin and Claude who I have spent so many joyful and cheerful times with.

I would like to take the opportunity to thank all my Chinese friends in the Netherlands, especially my former and current housemates, for making me feel like at home. I also want to thank my best friend Cai, for sharing my joy and sorrow.

Last but not least, I wish to thank my families: my parents, for their endless love and support; my parents in-law for their understanding and support of my work; Ran and Zongbo, for their care and encouragements; my son Kaiwen, for being my angel.

My most special thanks go to my husband Zhenyuan. I am blessed to have you to grow up and to grow old with.

Yu Bi
Delft, May 2012

About the Author

Yu Bi was born in Beijing, China, in 1983. She received the Bachelor degree in information engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 2005. and the Master of Science degree in microelectronics from Delft University of Technology, Delft, The Netherlands, in 2007. From 2006 to 2007, she worked as an intern in the Methods and Solutions group at NXP Semiconductors, Eindhoven, The Netherlands. Since 2007, she has been working towards the Ph.D. degree in the Circuits and Systems group (CAS) at Delft University of Technology.

Publications

- Yu Bi, S. de Graaf and N.P. van der Meijs,"Enhanced Sensitivity Computation for BEM Based Capacitance Extraction Using the Schur Complement Technique," in *Proc. CICC*, San Jose, CA, September 2011.
- P.J.A. Harpe, C. Zhou, Yu Bi, N.P. van de Meijs, X. Wang, K. Philips, G. Dolmans and H. de Groot, "A 26μw 8 bit 10 MS/s Asynchronous SAR ADC for Low Energy Radios," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1585-1595, 2011.
- Yu Bi, K.J. van der Kolk, J.F. Villena, L.M. Silveira and N.P. van der Meijs, "Fast Statistical Analysis of RC Nets Subject to Manufactur-

ing Variabilities," in *Proc. DATE*, Grenoble, France, pp. 32-37, March, 2011.

- Yu Bi, P.J.A. Harpe and N.P. van der Meijs, "Efficient Sensitivity-Based Capacitance Modeling for Systematic and Random Geometric Variations," in *Proc. ASP-DAC*, Yokohama, Japan, pp. 61-66, January, 2011.
- Yu Bi and N.P. van der Meijs, "Parameterized RC Extraction Using SPACE," in *Proc. ProRISC*, Veldhoven, Netherlands, November 2009.
- Yu Bi, K.J. van der Kolk and N.P. van der Meijs, "Sensitivity Computation Using Domain-Decomposition for Boundary Element Method Based Capacitance Extractors," in *Proc. CICC*, San Jose, CA, pp. 423-426, September, 2009.
- Yu Bi and N.P. van der Meijs, "Process variation aware modeling of interconnect capacitance", in *Proc. ProRISC*, Veldhoven, Netherlands, November 2008.
- Yu Bi, K.J. van der Kolk, D. Ioan and N.P. van der Meijs, "Sensitivity Computation of Interconnect Capacitances with respect to Geometric Parameters," in *Proc. EPEP*, San Jose, CA, pp. 209-212, October, 2008.
- Yu Bi, N.P. van der Meijs and D. Ioan, "Capacitance Sensitivity Calculation for Interconnects by Adjoint Field Technique," in *Proc. SPI*, Avignon, France, May 2008.
- Yu Bi, N.P. van der Meijs and W.H.A. Schilders, "Effects of Manufacturing Variability on Integrated Passive Components," in *Proc. ProRISC*, Veldhoven, Netherlands, November 2007.

Invited Talks

- "Fast Capacitance and Resistance Modeling Subject to Manufacturing Variations," Holst Centre/IMEC, Netherlands, January 2011.
- "Capacitance Modeling Considering Systematic and Stochastic Variations," Holst Centre/IMEC, Netherlands, March 2010.
- "Capacitance Sensitivity Calculation by Adjoint Field Technique", Mentor Graphics, Grenoble, France, May, 2008.