MODELING AND SIMULATION OF ADVANCED NANO-SCALE VERY LARGE SCALE INTEGRATION CIRCUITS

A Dissertation

by

YING ZHOU

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2010

Major Subject: Computer Engineering

MODELING AND SIMULATION OF ADVANCED NANO-SCALE VERY LARGE SCALE INTEGRATION CIRCUITS

A Dissertation

by

YING ZHOU

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Weiping Shi
Committee Members,	Jiang Hu
	Krzysztof A. Michalski
	Vivek Sarin
Head of Department,	Costas N. Georghiades

May 2010

Major Subject: Computer Engineering

ABSTRACT

Modeling and Simulation of Advanced Nano-Scale Very Large Scale Integration Circuits. (May 2010)

Ying Zhou, B.S.; M.S., Xi'an Jiaotong University, China

Chair of Advisory Committee: Dr. Weiping Shi

With VLSI(very large scale integration) technology shrinking and frequency increasing, the minimum feature size is smaller than sub-wavelength lithography wavelength, and the manufacturing cost is significantly increasing in order to achieve a good yield. Consequently design companies need to further lower power consumption. All these factors bring new challenges; simulation and modeling need to handle more design constraints, and need to work with modern manufacturing processes. In this dissertation, algorithms and new methodology are presented for these problems: (1) fast and accurate capacitance extraction, (2) capacitance extraction considering lithography effect, (3) BEOL(back end of line) impact on SRAM(static random access memory) performance and yield, and (4) new physical synthesis optimization flow is used to shed area and reduce the power consumption.

Interconnect parasitic extraction plays an important role in simulation, verification, optimization. A fast and accurate parasitic extraction algorithm is always important for a current design automation tool. In this dissertation, we propose a new algorithm named HybCap to efficiently handle multiple planar, conformal or embedded dielectric media. From experimental results, the new method is significantly faster than the previous one, 77X speedup, and has a 99% memory savings compared with FastCap and 2X speedup, and has an 80% memory savings compared with PHiCap for complex dielectric media.

In order to consider lithography effect in the existing LPE(Layout Parasitic Extraction) flow, a modified LPE flow and fast algorithms for interconnect parasitic extraction are proposed in this dissertation. Our methodology is efficient, compatible with the existing design flow and has high accuracy.

With the new enhanced parasitic extraction flow, simulation of BEOL effect on SRAM performance becomes possible. A SRAM simulation model with internal cell interconnect RC parasitics is proposed in order to study the BEOL lithography impact. The impact of BEOL variations on memory designs are systematically evaluated in this dissertation. The results show the power estimation with our SRAM model is more accurate.

Finally, a new optimization flow to shed area blow in the design synthesis flow is proposed, which is one level beyond simulation and modeling to directly optimize design, but is also built upon accurate simulations and modeling. Two simple, yet efficient, buffering and gate sizing techniques are presented. On 20 industrial designs in 45nm and 65nm, our new work achieves 12.5% logic area growth reduction, 5.8% total area reduction, 10% wirelength reduction and 770 ps worst slack improvement on average.

To My Family

ACKNOWLEDGMENTS

I wish to express my great thanks to my advisor Professor Weiping Shi. Dr. Shi is a wonderful teacher, has taught me a lot of fundamental VLSI design automation and parasitic extraction knowledge, and shared his professional manner of conducting research. I am very thankful to him for all the time he devoted to scientific discussions with me, as well as for his constant encouragement. Finally, I truly appreciate all his support for my life and work. Especially, he showed his understanding after my daughter was born. Many thanks to my committee members, Dr. Vivek Sarin, Dr. Jiang Hu, and Dr. Michalski; I gained a lot of useful knowledge from them.

Many thanks to Dr. Vivek Sarin for introducing the linear algorithm, which gave me more knowledge of matrix. Many thanks to Dr. Michalski for introducing the electromagnetic theory, which provides a strong theory backup for my research. Many thanks to Dr. Jiang Hu for the great courses on physical design. Many thanks to Dr. Peng Li for introducing the modeling and simulation to me. Many thanks to Dr. Charles Alpert of the IBM Austin Research Lab for opening another window to my life-understand gate-sizing, overall physical design flow. Many thanks to all of my colleagues, Dr. Nam, Dr. Cliff Sze, Dr. Natarajan Viswanathan, Dr. Jarrod Ray, David Papa, who provided a lot of help with my work. Many thanks to Dr. Rouwaida Kanj, and Dr. Sani Nassif for the useful discussion and comments on the work about BEOL effect on SRAM yield analysis. Many thanks to Dr. Frank Liu for the useful disscussion and help on the work about lithography impact on interconnect extraction.

I would like to express my gratefulness to all my officemates during my stay at Texas A&M University. Especially, thanks to Xiang Lu for learning Candence Tools.

Thanks to Shu Yan and Yang Yi for very useful discussions in interconnect extraction. Thanks to Chent-Ta Chiang for the course work discussion and introducing LaTex and many useful hints on that. Thanks to all the people in Zachary room 111, who give me the wonderful and unforgettable memories in my life. Without my friends in room 111, my life would not be so plentiful.

I wish to thank my parents and parents-in-law for their constant encouragement and support. After my daughter was born, they came to the US and helped us take care of Sophia for more than one year. Many thanks to my husband Zhuo Li. He is not only my husband, but also my good friend and mentor in my work. Without his support and encouragement, I could not finish my study and dissertation.

TABLE OF CONTENTS

CHAPTER Page

I INTRODUCTION 1

A. Technique Trend and Background 1
B. Contribution 4

II FAST CAPACITANCE EXTRACTION IN MULTILAYER, CONFORMAL AND EMBEDDED DIELECTRIC USING HYBRID BOUNDARY ELEMENT METHOD 8

A. Background 8
B. Preliminaries 11
1. Equivalent Charge Method (ECM) 12
2. Multilayer Green's Function Method 13
3. HybCap Algorithm 14
4. Correctness 15
C. Kernel Independent Preconditioned Solver 17
D. Ground Plane and Reflective Wall 18
E. Experimental Results 19
1. Multilayer Green's Function 19
2. Reflective Walls with Ground Plane 22
3. Conformal/Embedded Dielectric and Reflective Walls 23
F. Conclusion 23

III INTERCONNECT PARASITICS EXTRACTION CONSIDERING PHOTO-LITHOGRAPHY EFFECTS 28

A. Background 28
B. New LPE Methodology 29
C. Lithography Simulation 30
D. 3D Extraction 33
1. Surface Discretization 34
2. Side Wall Layer Selection 37
3. Shape Correction 43
4. Resistance Extraction 46
5. Inductance Extraction 48
E. Conclusion 48

CHAPTER Page

IV THE IMPACT OF BEOL LITHOGRAPHY EFFECTS ON THE SRAM CELL PERFORMANCE AND YIELD 50

A. Background 50
B. SRAM RC Model 51
1. SRAM RC Model 51
2. RC Extraction for SRAM Cell 53
C. The Methodology for SRAM Performance Analysis 55
D. BEOL Impact Analysis with New Methodology 56
1. BEOL Impact on SRAM Read Delay 57
2. BEOL impact on Stability 59
3. Misalignment Impact on SRAM Performance 61
E. Conclusion 64

V A SLEW BASED C_{EFF} 65

A. Background 65
B. C_{eff} for an RC Network for 10% Point 68
C. Equivalent Output Resistance 73
D. Detailed Comparison 77
E. Statistical Comparison 78
F. Conclusion 82

VI AREA AWARE PHYSICAL SYNTHESIS FLOW 83

A. Background 83
1. Buffer Insertion 85
2. Gate Sizing 87
3. Our Contribution 90
B. Overview of Existing Physical Synthesis Flow 90
C. Iterative EVE 92
D. Area Efficient Timing Driven Gate Sizing 95
E. New Area Efficient Optimization Flow 97
F. Other Practical Techniques 97
G. Experiments 98
1. Iterative EVE vs Single EVE 98
2. Timing Driven Gate Sizing 99
3. Overall Flow Comparison 100
H. Conclusion 102

VII CONCLUSION AND SUMMARY 105

CHAPTER Page

REFERENCES 109

VITA 118

LIST OF TABLES

TABLE Page

I Experimental results for the structure shown in Fig. 6. 24

II Experimental results for industrial test case shown in Fig. 7. 25

III Experimental results for reflective boundary walls and ground plane. 26

IV Experimental results for conformal/embedded dielectric, reflective boundary walls, ground plane and planar dielectrics. 27

V Capacitance comparison between lithography simulated and layout for two elbow conductors. 33

VI Resistance comparison between lithography simulated and layout for two elbow conductors. 33

VII Capacitances error with the different number of etching layers. 43

VIII Capacitances with new LPE methodology for three parallel buses. . . 46

IX Resistance comparison between new and old LPE methodologies for 1x1 elbow example. 48

X Inductance with new LPE strategy for 1x1 elbow example. 48

XI The relative RC value among all layouts in one piece of SRAM cell. . 57

XII SRAM yield analysis. Yield is in sigma. 64

XIII The variables used in the statistical simulation. 78

XIV The QOR comparison for iterative EVE. 99

XV The QOR comparison for area efficient gate sizing. 100

XVI The QOR comparison of baseline and new flow. 104

LIST OF FIGURES

FIGURE		Page
1	The lithography challenge.	2
2	Image distortion due to lithography effect.	2
3	The conductors buried in multilayer dielectric. Planar dielectric structures are XY plane. Embedded dielectric structures are a closed rectangular box region. Conformal dielectric structure is often modeled by embedded dielectric overridden by interconnect metal.	9
4	Multiconductor system conformal/embedded in a multilayer dielectric region. Black boxes are conductors, grey boxes are conformal/embedded dielectrics and dotted lines represent planar dielectric-planar dielectric interfaces.	16
5	Multiconductor system with reflective walls.	18
6	Crossing bus with planar structure. Shade boxes are conductors and dotted lines represent planar dielectric-planar dielectric interfaces.	20
7	Example with 48 conductors and 8 dielectric layer.	21
8	Conformal/embedded dielectric case. Shade boxes are conductors, green box is conformal/embedded dielectrics and dotted lines represent planar dielectric-planar dielectrics interfaces.	22
9	The etched profile vs. layout (top view).	28
10	Traditional LPE methodology.	30
11	New LPE methodology.	31
12	3D profile for elbow conductor.	32
13	Two cases to connect four points.	35
14	One example discretization for elbow-shape conductor.	36

FIGURE Page

15 Original points and projected points in a cross section view of one side wall surface. 38

16 One example approximation for elbow-shape conductor. 42

17 One examples for three parallel buses. 46

18 Resistance computation model. 47

19 Different lithographic profiles from the same layout profile of SRAM with different depth of focus (DOF). 52

20 SRAM MXN array. 53

21 6 transistor SRAM schematic. 54

22 6 transistor SRAM schametic with RC network. 55

23 The methodology flow for SRAM performance and yield analysis for BEOL variations. 56

24 Relative delay variation of ideal, best, nominal and worst RC model vs. basic NoRC model. 58

25 Relative delay varitation of best, nominal and worst. The reference model is ideal RC model. 58

26 Read yield of noRC, ideal RC, best RC, nominal RC and worst RC model. 59

27 Stability yield for noRC, ideal RC, best RC, nominal RC and worst RC model. 60

28 Misalignment. 61

29 Resistance vs. misalignment distance. 62

30 The effect of misalignment on the read delay τ_R variation. The reference model is idea RC model. 63

31 One and two-stage RC circuit. 68

FIGURE Page

32 η vs. α and β when we compute $0.1V_{DD}$ based effective capacitance. 71

33 η vs. α and β when we compute $0.5V_{DD}$ based effective capacitance. 72

34 The output waveform vs. α. $\beta = 1$. 73

35 The output waveform vs. β. $\alpha = 0.2$. 74

36 Comparsion of Eq. (5.5) and the solution of Eq. (5.4). 75

37 CMOS inverter circuit. 75

38 Inverter vs. effective capacitance. 78

39 Pi section vs. effective capacitance. 79

40 Inverter driving Pi section vs. effective capacitances. 80

41 Circuit used for statistical comparison. 80

42 Slew at V_A and V_B. Red squres represent our method. Green circules represent the single delay based effective capacitance method. 81

43 Histogram of percentage slew error. 81

44 Percentage slew error vs. α. 82

45 Horizontal congestion from timing driven physical synthesis flow. . . 85

46 Horizontal congestion from area efficient physical synthesis flow. . . . 86

47 Slew constrait (ns) vs. buffer area relationship is shown in green dots. Slew constraint (ns) vs. signal delay per mm relationship is shown in blue squares. 87

48 Delay vs. cap for a buffer in 45 nm node. Three different input slew values, 10, 20 and 40 ps, are used here. 11XM_0.2_r refers to 11X driving strength buffer, 20 ps input slew and the rising inputs. . 88

49 Delay vs. area for a buffer library in 65 nm node. 89

50 Delay vs. area for a buffer library in 45 nm node. 89

FIGURE Page

51 Flow diagram. 91

52 A simple example for iterative EVE. 92

53 Area aware gate sizing. 96

54 New optimization flow. 97

55 Logic area growth saving compared to baseline flow. 101

CHAPTER I

INTRODUCTION

A. Technique Trend and Background

As Moore's Law predicts, the number of transistors on Integrated Circuit is doubled every 24 months. So it has become a key benchmark for semiconductor development. Integration level, cost, speed, power, compactness and functionality obey the Moore's Law. From ITRS (The International Technology Roadmap for Semiconductors) [1], the minimum feature size used to fabricated the integrated circuit exponentially decreases during last four decades.

During the feature size shrinking, design automation company need face more challenges. Optical lithography is one of them. Optical lithography has been the mainstream technology for volume manufacturing since the earliest days of the microelectronics industry. And it is expected to continue as such through the 32 nm half-pitch technology generation. The minimum half pitch is proportional to the wavelength, and inversely proportional to the numerical aperture (NA). And depth of focuse is proportional to the wavelength, and inversely proportional to NA^2 [1]. In order to get the clearly printed image, the smaller wavelength, and higher NA imaging systems are required. Just as Chris A. Mack, vice president of KLA-Tencor, notes that optical lithography is encountering stiff challenges when moving to deep submicron production. He indicates that conventional dry lithography is encountering a bottleneck, as the numerical aperture (NA) approaches 0.9.

There are many new RET(resolution enhancement technique) such as off-axis illumination (OAI), phase shifting masks (PSM), and optical proximity corrections

---

The journal model is *IEEE Transactions on Automatic Control.*

Fig. 1. The lithography challenge.

(OPC) are being used with imaging systems at 193 nm wavelength. Also including 157nm, extreme ultraviolet (EUV) lithography are in development. The ITRS 2005 update seems to have come to a conclusion. It thinks the 193nm scanner (including the use of wet scanners) will be the mainstream solution at the next two technology nodes shown in Fig. 1 [2]. If one day water-based immersion technology can be extended in its application, the use of fluid rather than air for lithography will be the star technology for the 32nm and 22nm environments.

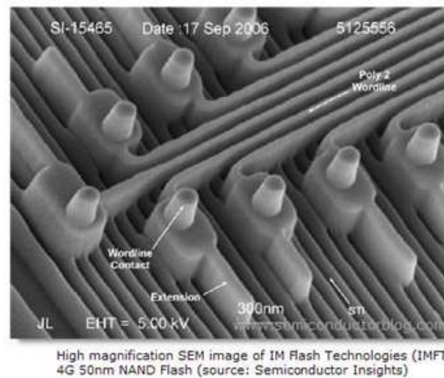


Fig. 2. Image distortion due to lithography effect.

Fig. 2 are the STEM image from Intel. With OAI technique, we still can see the wiggle on the metal boundary. It is obvious that the difference exists, especially around the corners. The traditional LPE methodology can not model and extract the litho/etch effects of nano-scale interconnect, including feature size shrinking, sub-wavelength of light, pattern-dependent effect, etc. To do so would force designer and manufacturers to make a change in entire design flow. Here, we propose a new LPE methodology considering lithographic effect compatible with the existing design flow to solve this problem.

SRAM cell is the main part of memory processor. What does lithographic distortion impact SRAM performance? In traditional SRAM performance analysis, the internal cell short interconnect effect has been neglected. all previous work only study the front end of line (FEOL) such as V_{dd}, V_{th} on SRAM performance. With the feature size shrinking beyond 65 nm, the transistor size and signal wire dimensions continues to decrease with the result of the unit wire capacitance and resistance become bigger. Here we proposed a SRAM RC model to consider the internal RC impact on SRAM performance. Also with this SRAM RC model, we study the lithography effect on SRAM performance.

Except for the lithography challenge due to the technology scaling advances beyond 65nm, the increased wire delay dominance due to finer wire width makes design closure an increasingly challenging problem, a lot of advanced technique skill is used to solve design closure problem. In modern technology, there are over 40 layers of dielectric, etch stop, shield and substrate. Most dielectric structures are planar, but embedded and conformal are also common.

Accurate and fast layout parasitic extraction will help designers a lot during timing verification and signal integrity analysis. It is a big challenge to have a fast scalable algorithm for capacitance extraction to meet the industrial practical require-

ment while dramatically reducing the extraction time, memory usage and the number of the iterations. Traditionally, the dielectric structures are handled by either equivalent charge method (ECM) such as FastCap [3] or PHiCap [4], or multilayer Green's function method (MGM) such as IES^3 [5] and HiCap [6]. Some other efficient methods, such as p-FFT [7] and FastImp [8], do not handle multilayer dielectric. In order to solve this problem, we propose a Hybrid algorithm for Capacitance extraction (HybCap) based on boundary element method.

As ITRS predicts, the density per chip is exponentially increased. The supply voltage will go to 0.4 V in 2016. Small area and low power design is the mainstream in the modern technology. During physical synthesis flow, what is the best way to realize small area and lower power design goal? We proposed an area efficient physical synthesis flow which is embeded in IBM physical synthesis flow.

B. Contribution

All of my work can be classified into three categories, namely (a) methodology, (b) extraction algorithm, (c) circuit simulation and modeling.

(a) Methodology. In this part work, I have two contributions. The first is about new Layout Parasitic Extraction (LPE) methodology. since the image distortion due to the technology limitation can not be neglected, lithography effect should be considered during Layout Parasitic Extraction flow. However, the tradition LPE flow doesn't support the extraction with subwavelength-light, pattern-dependent, etc. We proposed a smart new LPE methodology which made a minor change to include lithography simulation into the tradition LPE flow. Meanwhile, the algorithms for capacitance and resistance extraction are also presented, respectively. Lithography simulation and shape correction steps including a smart dynamic programming based

layer selection scheme are inserted into the traditional LPE methodology to form new LPE methodology. Compared with the traditional methodology, the new methodology will get much more accurate results. The new algorithm significantly reduces the running time of the 3D capacitance solver while keep the good accuracy. As well, we proposed a algorithm on how to do shape approximation. The new methodology are very quick, efficient and compatible with the current flow very well. This technique may be a little outdate, but it is a good option for industry, academy 2 years ago. This work was published in ASPDAC07 and got the best paper award.

Another one is about efficient area aware physical synthesis flow. Due to the demand of the small area and low power design, we proposed this efficient area aware physical synthesis flow based on IBM current physical design and synthesis tool. Based on the observation, we have several techniques to improve the current IBM PDS flow:

- An area efficient iterative slew-tighten approach for slew driven buffering and gate sizing (iterative EVE);
- A simple area efficient timing driven gate sizing method for cell library designs;

With those simple and efficient techniques, we got the overall successful results and improve the quality of run with IBM PDS tool.

(b) Fast extraction algorithm named HybCap. Since the multilayer dielectric are widely used in modern industry, fast and accurate interconnect extraction is important for timing verification and signal integrity analysis. ECM(Equivalent Charge Method) is popularly used to solve this problem in Boundary Element Method(BEM). Multilayer Green's Function is another option to solve this problem based on Boundary Element Method. Here, we proposed a methodology named hybrid to do interconnect exaction in multilayer dielectric media. We have main two contributions.

One is that we implemented MGM with independent solver, and another is that we combine ECM and MGM together to solve the cases for complex dielectric structures, ground plane, reflective walls. The capacitance matrix of our algorithm shows good accuracy with well known field solvers when there are complex dielectric structure and reflective boundary walls. With the experimental results, our method can speed 2x$\tilde{5}$x based on the complex of the interconnect structure, and save more than 80with less iteration number.

(c) Circuit modeling and simulation. This part includes two works. One is about BEOL(Back End of Line) impact on SRAM performance. A new SRAM parasitic analysis model is proposed to capture the internal cell interconnect parasitics RC network. Then we propose an SRAM performance/yield analysis flow which enables litho-aware parasitic extractions and simulation to the existing flows. With our proposed methodology, we can study the back-end-of-line(BEOL) variations on SRAM performance combined with FEOL(front-end-of-line) variations.

Another one is about slew C_{eff} model. With technology scaling, the ratio between the typical output resistance of the output stage of a cell and the interconnect resistance has been steadily rising, making the estimation of the single lumped-capacitance representation of interconnect load more complex. This was observed in [9] and an approach for computing an equivalent *effective capacitance* was proposed. However, one effective capacitance that captures the cell delay cannot accurately predict the slew at the cell output. For this problem, we present a new accurate and efficient approach to estimate the effective capacitance for the output slew of the cell based on a compact analytical model of MOS device operation. The modeling is done with two simple closed form formulas, which are easy to embedded in any STA tools.

The remainder of this dissertation is organized as follows:

- Chapter II : fast capacitance extraction with hybrid boundary element method
- Chapter III: a new methodology for interconnect parasitics extraction considering photo-lithography effects
- Chapter IV: the impact of BEOL Lithography Effects on the SRAM Cell Performance and Yield
- Chapter V: a Slew based C_{eff}
- Chapter VI: area aware physical synthesis flow

CHAPTER II

FAST CAPACITANCE EXTRACTION IN MULTILAYER, CONFORMAL AND EMBEDDED DIELECTRIC USING HYBRID BOUNDARY ELEMENT METHOD

A. Background

Fast and accurate capacitance extraction is very important for timing verification and signal integrity analysis for digital and mixed-signal integrated VLSI chips. Roughly speaking, there are two categories to compute capacitances: *2D/2.5D library looked-up*, where the layout is divided into sections and matched against a precharacterized library to derive the capacitance value, and *3D field solvers*, where the electromagnetic field is solved to compute the capacitance either by integral equations or differential equations. The library method is faster, while the field solver method is more accurate. In this section, we are targeting on fast and accurate field solver since it is important to critical net and clock tree analysis and library generation. We use boundary element method (BEM) as the baseline, which is used by many field solvers such as [3][4][7][10][11].

When the technology shrinks, more dielectric layers are used. One example is shown in Fig. 3. In modern technology, there are over 40 layers of dielectric, etch stop, shield and substrate. Most dielectric structures are planar, but embedded and conformal are also common. It is a big challenge to have a fast scalable algorithm for capacitance extraction to meet the trend of increasing dielectric layers. Traditionally, the dielectric structures are handled by either equivalent charge method (ECM) such as FastCap [3] or PHiCap [4], or multilayer Green's function method (MGM) such as IES^3 [5] and HiCap [6]. Some other efficient methods, such as p-FFT [7] and FastImp [8], do not handle multilayer dielectric.

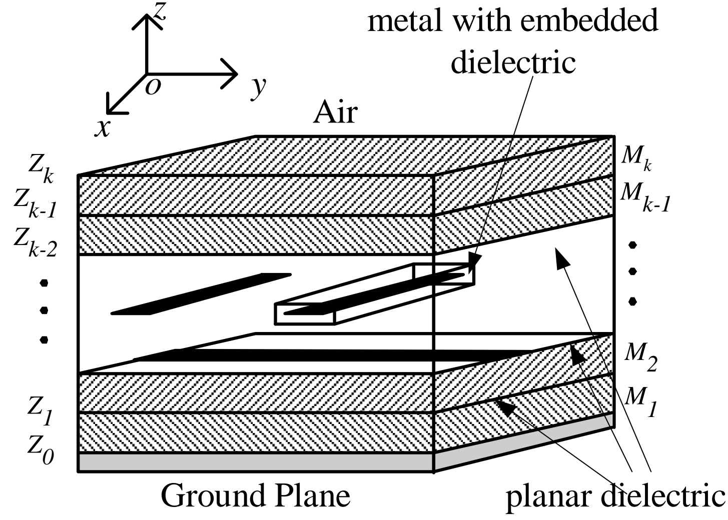


Fig. 3. The conductors buried in multilayer dielectric. Planar dielectric structures are XY plane. Embedded dielectric structures are a closed rectangular box region. Conformal dielectric structure is often modeled by embedded dielectric overridden by interconnect metal.

For the equivalent charge method, *dielectric-dielectric* interfaces are modeled and discretized to satisfy the interface condition [3][4]. The free space Green's function is used to construct the linear system. The charge density on both the *conductor-dielectric* and *dielectric-dielectric* interfaces are solved by iterative methods and acceleration techniques. Capacitances matrix is derived accordingly. ECM works for any dielectric structures with proper geometry processing and the Green's function is simple. However, it requires additional unknown charges on the discretized dielectric-dielectric interfaces and ground planes, thus resulting linear system is much larger. As the number of layers increases, this method becomes impractical.

For multilayer Green's function method, the Green's function for a multilayer planar dielectric medium are derived either directly in spatial domain, i.e., image theory [12], or in spectral domain [5]. The linear system is constructed by the new Green's function and the charge density of each discretized panel on conductors are

solved. MGM generally results in much smaller linear system compared to ECM since it avoids the discretization of dielectric-dielectric interfaces. However, MGM only works for planar dielectric structures and requires the algorithm be kernel independent. In [13], an equivalent dielectric constant approach is used to approximate all planar dielectric layers to only 4 layers by empirical formulas. Then double image Green's function is used, which works well only for up to 4 layers as shown in the paper. However, it is shown in [13] that the equivalent dielectric constants approach has over 15% error when combine two dielectric layers with relative bigger difference in dielectric constant, which is common for sub-130 nm technology. Moreover, the method cannot handle dielectrics other than planar dielectric.

In this section, we propose a Hybrid algorithm for Capacitance extraction (HybCap) based on a kernel independent fast multipole accelerated BEM field solver [4]. The new method combines ECM and MGM, and works extremely efficient for complex dielectric structures. Some of the main features of the algorithms are shown as follows:

- A linear system is built by multilayer Green's function for conductors and the interfaces between conformal/embedded dielectric regions and other dielectric regions;
- The system is transformed to a sparse system and solved with efficient preconditioner;
- The ground plane and reflective wall are handled.

To the best of our knowledge, this is the first hybrid method based on BEM to handle planar, conformal/embedded dielectric, ground plane, and reflective boundary walls.

B. Preliminaries

To compute the self and coupling capacitances, we need to compute the conductor surface charges, given certain conductor electrostatic potentials. In general, the surface charges satisfy the integral equation

$$\Phi(r) = \int_S \sigma(r')G(r,r')d\alpha', \tag{2.1}$$

where r is the observation point, r' is the source point, $\Phi(r)$ is the known conductor surface potential, S is the union of conductor-dielectric interfaces alone or the combination of conductor-dielectric interfaces and the dielectric-dielectric interfaces, σ is the charge density on S, $G(r,r')$ is the Green's function, $d\alpha'$ is the incremental conductor or dielectric surface area, and $r' \in d\alpha'$. Also additional electric displacement vector must satisfy Eq. (2.2) at dielectric-dielectric interfaces (interface conditions):

$$\varepsilon_a \frac{\partial \Phi_+(r)}{\partial n_a} = \varepsilon_b \frac{\partial \Phi_-(r)}{\partial n_a}, \tag{2.2}$$

where r is a point on the dielectric-dielectric interfaces, n_a is the normal to the dielectric interface at r that points into dielectric a; ε_a and ε_b are the permittivities of the corresponding dielectric regions; $\Phi_+(r)$ is the potential at r approached from the ε_a side of the interface, and $\Phi_-(r)$ is the analogous potential for the ε_b side.

Eq. (2.1) can be numerically solved. The $(i,\, j)$ entry of the capacitance matrix is the free charge on the ith conductor when the potential of the jth conductor is 1 V and the other conductors are grounded.

Before the new hybrid method HybCap is proposed, let us first review ECM and MGM methods.

1. Equivalent Charge Method (ECM)

Equivalent charge method is first proposed in [14] for capacitance extraction and followed by many researchers. In this method, surface charge layers are placed at the conductor-dielectric and dielectric-dielectric interfaces, with charge densities $\sigma_c(r)$ and $\sigma_d(r)$, respectively, and the dielectric medium is replaced with free space. Eq. (2.1) now becomes

$$\begin{aligned}\Phi_{ECM}(r) &= \int_{S_c} \sigma_c(r')G_F(r,r')d\alpha' \\ &\quad + \int_{S_d} \sigma_d(r')G_F(r,r')d\alpha',\end{aligned} \tag{2.3}$$

where S_c is the union of conductor-dielectric interfaces and S_d is the union of dielectric-dielectric interface, and $G_F = 1/(4\pi\varepsilon_0||r-r'||)$, r is the observation point and r' is the source point. The interface condition on dielectric-dielectric interfaces becomes

$$\varepsilon_a \frac{\partial \Phi_{ECM+}(r)}{\partial n_a} = \varepsilon_b \frac{\partial \Phi_{ECM-}(r)}{\partial n_a}. \tag{2.4}$$

To numerically compute σ_c and σ_d, the standard Galerkin scheme is used. The conductor-dielectric and dielectric-dielectric interfaces are discretized into $n = n_c + n_d$ small panels, $S_1, S_2, \ldots, S_n$, with n_c panels on conductor-dielectric interfaces and n_d panels on dielectric-dielectric interfaces.

A dense linear system is formed :

$$\begin{bmatrix} P_{cc} & P_{cd} \\ E_{dc} & E_{dd} \end{bmatrix} \begin{bmatrix} q_c \\ q_d \end{bmatrix} = \begin{bmatrix} v_c \\ 0 \end{bmatrix},$$

where q_c and q_d are the vector charges on the conductor-dielectric and dielectric-dielectric interface panels, respectively, and v_c are the vector of potentials on conductor panels. The dimension of potential matrix P is $(n_c + n_d)$. In Galerkin method

the $(i,\, j)$ entry of P_{cc} and P_{cd} are defined as

$$p_{ij} \quad = \quad \frac{1}{A(S_i)A(S_j)} \int_{S_i} \int_{S_j} G_F(r_i, r_j) d\alpha_j d\alpha_i, \tag{2.5}$$

where $A(S_i)$ and $A(S_j)$ are the area of panel S_i and S_j, respectively.

The entries of E_{dc} and off-diagonal entries of E_{dd} are defined as

$$e_{ij} \quad = \quad \frac{\partial}{\partial n_a} \frac{\varepsilon_a - \varepsilon_b}{A(S_i)A(S_j)} \int_{S_i} \int_{S_j} G_F(r_i, r_j) d\alpha_j d\alpha_i. \tag{2.6}$$

The ith diagonal entry $e_{ii} = \dfrac{(\varepsilon_a + \varepsilon_b)}{2\varepsilon_0 A(S_i)}$.

2. Multilayer Green's Function Method

For planar dielectric as shown in Fig. 3, where the permittivity within each layer is uniform in the x- and y- directions, we can derive the multilayer Green's function. Assume a point charge at r' in layer k, we have Poisson's equation:

$$\nabla^2 G_M(r, r') \quad = \quad -\frac{\delta(r - r')}{\varepsilon_k}. \tag{2.7}$$

Many works [12][15] [16] describe how to get the multilayer Green's function. No matter what methods they use, the Green's function must satisfy the continuous conditions $G_{M+} = G_{M-}$ everywhere and boundary conditions

$$\varepsilon_a \frac{\partial G_{M+}(r)}{\partial n_a} \quad = \quad \varepsilon_b \frac{\partial G_{M-}(r)}{\partial n_a} \tag{2.8}$$

at dielectric-dielectric interfaces. In this section, we use the methods similar to [5] to compute multilayer Green's function.

If we use multilayer Green's function as integral equation approach's kernel, Eq. (2.1) becomes

$$\Phi_{MGM}(r) \quad = \quad \int_{S_c} \sigma(r') G_M(r, r') d\alpha', \tag{2.9}$$

where S_c is just the conductor surface, σ is the charge density on S_c, r is the observation point and r' is the source point. The new linear system is $P'_{cc}q'_c = v_c$, where p'_{ij} is evaluated similarly to Eq. (2.5) with G_M as kernel now. It is well known that multilayer Green's function method doesn't need to consider the charge on dielectric-dielectric interfaces. The dimension of matrix P'_{cc} is n'_c, which is much smaller than that of the equivalent charge method which is $n_c + n_d$ even though that n'_c is different from n_c in general. Note that the evaluation of p'_{ij} is slower than p_{ij} in free space due to its complicated formulas, which results in a little overhead of matrix construction.

3. HybCap Algorithm

When the dielectric is nonplanar such as the example shown in Fig. 3, we can not solve this problem with MGM only. On the other hand, we need substantially large memory and long running time with ECM alone. In this section, we present HybCap, which is hybrid capacitance extraction algorithm to take advantage of general dielectric geometry with ECM and smaller system memory of MGM.

HybCap Algorithm includes the following steps:

1. Construct G_M based on given planar dielectric layers' information.

2. Change Eq. (2.1) and Eq. (2.2) to Eq. (2.10) and Eq. (2.11), respectively,

$$\begin{aligned}\Phi_{HybCap}(r) &= \int_{S_c} \sigma_c(r') G_M d\alpha' \\ &\quad + \int_{S_{ed}} \sigma_e(r') G_M d\alpha',\end{aligned} \tag{2.10}$$

and

$$\varepsilon_a \frac{\partial \Phi_{HybCap+}(r)}{\partial n_a} = \varepsilon_b \frac{\partial \Phi_{HybCap-}(r)}{\partial n_a}. \tag{2.11}$$

where S_c is the union of conductor-dielectric interfaces and S_{ed} is the union of

conformal/embedded dielectric-dielectric (conformal/embedded dielectric-planar dielectric or conformal/embedded dielectric-embedded dielectric) interfaces. Note that Eq. (2.11) only applies to S_{ed}.

3. Construct $\boldsymbol{Pq} = \boldsymbol{v}$ with Eq. (2.10) and (2.11) accordingly.

4. Solve the system with the kernel independent hierarchical algorithm such as PHicap [4].

5. Compute capacitance matrix.

4. Correctness

Theorem II.1. *Given a set of conductors, a layered planar dielectric structure, and rectangle regions of embedded dielectric, HybCap method produces the same result as ECM.*

Proof. Based on the previous analysis,

$$\Phi_{ECM}(r) = \int_{S_c} \sigma_c(r')G_F(r,r')d\alpha' + \int_{S_{pd}} \sigma_d(r')G_F(r,r')d\alpha' + \int_{S_{ed}} \sigma_e(r')G_F(r,r')d\alpha', \tag{2.12}$$

$$\varepsilon_a \frac{\partial \Phi_{ECM+}(r)}{\partial n_a} = \varepsilon_b \frac{\partial \Phi_{ECM-}(r)}{\partial n_a}, \tag{2.13}$$

where S_c is the union of conductor-dielectric interfaces, S_{pd} is the union of planar dielectric-planar dielectric interfaces, S_{ed} is the union of conformal/embedded dielectric-dielectric interfaces, the other terminologies are defined in previous sections, Eq. (2.13) applies on S_{pd} and S_{ed} in Eq. (2.12).

Based on the previous analysis, both MGM and ECM are satisfied with the same boundary continuous conditions which are $\Phi_+ = \Phi_-$ and $D_+ = D_-$. In ECM, the electric potential comes from the charge on conductors and the polarization charge on dielectric interface. If we remove the polarization charge on planar dielectric interface, the potential due to the dielectric difference will lose. The potential calculated with Multilayer Green's Function consider the potential caused not only by uniform dielectric but also by the dielectric difference. Therefore, we don't need to consider the polarization charge on dielectric interface in MGM since multilayer Green's function already considers it. Above all, the potential computed by ECM and MGM will be same in planar dielectric. We substitute free space Green's function G_F with multilayer Green's function G_M in Eq. (2.12). Since G_M already consider the potential due to the planar dielectric difference, the second item of the right hand side of Eq. (2.12) is zero. Then Eq. (2.12) becomes Eq. (2.10). Thus, HybCap method is equivalent to ECM method. □

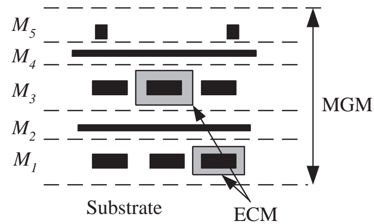


Fig. 4. Multiconductor system conformal/embedded in a multilayer dielectric region. Black boxes are conductors, grey boxes are conformal/embedded dielectrics and dotted lines represent planar dielectric-planar dielectric interfaces.

Note that in ECM, the whole dielectric medium is replaced by free space because the unknown charges are introduced at all dielectric boundary interfaces. In HybCap, however, each conformal/embedded dielectric region is replaced with the

planar dielectric surrounding the conformal/embedded dielectric. For the example shown in Fig. 4, where two dielectric boxes embeds in the layer M_1 and M_3, HybCap first employs ECM for two conformal/embedded dielectric regions and use multilayer Green's function for the whole planar system after transforming.

The new linear system is as follows:

$$\begin{bmatrix} P_{cc} & P_{ce} \\ E_{ec} & E_{ee} \end{bmatrix} \begin{bmatrix} q_c \\ q_e \end{bmatrix} = \begin{bmatrix} v_c \\ 0 \end{bmatrix},$$

where q_c and q_e are the vector charges on the conductor-dielectric and conformal/embedded dielectric-dielectric interface panels, respectively, and v_c are the vector of potentials on conductor panels. Note that p_{ij} is directly derived by multilayer Green's function, and e_{ij} is the electrical filed intensity derived by the corresponding p_{ij}.

C. Kernel Independent Preconditioned Solver

With multilayer Green's function, kernel dependent BEM field solver, such as FastCap [3] can not be used for hybrid method. HybCap algorithm uses PHiCap [4] as the underlying solver due to its kernel independence characteristic and efficient preconditioned solver. First, matrix P is hierarchically built on discretized panels of conductor-dielectric and conformal/embedded dielectric-dielectric interfaces with multilayer Green's function. Then the dense system $Pq = v$ is transformed to an equivalent sparse system $\tilde{P}\tilde{q} = \tilde{v}$. An incomplete factorization preconditioner for $\tilde{P}$ is computed next. Finally, $\tilde{P}\tilde{q} = \tilde{v}$ is solved by preconditioned GMRES or CG method.

For a system with n_c conductor panels, n_p planar dielectric-planar dielectric interface panels, and n_e embedded/conformal dielectric-dielectric interface panels, the dimension of matrix P is $n_c + n_e + n_p$ if ECM method is used alone. For HybCap algorithm, however, the dimension of matrix P could be from $n_c + n_e$ to $n_c + n_e + n_p - n_j$ de-

pending on the realistic chip structure assuming same discretization being performed (we can always model one or more planar dielectric-planar dielectric interfaces with ECM), where n_j is the minimum number of panels for one planar dielectric-planar dielectric interface among all k planar dielectric-planar dielectric interfaces.

D. Ground Plane and Reflective Wall

Ground planes is a perfect electric conductor (PEC), which can be easily modeled. In [5], transmission line theory is used and the ground plane can be modeled as a short circuit line.

One example of the reflective walls (reflective boundary walls), also called Neumann boundary, is shown in Fig. 5. The normal of electric field on the reflective boundary walls is zero.

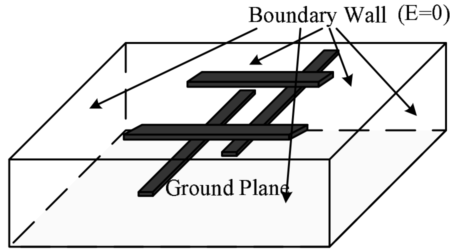


Fig. 5. Multiconductor system with reflective walls.

Reflective boundary walls can be easily modeled by our HybCap algorithm with treating reflective boundary walls as dielectric interfaces where $\varepsilon_a = \varepsilon_i$ and $\varepsilon_b = 0.0$

for panel i on the boundary. Now Eq. (2.10) changes to

$$\begin{aligned}\Phi_{HybCap}(r) &= \int_{S_c} \sigma_c(r')G_M d\alpha' \\ &+ \int_{S_{ed}} \sigma_e(r')G_M d\alpha' \\ &+ \int_{S_{bw}} \sigma_b(r')G_M d\alpha',\end{aligned} \tag{2.14}$$

where S_{bw} is the reflective boundary wall surfaces. We also need to add one more interface condition

$$\varepsilon_i \frac{\partial \Phi_{HybCap}(r_i)}{\partial n_i} = 0, \tag{2.15}$$

where r_i is on the interface of reflective boundary walls. The linear system is rewritten as follows;

$$\begin{bmatrix} P_{cc} & P_{ce} & P_{cb} \\ E_{ec} & E_{ee} & E_{eb} \\ E_{bc} & E_{be} & E_{bb} \end{bmatrix} \begin{bmatrix} q_c \\ q_e \\ q_b \end{bmatrix} = \begin{bmatrix} v_c \\ 0 \\ 0 \end{bmatrix},$$

where q_b denotes the vector of charges on boundary panels, P_{cb} is the coefficient between conductor surface panels and boundary wall panels, E_{eb} is the coefficient between conformal/embedded dielectric-dielectric interface panels and boundary wall panels. E_{bb} evaluates self term and coefficient between boundary wall panels. Other symbols are same as previous. Solve this linear system, then we can derive the capacitance value.

E. Experimental Results

1. Multilayer Green's Function

The experiments are executed on a 3.20GHz Intel Xeon CPU with 8GB memory. The average error E_{avg} in the capacitance matrix C' is defined as $||C-C'||_F/||C||_F$, where

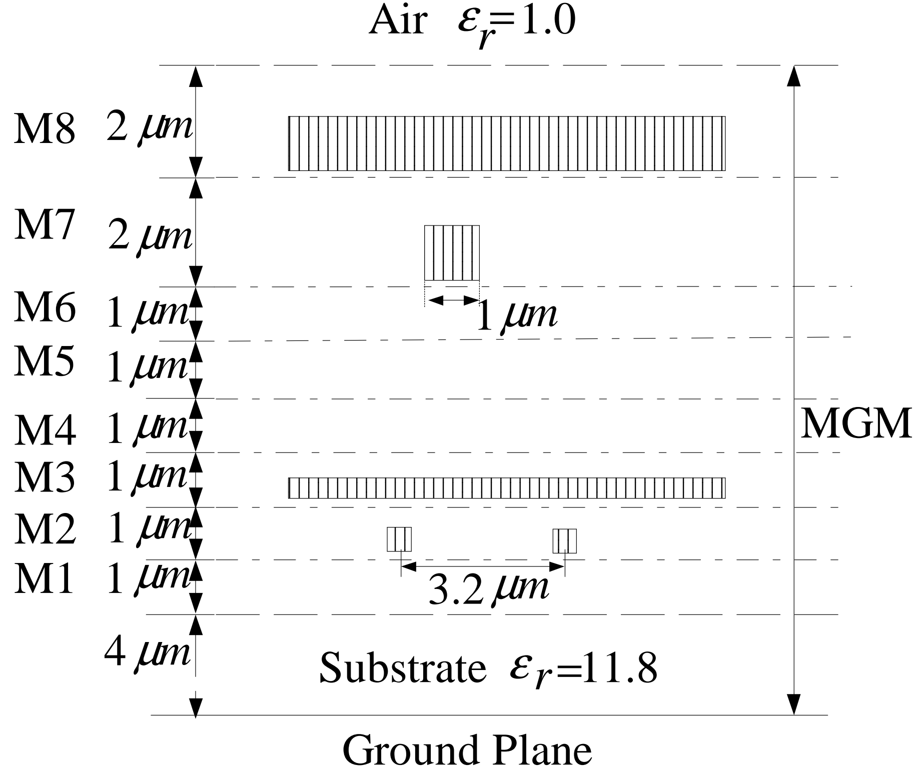


Fig. 6. Crossing bus with planar structure. Shade boxes are conductors and dotted lines represent planar dielectric-planar dielectric interfaces.

$||\cdot||_F$ denotes the Frobenius norm. In this case, there are only planar dielectric layers, $\varepsilon_{substrate} = 11.8$, $\varepsilon_{M_1,M_2,M_7} = 3.9$, $\varepsilon_{M_3,M_4,M_8} = 7.0$, $\varepsilon_{M_5,M_6} = 2.5$, and $\varepsilon_{Air} = 1.0$. Layer M2 and M8 have two buses each whereas layer M3 and M7 have one conductor each. The other dimension is shown in Fig. 6.

We compare FastCap 2.0 [17], PHiCap [4] with HybCap. The comparison results are shown in Table I. From the table, HybCap is almost 300 times faster than FastCap with 99% memory saving. HybCap is almost 30 times faster than FastCap with 98% memory saving. With 10% of discretized panels, HybCap achieves very good accuracy.

Another benchmark is an industrial test case containing eight layers of dielectric

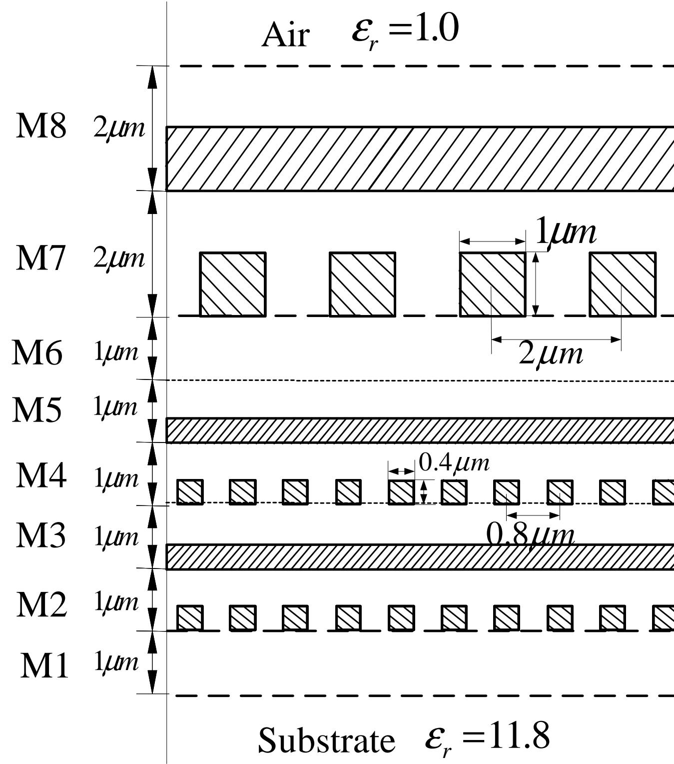


Fig. 7. Example with 48 conductors and 8 dielectric layer.

and 48 conductors from [4] and we modified it by adding the ground plane on the bottom. The case structure is shown in Fig. 7. Metal conductors are shade regions. Relative permittivity of M1 is 3.9, M2-M6 is 2.5 and M7-M8 is 7.0. Layers M2-M5 have ten conductors each whereas layer M7 and M8 have four conductors each. The computation result is shown in Table II. FastCap can not solve these examples because of prohibitive time and memory requirement, therefore we compare HybCap with PHiCap. Few items are compared shown in Table II because of the number of the conductors. From Table II, HybCap is 5 times faster than PHiCap with 85% memory saving. The number of panels is almost 1/4 of PHiCap since no dielectric panels are modeled.

2. Reflective Walls with Ground Plane

In this experiment, the example in Fig. 6 is added with reflective boundary walls. The boundary wall size is $20 \times 20 \times 16\ \mu m^3$.

We compare our experimental results with FastCap and PHiCap. The comparison results are shown in Table III. The relative error is within 2% entries with 90 times speedup and 0.5% memory usage compared with FastCap and 2 times speedup and 20% memory usage compared with PHiCap.

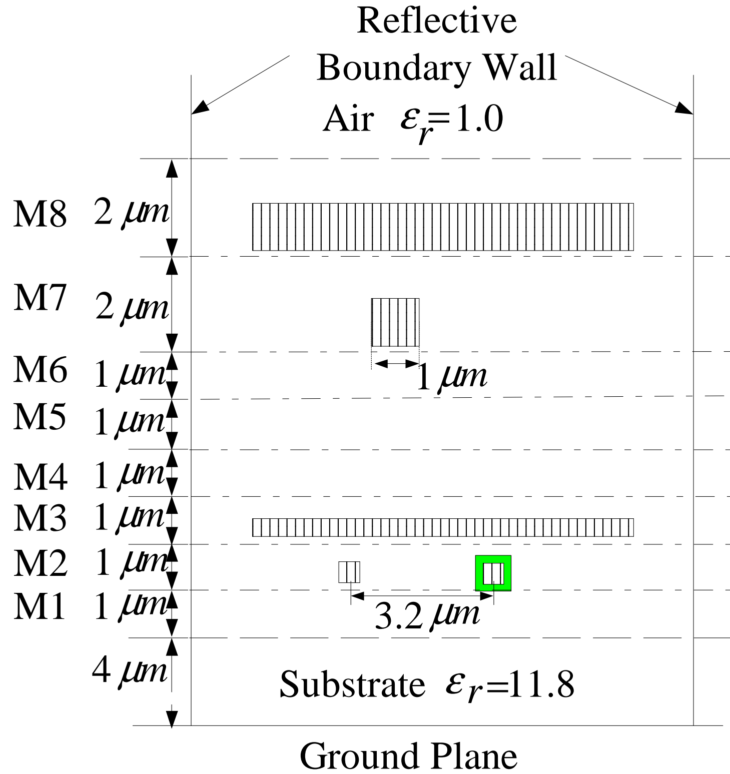


Fig. 8. Conformal/embedded dielectric case. Shade boxes are conductors, green box is conformal/embedded dielectrics and dotted lines represent planar dielectric–planar dielectrics interfaces.

3. Conformal/Embedded Dielectric and Reflective Walls

In this case, we consider the conformal/embedded dielectric, reflective boundary walls, ground plane and planar dielectric all together. The physical structure is shown in Fig. 8. The green area is the embedded dielectric. The boundary wall size is $20 \times 20 \times 16 \mu m^3$. All of other parameters are same as the first case in section 1 and $\varepsilon_{ed} = 2.0$.

The result is shown in Table IV. Compared with FastCap, HybCap gives about 70 times speedup and 99% memory saving. Compared with PHiCap, HybCap gives about 2 times speedup and 80% memory saving.

F. Conclusion

In this chapter, we present an efficient hybrid boundary element method for complex dielectric structures, ground plane, reflective walls. The new method optimally combines equivalent charge method and multilayer Green's function to meet the industrial practical requirement while dramatically reducing the extraction time, memory usage and the number of the iterations. The capacitance matrix of our algorithm shows good accuracy with well known field solvers when there are complex dielectric structure and reflective boundary walls.

Table I. Experimental results for the structure shown in Fig. 6.

Capacitance (aF)	FastCap (order=2)	PHiCap		HybCap	
			Error(%)		Error(%)
C_{11}	897.6	908.8	1.24	909.9	1.37
C_{22}	899.5	909.3	1.09	910.4	1.21
C_{33}	1168.0	1187.8	1.70	1177.4	0.80
C_{44}	1203.0	1214.0	0.91	1212.2	0.75
C_{55}	1308.0	1335.1	2.07	1324.3	1.25
C_{66}	1265.0	1287.8	1.80	1278.9	1.10
C_{12}, C_{21}	72.6	74.4	2.48	73.9	1.79
C_{13}, C_{31}	254.3	261.5	2.83	259.9	2.20
C_{14}, C_{41}	40.5	41.2	1.73	40.7	0.49
C_{15}, C_{51}	18.1	19.1	5.52	18.7	3.31
C_{16}, C_{16}	19.3	20.1	4.15	19.5	1.04
E_{avg}(%)	-	1.80		1.08	
Time(s)	476.92	38.62		1.28	
Iteration	14.7	2		1	
Memory(MB)	4027.2	200		4.3	
Panel	119296	168743		8316	

Table II. Experimental results for industrial test case shown in Fig. 7.

Capacitance (aF)	PHiCap	HybCap	
			Error(%)
C_{11}	854.1	854.9	0.09
C_{22}	992.9	993.6	0.07
C_{33}	1000.9	999.1	0.18
C_{12}, C_{21}	299.0	297.7	0.43
C_{13}, C_{31}	21.3	21.6	1.41
E_{avg}(%)	-	3.4	
Time(s)	138.65	23.59	
Iteration	2.04	1.96	
Memory(MB)	330	48	
Panel	231559	54832	

Table III. Experimental results for reflective boundary walls and ground plane.

Capacitance	FastCap	PHiCap		HybCap	
(aF)	(order=2)		Error(%)		Error(%)
C_{11}	895.3	905.5	1.14	906.4	1.24
C_{22}	896.7	907.0	1.17	907.5	1.20
C_{33}	1159.0	1181.5	1.94	1170.7	1.01
C_{44}	1188.0	1199.8	0.99	1199.3	0.95
C_{55}	1229.0	1252.5	1.91	1244.9	1.29
C_{66}	1155.0	1173.2	1.58	1169.5	1.26
C_{12}, C_{21}	75.1	76.6	2.00	76.3	1.60
C_{13}, C_{31}	258.4	266.6	3.17	264.4	2.32
C_{14}, C_{41}	46.4	46.9	1.08	45.9	1.08
C_{15}, C_{51}	26.6	27.7	3.76	26.8	0.75
C_{16}, C_{16}	28.7	29.6	3.14	28.4	1.05
E_{avg} (%)	-	1.78		1.18	
Time(s)	466.8	13.42		4.93	
Iteration	20.3	1.67		1	
Memory(MB)	2634.7	79		13.0	
Panel	123008	69583		19380	

Table IV. Experimental results for conformal/embedded dielectric, reflective boundary walls, ground plane and planar dielectrics.

Capacitance (aF)	FastCap (order=2)	PHiCap		HybCap	
			Error(%)		Error(%)
C_{11}	891.5	903.9	1.39	904.5	1.46
C_{22}	702.5	691.9	1.27	693.6	1.27
C_{33}	1120.0	1147.4	2.45	1135.9	1.41
C_{44}	1187.0	1199.3	1.04	1198.7	0.99
C_{55}	1228.0	1252.2	1.97	1244.7	1.36
C_{66}	1154.0	1172.8	1.63	1169.1	1.31
C_{12}, C_{21}	57.2	58.8	2.72	58.5	2.27
C_{13}, C_{31}	268.5	271.1	0.97	268.9	0.15
C_{14}, C_{41}	48.3	47.8	1.03	46.8	3.10
C_{15}, C_{51}	27.9	28.4	1.79	27.4	1.79
C_{16}, C_{16}	30.1	30.3	0.66	29.1	3.32
E_{avg} (%)	-	2.65		2.13	
Time(s)	467.9	14.27		6.05	
Iteration	20.2	2		1.17	
Memory(MB)	3101.5	82		14	
Panel	123062	70529		20326	

CHAPTER III

INTERCONNECT PARASITICS EXTRACTION CONSIDERING PHOTO-LITHOGRAPHY EFFECTS

A. Background

As the feature sizes decrease, interconnect variation is playing a greater role in circuit performance. In [18], a process model for sensitivity to different variations is proposed for a clock tree. In [19], a methodology is proposed to model the effect of systematic intra-die variations on circuit performance. In [20], an integrated variation analysis technique is proposed that considers both the effects of systematic and random variation. In [21], with a variational order reduction technique, authors show that interconnect variation can cause up to 25% clock skew variability in a microprocessor design. All these studies are based on accurate parasitic extraction data. Therefore, efficient and accurate extraction of interconnect parasitics under process variation becomes increasingly important. Due to sub-wave lithography effects and process

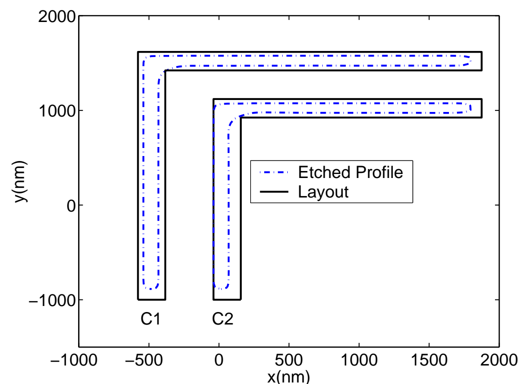


Fig. 9. The etched profile vs. layout (top view).

variations, such as mask size, etching speed, temperature, exposure dose and focal

position variations, it is difficult to reliably print the image intended by the designer. Therefore, the features fabricated mismatch from those drawn in the layout. Fig. 9 shows the top view of the layout and etched profile simulated by PROLITH[1] of a pairs of elbow structures. It is obvious that the difference is significant, especially around the corners. In this example, the error of RC parasitic due to lithographic effect is 20%.

The traditional LPE methodology can not model and extract the litho/etch effects of nano-scale interconnect, including feature size shrinking, subwavelength of light, pattern-dependent effect, etc. To do so would force designer and manufacturers to make a change in entire design flow. In the rest part of this section, we propose a new LPE methodology considering lithographic effect compatible with the existing design flow.

B. New LPE Methodology

The traditional LPE methodology flow is illustrated in Fig. 10. First, the common interconnect structures and the technology files are input to the 3D field solver. The field solver generates a pattern library to be used for layout parasitic extraction (LPE) tools. LPE tools then read the circuit layout and the pattern library to compute the parasitics of the entire circuit. No process variations or lithography effects are considered in the traditional flow.

With advanced sub-wavelength lithography and etching technology, the distortion is so severe and unpredictable that litho/etching software are developed to simulate this complex process.

Considering the litho/etching effect, we embedded lithography simulation into

[1]PROLITH is a trademark of KLA-Tencore Corporation.

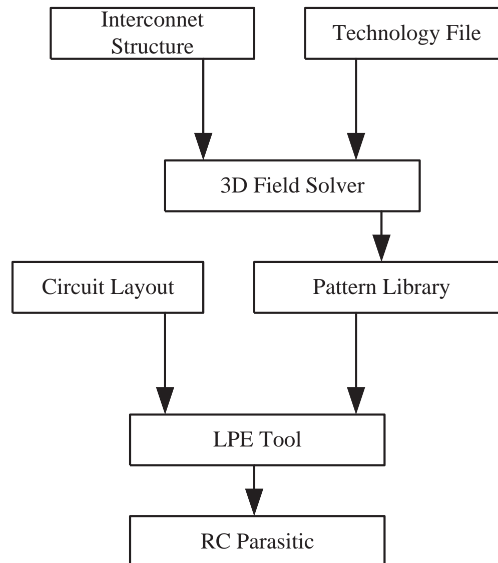


Fig. 10. Traditional LPE methodology.

the LPE flow. Fig. 11 is the new methodology proposed in this section. Two new procedures are added into the new methodology: lithography simulation and shape correction. The shape correction algorithm is used to simplify the complicated geometry from lithography simulation, so that current LPE tools can handle the etched profile, such as shown in Fig. 9. It is clear that the new LPE methodology improves the accuracy of parasitic extraction, and fits well into the existing design flow.

C. Lithography Simulation

Photo-lithography modeling and simulation have been used in the industry for about 30 years. Due to its speed and cost-effectiveness, lithography simulation is widely used to study the process development, determination of sensitivity to manufacture variables, mask design verification and yield analysis. Modern lithography simulation engine can provide accurate process models for the current lithography sequence. In

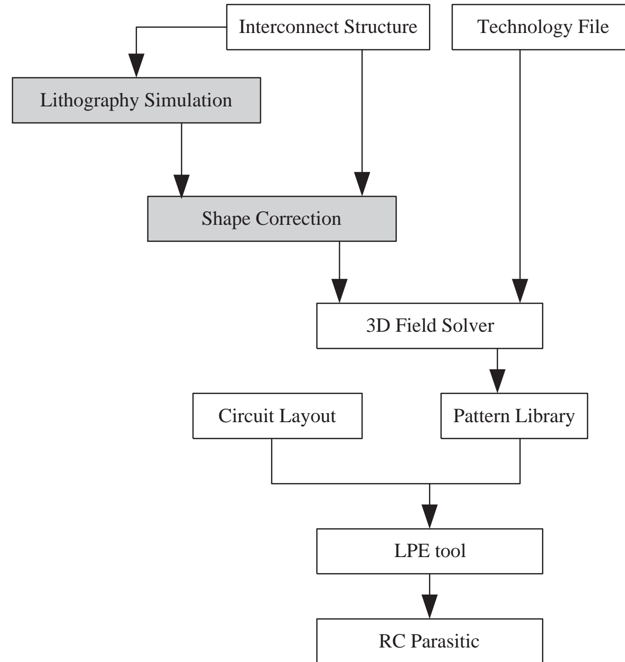


Fig. 11. New LPE methodology.

this section, we use 3D lithography simulator PROLITH version 8.1.2 provided by KLA-Tencor, to study the lithographic effect on interconnect. In our experiments, 90-nm technology is used with 193-nm UV light assumed as the lithographic light source.

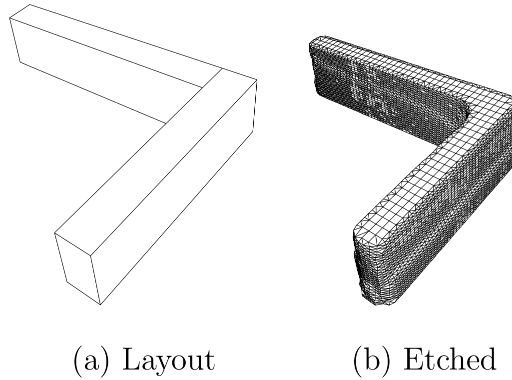
The lithography simulation step of our new LPE methodology includes four steps. First, the typical 3D structures are selected from the interconnect library. Second, the proper masks with OPC are derived for lithography simulation, and determine the process window and optimization of the manufacture parameters. Third, based on the masks and proper processing parameters selected, the lithography simulation is carried out to produce the final 3D geometry of the interconnect structures. Finally, the lithography images are post-processed to a GDS2 format.

The outputs from lithography simulators PROLITH after processed are contours

at different elevation. In other words, the output are points in the X-Y plane, organized for different heights, which we call layers. Note that such layer definition is different from metal layer in the traditional way. For each metal layer, there are often several elevations. The output format for the profile in Fig. 9 is as follows:

x	y	z	layer
480.5	-1394.151	0	0
531.5	-1398.848	0	0
⋮	⋮	⋮	⋮
480.5	-1394.4	-3	1
531.5	-1399.1	-3	1
⋮	⋮	⋮	⋮

We reconstruct the 3D profile for litho/etched conductors. The detail algorithm will be presented in next section. Fig. 12(a) and Fig. 12(b) show the original layout profile and the corresponding litho/etched profile, respectively.



(a) Layout (b) Etched

Fig. 12. 3D profile for elbow conductor.

The litho/etched profile is no longer in rectangular shape and hence the current commercial LPE tools can not solve it. The difference of the extracted RC value between them can reach 20%. For example, in the layout shown in Fig. 9, Table V and VI shows the parasitics difference between simulated litho/etch and original layout profile. The parasitic capacitance and resistance considering lithographic effect

are very different from those of the original layout. Note that such difference will become much bigger for 65 and 45 nm technology.

Table V. Capacitance comparison between lithography simulated and layout for two elbow conductors.

Capacitance (aF)	Etched	Layout	Error (%)
C_{11}	110.8	123.9	11.02
C_{22}	93.9	105.9	11.91
C_{12}, C_{21}	57.9	67.7	15.52

Table VI. Resistance comparison between lithography simulated and layout for two elbow conductors.

Resistance (Ω)	Etched	Layout	Error (%)
R_1	0.29	0.23	20.69
R_2	0.23	0.18	21.74

D. 3D Extraction

In the new LPE methodology, lithography simulation is inserted into the old LPE flow. However, it is a big challenge to accurately extract parasitics when the interconnect shape is no longer rectangular. In this section, a surface discretization algorithm, a dynamic programming based algorithm for selecting side wall layers, and a shape correction algorithm are introduced for the irregular shape interconnect capacitance extraction. An algorithm for resistance computation of irregular shapes is also proposed.

1. Surface Discretization

BEM (Boundary Element Method) capacitance extraction algorithms are based on surface discretization which is easier to reconstruct the irregular interconnect shape after lithography simulation. Start from Possion's equation, $\nabla^2\Phi = \rho$, where Φ is the surface potential, ρ is the charge density in dielectric. Discretize the conductor surfaces into small panels and formulate the problem using a linear system $Pq = V$, where $q \in R^n$ is the vector of unknown panel charges, $v \in R^n$ is the vector of known panel potentials, $P \in R^{n\times n}$ is the potential coefficient matrix computed by Green's Function and boundary condition and n is the number of panels.

The first problem that we need to solve is how to discretize irregular shape interconnect surface, that is, how to do 3D reconstruction for the interconnect after lithography simulation. Here, we proposed a surface discretization algorithm for any 3D field solver based on the Boundary Element Method such as FastCap[22] [17], which can handle quadrilateral and triangular shape. In current extraction flow, FastCap is used. The discretization includes two parts: 1) constructing the side wall, and 2) constructing the top and bottom surfaces.

Based on the characteristics of lithography simulation result, we read the simulation output into vector $L[n]$, where $L[i]$ stores the head of a list of points on the ith layer, and n is the number of layers. To construct the side wall from the contours is not a trivial task. Here, to trade off the speed and accuracy, we present a simple yet effective algorithm. The main idea is to connect points in neighbouring layers $L[i]$ and $L[i+1]$ according to the following rules. If point A and B are in layer $L[i]$, and point C and D are in layer $L[i+1]$, as shown in Fig. 13. Let point A' be the projection of point A on the plane of $i+1$th layer. There are two cases which we need to deal with:

1. If line AB and CD are coplanar, then we can connect these four points in clockwise to form a quadrilateral $ABDC$ as shown in Fig. 13(a).

2. If line AB and CD are not coplanar, then we can connect four points to form two triangles. If the distance AD is less than BC, then we generate two triangles ΔADC and ΔABD in clockwise shown in Fig. 13(b). Otherwise, we generate two triangles ΔABC and ΔBCD as shown in Fig. 13(c).

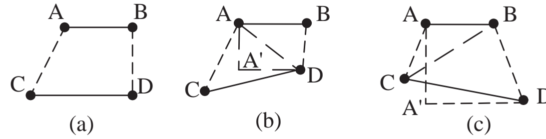
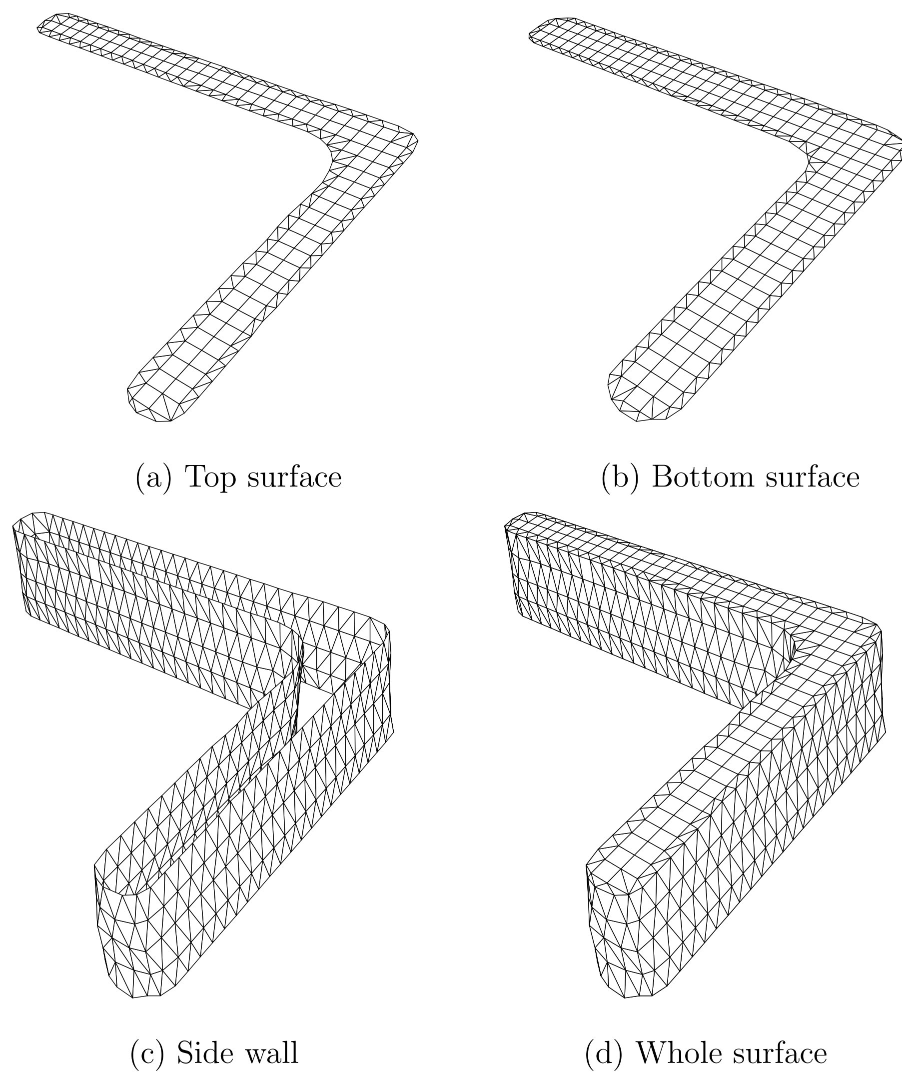


Fig. 13. Two cases to connect four points.

Based on the previous analysis, the complexity of the above algorithm is linear in term of the total sampling points.

For the bottom/top surfaces, we discretize them into squares for the inner part, and triangulate for the fringe area. Some points are inserted in the top/bottom surface. Note that, we can also run standard triangulation algorithm, such as Delaunay triangulation. However, based on our experience, it will either produce too many triangulars or generate many bad aspect ratio shapes, which in turn results in high computational cost or inaccuracy for 3D extraction tools. Fig. 14 shows discretization results on side wall and top/bottom surfaces with our surface discretization algorithm. Now the entire surface of the interconnect is discretized into triangles and quadrilaterals, which will be the input of 3D BEM algorithm. The total number of triangles and quadrilaterals has close relation with the number of the total sampling points in lithography simulation output.



(a) Top surface

(b) Bottom surface

(c) Side wall

(d) Whole surface

Fig. 14. One example discretization for elbow-shape conductor.

2. Side Wall Layer Selection

In general, the data size from the original lithography simulation is relative large due to the large number of layers. From extensive simulation, we found that selecting few middle layers (top and bottom layers must exist) from the original lithography simulation output is accurate enough while significantly speeding up the 3D capacitance solver simulation time. Comparing Fig. 12(b) with Fig. 14(d), where the latter one is generated by taking few middle layers from an original profile with 35 middle layers (37 layers totally) and performing our surface discretization algorithm, there are several times more triangles and quadrilaterals in Fig. 12(b). It takes significantly longer time and bigger memory for 3D BEM based solver to do extraction for the former case than the latter one, while the final capacitances of these two cases are almost same. Therefore, it is acceptable to use fewer middle layers and fewer sampling points to approximate the original interconnect.

How to choose the number and the locations of middle layers from the original etching files with acceptable accuracy is a hard problem. It is not affordable to simulate all combinations (i.e, for 35 middle layers, 3D capacitance solver need to be performed 324632 times to find a Five-layer approximation shape with smallest error). On another hand, simply choosing layers with fixed interval (i.e. choose a layer every 5 layer) may be too simple to handle some complex side wall shapes. Here, we propose a simple dynamic programming algorithm to choose the number and the location of layers based on error criteria.

Since there are less middle layers in the new approximation shape, every point $A(x, y, z)$ in the original profile has a projected point $A'(x', y', z)$ on the line $C'D'$ of new side wall surface with the same height, where C' is the closest points in the new shape to A with height higher than A, and D' is the closed points in the new

shape to A with height lower than A. Note that A' and A could be the same if A is on the selected layer, i.e., $C' = C$ and $D' = D$. A 2D example (cross section of one side wall) is shown in Fig. 15. In our algorithm, 3D cases are considered. In this section, we use the distance of the A to A', $d(A, A')$ as the error function to measure how close of our approximation shape to the original shape. Note that, our method is not limited to this error function. Other error functions could be used also, such as $1/d(A, A_r) - 1/d(A', A_r)$, where A_r is a reference point and this function correlates to free space Green's function. Actually, our algorithm is flexible enough to use any error function correlates to the physical location of points.

Suppose the number of layers in the original etching profile is n, and the number of points on each layer is k.

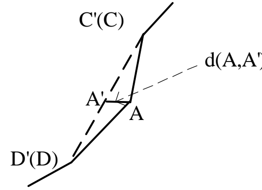


Fig. 15. Original points and projected points in a cross section view of one side wall surface.

Side Wall Layer Selection Problem : Given n layers and k points as described above, user specified error bound ϵ (or user specified number of layers), select m layers, where $m \geq 2$ and must include the first (top) and nth (bottom) layers, such that $\sum_{A \in S} d(A, A') < \epsilon$ (or minimize $\sum_{A \in S} d(A, A')$ for user specified m), where S is the set of all points in the original layer, and A' is the projected point of A on the side wall surfaces formed by new m layers with the same height.

Side Wall Layer Selection Algorithm: The algorithm is based on dynamic programming. First, we precomputed a $n \times n$ table ER which stores the information of error functions for every two layers. $ER[i, j]$ for $i < j$ is the sum of error function

for all points on layers from $i + 1$ to $j - 1$ if layer i and j are directly connected. $ER[j, i] = ER[i, j]$ and $ER[i, i] = 0$. It is easy to see that $ER[i, i+1] = 0$. Note that we only need to compute and store half of the table ER.

Let us define a two-dimensional table T, where each entry $t[i, j]$ is the minimum sum of error functions of points on layers from 1 to j when i middle layers are selected and the last middle layer is located at layer j (which implies the layer structure is 1, ..., j, n). Note that i is from 0 to $n - 2$, and j is from 1 to $n - 1$. $t[i, j]$ is computed as the following recursive formula

$$t[i,j] = \begin{cases} 0, \text{if } i = 0 \text{ and } j = 1 \\ \infty, \text{if } i \geq j \text{ or } (i = 0 \text{ and } j \neq 1) \\ min_{k=1,\ldots,j-1}(t[i-1,k] + ER[k,j]), \\ \quad \text{if } i > 0 \text{ and } i < j \end{cases} \tag{3.1}$$

Here is the intuitive explanation of how $t[i, j]$ is computed. When we want to choose i layers where the last layer is j, we look at all combinations of $i - 1$ layers plus a new link between the last layer of $i - 1$ layers and the layer j, and choose the one with minimum error provided that the selection of $i - 1$ layer is optimal. $t[0, 1] = 0$ means to choose zero layers, where the last layer is the first layer. Since it is not possible to choose i middle layers where the last layer index is less or equal to i, all $t[i, j]$ for $i \geq j$ is infinity. Similar argument for $t[0, j]$ when $j \neq 1$. It is easy to see from Eq. (3.1) that $c[i, i + 1] = 0$ and $c[1, j] = ER[1, j]$ when $j \neq 1$.

```
Side Wall Layer Selection
    Input: n * k sampling points on n layers, user specified ε or m
    Output: m * k sampling points on m layers
 1  Precompute ER table ;
 2  t[0,1] = 0 ;
 3  for  j = 2 ... n − 1 do
 4      t[1, j] = ER[1, j] ;
 5  end
 6  for i = 2 ... n − 2 do
 7      for j = i + 1 ... n − 1 do
 8          t[i, j] = ER(i, j);
 9          for k = i + 1 ... j − 1 do
10              t[i, j] = min(t[i, j], t[i − 1, k] + ER[k, j];
11          end
12      end
13  end
14  ME[0] = ER[1, n];
15  for i = 1 ... n − 2 do
16      ME[i] = ER[i + 1, n] ;
17      for j = i + 2 ... n − 1 do
18          ME[i] = min(ME[i], t[i, j] + ER[j, n]) ;
19      end
20  end
21  return m and location of m layers based on ME and user specified ε or m ;
```

After all $t[i,j]$ entries are computed, the minimum error $ME[i]$ for i middle layers is

$$ME[i] = min_{j=1,\ldots,n-1} t[i,j] + ER[j,n], i = 0, \ldots, n-2.$$

If the error bound ϵ is given, we select i and corresponding layer assignments such that $ME[i] < \epsilon$ and i is minimum. If the number of layer m is found, we directly find the layer assignments for $ME[m]$. The pseudo code of the algorithm is shown Algorithm 1. The code has taken into consideration that some entries of $t[i,j]$ is infinity. The location of m layers can be found easily with simple back-trace and the code is omitted here due to space limit.

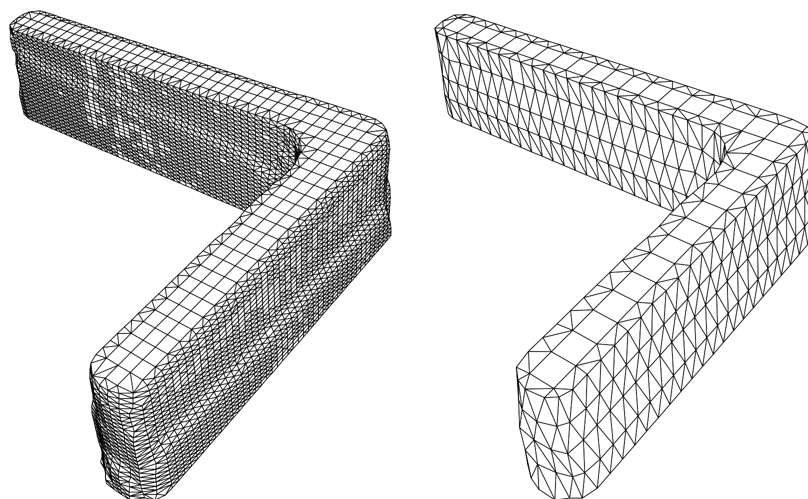
It is not to hard to see that by dynamic programming, the computation time of $t[i,j]$ and $ME[i]$ can be easily done in $O(n^3)$ time and $O(n^2)$ memory. It takes

$O(n^3k)$ time and $O(n^2)$ memory to compute ER table since for each $ER[i, j]$ entry it takes $O(nk)$ to compute the sum of error functions for all points between layer $i+1$ to $j+1$, and the projection points need to be recomputed each time for different $ER[i, j]$. Therefore, the total computation time is $O(n^3k)$ and memory consumption is $O(n^2)$. The optimality of the algorithm under our error function definition is guaranteed due to the optimal substructure of the problem all subsolutions are visited. The detail proof is omitted here due to space constraints.

If we draw a curve with y axis being $ME[i]$ and x axis being i, after removing redundant solutions (if $ME[i] < ME[j]$ and $j > i$, then $ME[j]$ is redundant), it is not surprising that the curve is a convex curve since more layers are selected, less error will be. From extensive experiment, we also found the $ME[i]$ has good correlation with the error of final capacitance matrix, which means our error function is valid. Again, it is possible to use other error functions to computer ER and get better results.

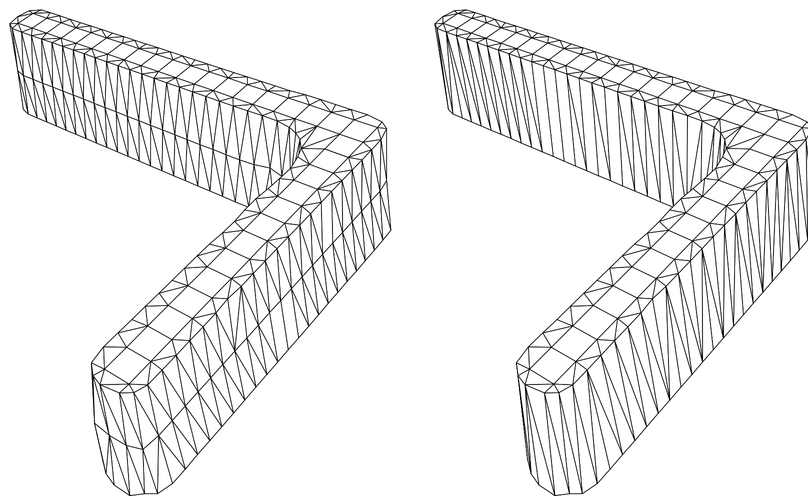
To verify our algorithm, Fig. 16 shows an elbow example with different m middle layers selecting from our algorithm. The total layer in the original file is 37, and the number of points chosen for every layer is 220. Our layer selection algorithm only takes 0.5 seconds on a SUN ULTRA SPARCV9 400 MHZ with 2GM memory machine. Note that one run of BEM solver for the original profile takes 20 seconds, therefore it is impractical to use 3D BEM solver directly to simulate all combinations of layer assignment.

The comparison results for single elbow are shown in Table VII. We found that the result with 5 layers has good accuracy compared to capacitance of the etched one. With the same method, some experimental results are obtained for more other structures, such as signal bus, parallel bus, 1x1 crossing bus, and 2x2 crossing bus. The running time of our algorithm for all cases (each with 200 to 400 points per



(a) Original shape with 37 layers

(b) 5-layer approximation



(c) 3-layer approximation

(d) 2-layer approximation

Fig. 16. One example approximation for elbow-shape conductor.

layer and 37 layers) are less than 1.2 seconds. The errors compared to etched profile are also shown in Table VII. We found that when the number of the layers is more than 4, both $ME[i]$ curve and capacitance error curves go to flat, which means the approximate capacitance has enough accuracy while the running time and memory have been dramatically reduced.

Table VII. Capacitances error with the different number of etching layers.

	Etched Profile 37 layer	5 layer Error (%)	3 layer Error(%)	2 layer Error(%)
Single elbow	-	1.45	1.73	2.30
Single bus	-	0.32	0.72	1.39
Parallel bus	-	1.31	4.80	7.92
1 X 1 bus	-	0.55	1.25	1.94
2 X 2 bus	-	1.14	3.68	5.74
Total time (s)	1404.4	148.4	91.2	72.5
Total memory(MB)	932.3	121.8	75.43	53.84

3. Shape Correction

Even though we can only select few layers from the lithography simulation profile, the conductor shape is still irregular and it can not be handled by current LPE commercial tool. Shape correction algorithms are introduced here.

```
3D Interconnect Shape Approximation
   Input: Lithographic Simulation Result, Circuit Layout
   Output: Shape Correction Result
1  ReadInput Circuit Layout Data and keep the edge of the circuit into vector
   E ;
2  ReadInput Lithographic Simulation Data and keep them into vector L ;
3  for k = 1 ··· n do
4      for i = 1 ··· p do
5          Flag = 2 ;
6          d_min = ∞ ;
7          for j = 1 ··· q do
8              A = L[k].pt; B = pt → next ;
9              if d(AB, E(i)) < d_min then
10                 d_min = d(AB, E(i)) ;
11                 Record the coordinate of point A and B ;
12             end
13         end
14         if edge // x axis then
15             Keep x coordinate of point A and B into vector Coord ;
16             if Flag ==2 then
17                 Flag = 0 ;
18             end
19         else
20             Keep y coordinate of point A and B into vector Coord ;
21             if Flag ==2 then
22                 Flag =1 ;
23             end
24         end
25     end
26     Generate Coordinates for every layer of Conductor ;
27 end
28 Fit the boundary wall using linear curving fitting;
```

The main idea of the shape correction is as follows:

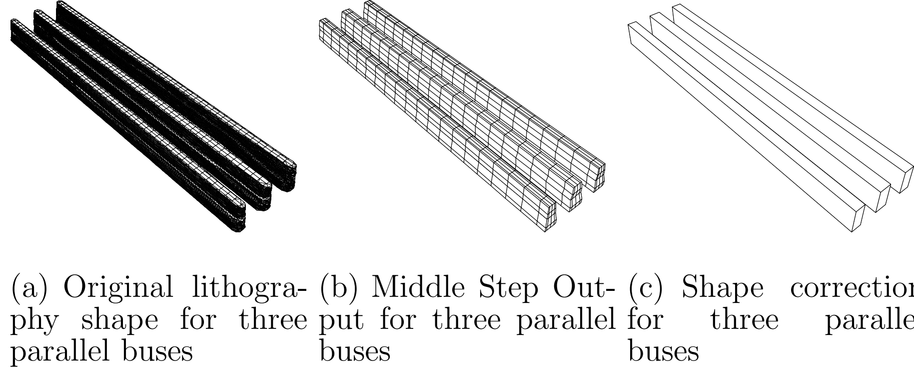
1. Read the original layout profile without process variation and lithography simulation. Keep layout profile in vector $E[p]$, where p is the number of total boundary edges of the layout profile. For the most selected interconnect pattern, p is a small number, i.e., p is 4 for the standard bus and p is 6 for a elbow shape. Meanwhile, keep lithography simulation results in vector $L[n]$, where n is the number of total layers. Based on the previous analysis, we set n to be 5.

2. For every edge in $E[p]$, find the point in lithography profile with minimum distance to it and keep its coordinates into vector Pos. The dimension of vector Pos is the same as that of vector $E[p]$.

3. Fit the boundary wall based on the previous results using least square method (LSM).

The pseudo code for shape correction is given in Algorithm 2. In the algorithm, $L[k].pt$ is the starting point of kth layer, $Flag$ indicates which one is firstly kept into vector Pos, x or y. The complexity of the algorithm is $O(pP)$, where P is the number of the total sampling points on all layers, and p is the number of the layout profile edges. Note that for most structures, p can be regarded as a constant.

Our shape correction algorithm significantly reduces the size of the input for FastCap. For example, the output shape from lithography simulation is shown in Fig. 17(a), and the output before fitting the boundary wall with LSM is shown in Fig. 17(b) which is much simpler than before. However, the side wall shape is still not regular, such as rectangles or quadrilaterals. After LSM, we fit the boundary points based on the previous step results into known patterns of the standard input format and get the final results shown in Fig. 17(c). Table VIII shows the capacitance values computed by FastCap, as well as the running time and memory usage, for the shape generated by the lithography simulation and shape correction algorithm. The shape correction result shows good accuracy from Table VIII. After shape correction, FastCap takes much less time and memory to compute the capacitance. The shape correction algorithm is executed on the laptop with Intel Pentium M 1.60 GHz and 512 MB RAM. The running time of the shape correction algorithm is only 5 seconds.



(a) Original lithography shape for three parallel buses (b) Middle Step Output for three parallel buses (c) Shape correction for three parallel buses

Fig. 17. One examples for three parallel buses.

Table VIII. Capacitances with new LPE methodology for three parallel buses.

Capacitance(aF)	Etched	Shape Correction	Error(%)
C_{11}	100.5	101.3	0.79
C_{22}	139.3	140.6	0.93
C_{33}	100.6	101.1	0.50
C_{12}, C_{21}	61.08	61.78	1.15
C_{13}, C_{31}	10.99	11.04	0.45
C_{23}, C_{32}	61.35	61.77	0.68
Time(s)	339.5	2.9	
Memory(MB)	285.6	5.5	

4. Resistance Extraction

In Fig. 12(b), the shape of the lithography simulated interconnect is irregular. In order to accurately compute the resistance of such irregular geometry, we can not directly use the classic equation $R = \rho L/A$, where ρ is resistivity, L is the length of the conductor and A is the area of the cross section.

We discretize the conductor into 3D grids, and build a linear system $GV = I$ [23] based on Kirchoff's Law, where G_i is the element conductor, $V_{i,j,k}$ is the voltage at node i, j, k and I is the independent current source. One model is shown in Fig 18. The linear system is shown as below,

$$
\begin{aligned}
I &= (G_1 + G_2 + G_3 + G_4 + G_5 + G_6)V_{i,j,k} \\
&\quad - G_1 V_{i,j-1,k} - G_2 V_{i,j+1,k} - G_3 V_{i-1,j,k} \\
&\quad - G_4 V_{i+1,j,k} - G_5 V_{i,j,k+1} - G_6 V_{i,j,k-1}.
\end{aligned}
$$

In the equation, $G_i = 1/R_i$. For a regular shape input, all G_i's are the same. But for irregular shape input, each G_i could be different. We solve the linear system $GV = I$ to get the node voltage at every grid point. Finally, we can get the average voltage drop along the conductor and use $R = U/I$ to obtain the resistance.

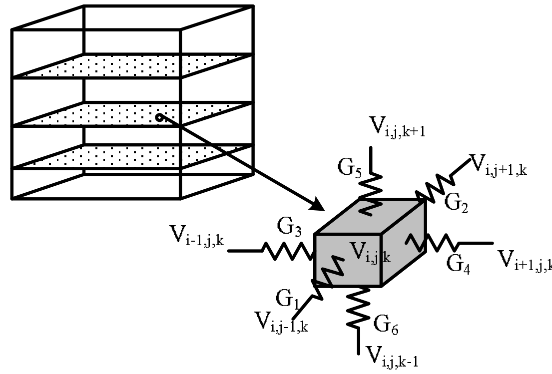


Fig. 18. Resistance computation model.

Table IX gives resistance extraction results corresponding to the example shown in Fig. 9. The resistance value based on the original layout profile could have 20% error compared to etched conductor. Meanwhile, we can observe that our shape correction algorithms are still efficient for resistance extraction, where the error is less than 5%. After shape correction, the conductor profile is regular now. We can use classical equation to obtain the resistance value.

Table IX. Resistance comparison between new and old LPE methodologies for 1x1 elbow example.

Resistance (Ω)	Etched Conductor	Layout Profile	Error (%)	Shape Correction	Error (%)
R_1	0.29	0.23	20.69	0.30	3.45
R_2	0.23	0.18	21.74	0.24	4.35

5. Inductance Extraction

Table. X gives inductance extraction results corresponding to the example shown in Fig. 9. The inductance values of layout profile and shape correction profile are obtained by FastHenry [24, 25] . Based on the results in Table. X, the lithography effect on inductance is insignificant under the current technology. Therefore, we can directly use the inductance of the layout profile.

Table X. Inductance with new LPE strategy for 1x1 elbow example.

Inductance (pH)	Layout Profile	Shape Correction	Error (%)
L_{11}	3.0011	2.9787	0.75
L_{22}	2.2028	2.1834	0.89
L_{12}, L_{21}	1.1732	1.1268	3.95

E. Conclusion

In this chapter, a new LPE methodology is proposed considering lithographic effect. Meanwhile, the algorithms for capacitance and resistance extraction are also presented, respectively. Lithography simulation and shape correction steps including a smart dynamic programming based layer selection scheme are inserted into the

traditional LPE methodology to form new LPE methodology. Compared with the traditional methodology, the new methodology will get much more accurate results. The new algorithm significantly reduces the running time of the 3D capacitance solver while keep the good accuracy.

CHAPTER IV

THE IMPACT OF BEOL LITHOGRAPHY EFFECTS ON THE SRAM CELL PERFORMANCE AND YIELD

A. Background

Manufacturing variations can be classified as systematic and random variations. Systematic variations are predictable in nature and depending on deterministic factors such as layout structure and surrounding topological environment. On the other hand, random variations are unpredictable and are caused by random uncertainties in the fabrication process such as microscopic fluctuations in the number and location of dopant atoms in the channel region. Random variations are harder to characterize and can have a detrimental effect on the yield of critical modules in a design.

For advanced technologies, the feature size is much smaller than wavelength, i.e. 65 nm node is exposed with 193 nm light. It is more difficult to print the desired layout shape on the wafer even with complex Resolution enhancement techniques (RET) [26] to maintain adequate pattern fidelity. RET developed for nominal lithography conditions (at tremendous computational cost) results in complex systematic variability in device and interconnect structures. On the other hand, RET techniques are not particularly robust across process windows and are amplifying other sources of lithographic variability, including defocusing, exposure dose, misalignment, len aberrations and resist and etch processing.

In addition to the above systematic variations, random variation such as misalignment is also an important BEOL factor. Under certain misalignment conditions, parasitic resistance becomes extremely big and SRAM performance will be seriously affected.

Some previous literatures described the impact of process variation on SRAM yield [27, 28]. However, no analysis has been done considering the internal cell interconnect parasitic BEOL variations, including both lithographic variation and misalignment, and front-end-of-line (FEOL) variations (i.e., variation on supply and threshold voltage) at the same time. In this section, we propose a methodology to analyze the combined impact of BEOL and FEOL process variations, and quantify the arising performance uncertainty. To better quantify such combined effect, we study the sensitivity of cell performance yield to these variation variables. We also present detailed analysis on the impact of the internal cell parasitic RC network parameters on SRAM stability, especially for the minimum cell operating voltage for a given yield tolerance level.

In this chapter, we use Calibre [1] as the lithographic simulation tool. To analyze the impact of different process variations such as dose/defocus variation, we use three different process settings, denoted as "Best", "Nominal" and "Worst", and generate three lithographic contours corresponding to each setting. Fig. 19 shows an example of the three different contours along with the corresponding original layout. Note that our methodology is not limited to a particular lithographic simulation tool, and can be extended to study more than three process corners.

B. SRAM RC Model

1. SRAM RC Model

General MXN SRAM array structure is shown in Fig. 20. In this section, the benchmark design is a 6-Tr (transistor) SRAM cell design shown in Fig. 21. All devices, interconnect and lithographic simulation parameters are based on 45 nm technology.

[1] Calibre is a trademark of Mentor Graphics

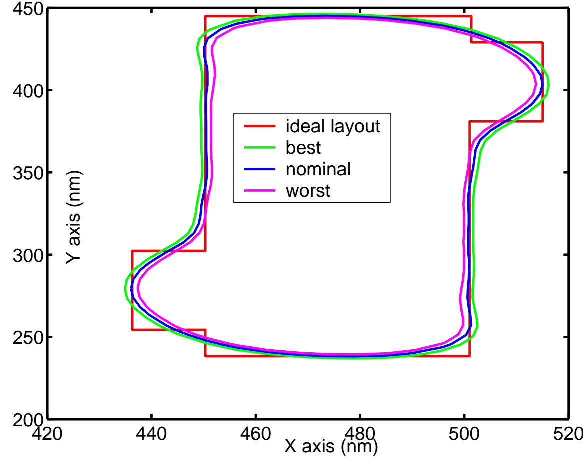


Fig. 19. Different lithographic profiles from the same layout profile of SRAM with different depth of focus (DOF).

In order to analyze the internal cell interconnect impact on SRAM peformance and yield, we propose a new SRAM schematic simulation model which includes SRAM internal cell parasitics, as illustrated in Fig. 22. Compared with Fig. 21, the internal cell interconnect RC network is the new addition. Lithographic simulation are performed under 3 different process windows, “Best”, “Nominal” and “Worst”, and three layout contours are generated. Each process window corresponds to a set of focus, dose and mask error settings. In the rest of the chapter, let “Basic NoRC” be the SRAM model without internal cell interconnect parasitic, as shown in Fig. 21. Let “Ideal RC” be the SRAM model including the internal cell interconnect parasitic network (Fig. 22) with all the parasitic parameters derived from the drawn profile (ideal layout in Fig. 19). Let “Best/Nominal/Worst RC” be the SRAM models including internal cell interconnect parasitic network with parameters deriving from corresponding interconnect lithography contours.

In this section, we rely on a simplified cross-subsection to model the memory

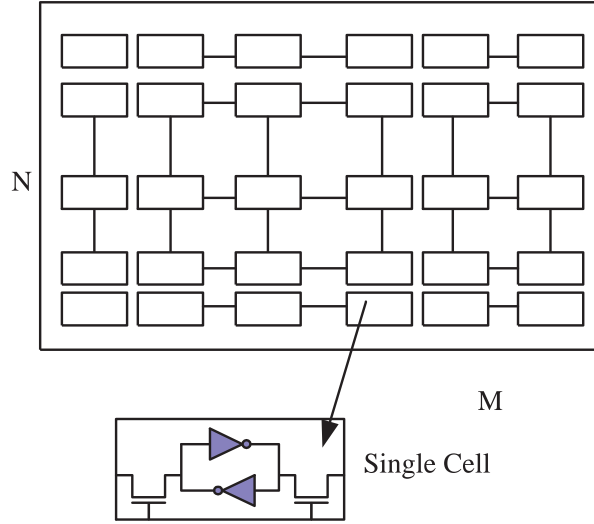


Fig. 20. SRAM MXN array.

block (array). The cross-subsection includes the local bitline accessed cell and load cells. Similar to state of the art designs [29], the number of cells per local bitline is 16. In a typical analysis the schematic of (Fig. 21) is used to represent the cell and load cells without the internal cell interconnect parasitics. For the BEOL analysis, we still use the 16 cell architecture cross-subsection to analyze SRAM performance and yield. However, we use our SRAM RC model (Fig. 22) to substitute NoRC model (Fig. 21). Next subsection, we will discuss how to get RC value for the internal cell parasitics.

2. RC Extraction for SRAM Cell

For extraction, we adopt the flow in [30] to handle non-regular contour shapes from lithography simulation. The basic idea is to use shape approximation method to get the approximated regular contours from non-regular contours. RC extraction is then

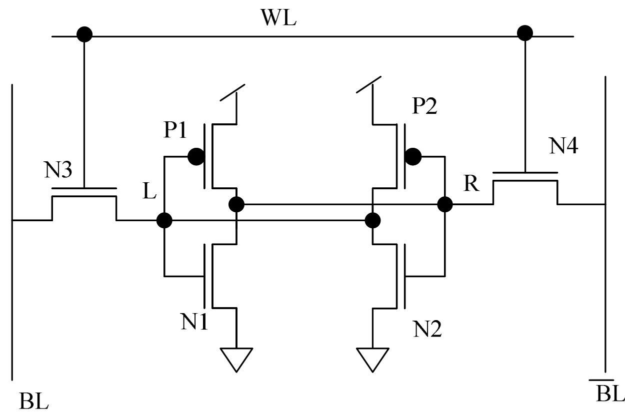


Fig. 21. 6 transistor SRAM schematic.

performed with Raphael [2] 3D field solver on the approximated contours, and multiple dielectrics are considered during the extraction.

For resistance extraction, since SRAM structure has repetitive patterns (repetitive cell by cell), we decompose $M \times N$ SRAM array into $M \times N$ cells according to the cell boundary. Resistance is extracted for the center cell.

For capacitance extraction, it is more complicated since the capacitance of each net also depends on its neighboring structure. When we use enough cells to do extraction, the center cell parasitics almost become constant. After getting the center cell RC netlist, we apply them into 16 cells and perform SPICE simulation to study the effect of internal cell parasitic on SRAM performance with 16 cells/bitline architecture.

[2]Raphael is a trademark of Synopsys

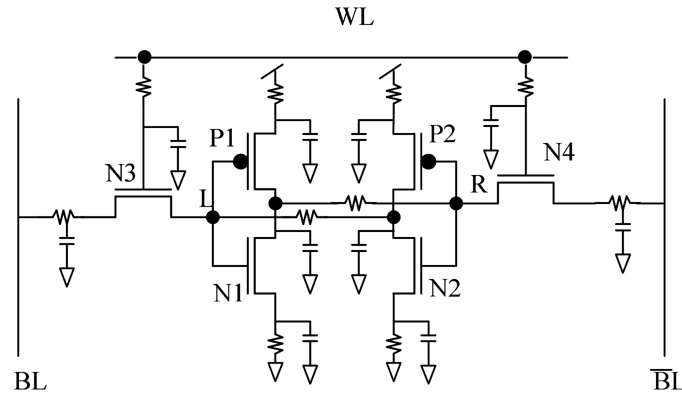


Fig. 22. 6 transistor SRAM schametic with RC network.

C. The Methodology for SRAM Performance Analysis

Section 2 discusses the process of RC extraction on a given layout contour of SRAM array. The whole methodology for SRAM performance analysis can be summarized as follows, and the flow diagram is illustrated in Fig. 23.

1. Pre-process the layout and recognize SRAM structure cell by cell;
2. Generate the three litho contours at different process corners;
3. Use polygon intersubsection algorithm to decompose the whole SRAM array into many small analysis units;
4. Use the pre-proposed method in [30] to get the approximated contour shape to speed up the RC extraction and extract RC value with Raphael;
5. Generate the whole SPICE netlist with BISM4_45 predictive model [31] to analyze SRAM performance.

With the above methodology, now it is viable to analyze the impact of lithography, misalignment, and all other BEOL parameters on SRAM performance.

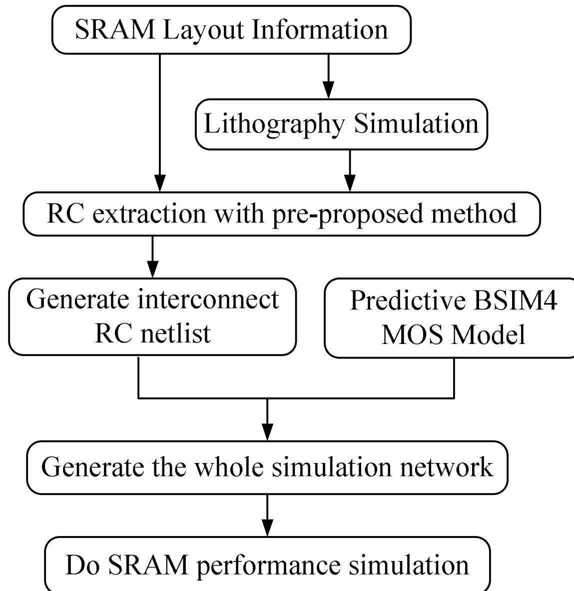


Fig. 23. The methodology flow for SRAM performance and yield analysis for BEOL variations.

D. BEOL Impact Analysis with New Methodology

In this section, we will perform detailed analysis for the impact of BEOL variations on SRAM performance. In general, SRAM yield and performance are highly dependent on the headroom between the supply voltage and the threshold of devices. In order to better quantify the BEOL impact on SRAM performance, we study the sensitivity of the cell functionality yield to these BEOL variations. We then model the yield sensitivity to predict ranges of tolerable fluctuations as function of desired V_{min} (minimum cell operating voltage for given tolerance level). We also studied the FEOL design at the same time for purposes of comparison. We swept the supply voltage V_{dd} and threshold voltage V_{th} in our experiments. Supply voltage ranges from 0.6 V to 0.9 V and threshold voltage variation is from 0 to 6σ. This leads to different worst case operating conditions of the SRAM cell. Hence we can analyze the cell performance

and stability under different FEOL conditions [32, 33, 34]. The additional BEOL are then evaluated under those different cell conditions.

In the following sections, we will analyze the performance in terms of read delay metric and transient node upset (often referred to as dynamic noise margin) of the cell [35]. We focus on studying the read delay variation due to BEOL since the read delay is more representative than write delay considering its magnitude.

1. BEOL Impact on SRAM Read Delay

As illustrated in Fig. 19, the lithographic profiles of SRAM cell are different from the drawn layout contour. Such difference can be translated to the change of RC value in the parasitic network of the cell. For the SRAM cell under study, the results of RC are shown in Table XI, where R_i and C_i are the RC values of the ideal layout profile, which can not be released here.

Table XI. The relative RC value among all layouts in one piece of SRAM cell.

	Ideal	Best	Nominal	Worst
R	R_i	$0.85R_i$	$0.90R_i$	$1.13R_i$
C	C_i	$1.14C_i$	$1.04C_i$	$0.95C_i$

Note that RC value under the "Nominal" case is still different from "Ideal" due to litho effect. With the RC netlist for ideal layout and lithographic contours extracted using our methodology, SPICE simulation are then used to obtain the SRAM read time delay.

Comparison results of read delay τ_R are shown in Fig. 24 and Fig. 25. For each RC model from Ideal, Best, Nominal and Worst shape, the delay variation compared to the Basic NoRC model is shown in Fig. 24. First, we can see that the modeling interconnect parasitics could introduce 20-34% delay difference. This highlights the

importance of interconnect modeling. Also, from Fig. 25, it is clear that lithographic variation itself could also cause up to 4% read delay variation compared to the ideal RC model.

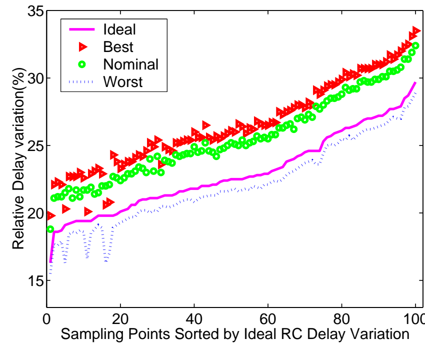


Fig. 24. Relative delay variation of ideal, best, nominal and worst RC model vs. basic NoRC model.

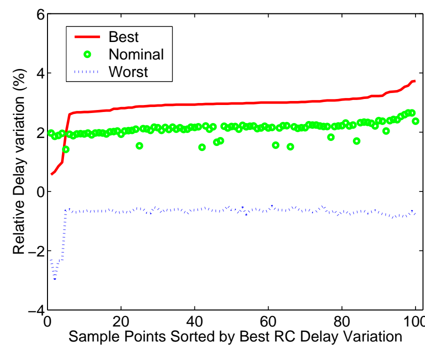


Fig. 25. Relative delay varitation of best, nominal and worst. The reference model is ideal RC model.

Next, we will discuss the BEOL impact on the read yield. For proper yield, the cell must satisfy a given probability of fail criteria. Without loss of generality, we define pass fail criteria for the read yield Y_{read} in the following way: if bitlines drops to half rail, then we call it "pass", and otherwise "fail". We swept V_{dd} and threshold

voltage for those three litho RC models, ideal RC model and Basic NoRC model. Fig. 26 illustrates the read yield of the SRAM cell for different parasitic RC models, where the black dotline represents the acceptable yield criteria (5 σ here).

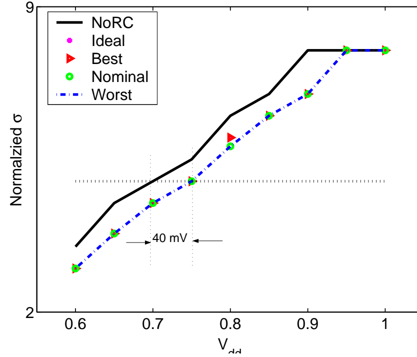


Fig. 26. Read yield of noRC, ideal RC, best RC, nominal RC and worst RC model.

It is obvious that the read yield becomes smaller after considering internal cell interconnect parasitics. The minimum cell operating voltage V_{min} (read V_{min}) increases by 40 mV due to internal cell interconnect parasitics and the power estimation may be off 9%. However, the difference between Ideal, Best, Nominal and Worst RC model is very trivial. It shows that litho variation at the level of our experiment does not have a big impact on V_{min} analysis.

2. BEOL impact on Stability

Traditionally, SRAM cells are designed to ensure that the contents of the cell do not get altered during read access. This can be satisfied by balancing the relative strengths of the devices in the design. However, the cell may still be vulnerable to the failures caused by random variations in the device strengths.

For the SRAM cell structure shown in Fig. 4, it is also desired that the cell stability maintain cell status during read "0". During read "0", the access and pull down

transistors, N_i (i=1,2,3,4) act as a voltage divider circuit between the precharged bitline BL_L, and node L shown in Fig. 21 and Fig. 22. Usually, this induces noise at node L. However, it is possible that the cell be week due to process variation. In this case, the induced noise can be significant enough to flip the content of the cell. This is known as a destructive read. For a cell to be stable, the maximum noise on node "L" must satisfy the "acceptability" criteria stated in the following equation.

$$V_{max}(L) < kV_{dd},$$

where $k < 1/2$. Usually, we want to account for the cell flipping. However, we can be more conservative by setting $1/3 < k < 1/2$. Here we use the later one.

Fig. 27 shows the stability yield result for NoRC, Ideal, Best, Nominal and Worst case. The black dotline represents the acceptable yield criteria. The minimum cell operating voltage V_{min} for the given tolerance level increased by 100 mV from 0.64 V mainly due to bitline parasitics loads.

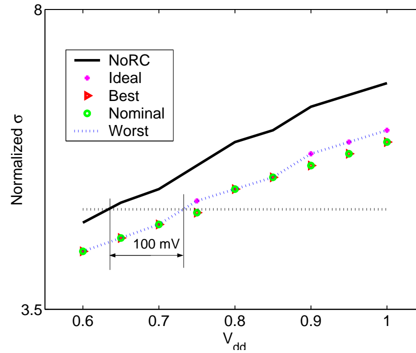


Fig. 27. Stability yield for noRC, ideal RC, best RC, nominal RC and worst RC model.

It is clear that it is important to model interconnect parasitics for stability yield analysis since V_{min} increased by 100 mV due to internal cell interconnect parasitics. Hence, Basic NoRC model analysis can be very optimistic and power estimation may

be off by 33%. Similar to read yield analysis, litho variations present very small sigma yield change.

3. Misalignment Impact on SRAM Performance

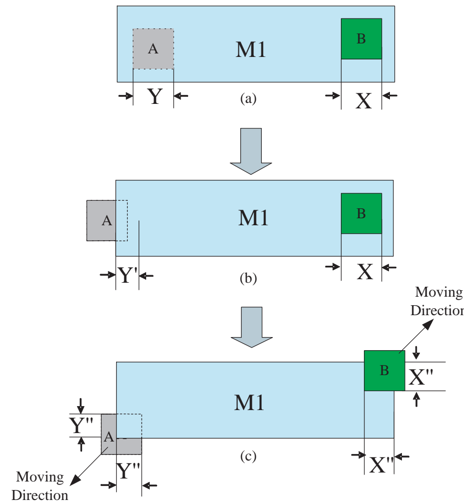


Fig. 28. Misalignment.

In the previous section, the systematic lithographic variations were considered. In this section, we will study another factor that could have an impact on SRAM performance: misalignment. It is possible during the fabrication steps that a layer gets misaligned compared to another layer.

Fig. 28 shows the possible misalignment that we consider in this section. The small green and gray squares represent the via and contact, respectively. The big rectangle is the metal interconnect. Fig. 28 (a) shows the original layout. In Fig. 28 (b), due to misalignment, the gray contact A is moved to the left and the green via B keeps the same position (contact A and via B are in different independent layers). The overlap distance Y becomes Y' which is smaller than Y and X remains

constant. As we know, the resistance has a close relation with electrode position. In this case, the resistance of metal and contact A are changed, while the resistance of via B remains constant. An example of both contact and via misalignment is shown in Fig. 28 (c). The contact A and via B are moved in the direction shown in Fig. 28 (c). Assume the current flow from via B to contact A. The size of contact A and via B is 1x1 $unit^2$, where $unit$ is the real edge length of contact A and via B. The contact unit and via unit may be different, but we use same value in this case. X'' and Y'' represent the normalized overlap distance between metal M1 and contact A, via B, respectively. The distance between contact A and via B becomes bigger when contact A and via B are moved in the opposite direction, respectively. Fig. 29 shows how the metal resistance changes when X'' and Y'' becomes smaller. We observe that R_{max}/R_{min} could be even bigger than 80 when X'' and Y'' is very small. R_{min} is the metal resistance shown in Fig. 28 (a).

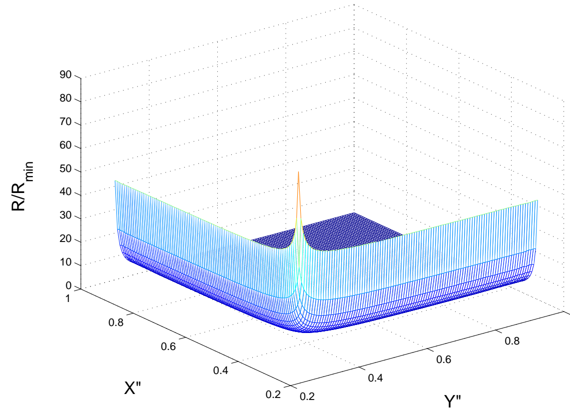


Fig. 29. Resistance vs. misalignment distance.

In our SRAM array study, there are 5 layers: CA, $M1$, $V1$, $M2$ and $V2$ (1 contact, 2 metal and 2 via layers); and every layer has the possiblity to be moved

in four direction such as left, right, up and down. In order to simplify our analysis, we only consider one case to study the misalignment impact on read delay τ_R. Let us define "shifted unit" be 10% of the contact critical dimension. In our study case, layer $V1$ is moved by 3 shifted units, 6 shifted units and 9 shifted units in right direction, respectively. As shown in Fig. 30, the read delay variation is increased as the misalignment becomes worse. The read delay could degrade up to 4%.

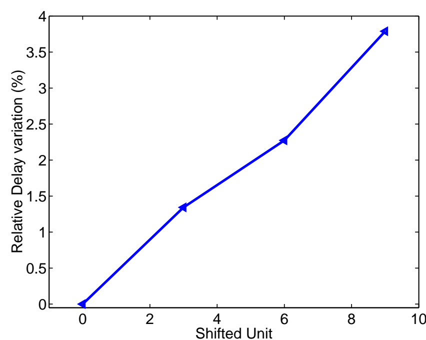


Fig. 30. The effect of misalignment on the read delay τ_R variation. The reference model is idea RC model.

Hence, in practice due to design rule requirements, the impact of misalignment is not very strong when the misalignment shift in the layers is less than the contact critical dimension because the resistance change is small. In extreme cases, the read delay variation can reach 32% compared to that of ideal RC model when R_{max} is around 3 $K\Omega$. In fact, misalignments only start to be critical when R value increases to be the same order as the nonlinear device resistance.

Next we study the performance yield variation (in σ values) as a function of resistance R_{max} which represents the cell maximum resistance. The yield results are studied for different supply voltage, V_{dd}, values. Table XII shows the corresponding read yield Y_{read} when we consider the misalignment conditions understudy. In order

to analyze the effect on the yield of more severe misalignment scenarios, R_{max} range is varied from 0.1 $K\Omega$ to 5 $K\Omega$. Significant yield drop is noticeable when R_{max} approaches the $K\Omega$ range as shown in Table XII.

Table XII. SRAM yield analysis. Yield is in sigma.

R_{max} (kΩ)	~0.1	~0.5	~1	~5
V_{dd} =0.9 V	~6.9	~6.8	~6.7	~6
V_{dd} =0.8 V	~5.7	~5.6	~5.5	~4.9
V_{dd} =0.7 V	~4.4	~4.4	~4.3	~3.7

E. Conclusion

In this chapter, we propose a new SRAM parasitic analysis model to capture the internal cell interconnect parasitics RC network. Then we propose an SRAM performance/yield analysis flow which enables litho-aware parasitic extractions and simulation to the existing flows. With our proposed methodology, we can study the back-end-of-line(BEOL) variations on SRAM performance combined with front-end-of-line(FEOL) variations. We show that the read delay can vary by 34% compared to traditional models which lack the internal cell interconnect parasitic modeling. Lithographic variations can introduce 4% read delay variation. V_{min} for both read and stability yield show significant change due to interconnect parasitics and power estimation may be off by 33% in our study case. In addition, the misalignment impact becomes dominant when the internal cell resistance change due to misalignment has the same magnitude of the nonlinear device resistances.

CHAPTER V

A SLEW BASED C_{EFF}

A. Background

Static timing analysis [36] has been successfully applied on integrated circuit sign-off since it has the capacity and speed to handle modern large multi-million-gate designs. The accuracy of STA is determined by the accuracy of the waveform and timing models used for logic cells and interconnect wires, and a large amount of research has focused on assessing and improving this accuracy.

In general, the following assumptions and abstractions are required for most STA methodologies considering the time-accuracy tradeoff:

1. Waveforms are modeled as saturated ramps and parameterized by (a) the start time, (b) the transition time (slew rate), and (c) a polarity indicating whether the waveform is rising or falling.

2. Logic cells are modeled as transformation on waveform function parameters. Such transformation is also dependent on the environment in which the cell is operating and can include quantities such as power supply, temperature and loading. Thus a cell timing model has the form:

$$\mathcal{P}_{out} = f(\mathcal{P}_{in}, E),$$

   where $\mathcal{P}$ represents the waveform parameters, and E represents the environment. An example of such a model is the so-called K-Factor equations [37], where E includes the load represented as a single lumped capacitance.

3. The interconnect is modeled via (a) the loading it presents to the driving cells,

and (b) the delay and slew degradation it introduces.

In order to insure the accuracy of STA, it is important to correctly model the interaction between cell timing models and interconnect loading models. Cell timing models are generated by performing detailed circuit-level simulation of the cell under various input and loading conditions using tools such as Spice [38]. For simplicity and generality, the cell loading is modeled as a lumped capacitor. The single lumped capacitance will be used to represent both the RC interconnect and the non-linear input capacitance of fanout cells being driven through the interconnect.

With technology scaling, the ratio between the typical output resistance of the output stage of a cell and the interconnect resistance has been steadily rising, making the estimation of the single lumped-capacitance representation of interconnect load more complex. This was observed in [9] and an approach for computing an equivalent *effective capacitance* was proposed. Most previous approaches to compute effective capacitance, either iterative [9, 39, 40] or non-iterative [41, 42, 43], use one single effective capacitance to capture the delay information only. However, one effective capacitance that captures the cell delay cannot accurately predict the slew at the cell output. It is shown in [44] that the slew error could be as high as 50% when the delay based effective capacitance is used. The slew rate, of course, is crucial to the computation of the interconnect delay and slew [45], noise, and the output waveform of downstream cell. Therefore it is important to model the slew correctly for accurate STA analysis.

The approach in [46] proposes a statistical multiramp driver model for distributed RLC network load and uses two effective capacitances to model different slew rates of output waveform, which is due to lossy transmission line effects. With high accuracy, however, the method needs to perform complicated statistical precharacterization and

moments computation. In [44], an iterative approach based on precharacterized table look-up is proposed to compute the effective capacitance to match the output slew. The method may not be suitable for fast STA on multi-million gate designs, especially when statistical STA are performed, where table precharacterization for environment and process variations are much expensive.

In this chapter we present a new accurate and efficient iterativeless approach to estimate the effective capacitance for the output slew of the cell based on a compact analytical model of MOS device operation. The slew in this chapter is defined as 2× 10/50 (90/50) slew for the falling (rising) input, i.e., two times the time difference between when the waveform crosses the 50% point and the 10% (90%) point for the falling (rising) waveform. Our approach can work with other slew metrics definition, i.e., 10/90 or 20/80 with little modification, but we choose this metric with the following observations and realistic concerns:

1. For the short interconnect (smaller interconnect resistance), this slew metric shows good correlation to the popular 10/90 metrics.

2. For the long interconnect (bigger interconnect resistance), this slew matric usually gives worst slew compared to other slew metrics, which provides larger safety margin for the following optimization, such as gate sizing and buffering.

3. The 10/50 (90/50) slew is harder to capture since generally 10% (90%) point for the falling (rising) waveform is in the strongly non-linear region for a Pi section load. With little modification, other slew metric defined in the weakly no-linear region, i.e., 20/80, can be easily modeled.

4. We use the closed-form effective capacitance from [43] to compute the 50% point of the waveform. Therefore, we only need to model the effective capacitance to

compute the 10% point (90% point for the rising waveform).

The modeling is done with two simple closed form formulas, which are easy to embedded in any STA tools. First we develop a simple formula for the equivalent capacitance of a Pi section driven by a constant resistance, then another formula for the equivalent constant output resistance of a CMOS inverter. Both formulas targets 10% point (90%) for the falling (rising) waveform. We then show the accuracy of the combination of the two formulas on example circuits.

B. C_{eff} for an RC Network for 10% Point

We begin to consider that the situation when the driver is modeled by a linear constant resistor, and compare the waveforms at driver's output under two models for the loading conditions:

1. A lump capacitance, which will eventually be the effective capacitance. This is illustrated in Fig. 31 (a).

2. A second order driving-point impedance circuit consisting of a single Pi section [47, 48]. This is illustrated in Fig. 31(b).

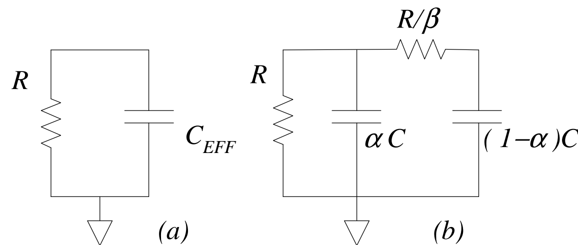


Fig. 31. One and two-stage RC circuit.

The solution for the simple one-stage RC circuit in Fig. 31(a) is

$$v = e^{-t/RC_{eff}} \tag{5.1}$$

The solution for the two stage RC circuit in Fig. 31(b) is

$$v = e^{-t/\tau_1}[-\frac{1-\alpha-\beta}{\xi}sinh(\frac{t}{\tau_2})+cosh(\frac{t}{\tau_2})] \tag{5.2}$$

where

$$\begin{aligned}\tau_1 &= \frac{2\alpha(1-\alpha)}{1-\alpha+\beta}\tau \\ \tau_2 &= \frac{2\alpha(1-\alpha)}{\xi}\tau \\ \tau &= RC \\ \xi &= \sqrt{(1-\alpha+\beta)^2-4\alpha\beta(1-\alpha)}\end{aligned}$$

Note that, $(1-\alpha+\beta)^2-4\alpha\beta(1-\alpha)$ is always greater than zero as long as β is greater than 0, which is proved in [43].

To match the delay of two models, two solutions needs to be matched at $0.5V_{DD}$, recognizing that the simple RC circuit will have an equivalent load capacitance denoted by $C_{eff} = \eta C$. From [43],

$$\eta_{0.5} = \frac{3\alpha+\beta^2}{3+\beta^2} = 1-3\frac{1-\alpha}{3+\beta^2} \tag{5.3}$$

To match the slew, we need to model the effective capacitance to match $0.1V_{DD}$. Note that another way is to directly match the slew of effective capacitance model to the Pi model. However, from extensive SPICE experiments, we empirically found that matching the slew directly gives worse results compared to matching $0.1V_{DD}$ and $0.5V_{DD}$ separately. The possible reason behind this is that for any slew metric, where one point in the strong nonlinear region of waveform and one point in the weak

nonlinear region of waveform are taken, one single equivalent resistance of nonlinear driver could not model both regions with the same accuracy.

In order to get the effective capacitance at $0.1V_{dd}$, we solve Eq. (5.1) and substitute in Eq.(5.2) which results in an equation in the quantities α, β and η that can be solved by any root finding routines, i.e., Newton-Raphelson:

$$0.1 = e^{-\phi_1}[-\frac{1-\alpha-\beta}{\xi} sinh(\phi_2) + cosh(\phi_2)] \tag{5.4}$$

where

$$\phi_1 = \frac{log(10)\eta_{0.1}(1-\alpha-\beta)}{2\alpha(1-\alpha)}$$
$$\phi_2 = \frac{log(10)\eta_{0.1}\xi}{2\alpha(1-\alpha)}$$

Fig. 32 shows the exact solution for $\eta_{0.1}$ of Eq. (5.4) for a wide range of αs and βs. For comparison, Fig. 33 shows the $\eta_{0.5}$ surface with the same set of αs and βs when $0.5V_{dd}$ is matched. Surprisingly, two figures are dramatically different. For example, $\eta_{0.1}$ decreases when α increases and β increases, while $\eta_{0.5}$ shows opposite behavior. Also, $\eta_{0.1} > 1$ while $\eta_{0.5} < 1$. Again, the reason is due to $0.1V_{dd}$ locating at the strong nonlinear region of Pi model. Figs. 34 and 35 show the output waveform of Pi model with different α and β parameters. First, we can observe that due to resistive shielding effect, the second order differential term is dominant. In such case, even the total lumped capacitance is used as effective capacitance, the time for the output of effective capacitance model reaches $0.1V_{DD}$ is still less than the time when the output of Pi model reaches $0.1V_{DD}$. When α or β increases, the second order effect is diminished and when α is near to 1 or β goes to infinity, the driving resistance sees the total capacitance as the load.

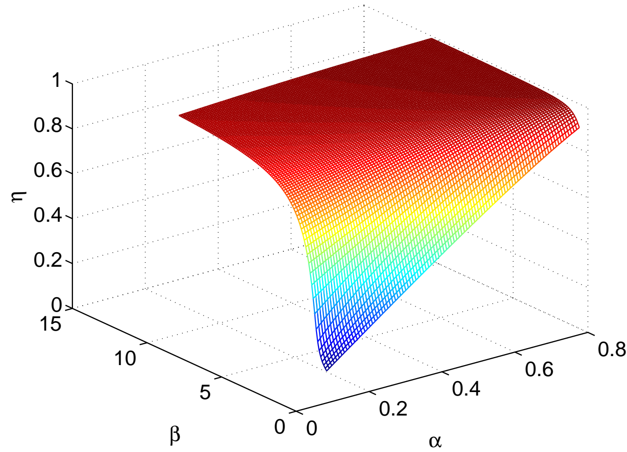


Fig. 32. η vs. α and β when we compute $0.1V_{DD}$ based effective capacitance.

From the physical property of the circuit, we know that Eq. (5.4) must have an answer in real number. It turns out that an excellent first order solution for $\eta_{0.1}$ of Eq. (5.4) is:

$$\eta_{0.1} = 1 + 0.5\frac{1-\alpha}{0.5+\beta} \tag{5.5}$$

A comparison between Eq. (5.4) and the exact solution of Eq. (5.5) is shown in Fig. 36 for value of α ranging from 0.1 to 0.8 and for values of β ranging from 0.4 to 10. The approximation is close enough to the exact solution, which is generally derived by a root finding routine. However, with the closed form Eq. (5.5), we can avoid the root finding process. Interestingly, we found that for points under 30% of V_{DD}, we can always use a formula similar to Eq. (5.5) to model the corresponding effective capacitance with tuning the 0.5 coefficients, while for the points over 50% of V_{DD}, we can uses a formula similar to Eq. (5.3) to model the corresponding effective capacitance.

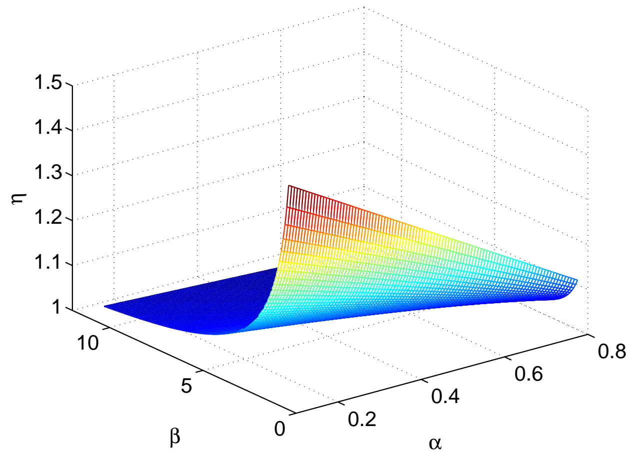


Fig. 33. η vs. α and β when we compute $0.5V_{DD}$ based effective capacitance.

It should be pointed out that when β is very small, usually less than 0.2, the driver will only see the near end capacitance, and $\eta_{0.1}$ do increases when α increases and $\eta_{0.1} \leq 1$. [1] In this section, we did not model the behavior of this region since: 1) in general such large interconnect resistance (the interconnect resistance is approximately two times of the resistance in the Pi model [47] for distribute RC load) will be either buffered or the net will be replaced in the first place in the synthesis level; 2) even in the rare case such region is necessary, we could use the model similar to Eq. (5.3) to capture the behavior.

Although Eq. (5.5) is convenient to apply, it requires knowledge of this equivalent output resistance of the driver R in Fig. 31. The next section will address this issue.

---

[1] When β approaches to zero, the resistance of Pi model goes to infinity and $\eta_{0.1} = \alpha$.

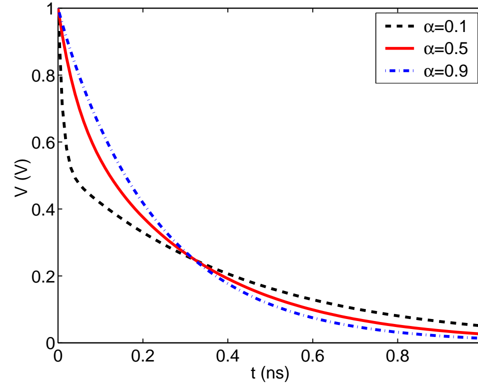


Fig. 34. The output waveform vs. α. $\beta = 1$.

C. Equivalent Output Resistance

The Shichman-Hodges (Spice Level-1) MOSFET model current equations are [49]:

$$I_{DS} = \begin{cases} 0 & : \quad V_{GS} \leq V_T \\ K(V_{GS} - V_T)^2 & : \quad V_{DS} > V_{GS} - V_T \\ \dfrac{K(V_{GS} - V_T)^2(2V_{DSAT} - V_{DS})V_{DS}}{V_{DSAT}^2} & : \quad V_{DS} < V_{DSAT} \end{cases} \tag{5.6}$$

where the constant K is drivability factor, $V_{DSAT} = (V_{GS} - V_T)$ is drain saturation voltage and V_T is threshold voltage. K can be expressed as $K = \dfrac{W}{L}\dfrac{K_p}{2}$ using standard Spice parameter names, where L is effective channel length, W is a channel width.

Now consider the CMOS inverter circuit in Fig. 37. We can write the time domain equation for the output voltage V_o as:

$$C_L \dot{V_o} = I_P - I_N$$

with:

$$I_P = K_P(L(V_{dd} - V_i - V_{PT})^2 - L(V_o - V_i - V_{PT})^2)$$

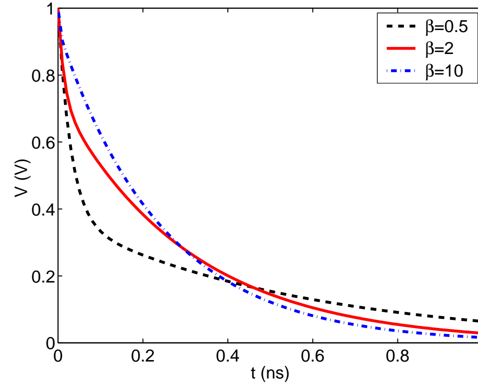


Fig. 35. The output waveform vs. β. $\alpha = 0.2$.

and

$$I_N = K_N(L(V_i - V_{NT})^2 - L(V_i - V_o - V_{NT})^2)$$

where the function $L(x)$ defined as:

$$L(x) = \begin{cases} 0 & : \quad x < 0 \\ x & : \quad x \geq 0 \end{cases}$$

Note that $L(x) \equiv x * Heaviside(x)$[50].

For a rising step input, we can write the time domain equation for the output voltage V_0 as:

$$-C_L \frac{dV_o}{dt} = \begin{cases} K(V_{DD} - V_T)^2 \ , & V_{DSAT} \leq V_o \leq V_{DD} \\ \frac{K(V_{DD} - V_T)^2}{V_{DSAT}^2}(2V_{DSAT} - V_o)V_o \ , & V_o < V_{DSAT} \end{cases} \tag{5.7}$$

where C_L is load capacitor.

The output waveform V_o is

$$V_o = \begin{cases} V_{DD} - \frac{t}{b} & : \quad V_{DSAT} \leq V_o \leq V_{DD} \\ \frac{2V_{DSAT}}{1 + ae^{2t/bV_{DSAT}}} & : \quad V_o < V_{DSAT} \end{cases} \tag{5.8}$$

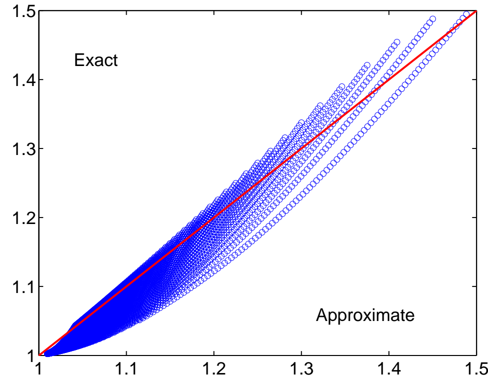


Fig. 36. Comparsion of Eq. (5.5) and the solution of Eq. (5.4).

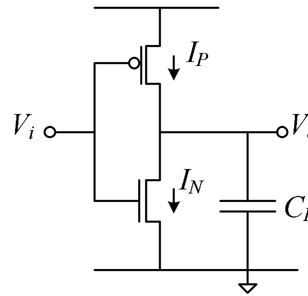


Fig. 37. CMOS inverter circuit.

where

$$a = e^{-2\frac{V_{dd}-V_{DSAT}}{V_{DSAT}}},$$
$$b = \frac{C_L}{K(V_{dd}-V_T)^2}$$

For any value x from 0 to 1, the time t_x at which the ouput waveform at V_o

reaches $x \cdot V_{DD}$, which means $V_o(t_x) = xV_{DD}$, is as follows:

$$t_x = \begin{cases} bV_{DD}(1-x)\ , \quad xV_{DD} \geq V_{DSAT} \\ \frac{bV_{DSAT}}{2}(ln(\frac{2V_{DSAT} - xV_{DD}}{xV_{DD}}) + 2(\frac{V_{DD} - V_{DSAT}}{V_{DSAT}})), \\ xV_{DD} < V_{DSAT} \end{cases} \tag{5.9}$$

From Eq. (5.9), we can derive any important timing information. For example, from Eq. (5.8), let us denote t_{SAT} as the time when the output waveform reach the boundary between linear and saturation regions, then

$$t_{SAT} = \frac{C_L(V_{DD} - V_{DSAT})}{K(V_{DD} - V_T)^2} = b(V_{dd} - V_{DSAT})$$

The equation in [43] is a special solution of Eq. (5.9) when $x = 0.5V_{DD}$.

For our requirement, we need to solve $t_{0.1V_{DD}}$ when V_o reaches $0.1V_{dd}$, which is:

$$t_{0.1} = \begin{cases} 0.9bV_{DD}\ , \quad V_{DSAT} \leq 0.1V_{DD}, \\ 0.5bV_{DSAT}(ln(\frac{20V_{DSAT}}{V_{DD}} - 1) + 2(\frac{V_{DD} - V_{DSAT}}{V_{DSAT}})) \\ 0.1V_{DD} < V_{DSAT} \end{cases} \tag{5.10}$$

To get the equivalent constant output resistance at $0.1V_{DD}$, we simply equate Eq. (5.10) to a simple RC circuit to get:

$$R_{Neff}|_{V_o=0.1V_{DD}} = \begin{cases} 0.9pV_{DD} \quad V_{DSAT} \leq 0.1V_{DD} \\ 0.5pV_{DSAT}(ln(\frac{20V_{DSAT}}{V_{DD}} - 1) + 2(\frac{V_{DD}}{V_{DSAT}} - 1)) \\ 0.1V_{DD} < V_{DSAT} \end{cases} \tag{5.11}$$

where

$$p = \frac{1}{ln(10)K(V_{DD} - V_T)^2}.$$

Recall that K is a function of the dimensions of the device, the gate oxide thickness

and the mobility [49]. The expression for the effective *pull-up* resistance is identical but would use the parameters corresponding to the P-channel device. For more complex gates, e.g., channel-connected components, K can be computed with the method proposed in [42].

Note that based on Eq. (5.9), for any voltage value, we can derive an equivalent resistance to match that voltage value for this MOSFET model.

D. Detailed Comparison

Fig. 38 is a comparison of the waveforms of an inverter driven by a fast (10 ps) rising pulse discharging a 3.818 pF capacitor vs. the equivalent effective resistance calculated from Eq. (5.11). For this example $V_{DD} = 2.5$, $V_T = 0.4$, the device dimensions for the N-channel device are $L = 0.5\mu$ and $W = 50\mu$. The Spice [38] level-1 model was used with $K_p = 2 \times 10^{-4}$ which translated to an $R_{Neff} = 32.476\Omega$. In the plot, the solid line is the output of the inverter while the dotted line are the output of the equivalent RC circuit.

For the second example, we use the circuit driven by the same 32.476Ω resistor in the previous example. From the example we determine that $\alpha = 0.215$ and $\beta = 0.4477$ which results in an effective capacitance multiplier (Eq. (5.5)) of $\eta = 1.4142$ and an effective capacitance of $C_{EFF} = 3.8183$ pF. Fig. 39 is a comparison of the resulting waveforms, with the solid line corresponding to the *Pi* model.

For the third example we combine the information from the two circuits above and compare the inverter driving the *Pi* section vs. the same inverter driving the effective capacitance. The results are plotted in Fig. 40 where the solid line corresponds to the *Pi* model.

These three examples show that the waveforms of simplified circuits modeled

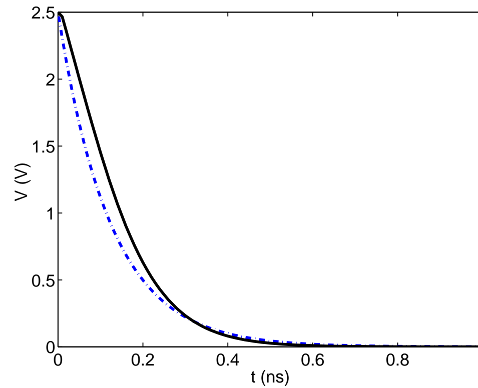


Fig. 38. Inverter vs. effective capacitance.

with our formulas Eq. (5.5) and (5.11) match the 10% point with the waveforms of original circuits.

E. Statistical Comparison

In order to validate the model over a wide range of operating conditions, we simulated the circuit shown in Fig. 41 with the same Spice level-1 model as above and over a wide range of randomly generated conditions. The variables varied are shown in Table XIII.

Table XIII. The variables used in the statistical simulation.

Parameter	Min	Max	Units	Distribution
Input Rise Time	1	100	ps	exponential
V_{dd}	1.5	2.1	V	uniform
V_{th}	0.35	0.45	V	uniform
C_L	50	500	pf	exponential
α	0.2	0.7	none	uniform

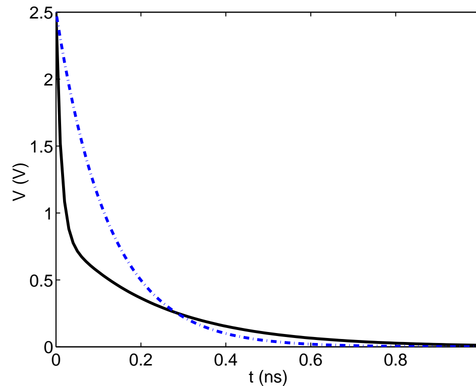


Fig. 39. Pi section vs. effective capacitance.

Some variables were sampled from a *uniform* distribution, while others which had a larger range were sampled from an *exponential* distribution. Note that we do not explicitely change β since the equivalent resistance for $0.5V_{DD}$ is different from the one for $0.1V_{DD}$.

For each sample, we calculated the equivalent output resistance from Eq. (5.11), and the effective capacitance from Eq. (5.5). We simulate the circuit and get the 10% points at V_A and V_B. Then another group of equivalent output resistance and effective capacitance are computed based on [43] and we simulate the new circuit and get the 50% points at V_A and V_B. The difference between 10% and 50% points are half the slew value, and the relative (percentage) errors of our method are computed. We performed a total of 5000 simulation. In aggregate, the error has a mean of -1.841 percent, and a standard deviation of 0.542 percent, with negative errors denoting an *over-estimation* of the effective capacitance. A histogram of the error distribution is shown in Fig. 43, and a plot of the computed slew versus real slew is shown in Fig. 42 in red squares.

For comparison, we also show the histogram of the slew error with only one

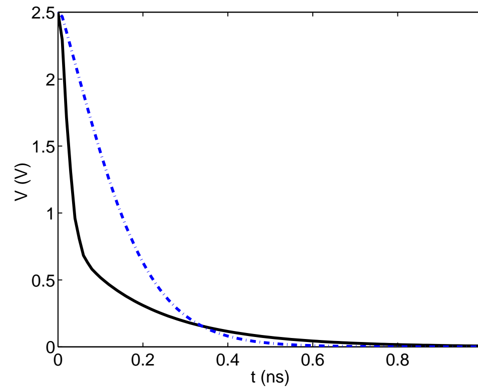


Fig. 40. Inverter driving Pi section vs. effective capacitances.

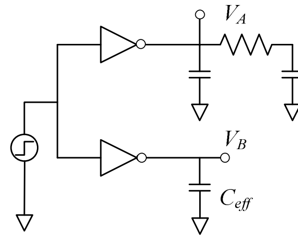


Fig. 41. Circuit used for statistical comparison.

effective capacitance proposed in [43] in Fig. 43, from which we can see that it has much bigger error compared to our method. The error has a mean of 3.856 percent, which is two times of our method, and more importantly, a standard deviation of 2.090 percent, which is 4 times larger than our method. A plot of the slew based on only one effective capacitance versus real slew is also shown in Fig. 42 in green circles. We also observed that single effective capacitance method always underestimate the slew, while our method mostly underestimate the slew. Generally designer tends to leave certain safety margin during analysis, which makes our model more appropriate while this trend also presents an opportunity for further refinement of the model, i.e., tune the coefficients in Eq. (5.5). It was shown in the [43] that, the largest delay

error dependency is on α. However, we found that for slew metric, the error is almost equally spanned to the whole range of α. Fig. 44 also shows a plot of the error vs. α in blue circles.

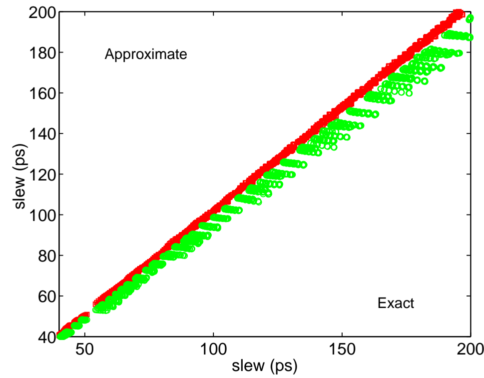


Fig. 42. Slew at V_A and V_B. Red squres represent our method. Green circules represent the single delay based effective capacitance method.

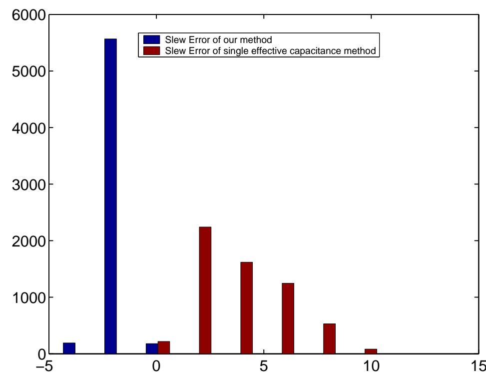


Fig. 43. Histogram of percentage slew error.

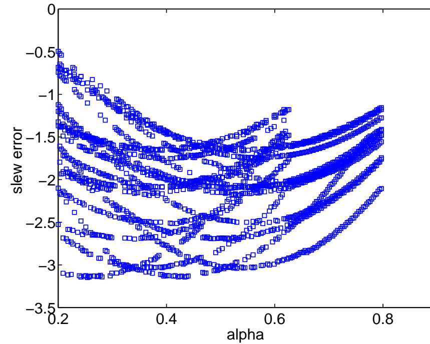


Fig. 44. Percentage slew error vs. α.

F. Conclusion

In this chapter we have presented an efficient and accurate method for calculating a new effective driving-point capacitance of RC interconnect to match the slew rate of logic drivers. With single delay based effective capacitance, we show that the slew error can not be ignored. Our method does not require iteration and gives simple closed-form formulas. The accuracy of the method was demonstrated to be more than adequate for applications in static timing analysis.

In the future, it may be possible to extend the analytical results from this model to other important performance metrics such as noise analysis, leakage power dissipation. With all analytical formulas, we can show the dependency of both the delay and slew on major technology variables such as threshold voltage, mobility and gate oxide thickness. This in turn can allow statistical static timing analysis, which often express manufacturing variations as distributional assumptions on the previous mentioned technology variables.

CHAPTER VI

AREA AWARE PHYSICAL SYNTHESIS FLOW

A. Background

Physical synthesis is the critical part of modern VLSI design methodologies. It refers to the process of placing the logic netlist of the design, as well as sizing, adding, and removing logic cells, concurrently optimizing multiple objectives under given constraints, where objectives and constraints are choices among area, power, timing and routability depending on design characteristics. In the last decade, timing closure is the main focus of physical synthesis flow [51], partially due to interconnect delay dominance over gate delay from technology mitigation.

So why do we suddenly care about area bloat? In 65 and 45 nm technologies, design companies tends to pack more logic functionalities into a small-sized die to balance the expensive fabrication cost, while they also want to keep low power budget to maintian their competitive margin. Such requirement could break the traditional timing driven flow which tends to consider area as merely a constraint and over-designs the chip.

Area bloat could cause several problems:

1. More power consumption. Area is the first order estimation of the power, especially for dynamic power. For leakage power, smaller area device tends to have less leakage even in the gate library family with same threshold-voltage (Vt).

2. Congestion problems. There are many causes for congestion problems, such as inferior floorplan and bad placement. Area bloat is one significant cause, which creates high logic gate density in certain regions, and there are not enough tracks to route all nets.

3. Timing problems. When the chip area has been fully occupied, there is no extra space for optimizations to further improve timing, or new buffers and resized gates are moved a long distance during legalization and big timing degradation happens.

Each problem described above or their combination could cause designers to increase die size, restart floorplanning and complete physical synthesis flow, which in turn lengthen the total time-to-market.

One example of area bloat causing congestion problems is shown below. For an industrial 45nm design with 102K input gates, we first run a traditional timing driven flow, and a global router with 5% detour length control to measure the congestion. The congestion picture is shown in Fig. 45 and the average congestion metric [1] is 94%, with 5535 nets pass through 100% routing congested tiles. Then with the techniques later discussed in this chapter, the area is reduced by 8%, and the congestion picture is shown in Fig. 46. The average congestion metric decreases to 89% with only 2856 nets passing through 100% routing congested tiles. The output netlist with the new technique can actually be routed with some further cell spreading techniques, where the original design has even no free space to be spreaded. Therefore, it is important to achieve a min-area physical synthesis flow.

The major source of the area bloat from the physical synthesis is buffering and gate sizing. Even though there are lots of existing literatures on these problems, there are still practical constraints that existing approaches do not model correctly.

[1]Measured by taking the worst 20% congested nets and averaging the congestion number of all routing tiles these nets pass through

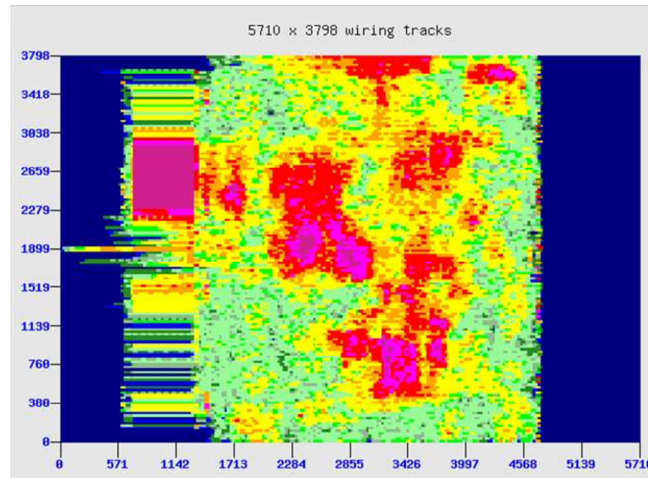


Fig. 45. Horizontal congestion from timing driven physical synthesis flow.

1. Buffer Insertion

Buffer (repeater) insertion, is a popular technique to reduce interconnect delay and fix electrical violations. Perhaps the most well-known buffer insertion algorithm is Van Ginneken's classic dynamic programming algorithm [52], which has led to several improvements for the kernel algorithm, such as speedup [53], resource control [54, 55], slew constrained buffering [56]. Other extensions or related work include buffer tree construction, buffering with more accurate delay models, buffering for noise reduction and simultaneous wire sizing and layer assignment.

Most of these literatures focus on single algorithm for a particular problem. However, creating an area efficient flow based upon these existing techniques and the way to handle all practical constraits are rarely discussed, which have big impact to the design area at the end of the flow. To list a few:

1. Should one use slew constrained buffering or timing driven buffering (they have different area and timing results)?

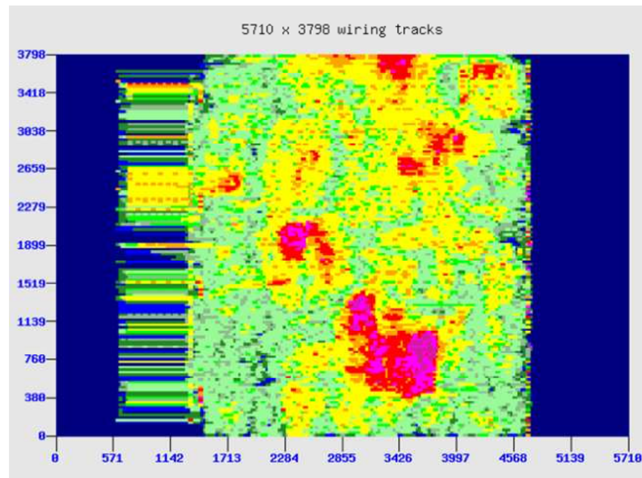


Fig. 46. Horizontal congestion from area efficient physical synthesis flow.

2. How to set the input slew and slew constraint for buffering algorithms, which has the big impact on the area?
3. Can one still use Elmore delay and linear gate delay model in buffer insertion for modern designs and any area impact?
4. How to handle rising and falling transitions inside the algorithm?

The impact of slew constraint on buffer area is shown in Fig. 47. The experiment is done at a 5 mm long line on a 2X wide/thick layers in 45 nm technology where buffers are placed at the max distance to meet the specified slew constrait in a repeated patten. As slew constraint becomes smaller, the distance between buffers is smaller, which results in more buffers and bigger buffer area. On another hand, the signal delay per mm, the sum of buffer delay and wire delay divided by the slew reach length, is measured and shown in the same curve. One can also see that by adjust the slew goal, one can achieve the optimal delay for a buffered wire without performing timing driven buffer.

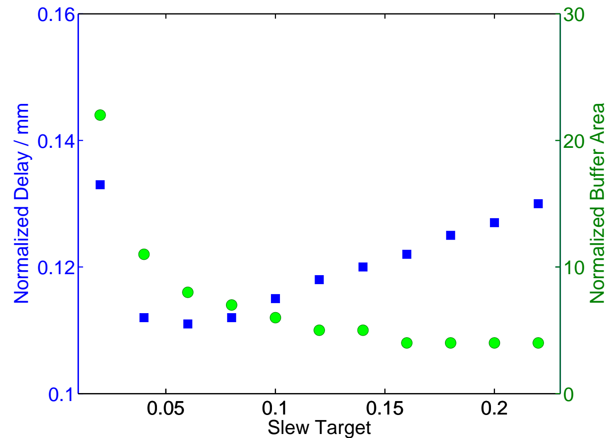


Fig. 47. Slew constrait (ns) vs. buffer area relationship is shown in green dots. Slew constraint (ns) vs. signal delay per mm relationship is shown in blue squares.

The impact of input slew and rising/falling inputs are illustrated in Fig. 48, where we choose we choose a buffer from a 45 nm node buffer and show the relationship between the delay and capacitance under different input slew values and rising/falling input transitions. It is clear that the delay is also quite sensitive to the input slew, as well as the rising and falling input directions, and the difference could be 10% to 15%.

2. Gate Sizing

Gate sizing has also been extensively studied in the literature. Most early work assumes the library is continuous, models the sizing problem as a convex programming with closed form solution [57] or Lagrangian relaxation [58]. These work ignore the fact that most cell library based designs have discretized gate sizes. Later, some rounding techniques have been proposed to map the continuous solutions to the discrete grid [59]. More recently, Liu and Hu [60] proposed to use dynamic-programming style algorithms for solving discretized gate sizing problem directly. However, the slew

impact is still ignored when propagating the solutions in the dynamic programming. Also all previous work tend to optimize the worst critical path, where the sum of negative slacks (referring to Figure of Merit) are always ignored, which is also an important factor to measure design quality, especially when the worst critical path stuck during the optimization with either logic structure problems or wrong timing assertions.

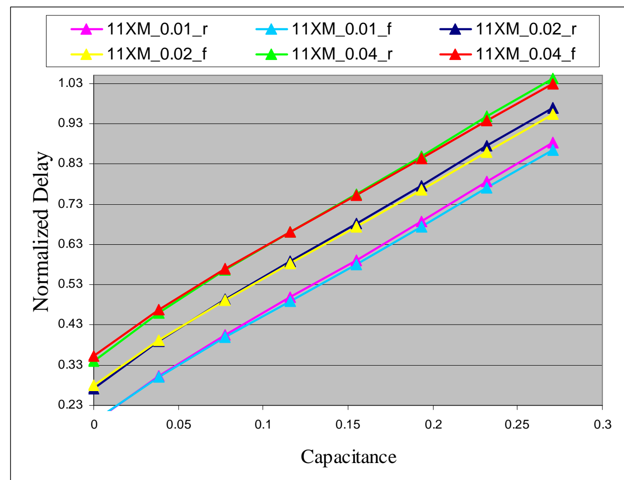


Fig. 48. Delay vs. cap for a buffer in 45 nm node. Three different input slew values, 10, 20 and 40 ps, are used here. 11XM_0.2_r refers to 11X driving strength buffer, 20 ps input slew and the rising inputs.

The following figures show the relationship between delay and area for one complete buffer library in 65 nm node (Fig. 49) and one in 45 nm node (Fig. 50). The delay is normalized and the buffer area is measured by its width. The capacitance load is the sum of the capacitance of a typical interconnect length plus the capacitance of a typical buffer for the corresponding technology. The input slew is set at 200 ps for 65 nm node and 40 ps for 45 nm node. Both rising and falling delay values are shown

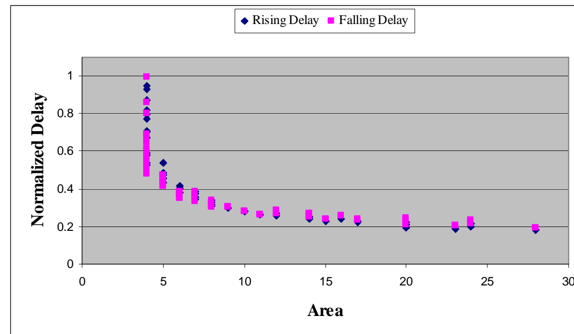


Fig. 49. Delay vs. area for a buffer library in 65 nm node.

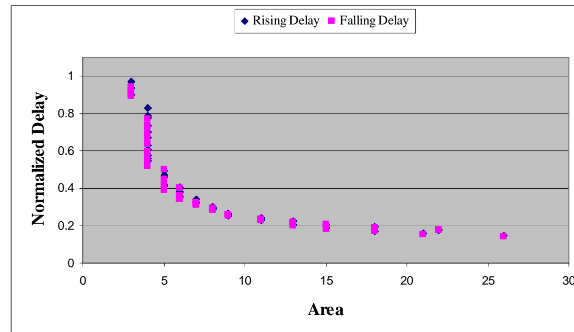


Fig. 50. Delay vs. area for a buffer library in 45 nm node.

in the figures. Note that not only the library is discrete [2], but also there are many buffers with the same area (or footprint) which have totally different delay values. It is caused by different N/P transistor strength for rising/falling balance or the choice of transistor sizes in the first and second inverters. Therefore, all assumptions such as "convex" and "continuous" do not work for cell based designs, and even rounding approach will meet problems when many gates share the same area. Also, as shown in Fig.48, the delay is sensitive to the input slew too. Such relationship between delay

[2]the area of a gate is generally measured by its width in the standard cell methodologies, since the vertical track is generally fixed

and area is also found in other logic gate library, such as AND/NAND/OR/XOR gates.

3. Our Contribution

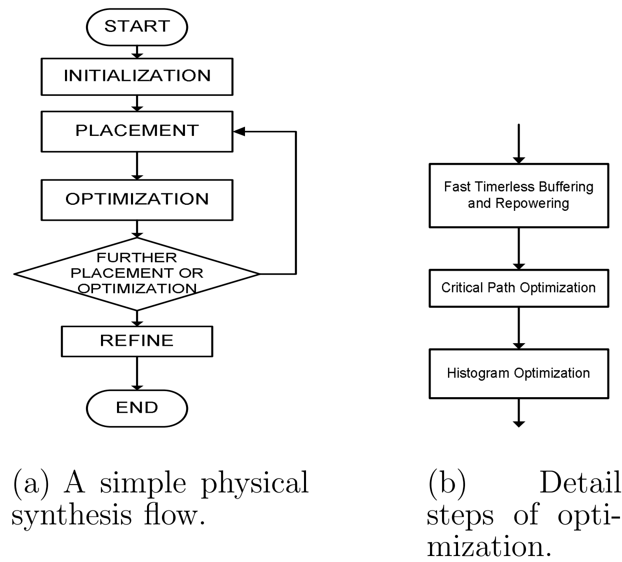
In this work, rather than providing a new theoretical algorithm on buffering and gate sizing problems, we present practical guide and experience of how to put an efficient incremental optimization step inside a physical flow with good area/timing tradeoff. Papers in this category are rarely seen. For a designer or a new EDA startup, we hope this work can provide a quick guide without looking through 100 papers on buffering and gate sizing. Our contributions in this chapter are

- An area efficient iterative slew-tighten approach for slew driven buffering and gate sizing (iterative EVE);
- A simple area efficient timing driven gate sizing method for cell library designs;
- A new area efficient optimization flow with practical buffering and gate sizing techniques to handle modern design constraints.

B. Overview of Existing Physical Synthesis Flow

A timing-driven physical synthesis flow described in [61] includes the following steps: 1) initial placement and optimization; 2) timing-driven placement and optimization; 3) timing-driven detailed placement; 4) clock insertion and optimization; 5) routing and post routing optimization; A simple diagram is shown in Fig. 51(a) with the placement and optimization part, where refine step refers to optimization at the finer level.

Further, optimization can also be broken into 3 steps: 1) electrical correction; 2) critical path optimization; 3) histogram compression.



(a) A simple physical synthesis flow.

(b) Detail steps of optimization.

Fig. 51. Flow diagram.

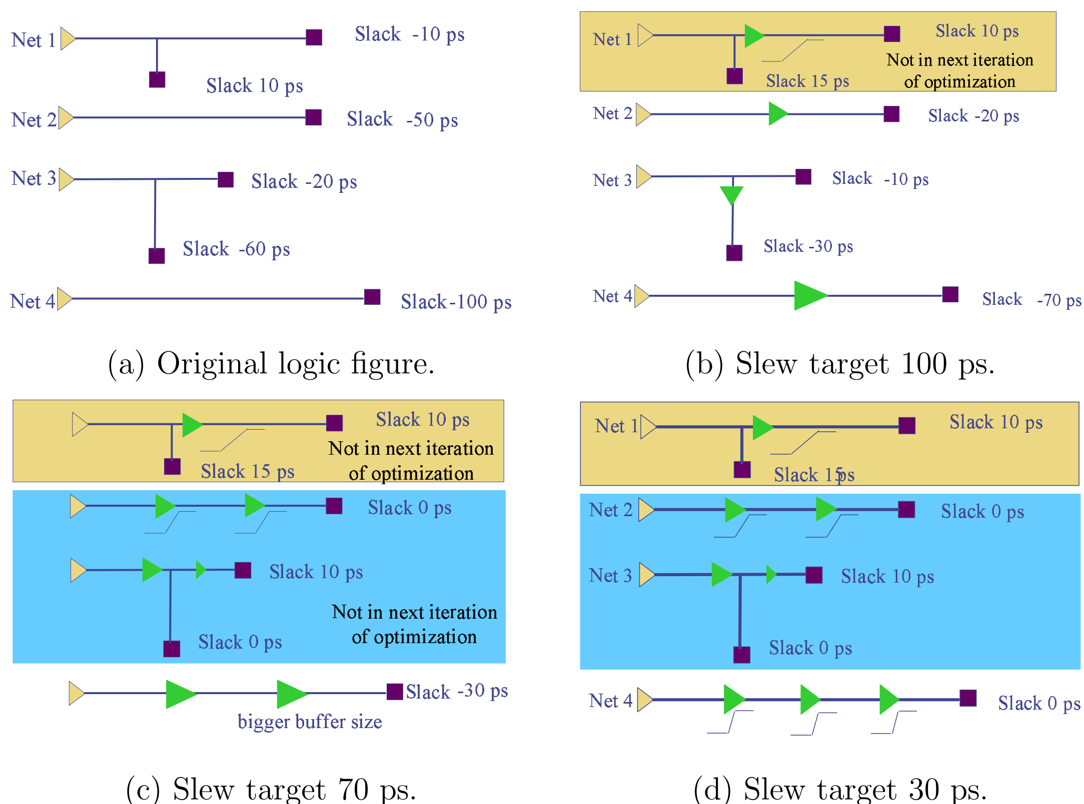
The purpose of electrical correction is to fix the capacitance and slew violations with buffering and gate sizing. In general, one wants to first perform electrical correction in order to get the design in a reasonable state for the subsequent optimizations. In [62], an electrical violation eliminator (EVE) technique is used to fix electrical violations through fast slew based gate sizing and buffering

Then more accurate but slower timing based buffering approach and gate sizing method is applied in the critical path optimization and histogram compression stages for the rest of the critical paths or nets. A simple optimization flow diagram is shown in Fig. 51(b).

This flow has the speed and timing performance advantage as shown in [61]. However, it has several drawbacks to cause the area bloat. In the following sections, we describe two main techniques to shed the area bloat for this flow, though the merit of techniques may benefit other flows too.

C. Iterative EVE

The concept of original EVE technique is to combine slew driven gate sizing and slew constrained min-cost buffering [56], and process gates from primary output (or latch inputs) to primary inputs (or latch outputs) in the combinational circuits. If a net has no violations, size the source gate down to save area and reduce load on inputs. If a net has violations, size the source up; if the biggest size cannot fix the slew violation, perform buffering on the net. This approach has several advantages, 1) combining buffering and gate sizing seamlessly; 2) high efficiency and 3) no reconvergency problem (all decisions are local for electrical corrections).



(a) Original logic figure.

(b) Slew target 100 ps.

(c) Slew target 70 ps.

(d) Slew target 30 ps.

Fig. 52. A simple example for iterative EVE.

EVE is motivated by the fact that most nets in modern designs either (1) have only electrical violations but without timing violations, or (2) have positive slacks

after electrical violations have been fixed. More importantly, slew constrained min-cost buffering is much faster than timing driven buffering because slew constraints can usually be handled locally.

However, as explained in [56], the buffering algorithm is very sensitive to the input slew and the slew target at the sinks. Buffering with a tight slew target (e.g., the slew target is set to be the saturation slew in a long optimal buffered-line[14]) can usually achieve similar timing performance as timing driven buffering, however, surplus buffer insertion will be done for nets with positive or near-positive slacks and loose original slew targets. On the contrary, using a relaxed slew target can save area but it unacceptably sacrifices performance, and more nets need timing driven buffering afterwards. It also slows down the runtime because for timing verification has to be done for each new solution to make sure no timing degradation occurs. Similar situation happens to gate sizing operations, though we use buffering as the main example for the rest of the section. In the following, we propose a new iterative method which gradually tightens slew target gradually.

Instead of starting with a tight slew target, the initial slew target is set based on the operating frequency of the design, which could be the tightest slew constraint for all data phases excluding scan. Sometimes, the initial slew constraint is provided by the designers. Comparing to the initial slew target, each net may have its own slew constraint (e.g., from design rules), and the tighter one is used during optimization for each net. In the first iteration, most of the nets may end up using its own slew constraint, but later in the process, the global slew target gets tighter and eventually overrides the local ones to guide timing optimization.

For the input slew, we start with the saturation slew along a long optimal buffer chain. This is a applicable for nets which need the most aggressive buffering. For the other nets, this is a conservative assumption and results in better area since the

saturation slew is usually smaller than the initial slew target.

With these settings, we run EVE and follow with a full static timing analysis. Then, the slew target is reduced by a given percentage (say 10% or 20%) and input slew is updated by taking a set of worst paths and averaging their input slew. EVE is run again for all negative nets with the right to left order. In all iterations, buffers on negative paths are completely ripped up and rebuilt. The process is repeated until the slew target is smaller than the saturation slew of an optimal buffered chain. Our experiments generally converge in 3 to 5 iterations for 65nm and 45nm technology, and the overhead due to static timing analysis is acceptable. This approach is significantly faster than the traditional timing driven buffering approach for all nets, which returns similar area results.

An exemplary circuit is shown in Fig. 52(a) to Fig. 52(d) . In this simple example, there are 4 nets and the initial design structure in Fig. 52(a) could be an optimized placement from a commercial tool, a random placement or a custom placement. In Fig. 52(a), there is no repeaters. The slack of all sinks is negative as shown in figure.

Fig. 52(b) represents the structure after the first iteration of EVE which generates the first optimized design structure. This iteration uses a global slew target of 100 ps, and a buffer has to be inserted in each net in order to meet the slew target. These buffers may have different sizes in a buffer library. After the first iteration, the slacks of all nets are shown in Fig. 52(b). Since the slack of both sinks in the first net become positive, the first net is skipped in the future iterations.

After the first iteration, the slew target is down to 70 ps from 100 ps. Fig. 52(c) shows the result of the second iteration of EVE. In this iteration, additional buffers have been inserted into the second, third nets to make their slack positive. They will then be skipped in future iteration.

In the third round, the slew target becomes 30 ps. The optimized design structure

is shown in Fig. 52(d). Since the first three nets were skipped for this iteration, additional buffer is only inserted in the forth net as shown in Fig. 52(d). The final slack of the forth net is 0 ps. This is an ideal case where all nets now have positive slack, and further timing driven optimization is not necessary.

While the example of Fig. 52(a) to Fig. 52(d) illustrates only 4 nets and a total of eight inserted buffers, in real designs, the number of nets is typically in the thousands, with the insertion of as many as 500,000 buffers.

With this method, we can insert as fewer buffers as possible to meet timing requirement and the total area and wirelength is greatly reduced.

D. Area Efficient Timing Driven Gate Sizing

Timing driven gate sizing is used in the critical path optimization and histogram compression stage. It needs to be accurate, incremental and harmless.

As discussed in the Sec. 2, the discrete nature of the cell library for standard-cell based designs, the slew impact and the FOM problems, make existing approaches such as convex programming either be unrealistic to use, or inaccurate. Also, previous approaches tend to find the sizing solution for the whole circuit, or a group of hundreds to thousands of gates with internal delay models, and then apply it. The scale of changes, combined with the model inaccuracy, may result in big rejection rate from the static timing analysis with slew propagation, and even some good partial solutions may be thrown away.

In our implementation, we choose to use a simple gate sizing approach. We first give an order of all boxes (could be based on slack or sensitivity), and then work on each single box at each time. After choosing the right power level, perform the incremental timing analysis to update the timing graph, and move on to the next

box. It looks like quite naive, but is more accurate for the local change, and also tends to give the best FOM result since even a box is on the not-near-critical paths, as long as the slack is still negative and can be improved, a better solution will be chosen. The whole process can also be iterated. One can speedup the process by simply limiting the update scope of static timing analysis engine when every size of the gate is evaluated.

We can resize boxes as long as the slack is improved, however, the slack only get a little bit improvement with lots of area resource sometimes. To be area efficient, the minimum improvement threshold δ is defined as the minimum required slack improvement per unit area change. When a particular gate is resized, we only accept the new solution, if the slack improvement of new solution compared to the previous best solution is bigger than δ times the area difference. As shown in Fig. 53, rather than choosing the best slack solution with area A_5, we pick the solution with much smaller area A_4, with acceptable timing degradation.

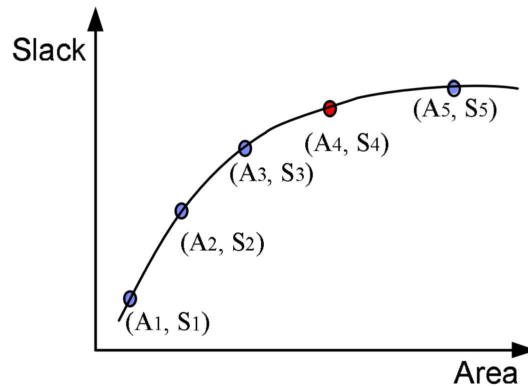


Fig. 53. Area aware gate sizing.

E. New Area Efficient Optimization Flow

The new flow assembled from iterative EVE is illustrated in Fig. 54. The area aware gate sizing is the critical path and histogram stages, where the cost can be tuned for different design requirement

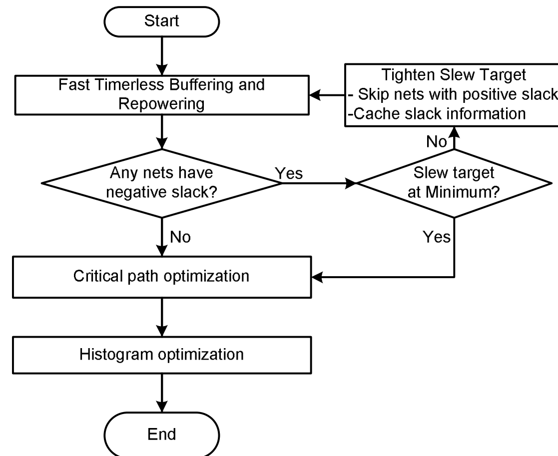


Fig. 54. New optimization flow.

F. Other Practical Techniques

In any physical synthesis flow, a timing driven buffering tool plays an essential role. We implement the techniques described in [55] to control the buffer resources, but to make it more practical, several tunings need to be done.

Handing rising and falling transitions: the buffering algorithm not only needs to handle polarities and inverters, also needs to distinguish the rising and falling signals during the bottom-up dynamic programming since the delay for both edges are noticable different (Fig. 48).

Delay Modeling: Elmore delay model is too conservative and causes over-buffering where moment based computation is too slow. We use scaled Elmore delay model (0.8 as factor) and linear gate delay models, and found the buffer locations are quite close to the solution from the moment based wire delay models and look-up tables based gate delay models, while the runtime is much faster. We suspect the buffering is a global scaled operation.

Speed: We implement following techniques: range search tree pruning, convex and predictive pruning techniques to gain speedup from solution pruning; buffer solution storage technique in [53] to gains speedup by avoiding naive copying solution list during adding wire operations; layer assignment techniques in [63] to gain speedup by efficiently utilizing available high level metals for each technology.

G. Experiments

We implement the new optimization flow in C++ and embed in the flow shown in Fig. 51(a). Twenty industrial designs (eighteen 65nm and two 45nm) are selected, ranging from 100,000 to 1 millon gates in the input netlist. All experiments are run on a 2.9G Hz machine with Linux operation system.

1. Iterative EVE vs Single EVE

In this experiment, we compare our iterative EVE algorithm with the single EVE algorithm which uses the aggressive slew target to get the best timing. Both optimizations are performed after initial placement, which produce a huge slack and FOM improvement since no net (including high fanout nets) has been buffered (other than polarity inverters). We compare the worst slack (WSLK) improvement (the difference of before/after worst slack), FOM improvement, and area increase due to the optimization. Data for only 10 designs are show in Table XIV to save the space. In

Table XIV. The QOR comparison for iterative EVE.

	Type	FOM Imp (ns)	WSLk Imp (ns)	Area Inc
test1	Single EVE	1150890.0	464.87	946003
	Iterative EVE	1153195.6	464.05	198297 (20.96%)
test2	Single EVE	3425630.0	718.20	677286
	Iterative EVE	3344172.7	593.24	189253 (27.94%)
test3	Single EVE	993808.0	125.45	369589
	Iterative EVE	1093409.7	148.62	151493 (40.98%)
test4	Single EVE	8564860.0	426.96	1221551
	Iterative EVE	8378022.8	395.96	490440 (40.14%)
test5	Single EVE	1416620.0	160.65	1611279
	Iterative EVE	1340335.3	167.00	331856 (20.60%)
test6	Single EVE	12550800.0	968.76	811444
	Iterative EVE	12767810.0	1027.0	271742 (33.49%)
test7	Single EVE	9480330.0	369.28	1200267
	Iterative EVE	7593774.2	366.07	454308 (37.85%)
test8	Single EVE	35511100.2	1918.0	1168543
	Iterative EVE	35530547.4	1928.0	414317 (35.45%)
test9	Single EVE	11674800.3	984.47	1103223
	Iterative EVE	11157401.0	1013.3	331287 (30.03%)
test10	Single EVE	66256.2	66.04	326920
	Iterative EVE	64255.5	65.47	93924 (28.73%)

summary, iterative EVE approach uses as less as 20% area of of single EVE, while achieving similar timing quality. There is runtime overhead since we run multiple iterations, which is generally 2 to 4 times depending on the design.

2. Timing Driven Gate Sizing

In this section, experiment results with different minimum improvement threshold δ for timing driven gate sizing are shown in Table XV. The timing driven gate sizing is performed in "critical path optimization" stage after iterative EVE, which also explains the scale of the improvement compared to Table XIV. The unit of δ is pico seconds per unit area change. When $\delta = 0$, it gives the best timing, and when $\delta > 0$, area cost is considered. From Table XV, the increased area becomes smaller when δ

Table XV. The QOR comparison for area efficient gate sizing.

	δ	WSLK Imp	FOM Imp	Area Inc
test1	0	0.08041	2373.05	200062
	0.005	0.00993	1198.84	62733
	0.010	0.00791	629.81	7217
	0.030	0.00146	268.06	-10162
	0.050	0.00146	252.72	-10370
test2	0	0.03775	1167.43	68958
	0.005	0.00902	646.73	18254
	0.010	0.00813	307.5	1940
	0.030	0.00813	138.93	-1785
	0.050	0.00813	123.46	-1862
test3	0	0.02486	164.51	31861
	0.005	0.00209	91.42	6859
	0.010	0.00209	45.89	325
	0.030	0.00209	19.15	-2575
	0.050	0.00209	19.09	-2593
test4	0	0.06008	309.02	13329
	0.005	0.02325	241.25	3862
	0.010	0.00021	83.67	-120
	0.030	0.00021	39.39	-862
	0.050	0.00021	39.05	-870

increases. It is interesting to see that when δ is big enough, the area starts to decrease and one can still achieve the worst slack and FOM improvement.

3. Overall Flow Comparison

In this part, we put iterative EVE and area aware gate sizing (with $\delta = 0.005$) in the flow and compare to the baseline (single EVE and $\delta = 0$). Both flows go through complete physical synthesis, include 2 iterations of placement (the second one is timing driven placement with net weight updated according to the timing information), optimizations, timing driven buffering, detail placement, legalization and the refine part. Both flows also use the practical techniques mentioned in Section V too.

For all experiments, we compare area, worst slack (WSLK), FOM, wirelength

(WL) and runtime at the end of physical synthesis and the results are shown in Table XVI.

From Table XVI, our new flow saves 5.8% total area compared to the the baseline flow on average, and the maximum area saving is 12.5%. Considering the area is the first order estimation of the gate power, the number is significant. In addition to total area, we reduce the logic area growth by 12%. Logic area is defined as the amount of area which a physical synthesis tool can "optimize", which excludes fixed and non-sizable cells , such as memory, SRAM, fixed macro blockages, etc. The logic area growth is then the ratio between the increased logic area and the initial logic area,

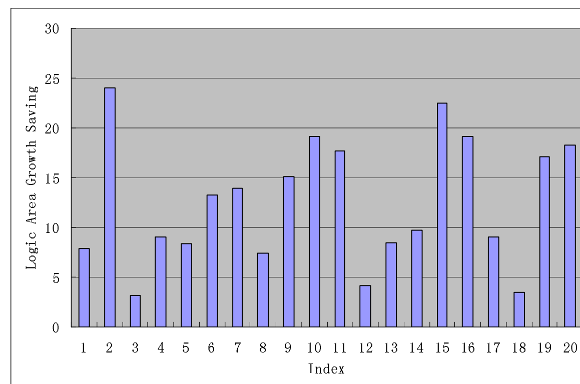


Fig. 55. Logic area growth saving compared to baseline flow.

Fig. 55 shows the results of the logic area growth saving for all designs comparing our new flow to the baseline flow. On average, we reduce the logic area growth by 12%.

For example, if the logic area growth of the baseline experiment is 30%, and the logic area growth of our flow is 10%, then the logic area growth saving is 20%. Compared to the baseline, the area recipe can save 12.5% logic area growth on average,

and up to 23% for some designs.

As we mentioned before, area bloat also causes the routing problems and timing problems. The design shown in Fig. 45 and Fig. 46 is actually the "test1" design in Table XVI. As discussed before, the routability of this design is very sensitive to the area. With our new flow, "test1" is not only routable, and the timing is near to close (the slack threshold is 0.1 ns). Compared to the baseline flow, the wirelength is reduced by 60.8%, and the worst slack is improved by over 12 ns. "Test10" shows the similar trend with 45.4% wirelength reduction and 1.8 ns worst slack improvement. On average, our new flow achieves 10.2% wirelength reduction. The worst slack is improved by 770 ps and FOM is improved by 7085 ns on average. Out of 20 designs, our new flow gives better slack for 12 designs compared to the baseline flow (same for 2 designs), and gives better FOM for 10 designs (same for 5 designs).

The main reason for the timing and the wirelength improvement is: 1) More free-space to insert buffers at the desirsed location or size gate up without moving; 2) Better timing before timing driven placement will make placement move less objects and results in less wirelength increase; 3) With more freespace, legalization tends to have minimal impact on the wirelength and timing.

As we see, timing, congestion and area problems are coupled together. By significantly reducing the area bloat, our techniques improve the timing, wirelength, and congestion.

H. Conclusion

In this chapter, we pointed out the major source of area increase in a typical physical synthesis flow is from buffer insertion and gate sizing, both of which have been discussed extensively in last two decade, where the main focus is individual optimized

algorithm. We present two simple, yet efficient, buffering and gate sizing techniques and achieve a physical synthesis flow with much smaller area bloat Compared to a traditional timing driven flow, our work achieves 12.5% logic area growth reduction, 5.8% total area reduction, 10% wirelength reduction and 770 ps worst slack improvement on average on 20 industrial designs in 65nm and 45nm.

Table XVI. The QOR comparison of baseline and new flow.

Circuit	Type	Area	CPU	WSLK	FOM	WL
test1	base	1.30777	2812.7	-12.738	-115965	46175031
#gates:102046	new	1.15874	4483.7	0.099	0	18100494
test2	base	20.2130	45229.6	-0.170	-2605	381929854
#gates:717075	new	19.5043	52505.1	-0.288	-5669	383202138
test3	base	1.92936	5948.8	-0.026	-2	34700143
#gates:110118	new	1.8862	6326.5	0.100	0	34175838
test4	base	10.9556	17771.0	-0.436	-928	121271648
#gates:253815	new	10.7207	22537.3	-0.436	-203	116457562
test5	base	5.99029	28559.2	0.099	0	150358750
#gates:663693	new	5.51185	31977.5	0.096	0	141428293
test6	base	8.29468	13441.5	-0.747	-37	54379592
#gates:94220	new	8.15084	17410.4	-0.754	-36	50603208
test7	base	16.3154	18454.7	-0.089	-1	273579209
#gates:367478	new	15.8426	22535.1	0.039	0	257814794
test8	base	7.46228	18598.5	0.012	0	136021272
#gates:476495	new	7.12724	22208.1	0.013	0	142562718
test9	base	4.81608	14899.8	-0.512	-357	101211621
#gates:168379	new	4.53251	17373.3	-0.375	-118	89450246
test10	base	5.54470	27398.3	-1.805	-27248	165713654
#gates:347502	new	4.85059	24215.5	-0.032	-53	90467156
test11	base	9.97560	30120.5	-0.528	-696	151761936
#gates:517459	new	9.27245	37599.5	-0.525	-726	144139156
test12	base	9.92720	28309.0	-0.347	-117	128984471
#gates:517583	new	9.27470	36257.1	-0.346	-156	124913294
test13	base	6.04183	18874.4	0.087	0	157218211
#gates: 554423	new	5.82644	20980.1	0.098	0	154116834
test14	base	3.11515	6193.2	0.100	0	33563568
#gates:142542	new	3.01044	6713.1	0.100	0	28902330
test15	base	9.95405	31284.2	-0.094	-1	269291914
#gates:797963	new	9.24077	39238.8	-0.057	-22	255280710
test16	base	13.7727	57974.5	-0.743	-4498	404978253
#gates:1066512	new	12.1618	71217.3	-0.646	-4048	397577302
test17	base	14.4742	30467.7	-0.253	-55	160451598
#gates:424465	new	13.7434	37764.0	-0.486	-63	142016166
test18	base	5.15163	15858.4	-0.412	-30	158948356
#gates:416142	new	4.79116	17207.0	-0.403	-29	137246466
test19	base	4.66864	9726.1	-0.200	-58	61616468
#gates:246524	new	4.59064	11020.5	-0.200	-58	62023304
test20	base	7.00458	27110.2	-0.688	-332	139740251
#gates:494645	new	6.41238	33254.7	-0.100	-54	130307063

CHAPTER VII

CONCLUSION AND SUMMARY

In this disseration, we firstly presented several methodologies and algorithms for interconnect extraction and circuit modeling and simulation. Then, we proposed a new optimization flow to shed area bloat in physical design synthesis flow.

With VLSI technology development, most of current ASIC chips, game processor chips and microprocessors are manufactured at 65 and 45 nanometers, and even several test/prototype chips manufactured in 32 nm technology is on the way. Since the feature size is much smaller than sub-wavelength lithograph wavelength, the manufacturing cost is significantly increasing in order to achieve a good yield. On the other hand, design companies need to get further aggressive to lower the power assumption, pack more logic functionalities on the fixed die size, and meet stringent performance requirement to keep the competitive margin in nowadays market. All these factors bring new challenges, as well as opportunities, for design automation tools in next decade. Simulation and modeling are areas which especially needs to have more accuracy, handle more design constraints and work with modern manufacturing process.

Interconnect parasitic extraction is the process of building the electrical and magnetic models for the physical shapes of interconnect metal layout and the media they are embeded and extracting the corresponding electrical and magnetic parameters for the circuit simulation. As interconnect performance is more dominant than logic performance since 90 nm, the accuracy and speed of interconnect parasitic extraction is more than important than ever for various steps in the design flow, such as synthesis, optimization, simulation and verification. Traditional 2D or 2.5D based method can not meet the new requirement of accuracy. Even 3D extraction needs to consider problems arising from new technology nodes, such as manufacturing variations, litho

process, multiple metal layers and complicated dielectric media. For example, there are 8 metal layers in 65 nm and 10 metal layers in 45 nm from IBM technology and each layer may have different dielectric constants with multiple planar, conformal or embedded dielectric media. In addition, even with OPC/PSM, the lithography process will cause the interconnect metal shapes on wafer different from those ideal rectangular shapes drawn on mask. Fast and accurate extraction algorithms with all these new constraints and challenges are extremely important. Starting with 3D capacitance extraction, a new method to efficiently handle multiple planar, conformal or embedded dielectric media is propsed. Previous algorithms based on Boundary Element Method (BEM) are inefficient due to the complex dielectric structures. We present a new algorithm (HybCap) that combines multilayer Greens function with the equivalent charge method to efficiently deal with the complex dielectrics. The multilayer Greens function is efficient to model layered dielectric media, while the equivalent charge method is powerful to model non-planar complex dielectric. Our method can also efficiently model ground plane and reflective boundary wall. From experimental results, the new method is significantly faster than previous methods in realistic conditions, i.e., 70X speedup and has a 99% memory savings compared with FastCap and 2X speedup, and has a 80% memory savings compared with PHiCap for complex dielectric structure with similar accuracy. Then, in order to consider lithography effect in the existing Layout Parasitic Extraction (LPE) flow, I presented a modified Layout Parasitic Extraction (LPE) flow and fast algorithms for interconnect parasitic extraction considering photo-lithography effects. Our techniques are efficient, compatible with the existing design flow and with high accuracy.

Even with extracted parasitic parameters, one still need to use them efficiently to build interconnect circuit models and study the interconnect impact on various new problems. One big question in SRAM simulation is the lack of knowledge if BEOL

lithography process has any impact on SRAM yield and performance in the advanced technology node. Traditionally SRAM performance is mainly dominated by FEOL process where gate length and voltage threshold (V_{th}) variations are dominant factors. As interconnect performance becomes more dominant and lithographic variation may cause misalignment, it is important to build a methodology built upon new extraction techniques to study BEOL parameter variation impact. With the new enhanced parasitic extraction flow, simulation of BEOL effect on SRAM performance becomes possible. A SRAM simulation model with internal cell interconnect RC parasitics is proposed in order to study the BEOL lithographic impact. The impact of BEOL variations on memory designs are systematically evaluated. The results shows the power estimation with our BEOL model is more accurate and misalignment impact became severe when the resistance is the same order of magnitude as the nonlinear device resistance.

Another popular but unsolved problem related to interconnect modeling is the effective capacitance modeling for logic gate delay and slew computation. Traditionally the effective capacitance is mainly computed to match the logic gate delay from the input to the output, but the same model is also used to compute the slew of the waveform at gate output, which may bring big inaccuracy. We proposed a new effective capacitance model which translates an interconnect pi-model to a single effective capacitance value for gate output slew computation. Based on the model proposed in this work, I recommend the effective capacitance should be separated modeled for delay and slew computation to get the best accuracy and the traditional method has the big flaw if only one capacitance model is applied. The conclusion not only hold for traditional voltage based models, but could also be used for new current source models, and can be seen as one step further to the final goal, an effective capacitance model for full waveform match.

Even with all accurate simulations and models, design flow may still need to be re-tuned to meet more stringent power and area requirement in advanced technology nodes, as well as not impacting the timing performance. A new optimization flow to shed area bloat in the design synthesis flow is proposed, which is one level beyond simulation and modeling to directly optimize the design, but is also built upon accurate simulations and modeling. Area bloat in physical synthesis not only increases power dissipation, but also creates congestion problems, forces designers to enlarge the die area, reruns the whole design flow, and postpones the design deadline. As a result, it is vital for physical synthesis tools to achieve timing closure with intelligent area control. In this dissertation, I present two efficient buffering and gate sizing techniques in order to achieve a physical synthesis flow with much smaller area bloat compared to a traditional timing driven flow. The results show that the new flow achieves 12.5% logic area growth reduction, 5.8% total area reduction, 10% wirelength reduction and 770 ps worst slack improvement on average on 20 industrial designs in 65nm and 45nm.

REFERENCES

[1] ITRS Organization, "International technology roadmap for semiconductors 2007 edition lithography. [Online]," Available: http://www.itrs.net/Links/2007ITRS/2007_Chapters/2007_Lithography.pdf, 2007.

[2] E. Chen and E. Lam, "From ITRS roadmap to latest lithography development. [Online]," Available: http://maltiel-consulting.com/ITRS_Roadmap_to_Lithograph_Copper_CMP_Planarization.html, 2006.

[3] K. Nabors and J. White, "Multipole-accelerated 3-D capacitance extraction algorithms for structures with conformal dielectrics," in *Proceedings of the 29th ACM/IEEE Design Automation Conference*, June 1992, pp. 710–715.

[4] S. Yan, V. Sarin, and W. Shi, "Sparse transformations and preconditioners for hierarchical 3-D capacitance extraction with multiple dielectrics," in *Proceedings of the 41st annual Design Automation Conference*, June 2004, pp. 788–793.

[5] J. Zhao, W. W. M. Dai, S. Kapur, and D. E. Long, "Efficient three-dimensional extraction based on static and full-wave layered green's functions," in *Proceedings of the 35th annual Design Automation Conference*, New York, NY, USA, 1998, pp. 224–229.

[6] W. Shi, J. Liu, N. Kakani, and T. Yu, "A fast hierarchical algorithm for 3-D capacitance extraction," in *Proceedings of the 35th annual Design Automation Conference*, June 1998, pp. 212–217.

[7] J. R. Phillips and J. K. White, "A precorrected-FFT method for electrostatic analysis of complicated 3-D structures," *IEEE Transaction on Computer-Aided*

Design of Integrated Circuits and Systems, vol. 16, no. 10, pp. 1059–1072, October 1997.

[8] Z. Zhu, B. Song, and J. White, "Algorithms in FastImp: a fast and wideband impedance extraction program for complicated 3-D geometries," in *Proceedings of the 40th annual Design Automation Conference*, June 2003, pp. 712–717.

[9] J. Qian, S. Pullela, and L. T. Pillage, "Modeling the 'effective capacitance' for the RC interconnect of CMOS gate," *IEEE Transcations on Computer-Aided Design of Integrated Circuits and Systems*, vol. 13, no. 12, pp. 1526–1535, December 1994.

[10] J. Tausch and J. White, "A multiscale method for fast capacitance extraction," in *Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, June 1999, pp. 537–542.

[11] M. W. Beattie and L. T. Pileggi, "Electromagnetic parasitic extraction via a multipole method with hierarchical refinement," in *Proceedings of the 1999 IEEE/ACM International Conference on Computer-Aided Design*, November 1999, pp. 437–444.

[12] C. Wei, R. F. Barrington, J. R. Mautz, and T. K. Sarkar, "Multiconductor transmission lines in multilayered dielectric media," *IEEE Transaction on Microwave Theory and Techniques*, vol. 32, no. 4, pp. 439–450, April 1984.

[13] W. Dai, Z. Li, and J. Mao, "Parameter extraction for on-chip interconnects by double-image green's function method combined with hierarchial algorithm," *IEEE Transaction on Microwave Theory and Techniques*, vol. 53, no. 7, pp. 2416–2423, July 2005.

[14] S. Rao, T. K. Sarkar, and R. F. Harrington, "The electrostatic field of conducting bodies in multiple dielectric media," *IEEE Transaction on Microwave Theory and Techniques*, vol. 32, no. 11, pp. 1441–1448, November 1984.

[15] K. A. Michalski and J. R. Mosig, "Multilayered media green's functions in integral equation formulations," *IEEE Transaction on Antennas and Propagation*, vol. 45, no. 3, pp. 508–519, March 1997.

[16] R. Crampagne, M. Ahmadpanah, and J. Guiraud, "A simple method for determing the green's function for a large class of mic lines having multilayered dielectric structures," *IEEE Transaction on Microwave Theory and Technique*, vol. 26, no. 2, pp. 82–87, February 1978.

[17] RLE Computational Prototyping Group at Massachusetts Institute of Technology, "FastCap. [Online]," Available: http://www.rle.mit.edu/cpg/research_codes.htm, 1992.

[18] P. Zarkesh-Ha and J. D. Meindl, "Optimum chip clock distribution networks," in *IEEE International Interconnect Technology Conference*, May 1999, pp. 18–20.

[19] V. Mehrotra, S. L. Sam, D. Boning, A. Chandrakasan, R. Vallishayee, and S. Nassif, "A methodology for modeling the effects of systematic within-die interconnect and device variation on circuit performance," in *Proceedings of the 37th annual Design Automation Conference*, June 2000, pp. 172–175.

[20] V. Mehrotra and D. Boning, "Technology scaling impact of variation on clock skew and interconnect delay," in *IEEE International Interconnect Technology Conference*, June 2001, pp. 122–124.

[21] Y. Liu, S. Nassif, L. T. Pileggi, and A. J. Strojwas, "Impact of interconnect

variations on the clock skew of a gigahertz microprocessor," in *Proceedings of the 37th annual Design Automation Conference*, June 2000, pp. 168–171.

[22] K. Nabors and J. White, "FastCap: A multipole accelereated 3-D capacitance extraction program," *IEEE Transcations on Computer-Aided Design of Integrated Circuits and Systems*, vol. 10, no. 11, pp. 1447 – 1459, November 1991.

[23] S. P. McCormick, "EXCL: A circuit extractor for IC designs," in *Proceedings of the 21st annual Design Automation Conference*, June 1984, pp. 616–623.

[24] M. Kamon, M. J. Tsuk, and J. White, "FastHenry: A multipole-accelerated 3-D inductance extraction program," *IEEE Transaction on Microwave Theory and Techniques*, vol. 42, no. 9, pp. 1750–1758, September 1994.

[25] RLE Computational Prototyping Group at Massachusetts Institute of Technology, "FastHenry. [Online]," Available: http://www.rle.mit.edu/cpg/research_codes.htm, 1996.

[26] A. B. Kahng and Y. C. Pati, "Subwavelength optical lithography: Challenges and impact on physical design," in *Proceedings of the 1999 International Symposium on Physical Design*, New York, NY, USA, April 1999, pp. 112–119, ACM.

[27] K. Agarwal and S. Nassif, "The impact of random device variation on SRAM cell stability in sub-90-nm CMOS technologies," *IEEE Transcations on Very Large Scale Integration Systems*, vol. 16, no. 1, pp. 86–97, January 2008.

[28] R. N. Kanj, R. Joshi, and S. Nassif, "SRAM yield sensitivity to supply voltage fluctuations and its implications on Vmin," in *IEEE International Conference on Integrated Circuit Design and Technology*, May 2007, pp. 269–272.

[29] J. Pille, C. Admas, T. Christensen, S. Cottier, S. Ehrenreich, T. Kono, D. Nelson, O. Takahashi, S. Tokito, O. Torreiter, O. Wagner, and D. Wendel, "Implementation of the CELL broadband engine in a 65nm soi technology featuring dual-supply SRAM arrays supporting 6GHz at 1.3V," in *IEEE International Solid-state Circuits Conference*, June 2007, pp. 322–323.

[30] Y. Zhou, Z. Li, Y. Tian, W. Shi, and F. Liu, "A new methodology for interconnect parasitics extraction considering photo-lithography effects," in *Proceedings of the 2007 Asia and South Pacific Design Automation Conference*, January 2007, pp. 450–455.

[31] Nanoscale Integration and Modeling (NIMO) Group at ASU, "Predictive technology models. [Online]," Available: http:://www.eas.asu.edu/~ptm, 2009.

[32] M. Pelgrom, A. Duinmaijer, and A. Welbers, "Matching propertities of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1440, October 1989.

[33] K. Lakshmikumar, R. Hadaway, and M. A. Copeland, "Characterization and modeling of mismatch in MOS transistors for precision analog design," *Journal of Solid-State Circuits*, vol. 21, no. 6, pp. 1057–1066, February 1986.

[34] T. Mizuno, J. Okamura, and A. Toriumi, "Experimental study of threshold voltage fluctuation due to statistical variation of channel dopant number in MOSFETs," *IEEE Transaction on Electron Devices*, vol. 41, no. 11, pp. 2216–2221, Novermber 1994.

[35] R. V. Joshi, S. Mukhopadhyay, D. W. Plass, Y. H. Chan, C. Chuang, and A. Devgan, "Variability analysis for sub-100 nm PD/SOI CMOS SRAM cell," in *Pro-*

ceeding of the 30th European Solid-State Circuits Conference, September 2004, pp. 211–214.

[36] T. I. Kirkpatrick and N. R. Clark, "PERT as an aid to logic design," *IBM Journal of Research and Development*, vol. 10, no. 2, pp. 135–141, Mar 1966.

[37] N. H. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, VLSI Systems. Addison-Wesley, Reading, Massachusetts, USA, 2nd edition, 1993.

[38] L. W. Nagel, "Spice2: A computer program to simulate semiconductor circuits," Ph.D. dissertation, University of California, Berkeley, California, 1975.

[39] R. Macys and S. McCormick, "A new algorithm for computing the "effective capacitance" in deep sub-micron circuits," in *Proceedings of the IEEE Custom Integrated Circuits Conference*, May 1998, pp. 313–316.

[40] S. Mei, J. Kawa, C. Chiang, and Y. I. Ismail, "An accurate low iteration algorithm for effective capacitance computation," in *IEEE International Workshop on System-on-Chip for Real-Time Applications*, July 2004, pp. 99–104.

[41] A. B. Kahng and S. Muddu, "New efficient algorithms for computing effective capacitance," in *Proceedings of the 1998 International Symposium on Physical Design*, April 1998, pp. 147–151.

[42] A. B. Kahng and S. Muddu, "Improved effective capacitance computations for use in logic and layout optimization," in *Proceedings of the 12th International Conference on VLSI Design - 'VLSI for the Information Appliance'*, January 1999, pp. 578–582.

[43] S. Nassif and Z. Li, "A more effective C_{eff}," in *IEEE International Symposium on Quality Electronic Design*, March 2005, pp. 648–653.

[44] B. N. Sheehan, "Osculating thevenin model for predicting delay and slew of capacitively characterized cells," in *Proceedings of the 39th annual Design Automation Conference*, June 2002, pp. 866–869.

[45] C. V. Kashyap, C. J. Alpert, F. Liu, and A Devgan, "Closed-form expressions for extending step delay and slew metric to ramp inputs for RC trees," *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23, no. 4, pp. 509–516, April 2004.

[46] J. M. Wang, J. Li, S. Yanamanamanda, L. K. Vakati, and K. K. Muchherla, "Modeling the driver load in the presence of process variations," *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 10, pp. 2264–2275, October 2006.

[47] P. R. O'Brien and L. T. Savarino, "Modeling of driving point characteristic of resistive interconnect for accurate delay estimation," in *International Conference on Computer-Aided Design*, November 1989, pp. 512–515.

[48] A. B. Kahng and S. Muddu, "Efficient gate delay modeling for large interconnect loads," in *IEEE Proceedings of Multi-Chip Module Conference*, February 1996, pp. 202–207.

[49] P. Antognetti and C. Massobro, *Semiconductor Device Mdeling with SPICE*, McGraw-Hill, Columbus, Ohio, USA, 2nd edition, 1993.

[50] J. W. Harris and H. Stocker, *Handbook of Mathematics and Computational Science*, Springer, New York, NY, USA, 1st edition, 1998.

[51] L. Trevillyan, D. Kung, R. Puri, L. N. Reddy, and M. A. Kazda, "An integrated environment for technology closure of deep-submicron IC designs," *IEEE Design*

and Test of Computer, vol. 21, no. 1, pp. 14–22, Januaray-February 2004.

[52] L. P. P. P. VAN GINNEKEN, “Buffer placement in distributed RC-tree networks for minimal elmore delay,” in *Proceedings of the International Symposium on Circuits and Systems*, May 1990, pp. 865–868.

[53] W. Shi and Z. Li, “A fast algorithm for optimal buffer insertion,” *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 879– 891, June 2005.

[54] J. Lillis, C. Cheng, and T. Y. Lin, “Optimal wire sizing and buffer insertion for low power and a generalized delay model,” in *Proceedings of the 1995 IEEE/ACM International Conference on Computer-Aided Design*, December 1995, pp. 138–143.

[55] Z. Li, C. N. Sze, C. J. Alpert, J. Hu, and W. Shi, “Making fast buffer insertion even faster via approximation techniques,” in *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, January 2005, pp. 13–18.

[56] S. Hu, C. J. Alpert, J. Hu, S. Karandikar, Z. Li, W. Shi, and C. N. Sze, “Fast algorithms for slew constrained minimum cost buffering,” in *Proceedings of the 43rd annual Design Automation Conference*, July 2006, pp. 308–313.

[57] C. Chu and D. F. Wong, “Closed form solutions to simultaneous buffer insertion/sizing and wire sizing,” *Transactions on Design Automation of Electronic Systems*, vol. 6, no. 3, pp. 343–371, July 2001.

[58] C. Chen, C. Chu, and D. F. Wong, “Fast and exact simultaneous gate and wire sizing by lagrangian relaxation,” in *Proceedings of the 1998 IEEE/ACM*

International Conference on Computer-Aided Design, February 1998, pp. 617–624.

[59] S. Hu, M. Ketkar, and J. Hu, “Gate sizing for cell library-based designs,” in *Proceedings of the 44th annual Design Automation Conference*, June 2007, pp. 847–852.

[60] Y. Liu and J. Hu, “A new algorithm for simultaneous gate sizing and threshold voltage assignment,” in *Proceedings of the 2009 International Symposium on Physical Design*, March 2009, pp. 27–34.

[61] C. J. Alpert, S. K. Karandikar, Z. Li, G. Nam, S.T. Quay, H. Ren, C. N. Sze, P. G. Villarrubia, and M. C. Yildiz, “Techniques for fast physical synthesis,” *Proceedings of the IEEE*, vol. 95, no. 3, pp. 573–599, March 2007.

[62] S. K. Karandikar, C. J. Alpert, M. C. Yildiz, P. Villarrubia, S. Quay, and T. Mahmud, “Fast electrical correction using resizing and buffering,” in *Proceedings of the 2007 Asia and South Pacific Design Automation Conference*, January 2007, pp. 553–558.

[63] Z. Li, C. J. Alpert, S. Hu, T. Muhmud, S. T. Quay, and P. G. Villarrubia, “Fast interconnect synthesis with layer assignment,” in *Proceedings of the 2008 International Symposium on Physical Design*, April 2008, pp. 71–77.

VITA

Ying Zhou received B.S. and M.S. degrees in electrical engineering from Xi'an Jiaotong University, China, in 1998 and 2001 respectively. She enrolled in Texas A&M University in the Fall of 2003, majoring in computer engineering. In January 2008, she did an internship at the IBM Austin Research Laboratory. Her research interests mainly focus on parasitic extraction, circuit modeling and simulation, and physical synthesis, especially on gate sizing.

She received the ASPDAC Best Paper Award in 2007. During her internship at IBM, she received the IBM Research Outstanding Technical Accomplishment Award in 2008, as a team member for the research project on "a next generation design closure flow for ASICs and CDP". The outstanding Technical Accomplishment Award is highly competitive since it needs to show more than $100 Million revenue impact to the IBM business. She also received the IBM Technical Accomplishment Award in 2009 as a team member for the research project on "ASIC design win".

She may be contacted by email at nancyzhou8@gmail.com and by mail at the following address:

Blg 904, 11501 Burnet Road

Austin, Texas, 78758

The typist for this dissertation was Ying Zhou.