ADVANCED ANALOG LAYOUT DESIGN AUTOMATION IN COMPLIANCE WITH DENSITY UNIFORMITY

by

©Gholamreza Shomalnasab

A thesis submitted to the School of Graduate Studies

in partial fulfillment of the requirements for the

Degree of Doctor of Philosophy

Faculty of Engineering of Applied Science

Memorial University of Newfoundland

October, 2017

St. John's Newfoundland and Labrador Canada

ABSTRACT

To fabricate a reliable integrated circuit chip, foundries follow specific design rules and layout processing techniques. One of the parameters, which affect circuit performance and final electronic product quality, is the variation of thickness for each semiconductor layer within the fabricated chips. The thickness is closely dependent on the density of geometric features on that layer. Therefore, to ensure consistent thickness, foundries normally have to seriously control distribution of the feature density on each layer by using post-processing operations.

In this research, the methods of controlling feature density distribution on different layers of an analog layout during the process of layout migration from an old technology to a new one or updated design specifications in the same technology have been investigated. We aim to achieve density-uniformity-aware layout retargeting for facilitating manufacturing process in the advanced technologies. This can offer an advantage right to the design stage for the designers to evaluate the effects of applying density uniformity to their drafted layouts, which are otherwise usually done by the foundries at the final manufacturing stage without considering circuit performance. Layout modification for density uniformity includes component position change and size modification, which may induce crosstalk noise caused by extra parasitic capacitance. To effectively control this effect, we have also investigated and proposed a simple yet accurate analytic method to model the parasitic capacitance on multi-layer VLSI chips. Supported by this capacitance modeling research, a unique methodology to deal with

density-uniformity-aware analog layout retargeting with the capability of parasitic capacitance control has been presented. The proposed operations include layout geometry position rearrangement, interconnect size modification, and extra dummy fill insertion for enhancing layout density uniformity. All of these operations are holistically coordinated by a linear programming optimization scheme. The experimental results demonstrate the efficacy of the proposed methodology compared to the popular digital solutions in terms of minimum density variation and acute parasitic capacitance control.

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere appreciation to my advisor, Dr. Lihong Zhang, for his guidance, inspiration, suggestions, criticism and financial support that he has provided throughout the course of my research and studies. He always encourages me to conduct innovative research and trains me to be an independent researcher. Special thanks are also extended to Dr. Howard Heys for his time and support as my co-supervisor for the first year of my PhD program and his instruction as one member on my Supervisory Committee. I also thank Dr. Ramachandra Venkatesan for taking the time to review this dissertation and serving on the Supervisory Committee.

I also gratefully acknowledge the Faculty of Engineering and Applied Science, the Memorial University of Newfoundland, Natural Sciences and Engineering Research Council of Canada (NSERC), Canada Foundation for Innovation (CFI), Research and Development Corporation (RDC) of Newfoundland and Labrador (through its Industrial Research and Innovation Fund and ArcticTECH R&D Award), for the financial support.

My appreciations also go to the entire group members of the CADLAMS and my colleagues: Xuan Dong, Tuotian Liao, Mohammad Torabi for their friendship and assistance in the course of my research programme. Additional gratitude is given to my friends Remzi Cej and Mohammad Hekmatnejad which have been a great part of my life at Memorial University.

Finally, I would like to express my special appreciation to my parents, Faramarz Shomalnasab and Sedigheh Tohidi, and my sisters for their continued support and love

which support me to pursue my ambitions. The deepest thanks are expressed to my lovely wife, Khoshrooz Kazemi, for her endless love, inspiration, encouragement and patience.

TABLE OF CONTENTS

ABSTRACT .... **II**

ACKNOWLEDGEMENT .... **IV**

TABLE OF CONTENTS .... **VI**

LIST OF FIGURES .... **IX**

LIST OF TABLES .... **XII**

LIST OF SYMBOLS AND ABBREVIATIONS .... **XIII**

CHAPTER 1 INTRODUCTION **1**

1.1 Background **1**

1.2 Statement of Problems **3**

1.3 Research Objectives **4**

1.4 Structure of the Thesis **5**

CHAPTER 2 LITERATURE REVIEW **7**

2.1 Introduction **7**

2.2 Capacitance Modeling **7**

2.3 Layout Synthesis and Retargeting **14**

2.4 Density Analysis and Fill Synthesis **18**

2.5 Summary **26**

CHAPTER 3 ANALATIC MODELING OF PARASITIC CAPACITANCE **27**

3.1 Introduction **27**

3.2 Preliminaries of Capacitance Modeling **27**

3.3 Modeling of Fringe Capacitance **29**

3.3.1 Substrate Capacitance 30

3.3.2 Coupling Capacitance 35

3.3.3 Lateral Capacitance 39

3.3.4 Linear Model 41

3.3.5 Curve-Fitting Technique 42

3.4 Summary 44

CHAPTER 4 DENSITY-UNIFORMITY-AWARE ANALOG LAYOUT RETARGETING 45

4.1 Introduction 45

4.2 Regular Retargeting Process 46

4.3 Density-Uniformity-Aware Analog Layout Retargeting Process Flow......52

4.4 Linear-Programming Formulation and Optimization 56

4.4.1 Density Analysis 56

4.4.2 Problem Formulation 58

4.5 Summary 64

CHAPTER 5 DENSITY-UNIFORMITY OPTIMIZATION AND CONTROL ALGORITHM...65

5.1 Introduction 65

5.2 Interconnect Widening Process 66

5.2.1 Smart Splitting 70

5.3 Area Shift Process 74

5.4 Parasitic Capacitance Control during Layout Retargeting 78

5.5 Summary 81

CHAPTER 6 EXPERIMENTAL RESULTS 82

6.1 Introduction 82

6.2 Verification of Capacitance Modeling 83

6.2.1 Substrate Capacitance 84

6.2.2 Coupling Capacitance 87

6.2.3 Lateral Capacitance..97

6.2.4 Analysis of Computation Time..100

6.2.5 Case Study..103

6.2.6 Extension to More Advanced Technologies 107

6.3 Experimental Result of Density-Uniformity-Aware Analog Layout Retargeting 108

6.3.1 Experimental Comparison of Various Density Control Schemes 108

6.3.2 Density-Uniformity-Aware Analog Layout Retargeting Verification.....111

6.4 Summary 128

CHAPTER 7 CONCLUSIONS AND RECOMMENDATIONS 129

7.1 Summary 129

7.2 Research Contributions 130

7.3 Recommendations for Future Research 133

REFERENCES 134

APPENDIX 142

Candidate’s Publication List 142

LIST OF FIGURES

Figure 1 – Fringe capacitance contribution with different sizes in 3 different technologies. 9

Figure 2 – Relationship between layer thickness and local density [48]....20

Figure 3 – An $l \times l$ layout is partitioned into smaller cells, each of which has a size of $(w/r) \times (w/r)$. Each $w \times w$ partition-window (light gray) consists of $r \times r$ partition-cells. A pair of partition-windows from different dissections may overlap with each other. 21

Figure 4 – Generating Hanan grid over a layout [48] 22

Figure 5 – Fringe capacitances (a) sidewall to sidewall fringe in the case where two metal blocks (M1 and M2) exactly overlap, (b) sidewall to top surface fringe capacitance when two metal blocks (M1 and M2) have overlap and displacement, (c) fringe capacitance between sidewall of top metal block (M2) and top surface of bottom metal block (M1). 29

Figure 6 – Electrostatic field simulation shows electric field between the dashed lines can be approximated by the circular lines. 31

Figure 7 – Splitting fringe capacitance of one side to several sub-components. 31

Figure 8 – Calculating C_2 in a circular electric field. t is the thickness of Metal-1 layer and d is the vertical distance between Metal-1 layer and substrate....32

Figure 9 – Actual charge density (shown in colorful density) reduces as the distance from origin O increases. 33

Figure 10 – Choosing a proper factor to calculate the fringe capacitance based on the displacement between *M1* and *M2*. 37

Figure 11 – Modeling details of fringe capacitance in Figure 5(c). 39

Figure 12 – Lateral capacitance with its different components. (a) Two metal blocks are in a line. (b) Two metal blocks have displacement. 41

Figure 13 – Linearization of $\ln(1+x)$ to 0.75 x in range of $0.65<x<1.2$ 42

Figure 14 – A mask layer representation in corner-stitching data structure [40]....47

Figure 15 – Flow diagram of the regular retargeting platform 47

Figure 16 – An exemplary horizontal constraint graph including a sub-graph representing the placement of three solid tiles. 49

Figure 17 – Flow diagram of the proposed density uniformity-aware layout retargeting methodology. 54

Figure 18 – Pseudo code of the interconnect widening operation 67

Figure 19 – Example of splitting a tile into smaller tiles and the corresponding constraint graphs in (a) horizontal and (c) vertical directions, (c) for split tiles and (d) for the relaxed vertical constraint graph....72

Figure 20 – Updating connectivity constraints after splitting the shared tiles....73

Figure 21 – Pseudo code of the area shift operation....75

Figure 22 – Area shift variables for the maximum or minimum partition-window in the local area shift scheme....77

Figure 23 – parasitic capacitance components between two adjacent nets. 80

Figure 24 – Substrate capacitance error comparison for 3 different technologies....86

Figure 25 – Several examples of coupling capacitance test cases (blue squares with orthogonal shadowed lines are on Metal-1 layer and green ones with vertical shadowed lines are on Metal-2 layer). 88

Figure 26 – Curve-fitting diagram of four samples to tune (11) for CMOS 90 nm technology....90

Figure 27 – Curve-fitting diagram of four samples to tune (16) for CMOS 90 nm technology....90

Figure 28 – Curve-fitting diagram of four samples to tune (11) for CMOS 65 nm technology....92

Figure 29 – Curve-fitting diagram of four samples (with geometry size of 10µm*10µm) to tune (16) for CMOS 65 nm technology....92

Figure 30 – Histogram of error occurrences in the Monte-Carlo verification for coupling capacitance in the CMOS 65 nm technology....97

Figure 31 – Two examples of lateral capacitance test case (top view). 99

Figure 32 – Case study of the capacitance modeling 105

Figure 33 – Different test cases to investigate the effects of dummy fill insertion and interconnect widening on the laterally-coupling capacitance. (a) The reference test case, (b) dummy scheme1 test case, (c) dummy scheme 2 test case, (d) interconnect widening test case....110

Figure 34 – Two-stage Opamp schematic (top) and folded-cascode Opamp schematic (middle), and 5-bit ADC analog comparator (bottom)....113

Figure 35 – Two-stage Opamp original layout (top) and folded-cascode Opamp original layout (bottom)....114

Figure 36 – the analog part of the 5-bit ADC layout....115

Figure 37 – Two-stage Opamp layouts before (top) and after (bottom) the AS+IW+DF operations in CMOS 180 nm technology. The splitting and widening operation is also shown in the red ellipse. 125

Figure 38 – Folded-cascode Opamp layouts before (top) and after (bottom) the AS+IW+DF operations in CMOS 180 nm technology. 126

Figure 39 – A small portion of the analog part of the 5-bit ADC layout (Metal 1) after modification by AS+IW+DF 127

LIST OF TABLES

Table 1 – Average of fringe contribution in substrate capacitance 84

Table 2 – Proper values obtained from the curve-fitting process 85

Table 3 – Comparison of coupling capacitance. All numbers represent error percentage. (*NA* means not applicable, and *NF* means not feasible) 93

Table 4 – Statistical results from the Monte Carlo verification for overlapping coupling capacitance in the CMOS 65 nm technology. 96

Table 5 – Lateral capacitance effective width (w_e) expressions and corresponding error estimations (R-squared) for different technologies (*dx* is the displacement in the corresponding direction) 98

Table 6 – Comparison of lateral capacitance. All numbers represent error percentage. (*NA* means not applicable, and *NF* means not feasible) 100

Table 7 – Computation time comparison 101

Table 8 – Complexity comparison among the methods that are able to handle fringe capacitance (*d* stands for the distance). 103

Table 9 – Comparison of substrate and coupling/lateral capacitance in the case study ..106

Table 10 – Laterally-Coupling Capacitance (top) and substrate capacitance (bottom) increase comparison for test cases of Figure 32 (b)-(d) in reference to the original test case of Figure 32 (a) 111

Table 11 – Comparison of density variation of two test layouts for CMOS 180 nm technology 117

Table 12 – Comparison of density variation of two test layouts for CMOS 90 nm technology 118

Table 13 – Comparison of density variation of two test layouts for CMOS 65 nm technology 119

Table 14 – Comparison of density variation of 5-bit ADC layouts for CMOS 180 nm technology 119

Table 15 – Performance comparison of two test layouts in CMOS 180 nm technology 124

Table 16 – Performance comparison of 5-bit ADC 124

LIST OF SYMBOLS AND ABBREVIATIONS

ADC	Analog to Digital Convertor
ALG	Analog Layout Generator
AS	Area Shift
BEM	Boundary Element Method
CAD	Computer-Aided Design
CBCM	Charge-Based Capacitance Measurement
CDT	Constrained Delaunay Triangulation
CG	Constraint Graph
CLP	Covering Linear Programming
CMOS	Complementary Metal-Oxide-Semiconductor
CMP	Chemical-Mechanical Polishing
CPU	Central Processing Unit
DF	Dummy Fill
DFM	Design for Manufacturability
EMCF	Electro-Magnetic Curve Fitting
FEM	Finite Element Method
GHZ	Giga Hertz

GP	Geometric Programming
IC	Integrated Circuit
IMC	Iterated Monte-Carlo
IPRAIL	Intellectual Property Reuse-based Analog IC Layout
IW	Interconnect Widening
LP	Linear Programming
MC	Monte-Carlo
MIN-FILL	Minimum Fill
MIN-VAR	Minimum Variation
MOSFET	Metal-Oxide-Semiconductor Field Effect Transistor
NA	Not Applicable
NF	Not Feasible
RF	Radio Frequency
TSV	Through-Silicon-Via
VLSI	Very Large Scale Integration

CHAPTER 1

INTRODUCTION

1.1 Background

Competitive consumer electronics demands high-end technologies to better serve the daily life of human beings. Fast growing integrated circuits play a significant role in making various modern technologies (such as communications and multimedia) both powerful and immersive, as demonstrated by the widely affordable mobile devices in use by the general public. Unprecedented innovation in microelectronics chip manufacturing technologies calls for redesigning of the entire chip, which was functional in the outdated technologies, for newly emerging more advanced technologies. As the chip redesigning is a time-consuming process, computer-aided design (CAD) tools are introduced to help designers to complete this procedure more effectively and efficiently [1].

There is a lot of work done on the CAD tools to support the redesigning of digital circuits. Nowadays, it is already possible to use hardware description languages and scalable technology libraries to automate the entire process of designing a complex digital system/circuit for any specific technology and/or migrating to a different technology. But in stark contrast, the situation for analog and radio-frequency (RF) integrated circuits is completely different due to the high sensitivity and nonlinearity nature of analog signals and devices. The CAD tools to retarget an existing analog design to a newer technology process or specification are under research and development.

Although some methods and corresponding CAD tools have been proposed in the literature, they are far from maturity as their digital counterpart.

Chemical Mechanical Polishing (CMP) is a manufacturing process, which uses both chemical and mechanical means to make the surface of wafer planar [2]. As a normal practice in the very-large-scale-integrated (VLSI) chip fabrication, CMP has been widely utilized to satisfy the local and global planarity requirements imposed by the advanced photolithography methods [3]. To reliably fabricate multi-layer VLSI chips with high yield, the deposition density of each layer occupied by any geometric blocks should be maintained uniform in order for the CMP process to work better [4]. Therefore, foundries usually modify the layout on the wafer at the back end by inserting dummy features, which do not have any logic functionality but only change the layer density. Obviously, the signal and functional integrity of the circuits (especially sensitive analog and RF blocks) originally well done by the designers might not be guaranteed during the fabrication.

Depending on the local density of the wafer, CMP may cause unwanted effects such as erosion and dishing. This can cause variation in thickness of interlayer dielectric, which may change interconnect properties (e.g., capacitance and resistance) and lead to timing uncertainty as well as other phenomena. To improve layer property and enhance reliability in the IC manufacturing process, the global material deposition within a layout must be uniformly distributed [5]. One solution is to use dummy fill insertion to improve pattern density [6]. Dummy fill blocks are non-functional features inserted into sparse

area of a layout to increase local density in that region. However, they may increase parasitic capacitance between interconnects and thus affect circuit performance [7].

1.2 Statement of Problems

1) Simple yet accurate model of parasitic capacitance

Based on the introduction above, accurate modeling of parasitic capacitance in VLSI chips is critical if one needs to understand and control the dummy fill insertion. Accurate modeling usually comes along with complex formulation and expensive computational process. Therefore, the demand for simple and accurate modeling is quite obvious. Researchers always have to make a trade-off between accuracy and simplicity for an analytic model based on the applications. For the optimization algorithms, a model to be utilized is preferred to be simpler yet accurate.

2) Design for manufacturability of analog layouts

Integration of manufacturability considerations into the design stage of chip layouts will increase the final product quality and yield. In the traditional process of designing circuits and their layouts, design for manufacturability may not be considered seriously. Therefore, manufacturability issues have to be addressed by the foundries in the fabrication stage. This may unfortunately alter and degrade circuit performance.

3) Improving analog layout retargeting process

Layout generation or layout retargeting may be performed automatically by computer-aided design tools. A vast number of commercial tools have been introduced

for synthesizing digital layouts, whereas very limited promising tools are available for the analog counterparts. The area of automatic analog layout generation and optimization is still under development.

To the best of our knowledge, none of the current commercial automated analog layout retargeting tools can deal with the density uniformity challenges with a dedicated method that is tailored for analog layouts. Instead they have to solely rely on the solutions developed for general digital layouts. It has been well recognized that analog layouts, which look much sparser, are largely different from digital ones due to complex analog constraints. Moreover, the capability of density uniformity is especially beneficial to the analog layout retargeting tools, which typically convert layouts from old technologies to advanced ones. In the old technologies, density rules are normally not critical or even not required, whereas they become mandatory for the advanced technologies. Therefore, the analog layout retargeting tools demand the functionality of density uniformity as one prerequisite for them to be accepted by the analog designers in the era of advanced technologies.

1.3 Research Objectives

Although the sole solution for digital layouts by means of dummy fill insertion has been already studied in the previous publications, density uniformity optimization has not yet been seriously addressed in the automated analog layout generation context. The main objective in this thesis work is to investigate a new layout retargeting methodology to generate a layout with uniform density distribution. The proposed layout retargeting

methodology can automatically produce an output layout following the same geometric floorplan of the input layout. To facilitate complex iterative loops in the optimization, a simple yet accurate model to include the parasitic capacitance into constraints is needed.

Towards these objectives, this PhD research has been conducted to gain the following achievements. A parasitic capacitance modeling methodology for a multi-layer VLSI chip has been developed. Then a linear-programming formulation and optimization method has been devised to globally schedule the resource allocation for various means to meet the density uniformity goals. Finally, several special functional features including layout density control, geometry modification, and dummy fill insertion have been integrated to form the proposed density-uniformity-aware analog layout retargeting flow. Our experiments show that all the features above are properly combined together to effectively contribute to optimum output layouts in compliance with density uniformity without compromising electrical performance.

1.4 Structure of the Thesis

This thesis is organized as follows. In Chapter 2, a comprehensive literature review, including capacitance modeling, density analysis, automated layout generation, and layout retargeting, is conducted. In Chapter 3, the proposed modeling of parasitic capacitance inside VLSI chips is discussed. Chapter 4 describes the second phase of this PhD research, that is, the control and optimization scheme of the proposed density-uniformity-aware analog layout retargeting process. The implementation details of the proposed methodology are explained in Chapter 5. Chapter 6 is dedicated to the

experimental results and verification of the proposed methodology. Chapter 7 concludes this study with some contributions listed and provides some recommendations for future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, the previous work on the major aspects of this dissertation research will be reviewed. Parasitic capacitance modeling approaches are surveyed in Section 2.2. Then the work on automatic layout synthesis is reviewed in Section 2.3. In Section 2.4, the previous work on layout density analysis and dummy fill synthesis is studied. Finally, a brief summary is presented in Section 2.5.

2.2 Capacitance Modeling

To ensure sufficient reliability margin, it is indispensable to effectively evaluate performance of laid-out circuitry before chip fabrication. Interconnect parasitic capacitance is one of the major contributors to performance degradation and signal integrity problems [8] [9]. As state-of-the-art technologies have multiple metal layers, calculation of parasitic capacitance between these layers is challenging due to the required accuracy and highly intensive computation effort. Unlike commercial tools that typically use empirical equations specialized for a particular technology, a general reliable modeling methodology for deriving simply expressed yet accurate reusable equations is strongly demanded [10].

Parasitic capacitances can make severe impact on circuit performance if they are not correctly modeled and taken into account in the circuitry design. They may lead to performance degradation due to the issues of signal integrity and crosstalk noise. In particular, capacitance modeling in the context of complex geometry of leading-edge technologies is even more difficult [11]. Without simply expressed analytic models, extraction of a complicated layout with thousands even millions of interconnects would be extremely time-consuming and thus impractical for circuitry optimization purpose [12]. Fringe capacitance, which is normally formed by sidewalls of interconnect blocks, has an increasingly important contribution in the total parasitic capacitance in advanced technologies. Due to small absolute values of fringe capacitances, they used to be safely ignored for the old technologies. However, for the modern deep-submicron or nanometer technologies, smaller geometries, which usually generate smaller total parasitic capacitance, are widely used. Thus, the impact of fringe capacitance cannot be overlooked any longer. Moreover, according to our experiments, the contribution of the fringe component within overall capacitance increases if the size of the related geometry is shrinking (as shown in Figure 1). For example, in CMOS 90 nm technology, the fringe contribution within the entire substrate capacitance[1] increases from 9% for 40 µm-size geometry to 28% for 10 µm-size one and even more for smaller sizes. Therefore, accurate

[1] Substrate capacitance is a capacitance formed between any geometry pattern on one layer and the substrate ground plane inside the chip.

modeling of the fringe capacitance is becoming more important in the advanced technologies.

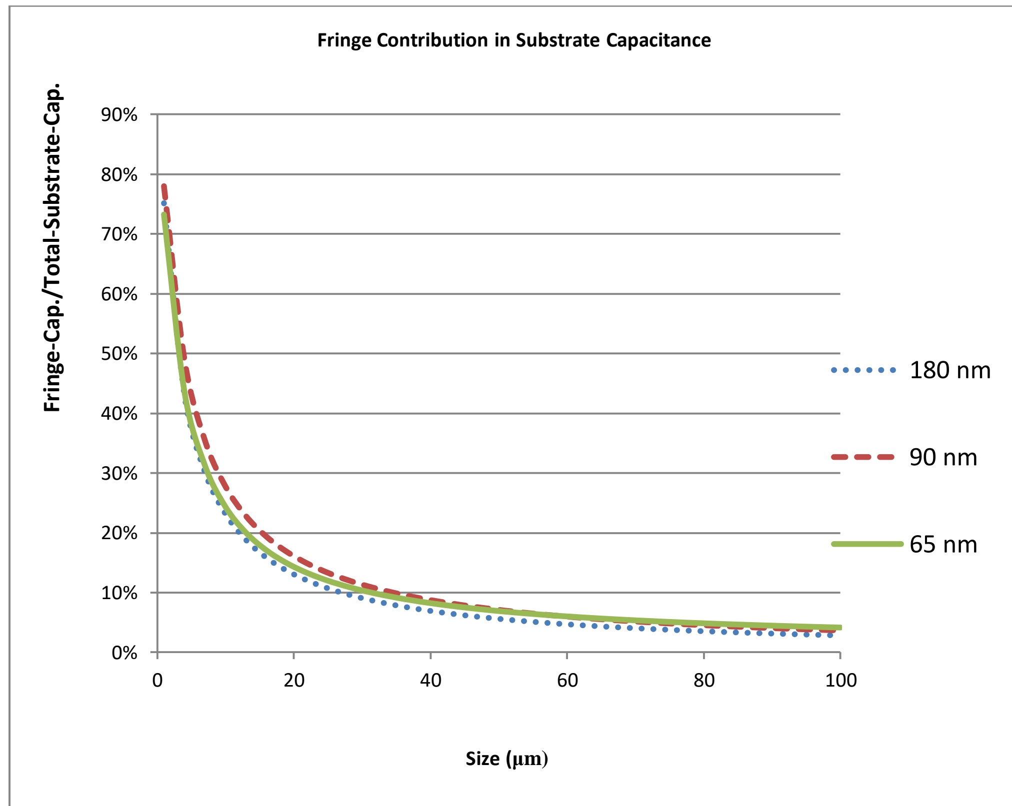


Figure 1 – Fringe capacitance contribution with different sizes in 3 different technologies.

Adding dummy fill is one of the Design-for-Manufacturability (DFM) attempts to meet density requirements for high-yield chip fabrication [3]. Although inserting dummy fill can help make uniform layers, this practice may degrade the performance of sensitive analog circuitry or high-speed digital blocks due to added parasitic capacitance. Therefore, accurate modeling of dummy fill impact on the parasitic capacitance is crucial [13] [14]. For this reason, this dissertation research has been first focused on the modeling study of parasitic capacitance in the advanced technologies.

Computer-Aided Design tools using optimization loops can optimally add dummy fill to a chip layout while minimizing negative impact of dummy blocks on circuit performance. However, adding thousands or millions of dummy fill to a well-designed layout makes it difficult for CAD tools to invoke accurate commercial capacitance extractors to estimate a huge number of capacitors and associated performance degradation within loops of optimization. Instead, traditional rough capacitance estimation equations have to be widely used in the state-of-the-art optimization work [3] [15]. This is just a vivid example that exhibits the necessity of this research. Here the motivation is to investigate simple yet accurate analytic parasitic capacitance models, which can be used in the CAD algorithms for loops of optimization.

The reported techniques for capacitance modeling can be classified into three categories: numerical methods, analytic methods, and test-based or empirical methods. Finite Element Method (FEM), Boundary Element Method (BEM), and multipole algorithms [16] are some of the available numerical methods. These methods solve electromagnetic equations and usually provide accurate results. However, they are typically computationally expensive when applied in the modern VLSI circuits [17] [18]. Thus, they are normally deployed for the verification purpose before tape-out rather than being used as an estimator within loops of optimization.

As an alternative, analytic methods use approximate equations derived from electromagnetic or mathematical models. Since the computation of overlap capacitance is easy to handle, [11] [19] focus on different analytic models, which are used to account for fringe capacitance between interconnects by using conformal mapping. These

methods try to model electric flux between plates by mapping an elliptical shape to a rectangle. Due to intensive mathematical computation, the developed models are not recommended for extracting the layout with a great number of interconnects although accuracy of the models is acceptable.

Bueno-Barrachina *et al.* [20] presented electrostatic field equations to calculate capacitance between inclined plates. FEM software like ANSYS was used to solve electrostatic equations and plot them to validate their assumption. The authors verified their equations with the geometries much larger than the feature size of modern submicron or nanometer VLSI circuits. Although their model was specialized for a certain case (i.e., metal blocks with the same size), our experiments show that such a model can still be used for smaller feature size with reasonably good accuracy. Samudra and Lee evaluated special interconnect geometry in their research [21]. As their analytic equations were derived for a specific shape of interconnects (trapezoid), such equations cannot be generalized to cover other interconnect shapes. Zhao *et al.* [22] proposed analytic modeling of interconnect capacitance based on electric filed distribution. They used several empirical factors in their model. Although the reported accuracy was promising, the authors did not report their model computation time. Moreover, in the analytic approaches above, the proposed equations require use of some technology-dependent factors, which are normally derived from experiments. However, the way how to obtain those factors is not clarified in the publications. This makes reproduction of the published methods, in particular for the modern technologies, extremely hard.

The third category is the empirical technique [23], where plenty of experimental data for a particular technology are collected and fitted to a certain model. In these methods some technology-dependent parameters have to be obtained for the best match. These methods are widely used in commercial tools where the capacitance extraction equations are different for special situations appearing in the layout. Using computationally intensive operators (e.g., exponentiation or logarithm) and many equations for one general scenario is deemed as the drawbacks of the empirical technique.

Barke [24] compared previous work on the estimation of line-to-ground capacitance, which are analytic equations combined with several empirical factors. Since this study was targeted at an old technology (where the author verified the equations with geometry sizes around 1 μm), several proposed equations function well for large geometries in the modern submicron technologies. However, these equations often fail to work well for smaller geometries. Tani *et al.* [18] also proposed approximation equations for parasitic capacitance extraction with promising results for some particular cases (e.g., two crossing metal lines). Their analytic method also includes several empirical factors that are hard to determine for new technologies. Moreover, estimation of coupling[2] capacitance for adjacent interconnects was postponed as their future work.

[2] Coupling capacitance is a capacitance formed between two parallel patterns belonging to different electrical nets inside a VLSI chip.

After comparing three numerical methods (FEM, BEM, and random walk method [25]) and analytic methods, Husain [26] suggested some guidelines to extract interconnect capacitances of a large layout. Several methods were also compared with each other along with the error sources analyzed for each of them. In conclusion, the author stressed that the numerical-method-based field solvers are too slow in extraction of parasitic capacitances for large layouts. Moreover, it was suggested that the existing analytic methods need to be modified for being used in the new technologies. In the work by Chang *et al.* [27], a Charge-Based Capacitance Measurement (CBCM) method was introduced to measure total capacitance of one interconnect. Using a test structure and a test signal, the authors could measure the capacitance of a pin. This is a general method, but out of our interest in this dissertation research.

Accurate modeling of parasitic capacitance has vast applications in VLSI industries. Hyun *et al.* [28] used a model similar to the one proposed in [11] for Through-Silicon-Via (TSV) coupling capacitance. The average reported error is around 5-8%. The sidewall parasitic capacitance in nano-scale trench isolated MOSFETs [29] and gate capacitance in gate-all-around cylindrical silicon nanowire MOSFETs [30] were also modeled by using a conformal mapping method, which is similar to [11] and [23]. Furthermore, capacitance modeling is important in Chemical Mechanical Polishing and density analysis. Kahng and Samadi [3] reported the significance of accurate parasitic capacitance estimation in dummy fill insertion stage where imprecise filling process without considering effect of parasitic capacitance from the layout can dramatically degrade circuit performance.

2.3 Layout Synthesis and Retargeting

Automated analog layout generation has always been one of the appealing topics in the area of electronic design automation. Such efforts can be categorized into two different streams. In the first division, an analog layout can be generated from scratch by using sequential operations of device generation, placement and routing. The associated methods have been studied so well that a variety of academic and commercial tools are currently available for decent layout generation directly from analog schematic. However, different from the mature digital hardware description language and physical design flow, these methods are still unable to convert a well-designed layout to fit for updated specifications or new technologies. Therefore, the second research stream in this disseration, called layout retargeting, has been called upon to address this issue in the context of restless technology upgrade. Normally, a modern analog layout retargeting method uses a silicon-proven input layout as template to produce an optimal solution suitable for modified specifications or updated technologies with simultaneous placement and routing operations.

LAYLA [31] is a hardware description language to develop parameterized cell libraries for VLSI layouts mostly focused on digital circuits. Hong *et al.* [32] introduced a performance-driven analog layout compiler. Their proposed compiler includes device generation, placement, and routing with performance considerations in order to meet the design specifications. KOAN/ANAGRAM II [33] is a tool based on macrocell layout automation to produce a complete analog layout from scratch. It includes two main modules: KOAN is responsible for device generation and placement, and ANAGRAM II

handles interconnections among the devices. The analog layout synthesis is broken down into sequential steps. To improve this work, the authors pointed out that a good understanding and communication between placement and routing is necessary. Malavasi *et al.* [34] presented a methodology for automatic synthesis of a full custom IC layout with complex analog constraints. By using this approach, high-level specifications can be translated into low-level constraints, which can control the supporting tools for placement, routing and compaction. Although parasitic constraints are taken into account in the layout generation, the projected parasitic bounds may create an over-constrained problem, which renders the problem unsolvable.

Zhang *et al.* [35] proposed a layout synthesis flow for RF circuits to consider parasitic closure issues. Their method utilizes performance-driven placement and global routing along with device tuning. Since layout effects related to performance are only modeled in terms of net length, parasitic coupling intricacy in the advanced technologies may not be considered by this method. In [36] a layout synthesis method, ALADIN, was proposed with sequential steps of module generation, placement and routing. The third version of the Analog Layout Generator (ALG) [37] was improved by Yilmaz and Dundar. Similarly this tool is capable of module generation, placement and routing. It offers optimization-based and rule-based module generation. The generated modules are fed to a placer to finalize the module locations in the layout. Finally in the routing stage, all of the modules are electrically connected globally and locally. During these sequential steps, this tool utilizes optimization algorithms to minimize performance degradation caused by parasitic effects and process variation.

An important factor for layout automation is how to use expert design knowledge in the physical design. In the following layout retargeting works, the expert design knowledge is normally reflected from the old layout or input template. Castro-López *et al.* [38] presented their work of analog layout retargeting by creating an extensive hierarchical template from an input layout. By generating constraints from input layout and applying new design specifications plus objectives, an iterative simulation-based optimization algorithm can generate a resized layout. In contrast, the authors used slicing tree to represent a layout in the layout synthesis process [39]. Since both methods require a lot of simulation and computation effort in the iterations to reach final solutions, the efficiency of these algorithms needs to be improved.

Jangkrajarng *et al.* introduced a layout retargeting tool called Intellectual Property Reuse-based Analog IC Layout (IPRAIL), which uses a symbolic template to preserve input layout properties and eventually generate a layout that meets all new design rules and specifications [40]. The symbolic template introduced in their work is directed constraint graphs, which are optimized in the context of new design constraints to generate a retargeted layout fit for new technology processes and/or specifications. In [41], Zhang and Liu improved the retargeting process by considering performance bounds when handling the symbolic template. Circuit performance sensitivities are first identified and modeled, and then the corresponding parasitic bounds are added to the pool of constraints. Finally the optimization problem is resolved by using combination of a graph-based technique with mixed-integer non-linear programming. Although the basic design rules and parasitic effects have been well covered, some design-for-

manufacturability tactics emerging in the advanced technologies cannot be handled by these methods.

Wang *et al.* [42] used layout symbolic template and Geometric Programming (GP) in the process of layout regatgeting. The presented experiment shows some promising results compared to the previous works. Chin *et al.* [43] presents an analog layout prototyping method by reusing layout knowledge while preserving routing behavior. The layout routing information is extracted into a graph with Constrained Delaunay Triangulation (CDT) algorithm. Multiple placements of the modules are explored, while for each placement candidate, the graph containing routing information is updated and resolved to derive the interconnected layout. For any left-over unrouted nets, manual routing is needed at the end of the process. Finally, the layout is tuned on the basis of simulation results with certain aid of users. Although this method can reduce layout prototyping time, it is hard to be fully automated and the final result heavily relies on expert knowledge.

LAYGEN II [44] proposed by Martins *et al.* is a tool for automated analog IC layout generation based on input template and optimization techniques. As an input, the layout template information, which is used for guiding placement and routing, is converted to a B*-tree representation by a placer. Once the layout floorplan is identified, an optimization-based router completes the interconnection with an evolutionary algorithm as the optimization engine. Although effective, the applied optimization method may cause degradation in the computational efficiency for the layout with a large number of blocks. Furthermore, an automated analog IC design flow from circuit level

specifications to physical layout description is included in AIDA [45]. The fully automated circuit-level synthesis is done by a multi-objective multi-constraint optimization approach. Then a layout is generated considering design rule constraints from a sized circuit-level description and high level layout guidelines. In this work, LAYGEN II is used to generate and optimize the final layout.

Weng *et al.* in [46] proposed an analog layout migration methodology to effeciently generate multiple layouts while keeping similar or better circuit performance. Various placement constraints are extracted from the original layout and hierarchically stored into a topology slicing tree. Pan *et al.* [47] proposed a prototyping framework for analog layout migration with planar preservation. In their work, constrained Delaunay triangulation is used to extract placement and routing features from an input layout into a crossing graph, which can be migrated into multiple layouts with placement and routing reconnection.

2.4 Density Analysis and Fill Synthesis

All of the analog layout automation tools above strive to meet design specifications and technology design rules. To fabricate a chip in deep sub-micron or nanometer technologies, a layout should normally meet certain density design rules. As shown in Figure 2, layer thickness and local density has a tight relationship [48]. Variation in layer thickness may cause some unwanted degradation in chip performance. To control layer thickness, foundries try to control the local density by imposing density design rules.

If these rules are not met, foundries would usually have to modify the layout by inserting non-functional dummy blocks to achieve the desired density coverage for increasing the chip yield. Typically, density analysis comes with area fill synthesis as a part of CMP process. The goal of density analysis is to find the windows that violate density constraints (upper bound or lower bound). When the density analysis is combined with fill synthesis, the optimal number of demanded dummy blocks is calculated by using an optimization approach. Following such a recommendation, some dummy blocks are inserted into the layout to achieve better pattern density distribution in terms of uniformity.

Foundries have enforced density rules on different layers to minimize the effect of CMP variation, which can consequently control variation of the layer thickness. For example, in 0.35 μm and below technologies, foundries require overall density of any metal layer to be maintained between 35% and 70%. These density design rules are defined based on a specific technology and/or context (such as application specific integrated circuit (ASIC), digital or analog layouts) [48].

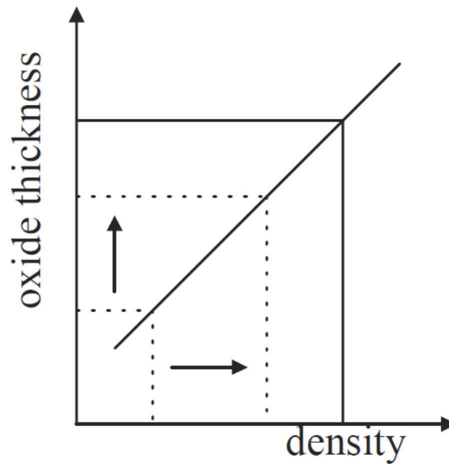


Figure 2 – Relationship between layer thickness and local density [48].

Kahng *et al.* [48] investigated three major approaches of analyzing layout density, including fixed dissection method, optimal extremal-density window analysis, and multilevel density analysis. All of these methods are based on scanning layout with different windows to find maximum and minimum window densities. The density of all $w \times w$ windows in the layout is checked to verify density uniformity distribution in the layout. The size of window (w) is a factor of a few parameters (e.g., process technology, CMP process, *etc*). Checking all eligible $w \times w$ windows in the layout to find extremal windows is a time consuming procedure. Therefore, foundries normally utilize some simpler methods to analyze the density of a layout.

One of such simpler methods is called fixed dissection density analysis method, which is nowadays quite popular and often used by foundries. In the fixed dissection regime as shown in Figure 3, a layout is partitioned to smaller partition-cells, and the density of all the windows are calculated over the layout to find the extremal windows.

As another method, all of the window corners are always on the Hanan grid [49], which is formed when drawing vertical and horizontal lines passing through all the tile edges as shown in Figure 4. The author also proposed a multi-level density analysis approach, which recursively subdivides the layout until the number of tiles in each cell is small for improving accuracy while its running time is still fast. Although the presented algorithms appear to be accurate, the experimental results show they are very slow when the number of tiles is large.

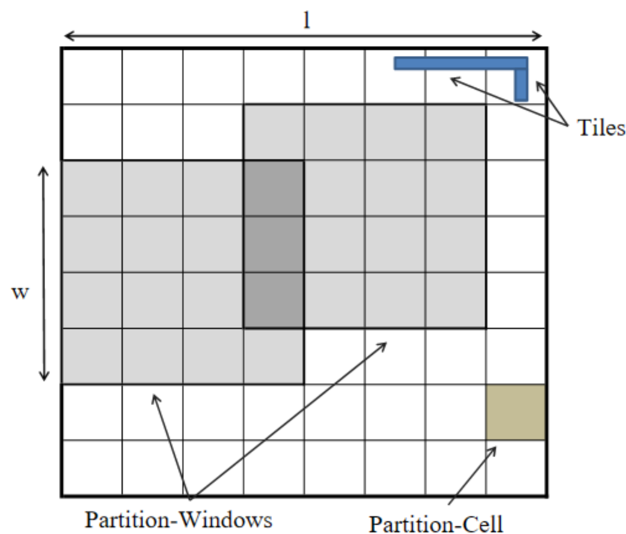


Figure 3 – An $l \times l$ layout is partitioned into smaller cells, each of which has a size of $(w/r) \times (w/r)$. Each $w \times w$ partition-window (light gray) consists of $r \times r$ partition-cells. A pair of partition-windows from different dissections may overlap with each other.

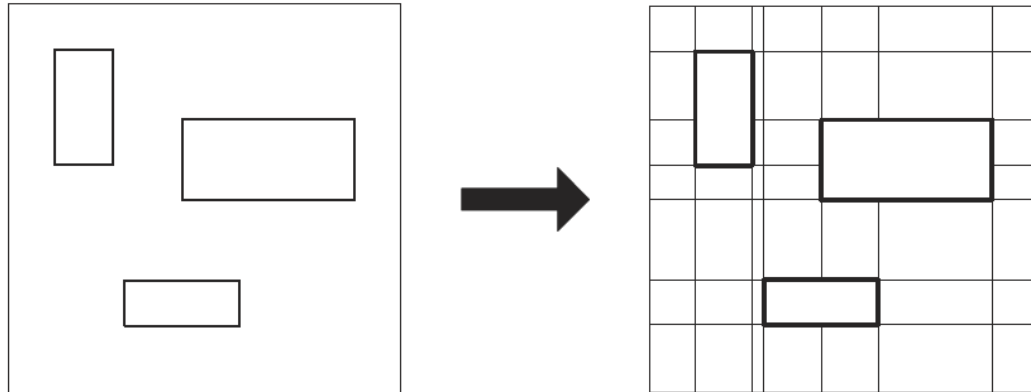


Figure 4 – Generating Hanan grid over a layout [48]

Besides that, the authors of [48] also studied the filling problem by using linear programming (LP) to minimize the density variation of windows (i.e., the maximum window density minus the minimum window density). This formulation is called MIN-VAR, which can suggest the amount of needed dummy fill to be inserted in different regions of the layout. This work is one of the earliest endeavors in the area of dummy fill insertion optimization. Although its optimization coverage and uniformity performance are relatively limited compared to the consequent works, its versatility of the LP formulation has profound impact on the further research of dummy fill synthesis.

In another work, Xiang *et al.* [50] presented a methodology based on the fixed dissection method. The main idea of their research is to recursively use fixed dissection approach with smaller cell size. To efficiently reduce run-time of the algorithm, the authors used a specific structure for mapping the tiles in the layout and avoided double

counting in their equations. This method was claimed to be faster than the previous methods yet accurate.

Chen *et al.* [51] presented Monte-Carlo-based filling methods with similar runtime compared to the previous LP-based methods. In their Monte-Carlo approaches, each partition cell is assigned with a certain priority, which is calculated as a function of multiple parameters, such as density upper bound, available filling area, *etc*. Then one partition cell is chosen based on its priority and filled with the predetermined amount of dummy fill. Although the reported accuracy looks somewhat promising, the drawback of this method is that an excessive amount of dummy fill may be inserted into the layout. The same authors improved their work in [52] by presenting iterative methods based on linear programming, Monte-Carlo and greedy algorithms. To reach optimum solutions, they alternated between two main formulations, minimum window density variation (i.e., MIN-VAR) and minimum amount of inserted dummy fill (called MIN-FILL). The experimental results show that the new methods can derive better solutions at the cost of slightly longer running time over the non-iterated methods.

Another fill synthesis method was proposed to consider the impact of fill insertion on the circuit performance in [53]. The authors first developed a method of estimating the effects of dummy fill on parasitic capacitance and timing overhead in the digital layout design. Then the problem was formulated to minimize the total delay based on integer linear programming and greedy algorithm. In comparison with the other works, this scheme showed a reduction in the total delay of digital circuits. However, since the accuracy and scope of the applied capacitance modelling were not clearly reported in this

work, it is uncertain whether the same modeling scheme can be applied to the advanced technologies. Tian *et al.* [54] also presented a modeling method for density analysis of a layout considering all dies on a wafer. Their formulation of fill placement is similar to MIN-VAR, but with an addition of variation budget. Although the reported results were promising, this paper failed to provide some comparison results with other approaches.

Mukherjee and Chakraborty in [55] presented a random greedy algorithm to insert pixel-wise dummy fill. In their density analysis method, they tried to use floating window to improve density uniformity. However, lack of the details in their implementation makes it unclear for the readers to understand how dummy insertion was performed. Despite the promising test results that showed the efficacy of their proposed method, they actually compared the experimental results with only one distinct method, which was literally aimed at different functionalities. Feng *et al.* [56] developed a polynomial-time approximation algorithm for covering LP (CLP) problem of dummy fill insertion, based on which they presented a greedy iterative algorithm to achieve better performance. While the run time of the proposed algorithm was shrunk significantly, the number of the inserted dummy blocks could be reduced only up to 3%. Moreover, the experimental results of the proposed greedy algorithm were only compared with a Monte-Carlo-based algorithm.

A novel density analysis method with the objective of minimizing gradient density was proposed in [57]. The density analysis was done with window coarsening and uncoarsening to minimize gradient density until the optimal number of dummy blocks was identified. In this work, coupling capacitance constraints were also considered.

However, although the proposed method could significantly reduce the density gradient and the number of inserted dummy blocks, there was no report on the final density variation of the experimental layouts in the paper. Dhumane and Kundu [58] proposed a method of concentrically growing windows to check density uniformity for better dummy fill insertion. With a focus on yield improvement, they intended to balance the competing goals between surface planarity and critical area minimization. However, in spite of the growing window approach for dummy fill insertion, their problem formulation is actually quite similar to the fixed dissection method used in [48]. Saha and Sur-Kolay [59] developed a planarization method to address the limitation of the conventional square partition-windows for dummy fill synthesis. They proposed a Voronoi-diagram-based tessellation for better selection of the positions where dummy fill blocks need to be inserted.

A novel design flow for dummy fill insertion using Boolean mask operations was presented in [60]. To improve computational efficiency, the authors claimed that dummy fill insertion can be moved to mask generation steps when a design is going to be fabricated by foundries. In this process, a mask for dummy blocks is prepared, optimized, and then combined with the masks for the active patterns included in the original design. This proposed flow can significantly reduce the product delivery time. Nevertheless, the performance of the modified layouts was not discussed in the paper although it may be affected due to dummy fill insertion. In a recent work, Liu *et al.* [61] proposed three new filling approaches, which first find fillable areas in a layout and then insert dummy features in a way to minimize the total overlap. In another similar work, Lin *et al.* [62]

introduced a dummy fill insertion method for digital layout design with coupling constraints considered. By using integer linear programming and min-cost flow graph optimization, the proposed method can effectively improve the layout density variation.

2.5 Summary

This chapter is started with a review of the previous works on parasitic capacitance modeling on multi-layer VLSI chips. Then the existing layout generation and retargeting methods/tools are surveyed in the domain of analog and RF integrated circuits. Finally the previous works on layer density analysis and layout density uniformity optimization methods are reviewed. The above literature review would provide a better understanding of the history and the state-of-the-art on the major areas covered in this dissertation research. The next chapter will be focused on the discussion of the proposed methodology for parasitic capacitance modeling in multi-layer layouts.

CHAPTER 3

ANALATIC MODELING OF PARASITIC CAPACITANCE[3]

3.1 Introduction

In this chapter, the analytic models developed for different fringe capacitance components in parasitic capacitance are explained. First a brief discussion about the preliminaries of capacitance modeling is presented. Then the proposed analytic models for substrate capacitance, coupling capacitance, and lateral capacitance are discussed. Finally, the approximated linear models and the utilized curve-fitting technique are discussed.

3.2 Preliminaries of Capacitance Modeling

To facilitate the modeling, as generally used in the literature, we can define two parasitic capacitive components as follows: 1) *overlap capacitance,* which is formed between two overlapping parallel conductors on different layers or the same layer; 2) *fringe capacitance,* which is formed between non-overlapping sidewall of one conductor and surface or sidewall of a second conductor on the same or different layer with reference to the first conductor.

---

[3] The research of this chapter has been published in [A2], [A4], [A8] and [A10].

The capacitance of overlapping metal interconnects can be easily computed by using the following parallel plate equation,

$$C_{ov} = \varepsilon_r \varepsilon_0 \frac{A}{d} \tag{1}$$

where ε_r is the relative dielectric coefficient, ε_0 is the vacuum permittivity, A is the overlap area, and d is the distance between two parallel plates. In the cases where there are multiple dielectric layers, we use weighted average of dielectric coefficients to calculate average dielectric coefficient. This is similar to the method documented in [28].

Figure 5 exhibits electric flux between two metal surfaces (i.e., block *M1* on Metal-1 layer and block *M2* on Metal-2 layer), which generates different fringe capacitive components (i.e., sidewall-sidewall and sidewall-surface). Analytic models of fringe capacitance often deploy technology or geometry dependent parameters [26]. Fringe capacitance modeling also has a close relationship with geometric parameters. In modern technologies, due to technology scaling-down and the trend of using smaller size geometries, fringe capacitance is playing an increasingly more important role in parasitic capacitance.

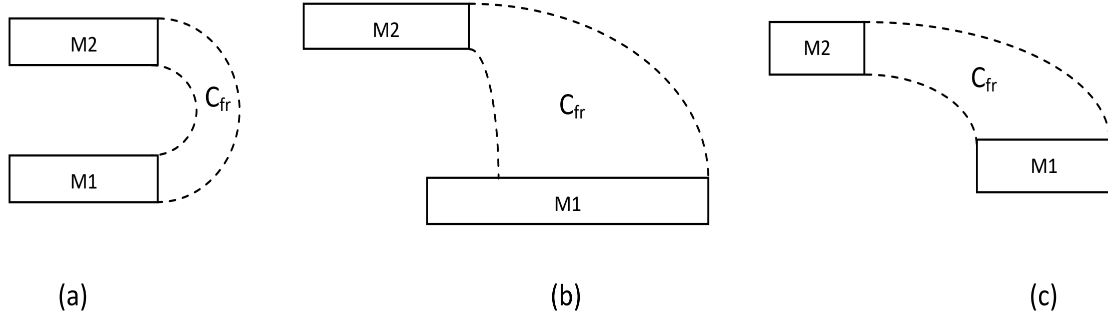


Figure 5 – Fringe capacitances (a) sidewall to sidewall fringe in the case where two metal blocks (M1 and M2) exactly overlap, (b) sidewall to top surface fringe capacitance when two metal blocks (M1 and M2) have overlap and displacement, (c) fringe capacitance between sidewall of top metal block (M2) and top surface of bottom metal block (M1).

Capacitance appearing in circuits can also be categorized from another point of view: *Substrate Capacitance* is formed between an interconnect block and the ground plane, *Coupling Capacitance* is generated between two interconnect blocks on different layers, and *Lateral Capacitance* is produced between two interconnect blocks on the same layer [63]. These terms are widely used in the literature. In some literature, coupling and lateral capacitances are generally called coupling capacitance. Each of the substrate capacitance, coupling capacitance, and lateral capacitance may have overlap and fringe components as described above. In the following sub-sections, the details of modeling these capacitances are explained.

3.3 Modeling of Fringe Capacitance

Our proposed modeling approach, which is aimed to be used as a general method for any kind of technology and geometry, is based on physics concept of electric field and charge between plates. The derived analytic equations are formulated based on

electrostatic field analysis and approximated to simple equations but preserving promising results. We divide the modeling into three different categories. First, we propose an equation to calculate fringe component of substrate capacitance. Then, it is extended to compute fringe component of coupling capacitance. Finally, the method is applied for computation of lateral capacitance.

3.3.1 Substrate Capacitance

Substrate capacitance is a type of capacitance between an interconnect block and substrate plane (usually as ground). An interconnect block may have four fringe capacitance components, one for each side. We will first derive equations for one side, and then we will apply them to the others. In our method, the key point of accurately calculating fringe component is to split it to several major sub-components.

As shown in Figure 7, the fringe capacitance between one side of a metal rectangle *M1* and the substrate plane is made up of C_1, C_2, C_3, and C_4. We can calculate C_2 based on electromagnetic equations by using geometric parameters such as metal thickness *(t)* and vertical distance *(d)* between the metal block and surface of the ground plane. C_1 and C_3 are also calculated in terms of C_2 based on charge density distribution concept. C_4 can actually be neglected due to very less contribution. Then the fringe capacitance for this side would be the sum of the first three sub-components as they are connected in parallel.

To calculate C_2, we assume the electric field lines are circular as shown in Figure 6 and Figure 7. Figure 6 shows the simulation of electric field which forms the capacitance between a metal block and the ground plane. Without loss of generality, we can also

assume the block is on Metal-1 layer and the substrate plane is connected to ground. Then the voltage between the metal block side wall and the ground plane can be calculated by using (2):

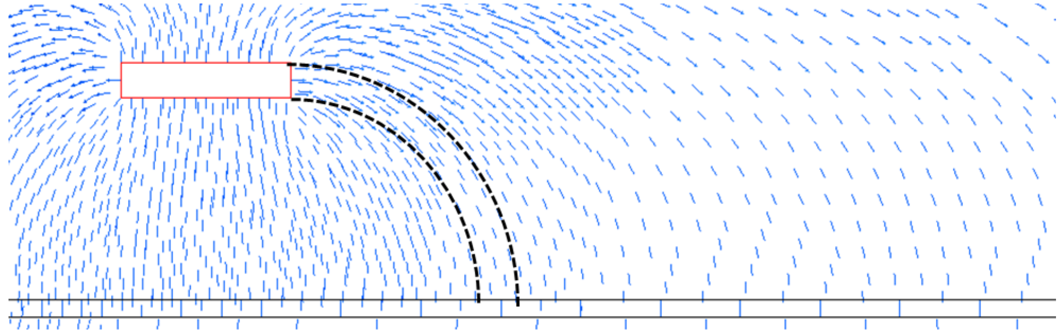


Figure 6 – Electrostatic field simulation shows electric field between the dashed lines can be approximated by the circular lines.

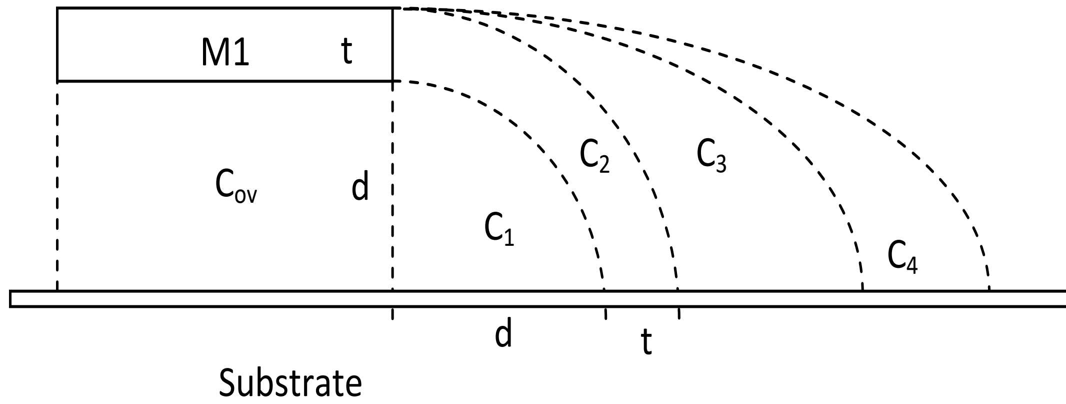


Figure 7 – Splitting fringe capacitance of one side to several sub-components.

$$V = \int_L \vec{E} \cdot \vec{dl} = \int_{\theta=0}^{\theta=\pi/2} E(r) \cdot r \cdot d\theta = E(r) \cdot r \cdot \frac{\pi}{2}. \quad (2)$$

In (2), the voltage is calculated by integrating vector multiplication between electric field vector $\vec{E}$ and length vector $\vec{dl}$ over arc L. As shown in Figure 8, electric field vector

can be converted to a function of radius, *E(r)*, which is independent of angle θ. Vector $\overrightarrow{dl}$ is also converted to $r \cdot d\theta$.

The superficial charge density, which is represented by σ at the distance r from the origin labeled as O on the ground plate, is:

$$\sigma(r) = \varepsilon_r \varepsilon_0 \cdot E(r). \tag{3}$$

After combining (2) and (3), we can obtain

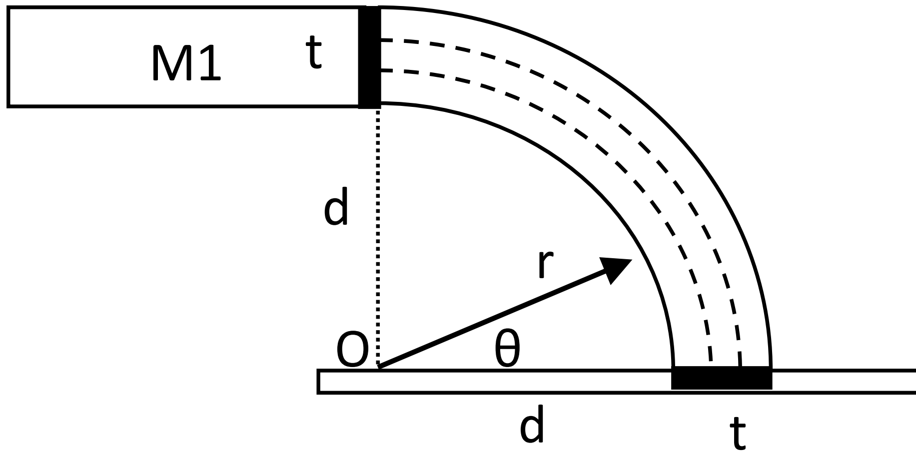


Figure 8 – Calculating C_2 in a circular electric field. *t* is the thickness of Metal-1 layer and *d* is the vertical distance between Metal-1 layer and substrate.

$$\sigma(r) = \frac{2\,\varepsilon_r \varepsilon_0}{\pi r} V. \tag{4}$$

An element of charge for unit length is defined as follows:

$$dq = \sigma(r)\, dS = \sigma(r) \cdot 1 \cdot dr, \tag{5}$$

where dS (shown in Figure 9) is the element of area, which is dr for unit length (i.e., $dy = 1$).

Now by integrating dq, the total stored charge Q_2 between the *M1* sidewall and the shadowed area *A2* on the substrate plane surface can be represented in the form of (6)

$$Q_2 = \frac{2\,\varepsilon_r \varepsilon_0}{\pi} V \int_d^{d+t} \frac{dr}{r} = \frac{2\,\varepsilon_r \varepsilon_0}{\pi} V \ln(1 + \frac{t}{d}). \tag{6}$$

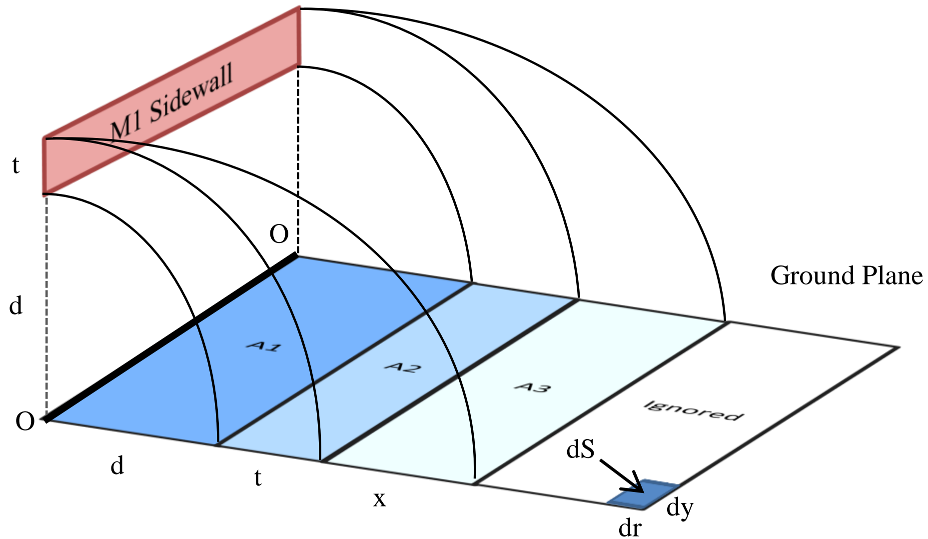


Figure 9 – Actual charge density (shown in colorful density) reduces as the distance from origin *O* increases.

Note that the calculated charge is not the total stored charge in the scenario above. Instead it only represents portion of the charge that is associated with capacitance C_2. Due to that, it can be named as Q_2.

The general equation for calculating capacitance (i.e., $C = Q/V$) can help calculate C_2, that is, for unit length:

$$C_2 = \frac{2\,\varepsilon_r \varepsilon_0}{\pi} \ln(1 + \frac{t}{d}) \tag{7}$$

To calculate the other components, we refer to charge density of the substrate plane surface. The actual charge density decreases when the distance from origin O increases (as shown in Figure 9).

If the average charge density $(\bar{\sigma})$ is considered for the surface, the relationship between C_1 and C_2 can be approximated to the following equation:

$$\frac{C_1}{C_2} = \frac{Q_1/V}{Q_2/V} = \frac{\bar{\sigma} A_1}{\bar{\sigma} A_2} => \frac{C_1}{C_2} = \frac{A_1}{A_2}, \tag{8}$$

where V is the voltage between two plates, Q_1 and Q_2 represent the charge of each related region, and A_1 and A_2 stand for the corresponding area of each segment on the substrate plate surface. Thus, we will have

$$C_1 = \frac{d}{t}\, C_2. \tag{9}$$

C_3 can be calculated in a similar way.

$$C_3 = \frac{x}{t}\, C_2. \tag{10}$$

Now the most challenging issue is how to determine a correct distance for x. As observed from Figure 9, beyond x, electric flux is not strong enough to contribute towards the fringe capacitance formation. So the curve-fitting technique is used to select the right value of x for one specific technology. Other factors such as corner capacitive effect as well as the error due to applying average charge density can be accounted for by choosing suitable technology-dependent values for x. The details of the applied curve-

fitting technique will be discussed in 3.3.5. Therefore, the fringe substrate capacitance for one side is calculated by summing up the three components above:

$$C_{fringe,sub} = C_1 + C_2 + C_3 = (\frac{d}{t} + 1 + \frac{x}{t})\, C_2. \tag{11}$$

The fringe capacitances of the other sides can be calculated in the same way. Thus, the substrate capacitance for a square metal block can be computed by adding overlap capacitance and four side-wall fringe components as follows:

$$C_{sub} = C_{ov} + 4C_{fringe,sub}\,. \tag{12}$$

3.3.2 Coupling Capacitance

Coupling capacitance is the capacitance between two metal blocks on different layers. Figure 5 shows different examples of this capacitance. It includes one overlap component (if applicable) and several fringe components. As discussed in 3.2, the overlap component is easy to compute by using (1), whereas the computation of the fringe components is much more complicated. We can categorize the fringe capacitance into 3 different situations. In Figure 5(a), fringe capacitance is formed between two sidewalls when they are in a line. In Figure 5(b), fringe capacitance is defined between sidewall of *M2* and top surface of *M1*. Fringe capacitance in Figure 5(c) has the same definition, but two metal blocks have no overlap. For each of the situations above we have developed an equation as explained below. Those equations will cover all of the coupling fringe capacitance situations appearing within a design.

Figure 5(a) shows a fringe capacitance when two metal blocks (say, *M1* and *M2*) exactly overlap on one side. In this case, the fringe component can be approximated to a capacitance, which is formed by semicircular shape electric flux between two side walls. If we change the upper bound angle in (2) to $\theta = \pi$ and continue the rest of the derivation, we can reach the following equation for the fringe capacitance in unit length:

$$C_2 = \frac{\varepsilon_r \varepsilon_0}{\pi} \ln(1 + \frac{2t}{d}). \quad (13)$$

Calculating the fringe capacitance in Figure 5(b) is similar to calculating the fringe component of substrate capacitance (11), where x can be determined by the curve-fitting technique.

In case the displacement (w in Figure 10) amount of the top surface of the bottom metal block is less than $(d + t + x)$, a proper value in calculating fringe capacitance should be considered. We can categorize it in the following four different cases as depicted in Figure 10(a)-(d):

- If w is less than vertical distance (d) between two metal layers as shown in Figure 10(a), only a portion of C_1 in (11) is considered as the fringe capacitance. Therefore, the coefficient of C_2 in the rightmost expression of (11) would be $(\frac{w}{t})$, that is, the unit fringe capacitance for that side would be $C_{fringe} = (\frac{w}{t}) \times C_2$.
- If w is between d and $(d + t)$ (where t is the thickness of the top layer) as shown in Figure 10(b), we can consider C_1 plus a portion of C_2 in the formulation. In this

case, we first calculate C_2 and then use the coefficient of C_2 in the rightmost expression of (11) as $(\frac{d}{t}+\frac{c}{t})$, where $c = w - d$.

- If w is greater than $(d + t)$ but less than $(d+t+x)$ as shown in Figure 10(c), the coefficient of C_2 in the rightmost expression of (11) would be $(\frac{d}{t}+1+\frac{c}{t})$, where $c = w - (d+t)$.
- If w is greater than $(d+t+x)$ as shown in Figure 10(d), the coefficient of C_2 in (11) would be $(\frac{d}{t}+1+\frac{x}{t})$ as described earlier.

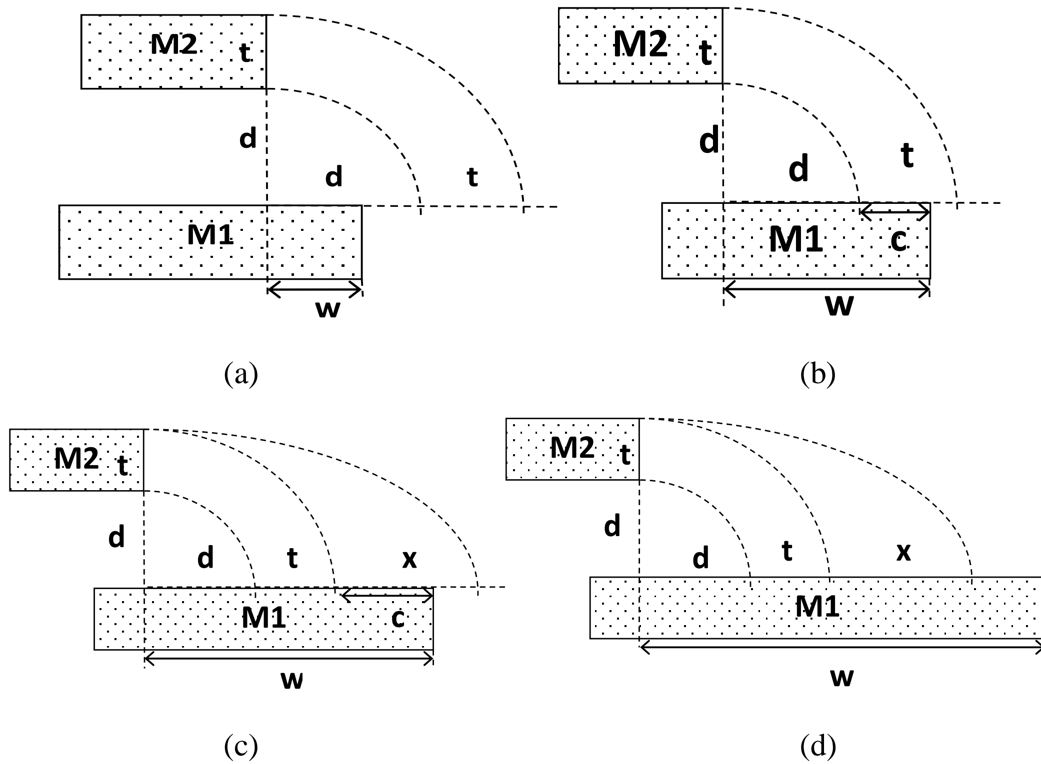


Figure 10 – Choosing a proper factor to calculate the fringe capacitance based on the displacement between *M1* and *M2*.

If two metal blocks have no overlap as displayed in Figure 5(c) and Figure 11, a partial solution from [11] is borrowed for a similar scenario, that is, the unit fringe capacitance is proportional to the expression as given in (14):

$$C_{fringe} \sim \frac{\varepsilon_r \varepsilon_0 t}{y}, \tag{14}$$

where y is the orthogonal distance between corners of two metal blocks, which is expressed as

$$y = \sqrt{d^2 + s^2}, \tag{15}$$

where d and s are vertical and horizontal distances of two corners respectively.

The suitable technology-dependent factor (i.e., the best fit for β in (16)) should be for the following equation in order to calculate fringe capacitance in this scenario:

$$C_{fringe} = \beta \varepsilon_r \varepsilon_0 \frac{t}{y}. \tag{16}$$

Eventually coupling capacitance would be the sum of overlap (if applicable) and different fringe components. Based on different situations, the fringe components can be different combination of the equations above.

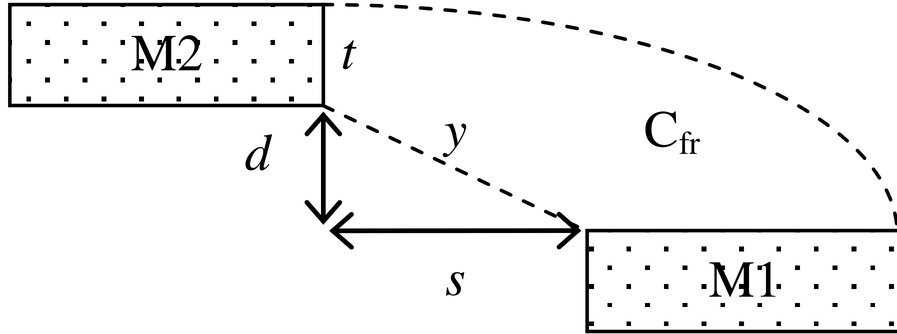


Figure 11 – Modeling details of fringe capacitance in Figure 5(c).

3.3.3 Lateral Capacitance

Lateral capacitance can be actually treated as a type of coupling capacitance between two metal blocks on the same layer. Figure 12 depicts examples of lateral capacitance. If rotating two inline metal blocks as shown in Figure 12(a) by 90 degrees clockwise, a configuration similar to Figure 5(a) in terms of electric flux is obtained. Therefore, the same logarithmic equations derived for coupling capacitance in 3.3.2 can be applied to calculate lateral capacitance between two metal blocks. As shown in Figure 12, the entire scope of metal blocks may not be uniformly competent in capacitance calculation. Based on the distance between two inline metal blocks, we can consider a portion of the entire surface area, because only the adjacent boundary region between the two metal blocks contributes the majority of the electric flux in forming the lateral fringe capacitance.

Here, concept of effective width (w_e) is deployed to adjust the equations for lateral capacitance. The electric flux beyond the effective width is actually not significant any more to contribute towards capacitance formation. Such an effective width is a function

of distance between two metal blocks. The curve-fitting technique is used to determine w_e. Following the observation above, we can see that the nature of lateral fringe capacitance is exactly the same as that of coupling fringe capacitance but with the consideration of utilizing new parameters.

Thus, for the top, bottom, or side (as the case shown in Figure 12(a)) unit capacitance, (13) is changed to the following:

$$C_{top/bottom/side} = \frac{\varepsilon_r \varepsilon_0}{\pi} \ln(1 + \frac{2w_e}{s}), \qquad (17)$$

where s is the distance between two metal blocks and w_e is the effective width coefficient obtained from the curve-fitting technique that will be explained in 3.3.5. For C_{top}, C_{bottom} or C_{side}, the corresponding ε_r should be used. And for side capacitance in Figure 12(b), (11) is applicable.

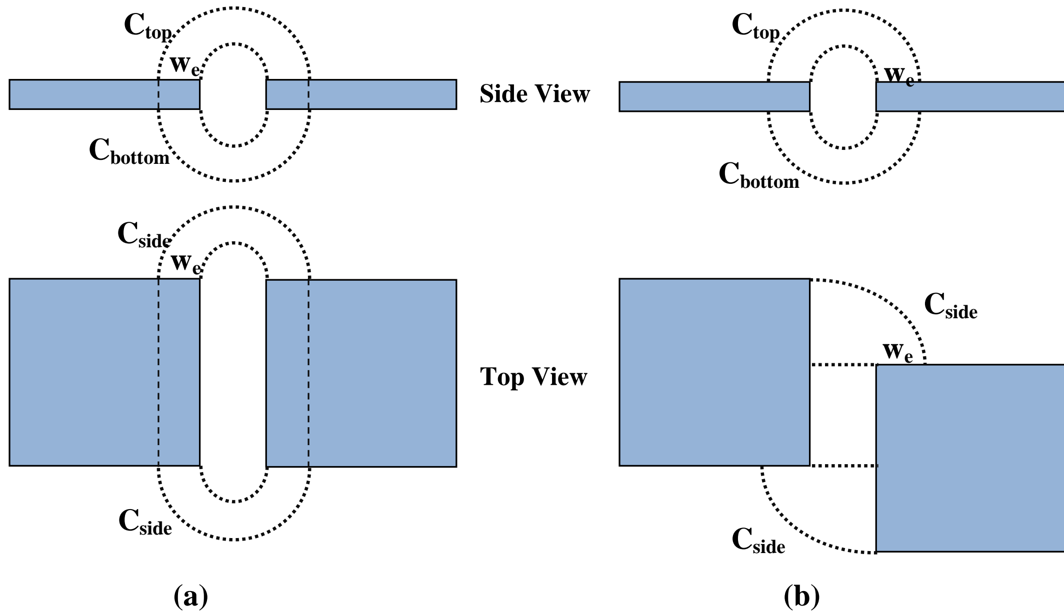


Figure 12 – Lateral capacitance with its different components. (a) Two metal blocks are in a line. (b) Two metal blocks have displacement.

3.3.4 Linear Model

To make the derived model even simpler for being applied to the dummy fill or any other loops of optimization in the CAD algorithms, we also consider the following substitution to make our equation linear:

$$ln\,(1 + x) \;=\; cx \qquad when\; a \leq x \leq b, \tag{18}$$

where a and b are the lower and upper bounds of the region where we can model the logarithm function to a linear function. This linear approximation is shown as an example of $c=0.75$ in Figure 13. As can be seen from Figure 13, two curves are very close to each other within the range of x between 0.65 and 1.2. Therefore, the modelling error would be ignorable. On the other hand, as this linear equation is much simpler in complexity, the

computation time can be definitely reduced compared to the logarithmic one. The range for linearization is selected based on the technology parameters.

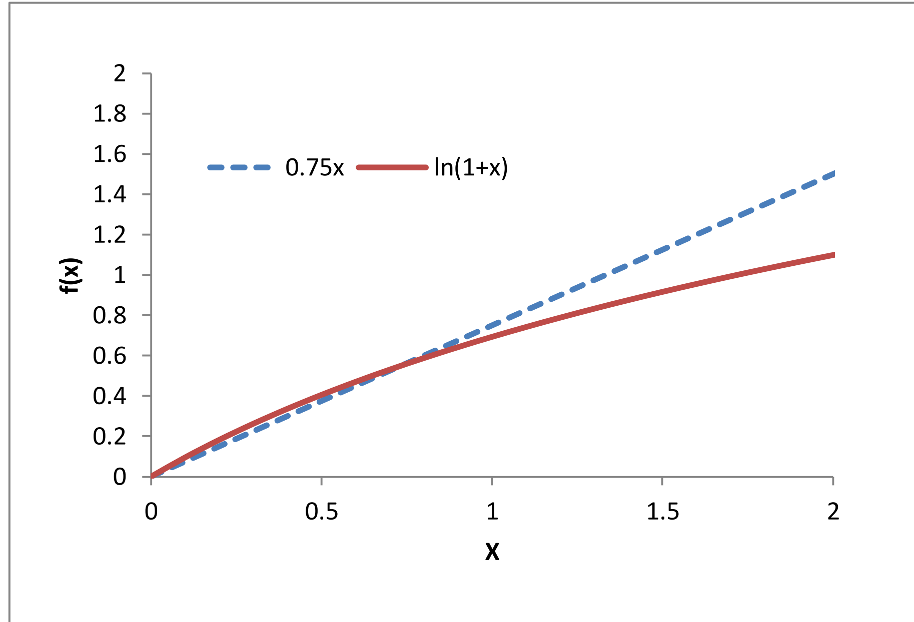


Figure 13 – Linearization of ln(1+x) to 0.75 x in range of $0.65<x<1.2$

3.3.5 Curve-Fitting Technique

There are two types of technology parameters. The first one is technology profile parameters, e.g., layer thickness, layer-to-substrate distance, and dielectric coefficients. They can normally be directly obtained from technology files or foundry process documentation. The second type is technology-dependent parameters for our proposed modeling methodology (e.g., x/t, β, and w_e). We deployed curve-fitting technique to tune the templates derived in the previous section in order to determine the suitable technology-dependent parameters.

To achieve this, we selected several samples for each technology characterization scenario. The corresponding parasitic capacitances were extracted for those samples and the trend of the data was obtained. To fit our template equations to the selected samples, we determined the parameters that are defined in the previous sections. These parameters are normally considered as a function of displacement as illustrated for each capacitance category below. For instance, to get the proper value of x/t for substrate capacitance modeling in a specific technology, we used four cases of 1, 3, 5, and 7 μm blocks. As another example, to determine the value of x/t for the coupling capacitance modeling, we selected several test cases in the following manner. As size and displacement in the X and Y directions may be changing, four samples with fixed size and fixed displacement in the Y direction, but different displacement in the X direction (with the same interval within the entire feasible range) were selected.

The derived technology-dependent parameters were then inserted to the modeling equations, which were later used to estimate other test cases so that estimation errors with reference to the extracted results could be analyzed and reported. Note that the process of determining technology-dependent coefficients is done only once for each technology and then the proposed equations along with the determined parameters can be fed to optimizers or extraction tools. Our test scenarios include different geometry sizes and distinct relative displacement values. In Section 6.2, parameter determination and experimental verification are detailed for substrate capacitance, coupling capacitance and lateral capacitance.

3.4 Summary

In this chapter, simple yet accurate analytic model for calculating substrate, coupling, and lateral capacitances based on geometric parameters has been presented. The proposed method can cover all the different parasitic capacitance forms that may appear in the VLSI layouts. In addition, it is easy to be utilized in any technology processes with little tuning effort. Next chapter will be focused on the general flow proposed for controlling density uniformity during layout generation process and its corresponding optimization formulation.

CHAPTER 4

DENSITY-UNIFORMITY-AWARE ANALOG LAYOUT RETARGETING[4]

4.1 Introduction

As explained in Section 2.3, analog IC layout automation is normally performed in two different ways. By using analog layout synthesis tools, a design typically presented in schematic form can produce its corresponding layout based on the applied constraints. With the second method called layout retargeting, the knowledge of a silicon-proven layout is utilized to generate a new layout for updated design specifications and/or new technology processes [64].

In this dissertation research, the layout retargeting method is selected since it is not only still immature from the technical perspective, but also owns great potential in the current VLSI CAD commercial market. The demand of conducting automated layout migration from an old technology to an advanced nanometer technology is expected to grow in the current technology exploding era. As a matter of fact, it is fairly flexible to add more constraints and optimization schemes to our existing constraint-graph-based layout retargeting method. Thus, we are able to add density uniformity feature and

---

[4] The research of this chapter has been published in [A1], [A3], [A6] and [A7].

parasitic capacitance consideration to make the layout retargeting process in compliance with the advanced design for manufacturability. In this chapter, our layout retargeting approach will be first described. And then the details of the proposed density-uniformity-aware methodology will be explained.

4.2 Regular Retargeting Process

To conduct layout migration for updated design specifications or new technologies, a platform has been developed to extract and store the layout information for further processing. The initial layout, normally a fine-tuned silicon-proven one, is extracted and stored by using corner-stitch [65], an efficient data structure for layout representation [40]. There are *solid* and *space* tiles on each layer within the layout where the solid tiles represent functional rectangles and the space tiles denote empty areas. Figure 14 shows an example of corner-stitch data structure used in our established platform to store layout information. The gray tiles are solid tiles and the white tiles are space tiles. The solid and space tiles are mutually connected with the aid of corners' information just like stitches in the cloth sewing for easy access. Figure 15 shows a simple flow diagram of the implemented retargeting platform.

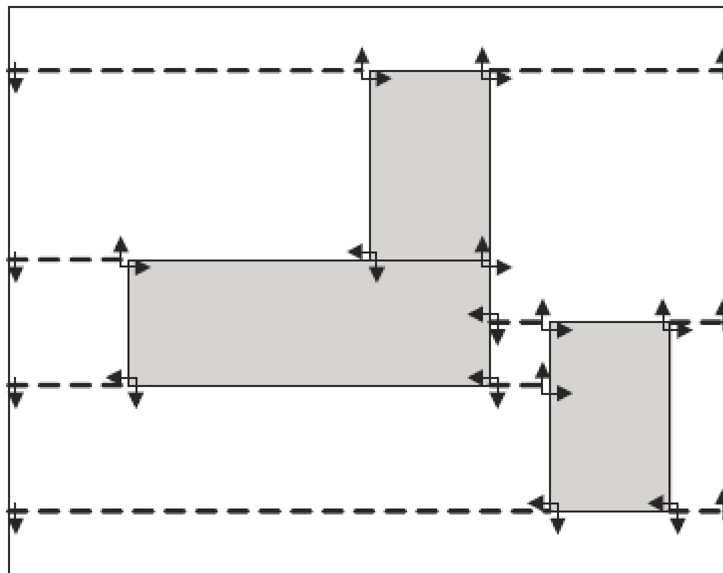


Figure 14 – A mask layer representation in corner-stitching data structure [40]

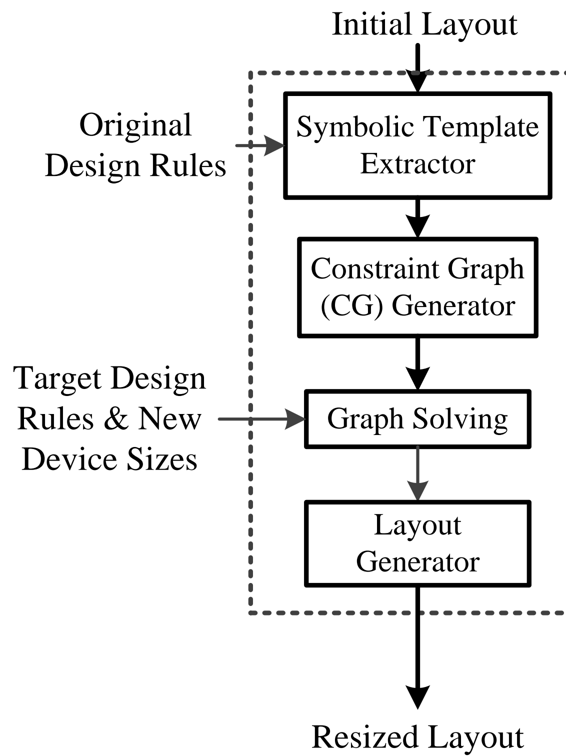


Figure 15 – Flow diagram of the regular retargeting platform

As shown in Figure 15, the Symbolic Template Extractor module can derive a symbolic template from the initial layout for retargeting process. A symbolic template is actually a group of constraints reflecting the abstract layout properties, such as devices, connectivity, technology design rules, and analog layout integrity features (such as symmetry and matching constraints). During the symbolic template extraction, the initial layout is scanned using scan-line algorithm [66] to recognize all layout features. Then the entire layout features such as active and passive devices, netlist, connectivity and layout symmetry features are extracted. All of these extracted layout properties will be reflected in the symbolic template. Such generated constraints help preserve the structure of the layout and ensure correctness of the design rules. The constraints enforced by the technology design rules fall into the following three categories: (1) *minimum size* of a tile, (2) *minimum spacing* between two electrically unconnected tiles on the same or different layers, and (3) *minimum extension* of two overlapping tiles on different layers.

The symmetry and matching constraints can be extracted from the original layout by the Symbolic Template Extraction module as shown in Figure 15. These analog special constraints can help preserve the topological features of sensitive devices during the layout retargeting process. These constraints are represented in the weighted directed constraint graphs (CG), on which a longest-path algorithm (e.g., Bellman–Ford algorithm [67] deployed in this work) can be performed to optimize the size and location of each tile in the layout. Moreover, the current flow and current density requirements from the analog circuit design perspective can also be represented as predefined electromigration

constraints, which are certain minimum sizes of the corresponding interconnect tiles in the constraint graphs.

The constraint graphs are generated in both horizontal and vertical directions and solved separately. Each solid tile is represented by two nodes and one arc in between within each of the constraint graphs. The space between two solid tiles is reflected by another arc. Figure 16 shows a sub-graph representation example of 3 tiles in the horizontal direction.

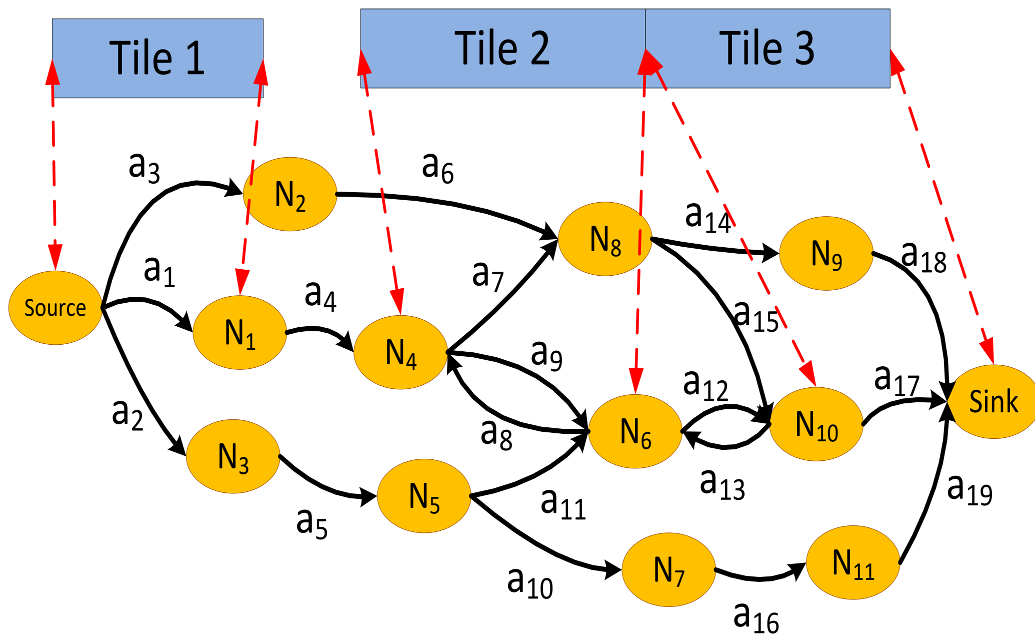


Figure 16 – An exemplary horizontal constraint graph including a sub-graph representing the placement of three solid tiles.

The nodes in the constraint graph along with their corresponding tile edges are visually marked (as cross-reference) with red dash double-ended arrows in the figure.

Here, we explain in detail how the horizontal constraint graph in Figure 16 is generated from the layout geometries:

After analyzing the layout, all of the solid tiles, their corresponding edges and their locations are identified. Let us call the locations of the left edge and the right edge of *Tile 1*, *Lt1* and *Rt1* respectively. We choose the same naming style for *Tile 2* and *Tile 3* as well. Therefore, the following constraints are extracted from the layout shown in Figure 16:

$$Rt1 - Lt1 \geq minimum_size \tag{19}$$

$$Lt2 - Rt1 \geq minimum_spacing \tag{20}$$

$$Rt2 - Lt2 = given_size^{5} \tag{21}$$

$$Lt3 - Rt2 = 0 \tag{22}$$

$$Rt3 - Lt3 \geq minimum_size \tag{23}$$

The equations or inequalities above demonstrate the layout symbolic template according to the technology design rules. Inequality (19) shows that the width of *Tile 1* cannot be smaller than the minimum size imposed by the design rule constraints. Inequality (20) shows that the distance between the left edge of *Tile 2* and the right edge of *Tile 1* should not be smaller than the minimum spacing enforced by the technology design rules. We assume the width of *Tile 2* should be fixed according to the designer

[5] Given_size is the size of a tile determined by designer as one of the inputs to the program.

preference. Thus, equation (21) implies that the distance between the right and left edges of *Tile 2* is constant. Equation (22) shows that *Tile 2* and *Tile 3* are electrically connected while their distance is described as 0. Based on (23), the width of *Tile 3* is also considered to be greater than the minimum size imposed by the technology design rules. As shown in the equations above, all the edge locations are not fixed. The extracted template can only show the relative positions between them. The final location of each tile and consequently the size of each tile have to be identified after running an optimization sorting on the constraint graphs.

Once each geometric component is recognized, the layout can be represented by its symbolic template, where each tile edge is assigned with a node in the constraint graph, and the arcs between two nodes are connected. For example, *Tile 1* corresponds to two nodes *source* and N_1 for its left and right edges, while arc a_1 represents the width constraint of *Tile 1* between *Source* and N_1, which is extracted from (19). The arc weight of a_1 is the minimum size requirement defined by the technology design rules. Similarly, *Tile 2* (or *3*) has two nodes N_4 (or N_{10}) and N_6 (or *Sink*) for both side edges. Arc a_4, whose weight can be the minimum spacing defined by the technology design rules, represents the distance between *Tiles 1* and *2* (as shown in (20)). If the length of *Tile 2* has to be kept fixed, two arcs a_8 and a_9 in the opposite directions with the same weight but different signs should be defined between nodes N_4 and N_6 as shown in Figure 16. These two arcs in fact represent equation (21). Moreover, arcs a_{12} and a_{13} with both weights of 0 show the connectivity constraint between nodes N_6 and N_{10} so that *Tiles 2* and *3* can always abut with each other (similar to equation (22)). The other nodes (e.g., N_2, N_3, N_5,

N_7, N_8, N_9, and N_{11}) represent the surrounding geometries around *Tiles 1-3*, which however are not exhibited in Figure 16.

When the graph is generated based on the original technology process, optimization process will be started as shown in Figure 15 in the "Graph Solving" module. First, the device sizes are updated to the new sizes provided by the designers, and the target technology design rules are applied to the constraint graphs. In the next stage, the graph compaction algorithm is applied to the constraint graphs to optimize the final location of each node. To get estimation about the minimum and maximum locations of each node in the constraint graphs, a longest-path algorithm is performed twice to pull all the nodes to left/bottom and right/top for each horizontal and vertical constraint graphs. In the final round, the optimum location of each node is identified based on their minimum and maximum locations by using wire length minimization algorithm [68]. And then the *Resized Layout* can be generated from the updated constraint graphs via the "Layout Generator" module with the nodes' final locations as input information.

4.3 Density-Uniformity-Aware Analog Layout Retargeting Process Flow

In the proposed methodology, we opt to modify solid or space arcs to apply desired changes in the layouts. In contrast to the previous density control work that handles fixed layouts mainly for digital circuits, the applications of our methodology in the context of analog layout migration has no constraint on the solidity of geometries in the layouts although the relative location of devices should be followed as per the extracted symbolic template. That is to say, by using our proposed approach, we are not necessarily only

bound to dummy fill insertion to achieve density uniformity. Instead, we are able to modify any geometries (including the sizes and locations) as long as these changes impose no impact on the circuit performance. A suitable fixture for this is our weighted directed constraint graphs, which can supply good guidelines to govern the modifications and balance the various factors to reach the most beneficial holistic solution. This is the advantages of our proposed schemes, which can offer the designers more flexibility to gain density uniformity through multiple options.

Our proposed density-uniformity-aware analog layout retargeting flow is composed of several steps as shown in Figure 17. First, a regular retargeting process (the left part within the dotted block of Figure 17) is performed on the input initial layout to derive a new layout called *resized layout*, since new device sizes are already taken in along with the target technology design rules. And then a density analysis followed by an LP optimization will determine the solution (in terms of density budget) in order to plan where and how much to improve the density distribution. We use fixed dissection approach for density analysis (as detailed in section 4.4.1). The obtained information from the density analysis is used as the input to the density planning module. Different from the previous works that solely rely on dummy fill insertion for density uniformity, in this dissertation research we have developed an innovative technique to reposition and enlarge the functional geometry features for the analog layouts.

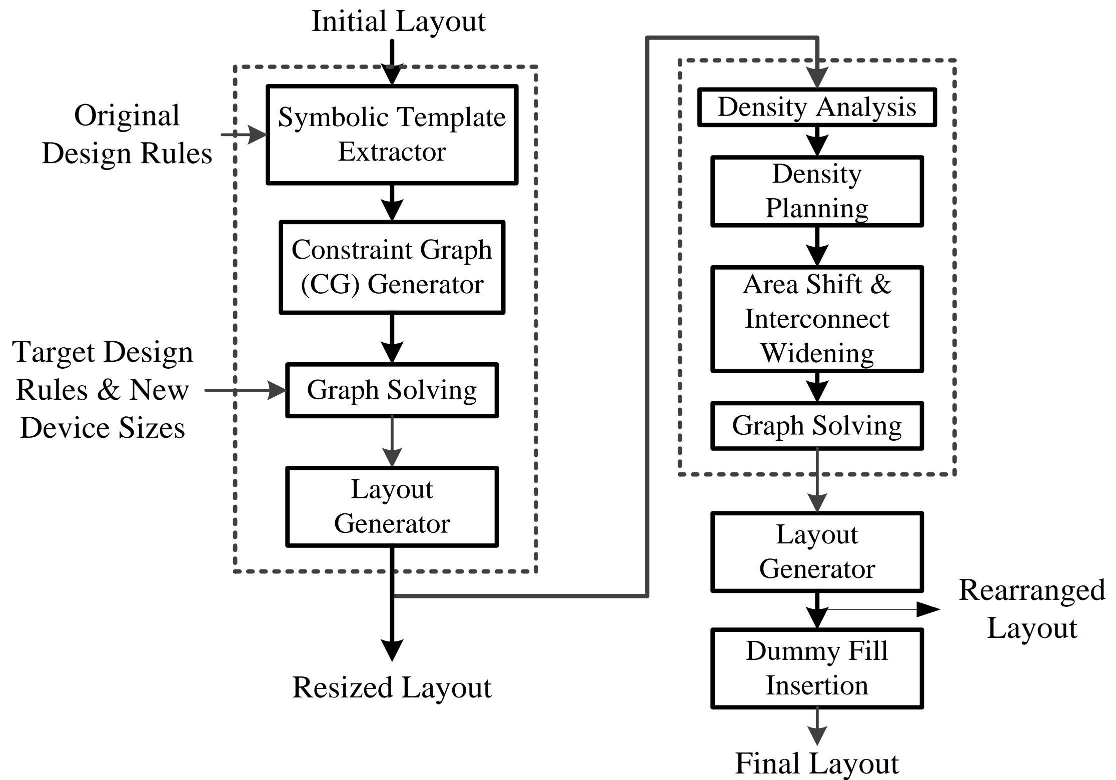


Figure 17 – Flow diagram of the proposed density uniformity-aware layout retargeting methodology.

We advocate to first shift some geometries and then modify some features (mainly interconnects due to the stringent performance-related geometry requirements for devices) in analog layouts as per the planned density budget. The recommended density allocation is integrated into the graph-based analog layout retargeting process as special constraints to impose the symbolic template. Then the updated constraint graphs are solved to generate the *rearranged layout* as shown in Figure 17, where the core of our density-oriented handling in the proposed analog layout retargeting flow is shown within the right dotted block. As the last resort, for empty spaces without interconnects inside

the rearranged layout, the traditional dummy fill insertion scheme will be applied to generate the *final layout* as shown in Figure 17, which features the improved pattern density distribution.

Therefore, by using the proposed methodology we are able to increase density in three different ways: (1) by applying *Area Shift (AS)* where we shift tiles from their current cell to neighboring cells; (2) by using *Interconnect Widening (IW)* where the interconnects are mainly widened to increase the pattern density; (3) and by Inserting *Dummy Fill (DF)* if the area shift and interconnect widening approaches cannot reach the best solution due to limited available resources (i.e., the existing geometries) in the layout. We can choose one of the schemes above or combine them to achieve better density uniformity. Based on the recommended solution and selected schemes, the designed algorithm changes or moves interconnect tiles in the layout. By investigating each tile and considering its maximum allowable size as well as the location range of its edges, the amount of increase in the size of each tile or its new location is calculated in order to be applied to the constraint graphs. After all tiles are explored, a longest path optimization algorithm is executed on the constraint graphs. The last option, dummy fill insertion, may be utilized to further decrease layout density variation. As shown in Figure 17, after solving the modified constraint graphs, we can use dummy fill insertion as a complementary approach to finalize the generated layout if the layout density variation still needs to be improved.

The rationale of our handling strategy above is, in the retargeting process of analog layouts, we do not need to consider the interconnect geometries in the layout as fixed

blocks. As a matter of fact, the symbolic template can change a fixed layout to the ones with floating rectangles featuring variable sizes and locations for various applications. Therefore, we can modify any existing geometries, which are not involved with performance-related sensitive analog building blocks, in the initial layout to increase the coverage density in different regions of a layout rather than only depending on insertion of external dummy fill to the layout.

4.4 Linear-Programming Formulation and Optimization

In Figure 17, our approach is distinguished from the other works by the modules inside the right dotted block. In this section we will discuss the general flow to tackle density-uniformity-aware retargeting approach by focusing on the first two modules inside the right dotted block. First, the layout generated from the regular retargeting approach, i.e., the resized layout, is analyzed for its pattern density distribution. Then by performing the density planning, the solution to improving density uniformity is identified. The details of our approach is explained in the following two sub-sections.

4.4.1 Density Analysis

In our approach, a resized layout generated from the regular retargeting process (as shown in Figure 17) needs to be modified to improve its pattern density distribution. The fixed-dissection density analysis is deployed in our work to analyze the resized layout to evaluate its current density status. As mentioned in Section 2.4, fixed-dissection density analysis is the most common method widely used by the foundries to control density

uniformity of the layouts. In this approach, a layout is partitioned to smaller cells and a sliding window scans the layout to find density violations within the layer. The size of the window is determined by various technology parameters.

The following notations are used to express the fixed dissection density analysis approach:

- The input is a given layout (with a size of $l \times l$), which includes rectangular geometries called *tiles* with any arbitrary sizes.
- A fixed window (called *partition-window*) with a size of $w \times w$ is sliding over the layout to check and apply density requirements. w is determined by a few parameters, such as the technology design rules, CMP process, *etc*.
- Each window includes $r \times r$ cells (called *partition-cells,* r is an integer divisor of w). In the r-fixed dissection approach, the input layout is partitioned into $(r \times l/w) \times (r \times l/w)$ smaller cells, each of which has a size of $(w/r) \times (w/r)$. In this work, we use horizontal and vertical indices (ranging from 0 to $r \times (l/w)-1)$ to refer to each partition-cell.

The density is checked on all of the partition-windows, each of which has a step size of w/r (i.e., one cell). For the example shown in Figure 3, the given layout has a size of 800µm×800µm (i.e., l=800µm), the window size is 400µm×400µm (i.e., w=400µm), and each window includes 4×4 cells (i.e., r=4). The entire layout is partitioned into 8×8 cells (i.e., $r \times l/w$=8). Upon finishing the density analysis, the density of each partition-window and partition-cell is identified. The result of density analysis will be used in the density planning formulation to compute the optimum solution for density uniformity problem. In

the next sub-section, the optimization formulation is presented. Density analysis is tightly correlated to the density planning problem. Therefore, using a proper window size and an appropriate number of partition-cells inside each partition-window is important for deriving better solution for the density planning problem. As pointed out in [48], a small value of r may reduce accuracy and cannot guarantee that the uniformity is completely applied to the layout. However, a large value of r may increase the number of partition-cells for high accuracy in the layout density analysis. But in the meantime it would dramatically increase the complexity of the problem as well as the density analysis time and solution convergence time.

4.4.2 Problem Formulation

Based on the current local densities of a given $l \times l$ layout that is partitioned into smaller partition-cells, we formulate the optimization problem to achieve uniform density distribution. In the fixed dissection method, a partition-window scans the entire layout (layer by layer separately) with a step size of one partition-cell. In our proposed approach, we intend to modify the resized layout by rearranging pattern elements, increasing size of interconnects, and/or inserting dummy fill. In the problem formulation, we use one set of variables for added area to each partition-cell and the other set of variables for transferring area between different partition-cells as follows:

- CA_{ij}: the amount of the added area to partition-cell C_{ij} (where i is used as row index and j is used as column index with C_{00} referring to the left-top corner partition-cell within a layout). This variable can be implemented either by

increasing size of any functional tiles in the partition-cell or by dummy fill added to the partition-cell.

- AXL_{ij}: the amount of the occupancy area to be shifted from partition-cell C_{ij} to its left neighbor partition-cell.
- AXR_{ij}: the amount of the occupancy area to be shifted from partition-cell C_{ij} to its right neighbor partition-cell.
- AYT_{ij}: the amount of the occupancy area to be shifted from partition-cell C_{ij} to its top neighbor partition-cell.
- AYB_{ij}: the amount of the occupancy area to be shifted from partition-cell C_{ij} to its bottom neighbor partition-cell.
- A_{ij}: a general parameter for occupancy area shift variables associated with partition-cell C_{ij}, which might be AXL_{ij}, AXR_{ij}, AYT_{ij} or AYB_{ij}.
- CD_{ij}: the current pattern density of partition-cell C_{ij}, which is obtained from the density analysis.
- SD_{ij}: the summation density of partition-cell C_{ij}, which includes the current density plus all the added and traded densities, that is, $SD_{ij} = CD_{ij} + CA_{ij} + AXL_{ij} + AXR_{ij} + AYT_{ij} + AYB_{ij}$. Note that the occupancy area shift variables can be interpreted as follows: a positive occupancy area shift represents the amount of the occupancy area that is moved in from the corresponding side of its neighbor partition-cell into C_{ij}, while a negative occupancy area shift represents the amount of the occupancy area moved out from C_{ij} into its neighbor partition-cell on the corresponding side.

- WD_{ij}: the density of partition-window W_{ij}, which includes $r \times r$ partition-cells. Therefore, it is calculated from average of all partition-cells: $WD_{ij} = \frac{1}{r \times r} \sum_{s=i}^{i+r-1} \sum_{t=j}^{j+r-1} SD_{st}$.
- M and N: two auxiliary variables for restricting partition-window densities. M is smaller than the minimum partition-window density and N is greater than the maximum partition-window density.
- $MaxCap(C_{ij})$: the available area for density control operations inside partition-cell C_{ij}, which is calculated during the density analysis process based on the upper bound of local density imposed by the technology design rules.

By using the notations above, we formulate the LP problem of minimum variation between sweeping windows as the defined objective and constraints below:

Objective: Maximize (M-N)

Subject to:

$$0 \leq \mathrm{CA}_{ij} \leq \mathrm{MaxCap}(\mathrm{C}_{ij}), \qquad \mathrm{i,j} = [0, \frac{rl}{w} - 1] \tag{24}$$

$$-\frac{\mathrm{CD}_{ij}}{4} \leq \mathrm{A_{ij}} \leq \frac{\mathrm{CD}_{ij}}{4}, \qquad \mathrm{i,j} = [0, \frac{rl}{w} - 1] \tag{25}$$

$$\mathrm{A}_{ij} = 0, \text{ for the area shift variables on layout borders} \tag{26}$$

$$\mathrm{AXL}_{ij} + \mathrm{AXR}_{i(j-1)} = 0, \ \mathrm{i} = [0, \frac{rl}{w} - 1], \mathrm{j} = [1, \frac{rl}{w} - 1] \tag{27}$$

$$\mathrm{AYT}_{ij} + \mathrm{AYB}_{(i-1)j} = 0, \quad \mathrm{i} = [1, \frac{rl}{w} - 1], \mathrm{j} = [0, \frac{rl}{w} - 1] \tag{28}$$

$$\mathrm{M} \leq \mathrm{WD}_{ij}, \qquad \mathrm{i,j} = [0, \frac{rl}{w} - r + 1] \tag{29}$$

$$N \geq WD_{ij}, \qquad i,j = [0, \frac{rl}{w} - r + 1] \tag{30}$$

$$L \leq WD_{ij} \leq U. \qquad i,j = [0, \frac{rl}{w} - r + 1] \tag{31}$$

In the formulation above, (24) is to ensure that the demanded area variables are positive and less than the maximum available capacity, $MaxCap(C_{ij})$. If variable CA_{ij} is supposed to be implemented by using dummy fill, $MaxCap(C_{ij})$ is calculated by collecting all the available empty spaces in the corresponding partition-cell as the upper bound of CA_{ij}. But if CA_{ij} is used as a guide for interconnect widening operation, the maximum amount of possibly added areas is calculated based on the interconnect occupancy within the current partition-cell. For instance, for a completely empty partition-cell, its $MaxCap(C_{ij})$ is zero as there is no interconnect to be enlarged. However, it is equal to the density upper bound of the corresponding partition-cell if using dummy fill insertion is chosen. Therefore, $MaxCap(C_{ij})$ amount is able to guide LP to provide more realistic solutions.

Equation (25) ensures that the area shift variables are limited to the current density of each partition-cell divided by 4. As there are four variables corresponding to four sides of each partition-cell, only one-fourth of the current density is allocated for each area shift variable. Equation (26) is only applied to the boundary cells, which ensures there is no area transferred to the outside of layout. Therefore, AXL_{i0} (i.e., left area shift variables of the cells located at the first column), $AXR_{i(rl/w-1)}$ (right area shift variables of all the cells located at the last column), AYT_{0j} (top area shift variables of all the cells located at the first row), and $AYB_{(rl/w-1)j}$ (bottom area shift variables of all the cells located at the last

row) are all zero. Equations (27) and (28) express that the sum of the area shifts among neighbor partition-cells are zero. This ensures the amount of the transferred area towards the left neighbor partition-cell of C_{ij} is equal to the amount of the transferred area towards the right neighbor partition-cell of $C_{i(j-1)}$ but with the opposite sign. So do the neighboring top and bottom partition-cells. For example, for partition-cell C_{22}, $AXL_{22}+AXR_{21}=0$ and $AYT_{22}+AYB_{12}=0$.

In (29), M, one of the auxiliary variables, behaves as the lower bound of density for all the partition-windows. Each partition-window includes $r \times r$ partition-cells, each of which has the summation density among the current density, added areas, and shift areas. In (30), N is another auxiliary variable functioning as the upper bound of density for all the partition-windows. Equation (31) defines the upper bound and lower bound for each partition-window density, which are imposed by technology design rules.

With the objective function, by maximizing M-N, the formulated program can minimize the density variation among all the partition-windows, which is similar to the MIN-VAR formulation presented in [48]. M is smaller than the minimum partition-window density and N is greater than the maximum partition-window density. Thus, the maximum amount of the term M-N is zero. In the best scenario, the LP solver may reach the objective function of zero, which means we have achieved zero density variation. Obviously, this meets the goal of achieving a uniform layout in terms of density distribution.

On the other side, our formulation above has a significant difference from the original MIN-VAR formulation by introducing area shift variables and their

corresponding constraints. Moreover, the traditional MIN-VAR formulation has only been applied to the dummy fill insertion scheme, whereas $MaxCap(C_{ij})$, in our proposed formulation, can address any feasible options including both interconnect widening and dummy fill insertion as long as the coverage of partition-cells can be increased. In addition, another auxiliary variable N is deployed beside the regular auxiliary variable M, which is used in the traditional dummy fill insertion formulations to increase density of the lowest coverage window and consequently to reduce density variation. In the traditional dummy fill insertion approaches, window density can only be increased by adding dummy features, whereas in our methodology the density of the highest coverage window can be reduced by moving area outwards. Therefore, in our proposed formulation, the variable N is used to reduce the highest density window in addition to increasing the lowest density window by using M variable in order to achieve better solutions.

By solving the aforementioned linear programming problem by a solver engine (Here simplex algorithm [69] is used from MATLAB [70] linear programming solver) optimum values for CA_{ij} and A_{ij} will be obtained. With this solution in hand, our algorithm understands how much density should be added to each partition-cell and also how much density should be transferred from one partition-cell to its neighboring partition-cell.

4.5 Summary

In this chapter, we have presented a novel approach to improve density uniformity for analog integrated circuits during layout retargeting process. Unlike the previously published works that are mainly focused on dummy fill insertion, our methodology is based on the nature of analog layouts, which allows one to modify the location and size of functional geometries to improve density uniformity. This appealing aspect is particularly effective when combined with symbolic template, which can determine the location and size of any geometry based on the updated constraints in the analog layout retargeting process. In the next chapter, we will explain the implementation details of our proposed density-uniformity-aware analog layout retargeting formulation.

CHAPTER 5

DENSITY-UNIFORMITY OPTIMIZATION AND CONTROL ALGORITHM[6]

5.1 Introduction

As mentioned in the previous chapter, the optimum solution from the LP-based formulation is the guide of the subsequent operations for improving the layout density uniformity. In this chapter, the fulfillment details of the density-uniformity optimization solution will be fully discussed. The operations of fulfilling the solution might include shifting tiles between partition-cells, enlarging interconnects, and inserting dummy features. In this dissertation research, the proposed density controlling approaches are integrated into the analog layout retargeting platform by smartly modifying the constraints and allocating proper weights to the arcs in the constraint graphs. Thus, our special density controlling schemes including area shift and interconnect widening can be achieved in addition to the conventional method of dummy fill insertion. By running graph optimization algorithms (such as Bellman-Ford longest-path algorithm), an optimized layout featuring uniform density distribution can be generated according to the updated design specifications and/or new technology design rules.

[6] The research of this chapter has been published in [A1], [A3], [A6] and [A7].

As shown in Figure 17, dummy fill insertion is deployed as a complementary method to generate the final layout. Since the dummy fill insertion method has been well studied in the literature, their operations will not be detailed in this chapter. Instead, the focus of this chapter would be on the special handling of area shift and interconnect widening operations. The planning process, including density analysis, LP problem formulation, and problem solving, is done in a layer-by-layer manner, while the constraint graph optimization and solution derivation are managed for all the layers together in a holistic way for a multi-layer layout. Thus, we will only elaborate on a simplified scenario of single-layer process in this chapter to mainly illustrate the underlying principles.

In the following two sections, the weight allocation algorithm for implementing interconnect widening and area shift approaches is presented. And then the parasitic coupling capacitance integration to the layout retargeting process is explained.

5.2 Interconnect Widening Process

The optimization variable CA_{ij} can be interpreted as the amount of area that should be added into the existing functional interconnects within partition-cell C_{ij} to implement the recommended solution. The algorithm for the interconnect widening operation is listed in Figure 18, which includes two nested loops to handle the density-rule-constrained layers one after another. For each of such layers in the resized layout, all the solid tiles are enumerated and split into separate ones if they are shared among multiple partition-cells. Layer density is analyzed in Line 3 by following the fixed dissection

method as described in Section 4.4.1. The LP problem formulation in Line 5 is detailed in Section 4.4.2.

```
    Algorithm: Interconnect Widening Operation
1.  Foreach (all the density-rule-constrained layers in the layout)
2.    Split any shared tiles among multiple partition-cells;
3.    Analyze the layer density;
4.    Calculate the available room for each node in the constraint graph (CG);
5.    Formulate and solve the LP problem;
6.    Foreach (all the tiles on the current layer)
7.       Identify the recommended solution for the corresponding partition-cell;
8.       Calculate new size based on the given solution;
9.       Remove extra arcs to unbind inline edges;
10.      Allocate redundant space based on the critical path and available room for expansion;
11.      Do depth-first search to update node room in the CG;
12.      Update the corresponding arcs in the CG;
13.   Endfor
14.   Solve the longest-path problem for the CG;
15. Endfor
```

Figure 18 – Pseudo code of the interconnect widening operation

For each density-rule-constrained layer, all the solid tiles are processed one by one. For the corresponding nodes of each solid tile in the constraint graphs, the maximum and minimum allowable locations are estimated (by running a longest-path algorithm on the constraint graphs in forward and backward directions). Since the weight allocation algorithm is aware of the recommended density solution for the partition-cell where the tile is located, it can propose new dimensions for the tile by assigning enlarged weights to the corresponding arcs in the constraint graphs. To take into account the critical path (as shown in Line 10 of Figure 18) that determines the size of the final layout, we always use

the maximum room space as the quota for weight assignment. Note that a critical path is the longest path among all the paths that we can find in one directed constraint graph from left to right (or from bottom to top) to identify layout dimensions.

As an example, part of the horizontal constraint graph is shown in Figure 16. The process of the graph generation is explained in Section 4.2. The minimum/maximum locations of graph nodes obtained by running the longest-path algorithm on the constraint graphs in forward and backward directions can serve as the guides for our algorithm. Based on such minimum/maximum locations, the allowable range, called *room*, for the location of node n can be derived:

$$room_n = maxLoc_n - minLoc_n \quad (32)$$

For each tile t and its corresponding width arc in the horizontal direction, the maximum allowable weight can also be calculated from minimum/maximum locations as follows:

$$max_arc_weight_t = maxLoc_{Re} - minLoc_{Le} \ , \quad (33)$$

where $maxLoc_{Re}$ is the maximum location of the right edge and $minLoc_{Le}$ is the minimum location of the left edge. Likewise, the maximum allowable weight of the tile length arc in the vertical direction can be also derived by using the maximum location of the top edge and the minimum location of the bottom edge.

In Figure 16, there are several paths from *Source* node to *Sink* node, while the longest one, which determines the layout horizontal size, is the critical path. The length

of a path is identified as the sum of arc weights along that path. Assume the path including a_1, a_4, a_9, a_{12} and a_{17} is the critical path with the length of $L_{critical}$, i.e.,

$$L_{critical} = a_1 + a_4 + a_9 + a_{12} + a_{17} \ . \tag{34}$$

For each path p from *Source* node to *Sink* node, the path *redundant space* is calculated as follows:

$$redundant_space_p = L_{critical} - L_p \ , \tag{35}$$

where L_p is the current length of path p. That is to say, we are allowed to increase any path length up to the length of the critical path by allocating the redundant space to any arc weights within that path. Hence, this calculated redundant space is going to be shared among all the arcs along the path, and changing any arc weight should be always followed by updating the redundant space of the path and the room of all the associated nodes dynamically, so that any path enclosing the updated arc should not exceed $L_{critical}$ as indicted in Lines 8 and 10 of Figure 18. Here a depth-first search is performed in the constraint graphs (Line 11) to update the room of the other nodes connected to that updated arc in the graph data structure to facilitate the allocation of the rest of the redundant space. Once all tiles are processed, the constraint graphs will be solved with a longest-path algorithm to generate the *Rearranged Layout* as shown in Figure 17.

In case there is no enough room to produce sufficient density by only increasing the size of interconnects, an integral solution has to be called upon by first increasing the area of interconnect geometries as much as possible and then inserting dummy fill to reach the quota derived from LP optimization. Note that enlarging interconnects cannot be

performed in certain situations. For the geometries, which are used to determine the electrical nominal parameters (e.g., channels of MOSFET devices, resistive stripes of passive resistors, or parallel plates of passive capacitors), it is not allowed to change their sizes due to their significance to the circuit performance.

The other functional geometries, which work as interconnections, are the candidates to be modified by our interconnect widening process. However, there is one challenging situation for our proposed approach to function well with full efficiency. If one interconnect, which itself is one piece of tile in the resized layout, is shared by multiple partition-cells, only one unique widening size with the minimum amount can be applied to the constraint graphs even though some of the partition-cells may allow for more widened room. In the next section, we will present the smart splitting algorithm to resolve this problem while improving interconnect widening efficiency.

5.2.1 Smart Splitting

Before explaining the splitting approach, we will look at one situation more closely, as shown in Figure 19, to understand the problem and its solution. Figure 19 shows a big tile when it is split into two smaller tiles and the corresponding constraint sub-graphs generated in horizontal and vertical directions. The horizontal CG is shown on the top and vertical CG is shown at the bottom of the tiles. All the edges and their corresponding nodes in the CGs are shown as well. *Lti*, *Rti*, *Tti* and *Bti* show the left, right, top and bottom edges for the corresponding nodes respectively. The two arcs with zero weight between *Rt1* and *Lt2* in the horizontal CG in Figure 19(a) show the electrical connectivity

constraint between *Tile 1* and *Tile 2*. In this way two tiles are always abutting with each other. Two zero arcs between the top edges of *Tile 1* and *Tile 2* (i.e. *Tt1* and *Tt2*) in vertical CG in Figure 19(c) shows the locations of these two top edges are always equal, which means they are always put in line with each other. This is exactly the same situation for the bottom edges of these two tiles. Therefore, it is shown that splitting a tile into two smaller tiles will not let the tile sizes change freely, unless the constraints of the top and bottom edges are relaxed. As shown in Figure 19(d), to relax the constraints for the top edges and bottom edges, we can remove one of the zero arcs in between these edges. Therefore, the location of *Tt2* can be higher than that of *Tt1*, while the location of *Bt2* can be lower than that of *Bt1*. This means the height of *Tile 2* can be greater than that of *Tile 1*. Although this can partially solve the problem, the height of *Tile 2* will increase consequently if the height of *Tile 1* becomes larger. Therefore, we need a better solution.

As mentioned before, to achieve better performance, our interconnect widening algorithm can set out to split a one-piece tile to several smaller ones that are not shared between adjacent partition-cells, meanwhile maintaining electrical properties of the interconnect nets. Although these multiple tiles may have any connectivity constraints, their associated edges have to be kept on the same line for abutting in order to guarantee the electrical connectivity. Moreover, newly generated tiles should have no dependency on each other if one of their sizes is changed.

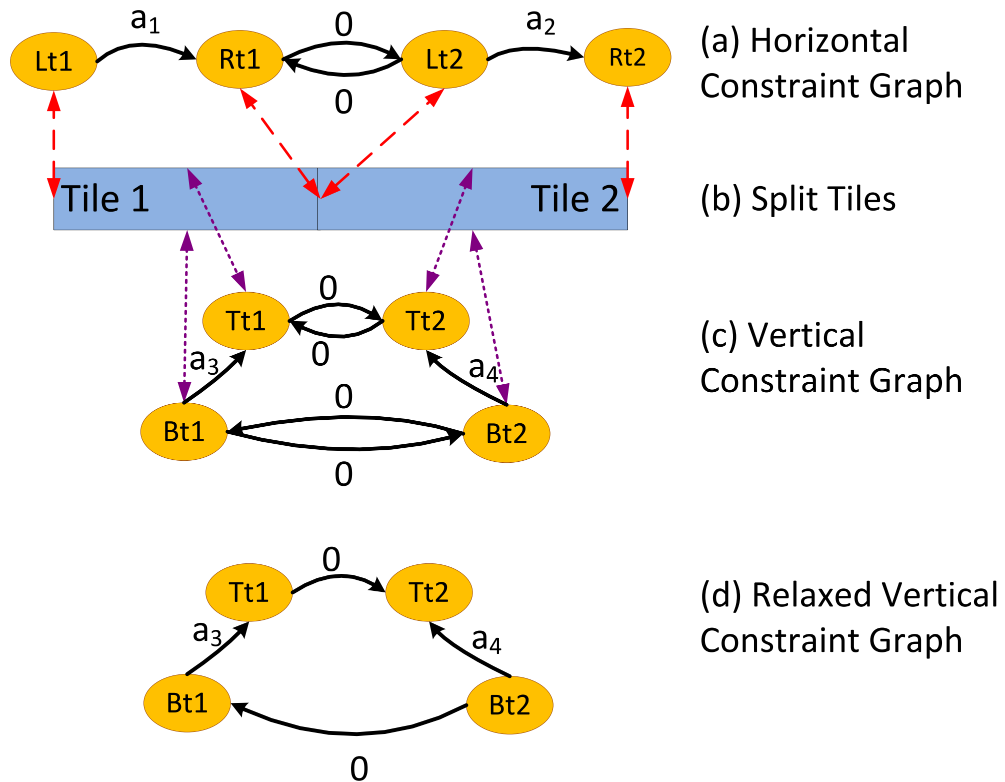


Figure 19 – Example of splitting a tile into smaller tiles and the corresponding constraint graphs in (a) horizontal and (c) vertical directions, (c) for split tiles and (d) for the relaxed vertical constraint graph

We achieve this by introducing an *intermediate tile* between two split tiles in the corresponding partition-cells as shown in Figure 20. The intermediate tile *M* is kept as the minimum size, while the connectivity constraints between two split tiles (i.e., *S1* and *S2*) and the intermediate tile are relaxed. As shown in Figure 20(b), there are two arcs with zero weight between the top edges of S_1 and *M* and similarly, between the top edges of S_2 and *M*. These four arcs would keep the top edges of the three tiles on the same line. This

actually discourages independent changing of each tile size. To resolve this problem, we unbind the top edges of S_1 and S_2 by removing two lower arcs from the constraint graphs as shown in Figure 20(c). Therefore, the only two arcs left with zero weight would let the top edges of S_1 and S_2 to move independently from each other but more than the top edge of M, while the intermediate tile M is kept as is. A similar operation can be done with the bottom edges of tiles S_1, M, and S_2. A possible shape of the interconnect after all of the operations above is shown in Figure 20(d).

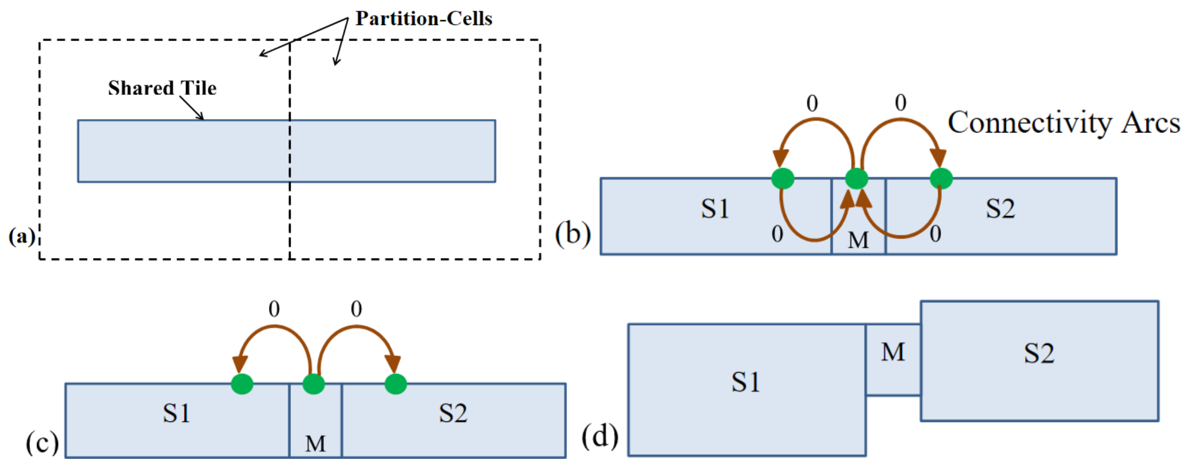


Figure 20 – Updating connectivity constraints after splitting the shared tiles

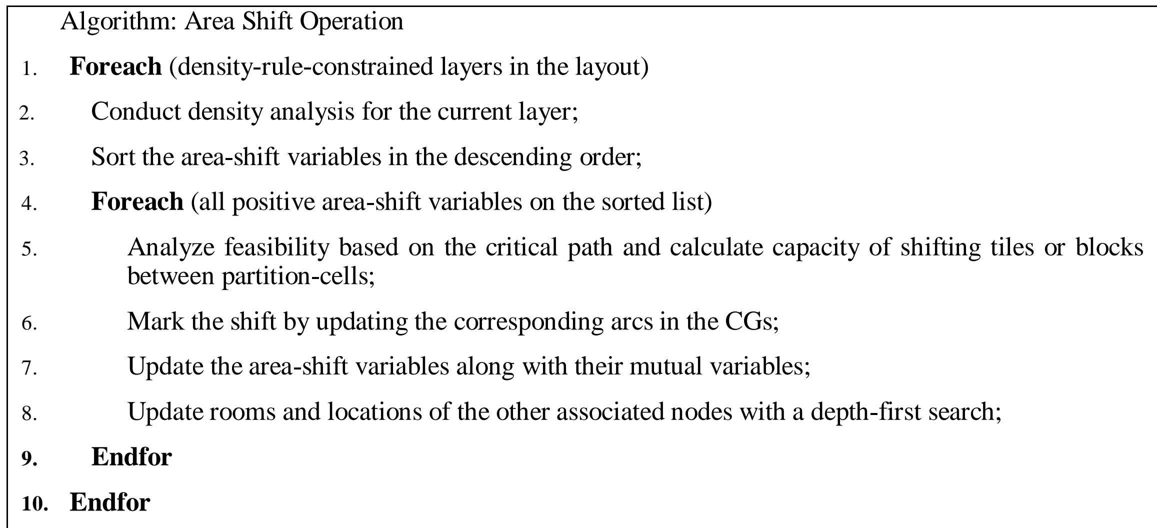
After updating the constraint graphs by removing some connectivity arcs (as listed in Line 12 of Figure 18) and modifying weights of the split tiles within different partition-cells, the longest-path algorithm would be performed on the constraint graphs to generate the rearranged layout. Since the interconnect widening scheme is aimed to add more pattern density by changing the size of the originally existing geometry, we do not necessarily expect a satisfactory minimum variation by this stage alone due to probably

limited available resources in the layout. Instead, we can still use the traditional dummy fill insertion as a complementary process to further shrink the density variation of partition-windows.

5.3 Area Shift Process

Variables A_{ij} obtained from the LP solution suggest the scheme of moving geometries around in the layout to achieve better density uniformity. The contribution from area shift for improving density variation is non-negligible since the variables can be up to plus or minus one fourth of the current densities. A positive solution means the area is moved into the partition-cell from the corresponding side and the negative solution for each variable indicates the area is moved out from the corresponding side of the partition-cell. The algorithm pseudo code for the area shift operation is listed in Figure 21. For each density-rule-constrained layer, the area-shift variables are first sorted in the descending order (Line 3).

To achieve geometry move-out for a suggested area amount, we need to first identify the tiles inside the partition-cell close to the corresponding side boundary. If the tile edges have enough room (the same room concept as described before) to move towards the neighboring partition-cell, we calculate the movement distance, which is consistent to the suggested shift area amount. This is implemented by updating the corresponding arcs in the constraint graphs (Line 6), which represent the spacing between different tile edges on the corresponding sides.



Algorithm: Area Shift Operation

1. **Foreach** (density-rule-constrained layers in the layout)
2. Conduct density analysis for the current layer;
3. Sort the area-shift variables in the descending order;
4. **Foreach** (all positive area-shift variables on the sorted list)
5. Analyze feasibility based on the critical path and calculate capacity of shifting tiles or blocks between partition-cells;
6. Mark the shift by updating the corresponding arcs in the CGs;
7. Update the area-shift variables along with their mutual variables;
8. Update rooms and locations of the other associated nodes with a depth-first search;
9. **Endfor**
10. **Endfor**

Figure 21 – Pseudo code of the area shift operation.

For instance, if the weight of arc a_4 in Figure 16 is increased, we can indeed move *Tile 2* to the right and make it to stay farther from *Tile 1*. After implementing one area-shift variable, its mutual area-shift variable on the opposite side has been also implemented, as described in (27) and (28). The handling of the area shift variables is only executed for all the positive ones on the sorted list in order to implement the obtained solution. When the amount of shift is applied to the constraint graph by updating an arc weight, a depth-first search is executed to update the room of any associated nodes and their final locations so that the entire paths passing through this arc can utilize the redundant space properly.

Since the modified constraints already consider the critical path that is supposed to be intact after the area shift operation, it is certain that the final layout will feature an improved density distribution. We have included two schemes in our realization of the area shift operation: *global area shift* and *local area shift*. In the global area shift scheme,

we use the formulation presented in Section 4.4, where all of the partition-cells contribute their area shift variables in the formulation.

In contrast, in the local area shift scheme, only the area shift variables corresponding to a few specific partition-cells are involved in the LP formulation. In this way, we can simplify the formulation by significantly reducing the number of variables. Compared to the global area shift scheme, the goal in the local area shift scheme is the same, which is to minimize density variation of partition-windows. This can be achieved by only modifying maximum and minimum partition-windows. The density of the maximum partition-window can be reduced by moving out some tiles, whereas the density of the minimum partition-window can be increased by shifting in some tiles. Thus, we reduce the density difference between the maximum and minimum partition-windows to consequently reduce window density variation in the layout.

As shown in Figure 22, for one maximum or minimum partition-window, we can only include the marked area shift variables in the LP formulation. In this way, the number of the used variables can be reduced. Thus, in the best case we can reduce the number of area shift variables from $4\times(l\times r/w)\times(l\times r/w)$ in the global area shift scheme to only $2\times2\times(r+r)$ in the local area shift scheme (when we only consider one maximum and one minimum partition-window in formulation), where l represents the layout size, w stands for the partition-window size, and r is the dissection number in a partition-window of the fixed dissection density analysis regime. To implement the local area shift scheme in this work, after density analysis for a density-rule-constrained layer, we look for maximum and minimum partition-windows (within the tolerance of a user-defined factor,

e.g., 5%) and group them into MAX-Group and MIN-Group partition-windows, respectively.

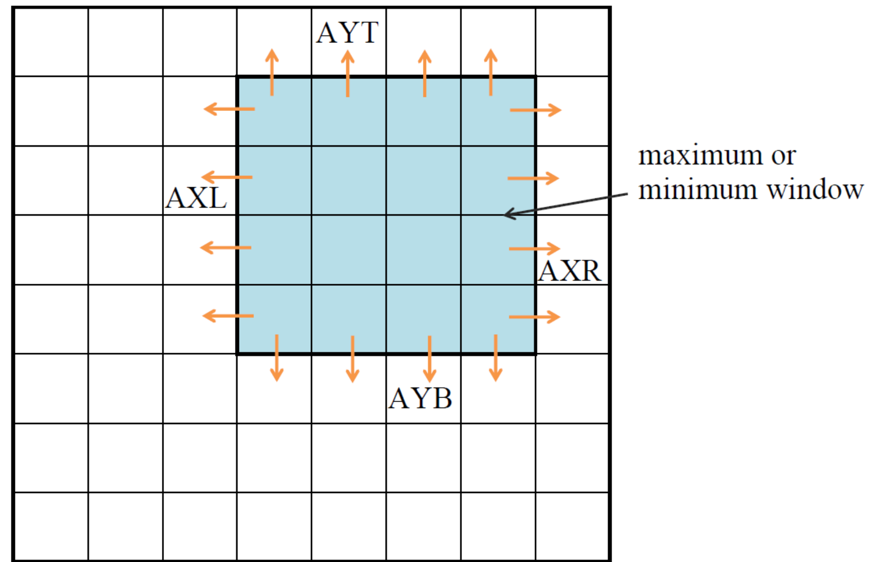


Figure 22 – Area shift variables for the maximum or minimum partition-window in the local area shift scheme.

That is to say, if the difference between density of the extreme partition-window and the second extreme partition-window is very small, we would include the area shift variables of both extreme partition-windows in the formulation. When the groups are created, we only include moving out area shift variables for the MAX-Group and moving in area shift variables for the MIN- Group partition-windows in the LP formulation. This can take place by setting proper negative and positive constraints for those variables in equation (25) respectively. Once the LP problem is resolved, we can only focus on the realization of the area shift variables within the partition-cells enclosed by the MAX-Group and MIN-Group partition-windows.

5.4 Parasitic Capacitance Control during Layout Retargeting

In this research, we aim to take into account the parasitic capacitance effects in the process of density-uniformity-aware analog layout retargeting. With the modeling scheme proposed in Chapter 3, we can add parasitic capacitance consideration into the optimization scheme for the density uniformity problem. To avoid increasing the complexity of the problem, we use the proposed linear capacitance model (expressed in Section 3.3.4 and equation (18)) for the constraints. As mentioned before, the overlapping capacitance and the fringe capacitance can be modeled to be proportional to geometry size. Therefore, the entire capacitance models for different categories and scenarios can be made linear to tile size. Here, the capacitance budge is defined for each individual net (for substrate capacitance) or each pair of nets (for coupling or lateral capacitance). The capacitance budget can be read from a pre-defined file provided by the users after running a bunch of simulations. Therefore, for each net or each pair of nets in the input layout, a capacitance value is assigned and used as its budget during layout modification. As shown in Figure 23, different capacitance components can be extracted between adjacent tiles for two different nets[7]. During the realization of area shift and interconnect widening where the constraint graphs are updated by processing tiles, the arc weight allocation scheme as described in Sections 4.2 and 4.3 would always consider the parasitic capacitance budget.

---

[7] Electrically connected tiles and interconnects are considered as one net in a circuit.

Each selected tile may be assigned with a new value to enlarge its size. The selected tile should be also checked with its parasitic capacitance related to other surrounding nets. The new dimension of the tile may change the parasitic capacitance between its belonging net and other surrounding nets.

The new parasitic capacitance is calculated based on the newly assigned sizes and distance between tiles. The amounts of enlargement in interconnect size and change in its new location are reflected in the calculated parasitic capacitance, which would consume a portion of the budget between each net or each pair of nets. The constraint graph arc weight allocation scheme controls the amount of arc weight change in the constraint graphs such that the capacitance budget constraint is not violated.

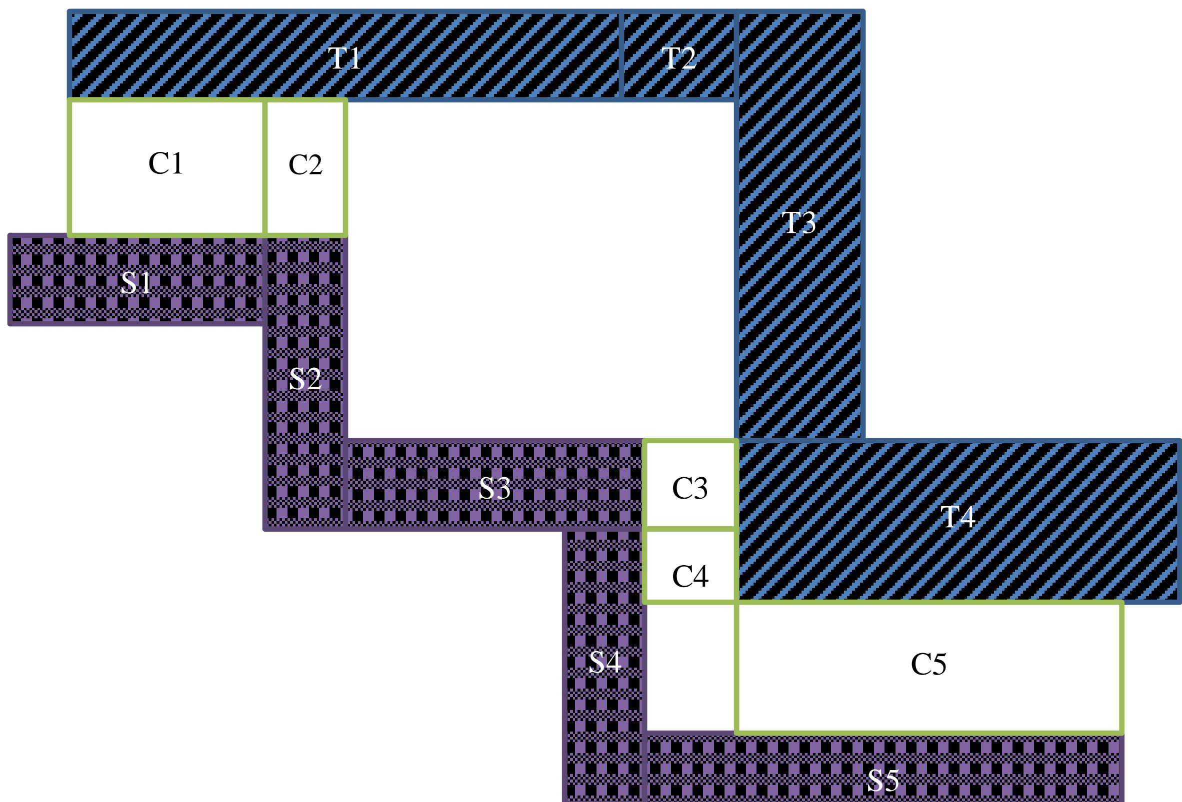


Figure 23 – parasitic capacitance components between two adjacent nets.

If a new capacitance value is going to be more than the budget, then the control scheme would adjust the new tile dimension to be compliant with the capacitance budget. One example is shown in Figure 23, where five different parasitic capacitance components are found between two closely opposite tiles (The tiles can be in different layers, therefore, proper capacitance equation is utilized to extract the capacitance value). The capacitances are marked with C_i, whose formation is displayed between two close-by corresponding tiles. Thus, the budget between two nets T (consists of tiles T_i) and S (consists of tiles S_i) can be defined as follows:

$$Capacitance\ Budget = C_1 + C_2 + C_3 + C_4 + C_5. \tag{36}$$

If the vertical dimension of tile T_4 intends to increase by the constraint-graph arc weight allocation scheme, capacitances C_3, C_4 and C_5 are subject to change because of the overlap amount between T_4 and S_3 (for C_3 component), increased overlap amount between T_4 and S_4 (for C_4 component) and decreased vertical distance between T_4 and S_5 (for C_5 component). If the sum of these increases in coupling capacitance between the aforementioned nets is more than the capacitance budget, the constraint-graph arc weight allocation scheme will limit the intended dimension change of tile T_4 to fit the capacitance budget accordingly. By running the capacitance control scheme above during the process of constraint graph modification, the degradation effect induced by the parasitic capacitance on circuit performance would be effectively controlled.

5.5 Summary

This chapter explains the control and optimization logic we have utilized to implement density-uniformity-aware analog layout retargeting methodology formulated in the previous chapter. First of all, interconnect widening scheme was explained. And then smart splitting scheme to improve interconnect widening performance was discussed. Then the area shift approach including its global and local shift scheme was explained. Finally the way how parasitic capacitance consideration is integrated into the density-uniformity-aware analog layout retargeting methodology was presented. The next chapter is focused on the experimental results for verifying our parasitic capacitance modeling as well as the density-uniformity-aware layout retargeting performance.

CHAPTER 6

EXPERIMENTAL RESULTS[8]

6.1 Introduction

In this chapter, the proposed methodologies described in the previous chapters are tested and verified. First of all, the developed model for fringe capacitance will be tested by running different test scenarios and a case study. To verify the proposed model, we developed scripts to generate thousands of layout samples, which cover all possible geometric situations for CMOS 180, 90 and 65 nm technologies. The proposed model is compared with previously published works in reference to the extracted results from commercial tools. The experimental results show that the estimation errors of the proposed method are much lower than 10% (2%-4% or less for most of the cases) but with significantly reduced computation effort. The proposed model is a general methodology that can be used for any nanometer technologies with different geometric parameters.

In Section 6.3, the result of testing density-uniformity-aware layout retargeting is presented. Our experimental results show that the proposed approach can account for up to 80% improvement towards the ideal density uniformity in the regular analog layouts.

---

[8] The research of this chapter has been published in [A1], [A2], [A3], [A4], [A6], [A7], [A8] and [A10].

This promising option can significantly decrease the capacitance-induced parasitic effects due to the traditional sole dummy-insertion operation.

6.2 Verification of Capacitance Modeling

This research aims to develop less complex models for parasitic capacitance of metal blocks used as interconnects or dummy fill within a layout. The desired equations are supposed to be accurate enough yet simple to be fit for optimization algorithms. We have verified the proposed model in different scenarios against the actual parasitic capacitance extraction in different CMOS technologies, including 180, 90 and 65 nm. For simplification purpose, in all the cases, the geometries used are square metal blocks. For substrate capacitance, square metal blocks on Metal-1 layer with different sizes are tested. For the coupling capacitance experiments, two square blocks having different overlapping areas between Metal-1 and Metal-2 layers are used. Lateral capacitance was examined for different geometries on Metal-1 layer. These test cases include two Metal-1 squares with different sizes as well as distinct distance and displacement from each other. According to the experiments, our proposed modeling methodology is independent of choice of any layers. The extraction tool used for CMOS 180 nm process is Cadence Diva [71], whereas Mentor Graphics Calibre PEX [72] is used for CMOS 90 nm and 65 nm processes. The extracted numbers are used as the reference for comparison.

We used Cadence *SKILL* [73] scripts to generate thousands of test cases to verify the proposed model. The model was also implemented in *C++* program to generate the same test scenarios and the final results are compared and shown in the following tables. In the

following sub-sections, we also compare the proposed models for substrate, coupling and lateral capacitance with other published models against the extracted results.

6.2.1 Substrate Capacitance

A series of square metal blocks on Metal-1 layer were used as test structures. Experiments were repeated for different sizes in different technologies. Size of metal blocks is from 1 µm to 100 µm with a step of 3 µm. To demonstrate significance of this research, we first assessed importance of fringe component in substrate capacitance. Figure 1 depicts the fringe contribution in 3 technologies and Table 1 shows the average of fringe contribution in each technology. From Figure 1, it can be seen that the contribution is as high as 75% for small geometries in all technologies. Even for the geometry with single-edge size of 10 µm, the contribution of the fringe component in the total substrate capacitance is well above 20% in all of the three technologies.

Moreover, it is observed that the fringe component contributions within substrate capacitance in the 90nm and 65nm technologies are larger than the contribution in the 180 nm technology for the geometry with the size between 1µm and 100µm.

Table 1 – Average of fringe contribution in substrate capacitance

Technology	CMOS 180 nm	CMOS 90 nm	CMOS 65 nm
Fringe contribution	10.81%	12.83%	11.99%

Some published works, including Bansal's [11], Elmasry's [74], and Sakurai's methods [75] for fringe capacitance, and our model are compared against extracted values in Figure 24. As mentioned before, all the models were implemented in *C++* to generate the same test scenarios. For our proposed model, (11) is used to compute the fringe substrate capacitance and (7) is employed to calculate C_2. To tune the equation for each technology, we chose a few samples from each test scenario and finally concluded that the template equation with a constant value of *x/t* can fit to the extracted values. The values of *x/t* for different technologies are shown in Table 2. For a specific technology, its profile parameters (e.g., layer-to-substrate distance, layer thickness, and dielectric coefficients) directly determine the value of *x/t*. From Table 2, it is read that the values of *x/t* are increasing among 180 nm, 90 nm, and 65 nm technologies. It can be generally understood that the amount of the fringe capacitance component gets larger for the same size geometry when a smaller technology node is used.

Table 2 – Proper values obtained from the curve-fitting process

Technology	CMOS 180 nm	CMOS 90 nm	CMOS 65 nm
x/t	0	0.4	2.5

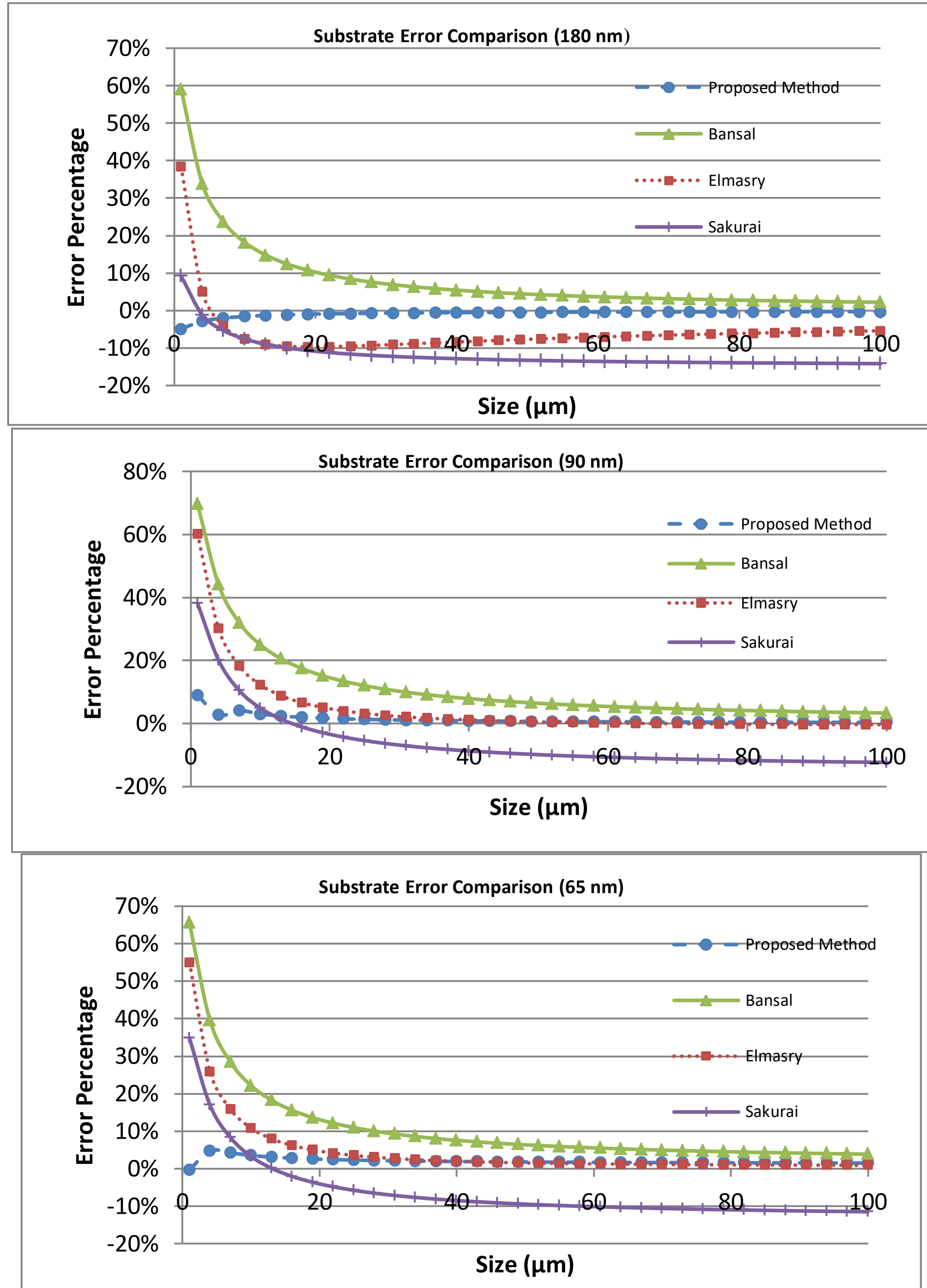


Figure 24 – Substrate capacitance error comparison for 3 different technologies.

As observed from Figure 24, all the previous methods involved in the comparison have large errors (up to 69%) for small geometries. The error and fringe contribution

reduce as the geometry size increases. On the other hand, our proposed model shows the errors are less than 9% (in CMOS 90 nm and CMOS 65 nm) and less than 5% (in CMOS 180 nm) for very small geometries, and approaching zero when the size is increasing. For large geometries, the errors for different methods are getting close because the overlap component, which is identical for all, becomes dominant and reduces the effect of fringe components.

In the modern technologies, size of interconnects or dummy fill is getting increasingly small. Therefore, the importance of the proposed method is obvious for the new nanometer technologies.

6.2.2 Coupling Capacitance

The test structure for coupling capacitance includes two square metal blocks on Metal-1 and Metal-2 layers. Experiments were repeated for different sizes and displacements. Figure 25 shows several examples of the test cases in the experiment. A blue square (with orthogonal lines) represents a block on Metal-1 layer and a green square (with vertical lines) shows a block on Metal-2 layer. Two sets of experiments were conducted to verify both equations proposed for coupling capacitance (i.e., (13) and (11) for the cases with certain overlap between two blocks and (16) for the cases without overlap between two blocks).

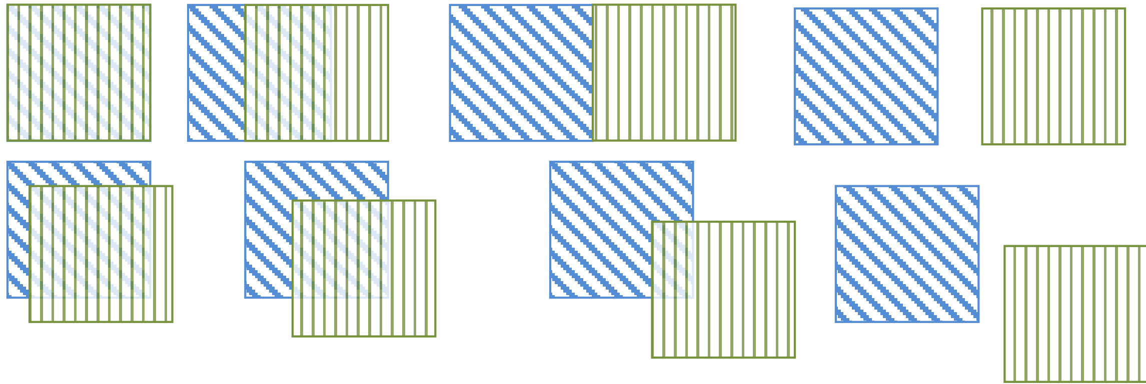


Figure 25 – Several examples of coupling capacitance test cases (blue squares with orthogonal shadowed lines are on Metal-1 layer and green ones with vertical shadowed lines are on Metal-2 layer).

In the first set of experiments, we considered Metal-1 block as fixed and shifted Metal-2 block to generate various test cases with different overlap area between two blocks, as well as with different displacement in the X and Y directions. Then this type of experiments was repeated for different block sizes. In the second set of experiments, to generate non-overlap test cases to verify (16), we considered different distance and displacement between two non-overlapping blocks. Then we repeated this type of experiments for different block sizes. The curve-fitting technique is applied to tune the template equations in terms of technology-dependent parameters, that is, x/t in (11) and β in (16). The tuning of β for coupling capacitance is done for several fixed geometry sizes, such as 1 µm and 10 µm. And then the derived factors along with the equation are applied to calculate the other test cases.

For CMOS 180 nm technology, (16) is not applicable because there is no extracted capacitance for the case where two metal blocks have no overlap. From the curve-fitting

process, x/t in (11) tends to be a constant. So it was selected to be 0.1 to reach the best fit. For CMOS 90 nm technology, after examining 4 samples, the ratio of x/t is obtained in terms of displacement ratio, which is the amount of displacement over size of the metal blocks under test. From the curve-fitting diagram outlined in Figure 26, the following equation is determined:

$$x/t = 1.672 \times displaceR - 0.454\,, \tag{37}$$

where *displaceR* is the displacement ratio between displacement and size of the corresponding side. To evaluate the approximation error, the value of R-squared for (37) is calculated to be 0.994, which exhibits the closeness of the linear model representation.

Figure 26 also shows that there is no need to use x/t for less than 25% displacement and more displacement needs to be compensated in capacitance calculation represented by larger x/t ratio. More than 100% displacement changes the equation to (16).

To tune (16) for this technology, β is also obtained from the curve-fitting process as follows

$$\beta = -0.0898 \times displace + 1.233, \tag{38}$$

where *displace* is the amount of displacement between two metal edges. In this case, commercial tools do not extract any coupling capacitance for distances more than 3 µm. Therefore, absolute value of displacement instead of relative value is used here. Increase in displacement reduces the coupling capacitance dramatically. Figure 27 shows the curve-fitting diagram of four samples (with geometry size of 10µm*10µm) when obtaining this equation. The value of R-squared is 0.995.

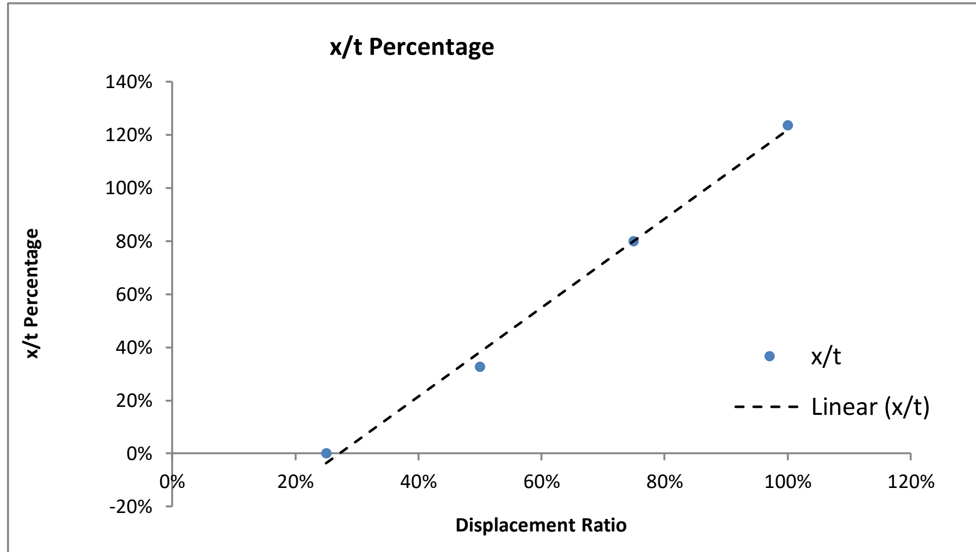


Figure 26 – Curve-fitting diagram of four samples to tune (11) for CMOS 90 nm technology

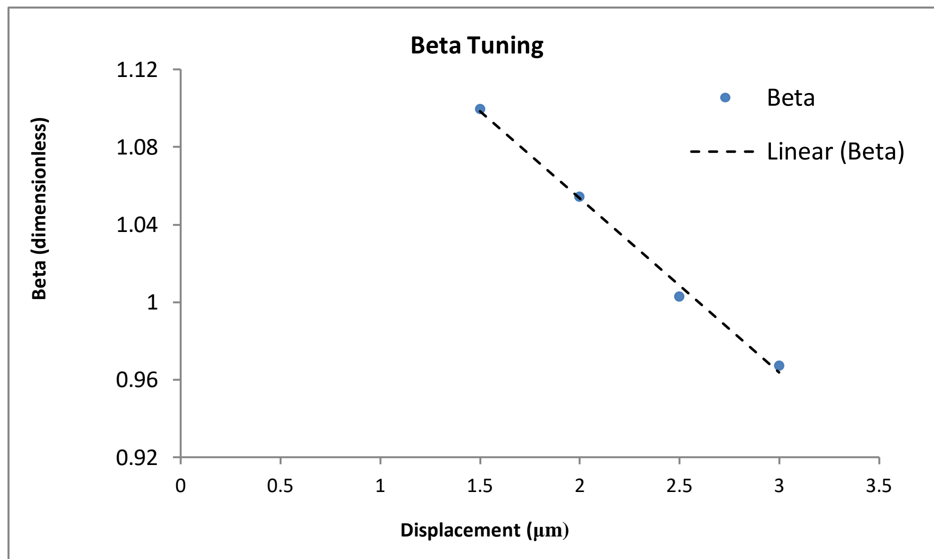


Figure 27 – Curve-fitting diagram of four samples to tune (16) for CMOS 90 nm technology.

For CMOS 65 nm technology, (11) is tuned based on the same principle with the curve fitting diagram shown in Figure 28. *x/t* is found dependent on displacement ratio linearly as the following:

$$x/t = 2.2 \times displaceR - 0.3 \quad (39)$$

To evaluate the approximation error, the value of R-squared for the derived equation is 0.988. For completeness, the β expression for (16) is provided below

$$\beta = -0.1382 \times displace + 2.0765 \quad (40)$$

The value of R-Squared is 0.97. The linear graph is shown in Figure 29.

After completing the curve-fitting process, we evaluated the developed model against the experimental results obtained from commercial extraction tools. The statistical analysis of the proposed method in comparison with two previous works ([11] and [15]) as well as two derivative streams is shown in Table 3. In [15], Kurokawa *et al.* introduced capacitance modeling in presence of dummy fill. In that work the authors only used basic overlap capacitance equation (i.e., (1)) in their modeling. In [11], conformal mapping is used to derive a set of analytic fringe capacitance models. Considering the significant contribution of fringe component in the parasitic capacitance for the small-size geometry, we differentiate the test scenarios with small-size and big-size categories. Geometry size below 10 µm is grouped as small-size category, whereas geometry size over 10 µm is considered as large-size category.

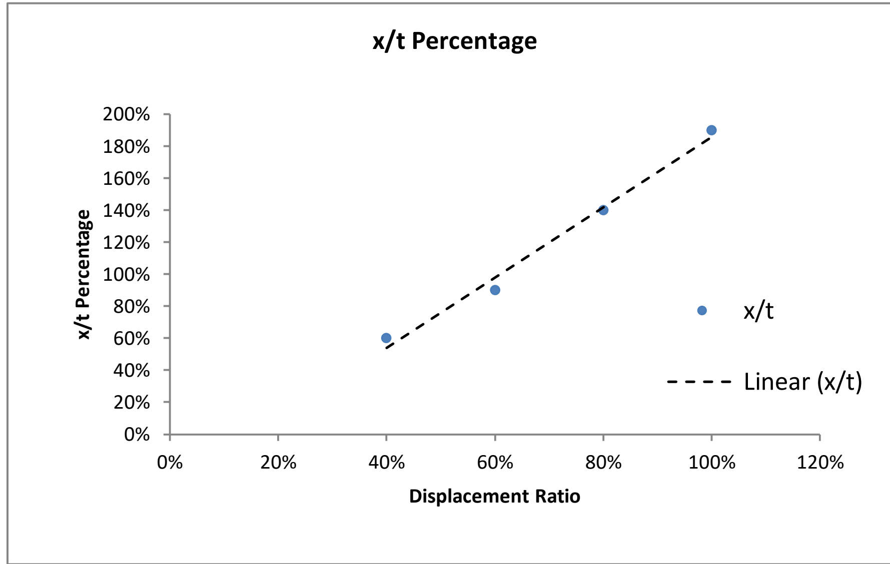


Figure 28 – Curve-fitting diagram of four samples to tune (11) for CMOS 65 nm technology.

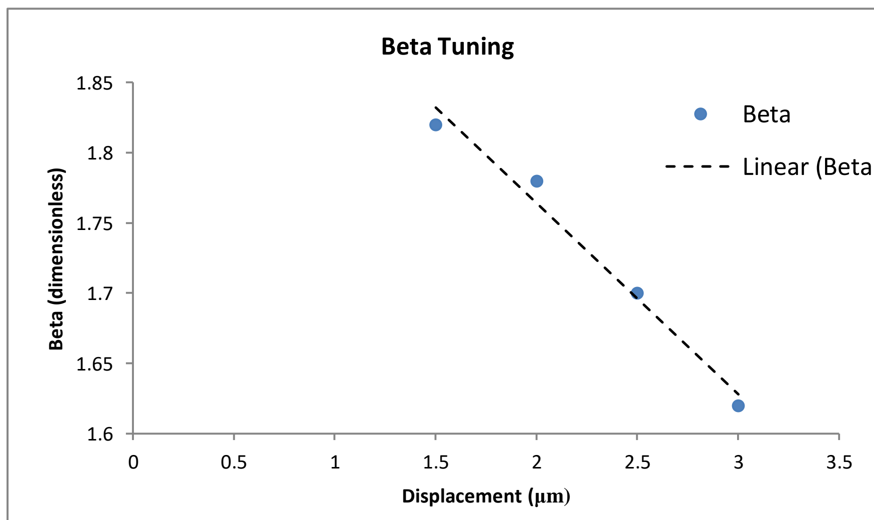


Figure 29 – Curve-fitting diagram of four samples (with geometry size of 10μm*10μm) to tune (16) for CMOS 65 nm technology.

Table 3 – Comparison of coupling capacitance. All numbers represent error percentage. (*NA* means not applicable, and *NF* means not feasible)

Technology			CMOS 180 nm		CMOS 90 nm		CMOS 65 nm	
Test scenario			Overlap Coupling	Non-overlap coupling	Overlap Coupling	Non-overlap coupling	Overlap Coupling	Non-overlap coupling
Kurokawa [15]	Small	Average	30.23	NA	59.03	NF	47.61	NF
		Std Dev	27.6	NA	30.98	NF	34.32	NF
		Max	80.03	NA	100+	NF	100+	NF
	Big	Average	9.27	NA	35.1	NF	27.79	NF
		Std Dev	10.93	NA	41.76	NF	40.74	NF
		Max	57.86	NA	100+	NF	100+	NF
Bansal [11]	Small	Average	6.87	NA	13.19	3.68	15.47	45.7
		Std Dev	6.23	NA	11.88	9.14	13.8	25.06
		Max	23.13	NA	35.28	11.52	43.21	67.73
	Big	Average	1.95	NA	10.38	16.03	11.56	48.4
		Std Dev	2.32	NA	13.8	11.79	17.79	28.17
		Max	12.15	NA	35.01	31.64	43.21	67.72
Unified-EMCF	Small	Average	0.98	NA	5.84	4.34	4.88	8.88
		Std Dev	0.93	NA	6.64	3.75	8.16	13.76
		Max	4.37	NA	31.69	9.11	38.36	46.37
	Big	Average	0.2	NA	1.66	4.34	0.51	2.01
		Std Dev	0.31	NA	1.25	3.75	0.56	1.77
		Max	1.66	NA	6.51	9.07	3.88	5.03
Segment-EMCF	Small	Average	0.98	NA	2.9	4.34	2.08	2.06
		Std Dev	0.93	NA	2.06	3.75	1.72	1.74
		Max	4.37	NA	8.43	9.11	7.23	5.03
	Big	Average	0.2	NA	1.66	4.34	0.51	2.01
		Std Dev	0.31	NA	1.25	3.75	0.56	1.77
		Max	1.66	NA	6.51	9.07	3.88	5.03
Linear Unified-EMCF	Small	Average	2.5	NA	4.66	4.34	7.44	2.06
		Std Dev	2.31	NA	3.66	3.75	5.55	1.74
		Max	6.8	NA	14.2	9.11	16.88	5.03
	Big	Average	0.67	NA	3.06	4.34	4.2	2.01
		Std Dev	0.9	NA	4.1	3.75	6.31	1.77
		Max	4.87	NA	10.57	9.07	15.35	5.03

Moreover, two streams were investigated for technology-dependent parameter determination with the aid of the curve-fitting technique. Unified-EMCF is our proposed EM-based modeling method supported by the curve-fitting technique for any size of geometries. Thus, in Unified-EMCF, one identical equation, which was obtained by running curve fitting on 10μm-size geometry, was used for both small (less than 10μm) and big (equal or more than 10μm) geometry in the latter verification stage. Segment-EMCF is also based on the proposed EM-based modeling method, but one set of technology-dependent parameters (e.g., x/t and β) is tuned specifically for small size (less than 10μm) geometries, and the other set of technology-dependent parameters for large size (equal or more than 10μm) geometries. Finally, two sets of equations with different technology-dependent parameters are used in Segment-EMCF. Our experimental results show that each equation fits the designated size range well, but fails to offer promising accuracy to the other size range.

Besides, in the comparison, we also include another derivative, Linear-EMCF, which uses (18) to replace logarithm function with linear function. Each method was applied to the different test scenarios in CMOS 180 nm, 90 nm, and 65 nm technologies and the errors between estimated values and extracted results were calculated. In Table 3, error averages, error standard deviations, and maximum error for each modeling method are listed. As mentioned in the previous section, (16) is not applicable for CMOS 180 nm technology, because there is no extracted coupling capacitance when there is no geometry overlap. Reference [15] is not able to provide non-overlap coupling capacitance. From the reported experimental results in Table 3, it can be read that [15] fails to provide

satisfactory estimation. Because it only uses overlap component to model interconnect capacitances, neglected fringe capacitance contribution brings forth huge deviation from the real case. For the coupling capacitance estimation, statistically its error averages and standard deviations are generally doubly or triply larger than any derivatives of our proposed method.

It is also observed that Segment-EMCF outperforms Unified-EMCF, especially for the small-size geometries. This is obvious because more effort has been put on the tuning of the technology-dependent parameters for small-size geometries. The general average accuracy of Segment-EMCF can reach well below 5% for all the situations. Therefore, this is the recommended method for high-accuracy estimation.

If there is a serious concern over computation time, Linear-EMCF can be used as a trade-off between accuracy and CPU time. This method has slightly degraded accuracy (up to 8% deviation from the extracted results on average), but with full linear expression in the analytic equations. All the presented curve-fitting equations above are obtained for the Unified-EMCF method. The equations for Segment-EMCF or Linear-EMCF can be done by following the same principle.

As shown in Table 3, the maximum errors for Segment-EMCF are all under 9.72% compared to the extracted values, whereas the maximum errors increase to 46.37% for Unified-EMCF. Although the occurrence rate of maximum errors is not high (normally less than 0.1% according to our experiments), this may impose a concern to certain users who really care about accuracy of any single case. In this situation, such users may consider to only utilize Segment-EMCF, which needs a little more effort in the curve-

fitting parameter characterization stage. For Linear-EMCF, the maximum errors reach 16.88%, which exhibits utilizing this modelling method for a faster estimation may have to tolerate certain accuracy degradation. Nevertheless, considering the other previously published works, whose maximum errors are normally very huge, even reaching 100% or more (marked as "100+" for this situation in Table 3), our proposed modeling methods are actually providing much more favorable accuracy.

To verify our coupling capacitance modeling, we applied the Monte-Carlo method in the CMOS 65 nm technology. We tested 5,000 samples, each of which includes two metal blocks with random size and displacement in X and Y directions from each other. The test results are shown in Table 4. And the error occurrence histogram is shown in Figure 30, where "E" stands for error value. As can be seen, the average error of our method is only 1.13% with standard deviation of 1.32%. Moreover, the error is less than 3% for more than 90.7% of the entire test cases.

Table 4 – Statistical results from the Monte Carlo verification for overlapping coupling capacitance in the CMOS 65 nm technology.

Average Error	**Standard Deviation**	**Maximum Error**
1.13%	1.32%	8.53%

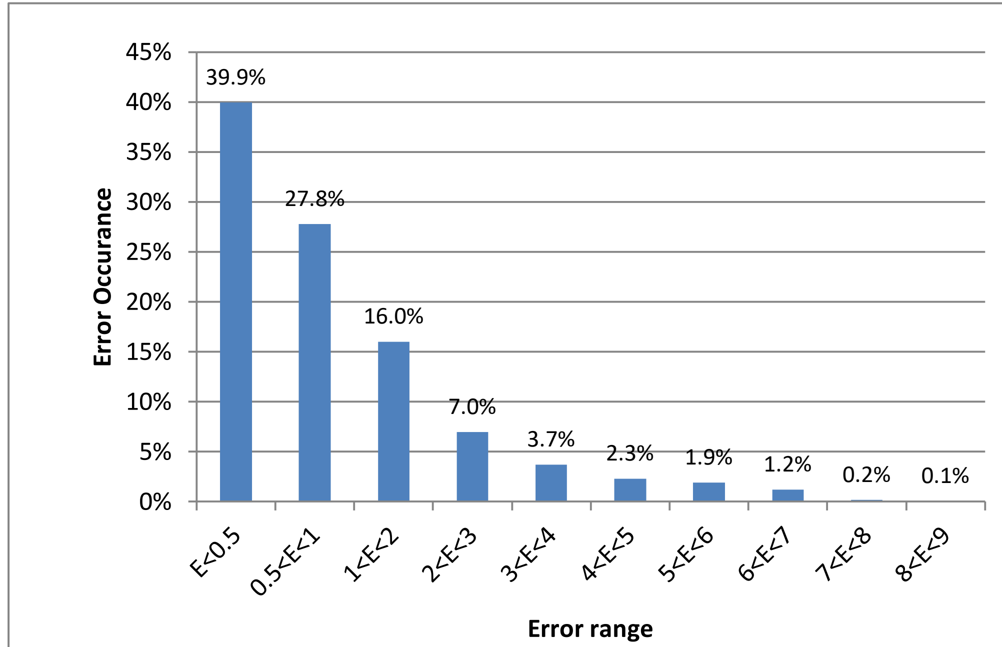


Figure 30 – Histogram of error occurrences in the Monte-Carlo verification for coupling capacitance in the CMOS 65 nm technology.

6.2.3 Lateral Capacitance

As mentioned in Section 3.3.3, to calculate lateral capacitance, the same template equations used for the computation of coupling capacitance can be employed. Effective width was used as a technology-dependent parameter to tune the equations. The expressions of effective width as well as error estimation R-squared for the lateral capacitance component in different CMOS technology can be found in Table 5. All the expressions were obtained by using the curve-fitting technique as discussed for the other capacitance types. After applying the derived parameters to the proposed equations, we compared the calculated lateral capacitance against the extracted results.

The scenarios to test lateral capacitance modeling are very similar to what have been defined to verify the non-overlapping coupling capacitance modeling. But instead of using one Metal-2 block, both metal blocks were drawn on Metal-1 layer. Figure 31 shows two lateral capacitance test cases. Two Metal-1 blocks feature different sizes, distance (i.e., displacement in the X direction), and displacement in the Y direction. In test case (a), two blocks have no displacement in the Y direction, but they may have different distances. In test case (b), two blocks have displacement in both X and Y directions. Thus, two blocks may have overlap in their confronting sidewalls. The amount of displacement in Y direction can change the facing sidewalls whereas the distance changes the space between two facing sidewalls. These two parameters have significant effects on lateral fringe capacitance.

Table 5 – Lateral capacitance effective width (w_e) expressions and corresponding error estimations (R-squared) for different technologies (*dx* is the displacement in the corresponding direction)

Technology	$\mathbf{w_e}$	**R-squared**
CMOS 180 nm	0.14	1
CMOS 90 nm	$-0.0895 \cdot dx + 0.44$	0.98
CMOS 65 nm	$-0.06981 \cdot dx+0.3309$	0.96

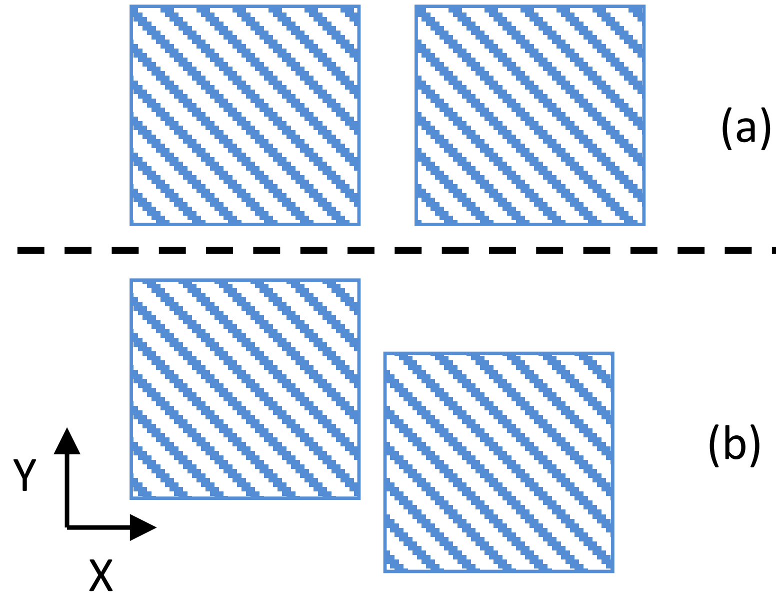


Figure 31 – Two examples of lateral capacitance test case (top view).

In the test, the size of the blocks varied from 1 µm to 100 µm. The horizontal distance between two blocks varied from 0.5 µm to 3 µm. As mentioned before, there is no extracted capacitance for distance more than 3 µm. Therefore, we restricted the range of horizontal distance within 3 µm. The displacement in the Y direction varied from 0 to the full size of the block. In this way, hundreds of different test cases could be generated to verify the model. Extracted capacitance by using commercial tools was compared to the proposed model and the results are presented in Table 6.

Reference [15] only uses overlap capacitance; however reference [11] is not able to provide a solution for lateral capacitance. Our Segment-EMCF method provides more accurate results with less than 3% average error and maximum error is less than 10%.

Table 6 – Comparison of lateral capacitance. All numbers represent error percentage. (*NA* means not applicable, and *NF* means not feasible)

Technology			CMOS 180 nm	CMOS 90 nm	CMOS 65 nm
Kurokawa [15]	Small	Average	3.69	47.78	46.96
		Std Dev	9.11	28.32	27.89
		Max	26.03	71.43	71.79
	Big	Average	16.92	47.92	46.85
		Std Dev	12.46	28.31	27.73
		Max	26.03	71.47	71.45
Bansal [11]	Small	Average	NF	NF	NF
		Std Dev	NF	NF	NF
		Max	NF	NF	NF
	Big	Average	NF	NF	NF
		Std Dev	NF	NF	NF
		Max	NF	NF	NF
Segment-EMCF	Small	Average	0.27	2.47	2.75
		Std Dev	1.05	2.5	2.66
		Max	6.03	8.43	8.58
	Big	Average	0.8	3.3	2.7
		Std Dev	1.25	3	2.89
		Max	5.76	9.72	9.49

6.2.4 Analysis of Computation Time

To compare the computation time of different methods, we picked the main equation of each method and included it in an iterative loop for running 100,000,000 times. The experimental results are reported in Table 7. The program was implemented in *C++* and executed on a 64-bit server with 48GB memory, six-core Intel Xeon CPU @ 2.67GHz, and Red Hat Enterprise Linux Server release 5.10 operating system.

Unified-EMCF and Segment-EMCF have the same nature as they both have a logarithmic function. Therefore, we used similar equation for them and their computation time is identical. As shown in Table 7, by replacing the logarithmic function with a linear one, the computation time of Linear EMCF is at least three times faster than Unified-EMCF or Segment-EMCF, only 1.45 seconds for 100 million runs. The computation time of the other methods (e.g., Bansal's, Sakurai's, and Elmasry's) is over 1.31 times longer than that of Unified-EMCF or Segment-EMCF.

As a comparison with commercial tools, we also executed Calibre PEX and Diva commands under Linux shell. Note that these executable commands may include some hidden tasks besides parasitic capacitance extraction since their source code cannot be accessed. Each of these tools run only once and the computation time was measured on the same server. It took 39 seconds for Calibre PEX to extract parasitic capacitance, whereas it took 4 seconds for Diva to report parasitic capacitance for a layout.

Table 7 – Computation time comparison

Method	Computation Time (Sec.)
Unified-EMCF	5.313
Segment-EMCF	5.313
Linear EMCF	1.451
Bansal [11]	6.97
Sakurai [75]	9.381
Elmasry [74]	10.243

Obviously, to include Calibre PEX [72] or Diva [71] in a big loop of optimization is no doubt impractical. However, the modeling methods proposed in this dissertation fit very well in the situation above.

Besides that, to gain a general time complexity comparison, we also summarize the applied arithmetic operators appearing in different methods in Table 8. The proposed method, especially its linear derivative Linear-EMCF, is the simplest in expression and easy to use in the loops of optimization. To the best of our knowledge, this proposed linear model is computationally cheapest among the previously published works and commercial tools that are able to handle fringe capacitance. Considering Table 7 and Table 8 as well as the accuracy comparison in the previous sub-sections, the users can determine the best method for them to use with a balanced consideration between accuracy and efficiency.

Table 8 – Complexity comparison among the methods that are able to handle fringe capacitance (*d* stands for the distance).

Methods	Operator	Linear	Polynomial (order)	Square root	Logarithm	Logarithm squared	Exponential
Unified-EMCF	Substrate				×		
	Coupling				×		
	Lateral				×		
Linear-EMCF	Substrate	×					
	Coupling	×					
	Lateral	×					
Bansal [11]	Substrate			×	×		
	Coupling			×	×		
Elmasry [74]	Substrate					×	
Sakurai [75]	Substrate		× (0.22)				
Comm. Tool1	Substrate		× (0.02*d)				×
	Coupling		× (1.42)				×
	Lateral		× (0.9)				×
Comm. Tool2	Substrate		× (2)				
	Coupling		× (2)				
	Lateral		× (5)				

6.2.5 Case Study

To evaluate our modeling in practice, we also made a case study of a simple but complete layout in CMOS 90 nm technology and compared the results of the proposed analytic modeling against extracted values obtained from an industry-strength commercial tool, Calibre PEX [72]. Figure 32 shows the layout in this case study, which includes 3 nets and 2 dummy blocks. All the nets consist of three metal layers (i.e.,

Metal-1, Metal-2 and Metal-3) and two dummy blocks have been drawn on Metal-1 layer. The case study is considered as an example of a portion of a layout.

First of all, substrate capacitance for each net and coupling capacitance between each pair of nets and dummy blocks were extracted by Calibre PEX. By using our proposed capacitance modelling, the capacitance of each net and coupling capacitance between each pair of nets and dummy blocks were calculated as well. As the tiles can be located in different layers, to calculate the coupling capacitance the geometric situation of each individual fringe capacitance component is investigated and based on that the proper equation and model is used. Table 9 shows the comparison of the extracted results and our calculations. For each net, the substrate capacitance derived from our proposed method was compared against the extracted ones with error calculated in reference to the extracted value. For each pair of nets, if the coupling/lateral capacitance exists, the coupling/lateral capacitance comparison between our proposed approach and the extracted value was conducted.

As a matter of fact, in each case the final obtained coupling/lateral capacitance consists of many overlap and fringe capacitance components calculated individually based on their geometrical situations. The equivalent capacitance was calculated by summarizing all those components. As listed in Table 9, the reported errors of our proposed analytic modeling for the substrate and coupling/lateral capacitances in this case study are in the range of ±7% with reference to the extracted results from the industry mainstream extraction tool Calibre PEX.

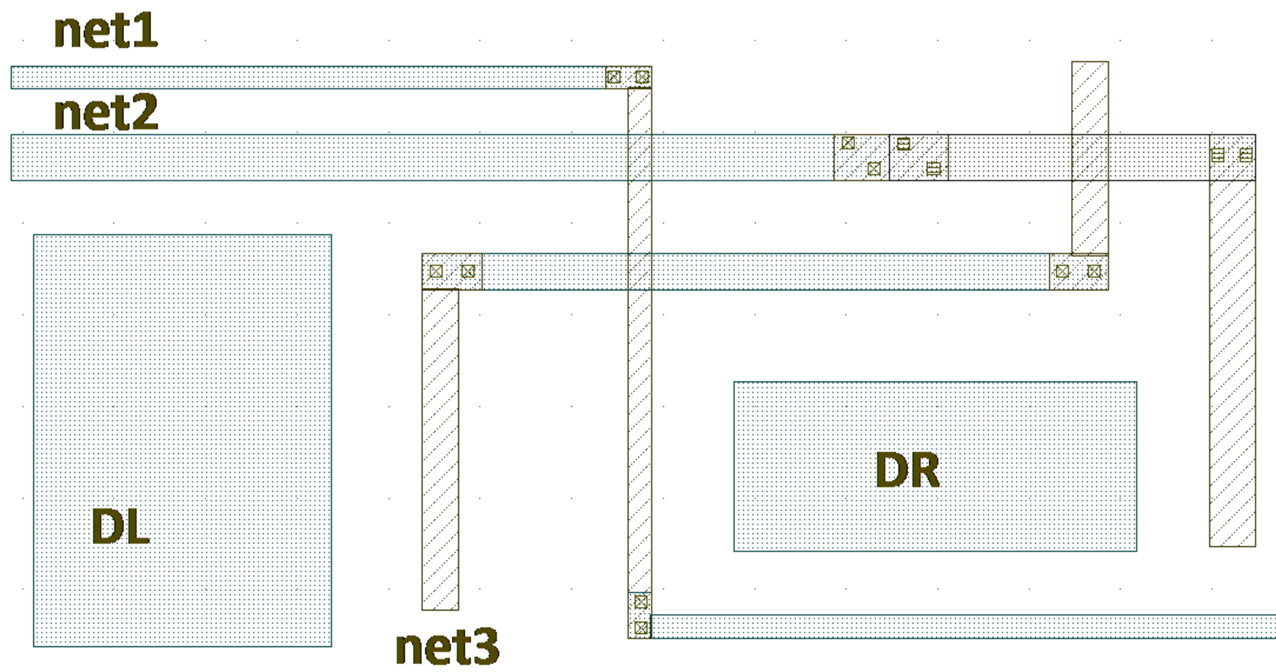


Figure 32 – Case study of the capacitance modeling

Table 9 – Comparison of substrate and coupling/lateral capacitance in the case study

			Substrate Capacitance	Coupling/Lateral Capacitance				
				1	2	3	4	5
				net1	net2	net3	DL	DR
1	net1	Extracted	1.38285E-15		3.44820E-16	1.01701E-16	no coupling capacitance	1.65085E-16
		Calculated	1.47093E-15		3.66525E-16	9.67812E-17		1.61053E-16
		Error	-6.37%		-6.29%	4.84%		2.44%
2	net2	Extracted	1.30351E-15	3.44820E-16		2.70103E-16	1.18184E-16	3.79546E-17
		Calculated	1.31899E-15	3.66525E-16		2.84851E-16	1.09949E-16	3.58022E-17
		Error	-1.19%	-6.29%		-5.46%	6.97%	5.67%
3	net3	Extracted	9.85878E-16	1.01701E-16	2.70103E-16		6.51117E-17	8.78580E-17
		Calculated	9.52202E-16	9.67812E-17	2.84851E-16		6.51177E-17	9.36660E-17
		Error	3.42%	4.84%	-5.46%		-0.01%	-6.61%
4	DL	Extracted	1.28675E-15	no coupling capacitance	1.18184E-16	6.51117E-17		no coupling capacitance
		Calculated	1.26561E-15		1.09949E-16	6.51177E-17		
		Error	1.64%		6.97%	-0.01%		
5	DR	Extracted	7.42629E-16	1.65085E-16	3.79546E-17	8.78580E-17	no coupling capacitance	
		Calculated	7.44290E-16	1.61053E-16	3.58022E-17	9.36660E-17		
		Error	-0.22%	2.44%	5.67%	-6.61%		

6.2.6 Extension to More Advanced Technologies

As an extension, the proposed modeling methodology was further applied to CMOS 45 nm technology. We conducted experiments on substrate, coupling, and lateral capacitance. It was found that our proposed modeling methodology is very applicable to this advanced technology node. By using the curve-fitting technique, the proper value of *x/t* for substrate capacitance in the CMOS 45 nm technology was found to be 3. If applying it to a big set of test cases, the average error percentage is 1.35% with standard deviation of 2%, and the maximum error percentage is 9.56%. Considering the *x/t* values in the other three technologies reported in Table 5, the trend of increasing *x/t* for a smaller-node technology still holds. However, the *x/t* value change between 65 nm and 45 nm technologies is not as significant as that between 90 nm and 65 nm technologies. This is because other technology profile parameters, such as dielectric coefficients and ratio between layer-to-substrate distance and layer thickness, may also influence the value of *x/t* besides the value of layer-to-substrate distance. In our experiments, it was found that the dielectric coefficients of the 45 nm technology are quite different from those in the 180 nm, 90 nm, or 65 nm technologies. We expect this phenomenon to be continuing in the advanced technologies. Nevertheless, this will not make any negative impact on our proposed modeling methodology as *x/t* just reflects the technology nature and keeps unchanged for any geometry once being first characterized for a specific technology.

Similarly, we used the curve-fitting technique to characterize the overlapping coupling capacitance for the 45 nm technology and obtained the following technology-dependent parameter:

$$x/t = 0.615 \times displaceR + 0.549 \quad (41)$$

If applying it to a big set of test cases, the average error percentage is 1.81% with standard deviation of 1.45%, and the maximum error percentage is 4.71%.

6.3 Experimental Result of Density-Uniformity-Aware Analog Layout Retargeting

In this section, the proposed approach of density-uniformity-aware analog layout retargeting with parasitic capacitance consideration is tested and verified. In the first sub-section, the importance of using the proposed scheme in comparison with the traditional dummy fill insertion method is shown by running a set of simple test scenarios. Then the proposed method is tested in practice for a few test layouts and the verification and simulation results are presented for different CMOS technologies. The result shows that the developed scheme can improve density uniformity of an analog layout up to 80% towards the ideal layout with less degradation due to less number of inserted dummy features.

6.3.1 Experimental Comparison of Various Density Control Schemes

Our proposed method has two major advantages in comparison with the previous works. First of all, for compact layouts where there is no enough room for dummy fill to be inserted among interconnects without violating design rules, the interconnect widening

operation can effectively improve layout density uniformity by slightly widening the interconnects. The area shift operation can also beneficially rearrange the functional geometries to reduce the density variation throughout the layout. Secondly, dummy fill among functional geometries in the layout would incur more parasitic capacitance than the interconnect widening or area shift operation, which may affect the circuit performance.

One set of simple experiments have been conducted in order to lead us to seriously think about such difference. As shown in Figure 33, the increase in parasitic laterally-coupling capacitance and parasitic substrate capacitance between two Metal-1 tiles is investigated after adding dummy fill and widening interconnect sizes. Figure 33(a) is the reference (including two Metal-1 blocks with the size of 2μm×10μm and 3μm away from each other) to define a default laterally-coupling capacitance between two metal tiles and sum of substrate capacitances. Without loss of generality, here we assume the density constraint is to add 25% of the existing area in between two metal tiles. This can be achieved by several different ways as depicted in Figure 33(b), (c) and (d). In Figure 33(b) and (c), both are implemented by using dummy fill to increase feature coverage.

Figure 33(b) includes one dummy fill as a narrow tile in the middle, while Figure 33(c) adds one thicker square in the center but with less overlap from the two surrounding parallel interconnect tiles. In Figure 33(d), the added area is obtained by widening two facing metal tiles, which effectively causes a small amount of distance decrease between the two inner edges.

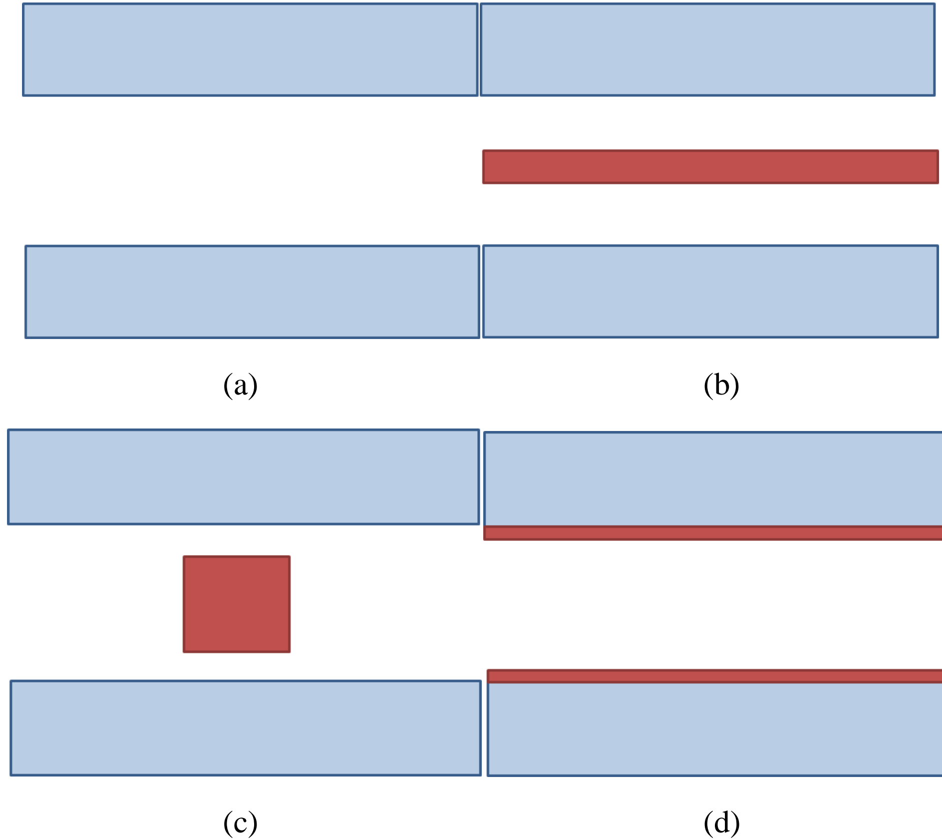


Figure 33 – Different test cases to investigate the effects of dummy fill insertion and interconnect widening on the laterally-coupling capacitance. (a) The reference test case, (b) dummy scheme1 test case, (c) dummy scheme 2 test case, (d) interconnect widening test case.

The amounts of increase in laterally-coupling capacitance and substrate capacitance in the three different cases of Figure 33(b)-(d) are recorded for three different technologies as listed in Table 10.

One can clearly observe that our proposed interconnect widening scheme can offer at least 24% - 47% less laterally-coupling capacitance and 3% - 12% less substrate capacitance compared to the dummy fill insertion scheme in the 180 nm, 90 nm, and 65 nm CMOS technologies. As for our proposed area shift scheme, if it is possible to shift

any geometries from excess-density regions to shortage-density regions, the size of such geometries should be kept intact while the intermediate distance can be the same or increased, which leads to no worry of parasitic capacitance change at least or reduction of the laterally-coupling capacitance.

Table 10 – Laterally-Coupling Capacitance (top) and substrate capacitance (bottom) increase comparison for test cases of Figure 33 (b)-(d) in reference to the original test case of Figure 33 (a)

Coupling/Lateral Capacitance Comparison			
Test case / Technology	Dummy scheme 1 (Figure 33(b))	Dummy scheme 2 (Figure 33(c))	Interconnect widening (Figure 33(d))
180 nm	99%	162%	75%
90 nm	129%	189%	88%
65 nm	121%	193%	74%

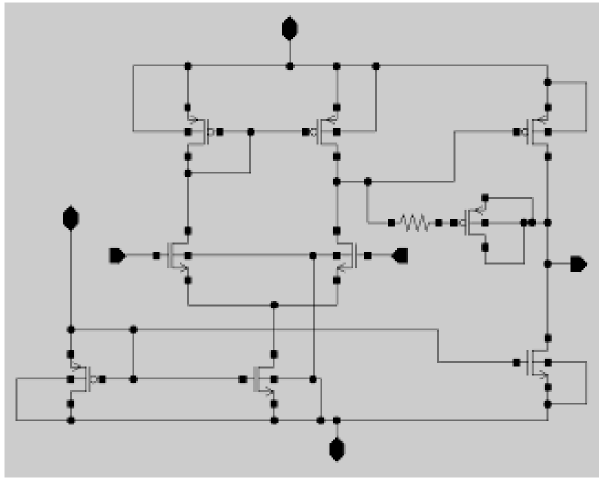
Substrate Capacitance Comparison			
Test case / Technology	Dummy scheme 1 (Figure 33(b))	Dummy scheme 2 (Figure 33(c))	Interconnect widening (Figure 33(d))
180 nm	35%	27%	15%
90 nm	20%	16%	13%
65 nm	23%	18%	15%

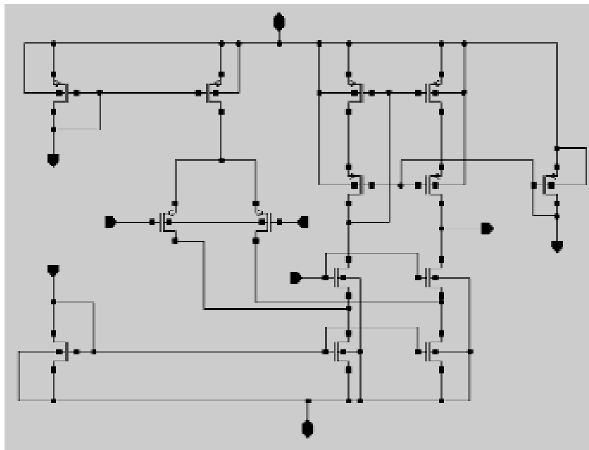
6.3.2 Density-Uniformity-Aware Analog Layout Retargeting Verification

We have developed our density-uniformity-aware analog layout retargeting platform in *C++*. To verify the performance of our proposed method, we used three analog circuits in our experiments: a two-stage operational amplifier (Opamp), a folded-Cascode Opamp, and a 5-bit flash analog-to-digital converter (ADC). The 5-bit ADC uses 32 modules of analog comparator along with flip-flops and digital priority encoders. In our experiments, the analog core of the 5-bit ADC was tested. All of the circuits were

retargeted for technology and specification update. Figure 34 shows the schematics of two Opamps and 5-bit ADC analog comparator circuits under test. Figure 35 shows the Opamps original layouts, which were well designed in the 0.25 μm CMOS technology. Figure 36 also shows the analog part of the 5-bit ADC layout, which includes 32 modules of comparators. To demonstrate the efficacy of our proposed LP and graph-based optimization method as explained in Chapters 4 and 5, we also implemented other approaches that were published in the literature. For the comparison purpose, we also studied different combination among area shift, interconnect widening, and dummy fill insertion operations.

An original layout is loaded in the retargeting platform along with the original technology design rule constraints, target technology parameters and new transistor sizes (based on the design specifications). Then after generating resized layout by running retargeting process, the layout density is analyzed by using the fixed dissection method. In the next stage, the LP formulation explained in Section 4.4.2 suggests the optimum solution to improve density uniformity based on the selected approach. Since the optimization algorithms to control layer density by dummy fill insertion vary, we implemented 3 different optimization schemes including linear programming [48], Monte-Carlo [51] and iterative Monte-Carlo [52], each of which suggests unique optimum solution to fill resized layout by dummy fill blocks. We implemented our proposed methodology with different combination options so that the users can choose area shift operation, interconnect widening operation, or their combination along with dummy fill insertion.





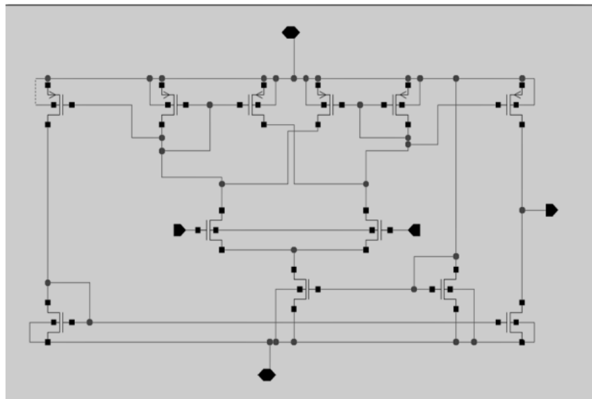


Figure 34 – Two-stage Opamp schematic (top) and folded-cascode Opamp schematic (middle), and 5-bit ADC analog comparator (bottom)

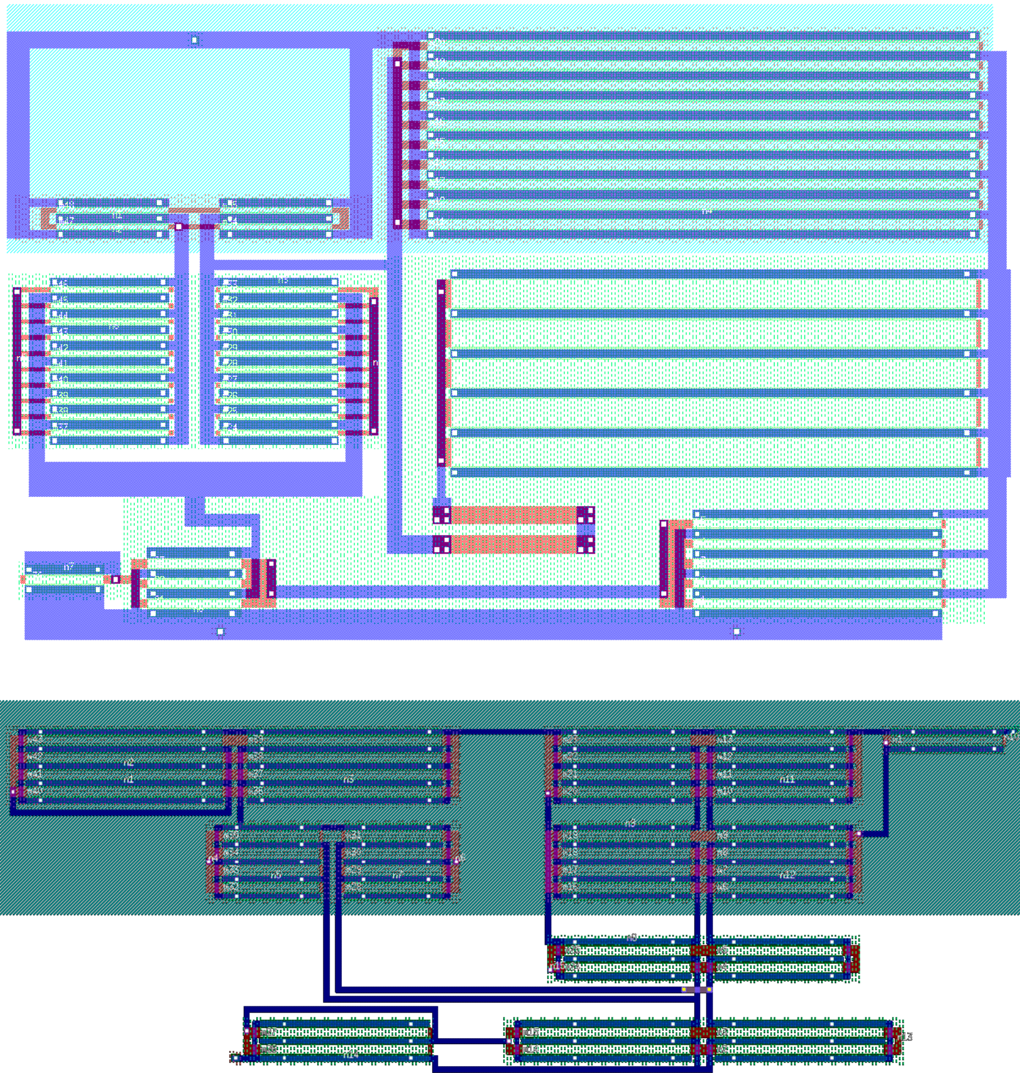


Figure 35 – Two-stage Opamp original layout (top) and folded-cascode Opamp original layout (bottom)

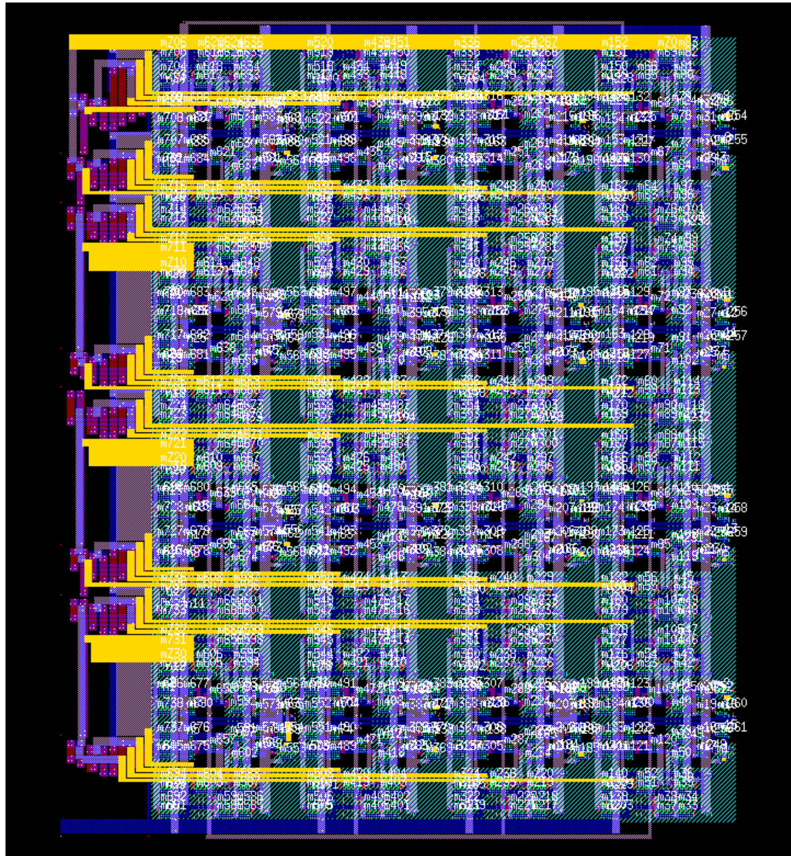


Figure 36 – the analog part of the 5-bit ADC layout

Table 11, Table 12 and Table 13 show the comparison of density variations of the two Opamps example circuits in the CMOS 180 nm, 90 nm and 65 nm technologies, while Table 14 shows the test results of density variation for the 5-bit ADC in the CMOS 180 nm technology. The first three rows of each test case are attributed to the methods presented in [48], [51], and [52] for dummy fill (DF) insertion, respectively. Since the optimization algorithms utilized in these three methods are different, the resulting layouts from our initial layout retargeting process are filled with dummy blocks based on the corresponding DF optimization algorithms. For the first one *DF-MC*, Monte-Carlo (MC)

optimization scheme is used to find the optimum solution. In the second method *DF-IMC*, a better version of Monte-Carlo, i.e., iterative Monte-Carlo (IMC), is utilized to obtain the optimal filling solution. And the third row presents the results of using linear programming (LP) approach for filling location and amount optimization. In the last five rows of each test case, our proposed methodology with different combination options are presented. In the fourth row (marked as IW standing for interconnect widening operation) of each test circuit in all the four tables, the optimization was handled by the interconnect widening operation only, while in the fifth row (marked as AS where AS stands for area shift operation) of each test circuit, the optimization was conducted by the area shift operation. In the sixth and seventh rows area shift operation was performed and then interconnect widening (marked as AS+IW) or dummy fill insertion (marked as AS+DF) was conducted. In the last row of each test circuit, AS+IW+DF stands for the integral operation of the area shift as the first stage, the interconnect widening as the second stage, and dummy fill insertion as the last stage. To simplify the presentation, only the test result of different methods on Metal-1 layer is reported.

Table 11 – Comparison of density variation of two test layouts for CMOS 180 nm technology

Process Technology		CMOS 180 nm					
Test Layouts	Density Control Method	Window Variation (%)		Improvement (%)	number of fill	Fill Reduction (%)	Run-time (Sec.)
		Before	After				
Two-Stage Opamp	DF-MC	11.7	7.7	-34%	507	--	0.88
	DF-IMC	11.7	3.8	-68%	311	-39%	0.89
	DF-LP	11.7	4.3	-63%	349	-31%	0.86
	IW	11.7	8.1	-31%	0	-100%	2.89
	AS	11.7	9.3	-21%	0	-100%	2.9
	AS+IW	11.7	6.9	-41%	0	-100%	2.94
	AS+DF	11.7	3.7	-68%	277	-45%	2.82
	AS+IW+DF	11.7	3	-74%	153	-70%	2.99
Folded-Cascode Opamp	DF-MC	12.5	5.2	-58%	463	--	0.69
	DF-IMC	12.5	4.4	-65%	328	-29%	0.71
	DF-LP	12.5	4.9	-61%	397	-14%	0.75
	IW	12.5	8	-36%	0	-100%	2.21
	AS	12.5	10.7	-14%	0	-100%	2.23
	AS+IW	12.5	6.7	-46%	0	-100%	2.27
	AS+DF	12.5	4.5	-64%	229	-51%	2.21
	AS+IW+DF	12.5	3.7	-70%	228	-51%	2.3

Table 12 – Comparison of density variation of two test layouts for CMOS 90 nm technology

Process Technology		**CMOS 90 nm**					
Test Layouts	Density Control Method	Window Variation (%)		Improvement (%)	number of fill	Fill Reduction (%)	Run-time (Sec.)
		Before	After				
Two-Stage Opamp	DF-MC	7.2	5.7	-21%	833	--	0.92
	DF-IMC	7.2	5.2	-28%	551	-34%	0.93
	DF-LP	7.2	3.9	-46%	691	-17%	0.963
	IW	7.2	4.6	-36%	0	-100%	2.83
	AS	7.2	4.8	-33%	0	-100%	2.83
	AS+IW	7.2	4.4	-39%	0	-100%	2.84
	AS+DF	7.2	3.4	-53%	530	-36%	2.86
	AS+IW+DF	7.2	2.7	-63%	477	-43%	2.9
Folded-Cascode Opamp	DF-MC	17.9	9.3	-48%	487	--	0.61
	DF-IMC	17.9	7.1	-60%	304	-38%	0.66
	DF-LP	17.9	6	-66%	419	-14%	0.65
	IW	17.9	13.4	-25%	0	-100%	2.13
	AS	17.9	12.4	-31%	0	-100%	2.12
	AS+IW	17.9	12.7	-29%	0	-100%	2.16
	AS+DF	17.9	5.1	-72%	316	-35%	2.15
	AS+IW+DF	17.9	4.4	-75%	270	-45%	2.21

Table 13 – Comparison of density variation of two test layouts for CMOS 65 nm technology

Process Technology		**CMOS 65nm**					
Test Layouts	Density Control Method	Window Variation (%)		Improvement (%)	number of fill	Fill Reduction (%)	Run-time (Sec.)
		Before	After				
Two-Stage Opamp	DF-MC	9.1	8.1	-11%	829	--	0.96
	DF-IMC	9.1	5.9	-35%	653	-21%	0.92
	DF-LP	9.1	6	-34%	722	-13%	0.97
	IW	9.1	6.6	-27%	0	-100%	3.07
	AS	9.1	6.8	-25%	0	-100%	3.06
	AS+IW	9.1	8.3	-9%	0	-100%	3.09
	AS+DF	9.1	5.5	-40%	472	-43%	3.11
	AS+IW+DF	9.1	3.3	-64%	642	-23%	3.18
Folded-Cascode Opamp	DF-MC	16.7	12.7	-24%	309	--	0.68
	DF-IMC	16.7	9.1	-46%	241	-22%	0.68
	DF-LP	16.7	7.4	-56%	233	-25%	0.68
	IW	16.7	13.2	-21%	0	-100%	2.28
	AS	16.7	11.6	-31%	0	-100%	2.14
	AS+IW	16.7	8.3	-50%	0	-100%	2.32
	AS+DF	16.7	8	-52%	168	-46%	2.31
	AS+IW+DF	16.7	2.9	-83%	167	-46%	2.37

Table 14 – Comparison of density variation of 5-bit ADC layouts for CMOS 180 nm technology

Layouts for Test/Methods		Window Variation (%)		Improvement (%)	Numbe r of Fill	Fill Reduction (%)	Run-Time (Sec.)
		Before	After				
5-bit ADC	DF-MC	11.4	8.6	24.43	1456	-	103.2
	DF-IMC	11.4	8.2	27.94	1056	-27.5	103.2
	DF-LP	11.4	6.95	38.93	1088	-25.3	104.16
	IW	11.4	4.6	59.58	0	-100	300.8
	AS	11.4	8.4	26.32	0	-100	291.6
	AS+IW	11.4	4.4	61.34	0	-100	308.4
	AS+DF	11.4	5.4	52.63	416	-71.4	311.3
	AS+IW+DF	11.4	3.9	65.73	368	-74.7	317.1

In all of the four tables, density variations before and after applying any specific approach are listed with improvement percentages marked as well. *Number of Fill* column shows the number of dummy features inserted on the layer and *Fill Reduction* column reports the number reduction percentage of dummy fill blocks with reference to that in the first row. One individual dummy unit is a 1μm×1μm square for CMOS 180 nm technology and a 0.5μm×0.5μm square for CMOS 90 and 65 nm technologies. In the last column, the run time of the different methods is also listed for the comparison purpose.

As shown in Table 11, for both test layouts, DF-MC method can improve density uniformity to some extent while it inserts an excessive amount of fill blocks. In comparison, DF-IMC and DF-LP can improve density uniformity much better with significantly less number of dummy features inserted. By using the proposed interconnect widening operation alone (i.e., in the IW rows), the density variation can be reduced to 31% and 36% for the two-stage Opamp and folded-cascode Opamp, respectively. By combining the contribution from the area shift operation (i.e., in the AS+IW rows), the proposed approach can improve density uniformity to 41% and 46% for the two-stage Opamp and folded-cascode Opamp, respectively.

If further adding the complementary dummy insertion operation, the proposed method (i.e., AS+IW+DF) can improve the density variation by 74% and 70% for the two test layouts, much better than the performance achieved by DF-MC, DF-IMC, and DF-LP. In addition, our proposed approach features about 20% to 40% reduction in the number of inserted dummy blocks compared to DF-IMC and DF-LP. Due to the

complexity of our algorithm and its graph handling procedure, the run time of our proposed method is more than that of the other previously published works.

Table 12 and Table 13 show a very similar trend. For CMOS 90 nm technology in Table 12, DF-MC improves the layout density of 21% and 48% for the two-stage and folded-cascode Opamps respectively. By using DF-IMC and DF-LP methods, the layout density can improve more yet with less numbers of inserted dummy blocks. Although the interconnect widening and area shift methods alone can only improve the density uniformity a little bit, combining them together with dummy fill insertion, i.e. AS+IW+DF, can achieve 63% and 75% improvement in the density uniformity for two-stage and folded-cascode Opamps respectively, which is much better than DF-LP and DF-IMC methods. In addition, AS+IW+DF can insert at least 7% less number of dummy blocks compared to the previous dummy insertion methods. Table 13 shows similar test results for CMOS 65 nm technology. Our proposed method AS+IW+DF outperforms other previously published works, DF-MC, DF-IMC, and DF-LP with better density uniformity distribution and less number of inserted dummy blocks.

The 5-bit ADC layout test results are shown in Table 14. Although the layout size is significantly bigger than the two Opamp layouts, the trend is consistent with the data presented in Table 11, Table 12 and Table 13. As expected, DF-MC method improved density uniformity a little but with a large number of small dummy blocks. By using DF-IMC and DF-LP methods, the density variation improves 28% and 39% respectively with almost 25%-27% less number of inserted dummy blocks. Using the area shift only can improve density variation by 26% due to the layout complexity and limitation of moving

interconnects, but combining it with dummy fill insertion can improve density variation to 52.6% with almost 71.5% less inserted dummy blocks. The interconnect widening approach can improve density uniformity by 59% individually and 61% if it is combined with the area shift approach. Moreover, by using the area shift, interconnect widening along with dummy fill insertion can improve density uniformity up to 65.73% with 74.7% less number of inserted dummy fill blocks.

To show the effect of our proposed approach on layout parasitics, we extracted parasitic coupling/lateral capacitance from the final DF-LP and AS+IW+DF layouts for the two Opamp test layouts in CMOS 90 nm technology. The coupling/lateral capacitance between each pair of electrical nets in the final layout generated by our proposed AS+IW+DF scheme was recorded and compared with that of the same nets in the layout generated by scheme DF-LP. For the two-Stage Opamp, the maximum coupling/lateral capacitance increase in the DF-LP layout is 16.49% more than that in the layout generated by our proposed AS+IW+DF method. The average increment of coupling/lateral capacitance is also compared for both approaches and it was found that our proposed method could produce 11.35% less capacitance than the DF-LP method. For the folded-Cascode Opamp, the amount of the maximum increment in coupling/lateral capacitance by the DF-LP method is 14.18% more than that from our proposed method, while the average increment of coupling/lateral capacitance in the layout generated by the DF-LP method is 26.85% more than that in the AS+IW+DF layout. The comparison of average increment in the coupling/lateral capacitance between

different nets of the ADC layout shows that DF-LP method introduces 16.12% more parasitic capacitance compared to AS+IW+DF method.

To ensure any density-uniformity-related modification has no negative impact on circuit performance, we presented circuit specifications and post-layout simulations of the Opamps in Table 15 and the 5-bit ADC in Table 16 for the comparison purpose. One can observe that the modified layouts can satisfy the due circuit specifications with little change compared to the resized layouts. The comparator circuit used in the 5-bit ADC design is sensitive to parasitics. Therefore, it can be seen in Table 16 that the layout generated by using DF-LP method cannot meet the specifications while the layout generated by AS+IW+DF method meets the specifications with less change in comparison with the schematic simulation results.

Figure 37 exhibits the two-stage Opamp with Metal-1 layer highlighted layouts before and after AS+IW+DF operation. A zoom-in area within the ellipse bubble is focused on to illustrate the applied area-shift and interconnect widening operations in detail. Figure 38 shows the folded-cascode Opamp layouts before and after AS+IW+DF operation. It is observed in both layouts that some modules and interconnects are moved and some of interconnects are widened to implement the recommended solution.

Table 15 – Performance comparison of two test layouts in CMOS 180 nm technology

Performance	**Spec.**	**Resized Layout**	**DF-LP Layout**	**AS+IW+ DF Layout**
Two-Stage Opamp				
Gain (dB)	60	62.65	62.65	62.65
Bandwidth (MHz)	80	123.74	123.73	123.74
Phase Margin (˚)	60	79.47	79.48	79.95
Gain Margin (dB)	10	16.1	16.1	16.4
Folded-Cascode Opamp				
Gain (dB)	60	60.82	60.82	60.82
Bandwidth (MHz)	30	49.05	49.16	48.96
Phase Margin (˚)	60	77.8	77.86	78
Gain Margin (dB)	10	44.8	45.11	44.9

Table 16 – Performance comparison of 5-bit ADC

Parameters		**Spec.**	**Schematic**	**DF-LP Layout**	**AS+IW+DF Layout**
DNL - Differential Non-Linearity	Max	0.25	0.19	0.27	0.22
	Diff. (%)	NA	NA	39.20	14.42
INL- Integral Non-Linearity	Max	0.6	0.49	0.65	0.53
	Diff. (%)	NA	NA	30.90	7.42
Power Consumption (mW)		3.5	3.04	3.05	3.04

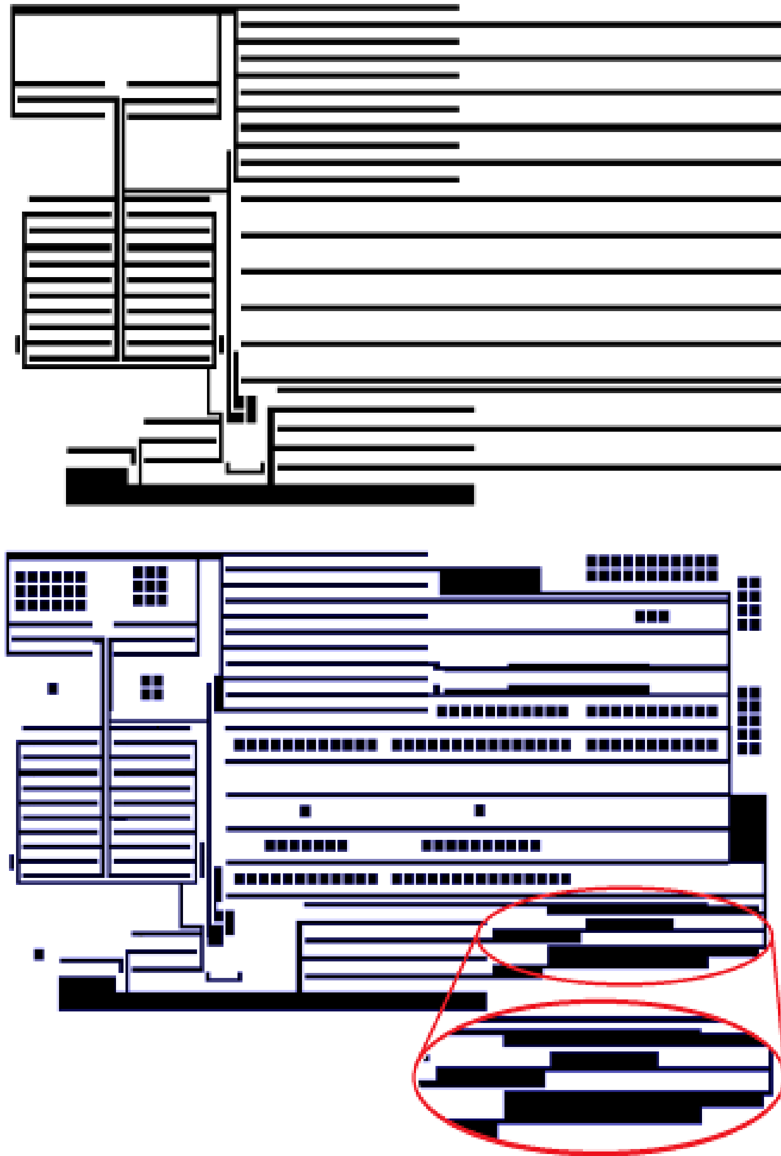


Figure 37 – Two-stage Opamp layouts before (top) and after (bottom) the AS+IW+DF operations in CMOS 180 nm technology. The splitting and widening operation is also shown in the red ellipse.

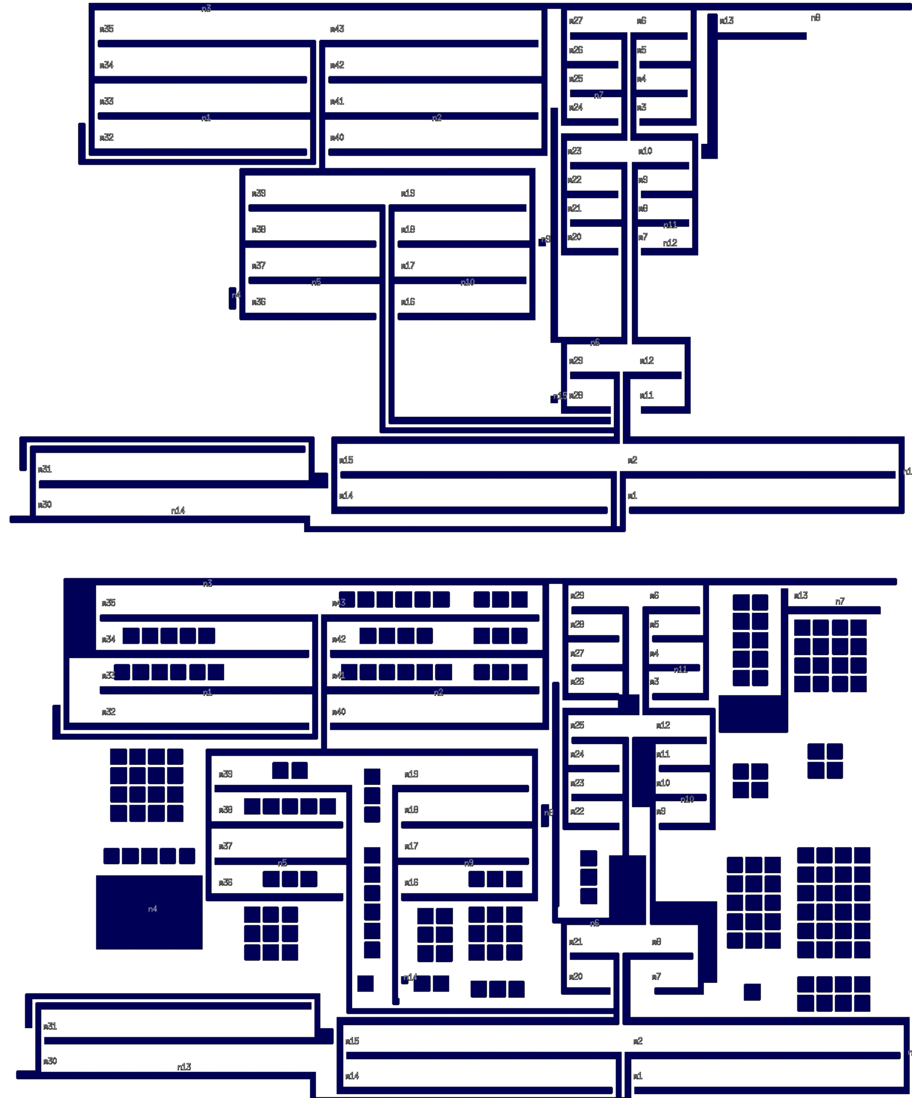


Figure 38 – Folded-cascode Opamp layouts before (top) and after (bottom) the AS+IW+DF operations in CMOS 180 nm technology.

Figure 39 – A small portion of the analog part of the 5-bit ADC layout (Metal 1) after modification by AS+IW+DF

Figure 39 shows a small portion of 5-bit ADC layout after applying AS+IW+DF method. Once area shift and interconnect widening have been performed, the layout density distribution is analyzed and the rest of the solution would be implemented by inserting dummy blocks. As shown in Figure 37, Figure 38 and Figure 39, some empty areas in the layout have been filled with arrays of small dummy blocks.

6.4 Summary

In this chapter we presented the testing and verification results of our proposed methodologies on parasitic capacitance analytic modeling and density-uniformity-aware analog layout retargeting. First, the accuracy of the parasitic capacitance modeling was testified with reference to one industry mainstream layout extraction tool. Then, we used the developed capacitance models in the context of our proposed density-uniformity-aware analog layout retargeting for improving the layout quality through handling manufacturability issues and controlling parasitic capacitance. The next chapter is focused on the conclusions and recommendations for the future research.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Summary

In this dissertation research work, a general density-uniformity-aware layout retargeting methodology for analog layouts was proposed to take into account parasitic capacitance constraints in the layout retargeting optimization. To achieve this goal, we worked on a complete analytic model for estimating parasitic capacitance in a multi-layer layout and a software platform for density-uniformity-aware analog layout migration. The comprehensive analytic modeling of parasitic capacitance was developed to cover all possible situations appearing inside a multi-layer layout. Our proposed modelling was tested and verified in terms of accuracy and simplicity. Another advantage of our proposed modeling is its reusability, that is, with a little effort of tuning, the model can be used in a variety of technologies and topological situations.

In the second phase of this dissertation work, we have studied and proposed new strategies of layout element position movement and size modification to address density uniformity issues for analog layout retargeting. Based on the thorough literature review, it is understood that the traditional procedure to control density uniformity especially used in digital layouts is dummy fill insertion methods. The previously published works usually consider the layout under test as a set of non-changeable geometries. However, our proposed approach to control density of an analog layout is significantly different from the previous works by modifying layout elements for improving density uniformity.

As modifying layout elements may affect both technology design-rule compliance and circuit performance, this operation should be done with caution in a holistic way.

Therefore, a linear-programming-based formulation scheme was developed to plan the density distribution, which is subsequently fulfilled by our special constraint-graph-based area shift, interconnect widening, and dummy fill insertion operations. We also took into account parasitic capacitance impact during layout modification process by using our proposed capacitance modeling to control the degradation effect on circuit performance. The experimental results exhibit that the proposed methodology can outperform the traditional dummy fill insertion methods in terms of density variation and parasitic capacitance control.

7.2 Research Contributions

The contributions of this PhD dissertation include the following:

- The comprehensive parasitic capacitance modeling proposed in the first phase of this research, which is simple yet accurate, can cover all the possible topological situations that may appear in a multi-layer layout. It is designed to be tunable for reuse in any technologies and geometric scenarios by using specific technology parameters with an aid of the curve-fitting technique.
- We have proposed a linear model for estimating parasitic capacitance in substitution of any expensive logarithmic models. Although the accuracy of the linear model is relatively lower, its sound trade-off between simplicity and

accuracy can sufficiently justify its application to the linear programming optimization or loops of optimization in the VLSI CAD domain.

- Testing and verification of our proposed parasitic capacitance modeling have demonstrated its promising characteristics in terms of accuracy and simplicity. Its reusability and fast run-time are also the advantageous features compared to the other published works.
- In the second phase of the dissertation research, a new approach has been proposed to address density uniformity challenges in design for manufacturability of analog layouts. Considering the fast advancement of IC technologies, we have focused our research on the analog layout retargeting process, which is still immature in the current EDA market. We have aimed to improve density uniformity in a multi-layer analog layout during the layout migration from an old technology to a new technology and/or updated specifications. We have integrated the layout resource information and layer density requirements into the constraint graph optimization to address the density uniformity problem. Unlike the traditional dummy fill insertion methods that only deal with fixed layouts, our proposed method can dedicatedly treat flexible layouts, which are actually the nature of analog layout retargeting process. While updating layout elements during the layout migration, the layer density distribution in the layout can be rearranged by using the existing elements in the layout. Therefore, the traditional exclusive dependence on the dummy fill insertion especially in the digital layout design can be reduced.

- Our proposed density-uniformity formulation is so general that it can be used for any analog layouts. It is based on linear programming optimization aimed for layout modification through area shift and interconnect widening operations. A reduced form of linear programming, which includes less number of variables, has been proposed as well for shrunk algorithmic complexity.
- Fulfillment of the recommended solution has been integrated to the constraint graph modification and handling. By using constraint graph optimization algorithms, a resized layout can be smartly modified to not only meet new design specifications but also improve its density uniformity. Thus in this work, a novel LP-based formulation for density uniformity is merged with constraint graph optimization and layout retargeting process. This has not been reported elsewhere thus far.
- Any modification in a layout should be performed in a controlled manner, while any improper change to parasitic capacitance may degrade circuit performance. Therefore, the capacitance modeling proposed in the first phase of this research has been utilized in our proposed layout retargeting process to control the effect of layout modification on circuit performance. This feature can ensure to maintain electrical performance correctness and reliability during the layout migration process.
- The experiments conducted on a couple of commonly used analog Opamps and a 5-bit ADC confirm the high effectiveness and efficiency of our proposed

methodologies in terms of density uniformity improvement and parasitic capacitance control.

7.3 Recommendations for Future Research

- New advanced nanometer technologies bring more challenges to simple yet accurate parasitic capacitance modelling for a multi-layer chip. Our proposed model in this dissertation can be adapted to the sub-45nm technologies with more reliable tuning methods being investigated.
- The proposed LP formulation for density uniformity optimization can be extended to multi-layer optimization, which considers multiple layers at the same time right in the planning stage to improve density uniformity for a multi-layer layout.
- Different optimization methods such as non-linear programming or even evolutionary algorithms may be investigated to not only enclose more constraints but also improve recommended solutions for better fulfillment of density uniformity in the subsequent stage.
- More design for manufacturability factors (e.g., double or multiple patterning) may be taken into account in the context of analog layout retargeting.

REFERENCES

[1] N. Jangkrajarng, "Analog/RF VLSI Layout Generation Layout Retargeting via Symbolic Template," Ph.D. Dissertation, Univ. of Washington, Seattle, WA, 2006.

[2] D. O. Ouma, "Modeling of Chemical Mechanical Polishing for Dielectric Planarization," Ph.D. Dissertation, Dept. of Elect. and Comp. Sci., Massachusetts Inst. of Tech. (MIT), Boston, MA, 1998.

[3] A. B. Kahng and K. Samadi, "CMP Fill Synthesis: A Survey of Recent Studies," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 27, no. 1, pp. 3-19, Jan. 2008.

[4] L. Zhang, "VLSI Circuit Layout," in *Encyclopedia of Computer Science and Engineering*, vol. 5, Wiley, 2009, pp. 3034-3044.

[5] D. Z. Pan and M. D. F. Wong, "Manufacturability-aware Physical Layout Optimizations," in *Proc. Int'l Conference on Integrated Circuit Design and Technology (ICICDT)*, pp. 149-153, Austin, TX, USA, May 2005.

[6] Y. Chen, "Layout Density Control for Improved VLSI Manufacturability," Ph.D. Dissertation, Univ. of California, Los Angeles, CA, 2003.

[7] I. Nitta et al., "“Condition-based” Dummy Fill Insertion Method based on ECP and CMP Predictive Models," in *Proc. Int'l Symposium on Quality Electronic Design (ISQED)*, pp. 198-205, San Jose, CA, USA, March 2010.

[8] S. Wane and D. Bajon, "Characterisation and Macro-modeling of Patterned Micronic and Nano-scale Dummy Metal-fills in Integrated Circuits," in *Proc. of IEEE Radio Frequency Integrated Circuits Symposium*, pp. 171-174, Boston, MA, USA, 2009.

[9] L. Nan, K. Mouthaan, Y. Z. Xiong, J. Shi, S. C. Rustagi and B. L. Ooi, "Impact of Metal Dummy Fills on the Performance of CMOS Inductors," in *Proc. IEEE Conference on Electron Devices and Solid-State Circuits*, pp. 251-254, Tainan, Taiwan, Dec. 2007.

[10] L. Carloni, A. B. Kahng, S. Muddu, A. Pinto, K. Samadi and P. Sharma, "Interconnect Modeling for Improved System-Level Design Optimization," in *Proc. of Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 258-264,

Seoul, 2008.

[11] A. Bansal, B. C. Paul and K. Roy, "An Analytical Fringe Capacitance Model for Interconnects using Conformal Mapping," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 25, no. 12, pp. 2765-2774, Dec. 2006.

[12] G. Shomalnasab and L. Zhang, "New Analytic Model of Coupling and Substrate Capacitance in Nanometer Technologies," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI),* vol. 23, no. 7, pp. 1268 - 1280, 2015.

[13] A. B. Kahng and R. O. Topaloglu, "DOE-Based Extraction of CMP, Active and Via Fill Impact on Capacitances," *IEEE Trans. on Semiconductor Manufacturing,* vol. 21, no. 1, pp. 22-32, Feb. 2008.

[14] A. Kurokawa, T. Sato, T. Kanamoto and M. Hashimoto, "Interconnect Modeling: A Physical Design Perspective," *IEEE Transactions on Electron Devices,* vol. 56, no. 9, pp. 1840-1851, Sept. 2009.

[15] A. Kurokawa, T. Kanamoto, A. Kasebe, Y. Inoue and H. Masuda, "Efficient Capacitance Extraction Method for Interconnects with Dummy Fills," in *Proc. IEEE Conference on Custom Integrated Circuits Conference*, pp. 485-488, Oct. 2004.

[16] K. Nabors and J. White, "Multipole-Accelerated Capacitance Extraction Algorithms for 3-D Structures with Multiple Dielectrics," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications,* vol. 39, no. 11, pp. 946-954, Nov. 1992.

[17] N. V. K. S. a. D. S. C. R. Sharma, "Closed-form expressions for extraction of capacitances in multilayer VLSI interconnects," in *Proc. IEEE Regional Ten Conference*, pp. 1-4, 2008.

[18] S. Tani, Y. Uchida, M. Furuie, S.Tsukiyama, B. Lee, S. Nishi, Y. Kubota, I. Shirakawa and S. Imai, "Parasitic Capacitance Modeling for Multilevel Interconnects," in *Proc. Asia-Pacific Conference on Circuits and Systems*, pp. 59-64, 2002.

[19] Y. Xiang, "The Electrostatic Capacitance of An Inclined Plate Capacitor," *Journal of Electrostatics,* vol. 64, no. 1, pp. 29-34, 2006.

[20] J. M. Bueno-Barrachina, C. S. Cañas-Peñuelas and S. Catalan-Izquierdo, "Capacitance Evaluation on Non-parallel Thick-Plate Capacitors by Means of Finite Element Analysis," *Journal of Energy and Power Engineering,* vol. 5, no. 4, pp. 373-378, Apr. 2011.

[21] G. S. Samudra and H. L. Lee, "A Set of Analytic Formulas for Capacitance of VLSI Interconnects of Trapezium Shape," *IEEE Trans. On Electron Devices,* vol. 41, no. 8, pp. 1467-1469, Aug. 1994.

[22] W. Zhao, X. Li, S. Gu, S. Kang, M. Nowak and Y. Cao, "Field-Based Capacitance Modeling for Sub-65-nm On-Chip Interconnect," *IEEE Trans. on Electron Device,* vol. 56, no. 9, pp. 1862-1872, 2009.

[23] N. D. Arora, K.V. Raol, R. Schumann and L.M. Richardson, "Modeling and Extraction of Interconnect Capacitances for Multilayer VLSI Circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 15, no. 1, pp. 58-67, Jan. 1996.

[24] E. Barke, "Line-to-ground Capacitance Calculation for VLSI: A Comparison," *IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems,* vol. 7, no. 2, pp. 295-298, Feb. 1988.

[25] S. Batterywala, R. Ananthakrishna, Y. Luo and A. Gyure, "A Statistical Method for Fast and Accurate Capacitance Extraction in the Presence of Floating Dummy Fills," in *Proc. International Conference of VLSI Dsign*, 2006.

[26] A. Husain, "Models for Interconnect Capacitance Extraction," in *Proc. International Symposium on Quality Electronic Design (ISQED)*, pp. 167-172, 2001.

[27] Y. Chang, H. Chang, T. Lu, Y. King, W. Ting, J. Ku and C. Lu, "A Novel CBCM Method Free from Charge Injection Induced Errors: Investigation into the Impact of Floating Dummy-Fills on Interconnect Capacitance," in *Proc. International Conference on Microelectronic Test Structures*, pp. 235-238, 2005.

[28] K. D. Hyun, S. Mukhopadhyay and S. Lim, "Fast and Accurate Analytical Modeling of Through-Silicon_Via Capacitive Coupling," *IEEE Trans. on Components, Packaging and Manufacturing Technology,* vol. 2, pp. 168-180, 2011.

[29] S. Pandit and C. Sarkar, "An Analytical Model for Sidewall Parasitic Capacitance of nano-Scale Trench Isolated MOSFETs," in *International Conference on Electrical and Computer Engineering*, pp. 578-581, 2010.

[30] T. An and S. Kim, "3-D Modeling of Fringing Gate Capacitance in Gate-All-Around Cylindrical Silicon Nanowire MOSFETs," in *Proc. Conference on Simulation of Semiconductor Processes and Devices*, pp. 256-259, 2013.

[31] W. E. Cory, "Layla: A VLSI Layout Language," in *Proceeding on Design*

Automation Conference, pp. 245-251, June 1985.

[32] S. K. Hong and P. E. Allen, "Performance Driven Analog Layout Compiler," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 835-838, May 1990.

[33] J. M. Cohn, D. J. Garrod, R. A. Rutenbar and L. R. Carley, "Additional Koan/Anagram II Results," in *Analog Device-Level Layout Automation*, Springer US, 1994, pp. 253-269.

[34] E. Malavasi, E. Charbon, E. Felt and A. Sangiovanni-Vincentelli, "Automation of IC Layout with Analog Constraints," *IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems,* vol. 15, no. 8, pp. 923-942, Aug. 1996.

[35] G. Zhang, A. Dengi, R. A. Rohrer, R. A. Rutenbar and L. R. Carley, "A Synthesis Flow toward Fast Parasitic Closure for Radio-Frequency Integrated Circuits," in *Proc. of Design Automation Conference (DAC)*, pp. 155-158, 2004.

[36] L. Zhang, U. Kleine and Y. Jiang, "An Aautomated Design Tool for Analog Layouts," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems,* vol. 14, no. 8, pp. 881-894, Aug. 2006.

[37] E. Yilmaz and G. Dundar, "Analog Layout Generator for CMOS Circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 28, no. 1, pp. 32-45, Jan. 2009.

[38] R. Castro-López, F. V. Fernández and A. Rodríguez-Vázquez, "Generation of Technology-Independent Retargetable Analog Blocks," *Journal of Analog Integrated Circuits and Signal Processing,* vol. 33, no. 2, pp. 157-170, Nov. 2002.

[39] R. Castro-López, O. Guerra, E. Roca and F. V. Fernandez, "An Integrated Layout-Synthesis Approach for Analog ICs," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 27, no. 7, pp. 1179-1189, July 2008.

[40] N. Jangkrajarng, S. Bhattacharya, R. Hortono and C.J. R. Shi, "IPRAIL-Intellectual Property Reuse-based Analog IC Layout Automation," *Journal of Integration,* vol. 36, no. 4, pp. 237-262, Nov. 2003.

[41] L. Zhang and Z. Liu, "Directly performance-constrained template-based layout retargeting and optimization for analog integrated circuits," *Integration, the VLSI Journal,* vol. 44, no. 1, pp. 1-11, Jan. 2011.

[42] S. Wang, X. Fan, S. Zhang and J. Ming-e, "Analog Layout Retargeting with Geometric Programming and Constrains Symbolization Method," in *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 353-356, Seoul, 2012.

[43] C. Y. Chin, P. C. Pan, H. M. Chen, T. C. Chen and J. C. Lin, "Efficient Analog Layout Prototyping by Layout Reuse with Routing Preservation," in *Proc. of IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 40-47, 2013.

[44] R. Martins, N. Lourenco and N. Horta, "LAYGEN II-Automatic Layout Generation of Analog Integrated Circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 32, no. 11, pp. 1641-1654, 2013.

[45] R. Martins, N. Lourenco, S. Rodrigues, J. Guilherme and N. Horta, "AIDA: Automated Analog IC Design Flow from Circuit Level to Layout," in *Proc. of Intl. Conf. on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, pp. 29-32, 2012.

[46] Y. P. Weng, H. M. Chen, T. C. Chen, P. C. Pan, C. H. Chen and W. Z. Chen, "Fast Analog Layout Prototyping for Nanometer Design Migration," in *Proc. IEEE/ACM Int'l Conf. on Computer-Aided Design (ICCAD)*, pp. 517-522, San Jose, CA, 2011.

[47] P. C. Pan, C. Y. Chin, H. M. Chen, T. C. Chen, C. C. Lee and J. C. Lin, "A Fast Prototyping Framework for Analog Layout Migration with Planar Preservation," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD),* vol. 34, no. 9, pp. 1373-1386, 2015.

[48] A. B. Kahng, G. Robins, A. Singh and A. Zelikovsky, "Filling Algorithms and Analyses for Layout Density Control," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems (TCAD),* vol. 18, no. 4, pp. 445-462, Apr. 1999.

[49] M. Hanan, "On Steiner's Problem with Rectilinear Distance," *SIAM Journal of Applied Math,* vol. 14, pp. 255-265, 1966.

[50] H. Xiang, K.-Y. Chao, R. Puri and M. D. F. Wong, "Is Your Layout Density Verification Exact? -A Fast Exact Deep Submicrometer Density Calculation Algorithm," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and systems,* vol. 27, no. 4, pp. 621-632, 2008.

[51] Y. Chen, A. B. Kahng, G. Robins and A. Zelikovsky, "Monte Carlo Algorithms for Layout Density Control," in *Proc. ASP-Design Automation Conference*, pp. 523-

528, 2000.

[52] Y. Chen, A. B. Kahng, G. Robins and A. Zelikovsky, "Area Fill Synthesis for Uniform Layout Density," *IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems,* vol. 21, no. 10, pp. 1132-1147, Oct. 2002.

[53] Y. Chen, P. Gupta and A. B. Kahng, "Performance-Impact Limited Area Fill Synthesis," in *Proc. IEEE Design Automation Conference (DAC)*, pp. 22-27, Jun. 2003.

[54] R. Tian, D. Wong and R. Boone, "Model-based Dummy Feature Placement for Oxide Chemical-Mechanical Polishing Manufacturability," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 20, no. 7, pp. 902-910, 2001.

[55] M. Mukherjee and K. Chakraborty, "A Randomized Greedy Method for Rectangular-Pattern Fill Problems," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 27, no. 8, pp. 1376-1284, Aug. 2008.

[56] C. Feng, H. Zhou, C. Yan, J. Tao and X. Zeng, "Efficient Approximation Algorithms for Chemical Mechanical Polishing Dummy Fill," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD),* vol. 30, no. 3, pp. 402-415, 2011.

[57] H.Y. Chen, S. J. Chou and Y. W. Chang, "Density Gradient Minimization with Coupling-Constrained Dummy Fill for CMP Control," in *Proc. of the international symposium on Physical design (ISPD)*, pp. 105-111, 2010.

[58] N. Dhumane and S. Kundu, "Critical Area Driven Dummy Fill Insertion to Improve Manufacturing Yield," in *Proc. of Intl. Symposium on Quality Electronic Design (ISQED)*, pp. 334-341, 2012.

[59] D. Saha and S. Sur-Kolay, "Planarization of Metal Layers in a Chip based on Voronoi Diagram," in *Proc. of Int'l Conf. on Computers and Devices for Communication (CODEC)*, pp. 1-4, Kolkata, 2012.

[60] T. C. Luo, M. C. T. Chao, P. A. Fisher and C. R. Kuo, "A Novel Design Flow for Dummy Fill using Boolean Mask Operations," *IEEE Trans. On Semiconductor Manufacturing,* vol. 25, no. 3, pp. 468-479, 2012.

[61] C. Liu, P. Tu, P. Wu, H. Tang, Y. Jiang, J. Kuang and E. F. Y. Young, "An Effective Chemical Mechanical Polishing Fill Insertion Approach," *ACM Trans. on Design*

Automation of Electronic Systems (TODAES), vol. 21, no. 3, pp. 54-75, 2016.

[62] Y. Lin, B. Yu and D. Z. Pan, "High Performance Dummy Fill Insertion with Coupling and Uniformity Constraints," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD),* no. 99, 2016.

[63] G. Shomalnasab, H. M. Heys and L. Zhang, "Analytic Modeling of Interconnect Capacitance in Submicron and Nanometer Technologies," in *International Symposium on Circuits and Systems (ISCAS)*, pp. 2553-2556, Beijing, China, May 2013.

[64] G. Shomalnasab and L. Zhang, "Analog Layout Density Uniformity Improvement using Interconnect Widening and Dummy Fill Insertion," in *International Symposium on Circuits and Systems (ISCAS)*, pp. 1826 - 1829, Baltimore, MD, USA, May 2017.

[65] J. K. Ousterhout, "Corner Stitching: A Data Structuring Technique for VLSI Layout Tools," *IEEE trans. on Computer-Aided Design of Integrated Circuits and Systems,* vol. 3, no. 1, pp. 87-100, Jan. 1984.

[66] S. L. Lin and J. Allen, "Minplex- A Compactor that Minimizes the Bounding Rectangle and Individual Rectangles in A Layout," in *Proc. of IEEE/ACM Design Automation Conference*, pp. 123-130, 1986.

[67] T.H. Cormen, C.E. Leiserson and R.L. Rivest, Introduction to Algorithms, Cambridge, MA: MIT Press, 2001.

[68] G. Lakhani and R. Varadarajan, "A Wire-Length Minimization Algorithm for Circuit Layout Compaction," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, pp. 276-279, Philadelphia, PA, USA, May 1987.

[69] K. G. Murty, Linear Programming, New York: John Wiley & Sons Inc., 1983.

[70] "MATLAB," Mathworks, 2014. [Online]. Available: http://mathworks.com.

[71] Cadence Design Systems, Inc., San Jose, CA, "Diva," [Online]. Available: http://www.cadence.com.

[72] Mentor Graphics Corporation, Wilsonville, OR,, "Calibre v2011," [Online]. Available: http://www.mentor.com.

[73] "Cadence SKILL," [Online]. Available:

http://en.wikipedia.org/wiki/Cadence_SKILL. [Accessed December 2012].

[74] M. I. Elmasry, "Capacitance Calculations in MOSFET VLSI," *IEEE Trans. on Electron Device Letters,* vol. 3, no. 1, pp. 6-7, Jan. 1982.

[75] T. Sakurai and K. Tamaru, "Simple Formulas for Two- and Three-Dimensional Capacitances," *IEEE Trans. on Electron Devices,* vol. 30, no. 2, pp. 183-185, Feb. 1983.

APPENDIX

Candidate's Publication List

[A1] G. Shomalnasab and L. Zhang, "Density-Uniformity-Aware Analog Layout Retargeting," *IEEE Trans. on Computer-Aided Design of Circuits and Systems (TCAD)* , Submitted in February 2017 (under minor revision at present).

[A2] G. Shomalnasab and L. Zhang, "New Analytic Model of Coupling and Substrate Capacitance in Nanometer Technologies," *IEEE Trans. on Very Large Scale Integration (TVLSI) Systems*, vol. 23, no. 7, pp. 1268 - 1280, 2015.

[A3] G. Shomalnasab and L. Zhang, "Analog Layout Density Uniformity Improvement using Interconnect Widening and Dummy Fill Insertion", *International Symposium on Circuits and Systems (ISCAS)*, pp. 1826 - 1829, May 2017, Baltimore, MD, USA.

[A4] G. Shomalnasab, H. M. Heys and L. Zhang, "Analytic Modeling of Interconnect Capacitance in Submicron and Nanometer Technologies", *International Symposium on Circuits and Systems (ISCAS)*, pp. 2553-2556, May 2013, Beijing, China.

[A5] W. Ren, G. Shomalnasab and L. Zhang, "Acoustic Underwater Positioning on FPGA", *IEEE OCEANS14*, September 2014, St. John's, Canada.

[A6] G. Shomalnasab and L. Zhang, "Density uniformity aware retargeting by IPRAIL", *Newfoundland Electrical and Computer Engineering Conference (NECEC2014)*, November 2014, St. John's, Canada.

[A7] G. Shomalnasab and L. Zhang, "Dummy Fill Insertion Challenges in IPRAIL", *Newfoundland Electrical and Computer Engineering Conference (NECEC2013),* November 2013, St. John's, Canada.

[A8] G. Shomalnasab, H. M. Heys and L. Zhang, "Analytical Study of Coupling Capacitance Modeling", *Newfoundland Electrical and Computer Engineering Conference (NECEC2012)*, November 2012, St. John's, Canada.

[A9] G. Shomalnasab and L. Zhang, "Fish Tagging: How Technology Can Help Marine Life", *Newfoundland Electrical and Computer Engineering Conference (NECEC2012)*, November 2012, St. John's, Canada.

[A10] G. Shomalnasab, H. M. Heys, L. Zhang, "Interconnect Capacitive Modeling in Submicron and Nano Technologies", *Newfoundland Electrical and Computer Engineering Conference (NECEC2011)*, November 2011, St. John's, Canada.