Title

New Methodologies for Evaluating Design Rules

Author

Gupta, Mukul

Publication Date

2013

UNIVERSITY OF CALIFORNIA

Los Angeles

New Methodologies for Evaluating Design Rules

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Electrical Engineering

by

Mukul Gupta

2013

ABSTRACT OF THE THESIS

New Methodologies for Evaluating Design Rules

by

Mukul Gupta

Master of Science in Electrical Engineering

University of California, Los Angeles, 2013

Professor Puneet Gupta, Chair

Design Rules (DRs) are the biggest design-relevant quality metric for a technology. Even small changes in DRs can have significant impact on manufacturability as well as circuit characteristics including layout area, variability, power, and performance. To systematically evaluate design rules several works have been published. The most recent among them is the Design Rule Evaluator (UCLA_DRE), a tool developed by NanoCad lab at UCLA, for fast and systematic evaluation of design rules and layout styles in terms of major layout characteristics of area, manufacturability, and variability. The framework essentially creates a virtual standard-cell library and performs the evaluation based on the virtual layout using first order models of variability and manufacturability (instead of relying on accurate simulation) and layout topology/congestion-based area estimates (instead of explicit and slow layout generation).

However, UCLA_DRE suffers from few major limitations. First, UCLA_DRE currently does not have the capability to evaluate the interaction between overlay design rules and overlay control, which is becoming more critical and more challenging with the move toward multiple-patterning(MP) lithography. Second, UCLA_DRE currently evaluates design rules at the cell level which may lead to misleading conclusions because most designs are routing-limited and, hence, not every change in cell area results in a corresponding change in chip area. Third, delay was not evaluated but it is well-known that *delay-change can affect chip-area* due to different buffering and gate sizing to meet timing requirements.

The first part of this dissertation offers a framework to study interaction between overlay design rules and overly control options in terms of area, performance and yield. The framework can also be used for designing informed, design-aware overlay metrology and control strategies. In this work, the framework was used to explore the design impact of LELE double-patterning rules and poly-line end extension rule defined between poly and active layer for different overlay characteristics (i.e., within-field vs. field-to-field overlay) and different overlay models at the 14nm node. Interesting conclusions can be drawn from the results. For example, one result shows that increasing the minimum mask-overlap length by 1nm would allow the use of a third-order wafer/sixth-order field-level overlay model instead of a sixth-order wafer/sixth-order field-level model with negligible impact on design.

In the second part of the dissertation, a new methodology called chipDRE, a framework to evaluate design rules at the chip-level, is described. chipDRE uses a good chips per wafer metric to unify area, performance, variability and functional yield. It uses UCLA_DRE to generate virtual standard-cell library and uses a mix of physical design and semi-empirical models to estimate area change at the chip-level due to both cell delay and cell area change. One interesting result for well to active spacing shows non-monotonic relationship of “good chips per wafer" with the rule value

The thesis of Mukul Gupta is approved.

Chi On Chui

Lei He

Puneet Gupta, Committee Chair

University of California, Los Angeles

2013

To my family

Table of Contents

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank Prof. Puneet Gupta for his brilliant guidance and constant support without which this thesis would not have been possible. Things that I have learnt working with him have helped me to achieve desired career goals.

I would also like to thank IMPACT+ research consortium at the University of California (http://impact.ee.ucla.edu) and Semiconductor Research Corporation for their generous funding for this research, Dr. Robert Socha from ASML and Dr. Alexander Starikov for the fruitful discussions and their valuable suggestions regarding the work on overlay.

I would like to sincerely thank my commitee members, Prof. Lei He and Prof. Chi on Chui for their valuable feedback on my research and teaching.

I would also like to thank my co-authors Rani S. Ghaida, Yasmine Badr and Ning Jin, and my colleagues at NanoCAD lab over the last two years : Abde Ali Kagliwala, Ankur Sharma, Shaodi Wang, Liangzhen Lai and Mark Gottsho

Finally, I would like to thank Arundhati for encouraging me at every step and my Mom, Dad, sisters for their unconditional support.

CHAPTER 1

Introduction

Semiconductors have fueled wealth creation, making new applications (cost-) feasible with each successive technology generation. Keeping Moore's Law alive would require rapid technology changes over the next decade and beyond. Accurate projection of the design impact of device and technology changes is key for making informed technology/design decisions, thereby, ensuring timely and cost-effective development of technology and design flows.

The evaluation of technology impact on design is traditionally inferred from the evaluation of design rules, which are the biggest design-relevant quality metric for a technology. UCLA_DRE [1] is a first of its kind framework developed at NanoCad Lab, UCLA to systematically evaluate design rules in terms of area, variability and yield. An overview of UCLA_DRE is shown in Fig. 1.1. It takes in a set of design rules, standard cell schematics, layout styles (such as cell height, M1/M2/diffusion

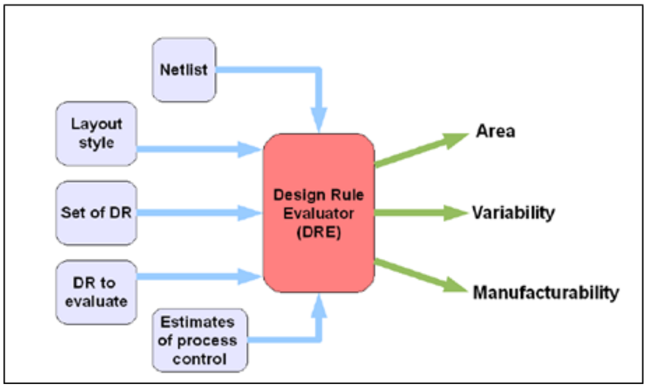


Figure 1.1: UCLA_DRE :a tool for systematic evaluation of Design rules developed by NanoCad Lab at UCLA [1]

power rail etc) and process related parameters (e.g. overlay error distribution, defect density) and generates the following:

- Virtual standard cell layouts with mainly front-end-of-the-line(FEOL) layers.
- Compute layout area for each standard cell
- Determine variability in terms of line-end tapering, poly rounding (mainly due to L-shape poly) and diffusion rounding(mainly due to L-shape diffusion).
- Determine contact failure yield, defect failure yield and overlay yield. Overlay Errors are assumed to be purely random and fully-correlated for all the features on the design. This implies that overlay yield is equal to yield of one feature.

Although, UCLA_DRE can evaluate a large number of design rules in a reasonable amount of time, it suffers from a few major limitations. First, interaction between overlay and design rules is limited to overlay errors being purely random and fully-correlated. It does not evaluate this interaction in terms of various overlay characteristics such as breakdown between inter-field/intra-field overlay errors and the alignment strategy used. Second,evaluation was performed at the cell-level, which does not truly reflect the area change at the chip-level since most of the time chip area is routing limited. Also, cell-level evaluation cannot evaluate the change in chip area due to sizing and buffering.

In this dissertation, second chapter deals with Overlay and its interaction with design rules, overlay characteristics and alignment strategies. Third chapter explain in detail the new methodology used for chip-level design rule evaluation. Finally, main contributions are summarized.

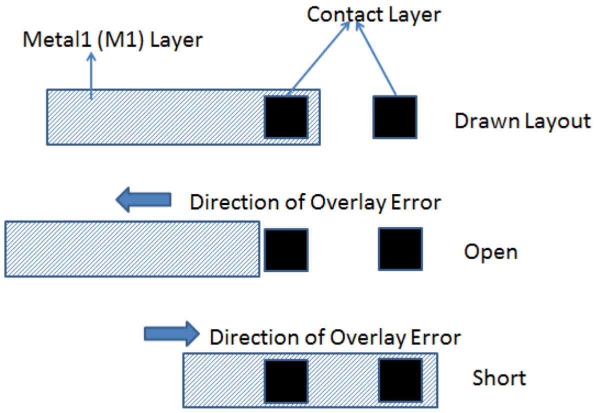


Figure 1.2: Illustration of overlay errors leading to short circuit and/or open circuit

1.1 Overlay

Overlay is the positional accuracy with which a pattern is formed on top of an existing pattern on the wafer [4] for ensuring correct functionality of the circuit. For example, It is well known that metal layer is exposed after the contact layer has been printed on the wafer. Any misalignment between the wafer and metal layer mask may lead to opens or shorts in the integrated circuit as shown in Fig. 1.2. According to ITRS, overlay budget is roughly 20% of the technology node. However, As technology scaling continues, overlay control is becoming more important than ever to allow smaller and smaller feature sizes. Moreover, the introduction of multiple-patterning (MP) lithography, where overlay effectively translates into CD variability [5, 6] (see Figure 1.3), has made overlay control even more critical and more challenging. Meeting the requirements for overlay control is believed to be one of the biggest challenges for deploying MP technology [7].

Overlay has been traditionally modeled using a linear model with major overlay components of translation, magnification, and rotation in the wafer and field coordinate systems [8, 9]. This linear model required a simple 2-point alignment. In recent years, the industry has moved toward high-order overlay modeling and more sophisticated alignment strategies, which requires more overlay sampling and excessive align-

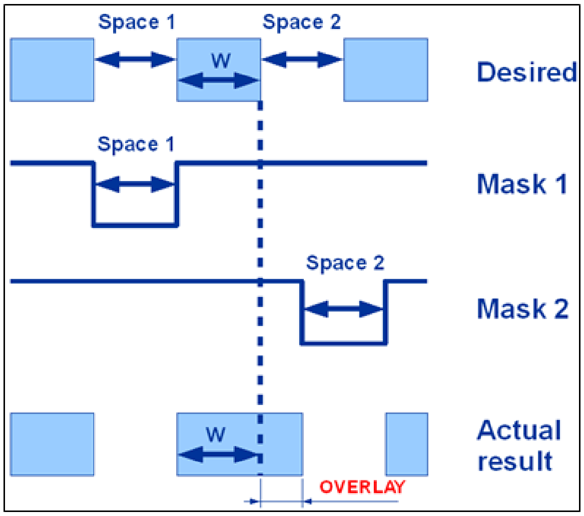


Figure 1.3: CD variaiton due to overlay in Double Patterning(DP) technology

ment [10, 3, 11, 12, 13]. These improvements in overlay control are capable of reducing overlay errors considerably when a high-order overlay model is used. On the downside, high-order modeling of overlay requires more advanced exposure scanners, more alignment measurements, and excessive off-line overlay metrology. Hence, the overlay improvement of high-order modeling comes at a huge cost in tool migration and diminished throughput capability due to the additional measuring time.

Design rules and overlay have strong interaction and can have a considerable impact on the design area, yield, and performance. Design rules that define interactions between different layers (e.g., metal overhang on via rule) or different mask-layouts of the same layer (e.g., mask overlap) effectively serve as guard band for overlay errors. For defining these rules during process development, a prediction of the yield loss due to overlay is needed. If overlay is characterized entirely as a field-to-field error, then the probability of survival (POS) for the die is equal to the POS of the most overlay-critical spot in the layout, say k. On the other extreme, if overlay is characterized entirely as a random within-field variation, then POS of the die is k^n, where n is the total number of critical spots in the design (see Fig. 1.4). Hence, depending on the overlay characteristics, rules can either be grown to suppress yield loss or shrank to reduce the layout

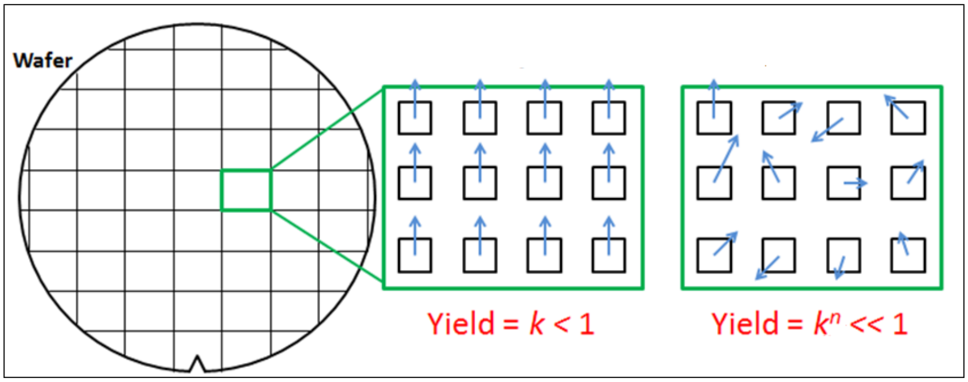


Figure 1.4: field-field(fully-correlated) vs within-field (independent)

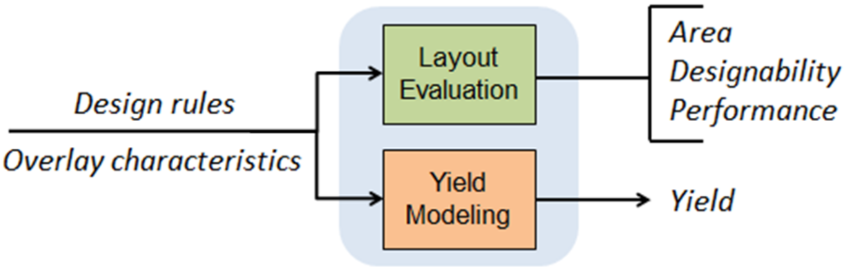


Figure 1.5: framework for evaluating interaction between overlay design rules and overlay control

area.

The thesis offers a framework to study this interaction and evaluate the overall design impact of rules, overlay characteristics, and overlay control options. An overview of the framework is given in Fig. 1.5. The framework can also be used for designing informed, design-aware overlay metrology and control strategies. As an illustration, The framework was used to explore the design impact of LELE double-patterning rules and poly-line end extension rule defined between poly and active layer for different overlay characteristics (i.e., within-field vs. field-to-field overlay) and different overlay models at the 14nm node. Interesting conclusions can be drawn from the results. For example, one result shows that increasing the minimum mask-overlap length by 1nm would allow the use of a third-order wafer/sixth-order field-level overlay model instead of a sixth-order wafer/sixth-order field-level model with negligible impact on design.

1.2 Chip-level Design Rule Evaluation

This thesis offers a new methodology Chip-level Design Rule Evaluator (ChipDRE), a framework for systematic evaluation of design rules and their interaction with layouts, performance, margins and yield at the chip-scale. ChipDRE uses a "good chips per wafer" (GCPW) metric to unify area, performance, variability and functional yield. It uses a generated virtual standard-cell library coupled with a mix of physical design and semi-empirical models to estimate area, delay and yield at the chip-level. To predict the design-rule/layout impact on delay and delay variability, Chip-DRE employs a Static Timing Analysis model to estimate cell-delay and a semi-empirical model to predict delay-margin dependent area penalty. Chip-level area is estimated from cell area – including the delay-margin area penalty – and a cell-change to chip-change area model that is calibrated using actual Synthesis, Place and Route (SPR) data. Finally, "good chips per wafer" is calculated taking into consideration a chip-level functional yield estimate. The result is a unified design-quality estimate that can be computed fast enough to allow using Chip-DRE to optimize a large number of complex design rules. The framework makes rule generation and optimization easier and much faster. Rather than exploring the entire search space of design rules manually or with conventional compute-expensive methods, the framework can be used to quickly eliminate poor rule choices. It also has the capability to do evaluation studies of major design rules at advanced nodes (some FinFET-specific) including: gate to local-interconnect spacing, gate-to-well edge spacing and fin pitch.

CHAPTER 2

Overlay control and Design Interaction

2.1 Overlay error modeling

The overlay error can be divided into intra-field and inter-field errors. An overlay model is a mathematical representation of the overlay between two patterning steps which typically takes the form of an algebraic function of 2 Cartesian coordinates in the plane of the wafer and 2 Cartesian coordinates in the plane of the reticle. In production dis-positioning and control, the model has historically been constrained to 6 linear wafer/grid terms (Xtranslation, Ytranslation, Xscale, Yscale, Xrotation, and Yrotation, or linear combinations thereof depending on the scanner) and similarly for the field [14]. So, a standard linear model is conveniently described by the following terms:

linear Inter-field Overlay Model

$$\Delta X = m1 + m3 * X + m5 * Y + resx_{wafer} \tag{2.1}$$

$$\Delta Y = m2 + m4 * X + m6 * Y + resy_{wafer} \tag{2.2}$$

linear Intra-field Overlay Model

$$\Delta x = k1 + k3 * x + k5 * y + resx_{field} \tag{2.3}$$

$$\Delta y = k2 + k4 * x + k6 * y + resy_{field} \tag{2.4}$$

The transition to higher order correction includes additional parametric terms, for both grid and field, where the specific details depend on the scanner involved. Higher Order Inter-field errors are caused mainly due to stage grid difference between machines, wafer loading error, wafer process effect, wafer thermal deformation, and so on . Higher order Intra-field errors are caused mainly due to lens distortion and reticle manufacturing error [15]. General form of intra-field and inter-field overlay errors are then given by

Intra-field Overlay Model

$$\Delta x = \sum_{i=1}^{m} \sum_{j=1}^{i} a_{ij} * x^{j} * y^{i-j} + resx_{field} \tag{2.5}$$

$$\Delta y = \sum_{i=1}^{m} \sum_{j=1}^{i} b_{ij} * x^{j} * y^{i-j} + resx_{field} \tag{2.6}$$

Inter-field Overlay Model

$$\Delta X = \sum_{i=1}^{m} \sum_{j=1}^{i} c_{ij} * X^{j} * Y^{i-j} + resx_{wafer} \tag{2.7}$$

$$\Delta Y = \sum_{i=1}^{m} \sum_{j=0}^{i} d_{ij} * X^{j} * Y^{i-j} + resy_{wafer} \tag{2.8}$$

where (X,Y) denotes the coordinate of a point on the wafer corresponding to the center of some exposure field and (x,y) the coordinates of a point on the wafer relative to the center of the exposure field in which it is contained as shown in Figure 2.1. $resx_{field}$, $resy_{field}$, $resx_{wafer}$ and $resy_{wafer}$ are the unmodeled components.

2.1.1 Strategies for overlay control

As part of usual production, the overlay of the exposed wafers is checked on an off-line metrology system and process corrections are calculated from the metrology data. Figure 2.2 [2] shows different data flows in the fab environment during the application of

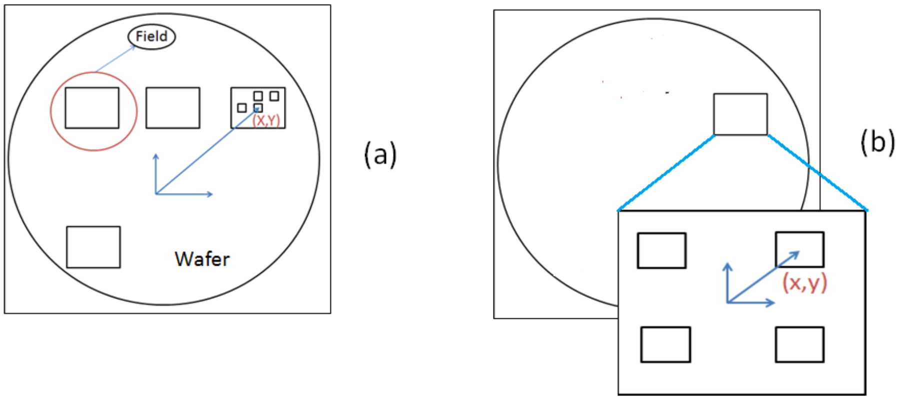


Figure 2.1: (a) (X,Y) : centre of exposure field w.r.t wafer centre (b) (x,y): a point in the exposure field w.r.t. centre of exposure field

process corrections. Usually, a sample of a few wafers per lot is measured. Then the measurement data is collected and sent to the APC (Advanced Process Control) system for modeling and calculations of new corrections.Then Manufacturing Execution System(MES) updates the process job with the new corrections and sends it to the lithography system where the corrections are applied during exposure of the subsequent lots. Systematic overlay introduced by various lithographic tools are identifed and eventually compensated. The most accurate process correction mechanism is the application of corrections per exposure (CPE), a look-up table based correction methodology. However, this requires extensive measurement on overlay metrology tool (measurement on all fields) which affects throughput. To reduce the measurement effort, higher order

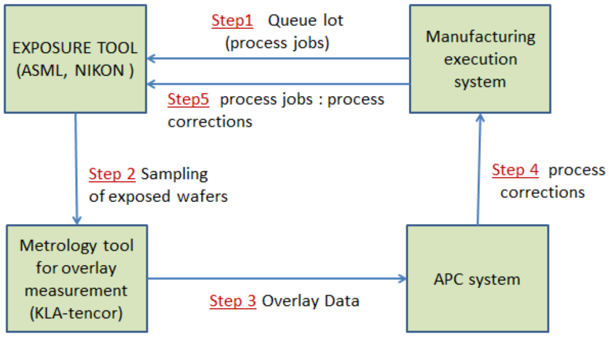
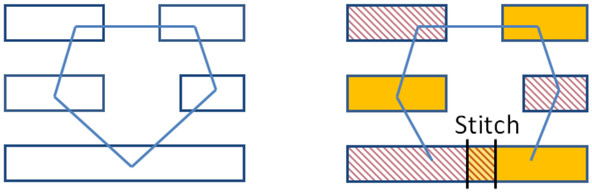
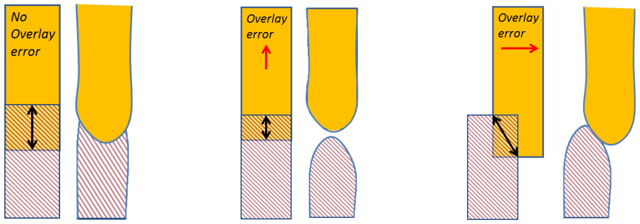


Figure 2.2: a typical production flow for overlay correction [2]



(a)

Figure 2.3: Example of a DP-problematic layout pattern with an odd cycle in its conflict graph (a) that was broken by introducing a stitch (b).



(a) (b) (c)

Figure 2.4: Example of a stitch (drawn and on-wafer) in a vertical line (a), a possible failure with overlay error in Y direction that may occur after line-end pullback (b), and a possible failure with overlay error in X direction due to narrowing (c).

process corrections (HOPC) have been introduced, in which only a subset of all fields and/or a subset of overlay targets per exposure field are measured and polynomial are fitted to these measurements. Although higher order correction results in smaller overlay error and hence improved yield, it also results in lower throughput. Hence there is an interesting trade-off between yield and throughput.

2.2 Overlay and Design Rule Interaction

In this thesis, DP-related design rules, namely the mask-overlap length rule and the minimum line-width and spacing design rules, and poly line-end extension rule and their interaction with overlay are studied.

The overlap-length rule is triggered whenever a stitch is introduced between the different mask layouts of the same layer. Although stitches may be a cause for yield

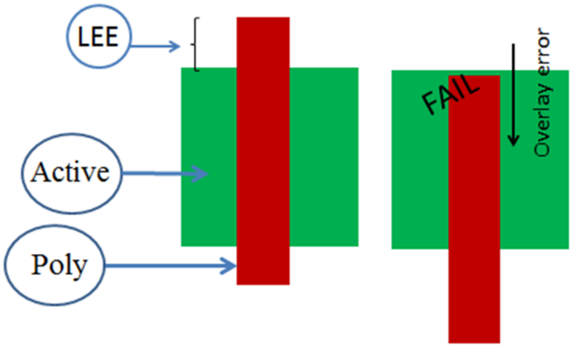


Figure 2.5: Poly line-end extension rule and failure criteria. The assumed process is one that does not define poly line-ends with a separate cut-exposure.

loss, stitching is needed to conform many problematic layout patterns to DP without the need for layout modification (by breaking odd cycles in the conflict graph as in the example of Figure 2.3).

One of the main reasons for yield loss associated with stitches is overlay errors between the first and second exposures in DP. Therefore, the minimum overlap-length rule – a.k.a. overlap margin – has a direct impact on yield. Consider for example a stitch in the center of a vertical line as shown in Figure 2.4. An overlay in the Y direction may result in an insufficient mask overlap and cause an open defect after line-end pullback; an overlay in the X direction may cause the wire to become too narrow at the stitch leading to failure. In addition, the overlap-length rule affects the DP-compatibility of the layout. The larger the overlap length is, the lesser candidate-stitch locations the layout will have. Hence, while a large and conservative overlap-length rule is likely to inhibit most yield loss of stitches caused by overlay, such overlap length may result in excessive re-design efforts and area overhead to ensure the layout conforms to DP. Another design rule that may affect the yield loss of stitches due to overlay (in x direction for the example in Figure 2.4) is the line-width rule. Clearly, failure from narrowing for initially narrow lines is more severe than such failure in wide lines.

The minimum line-spacing design rule impacts the delay variation of wires caused by overlay errors between the two exposures of DP [16, 17, 18, 19]. Since overlay translates directly into line-spacing variation (with a positive dual-line process), the coupling capacitance between neighboring wires on different exposures will be affected by both overlay and the minimum line-spacing rule. The line-spacing rule has also a direct impact on the layout area. While a large line-spacing rule may confine the wire-delay variation, such spacing rule is likely to induce an area overhead.

Poly line-end extension over active rule (LEE) is subject to failure due to overlay error between the polysilicon and the active layer. Consider for example an overlay instance shown in Figure 2.5. An overlay error in the Y direction may lead to a low resistance path between source and drain of the transistor after line-end pullback[1]. Therefore, LEE has direct impact on yield since a larger poly line-end extension is likely to inhibit most yield loss caused by overlay. In addition, poly line-end extension rule also affects the design area. The larger the extension rule value is, greater is the amount of folding in poly gates which will result in a larger design area. Hence there is any interesting trade-off between yield and area(in case of LEE) or designer effort(in case of min. overlap length)

2.3 Overlay and Yield Modeling

The yield from overlay, $Y_{overlay}$, is equal to the probability of survival (POS) from the overlay error remaining after any overlay correction and referred to as residue[2]. Overlay-residue vector components in x and y directions are typically described by a normal distribution with zero mean and process-specific 3σ estimate. Therefore, given the fraction, p, of the overlay-residue variance breakdown between field-to-field and within-field components, the probability distribution of each type of overlay error can

[1] Instead of simple geometric line-end failure model, a more complex electrical failure model [20] can be used as well

[2] Coupled with the lithographic line-end pullback, which we model as an offset of fixed value.

be calculated as follows:

$$\begin{aligned} f_{field-to-field} &= \frac{1}{\sigma\sqrt{2\pi p}} e^{\frac{-u^2}{2p\sigma^2}}, \\ f_{within-field} &= \frac{1}{\sigma\sqrt{2\pi(1-p)}} e^{\frac{-v^2}{2(1-p)\sigma^2}}. \end{aligned} \tag{2.9}$$

The probability for each type of overlay error to have a value between a and b is then given by

$$\begin{aligned} P_{field-to-field} &= \frac{1}{\sigma\sqrt{2\pi p}} \int_a^b e^{\frac{-u^2}{2p\sigma^2}} \, \mathrm{d}u, \\ P_{within-field} &= \frac{1}{\sigma\sqrt{2\pi(1-p)}} \int_a^b e^{\frac{-v^2}{2(1-p)\sigma^2}} \, \mathrm{d}v. \end{aligned} \tag{2.10}$$

It is assumed that that overlay residue coming from *field-to-field sources* (i.e., wafer-level) is identical at all features of the same layer in the design. The overlay residue coming from *within-field sources*, however, can be different at features of the same die.

overlay residue is modeled (within-field and field-to-field) as partly systematic and partly random.

2.3.1 Yield model with purely random overlay residue

The random part of the overlay residue comes from un-modeled overlay components as well as imperfections in the correction process. In our yield model, the random component of the within-field overlay residue is assumed to be independent from one feature to another across the design while field-to-field overlay residue is assumed to be fully correlated for all the features in the design. Hence, when the overlay residue is entirely random, the die yield caused by overlay in one direction is equivalent to the probability of all features – say n – in the design surviving such overlay error and it is

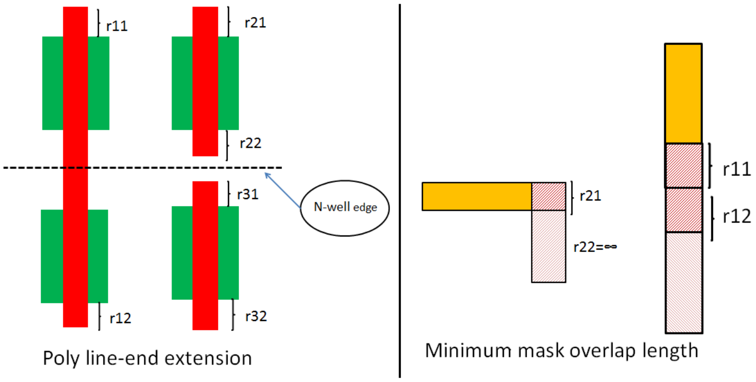


Figure 2.6: Example of various overlay instances scenarios for poly line-end extension and minimum mask overlap length

calculated as follows:

single instance:

$$POS_{within-field} = \frac{1}{\sigma\sqrt{2\pi(1-p)}} \int_{-r_{12}}^{r_{11}} e^{\frac{-v^2}{2(1-p)\sigma^2}} \, \mathrm{d}v; \tag{2.11}$$

where r_{11} and r_{12} are the extension rule values for the overlap instance (e.g. Figure 2.6)

all instances n in the design:

$$POS_{within-field} = \prod_{i=1}^{n} \left[\frac{1}{\sigma\sqrt{2(1-p)\pi}} \int_{-r_{i2}}^{r_{i1}} e^{\frac{-v^2}{2(1-p)\sigma^2}} \, \mathrm{d}v \right]; \tag{2.12}$$

Now taking into account the wafer-level random component, say u, Die yield is given by

$$Y_{x|y} = \frac{1}{\sigma\sqrt{2p\pi}} \int_{u_{min}}^{r_{max}} \prod_{i=1}^{n} \left[\int_{-r_{i1}-u}^{r_{i2}-u} \frac{e^{\frac{-v^2}{2(1-p)\sigma^2}}}{\sigma\sqrt{2(1-p)\pi}} \mathrm{d}v \right] e^{\frac{-u^2}{2p\sigma^2}} \, \mathrm{d}u, \tag{2.13}$$

where r_{max} is the value of the maximum of all given extension rule in the design (see Fig. 2.8). u and v are variables denoting overlay. For yield calculation purpose, maximum value of wafer-level random error u is taken as r_{max} since any overlay error

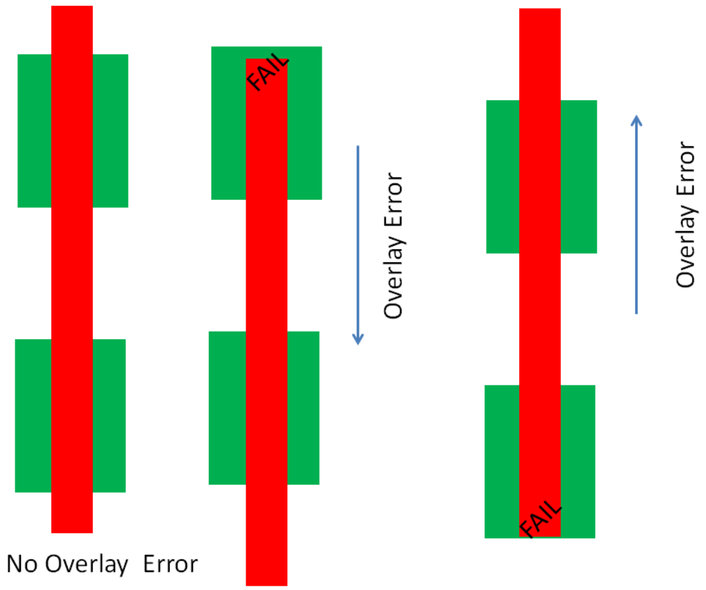


Figure 2.7: Example of overlay instance scenarios for which failure can occur because of overlay error in both direction

beyond this limit will cause all features to fail and hence yield will be zero. Minimum value of u , say u_{min}, can either be -r_{max}, when overlay error causes failure in both direction (for e.g. +/- y direction in Figure 2.7) or $-\infty$, when the overlay in a particular direction effectively increases the overlap at the feature (for e.g. Figure 2.9(c)). r_{i1} and r_{i2} represents the values of the i^{th} instance of layer-overlap in the design (e.g. Figure 2.6).

2.3.2 Yield model in presence of systematic overlay residue

The systematic part of the overlay residue comes from un-corrected high-order overlay components (up to the sixth-order components in our experiments). The reason for not correcting for those high-order terms is because scanner tools have limited correction capability (e.g., previous-generation tools could not correct terms beyond the third order) and sophisticated alignment and overlay measurement strategies needed for high-order terms correction reduces the manufacturing throughput [3]. For yield computation, we divide the design into grids (see Figure 2.10). While we assume the

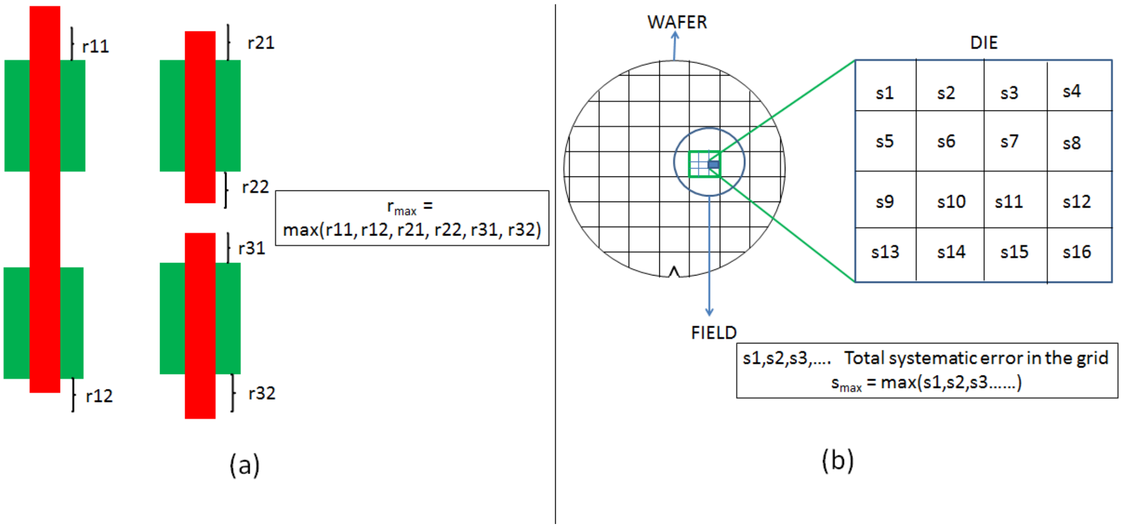

r11
r21
r22
r31
r12
r32
r_{max} =
max(r11, r12, r21, r22, r31, r32)
(a)
WAFER
DIE
s1
s2
s3
s4
s5
s6
s7
s8
s9
s10
s11
s12
s13
s14
s15
s16
FIELD
s1,s2,s3,.... Total systematic error in the grid
s_{max} = max(s1,s2,s3......)
(b)

Figure 2.8: example illustrating how r_{max} and s_{max} are calculated

field-to-field systematic overlay residue is identical at all features in the field, we assume the within-field systematic overlay residue is identical for features of the same grid only but different from one grid to another. Therefore, the total systematic overlay residue at an overlap-instance is the sum of the systematic within-field overlay residue in the grid containing the instance and the systematic field-to-field overlay residue of the field containing the instance. Unmodeled overlay error is assumed to be purely random. This random residue is further broken down into wafer-level component and field-level component. Therefore, given the fraction, p, of the random overlay-residue variance (σ^2) breakdown between field-to-field and within-field and systematic overlay residue as described earlier , the probability of survival from within-field overlay for a single instance, all instances in a grid, and the entire die is as follows:

single instance with systematic overlay s:

$$POS_{within-field} = \frac{1}{\sigma\sqrt{2\pi(1-p)}} \int_{-r_{12}-s}^{r_{11}-s} e^{\frac{-v^2}{2(1-p)\sigma^2}} \, \mathrm{d}v; \tag{2.14}$$

where r_{11} and r_{12} are shown in Figure 2.6

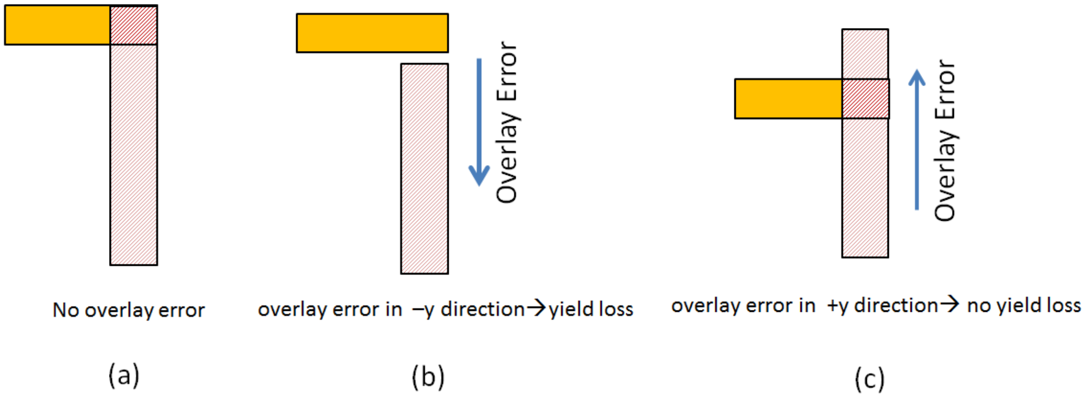

Overlay Error
Overlay Error
No overlay error
overlay error in −y direction→yield loss
overlay error in +y direction→ no yield loss
(a)
(b)
(c)

Figure 2.9: Example of an overlay instance causing failure only in one direction (a) stitch in a L-shaped wire segment (b) for no failure, overlay error should be less than mask overlap length in the given direction (c)no failure in this direction for any value of overlay error

all instances (n/g) of same grid of design with g grids:

$$POS_{within-field} = \prod_{j=1}^{n/g} \left[\frac{1}{\sigma\sqrt{2(1-p)\pi}} \int_{-r_{j2}-s}^{r_{j1}-s} e^{\frac{-v^2}{2(1-p)\sigma^2}}\, \mathrm{d}v \right]; \tag{2.15}$$

all instances in the die:

$$POS_{within-field} = \prod_{i=1}^{g} \prod_{j=1}^{n/g} \left[\frac{1}{\sigma\sqrt{2\pi(1-p)}} \int_{-r_{ij1}-s_i}^{r_{ij2}-s_i} e^{\frac{-v^2}{2(1-p)\sigma^2}}\, \mathrm{d}v. \right], \tag{2.16}$$

where s_i is the systematic overlay residue at the center of the i^{th} grid, which includes field-to-field and within-field sources. A model to estimate s_i will be presented in the next section.

Now taking into account the wafer-level random component,say u, Die yield is given by

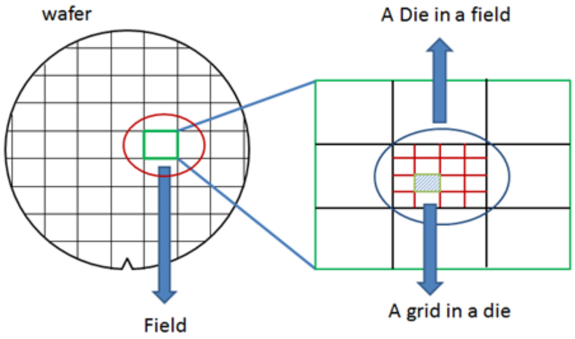


Figure 2.10: Pictorial representation of wafer, exposure fields, dies and the grid structure on each die.

$$Y_{x|y} = \frac{1}{\sigma\sqrt{2\pi p}} \int_{u_{min}}^{r_{max}+s_{max}} \prod_{i=1}^{g} \prod_{j=1}^{n/g} \left[\int_{-r_{ij2}-u-s_i}^{r_{ij1}-u-s_i} \left(\frac{e^{\frac{-v^2}{2(1-p)\sigma^2}}}{\sigma\sqrt{2\pi(1-p)}} \mathrm{d}v \right) \right] e^{\frac{-u^2}{2p\sigma^2}} \mathrm{d}u, \tag{2.17}$$

where r_{ij1} and r_{ij2} are the values of the j^{th} overlay instance in the i^{th} grid, u is the random component of the field-to-field overlay residue and s_{max} (see Fig. 2.8) is the maximum systematic overlay error in the die. The maximum value of u is chosen to be (r_{max}+s_{max}) because beyond this limits all features will definitely fail and POS will be zero. The minimum value of u, say u_{min}, can either be $-(s_{max} + r_{max})$ when overlay error causes failure in both direction or $-\infty$, when the overlay in particular direction effectively increases the overlap at the feature. Table 2.1 summarizes all the assumptions made in the derivation of the yield model of Equation 2.17.

Finally, the overall yield from overlay in any direction is approximated as the prod-

Table 2.1: Summary of all assumptions made in the derivation of the yield model of Equation 2.17.

Overlay Component	**Assumption**
Random field-to-field	Identical for all feature within the same field
Systematic field-to-field	Identical for all feature in the same field
Random within-field	Independent for all feature in the same field
Systematic within-field	Identical for all feature within the same grid

uct of the yield in the x and y directions[3]:

$$(Y)_{overlay} = (Y)_x \times (Y)_y. \tag{2.18}$$

2.3.3 Modeling the systematic overlay residue

In this section, we describe our method for estimating the systematic overlay residue at the center of each grid (s_i in Equation 2.17).

Systematic overlay error is typically described using a polynomial model function of wafer and field levels coordinates as in [21]. When the maximum polynomial order of the model is m but correction is performed for up to the k^{th} order only, then the polynomial model can be used to describe the uncorrected systematic overlay error s_x in x direction and s_y in y direction as follows:

$$\begin{aligned} s_x &= \sum_{q=k+1}^{m} \sum_{t=0}^{q} a_{qt} * x^t * y^{q-t} + \sum_{q=k+1}^{m} \sum_{t=0}^{q} b_{qt} * X^t * Y^{q-t} \\ s_y &= \sum_{q=k+1}^{m} \sum_{t=0}^{q} c_{qt} * x^t * y^{q-t} + \sum_{q=k+1}^{m} \sum_{t=0}^{q} d_{qt} * X^t * Y^{q-t} \end{aligned} \tag{2.19}$$

where x, y are the field level coordinates and X, Y are the wafer level coordinates. a

[3] This equation slightly underestimates the yield loss as, in reality, yield loss from overlay is defined by the area of the overlap region, which is influenced by overlay in both x and y directions.

Table 2.2: σ^2 values in nm^2 for second to sixth polynomial order of field-to-field and within-field overlay sources using overlay characterization data reported in [3].

Order	**Field-to-field (X)**	**Within-field (X)**	**Field-to-field (Y)**	**Within-field (Y)**
2^{nd},3^{rd}	$0.14nm^2$	$0.17nm^2$	$0.22nm^2$	$0.055nm^2$
4^{th},5^{th}, 6^{th}	$0.045nm^2$	$0.028nm^2$	$0.037nm^2$	$0.037nm^2$
Random	$0.07nm^2$	$0.07nm^2$	$0.028nm^2$	$0.028nm^2$

and c are the coefficients for field-level and b and d are the coefficients for wafer-level terms.

The coefficients of the model of Equation 2.19 can be estimated from overlay measurement data. For our experiments, we estimate these coefficients as follows. We use overlay variance values for each polynomial order reported in [3], where a source of variance analysis has been conducted to characterize overlay error at 32nm node up to the sixth order wafer and sixth order field components. Since our experiments were performed for the 14nm node, we scaled the variances by a factor of 2 to account for possible improvements of scanner tools correction accuracy. We also assume that the source of variance coming from the random component is split equally between field-to-field and within-field overlay sources. Table 2.2 shows the σ^2 values used in this work for each order. To simplify the estimation of the model's coefficients using variance values, coefficients for all components of a given order are assumed to be same (i.e., for a given q, all a_{qt}, b_{qt}, c_{qt} and d_{qt} coefficients of Equation 2.19 are the same). Using the coordinates at a number of points in the wafer and field, the coefficient values of each polynomial order are then inferred from Equation 2.19 and the estimated variance values. For example, the coefficient of the within-field second polynomial order, a_2, can be calculated as follows:

$$s_x(2nd\ order\ within-field) = a_2 * (x^2 + y^2 + xy) \tag{2.20}$$
$$a_2 = \frac{\sigma_{2^{nd}orderfield}}{\sigma(x^2 + xy + y^2)}.$$

Table 2.3: Coefficients for the systematic overlay residue model of Equation 2.19 using a field size of 33x26mm and assuming 63 fields per wafer.

Within-field		Field-to-field	
a_{20}, a_{21}, a_{22}	0.5203	b_{20}, b_{21}, b_{22}	0.0090
$a_{30}, a_{31}, a_{32}, a_{33}$	0.2681	$b_{30}, b_{31}, b_{32}, b_{33}$	4.8183×10^{-4}
$a_{40}, a_{41}, a_{42}, a_{43}, a_{44}$	0.0811	$b_{40}, b_{41}, b_{42}, b_{43}, b_{44}$	3.4968×10^{-5}
$a_{50}, a_{51}, a_{52}, a_{53}, a_{54}, a_{55}$	0.0491	$b_{50}, b_{51}, b_{52}, b_{53}, b_{54}, b_{55}$	2.272×10^{-6}
$a_{60}, a_{61}, a_{62}, a_{63}, a_{64}, a_{65}, a_{66}$	0.0338	$b_{60}, b_{61}, b_{62}, b_{63}, b_{64}, b_{65}, b_{66}$	2.592×10^{-7}
c_{20}, c_{21}, c_{22}	0.3025	d_{20}, d_{21}, d_{22}	0.0114
$c_{30}, c_{31}, c_{32}, c_{33}$	0.1543	$d_{30}, d_{31}, d_{32}, d_{33}$	6.0713×10^{-4}
$c_{40}, c_{41}, c_{42}, c_{43}, c_{44}$	0.0933	$d_{40}, d_{41}, d_{42}, d_{43}, d_{44}$	3.1309×10^{-5}
$c_{50}, c_{51}, c_{52}, c_{53}, c_{54}, c_{55}$	0.0565	$d_{50}, d_{51}, d_{52}, d_{53}, d_{54}, d_{55}$	2.0141×10^{-6}
$c_{60}, c_{61}, c_{62}, c_{63}, c_{64}, c_{65}, c_{66}$	0.0389	$d_{60}, d_{61}, d_{62}, d_{63}, d_{64}, d_{65}, d_{66}$	2.2976×10^{-7}

Table 2.3 shows all coefficient values that we use in our experiments.

2.4 Evaluation of Rules Impact on Design

This section presents the methods we used for evaluating the design impact of overlay-related rules.

2.4.1 Evaluation of Design Area

Evaluation for the design area associated with poly line-end extension rule is achieved using the Design Rules Evaluator (UCLA_DRE[4]) from [22]. To evaluate area, DRE essentially creates a virtual standard-cell layouts from a set of DRs and transistor-level netlists of standard-cells. Using estimated area of the virtual layouts as well as instance-counts of cells in the design, the total cell-area in the design is evaluated.

[4] UCLA_DRE is available for public use and can be downloaded at nanocad.ee.ucla.edu/Main/DownloadForm

2.4.2 Evaluation of DP-Compatibility

A layout is said to be DP-compatible, if its features can be assigned to the first and second masks without any spacing violations in each mask-layout. Hence the number of spacing violations was chosen as the metric for DP-compatibility. The mask-assignment algorithm of [23] was used, which guarantees to a mask-assignment solution if one exists. To further reduce the number of spacing violations in DP-incompatible layouts, the algorithm was modified to flip the mask-assignment of violating features if the flipping reduces the number of violations.

2.4.3 Evaluation of Overlay-Induced Delay Variation

The method described in [16] to evaluate the electrical variation of wires formed with DP. In essence, the method consists modeling the wire resistance and capacitance, which are the main elements of wire delay, as a function of overlay and its different components. Since the method in [16] assumes a linear overlay model, we limit our experiments on the minimum line-spacing rules to the case of overlay control with a linear model.

2.5 Experimental Results

In this section, Interaction of DP related design rules and poly line-end extension rule with overlay at the 14nm technology node was explored.

2.5.1 Testing setup

Experiments were performed using AE18 design from [24], synthesized using Nangate Open Cell-Library [25], and FreePDK open-source process [26]. Since the PDK and standard cell-library are for a 45nm process, all rules and layouts were scaled by $2\times\sqrt{2}$ to run the experiments for the 14nm node (M1 half-pitch becomes 23nm). In all exper-

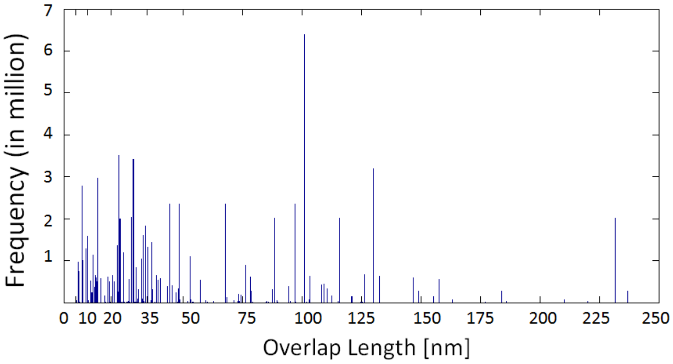


Figure 2.11: Histogram of overlap-length values in the design.

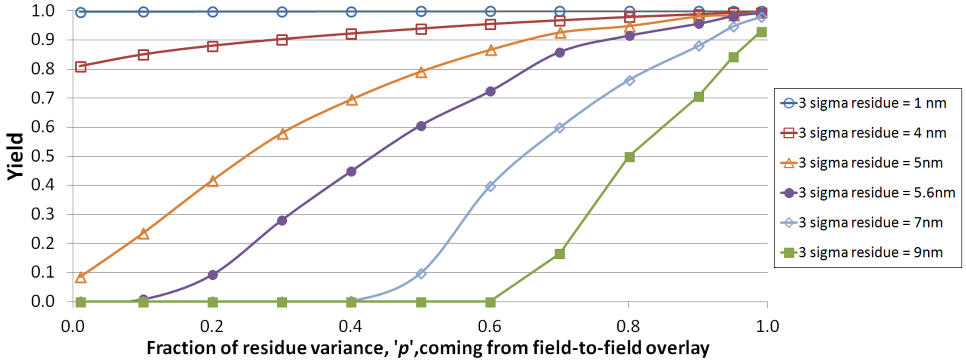


Figure 2.12: Plots showing the effects of the breakdown of overlay among field-to-field and within-field overlay components for different overlay-residue values.

iments, we assume a line-end pullback of 5nm. We use a field size of $33 \times 26mm$ and a design grid size for yield computation of $2.5 \times 2.5mm$.

Since the area of the benchmark design is relatively small (10K-cell instances), we normalize the yield results to a $100mm^2$ die area to have a realistic number of structures that are susceptible to yield loss (e.g, number of stitches in our experiments). We determine for the base case in each experiment the number of design copies that can fit in $10 \times 10mm$ chip size and find the corresponding number of stitches as well as the overlap length and direction of stitches in the design[5]. Figure 2.11 depicts a histogram of overlap-length values for all stitches in the design.

[5] Note that, for corner stitches, we assume that half are in vertical lines and the other half are in horizontal lines to estimate the yield loss for the open-circuit failure shown in Figure 2.4(b). Layout context effects for more accurate modeling is part of ongoing work.

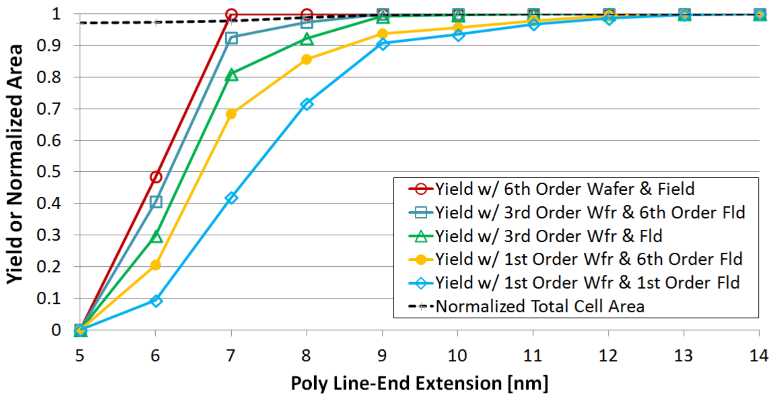


Figure 2.13: Plots showing the interaction between the polysilicon line-end extension rule and overlay control and their impact on yield and die area.

2.5.2 Projecting the overlay capability of the process

In the first experiment, the framework is used to analyze the yield loss for various values of variance of unmodeled residue and breakdown p of the residue between field-to-field and within-field components. This experiment has been done for Poly line-end extension rule (LEE) value of 13nm and for first order wafer/first order field correction model. Figure 2.12 plots the yield of LEE for different cases. The results show that the larger the fraction of within-field overlay component, the larger the yield loss. The plots also identify the value of the residue for which a close to 100% yield can be achieved for a given overlay breakdown between field-to-field and within-field components. Such result can project the overlay capability of the process and serve as early hint for design-rules development.

2.5.3 Interaction between Poly Line-end extension rule and overlay control

The framework was also used to evaluate poly line-end extension rule (LEE) . Figure 2.13 shows yield and design area curves as minimum poly line-end extension rule is varied for various overlay control options. Impressively, increasing the rule value by just a few nanometer can allow the use of less complex overlay control while keeping yield and die area virtually unaffected. For example, increasing the rule from 8nm to

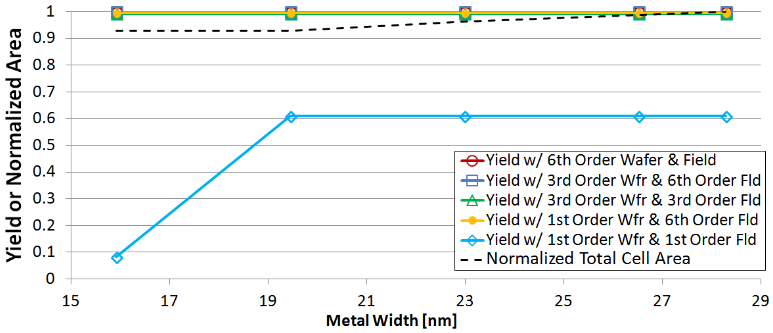


Figure 2.14: Plots showing the interaction between the minimum line-width rule and overlay control and their impact on yield and layout area of the design with minimum overlap-length rule of 14nm.

9nm would allow the use of third-order wafer and field-level model instead of sixth-order wafer and field-level model with negligible impact on area and yield (less than 1% area increase while yield drops from 100% to 99.3%). This can have important implications such as increased throughput and extending the lifespan of current scanner tools that are not capable of high-order overlay correction.

2.5.4 Interaction between DP-related rules and overlay control

The framework was also used to study the effects of DP rules on stitch failure and the area and DP-compatibility of the design. In one experiment, the line-width was varied by few nanometers from the nominal value at 23nm and report the yield loss and the normalized design area for the different overlay-modeling options. The results, depicted in Figure 2.14, show that the line-width has almost no impact on stitch failure. The reason is that the nominal rule value is large enough to avoid stitches failure from overlay in the direction perpendicular to lines. Hence, stitches yield loss may be neglected when deciding on the minimum line-width rule. It can also be clearly seen from Figure 2.14 that the first-order wafer/first-order field-level overlay model, i.e., the linear model, is insufficient for controlling overlay at the 14nm node.

In another experiment, the minimum mask-overlap length was varied and the yield

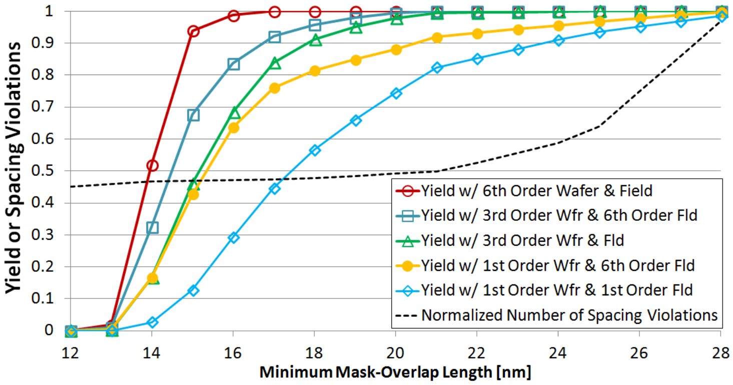


Figure 2.15: Plots showing the interaction between the overlap-length rule and overlay control and their impact on yield and DP-compatibility of the design at the nominal line-width of 23nm.

loss and number of DP-spacing violations in the design for the different overlay-modeling options was reported. The results, depicted in Figure 2.15[6], show the strong interaction between the rule value and overlay-control options as well as the overall impact on yield and DP-compatibility. Interestingly, few nanometer changes in the rule value may allow the use of a less stringent overlay control without significant impact on DP-compatibility. For example, increasing the minimum mask-overlap length from 19nm to 20nm would allow the use of third-order wafer/sixth-order field-level overlay model instead of sixth-order wafer/sixth-order field-level model while yield remains at 100% and DP-spacing violations increase by just 1%.

The last experiment is about studying the effects of the line-spacing rule on wire-delay variation and layout area. The line-spacing rule is changed from the nominal value at 23nm by few nanometers. The results, given in Figure 2.16[7], indicates that the impact of this rule on the average RC variation is minor, while its impact on area is considerable. Hence, tweaking the line-spacing rule with the intention of reducing the electrical variation is ineffective.

[6] The number of DP-spacing violations are normalized with respect to the case with the largest number and DP mask-assignment of the layouts was performed using a minimum same-color spacing of 1.5× the half-pitch.

[7] Note that there is always some electrical variation due to overlay errors with any realistic line-spacing rule.

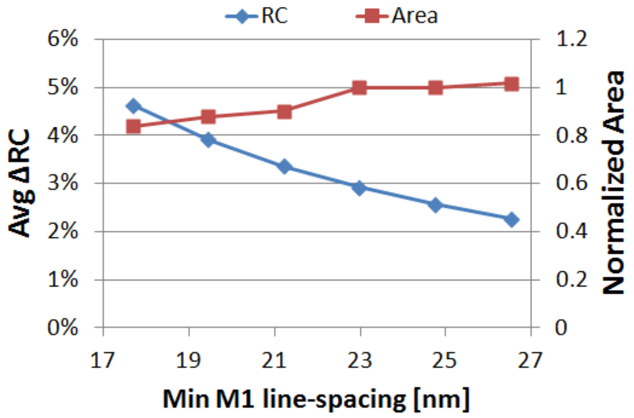


Figure 2.16: Plot for the average ΔRC and the normalized design area for different values of the minimum line-spacing rule.

2.6 Conclusions

The thesis proposes a general framework, built over UCLA_DRE, to explore the interactions between design rules, overlay characteristics, and overlay modeling options. Yield loss due to overlay is modeled as a function of design-rule values and the overlay characteristics. The proposed framework is the first of its kind and it can be used during process development to better define overlay-related design rules and project overlay requirements for the process. For demonstration purposes, the framework was used in this work to explore DP and overlay-related rules for the M1 layer as well as the polysilicon line-end extension over active rule at the 14nm node. Important conclusions could be drawn from our experimental results. One result shows that increasing the minimum mask-overlap length by 1nm would allow the use of a third-order wafer/sixth-order field-level overlay model instead of a sixth-order wafer/sixth-order field-level model with negligible impact on design. Another result shows that the minimum line-width and spacing rules have an insignificant impact yield and electrical variation. Although our studies were performed for a few rules at the M1 and poly layers, the framework is more general and can be used to explore other inter-layer overlay rules, for different MP technologies, and for different layers.

CHAPTER 3

Chip-level Design Rule evaluation

3.1 Overview

An overview of Chip-DRE is depicted in Figure 3.1. The framework takes the following inputs: transistor-level netlists (SPICE) of cells, rules and their values, estimates of process control (e.g., overlay error distribution), and cell-usage statistics of the design to evaluate the rules on. In Chip-DRE, only the values of rules under evaluation are modified while all others remain unchanged. This modified set of rules is then used to estimate the cell-layout and perform the design-level evaluation.

Concisely, the first stage of Chip-DRE is to estimate the cell layout/area and cell delay for a given set of rules. If the cell delay changes in comparison with the delay obtained using a base set of rules, the cell-delay change is converted into a delay-scaling factor which is used to scale the timing characteristics of the standard-cell library (in Liberty file format). A semi-empirical model is then used to estimate the impact of cell-delay change on the design overall cell-area. The model essentially predicts gate-sizing and buffer-insertion to meet the setup timing requirements with the new cell-delay characteristics. The model is derived analytically but it also contains few parameters that are fitted to actual SPR (Synthesis, Placement and Routing) data for improved accuracy. In the second stage of Chip-DRE, another semi-empirical model is used to predict how the cell-area change translates into chip-area change. This cell-to-chip area model also contains parameters that are fitted to actual SPR data for improved accuracy. The final stage of Chip-DRE, chip-level functional yield is estimated and a unified design-quality

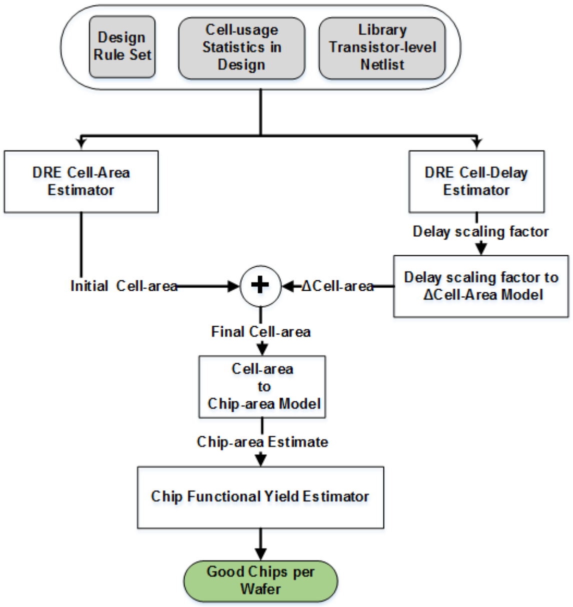


Figure 3.1: Overview of Chip-DRE and its main components.

metric, number of “good chips per wafer” (GCPW), is calculated.

3.2 Cell-area estimation

The cell area estimator is based on the existing cell-level DRE framework from [1]. Cell-level DRE estimates the cell area through fast generation of front-end-of-line (FEOL) layers and congestion-based estimation of wiring area. This approach has been shown to be well-suited for efficient rule exploration while resulting in good accuracy; for a 96-cell library, the cell-area for a given rule set is estimated with $< 1\%$ absolute error with a run-time of 80 minutes or $\sim 2\%$ absolute error with a run-time of just 10 minutes.

The cell-layout estimation of the cell-level DRE was extended in this work to enable evaluation of state-of-the-art technologies at the chip-level.

For chip-level evaluation, DRE was modfied to generate I/O pin segments and the physical specifications of the technology and standard-cells (in Library Exchange Format or LEF format)). For the purpose of LEF creation, pin segments are kept at mini-

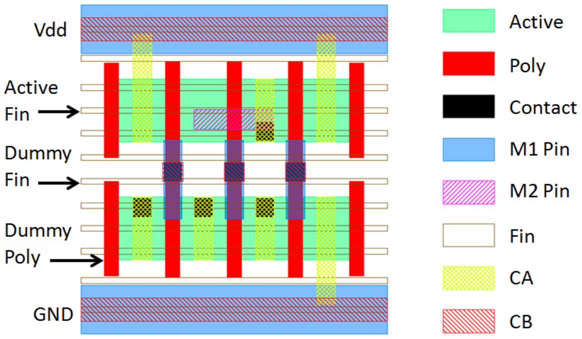


Figure 3.2: Example layout for OAI21_X1 cell generated by Chip-DRE with FinFETs, local interconnects, and DRC-clean I/O pin segments.

mum possible dimensions while meeting the minimum area design rule.

3.3 Cell-Delay Estimation

A crucial aspect of Design Rule Evaluation is the assessment of the impact of the DRs on performance. To be able to characterize a digital chip-level delay, it is required to model the delay for each standard cell. First-order delay models are emplyed in order to have a fast and approximate delay estimation.

3.3.1 Cell Delay Model

To characterize the cell rise or fall delay, the cell is considered as a sequence of stages, where each stage can be considered a gate. The delays of these stages are then added up. For each stage, all paths connecting the output to the power supply (Vdd or ground) are enumerated. An RC tree is constructed for each path and Elmore delay [27] is applied to compute the path delay [28]. The worst case pull up and pull down delays are determined for each stage. Identical paths (paths which switch simultaneously) are considered as parallel resistances and their capacitances are added up. We apply an RC approximation for each transistor. The capacitance model [28] considers the gate capacitance (including channel and overlap capacitances) as well as diffusion capacitances, and accounts for Miller effect.

3.3.2 Liberty Timing File Generation

For the baseline set of design rules, we assume that liberty file already exists.[1]. To generate the liberty file for virtual standard-cell library corresponding tothe set of rules under evaluation, the worst-case pull-up and pull-down delays for the gates are computed as explained in section 3.3.1. This is also done for the baseline set of design rules to create a reference gate delay (computed by DRE). The ratios between the gate delays in the case of design rules under evaluation and those of the baseline design rules are used to scale the baseline liberty file to obtain an estimated liberty file for the design rules under evaluation. For sequential elements, their hold and setup times are left unchanged (same as baseline liberty file), and their clock to output delay is scaled by the same scaling factor as inverter. The entire flow of generating layouts, estimating delays and generating .libs within Chip-DRE takes less than 49 minutes for a 100 cell library as opposed to commercial library characterization tools which take several CPU days. This basically makes Chip-DRE style delay estimation the only viable way to infer delay dependence on rules.

3.4 Cell-Delay to Design Cell-Area Modeling

One of the major issues Chip-DRE addresses which typical cell-based design rule optimization approaches suffer from is the effect of timing optimization during physical synthesis on area. Earlier, it was demonstrated how Chip-DRE estimates delay of each standard-cell using first-order delay models. In this section, a cell-delay to chip-area model is developed .

Two of the most common and powerful timing optimization knobs employed by

[1] This could be a characterized or scaled version from a previous technology node. The absolute values of delays in the .lib are not very important for Chip-DRE as we are more interested in delay changes with rule changes.

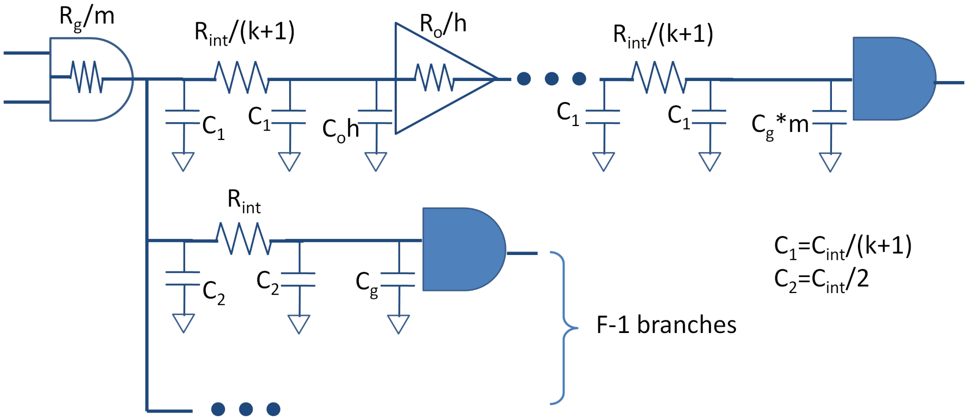


Figure 3.3: Critical path delay model illustration

physical synthesis tools are gate sizing and buffer insertion [2]. As delay of standard cell increases, we can expect an increase in the chip area. Previous work [29] has experimentally characterized the impact of timing guardband reduction on some metrics of the circuit by running synthesis, place and route for several scaled libraries. However this is impractically slow to explore design rule choices. Moreover, the work of [30] has demonstrated that little noise can have huge effect on place-and-route solution quality; this makes using a model-based estimate even more attractive. Delay variability is modeled as a worst-case delay change (i.e., a "corner" model) and hence is implicitly accounted for in the delay-to-area model.

In this model, we focus on the delay of critical path, since we assume that the change in chip area is due to the gate sizing and buffer insertion performed on the critical path. Table 3.1 lists the definition of the used symbols. An illustration of a critical path logic stage (including interconnect and added buffers) is shown in figure 3.3. The delay of one logic stage (T_{ls}) is - like in cell delay estimator - estimated by applying Elmore delay on the equivalent RC tree.

Similar to [31], optimal sizing and buffer insertion can be done analytically. With no buffers on the critical path, the delay $T_{ls_{nobuffers}}$ is as follows.

[2] Though we realize that several other timing optimizations may be employed, as we show later, for model form development purposes, sizing and buffering is sufficient

$$T_{ls_{nobuffers}} = 0.69[\frac{R_{g\,F}C_{int}}{m} + R_gC_g + \frac{R_{int}C_{int}}{2} + mR_{int}C_g + \frac{R_g}{m}(F-1)C_g] \tag{3.1}$$

The optimum gate size can be calculated as below to minimize the above stage delay.

$$m_{opt} = \sqrt{\frac{F\,R_gC_{int} + R_gC_g(F-1)}{R_{int}C_g}} \tag{3.2}$$

Next, delay is improved by adding buffers of optimal number (k_{opt}) and optimal size (h_{opt}) between gates on the critical paths. Similar to the procedure followed for gate sizing, k_{opt} and h_{opt} are picked, successively, in order to minimize critical path delay.

The total area of critical paths is the sum of the areas of the gates on critical paths in addition to the areas of the added buffers. Chip-DRE calculates average change in delay of cells (used in a design) by changing design rules (as discussed in the previous section) as a delay scaling factor S. For simplicity, we assume that the scaling of gate delay is due to a change in the driving resistance of the gate. Thus R_g=S* Rg_0 where Rg_0 is the gate resistance at the nominal delay. Also the buffer resistance is scaled by S, R_o=S* Ro_0. Thus estimate of critical path area as a function of as the delay scaling factor S is given below.

$$A_{CP} \sim c1 * \sqrt{S} + c2 * S^{0.25} + c3 \tag{3.3}$$

The model was fitted and validated on two benchmark designs [24] by running synthesis, placement and routing (using Cadence Encounter) in order to obtain the experimental design area. Figure 3.4 shows the estimated cell area for MIPS in comparison

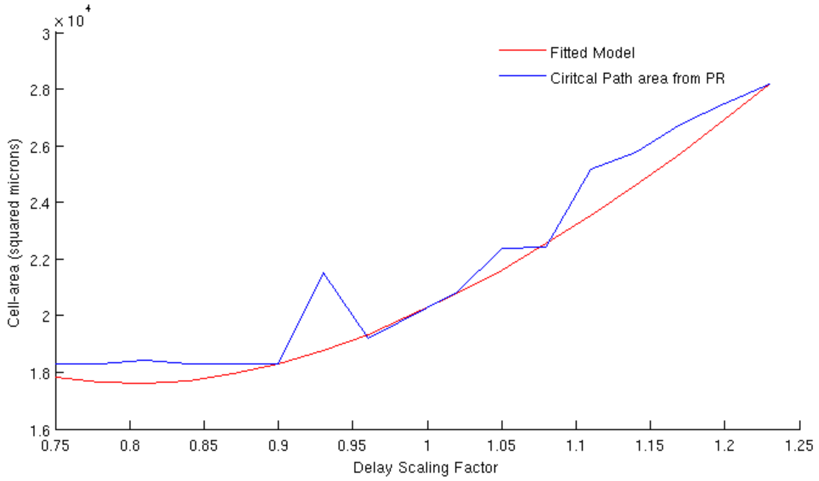


Figure 3.4: Model validation on MIPS

Table 3.1: Symbols used in the model

Symbol	**Definition**
T_{ls}	Delay of a logic stage
$T_{ls_{nobuffer}}$	Delay of logic stage with no buffers inserted
C_g	Input capacitance of the gate
R_g	Gate resistance
F	Average fanout of all instances on critical paths
R_o	Resistance of minimum size inverter
C_o	Input capacitance of minimum size inverter
R_{int}	Average resistance of interconnect for a logic stage
C_{int}	Average capacitance of interconnect for a logic stage
C_{unit}	Unit capacitance per area
c1, c2, c3	Correction coefficients fitted empirically

to the cell-area obtained after placement and routing.

3.5 Chip-Area and Yield Modeling

3.5.1 Minimum Routable Area

Minimum routable area (MRA) of a design requires the estimation of maximum utilization at which the number of DRC errors cease to be zero. This implies that for finding MRA, multiple Place & Route (P&R) runs are required that makes the whole process time consuming (detailed routing being the main culprit). For instance, an experiment to estimate MRA of AES ($\sim 10K$ gate design) using binary search took 14 hrs (as shown in column 6 of Table 3.2). Such excessive runtime makes chip-level evaluation of multiple design rules impractical.

To overcome runtime problem, we propose a new methodology that estimates MRA using global routing congestion estimates (AEGR). Global routing congestion estimates requires the estimation of wiring demand and wiring supply on each of the global routing cell called G-cell. A G-cell represents a fixed number of available routing tracks in each layer. If wiring demand exceeds supply, the detailed routing is unlikely to implement a design rule correct wire pattern. This is the most common way to detect wire congestion. If the wiring demand is less than supply, however, the G-cell is likely to be routable. Congestion in an arbitrary grid is thus given by

$$C = \frac{\text{routing demand (d)}}{\text{routing supply (s)}} \tag{3.4}$$

Now suppose there are $m \times n$ G-cells used in a rectangular floorplan, congestion in the i^{th} row and j^{th} column is given by

$$C_{ij} = \frac{d_{ij}}{s_{ij}}, \tag{3.5}$$

and average congestion, C_{avg} and peak congestion, C_{peak} are given by

$$\begin{aligned} C_{avg} &= \sum_{i=1}^{m}\sum_{j=1}^{n} C_{ij}/(mn) && (3.6)\\ C_{peak} &= \max_{i,j}(C_{ij}). && (3.7) \end{aligned}$$

P&R tools cannot resolve all instances of congestion and for very high congestion values tool might not find enough unused G-cells to successfully route the design. Hence we propose that there exist a threshold on congestion beyond which tool cannot successfully route the design. Based on this we define a metric, $m(u)$, in the following manner

$$m(u) = \alpha * C_{peak}(u) + \beta * C_{avg}(u), \tag{3.8}$$

where C_{avg} is the average congestion over all G-cells and C_{peak} is the maximum congestion over all G-cells , and α and β are the tool dependent parameters. The utilization for which $m(u)$ is 1 is classified as the maximum utilization of the design, say u_{max}.

To further refine the estimation of maximum utilization, we run detailed routing in the range $[0.9u_{max}, 1.1u_{max}]$ to get two utilization values where number of DRC errors is greater than zero. Then linear extrapolation is done using these two points to estimate the utilization value where number of DRC errors is equal to zero. This estimated utilization value is termed as the maximum utilization value.

Using this methodology substantial runtime improvement was achieved as illustrated in Table 3.2.

3.5.2 Model formulation

Although AEGR gives substantial improvement in runtime, it still requires running P&R for all the designs and large number of FEOL design rules (increasing with every new technology node). Also, tool noise leads to problems in optimization. To overcome these problems, we model chip area as a function of total cell area thereby skipping P&R to the maximum possible extent. Our proposed model in differential form is given in Equation (3.9). Here y is the chip area and x is the total cell area. $\frac{x}{y}$ is the utilization of the design. In the proposed model, as the utilization increases or equivalently white space decreases, change in chip area is more sensitive to any change in cell area. The final analytical equation is given in Equation (3.15).

$$\frac{dy}{dx} = k1 - k2 * (y/x) \tag{3.9}$$

substituting $y = tx$ in Eq. (3.9) we get,

$$x\frac{dt}{dx} = k1 - k2 * t - t \tag{3.10}$$

$$\int_{\frac{y0}{x0}}^{\frac{y}{x}} \frac{dt}{k1 - k2 * t - t} = \int_{x0}^{x} \frac{dx}{x} \tag{3.11}$$

$$ln\left(\frac{k1 - (k2+1) * \frac{y}{x}}{k1 - (k2+1) * \frac{y0}{x0}}\right) = -(k2+1) * ln\left(\frac{x}{x0}\right) \tag{3.12}$$

$$k1 - (k2+1) * \frac{y}{x} = \left(k1 - (k2+1) * \frac{y0}{x0}\right) * \left(\frac{x}{x0}\right)^{-(k2+1)} \tag{3.13}$$

$$y = \frac{k1}{k2+1} * x + \left(y0 - (\frac{k1}{k2+1}) * x0\right) * (\frac{x}{x0})^{-k2} \tag{3.14}$$

$$y = \frac{k1}{k2+1} * x + \left(y0 - (\frac{k1}{k2+1}) * x0\right) * (\frac{x}{x0})^{-k2} \tag{3.15}$$

There are four unknowns in the model viz. $k1$, $k2$, $x0$ and $y0$. $y0$ can be thought of as the routing limited chip area [32]. $x0$ can be thought of as any unutilized whitespace area[3] when the chip area is $y0$. $x0$ depends on the cell routability which in turn is dependent on the pin access and congestion within the cell [33]. Larger congestion implies router needs to drop more vias outside the cells to make connections with the cell instance pins, effectively decreasing any unutilized whitespace and hence decreasing $x0$.

To find $k1$ and $k2$ we apply the following boundary conditions

[3] unutilized whitespace is the chip area minus the area required by the router to make connection with the cell instance pins using M1 layer.

$$
\begin{aligned}
k1 - k2 &= 1, & (3.16)\\
k1 - k2 * \frac{y0}{x0} &= 0. & (3.17)
\end{aligned}
$$

Equation (3.16) is based on the fact that for very high utilization values, change in chip area is roughly equal to change in total cell area. This implies that as $u \to 1$, $\frac{dy}{dx} \to 1$. Hence the boundary condition follows from Equation (3.9). Similarly from the other extreme, For any total cell area less than $x0$, chip area is routing limited and is equal to $y0$. Hence, Equation (3.17) follows from Equation (3.9). Based on these boundary conditions, model coefficients and final analytical equation are given by

$$
\begin{aligned}
k1 &= \frac{y0}{y0 - x0} & (3.18)\\
k2 &= \frac{x0}{y0 - x0} & (3.19)
\end{aligned}
$$

$$y = x + (y0 - x0) * \big(\frac{x0}{x}\big)^{\frac{x0}{y0-x0}} \text{ for } x > x0 \qquad (3.20)$$

$$y = y0 \text{ for } x <= x0 \qquad (3.21)$$

Since $y0$ and $x0$ are design dependent parameters, we estimate them by actual P&R runs for each design under consideration. $x0$ and $y0$ needs to be estimated only once for a given back-end interconnect stack and library architecture. This gives substantial improvement in runtime making it possible to simultaneously evaluate large number of design rules.

Our experiments to validate our methodology were performed on 5 designs from [24], synthesized using Nangate Open Cell-Library [25], and FreePDK open-source pro-

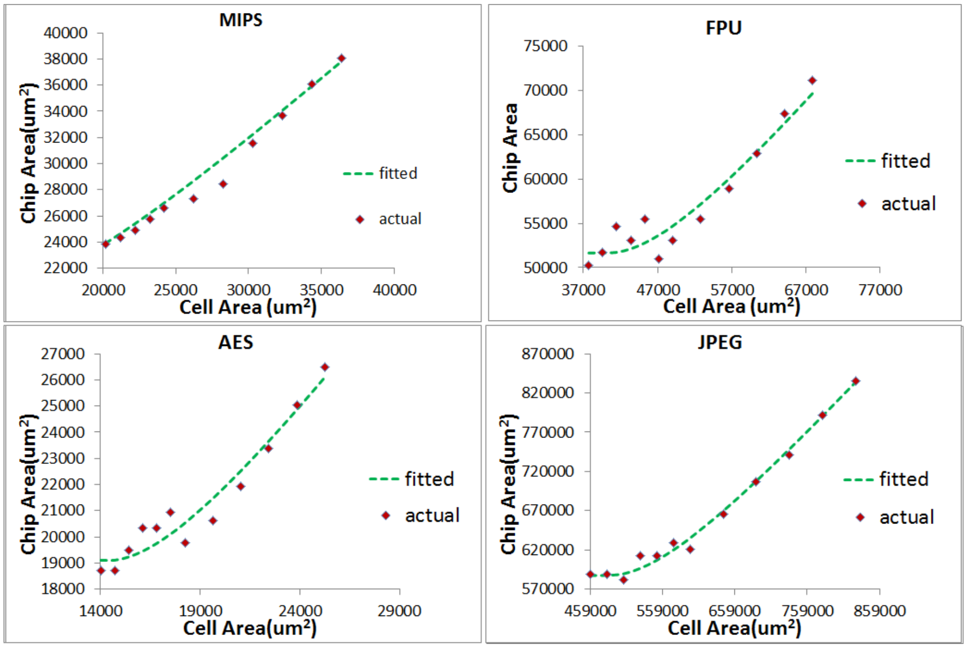


Figure 3.5: Plots showing the actual chip area and chip area estimated using the analytical model vs total cell area

cess [26]. First a data for actual P&R was created for all the designs using cadence encounter, with router objective function as "minimize congestion", and for varying number of routing layers. Based on these runs α and β (in Equation (3.8)) were estimated to be $\frac{1}{3}$. The comparison between AEGR and actual P&R for MRA estimation is given in Table 3.2. For actual P&R, maximum utilization was found using binary search algorithm.

To evaluate the cell area to chip area model, the area of various cells was increased in the LEF file to closely imitate cell area change due to FEOL design rule changes. However, the pin shapes and pin positions were not modified. Chip area was then estimated using AEGR for every instance of increase in total cell area and the proposed model was fitted on the resulting data. The plots are shown in Figure 3.5.2 and values of $x0$ and $y0$ are shown in Table 3.3

Design	**Routing Layers**	**AEGR Util.**	**P&R Util.**	**Runtime in mins (AEGR)**	**Runtime in mins (P&R)**	**Runtime Reduction**
MIPS	3	0.83	0.83	97	322	3.3x
JPEG	3	0.93	0.93	345	892	2.6x
AES	3	0.44	0.47	57	1267	22x
FPU	3	0.91	0.90	52	261.17	5x
MIPS	4	0.97	0.97	23	145	6.3x
AES	4	0.76	0.76	110	842	7.6x
NOVA	4	0.88	0.88	296	519	1.8x
AES	5	0.85	0.84	52.4	141	2.7x

Table 3.2: runtime comparison between global routing congestion based estimation and actual P&R for calculating minimum routable area

Table 3.3: Values of $x0$ and $y0$ for various designs (see Figure 3.5.2 for the plots)

Design Name	**x0**(um^2)	**y0**(um^2)
MIPS	12526	20437
FPU	40265	51649
AES	14365	19090
NOVA	95986	122050
JPEG	485700	587480

3.5.3 Functional Yield Modeling

Functional yield at the cell-level is computed similarly to [1]. It includes three yield-loss source: overlay error (i.e. misalignment between layers) coupled with lithographic line-end shortening (a.k.a. pull-back), contact-hole failure, and random particle defects. The yield at the cell-level is extended to the chip-level using the well-known negative binomial model [4]. GCPW can then be calculated as the ratio of $\frac{\text{wafer area}}{\text{chip area}} \times yield$.

3.6 Experimental Results

As examples, we study three interesting rules in Chip-DRE: (1) well-to-active spacing rule which affects number of transistor folds (hence area and delay variability) as well

[4] Yield loss in routing-layers will be addressed in future work.

Table 3.4: Chip area comparison between SPR and model based prediction. The run-time for Chip-DRE is just the cell estimation time which is about 49 minutes for a 100 cell library. The golden flow uses Chip-DRE generated libraries with commercial tools for physical design with the proposed AEGR routing area estimation method proposed in this paper.

Design	Well-to-active spacing	Chip area SPR (in um^2)	Chip Area model (in um^2)	Error in %	GCPW model based	SPR Run-time in mins
MIPS	140	29645	30364	2.42	667	118
	185	29645	28372	-4.29	713	356
	200	33516	31914	-4.7	633	240
	210	32881	32196	-2.08	627	207

as threshold voltage and mobility of transistors (hence delay); (2) local-interconnect to gate spacing rule which affects capacitances as well as area; and (3) fin-pitch rule for a candidate FinFET technology. In all three cases, we observe that simple cell-based estimates (as is the state of the art) to assess rule quality can be misleading highlighting the importance of the Chip-DRE framework.

3.6.1 Well-to-active spacing rule

Chip-DRE was used to perform a study of the well-to-active spacing rule, which impacts cell-delay without affecting the cell-area. The rule values that were chosen are 140nm, 185nm, 200nm and 210nm with 140nm as the baseline value. SPR data was generated for MIPS design using the DRE generated LEF and LIB files for each spacing rule with timing optimization done at both placement and post-routing stages while keeping the congestion effort “high”. The clock period was chosen such that minimum positive slack was achieved for the baseline case. The maximum utilization at which the number of DR violations is equal to zero and timing slack is positive is used to compute the chip area. Chip area comparison between actual SPR and using proposed model based flow is given in Table 3.4. The table also shows the GCPW metric for the design rule. It takes around 49 minutes to generate standard cells and compute delays for a 100 cell library on a single CPU. We do not use commercial library characteri-

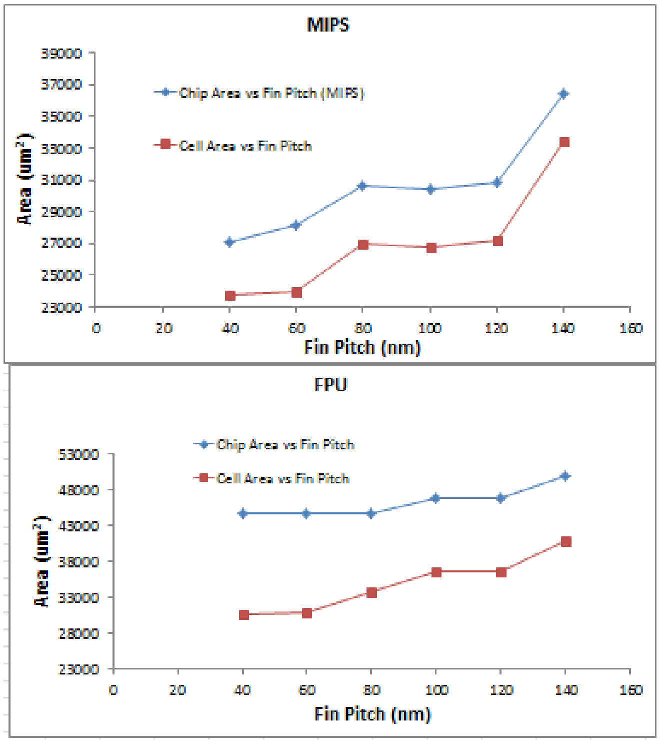


Figure 3.6: Plots for chip area and cell area vs fin pitch

zation tools for golden flow as the run-time is impractically high (few CPU days for one set of rules for a 100 cell library). As is visible from the results from Table 3.4, Chip-DRE predictions are in strong agreement with the full physical implementation based flow and match the trends well. This shows the validity of the Chip-DRE models. Interestingly, the dependence of GCPW as well as chip area on the rule value is not monotone. This is despite the fact that library cell area montonically increases as the rule value increases since increasing well-to-active spacing improves delay that results in less buffering and resizing at the chip-level.

3.6.2 FinFET Fin-Pitch study

In this experiment, the impact of fin pitch on chip-area was studied. The impact of fin pitch on delay was ignored in this experiment since its impact on parasitic capacitances was not modeled in this work[5]. Fin spacing was varied from 60nm to 120nm in steps of 20nm and for each value standard cell layouts were generated. Based on the standard cell usage of MIPS and FPU design, total cell area was computed. The cell area was then plugged into "cell to chip area model" and chip area was computed for the two designs. Figure 3.6 shows the chip area and cell area variations as the fin pitch is varied. Interesting results can be deduced from this experiment. For. eg., in case of FPU, increasing the fin pitch from 60 to 80nm increases cell area by almost 9% while there is negligible increase in chip area. GCPW trends are similar to chip area trends in this case.

3.6.3 LI to gate spacing

Local interconnect is commonly used in modern technologies to relieve congestion on local metal layers. One of the primary purposes is to make the power and ground rail connections from corresponding active areas in the devices. These connections replace contacts and metal. Unfortunately, these long contacts also increase capacitive coupling between gate and the local interconnect resulting in increased C_{gs}. To complicate matters further, increased spacing between gate and local interconnect can cause increase in the active area resulting in increased diffusion capacitance as well. We model both these effects in Chip-DRE for the planar process and explore this spacing rule. Figure 3.7 shows the effect of changing the LI to gate spacing on the chip area. In this case, the cell area increase due to rule value increase dominates the potential delay improvement coming from reduced gate to LI coupling capacitance. GCPW trends are similar, so we do not report them. Note the difference in rule behavior compared to the well-to-active spacing rule which has a much stronger delay impact.

[5] This will be part of future work.

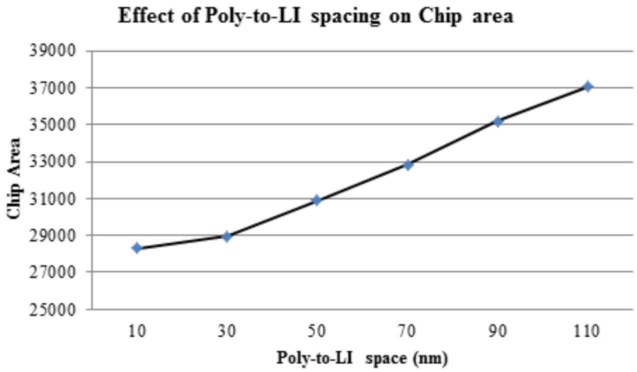


Figure 3.7: Poly to LI design rule evaluation and effect on chip area

3.7 Conclusions

We presented Chip-DRE, the first framework for *fast*, *early* and *systematic* collective evaluation of design rules, layout styles, and library architectures at the chip-scale. The framework makes rule definition and optimization easier, efficient, and much more systematic. Rather than exploring the entire search space of design rules manually or with conventional compute-expensive methods, the framework can be used to quickly eliminate poor rule choices. By using fast layout-estimation methods coupled with semi-empirical models for cell-area/delay impact and trade-offs at the chip-level, our framework can evaluate big decisions *before* exact process and design technologies are known. In this paper, we illustrated the potential applications of our framework for the collective evaluation rules. We use the Chip-DRE framework to perform evaluation studies of debatable rules for state-of-the-art technologies, including FinFETs and local-interconnects, at the chip-scale. For instance *a study of well-to-active spacing rule reveals a non-monotone dependence of rule value to chip area* (although the cell area relationship is monotone) due to delay changes coming from well-proximity effect.

CHAPTER 4

Conclusions

To conclude this work key contributions are summarized and some directions for future work are discussed.

4.1 Key contributions

This thesis improves upon some major limitations in UCLA_DRE, a tool developed by NanoCad lab at UCLA for fast and systematic evaluation of design rules at the cell level in terms of area, variability and manufacturability. The key contributions of this work are mentioned below:

- A framework was developed to evaluate interaction between design rules and overlay control. In this work, a comprehensive yield model that takes into account various overlay characteristics such as the overlay model used, 3σ residue for unmodeled components and the model used for overlay correction was developed. Using this framework, users can also generate trade-off curves between yield and area/designer effort and systematically make decisions on the design rule value to be used. Information regarding design rule usage and design area is computed using UCLA_DRE. Interesting results can be deduced from the results such as increasing the minimum mask-overlap length by 1nm would allow the use of a third-order wafer/sixth-order field-level overlay model instead of a sixth-order wafer/sixth-order field-level model with negligible impact on designer effort on resolving all spacing violations in the design. Similarly for poly line-end extension rule, increasing the rule value from 8nm to 9nm would allow the use of third order wafer-level and third order field-level model instead of the sixth order wafer-level and sixth order field-level model.

- We offer Chip-DRE, the first framework for *fast*, *early* and *systematic* collective evaluation of design rules, layout styles, and library architectures at the chip-scale. We evaluate the rule impact on delay and report the evaluation in terms of "good chips per wafer" unifying area, performance, variability and functional yield metrics. This comprehensive evaluation allows studying interesting trade-offs that occur at the chip level like the one between variability, performance and area.We perform evaluation studies of major design rules at advanced nodes (some FinFET-specific) including: gate to local-interconnect spacing, gate-to-well edge spacing and fin pitch. Some of the interesting results from this work are: non-monotone relationship of GCPW with well to active spacing rule, negligible change in FPU chip area although cell area changes by 9% and negligible reduction in input pin capacitance by increasing LI to poly spacing due to increase in diffusion capacitance although LI to gate capacitance decreases.

4.2 Future Work

In future work, yield and design-impact analysis will be extended to a chip-level analysis across all layers in the design and explore other overlay-related rules especially rules related to cut-masks. For chipDRE, we want to include the impact of fin pitch on cell delay and its subsequent effect on chip area. Further, we want to explore the nature of $x0$ and $y0$, coefficients in the cell area to chip area model, as the library architecture and/or interconnect stack parameters (such as metal pitch, number of metal layers) are changed.

References

[1] R. S. Ghaida and P. Gupta. DRE: A framework for early co-evaluation of design rules, technology choices, and layout methodologies. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 31(9):1379–1392, Sept 2012.

[2] Yixin Lu. Optimization of Process Corrections. Master's thesis, Technische Universiteit Eindhoven, Eindhoven, 2008.

[3] B. Eichelberger, K. Huang, K. O'Brien, D. Tien, F. Tsai, A. Minvielle, L. Singh, and J. Schefske. 32nm overlay improvement capabilities. In *Proc. SPIE*, volume 6924, page 69244C, 2008.

[4] C. A. Mack. How to characterize overlay errors. In *Yield Management Solutions*, pages 46–47, 2006.

[5] W. H. Arnold. Toward 3nm overlay and critical dimension uniformity: an integrated error budget for double patterning lithography. In *Proc. SPIE*, volume 6924, page 692404, 2008.

[6] M. Dusa et al. Pitch doubling through dual-patterning lithography challenges in integration and litho budgets. In *Proc. SPIE*, volume 6520, page 65200G, 2007.

[7] W. Arnold, M. Dusa, and J. Finders. Manufacturing challenges in double patterning lithography. In *IEEE Intl. Symp. Semiconductor Manufacturing*, pages 283–286, Sept 2006.

[8] D. Laidler, P. Leray, K. D'havé, and S. Cheng. Sources of overlay error in double patterning integration schemes. In *Proc. SPIE*, volume 6922, page 69221E, 2008.

[9] C. Chien and K. Chang. Modeling overlay errors and sampling strategies to improve yield. *Journal of the Chinese Institute of Industrial Engineers*, 18(3):95–103, 2001.

[10] R. Wang, CY Chiang, W. Hsu, R. Yang, J. Chiu T. Shih, J. Chen, and W. Lin. Overlay improvement by ASML HOWA 5th alignment strategy. In *Proc. SPIE*, volume 7520, page 752023, 2009.

[11] S. Wakamoto, Y. Ishii, K. Yasukawa, A. Sukegawa, S. Maejima, A. Kato, J. C. Robinson, B. J. Eichelberger, P. Izikson, and M. Adel. Improved overlay control through automated high-order compensation. In *Proc. SPIE*, volume 6518, page 65180J, 2007.

[12] Harry J. Levinson. *Principles of Lithography*. SPIE Press, Bellingham, WA, 2010.

[13] D. Choi, C. Lee, C. Bang, D. Cho, M. Gil, P. Izikson, S. Yoon, and D. Lee. Optimization of high order control including overlay, alignment, and sampling. In *Proc. SPIE*, volume 6922, page 69220P, 2008.

[14] B. Eichelberger et al. 32nm overlay improvement capabilities. In *Proc. SPIE*, volume 6924, page 69244C, 2008.

[15] Ayako Sukegawa, Shinji Wakamoto, Shinichi Nakajima, Masaharu Kawakubo, and Nobutaka Magome. Overlay improvement by using new framework of grid compensation for matching. 6152:61523A–61523A–8, 2006.

[16] R. S. Ghaida and P. Gupta. Within-layer overlay impact for design in metal double patterning. *IEEE Transactions on Semiconductor Manufacturing*, 23(3):381–390, Feb 2010.

[17] J. Yang and D. Z. Pan. Overlay aware interconnect and timing variation modeling for double patterning technology. In *Intl. Conf. on Computer-Aided Design*, pages 488–493, November 2008.

[18] E. Y. Chin and A. R. Neureuther. Variability aware interconnect timing models for double patterning. In *Proc. SPIE*, volume 7275, page 727513, 2009.

[19] K. Jeong, A. B. Kahng, and R. O. Topaloglu. Is overlay error more important than interconnect variations in double patterning? In *System Level Interconnect Prediction*, pages 3–10, June 2009.

[20] P. Gupta, A. B. Kahng, Y. Kim, S. Shah, and D. Sylvester. Line end shortening is not always a failure. In *ACM/IEEE Design Automation Conference*, pages 270–271, June 2007.

[21] Ayako Sukegawa, Shinji Wakamoto, Shinichi Nakajima, Masaharu Kawakubo, and Nobutaka Magome. Overlay improvement by using new framework of grid compensation for matching. pages 61523A–61523A–8, 2006.

[22] R. S. Ghaida and P. Gupta. A framework for early and systematic evaluation of design rules. In *Intl. Conf. on Computer-Aided Design*, pages 615–622, Nov 2009.

[23] R. S. Ghaida, K. B. Agarwal, S. R. Nassif, X. Yuan, L. W. Liebmann, and P. Gupta. A framework for double patterning-enabled design. In *Intl. Conf. on Computer-Aided Design*, pages 14–20, Nov. 2011.

[24] http://www.opencores.org/.

[25] Nangate open cell library v1.3. 2009. http://www.si2.org/openeda.si2.org/projects/nangatelib.

[26] FreePDK. http://www.eda.ncsu.edu/wiki/FreePDK.

[27] W. C. Elmore. The transient response of damped linear networks with particular regard to wideband amplifiers. *Journal of Applied Physics*, 19(1):55–63, 1948.

[28] Jan M. Rabaey, Anantha Chandrakasan, and Borivoje Nikolic. *Digital integrated circuits- A design perspective*. Prentice Hall, 2nd edition, 2004.

[29] Kwangok Jeong, A.B. Kahng, and K. Samadi. Impact of Guardband Reduction On Design Outcomes: A Quantitative Approach. *IEEE Transactions on Semiconductor Manufacturing*, 22(4):552 –565, November 2009.

[30] A.B. Kahng and S. Mantik. Measurement of inherent noise in EDA tools. In *Quality Electronic Design, 2002. Proceedings. International Symposium on*, pages 206–211, 2002.

[31] H.B.Bakoglu. *Circuits, Interconnections and Packaging for VLSI*. 1990.

[32] R. Venkatesan, J.A. Davis, K.A. Bowman, and J.D. Meindl. Optimal n-tier multilevel interconnect architectures for gigascale integration (gsi). *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 9:899–912, Dec. 2001.

[33] T. Taghavi, C. Alpert, A. Huber, Zhuo Li, Gi-Joon Nam, and S. Ramji. New placement prediction and mitigation techniques for local routing congestion. In *Computer-Aided Design (ICCAD), 2010 IEEE/ACM International Conference on*, pages 621–624, 2010.