University of California
Los Angeles

Design-Aware Mask Manufacturing

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

by

Abde Ali Kagalwalla

2011

© Copyright by
Abde Ali Kagalwalla
2011

The thesis of Abde Ali Kagalwalla is approved.

Lei He

Dejan Markovic

Puneet Gupta, Committee Chair

University of California, Los Angeles

2011

To mom and dad.

Table of Contents

1 Introduction **1**

1.1 Mask Inspection Primer 3

1.2 EUV Lithography 7

1.3 Thesis Outline 8

2 Assigning Criticality to Layout Features **10**

2.1 Non-functional Feature Finding Algorithm 11

2.2 Criticality Assignment Methodology 16

2.3 Experimental Results 21

2.4 Chapter Summary 23

3 Design-Aware Mask Inspection **25**

3.1 Reticle Partitioning 26

3.2 Experimental Results 30

3.3 Chapter Summary 32

4 Defect-Aware Floorplanning for EUV Masks **36**

4.1 Modeling CD impact of Buried Defects 37

4.2 Defect-Aware Reticle Floorplanning 41

4.3 Experimental Results 45

4.4 Chapter Summary 53

5 Conclusions **54**

5.1 Summary of Contributions 54

5.2 Future Work 55

References **57**

List of Figures

1.1 Mask cost increase with technology [itr09] 2

1.2 Key steps of reticle inspection along with our contributions in circular blocks 4

1.3 Various categories of defects reported by inspection tool 5

1.4 An EUV mask with buried defects [CN08a]. 8

2.1 Shape simplification for a distored T-shape 14

2.2 Illustration of various steps of redundancy-finding algorithm . . . 14

2.3 Impact of various defect types on polysilicon layer features 17

2.4 Impact of different defect types on a via layer feature 21

3.1 Incremental cost evaluation for partitioning 30

3.2 First-pass yield with design-aware inspection. The number of defects sprinkled are $\mu + \sigma$ and $\mu + 3\sigma$ which is derived from the reticle inspection statistics we have available from a commercial mask shop. Note that the yield is zero for the conventional design-unaware inspection strategy in all these cases. 34

4.1 A Gaussian defect with height H and full width at half maximum FWHM 38

4.2 Illustration of some assumptions for modeling of CD impact of buried defect 39

4.3 A defect and absorber with r as distance between center of defect and closest absorber edge 40

4.4 Various degrees of flexibility for floorplanner 45
4.5 Mask yield for different defect dimensions 51
4.6 Improvement in post-floorplanning mask yield with free space . . 52

List of Tables

2.1 Impact of different defect types on polysilicon layer 17

2.2 Impact of different defect types on a metal layer 19

2.3 Impact of different defect types on a via layer 20

2.4 Shape simplification results 22

2.5 Experimental results for redundancy finding 22

2.6 Layer by layer redundant vias and dummy regions 23

3.1 Experimental results for partitioning with false and real defect count before and after design-aware inspection 33

4.1 Different die sizes considered 46

4.2 Experimental results for reticle floorplanning without design information (Die Yield (DY) and Mask Yield (MY)) 48

4.3 Experimental results for reticle floorplanning with design information 49

4.4 Mask yield for different orientations 53

Acknowledgments

This thesis would not have been possible without the constant guidance and support of my advisor, Prof. Puneet Gupta. Our frequent technical discussions have been very fascinating and educative. I would also like to thank Prof. Andrew Kahng at UCSD for introducing me to the field of physical design through a summer undergraduate internship. I would probably never have taken up graduate studies without that experience.

I would like to acknowledge the support of our industrial partners who provided me with data and feedback. Chris Progler and Steve McDonald (Photronics Inc.) were tremendously helpful in our mask inspection project. Discussions with Stan Stokowski (KLA-Tencor) and Alexander Starikov (I&I Consulting) provided me with much needed feedback for my inspection and EUV projects, respectively. Duck Hyung Hur and Chul Hong Park (Samsung) provided us with relevant data for EUV project.

My colleagues at NanoCAD lab over the last two years: Tuck Boon Chan, John Lee, Rani Ghaida, Aashish Pant, Santiago Mok, Liangzhen Lai, Parag Kulkarni, Charwak Apte, Tanaya Sahu and Lerong Cheng have all been very helpful and fun to be around. My work would not have been the same without their daily contributions.

Lastly, I would like thank my parents and my brother for their unconditional support and encouragement.

Abstract of the Thesis

Design-Aware Mask Manufacturing

by

Abde Ali Kagalwalla

Master of Science in Electrical Engineering

University of California, Los Angeles, 2011

Professor Puneet Gupta, Chair

The cost per chip benefit of semiconductor technology scaling is rapidly eroding due to increasing manufacturing costs. Photomasks are a significant contributor to this problem. The primary goal of our research has been to exploit design information, which can inform mask makers of the criticality of different patterns on the mask, to help reduce mask cost. In this thesis, we first focus on developing a methodology to evaluate criticality of different features in a layout. Then we explore two potential applications of this information towards reducing mask cost: mask inspection and mask floorplanning for EUV masks.

We propose a method to access the criticality of different features of a given layout. For metal and via layers, it essentially boils down to finding the redundant vias and dummy fill. To do this, we develop a scan-line based algorithm to construct a connectivity graph which helps identify these structures without using any design information. Using our method, we can evaluate post-OPC design layouts with more than 40k gates in just 80 minutes with almost 100% accuracy. Using this information along with timing information and design rules, we assign a minimum-size defect to each reticle feature that could cause the design to fail.

The criticality of various design features on the reticle is then used to partition

the reticle such that each partition is inspected at a different pixel size and sensitivity so that the false defect + nuisance defect count is reduced without missing any critical defect. Up to a $4X$ improvement in false defect + nuisance defect count is observed with our technique, resulting in up to 55% improvement in first-pass yield. These improvements lead to substantial reduction in time/effort needed for defect review.

Fabricating defect-free mask blanks remains a major "show-stopper" for adoption of EUV lithography. One promising approach to alleviate this problem is reticle floorplanning with the goal of minimizing the design impact of buried defects. We propose a simulated annealing based gridded floorplanner for single project reticles that minimizes the design impact of buried defects. Our results show a substantial improvement in mask yield with this approach. For a mask with 40 defects, our approach can improve mask yield from 53% to 94%. Criticality information of reticle features can be exploited for more accurate yield computation. This helps further improve mask yield, up to 99% for a 40-defect mask. These improvements are achieved with a limited area overhead of 0.03% on the exposure field. Defect-aware floorplanning also reduces sensitivity of mask yield to defect dimensions.

CHAPTER 1

Introduction

Over the last decade, semiconductor technology scaling continues to keep pace with Moore's law. The biggest driver for scaling has been cost reduction due to more chips per wafer. But this cost benefit is rapidly eroding due to huge increase in manufacturing cost with each new technology generation. This trend in illustrated in Figure 1.

As can be seen from Figure 1, a substantial component of the manufacturing cost is due to photomasks. The use of aggressive resolution enchancement techniques such as optical proximity correction (OPC) and phase-shiting masks (PSM) have led to extremely complex mask features which are hard to manufacture [GKS03]. For 45nm commercial digital designs, a complete mask set can easily cost more than a million dollars. Current projections from ITRS suggest that this situation is likely to get worse with newer lithography patterning technologies like double patterning or extreme ultraviolet (EUV) lithography.

One major cause of such a high manufacturing cost is a lack of design-manufacturing interaction. Most digital design layouts have many non-functional or non-critical features. Knowledge about the criticality of various shapes/features in a circuit layout can be exploited to reduce manufacturing cost by adapting the "manufacturing effort" based on the criticality of different design features.

There has been considerable interest in reducing manufacturing cost by using

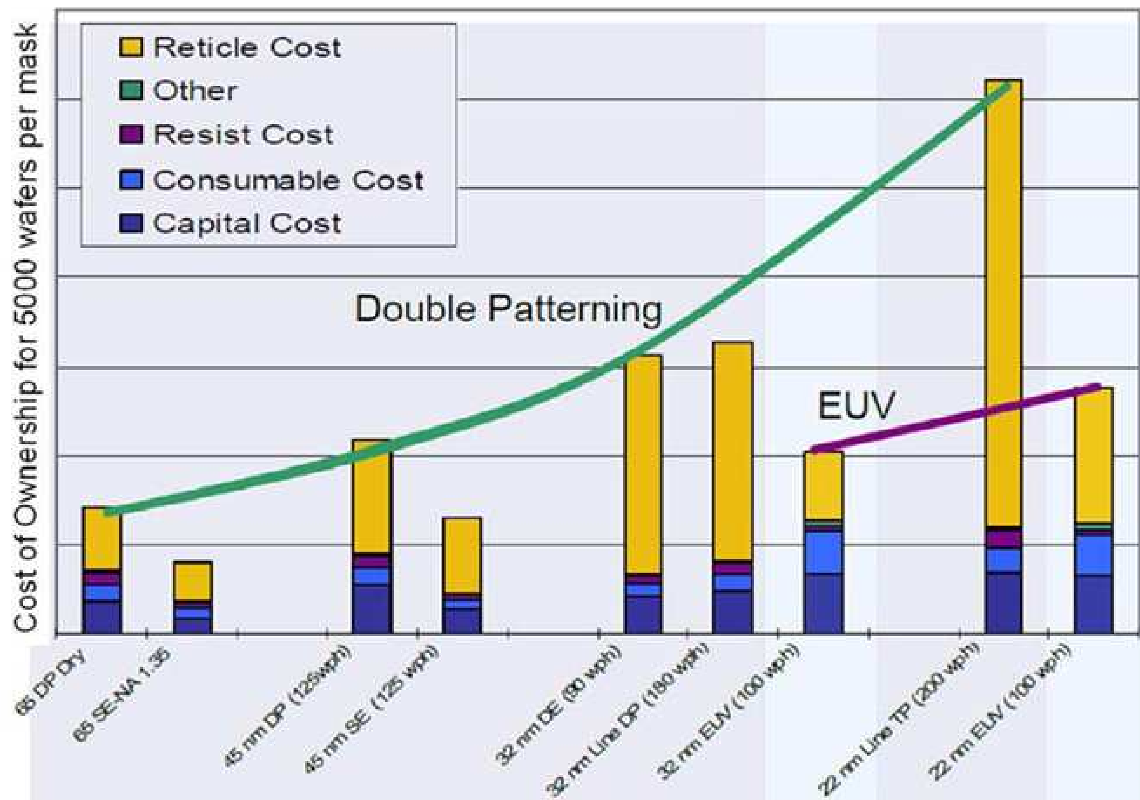


Figure 1.1: Mask cost increase with technology [itr09]

design intent. For instance, Banerjee et al. [BEL08] use estimates of on/off current of transistors, based on simulated resist contours, to reduce OPC runtime and mask complexity. Zhang et al. [ZA07] modeled the impact of corner rounding in printed transistors on saturation current and integrated their model into a OPC framework. Similarly, [THT08, KNO07] used device performance estimates to tune the aggressiveness of optical correction achieving up to 93% reduction in mask complexity. Gupta et al. [GKS05] reduce mask write cost by adapting mask fracturing based on timing slack. Chan et al. [CKG11] use estimates of design metrics like delay and power to improve the evaluation of process window. These approaches indicate that considerable benefit can be derived by using design intent to reduce the inherent pessimism of various mask manufacturing steps. In this thesis, we focus on two manufacturing steps that can benefit from design-awareness; mask inspection and reticle floorplanning for EUV masks.

This chapter is organized as follows. First we give an overview of the current mask inspection methodology in Section 1.1. This is followed by a brief introduction to EUV lithography and some of the challenges facing it in Section 1.2. Section 1.3 then gives an outline of the overall thesis.

1.1 Mask Inspection Primer

A comprehensive inspection of reticles must be done by the mask shops before sending it to the foundry. Mask inspection has become a major bottleneck in the manufacturing flow taking up as much as 30% of the total manufacturing time [HK08]. The basic steps of inspection are shown in Figure 1.2.

Initially the reticle is passed through an inspection tool (e.g. KLA-Tencor's TeraScan [DMI08] or NEC's LM series [MBM08]) which takes an image of a die

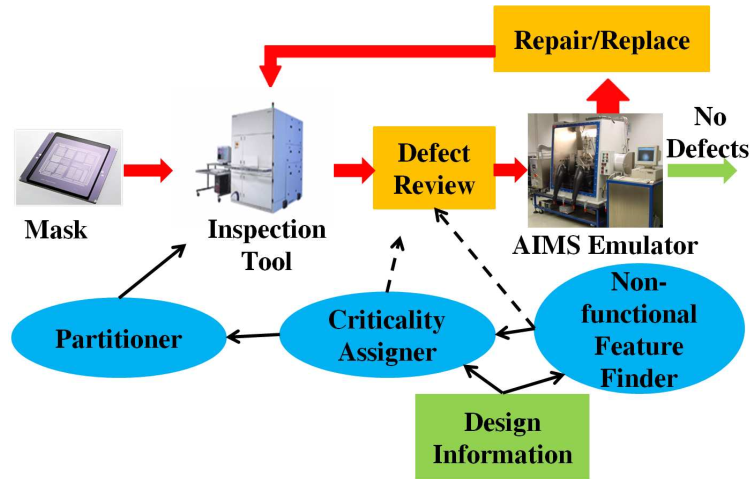


Figure 1.2: Key steps of reticle inspection along with our contributions in circular blocks

and compares it to a reference database (die-database mode) or another die (die-die mode). The difference between the two images is found and if the intensity of the difference exceeds a predefined threshold, the difference pattern is labeled a defect. The inverse of this threshold is referred to as sensitivity. These tools can have a pixel size as low as $55nm$ and can detect critical dimention (CD) defects as small as $20nm$ (equivalent to roughly $5nm$ on wafer) on the mask at maximum sensitivity (minimum threshold) [DMI08].

Inspection tools can generate a very large number of defects (100+) most of which do not impact the final design. Defects can be classified as shown in Figure 1.3. A *false defect* is an incorrect detection reported by the inspection tool due to vibration, misalignment, optical distortion, error in database rendering (die-database mode), etc. *Real defects* are caused either due to misalignment or vibration of the mask writer (CD defects) or contamination of the mask (contamination defects). Inspection tools typically have different algorithms to detect these two categories of defects and hence have different sensitivities for these de-

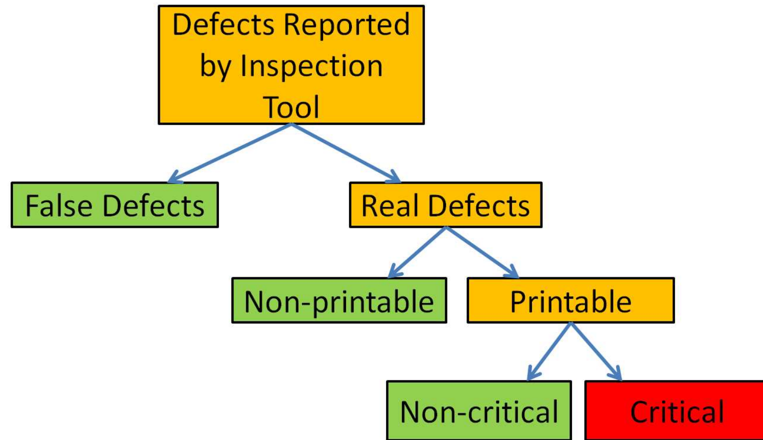


Figure 1.3: Various categories of defects reported by inspection tool

fects. Only a small fraction of real defects actually print on the wafer. Among the printable defects, some lie on non-critical regions of the design such as dummy fill or redundant vias. Hence, only a small fraction of the defects reported by the inspection tool really matter. All the non-printable and non-critical defects are also called *nuisance defects*.

Reducing the number of false defects + nuisance defects reported by the inspection tool is essential to reduce inspection cost. Reducing nuisance defects is particularly important to mask shops as it impacts *first-pass yield*, which is the fraction of total masks manufactured that can be shipped without repair or detailed review.

The next step in mask inspection is defect review, where each defect reported by the inspection tool is checked to find out if it really matters. False, non-printable and non-critical defects are filtered out during this step. Images of defects reported by the inspection tool are analyzed using software tools [PLC03, KAC01] or manually. Often defect images need to be recaptured at a better resolution. For this, the inspection tool could be reused (Online Review) or an

e-beam is employed [KLZ08].

After pruning out a significant fraction of false/non-printable/non-critical defects, the mask is passed through an aerial imaging tool such as the Carl Zeiss AIMS system [DZB06]. AIMS is essentially a hardware emulator of the wafer stepper that operates at the same optical settings as the stepper and gives a very accurate estimate of the printability of defects. Although extremely accurate, AIMS is slow and cumbersome. Hence, minimizing the number of defects that have to pass through AIMS is important in order to ensure a reasonable turnaround time. Defects which are found to be printable by AIMS are then either repaired, or if they are unrepairable, the reticle must be replaced. The repaired or replaced reticle must again go through this inspection cycle. Because of the manual steps and the use of AIMS, defect review is typically the slowest part of reticle inspection.

The increasing complexity of reticles has also increased the burden of reticle inspection tools. In fact, mask inspection is more challenging than mask writing itself [HK08]. High resolution reticle inspection tools are required to detect every potential printable defect in order to prevent yield loss. But this also produces a large number of "nuisance" defects, i.e. defects that do not affect yield. As a result post-inspection review of defects has become very time consuming. The overall inspection flow has a considerable impact on mask cost and turnaround time (TAT). Keeping mask cost in control is extremely critical, especially for low volume SoCs. The problem is likely to get worse for future patterning technologies like multi-layer EUV lithography and nanoimprint templates. Hence there is a strong need to improve the inspection flow.

1.2 EUV Lithography

Extreme ultraviolet (EUV) lithography is considered one of the most promising next generation lithography solutions to replace the current deep ultraviolet (DUV) lithography. It proposes moving from current 193nm lithography to 13.5nm. At 13.5nm wavelength most materials absorb light, but it is possible to have reflective optics using Bragg reflectors. EUV technology still faces several challenges before it can actually be used for volume production. In addition to source and resist, fabricating defect-free mask blanks still remains one of the major challenges for EUV [Lev09].

A key problem associated with the fabrication of EUV mask blanks, which are essentially multilayer reflecting structures, is buried defects. A sample EUV mask with buried defects is shown in Figure 1.4. Buried defects are caused due to pits on the substrate surface, or particles that get introduced either on the substrate surface, or during multi-layer deposition. Around 75% of the defects are caused due to substrate defects. Current technology has enabled mask makers to reduce the density of buried defects down to 0.005 defects/cm^2 for defects larger than 53nm [Ras09]. But the detection accuracy of most inspection tools used today is questionable and actual defect densities are expected to be much higher. Although these defects can be partially repaired using e-beam, there is a considerable risk of damaging the multi-layer structure [DFN02]. Because of these issues, it might not be feasible to produce defect-free EUV mask blanks at a reasonable cost. This necessitates developing methods to have functional masks using blanks which are not defect free.

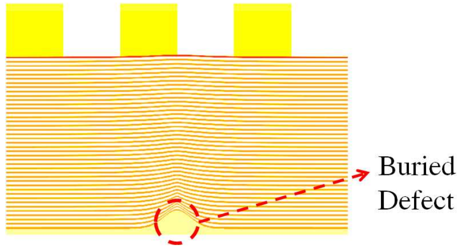


Figure 1.4: An EUV mask with buried defects [CN08a].

1.3 Thesis Outline

The key contributions of this thesis are as follows:

- We develop a methodology to assign criticality to each feature of a design layout. We propose an algorithm to identify non-functional features in a post-OPC layout and use this information, along with timing slack, to assign criticality.
- We propose a design-aware mask inspection flow that uses the assigned criticality of different features to reduce manufacturing cost.
- We propose a reticle floorplanning algorithm that can help alleviate the problem of buried defects in EUV mask blanks which improves mask yield. Using criticality assignment can help improve yield even further.

This thesis is organized as follows. Chapter 2 develops a methodology to assign criticality to each feature of a layout. To achieve this, first we develop an algorithm to locate non-functional features of a layout. This information, along with timing slack, is used to assign criticality to design features. Chapter 3 then develops a partitioning algorithm that utilizes this criticality information to improve the mask inspection flow. Chapter 4 proposes a reticle floorplanning

approach to deal with buried defects in EUV mask blanks. Criticality assignment methodology is exploited to further improve mask yield. Finally, Chapter 5 concludes the thesis.

CHAPTER 2

Assigning Criticality to Layout Features

In order to reduce manufacturing cost by using design awareness, we first try to analyze the importance or criticality of different shapes for any given design layout. This criticality corresponds to the largest defect size that a particular shape can tolerate without causing the final chip to fail. For back end layers like metal and via which do not have significant impact on delay, we identify non-critical features like redundant vias and dummy fill. For front end physical layers like polysilicon or active this criticality assignment can be done by using the available timing slack. The work presented in this chapter first appeared in [KGP10].

In this chapter, we first propose an algorithm to identify the non-critical features of back end layers in a post-OPC layout which is discussed in Section 2.1. Although we focus only on redundant vias and dummy fill in this chapter, other non-functional features such as spare cells, non-tree routes and assist features can also be found using our graph based approach. We assume that the layout has only rectilinear shapes, and that floating dummy fill in different metal layers are not connected through vias. This is consistent with most commercial fill synthesis tools[1].

We then propose a methodology to assign criticality to different features of a design layout in Section 2.2. For polysilicon layer, we use the timing slack of

[1] Future work can extend this to grounded fill and via fill as well.

various critical paths to assign a minimum-size defect for each polysilicon shape corresponding to a transistor pair (PMOS+NMOS) on the critical path. The location of redundant vias is used to assign the minimum-size defect for via layer and the location of dummy fill for metal layer.

To summarize, the two problems we attempt to answer in this section are:

1. **Non-functional Feature Finding:** *Given a post-OPC layout, identify non-functional features of the layout.*

2. **Criticality Assignment:** *Given the timing of critical paths and non-functional features, find the minimum-size defect at each location in the layout which can cause failure.*

2.1 Non-functional Feature Finding Algorithm

In order to find non-functional features in a design layout, we first fracture the layout features into rectangles. We then use a scan-line based algorithm to construct a neighborhood graph for these rectangles which essentially establishes electrical connectivity. We then perform some edge contraction operations to construct a reduced neighborhood graph (RNG) that can then help identify dummy fill and redundant vias. The various steps of our approach are detailed below.

2.1.1 Algorithm Steps

- *Fracturing Polygons:* The rectilinear polygons are fractured into rectangles using a simple horizontal slicing method [GG83]. The rectangles are then stored in different sets based on their layer. For example, a rectangle corresponding to a Metal 2 shape is stored in two sets, M_2V_1 and M_2V_2. A set

M_iV_j corresponds to all rectangles belonging to the same/adjacent metal or via layers and a via layer V_j connects M_j and M_{j+1}.

- *Neighborhood Graph Construction:* The new layout with fractured polygons is used to construct an undirected neighborhood graph, $G(V, E)$ in which every rectangle of the fractured layout corresponds to a vertex and an edge $(u, v) \in E$ if the two corresponding rectangles are physically in contact with each other in the layout.

  A scan-line based one-pass optimal algorithm is used to solve the rectangle intersection problem as described in [SW80]. The problem is reduced to two sub-problems, an interval query and a point query. Interval tree and range tree are two "semi-dynamic" tree data structures that are used to solve this problem. We shall refer to these two sets of trees as scan-line trees. A separate scan-line is used for each set M_iV_j but there is a single graph for the entire layout. Both these trees can perform INTERSECTSEARCH[2], INSERT and DELETE operations in $O(log(m))$, where m is the number of nodes in the tree, which depends linearly on the number of rectangles.

- *Edge Contraction:* All neighboring vertices of the neighborhood graph that correspond to rectangles of the same layer are merged. At the end of this operation each vertex has edges only to vertices belonging to an adjacent layer. Hence, a vertex corresponding to Metal 2 in RNG will have edges only to vertices of Via 1 or Via 2 and so on.

- *Graph Analysis:* Floating fill is identified by looking for isolated vertices. Cycles in the RNG correspond to redundant vias which can be identified

[2]INTERSECTSEARCH returns all rectangles stored in the scan-line tree that intersect the input rectangle and construct edges in the neighborhood graph between the input and all returned rectangles.

using depth first search (DFS). Double and even multi-cut vias can be identified by scanning the reported cycles and identifying the set of vias connected to the same pair of metal layer vertices in the RNG.

2.1.2 Runtime Optimization Techniques

- *Routing-aware Scan-line:* The routing direction of each set of rectangles, M_iV_j can be found by taking the larger of the average length and width of all rectangles in the set. If the routing direction is X (Y) we define y (x) coordinates of the rectangles as scan-line events so that the average duration for which a rectangle needs to be stored in the tree reduces, thus improving the INTERSECTSEARCH time.

- *Shape Simplification:* Before fracturing the polygons into rectangles, we perform shape approximation on the post-OPC polygons to reduce the number of rectangles created after fracturing. We create two sets of buckets for the coordinates of each polygon. Each point is included in two buckets, one in the x-direction and another in the y-direction such that x (or y) coordinate of each point in a bucket is within a certain threshold distance of others. All of the x (or y) coordinates of a bucket are then changed to the average x (or y) coordinate of the corresponding bucket. This approach reduces small deviations along a straight line as shown in Figure 2.1 and hence reduces rectangle count.

Algorithm 1 is the pseudo-code that summarizes the all the steps of the non-functional feature finding algorithm. Figure 2.2 illustrates the non-functional feature finding algorithm for a sample double via.

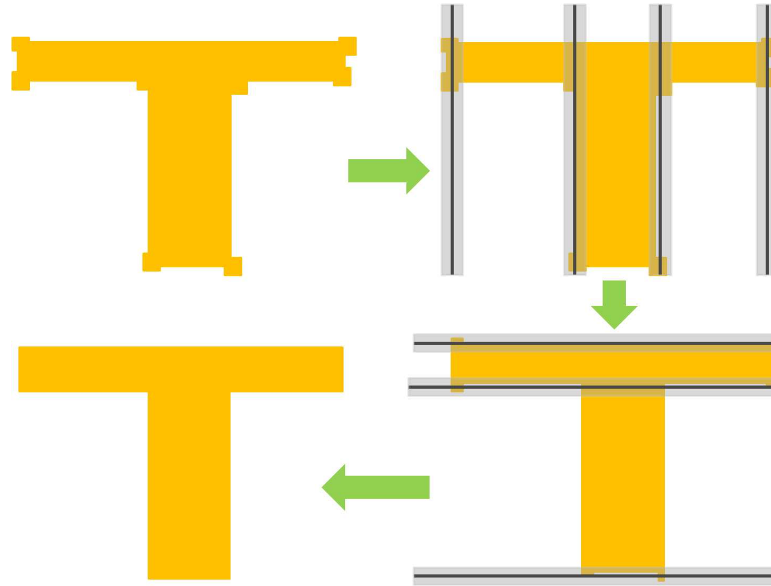


Figure 2.1: Shape simplification for a distored T-shape

We can perform the complexity analysis of this algorithm as follows. Except for sorting and scan-line based graph construction all other steps can be completed in linear time. If N is the number of rectangles in the layout, the total number of events for scan-line are $2N$. Each scan-line tree operation can be completed in $O(log(m))$, where m is number of rectangles currently stored in the tree, which can be $O(N)$ at worst. Hence the algorithm runtime is $O(Nlog(N))$ since sorting also has a worst case complexity of $O(Nlog(N))$.

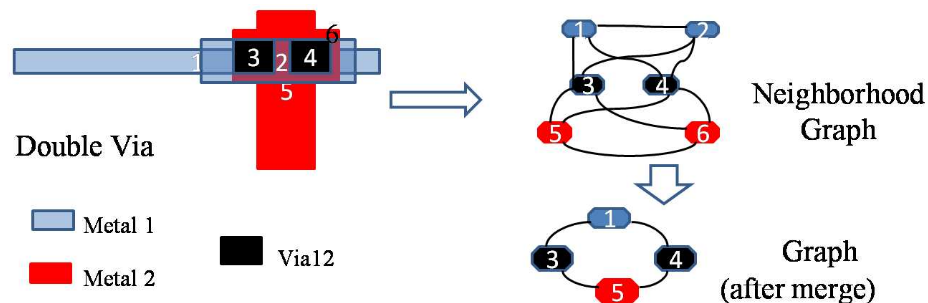


Figure 2.2: Illustration of various steps of redundancy-finding algorithm

Algorithm 1 Non-functional feature finding

Require: Shapes of all metal and via layers, S.

```
1: for all Shape s ∈ S do
2:    SHAPE-SIMPLIFICATION(s)
3:    Set of rectangles, B_s = FRACTURE(s)
4:    Store B_s in set M_iV_j corresponding to shape layer
5: end for
   //EVENT DEFINITION
6: Find routing direction R of each rectangle set, M_iV_j
7: if Routing direction R is X(Y) then
8:    Store bottom(left) and top(right) of each rectangle in set as separate events
      in E_ij.
9: end if
   //SCAN-LINE
10: for all Events e ∈ E_ij for each set E_ij do
11:    if e is bottom(left) then
12:       INTERSECTSEARCH(Scan-line Tree, e.rect)
13:       INSERT(Scan-line Tree, e.rect)
14:    else
15:       DELETE(Scan-line Tree, e.rect)
16:    end if
17: end for
   //EDGE CONTRACTION
18: Edge Contract G(V, E) to obtain RNG G(V', E')
   //GRAPH ANALYSIS
19: Mark all isolated vertices as dummy fill
20: Find cycles in G(V', E') using DFS to detect redundant vias
```

2.2 Criticality Assignment Methodology

On the basis of geometry, reticle defects are classified as pindots, pinholes, intrusions and extrusions. Intrusions and extrusions are considered CD defects. Pindots and pinholes are usually classified as contamination defects. Apart from the size, type and location of the defect, the CD impact on the wafer also depends on the type of reticle (bright-field or dark-field), type of resist (positive or negative) and mask error enhancement factor (MEEF) at the defect location. In this section, we will develop a method for estimating the CD impact of reticle defects for polysilicon, metal and via layers only. Phase defects are not considered since defect data from a commercial mask shop suggest they are rare. We use a square approximation for defects in our analysis (as do most critical area analysis methods).

In this section, we denote the size of a square defect by a and the minimum detectable defect size by a_{min}. W_{min} and S_{min} are the width and spacing design rules (DR) of the corresponding layer, respectively[3]. Df_{min}^{CD} and Df_{min}^{Con} are the minimum-size CD and contamination defects that matter for design functionality.

2.2.1 Polysilicon Layer

Poly layer printing typically uses bright-field masks with positive photoresist. Their impact is illustrated in Figure 2.3 and described in Table 2.1.

Our evaluation of minimum defect size must take the timing criticality of different cells into account apart from the possibliliy of open or short. For each transistor, we find the timing slack of the corresponding cell from the design

[3]For simplicity and pessimism we use minimum DR rules instead of using exact design values.

Table 2.1: Impact of different defect types on polysilicon layer

Type	Gate Length	Design Impact
Intrusion	Decrease	Open/Delay Decrease
Extrusion	Increase	Short/Delay Increase
Pinhole	Decrease	Open
Pindot	No change	None

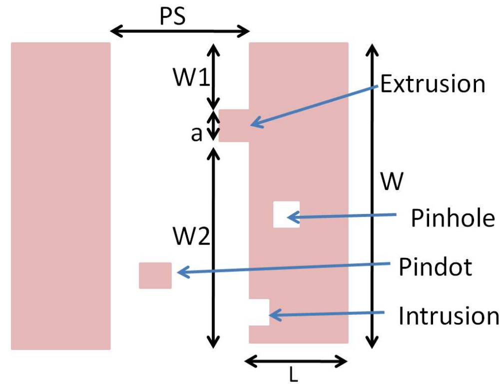


Figure 2.3: Impact of various defect types on polysilicon layer features

timing report. Assuming that at most K defects lie on a critical path[4], we can evaluate the minimum-size defect that changes the delay of the transistor and hence the path delay by less than T_{slack}/K. The formula for estimating the maximum gate length deviation, $\delta L_{critical}$, is derived in Equations (2.1)-(2.3) using a basic transistor model and first-order approximations where the various parameters are shown in Figure 2.3.

$$\frac{W}{L}_{new} = \frac{W1}{L} + \frac{W2}{L} + \frac{a}{L-a} \tag{2.1}$$

$$\frac{\triangle Delay}{Delay_{nom}} = -\frac{\triangle \frac{W}{L}}{\frac{W}{L}} = \frac{a^2}{WL} \tag{2.2}$$

$$\begin{aligned} \delta L_{critical} = a_{critical} &= \sqrt{\frac{(T_{slack} - \alpha_{cycle})/K}{Delay_{nom}} WL} \\ &= a_{min} \quad if \quad T_{slack} < \alpha_{cycle} \end{aligned} \tag{2.3}$$

Here α_{cycle} is taken as 1% of the design cycle time. This additional delay margin for each transistor is chosen to guardband against later variations. Hence, transistors on extremely critical paths (which have slack less than 1% of the cycle time) are assigned the minimum detectable defect size.

To guardband against downstream process variations we set the minimum defect size as 20% the width (opens) and spacing (shorts) dimensions. We assume that pinholes do not have any parametric impact and can only cause an open if they are bigger than the gate length. Hence, we can assign the minimum-size defect to polysilicon features as shown in Equation (2.4) and Equation (2.5).

[4]A critical path typically consists of only 20-30 transistors and hence the area occupied by a critical path is very small compared to the area of the chip, therefore $K = 10$ is chosen as a pessimistic guardband in our experiments.

Table 2.2: Impact of different defect types on a metal layer

Type	Wire Width	Design Impact
Intrusion	Decrease	Open
Extrusion	Increase	Short
Pinhole	Decrease	Resistance Change
Pindot	No change	None

$$Df_{min}^{CD} = \frac{min(0.2W_{min}, 0.2S_{min}, \delta L_{critical})}{MEEF} \tag{2.4}$$

$$Df_{min}^{Con} = \frac{0.2W_{min}}{MEEF} \tag{2.5}$$

2.2.2 Metal Layer

Bright-field masks with negative resist are typically used to make trenches for depositing copper (dual damascene process). The impact of various types of defects is shown in Table 2.2.

Dummy fill does not have any design impact and can be assigned a large Df_{min}^{CD} and Df_{min}^{Con}. Delay impact of metal layer mask defects is negligible. Hence for a metal layer shape, Df_{min}^{CD} and Df_{min}^{Con} can be evaluated as shown in Equation (2.6) and (2.7).

$$Df_{min}^{CD} = \frac{0.2min(W_{min}, S_{min})}{MEEF} \tag{2.6}$$

$$Df_{min}^{Con} = \frac{0.2S_{min}}{MEEF} \tag{2.7}$$

Table 2.3: Impact of different defect types on a via layer

Type	Via Width	Design Impact
Intrusion	Increase	Short
Extrusion	Decrease	Open/Resistance Increase
Pinhole	None	Metal Short
Pindot	Decrease	Resistance Increase

2.2.3 Via Layer

Dark-field masks with positive resist are typically used to print via layer. Impact of various defect types on via layer is summarised in Table 2.3 and shown in Figure 2.4. Note that regions where the non-fill shapes on adjacent metal layers overlap must be assigned minimum detectable defect size of the inspection tool for contamination defects since even the smallest pinhole defect could cause a short. A 20% change in via area is taken as the constraint to assign defect size for CD and contamination (pindot) defects. Redundant vias will have a larger Df_{min}^{CD} and Df_{min}^{Cont}. We can write the minimum-size defects Df_{min} for a set of mXn redundant vias ($m = 1, n = 1$ for single via) as shown in Equation (2.8)-(2.10).

$$Df_{min}^{CD} = \frac{0.2max(m,n)min(W_{min}, S_{min})}{MEEF} \tag{2.8}$$

$$Df_{min}^{Cont} = \frac{0.2max(m,n)W_{min}}{MEEF} \quad for \quad via \quad features \tag{2.9}$$

$$Df_{min}^{Cont} = a_{min} \quad for \quad metal \quad intersect \quad regions \tag{2.10}$$

In order to accurately assign a minimum-defect size for each of the cases discussed above, the value of MEEF needs to be assigned correctly. Since the value of MEEF varies considerably with optical process paramaters, we must take the largest value across the focus-exposure process window in these calculations. *Current inspection tools support adaptive thresholding where the threshold value is*

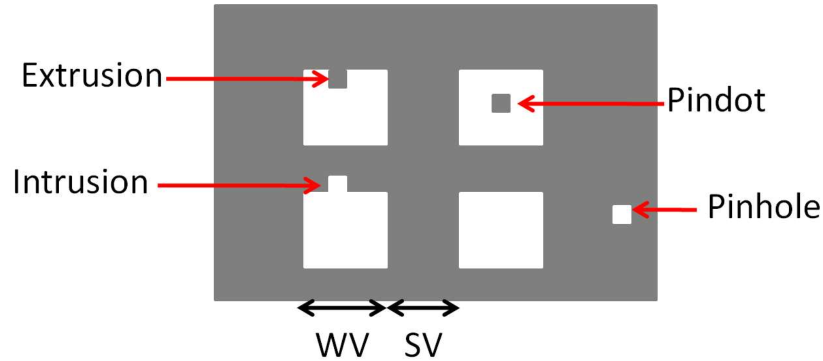


Figure 2.4: Impact of different defect types on a via layer feature

dynamically changed by the tool depending on online MEEF estimation [DMI08]. Hence, we take MEEF=1 in our experiments instead of relying on a lithography simulator to compute it.

2.3 Experimental Results

We implement our neighborhood graph based algorithm to identify redundant vias and dummy fill in C++ using OpenAccess (OA) API [oa]. Layouts of our benchmark circuits implemented in 45nm Nangate OpenCell library along with insertion of double vias and dummy fill was created in Cadence Encounter [enc08]. OPC was performed on the generated GDSII files using Mentor Calibre [cal08].

The size of the post-OPC benchmark circuits that we considered along with improvement in rectangle count due to shape simplification is shown in Table 2.4. The threshold for bucketing was taken as 20nm, which is less than the minimum metal width for 45nm Nangate Design. Around a 50% reduction in number of shapes is observed. Table 2.5 summarizes the results of redundancy finding. Runtime ranges from 5 minutes to 1 hour 19 minutes for the designs considered. The number of redundant vias and dummy fill reported by our approach are

Table 2.4: Shape simplification results

Design Name	# Gates	Area (um^2)	# Rectangles (before)	# Rectangles (after)
AesCipher (8-Metal)	15467	102494	2512023	1012226
Mips (6-Metal)	11577	59461	2876871	1721828
Nova (6-Metal)	43156	268594	13243201	8041773

Table 2.5: Experimental results for redundancy finding

Design Name	# Double Vias	# Dummy Fill	Runtime (min.)	Memory Usage (MB)
AesCipher (8-Metal)	131464	97772	8	910
Mips (6-Metal)	44004	67341	5	1190
Nova (6-Metal)	209623	303792	79	4814

verified with the number obtained from DEF file of the corresponding design. Double vias are reported with 100% accuracy by our approach and there is less than 1% error in dummy fill due to some outliers. The runtime of this algorithm can be improved by partitioning the layout into smaller blocks and using a separate graph for each region. The algorithm can also be parallelized easily by running the critical graph construction step for each set M_iV_j in parallel. These techniques are left for future work.

Table 2.6 shows the percentage vias that are redundant for various via layers along with percentage of dummy area for metal layers. The results are shown for two benchmarks which indicate the potential benefits that can be derived from design-aware inspection of metal and via layers. For metal layers, dummy area is reported as a percentage of the total die area. Note that since higher metal layers typically have less congestion after routing they have a greater percentage

Table 2.6: Layer by layer redundant vias and dummy regions

Design	Via Layer	# Vias	% Redundant	Metal Layer	% Dummy Area
Mips	Via1	71724	23	Metal1	3.6
	Via2	72467	78	Metal2	6.4
	Via3	29970	65	Metal3	8.7
	Via4	12642	36	Metal4	10.9
	Via5	4850	46	Metal5	12.9
				Metal6	20.5
Nova	Via1	266215	22	Metal1	1.8
	Via2	324409	80	Metal2	4.5
	Via3	125926	75	Metal3	5.5
	Via4	37474	49	Metal4	9.9
	Via5	10992	64	Metal5	15.9
				Metal6	25.4

of dummy area.

2.4 Chapter Summary

In this chapter, we developed a comprehensive methodology to assign criticality to each feature of a design layout. To do this, we first proposed and implemented a graph based algorithm that finds non-functional features (dummy fill and redundant vias) in a post-OPC layout with almost 100% accuracy. We then formulated a method to assign a minimum-size defect to each feature of a reticle using timing slack information along with location of non-functional features. This assignment of criticality is then used for two applications:

- Partitioning a reticle such that different partitions can be inspected at

different resolutions (pixel size and sensitivity). This helps reduce mask inspection cost by reducing false/nuisance defects reported by inspection tools. A partitioning algorithm along with the potential benefits of this apporach is covered in Chapter 3.

- Floorplanning an EUV mask with buried defects such that defects lie in non-critical regions of the mask so that they have minimal design impact. A simulated annealing based algorithm along with simulation results for this work is presented in Chapter 4.

CHAPTER 3

Design-Aware Mask Inspection

The traditional approach to mask inspection discussed in Chapter 1 does not use any design information to assess the criticality of defects. It assumes that all printable defects larger than a threshold size (say 10% of mask critical dimension) are critical. If design information is available to mask shops and fabs, they might be able to avoid the expensive process of repair/replacement of the mask due to non-critical defects.

Design information can also be used to reduce false and nuisance defects reported by the inspection tool. Communicating design intent to the inspection tool in the form of additional control layers has been suggested before [VHR03, HLE08, TTN08]. Mask shops can use design information to lower the inspection sensitivity of non-critical regions in order to reduce the number of false defects and nuisance defects. Hedges et al. [HLE08] have shown that up to $100X$ reduction in nuisance defect count is possible just by using variable sensitivity during reticle inspection. Current inspection tools allow the user to define inspection sensitivity on a per pixel basis. But the memory required to store this sensitivity information is impractical since a reticle can have up to 10^{12} pixels. These approaches assume that mask shops know the design criticality of the layout which is rarely the case. Driessen et al. [DGS08] analyze a post-OPC layout to extract some non-critical features in the absence of any design data. Stoler et al [SRM07] extract some criticality information as part of Manufacturing Rule Check (MRC). Both these

approaches focus on extracting assist features from the layout which are a major source of nuisance defects.

In this chapter, we demonstrate how the criticality assignment methodology proposed in Chapter 2 can be exploited to improve mask inspection. The work presented in this chapter first appeared in [KGP10]. We propose an algorithm to partition the reticle, such that each partition can be inspected at a different pixel size and sensitivity in Section 3.1. This partitioning is done in order to reduce false and nuisance defects without missing any critical defects. We show simulation results of our method in Section 3.2. We summarize the chapter in Section 3.3.

3.1 Reticle Partitioning

3.1.1 Problem Formulation

Given the minimum-size defect for each feature on a reticle, partition the reticle such that each partition has length and breadth greater than a predefined value (inspection tool requirement) and is assigned a pixel size and sensitivity (CD and contamination) such that no critical defects are missed and the number of false defects + nuisance defects reported by the inspection tool are minimized.

In order to solve this problem we need to find the minimum detectable defect size as a function of sensitivity and pixel size. The resolution of any digital imaging system scales linearly with pixel size. Also, increasing the sensitivity helps in detecting smaller features. Hence, for an inspection with pixel size p and sensitivity s, we shall model this as shown in Equation (3.1) for both CD and contamination defects. Current inspection tools are capable of inspecting a $20nm$ defect (on the mask) which corresponds to $5nm$ on the wafer ($MEEF = 1$) at a

pixel size of $55nm$ and sensitivity of 100 [DMI08]. Hence $K_c \approx 9$[1].

$$D_{min} = K_c \frac{p}{s} \quad (3.1)$$

The total number of false defects (noise) reported by the inspection tool is a function of pixel size and sensitivity. The largest component of image noise, photon noise, depends linearly on the density of pixels on the sensor and is hence inversely proportional to the square of pixel size. To model other noise sources as well, we assume that noise is proportional to $p^{-\alpha}$, where p is the pixel size. Since false defects are essentially noise, we can model the light intensity falling on a pixel as a Gaussian function, and hence the number of false defects as a function of threshold (inverse of sensitivity) is a complementary error function [S10]. The model for false defects as a funcion of pixel size and two sensitivity levels (S^{CD} and S^{Con}) is shown in Equation (3.2). K^{CD}, K^{Con}, σ^{CD}, σ^{Con} and α depend on the inspection tool. We used a commercial mask shop's inspection data from over 800 reticles with inspection areas ranging from $8000-15000mm^2$, pixel sizes ranging from $72-250nm$, and sensitivities ranging from $75-100$ to fit these parameters and got $\alpha = 2.35$, $K^{CD} = 489.38$, $K^{Con} = 489.38$, $\sigma^{CD} = 0.01$ and $\sigma^{Con} = 0.01$ if the area is taken in mm^2 and pixel size in nm.

$$FD = \frac{A}{p^{\alpha}}(K^{CD} erfc\left(\frac{1}{\sigma^{CD} S^{CD}}\right) + K^{Con} erfc\left(\frac{1}{\sigma^{Con} S^{Con}}\right)) \quad (3.2)$$

The number of nuisance defects depends on the design and the total number of real defects on the mask. Assuming that the defect distribution for a reticle follows the same negative binomial distribution as wafer defects, we can derive a model for the total number of real defects for a reticle of area A, inspected with

[1] K_c is slightly different for CD and contamination types of defects but we assume a constant value for simplicity.

pixel size p, and a single sensitivity s using the following equation:

$$RD = \sum_{DefectTypes} \int_{D_{min}}^{\infty} \frac{K_2}{D^{\alpha}} dD = \sum_{DefectTypes} \frac{K_2}{\alpha - 1} \left(K_c \frac{p}{s} \right)^{\alpha - 1}$$
$$= T^{CD} \left(\frac{p}{S^{CD}} \right)^{\beta^{CD} - 1} + T^{Con} \left(\frac{p}{S^{Con}} \right)^{\beta^{Con} - 1} \quad (3.3)$$

The constants in the above equation were fitted using the same mask shop data described above to obtain $T^{CD} = 0.0002555$, $\beta^{CD} = 1.3$, $T^{Con} = 0.00008208$ and $\beta^{Con} = 0.88$.

Current inspection tools can take DNIR (Do Not Inspect Regions) as inputs. DNIR rules specify that a DNIR region can be as small as one pixel but there is a forty pixel band in each direction that is not inspected. For our partitioning problem this essentially means that a partition must have dimentions of at least 80 pixels (recall that multiple pixel sizes are implemented as multiple scans with DNIRs). For simplicity we assume the same partition for both pixel size and sensitivity and use the largest pixel size in our experiments to define minimum-dimension of a partition. Based on the above discussion, our problem can be stated more precisely as:

Given a reticle with minimum-size defect for each feature, create a partition with rectangular blocks, each of width W_j and height H_j assigning a pixel size p_j, and sensitivities S^{CD}, S^{Con} such that the following function is minimized:

$$F = FalseDefects + \gamma TotalDefects \quad (3.4)$$

and the following constraints are obeyed:

- *Minimum-dimension constraint:*

$$\min(W_j, H_j) > L_{min} \quad (3.5)$$

- *For any feature with minimum-size defect D_{CD} and D_{Con} lying in the jth block of the partition:*

$$D^{CD} > K_c \frac{p_j}{S^{CD}}, \quad D^{Con} > K_c \frac{p_j}{S^{Con}} \tag{3.6}$$

Here γ is a weighting factor that we can choose and L_{min} is the minimum-dimension constraint.

3.1.2 Partitioning Algorithm

We use a scan-line based approach which consists of alternatively moving horizontal and vertical lines across the reticle and placing them at the location which minimizes the cost function[2]. This procedure is done for a finite number of iterations. The number of iterations is chosen such that increasing them does not have a significant improvement in the cost function. The minimum unit by which the scan-lines are moved is equal to the minimum-dimension constraint for the partitions ensuring that Equation (3.5) is obeyed by construction. The change in cost if a line is introduced at a particular position is calculated incrementally to improve runtime. For a particular position of a new scan-line, we only need to look at the neighboring lines in each direction and calculate the cost function for that region. Figure 3.1 gives an illustration of this idea. The current partitioning has blocks P1-9. When a new vertical scan-line is introduced, P2, P5 and P8 are split into two blocks and the improvement in cost function can be evaluated as

[2] This method bears some resemblance to the DNIR placement algorithm proposed in [CLM06].

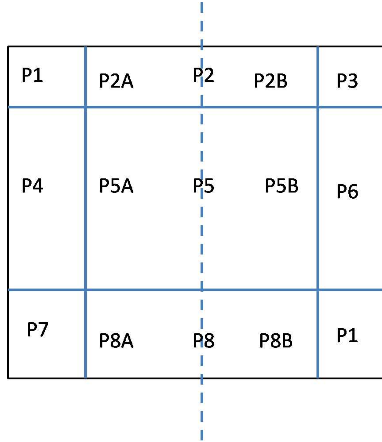


Figure 3.1: Incremental cost evaluation for partitioning

shown in Equation (3.7).

$$\begin{aligned}\delta Cost = & Cost(P2A) + Cost(P2B) - Cost(P2) \\ & + Cost(P5A) + Cost(P5B) - Cost(P5) \\ & + Cost(P8A) + Cost(P8B) - Cost(P8)\end{aligned} \tag{3.7}$$

The cost of any single partition block can be computed by finding the minimum size defect inside the block and finding the sensitivity for each pixel size (only a small number of discrete pixel sizes are available in the inspection tool) to minimize Equation (3.4).

The runtime complexity for this method is $O(W_{chip}N_{iter}/L_{min})$, where W_{chip} is chip size, L_{min} is the moving distance of the scan-line, and N_{iter} is the number of iterations.

3.2 Experimental Results

For assigning minimum-size defect to each layout feature, we use the design rules from Free PDK [fre09]. Timing analysis was done on the post-routed design using

Cadence Encounter [enc08]. Using this criticality assignment, reticle partitioning was implemented in C++. From the commercial mask shop's defect data, it is clear that false defects are typically 10-20X the nuisance defects. But nuisance defects are more important to mask shops as they help improve first-pass yield. Hence, we took $\gamma = 10$ for our cost function in these experiments. Only two pixel sizes, $72nm$ and $90nm$, were used in our experiments. The minimum-dimension constraint was taken as $2\mu m$, which is slightly larger than the dimension of 80 pixels with $90nm$ pixel size. The number of iterations for the scan-line was taken as 500.

Table 3.1 shows a comparison of post-partitioning results with the default case of design-unaware inspection at a single pixel size and sensitivity for the entire reticle. The false and total defect count are evaluated on the basis of our fitted equations derived in Section 3.1.1. Experiments were done for all metal, via and polysilicon layers on two designs, for which the non-functional features have also been reported. Note that the reduction in real defects (non-false defects reported by inspection tool) is due to the decrease in nuisance defects since the partitioning problem is constrained to not miss any critical defect. Since the designs we consider are very small compared to real reticle sizes, we scaled up the values by $1000X$ to obtain defect count comparable to realistic size reticles.

The results of Table 3.1 indicate that up to $4X$ improvement in both false and nuisance defects is observed for higher via layers. The initial value of false and real defect count for Via1-3 and Via4-5 are the same due to similar minimum width design rule. Via layers show the most reduction in defect count due to a large number of redundant vias, some of which are arrays with more than 10 vias. The improvement is smaller in lower via layers due to the metal intersect regions that need to be inspected at high resolution for pinhole defects as discussed in Section

2.2.3. Since lower metal layers are dense, such regions occupy a large fraction of reticle area. The improvement in polysilicon layers is not very substantial because in the designs we used, a large fraction of cells had very small slack and hence the inspection resolution could not be lowered in most regions of the reticle. Metal layers, due to their larger sizes, have the smallest number of false and nuisance defects initially and the improvement is due to dummy regions. *Note that the metal/via layer processing does not require any explicit timing information, while the polysilicon layer leverages it heavily.*

For evaluating the first-pass yield improvement due to design-aware inspection, we implemented a Monte Carlo simulation for both Mips and Nova where real defects were randomly placed all over the reticle corresponding to metal, via and polysilicon layers. The minimum-defect size placed was $7nm$ which is the smallest detectable feature size at a $72nm$ pixel size inspection. Defect size distribution was taken as K/r^3 for a defect of size r, where K is found by taking the maximum defect size of $150nm$. Spatial distribution of the defects was uniform. If a reticle was inspected in a design-unaware fashion, *no* reticle will pass without repair/replacement. With this setup, we iterated over 10000 reticles and we find number of reticles R which do not report any defect with the design-aware inspection. This gives the first-pass yield as $R/10000$. Results for the average first-pass yield of the two designs for each reticle is shown in Figure 3.2.

3.3 Chapter Summary

In this chapter we developed a scan-line based partitioning algorithm to inspect different regions of the layout with different pixel sizes and sensitivities. The method exploits the criticality assignment methodology of Chapter 2 to adjust inspection resolution for different regions. Our simulation results show that up

Table 3.1: Experimental results for partitioning with false and real defect count before and after design-aware inspection

Design Name	Layer	Before		After	
		# False	# Real	# False	# Real
Mips	Polysilicon	14.35	1.63	4.90	1.11
	Via1	15.23	1.43	10.46	1.10
	Via2	14.97	1.35	9.79	0.93
	Via3	14.97	1.35	8.38	0.827
	Via4	14.70	0.88	6.22	0.42
	Via5	14.70	0.88	2.91	0.21
	Metal1	1.06	1.20	0.98	1.11
	Metal2	0.52	1.10	0.44	0.92
	Metal3	0.52	1.10	0.39	0.82
	Metal4	0	0.49	0	0.36
	Metal5	0	0.49	0	0.33
	Metal6	0	0.49	0	0.23
Nova	Polysilicon	69.32	9.04	32.44	6.63
	Via1	137.5	11.3	54.5	5.3
	Via2	67.51	6.10	51.14	4.97
	Via3	67.51	6.10	37.93	4.00
	Via4	66.34	3.99	16.1	1.32
	Via5	66.34	3.99	4.09	0.43
	Metal1	4.79	5.42	4.56	5.17
	Metal2	2.36	4.97	2.07	4.36
	Metal3	2.36	4.97	2.01	4.23
	Metal4	0	2.21	0	1.77
	Metal5	0	2.21	0	1.37
	Metal6	0	2.21	0	0.65

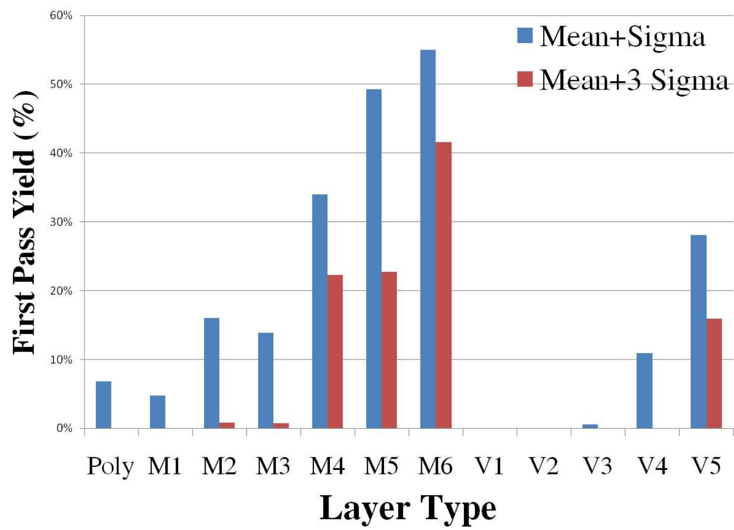


Figure 3.2: First-pass yield with design-aware inspection. The number of defects sprinkled are $\mu+\sigma$ and $\mu+3\sigma$ which is derived from the reticle inspection statistics we have available from a commercial mask shop. Note that the yield is zero for the conventional design-unaware inspection strategy in all these cases.

to $4X$ reduction in nuisance and false defects is possible with this method, along with up to 55% improvement in first-pass yield. These results suggest that design aware mask inspection can significantly lower inspection cost by improving first-pass yield and overall turn-around time due to easier defect review.

CHAPTER 4

Defect-aware Floorplanning for EUV Masks

Due to the challenges associated with producing defect-free EUV blanks as described in Chapter 1, we need to look at methods to print patterns on an EUV mask blank in the presence of buried defects. One potential approach to solve this problem is to compensate for the CD impact of these buried defects by modifying the absorber pattern of the design. [CCN10b] uses a fast defect printability simulator to iteratively modify the absorber pattern based on the thresholded difference between the target and simulated images. [CCN10a] demonstrates two simple techniques; removing part of the absorber and covering the defects with an absorber. Due to the fact that buried defects are actually phase defects, compensated layouts are still very sensitive to change in focus and hence have a small process window.

An alternative method is to move the mask pattern so that defects are avoided. Burns et al. [BA10] demonstrated that 70% of the set of benchmark designs they considered can be matched to a defective blank using a simple enumerative technique of moving the entire mask pattern to avoid the buried defects. The method assumes that a defect is harmless if it lies under the absorber. This assumption is however inaccurate, as the Gaussian shape of the buried defects means that there can be some design impact of the defect even if most of it lies beneath the absorber. Another observation is that many defects that are not under the absorber may still be harmless either due to non-critical absorber

patterns or a large distance of the defect from any absorber. In addition, the method entails moving the entire exposure field of the mask away from the center of the blank, which may not be supported by the current wafer stepper.

In this chapter, we propose a reticle floorplanning based approach to deal with buried defects in EUV blanks. This work first appeared in [KGH11]. We model the buried defects as Gaussian shaped, which is more accurately than [BA10], and then propose a design CD impact metric in Section 4.1. We then propose a simulated annealing based reticle floorplanning algorithm to minimize the CD impact of buried defects in Section 4.2. Then we show results from our simulation study in Section 4.3. Finally, we summarize the chapter in Section 4.4.

4.1 Modeling CD impact of Buried Defects

Although CD impact of buried defects has been extensively studied through experiments and simulations [CN09, TYT10] for different defect dimensions and optical conditions, the focus has always been on dense parallel line patterns. In this work, we assume that their models will hold for general layout patterns. We use a simple pessimistic linear model for CD impact of buried defects that was proposed by Clifford et al. [CN08b] and we use the same image slope and fitting constants. In addition, we make the following assumptions.

- All defects have a Gaussian shape as shown in Figure 4.1. This assumption is reasonable due to the smoothing process during multi-layer deposition [CN08b]. As shown in Figure 4.1, H is the maximum height of the Gaussian defect and Full Width Half Maximum (FWHM) is the width of the defect at height $H/2$.

- CD impact of a defect on a particular absorber is proportional to the height

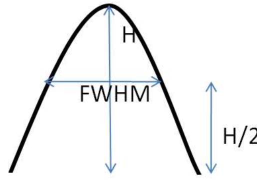


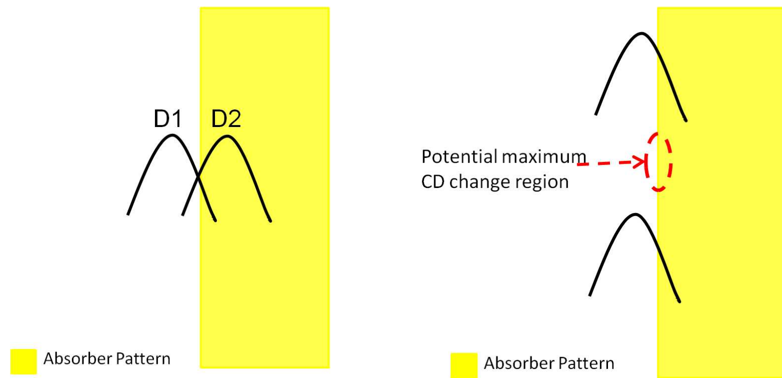
Figure 4.1: A Gaussian defect with height H and full width at half maximum FWHM

of the defect at the closest edge of the absorber. There are two potential sources of error with this assumption. One potential issue is shown in Figure 4.2(a) below, where our assumption implies that defects D1 and D2 have the same CD impact. In reality, the intensity drop of the aerial image, due to the defect, would be larger when most of the defect is not covered by the absorber. The second source of error is due to shadowing, which makes the defect location that causes maximum CD change non-symmetric.

- To account for defocus, which can have a significant impact on CD [CN09, TYT10], we scale the values obtained from the linear model by 3X as a pessimistic approximation. The scaling factor is based on simulation results from [Cli10], for defocus values of $\pm 75nm$.

- A single absorber pattern cannot be affected by more than one defect. This assumption is reasonable considering the fact that defects are randomly distributed across an entire $104mm \times 132mm$ exposure field. Unless the defect density is very high, two defects are unlikely to lie close to a single absorber pattern as shown in Figure 4.2(b).

With these assumptions, the CD impact for a buried defect, as shown in Figure 4.3, can be derived using Equation (4.1) and Equation (4.2), where $I_{NoDefect}$,



(a) Two potential locations of a defect, D1 and D2 relative to absorber edge. Although D1 typically has greater CD impact than D2, we assume both have same CD impact for simplicity

(b) A scenario with two defects changing CD of a single absorber. The worst case CD change may not lie at minimum distance edge fragement of either defect

Figure 4.2: Illustration of some assumptions for modeling of CD impact of buried defect

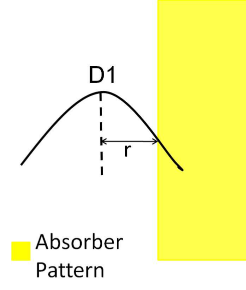


Figure 4.3: A defect and absorber with r as distance between center of defect and closest absorber edge

m_{defect}, b_{defect} and $ImageSlope$ are constants whose values are taken from [CN08b][1].

$$DefectHeight = e^{-r^2/(FWHM/2)^2} \tag{4.1}$$

$$CD_{defect} = \frac{3 \times \sqrt{I_{NoDefect}}(m_{defect}DefectHeight + b_{defect})}{ImageSlope} \tag{4.2}$$

To find whether a buried defect will cause the design to fail or not, we also need to know the acceptable CD deviation that each design shape can tolerate. This CD tolerance can be computed using the method proposed in Chapter 2 when some design information (timing slack) is available to the mask manufacturers. Otherwise, a single conservative CD tolerance can be assigned to each shape in the design. Using the criticality assignment and CD impact of each buried defect, we propose a simple cost metric that estimates the overall design impact of buried defects as shown in Equation (4.3), where D is the set of dies on the mask, BD is the set of buried defects and S is the set of absorber shapes in the design.

$$Cost = \sum_{i \in D} \sum_{d \in BD} \sum_{s \in S} e^{CD_{defect}(i,d,s) - CD_{tol}(i,s)} \tag{4.3}$$

[1] $m_{defect} = 0.191nm^{-1}, b_{defect} = 0.094, I_{NoDefect} = 0.3$ and $ImageSlope = 0.0471nm^{-1}$.

It is worth noting that this cost metric is not equivalent to yield but it is indicative of the overall electrical impact of buried defects on the design. For example, if a single die has multiple defects, moving the die may not improve yield at all but it could still reduce this cost metric. Another important point is that although we have used a closed form expression to calculate the CD impact of a buried defect, it is possible to use a fast simulator such as RADICAL [CN09] for layout snippets around each buried defect to evaluate the design impact more accurately.

4.2 Defect-Aware Reticle Floorplanning

4.2.1 Problem Formulation

The reticle floorplanning problem for defective EUV mask blanks can be formally stated as follows:

Given a die of dimensions $L_d \times W_d$, the criticality of each design shape, and a reticle of dimensions $L_r \times W_r$ along with the location and size of buried defects on it, find a reticle floorplan such that the cost function in Equation (4.3) is minimized.

Floorplanning of dies on a mask is an extremely well studied problem in DUV lithography. Most approaches focus on multi-project reticles with dies of different dimensions. The earliest works focused on achieving the most compact placement of rectangles in a given area [CL03]. B*-tree is an efficient data structure to solve the compact floorplan problem [YCC00]. Many later approaches looked at maximizing the number of chips after dicing the wafer. Kahng et al [Kah07] solved this problem using quadrisection based simulated annealing. The problem was solved as a mixed-ILP in [WLT08].

In this work, we focus on single project reticles and the objective of floorplanning is to maximize yield in the presence of buried defects. We solve the problem for only a single physical layer assuming that all other layers lie on defect-free mask blanks (or are patterned using DUV lithography).

4.2.2 Problem Solution

To solve the single-project reticle floorplanning problem formulated above, we consider only gridded solutions since they guarantee that no die is lost after side-to-side wafer dicing. A non-gridded solution can be more compact but will usually lose some dies during dicing which needs to be accounted for during yield computation.

We use simulated annealing to solve this optimization problem since previous work on floorplanning suggests that it is a good heuristic for floorplanning problems. In this technique, an initial solution is randomly chosen. Any perturbation or change in that solution increases or decreases the cost. If a change or move reduces the cost it is accepted; otherwise, it is accepted with a finite probability depending on the increase in cost and the number of prior iterations. A temperature is usually used as a parameter for this in analogy to thermal annealing. So, initially when the system is hot, most moves, even those that increase cost, are accepted. As the system cools down, the optimizer behaves like a greedy algorithm.

Since we are looking for gridded solutions, we define a set of horizontal and vertical grid-lines. If we have an initial compact floorplan with n_r rows and n_c columns of dies, then we have n_r horizontal grid-lines and n_c vertical grid-lines. Each horizontal (vertical) grid-line has its corresponding $y(x)$ coordinate linked to all dies with the same bottom (left) coordinate. So, each die is linked to two

grid-lines, one vertical and one horizontal. Both horizontal and vertical grid-lines are sorted by the coordinate of the grid-lines. Each grid-line coordinate (and all the linked dies) can be increased or decreased by a predefined value, say δ. This is a move or a perturbation. Hence a vertical (horizontal) grid-line $L_i^V(L_i^H)$ has two possible moves:(1) $x_i(y_i) = x_i(y_i) + \delta$;(2) $x_i(y_i) = x_i(y_i) - \delta$. A move can be valid or invalid based on whether spatial constraints are obeyed after the move is made. The spatial constraints that must be obeyed by every grid-line are as follows:

1. $x_0(y_0) > 0.0$
2. $x_i(y_i) - x_{i-1}(y_{i-1}) >= W_d(H_d)$
3. $x_k(y_k) <= W_d(H_d) + W_r(H_r)$

Here k is the total number of vertical (horizontal) grid-lines, $W_r(H_r)$ is reticle width (height), $W_d(H_d)$ is die width (height) and $i \in 1, 2...k-1$. Figure 4.4(a) graphically illustrates this definition of moves and their validity.

Apart from moving, the die's orientation can also be changed. Each die can have four possible orientations as shown in Figure 4.4(b). 90^0 rotation is not considered since it will typically not be allowed due to lithographic patterning constraints. Other orientation changes can have significant manufacturing overheads as well. Flipping the die would lead to different pin locations and hence require different packages for different dies. Rotation by 180^0 makes wafer testing significantly harder (potentially requiring a different probe-card). Due to these overheads, we have disallowed any orientation changes in our algorithm.

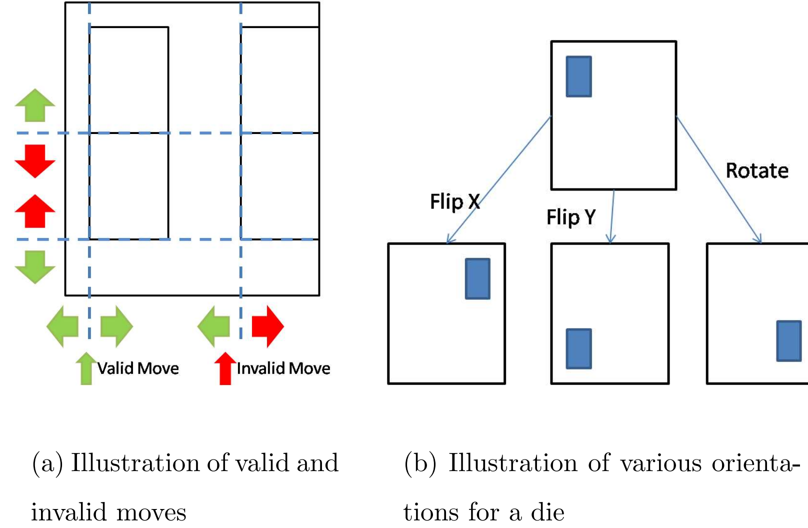
Algorithm 2 describes the defect-aware EUV floorplanning algorithm. Lines 1-3 define an initial partition where the dies are placed in a compact grid on

Algorithm 2 Reticle floorplanning algorithm for EUV mask

Require: Width (W_d) and Height (H_d) of reticle, width(W_r) and height (H_r) of die, CD tolerance of design shapes and location of defects on mask blank.

Ensure: Location of die such that number of defects in critical areas is minimized.

1: Define $n_c = W_d/W_r$ vertical grid-lines starting from origin with spacing W_r.
2: Define $n_r = H_d/H_r$ horizontal grid-lines starting from origin with spacing H_r.
3: Place $n_c \times n_r$ dies on the reticle such that each die is linked to one vertical and one horizontal grid-line based on bottom left co-ordinate.
4: $T = T_{initial}$, r is cooling rate
5: **while** $T > T_{final}$ **do**
6: Randomly pick one valid move m^* that satisfies spatial constraints.
7: **if** cost(m^*) <= 0 **then**
8: Accept m^*.
9: **end if**
10: **if** cost(m^*) > 0 **then**
11: Accept m_{min} with probability $P = \exp(-cost(m^*)/T)$.
12: **end if**
13: $T = T * r$.
14: **end while**



(a) Illustration of valid and invalid moves

(b) Illustration of various orientations for a die

Figure 4.4: Various degrees of flexibility for floorplanner

the reticle along with the grid-line data structure. Line 4-5 and 13 define the standard SA iterations. A valid move is chosen and accepted/rejected based on the conventional SA criteria in Lines 6-12.

4.3 Experimental Results

4.3.1 Setup

For our experiments, we chose buried defects of height $H = 2nm$ and width $FWHM = 100nm$ which are randomly distributed over the maximum reticle field area of $104mmX132mm$ [2]. A typical mask blank is larger than the field size and there is some flexibility in choosing the size and the location of the exposure field. But in this work, we assume the field is always centered as this is typically the case. All results are reported as an average of 1000 random defect distributions. We consider masks as brightfield with GDS shapes corresponding

[2]All dimensions in this section are mask scale unless explicitly stated.

Table 4.1: Different die sizes considered

Design Label	Die Size ($mm \times mm$)	# Die/reticle	Exposure field ($mm \times mm$)
Design A	51.85×65.77	4	103.7×131.5
Design B	51.85×43.85	6	103.7×131.5
Design C	34.45×32.71	12	103.3×130.8

to absorbers. This is a reasonable assumption for the polysilicon layer, which is our focus in this work. The schedule for simulated annealing was taken as $T_{initial} = 100000$, $r = 0.99$ and $T_{final} = 0.001$. When making moves, the distance $\delta = 0.5um$. The implementation was done in C++ using OpenAccess API [oa].

We perform floorplanning with the three different die sizes shown in Table 4.1, using the number of dies that can be fit inside the maximum exposure field area. All three benchmark designs are constructed by tiling copies of the polysilicon layer of a $45nm$ Mips design, which was placed and routed with 75% utilization in Cadence SoC Encounter [enc08] using the Nangate $45nm$ library [nan]. The $45nm$ design is scaled down to $22nm$ before tiling. We then change the exposure field area slightly so that in each case 99.97% of the exposure field is occupied by the design pattern. The exact value of the field size for each die is also mentioned in Table 4.1.

4.3.2 Impact of Design Information

In the absence of any design information, the mask maker can assign a fixed CD tolerance to each absorber shape and then use it to perform floorplanning. In this experiment, we assign a conservative CD tolerance of $0.25nm$, which is 1% of the transistor gate length in a $22nm$ design (wafer scale). The results with this design-unaware approach are shown for different numbers of defects in Table 4.2.

The proposed cost function before and after the floorplanning, along with die yield[3], and mask yield[4] are shown. Mask yield is typically the most important metric for mask makers as it strongly affects the manufacturing cost of masks. We can see that mask yield can be improved to around 90% for up to 60 defects for all three dies. The results indicate that die size does not have a significant impact on initial mask yield or the results of floorplanning optimization.

If mask makers are provided with some design information, they can exploit it to assign different CD tolerances to different absorber shapes based on their criticality. We use the method proposed in Chapter 2 to assign criticality (which is equivalent to CD tolerance) to each feature of the layout. Assigning CD tolerances based on criticality reduces the pessimism in yield computation caused by assigning a single CD tolerance to each shape. This can be clearly seen if we compare the initial mask yield of Table 4.3 compared to that of Table 4.2. Design awareness allows the floorplanner to improve yield by placing non-critical absorber edges close to buried defects. The post-floorplanning mask yields of Table 4.2 and Table 4.3 illustrate this. For example, with 40 defects, the mask yield of a reticle with Die B can be improved only up to 94.2% without any design information, but it can be improved up to 98.4% with design awareness. The difference in yield is even bigger for larger defect count.

4.3.3 Impact of Defect Dimensions

In this section, we explore the impact of defect size on mask yield before and after floorplanning. All results in this section correspond to design B with a defect count of 80. We first study the impact of defect height on mask yield.

[3]Die yield is the average percentage of dies per mask for which CD impact of defect is less than CD tolerance.

[4]Mask yield is the average percentage of masks with all die functioning.

Table 4.2: Experimental results for reticle floorplanning without design information (Die Yield (DY) and Mask Yield (MY))

Design Label	# Defects	Initial			Final		
		Cost	DY(%)	MY(%)	Cost	DY(%)	MY(%)
Design A	20	3038.7	92.7	73	19.2	99.6	98.4
	40	5754.8	86.5	53.3	34.61	98.3	93.7
	60	9045.6	81.2	37.4	72.10	97.0	89.6
	80	11120	77.8	31.2	110.58	93.9	80.8
	100	13761	72.3	19.5	234.51	91.3	71.8
Design B	20	3038.7	95.1	73	18.59	99.8	98.6
	40	5754.8	90.3	53.3	32.75	98.9	94.2
	60	9045.7	86.8	37.4	126.75	98.0	90.0
	80	11120	83.8	31.2	217.4	96.1	82.1
	100	13761	79.2	19.5	319.4	94.4	74.2
Design C	20	2648.05	97.7	75.0	18.45	99.9	98.7
	40	5104.3	95.5	56.7	35.42	99.5	94.9
	60	7984.69	93.7	41.7	150.26	99.1	91.5
	80	9872.4	92.2	35.4	128.60	98.3	84.8
	100	12418	89.9	22.7	300.35	97.6	78.0

Table 4.3: Experimental results for reticle floorplanning with design information

Design Label	# Defects	Initial			Final		
		Cost	DY(%)	MY(%)	Cost	DY(%)	MY(%)
Design A	20	95.52	95.5	82.9	0.03	99.9	99.8
	40	214.49	917	69.2	0.39	99.8	99.2
	60	302.94	87.9	57.3	0.91	99.3	97.5
	80	324.31	85.6	50.4	2.33	98.9	95.8
	100	451.14	81.4	39.3	7.43	98.2	93.3
Design B	20	97.34	97.0	82.9	0.03	99.9	99.8
	40	214.49	94.3	69.2	0.17	99.8	99.1
	60	302.94	91.7	57.3	0.74	99.6	97.7
	80	324.31	89.8	50.4	1.42	99.3	96.0
	100	451.14	86.5	39.3	4.44	98.9	93.9
Design C	20	56.04	98.62	84.4	0.19	99.9	99.7
	40	200.055	97.4	71.8	0.39	99.9	99.1
	60	292.79	96.1	60.8	0.75	99.8	97.9
	80	273.5	95.3	54.5	1.25	99.7	97.0
	100	411.24	93.7	42.8	5.45	99.4	93.4

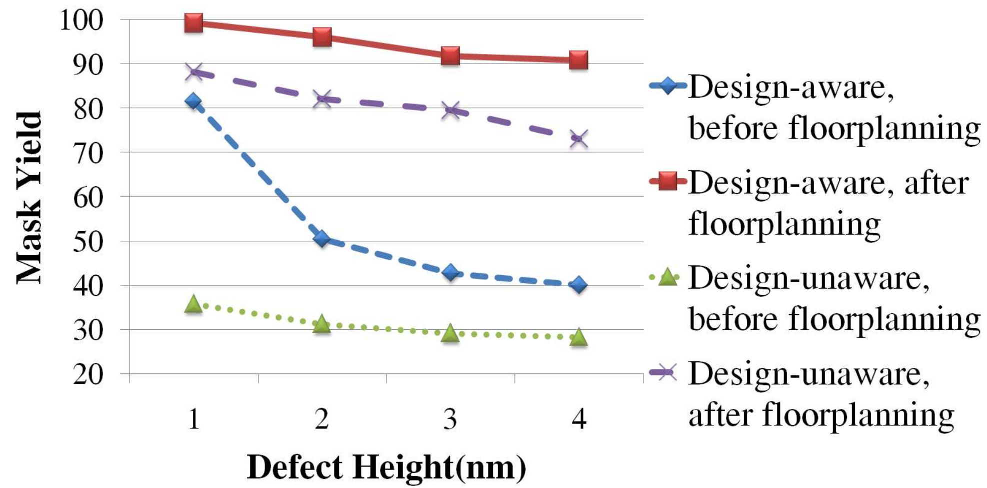
Figure 4.5(a) shows a plot of mask yield for four different defect heights of 1nm, 2nm, 3nm and 4nm. It shows that design-aware yield decreases exponentially as defect height increases before floorplanning. After floorplanning the impact of defect height is less severe and there is only a small linear decrease in yield. For the design-unaware case, yield before and after floorplanning shows a linear dependence, but has a larger slope compared to the design-aware case.

Figure 4.5(b) shows the mask yield before and after floorplanning for $2nm$ buried defects with FWHM ranging from $20nm$ to $100nm$. It shows the strong dependence of mask yield on defect width before floorplanning. This dependence is considerably weakened post-floorplanning as the floorplanner is able to achieve yield above 95% even for $100nm$ wide defects. The figure also shows the benefit of design-aware floorplanning, where the dependence of yield on FWHM is almost negligible.

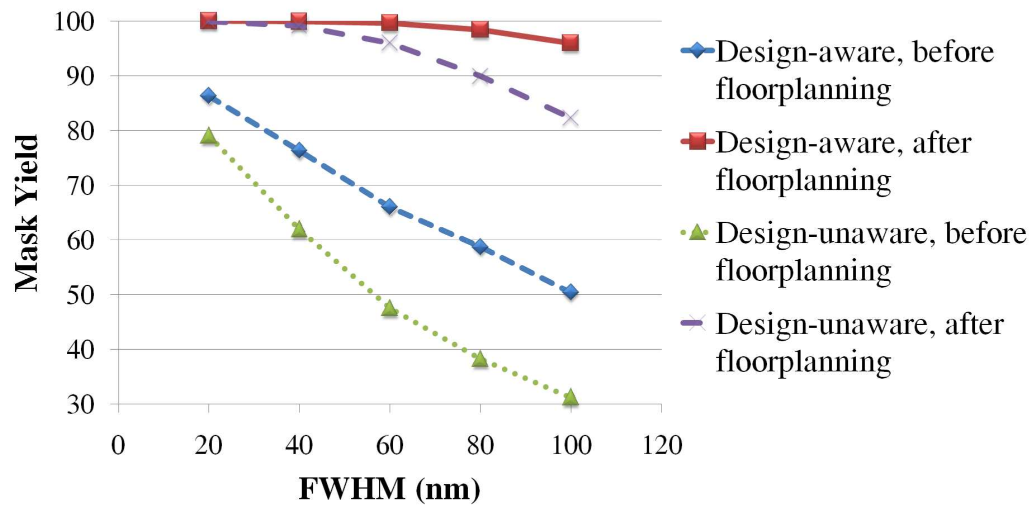
These two plots demonstrate the strong sensitivity of mask yield to defect dimensions. Our floorplanner is able to reduce this sensitivity since it takes defect dimensions into account while computing the design impact of defects.

4.3.4 Impact of Available Free Area

Allowing empty space on the exposure field allows more flexibility in placing dies to avoid buried defects, but it comes at a price. The wasted space reduces the number of dies per wafer. Hence, we need to evaluate how much free space is necessary in the exposure field in order to achieve an acceptable mask yield. To perform this experiment, we choose Die B and modify slightly the dimensions of the exposure field in order to change the free space. The results for a few cases of design B with defect height of $2nm$ and FWHM of $100nm$ are shown in Figure 4.6. This shows that our floorplanner can achieve almost 100% yield



(a) Mask yield before and after floorplanning for different defect height



(b) Mask yield before and after floorplanning for different defect widths

Figure 4.5: Mask yield for different defect dimensions

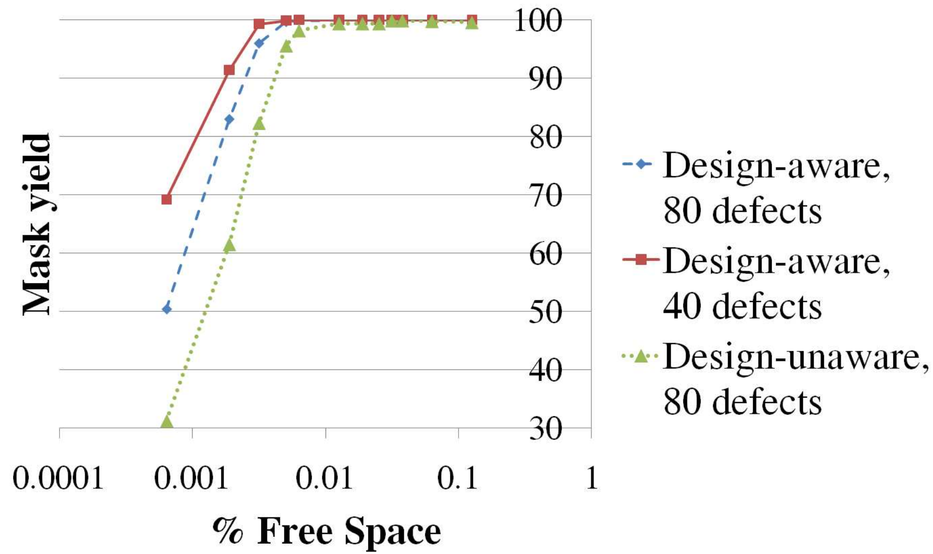


Figure 4.6: Improvement in post-floorplanning mask yield with free space

with a small area overhead of less than 0.01%. The plot also demonstrates that design-awareness and lower defect density helps reduce the area overhead of this approach.

4.3.5 Allowing Orientation Change

In Section 4.2, we briefly discussed the additional overheads of allowing orientation changes during floorplanning. Allowing this degree of freedom can have a significant improvement in mask yield, especially at higher defect densities or with design-unaware floorplanning. Table 4.4 shows the additional improvement in mask yield due to the additional degrees of freedom. The improvement looks small for the design-aware case only due to the fact that the post-floorplanning yield is higher than 95% even without the additional degrees of freedom. Overall the results demonstrate that potential improvements are possible from this additional flexibility, which can be utilized to reduce the area overhead.

Table 4.4: Mask yield for different orientations

Design Type	Defects	Final Mask Yield (%)		
		No Orient. change	Only 180°	All allowed
Design B, design-unaware	80	82.3	88.8	99.2
Design B, design-unaware	60	90	93.3	99.5
Design B, design-aware	80	96	96.6	100

4.4 Chapter Summary

In this chapter, we proposed a simulated annealing based reticle floorplanner that minimizes the design impact of buried defects on EUV mask blanks. We developed a simple model to estimate the CD impact of Gaussian shaped buried defects in the presence of absorber patterns, based on existing literature on EUV defect simulations. We then optimize the CD impact cost metric by floorplanning single project masks in a gridded fashion which improves mask yield substantially, from around 53% to 94% for a 40-defect mask. Adding design information, which essentially allows assigning different CD tolerances to different shapes, can result in further improvements in mask yield, up to 99% for a 40-defect mask. This improvement was achieved with a limited area overhead on the exposure field of only 0.03%. Our floorplanning approach also reduces the sensitivity of mask yield to defect dimensions. Allowing change in orientation of dies, which can have a significant overhead, allows us to reduce the exposure area even further.

CHAPTER 5

Conclusions

To conclude this work, we summarize our key contributions and discuss some directions for future work.

5.1 Summary of Contributions

The exponential increase in the manufacturing cost of integrated circuits with each technology generation has eroded the cost benefit of scaling. A major component of this manufacturing cost is the mask cost. Incorporating design-awareness into the mask manufacturing flow can reduce the inherent pessimism of many manufacturing steps, like mask writing and mask inspection. In this thesis, we propose some techniques to incorporate design intent into mask manufacturing.

In Chapter 2, we developed a systematic methodology to assign criticality (minimum-size defect that causes design failure) to each feature of a design layout. To do this, we proposed a graph based algorithm to locate the non-functional features (redundant and dummy features) in a post-OPC layout (flat and 10X more complex than pre-OPC layout) in the absence of any design information. The implemented algorithm can analyze a 40k-gate design in just 80 minutes with almost 100% accuracy. Based on the location of non-functional features, along with the timing slack of critical paths, we assign the minimum-size defect that impacts the design to each feature of the design layout. This criticality assignment

is then used to develop a design-aware mask inspection flow in Chapter 3 and reticle floorplanning for EUV masks in Chapter 4.

In Chapter 3, we develop a design-aware mask inspection scheme. We use the criticality assignment from Chapter 2 to partition the layout using a scan-line based heuristic, where each partition is assigned a different pixel size and sensitivity to minimize false defects and nuisance defects. Inspection tools impose minimum size constraints on these partitions which are also accounted for in our method. Simulation studies based on our approach show up to $4X$ improvement in false defect + nuisance defect count along with 55% improvement in first-pass yield. These results suggest that considerable reductions in mask manufacturing costs can be achieved by design-aware inspection. In the future, we plan to test our approach in an actual commercial mask shop.

In Chapter 4, we proposed a comprehensive simulated annealing based reticle floorplanning algorithm that can help alleviate the problem of buried defects in EUV masks. Using a Gaussian defect model, we proposed a simple CD impact metric for the entire mask which we minimize during floorplanning. Simulation results from our floorplanner show that for a mask with 40 defects, we can potentially improve mask yield from 53% to 99% with an area overhead of less than 0.03%. Our floorplanner also reduces the dependence of mask yield on defect dimensions.

5.2 Future Work

This thesis proposes techniques to make mask manufacturing design-aware, thereby reducing cost. During the course of this research, several other interesting problems have emerged, especially in the area of EUV masks. For example, in this

thesis, we assume that inspection tools can accurately find the position and the shape of buried defects on the mask blank. But most current inspection tools cannot do this. Hence, one future improvement to the floorplanner is to make it more robust to error in exact defect location.

When multiple physical layers of the design are patterned on defective EUV blanks, the floorplanning problem becomes more challenging due to inter-layer connectivity. Another problem associated with the multiple layer scenario is physical layer-to-mask-blank mapping to maximize the mask yield. We are currently looking into these problems. In addition, our floorplanner assumes a very pessimistic approximate model for CD impact of buried defects. Hence, we plan to incorporate a fast image simulator for EUV masks to predict CD impact more accurately as a part of our floorplanner.

References

[BA10] John Burns and Mansoor Abbas. "EUV mask defect mitigation through pattern placement." volume 7823, p. 782340. SPIE, 2010.

[BEL08] S. Banerjee, P. Elakkumanan, L. Liebmann, and M. Orshansky. "Electrically driven optical proximity correction based on linear programming." In *ICCAD 2008*, pp. 473–479. IEEE Press, 2008.

[cal08] "Mentor Calibre." `http://www.mentor.com/`, 2008.

[CCN10a] Chris H. Clifford, Tina T. Chan, and Andrew R. Neureuther. "Compensation methods for buried defects in extreme ultraviolet lithography masks." volume 7636, p. 763623. SPIE, 2010.

[CCN10b] Chris H. Clifford, Tina T. Chan, Andrew R. Neureuther, Ying Li, Danping Peng, and Linyong Pang. "Compensation methods using a new model for buried defects in extreme ultraviolet lithography masks." volume 7823, p. 78230V. SPIE, 2010.

[CKG11] Tuck-Boon Chan, Abde Ali Kagalwalla, and Puneet Gupta. "Measurement and optimization of electrical process window." *Journal of Micro/Nanolithography, MEMS and MOEMS*, **10**(1):013014, 2011.

[CL03] Shih-Ying Chen and Eric C. Lynn. "Effective placement of chips on a shuttle mask." volume 5130, pp. 681–688. SPIE, 2003.

[Cli10] Chris Heinz Clifford. *Simulation and Compensation Methods for EUV Lithography Masks with Buried Defects*. PhD thesis, EECS Department, University of California, Berkeley, May 2010.

[CLM06] K. Chakraborty, A. Lvov, and M. Mukherjee. "Novel algorithms for placement of rectangular covers for mask inspection in advanced lithography and other VLSI design applications." *IEEE TCAD*, **25**(1):79 – 91, Jan. 2006.

[CN08a] Chris H. Clifford and Andrew R. Neureuther. "Smoothing based fast model for images of isolated buried EUV multilayer defects." volume 6921, p. 692119. SPIE, 2008.

[CN08b] Chris H. Clifford and Andrew R. Neureuther. "Smoothing based model for images of buried EUV multilayer defects." volume 7122, p. 71221X. SPIE, 2008.

[CN09] Chris H. Clifford and Andrew R. Neureuther. "Fast simulation methods and modeling for extreme ultraviolet masks with buried defects." *Journal of Micro/Nanolithography, MEMS and MOEMS*, **8**(3):031402, 2009.

[DFN02] Yunfei Deng, Bruno La Fontaine, and Andrew R. Neureuther. "Performance of repaired defects and attPSM in EUV multilayer masks." volume 4889, pp. 418–425. SPIE, 2002.

[DGS08] F. Driessen, J. Gunawerdana, Y. Saito, H. Tsuchiya, and Y. Tsuji. "Flexible sensitivity inspection with TK-CMI software for criticality-awareness." volume 7122, p. 712222. SPIE, 2008.

[DMI08] A. Dayal, B. Mu, V. Iyer, P. Lim, A. Goonesekera, and B. Broadbent. "Results from the KLA-Tencor TeraScanXR reticle inspection tool." volume 7122, p. 71223G. SPIE, 2008.

[DZB06] A. Dürr, A. Zibold, and K. Böhm. "An advanced study for defect disposition through 193-nm aerial imaging." volume 6152, p. 61522M. SPIE, 2006.

[enc08] "Cadence SOC Encounter." `http://www.cadence.com/`, 2008.

[fre09] "FreePDK 45nm Design Rules." `http://www.eda.ncsu.edu/wiki/FreePDK45:Contents`, 2009.

[GG83] K.D. Gourley and D.M. Green. "A Polygon-to-Rectangle Conversion Algorithm." *Computer Graphics and Applications, IEEE*, **3**(1):31 –36, jan. 1983.

[GKS03] P. Gupta, A. B. Kahng, D. Sylvester, and J. Yang. "A cost-driven lithographic correction methodology based on off-the-shelf sizing tools." In *Proc. DAC*, pp. 16–21, 2003.

[GKS05] P. Gupta, A. B. Kahng, D. Sylvester, and J. Yang. "Performance Driven OPC for Mask Cost Reduction." In *ISQED 2005*, pp. 270–275, 2005.

[HK08] K. Hosono and K. Kato. "PMJ panel discussion overview on mask complexities, cost, and cycle time in 32-nm system LSI generation: conflict or concurrent?" volume 7122, p. 712207. SPIE, 2008.

[HLE08] S. Hedges, C. Le, M. Eickhoff, M. Wylie, T. Simmons, V. Vellanki, and J. McMurran. "Novel mask inspection flow using Sensitivity Control Layers (SCL) on the TeraScanHR-587 platform." volume 7122, p. 71221G. SPIE, 2008.

[itr09] “International Technology Roadmap for Semiconductors(ITRS).” http://www.itrs.net/, 2009.

[KAC01] L. Karklin, M. Altamirano, L. Cai, K. Phan, and C. Spence. “Automatic defect severity scoring for 193-nm reticle defect inspection.” volume 4346, pp. 898–906. SPIE, 2001.

[Kah07] Mandoiu-I. I. Xu X. Zelikovsky A. Z. Kahng, A. B. “Enhanced Design Flow and Optimizations for Multiproject Wafers.” *IEEE Tran. on CAD of Integrated Circuits and Systems*, **26**(2):301 –311, Feb. 2007.

[KGH11] Abde Ali Kagalwalla, Puneet Gupta, Duck-Hyung Hur, and Chul-Hong Park. “Defect-aware reticle floorplanning for EUV masks.” volume 7974, p. 79740Z. SPIE, 2011.

[KGP10] A.A. Kagalwalla, P. Gupta, C. Progler, and S. McDonald. “Design-aware mask inspection.” In *Computer-Aided Design (ICCAD), 2010 IEEE/ACM International Conference on*, pp. 93 –99, nov. 2010.

[KLZ08] T.-Y. Kang, H.-C. Lee, H. Zhang, K. Yamada, Y. Kitayama, K. Kobayashi, and Peter Fiekowsky. “Auto-classification and simulation of mask defects using SEM and CAD images.” volume 7122, p. 71221F. SPIE, 2008.

[KNO07] Kaoru Koike, Kohichi Nakayama, Kazuhisa Ogawa, and Hidetoshi Ohnuma. “OPC to reduce variability of transistor properties.” volume 6521, p. 65210J. SPIE, 2007.

[Lev09] Harry J. Levinson. “Extreme ultraviolet lithography's path to manufacturing.” *Journal of Micro/Nanolithography, MEMS and MOEMS*, **8**(4):041501, 2009.

[MBM08] H. Moribe, T. Bashomatsu, K. Matsumura, A. Uehara, and H. Takahashi. “Improvement of image quality and inspection speed in LM7500 reticle inspection system.” volume 7028, p. 70282K. SPIE, 2008.

[nan] “Nangate Open Cell Library.” https://www.nangate.com/.

[oa] “OpenAccess API.” http://www.si2.org/.

[PLC03] L. Pang, A. Lu, J. Chen, E. Guo, L. Cai, and J.-H. Chen. “Enhanced dispositioning of reticle defects for advanced masks using virtual stepper with automated defect severity scoring.” In *SPIE*, volume 5256, pp. 461–473, December 2003.

[Ras09] Abbas Rastegar. "Overcoming mask blank defects in EUV lithography." Technical report, SPIE Newsroom, 2009.

[S10] "Personal Communication with Dr. Stan Stokowski, KLA-Tencor.".

[SRM07] D. Stoler, W. Ruch, W. Ma, S. Chakravarty, S. Liu, R. Morgan, J. Valadez, B. Moore, and J. Burns. "Optimizing defect inspection strategy through the use of design-aware database control layers." In *SPIE*, volume 6730, October 2007.

[SW80] H.W. Six and D. Wood. "The rectangle intersection problem revisited." *BIT Numerical Mathematics*, **20**(4):426–433, 1980.

[THT08] Siew-Hong Teh, Chun-Huat Heng, and Arthur Tay. "Design-process integration for performance-based OPC framework." In *Proceedings of the 45th annual Design Automation Conference*, DAC '08, pp. 522–527, New York, NY, USA, 2008. ACM.

[TTN08] H. Tsuchiya, M. Tokita, T. Nomura, and T. Inoue. "Die-to-database mask inspection with variable sensitivity." volume 7028, p. 70282I. SPIE, 2008.

[TYT10] Tsuneo Terasawa, Takeshi Yamane, Toshihiko Tanaka, Osamu Suga, Takashi Kamo, and Ichiro Mori. "Actinic phase defect detection and printability analysis for patterned EUVL mask." volume 7636, p. 763602. SPIE, 2010.

[VHR03] W. W. Volk, C. Hess, W. Ruch, Z. Yu, W. Ma, L. Fisher, C. Vickery, and Z. M. Ma. "Investigation of smart inspection of critical layer reticles using additional designer data to determine defect significance." In *SPIE*, volume 5256, pp. 489–499, December 2003.

[WLT08] Meng-Chiou Wu, Rung-Bin Lin, and Shih-Cheng Tsai. "Chip placement in a reticle for multiple-project wafer fabrication." *ACM Trans. Des. Autom. Electron. Syst.*, **13**(1):1–21, 2008.

[YCC00] Yun-Chih Chang Yao-Wen, Yun chih Chang, Yao wen Chang, Guang ming Wu, and Shu wei Wu. "B*-Trees: A New Representation for Non-Slicing Floorplans." In *Proc. DAC*, pp. 458–463, 2000.

[ZA07] Qiaolin Charlie Zhang and Paul van Adrichem. "Determining OPC target specifications electrically instead of geometrically." volume 6730, p. 67303V. SPIE, 2007.