Modeling OPC Complexity for Design for Manufacturability

Puneet Gupta[a], Andrew B. Kahng[a,b,c], Swamy Muddu[c], Sam Nakagawa[a] and Chul-Hong Park[c]

[a]Blaze DFM Inc, Sunnyvale CA, USA 94089
[b]CSE Department, UCSD, La Jolla CA, USA 92093
[c]ECE Department, UCSD, La Jolla CA, USA 92037

ABSTRACT

Increasing design complexity in sub-90nm designs results in increased mask complexity and cost. Resolution enhancement techniques (RET) such as assist feature addition, phase shifting (attenuated PSM) and aggressive optical proximity correction (OPC) help in preserving feature fidelity in silicon but increase mask complexity and cost. Data volume increase with rise in mask complexity is becoming prohibitive for manufacturing. Mask cost is determined by mask write time and mask inspection time, which are directly related to the complexity of features printed on the mask. Aggressive RET increase complexity by adding assist features and by modifying existing features.

Passing design intent to OPC has been identified as a solution for reducing mask complexity and cost in several recent works[2,3,4] . The goal of *design-aware* OPC is to relax OPC tolerances of layout features to minimize mask cost, without sacrificing parametric yield. To convey optimal OPC tolerances for manufacturing, design optimization should drive OPC tolerance optimization using models of mask cost for devices and wires. Design optimization should be aware of impact of OPC correction levels on mask cost and performance of the design. This work introduces mask cost characterization (MCC) that quantifies OPC complexity, measured in terms of fracture count of the mask, for different OPC tolerances. MCC with different OPC tolerances is a critical step in linking design and manufacturing.

In this paper, we present a MCC methodology that provides models of fracture count of standard cells and wire patterns for use in design optimization. MCC cannot be performed by designers as they do not have access to foundry OPC recipes and RET tools. To build a fracture count model, we perform OPC and fracturing on a limited set of standard cells and wire configurations with all tolerance combinations. Separately, we identify the characteristics of the layout that impact fracture count. Based on the fracture count (FC) data from OPC and mask data preparation runs, we build models of FC as function of OPC tolerances and layout parameters.

Keywords: OPC, Fracturing, Mask cost, DFM

1. INTRODUCTION

RET such as assist feature insertion and OPC are mandatory post tapeout steps to ensure printability of features in sub-90nm technology nodes. Doubling of layout data volume every technology node combined with aggressive RET is driving mask set cost to prohibitive levels. Transferring design intent to OPC process can reduce the increasing complexity of masks in sub-90nm technology nodes. Design intent-aware OPC applies different levels of OPC correction to different regions of a design based on their criticality.

There are two approaches for minimizing mask cost using design information. In the first approach, timing and power analysis are performed on the design to identify all critical paths and their corresponding layout features. OPC is then performed with tight tolerances on all critical features and with relaxed tolerances on all non-critical features to minimize mask cost. This approach (e.g., Cote et.al[2]) does not modify the design flow prior to the tapeout. However, relaxing OPC tolerance uniformly on all non-critical features does not lead to the best possible mask cost reductions. In the second approach, tolerance optimization is performed by choosing OPC tolerance combination specific to the standard cell or wire pattern by analyzing its impact on mask cost and design performance simultaneously. Gupta et.al.[3] propose such an approach for minimizing mask cost by relaxing OPC tolerances on standard cells, subject to meeting timing constraints . In this flow, OPC tolerance optimization is performed prior to tapeout.

25th Annual BACUS Symposium on Photomask Technology, edited by J. Tracy Weed, Patrick M. Martin,
Proc. of SPIE Vol. 5992, 59921W, (2005) · 0277-786X/05/$15 · doi: 10.1117/12.633416

Characterization of mask cost and timing impact of different OPC tolerances is the basic step for a complete *design-aware* OPC flow. In this work, we characterize mask cost of standard cells and wires without performing OPC repeatedly with different tolerances. Based on the statistical analysis of FC of standard cell and wire patterns, we construct models and lookup tables of mask cost that can be used for OPC tolerance optimization. For standard cells, we give models of mask cost of polysilicon layer with inner tolerance (IT), outer tolerance (OT), starting side (SSIDE) and fragmentation edgelength (FRAG) of feature edges in the layout. Design engineers can perform trade-offs between parametric yield and mask cost using this model. RET engineers can use the model of mask cost to tune their OPC recipes without running OPC and fracturing. Since the model provides mask cost as a function of layout parameters, library designers can modify device layout to minimize mask cost without running OPC and fracturing repeatedly. Further, MCC can be used to drive mask-friendly layout optimizations that can potentially improve yield.

OPC adjusts edge placement of features in the layout according to tolerance combination within the specified number of iterations. In addition to tolerance combination, the final fracture count of an OPC'ed layout depends on the convergence criterion of the OPC algorithm and the edgelength of fragments. Since OPC algorithm is iterative, modeling edge placement of features and fracture count is very difficult. Instead, we model the response of the OPC algorithm as a function of OPC tolerances and layout parameters. Layout dimensions and geometries of devices in standard cells is different from that of wires. Hence, we perform MCC for standard cells and wires differently.

To build mask cost models, we assume that fracture counts are generated with sign-off OPC recipes and optical models. Unless otherwise mentioned, fracture count of a standard cell refers to the fracture count of polysilicon (poly) layer only. In this work, we do not consider assist feature insertion during MCC. This paper is organized as follows. In Section 2 we present MCC methodology for standard cells. In Section 3 we present MCC methodology for wires. Layout styles and properties for standard cells are very different from those in wires. Hence, MCC approach is different for each. Details of experimental setup and results for standard cell and wire MCC are presented in their respective sections. Section 4 gives a summary of MCC and presents future directions.

2. LIBRARY MCC (LMCC)

Fracture count of a standard cell is a function of OPC tolerances (IT, OT, SSIDE, FRAG) and its layout context. In the absence of any layout feature within the optical radius of influence, fracture count depends entirely on IT, OT, SSIDE and FRAG. We refer to the absence of features within the distance of optical radius as isolated context. MCC for isolated context can be performed by running OPC followed by fracturing on individual standard cells for different IT, OT, SSIDE and FRAG. But in the presence of other standard cells, MCC with IT, OT, SSIDE and FRAG variation is CPU intensive. In real layouts, standard cells exist in many different layout contexts with other standard cells. Apart from the different types of standard cells surrounding a given cell, the placement of cells within the optical radius also impacts the fracture count. Running OPC and fracturing on all possible contexts with different spacing between standard cells is practically infeasible. To characterize mask cost of standard cells in a real layout context, we first identify different layout parameters that impact fracture count in the isolated context. In this work, we perform MCC for isolated context only.

Fracture count of poly layer of a standard cell in isolated context varies primarily with IT, OT, SSIDE and FRAG. Inner tolerance (IT) specifies the maximum tolerable edge movement *inside* the drawn feature. Outer tolerance (OT) specifies the maximum tolerable edge movement *outside* the drawn feature. Starting side (SSIDE) provides offset distances (inside and outside a drawn edge) that can be used by the OPC tool to converge on the edge movements faster. Fragmentation (FRAG) is one of the parameters that can have a significant impact on fracture count. OPC tools fragment a layout feature into segments and perform movement of these segments to correct the feature. The number of segments and hence the number of edge movements depend on the granularity of fragmentation. Fracture count of poly is inversely proportional to the fragment edgelength. Fine-grained fragmentation may result in fine-grained edge movements, that improve image quality. However, fracture count increases rapidly as maximum fragment edgelength is decreased. Variation of fracture count for different IT and OT for different fragmentation edgelengths is shown in Figure 1. For any given IT (or OT), we can observe a decrease in fracture count as the fragment edgelength is increased. In addition to the parameters

described above, fracture count also depends on OPC corrections performed at line-ends and corners. To keep our exploration space limited, we do not vary line-end and corner correction parameters.

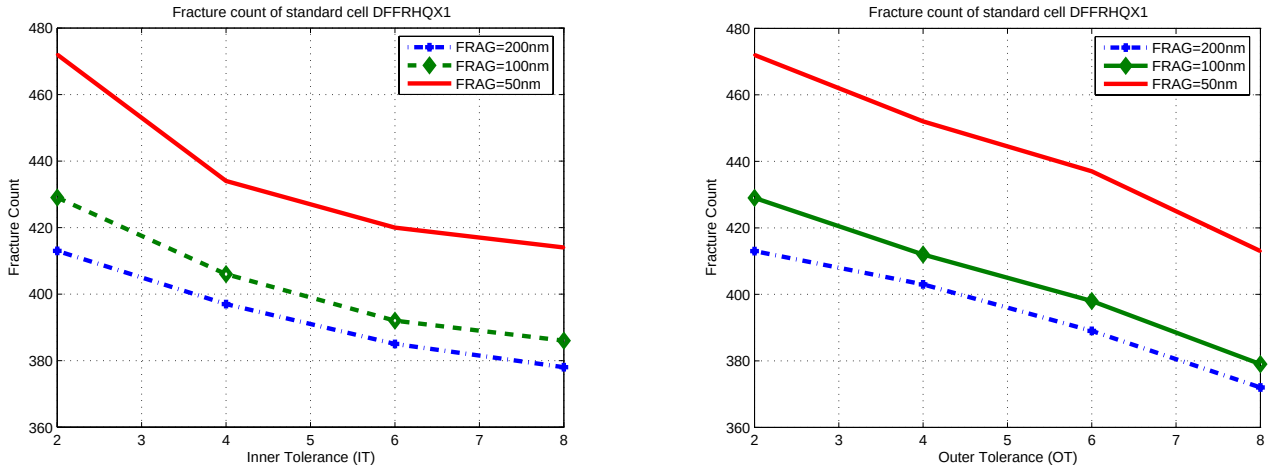

Figure 1. Fracture count variation with IT and OT for maximum fragment edgelengths of 50nm, 100nm and 200nm.

Figure 2 shows the flow chart for building MCC model for standard cell library. To construct a FC model, we choose a subset of standard cells from the library and run OPC exhaustively on all combinations of IT, OT, SSIDE and FRAG. We then perform fracturing on all OPC stream files and collect fracture count data. We identify layout parameters that are the source of fracture count variation between any two standard cells for a given tolerance combination. We then perform linear regression to fit the fracture count of standard cells as a function of layout parameters for each tolerance combination. We then perform linear regression to fit the coefficients of layout parameters as a function of OPC tolerances.

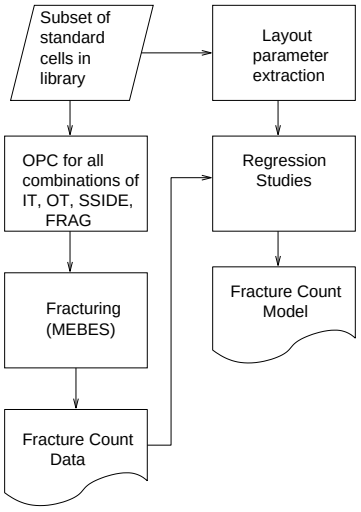

Figure 2. Flow chart of LMCC methodology.

2.1. Layout Parameter Extraction

In this section, we explore different characteristics of standard cell layouts that are the source of variation in fracture count for a given tolerance combination. Figure 3 shows poly and active layers of two standard cells.

λ	0.193
NA	0.68
σ_1, σ_2	0.85, 0.57
Defocus	-0.135
Illumination	Annular
Reference threshold	0.3

Table 1. Optical model parameters.

IT	$\{-2, -4, -6, -8\}$nm
OT	$\{2, 4, 6, 8\}$nm
SSIDE	$\{-20, -10, 0, 10, 25\}$nm
FRAG	$\{50, 100, 200, 500\}$nm

Table 2. OPC parameters.

The poly fracture counts of these two cells for any given tolerance combination. The source of fracture count discrepancy is the layout of the cells. From an initial observation, we can notice that the two cells differ in number of poly features (NP), cell width (CW) and spacing between poly (PS). The actual fracture count depends on optical interactions between the features, which in turn depends on the distribution of spacing between the features. Capturing the distribution of spacing between the poly increases the complexity of analysis. In this work, we only consider the average spacing between poly, defined as the ratio of cell width to the number of poly features. The parameter NP does not differentiate between a "simple" vertical poly and a "fingered" poly with parallel devices. To capture the complexity of features, we consider poly perimeter (PW), which is the total perimeter of poly in the standard cell. To capture the impact of line ends and corners, we consider poly vertex count (PVC) which is the total number of vertices of all poly features in the standard cell. A simple vertical poly has just four vertices. If a notch is added to the vertical poly, the vertex count increases to eight, reflecting the addition of two convex corners and two concave corners.

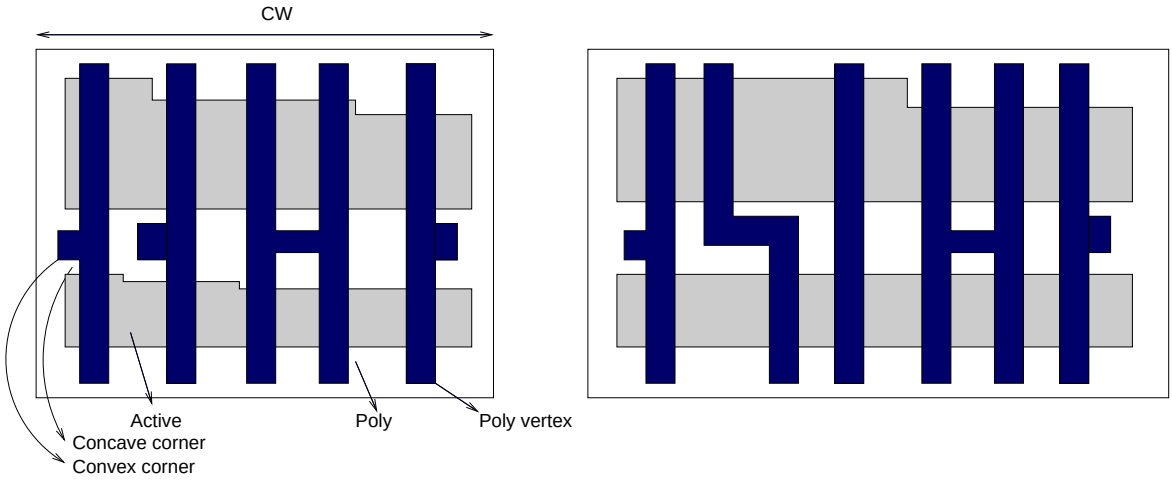

Figure 3. Poly and active regions of two standard cells.

2.2. Experimental Setup

In this section, we give details of our OPC setup and regression studies. To build the fracture count model, we choose 15 of the most frequently used cells from standard cell benchmarks in the 90nm technology. We construct optical model with the parameters given in Table 1. We construct OPC recipes to run OPC exhaustively on all 15 cells for all combinations of IT, OT, SSIDE and FRAG given in Table 2. We use CalibreOPC[1] to run OPC and FractureM (MEBES) to compute total fracture count. IT and OT are implemented using `epeToleranceTag`. SSIDE is implemented using `opcTag -hintOffset` and FRAG using `maxedgelength` parameter in the fragmentation algorithm. Layout parameters outlined in Section 2.1 are extracted by analyzing standard cell GDSII.

Based on the fracture count data and layout parameter values for 15 cells, we perform regression studies to construct FC model using SPLUS[7] software. Figure 4 shows a pair-wise scatter plot of poly fracture count (PFC) along with layout parameters. Column 1 of the figure shows the trend in PFC with NP, CW, PS, PVC

and PW. From the plots we can observe that PFC is strongly correlated with NP, CW, PVC and PW. Since PW is strongly correlated with NP, we choose one of these two for regression.

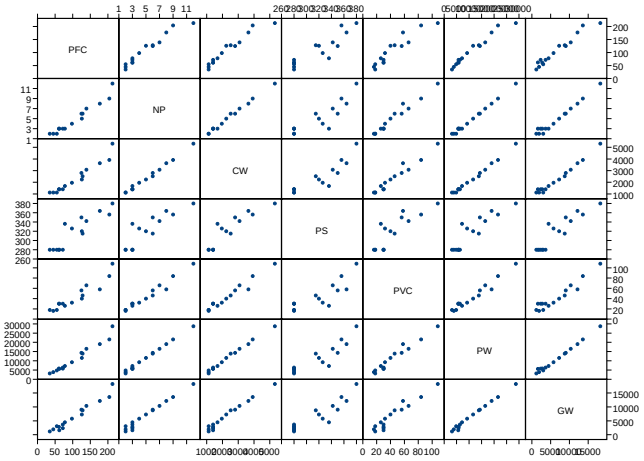

Figure 4. Pair-wise scatter plot showing trends between poly fracture count (PFC) and layout parameters and between different layout parameters.

For each tolerance combination, we run linear regression to fit FC of all 15 cells as a function of layout parameters. Figure 5 shows the response and the fit for all 15 cells for a single tolerance combination (IT = -6, OT = 6, SSIDE = 0, FRAG = 50). Average variance of fit for all 80 tolerance combinations is 0.96, which implies that 96% of FC variation is accounted for using NP, PVC and PW. To test the fidelity of the fit, we predict FC of 100 cells using the 15-cell model. We compare predicted FC values to actual FC numbers obtained from OPC and fracturing. Figure 6 shows predicted and actual FC of 100 cells. From the results we observe that around 62% of the cells have less than 5% error between predicted FC and actual FC. Around 77% of the cells have less than 10% error. For all the remaining cells, the trend in predicted FC tracks that of actual FC closely. Even though the predicted FC differs from the actual FC, the trend in FC is useful for performing tolerance optimization, since the designer needs to be aware of the change in FC rather than the absolute value.

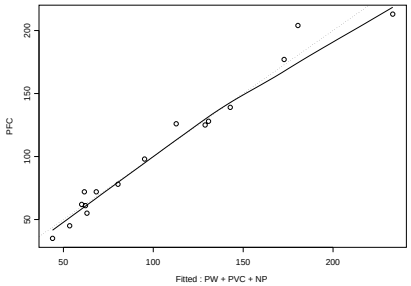

Figure 5. Scatter plot showing actual PFC (dots) versus fit (line) based on three variables, NP, PVC and PW.

3. WIRE MCC (WMCC)

Wire mask cost (WMC) model predicts the FC of wires before running OPC using pre-characterized models and look-up tables. The objectives of WMCC are: (a) to estimate the change of FC due to different OPC tolerances and (b) to predict total FC of a layout prior to OPC and fracturing. WMC model can be used to guide choice

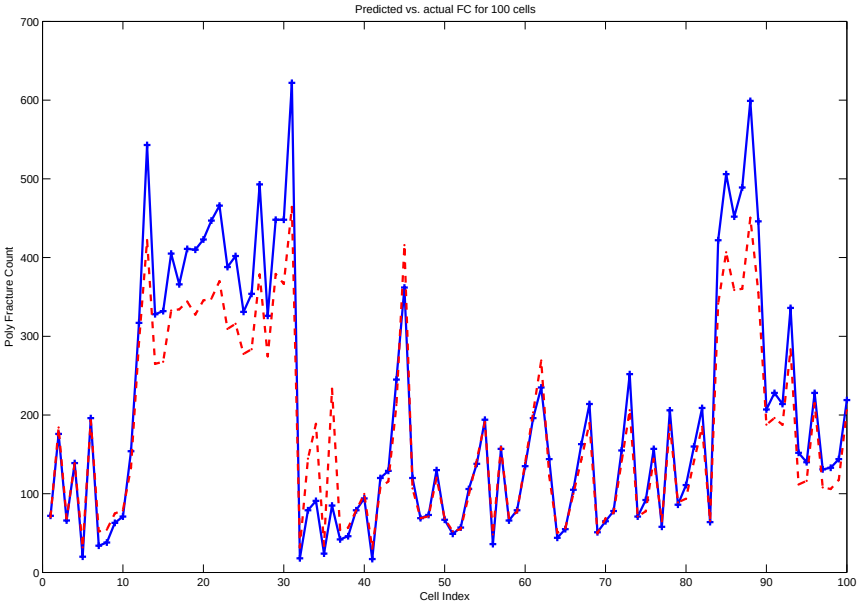

Figure 6. Predicted vs. actual FC of 100 standard cells. The prediction is based on model built using 15 standard cells only.

of OPC tolerances for different regions of a metal layer. In addition to tolerance optimization, WMC model can be used for guiding wire sizing optimizations to minimize FC.

WMC model is a combination of closed-from equations and look-up tables (LUTs) to estimate FC of wires in a layout. The model captures the three major contexts of a wire such as the line-body (L), line-end (LE) and the line-corner (C) as shown in Figure 7. To construct a model for WMC, we characterize the response of each of the different parts of a wire pattern in various layout contexts for different OPC tolerances. The first step in WMC is the construction of different wire patterns that vary layout parameters of interest. In the next step, we run OPC and fracturing on the wire patterns and collect FC data. Using the FC data, we construct closed-form expressions and populate LUTs representing the model. To classify a wire segment as a line-body, a line-end or a line-corner, we first study how FC varies for each part. We introduce a new concept called FC saturation that helps in this classification. Section 3.1 describes FC saturation in detail. Section 3.2 describes WMC model construction and validation for each context of a wire in detail.

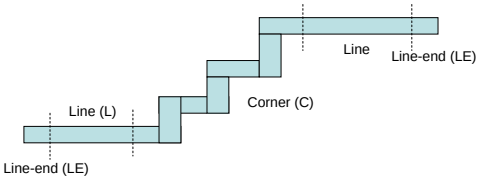

Figure 7. An example of context with line-body, line-end and corner shapes.

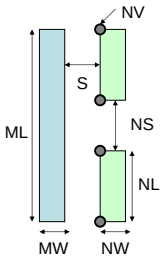

Figure 8. An example of context for WMC glossary.

3.1. FC Saturation

The geometries of wires in the layout along with context can be very complex. To model FC of wires in real layout contexts, we analyze the impact of increasing the number of neighboring wires to a main wire. Figure 8 shows a context of main pattern with two small neighbors. Parameters for the main pattern start with the letter 'M' and those of the neighbors start with the letter 'N'. A glossary of different parameters of the pattern in Figure 8 are summarized in Table 3.

Parameter name	Description
ML	Main (Primary) pattern length
MW	Main (Primary) pattern width
NL	Neighboring pattern length
NW	Neighboring pattern width
NV	Vertex of neighboring pattern
S	Distance between the main pattern and the neighboring pattern
NS	Distance between line-ends of neighboring patterns

Table 3. WMC Glossary.

FC of the main pattern typically decreases as S, NS and NL increase. As S increases, the impact of top right neighbor on OPC of the main pattern decreases. As NS increases, the combined impact of the neighbors on the main pattern decreases. It is interesting to note that as NL increases, FC of the main pattern decreases. The diffraction effects caused by a small neighbor in the vicinity of the main pattern are corrected aggressively by the OPC tool. As NL increases, the diffraction effects spread across the entire length of the main pattern, which are corrected uniformly across the entire length of main pattern. This results in a smaller impact on FC of the main pattern. The decrease in FC of the main pattern ceases as S, NS and NL increase beyond a certain limit. This phenomenon is referred to as saturation and the distance at which saturation takes place is called the saturation point of that parameter. The saturation point of each parameter is identified by a subscript 'o' to the parameter. Saturation point is experimentally determined by varying a layout dimension and measuring the corresponding FC. From our experimental results, we observe that the saturation point of S (i.e., S_o) is equal to the optical radius (OR) of the model used for OPC. The parameters S_o, NS_o and NL_o specify the saturation point of S, NS and NL respectively. The OPC treatment of line-ends and corners of a wire pattern is different from that of the line-body. To determine the extent over which OPC treatment of a line-end or a corner impacts that of a line-body, we construct test patterns shown in Figure 9. We define two parameters, line-end characteristic length (LECL) and corner characteristic length (CCL) to quantify the impact of line-ends and corners respectively. The saturation points of these parameters are $LECL_o$ and CCL_o. The general trend in saturation points of various parameters are shown in Figure 10 and their values are summarized in Table 4. From the results, we observe the following relationships between saturation parameters and OR. We verified these across a different set of OPC recipe and optical model.

$$S_o, LECL_o, CCL_o = OR \tag{1}$$

$$NS_o, NL_o = 2 \times OR \tag{2}$$

3.2. Line-body, End and Corner Models

Based on the saturation points for different parameters, we construct different configurations of the line-body, line-end and line-corners and run OPC followed by fracturing. To analyze FC for each type of pattern, we define a parameter, FC window, that determines the region over which FC is measured. Size of FC window is determined by saturation points.

For the line-body pattern, we observe that the FC is a first degree polynomial function of ML with a slope α and intercept β (FC = αML + β(S,EPE_{tol})) where, EPE_{tol} is the EPE tolerance specification of the main line.

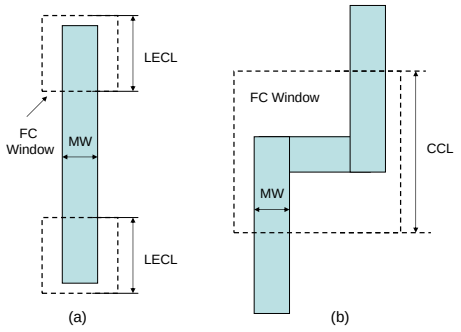

Figure 9. Test structures for (a) line-end characteristic length (LECL) and (b) corner characteristic length (CCL).

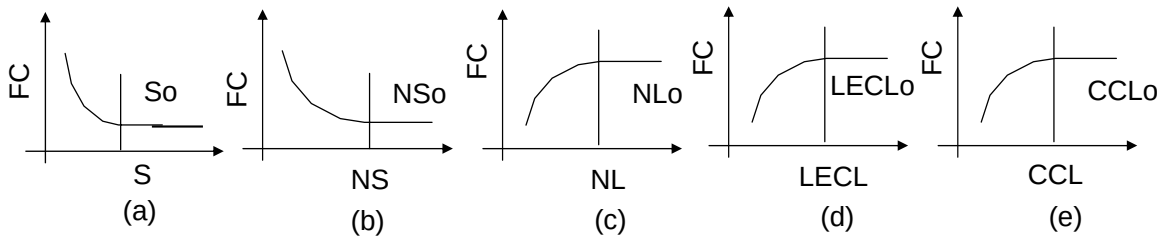

Figure 10. General trend in FC saturation with (a) S, (b) NS, (c) NL, (d) LECL and (e) CCL.

The slope of the function is independent of pattern parameters and is dependent only on the optical parameters. In the presence of neighbors, we observe a shift in the FC of the line-body and this captured by β. The intercept itself is a function of spacing between the main line and the neighbor and EPE tolerance. To compute the slope of the model, we construct the test patterns shown in Figure 12(a) which shows two main lines M1 and M2 of lengths ML_1 and ML_2 respectively. The value of α is computed as $FC(M2) - FC(M1)/ML_2 - ML_1$. To obtain the value of β we construct LUT with S and EPE_{tol} as parameters. The test pattern shown in Figure 12(b) is constructed for different values of S and OPC is run at different EPE_{tol}. To verify the model, we compute α and generate LUT for β and predict FC of two test patterns as shown in Figure 11. The LUT for β with S and EPE_{tol} and the comparison of predicted FC with experimental results for two examples are summarized in Table 5. FC increase with increase in number of neighbors is additive. E.g., FC of main line in Figure 11(b) is $\alpha\text{ML} + \beta(S1, EPE_{tol}) + \beta(S2, EPE_{tol})$.

For the line-end model, we consider two types of patterns as shown in Figure 13. Part (a) of the figure shows a line-end with a single neighbor and part (b) shows the line-end with two neighbors at equal spacings from the main line. FC model for line-end is based completely on LUTs. For a line-end with single neighbor, we construct the LUT by varying S and EPE_{tol}. For a line-end with two neighbors, we observe that FC variation is a function of spacing to the smaller of the two neighbors. To validate the line-end model, we construct test patterns shown in Figure 14. FC window for the line-end is determined by value of $LECL_o$. Table 6 gives the LUTs for line-end for single and double neighbor cases. Table 7 compares the predicted versus measured FC for line-ends shown

Dimension (S/NS/NL) (μm)	0.3	0.65	1.3	1.5
FC for S	101	79	79	79
FC for NS	158	138	124	124
FC for NL	86	93	101	101
FC for LECL	47	59	60	60
FC for CCL	69	71	71	71

Table 4. Saturation points characteristics: $S_0 = 0.65\mu$m, $NS_0 = 0.13\mu$m, $NL_0 = 0.13\mu$m, $LECL_0 = 0.65\mu$m and $CCL_0 = 0.65\mu$m.

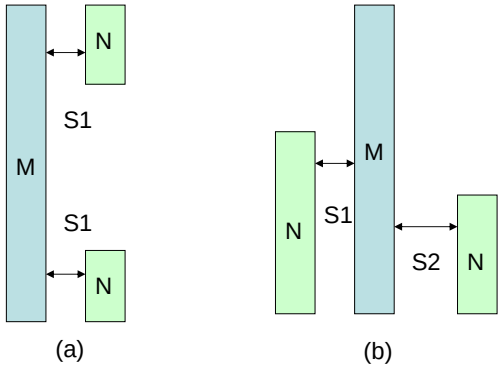

Figure 11. Two examples for validation of line-body model: (a) two neighbors with same space ($S1 = 200nm$) and (b) two neighbors with different spaces of ($S1 = 200nm$ and $S2 = 400nm$)

EPE_{tol}	**LUT (Space: nm)**			**FC of example (a)**		**FC of example (b)**	
	200	400	600	Experiment	Simulation	Experiments	Simulation
0.002	23	10	3	123	124	114	111
0.003	24	12	0	127	126	109	114
0.005	15	6	0	109	108	101	99
0.010	13	6	0	103	104	98	97

Table 5. LUT for line-body model and comparison of FC with simulation and experimental results for two examples: maximum difference of FC is 5.)

in Figure 14 for different neighbor spacings. If a line-end with asymmetrically-spaced neighbors is encountered in a real layout, we choose the neighbor that is closer to the main line. Line-ends are fragmented heavily during OPC. The presence of wire patterns around the line-end does not change the fragmentation significantly and hence, we see a small change in FC with change in neighbor spacing.

WMC model for the line-corner is generated by running OPC and fracturing on line-corner patterns with different CCL and EPE_{tol}. Based on the analysis of real layouts, we observe that the number of line-corners within the FC window (determined by CCL_o) does not exceed two. Hence, we construct LUTs with CCL and EPE_{tol} as parameters and use it for predicting FC of line-corners in real layouts.

3.3. FC Prediction

To validate the line-body, line-end and line-corner WMC models presented above, we analyze a real layout and predict its FC and compare it with real FC after OPC and fracturing. For each metal layer, the model generator

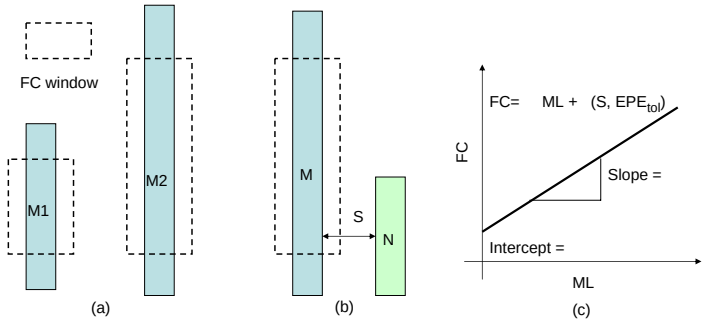

Figure 12. Line-body model: (a) Test patterns for α calculation, (b) test patterns for β LUTs generation and (c) line-body model ($FC = \alpha \mathrm{ML} + \beta(S, EPE_{tol})$).

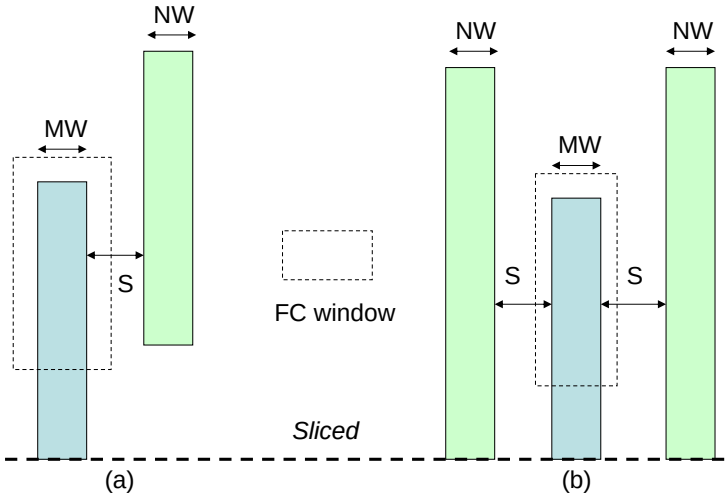

Figure 13. Pattern examples for line-end model: (a) line-end with single neighbor and (b) line-end with double neighbors.

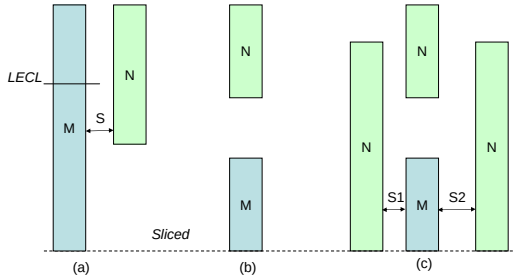

Figure 14. Three examples for line-end model validation: (a) line-end with S = 200nm. (b) line-end perpendicular neighbor. (c) line-end with S1 = 200nm and S2 = 400nm.

constructs a test layout that captures the three contexts of a wire pattern. We then extract the optical radius (OR) of the OPC model from a line and space test pattern. We then run OPC and fracturing on the test layout and record FCs of contexts. These FCs are used to compute α and populate LUTs.

To predict FC of real layouts, the predictor decomposes wire patterns from real layouts into the three contexts based on FC windows. FCs for patterns in each category are computed using the models and LUTs. Total FC of the layout is the sum of FC for each context. Table 8 shows the real and predicted FC for metal layer 2 of ALU128 benchmark in the 90nm technology. The maximum error in prediction is ±6%.

4. CONCLUSIONS

In this paper, we have presented methodologies for characterizing OPC of standard cells and wire patterns in terms of fracture count. The characterization provides models of FC as a function of layout parameters and

EPE_{tol}	**LUT for single neighbor (Space: nm)**				**LUT for double neighbor (Space:nm)**			
	200	400	600	650	200	400	600	650
0.002	16	15	13	12	14	13	12	12
0.003	17	14	13	12	13	12	11	12
0.005	15	14	11	11	12	12	11	11
0.01	12	10	10	9	10	9	9	9

Table 6. LUTs for single and double neighbors cases of the line-end model: $LECL_0$ is 650nm.

EPE_{tol}	FC for example (a)		FC for example (b)		FC for example (c)	
	Experiment	Simulation	Experiment	Simulation	Experiment	Simulation
0.002	15	16	12	12	13	14
0.003	14	17	11	12	12	13
0.005	13	15	11	11	11	12
0.010	11	12	10	9	10	10

Table 7. Comparison of predicted FC with real FC for the three line-end test patterns in Figure 14.

EPE_{tol} **(nm)**	**Real FC**	**Predicted FC**	**Error(%)**
0.002	306234	297022	3
0.004	282202	265312	6
0.006	227529	242100	-6
0.008	213306	221046	4.5

Table 8. Real versus predicted FC for different EPE_{tol} of metal layer 2 from ALU128 benchmark implemented in the 90nm technology

OPC tolerances. FC model constructed for library MCC using a limited set of library cells for a given tolerance combination can predict the FC trend of up to 75% of cells in the library within 5% error. FC model for wires can predict actual FC of layouts within 6% error. These FC models can be used by designers to choose between different OPC tolerance combinations to minimize mask cost. RET engineers can extend the presented models by adding other OPC parameters and use it for tuning OPC recipes and optical model parameters. Library designers can use these models for constructing design rules that minimize mask cost without actually running OPC and fracturing. The placement of standard cells and the type of standard cells surrounding a given cell have significant impact on its FC. We are currently working on extending the library MCC approach to include the impact of layout context. Line-ends and corners of poly features contribute significantly to FC. Optimizing line-end corner fragmentation parameters can enable further reductions in FC.

REFERENCES

1. `http://www.mentor.com/products/ic_nanometer_design/litho_modeling/calibre_ret/`
2. M. L. Cote, P. Hurat, A. Miloslavsky, D. Goinard and M. L. Rieger, "Mask Cost Reduction and Yield Optimization Using Design Intent", *Proc. SPIE on Design and Process Integration for Microelectronic Manufacturing*, 2005, pp. 389-396.
3. P. Gupta, A. B. Kahng, D. Sylvester and J. Yang, "A Cost-Driven Lithographic Correction Methodology Based on Off-the-Shelf Sizing Tools", *Proc. ACM/IEEE Design Automation Conf.*, 2003, pp. 16-21.
4. M. E. Mason, "Rising cost of RETs: Understanding the Value Proposition", *Proc. SPIE on Design and Process Integration for Microelectronic Manufacturing*, 2004, pp.10-19.
5. M. L. Rieger, *Mask EDA Workshop*, 2001, `http://www.sematech.org/resources/litho/meetings/mask/20010711/G_AVANTI.pdf` .
6. S. F. Schulze and J. Word, "Interaction of RET and MDP: Optimization for Reducing the Mask Writing Time", *Proc. SPIE on Design and Process Integration for Microelectronic Manufacturing*, 2004, pp. 170-181.
7. `http://www.insightful.com/products/splus/`
8. Y. Zhang, S. Chou, B. Rockwell, G. Xiao, H. H. Kamberian, R. Cottle and C. J. Progler, "Mask Cost Analysis via Write-time Estimation", *Proc. SPIE on Design and Process Integration for Microelectronic Manufacturing*, 2005, pp. 313-318.