Metrics to Assess Fracture Quality for Variable Shaped Beam Lithography

M. Bloeckera*, R. Gladhillb, P. D. Buckb, M. Kempfa, D. Aguilarc, R. B. Cincua

aAdvanced Mask Technology Center (AMTC), Rähnitzer Allee 9, D-01129 Dresden, Germany;
bToppan Photomasks, Inc., 23932 N.E. Glisan, Gresham, OR, USA 97030;
cToppan Photomasks, Inc., 400 Texas Avenue, Round Rock, TX, USA 78664

ABSTRACT

CD control requirements for advanced node masks are in the low single digit nanometer range. CD control for Variable Shaped Beam (VSB) lithography that is used to manufacture these masks is dependent on the post-fracture figure layout. Shot linearity, shot size repeatability, and shot placement repeatability can affect CD control differently based on the figure layout. The potential CD error contribution from poorly optimized fracture strategies thus can be a significant contributor to the total CD error.

In this paper we present a set of fracture quality metrics based on the impact on mask CD control and methods using EDA software to grade fracture strategies based on these fracture quality metrics. We also discuss applications of this metric for fracture tool design and the implementation of different fracture strategies into mask manufacturing including examinations of the predictability of fracturing results. Finally, we will discuss the usage of existing information about the design (such as design intent) in conjunction with the proposed quality metrics to judge different fracture strategies.

Keywords: Mask Data Preparation (MDP), Variable Shaped Beam (VSB), Fracturing, Sliver, CD uniformity, OPC

1. INTRODUCTION

With each technology node introduced into semiconductor manufacturing the requirements for photomask pattern fidelity and critical dimension (CD) control are becoming tighter. At the 65nm technology node the International Technology Roadmap for Semiconductors (ITRS) requires a CD uniformity of 4.8nm (3σ) for dense lines. Starting with the 130nm technology node these ever tightening mask specifications have lead to the deployment of Variable Shaped Beam (VSB) electron beam lithography tools into mask manufacturing of critical layers. As these write tools form the mask pattern by the sequential exposure of basic trapezoidal shapes (usually rectangular and right-angled triangular shots) they provide superior pattern fidelity and especially critical dimension (CD) control over raster scan pattern generators while still achieving reasonable write times.

Due to the method of pattern formation in VSB write tools the machine-specific data formats only need to (and by design can only) represent these basic shapes. The final decomposition of post-fracture figures into the final exposure shots is done on the write tool. The major mask data preparation (MDP) process in this context is to break up complex polygons present in the incoming GDSII or OASIS stream formats into these basic shapes. This process is usually referred to as “fracturing”. Fracturing for VSB write tools has to fulfill two sometimes competing requirements regarding the figure division (see Fig. 1 for an example):

1. Historically mask lithography tools have been the most expensive mask manufacturing equipment and the electron beam lithographic process is still one of the major drivers for mask cost. As the number of shots directly correlates to the required write time for a certain dataset the number of shots required to expose the whole mask pattern needs to be minimized to contain mask cost. Since the critical feature size in advanced node patterns is typically less than the maximum shot size post-fracture figures are often equivalent to shots since no further decomposition is necessary to expose the figures. This is also driven to some extent by the increasing complexity of Optical Proximity Compensation (OPC), which has the effect of breaking long, straight edges

* Martin.Bloecker@amtc-dresden.com; Phone +49 (351) 4048-381; Fax +49 (351) 4048-9381; www.amtc-dresden.com
† The maximum shot size for a NuFlare EBM 5000 series tool is 1µm, which is determined by write tool hardware constraints.
into short jogs. This results in an increased number of vertices that need to be considered during fracturing. During polygon division the fracturing algorithm thus needs to minimize the number of figures while ideally also considering the maximum shot size of the target tool to minimize write time and mask cost.

2. As will be discussed in detail later in this paper the figure division for VSB write tools can also have an impact on CD control. Properties of the mask lithography process such as shot linearity, shot size and dose repeatability, and shot placement repeatability can affect CD control differently based on the figure layout. A simple example is the case of a critical feature composed of one vs. two figures in the critical axis. The constituent errors that affect CD control are different in these two cases. Due to this the fracturing process needs to avoid splitting critical features into two shots and especially avoid the creation of small figures (so called slivers), e.g. figures with one dimension smaller than a certain threshold value. Slivers are more critical in this context compared to split critical features as the effect of shot size linearity is more pronounced for smaller target feature sizes.

The fracturing process can be considered to be an optimization problem to find a polygon partition that meets all requirements for figure division which is a well studied problem. As maintaining the data integrity during fracturing is the foremost requirement for the MDP process well-proven commercially available electronic design automation (EDA) tools such as CATS from Synopsys, CalibreMDP from Mentor Graphics or PATABCON from Nippon Control Systems (NCS) are used for the polygon division in production environments. Apart from optimizing the fracturing result regarding shot count and CD control the fracturing process also has to fulfill production requirements with respect to speed and throughput. To achieve this throughput commercially available MDP tools in general use at least some heuristics to find an optimized fracture solution. The heuristic methods applied can depend on certain aspects of the input data configuration. Practically this leads to the effect that only slight changes in the input data (such as applying a different data bias) can significantly influence the figure division in the output file. Phenomenologically this is a non-predictable behavior of the fracturing algorithm. Due to their possible impact on mask quality these cases need to be detected in a production environment. It is necessary to verify the output data quality by applying a certain quality metric to it. Historically the main metric has been the number of small figures (also referred to as sliver count).

Well defined metrics exist to assess the quality of post-MDP datasets with respect to write time. These metrics mainly are shot count and file size. Those metrics will not be discussed in detail in this paper. We will focus on the discussion of fracture quality metrics that are related to the CD impact of the figure division. We will show that sliver count as a fracture data quality metric is unsuitable for certain applications.

Previous work of some of the authors has focused on the application of design rule check (DRC) methodologies for optimizing the reticle inspection process. In this paper we discuss the application of DRC methodologies to define and implement more suitable metrics for fracture quality. We discuss two possible enhanced fracture metrics and furthermore we also discuss applications for this quality metric and some results we obtained in applying it. Specifically we present benchmark results we obtained in comparing commercially available MDP tools.

2. MOTIVATION

Optical lithography with 193nm wavelength is expected to be the lithography solution down to the 45nm technology node and may even be extended to the 32nm technology node if the development of immersion lithography tools and materials such as high-NA fluids can be completed in time. Due to the growing gap between the exposure wavelength and the minimum feature size on wafer more and more aggressive resolution enhancement techniques such as optical proximity correction (OPC), sub-resolution assist features (SRAF) and off-axis illumination (OAI) are applied. From a mask layout perspective these techniques lead to additional vertices and corresponding edges being introduced into the layout to be exposed on the mask. These additional vertices can lead to layout situations that cannot be resolved without small figures or split features. Due to the increasing number of edge fragments for smaller technology nodes the possibility of layout situations that require slivers to be created is rising. This in turn raises the complexity of the polygon division optimization process.

Where we present fracture results in this paper they have been obtained by using one of the mentioned MDP tools. Due to non-benchmarking restrictions in the MDP tool license terms all results have been made anonymous.
Fig. 1: Different fracturing solutions for a design situation that cannot be resolved without slivers. The polygon division can either be optimized for figure count, sliver count or CD impact.

The analysis of fracture quality using sliver count as a metric in these cases can be problematic because a significant amount of slivers cannot be avoided due to the design. In addition to that sliver count does not take into account if the figures were placed in such a way that their impact on the CD performance is minimized. Fig. 1 shows a (hypothetical) example of a polygon that cannot be fractured without creating slivers. Two divisions exist that both yield the same sliver count (two slivers) while their impact on CD performance is likely to be different. The fracturing solution shown in the middle only requires one sliver but may not be optimal with respect to CD performance either.

The mentioned increasing complexity of photomask designs and as a consequence the more difficult optimization of polygon slicing is contrasted with a rising demand for improved CD control. To ensure this CD control the result of the MDP process needs to be verified. Without a well defined and meaningful quality metric this process needs to rely on human rating of the resulting fractured data during review. To support this review commercial EDA tools offer methods to highlight both slivers and/or split critical features.

While the quality requirements for the MDP process have increased the cycle time budget for the MDP process has remained constant over the technology nodes. Typically a certain amount of the total cycle time is attributed to the complete MDP flow. Apart from the increase in compute power to handle increasingly complex designs this cycle time target can only be met by improving the efficiency of the MDP process. Main leverages for cycle time improvements are the reduction of human interaction, avoidance of queue times and the automation of the MDP process. An automation of the fracture quality review process in this context would yield the following advantages

- “Known good” datasets can be automatically introduced to the subsequent mask manufacturing process steps without further human interaction thus eliminating both queue and manual review times,
- A well defined quality metric eliminates subjective human judging in classifying the quality of fractured datasets and thus provides a more robust quality review process and
- An automated fracture quality review is a prerequisite to automatically initiate alternative MDP processes such as refracturing the data with different parameter settings or a different EDA tool. This again reduces queue time as the alternative MDP process can be started automatically even during off-hours.

The introduction of an automated fracture quality review process is quite similar to the case of the optical rule check (ORC) to verify the optical proximity correction (OPC). It is to be noted that the above mentioned advantages only hold true if a well defined and meaningful metric is applied in the automated fracture quality review process.

2.1 Constituent Errors for Variable Shaped Beam Lithography

As mentioned earlier the pattern formation in variable shaped beam lithography is done by exposing simple rectangular or triangular shots into the resist. During this pattern transfer process different error sources contribute to the CD width finally produced on the mask. The CD error contribution of the measurement process can be neglected in this context as its impact at least to first order is independent of the figure division. The CD result $CD_{undivided}$ obtained for a figure
exposed by a single shot with nominal size $C_D$ is influenced by both a statistical (or noise) error $\Delta C_D$ as well as a systematic or deterministic error $\Delta C_D$:

$$C_D_{\text{undivided}} = C_D + \Delta C_D + \Delta C_D$$  \hspace{1cm} (1)$$

Sources for deterministic error contributions are the shot linearity as well as an imperfect shot size calibration. The shot size linearity is caused by interactions in the electron optical system. As a result the deposited energy distribution and thus the resulting CD value are a function of the target shot size:

$$\Delta C_D = f(C_D)$$  \hspace{1cm} (2)$$

In the case of divided shots in addition to the error sources for single shots additional error sources due to the relative placement of the two shots contribute to the overall CD result:

$$C_D_{\text{divided}} = C_D + \Delta C_D + \Delta C_D + \Delta C_D + \Delta C_D + \Delta C_D + \Delta C_D + \Delta C_D$$  \hspace{1cm} (3)$$

While deterministic error contributions (the first bracket in equation 3) can in principle be addressed by process optimizations\(^\dagger\) such as tuning of the electron beam system the noise component represented by the last bracket of equation 3 will be larger for the divided figure compared to the undivided case. This alone results in a reduced CD control for the divided figure case. Due to the above considerations MDP tools should in general prefer to create embedded slivers\(^\ddagger\) (see Fig. 2) and avoid edge slivers\(^\dagger\dagger\) or splitting critical features in general.

### 3. METHODS TO ASSESS FRACTURE QUALITY

#### 3.1 Requirements for Fracture Quality Metrics

To assess fracture quality in an automated fashion the metric applied needs to fulfill the following requirements:

- Due to their different impact on CD control the metric needs to reflect the different severity of slivers due to their different impact on CD control. Specifically embedded slivers should be favored over edge slivers.
- To be able to do benchmarking between different MDP algorithms or polygon division strategies the definition and implementation of the fracture quality metric needs to be independent from the MDP tool used to generate the VSB data.
- As this metric is targeted towards a use in a production environment the implementation of the sliver metric needs to be production compatible in terms of runtime and resource requirements. The runtime required for the fracture quality analysis should be significantly lower than the initial VSB data conversion time.

In the following we discuss two possible fracture quality metrics and judge them against these requirements.

#### 3.2 Sliver Partitioning

Due to the heuristic methods used to speed up the polygon fracturing process slivers may be introduced into the polygon division that are not required due to the design. To detect these situations DRC ruledecks have been implemented.

\(^\dagger\) Systematic errors induced by the shot size linearity could in principle be compensated by appropriate modifications of the exposure process such as size dependent dose control. Not all available VSB mask lithography tools offer this kind of compensation methodologies.

\(^\ddagger\) Embedded slivers are small figures that are stacked between other figures and thus share most of their periphery with other figures. Only a small amount of their periphery is coincident with the edges of the original design polygon. Usually they are placed perpendicular to the polygon edge and thus have only a small impact on CD control.

\(^\dagger\dagger\) Edge slivers are figures the smaller dimension of which is below the sliver threshold while one of their longer edges is coincident with the edge of the original design polygon.
These ruledecks are used to analyze the fractured data for

1. series of jogs,
2. offset jogs on the same geometry,
3. angled jogs,
4. jogs near corners and
5. jogs across and offset from corners.

These design situations will lead to unavoidable slivers when the length of the jogs in question is below the sliver threshold. If a sliver present in the fractured data interacts with any of these unavoidable sliver sites, then that sliver is deemed to be unavoidable. If it does not interact with any of these sites, it is deemed avoidable or optional. We are using the term “optional sliver” for this case because locally other polygon divisions exist that would avoid these slivers. In addition to the partitioning of slivers into unavoidable and optional slivers DRC rules are being applied to detect if a sliver is an embedded or if it is an edge sliver. We achieved runtimes for this analysis method between 0.7h and 3h on 4 CPUs for 90nm single chip data. The metrics for this fracture quality analysis are the number of optional slivers and the number of edge slivers. Fig. 2 shows an exemplary fractured dataset with the detection of the four possible sliver classes.

The fracture quality detect by applying the sliver partitioning metric to three fractured datasets of different device technology is shown in Fig. 3. It is evident that the fracture algorithm used achieves significantly different quality in each of the cases. For the logic gate layer almost all slivers are due to the design and thus cannot be avoided. An improvement of the fracturing algorithm for this case would be to introduce embedded slivers for those design situations that were resolved with edge slivers.

In contrast to that the fracturing algorithm seems to be unsuitable for DRAM gate layers. The majority of slivers has been introduced by the fracturing algorithm and is not required by layout situations. In addition to that both the optional as well as the unavoidable slivers have predominantly been introduced as edge slivers which have the biggest impact on CD control.

![Image](http://proceedings.spiedigitallibrary.org/proceedingsopiae/6349/63490Z-5)

Fig. 2: Definition of different sliver types as found in a sample dataset. Embedded slivers are stacked between two other figures and thus share most of their longer dimension with other figures. Optional slivers are introduced by the fracturing algorithm although other polygon divisions exist that could have avoided those slivers. Especially short slant edges cannot be resolved during fracturing without the creation of small figures.
3.3 Sliver Shoreline Analysis

Sliver partitioning as well as sliver count have one fundamental disadvantage as fracture quality metric for line type layers\textsuperscript{\textdegree}: Both metrics do not distinguish between (edge) slivers of different length. It is obvious that a longer sliver that may interfere with the CD critical regions for multiple devices has a bigger impact on device performance than a shorter sliver that may only impact a single device. An appropriate fracture quality metric in this case would be to count the number of CD critical regions that are affected by slivers or split features.

This type of analysis would require information about which parts of the design actually are sensitive for CD control. In a typical merchant mask manufacturing environment this information usually is not available. As consequence a fracture quality metric needs to be defined that yields a similar fracture quality assessment without the need for information about the design intent. This metric should be extendable to include design intent if this kind of information is available. To overcome this shortcoming we propose a fracture quality metric that we call “shoreline” analysis. It adopts and extends a principle proposed by Nakao et al.\textsuperscript{3} that only the exposed edges of slivers contribute to the degradation of CD control.

\textsuperscript{\textdegree} NB: Sliver count is an appropriate metric for contact type layers as in this case the number of contacts influenced by slivers and the number of slivers usually are equivalent. As commercial fracture tools offer fast methods to determine sliver count we restrict our discussion to fracture quality metrics for line type layers.
Table 1: Fracture tool comparison using sliver shoreline metric. Best results are marked with grey.

<table>
<thead>
<tr>
<th>Tool</th>
<th>DRAM metal</th>
<th>DRAM gate</th>
<th>DRAM active</th>
<th>DRAM trench</th>
<th>DRAM contact</th>
<th>MPU active</th>
<th>MPU gate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool A</td>
<td>238,022</td>
<td>208</td>
<td>17,006,284</td>
<td>39,296,639</td>
<td>0</td>
<td>17,899,081</td>
<td>5,250,183</td>
</tr>
<tr>
<td>Tool B</td>
<td>763,137</td>
<td>500</td>
<td>39,909,503</td>
<td>70,747,329</td>
<td>30,819</td>
<td>21,944,283</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool C</td>
<td>4,491</td>
<td>206</td>
<td>6,308</td>
<td>21,662,405</td>
<td>50,000</td>
<td>1,580,850</td>
<td>671,136</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tool</th>
<th>DRAM metal</th>
<th>DRAM gate</th>
<th>DRAM active</th>
<th>DRAM trench</th>
<th>DRAM contact</th>
<th>MPU active</th>
<th>MPU gate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool A</td>
<td>234,003</td>
<td>188</td>
<td>15,855,532</td>
<td>30,503,372</td>
<td>30,819</td>
<td>15,323,162</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool B</td>
<td>543,791</td>
<td>582</td>
<td>38,581,480</td>
<td>55,597,319</td>
<td>140</td>
<td>6,621,121</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool C</td>
<td>4,865</td>
<td>188</td>
<td>2,465</td>
<td>11,558,762</td>
<td>23</td>
<td>983,495</td>
<td>526,903</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tool</th>
<th>DRAM metal</th>
<th>DRAM gate</th>
<th>DRAM active</th>
<th>DRAM trench</th>
<th>DRAM contact</th>
<th>MPU active</th>
<th>MPU gate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool A</td>
<td>219,346</td>
<td>148</td>
<td>1,328,023</td>
<td>15,150,010</td>
<td>30,679</td>
<td>6,621,121</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool B</td>
<td>4,019</td>
<td>200</td>
<td>219,346</td>
<td>10,103,643</td>
<td>27</td>
<td>597,355</td>
<td>144,235</td>
</tr>
<tr>
<td>Tool C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8,793,267</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tool</th>
<th>DRAM metal</th>
<th>DRAM gate</th>
<th>DRAM active</th>
<th>DRAM trench</th>
<th>DRAM contact</th>
<th>MPU active</th>
<th>MPU gate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Tool C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Fig. 4 shows a sample polygon with three slivers and a split critical feature. By measuring the length of the edge of the polygon that is either coincident with a sliver ("sliver shoreline" as marked in Fig. 4) or with a region that is composed of multiple figures ("split CD shoreline" as marked in Fig. 4) a metric can be obtained. With this metric a better fracture quality is achieved when the sliver or split CD shoreline length are smaller§§. Under the assumption that CD critical areas for a certain design approximately have the same length the sliver shoreline length directly correlates with the number of devices influenced by slivers.

As discussed above embedded slivers are the preferred figure division for design situations that cannot be resolved without slivers. Thus a well defined sliver metric should not penalize this design situation. The potential effect of short edges (rectilinear edges with \( l \leq \varepsilon \) and 45° slant edges with \( l \leq \sqrt{2} \varepsilon \), where \( \varepsilon \) is the sliver threshold) on the CD performance can be considered in the context of defects. The effect of a sliver on the pattern fidelity is to introduce a certain edge placement error \( \Delta CD_{\text{max}} \). The size of the resulting sliver induced defect is then given by the edge length \( l \) and \( \Delta CD_{\text{max}} \). For the worst case sliver as shown in Fig. 4 the area of this defect is given by:

\[
A_{\text{DEF}} \approx 2 \cdot \varepsilon \cdot \Delta CD_{\text{max}}
\]

For a typical sliver threshold of \( \varepsilon = 100\,\text{nm} \) a defect size below \( s_D = 52\,\text{nm} \) as required by the ITRS for the 65nm technology node can be achieved if \( \Delta CD_{\text{max}} \) is below 10nm. Although experimental proof is pending it is assumed that \( \Delta CD_{\text{max}} \) is significantly lower and lies in the low single digit range. Due to the above considerations short edges are filtered in the sliver shoreline analysis. Nakao et al.\(^3\) propose a similar approach by neglecting embedded slivers in the cost analysis of different polygon divisions. The resulting metric is referenced to as "relevant" edge sliver length. The inclusion of design intent into the shoreline analysis can be done by measuring shoreline only in areas of the design that overlap with a marker layer. This layer marks those design regions that require tight CD control. We have implemented some empiric methods to extract marker layers for the DRAM core (see Table 1). In principle this kind of marker layer information could be included in the dataset handed-off to the mask house. Data flows in which single mask layer data are handed-off to the mask manufacturer would need to be modified to include this kind of design intent. In the case that stream file formats such as GDSII or OASIS are sent to the mask house the required information (such as gate over active regions) might already be present in the data and thus could be extracted e.g. via Boolean operations.

The sliver and split CD shoreline analysis was implemented using DRC rule decks. For performance reasons the final fractured data were converted back into a native format readable by the DRC engine. The runtimes achieved for the shoreline analysis method typically are approximately 10% of the required time for final fracture (including format conversion if required) and XOR verification on the same set of fracture hardware.

§§ NB: The following arguments apply both to the sliver shoreline as well as the split CD shoreline analysis method.
4. APPLICATIONS FOR SLIVER METRICS

Apart from the verification of production datasets the sliver shoreline fracture quality metric can also be applied during MDP process development and tuning. Below we discuss two possible applications.

4.1 Fracture Tool Benchmarking

Table 1 shows the result of a fracture tool comparison that was performed using the sliver and split CD shoreline metrics on different DRAM and MPU datasets. For this experiment default MDP tool parameter sets were used and no dataset specific parameter tuning was performed.

The comparison of sliver and split CD shoreline analysis shows that the fracture quality differences between the MDP tools compared can be of orders of magnitude. The restriction of the shoreline analysis to relevant edge slivers pronounces these quality differences. This pronunciation of fracture quality difference is due to the fact that datasets with better fracture quality usually contain more embedded slivers which are filtered out in the analysis. Datasets with lower fracture quality contain a higher amount of edge slivers that are not being filtered.

For certain datasets a restriction to the DRAM core was done in the analysis. While fracture tool B and C achieve comparably good results for fracture quality in the DRAM core area, tool C only achieves significantly worse fracture quality for the most critical DRAM metal and gate layers. It is to be noted that the sliver shoreline results for contact type layers yield not as big differences as for the other layers. As mentioned earlier sliver shoreline in this case is not as good of a metric as sliver count. Sliver shoreline analysis thus should be restricted to line type layers.

4.2 Fracture Tool Parameter Tuning

In general all commercial MDP tools offer parameters to influence the fracture results. Usually these parameters are meant to control the way critical features in the design are being fractured. As mentioned earlier the fracture tools need to balance figure count for write time considerations versus creation of slivers and split features. Often parameters exist to favor one of these criteria over the other.

Table 2: Result of a fracture tool parameter optimization for a logic gate layer using sliver and split CD shoreline metric.

<table>
<thead>
<tr>
<th>Parameter A</th>
<th>Parameter B (cd_max_width)</th>
<th>Parameter C (cd_min_length)</th>
<th>File size (Bytes)</th>
<th>JEOL shots</th>
<th>PEC Usage (MBytes)</th>
<th>Exposed sliver edge (µm)</th>
<th>Split feature length (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter A (small_value)</td>
<td>Parameter B (cd_max_width)</td>
<td>Parameter C (cd_min_length)</td>
<td>File size (Bytes)</td>
<td>JEOL shots</td>
<td>PEC Usage (MBytes)</td>
<td>Exposed sliver edge (µm)</td>
<td>Split feature length (µm)</td>
</tr>
<tr>
<td>0.05</td>
<td>1</td>
<td>1</td>
<td>4,071,424</td>
<td>3.08E+07</td>
<td>10.16</td>
<td>78,285.33</td>
<td>15.96</td>
</tr>
<tr>
<td>0.05</td>
<td>1</td>
<td>0.5</td>
<td>4,132,864</td>
<td>3.08E+07</td>
<td>10.24</td>
<td>76,555.52</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>0.5</td>
<td>1</td>
<td>3,788,800</td>
<td>3.10E+07</td>
<td>9.706</td>
<td>504,599.21</td>
<td>669,393.51</td>
</tr>
<tr>
<td>0.05</td>
<td>0.5</td>
<td>0.5</td>
<td>3,788,800</td>
<td>3.10E+07</td>
<td>9.706</td>
<td>504,596.20</td>
<td>669,393.51</td>
</tr>
<tr>
<td>0.1</td>
<td>1</td>
<td>0.7</td>
<td>4,009,984</td>
<td>3.08E+07</td>
<td>10.07</td>
<td>72,182.56</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.5</td>
<td>0.7</td>
<td>3,817,472</td>
<td>3.11E+07</td>
<td>9.697</td>
<td>472,132.41</td>
<td>669,868.21</td>
</tr>
<tr>
<td>0.1</td>
<td>0.7</td>
<td>1</td>
<td>3,977,216</td>
<td>3.08E+07</td>
<td>9.975</td>
<td>78,978.83</td>
<td>15.78</td>
</tr>
<tr>
<td>0.1</td>
<td>0.7</td>
<td>0.5</td>
<td>4,001,792</td>
<td>3.13E+07</td>
<td>10.02</td>
<td>79,384.54</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.7</td>
<td>0.7</td>
<td>3,993,600</td>
<td>3.13E+07</td>
<td>10.01</td>
<td>79,075.52</td>
<td></td>
</tr>
<tr>
<td>0.15</td>
<td>1</td>
<td>0.5</td>
<td>3,993,600</td>
<td>3.08E+07</td>
<td>10.03</td>
<td>71,723.66</td>
<td>13.85</td>
</tr>
<tr>
<td>0.15</td>
<td>0.5</td>
<td>1</td>
<td>4,038,656</td>
<td>3.08E+07</td>
<td>10.1</td>
<td>70,929.06</td>
<td></td>
</tr>
<tr>
<td>0.15</td>
<td>0.5</td>
<td>0.5</td>
<td>3,817,472</td>
<td>3.12E+07</td>
<td>9.675</td>
<td>475,356.79</td>
<td>676,747.73</td>
</tr>
<tr>
<td>0.15</td>
<td>0.5</td>
<td>0.5</td>
<td>3,817,472</td>
<td>3.12E+07</td>
<td>9.675</td>
<td>475,352.51</td>
<td>676,742.30</td>
</tr>
</tbody>
</table>
Two major fracture tool parameter classes exist:

1. Parameters that directly control certain aspects of the fracturing algorithm such as the favorable creation of embedded slivers. In this case tuning of this fracture tool parameter will result in a different behavior of the fracture algorithm on the whole design.

2. The second class of fracture tool parameters is designed to detect regions with certain design intent. Usually these parameters are meant to find the regions in which tight CD control is required. These regions in turn are treated differently by the fracturing algorithm than the rest of the design.

Fracture tool parameters often are either multi level switches or threshold values. Due to this the fracture parameter optimization problem is highly non-linear. A linear interpolation between sample points as done in a DOE analysis thus is not appropriate in this case. For a reduced number of parameters an optimized parameter set still can be obtained by selecting the best result out of a series of test fractures. This approach is especially suitable for the first class of fracture tool parameters.

As shown in Table 2 the sliver shoreline analysis methodology can be used to optimize fracture parameters in this kind of a setup. For this experiment three parameter values have been varied with three levels each. In Fig. 5 a detail of the post-fracture figure division is shown both for the optimum parameter set as well as for the parameter set that yields the largest edge sliver length. It can be seen that the parameter that influences the fracture quality most is parameter B (cd_max_width). If this parameter is chosen too low areas of the design that have a larger target CD are sliced into multiple shots.

The optimum value of parameter B in this case directly correlates with the designed target size of the CD critical regions. This target value thus could be provided as part of the data package transferred during the order process. Alternatively, in the case that design intent is conveyed in the dataset (e.g. via marker shapes) DRC methodologies could be applied to deduce an appropriate parameter value by determining the target size of features in the areas of the design that require tight CD control.

The “test fracture” approach to the parameter optimization problem can be problematic in terms of runtime due to the number of possible parameter combinations. To achieve acceptable run times the design could be reduced to certain representative areas. While the fracture results for those areas can be compared among each other using edge sliver length as quality metric the results obtained cannot be directly compared with the edge sliver length achieved on a full dataset. This can be overcome by scaling the edge sliver length by the total edge length of the analysis area. The resulting so called sliver shoreline percentage is invariant to changes of the design size. The comparison of the sliver shoreline percentage result obtained on a representative design part to the result obtained on the whole dataset then also gives a quantitative measure of how well the sample region was chosen.

For a very large number of fracture tool parameters the number of required sample fractures can be very large. It has been shown that as an alternative optimization strategy genetic algorithms can be used to obtain optimized parameter sets.
while minimizing the number of required test fractures. In this context sliver shoreline percentage can be used as the so-called fitness measure in such an optimization scheme.

5. DISCUSSION AND FUTURE WORK

In this paper we have proposed novel metrics to assess the quality of mask datasets for variable shaped beam (VSB) electron beam lithography. We have shown that sliver partitioning is a suitable metric to assess the fracture algorithm quality. Due to the required runtimes it is not suitable for a productive implementation though.

In contrast to that the sliver and split CD shoreline metric both is suitable for fracture algorithm quality assessment as well as the assessment of the quality of a specific fractured dataset. This metric is favorable for a productive implementation due to the required runtimes and its simplicity. Only one figure of merit, the shoreline length (or sliver shoreline percentage), can be used for fracture quality assessment. The compilation of a single figure of merit is crucial for an automated fracture quality assessment.

We have also shown that these metrics can be restricted to only those regions of the design that require tight CD control. This restriction can be done by making use of information about the design intent such as marker layers or empirical rules that define the CD critical regions.

Future work includes the integration of this fracture quality metric into a fully automated production flow. In addition to that a detailed determination of the CD impact of slivers in different design situations is necessary. This information in turn can be used to improve the fracture quality metric by improving the weighing of slivers based on their impact.

ACKNOWLEDGEMENTS

AMTC is a joint venture of Infineon, AMD and Toppan Photomasks, Inc. and gratefully acknowledges the financial support by the Free State of Saxony in the framework of the technology grants based upon European Regional Development Funds (2000-2006) and funds of the Free State of Saxony.

REFERENCES