Fracture Friendly Optical Proximity Correction

Frank Amoroso, Michel Cote, Tanya Do, Robert Lugg, John Nogatch*
Synopsys, 700 E. Middlefield Road, Mountain View, CA, USA 94043

ABSTRACT

Optical Proximity Correction (OPC) improves image fidelity by adding and subtracting small enhancement shapes from the original pattern data. Although the presence of these small shapes improves the final wafer image quality, it causes an increase in total figure count, longer fracture processing time, and the introduction of sliver figures. These undesirable artifacts can have a negative impact on the mask write time and mask image quality. In this paper we outline alternative OPC treatments which reduce the additional figures produced, and make the layout configurations friendlier to the subsequent mask fabrication phase. These include the alignment of neighboring small shapes during the OPC operation, and the preservation of jog alignment during the biasing phase. Illustrations of example pattern data, and improvement results in terms of figure counts are described.

Keywords: fracture, jogs, Optical Proximity Correction (OPC), process bias, reverse tone, slivers.

1. INTRODUCTION

As line width and spacing have become smaller, OPC treatment is required in order to achieve accurate formation of features on the wafer. By considering the cumulative effect of nearby shapes, OPC introduces many small shape adjustments "jogs", so that the total exposure in each small area is compensated for the contribution of its neighborhood. The jogs inserted by OPC increase the figure count output from the fracture phase, often by a significant amount (Figures 1 and 2). Larger figure count requires more fracture processing time, and more time to expose the mask. The presence of figures which are small in one or both dimensions is undesirable, because these "slivers" do not expose with the same fidelity as larger figures.

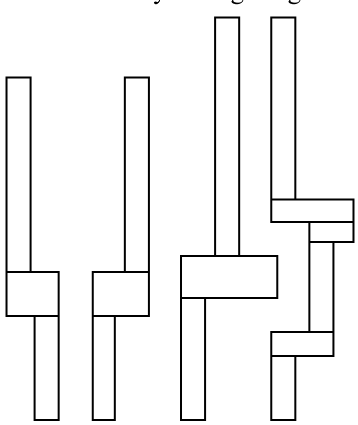

Fig. 1 Design data without OPC, after fracture to 15 primitive figures.

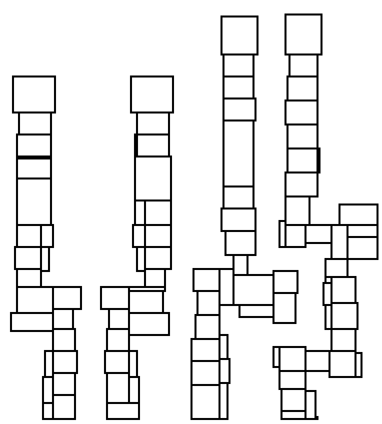

Fig. 2 Data treated by OPC and fracture, increases to 100 figures.

* Email: jnogatch@synopsys.com; phone 650-584-5000

25th Annual BACUS Symposium on Photomask Technology, edited by J. Tracy Weed, Patrick M. Martin,
Proc. of SPIE Vol. 5992, 599222, (2005) · 0277-786X/05/$15 · doi: 10.1117/12.632220

The solution obtained during the OPC calculation is not unique. A different choice of jog sizes and positions may achieve equally good exposure of the original design data. By modifying OPC recipes to prefer aligned jogs, instead of jogs placed at random, the number of slivers and the total number of figures can be reduced. The choice of jogs to be aligned should consider whether the data will subsequently be processed as normal versus reverse tone, in order to produce the least figure count after fracture.

The alignment of nearby jogs is perturbed when process bias is applied by the fracture tool. Some aligned jogs will remain aligned after process bias is applied. However, an "inside corner" paired with an "outside corner" will grow in opposite directions, resulting in two edges which are no longer aligned, likely resulting in an additional sliver figure. Unless this effect of process bias is compensated for, the benefit of aligned jogs is substantially reduced.

2. METHODOLOGY

Methods have been proposed to limit the increase in figure count which is induced by OPC treatment[1,2]. If design intent is known, then OPC treatment can be restricted to the important areas, e.g. gate regions on a polysilicon layer[3,4]. Design intent can sometimes be inferred, if important areas can be recognized by feature size[5,6]. The approach described here is not based on design intent, but rather the recognition of partial alignment in small neighborhoods, which can be enhanced by the OPC tool, and retained by the fracture tool.

2.1. Jog Alignment

Preferential alignment of OPC features has been implemented as an application-level recipe module in Proteus™. Jog positions are determined in an early phase of the model-based OPC called "dissection". The main "Fracture Friendly" alignment step takes place after dissection, but before the correction phase, when the pattern data is actually altered. Figure 3 illustrates two of the many possible situations where jog alignment may be considered. Additionally, the alignment process may need to compensate for a subsequent mask bias, as described in section 2.3.

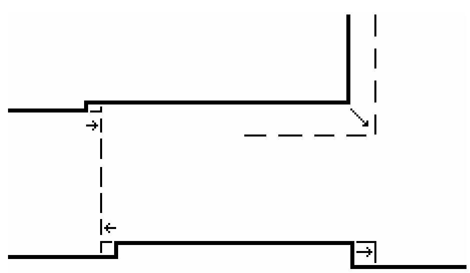

Fig. 3 Jog-to-jog and corner-to-jog alignment

Alignment is controlled by several parameters. The maximum search width for alignment can be specified. Jogs will not be aligned across shapes wider than this limit. A larger search width may increase the number of alignments, but will also increase runtime. A maximum allowable movement places a limit on how far each jog may be moved. This range can be overridden by a minimum distance between jogs, usually determined by Mask Rules Check (MRC) constraints.

Jog alignment can be done after each iteration of correction, or just once, at the end. For the experimental data, the increase in OPC runtime was between 5% and 15%.

Figure 4 shows a small region of poly data, which was given OPC treatment without consideration of jog alignment, and then fractured. With the OPC recipe modified to prefer aligned jogs, the subsequent fracture contains fewer figures (Figure 5). Note that the number of jogs is identical, but the likelihood of a jog causing an additional figure is reduced.

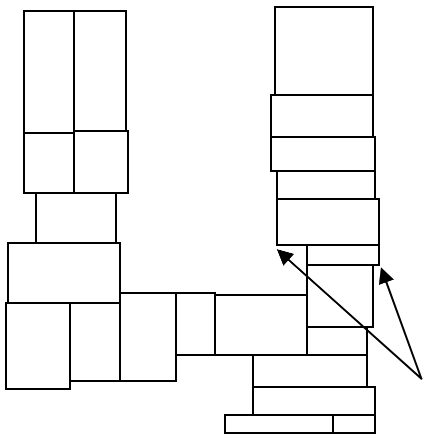

Fig. 4 OPC without jog alignment,
fracture produces 23 figures.

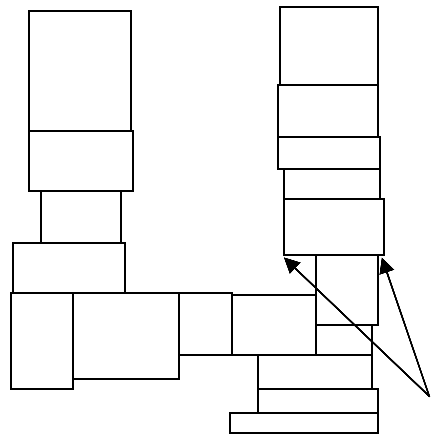

Fig 5 OPC with jog alignment,
fracture produces 18 figures.

Figures 6 and 7 show a larger neighborhood, including the region from the previous figures. Note that the division of areas into primitive figures is complex, and that small changes such as jog alignment often precipitate additional changes in the connected neighborhood.

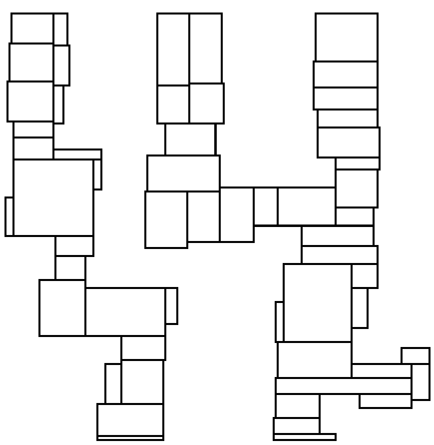

Fig. 6 OPC without jog alignment,
fracture produces 56 figures.

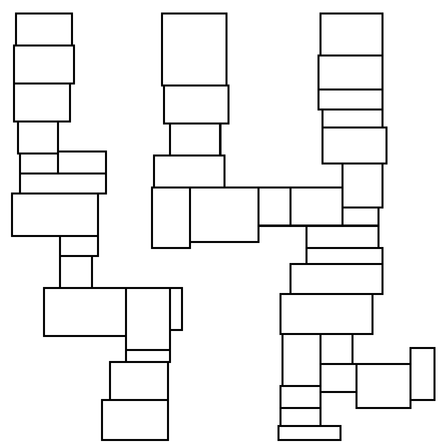

Fig. 7 OPC with jog alignment,
fracture produces 43 figures.

2.2. Reverse Tone

For normal tone data, the jog-to-jog alignment considers opposing interior sides of a shape. To reduce figures on reverse tone data, the jog-to-jog alignment instead needs to consider opposing external sides of nearby shapes. Figure 8 shows the result of normal tone jog alignment, followed by reverse tone fracture. When the jog alignment is aware of reverse tone, an improved result is obtained, as shown in Figure 9.

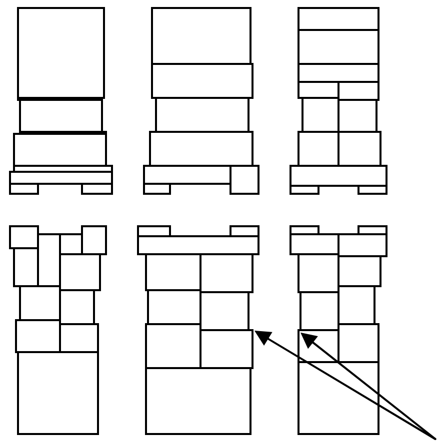

Fig. 8 normal tone OPC jog alignment, followed by reverse tone fracture fracture produces 60 figures.

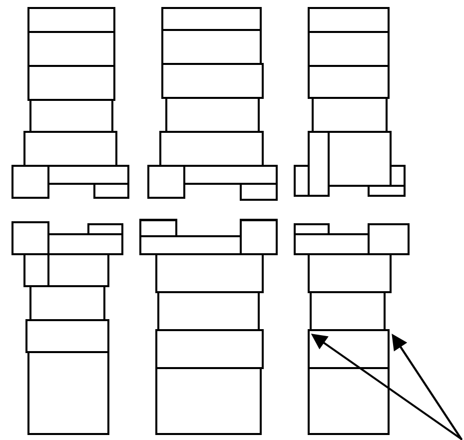

Fig. 9 reverse tone OPC jog alignment, followed by reverse tone fracture fracture produces 47 figures.

2.3. Process Bias

Adjustment for process bias is often applied to data during the fracture operation. The edge positions of the data are moved, in order to compensate for distortions which occur during wafer processing, e.g. over-etching. Note that this bias or "sizing" operation is not the same as magnification or scaling. Bias has the unfortunate effect of making some jogs become non-aligned, even though they were aligned before the bias was applied. Figure 10 shows 4 simple situations where 2 abutting rectangles result in either 2 or 3 rectangles after bias is added. In these diagrams, shaded areas are the figures before bias, and the larger, unshaded areas are the output figures, after positive bias. Of the 4 cases, the one in the lower right demonstrates that small, opposing jogs, which were aligned before bias is applied, become unaligned, and result in one additional figure.

2.3.1. OPC Compensation for Bias

To account for a mask bias which is not aware of OPC jogs, the alignment algorithm maintains a record of jog-to-jog and jog-to-corner pairings. Each pair of jogs is examined to determine whether or not they will be misaligned by the process bias. If they will be misaligned, both jogs are moved away by the bias amount. For a corner-to-jog alignment, the jog receives the entire compensation, moving by twice the bias amount. This adjustment should take place in OPC after all correction has been completed. At the user's discretion, this process may or may not be constrained by the MRC limits.

Though bias compensation is possible in OPC, it is inconvenient for most mask data preparation flows to do this. The process bias amount is often not known at the time of the OPC computation. In some cases, the output of the OPC tool may be subsequently fractured for exposure by two different mask making machines, each requiring a different bias. In other cases, the bias amount may be changed periodically, sometimes on a daily basis, as part of machine calibration. Different teams within the organization may be separately responsible for applying OPC and process bias. For these reasons, it is more appropriate that the bias be added by the fracture tool, and that jog alignment be preserved by the fracture tool, without rerunning OPC.

2.3.2. Jog Sensitive Bias

Given input data containing jogs that were preferentially aligned during the OPC phase, an enhancement to the CATS® fracture tool was undertaken, in order to preserve this alignment during the bias operation. A numeric parameter "JOGSIZE" specifies the maximum edge length to be classified as an OPC induced jog. These edges are biased, or not biased, as appropriate, in order to maintain alignment. Any edges that are longer than the specified limit receive normal bias treatment. Figure 10 shows conventional sizing of simple situations containing aligned jogs. Figure 11 shows the effect of using Jog Sensitive Bias on the same data.

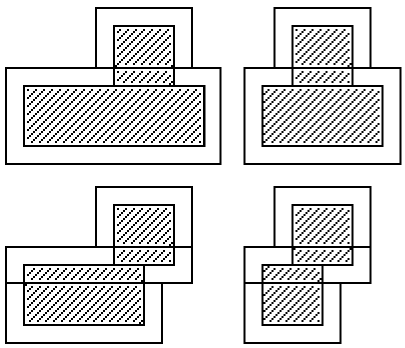

Fig. 10 Conventional bias treatment during fracture produces 10 figures.

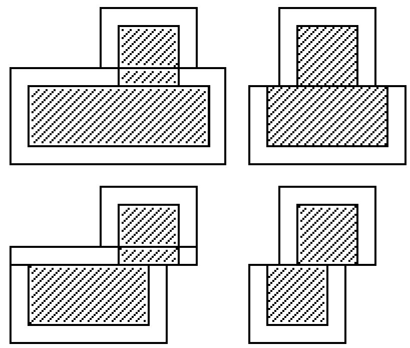

Fig. 11 Jog Sensitive Bias treatment during fracture produces 9 figures.

Of the 4 situations shown in Figures 10 and 11, the lower right case is the one where Jog Sensitive Bias results in fewer numbers of shapes. The lower left case will always result in an additional figure, regardless of whether the jog is biased or not. The case at upper left is best handled by biasing the jog, since it would otherwise require an additional figure.

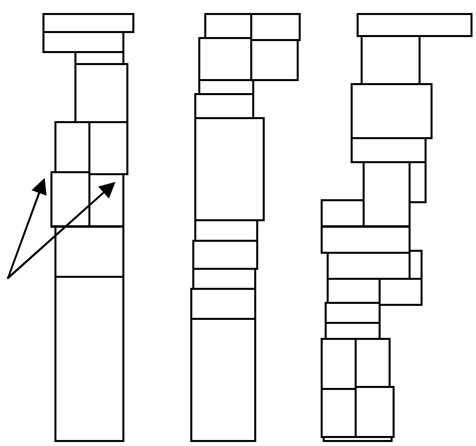

Fig. 12 OPC jog alignment disturbed by process bias applied during fracture (41 figures).

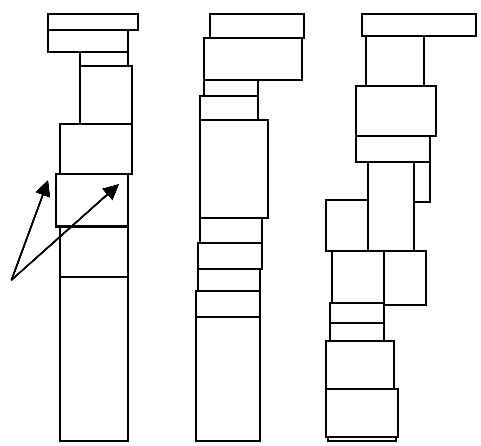

Fig. 13 OPC jog alignment preserved by Jog Sensitive Bias during fracture (32 figures).

In the current implementation, Jog Sensitive Bias operates only on Manhattan edges, in both oversize and undersize modes, and for normal and reverse tones. Non-Manhattan edges are not tested for edge length, so their bias treatment is not affected by the JOGSIZE parameter. Figures 12 and 13 illustrate the reduction in fracture figures that is achieved by setting JOGSIZE.

3. RESULTS

Figure 14 displays the results of five test cases which were processed by the new Fracture Friendly OPC and Jog Sensitive Bias treatment, as compared to the same data processed with the conventional OPC and bias. The five data sets were also processed with reverse tone and negative bias, yielding 10 data points. Normal tone cases are depicted as diamonds, and reverse tone cases as squares.

The vertical axis is the normalized figure count, i.e. the new figure count divided by the old figure count; a value of 1 represents no change in figure count, and values less than 1 indicate a desirable reduction in the number of figures. 8 of the 10 cases show a desirable reduction in figure count, which yields a reduction in the time required to write the mask.

The horizontal axis is the normalized fracture time, i.e. the new execution time divided by the old time, e.g. a value of 2 indicates that the new fracture time is twice as long as the old fracture time. OPC runtime is not included.

9 of the 10 cases were between 1X and 2.2X the execution time of the old time. One of the cases took 3.5X longer than its old time. Although the increase in fracture time is undesirable, it can be overcome by adding additional computer resources. This additional effort is worthwhile, since it reduces the mask writer processing time, which is not amenable to parallelization.

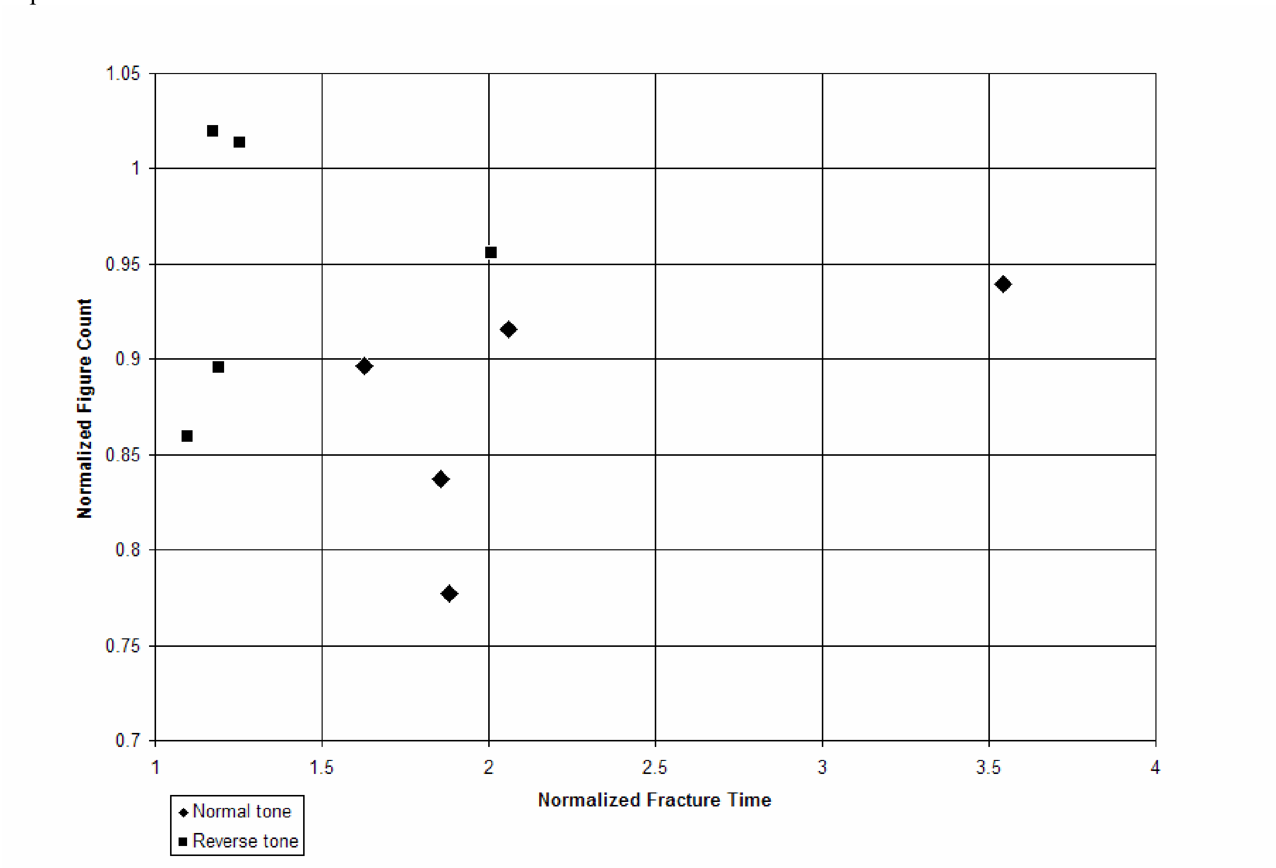

Fig. 14 Change in figure count (normalized) versus change in fracture time (normalized).

Reverse tone data did not show as much improvement as normal tone data. Two of the reverse tone cases show an undesirable increase of 2%. The distances between aligned jogs are greater for reverse tone, as compared to normal tone, which leads to less chance that jog alignment will result in an opportunistic decomposition into primitive figures.

4. CONCLUSIONS

Optical Proximity Correction is a necessity for achieving good results at advanced technology nodes, but it causes increases in total figure count, sliver count, fracture processing time, and mask writing time. These undesirable effects can be partially mitigated by making additional alignment constraints part of the OPC recipe. This modification is termed “Fracture Friendly OPC”.

The intended tone of the pattern data, i.e. normal or reverse, should be included in the OPC recipe, to maximize feature alignment during the subsequent fracture phase.

Although process bias can be included as one of the additional constraints during the OPC calculation, the bias amount is not often known at this point in the MDP flow. An enhancement to the fracture software, termed "Jog Sensitive Bias”, allows the benefit of Fracture Friendly OPC aligned features to be retained.

The combination of Fracture Friendly OPC and Jog Sensitive Bias fracture achieves noticeable improvement, on the order of 10% reduction in figure count. The amount of improvement depends on the nature of the original design data. These techniques can be applied to any data, without requiring design intent information.

REFERENCES

1. Michael L. Rieger, Jeffrey P. Mayhew, Jiangwei Li, James Shiely,"OPC Strategies to Minimize Mask Cost and Writing Time", SPIE Proc. Vol. 4562, pp. 154-160, SPIE BACUS 2001.

2. Lawrence S. Melvin III, Michael L. Rieger, "A Model-Based Methodology for Reducing OPC Output Pattern Complexity", SPIE Proc. Vol. 5256, pp. 238-245, SPIE BACUS 2003.

3. Michael L. Rieger, Jeffrey P. Mayhew, Lawrence Melvin, Robert Lugg, and Daniel Beale, "Anticipating and Controlling Mask Costs Within EDA Physical Design", SPIE Proc. Vol. 5130, pp. 617-627, SPIE PMJ 2003.

4. James Word, Steffen Schulze, "Optimization of Data Handling Prior to Fracture for Reduction of Mask Writing Time", SPIE Proc. Vol. 5256, pp. 766-776, SPIE BACUS 2003.

5. Lars W. Liebmann, "Vertex minimization in a smart optical proximity correction system", United States Patent 5553273, September 3, 1996

6. Walter Iandolo, Yitzik Gilboa, Bill Phan, and Artur Balasinski, "Optimization of Dummy Pattern for Mask Data Size Reduction", SPIE Proc. Vol. 5256, pp. 798-805, SPIE BACUS 2003.