Closing the Smoothness and Uniformity Gap in Area Fill Synthesis *

Yu Chen
CS Department
UCLA
Los Angeles, CA 90095
yuchen@cs.ucla.edu

Andrew B. Kahng
CSE and ECE
Departments, UCSD
La Jolla, CA 92093
abk@ucsd.edu

Gabriel Robins
CS Department
University of Virginia
Charlottesville, VA 22903
robins@cs.virginia.edu

Alexander Zelikovsky
CS Department
Georgia State Univ.
Atlanta, GA 30303
alexz@cs.gsu.edu

ABSTRACT

Control of variability in the back end of the line, and hence in interconnect performance as well, has become extremely difficult with the introduction of new materials such as copper and low-k dielectrics. Uniformity of chemical-mechanical planarization (CMP) requires the addition of area fill geometries into the layout, in order to *smoothen* the variation of feature densities across the die. Our work addresses the following *smoothness gap* in the recent literature on area fill synthesis. (1) The very first paper on the filling problem (Kahng et al., ISPD98 [7]) noted that there is potentially a large difference between the optimum window densities in fixed dissections vs. when all possible windows in the layout are considered. (2) Despite this observation, all filling methods since 1998 minimize and evaluate density variation only with respect to a *fixed dissection*. This paper gives the first evaluation of existing filling algorithms with respect to "gridless" ("floating-window") mode, according to both the *effective* and *spatial* density models. Our experiments indicate surprising advantages of Monte-Carlo and greedy strategies over "optimal" linear programming (LP) based methods. Second, we suggest new, more relevant methods of measuring a *local uniformity* based on Lipschitz conditions, and empirically demonstrate that Monte-Carlo methods are inherently better than LP with respect to the new criteria. Finally, we propose new LP-based filling methods that are directly driven by the new criteria, and show that these methods indeed help close the "smoothness gap".

Categories and Subject Descriptors

B.7.2 [**Hardware**]: IC— *Manufacturability*; J.6 [**Computer Applications**]: CAD; F.2.2 [**Analysis of Algorithms**]: Problem Complexity

* This research was supported by a grant from Cadence Design Systems, Inc., by the MARCO/DARPA Gigascale Silicon Research Center, by a Packard Foundation Fellowship, by a National Science Foundation Young Investigator Award (MIP-9457412) by NSF grant CCR-9988331 and by the State of Georgia's Yamacraw Initiative.

ISPD'02, April 7-10, 2002, San Diego, California, USA.

General Terms

Algorithms, Design, Reliability, Theory

Keywords

VLSI Manufacturability, Chemical-Mechanical Polishing, Dummy Fill Problem, Density Analysis, Monte-Carlo

1. INTRODUCTION

Chemical-mechanical planarization (CMP) and other manufacturing steps in nanometer-scale VLSI processes have varying effects on device and interconnect features, depending on the local characteristics of the layout. To improve manufacturability and performance predictability, foundry rules require that a layout be made uniform with respect to prescribed density criteria, through insertion of *area fill* (dummy fill) geometries.

All existing methods for synthesis of area fill are based on discretization: the layout is partitioned into *tiles*, and filling constraints or objectives (e.g., minimizing the maximum density variation) are enforced for square *windows* that each consists of $r \times r$ tiles. Thus, to practically control layout density in *arbitrary* windows, density bounds are enforced in only a *finite* set of windows. More precisely, both foundry rules and EDA physical verification and layout tools attempt to enforce density bounds within r^2 overlapping *fixed dissections*, where r determines the "phase shift" w/r by which the dissections are offset from each other. The resulting *fixed r-dissection* (see Figure 1) partitions the $n \times n$ layout into tiles T_{ij}, then covers the layout by $w \times w$-windows W_{ij}, $i, j = 1, \dots, \frac{nr}{w} - 1$, such that each window W_{ij} consists of r^2 tiles T_{kl}, $k = i, \dots, i + r - 1$, $l = j, \dots, j + r - 1$.

Two main filling objectives are considered in the recent literature:

- *(Min-Var Objective)* the *variation* in window density (i.e., maximum window density minus minimum window density) is minimized while the window density does not exceed the given upper bound U;
- *(Min-Fill Objective)* the number of inserted fill geometries is minimized while the density of any window remains in the given range (L, U).

Recent methods on area fill synthesis also focused exclusively on the fixed-dissection context, including:

- Linear Programming (*LP*) methods based on rounding relaxation of the corresponding integer linear program formulations. The LP formulations for filling were first proposed by

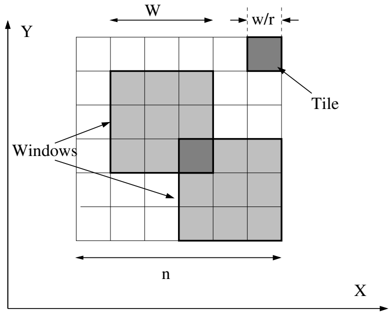

Figure 1: In the fixed r-dissection framework, the n-by-n layout is partitioned by r^2 (here, $r = 3$) distinct overlapping dissections with window size $w \times w$. The layout is partitioned into $\frac{nr}{w} \times \frac{nr}{w}$ *tiles*. Each dark-bordered $w \times w$ window consists of r^2 tiles.

Kahng et al. in [6] and adapted to other objectives and CMP models in [12, 13]);

- Greedy methods which iteratively find the best tile for the next filling geometry to be added into the layout. These methods were first used in [3] for ILD thickness control, and also used for shallow-trench isolation (STI) CMP model in [13]);
- Monte-Carlo (*MC*) methods, which are similar to greedy methods but insert the next filling geometry randomly. Due to its efficiency and accuracy, these were used for both flat [3, 4] and hierarchical [2] layout density control; and
- Iterated Greedy (*IGreedy*) and Iterated Monte-Carlo (*IMC*) methods, which improve the solution quality by iterating the insertions and deletions of dummy fill features with respect to the density variation ([3]).

The motivation for our present work is a "smoothness gap" in the fill literature. All existing filling methods fail to consider the potentially large difference between extremal densities in fixed-dissection windows and extremal densities when all possible windows are considered. On the other hand, the very first paper in the fill synthesis literature (Kahng et al., ISPD98 [7]) already pointed out the gap between fixed-dissection and "gridless" analyses, for both tile density and window density.[1] The potential consequence of the smoothness gap is that the fill result will not satisfy either given upper bounds on post-fill window density, or given bounds on density variation between windows. As post-CMP variation for oxide ILD polishing is essentially monotone in window density variation [11], this smoothness gap can compromise manufacturability of the layout, particularly given the small values of r in recent design rules.

We first address the discretization gap in existing analyses (i.e., evaluations) methods. Previous works compare density control methods only with respect to a given fixed grid, which underestimates the actual "gridless" density variation, but has been justified on grounds that gridless analysis is impractical. In this paper, we show for the first time the viability of gridless or *floating window* analyses, originally developed for the spatial density model [6], and extend it for the more accurate effective density model [9]. Second, previous research in layout density control concentrated on the *global* uniformity achieved by minimizing the window density variation over the entire layout. However, the density variation between locations which are far from each other is actually not so critical, in that the pressure/speed of the polishing pad can be (self-)adjusted during CMP. Thus, we propose and analyze criteria for "local uniformity" as a measure of smoothness in filling solutions. We evaluate existing methods with respect to the new criteria, and we suggest LP-based methods that directly optimize filling solutions with respect to smoothness.

The rest of the paper is organized as follows. In Section 2 we show how to apply floating window density analysis methods (such as extremal-density window and multilevel density analyses) to spatial and effective density models. We then give the first "gridless" evaluation of existing filling algorithms, under the *effective* as well as *spatial* density models. Our experiments indicate surprising advantages of Monte-Carlo and greedy methods over "optimal" linear programming (LP) based methods. In Section 3 we introduce new Lipschitz-like measures for layout smoothness and describe new LP-based filling methods driven by such measures. We also compare the results of existing and new filling approaches, with respect to the new smoothness criteria. Section 4 concludes with directions for future work.

2. LAYOUT DENSITY ANALYSES

As noted above, for the sake of tractability, previous works have used fixed dissections to decide the amount and positions of dummy fill features[6]. A smoothness gap thus exists because a filling solution based on a fixed-dissection does not address the true post-filling density variation. Here we first summarize the two main density models used in the current literature. We then introduce two extremal-density analysis algorithms (for spatial and effective density, respectively) which we use to compute post-fill layout density. Finally, we evaluate existing filling methods according to (near-)gridless density variation.

2.1 Density Models for Oxide CMP

We focus on layout density control for (oxide) interlevel dielectric CMP.[2] Several models have been proposed in [8], including the model of [10], where the interlevel dielectric thickness z at location (x, y) is calculated as:

$$z = \begin{cases} z_0 - \left(\frac{K_i t}{\rho(x,y)}\right) & t < (\rho_0 z_1)/K_i \\ z_0 - z_1 - K_i t + \rho_0(x,y) z_1 & t > (\rho_0 z_1)/K_i \end{cases} \quad (1)$$

The crucial element of this model is the determination of the effective initial pattern density, $\rho(x, y)$. The simplest model for $\rho(x, y)$ is the local areal feature density, i.e., the window density is simply equal to the sum:

$$\rho(W_{ij}) = \sum_{k=i}^{i+r-1} \sum_{l=j}^{j+r-1} area(T_{kl}) \quad (2)$$

where $area(T_{kl})$ denotes the original layout area of the tile T_{kl}. This *spatial density* model is due to [6], which solved the resulting filling problem using linear programming.

A more accurate model considers the deformation of the polishing pad during the CMP process [5]: effective local density $\rho(x, y)$

[1] Bounding the spatial density in a fixed set of $w \times w$ windows can incur substantial error, since other windows may still violate the density bounds [6].

[2] Several recent works, particularly by Wong et al., have studied alternative arenas for dummy fill, including shallow-trench isolation and dual-damascene copper. For such arenas, density calculations and physical polish mechanisms are different from those in the oxide context. Consideration of these alternate models is orthogonal to our contribution; certainly, the concept of a "smoothness gap" applies to all filling contexts.

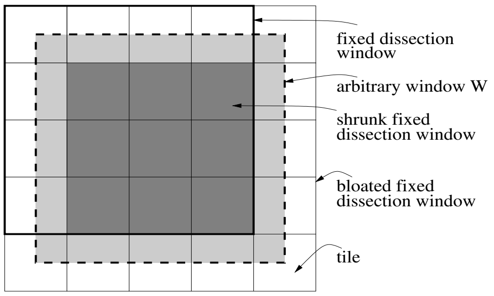

Figure 2: An arbitrary floating $w \times w$-window W always contains a *shrunk* $(r-1) \times (r-1)$-window of a fixed r-dissection, and is always covered by a *bloated* $(r+1) \times (r+1)$-window of the fixed r-dissection. A *standard* $r \times r$ fixed-dissection window is shown with thick border. A floating window is shown in light gray. The white window is the *bloated* fixed-dissection window, and the dark gray window is the *shrunk* fixed-dissection window.

Multi-Level Density Analysis Algorithm

Input: $n \times n$ layout and accuracy $\epsilon > 0$
Output: maximum density of $w \times w$ window with accuracy ϵ

(1) Make a list *ActiveTiles* of all $w/r \times w/r$-tiles
(2) *Accuracy* $= \infty$, $r = 1$
(3) **While** *Accuracy* $> 1 + 2\epsilon$ **do**
 (a) Find all rectangles in $w/r \times w/r$-tiles from *ActiveTiles*
 (b) Find area of each window consisting of tiles from *ActiveTiles*, add such window to the list *WINDOWS*
 (c) *Max* = maximum area of standard window with tiles from *ActiveTiles*
 (d) *BloatMax* = maximum area of bloated window with tiles from *ActiveTiles*
 (e) **For** each tile T from *ActiveTiles* which do not belong to any bloated window of area more than *Max* **do**
  if *Accuracy* $> 1 + \epsilon$, then put T in *TILES*
  remove T from *ActiveTiles*
 (f) Replace in *ActiveTiles* each tile with four of its subtiles
 (g) *Accuracy* = *BloatMax*/*Max*, $r = 2r$
(4) Move all tiles from *ActiveTiles* to *TILES*
(5) **Output** max window density = *(Max +BloatMax)*$/(2 \cdot w^2)$

Figure 3: Multi-level density analysis algorithm.

is calculated as the sum of *weighted* spatial pattern densities within the window, relative to an elliptical weighting function:

$$f(x, y) = c_0 \exp[c_1(x^2 + y^2)^{c_2}] \quad (3)$$

with experimentally determined constants c_0, c_1, and c_2 [12]. The *discretized* effective local pattern density ρ for a window W_{ij} in the fixed-dissection regime (henceforth referred to as *effective density*) is:

$$\rho(W_{ij}) = \sum_{k=i}^{i+r-1} \sum_{l=j}^{j+r-1} area(T_{kl}) \cdot f(k - (i + \frac{r}{2}), l - (j + \frac{r}{2})) \quad (4)$$

where the arguments of the elliptical weighing function f are the x- and y-distances of the tile T_{kl} from the center of the window W_{ij}.

2.2 Window Density Analyses

The authors of [6] proposed optimal extremal-density (i.e., minimum or maximum window density in the layout) analysis algorithms. Their ALG1, with complexity $O(k^2)$ (k is the number of rectangles in layout), is proposed as a means of checking the gridless post-filling density variation. However, with a large number of original and dummy fill features, this algorithm may be infeasible in practice.

Another method of [6] overcomes the intractability of optimal extremal-density analysis, based on the following fact (see Fig. 2).

LEMMA 1. *Given a fixed r-dissection, any arbitrary $w \times w$ window will contain some* shrunk $w(1 - 1/r) \times w(1 - 1/r)$ *window of the fixed r-dissection, and will be contained in some* bloated $w(1 + 1/r) \times w(1 + 1/r)$ *window of the fixed r-dissection.* □

The authors of [6] implemented the above Lemma within a *multi-level* density analysis algorithm (see Fig. 3). Here $\epsilon > 0$ is used to denote the required user-defined accuracy in finding the maximum window density. The lists *TILES* and *WINDOWS* are byproducts of the analysis. Since any floating $w \times w$-window W is contained in some bloated window, the filled area in W ranges between *Max* (maximum $w \times w$-window filled area found so far) and *BloatMax* (maximum bloated window filled area found so far). The algorithm terminates when the relative gap between *Max* and *BloatMax* is at most $2 \cdot \epsilon$, and then outputs the middle of the range (*Max*,*BloatMax*).

We use this algorithm (with accuracy = 1.5%) throughout this paper to achieve an accurate, efficient post-filling density analysis.[3] To handle the *effective* density model, the multi-level density analysis based on bloated and shrunk windows must be refined somewhat. To obtain more accurate results, the multi-level density analysis algorithm divides the r-dissection into smaller grids, so that more windows will be considered. With the effective density model, the discretized formulation (effective) shows that effective local pattern density is dependent on the window size w and the r-dissection. That is, we have to consider the effect on the formulation of the further division of layout during post-filling density analysis. We assume here that the effective local pattern density is still calculated with the value of r-dissection used in the filling process. The only difference is that the windows phase-shift will be smaller. For example, in Figure 4(a) we calculate the effective density of the window shown in light gray by considering 5×5 tiles (also called "cells") during the filling process. In Figure 4(b) the layout is further partitioned by a factor of 4. The effective density of the light gray window will be still calculated with the 5×5 "cells". Here each "cell" has the same dimension as a tile in the filling process and consists of 2×2 smaller tiles. More windows (e.g., the window with thick border) with smaller phase-shifts will be considered in the more gridded layout.

2.3 Accurate Analysis of Existing Methods

Here we compare the performance of existing fill synthesis methods, using the accurate multilevel floating-window density analysis. All experiments are performed using part of a metal layer extracted from an industry standard-cell layout[4] (Table 2.3). Benchmark L1 is the M2 layer from an 8,131-cell design and benchmark L2 is the M3 layer from a 20,577-cell layout.

Table 2 shows that underestimation of the window density variation as well as violation of the maximum window density in fixed-

[3]For the test cases used in this paper, the runtimes of the multi-level analysis with $accuracy = 1.5\%$ appear reasonable. Our (unoptimized) implementation has the following runtimes for Min-Var LP solutions and the spatial density model: L1/32 (45 sec), L1/16 (183 sec), L2/28 (99 sec), L2/14 (390 sec), For the effective density model, the runtimes are: L1/32 (49 sec), L1/16 (194 sec), L2/28 (109 sec), L2/14 (416 sec).

[4]Our experimental testbed integrates GDSII Stream input, conversion to CIF format, and internally-developed geometric processing engines, coded in C++ under Solaris. We use CPLEX version 7.0 as the linear programming solver. All runtimes are CPU seconds on a 300 MHz Sun Ultra-10 with 1GB of RAM.

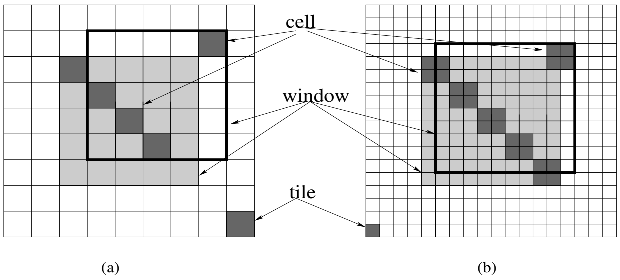

Figure 4: Post-filling density analysis for the effective density model. (a): a fixed-dissection, where each window consists of 5×5 cells (the same size as tiles); (b): a fixed-dissection for post-filling density analysis, where each window consists of 10×10 smaller tiles and each cell consists of 2×2 tiles.

dissection filling can be severe: e.g., for the LP method applied to the case L2/28/4 for the spatial (resp. effective) density model, the density variation is underestimated by 210% (resp. 264%) and the maximum density is violated by 21% (resp. 15%). Even for the finest grid (L2/28/16), the LP method may still yield considerable error: 11% (resp. 23%) in density variation and 1.2% (resp. 3.2%) in maximum density violation. Note that the LP method cannot easily handle a finer grid since the runtime is proportional to r^6.

Our comparisons show that the winning method is IMC and the runner-up is IGreedy. IMC and IGreedy can be run for much finer grids since its runtime is proportional to r^2 (resp. $r^2 \log r$). Although for L2/28/16 errors in density variation and the maximum density violation are similar, the iterative methods become considerably more accurate.

test case	L1	L2
layout size n	125,000	112,000
# rectangles k	49,506	76,423

Table 1: Parameters of four industry test cases. Here 40 units are equivalent to 1 micron.

3. LOCAL DENSITY VARIATION

The main objective of layout filling is to improve CMP and increase yield. Traditionally, layout uniformity has been measured by global spatial or effective density variation over all windows. Such a measure does not take in account that the polishing pad during CMP can change (adjust) the pressure and rotation speed according to the pattern distribution (see [11]). Boning et al. [1] further point out that while the effective density model is excellent for local CMP effect prediction, it fails to take into account global step heights. The influence of density variation between far-apart regions can be reduced by a mechanism of pressure adjustment, which leads to the *contact wear model* proposed in [1]. Within each local region, the area fill can be used to improve the CMP performance. Therefore, density variation between two windows in opposite corners of the layout will not cause problems because of the polishing dynamics. According to the integrated contact wear and effective density model, only a significant density variation between neighboring windows will complicate polishing pad control and may cause either dishing or underpolishing. Thus, it is more important to measure density variation between *neighboring* windows.

3.1 Lipschitz Measures of Smoothness

Depending on the CMP process and the polishing pad movement relative to the wafer, we may consider different window "neighborhoods". Below we propose three relevant Lipschitz-like definitions of local density variation which differ only in what windows are considered to be neighbors.

- **Type I:** The maximum density variation of every r neighboring windows in each row of the fixed-dissection. The intuition here is that the polishing pad is moving along window rows and therefore only overlapping windows in the same row define a neighborhood.
- **Type II:** The maximum density variation of every cluster of windows which cover one tile. The idea here is that the polishing pad can touch *all* overlapping windows almost simultaneously.
- **Type III:** The maximum density variation of every cluster of windows which cover one square consisting of $r/2 \times r/2$ tiles. The difference between this and the previous definition is the assumption that the polishing pad is moving slowly; if windows overlap but are still too far from each other, then we can disregard their mutual influence.

We compared the behaviors of existing filling methods with respect to these Lipschitz characteristics. The results in Table 3 show that there is a gap between the traditional Min-Var objective and the new "smoothness" objectives: the solution with the best Min-Var objective value does not always have the best value in terms of "smoothness" objectives. For the spatial (resp. effective) density model, though LP yields the best result for the case L2/28/4 with Min-Var objective with fixed-dissection model, it can not obtain the best result with respect to Lipschitz type-I variation. Thus, "smoothness" objectives may be considered separately for the filling process. We also notice that Monte-Carlo methods can achieve better solutions than LP with respect to the "smoothness" objectives (note that although LP is "optimal", it suffers from rounding and discreteness issues when converting the LP solution to an actual filling solution).

3.2 Smoothness Objectives for Filling

Obviously, all Lipschitz conditions are linear and can be implemented as linear programming formulations. We describe four linear programming formulations for the "smoothness" objectives with respect to the spatial density model. (The linear programming formulations for the effective density model are similar.)

The first Linear Programming formulation for the Min-Lip-I objective is:

Minimize: L
Subject to:

$$p_{ij} \geq 0 \quad i,j = 0,\ldots,\frac{nr}{w}-1 \tag{5}$$

$$p_{ij} \leq slack(T_{ij}) \quad i,j = 0,\ldots,\frac{nr}{w}-1 \tag{6}$$

$$\sum_{s=i}^{i+r-1}\sum_{t=j}^{j+r-1} p_{st} \leq \alpha_{ij}\left(U \cdot w^2 - area_{ij}\right) i,j = 0,\ldots,\frac{nr}{w}-1 \tag{7}$$

$$W_{ij} - W_{ik} \leq L, \;\; i,j,k = 0,\ldots,\frac{nr}{w}-1 \tag{8}$$

where $\alpha_{ij} = 0$ if $area_{ij} > U \cdot w^2$, and = 1 otherwise, and

$$W_{ij} = \sum_{s=i}^{i+r-1}\sum_{t=j}^{j+r-1} area(T_{st}) + \sum_{s=i}^{i+r-1}\sum_{t=j}^{j+r-1} p_{st}$$

Test case	OrgDen		LP			Greedy			MC			IGreedy			IMC		
			FD	Multi-Level		FD	Multi-Level		FD	Multi-Level		FD	Multi-Level		FD	Multi-Level	
T/W/r	MaxD	MinD	DenV	MaxD	DenV	DenV	MaxD	DenV	DenV	MaxD	DenV	DenV	MaxD	DenV	DenV	MaxD	DenV
Spatial Density Model																	
L1/16/4	.2572	.0516	.0639	.2653	.0855	.0621	.2706	.0783	.0621	.2679	.0756	.0621	.2653	.0840	.0621	.2653	.0727
L1/16/16	.2643	.0417	.0896	.2653	.0915	.0705	.2696	.0773	.0705	.2676	.0758	.0705	.2653	.0755	.0705	.2653	.0753
L2/28/4	.1887	.0500	.0326	.2288	.1012	.0529	.2244	.0986	.0482	.2236	.0973	.0326	.2202	.0908	.0328	.2181	.0898
L2/28/16	.1887	.0497	.0577	.1911	.0643	.0672	.1941	.0721	.0613	.1932	.0658	.0544	.1921	.0646	.0559	.1919	.0655
Effective Density Model																	
L1/16/4	.4161	.1073	.0512	.4244	.0703	.0788	.4251	.0904	.0520	.4286	.0713	.0481	.4245	.0693	.0499	.4251	.0724
L1/16/16	.4816	.0000	.2156	.4818	.2283	.2488	.5091	.2787	.1811	.5169	.2215	.1850	.4818	.2167	.1811	.4818	.2086
L2/28/4	.2977	.1008	.0291	.3419	.1060	.0630	.3385	.1097	.0481	.3340	.0974	.0480	.3186	.1013	.0397	.3240	.0926
L2/28/16	.5577	.0000	.2417	.5753	.2987	.2417	.5845	.2946	.2617	.5800	.3161	.2302	.5691	.2916	.2533	.5711	.3097

Table 2: Multi-level density analysis on results from existing fixed-dissection filling methods. Notation: *T/W/r:* Layout / window size / r-dissection; *LP:* linear programming method; *Greedy:* Greedy method; *MC:* Monte-Carlo method; *IGreedy:* iterated Greedy method; *IMC:* iterated Monte-Carlo method; *OrgDen:* density of original layout; *FD:* fixed-dissection density analysis; *Multi-Level:* multi-level density analysis; *MaxD:* maximum window density; *MinD:* minimum window density; *DenV:* density variation.

Test case	LP			Greedy			MC			IGreedy			IMC		
T/W/r	LipI	LipII	LipIII	LipI	LipII	LipIII	LipI	LipII	LipIII	LipI	LipII	LipIII	LipI	LipII	LipIII
Spatial Density Model															
L1/16/4	.0832	.0837	.0713	.0712	.0738	.0627	.0678	.0709	.0600	.0818	.0824	.0630	.0673	.0698	.0597
L1/16/16	.0854	.0868	.0711	.0730	.0742	.0644	.0708	.0742	.0643	.0724	.0725	.0617	.0707	.0730	.0610
L2/28/4	.0414	.0989	.0841	.0412	.0960	.0893	.0289	.0947	.0852	.0333	.0883	.0755	.0286	.0873	.0766
L2/28/16	.0330	.0642	.0632	.0388	.0713	.0707	.0248	.0658	.0658	.0272	.0619	.0604	.0265	.0631	.0606
Effective Density Model															
L1/16/4	4.048	4.333	3.864	5.332	5.619	5.190	3.631	4.166	3.448	3.994	4.254	3.132	4.245	4.481	3.315
L1/16/16	0.843	0.843	0.835	0.978	1.051	1.051	0.814	0.847	0.847	0.839	0.847	0.847	0.763	0.770	0.770
L2/28/4	2.882	5.782	4.855	2.694	6.587	6.565	1.498	5.579	5.092	2.702	6.317	5.678	2.532	5.640	4.981
L2/28/16	1.000	1.159	1.159	1.061	1.147	1.147	1.115	1.235	1.230	0.936	1.136	1.128	1.112	1.204	1.189

Table 3: Different behaviors of existing filling methods on "smoothness" objectives. Note: All data for effective density model have been timed by 10^3. Notation: *LipI:* Lipschitz condition I; *LipII:* Lipschitz condition II; *LipIII:* Lipschitz condition III.

Here, U is the given upper bound of the effective tile densities. The constraints (5) imply that features can be added but not deleted from any tile. The slack constraints (6) are computed for each tile. The pattern-dependent coefficient $pattern$ denotes the maximum pattern area which can embedded in an empty unit square. If a tile T_{ij} is originally overfilled, then we set $slack(T_{ij}) = 0$. In the LP solution, the values of p_{ij} indicate the fill amount to be inserted in each tile T_{ij}. The constraint (7) says that no window can have density greater than than U (unless it was initially overfilled). The constraints (8) imply that the auxiliary variable L is an upper bound on all variation between $(2r+1)$ windows in the same row.

The second Linear Programming formulation for the Min-Lip-II objective replaces the constraints (8) with the following constraints (9):

$$\begin{aligned} minDen(i,j) \le W_{lm} \le maxDen(i,j) \\ maxDen(i,j) - minDen(i,j) \le L \\ i,j = 1,\ldots,\frac{nr}{w} - 1, l(m) = i(j) - r, \ldots, i(j) + r \end{aligned} \qquad (9)$$

Here, the auxiliary variables $minDen(i,j)$ and $maxDen(i,j)$ are the minimum and maximum tile effective densities in square $(2r+1 \times 2r+1)$ centered at $T_{i,j}$. The constraints above ensure that the density variations among all windows which cover $T_{i,j}$ is less than the auxiliary variable L.

The Min-Lip-III objective strives to minimize the maximum density variation of every cluster of windows which cover one square consisting of k tiles. The constraints (9) are changed to the following:

$$i,j = 1,\ldots,\frac{nr}{w} - 1, l(m) = i(j) - \frac{r}{2}, \ldots, i(j) + \frac{r}{2} \qquad (10)$$

The constraints (10) ensure that the density variation between any two windows which cover $r \times r$ tiles is less than the auxiliary variable L.

Finally, in order to consider the "smoothness" objectives together with the Min-Var objective, we propose another LP formulation with the **combined** objective which is the linear summation of Min-Var, Lip-I, and Lip-II objectives with specific coefficients.

Minimize: $\quad C_0 * M + C_1 * L_I + C_2 * L_{II}$

Lip-I Constraints (8), Lip-II (9) and Min-Var constraints (11) are added for the combined objective:

$$M \le W_{i,j}, \; i,j = 0,\ldots,\frac{nr}{w} - 1 \qquad (11)$$

Here, the auxiliary variables L_I and L_{II} are the maximum Lipschitz condition type-I and type-II, and the auxiliary variable M is a lower bound on all tile densities.

3.3 Computational Experience

We tested the smoothness of filling solutions generated using the same test cases, with smoothness evaluated using finest-r density analysis with $r = 64$. Runtimes of the new methods are substantially longer than for the original Min-Var LP formulation, because many more constraints are added for each layout window due to the Lipschitz condition objectives. For example, for L2/28/8 and the spatial density model, the runtime of Min-Var LP is 6.9 sec-

Test case	Min-Var LP				LipI LP				LipII LP				LipIII LP				Comb LP			
T/W/r	DenV	Lip1	Lip2	Lip3	DenV	Lip1	Lip2	Lip3	DenV	Lip1	Lip2	Lip3	DenV	Lip1	Lip2	Lip3	DenV	Lip1	Lip2	Lip3
Spatial Density Model																				
L1/16/4	.0855	.0832	.0837	.0713	.1725	.0553	.1670	.1268	.1265	.0649	.0663	.0434	.1273	.0733	.0734	.0433	.1143	.0574	.0619	.0409
L1/16/8	.0814	.0734	.0777	.0670	.1972	.0938	.1932	.1428	.1702	.1016	.1027	.0756	.1835	.1158	.1224	.0664	.1707	.0937	.1005	.0766
L2/28/4	.1012	.0414	.0989	.0841	.0724	.0251	.0720	.0693	.0888	.0467	.0871	.0836	.0943	.0462	.0928	.0895	.0825	.0242	.0809	.0758
L2/28/8	.0666	.0340	.0658	.0654	.0871	.0264	.0825	.0744	.0700	.0331	.0697	.0661	.1188	.0594	.1033	.0714	.0747	.0255	.0708	.0656
Effective Density Model																				
L1/16/4	.0703	.0045	.0043	.0039	.2662	.0040	.0154	.0100	.1594	.0039	.0047	.0033	.1792	.0043	.0051	.0030	.1753	.0040	.0045	.0034
L1/16/8	.1709	.0025	.0025	.0023	.3939	.0020	.0060	.0052	.2902	.0025	.0025	.0018	.2906	.0028	.0029	.0018	.2680	.0021	.0022	.0019
L2/28/4	.1060	.0029	.0058	.0049	.1051	.0013	.0061	.0061	.1022	.0029	.0064	.0054	.1039	.0026	.0064	.0052	.0953	.0015	.0057	.0049
L2/28/8	.1483	.0015	.0023	.0022	.1527	.0007	.0024	.0024	.1559	.0015	.0023	.0022	.2063	.0018	.0032	.0022	.1382	.0007	.0021	.0021

Table 4: Comparison among the LP methods on Min-Var and Lipschitz condition objectives. Notation: *Min-Var LP:* LP with Min-Var objective; *LipI LP:* LP with Min-Lip-I objective; *LipII LP:* LP with Min-Lip-II objective; *LipIII LP:* LP with Min-Lip-III objective; *Com LP:* LP with combined objective.

onds, while the Lip-I LP runtime is 3.41 seconds, the Lip-II LP runtime is 994 seconds and the Lip-III LP runtime is 71.4 seconds. For L2/28/8 and the effective density model, the runtime of Min-Var LP is 2.3 seconds, while the Lip-I LP runtime is 708 seconds, the Lip-II LP runtime is 5084 seconds and the Lip-III LP runtime is 2495 seconds. Since fill generation is a post-processing step (currently performed in PV tools), we do not believe that these runtimes are prohibitive. Our major win is that LP is tractable with Lipschitz objectives (as opposed to intractable with large values of r). Of course, finding smoothness objectives that result in smaller LPs is a direction for future work.

The performances of the new LP formulations with "smoothness" objectives are studied in Table 4. We use the coefficients (0.4/0.4/0.2) in the combined objective; these values were derived from the greedy testing of all coefficient combinations. Because of LP's rounding error,[5] some new LPs do not achieve the best value on certain test cases. From the comparison between the new LPs and Min-Var LP, it appears that neither Min-Var LP nor the Lipschitz condition-derived LPs are dominant. At the same time, when compared against existing filling methods in Table (3), the new LP with combined objective normally achieves the best comprehensive solutions in terms of trading off among the Min-Den, Lipschitz conditions I and II. Another interesting observation is that the LP with combined objective can achieve even smaller density variations than the Min-Var LP. This shows that the solution qualities of LP methods can be significantly damaged by rounding effects, and that a better non-LP method may be possible.

4. CONCLUSIONS & FUTURE RESEARCH

To improve manufacturability and performance predictability, it is necessary to "smoothen" a layout by the insertion of "filling (dummy) geometries". In this paper, we pointed out the potentially large difference between fixed-dissection filling results and the actual maximum or minimum window density in optimal density analyses. We compared existing filling algorithms in gridless mode using the effective as well as the spatial density models. We also suggested new methods of measuring local uniformity of the layout based on Lipschitz conditions and proposed new filling methods based on these properties. Our experimental results highlight the advantages of Monte-Carlo and greedy -based methods over previous linear program based approaches.

Ongoing work addresses extensions of multi-level density analyses to measuring local uniformity ("smoothness") with respect to other CMP physical models. We also seek improved methods for optimizing fill synthesis with respect to our new (and possibly other alternative) local uniformity objectives.

[5] The desired fill area specified for each tile in the LP solution must be rounded to an area that corresponds to an an integer number of dummy fill features.

5. REFERENCES

[1] D. Boning, B. Lee, T. Tubawa, and T. Park, "Models for Pattern Dependencies: Capturing Effects in Oxide, STI, and Copper CMP", *Semicon/West Tech. Symp.: CMP Tech. for ULSI Manuf.*, July 2001.

[2] Y. Chen, A. B. Kahng, G. Robins and A. Zelikovsky, "Hierarchical Dummy Fill for Process Uniformity", *Proc. ASP-DAC*, Jan. 2001, pp.139-144.

[3] Y. Chen, A. B. Kahng, G. Robins and A. Zelikovsky, "Practical Iterated Fill Synthesis for CMP Uniformity", *Proc. Design Automation Conf.*, Los Angeles, June 2000, pp. 671-674.

[4] Y. Chen, A. B. Kahng, G. Robins and A. Zelikovsky, "New Monte-Carlo Algorithms for Layout Density Control", *Proc. ASP-DAC*, 2000, pp. 523-528.

[5] R. R. Divecha, B. E. Stine, D. O. Ouma, J. U. Yoon, D. S. Boning, et al., "Effect of Fine-line Density and Pitch on Interconnect ILD Thickness Variation in Oxide CMP Process", *Proc. CMP-MIC*, 1998.

[6] A. B. Kahng, G. Robins, A. Singh, H. Wang and A. Zelikovsky, "Filling Algorithms and Analyses for Layout Density Control", *IEEE Trans. Computer-Aided Design* 18(4) (1999), pp. 445-462.

[7] A. B. Kahng, G. Robins, A. Singh, H. Wang and A. Zelikovsky, "Filling and Slotting: Analysis and Algorithms", *Proc. ACM/IEEE Intl. Symp. on Physical Design*, April 1998, pp. 95-102.

[8] G. Nanz and L. E. Camilletti, "Modeling of Chemical-Mechanical Polishing: A Review", *IEEE Trans. on Semiconductor Manuf.* 8(4) (1995), pp. 382-389.

[9] D. Ouma, D. Boning, J. Chung, G. Shinn, L. Olsen, and J. Clark, "An Integrated Characterization and Modeling Methodology for CMP Dielectric Planarization", *Intl. Interconnect Technology Conference*, San Francisco, CA, June 1998.

[10] B. Stine, "A Closed-Form Analytical Model for ILD Thickness Variation in CMP Processes", *Proc. CMP-MIC*, 1997.

[11] B. Stine, D. Ouma, R. Divecha, D. Boning and J. Chung, "Rapid Characterization and Modeling of Pattern Dependent Variation in Chemical Mechanical Polishing", *IEEE Trans. Semi. Manuf.*, Feb. 1998.

[12] R. Tian, D. Wong, and R. Boone, "Model-Based Dummy Feature Placement for Oxide Chemical Mechanical Polishing Manufacturability", *Proc. Design Automation Conf.*, June 2000, pp. 667-670.

[13] R. Tian, X. Tang and D. F. Wong, "Dummy feature placement for chemical-mechanical polishing uniformity in a shallow trench isolation process ", *International Symposium on Physical Design*, April 2001, pp. 118-123.