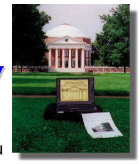
Area Fill Synthesis Algorithms for Enhanced VLSI Manufacturability

Y. Chen, A. B. Kahng, G. Robins, A. Zelikovsky

yuchen@cs.ucla.edu, abk@cs.ucsd.edu, robins@cs.virginia.edu, alexz@cs.gsu.edu



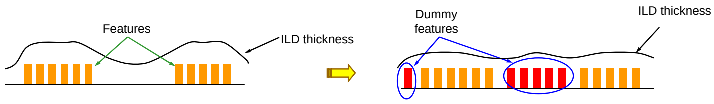
Department of
Computer Science
University of Virginia
School of Engineering & Applied Science

Introduction and Problem Statement

www.cs.virginia.edu/robins

Chemical-Mechanical Polishing (CMP)

- Rotating pad polishes each layer on wafers to achieve planarized surfaces
- Uneven features cause polishing pad to deform
- Density control is achieved by adding fill geometries into layout



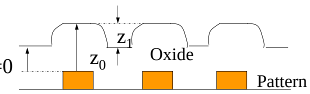
- Industry context: *fixed-dissection regime:*
  - Density constraints imposed only for ***fixed*** set of $w \times w$ windows
  - Layout partitioned by r^2 fixed dissections
  - Each $w \times w$ window is partitioned into r^2 ***tiles***

The Filling Problem

- Given: design rule-correct layout in an n × n layout region
- Find design rule-correct filled layout, such that:
  - No fill geometry is added within distance B of any layout feature, and:
  - **Min-Var objective**: no fill is added into any window with density ≥U, and minimum window density in the filled layout is maximized, or:
  - **Min-Fill objective**: number of filling features is minimized, and density of any window remains within given range (LB, UB)

Model for oxide planarization via CMP

$$z = \begin{cases} z_0 - \left(\dfrac{K_i t}{\rho(x,y)} \right) & t < (\rho_0 Z_1)/K_i \\ z_0 - z_1 - K_i t + \rho_0(x,y) z_1 & t > (\rho_0 z_1)/K_i \end{cases}$$



Crucial model element: determining the effective initial pattern density ρ(x,y)

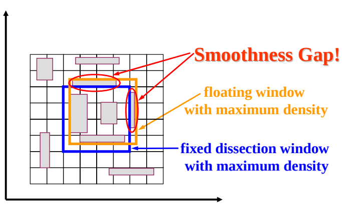
Smoothness Gap & New Local Density Smoothness

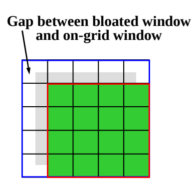
- Spatial Local Density:

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

- Effective Local Density:

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





- Fixed-dissection analysis ≠ floating window analysis
- Fill result will not satisfy the given bounds
- Previous filling methods fail to consider smoothness gap

Accurate Layout Density Analysis

- Optimal extremal-density analysis (K^2)
  - ⇨ Computational inefficiency!
- Multi-level density analysis algorithm:
  - Any window contained by *bloated* on-grid window
  - Any window contains *shrunk* on-grid window
  - Gap between max bloated and max on-grid window
  - ⇨ Algorithmic inaccuracy!

Three Types of Local Density Variation:

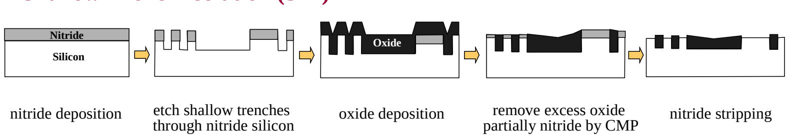
1: Max density variation of every r neighboring windows in each fixed-dissection row

2: Max density variation of every cluster of windows which cover one tile

3: Max density variation of every cluster of windows which cover $\frac{r}{2} \times \frac{r}{2}$ tiles

STI Dual-Material Dummy Fill

Shallow Trench Isolation (STI)



STI CMP Model

- **STI post-CMP variation:** controlled by changing the feature density distribution
- **Compressible pad model:** polishing occurs on up/down areas after some step height
- **Dual-material polish model:** two different materials for top & bottom surfaces

STI Fill is a non-linear programming problem!

Previous methods

- **Min-Var objective:** minimize max height variation
- **Min-Fill objective:** minimize total inserted fills, while keeping given lower bound

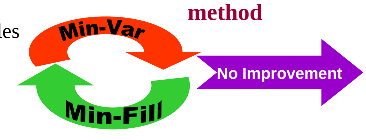
Drawbacks of previous work

- Can not guarantee to find a global minimum since it is deterministic
- Simple termination is not sufficient to yield optimal/sub-optimal solutions

Monte-Carlo method for STI Min-Var

- Calculate priority of tile(i,j) as *ΔH - ΔH (i, j, i', j')*
- Pick the tile for next filling randomly
- If the tile is overfilled, lock all neighboring tiles
- Update tile priority

Iterated Monte-Carlo method

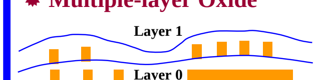


MC/Greedy methods for STI Min-Fill

- Find a solution with **Min-Var** objective to satisfy the given lower bound
- Modify the solution with respect to **Min-Fill** objective

Multiple-Layer Oxide CMP Dummy Fill

Multiple-layer Oxide



Multiple-layer density model

$$\hat{\rho}_{0(k)} = \begin{cases} [\hat{d}_k + \dfrac{z_{k-1}}{z_k} \hat{\rho}_{0(k-1)}] \times \hat{f} & k > 1 \\ \hat{d}_1 \times \hat{f} & k = 1 \end{cases}$$

Multiple-layer Oxide Fill Objectives

- **Sum of density variations can not guarantee the Min-Var objective on each layer**
- **Maximum density variation across all layers**

Linear Programming formulation

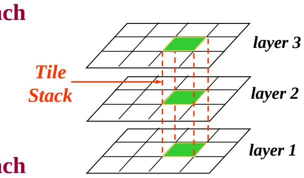
- Min $\quad M$
- Subject to: $\quad 0 \le L_k \le \rho_0(i,j,k) \le H_k \le \sum_{l=1}^{k} \left(\frac{z_l}{z_k} \right) \quad i,j = 0,\ldots,\frac{nr}{w}-1, k=1,\ldots,K$

$$(H_k - L_k) \le M \qquad k = 1,\ldots,K$$

$$0 \le x_{ijk} \le slack(T_{ijk}) \qquad i,j = 0,\ldots,\frac{nr}{w}-1, k=1,\ldots,K$$

Multiple-Layer Monte-Carlo Approach

- **Tile stack**: column of tiles
- **Effective density of tile stack**: sum of effective densities of all tiles in stack



Multiple-Layer Monte-Carlo Approach

- Compute slack area, cumulative effective density of tile stack
- Calculate tile stack's priority according to cumulative effective density
- While (sum of priorities > 0) Do:
  - Randomly select a tile stack according to its priority
  - From bottom to top layer, check for fill insertion feasibility
  - Update slack area and priority of the tile stack
  - If no slack area is left, lock the tile stack