Inverse Lithography

Abde Ali Kagalwalla
Why Resolution Enhancement?

The main problem is sub-wavelength patterning.

Figure from F. Schellenberg, “A little light magic”, IEEE Spectrum, 2003
Inverse Lithography Problem

Formulation

\[ m(x, y) \xrightarrow{\text{Imaging System: } T\{.\}} z(x, y) \approx z^*(x, y) \]

\[ T^{-1}\{.\} \]

\[ \hat{m}(x, y) = \operatorname*{arg\ min}_{m(x, y)} d(z^*(x, y), T\{m(x, y)\}) \]

Equation and Figure taken from A. Poonawala and P. Milanfar, AIP 2007
Discrete approaches

• Many “pixel flipping” based discrete approaches using SA, genetic algorithms, ILP

• One recent “gradient-inspired” pixel flipping by J. Zhang et al at ICCAD 2008 [“A Highly Efficient Optimization Algorithm for Pixel Manipulation in ILT”]

\[ \Delta E_c = \frac{\partial E_c}{\partial m} \Delta m + \frac{1}{2} \frac{\partial^2 E_c}{\partial m^2} (\Delta m)^2 + O[(\Delta m)^3] \]

• Flip the most cost reducing pixels which are not too close to each other for every iteration
• Terminate when no pixel flip reduces cost by more than \( \varepsilon \)
Gradient based continuous method

• Objective: Minimize $||z^* - \text{sig}(|Hm|^2)||_2$
• Relax $m = \{0, 1\}$ to $0 < m < 1$ then transform $m = (1 + \cos \theta)/2$ to make it an unconstrained optimization
• Reduce rounding error by adding $m(1 - m)$ to the objective
• Added regularization terms for manufacturability
Level set based method

• Numerical PDE based algorithm to model moving surfaces

• Formulation of Y. Shen, et. al. [Dec. 2009]

\[
\text{minimize } \int_{\Omega} |\nabla U| \, dx \\
\text{subject to } \int_{\Omega} (\text{sig}(|H \ast U|^2) - I_0)^2 \, dx = \varepsilon,
\]

• Final level set PDE solved using methods proposed by Osher et. al.

\[
\frac{\partial U}{\partial t} = -\alpha(x, t) + \lambda \nabla \cdot \left( \frac{\nabla U}{|\nabla U|} \right),
\]
Comparison with OPC

- Main advantage of OPC is more manufacturable masks
  - Speed used to be another one but it scales worse than ILT
- ILT has better litho performance
  - Not restricted to local solutions
  - No need for separate rule-based assist feature insertion step.

OPC algorithm. Figure taken from N. Cobb’s PhD thesis
Some unintuitive ILT patterns: Contact Array

- pitch=240nm
- pitch=340nm
- pitch=440nm
- pitch=540nm
- pitch=640nm
- pitch=740nm
Some unintuitive ILT patterns: Poly Pattern
Some current results: Full chip ILT (Samsung + Luminescent)

<table>
<thead>
<tr>
<th></th>
<th>Contact</th>
<th>Poly</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Conv. OPC</td>
<td>ILT</td>
</tr>
<tr>
<td>Input File Size (OASIS)</td>
<td>0.13 GB</td>
<td>4.2 GB</td>
</tr>
<tr>
<td>Output File Size (VSB12)</td>
<td>1.7 GB</td>
<td>58 GB</td>
</tr>
<tr>
<td>Data Conversion (Patacon)</td>
<td>0.5 hrs</td>
<td>2 hrs</td>
</tr>
<tr>
<td>Shot Count (Mshots)</td>
<td>209M</td>
<td>1,496M</td>
</tr>
<tr>
<td>Max Shot Density (Mshots/mm²)</td>
<td>4.7</td>
<td>32.6</td>
</tr>
<tr>
<td>Writing Time (EBM6000)</td>
<td>10.2 hrs</td>
<td>24.5 hrs</td>
</tr>
<tr>
<td>Estimated Writing Time (EBM7000)</td>
<td>7.1 hrs</td>
<td>17.2 hrs</td>
</tr>
</tbody>
</table>
Some current results: Full chip ILT (Samsung + Luminescent)

<table>
<thead>
<tr>
<th></th>
<th>Contact</th>
<th>Poly</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Conv. OPC</td>
<td>ILT (Seg20)</td>
</tr>
<tr>
<td>Shot Count (Mshots)</td>
<td>209M</td>
<td>1,496M</td>
</tr>
<tr>
<td>Max Shot Density (Mshots/mm²)</td>
<td>4.7</td>
<td>32.6</td>
</tr>
<tr>
<td>Writing Time (EBM6000)</td>
<td>10.2 hrs</td>
<td>24.5 hrs</td>
</tr>
<tr>
<td>Estimate Writing Time (EBM7000)</td>
<td>7.1 hrs</td>
<td>17.2 hrs</td>
</tr>
</tbody>
</table>

![Graphs showing data at nominal and defocus conditions for contact mode.](image-url)