A Fast Machine Learning-based Mask Printability Predictor for OPC Acceleration

Bentian Jiang[1], Hang Zhang[2], Jinglei Yang[3], Evangeline F. Y. Young[1]

[1]CSE Department, The Chinese University of Hong Kong
[2]ECE Department, Cornell Univeristy
[3]ECE Department, University of California, Santa Barbara

Jan. 23, 2019

香港中文大學
The Chinese University of Hong Kong

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

Application on OPC Acceleration

Experimental Results

Conclusion

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

Application on OPC Acceleration

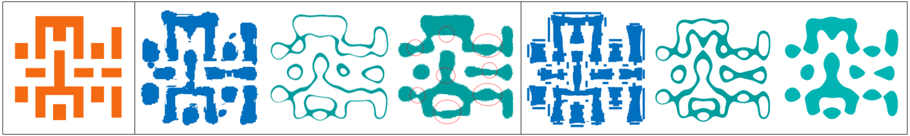
Experimental Results

Conclusion

Optical Proximity Correction

Conventional OPC approaches include

- Inverse lithography-based OPC (ILT)*
  - Pixel-based optimization, larger solution space, expected to achieve better mask printability.
  - Relatively slow, mask manufacturability, etc.
- Forward model-based OPC (FM) †
  - Relatively fast, good performance in practice.
  - Limited solution space, may not converge to optimal for advanced tech nodes.

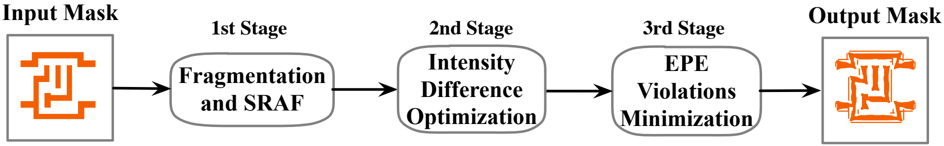


(a) Target (b) ILT (c) (d) (e) FM (f) (g)

*[1] MOSAIC, DAC2014, Gao
†[2] Robust-OPC, DATE2015, Kuang

Key Challenges of Optical Proximity Correction

- Both model-based and ILT OPC involves multiple rounds of litho-simulation.
- Conventional lithography simulation for new feature size suffers from large computational overhead, which makes the OPC process extremely time consuming.
  - General flow of typical model-based OPC tool. ‡
  - The 3rd stage is performed incrementally under the guidance of litho-simulation result, which takes up around **80%** runtime of the entire process.



‡[2] Robust-OPC, DATE2015, Kuang

Motivation

- The large computational overhead of conventional lithography simulation dominates the runtime of conventional OPC process
- Previous learning-based OPC works usually have low scalability for new technology node due to the large time-overhead in training step.
  - E.g. GAN-OPC [§]
  - Customized GAN architecture designed for OPC, reasonably good performance.
  - Time consuming, around 28000 Secs for training, and 400 Secs to perform OPC on a 2x2 um^2 clip.

[§] [3] GAN-OPC, DAC2018, Yang

Our Approach

Feature Extraction and Feature Selection

- Matrix-based Concentric Circle Sampling with CUDA Acceleration
- Second Order Circle Subset Selection via Mutual Information

Learning Models

- Machine Learning-based Mask Printability Predictor

Application

- Machine Learning-based OPC Acceleration

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

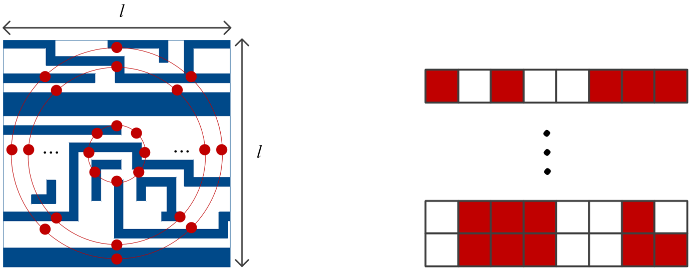
Application on OPC Acceleration

Experimental Results

Conclusion

Matrix-based Concentric Circle Sampling

- We perform matrix based concentric circle sampling (MCCS) for feature extraction ¶
- l denotes the length of the sample clip, n_p denotes the number of sample points on each sample circle.
- R_{in}, inc_1 and inc_2 control the sampling density manually.



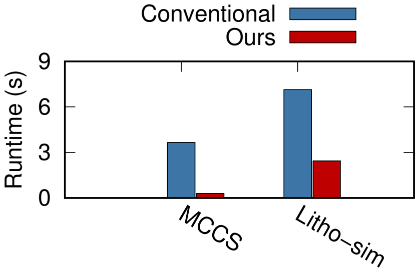
(a)

(b)

¶[4] Bilinear-HSD, ISPD2017, Zhang

CUDA Acceleration of MCCS

- Conventional implementation of MCCS suffers from long runtime. The runtime issue affect the performance of the proposed acceleration framework.
- Implement the MCCS with CUDA (c/c++) to achieve real-time feature extraction, and achieve **91.99%** runtime reduction comparing to conventional CPU-based MCCS (3.652s to 0.2925s).



Second Order Circle Subset Selection via Mutual Information

Primal objective: select n_c^* circles from original circle set that can maximize the dependency of selected circle subset with the target variable y (equivalent to minimize the conditional entropy)

$$\mathcal{I}_{n_c^*} = \arg\max_{\mathcal{I}_{n_c^*} \subseteq \mathcal{I}} \sum_{i \in \mathcal{I}_{n_c^*}} \sum_{j \in \mathcal{I}_{n_c^*}} I(C_i, C_j; Y), \tag{1}$$

where the mutual information $I(C_i, C_j; Y)$ defines the dependency between i^{th} and j^{th} circles with classification variable y

$$I(C_i, C_j; Y) = \sum_{c_i \in C_i} \sum_{c_j \in C_j} \sum_{y \in Y} p(c_i, c_j, y)\ log \frac{p(c_i, c_j, y)}{p(c_i, c_j)\ p(y)}. \tag{2}$$

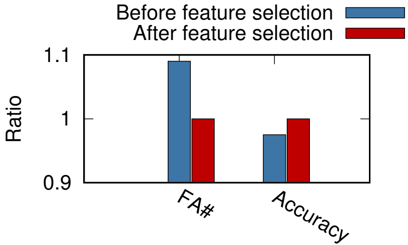
For each pair of circle c_i and circle c_j, encode them into decimal number, and perform statistic on the entire dataset (corresponding c_i, c_j). Finally, construct the mutual information matrix $\mathbf{M}$, where the entry in i^{th} row and j^{th} coloum is $I(C_i, C_j; Y)$.

Second Order Circle Subset Selection via Mutual Information

Rewrite Eqn.(1) as a mixed-integer quadratic programming (MIQP) problem

$$
\begin{aligned}
\max \quad & \mathbf{w}^\top \mathbf{M}\mathbf{w}, \\
\text{s.t.} \quad & \sum_{i=1}^{n_c} w_i = n_c^*, \; w_i \in \{0,1\}, \; \forall i.
\end{aligned}
\tag{3}
$$

The feature selection method reduces 9.4% false alarms and improves 2.38% accuracy



Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

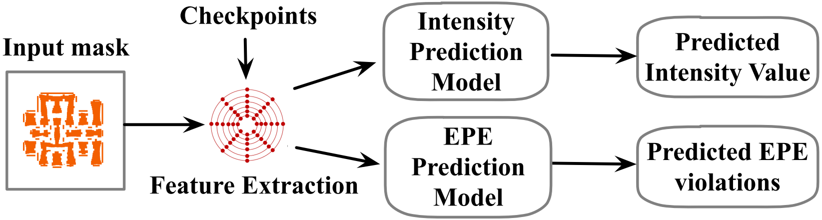
Application on OPC Acceleration

Experimental Results

Conclusion

ML-based Mask Printability Predictor

- The proposed ML-based mask printability predictor consists of
  - A EPE [‖] (edge placement error violation) prediction model (classification)
  - A light intensity prediction model (regression)



[‖] EPE is the horizontal and vertical geometric displacement of the image contour from the corresponding edge of the target layout polygon

ML-based Mask Printability Predictor

For each checkpoint on layout polygon

- EPE prediction model ⇒ whether this checkpoint is a EPE violation, the label is encoded as

$$Label_{EPE}(x, y) = \begin{cases} 1, & \text{if displacement} > \text{Threshold}, \\ -1, & \text{if displacement} \leq \text{Threshold}. \end{cases} \tag{4}$$

- Intensity prediction model ⇒ the adjust direction and step-size for this checkpoint segment, and intensity label is from the ground-truth resist image (continuous).

- Apply **XGBoost** to train both EPE model (classification) and light intensity prediction model (regression), and conduct cross-validation during the training.

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

Application on OPC Acceleration

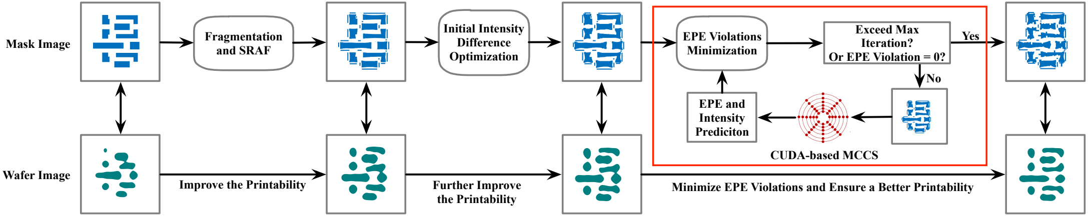
Experimental Results

Conclusion

Flow Overview

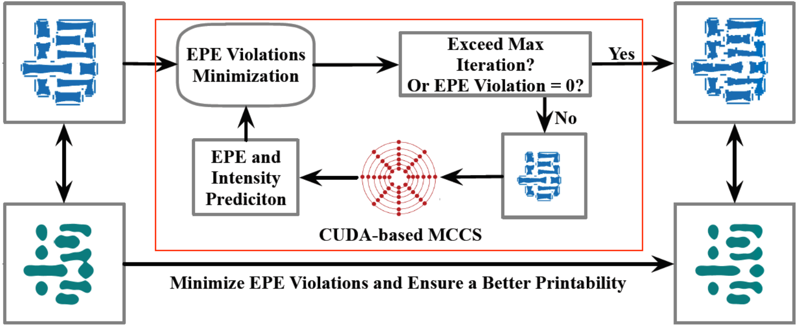
The target is to leverage machine learning technique to break the runtime bottleneck for conventional OPC.

- We regard the conventional mask quality evaluation step as a black-box, and replace it by our ML-based mask printability predictor.



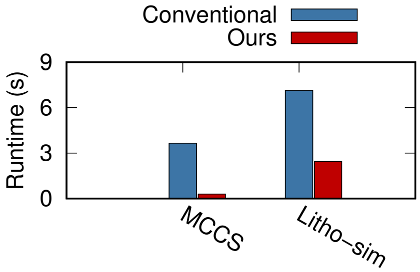
Application on OPC Acceleration

- The outputs of our learning models can determine the behavior of 3rd stage in the model-based OPC (*where* to optimize and *how* to optimize).
- We embed our CUDA-based MCCS extractor and pre-trained EPE/intensity prediction models into the 3rd stage, to skip the lithography simulation process.

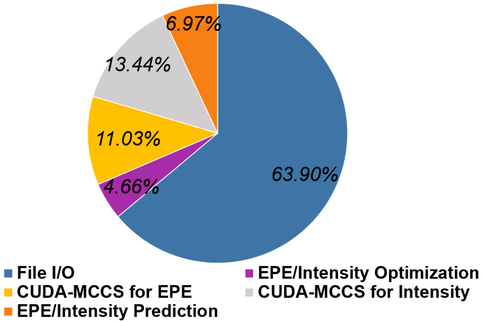


Application on OPC Acceleration

- Fast mask printability evaluation scheme reduces **65.8%** runtime comparing to conventional litho-simulation
- Due to the cross-platform implementation, extra I/O time (**63.9%** of total time) is introduced for communication.



(a) Runtime comparision



(b) Runtime distribution of accelerated stage3

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

Application on OPC Acceleration

Experimental Results

Conclusion

Experimental Results

- On ICCAD 2013 Contest Benchmarks
- 2048nm x 2048nm layout clips of 32nm M1 layer

	EPE prediction model			Intensity prediction model		PGAN-OPC**
	CPU (s)	FA#	Acccuracy	CPU (s)	RMSE	CPU (s)
Regular feature	225	360	91.45%	122	0.003963	28000
Selected feature	1533	329	93.83%	-	-	-
Ratio	0.146	**1.094**	**0.975**	-	-	-

* number of test instances = 4309, mean value of intensity labels = 0.22436.

- Our second order circle subset selection method improves the prediction accuracy for model by **2.38%** and reduces **9.4%** false alarms.
- Our training times are only **5.48%** and **0.44%** of PGAN-OPCs.

** [3] GAN-OPC, DAC2018, Yang

Experimental Results

Benchmarks		Ours		Original mask optimizer		ILT		GAN-OPC		PGAN-OPC	
ID	Area (nm^2)	RT (s)	L_2 (nm)	RT (s)	L_2 (nm)	RT (s)	L_2 (nm)	RT (s)	L_2 (nm)	RT (s)	L_2 (nm)
case1	215344	69.14	44721	278	53816	1280	49893	380	54970	358	52570
case2	169280	69.09	37418	142	41382	381	50369	374	46445	368	42253
case3	213504	80.86	80491	152	79255	1123	81007	379	88899	368	83663
case4	82560	67.61	19038	307	21717	1271	20044	376	18290	377	19965
case5	281958	71.69	47423	189	48858	1120	44656	378	42835	369	44733
case6	286234	76.53	44762	353	46320	391	57375	367	44313	364	46062
case7	229149	70.18	30400	219	31898	406	37221	377	24481	377	26438
case8	128544	68.87	18200	99	23312	388	19782	394	17399	383	17690
case9	317581	73.38	55767	119	55684	1138	55399	427	53637	383	56125
case10	102400	68.74	14451	61	19722	387	24381	395	9677	366	9990
Average	-	**71.61**	**39267.1**	191.9	42196.4	788.5	44012.7	383.6	40094.6	371.9	39948.9
Ratio	-	**1.00**	**1.00**	2.68	1.0746	11.011	1.1209	5.356	1.0211	5.193	1.0174

* $\mathbf{L_2}$ error is $||\mathbf{M}-\mathbf{R}||_2^2$, where $\mathbf{M}$ is the target mask image and $\mathbf{R}$ is the wafer image. RT denotes runtime (sec).

- Compared with current mainstream works, we achieve
  - 1.75% - 12.1% better image-fidelity in terms of L_2 (nm);
  - 2.6× - 11× runtime speed-up.

Outline

Introduction

Feature Extraction and Feature Selection

ML-based Mask Printability Predictor

Application on OPC Acceleration

Experimental Results

Conclusion

Conclusion

- Propose a set of fast and accurate mask printability prediction models based on machine learning techniques.
- Develop with CUDA a matrix-based concentric circle sampling method for feature extraction, followed by a second order circle subset selection algorithm for feature selection.
- Develop a machine learning-based OPC acceleration framework, which achieves 2.6X-11X runtime speedup with a comparable printability.

Q&A

Thanks and Questions?

GAO, J.-R., XU, X., YU, B., AND PAN, D. Z.
MOSAIC: Mask optimizing solution with process window aware inverse correction.
In *Proc. DAC* (2014), pp. 52:1–52:6.

KUANG, J., CHOW, W.-K., AND YOUNG, E. F. Y.
A robust approach for process variation aware mask optimization.
In *Proc. DATE* (2015), pp. 1591–1594.

YANG, H., LI, S., MA, Y., YU, B., AND YOUNG, E. F. Y.
Gan-opc: Mask optimization with lithography-guided generative adversarial nets.
In *Proc. DAC* (2018).

ZHANG, H., ZHU, F., LI, H., YOUNG, E. F. Y., AND YU, B.
Bilinear lithography hotspot detection.
In *Proc. ISPD* (2017), pp. 7–14.