Model-based OPC Extension in OpenILT

Su Zheng[1], Gang Xiao[2], Ge Yan[2], Meng Dong[2], Yao Li[2],
Hong Chen[2], Yuzhe Ma[3], Bei Yu[1], Martin Wong[4]

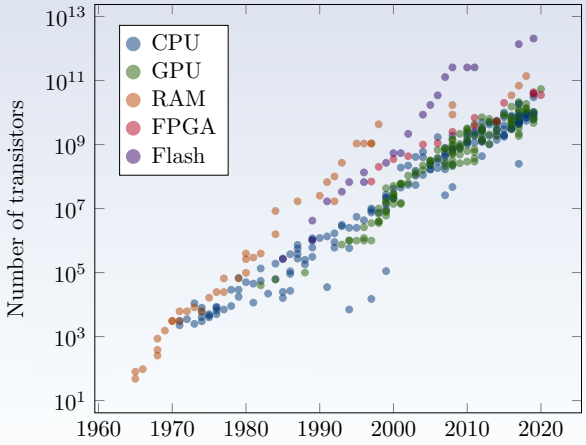
[1]Chinese University of Hong Kong
[2]Shenzhen GWX Technology Co., Ltd.
[3]Hong Kong University of Science and Technology (Guangzhou)
[4]Hong Kong Baptist University

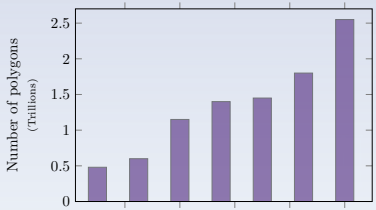
Outline

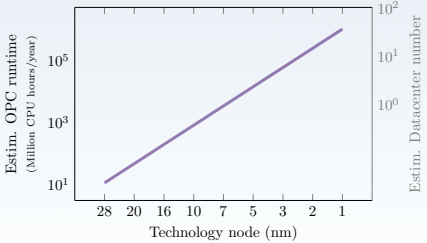
Introduction

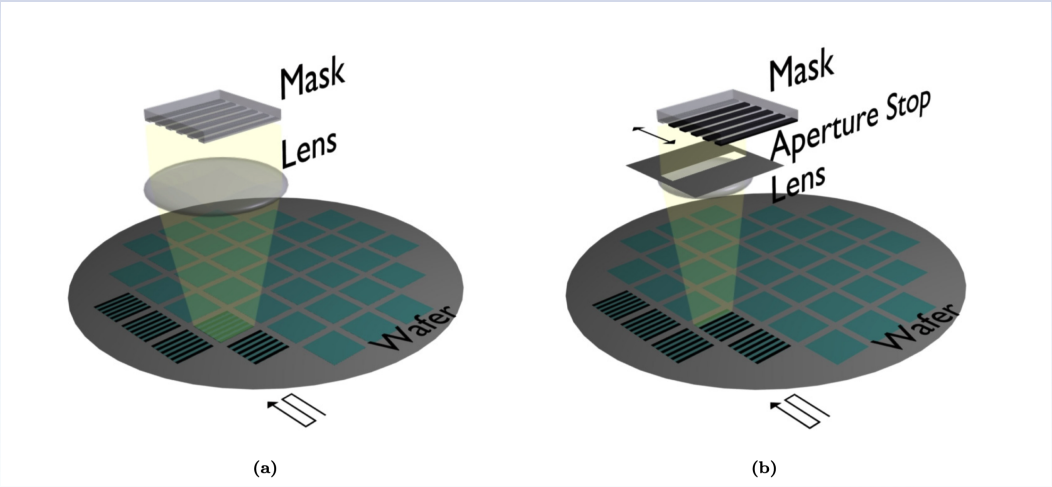
Moore's Law to Extreme Scaling

- Billions of transistors on a chip → ... Trillions of polygons

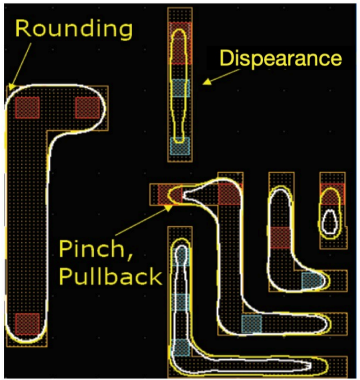


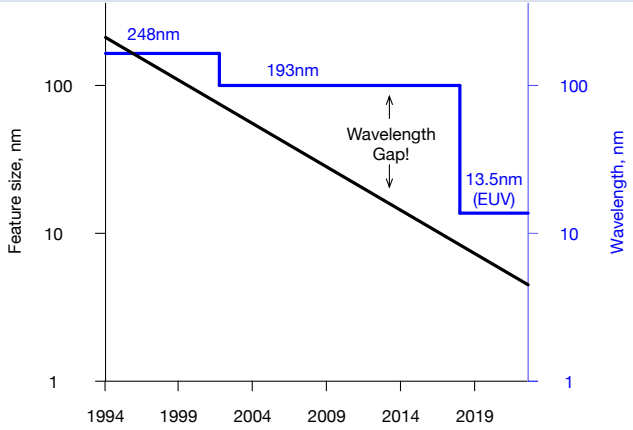




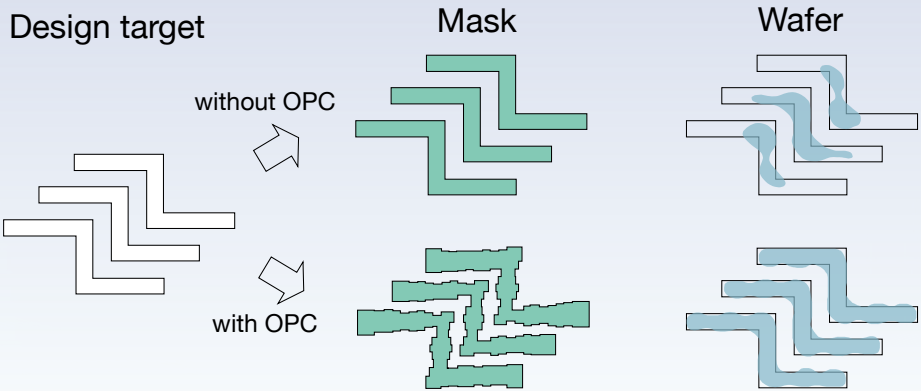

Mask
Lens
Wafer
(a)
Mask
Aperture Stop
Lens
Wafer
(b)

When feature is small: what you see ≠ what you want





Mask Optimization



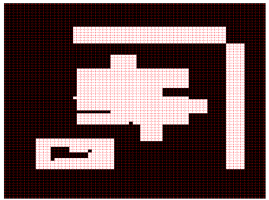
Classic OPC

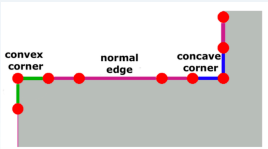
- Model/Rule-based OPC [Cobb+,SPIE'02][Kuang+,DATE'15] [Awad+,DAC'16] [Su+,ICCAD'16]
  1. Fragmentation of shape edges;
  2. Move fragments for better printability.
- Inverse Lithography [Pang+,SPIE'05] [Gao+,DAC'14] [Poonawala+,TIP'07] [Ma+,ICCAD'17]
  1. Efficient model that maps mask to aerial image;
  2. Continuously update mask through descending the gradient of contour error.

Machine Learning OPC

[Matsunawa+,JM3'16] [Choi+,SPIE'16] [Xu+,ISPD'16] [Shim+,APCCAS'16]

1. Edge fragmentation;
2. Feature extraction;
3. Model training.





github.com/OpenOPC/OpenILT/

README.md

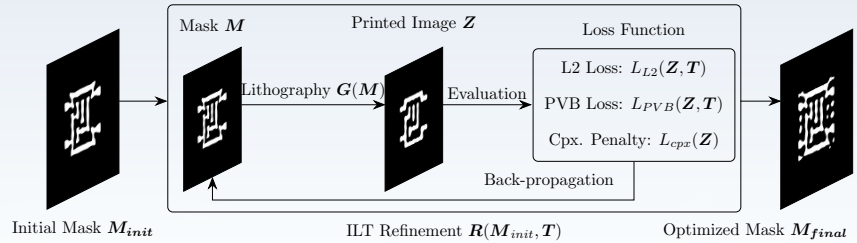
```
   ____                    ____ __   _____
  / __ \ ____   ___   ____    /  _// /  /_  __/
 / / / // __ \ / _ \ / __ \   / / / /    / /
/ /_/ // /_/ //  __// / / / _/ / / /___ / /
\____// .___/ \___//_/ /_/ /___//_____//_/
     /_/
```

OpenILT: An Open-source Platform for Inverse Lithography Technology Research

OpenILT is a open-source platform for inverse lithography technology (ILT) research. It has a comprehensive and flexible ecosystem of libraries that enable the efficient development and evaluation of ILT algorithm. OpenILT decouples the ILT flow into different components, lithography simulation, initialization, optimization, and evaluation. ILT researchers can implement and evaluate their ideas quickly by replacing a component with the novel method. Moreover, the platform is implemented with *pytorch*, which enables easy GPU acceleration and deep-learning integration.

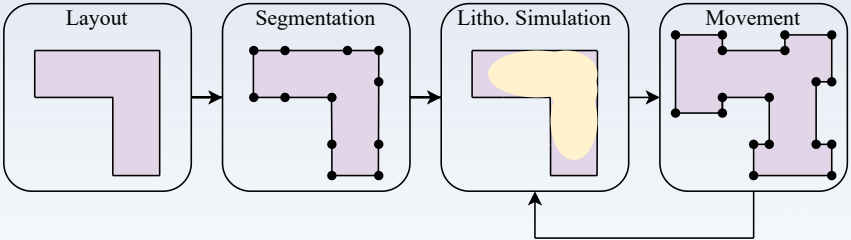
OpenILT Flow

- Lithography simulation
- Initialization
- Solver
- Evaluation



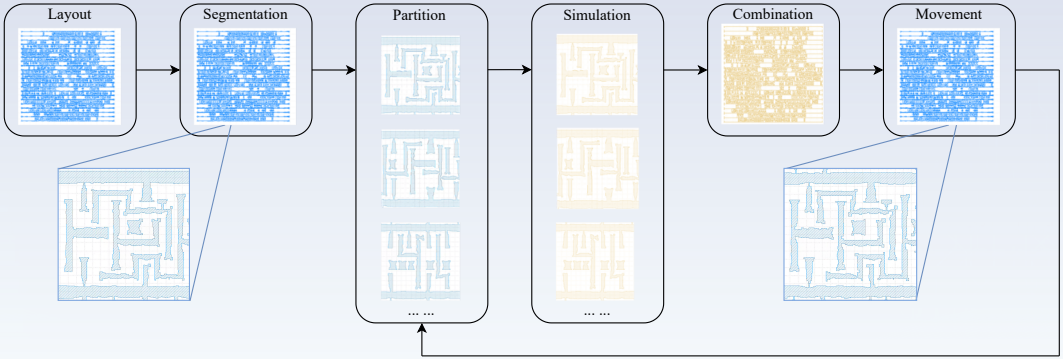
OpenILT Weaknesses

- Lack of Large-scale Lithography simulation
- No Support of Model-based OPC

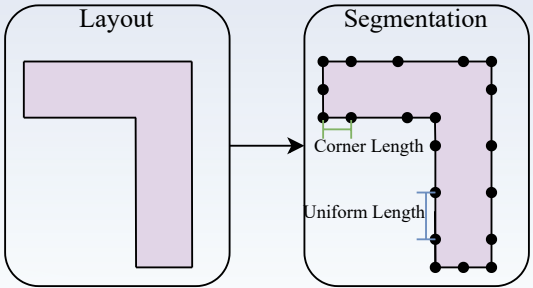


Model-based OPC

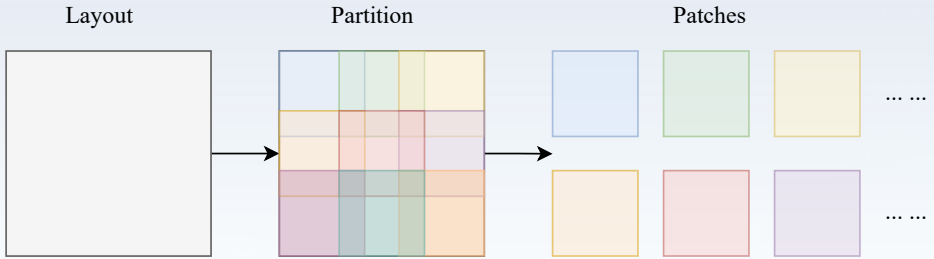
Overview


Layout
Segmentation
Partition
Simulation
Combination
Movement
... ...
... ...

- Segmentation: polygon to segments
  - Corner length
  - Uniform length

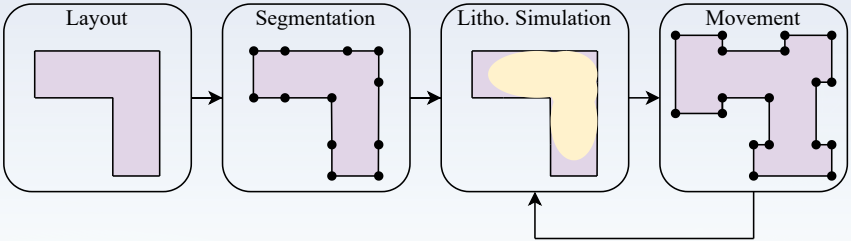


- Partition+Simulation+Combination:
  - **GPU acceleration lithography simulation**
  - Partition the layout to fit the size of lithography simulator
  - Support large-scale lithography simulation in OpenILT



Model-based OPC Steps

- Movement:
  - EPE-driven optimization
  - Outside edge displacement $\rightarrow$ move inwards
  - Inside edge diplacement $\rightarrow$ move outwards



Experiments

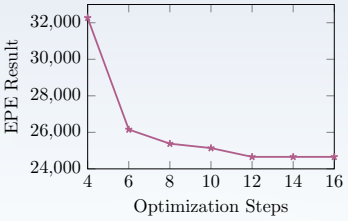
MB-OPC Results on Different Testcases

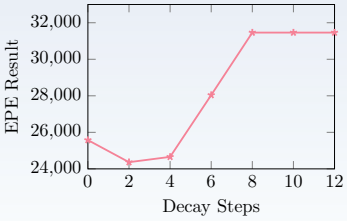
- **GPU acceleration gains significant speed up**

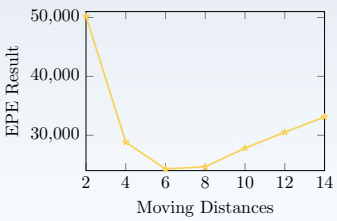
Layout	Size	Tiles	EPE	L2	PVB	Runtime@GPU	Runtime@CPU
`gcd`	$30 \times 30 \mu m^2$	1	24657	595317	149969	150 s	801 s
`aes`	$250 \times 250 \mu m^2$	144	2633481	62804952	16102282	18,612 s	>24 hours
`dynamicnode`	$246 \times 246 \mu m^2$	144	2326174	58335791	14815929	19,494 s	>24 hours

Results under different configurations

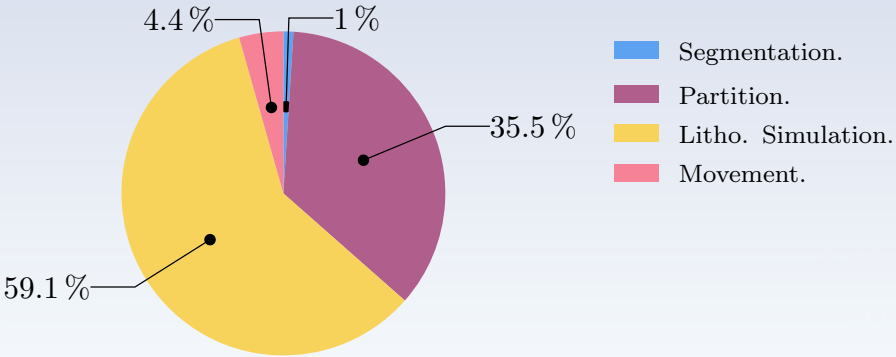
- Optimization steps
- Step size decay
- Movement step size



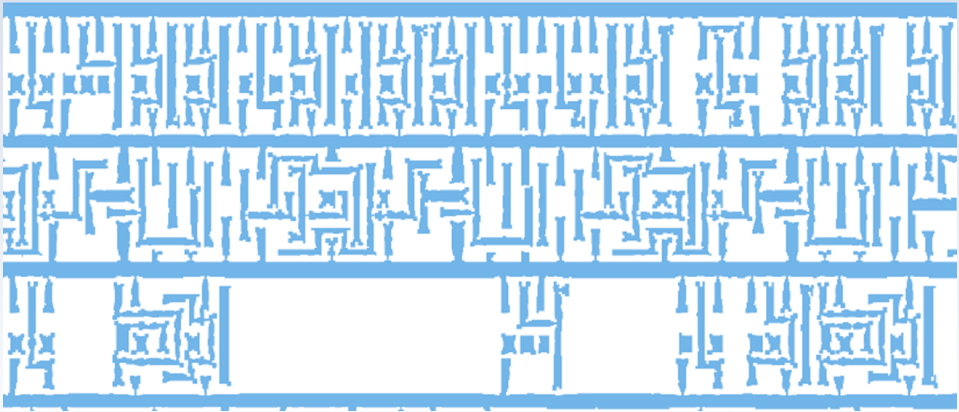




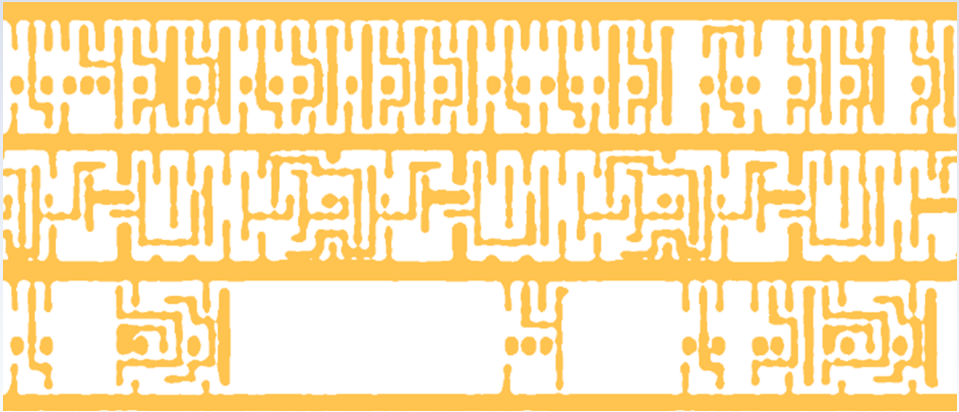
Runtime analysis



Example: OPC Result



Example: Simulation Result



Conclusion

Conclusion

- **Open-source model-based OPC built on OpenILT**
- **GPU acceleration gains significant speed up**
- **Set up a robust baseline for future OPC research**

THANK YOU!