2025 IEEE 16th International Conference on ASIC
Oct. 21-24, 2025, Crowne Plaza Kunming City Centre, Kunming, China

TorchLitho 2.0: Differentiable Lithography Simulation Engine for Large-Scale Layouts

Shuo Yin, **Su Zheng**, Ziyang Yu, Bei Yu

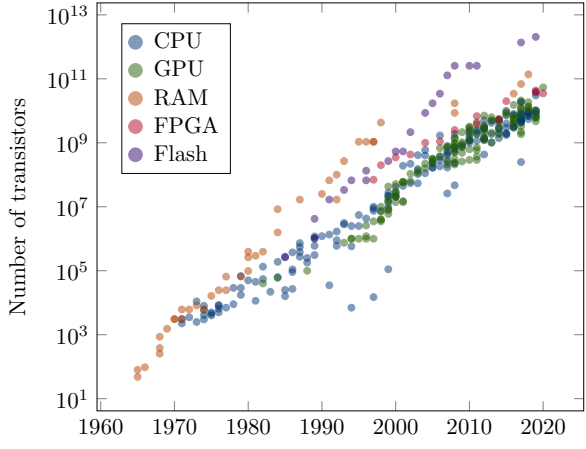
The Chinese University of Hong Kong
{syin22, szheng22, zyyu, byu}@cse.cuhk.edu.hk

Oct. 22, 2025

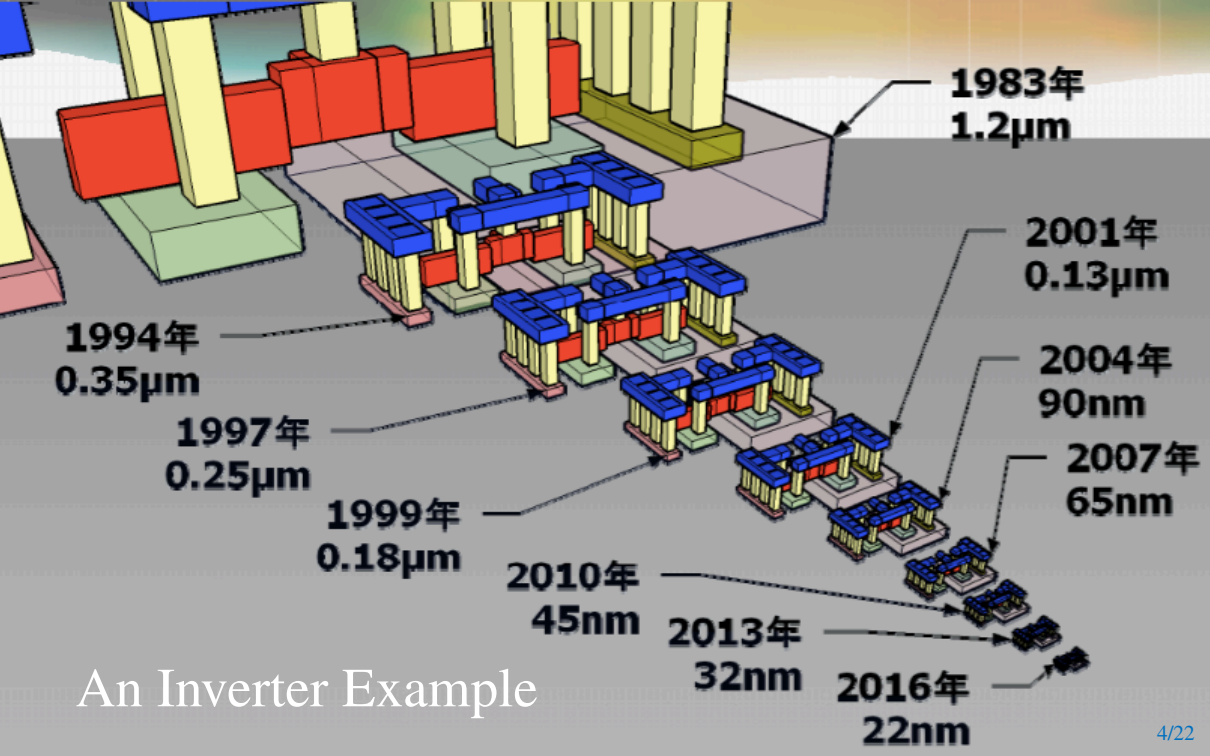
Introduction

Moore's Law to Extreme Scaling

- The designs are becoming increasingly complex.
- Billions of transistors on a chip → ... Trillions of polygons.

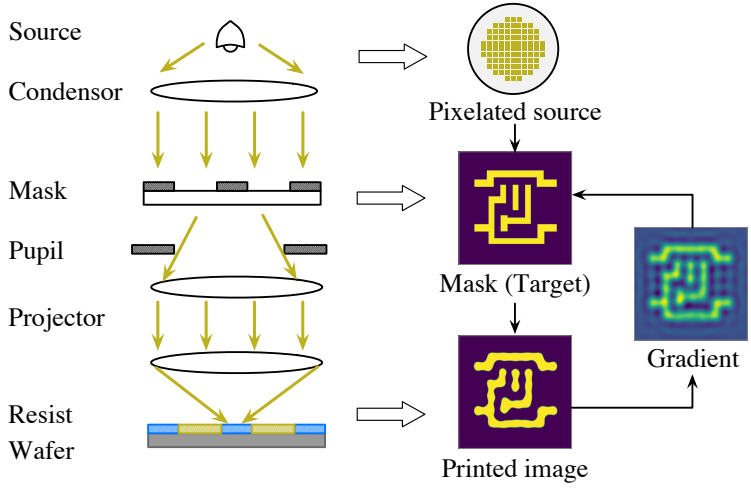


While the technology node keeps shrinking ...



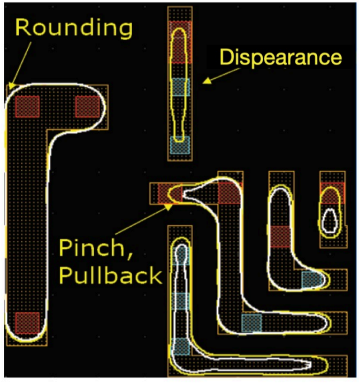
An Inverter Example

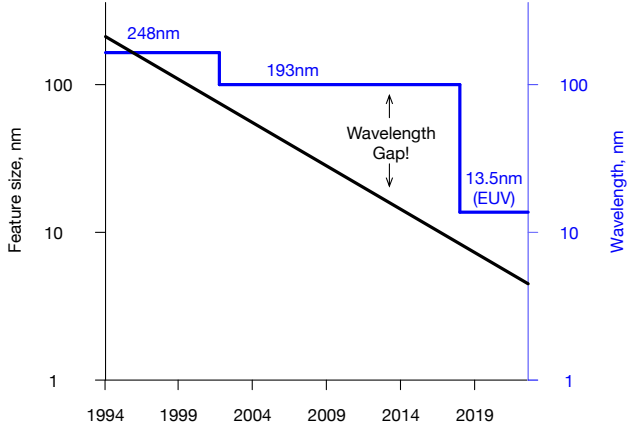
Lithographic System



Simplified schematic diagram of a lithographic system.

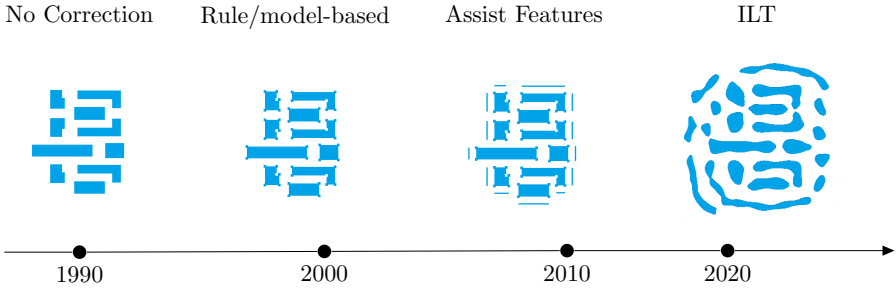
When feature is small: what you see $\neq$ what you want





The Necessity of Optical Proximity Correction (OPC) ...

Historical Timeline of OPC Techniques



Why Inverse Lithography Technology (ILT)?

1. A large optimization space enabled by pixel-based representation.
2. Highly compatible with GPU computation and potentially aided by AI techniques.

Inverse Lithography Technology

- **Optimization-based**: MOSAIC[1], LevelSet-GPU[2], CircleOPC[3], CTM-SRAF[4] ...
- **AI-based**: DevelSet[5], NeuralILT[6], Camo[7], L2O-ILT[8] ...
- **Generative-Based**: GAN-OPC[9], EMOGen[10] ...

[1] **Jhih-Rong Gao et al. (2014). "MOSAIC: Mask Optimizing Solution With Process Window Aware Inverse Correction". In:** *Proc. DAC*. **San Jose, California, 52:1–52:6.**

[2] **Ziyang Yu, Guojin Chen, et al. (2021). "A GPU-enabled Level Set Method for Mask Optimization". In:** *Proc. DATE*.

[3] **Xinyun Zhang et al. (2024). "Fracturing-aware curvilinear ILT via circular e-beam mask writer". In:** *Proc. DAC*, **pp. 1–6.**

[4] **Ziyang Yu, Peiyu Liao, et al. (2023). "CTM-SRAF: Continuous transmission mask-based constraint-aware subresolution assist feature generation". In:** *IEEE TCAD* **42.10, pp. 3402–3411.**

[5] **Guojin Chen et al. (2021). "DevelSet: Deep neural level set for instant mask optimization". In:** *Proc. ICCAD*.

[6] **Bentian Jiang et al. (2020). "Neural-ILT: Migrating ILT to Nerual Networks for Mask Printability and Complexity Co-optimizaton"". In:** *Proc. ICCAD*.

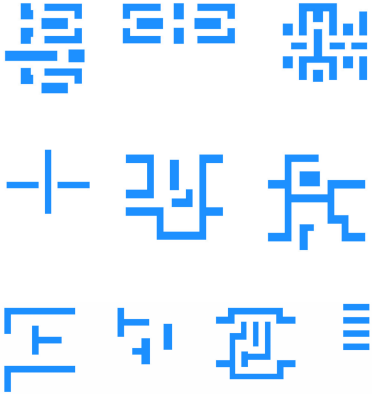
[7] **Xiaoxiao Liang et al. (2024). "Camo: Correlation-aware mask optimization with modulated reinforcement learning". In:** *Proc. DAC*, **pp. 1–6.**

[8] **Binwu Zhu et al. (2023). "L2O-ILT: Learning to optimize inverse lithography techniques". In:** *IEEE TCAD* **43.3, pp. 944–955.**

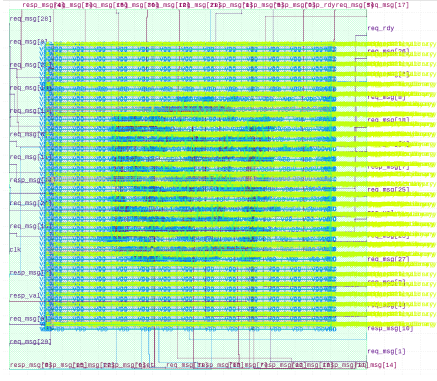
[9] **Haoyu Yang et al. (2018). "GAN-OPC: Mask Optimization with Lithography-guided Generative Adversarial Nets". In:** *Proc. DAC*, **131:1–131:6.**

[10] **Su Zheng et al. (2024). "EMOGen: Enhancing mask optimization via pattern generation". In:** *Proc. DAC*, **pp. 1–6.**

Gap Between Academia & Industry



ICCAD Contest 2013 Design

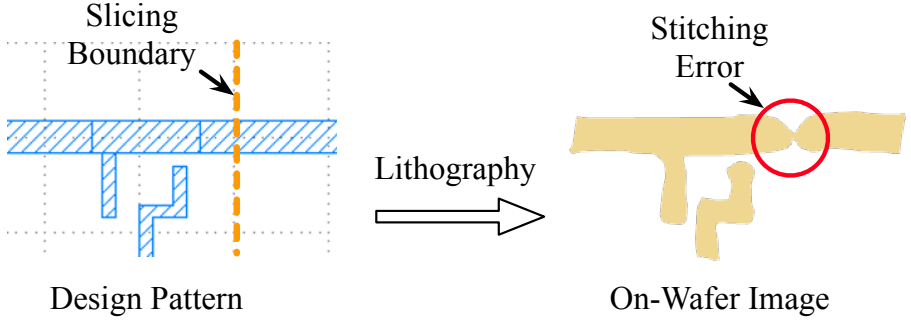


GCD layout generated by OpenROAD throuogh FreePDK45

Gap Between Academia & Industry

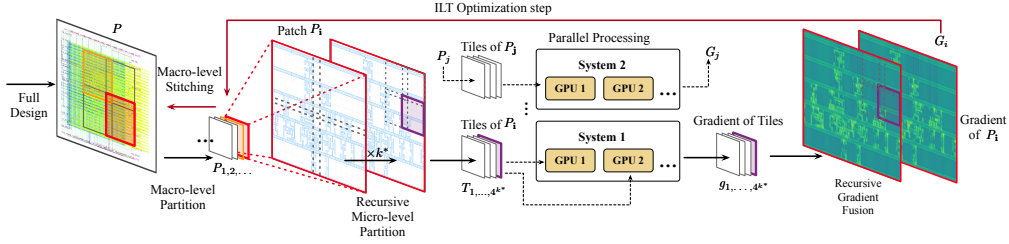
- **Memory Limitation**: The mask/target pixel image exceeds the capacity of main memory for storage.
- **Computation Limitation**: Simulation or optimization processes are computationally intensive.

Lithography defects:



New Direction: Full-Chip ILT[11]

Full Chip ILT System:

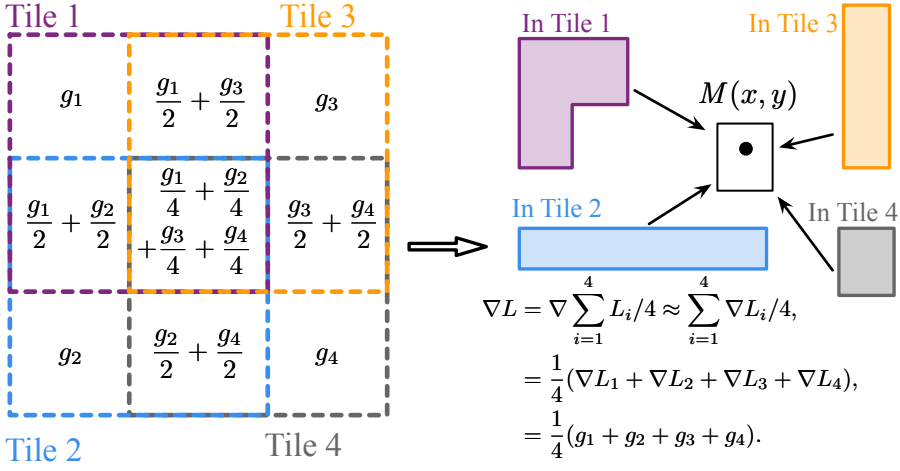


New challenges: Partitioning; Distributed computing; Boundary healing ...

[11] **Shuo Yin et al. (2024). "FuILT: Full chip ilt system with boundary healing".** In: *Proc. ISPD*, pp. 13–20.

New Direction: Full-Chip ILT[12]

Gradient Fusion:



[12] **Shuo Yin et al. (2024). "FuILT: Full chip ilt system with boundary healing".** In: *Proc. ISPD*, pp. 13–20.

Needs of Efficient Differentiable Lithography Simulator

TorchLitho 2.0[13]

User-friendly, efficient, and scalable differentiable lithography simulation engine based on PyTorch for large-scale layouts.

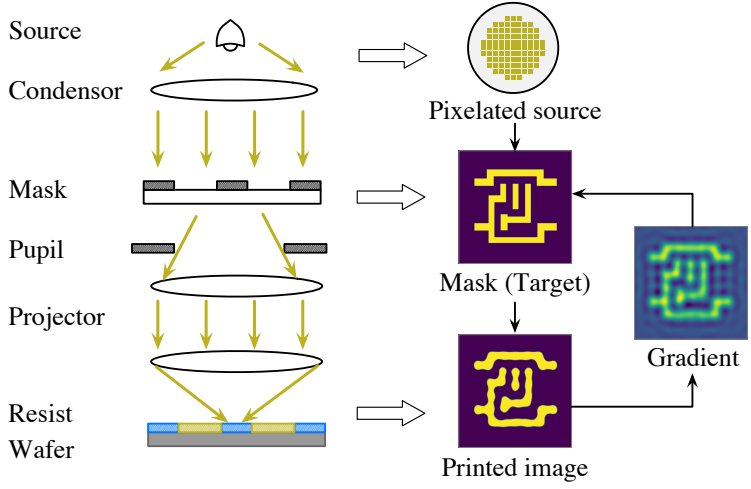
Table: Overview comparisons between TorchLitho 2.0 and existing lithography simulator frameworks.

Model	Hopkins	Abbe	Custom Parameters	Large Layout	Runtime
ICCAD-13	✓	✗	✗	✗	fast
TorchLitho	✓	✓	✓	✗	slow
TorchLitho 2.0	✓	✓	✓	✓	fast

[13] https://github.com/OpenOPC/TorchLitho-Lite

TorchLitho 2.0

Optical System Modeling



Schematic diagram of differentiable lithography system.

Abbe Imaging:

$$I(x,y) = \iint \tilde{J}(f,g) \left[\left| \iint \tilde{H}(f+f', g+g') \tilde{O}(f',g') e^{2\pi i[f'x+g'y]} \, df' \, dg' \right|^2 \right] df \, dg. \tag{1}$$

TorchLitho 2.0: Source-level parallel integration for fast Abbe imaging.

Hopkins Imaging:

$$I(x,y) = \iint \tilde{O}(f',g') \, \tilde{O}^*(f'',g'') \, TCC(f',g',f'',g'') e^{i2\pi[(f'-f'')x+(g'-g'')y]} \, df' \, df'' \, dg' \, dg'', \tag{2}$$

where the TCC is defined as

$$TCC(f',g',f'',g'') = \iint \tilde{J}(f,g) \, H(f'+f, g'+g) H^*(f''+f, g''+g) \, df \, dg. \tag{3}$$

TorchLitho 2.0: TCC matrix generation using randomized SVD.

TorchLitho 2.0 APIs

- **AutoGrad**: TorchLitho provides a customized backward propagation mechanism that supports automatic differentiation.
- **Dynamic Size**: It supports variable mask sizes within the same physical environment.
- **Batching**: The batching axis is designed to fully utilize GPU resources.

Param	Type	Description	Default
Pixel	`int`	Pixel size of the mask (unit: *nm*).	14
Sigma	`float`	Source broadening parameter σ.	0.05
NA	`float`	Numerical Aperture.	1.35
Wavelength	`int`	Illumination wavelength (unit: *nm*).	193
Defocus	`List[int]`	List of defocus values.	None
Parallel	`bool`	Enable parallel mode for Abbe imaging simulation (performance only; no physical change).	False
Batch	`bool`	Enable a batch dimension to simulate multiple masks at once (inputs must include a batch axis).	False

Table: Parameter descriptions and defaults for the simulation.

Simulation Result

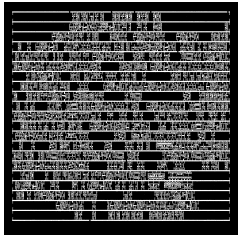
Visualization on ICCAD-13 Benchmarks

The first row shows the input mask patterns, the second row shows the corresponding aerial images, and the third row shows the gradient map of the lithography process.

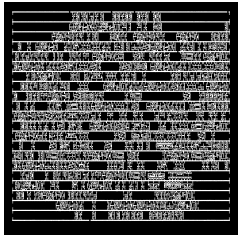
Experimental results on the ICCAD-2013 contest benchmark with 1 nm^2 pixel size.

Visualization on Full-chip Layouts

The printed images are generated through direct simulation without any partitioning.



(a) Mask of metal layer



(b) Printed image of metal layer

(c) Mask of via layer

(d) Printed image of via layer

Experimental results on the full-chip mask of the GCD design, where the mask pattern is optimized using FuILT[14].

[14] **Shuo Yin et al. (2024). "FuILT: Full chip ilt system with boundary healing".** In: *Proc. ISPD*, pp. 13–20.

OpenFuILT[15]

The open-source repository of FuILT based on TorchLitho 2.0[16].

The logo of OpenFuILT.

`TorchLitho` link

`OpenFuILT` link

OpenFuILT-Eval[17]: An open-source evaluation framework for full-chip scaled OPC.

[15] https://github.com/OpenOPC/OpenFuILT
[16] https://github.com/OpenOPC/TorchLitho-Lite
[17] https://github.com/OpenOPC/OpenFuILT-Eval

THANK YOU!