ASU Arizona State University | ASU Center for Semiconductor Microelectronics (ACME) | NVIDIA | Rensselaer

Apollo: Automated Routing-Informed Placement for Large-Scale Photonic Integrated Circuits

Hongjian Zhou[1], Haoyu Yang[2], Nicholas Gangi[3], Haoxing Ren[2], Rena Huang[3], Jiaqi Gu[1]

[1]Arizona State University, [2]NVIDIA, [3]Rensselaer Polytechnic Institute

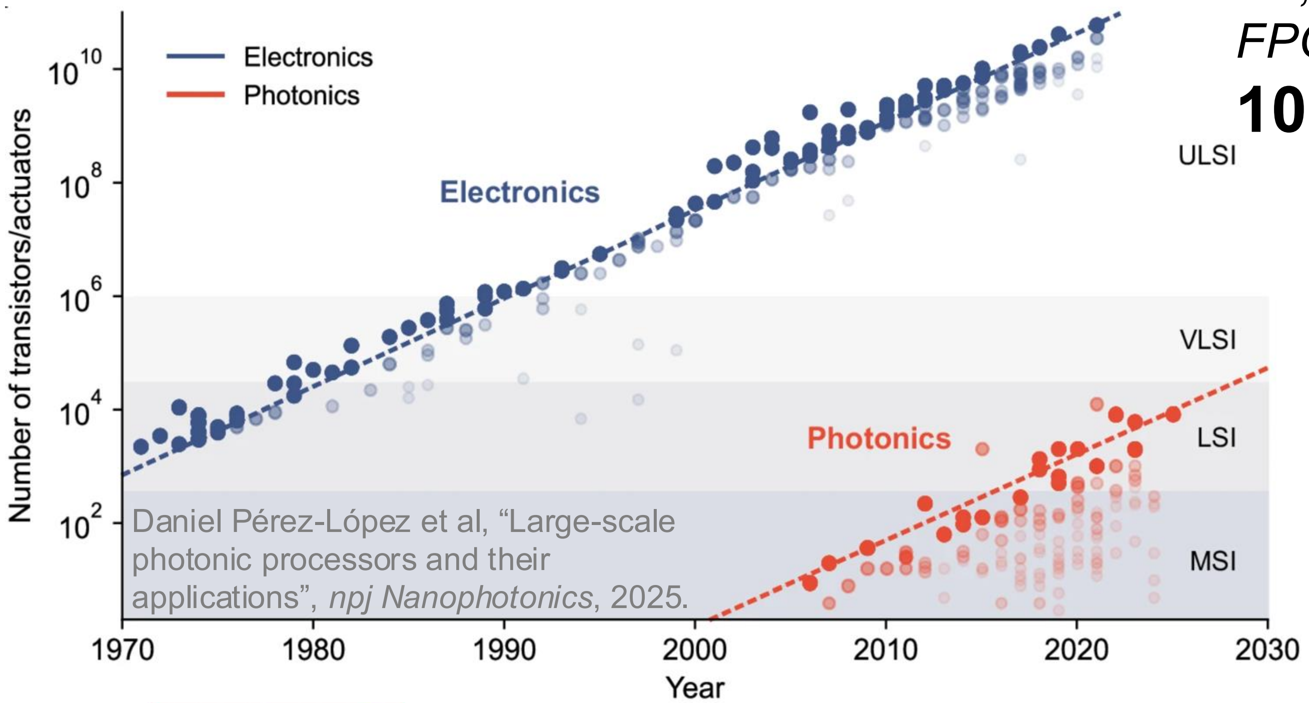
hzhou144@asu.edu

jiaqigu@asu.edu | scopex-asu.github.io

APOLLO

SCOPE

VLPI + EPDA Era

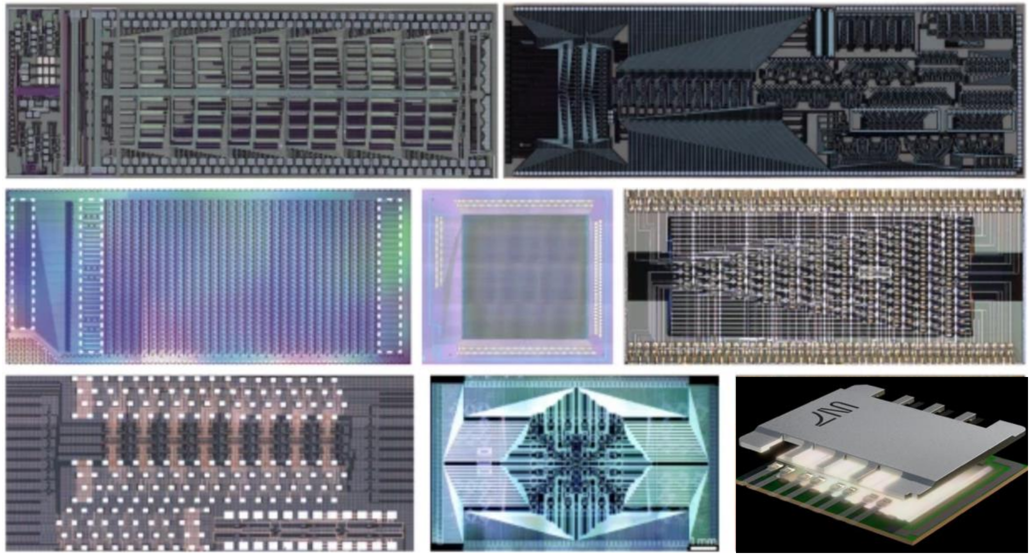


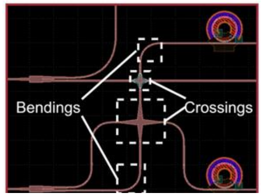
Daniel Pérez-López et al, “Large-scale photonic processors and their applications”, *npj Nanophotonics*, 2025.

Very Large-scale Photonic integration (VLPI) Era

RF, switching, interconnect, LiDAR, beamformer, photonic FPGA, photonic computing, heterogeneous 3D EPIC…

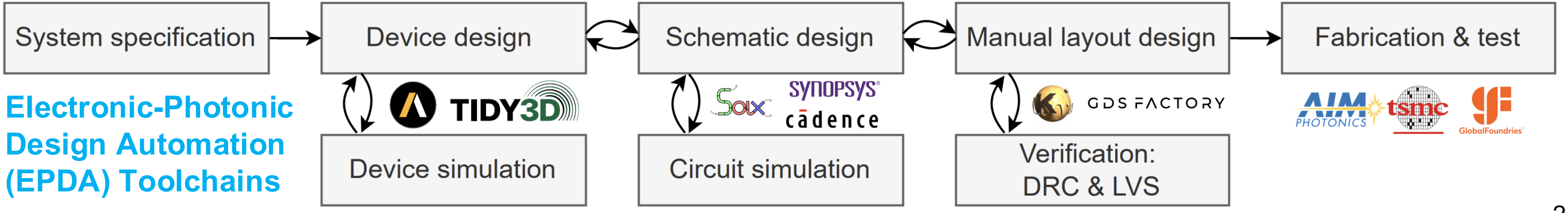
100~10k components and beyond





Heavily relies on manual design in several months!

Time-consuming & Not scalable for large-scale EPICs



System specification → Device design ⇄ Schematic design ⇄ Manual layout design → Fabrication & test

Device design ⇅ Device simulation

Schematic design ⇅ Circuit simulation

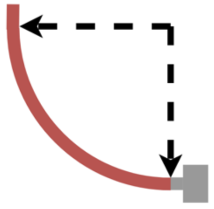
Manual layout design ⇅ Verification: DRC & LVS

Electronic-Photonic Design Automation (EPDA) Toolchains

H. Zhou, P. Ma, J. Guet al, “Toward Intelligent Electronic-Photonic Design Automation for Large-Scale Photonic Integrated Circuits: from Device Inverse Design to Physical Layout Generation”, *SPIE ODA*, 2025.

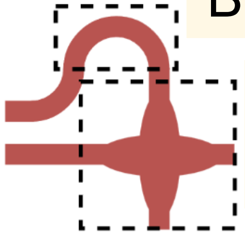
What Makes PIC Placement Different

- Waveguide routing is highly sensitive to component placement
  - Curvilinear structure and **highly space-consuming**
    - Curvy bend with **minimum bend radius**
    - Waveguide crossing: **90°** intersection in the same layer



Bend radius

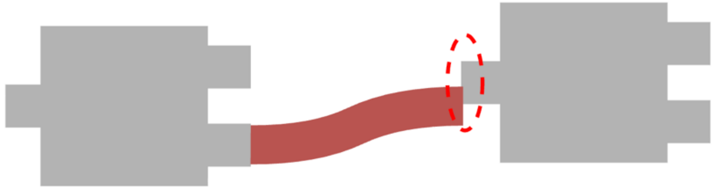
e.g. 5 μm



Bend

Crossing component

e.g. 20×20 μm^2



Port misalignment

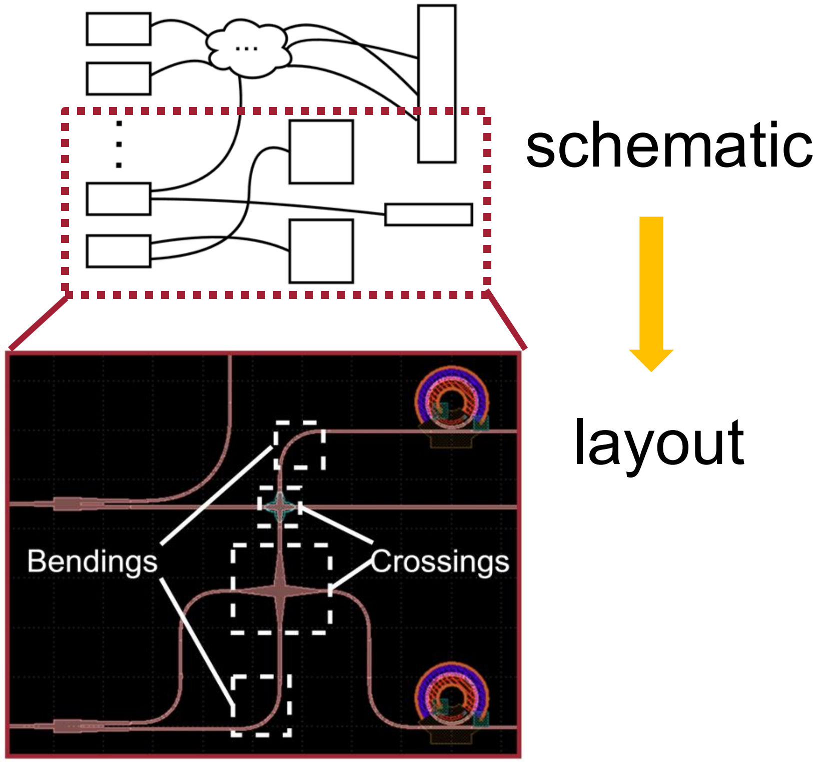
Waveguide should satisfy the bend radius

  - Directional port alignment
    - Inappropriate placement leads to misalignment
  - Limited routing layer: e.g. one silicon layer
    - Share the same silicon layer with component: routing resource contention

Routability highly depends on the placement solution!

Current PIC Placement Solutions

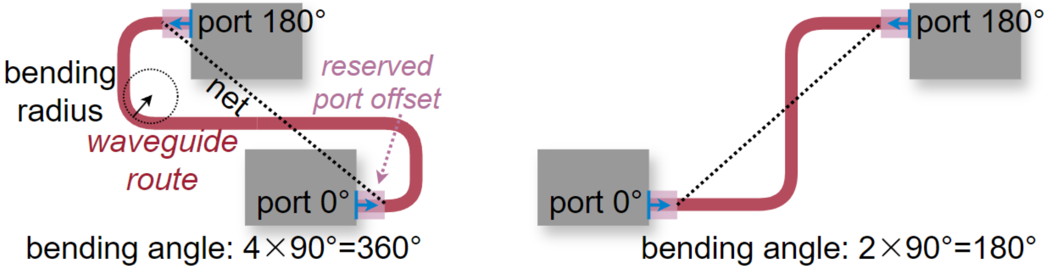
- Schematic-Driven Manual Layout: Place and route simultaneously
  - Manually *plan routing* paths in schematic
  - Path is also treated as photonic components
    - Segment, bend, crossing
  - Place each component carefully
    - Spacing constraint
    - Alignment constrain
  - Need Back-and-forth modifications
  - ☹ Time-consuming & not scalable

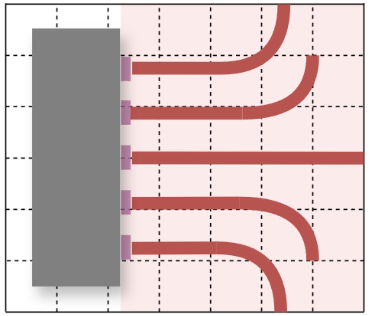


- Existing automated PIC placement works
  - PLATON [Beuningen+, ISPD’16]: minimize crossings
  - PlanarONoC [Chuang+, DAC’ 18]: planar graph-based method
  - CPONoC [Chen+, ASP-DAC’25]: introduce device flipping and rotation
  - ☹ Fail to account for critical routability considerations -> **illegal layout**
    - Bend radii, port accessibility, and area overhead of crossings

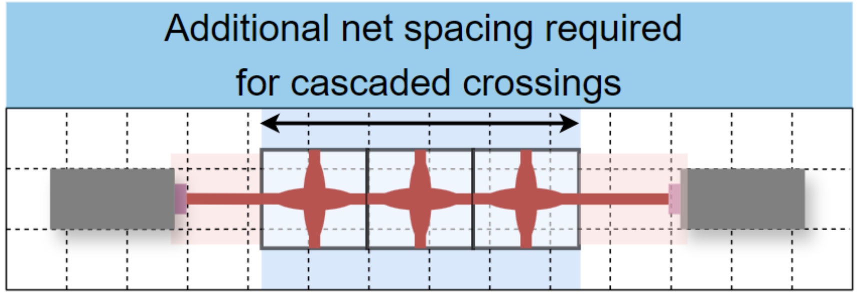
What We Need in Automated PIC Placement?

- Aware of **port orientation** and accessibility
  - Improper port location introduces excessive bends and detours
  - Satisfy basic spacing demand for device port access

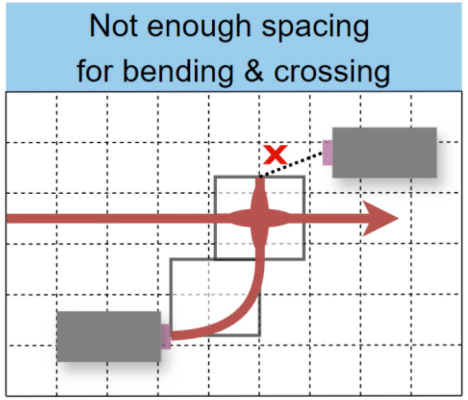




- Aware of spacing needed by bends and crossings

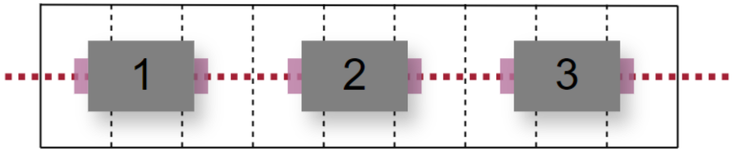


Additional net spacing required for cascaded crossings



Not enough spacing for bending & crossing

- Satisfy the user-defined constraints
  - E.g., component alignment constraints
  - Reduce crosstalk and achieve phase matching



Y-center alignment

Proposed PIC Placer: Apollo

- How to be aware of **port orientation** during placement?
  - *Sol:* Asymmetric bending-aware wirelength

- How to **save spacing** for bending and crossings?
  - *Sol:* Routing-informed net spacing model

- How to satisfy the **designer constraints**?
  - *Sol:* Conditional projected gradient descent

- How to stabilize the **mixed-size PIC** placement?
  - *Sol:* Blockwise adaptive Nesterov optimizer

Placement Formulation with Designer Constraints

$$\min_{x,y} \sum_{e \epsilon E} WL(e;x,y) + NS(e;x,y) + D(x,y)$$

Minimize waveguide lengths

Optimize spacing

Spread cells in electrostatic fields

$s.t.$ alignment constraints

$WL(e;x,y)$: Wirelength function (reduce bend)

$NS(e;x,y)$: Routing-informed net spacing model (improve routability)

$D(x,y)$: component density function (ePlace)

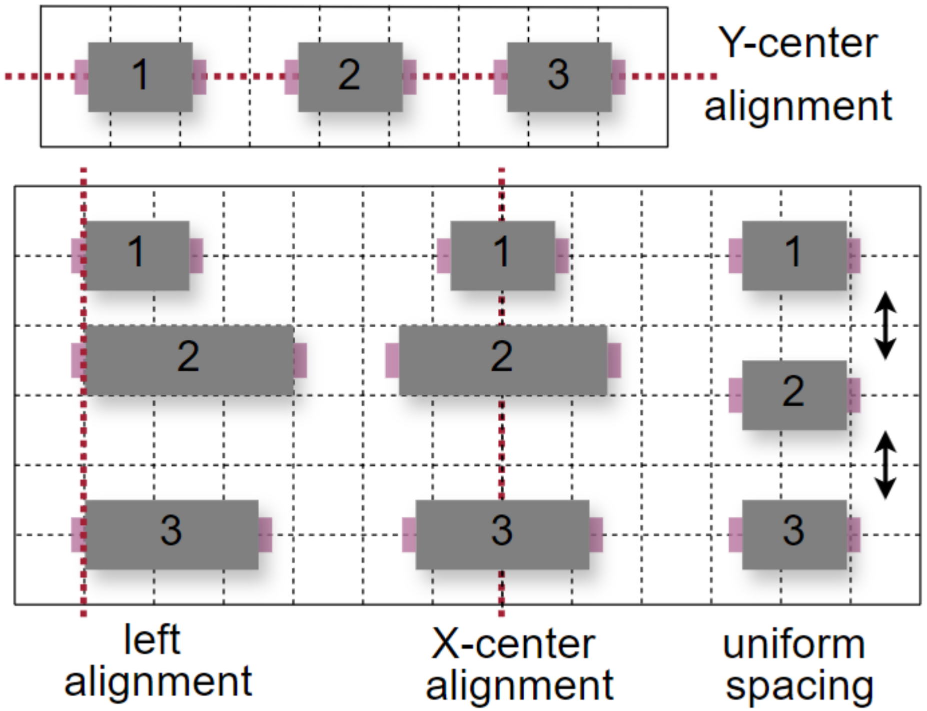
- Enforce constraints by conditional projected gradient descent
  - Apply *soft projection* before position update
  - Gradually *tighten* constraints

$$x_i = (1 - s_t) \cdot x_i + s_t \cdot x_{new,i}$$

$$s_t = s_0 + (s_T - s_0) \cdot \frac{1 - \cos({}^{\pi t}/_{T})}{2}$$

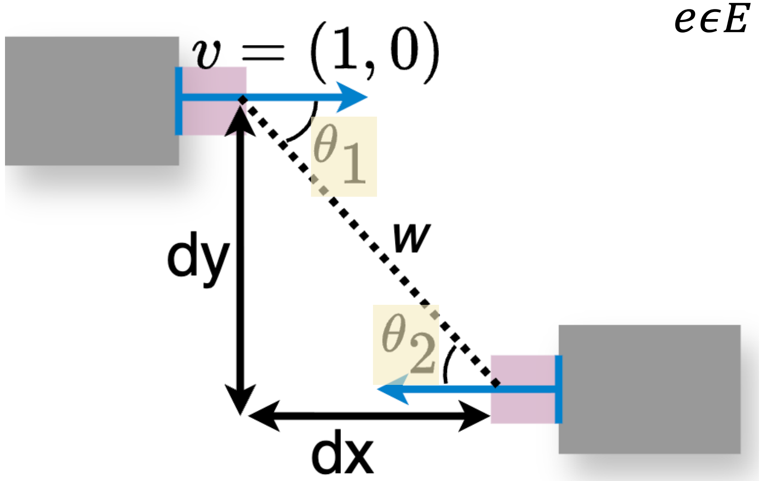
  - e.g., Alignment: move toward avg. locations

$$x_{new,i} \leftarrow \frac{1}{|G|} \sum_{j \in G} x_j$$



Asymmetric Bending-Aware Wirelength

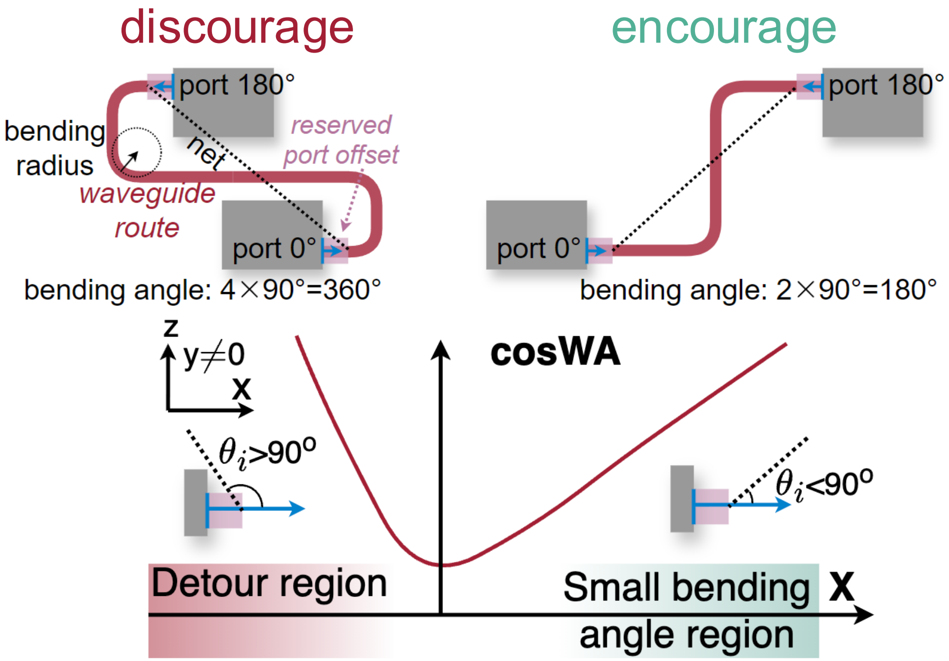
$$\min_{x,y} \sum_{e \epsilon E} WL(e;x,y)$$



- We augment standard weighted-average wirelength to reduce wire bending

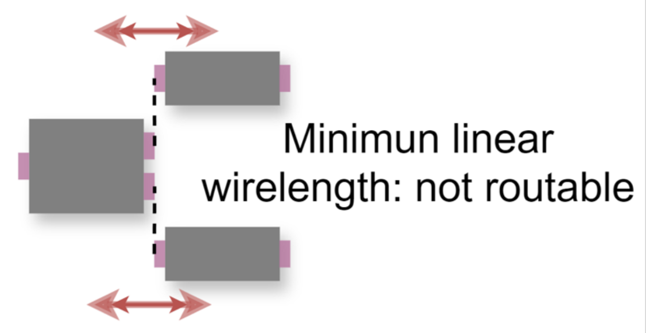
$$\cos WA_x = (1+\boldsymbol{W_\theta})\left(\frac{\sum_{i\in e} x_i e^{\frac{x_i}{\gamma}}}{\sum_{i\in e} e^{\frac{x_i}{\gamma}}} - \frac{\sum_{i\in e} x_i e^{-\frac{x_i}{\gamma}}}{\sum_{i\in e} e^{-\frac{x_i}{\gamma}}}\right)^{\alpha}$$

$$\boldsymbol{W_\theta} = [(c - cos\theta_1)_+]^2 + [(c - cos\theta_2)_+]^2$$



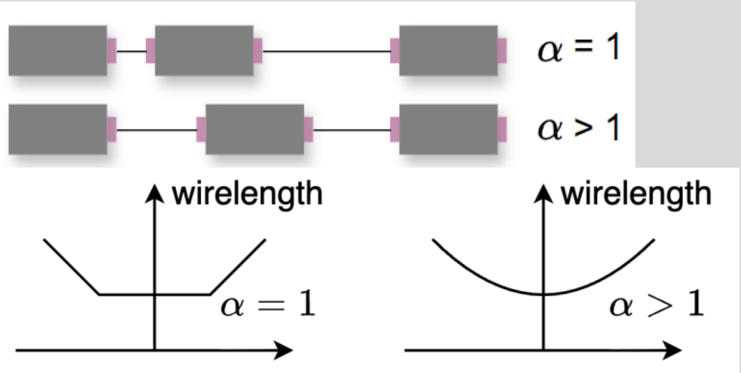
Q1. Why need $(1+\boldsymbol{W_\theta})$?

Need to consider bending



Q2. Why α exponent ?

Encourage symmetry



Routing-Informed Net Spacing Model

$$\min_{x,y} \sum_{e \epsilon E} NS(e; x, y)$$

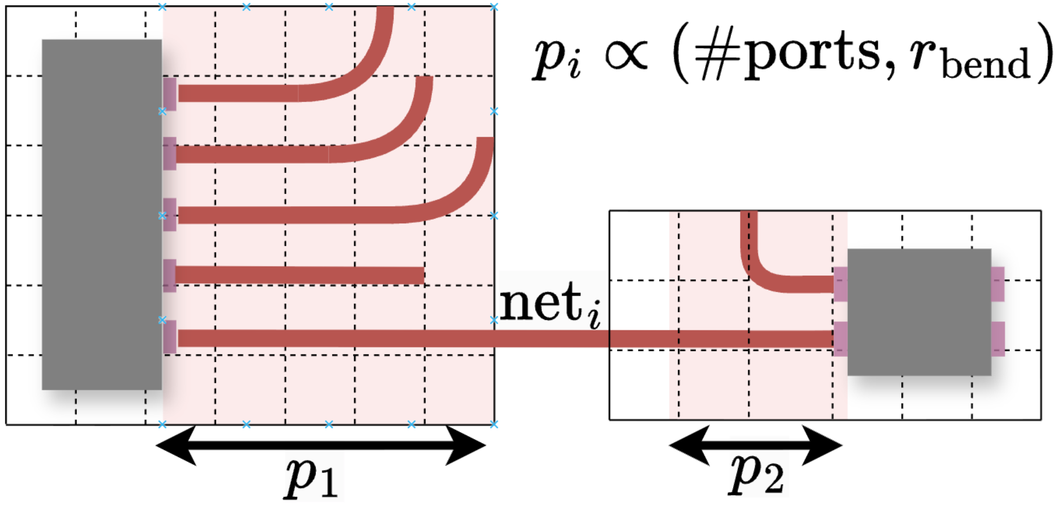
- Root cause for routing failure: *port accessibility, crossing insertion*
- ***Estimated net spacing*** = **basic spacing** + **crossing spacing**

Basic spacing for $net_i = sum(p_1, p_2)$

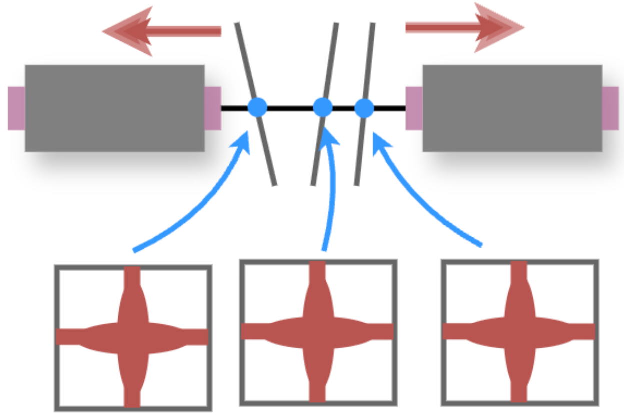
Crossing spacing for $net_i = \#cross \times cross_{size}$

Estimate by port count and bending radius

Estimate **#cross** by # of wire intersections every 100 iterations



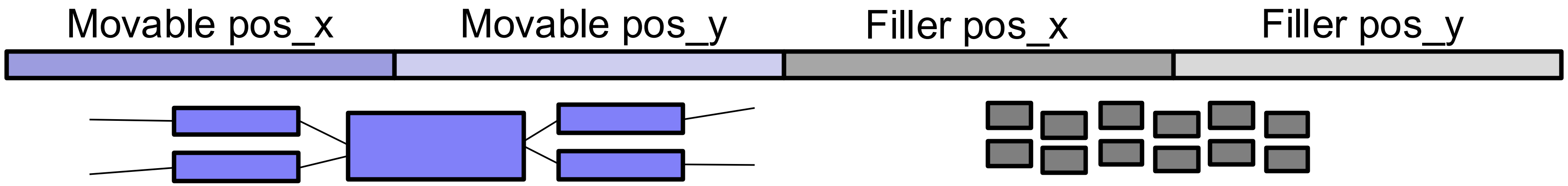
+



3 crossings to be inserted

Blockwise Adaptive Nesterov Optimizer

- Challenge for optimization stability and convergence
  - Significant **heterogeneity in cell sizes…**
    - Mach-Zehnder modulators (1000× $100\ \mu m^2$); filler (10× $5\ \mu m^2$)
  - *Idea*: decouple large cell updates from small cells
  - *Sol*: independent Barzilai-Borwein step for different variables
    - 4 variable blocks: movable instances and dummy fillers in x and y directions
    - Encourage faster convergence of filler cells to surround movable instances

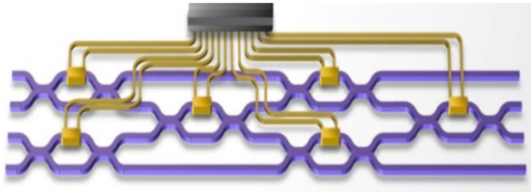


- Stabilize the optimization when cells are near-optimal
  - *Sol*: global cosine annealing schedule
    - Gradually reduce the effective step size over time

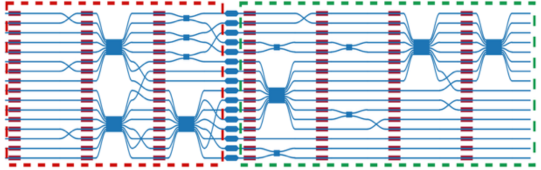
Experimental Setup

- Machine
  - AMD EPYC 7763 Linux server with NVIDIA RTX A6000 GPU
- Benchmark: photonic computing unit
  - Clements: MZI array for matrix multiply [Shen+, NatPhoton’17]
    - **Regular structure, no crossing**
  - ADEPT: auto-searched photonic tensor core [Gu+, DAC’22]
    - **Multi-port, Irregular, high density, unavoidable crossings**
  - 2 settings with different area budget:
    - *-S*: **compact die** size with a <u>5 μm bending radius</u>
    - *-L*: **spacious die** size with a <u>10 μm bending radius</u>

Clements

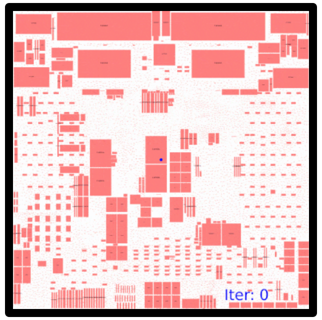


ADEPT

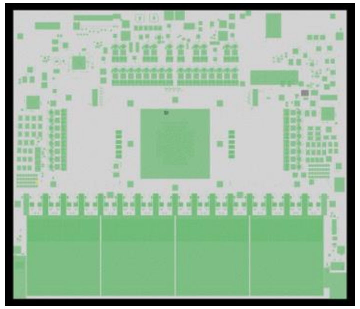


- Placers for comparison

`DREAMPlace`: **VLSI** placer
w/ routability optimization
[Lin+, TCAD’20]

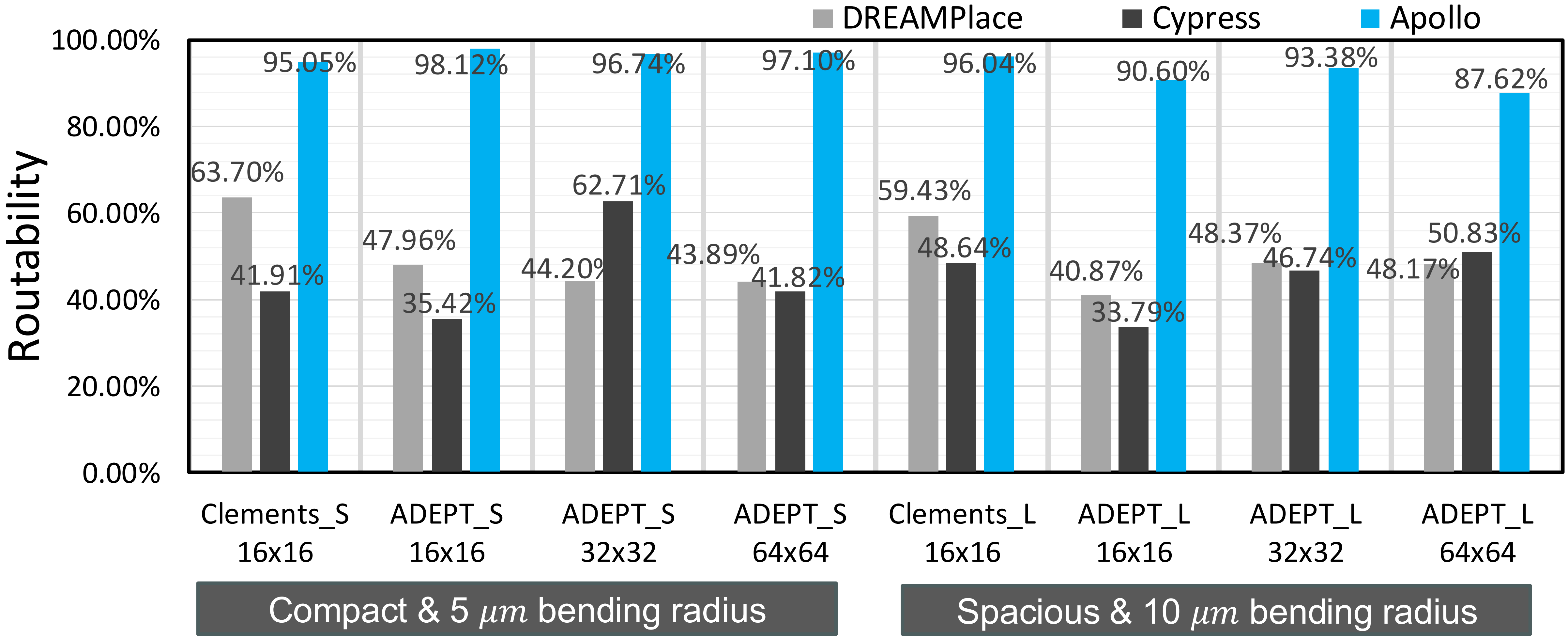


`Cypress`: **PCB** placer
w/ crossing optimization
[Zhang+, ISPD’25]



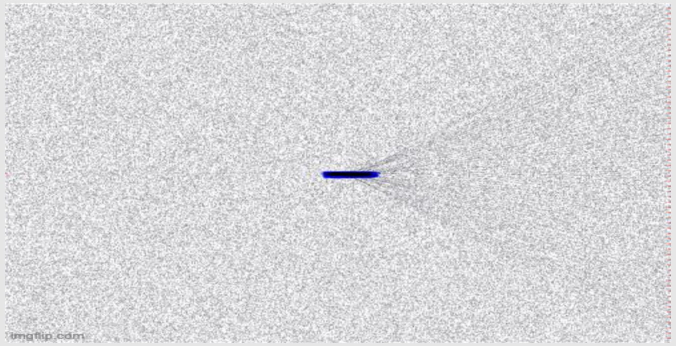
Routability Comparison

- ♦ Apollo outperforms other placers in routability
  - › Waveguide routing using PIC router: **LiDAR** [Zhou+, ISPD’25, TCAD’25]
  - › **94% versus 50.85% (DREAMPlace) and 51.38% (Cypress)**



Apollo Placement Visualization

ADEPT_64x64: #nets=1791, #cells=1201

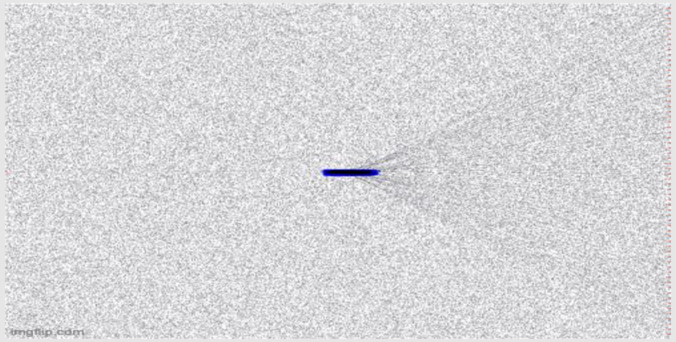


DREAMPlace

235 s

43.9%

nets are routable

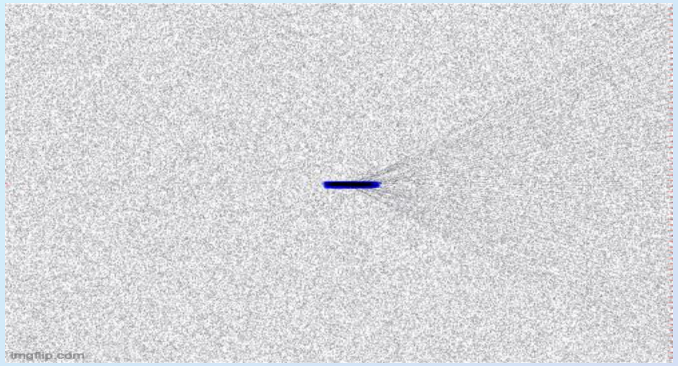


Cypress

133 s

41.8%

nets are routable



Apollo

105 s *43%▼*

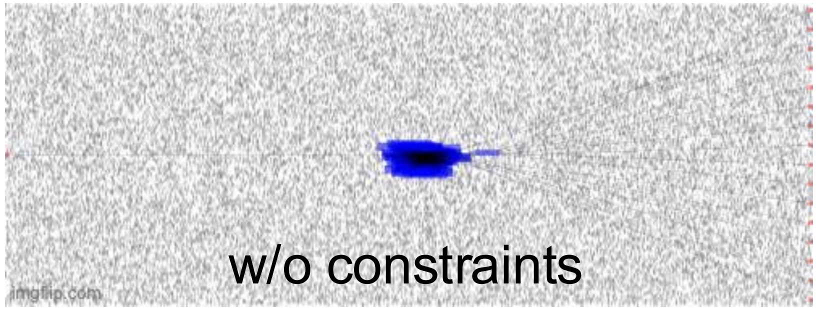
97.1% *54%▲*

nets are routable

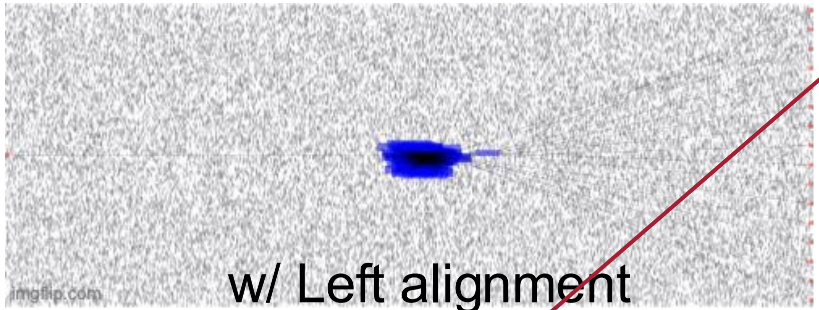
Apollo *w/ different constraints*

🟦 photonic device

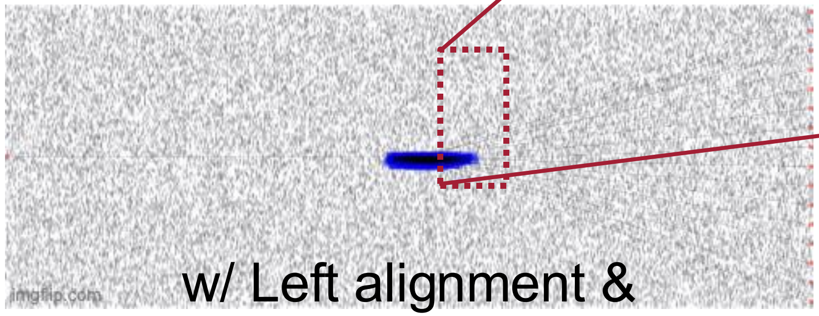
⬛ dummy filler



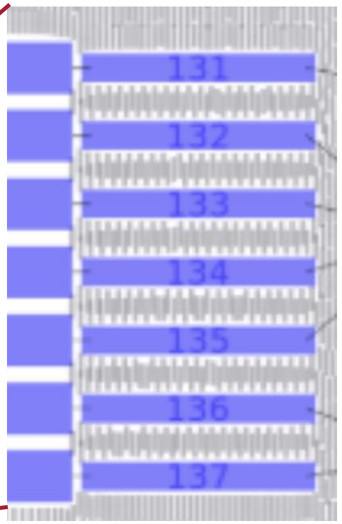
w/o constraints



w/ Left alignment



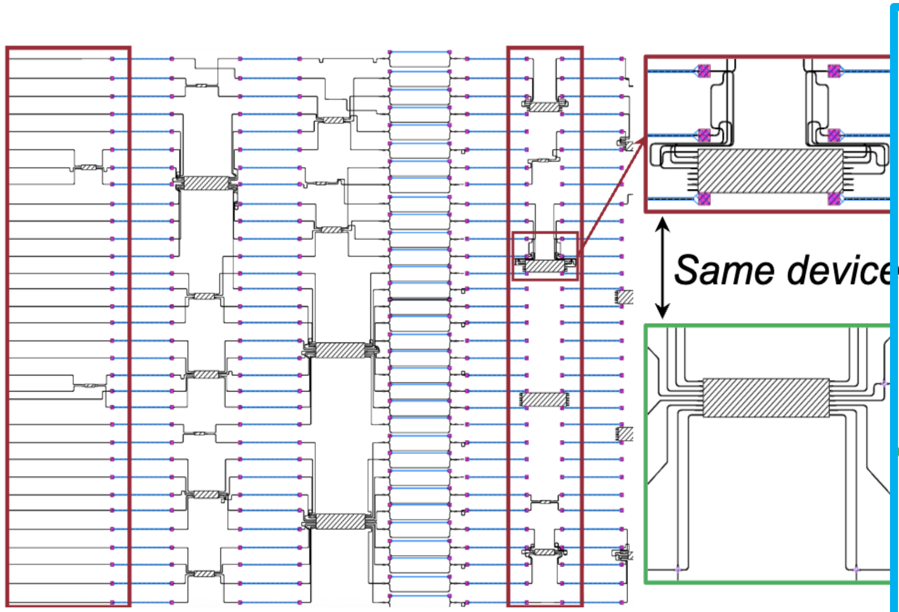
w/ Left alignment & Y-center alignment



Inflate thermal-sensitive components to ensure sufficient spacing

Final GDSII Layout Visualization

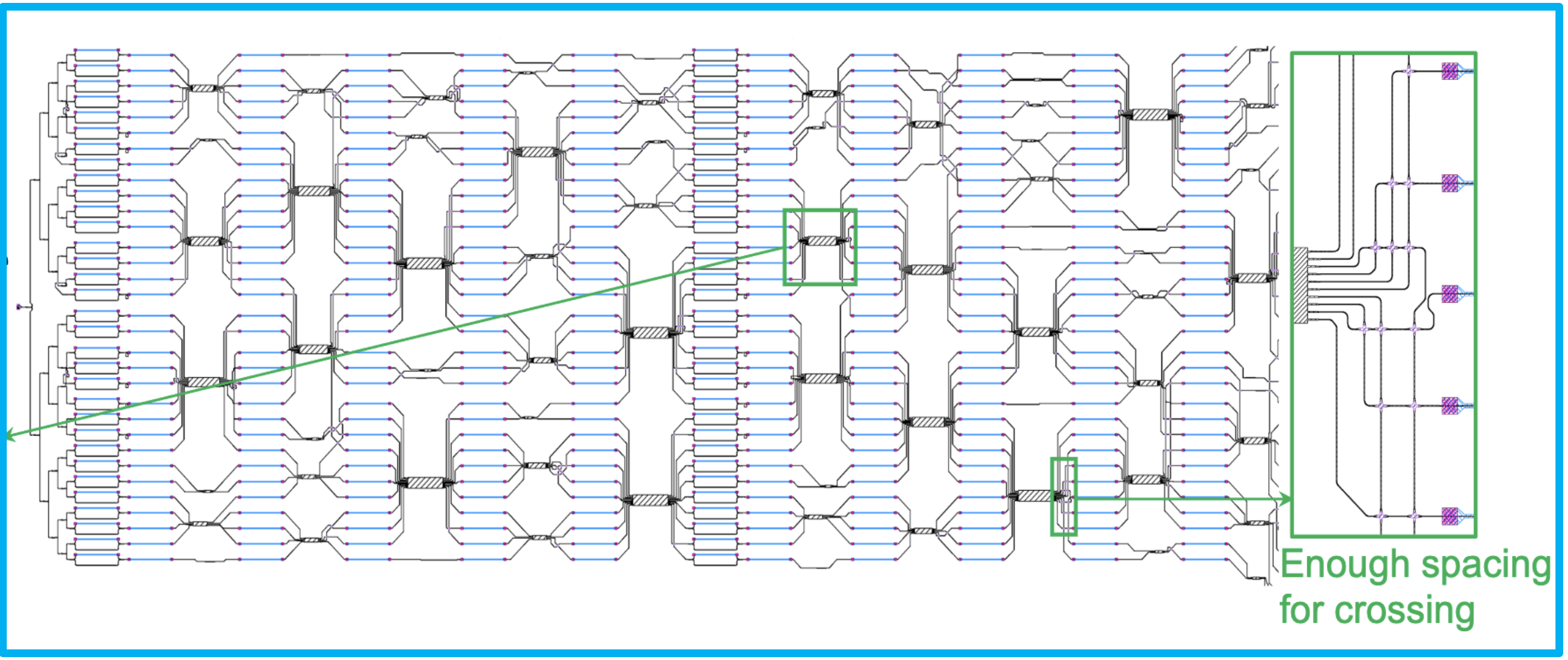
Layout generated by **Cypress**



Inappropriate spacing

Not aware of bending

Layout generated by **Apollo**



Routing-informed placement leads to compact, routable layout

Photonic tensor core: ADEPT_32x32

Routed by our PIC router - **LiDAR** [Zhou+, ISPD’25, TCAD’25]

Open-Source
PIC placer Apollo

Thank you!

Apollo: Automated Routing-Informed Placement for Large-Scale Photonic Integrated Circuits

Hongjian Zhou[1], Haoyu Yang[3], Nicholas Gangi[2], Zhaoran (Rena) Huang[2], Haoxing Ren[3], Jiaqi Gu[1,†]
[1]Arizona State University [2]Rensselaer Polytechnic Institute [3]NVIDIA Corporation
[†]jiaqigu@asu.edu

arXiv Preprint

Automated PIC placement tool
Seamless w/ LiDAR PIC Router

Arizona State University

ASU Center for Semiconductor Microelectronics (ACME)

SCOPE