Arizona State University

ASU Center for Semiconductor Microelectronics (ACME)

LiDAR: Automated Curvy Waveguide Detailed Routing for Large-Scale Photonic Integrated Circuits

Hongjian Zhou[1], Keren Zhu[2], Jiaqi Gu[1]

[1]Arizona State University, [2]Fudan University
School of Electrical, Computer and Energy Engineering
hzhou144@asu.edu
jiaqigu@asu.edu | scopex-asu.github.io

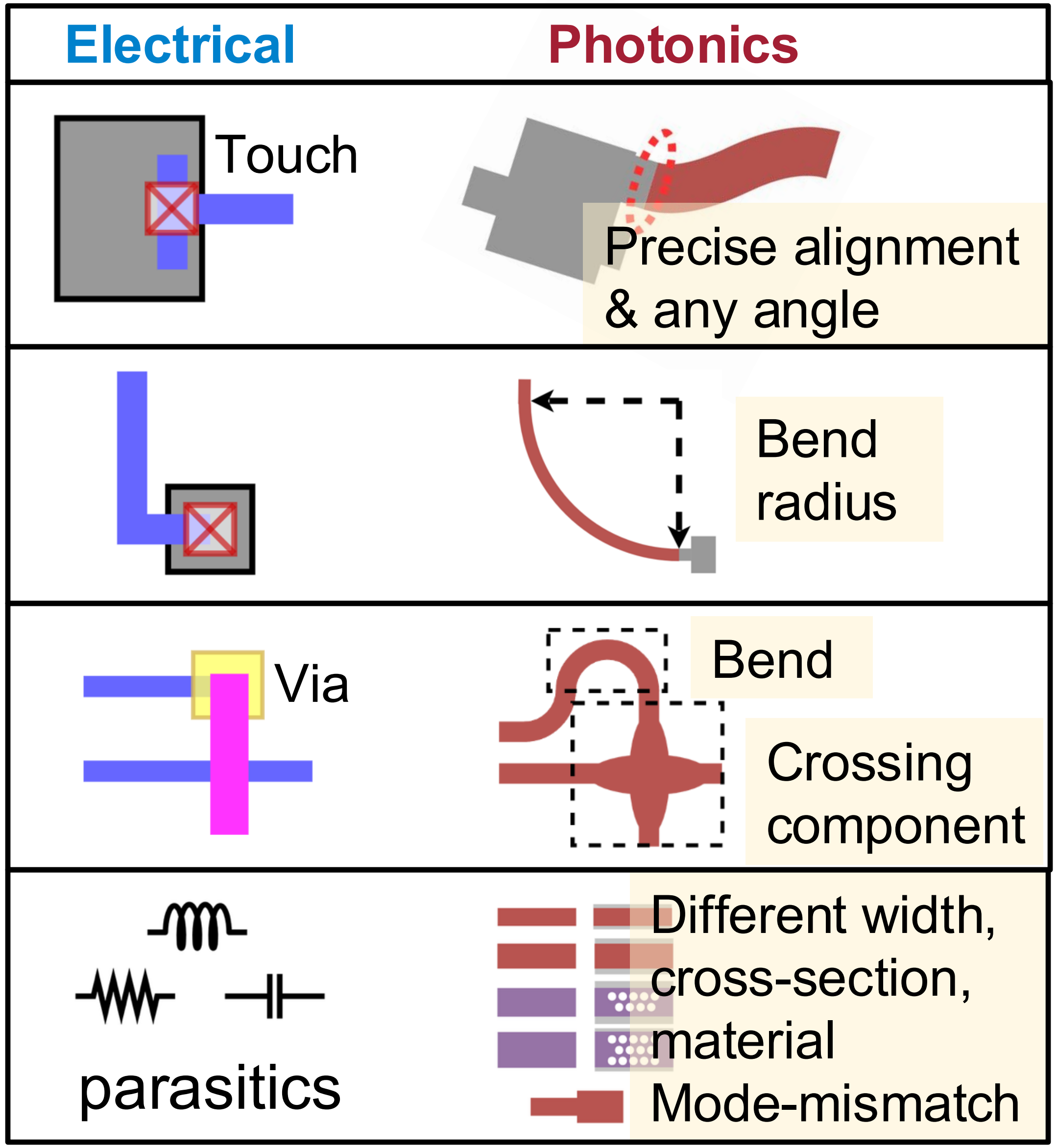
LIDAR

SCOPE

What Makes PIC Routing Different from EIC?

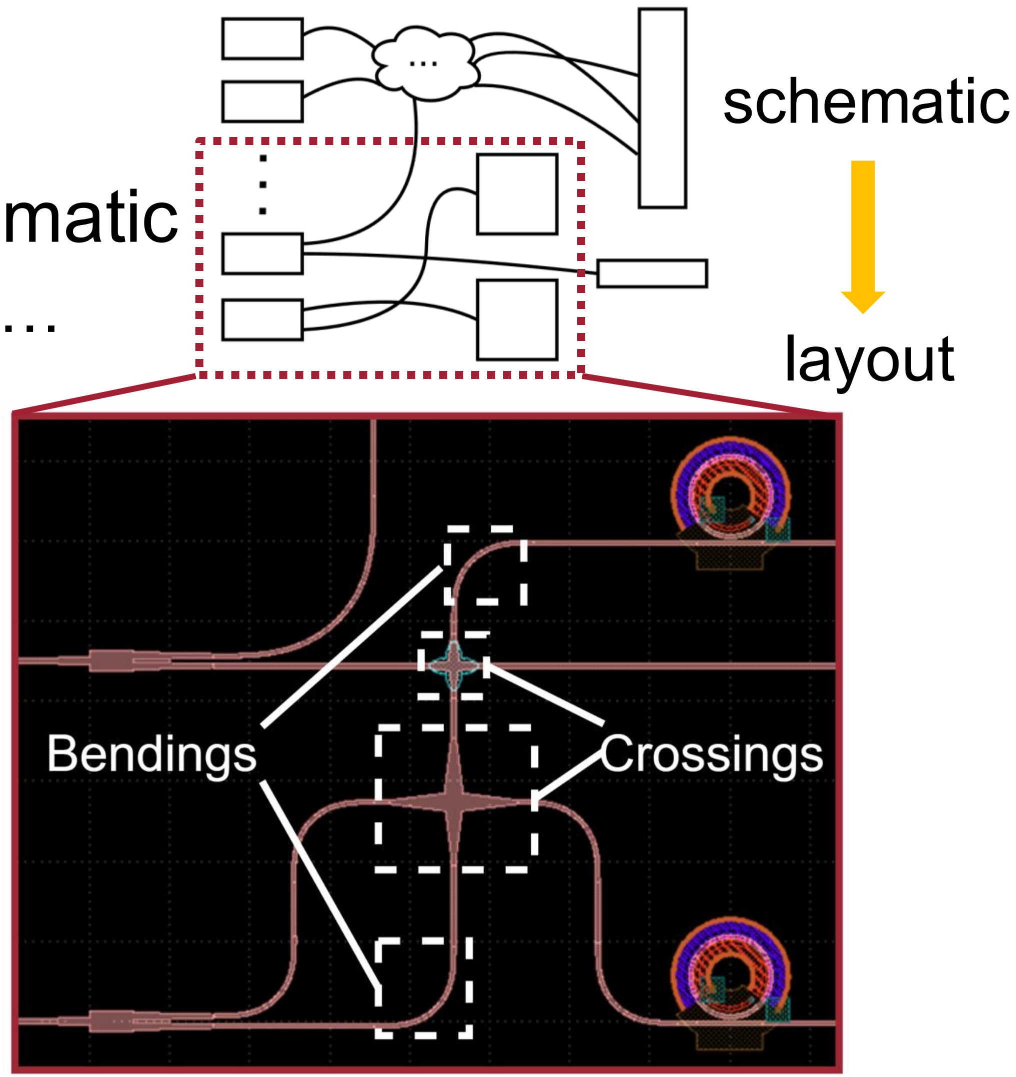
- Port access
  - Need to align **port orientation**
- **Curvy** bend
  - Need additional space
- Crossing (*similar to via*)
  - **90°** intersection in same layer
  - **Area**-consuming
- Signal integrity (*analog/RF nature*)
  - Phase/modal matching
  - Thermal crosstalk
  - ...

Heavily relies on manual design!



How Human Routes Waveguides?

- **Schematic-driven layout**
  - Manually *plan routing* solutions in schematic
    - Even wire crossings need planning ahead…
  - Path is formed by *separate instances*
    - Segment, bending, crossing…
  - Connect each instance *carefully*
    - Bending radius constraint
    - Spacing constraint
    - Alignment constraint…
  - *Back-and-forth* modifications
    - Instances are highly coupled



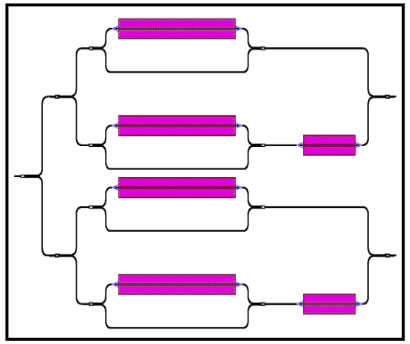
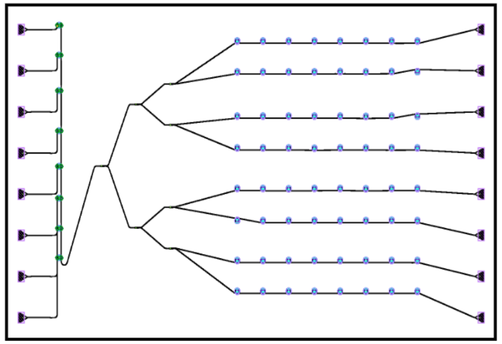
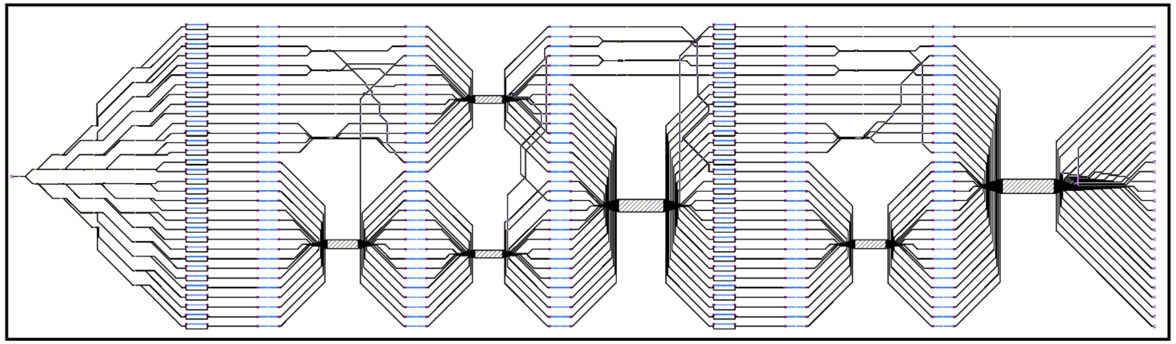
Time-consuming & Not scalable for large-scale PICs

PIC Scale and Design Complexity Grow Rapidly

- From **tens** to **hundreds** of instances/nets
- From **well-structured** designs to **irregular** designs
- From **basic geometry** to **stringent and multi-disciplinary** rules

<2017 · 2017 · ··· · 2024 · ···

IQ modulator · *Micro-ring weight bank* · *Photonic tensors core, system-level interconnects...*

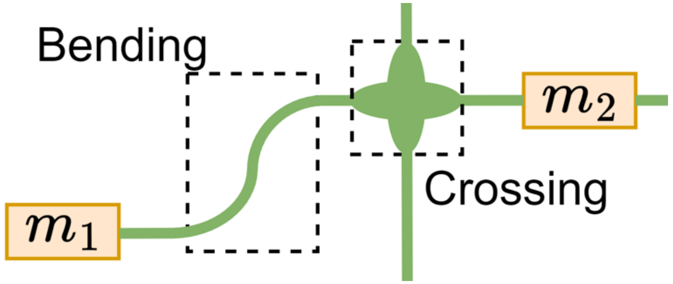
Small-scale, manually routable (~1day)

Large-scale, complex PICs (~weeks)

Time for EPDA! Require **auto detailed routing tool** to increase productivity, efficiency & design quality

What Makes A Good PIC Routing: (Metric and Formulation)

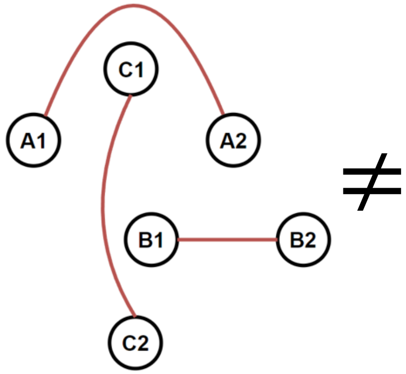
- ***Quality Metric***: minimize **critical-path insertion** loss: IL_{max}
  - *Link budget* is critical to **required laser power & SNR**
  - Path insertion loss = device insertion losses + net insertion losses

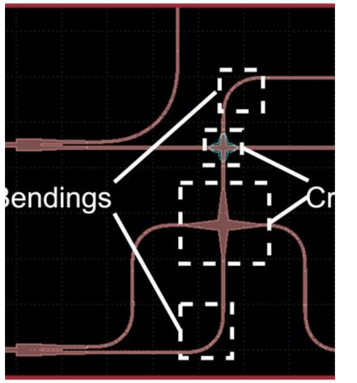


  - Net loss: PIC only has **2-pin** net, net loss contains 3 parts
    - *Straight* waveguide propagation loss: ∝ **path length**
    - Curvy waveguide bending loss: ∝ **bending angle**
    - *Waveguide crossing* loss: ∝ **number of crossings**
- ***Problem formulation***
  - Given a set of nets and placed devices, generate **legal** routing for each net

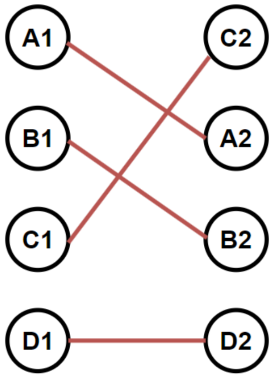
$$\boxed{\min\ IL_{max} \quad \text{s.t. Design rules}}$$

Prior Work and Limitations





- Focus on global route ***planning***:
  - Proton [Boos+, ICCAD'13]: Adaptive crossing penalty
  - ToPro [Zheng+, ICCAD'21]: Dynamic pushing algorithm
  - PlanarNoC [Chuang+, DAC'19]: Introduce flipping and rotation of devices
- ☹ **Overlook physical implementation --- no legal GDS layout generated**
  - **Not aware of curvy waveguides & bending**
  - **Not aware of crossing insertion**
- Photonic detailed channel routing:
  - Manhattan grid-based left-edge method [Condrat+, MWSCAS'12]
  - Non-Manhattan channel routing [Condrat+, SLIP'13]
- ☹ **Cannot optimize #crossing**



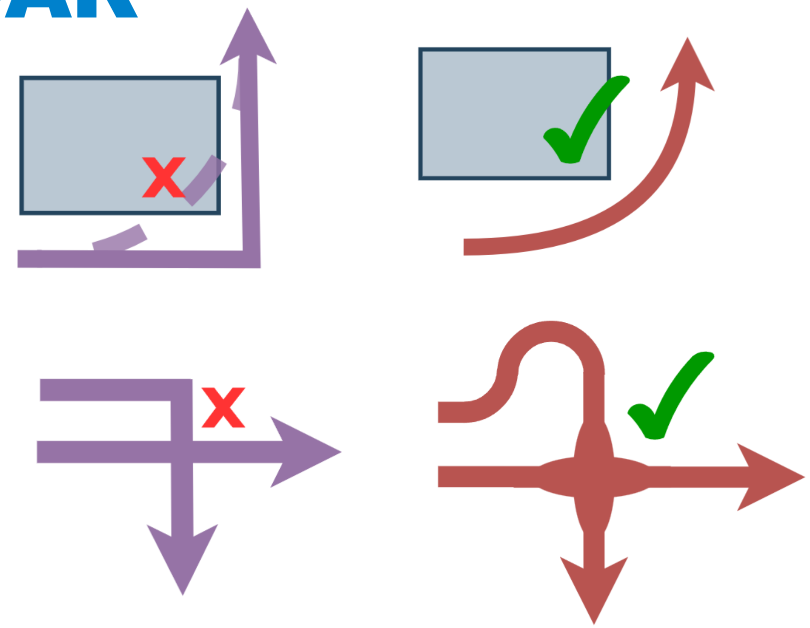
We will fill the gap

generate *implementable* routing solution while *minimizing* IL_{max}

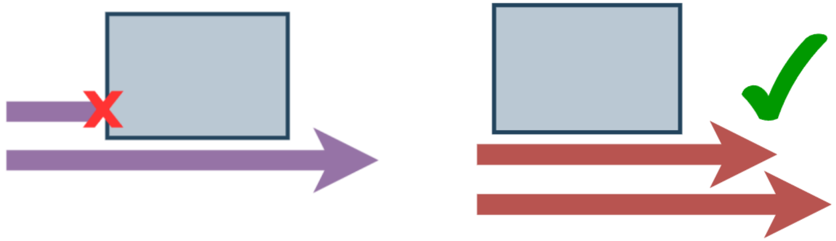
Proposed PIC Detailed Router: LiDAR

- ***How to find a path that is <u>physically</u> implementable?***
  - *<u>Sol:</u>* Curvy-Aware *A** Search
    - **<u>Parametric</u>** neighbors' generation
    - **<u>Dynamic</u>** crossing insertion

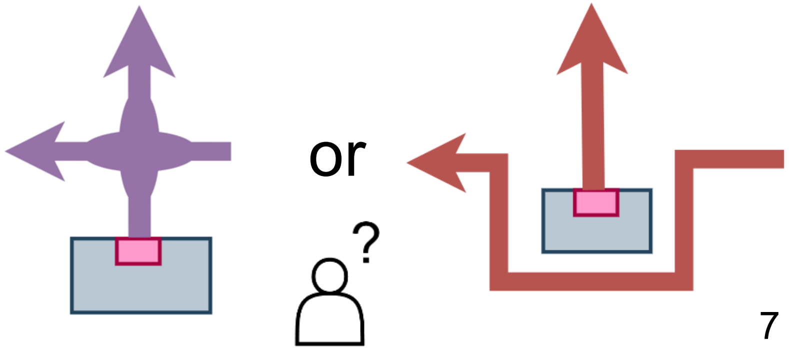
Not enough space for bending & crossing



- ***How to mitigate routing <u>congestion</u> on a single layer?***
  - *<u>Sol:</u>* Reserve routing resource
    - **<u>Predictively</u>** reserve space near ports
    - **<u>Joint</u>** planning for a group of nets

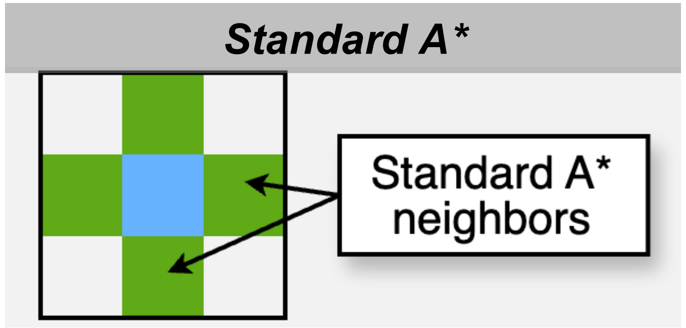


- ***How to balance <u>crossing vs. detour</u>?***
  - *<u>Sol:</u>* Detect & remove undesired crossing

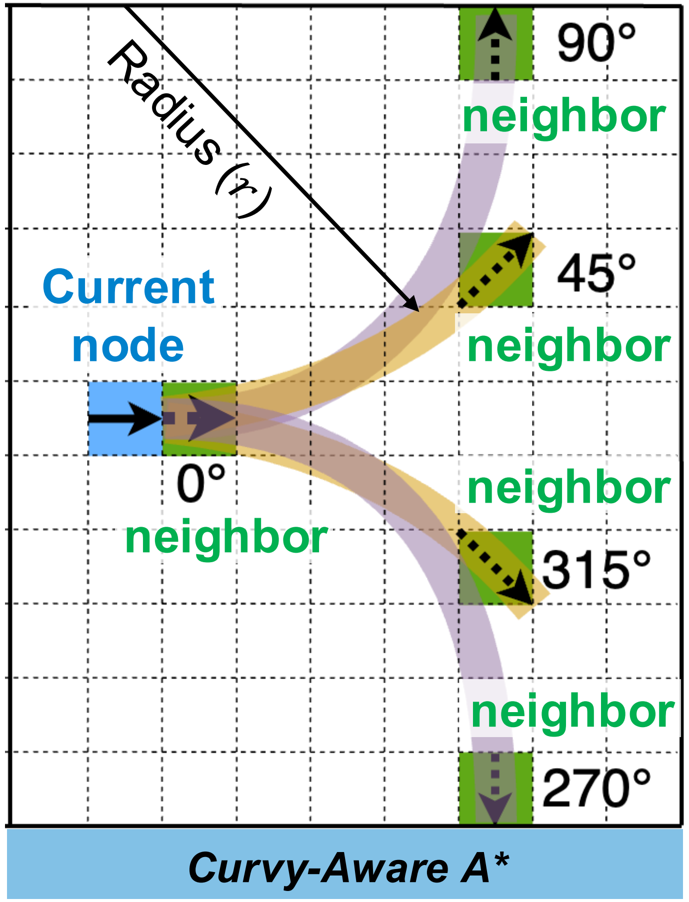


Curvy-Aware A^* Search

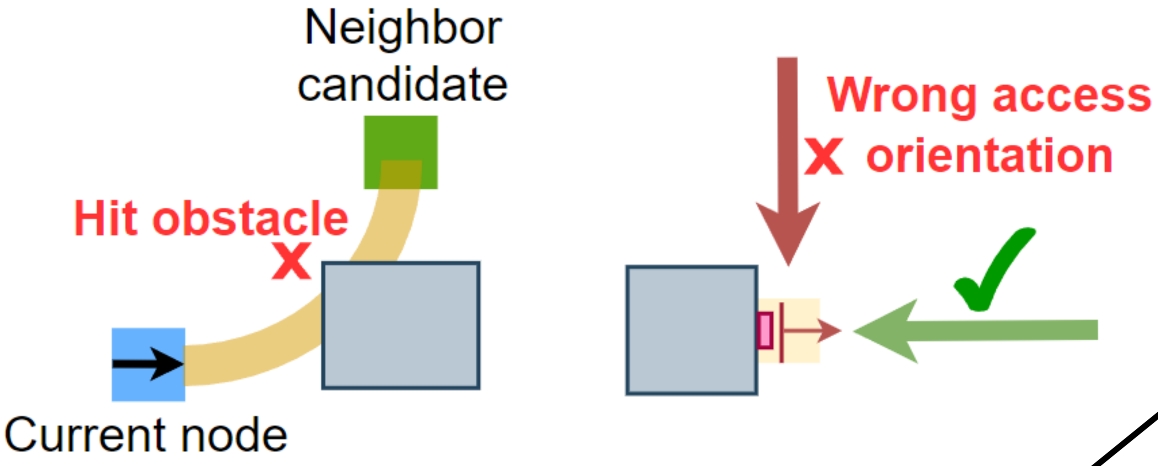
- We *augment* standard $\boldsymbol{A}^*$ search to support <u>curvy</u> waveguide + <u>non-Manhattan</u> routes

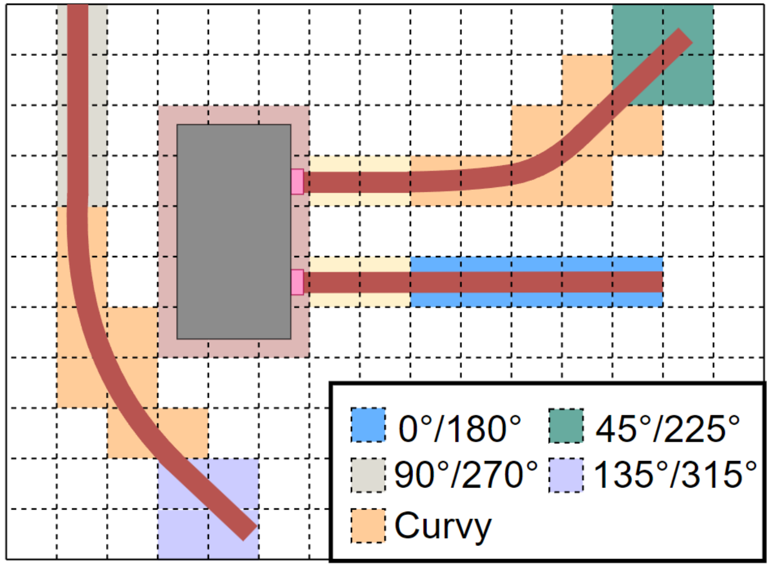


- ***How to find next neighbors to explore?***
- Depend on **current path direction**
  - *<u>Sol</u>*: Extend $\boldsymbol{A}^*$ node state to remember orientation: $(x, y,$ **orientation**$)$
- Depend on **bend radius**
  - *<u>Sol</u>*: redefine curvy-aware neighbors
  - Locations **adaptively calculated** based on:
    - Radius (r) & node direction

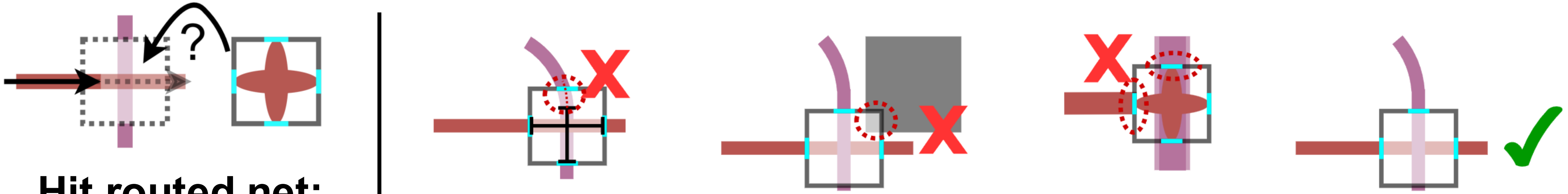


How to Ensure Neighbors’ Legality





- Propose ***oriented*** grid map
  - Ensure legal 90º crossing insertion & correct connection direction



Hit routed net; Insert crossing neighbor?

Not enough straight length

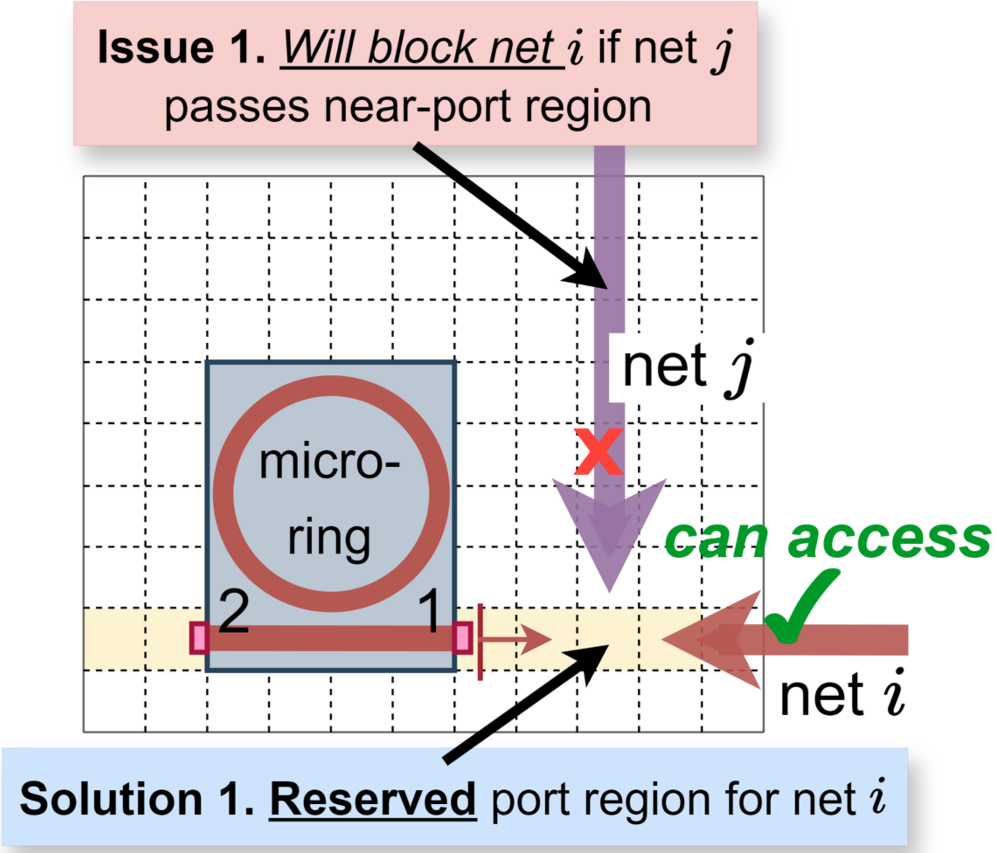
Conflict with blockage

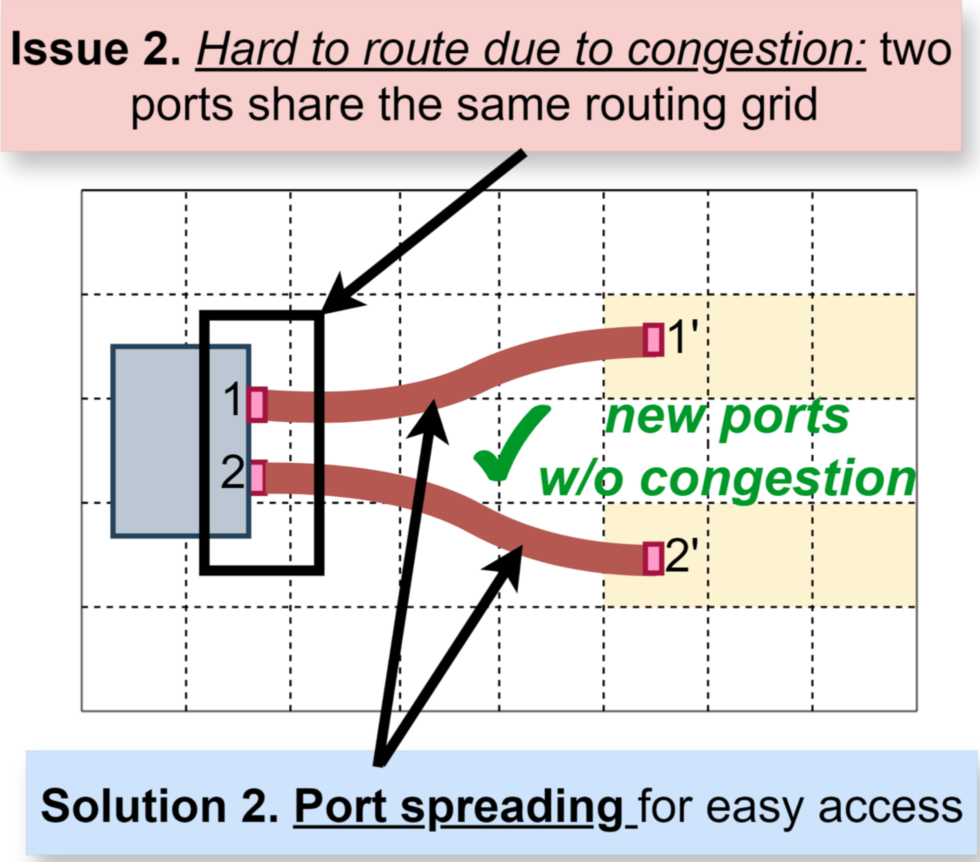
type/size mismatch

Crossing allowed

How to Mitigate Waveguide Routing Conflicts?

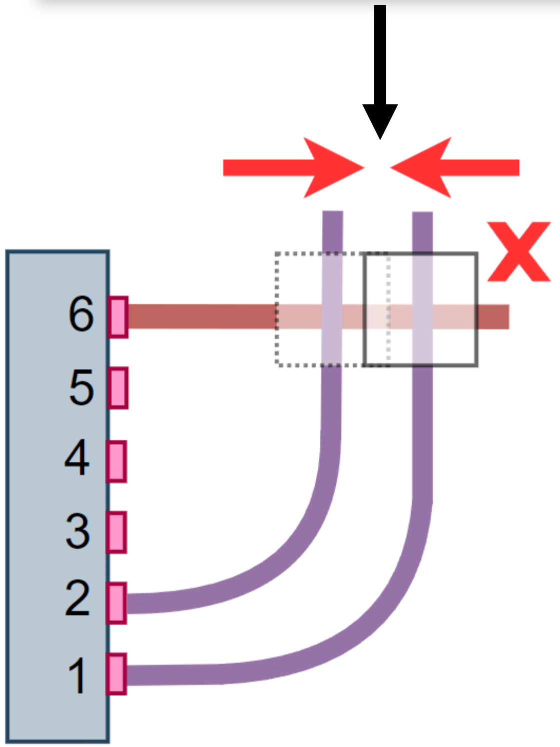
- Waveguide conflict: routing resource competition among waveguides
- ***Predictively reserve*** routing resource **near port regions**

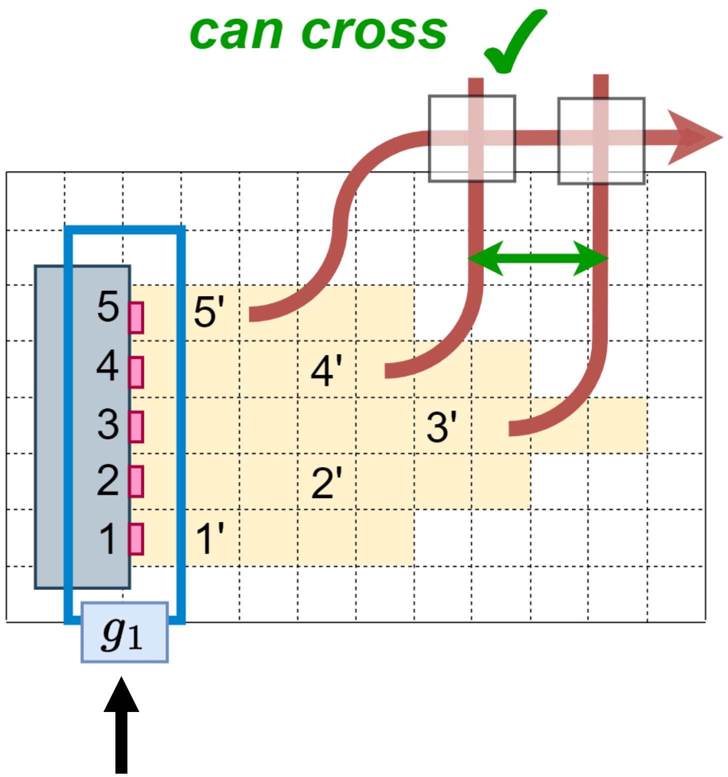




Joint Planning for A Group of Nets: Routability ↑↑

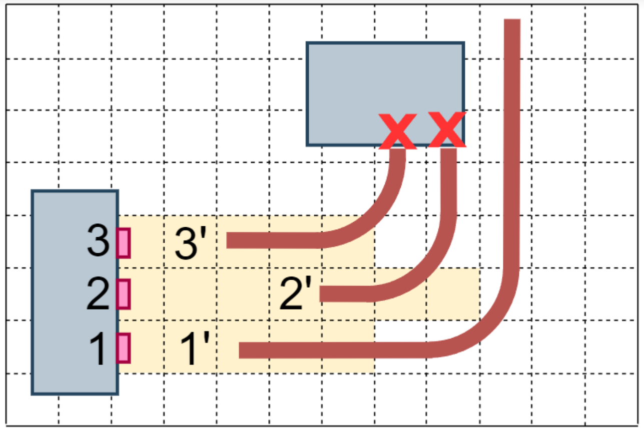
Issue 3. *hard to escape* due to high port density

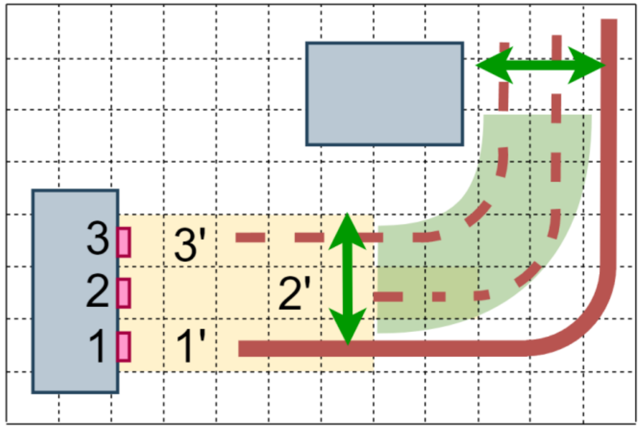




Solution 3. **Port-group based net planning:**
mountain-shape port region & route group by group

Issue 4. *Routing congestion*: no resource planning



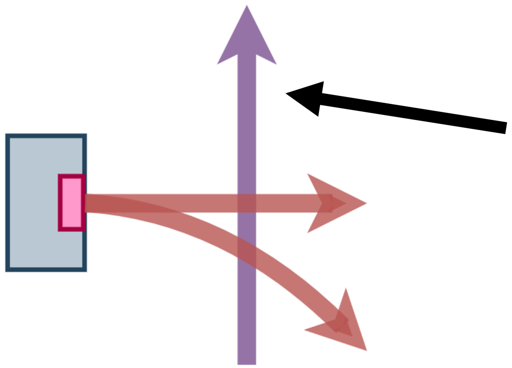


Solution 4. **Group congestion penalty:**
reserved routing resources $\propto$ *{unrouted nets in group}*

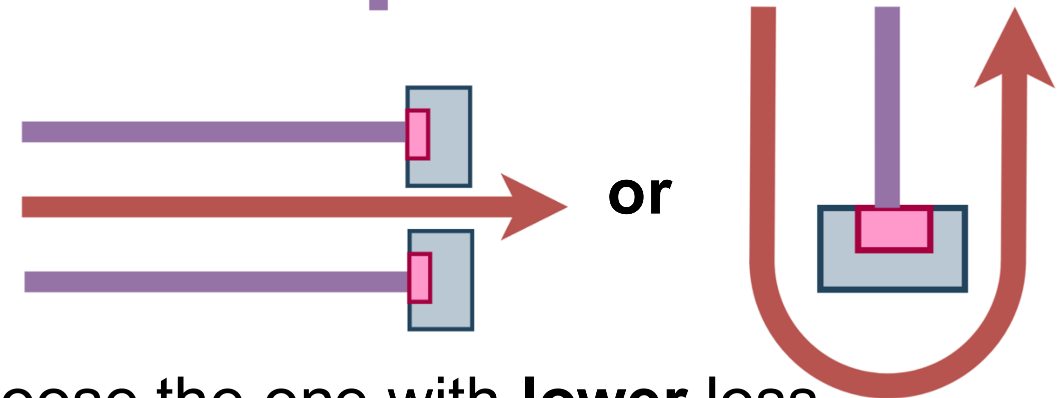
Crossing Optimization & Waveguide Refinement

- Crossing optimization
  - Try **<u>crossing-disabled routing</u>**
  - If failed:
    - Blocked by other net
  - If success:
    - Go through congested region

      or
    - Long detour w/o crossing

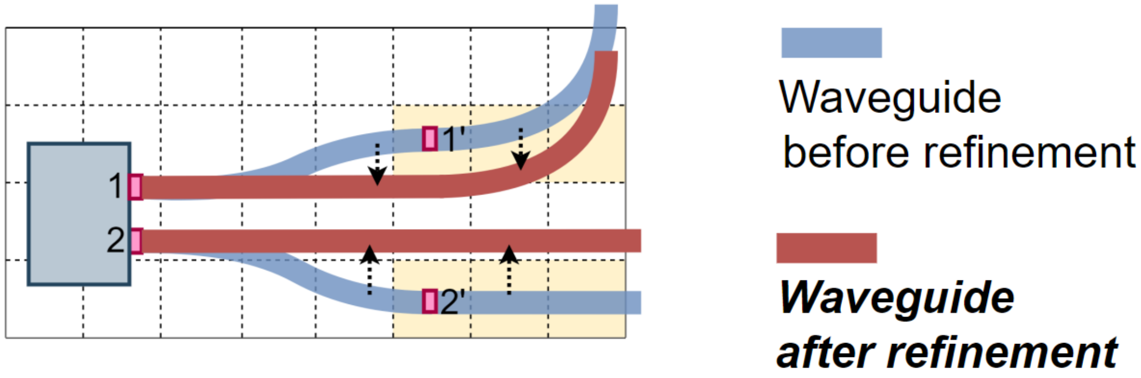


Need to ripup the **blocking waveguide?**



Choose the one with **<u>lower</u>** loss

- Waveguide refinement
  - Shift & stretch to remove unnecessary offset/curves

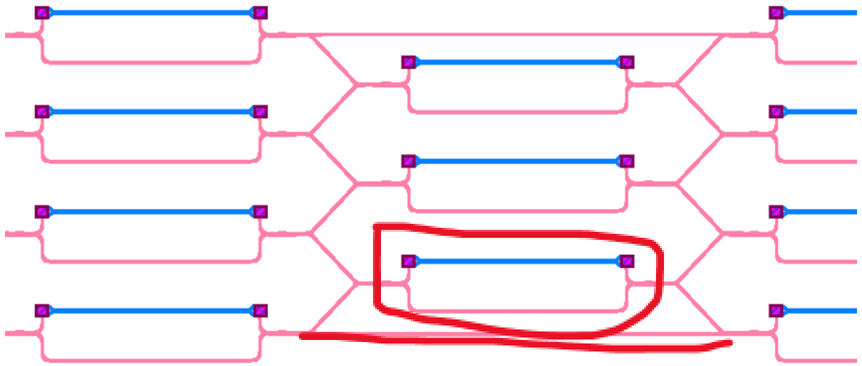


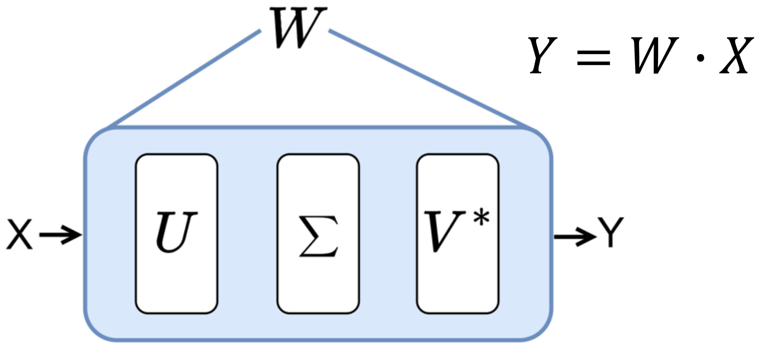
Evaluation Setup

- Machine & platform
  - Intel i5-125600KF 3.7GHz CPU 32 GB RAM
  - Python 3.11, based on **latest** GDS FACTORY
- Baseline PIC routers
  - Base-1: Proton [Boos+, ICCAD'13] with rip-up & reroute
  - Base-2: Proton [Boos+, ICCAD'13] with diagonal neighbors
- Benchmark suits (customized LEF/DEF-like format for PIC)
  - **Computing**: photonic tensor core (PTC)
    - Clements-style MZI arrays [Shen+, NatPhoton'17]
    - ADEPT auto-searched PTC [Gu+, DAC'22]
  - **Interconnect**: Wavelength-routed Optical Network-on-Chip (WRONOC)

Photonic Computing Benchmarks

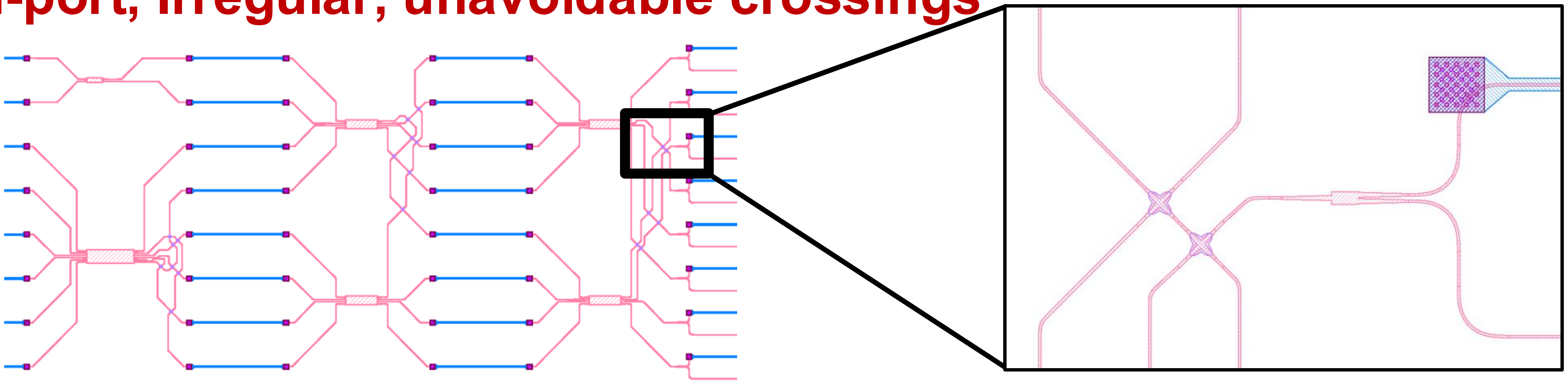
- **Clements:** classic MZI-based matrix multiplication unit [Shen+, NatPhoton'17]
  - **Regular structure, no crossing**





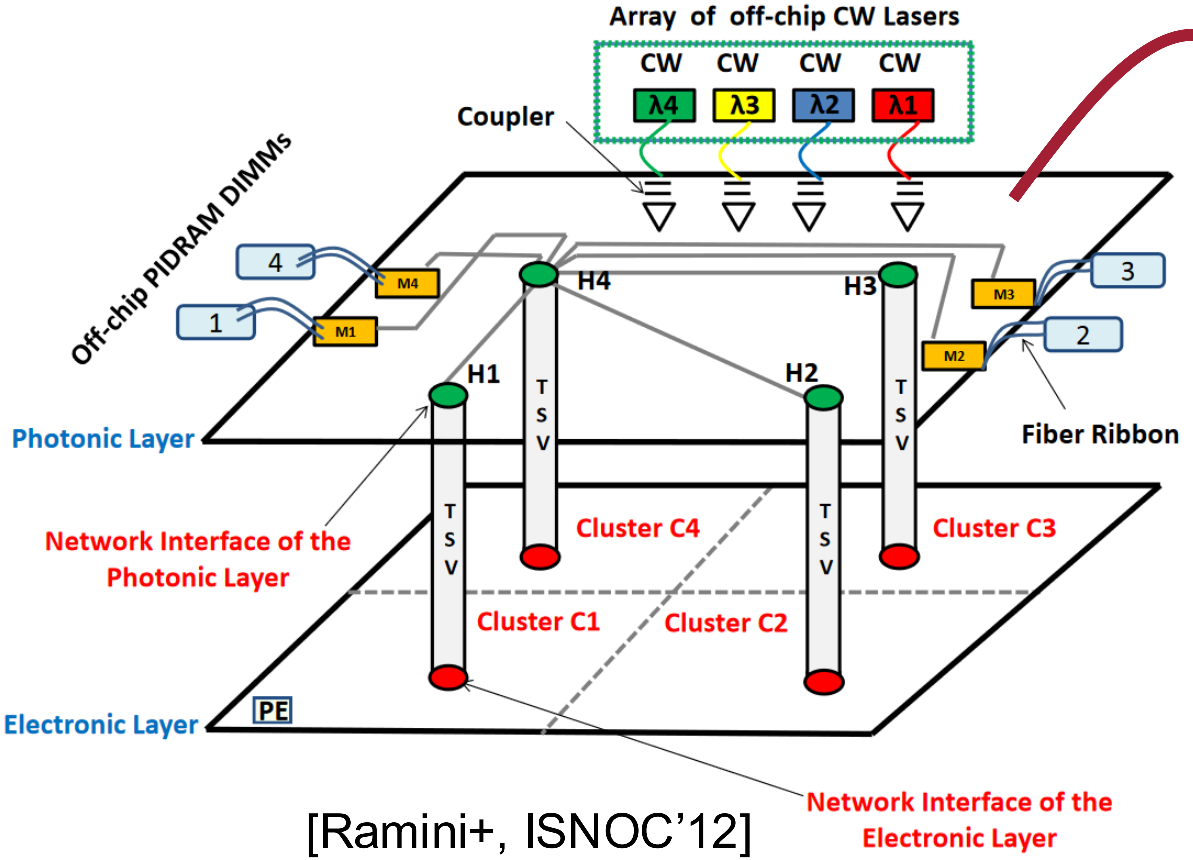
Optical Inference Unit

- **ADEPT:** auto-searched subspace photonic tensor core [Gu+, DAC'22]
  - **Multi-port, Irregular, unavoidable crossings**

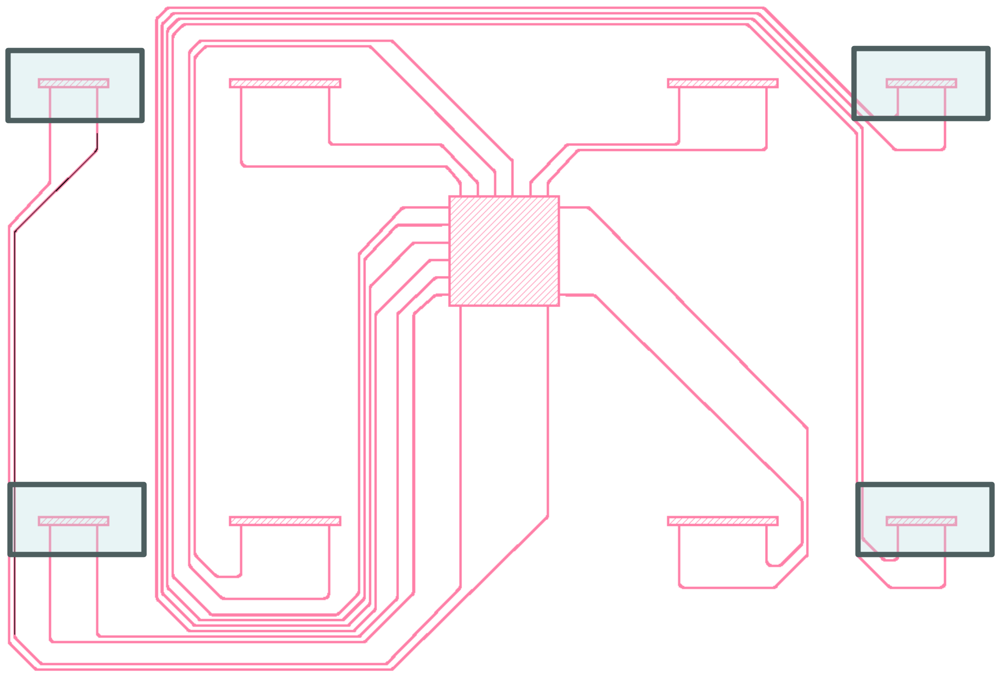


Optical Interconnect Benchmarks

- Wavelength-routed Optical Network-on-Chip (WRONOC)
  - Different position of memory controls: north, one-side, pair-wise, corner
  - <u>Exist optimal solution (no crossing)</u>



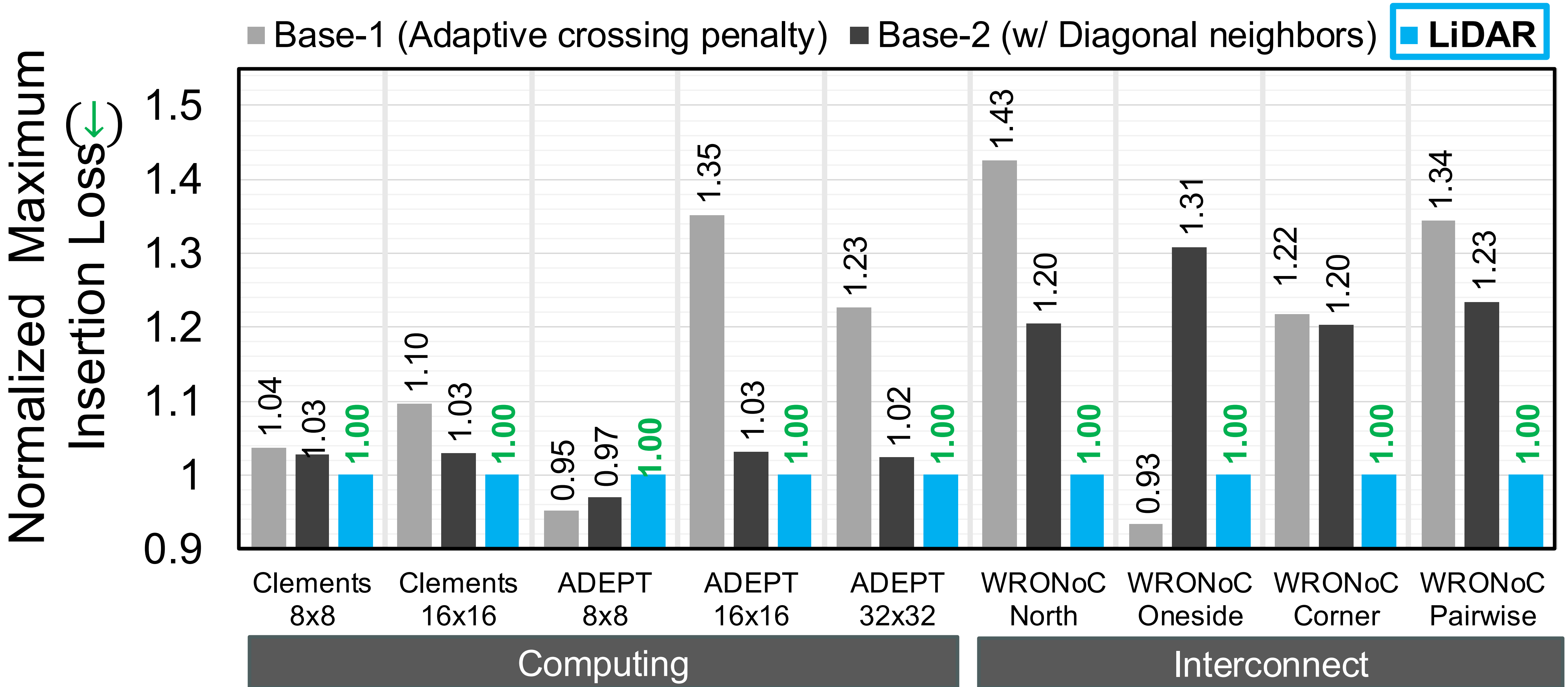
[Ramini+, ISNOC'12]



Pair-wise benchmark

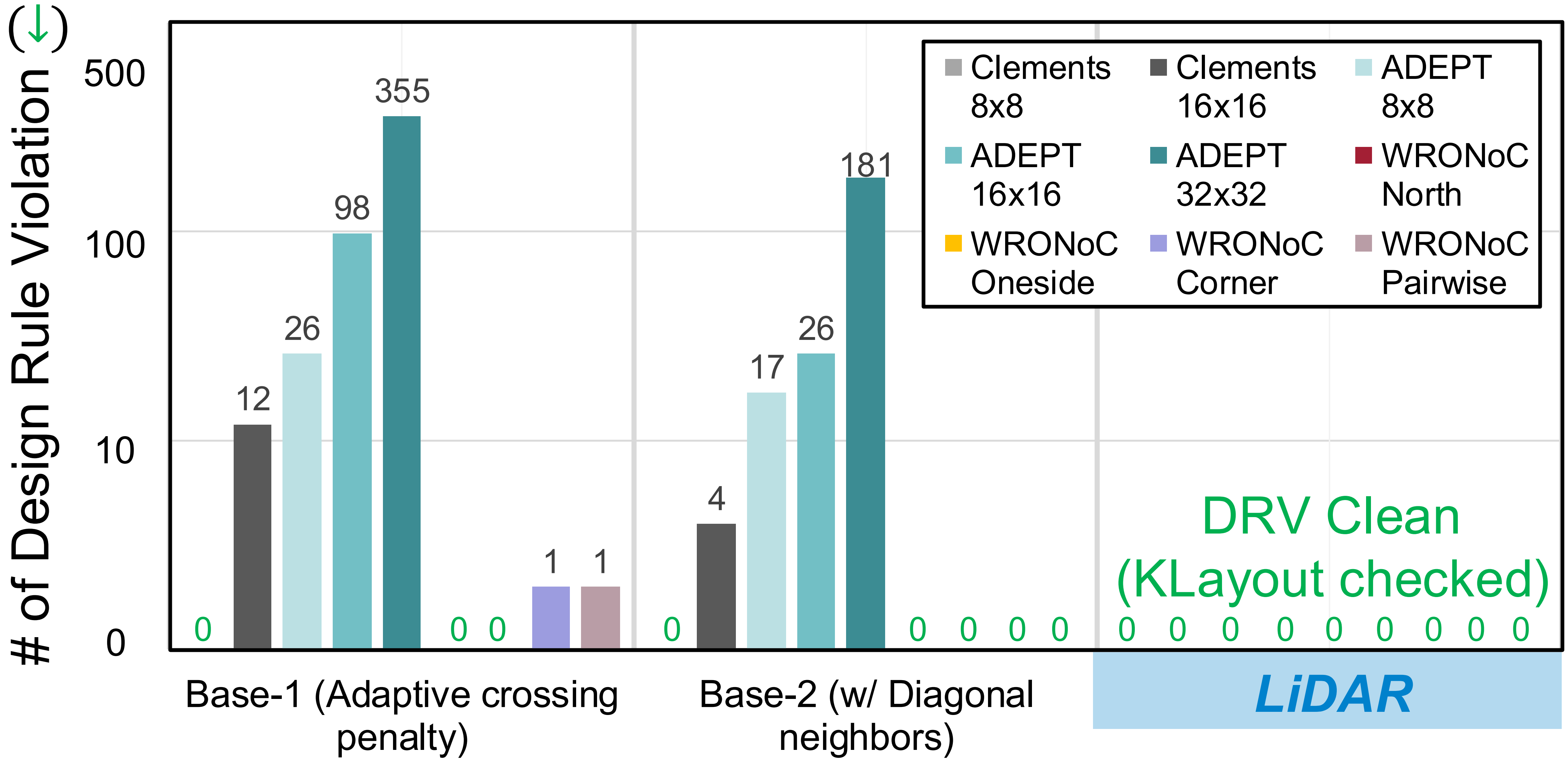
Maximum Insertion Loss Comparison

- ♦ ***LiDAR*** outperforms other routers in IL_{max}
  - › **14%** better than Base-1 **5%** better than Base-2



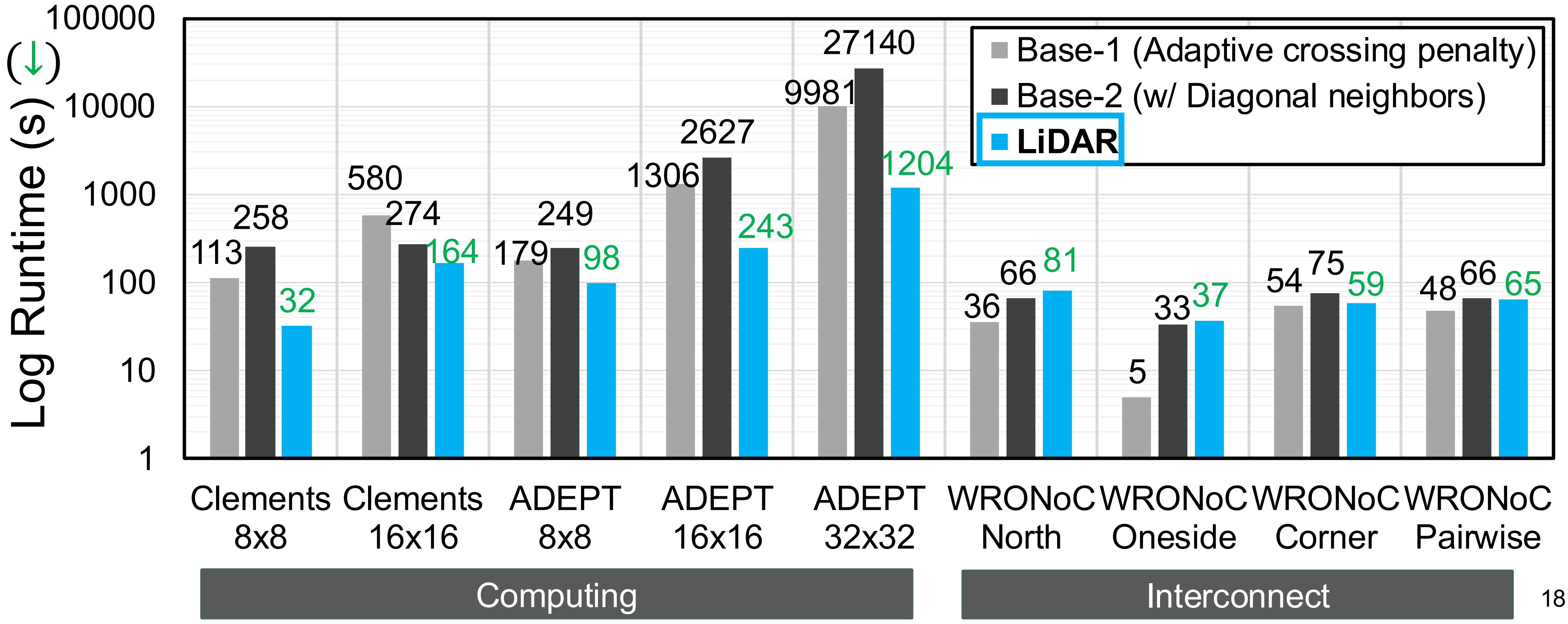
of Design Rule Violation Comparison

♦ ***LiDAR*** generates **DRV-free** solutions on all benchmarks



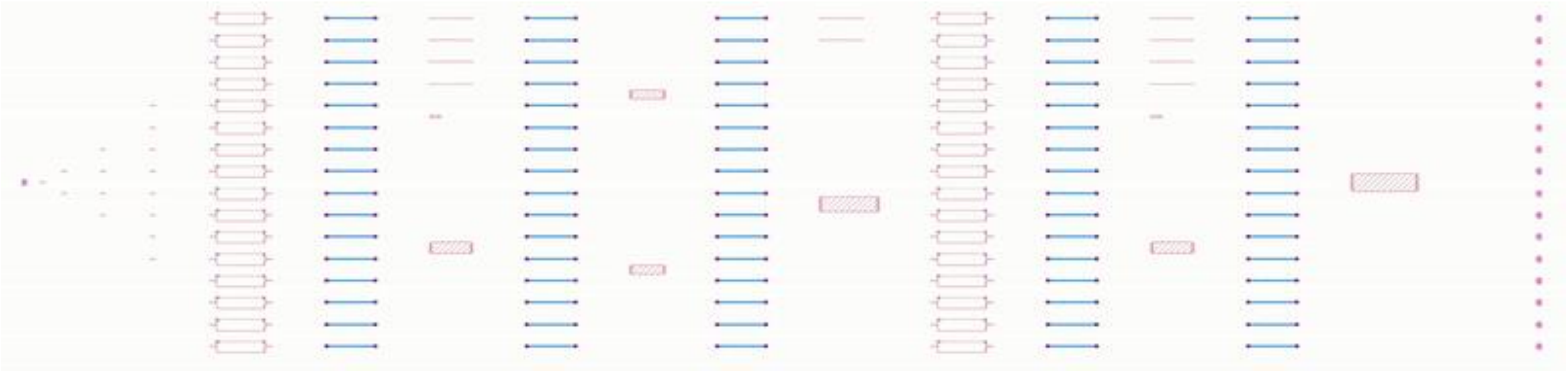
Runtime Comparison

- ***LiDAR*** is 2.75× faster than Base-1 and is 5.51× faster than Base-2
  - Smart crossing insertion → Less ripup & reroute

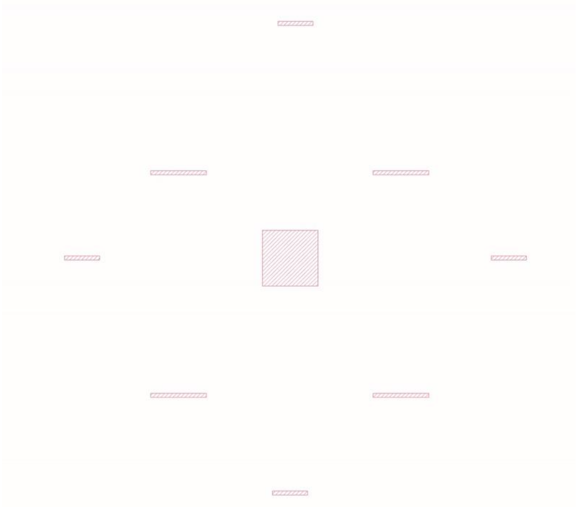


Animation of *LiDAR* for PIC Detailed Routing

- Photonic computing: ADEPT 16x16 PTC (243 s + 0 DRV)

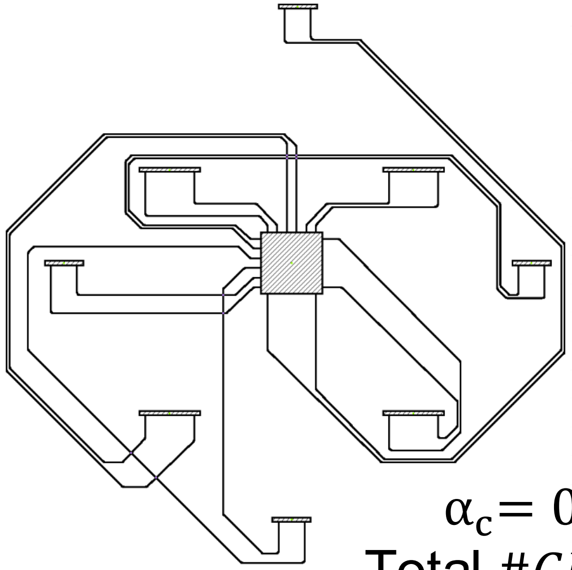


- Optical interconnect: WRONoC_north (81 s + 0 DRV)

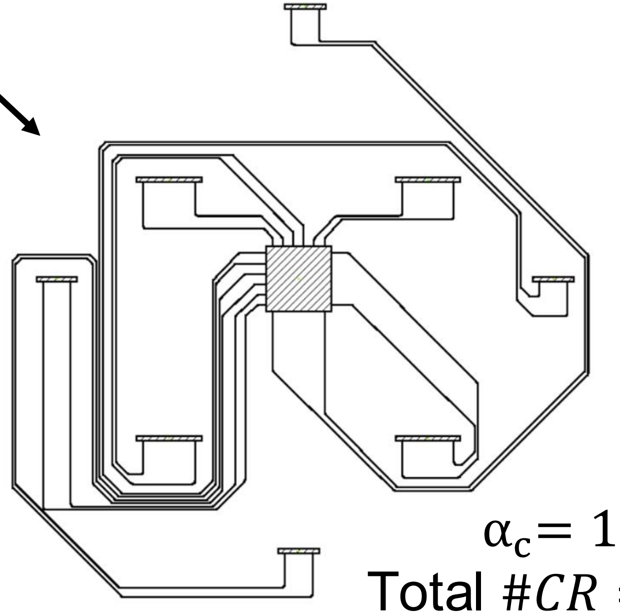


Designer-Controlled, PDK-Adaptive Congestion Penalty (GCP)

- User-defined crossing penalty strength adaptive to different PDKs
- Larger **crossing loss $\boldsymbol{\alpha_c}$** encourages fewer crossings: $\boldsymbol{\alpha_c}\uparrow \rightarrow$ **#CR**$\downarrow$
- GCP improves routing legality



$\alpha_c = 0.3$ dB
Total $\#CR = 8$



$\alpha_c = 1$ dB
Total $\#CR = 0$

Metrics	α_c=1 dB		α_c=0.3 dB	
	w/o GCP	**LiDAR**	w/o GCP	**LiDAR**
# CR	6	0	5	5
WL (mm)	20.72	31.11	25.11	26.04
IL_{max} (↓)	15.21	10.78	7.18	7.31
DRV	0	0	**1**	**0**
Time (s)	129	73	261	197

Thank you!

Q & A?

LiDAR: Automated Curvy Waveguide Detailed Routing for Large-Scale Photonic Integrated Circuits

Hongjian Zhou
hzhou144@asu.edu
Arizona State University
Tempe, Arizona, USA

Keren Zhu
krzhu@fudan.edu.cn
Fudan University
Shanghai, China

Jiaqi Gu
jiaqigu@asu.edu
Arizona State University
Tempe, Arizona, USA

Open-Source PIC router LiDAR

PIC detailed router for auto waveguide routing

Seamless w/ GDSFactory 8

arXiv Preprint

Arizona State University

ASU Center for Semiconductor Microelectronics (ACME)

SCOPE