ADEPT: Automatic Differentiable DEsign of Photonic Tensor Cores

Jiaqi Gu[1], Hanqing Zhu[1], Chenghao Feng[1], Zixuan Jiang[1], Mingjie Liu[1], Shuhan Zhang[1], Ray T. Chen[1], David Z. Pan[1]

[1]*ECE Dept., University of Texas at Austin*

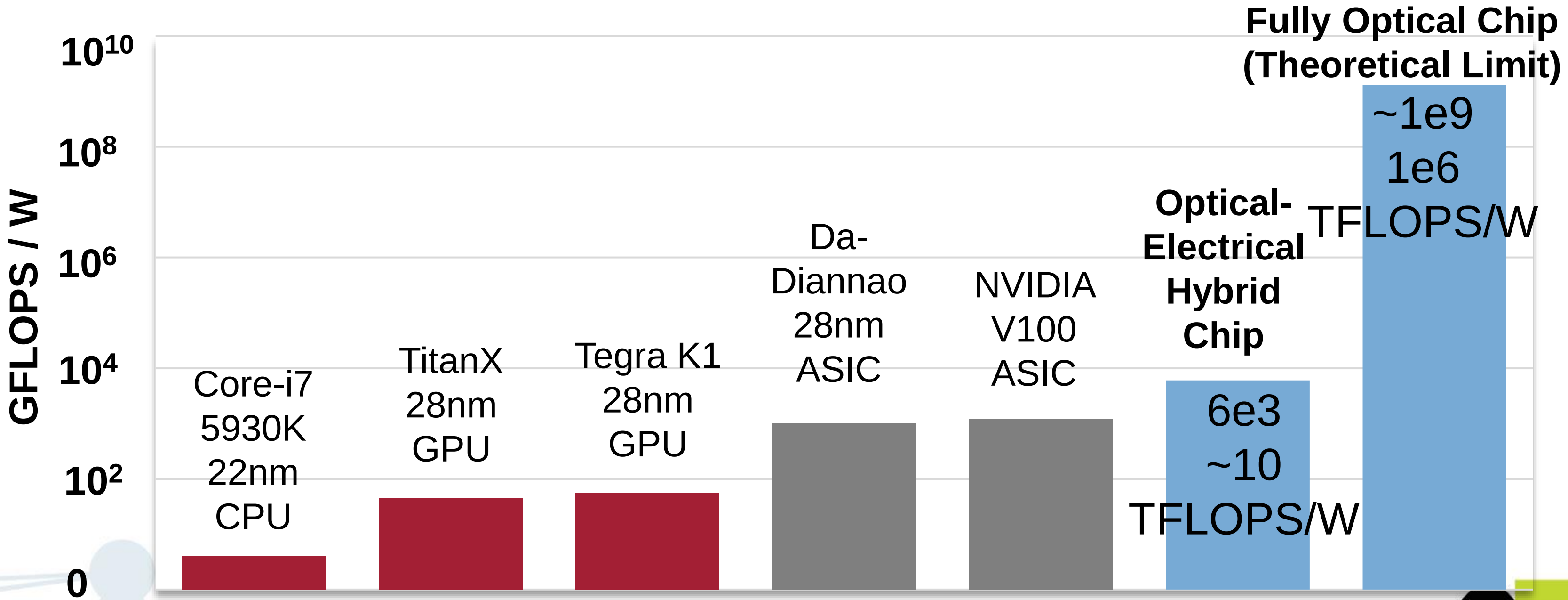
This work was supported in part by AFOSR MURI

jqgu@utexas.edu, jeremiemelo.github.io

Photonic AI

- AI compute requirement has **5×** the doubling rate of Moore's law
- Optics as next-generation AI solution

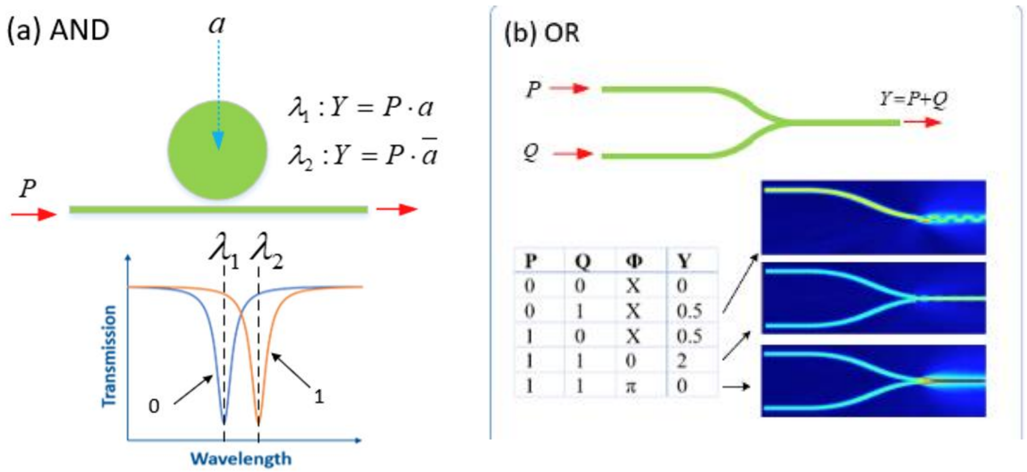
Ultra-high speed & **Ultra-low energy**



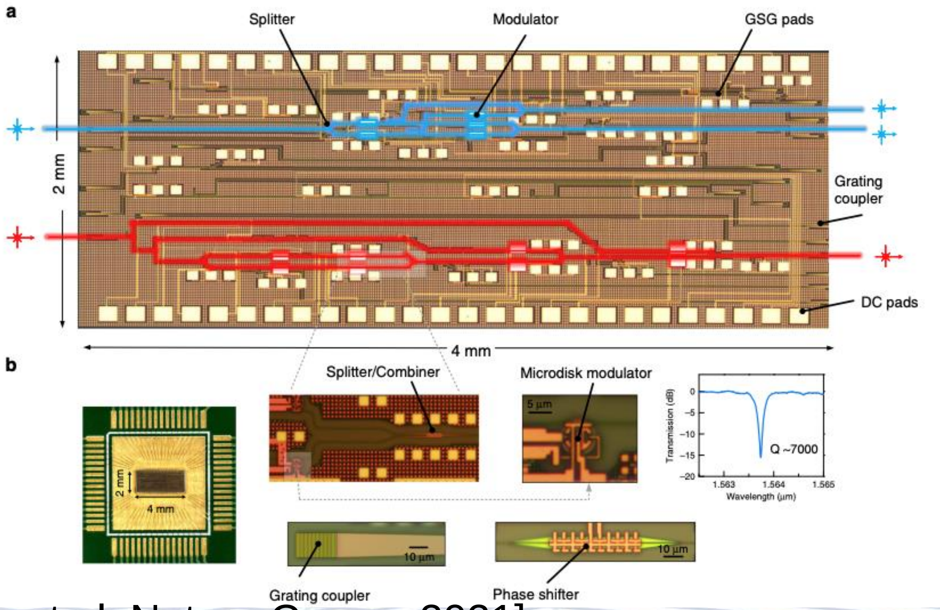
[Shen+, *Nature Photonics* 2017]

Optical Computing Basics

Digital computing



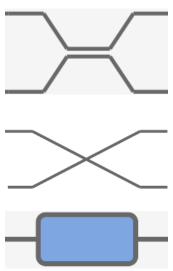
P	Q	Φ	Y
0	0	X	0
0	1	X	0.5
1	0	X	0.5
1	1	0	2
1	1	π	0



[Ying et al, Nature Comm. 2021]

Analog computing

***Basic Components**

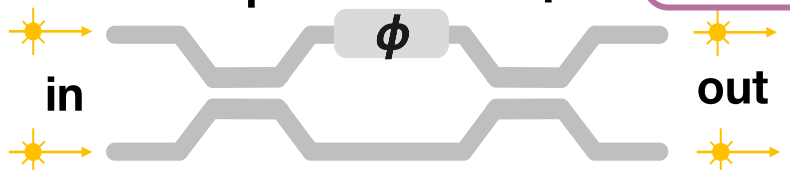


$$\begin{pmatrix} t & \sqrt{1-t^2}j \\ \sqrt{1-t^2}j & t \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$y = e^{-j\phi}x$$

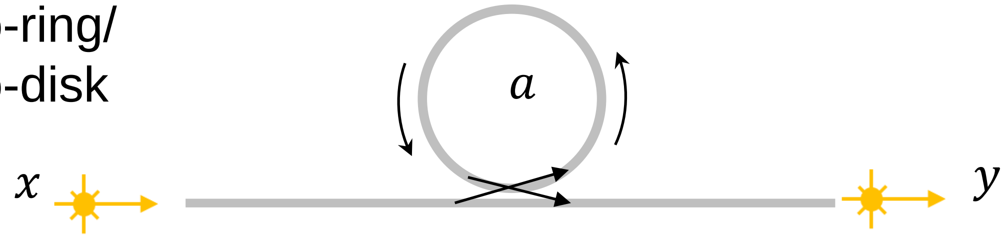
MZI



$$\begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

2×2 Unitary Matmul (~100×20 μm^2)

Micro-ring/
Micro-disk



$$y = a \cdot x$$

Scalar Mul. (~10×10 um^2)

WDM+PD

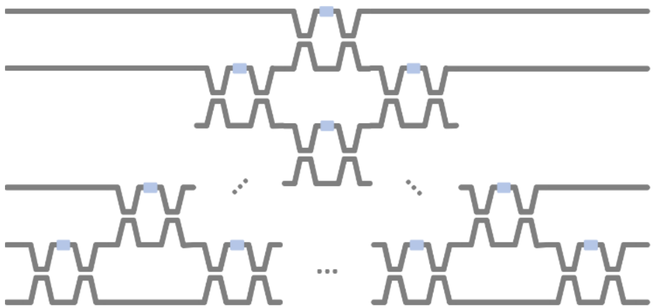
$$y = \sum x_i$$

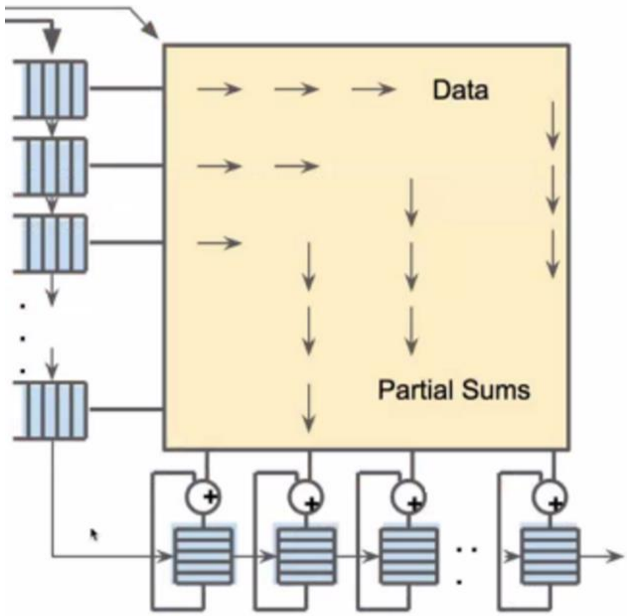
[Lightelligence]

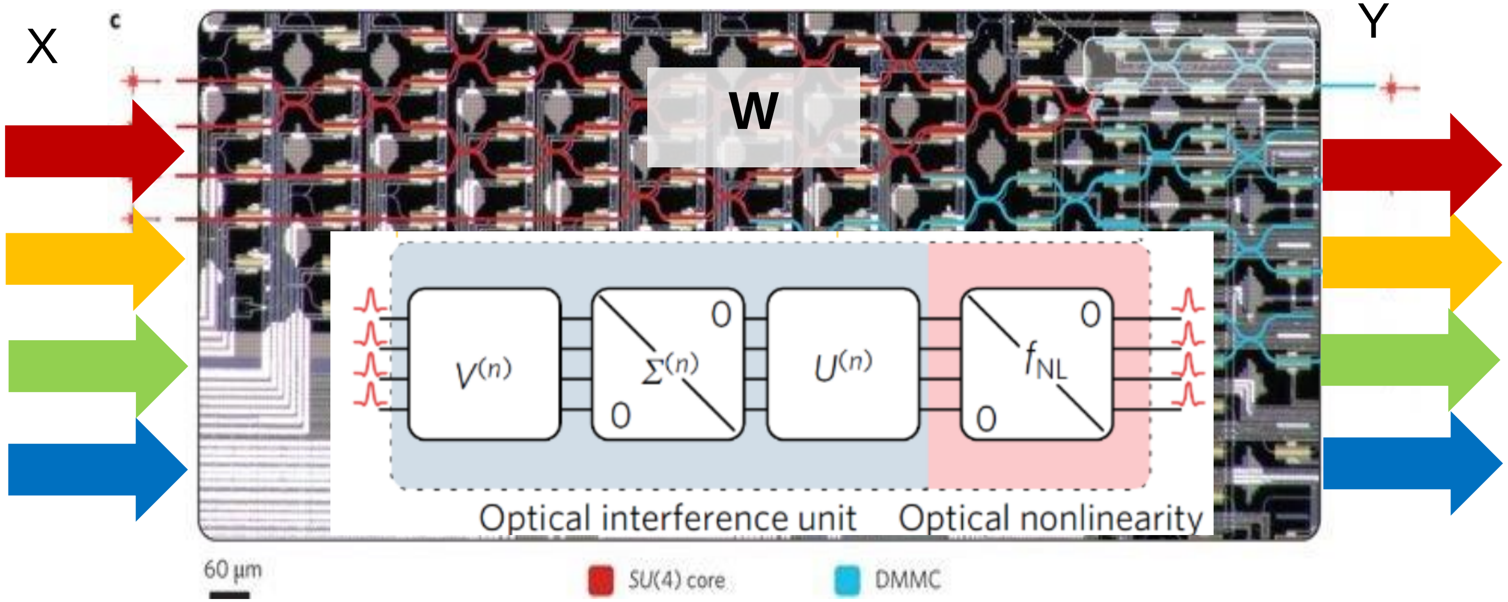
WDM-based Summation

ONN: **P**hotonic **T**ensor **C**ore (PTC)

- DNNs: linear projection + nonlinear activation
  - Matrix multiplication is computation-intensive
- Photonics is good at ultra-fast linear operations





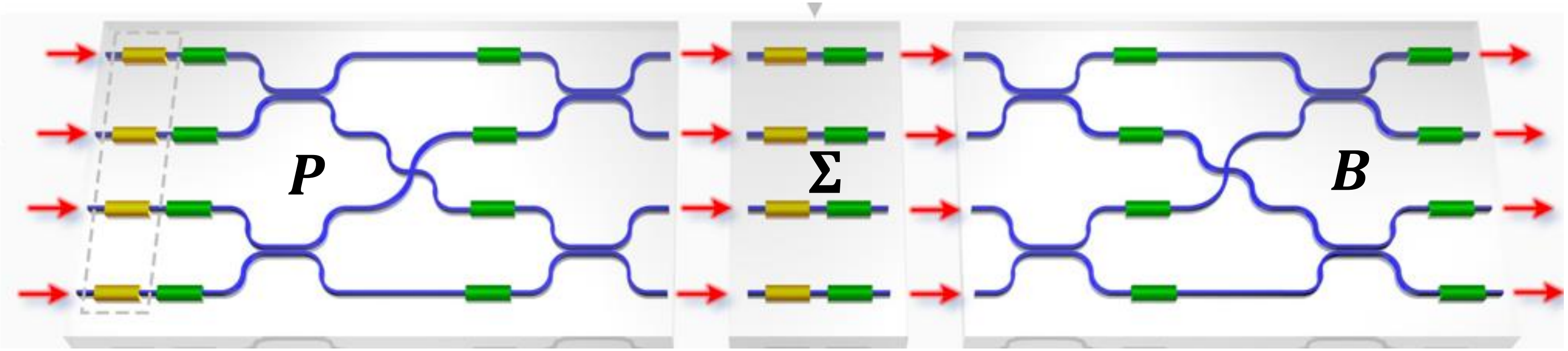


Electrical systolic array for digital GEMM [Google TPU]

Photonic tensor unit for analog GEMM [Nat. Photonics'17]

General → Subspace Photonic Tensor Core

- $\boldsymbol{U\Sigma V} \rightarrow \boldsymbol{B\Sigma P}$: Compact butterfly photonic mesh
  - Compact footprint: *No MZI*, use basic optical devices



- Trade universality for higher hardware efficiency

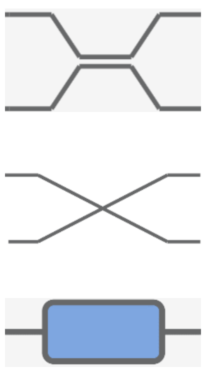
J. Gu, Z. Zhao, C. Feng, Z. Ying, R.T. Chen, and D.Z. Pan, ASP-DAC, 2020 (BPA)
C. Feng, J. Gu, H. Zhu., Z. Ying, Z. Zhao, D.Z. Pan, R.T. Chen, Under Review, 2022

Motivations

Manual Design → Automated PTC Design

Can we automatically explore larger design space adaptive to different design constraints ?

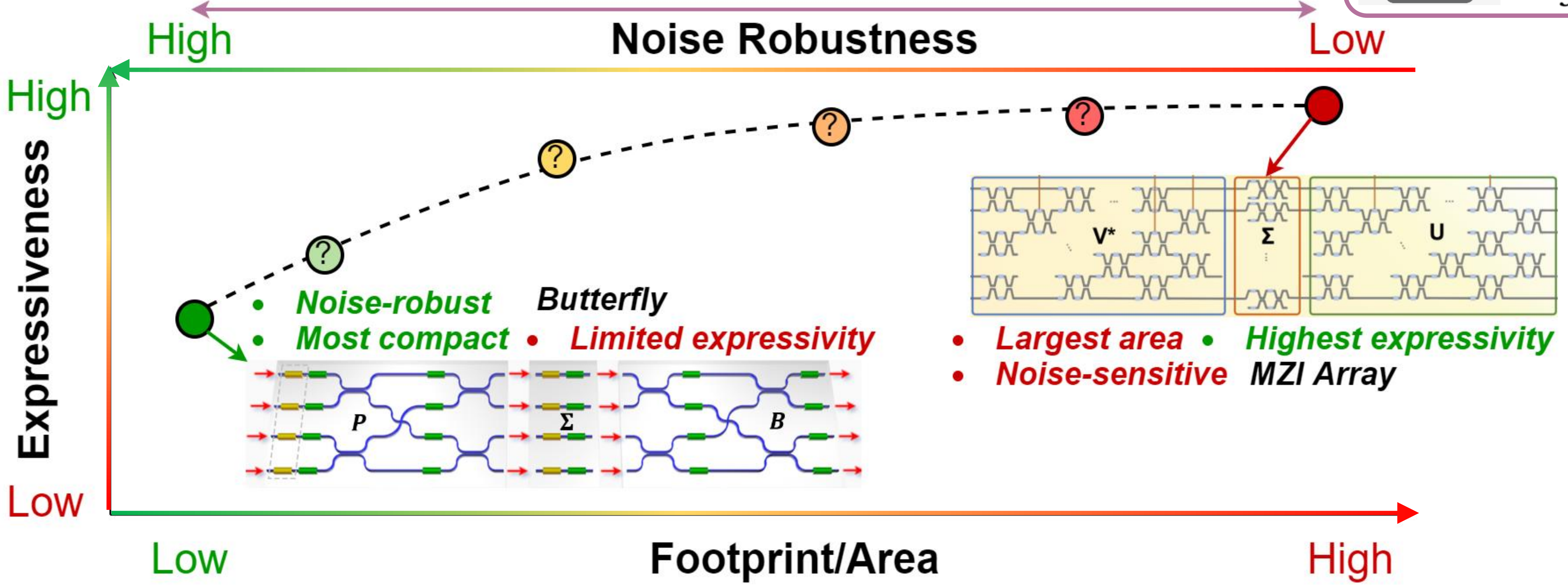
***Basic Components**



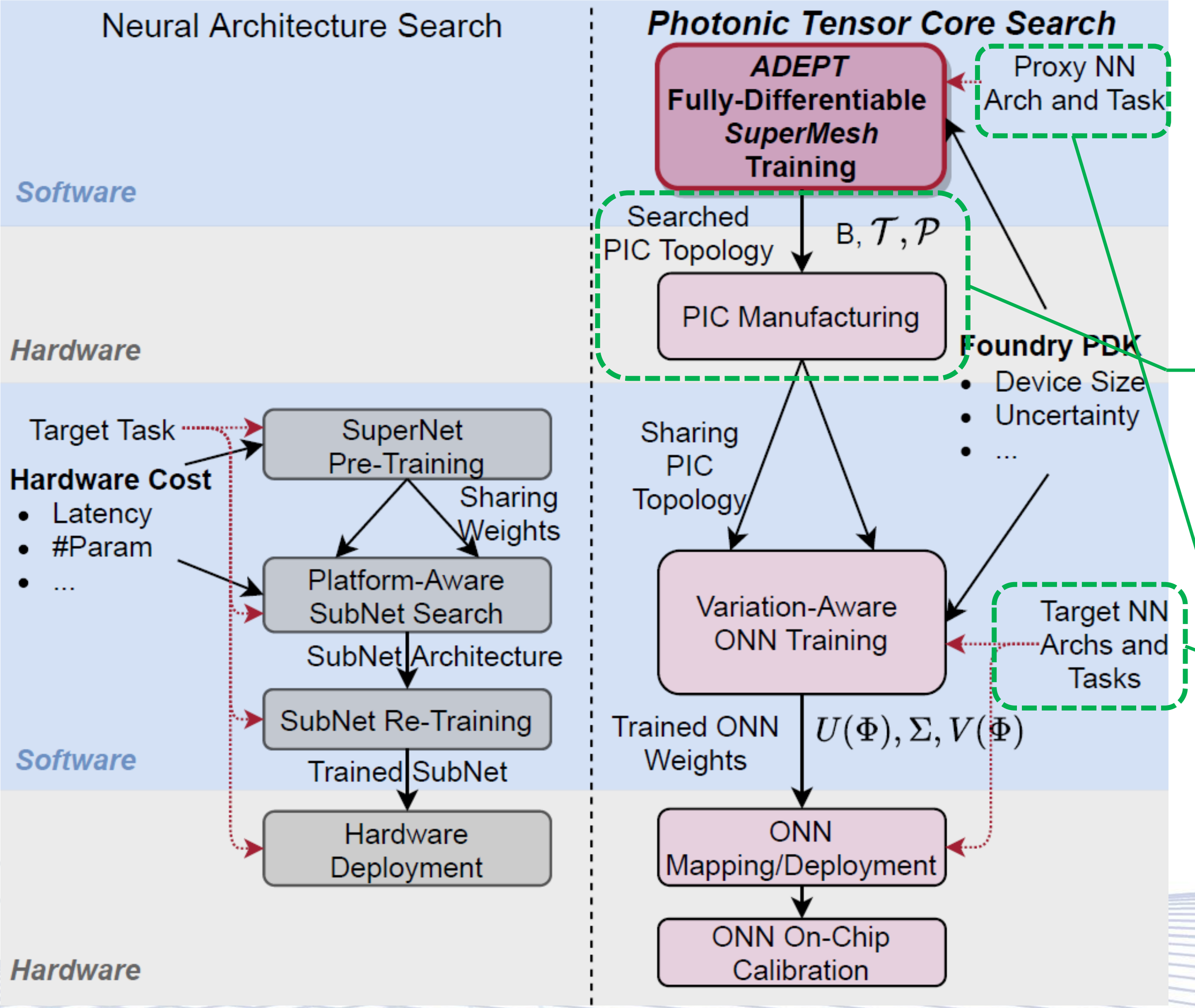
$$\begin{pmatrix} t & \sqrt{1-t^2}j \\ \sqrt{1-t^2}j & t \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$y = e^{-j\phi}x$$



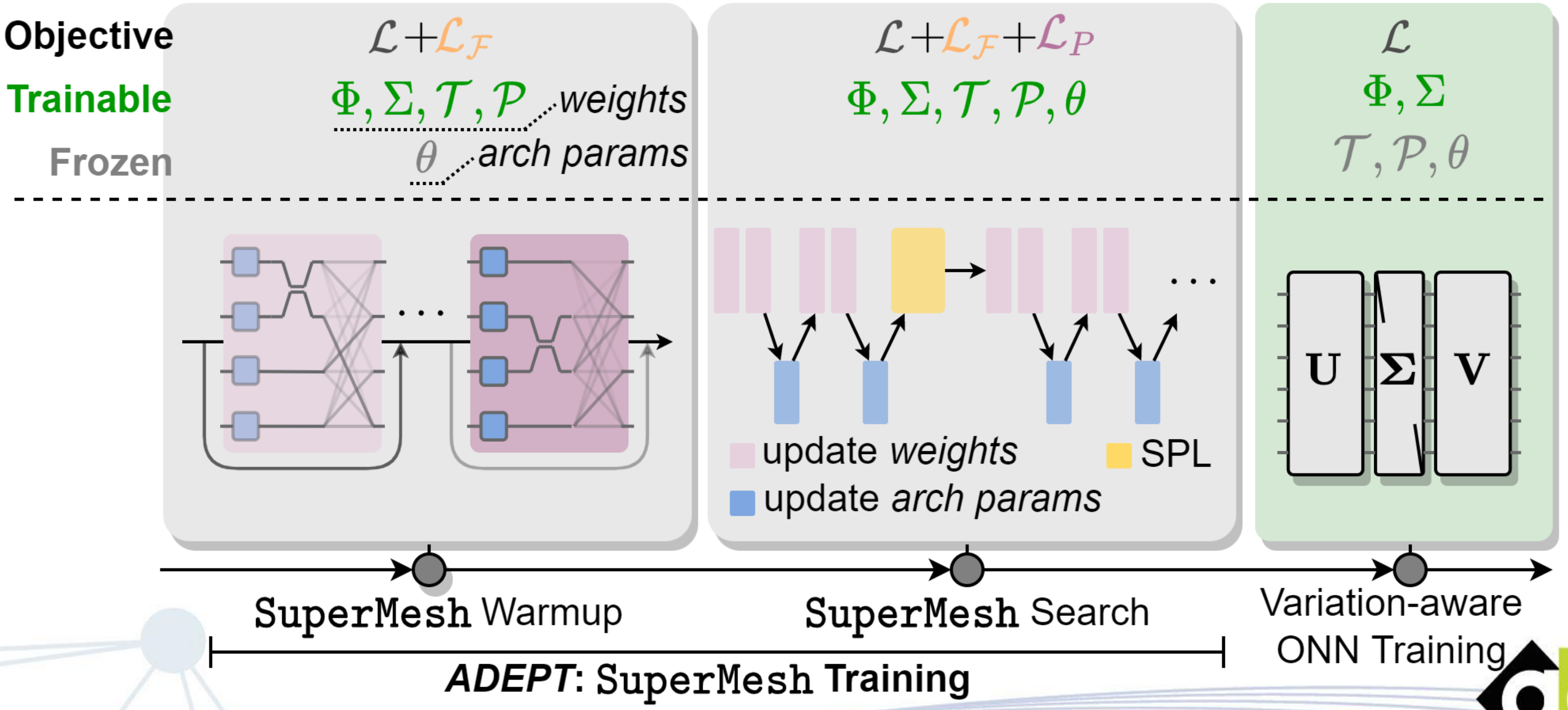
NAS vs. PTC Search



- Large and highly-discrete search space
- PTC Circuit topology search
  - Unchangeable after chip manufacturing
- Search on proxy tasks → Adapt to various NN workloads

Our Proposed *ADEPT*

- The first *differentiable* PTC topology search framework
- *Automatically* find a coherent PTC design with basic components
- Adapt to different *foundry PDKs* and *footprint constraints*

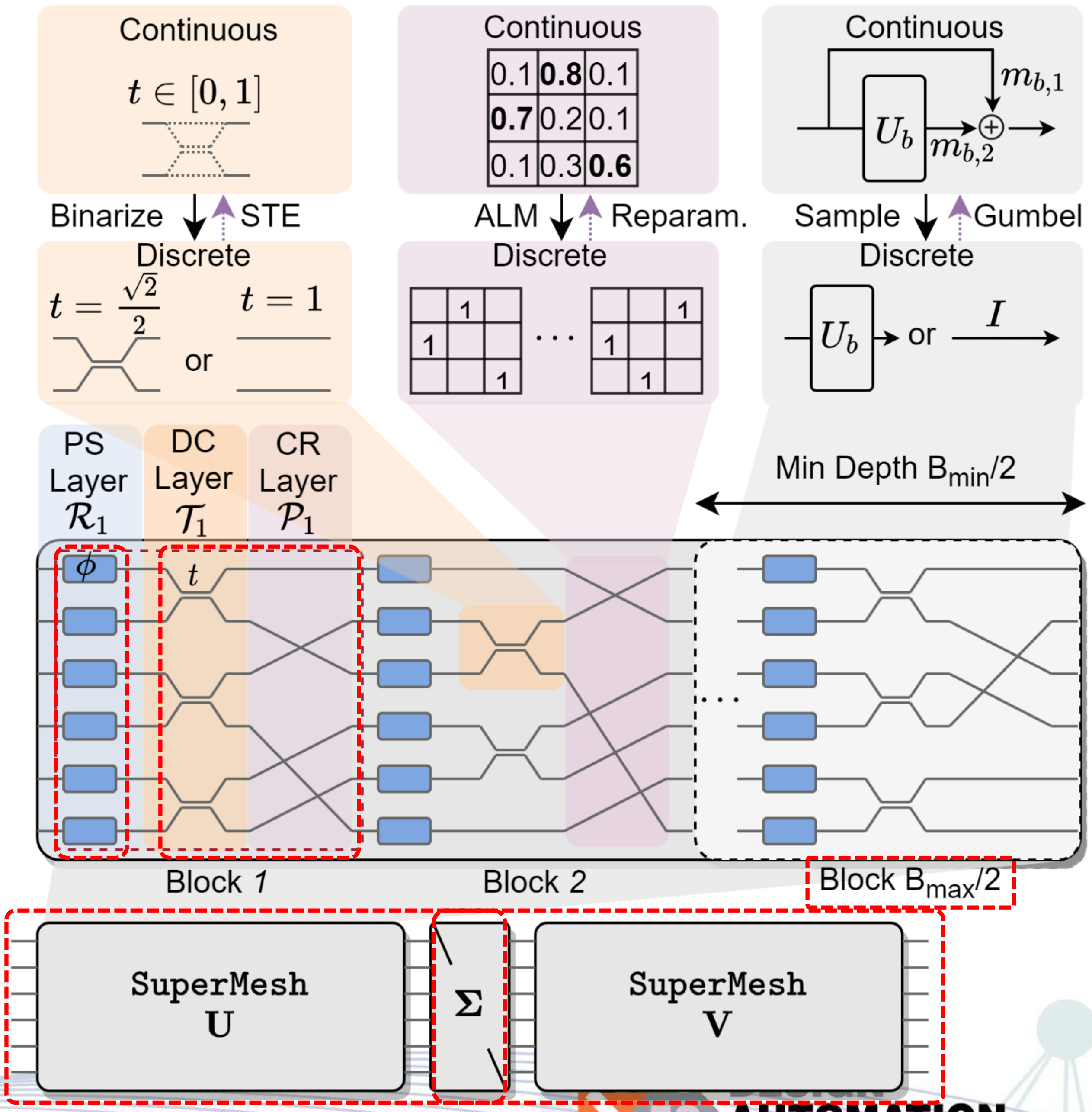


ADEPT Formulation

- Block: PS/DC/CR layers
- SuperMesh *weights*
  - Phases: Φ^U, Φ^V
  - Diagonal: Σ
- SuperMesh *arch params* (α)
  - Block number: B^U, B^V
  - Couplers: $\mathcal{T}_b$
  - Crossings: $\mathcal{P}_b$
- Bilevel optimization with footprint constraint: F_{min}, F_{max}

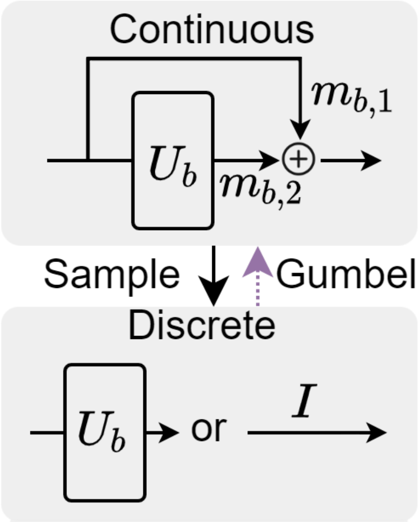
$$\min_{\alpha \in \mathcal{A}} \mathcal{L}(W^{*\alpha};\ \mathcal{D}^{val}), \quad \alpha = (B^U, B^V, \mathcal{P}, \mathcal{T})$$

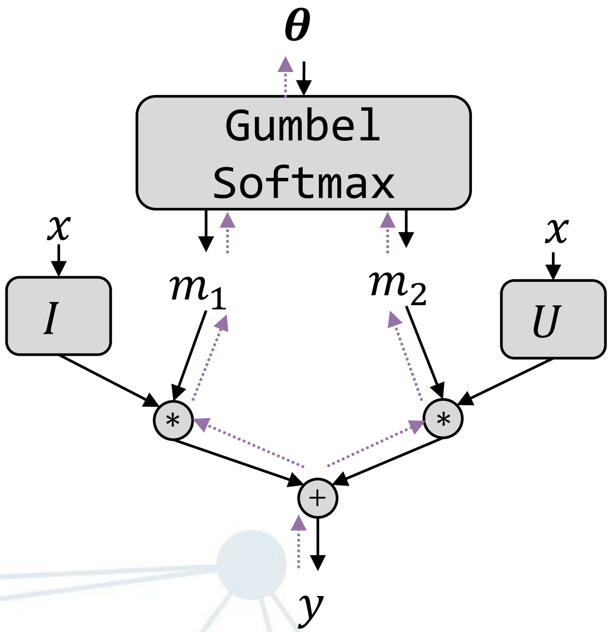
$$\text{s.t. } W^* = \underset{W}{\operatorname{argmin}}\, \mathcal{L}(W^{\alpha};\ \mathcal{D}^{trn}), \boxed{F_{min} \le \mathcal{F}(\alpha) \le F_{max}}$$

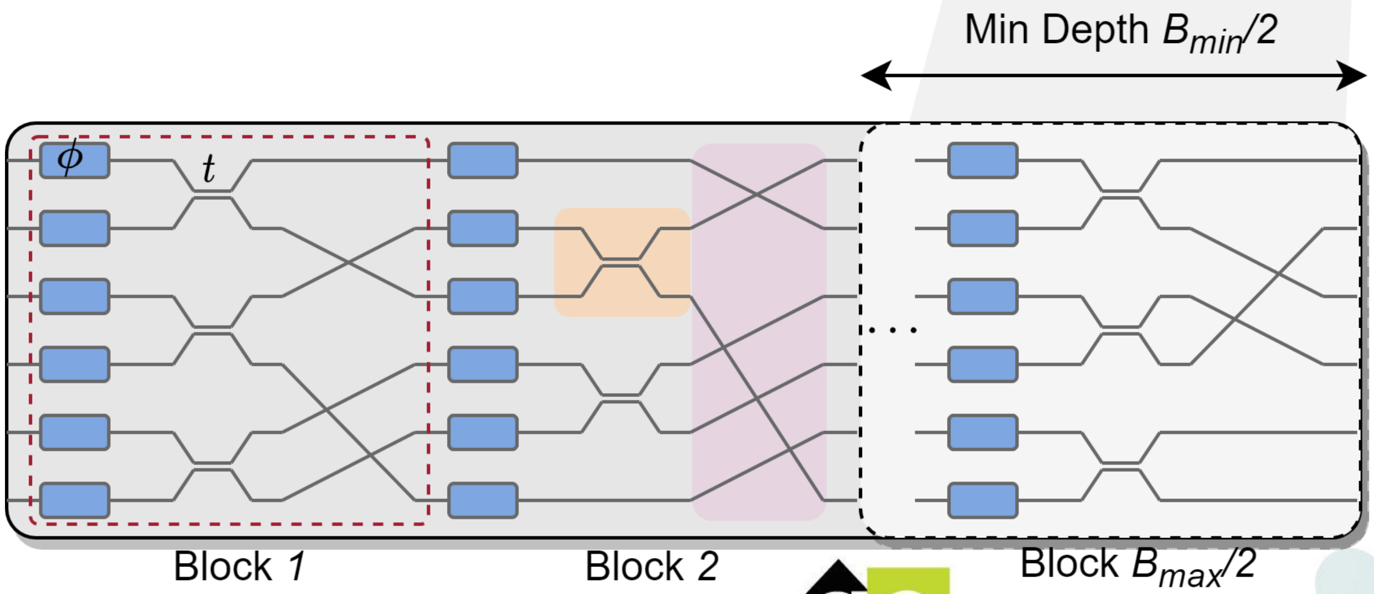


Optimize SuperMesh Depth B

- Discrete variable: *skip* $(U_{b,1})$ *or* *keep* $(U_{b,2})$
- Probabilistic *SuperMesh* block
  - Sample mask m from a distribution
  - Learn the distribution with Gumbel softmax trick [B. Wu+, FBNet, CVPR'19]

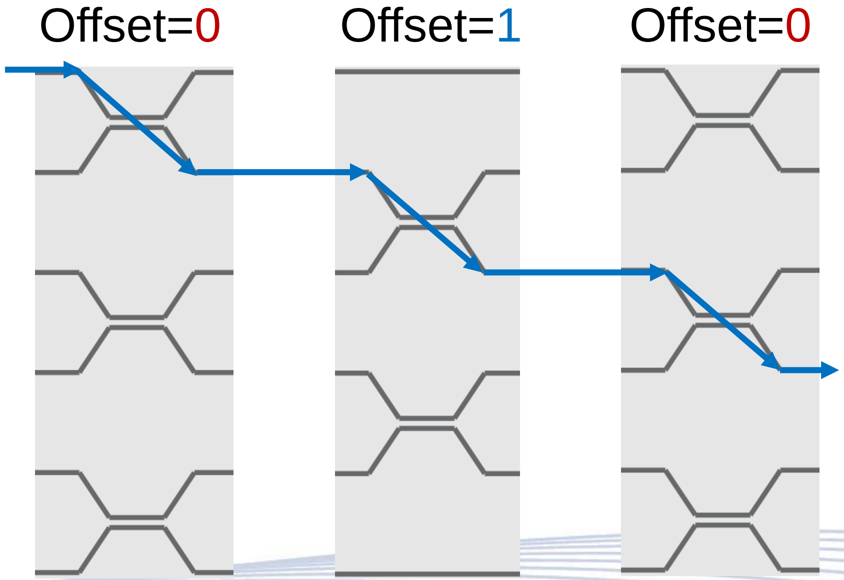


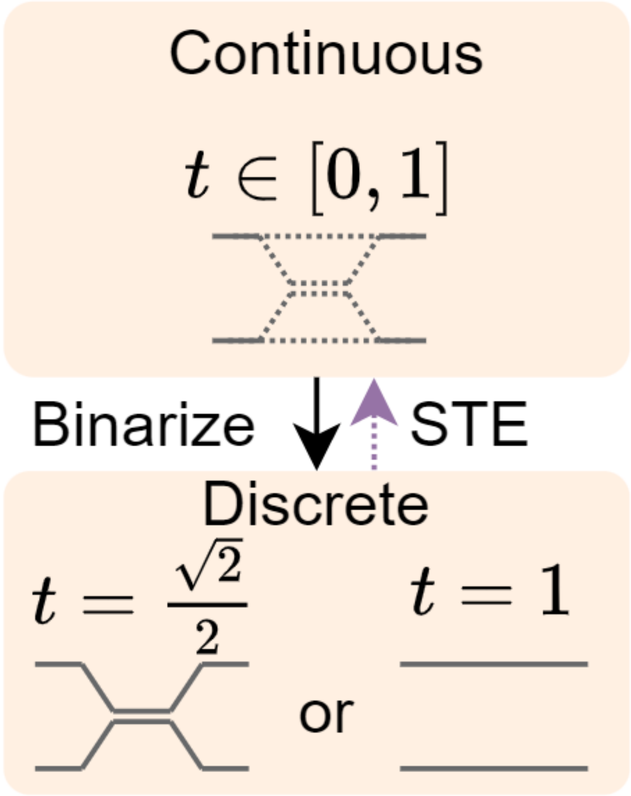




Optimize Coupler Layer $\mathcal{T}$

- Discrete variable: *identity* ($t = 1$) or *coupler* ($t = \frac{\sqrt{2}}{2}$)
- Train continuous transmission $t \in [0, 1]$
- Binarize $t \in \{\frac{\sqrt{2}}{2}, 1\}$ with QAT
  - Estimate gradients with STE
- Interleaved DC layers
  - Information interaction



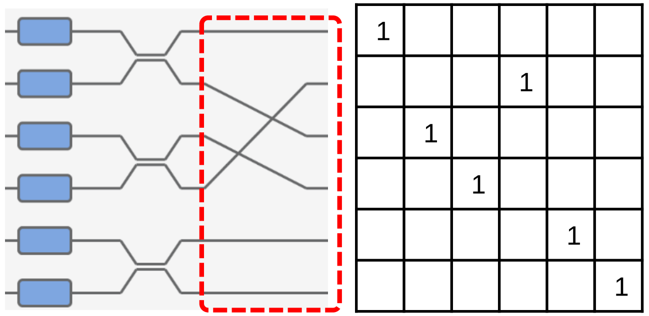


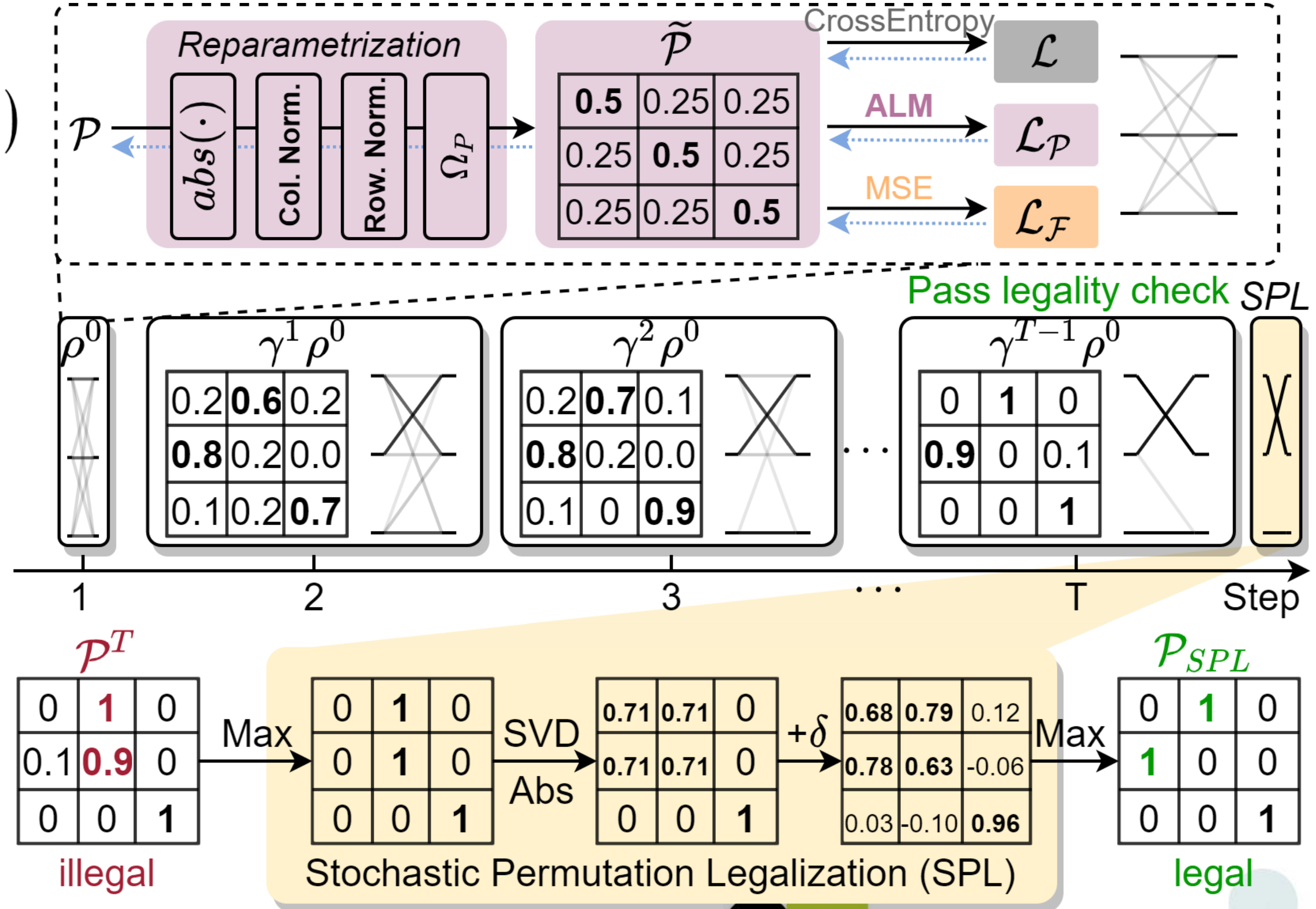
Optimize Permutation Layer $\mathcal{P}$

- All-to-all waveguide routing:
  - Permutation matrix
  - Huge design space $\mathcal{O}((K \cdot K!/2)^{B_{max}})$
- How to make it differentiable?
  - *Reparametrization*
  - *Relaxation* to its convex hull $\tilde{\mathcal{P}}$

$$\mathcal{P}_b \geq 0;\ \|\mathcal{P}_b^{i,:}\|_1 = \|\mathcal{P}_b^{i,:}\|_2, \forall i;\ \|\mathcal{P}_b^{:,j}\|_1 = \|\mathcal{P}_b^{:,j}\|_2$$

- Augmented Lagrangian (ALM)
  - Gradually push $\tilde{\mathcal{P}}$ to permutation $\mathcal{P}$
- Permutation legalization (SPL)

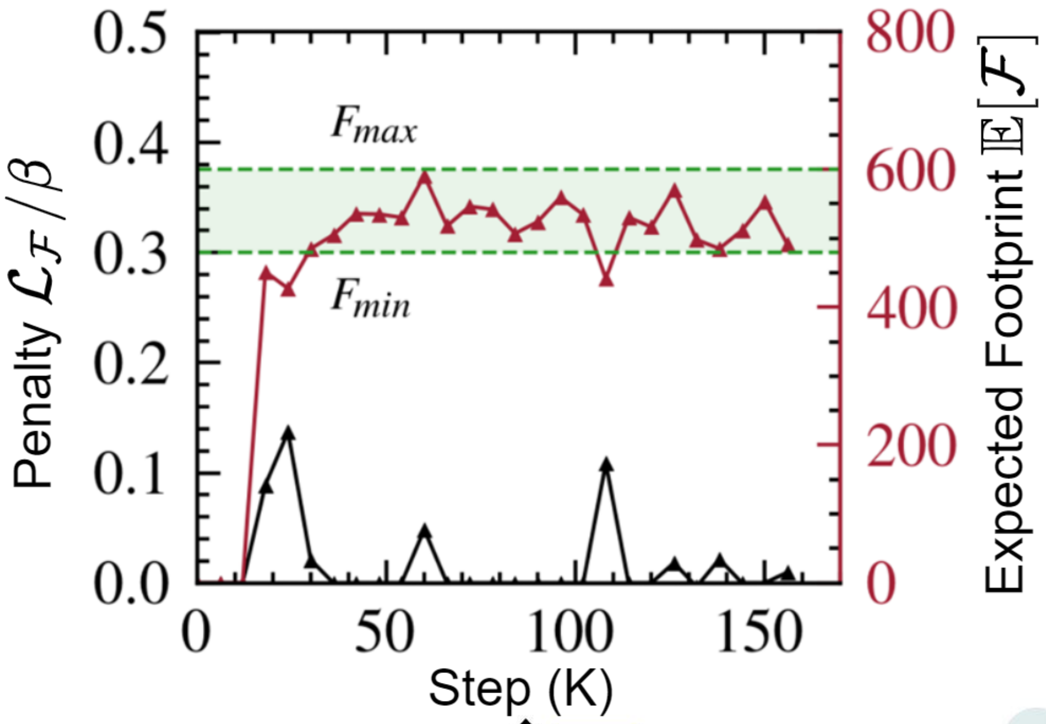




PDK-Adaptive Footprint-Constrained Optimization

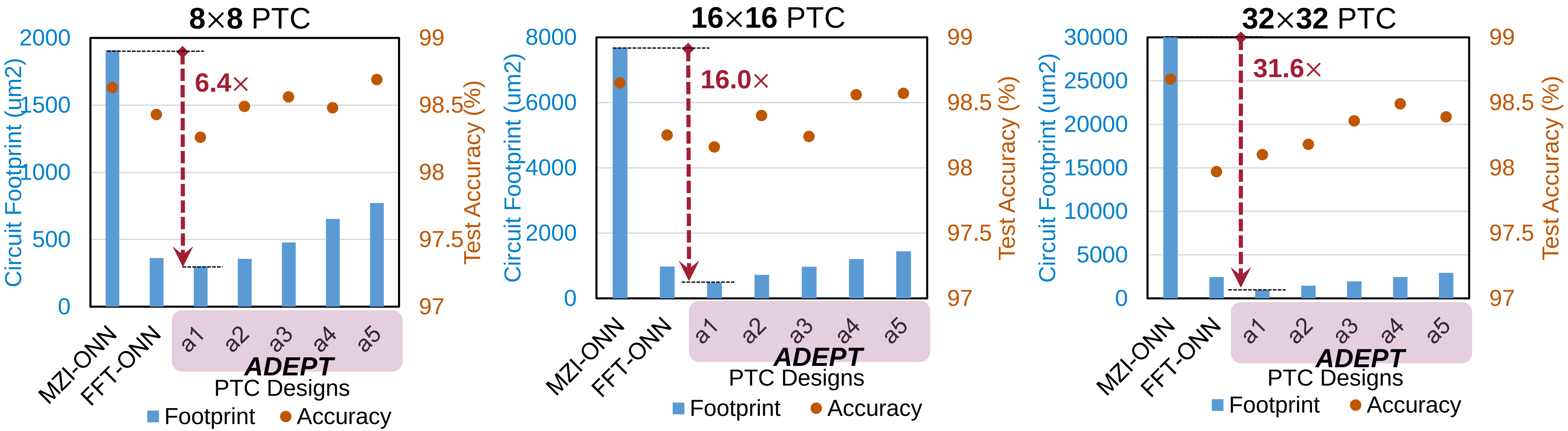
- Restrict *expected footprint* of the probabilistic *SuperMesh*
  - Models the device specification from foundry PDKs
- Restricted by lower/upper bounds
- If exceeds bounds
  - Penalize/encourage blocks, DC, CR

$$\mathcal{L}_{\mathcal{F}}=\begin{cases}\beta\Big(\mathbb{E}[\mathcal{F}_{\text{prox}}(\alpha)]/\widehat{F}_{max}\Big), & \mathbb{E}[\mathcal{F}(\alpha)]>\widehat{F}_{max},\\ -\beta\Big(\mathbb{E}[\mathcal{F}_{\text{prox}}(\alpha)]/\widehat{F}_{min}\Big), & \mathbb{E}[\mathcal{F}(\alpha)]<\widehat{F}_{min},\\ 0, & \text{otherwise},\end{cases}$$



Experimental Results on *AMF* PDK

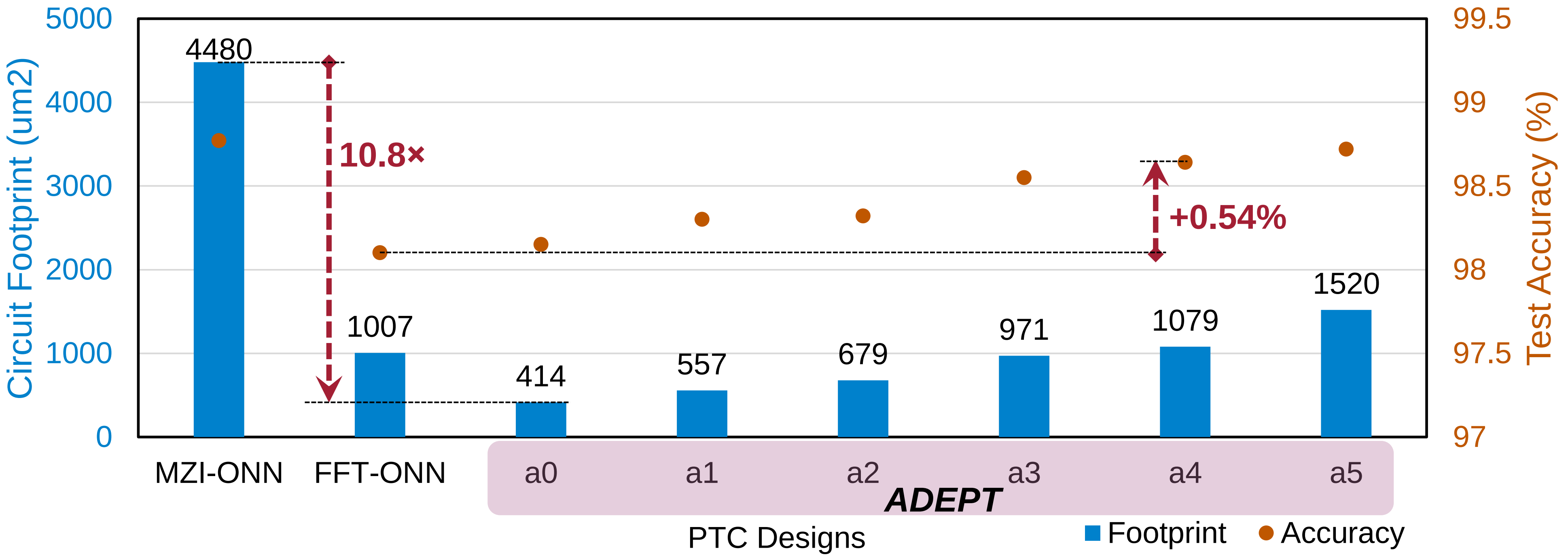
- Comparable expressiveness (<0.5% acc drop)
- 2-30× smaller footprint than MZI-ONN [Nat. Photon'17]
- 2.5× more compact than FFT-ONN [ASP-DAC'20] with higher accuracy



*2-layer CNN on MNIST

Foundry PDK Adaptation

- Adapt to *AIM Photonics* (with much larger CRs than *AMF*)
- 3-11× more compact than MZI-ONN [Nat. Photon'17]
- 0.5% higher accuracy than FFT-ONN [ASP-DAC'20]



*2-layer CNN on MNIST

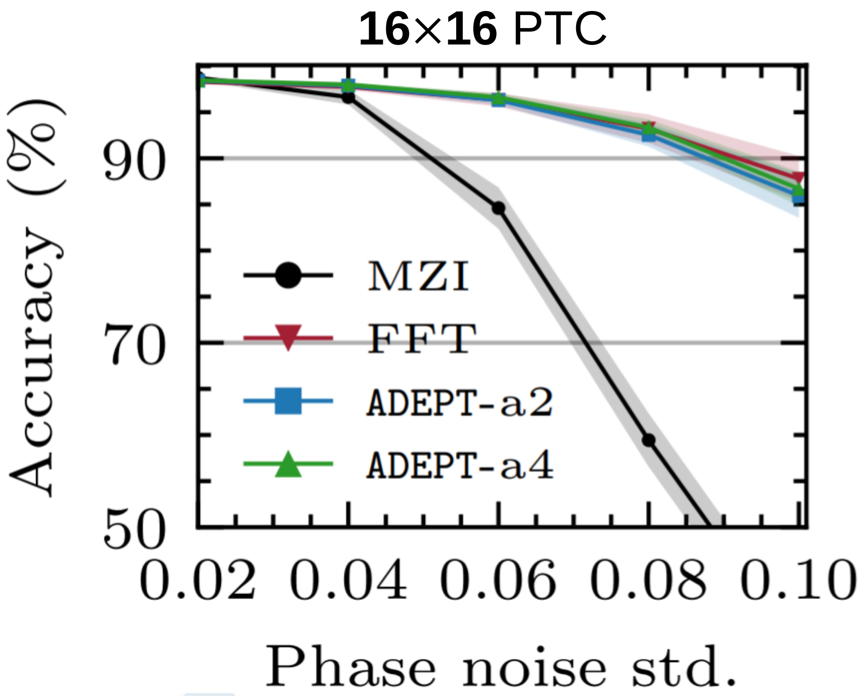
Generalize to Different Tasks/Models

- Search on 2-layer CNN + MNIST → Train on different tasks/models
- 84% smaller than MZI-ONN [Nat. Photon'17] with comparable accuracy
- 26% smaller than FFT-ONN [ASP-DAC'20] with +2.1% higher accuracy

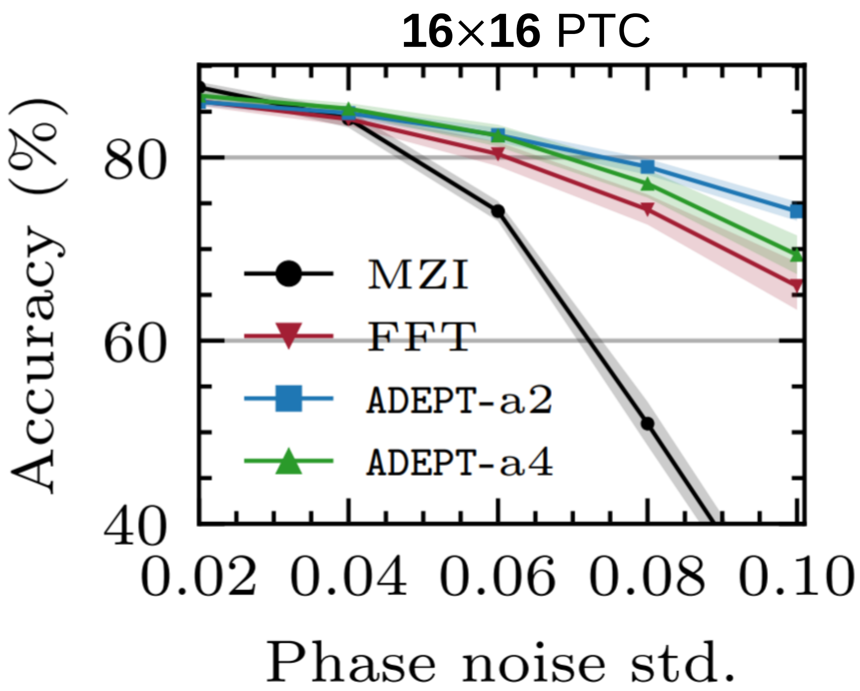
Model	Datasets	MZI [14]	FFT [5, 6]	ADEPT-a2	ADEPT-a4
Footprint		7683	972	722	1206
LeNet-5	FMNIST	87.33	85.87	85.89	87.07
	SVHN	69.91	65.04	65.26	69.20
	CIFAR-10	51.40	42.75	51.26	52.42
VGG-8	FMNIST	89.59	88.62	89.23	89.16
	SVHN	77.87	75.22	75.86	77.20
	CIFAR-10	68.90	63.57	66.30	68.50

Noise Robustness of Searched PTC

- Shallow network depth → Superior noise robustness
- ***ADEPT*** is even more robust than FFT-based butterfly photonic mesh



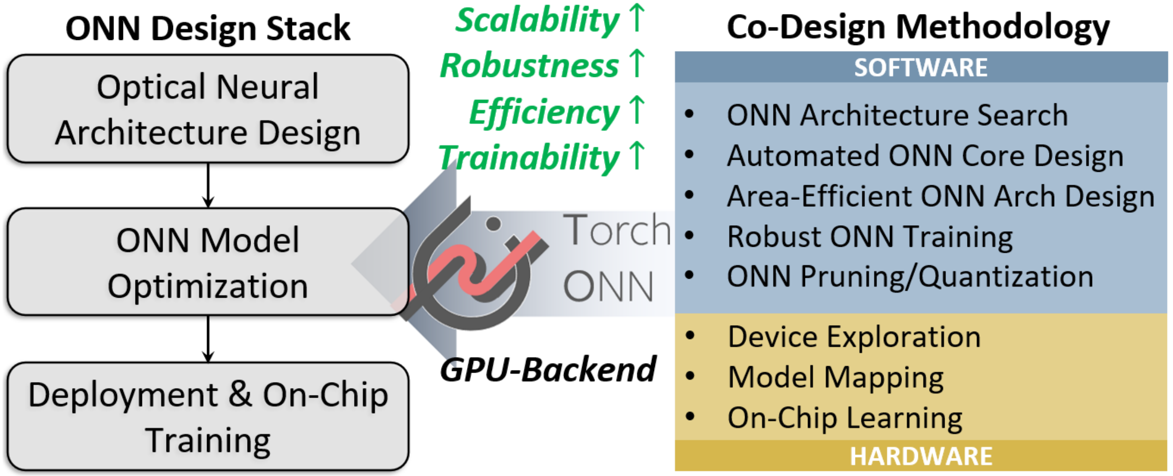
2-layer CNN MNIST



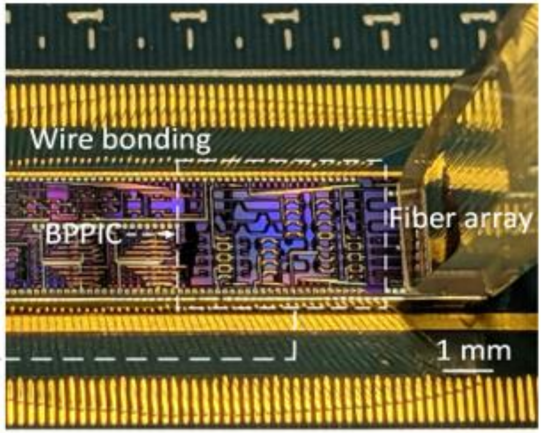
LeNet-5 FashionMNIST

The Future of *Light*-*AI* *Interaction* is **Bright**

- ***ADEPT***: first automatic differentiable PTC search framework
- **Compactness**: 2-30× more compact than MZI-ONN
- **Expressiveness**: Comparable (<0.5%↓) accuracy to MZI-ONN
- **Adaptability**: Adapt to foundry PDKs and area constraints
- **Robustness**: More noise-resilient than FFT-ONN



TorchONN



The University of Texas at Austin

ADVANCED MICRO FOUNDRY

github.com/JeremieMelo/pytorch-onn

Optics for AI ↔ AI for Optics

DESIGN AUTOMATION CONFERENCE 59