APL Machine Learning

RESEARCH ARTICLE | JANUARY 24 2024

M³ICRO: Machine learning-enabled compact photonic tensor core based on programmable multi-operand multimode interference

Jiaqi Gu ; Hanqing Zhu ; Chenghao Feng ; Zixuan Jiang ; Ray T. Chen ; David Z. Pan

APL Mach. Learn. 2, 016106 (2024)
https://doi.org/10.1063/5.0170965

M³ICRO: Machine learning-enabled compact photonic tensor core based on programmable multi-operand multimode interference

Cite as: APL Mach. Learn. 2, 016106 (2024); doi: 10.1063/5.0170965
Submitted: 31 August 2023 • Accepted: 28 December 2023 •
Published Online: 24 January 2024

Jiaqi Gu,[1,2,a] Hanqing Zhu,[1] Chenghao Feng,[1] Zixuan Jiang,[1] Ray T. Chen,[1] and David Z. Pan[1]

AFFILIATIONS

[1] Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, Texas 78712, USA
[2] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA

[a] Author to whom correspondence should be addressed: jiaqigu@asu.edu

ABSTRACT

Photonic computing shows promise for transformative advancements in machine learning (ML) acceleration, offering ultrafast speed, massive parallelism, and high energy efficiency. However, current photonic tensor core (PTC) designs based on standard optical components hinder scalability and compute density due to their large spatial footprint. To address this, we propose an ultracompact PTC using customized programmable multi-operand multimode interference (MOMMI) devices, named M³ICRO. The programmable MOMMI leverages the intrinsic light propagation principle, providing a single-device programmable matrix unit beyond the conventional computing paradigm of one multiply-accumulate operation per device. To overcome the optimization difficulty of customized devices that often requires time-consuming simulation, we apply ML for optics to predict the device behavior and enable differentiable optimization flow. We thoroughly investigate the reconfigurability and matrix expressivity of our customized PTC and introduce a novel block unfolding method to fully exploit the computing capabilities of a complex-valued PTC for near-universal real-valued linear transformations. Extensive evaluations demonstrate that M³ICRO achieves a 3.5–8.9× smaller footprint, 1.6–4.4× higher speed, 9.9–38.5× higher compute density, 3.7–12× higher system throughput, and superior noise robustness compared to state-of-the-art coherent PTC designs. It also outperforms electronic digital A100 graphics processing unit by 34.8–403× higher throughput while maintaining close-to-digital task accuracy across various ML benchmarks.

I. INTRODUCTION

Photonic computing has emerged as a promising technology for high-performance and energy-efficient computing, particularly in computation-intensive artificial intelligence (AI) applications.[1–10] Photonic tensor cores (PTCs) have been developed using standard optical components to enable matrix multiplication in the analog domain at the speed of light, including free-space diffractive designs[10] and integrated photonic circuit-based designs.[1,5,7,8] However, concerns regarding area efficiency and scalability arise due to the large number of bulky components used in existing PTC designs, shown in Figs. 1(a)–1(d). Based on matrix decomposition, general matrix multiplication (GEMM), i.e., universal linear operations, can be mapped to cascaded Mach–Zehnder interferometer (MZI) arrays.[1] The large number of bulky MZIs used in the tensor core raises concerns about area efficiency and scalability. Efforts have been made to reduce the circuit footprint through approaches such as butterfly-style photonic mesh[11–13] with logarithmic network depth, automatically searched circuit topologies,[14] and low-rank MZI arrays.[15] There are also integrated diffractive optical neural networks (DONNs) that leverage on-chip diffractive components for high-parallelism computing.[9,16,17] Moreover, incoherent PTCs based on micro-ring resonator (MRR) weight banks,[18–21] phase-change material (PCM) crossbar arrays,[6,22] and frequency micro-comb have been proposed for compact GEMM using multiple wavelengths. However, the above works are based on standard components designed for optical communications. Their compute density is still limited by approximately one multiply-accumulate (MAC) per device, which intrinsically limits their scalability and efficiency.

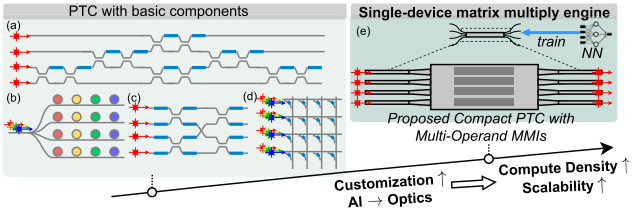

FIG. 1. Overview of photonic tensor core designs with increasing compute density. PTCs with standard devices: (a) MZI array,[1] (b) MRR weight bank,[18] (c) butterfly-style PTC,[11] and (d) PCM crossbar.[6] (e) Our proposed M³ICRO PTC with customized MMI devices and trained with a machine learning-based approach.

To address the limitations of current PTC designs and enhance area efficiency, customized photonic devices tailored for optical computing have attracted attention. Multi-operand (MO) photonic devices have been explored to increase compute density. A compact photonic neuron based on multi-operand rings (MORRs)[12,23] was proposed to squeeze vector dot-product and Lorentzian nonlinear transmission into a single MORR by setting multiple controllers inside the ring, i.e., $y = f\left(\sum_i \phi_i(w_i x_i^2)\right)$. Compared to single-operand MRR weight banks,[18,19] MORR arrays can significantly reduce ring resonator and wavelength usage. In the multi-operand device family, another member, multi-operand MZI (MOMZI),[24] was recently presented to partition the phase shifter in the MZI for vector dot-product with a sinusoidal nonlinear transmission, i.e., $y = \cos\left(\sum_i \phi(w_i x_i)\right)$. By squeezing vector/tensor operations into a single device, multi-operand devices represent a new design paradigm to scale up the compute density of optical computing. However, for previous multi-operand devices, inputs and weights were encoded as the electrical control signals and controller tuning coefficients, respectively. Hence, they face challenges such as limited weight reconfigurability and trainability difficulties associated with nonlinear transmission.

To achieve breakthrough in regard to area efficiency compared to coherent PTCs based on basic devices while overcoming the limitation of existing multi-operand PTCs, we propose a novel coherent multipath PTC design M³ICRO based on customized programmable multi-operand multimode interference (MOMMI) devices, shown in Fig. 1(e). By leveraging the principles of light propagation and interference, combined with fine-grained refractive index tuning within the multimode waveguide, MOMMIs enable the realization of ultracompact programmable analog matrix multiplication cores. Our proposed PTC, equipped with a machine learning (ML)-enabled training flow and a block unfolding method, facilitates efficient and differentiable training of complex-valued coherent PTCs based on customized devices and supports real-valued linear operations.

The contributions of M³ICRO are summarized as follows:

- **Closing the loop of photonics for AI and AI for photonics:** We propose the first ML-enabled programmable photonic tensor core (PTC) based on customized optical devices.
- **Ultracompact single-device optical matrix unit**: We introduce an ultracompact photonic tensor core based on customized programmable MOMMIs, a single-device matrix unit beyond the conventional paradigm of one MAC/device, significantly improving compute density and area efficiency.
- **Superior expressivity and footprint efficiency**: We enhance the expressivity of MOMMIs by developing a multipath PTC architecture called M³ICRO, offering superior matrix representability and improvements in footprint efficiency over previous coherent PTCs.
- **ML-assisted PTC training method**: We propose a novel ML-assisted training method that estimates device gradients and enables differentiable optimization of MOMMIs, eliminating the need for time-consuming simulations and accelerating the training process.
- **Efficient complex PTCs with block unfolding**: We introduce a novel block unfolding technique, achieving efficient, full-range, real-to-real linear transformations with 4× higher efficiency than previous differential photodetection approaches.
- **Significant performance advantages**: Extensive evaluations show that our customized M³ICRO PTC demonstrates near-universal matrix expressivity, close-to-digital accuracy on various ML tasks with 3.5–8.9× smaller footprint, 1.6–4.4× higher speed (TOPS), 9.9–38.5× higher compute density (TOPS/mm²), 3.7–12× higher system throughput frame-per-second (FPS), and superior noise robustness than prior state-of-the-art (SoTA) coherent PTCs. It also outperforms electronic digital Nvidia A100 graphics processing unit (GPU) by 34.8–403× higher throughput. These results highlight the potential of device customization for advancing scalable photonic ML computing.

II. PROPOSED MOMMI-BASED PTC M³ICRO

We introduce a compact photonic tensor core (PTC) design M³ICRO based on customized programmable multi-operand MMI devices. We design a programmable MOMMI and investigate its matrix expressivity. Based on it, we construct the multipath PTC M³ICRO with a compact footprint and near-universal matrix representability. We introduce an efficient ML-based training method for customized photonic devices. Additionally, we present a novel block unfolding method to overcome optimization challenges in complex coherent PTCs.

A. Initial state design of general MMI device

We start our PTC design from an initial MMI structure with a compact footprint, low insertion loss, and near-uniform power splitting ratios. This requires us to carefully determine the width and length of the MMI. Consider a two-dimensional (2D) horizontal plane of an MMI, we denote its length as L, width as W_{MMI}, effective refractive index of the multimode region as n_{eff}, and index of the cladding as n_c. We define $L_\pi \approx \frac{4 n_{eff} W_{e0}^2}{3\lambda_0}$, where W_{e0} is the effective width of the zeroth mode. Based on the dispersion equation,[25] we obtain the propagation constant spacing between the zeroth and v-th mode as $\beta_0 - \beta_v \approx \frac{v(v+2)\pi}{3L_\pi}$. The field profile $\Psi(y,z)$ at the output ports can be written as a superposition of all guided modes at $z = L$, $\Psi(y,L) = \sum_{v=0}^{m-1} c_v \psi_v(y) \exp\left[jL\frac{v(v+2)\pi}{3L_\pi}\right]$. The output field should be

a multiple self-imaging of the input field $\Psi(y,0)$, which holds under the condition that

$$\exp\left[jL\frac{v(v+2)\pi}{3L_\pi}\right] = 1 \text{ or } (-1)^v, \quad L = p(3L_\pi/N), \quad p \in \mathbb{Z}. \tag{1}$$

To obtain an MMI with the shortest length, we set $p = 1$ and $L = 3L_\pi/N$, which corresponds to the first N-fold self-imaging. Based on this initialization, we simulate the figure of merit (FoM) of the MMI, defined as the product of insertion loss and imbalance of power splitting, while performing hyperparameter search on L and W_{MMI} to optimize the FoM. Ideally, the transfer function of a general $k \times k$ MMI corresponds to a symmetric unitary matrix, given its geometric symmetry and energy conservation. For example, after device optimization, the spatial dimensions and transfer matrix of our optimized 4×4 MMI are shown in Fig. 2. We observe a nearly symmetric unitary transfer matrix and a near-uniform power splitting ratio, which is a good initial state of the MMI.

B. M^{3}ICRO: Programmable MOMMI-based PTC

Now we discuss how to make an MMI reprogrammable, and then we will introduce how to construct our M^{3}ICRO tensor core using this customized device.

1. Programmable MOMMI

By changing the refractive index inside the multimode waveguide region, we can program the transfer matrix of the MMI. As shown in Fig. 3, we introduce a customized multi-operand MMI (MOMMI) by placing d tunable regions within an MMI to change their local refractive indices $(\varepsilon_1, \ldots, \varepsilon_d)$. In this way, we can perform fine-grained manipulation of the device transmission. Discussion

$$\mathbf{W}(\epsilon) = \begin{pmatrix} w_{11} & \cdots & w_{1k} \\ w_{21} & \cdots & w_{2k} \\ \vdots & \ddots & \vdots \\ w_{k1} & \cdots & w_{kk} \end{pmatrix} \in \mathbb{C}^{k\times k}, \quad \epsilon \in \mathbb{R}^d$$

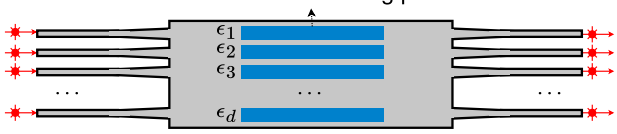

FIG. 3. A d-op $k \times k$ programmable multi-operand MMI.

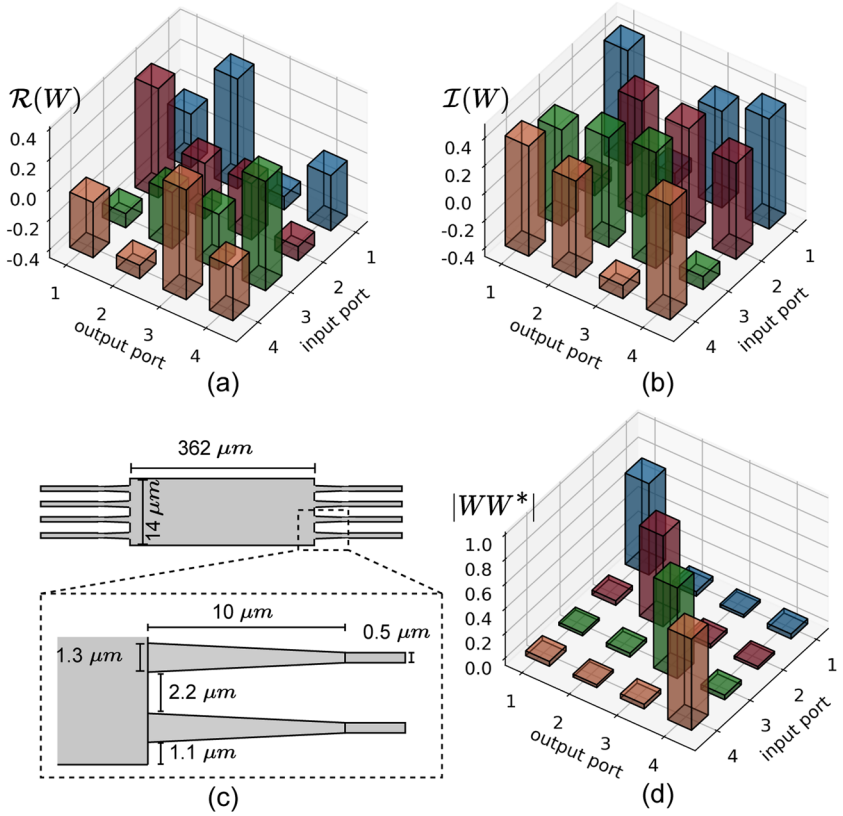

FIG. 2. (a) Real and (b) imaginary parts of the transfer matrix of the optimized 4×4 MMI. (c) Detailed sizes of the MMI. (d) The transfer matrix of the optimized MMI is close to a unitary matrix.

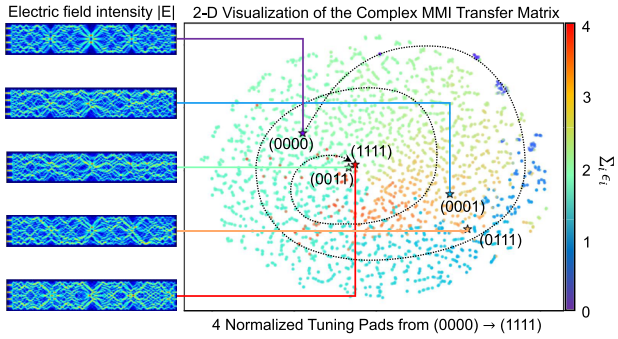

FIG. 4. Visualization of the complex transfer matrix $W(\varepsilon) \in \mathbb{C}^{4\times4}$ of a 4-op 4 × 4 MOMMI in the projected 2D space using t-SNE. Each pad is discretized to eight uniform levels (3-bit) and normalized to [0, 1] by the maximum index change (0.03). Matrices are colored based on $\sum_i \varepsilon_i$.

on the practicality of the device implementation is provided in Sec. III H. Note that the complex-valued transfer matrix $W(\varepsilon)$ of a d-op $k \times k$ MOMMI is reparametrized by d refractive indices, leading to a reduced degree of freedom with only d real latent variables. Therefore, the representable matrices are restricted to a subspace of arbitrary complex matrices. We sweep the refractive indices for each tuning pad and visualize the simulated transfer matrices of a 4-op 3-bit 4 ×4 MOMMI in Fig. 4. A clear spiral-like matrix distribution in the parameter space can be observed as we gradually increase the normalized indices from (0,0,0,0) to (1,1,1,1) with 3-bit resolution on each pad, which represents the implementable matrix subspace.

A single MOMMI itself is an ultracompact matrix unit. However, with a reduced number of parameters ($d < 2k^2$), it shows limited expressivity and lacks flexible controllability over matrix norm and signs, evidenced in Fig. 4. Therefore, we need to enhance its expressivity with a specialized tensor core design.

2. Multipath PTC *M³ICRO*

To enhance the matrix expressivity, we introduce a multipath PTC M³ICRO in Fig. 5, constructed by cascading C blocks of interleaved MOMMIs and modulators with P parallel paths. Each MOMMI serves as an all-to-all channel mixer to create dense signal interactions. The internal diagonal complex matrix $\Sigma \in \mathbb{C}^k$ handles row-column scaling to modulate the matrix norm and signs. The formulation of the multipath PTC M³ICRO is as follows:

$$W = \frac{1}{P}\sum_{p=1}^{P}\left(\prod_{c=2}^{C}(U^{pc}\Sigma^{pc})U^{p1}\right) \in \mathbb{C}^{k\times k}. \quad (2)$$

For a $k \times k$ multipath PTC, instead of having $2k^2$ real parameters in a general complex matrix, it has a reduced number of latent real variables, i.e., $PCd + 2P(C-1)k$. As the architecture design variables of M³ICRO, P and C can be adjusted to trade off hardware efficiency and matrix expressivity. For example, if $(k, d, P, C) = (4, 4, 2, 2)$, it has exactly the same parameter count as a general complex matrix unit, i.e., 32.

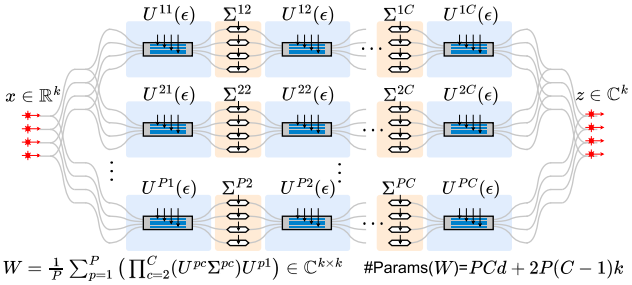

FIG. 5. The proposed MOMMI-based photonic tensor core M³ICRO with P parallel paths and C cascaded components.

C. Efficient complex tensor core via block unfolding

A complex matrix unit seems to have a higher expressivity than a real counterpart since it doubles the parameter count. However, it is often not true when applied to neural networks that require real-valued operations, e.g., activation functions, normalization, pooling, and loss functions. Therefore, to fit into the widely used real-valued DNN paradigm, we need to construct a photonic tensor core that supports *full-range real-valued inputs/outputs*. Previous methods either (1) enforce a real transfer matrix that wastes the multiplication of the imaginary part, e.g., MZI arrays,[1,11,26–28] or (2) remove the phase information by extracting the light intensity through photodetection, which only supports non-negative output.[13,16,17] For case (2), differential photodetection is widely used to create full-range output vectors, i.e., $y = |W_+x| - |W_-x|$, shown in Fig. 6(a). However, this method introduces undesired nonlinearity, which breaks the linear property and leads to optimization difficulty. Moreover, such a method is not efficient as it uses two $k \times k$ complex matrix units while the *effective computing* is one $k \times k$ real matrix-vector multiplication.

To solve those problems caused by complex-valued tensor cores, we propose a block unfolding method to enable *efficient, full-range, real-to-real linear transformation*. Figure 6(b) illustrates the principle of block unfolding. For an N-input M-output real linear layer, we first construct a $\frac{M}{2} \times N$ complex matrix and partition it into a series of $k \times k$ blocks. Each complex submatrix $W_{ij} \in \mathbb{C}^{k\times k}$ is implemented by a $k \times k$ complex PTC. The real and imaginary parts of the output vector $z \in \mathbb{C}^{\frac{M}{2}}$ are unfolded blockwise,

$$y = \texttt{Unfold}(z) = \left[R(z_1); I(z_1); \cdots; R(z_{N/k}); I(z_{N/k})\right]^T \in \mathbb{R}^M. \quad (3)$$

Note that unfolding the output vector is equivalent to unfolding the complex weight matrix $W \in \mathbb{C}^{\frac{M}{2}\times N}$ to a 2× larger real-valued matrix $\widetilde{W} \in \mathbb{R}^{M\times N}$. With this method, we fully leverage the actual computing capability of the tensor core with only $MN/k \cdot \#\text{Params}(W_{ij})$ parameters, which is twice more efficient than enforcing a real transfer matrix and four times more efficient than the differential photodetection method. Note that this method is generic: Any $k \times k$ complex-valued PTC, once equipped with our block unfolding, can support $(2k) \times k$ real matrix multiplication in one shot. The coherent detection with phase and magnitude detection can be implemented by using self-analyzers.[29,30]

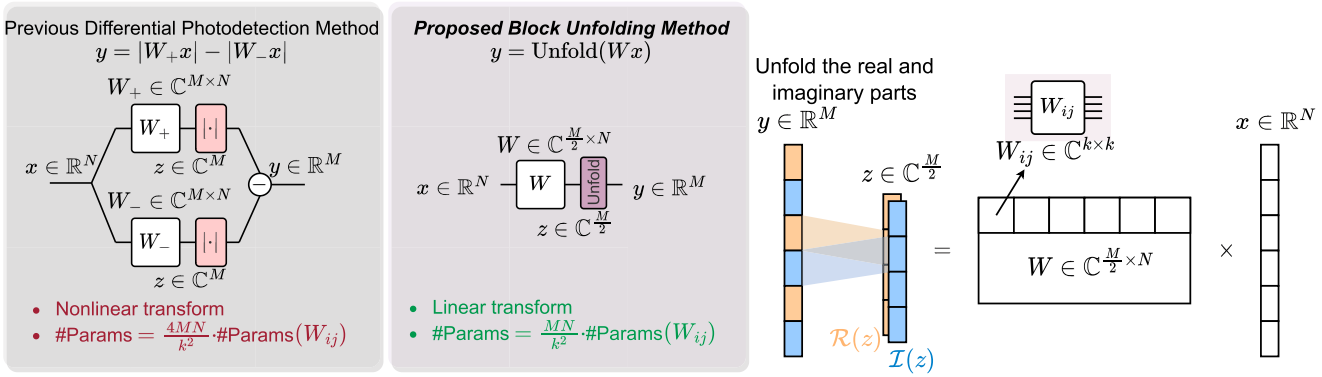

FIG. 6. (a) Compared to previous complex-valued photonic tensor core designs with differential photodetection, our proposed block unfolding method supports pure linear transform with four times fewer parameters. (b) Illustration of block unfolding that interleaves the output vector's real part and imaginary part blockwise.

D. Machine learning-enabled differentiable optimization

Optimizing customized photonic devices is challenging since it relies on time-consuming optical simulation involving Maxwell equations solving, eigenmode decomposition, and S-parameter extraction. Such a complicated process is usually treated as a black-box and cannot be embedded into the outer-loop NN training. To enable efficient optimization of the device variables ε, we employ ML for photonics by introducing a differentiable photonic hardware estimator (DPE):

$$W_\theta = (y + y^T)/2; \quad y = f_\theta(\cos(\omega Q(\varepsilon) + \phi); Q(\varepsilon)), \tag{4}$$

where $f_\theta(\cdot) : \mathbb{R}^d \to \mathbb{C}^{k\times k}$ is a multilayer perceptron, $Q(\varepsilon)$ is the quantized refractive index, ω and ϕ are learnable parameters in the predefined sinusoidal features. The reparameterization on W_θ guarantees a symmetric transfer matrix based on prior knowledge. As shown in Fig. 7, we build a differentiable training method with gradient replacement and straight-through estimator (STE) techniques. In the forward procedure, we quantize the refractive indices to b-bit levels and look up the ground truth table to obtain the transfer matrix $W(\varepsilon)$ for forward propagation. During backward propagation, we redefine the gradient calculation as follows:

$$\frac{\partial \mathscr{L}}{\partial \varepsilon} = \frac{\partial \mathscr{L}}{\partial W(\varepsilon)}\frac{\partial W(\varepsilon)}{\partial Q(\varepsilon)}\frac{\partial Q(\varepsilon)}{\partial \varepsilon} \approx \frac{\partial \mathscr{L}}{\partial W(\varepsilon)}\frac{\partial W_\theta}{\partial Q(\varepsilon)}, \qquad \frac{\partial \mathscr{L}}{\partial x} = W(\varepsilon)^T\frac{\partial \mathscr{L}}{\partial y} \tag{5}$$

where $\frac{\partial Q(\varepsilon)}{\partial \varepsilon}$ is estimated as 1 using STE. Note that only $\frac{\partial W_\theta}{\partial Q(\varepsilon)}$ is calculated by the auto-differentiation through the NN predictor. All other terms during forward and backward propagation are based on $W(\varepsilon)$ to eliminate gradient approximation error accumulation for higher estimation fidelity. Figure 8 visualizes the predicted device behavior and shows superior fidelity with 1.1×10^{-4} mean-square error (MSE) compared to the ground-truth targets. Most importantly, the predictor behaves as a high-quality first-order oracle with a very smooth landscape that can provide reliable and informative first-order gradient information to guide optimization.

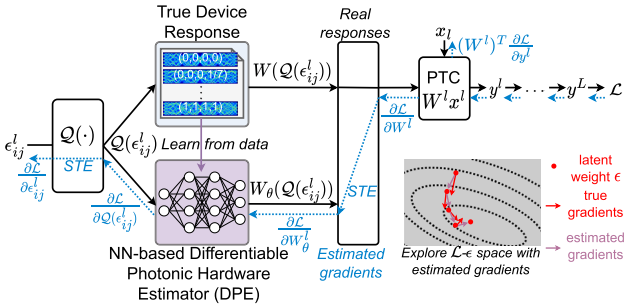

FIG. 7. Proposed ML-enabled differentiable optimization flow for customized programmable MOMMI-based PTCs.

E. Expressivity of programmable MOMMI

To evaluate the matrix expressivity of our multipath MOMMI-based PTC M^{3}ICRO, we perform numerical analysis on different PTC designs in Fig. 9. We randomly generate 40k real matrices from Gaussian distribution, train the differentiable surrogate model of each PTC design with block unfolding to approximate those random real matrices, and then evaluate the average relative ℓ_2 matrix distance as the fidelity, i.e., $F = \frac{1}{N}\sum_{i=1}^{N} \|W_\theta(\varepsilon_i) - \widetilde{W}_i\|_{\mathcal{F}}^2 / \|\widetilde{W}_i\|_{\mathcal{F}}^2$. First of all, the diagonal matrix used for norm and phase tuning is critical to the expressivity. From the expressivity colormap, we can conclude the following trade-off: (1) Increasing the cascading depth C is more effective in boosting the expressivity, but it will significantly increase the circuit depth, leading to higher delay and insertion loss. (2) Increasing the parallel path count P is not as effective in expressivity boost since it only interpolates inside the convex hull of the subspace and also introduces extra signal splitting and combining cost, but it does not increase the critical path length. (3) With large enough photonic mesh width and depth, our M^{3}ICRO can potentially realize 100% matrix expressivity as a universal linear unit. We also compare

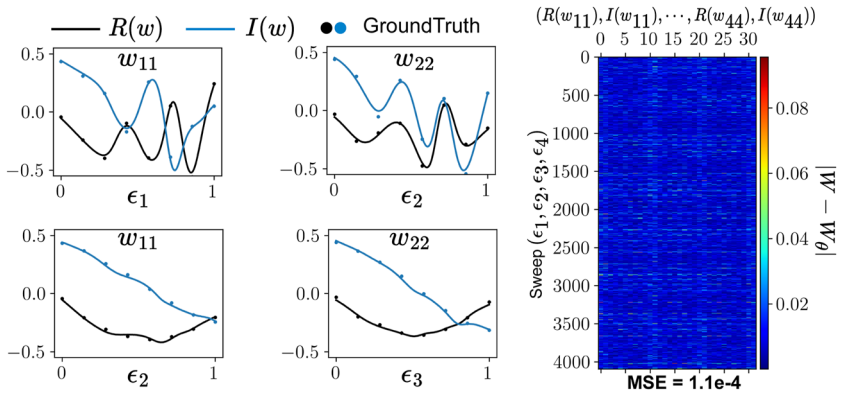

FIG. 8. Visualization of the prediction fidelity of our NN-based device predictor on a 4 × 4 programmable MMI.

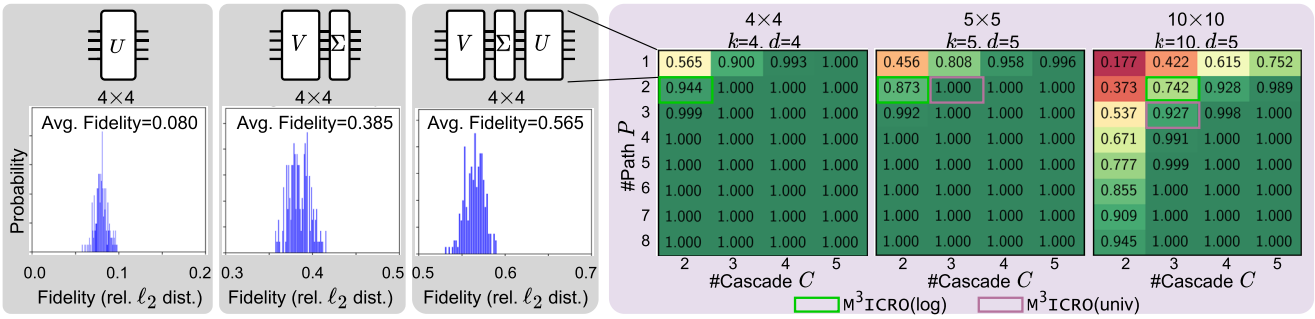

FIG. 9. Numerical analysis on the matrix expressivity (fidelity) of different MOMMI-based PTC designs. The colormap shows how expressivity changes with different number of parallel paths *P* and cascaded blocks *C*.

our M³ICRO with previous PTC designs in Fig. 10 across different matrix sizes. M³ICRO variants have comparable expressivity to that of the universal MZI array and significantly outperform previous compact PTC designs based on FFT[11,16] and trainable butterfly[13,31] topology.

F. Hardware performance and efficiency analysis

Section II E discussed the trade-offs between hardware cost and matrix expressivity with different depth *C* and parallel path count *P*.

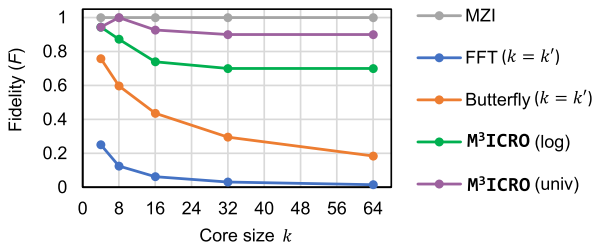

FIG. 10. Comparison of matrix expressivity/fidelity (F) on different PTC designs across various matrix sizes. k' is the butterfly mesh size.

To cover two representative design points for the following discussion, we design a compact variant named M³ICRO (log) and a larger but more expressive variant M³ICRO (univ). For M³ICRO (log), we prioritize area efficiency and target ~70% expressivity. We design M³ICRO (log) as a dual-path PTC, i.e., $P = 2$, with a logarithmic circuit depth $C = \lfloor \log_2 k \rfloor$. For M³ICRO (univ), we prioritize expressivity with >90% fidelity and design it as a near-universal PTC. We empirically set 70% parameter count as a target, assume $P \approx C$, $d = k$, $\alpha = 0.7$, and have

$$PCk + 2P(C-1)k \approx 2\alpha k^2;$$
$$P = \left\lfloor \frac{1+\sqrt{1+6\alpha k}}{3} \right\rceil; C = \left\lceil \frac{1+\sqrt{1+6\alpha k}}{3} \right\rceil. \quad (6)$$

The following analysis mainly focuses on those two variants of M³ICRO architecture.

1. Footprint

We derive the total device footprint of a PTC as

$$A_{\text{total}} = A_{\text{laser}} + (k-1)A_{\text{Y}} + kA_{\text{MZM}} + A_{\text{core}} + kA_{\text{PD}}, \quad (7)$$

where the footprint of the computing core A_{core} is derived in Appendix A and Table IV. A_{laser}, A_{Y}, A_{MZM}, and A_{PD} represent the footprint of laser, Y-branch used for on-chip channel splitting,

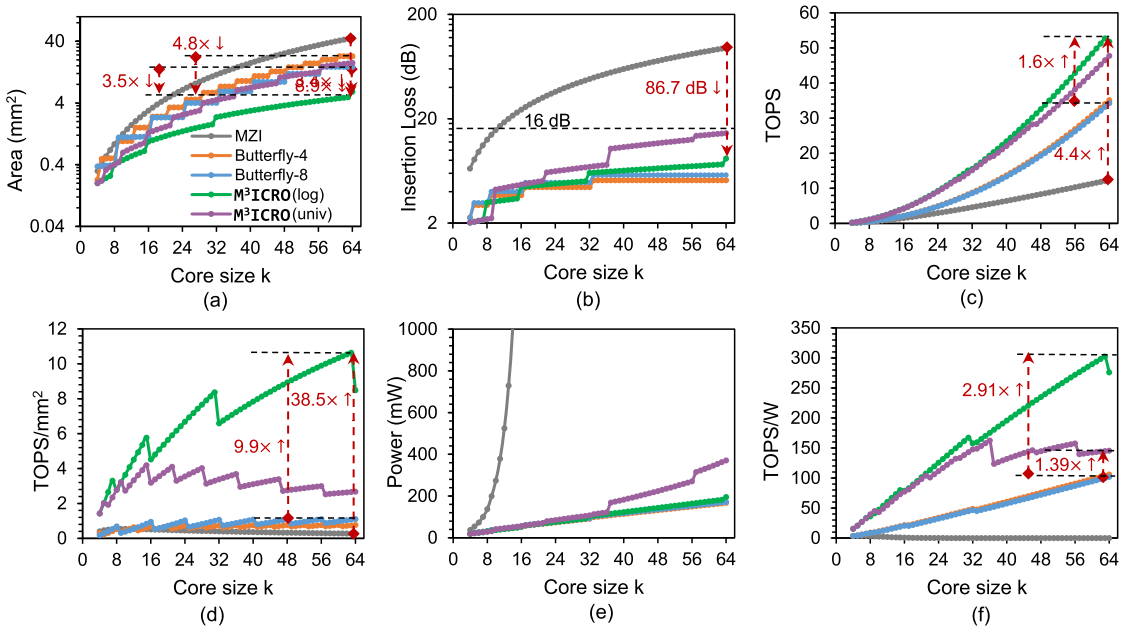

FIG. 11. Comparison of the (a) footprint, (b) insertion loss (IL), (c) computing capacity (TOPS), (d) compute density (TOPS/mm^2), (e) power, and (f) energy efficiency (TOPS/W) among different PTC designs with increasing core sizes. Butterfly PTCs adopt differential photodetection, and our M^{3}ICRO adopts the proposed block unfolding method. The insertion loss excludes the input splitting and signal modulator, while others include laser, splitting, modulators, tensor cores, and photodetectors.

input modulators, and photodetectors. We plot the total footprint A_{total} of different PTCs with increasing core sizes in Fig. 11(a). Our M^{3}ICRO (log) PTC shows good footprint scalability and is 1.6~8.9× more compact than the MZI array, 1.1~4.8× smaller than FFT/butterfly-style PTCs with a block size of 4, and 1~3.5× smaller than FFT/butterfly PTCs with a block size of 8. M^{3}ICRO (univ) generally has a comparably compact footprint to Butterfly-8 PTC.

2. Insertion loss

As an important design metric, circuit insertion loss (IL) impacts the required laser power. High insertion loss fundamentally limits the PTC's power efficiency and scalability. The theoretical insertion loss IL_{core} in the unit of dB of different PTCs is summarized in Table IV. Figure 11(b) shows the insertion loss scalability of different photonic computing cores, excluding signal splitting and input modulators. With a 64 × 64 core size, the MZI mesh has almost 97 dB insertion loss, while our M^{3}ICRO shows less than 16 dB IL. Such a low insertion loss of M^{3}ICRO can fundamentally enable further scaling of larger core sizes with affordable laser power.

3. Peak compute speed and density

To estimate the peak speed, we derive the PTC delay by accumulating the delay from electrical control to the final result readout[13,32] as follows:

$$\tau = \tau_{\text{E-O}} + \tau_{\text{core}} + \tau_{\text{PD}} + \tau_{\text{ADC}}. \tag{8}$$

We assume $\tau_{\text{E-O}}$ = 10 ps for the electrical-to-optical (E-O) conversion, 10 ps for photodetection, and 200 ps for 5 GSPS analog-to-digital conversion (ADC). The optical path delay of the tensor core τ_{core} is derived from the total length of cascaded devices along the critical path, which is summarized in Table IV. The peak computing speed on a $k \times k$ matrix-vector multiplication workload is defined as $2k^2/\tau$. Note that if our block unfolding method is applied, the peak computing speed will double, i.e., $4k^2/\tau$, as it finishes twice the computations in one shot compared to the differential detection method. The peak computing speed (TOPS) of different PTCs with increasing core sizes is compared in Fig. 11(c). Our M^{3}ICRO (log) has 4.4× and 1.6× faster peak computing speed than MZI arrays and butterfly-style PTCs, respectively. The speed can scale up when using a larger core size or wavelength-division multiplexing (WDM) for multi-wavelength parallel computing due to the broadband property of our design.

In terms of area efficiency (compute density), shown in Fig. 11(d), since our M^{3}ICRO is very compact in spatial footprint, it shows 38.5× and 9.9× higher TOPS/mm^2 than MZI arrays and butterfly structures, respectively.

4. Power and energy efficiency

The power of the photonic tensor core is mainly comprised of four parts, i.e., laser, input modulators, weight programming in the core, and photodetection,

$$P_{\text{total}} = P_{\text{laser}} + P_{\text{mod}} + P_{\text{wt}} + P_{\text{PD}}. \tag{9}$$

The weight programming power P_{wt} is zero when using nonvolatile phase shifters.[33] The input modulation power P_{mod} and detection power P_{PD} are the same for all coherent PTCs using MZMs. Given the photodetector sensitivity S, ADC resolution of b-bit (we assume 8-bit here), and laser wall-plug efficiency η, the required wall-plug

power is $P_{\text{laser}} = 10^{(S+IL)/10} \times 2^b/\eta$, where the total insertion loss IL includes the loss of the computing core IL_{core} (in Table IV) and the loss of Y-branch splitting tree and input MZMs for k channels, i.e.,

$$IL = \log_2 k \cdot IL_Y + IL_{\text{MZM}} + IL_{\text{core}}. \tag{10}$$

The detailed device parameters used in the calculation are listed in Table III. With different core sizes k, we show the power consumption of different designs in Fig. 11(e). Butterfly-style PTCs and our M^{3}ICRO have much lower insertion loss than MZI meshes, which shows considerably better power scalability to larger core sizes. The energy efficiency is defined as the ratio of peak computing speed to power (TOPS/W). Figure 11(f) shows our 64×64 M^{3}ICRO (log) and M^{3}ICRO (univ) architectures have 289.9 TOPS/W and 128.4 TOPS/W energy efficiency, outperforming butterfly-style PTCs by 2.91× and 1.39×, respectively.

III. EVALUATION

We conduct various simulation-based evaluations on our M^{3}ICRO PTC designs in terms of expressivity, quantization tolerance, and noise robustness. We mainly compare M^{3}ICRO with (1) MZI array,[1] (2) FFT-based PTC with fixed optical Fourier transform modules,[11,16] and (3) butterfly-style PTC with trainable butterfly transforms.[12,13] Note that we do not compare with other multi-operand tensor cores since they are incoherent architectures with nonlinear transmissions and limited training scalability, especially on large NN models. We also show the effectiveness of our block unfolding method.

A. Training setups

We train optical neural network models based on the open-source library TorchONN and adopt the same settings for all PTC designs. We first train a software digital NN model and use it as a teacher model T. Its optical analog version is called the student model S. As an initialization, we map the teacher's weight matrices W_{ij}^T blockwise to the student counterpart $W_{ij}(\varepsilon, \Sigma)$ by solving the optimization problem $\varepsilon^{init}, \Sigma^{init} = \arg\min_{\varepsilon,\Sigma} \sum_{i,j} \| W_{ij}^T - W_{ij}(\varepsilon, \Sigma) \|_2^2$. After mapping, we fine-tune the student with knowledge distillation, $\min_{\varepsilon,\Sigma} \mathcal{L}_{CE}(y^S, \hat{y}) + \eta\beta^2 D_{KL}\left(\frac{y^S}{\beta}, \frac{y^T}{\beta}\right)$, where $\mathcal{L}_{CE}$ is the cross-entropy loss between the student predictions and the labels, D_{KL} is the KL divergence between student and teacher predictions, β is the temperature ($\beta = 2$), and η is set to 0.1 to balance two loss functions. During the 3000-step mapping stage, we use Adam optimizer with an initial learning rate of 1×10^{-2} for Σ and 1×10^{-3} for ε. Cosine learning rate decay is adopted. The fine-tuning stage learning rate is set to 3×10^{-4} for Σ and 4×10^{-4} for ε. The NN device predictor is a 6-layer MLP: $(2k)$-$(256)_{\times 3}$-$(128)_{\times 2}$-$(2k^2)$.

B. Accuracy evaluation

Table I shows a comprehensive comparison among different PTC designs on three different NN models and image classification datasets. The universal MZI array represents the ideal software NN accuracy. We observe unsatisfying FFT-based PTC due to its fixed Fourier transform and limited matrix expressivity. The butterfly PTC has trainable phases in the butterfly transform, showing enhanced accuracy on different tasks compared to the FFT designs. Across different MOMMI sizes, our M^{3}ICRO (log) and M^{3}ICRO (univ) series outperform the compact butterfly designs on all benchmarks. Our specially designed universal M^{3}ICRO variants show the best accuracy. Even with 10×10 MOMMIs, the universal variant maintains >0.9 matrix expressivity with <0.5% accuracy degradation compared to the ideal digital software model.

C. Quantization tolerance evaluation

In practice, the index tuning precision inside the MOMMI is quantized for control efficiency consideration. In Fig. 12, we illustrate the impact of ε bitwidth on the PTC matrix expressivity and the corresponding accuracy on the ResNet-20 CIFAR-10 benchmark. To stabilize the optimization of discrete device control variables ε, we set the following initial learning rate to $\min(\alpha_0, \alpha_0 \times 2^{b-2})$, $\alpha_0 = 5e-6$. For M^{3}ICRO (log)-4, the expressivity drops with fewer bitwidth while the task accuracy can maintain a value <1% drop with above 4-bit resolution, which is suitable for efficient device control. For M^{3}ICRO (univ)-5, the fidelity maintains a value >0.8 even with a fixed MOMMI (0-bit), and the accuracy can almost maintain the same value with more than 2-bit. Binary and fixed MOMMIs suffer

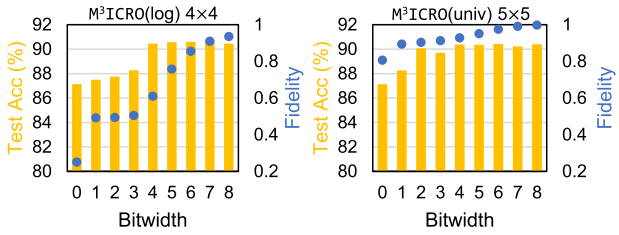

FIG. 12. Matrix expressivity and test accuracy under ε quantization on ResNet-20 CIFAR-10 for 4×4 M^{3}ICRO (log) and 5×5 M^{3}ICRO (univ). 0-bit means the MOMMI is fixed to its initial state. Activations are quantized to 8-bit.

TABLE I. Comparison of accuracy across different PTC designs on various models and datasets. Boldface values represent the best accuracy.

	MZI[1]	FFT-4[11]	FFT-8[11]	Butterfly-4[13]	Butterfly-8[13]	M^{3}ICRO (log)-4	M^{3}ICRO (log)-5	M^{3}ICRO (univ)-5	M^{3}ICRO (log)-10	M^{3}ICRO (univ)-10
ResNet20-CIFAR10	90.29	86.04	82.94	90.38	88.27	90.56	90.36	**90.77**	90.26	90.10
ResNet18-CIFAR100	73.45	70.88	67.25	72.63	72.01	74.18	**74.22**	74.00	72.05	73.53
MobileNetV3-SVHN	95.57	95.2	94.65	95.57	95.03	**95.61**	95.38	95.59	95.19	95.28

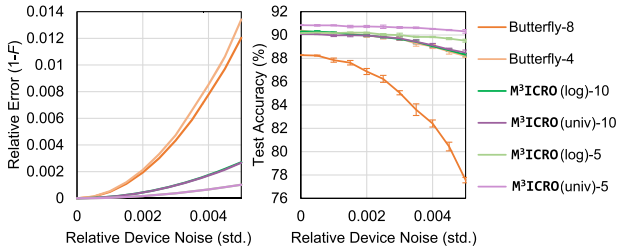

FIG. 13. Noise robustness evaluation for different tensor core designs on ResNet-20 CIFAR10 with various device noise intensity. All tensor cores adopt the proposed block unfolding method. Error bars show the accuracy standard deviation. The proposed M³ICRO-series shows superior robustness compared to previous SoTA butterfly-style PTCs.

from overly limited matrix expressivity, which further necessitates and proves the superiority of the programmability of our MOMMI device over previous passive/fixed designs.[11,16] In practical settings, 4-bit to 8-bit resolutions are considered efficient and practical settings for most analog ML accelerators. Overall, our MOMMI-based PTC M³ICRO shows great task accuracy and quantization tolerance with 4- to 8-bit device controls.

D. Device noise robustness evaluation

We evaluate the noise tolerance of our proposed M³ICRO PTC design against random index perturbation from nonideal control signals or environmental variations. We mainly compare with butterfly PTC since it has the SoTA noise tolerance due to its logarithmic network depth.[12,13] In Fig. 13, we first compare the relative matrix ℓ_2 error, i.e., $1 - F$, caused by various device noise intensities. The noise is sampled from $\Delta\varepsilon \sim \mathcal{N}(0, \sigma^2)$ for the index of M³ICRO with the maximum tuning range of 1 and $\Delta\phi \sim \mathcal{N}(0, (2\pi\sigma)^2)$ for the phases in the butterfly PTC with a maximum tuning range of 2π.[34] We observe significantly lower sensitivity of M³ICRO compared to that of butterfly designs. We further evaluate the accuracy degradation on ResNet-20 CIFAR-10. All M³ICRO variants outperform the butterfly designs with better noise tolerance.

E. Ablation study on block unfolding

Table II compares PTCs with differential photodetection and block unfolding on different benchmarks. Differential photodetection consumes four times the parameters and hardware cost to perform a nonlinear real-to-real transformation with a balanced output range. It leads to significant optimization difficulty, leading to severe accuracy drop or even divergence on MobileNetV3. In contrast, our block unfolding achieves close-to-digital accuracy because it enables a real-to-real full-range linear transform, which is compatible with direct weight matrix mapping without optimization instability issues.

F. Advance compute density vs efficiency Pareto frontier

In Fig. 14, we plot different NN hardware designs in the compute density (TOPS/mm²) and energy efficiency (TOPS/W) space, including analog electronics,[35–37] digital electronics,[38–42] and analog

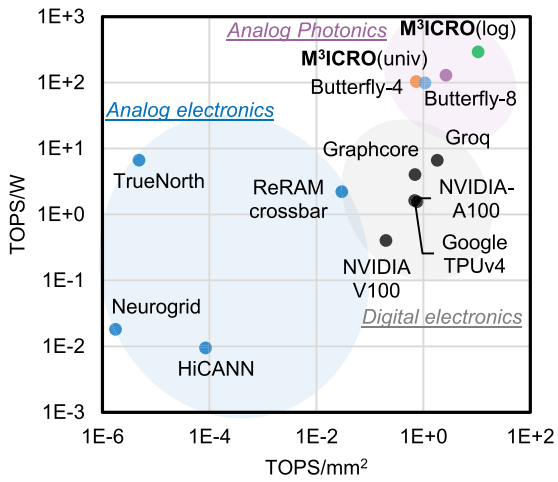

FIG. 14. Compute density vs energy efficiency Pareto frontier of different hardware technologies, including analog electronics, digital electronics, and analog photonics (64 × 64 cores). Our M³ICRO achieves the best Pareto frontier.

TABLE II. Evaluation of the effectiveness of our proposed block unfolding (*unfold*) and previous differential photodetection (*diff*) method. *div* means divergence due to instability caused by nonlinear absolute operations.

		FFT-4[11]		Butterfly-4[13]		M³ICRO (log)-4	
		Diff	Unfold	Diff	Unfold	Diff	Unfold
ResNet-20 CIFAR100	Acc	85.17	86.04	86.86	90.38	88.44	90.56
	#Params	0.27 M	67.5 K	1.09 M	0.27 M	1.09 M	0.27 M
ResNet-18 CIFAR100	Acc	67.2	70.88	68.77	72.63	70.57	74.18
	#Params	11.22 M	2.81 M	44.85 M	11.22 M	44.85 M	11.22 M
MobileNetV3 SVHN	Acc	Div	95.2	Div	95.57	Div	95.61
	#Params	1.53 M	0.396 M	6.08 M	1.54 M	6.08 M	1.54 M

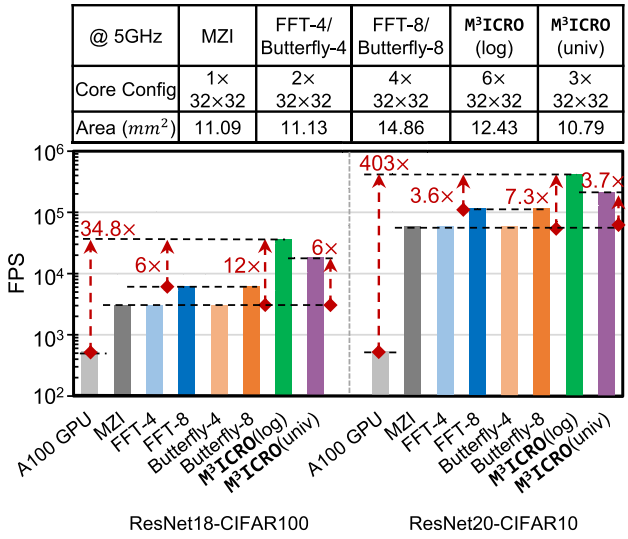

FIG. 15. Comparison of the system-level throughput of the single-example inference task in frame-per-second (FPS) among PTCs (@5 GHz clock) and NVIDIA A100 GPU. The FPS of A100 is measured by the benchmarking tool from PyTorch with mixed precision.

photonic tensor cores. Note that PTCs are configured to have a single 64 × 64 core, much smaller than electronics counterparts with multiple large-size (>1024) cores. Analog electronics have relatively high energy efficiency with low compute density. SoTA digital processors, e.g., TPUv4[40] and A100 GPU,[39] show comparable energy efficiency with around 1 TOPS/mm^2 area efficiency. Analog photonic tensor cores outperform SoTA digital electronics by over two orders of magnitude in energy efficiency, while the compute density is still around 1 TOPS/mm^2. With customized MOMMI devices, our M^{3}ICRO designs, especially the M^{3}ICRO (log) variant, show 3–10 TOPS/mm^2 compute density, significantly advancing the Pareto frontier. With more compact MMI designs and multiple wavelengths, the compute density of M^{3}ICRO can potentially reach an even higher level.

G. System throughput comparison

We use an internal system-level photonic accelerator simulator to evaluate the throughput of different PTC designs in Fig. 15. The detailed architectural simulation is provided in Appendix B. We adjust the core configurations to maintain similar area budgets for all PTCs for fair comparison. Our compact MOMMI-based design equipped with block unfolding allows more cores on chip while boosting the effective computing speed. Our M^{3}ICRO variants, on average, show 3.7–12× higher throughput (FPS) than baseline PTCs and 34.8–403× higher throughput than NVIDIA A100 GPU.

H. Discussion on implementation of programmable MOMMI

The practicality of index tuning for a multimode waveguide has been widely discussed in the literature.[43–45] In this work, the MOMMI device is designed for weight-static linear transformation, which does not require high-speed modulation. Hence, we can use existing low-speed phase modulators as tuning pads. For example, we can put thermal tuning pads on top of the multimode waveguide region, which has been experimentally demonstrated.[44,45] To reduce the power consumption, we can also use nonvolatile phase-change materials (PCMs)[46] or liquid crystal (LC)[43] as the tuning pads with high index contrast and low static power consumption. If the application requires high-speed weight reconfiguration, we can adopt electro-optic (EO) index-tuning materials, such as thin-film lithium niobate.[47]

IV. CONCLUSION

In this study, we propose the first machine learning-enabled multipath photonic tensor core M^{3}ICRO based on customized programmable multi-operand multimode interference devices. We thoroughly investigate its matrix expressivity and enable efficient PTC optimization with an ML-based training method. We further introduce a block unfolding technique to enable full-range real-to-real linear transform for complex-valued PTC with four times higher efficiency than the differential photodetection approach. Extensive evaluation shows that our customized M^{3}ICRO PTC has close-to-digital task accuracy, 1.6–4.4× higher speed, 9.9–38.5× higher compute density, superior noise robustness, and 3.7–12× higher system throughput than previous SoTA coherent PTCs and that it is 34.8–403× faster than A100 GPU. This study opens up new possibilities for device customization and strengthens the integration of photonics and machine learning, driving the scalability and efficiency of photonic ML computing.

ACKNOWLEDGMENTS

The authors acknowledge the support provided by Multidisciplinary University Research Initiative (MURI) program through the Air Force Office of Scientific Research (AFOSR), Contract No. FA 9550-17-1-0071, monitored by Dr. Gernot S. Pomrenke.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Jiaqi Gu: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Validation (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Hanqing Zhu**: Data curation (equal); Writing – review & editing (equal). **Chenghao Feng**: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Writing – review & editing (equal). **Zixuan Jiang**: Formal analysis (equal); Writing – review & editing (equal). **Ray T. Chen**: Funding acquisition (equal); Project administration (equal); Resources (equal); Software (equal); Supervision (equal). **David Z. Pan**: Funding acquisition (equal);

TABLE III. Adopted component parameters in M³ICRO. IL represents insertion loss.

Device	Parameter	Value
Crossing (CR)[49]	IL	0.02 dB
	Area	$7.4 \times 7.4\ \mu m^2$
Phase shifter (PS)[33]	IL	0.04 dB
	Area	$90 \times 40\ \mu m^2$
	Response time	10 ns
	Static power	0 mW
Y-branch (Y)[50]	IL	0.3 dB
	Area	$1.8 \times 1.3\ \mu m^2$
2 × 2 beam splitter (BS)[51]	IL	0.33 dB
	Area	$29.3 \times 2.4\ \mu m^2$
4 × 4 MMI[51]	IL	0.33 dB
	Area	$55.4 \times 4.8\ \mu m^2$
MZM	Tuning power	2.25 mW[52]
	IL	1.2 dB[53]
	Area	$260 \times 20\ \mu m^2$[53]
Photodetetcor[54]	Power	1.1 mW
	Sensitivity	−25 dBm
	Area	$4 \times 10\ \mu m^2$
Laser[55]	Wall-plug efficiency	0.2
	Area	$400 \times 300\ \mu m^2$

Project administration (equal); Resources (equal); Software (equal); Supervision (equal).

DATA AVAILABILITY

The data that support the findings of this study are openly available in M3ICRO-MOMMI.[48]

APPENDIX A: FOOTPRINT, INSERTION LOSS, AND DELAY OF PHOTONIC TENSOR CORE

In Table IV, we summarize the theoretical footprint A_{core}, insertion loss IL_{core}, and delay (latency) τ_{core} of different photonic tensor core designs with a core size of $k \times k$, which is used in Sec. II F. A list of parameters used in the performance and efficiency calculation is provided in Table III. The A_{core}, IL_{core}, and τ_{core} are used in the calculation of total footprint, total insertion loss, and total delay in Sec. II F.

APPENDIX B: SYSTEM-LEVEL PERFORMANCE SIMULATION

We adopt a system-level photonic accelerator simulator to simulate the performance and efficiency.[56] The multicore architecture has a DRAM, global static random access memory (SRAM) buffer, input/activation SRAM buffers for each core, and multiple photonic tensor cores. Optical interconnect is assumed for inter-core input operand broadcast. The area, leakage power, and access energy of the 14 nm memory hierarchy are modeled by PCACTI.[57] High-bandwidth memory (HBM) is used to supply data to the photonic

TABLE IV. Footprint (A_{core}), insertion loss (IL_{core}), and delay (τ_{core}) analysis of photonic tensor cores. A is the footprint, IL is the insertion loss, and L is the device length. W_0L_0 is the area of a reference $k_0 \times k_0$ MMI. We assume the MMI size scales with k^2 based on Eq. (1). FFT/Butterfly-k' means that the PTC is of size k'. If $k > k'$, the matrix is chunked into $(k/k') \times (k/k')$ blocks of size $k' \times k'$. #CR(k') and #CCR(k') are the total crossing count and the number of cascaded crossings in the critical path, respectively. n_g is the group index and c is the free-space light speed.

Design	Metric	Value
MZI[1]	Footprint (A_{core})	$k^2(3A_{PS} + 2A_{BS})$
	IL (IL_{core})	$(2k+1)\ (2IL_{BS} + 2IL_{PS})$
	Delay (τ_{core})	$(2k+1)\ (2L_{BS} + 2L_{PS})n_g/c$
FFT-k'[11] Butterfly-k'[13]	Footprint (A_{core})	$\lceil k/k' \rceil^2 \left(k'(\log_2 k' + 2)A_{BS} + k'(2\log_2 k' + 2)A_{PS} + \#CR(k') \cdot A_{CR}\right) + 2k(\lceil k/k' \rceil - 1)A_Y$
	IL (IL_{core})	$2\lceil \log_2(k/k') \rceil IL_Y + (2\log_2 k' + 2)(IL_{BS} + IL_{PS}) + \left(2\lceil \log_2(k/k') \rceil (k'-1) + \#CCR(k')\right) IL_{CR}$
	Delay (τ_{core})	$\left(2\lceil \log_2(k/k') \rceil L_Y + (2\log_2 k' + 2)(L_{BS} + L_{PS}) + \left(2\lceil \log_2(k/k') \rceil (k'-1) + \#CCR(k')\right) L_{CR}\right) n_g/c$
M³ICRO (log)	Footprint (A_{core})	$2\lfloor \log_2 k \rfloor L_0 W_0 k^2/k_0^2 + 4k(\lfloor \log_2 k \rfloor - 1)(A_{PS} + A_Y) + 2kA_Y + k(k-1)A_{CR}$
	IL (IL_{core})	$2 \cdot IL_Y + \lfloor \log_2 k \rfloor \cdot IL_{MMI} + (\lfloor \log_2 k \rfloor - 1)\ (2IL_Y + IL_{PS}) + 2(k-1)IL_{CR}$
	Delay (τ_{core})	$(2 \cdot L_Y + \lfloor \log_2 k \rfloor \cdot L_0 k/k_0 + (\lfloor \log_2 k \rfloor - 1)\ (2L_Y + L_{PS}) + 2(k-1)L_{CR})n_g/c$
M³ICRO (univ)	Footprint (A_{core})	$PCL_0 W_0 k^2/k_0^2 + 2kP(C-1)(A_{PS} + A_Y) + 2(P-1)kA_Y + (P-1)k(k-1)A_{CR};\ P, C = (6)$
	IL (IL_{core})	$2\lceil \log_2 P \rceil \cdot IL_Y + C \cdot IL_{MMI} + (C-1)\ (2IL_Y + IL_{PS}) + 2\lceil \log_2 P \rceil (k-1)IL_{CR};\ P, C = (6)$
	Delay (τ_{core})	$(2\lceil \log_2 P \rceil \cdot L_Y + C \cdot L_0 k/k_0 + (C-1)\ (2L_Y + L_{PS}) + 2\lceil \log_2 P \rceil (k-1)L_{CR})n_g/c$

system with a memory system bandwidth >1 TB/s.[58] We use the ADC[59] and DAC[60] with 14 and 16 nm technology nodes, respectively, while their bitwidths and frequency are scaled accordingly.[61] The scheduling of the multicore accelerator adopts the weight-stationary dataflow to amortize the PTC weight reprogramming cost.

APPENDIX C: LIMITATIONS AND FUTURE DIRECTIONS

Potential limitations of our proposed design are

- **Subspace tensor core**: Our MOMMI's transfer matrix cannot exactly express arbitrary matrix. The relation between ε and transfer matrix W is limited by the working principle of MMI. Theoretically, our multipath PTC M^{3}ICRO, though it can have multiple MMIs cascaded and connected in parallel, only numerically approximates a target matrix with a low error, which does not have a theoretical guarantee to express the exact matrix. Mapping a target matrix to our designs requires optimization-based mapping.
- **Weight-static tensor core**: The transfer matrix of M^{3}ICRO is not a simple explicit function of ε, which is a complicated function learned by the neural network-based DPE. Hence, it can only be trained as a static weight for weight-stationary linear operation, e.g., $Z = Wx$ in linear/convolution layers in neural networks. A dynamic tensor product, e.g., $Z = XY$ in self-attention operations, cannot be realized, as the arbitrary dynamic tensor X needs to be mapped to M^{3}ICRO through training, which cannot be efficiently performed in real time.

To further validate our design and improve its performance/efficiency, here are several future directions:

- Experimentally demonstrate the usage of the proposed MOMMI and M^{3}ICRO PTC for real-world machine learning tasks.
- Explore other tuning mechanisms with different tuning pad geometries, locations, and materials, and customize the MMI structure to reduce static power consumption, reduce the size, and increase performance/robustness.
- Combine MOMMI with other multi-operand devices to realize more scalable tensor computations.

REFERENCES

[1] Y. Shen, N. C. Harris, S. Skirlo *et al.*, "Deep learning with coherent nanophotonic circuits," Nat. Photonics **11**, 441 (2017).

[2] Q. Cheng, J. Kwon, M. Glick, M. Bahadori, L. P. Carloni, and K. Bergman, "Silicon photonics codesign for deep learning," Proc. IEEE **108**, 1261 (2020).

[3] G. Wetzstein, A. Ozcan, S. Gigan, S. Fan, D. Englund, M. Soljačić, C. Denz, D. A. B. Miller, and D. Psaltis, "Inference in artificial intelligence with deep optics and photonics," Nature **588**, 39 (2020).

[4] B. J. Shastri, A. N. Tait, T. F. de Lima, W. H. P. Pernice, H. Bhaskaran, C. D. Wright, and P. R. Prucnal, "Photonics for artificial intelligence and neuromorphic computing," Nat. Photonics **15**, 102 (2021).

[5] X. Xu, M. Tan, B. Corcoran, J. Wu, A. Boes, T. G. Nguyen, S. T. Chu, B. E. Little, D. G. Hicks, R. Morandotti, A. Mitchell, and D. J. Moss, "11 TOPS photonic convolutional accelerator for optical neural networks," Nature **589**, 44 (2021).

[6] J. Feldmann, N. Youngblood, M. Karpov, H. Gehring, X. Li, M. Stappers, M. Le Gallo, X. Fu, A. Lukashchuk, A. Raja, J. Liu, D. Wright, A. Sebastian, T. Kippenberg, W. Pernice, and H. Bhaskaran, "Parallel convolutional processing using an integrated photonic tensor core," Nature **589**, 52 (2021).

[7] C. Huang, S. Fujisawa, T. F. de Lima *et al.*, "A silicon photonic–electronic neural network for fibre nonlinearity compensation," Nat. Electron. **4**, 837 (2021).

[8] B. Bai, Q. Yang, H. Shu, L. Chang *et al.*, "Microcomb-based integrated photonic processing unit," Nat. Commun. **14**, 66 (2023).

[9] T. Fu, Y. Zang, Y. Huang, Z. Du *et al.*, "Photonic machine learning with on-chip diffractive optics," Nat. Commun. **14**, 70 (2023).

[10] T. Zhou, X. Lin, J. Wu, Y. Chen *et al.*, "Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit," Nat. Photonics **15**, 367 (2021).

[11] J. Gu, Z. Zhao, C. Feng *et al.*, "Towards area-efficient optical neural networks: An FFT-based architecture," in *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)* (IEEE, 2020).

[12] J. Gu, C. Feng, H. Zhu, Z. Zhao, Z. Ying, M. Liu, R. T. Chen, and D. Z. Pan, "Squeezelight: A multi-operand ring-based optical neural network with cross-layer scalability," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)* (IEEE, 2022).

[13] C. Feng, J. Gu, H. Zhu, Z. Ying, Z. Zhao, D. Z. Pan, and R. T. Chen, "A compact butterfly-style silicon photonic-electronic neural chip for hardware-efficient deep learning," ACS Photonics **9**, 3906–3916 (2022).

[14] J. Gu, H. Zhu, C. Feng, Z. Jiang, M. Liu, S. Zhang, R. T. Chen, and D. Z. Pan, "ADEPT: Automatic differentiable design of photonic tensor cores," in ACM/IEEE Design Automation Conference (DAC) 2022.

[15] X. Xiao, M. B. On, T. Van Vaerenbergh, D. Liang, R. G. Beausoleil, and S. J. B. Yoo, "Large-scale and energy-efficient tensorized optical neural networks on III–V-on-silicon MOSCAP platform," APL Photonics **6**, 126107 (2021).

[16] H. H. Zhu, J. Zou, H. Zhang, Y. Z. Shi, S. B. Luo *et al.*, "Space-efficient optical computing with an integrated chip diffractive neural network," Nat. Commun. **13**, 1044 (2022).

[17] Z. Wang, L. Chang, F. Wang, T. Li, and T. Gu, "Integrated photonic metasystem for image classifications at telecommunication wavelength," Nat. Commun. **13**, 2131 (2022).

[18] A. N. Tait, T. F. de Lima, E. Zhou *et al.*, "Neuromorphic photonic networks using silicon photonic weight banks," Sci. Rep. **7**, 7430 (2017).

[19] W. Liu, W. Liu, Y. Ye, Q. Lou, Y. Xie, and L. Jiang, "Holylight: A nanophotonic accelerator for deep learning in data centers," in *IEEE/ACM Proceedings Design, Automation and Test in Europe (DATE)* (IEEE, 2019).

[20] F. Sunny, M. Nikdast, and S. Pasricha, "Reclight: A recurrent neural network accelerator with integrated silicon photonics," in *IEEE Annual Symposium on VLSI (ISVLSI)* (IEEE, 2022), p. 6.

[21] F. Sunny, A. Mirza, M. Nikdast, and S. Pasricha, "Crosslight: A cross-layer optimized silicon photonic neural network accelerator," in *ACM/IEEE Design Automation Conference (DAC)* (IEEE, 2021), pp. 1069–1074.

[22] M. Miscuglio and V. J. Sorger, "Photonic tensor cores for machine learning," Appl. Phys. Rev. **7**, 031404 (2020).

[23] J. Gu, C. Feng, Z. Zhao, Z. Ying, M. Liu, R. T. Chen, and D. Z. Pan, "Squeezelight: Towards scalable optical neural networks with multi-operand ring resonators," in *IEEE/ACM Proceedings Design, Automation and Test in Europe (DATE)* (IEEE, 2021).

[24] C. Feng, J. Gu, H. Zhu, S. Ning, R. Tang, M. Hlaing, J. Midkiff, S. Jain, D. Pan, and R. Chen, "Integrated multi-operand optical neurons for scalable and hardware-efficient deep learning," Nanophotonics (published online, 2023).

[25] L. B. Soldano and E. C. M. Pennings, "Optical multi-mode interference devices based on self-imaging: Principles and applications," J. Lightwave Technol. **13**, 615–627 (1995).

[26] C. Demirkiran, F. Eris, G. Wang, J. Elmhurst, N. Moore, N. C. Harris, A. Basumallik, V. J. Reddi, A. Joshi, and D. Bunandar, "An electro-photonic system for accelerating deep neural networks," arXiv:2109.01126 [cs] (2022).

[27] Z. Zhao, D. Liu, M. Li *et al.*, "Hardware-software co-design of slimmed optical neural networks," in *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)* (IEEE, 2019).

[28]Y. Xiao, X. Peng, H. Tang, and Y. Tang, "Optical neural network with complementary decomposition to overcome the phase insensitive constrains," IEEE J. Sel. Top. Quantum Electron. **29**, 6100708 (2023).
[29]D. A. B. Miller, "Analyzing and generating multimode optical fields using self-configuring networks," Optica **7**, 794 (2020).
[30]S. Pai, Z. Sun, T. W. Hughes *et al.*, "Experimentally realized in situ backpropagation for deep learning in photonic neural networks," Science **380**, 398 (2023).
[31]J. Gu, Z. Zhao, C. Feng *et al.*, "Towards hardware-efficient optical neural networks: Beyond FFT architecture via joint learnability," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)* (IEEE, 2020).
[32]H. Zhu, J. Gu, H. Wang, R. Tang, Z. Zhang, C. Feng, S. Han, R. T. Chen, and D. Z. Pan, "Lightening-transformer: A dynamically-operated optically-interconnected photonic transformer accelerator," in *IEEE International Symposium on High Performance Computer Architecture (HPCA)* (IEEE, 2024).
[33]R. Baghdadi, M. Gould, S. Gupta, M. Tymchenko, D. Bunandar, C. Ramey, and N. C. Harris, "Dual slot-mode NOEM phase shifter," Opt. Express **29**, 19113 (2021).
[34]Y. Zhu, G. L. Zhang, B. Li *et al.*, "Countering variations and thermal effects for accurate optical neural networks," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)* (IEEE, 2020).
[35]M. V. DeBole, B. Taba, A. Amir, F. Akopyan, A. Andreopoulos, W. P. Risk, J. Kusnitz, C. Ortega Otero, T. K. Nayak, R. Appuswamy, P. J. Carlson, A. S. Cassidy, P. Datta, S. K. Esser, G. J. Garreau, K. L. Holland, S. Lekuch, M. Mastro, J. McKinstry, C. di Nolfo, B. Paulovicks, J. Sawada, K. Schleupen, B. G. Shaw, J. L. Klamo, M. D. Flickner, J. V. Arthur, and D. S. Modha, "TrueNorth: Accelerating from zero to 64 million neurons in 10 years," Computer **52**, 20–29 (2019).
[36]B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. A. Merolla, and K. Boahen, "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations," Proc. IEEE **102**, 699–716 (2014).
[37]J. Schemmel, D. Brüderle, A. Grübl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *2010 IEEE International Symposium on Circuits and Systems (ISCAS)* (IEEE, 2010), pp. 1947–1950.
[38]J. Choquette, O. Giroux, and D. Foley, "Volta: Performance and programmability," IEEE Micro **38**, 42–52 (2018).
[39]J. Choquette, W. Gandhi, O. Giroux, N. Stam, and R. Krashinsky, "NVIDIA A100 tensor core GPU: Performance and innovation," IEEE Micro **41**, 29–35 (2021).
[40]N. P. Jouppi, G. Kurian, S. Li, P. Ma, R. Nagarajan, L. Nai, N. Patil, S. Subramanian, A. Swing, B. Towles, C. Young, X. Zhou, Z. Zhou, and D. Patterson, "TPU v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings," arXiv:2304.01433 [cs.AR] (2023).
[41]I. Kacher, M. Portaz, H. Randrianarivo, and S. Peyronnet, "Graphcore C2 card performance for image-based deep learning application: A report," arXiv:2002.11670 [cs.CV] (2020).
[42]L. Gwennap, Groq rocks neural networks, microprocessor report, 2020, http://groq.com/wp-content/uploads/2020/04/Groq-RocksNNs-Linley-Group-MPR-2020Jan06.pdf.
[43]H. Larocque and D. Englund, "Universal linear optics by programmable multimode interference," Opt. Express **29**, 38257–38267 (2021).
[44]M. van Niekerk, J. A. Steidle, G. A. Howland, M. L. Fanto, N. Soures, F. T. Zohora, D. Kudithipudi, and S. F. Preble, "Approximating large scale arbitrary unitaries with integrated multimode interferometers," Proc. SPIE **10984**, 109840J (2019).
[45]T. Chen, Z. Dang, Z. Deng, Z. Ding, and Z. Zhang, "Micro light flow controller on a programmable waveguide engine," Micromachines **13**, 1990 (2022).
[46]M. Delaney, I. Zeimpekis, H. Du, X. Yan, M. Banakar, D. J. Thomson, D. W. Hewak, and O. L. Muskens, "Nonvolatile programmable silicon photonics using an ultralow-loss Sb_2Se_3 phase change material," Sci. Adv. **7**, eabg3500 (2021).
[47]C. Wang, M. Zhang, X. Chen, M. Bertrand, A. Shams-Ansari, S. Chandrasekhar, P. Winzer, and M. Lončar, "Integrated lithium niobate electro-optic modulators operating at CMOS-compatible voltages," Nature **562**, 101–104 (2018).
[48]Dataset: J. Gu, H. Zhu, C. Feng, R. T. Chen, and D. Z. Pan (2023). "M3icro-MOMMI open-source codes and data," Github. https://github.com/JeremieMelo/M3ICRO-MOMMI.
[49]Y. Zhang, A. Hosseini, X. Xu, D. Kwong, and R. T. Chen, "Ultralow-loss silicon waveguide crossing using Bloch modes in index-engineered cascaded multimode-interference couplers," Opt. Lett. **38**, 3608–3611 (2013).
[50]D. P. Nair and M. Ménard, "A compact low-loss broadband polarization independent silicon 50/50 splitter," IEEE Photonics J. **13**, 6600207 (2021).
[51]C.-H. Lin, Study of ultra-small NXN photonic multimode interference splitter and applications, 2007, http://rportal.lib.ntnu.edu.tw:8080/server/api/core/bitstreams/bf2cc632-4a39-49d5-9ac5-b93b5b6c774f/content.
[52]P. Dong, W. Qian, H. Liang, R. Shafiiha, D. Feng, G. Li, J. E. Cunningham, A. V. Krishnamoorthy, and M. Asghari, "Thermally tunable silicon racetrack resonators with ultralow tuning power," Opt. Express **18**, 20298–20304 (2010).
[53]S. Akiyama, T. Baba, M. Imai, T. Akagawa, M. Takahashi, N. Hirayama, H. Takahashi, Y. Noguchi, H. Okayama, T. Horikawa, and T. Usuki, "12.5-Gb/s operation with 0.29-V cm $V_\pi L$ using silicon Mach-Zehnder modulator based-on forward-biased pin diode," Opt. Express **20**, 2911–2923 (2012).
[54]Z. Huang, C. Li, D. Liang, K. Yu, C. Santori, M. Fiorentino, W. Sorin, S. Palermo, and R. G. Beausoleil, "25 Gbps low-voltage waveguide Si–Ge avalanche photodiode," Optica **3**, 793–798 (2016).
[55]H. Wang, R. Zhang, Q. Kan, D. Lu, W. Wang, and L. Zhao, "*High-power wide-bandwidth 1.55-μm directly modulated DFB lasers for free space optical communications*," in *2019 Conference on Lasers and Electro-Optics (CLEO)* (IEEE, Piscataway, NJ, 2019), pp. 1–2.
[56]H. Zhu, J. Gu, H. Wang, Z. Jiang, Z. Zhang, R. Tang, C. Feng, S. Han, R. T. Chen, and D. Z. Pan, "Lightening-Transformer: A dynamically-operated optically-interconnected photonic transformer accelerator," arXiv:2305.19533 (2023).
[57]A. Shafaei, Y. Wang, X. Lin, and M. Pedram, "FinCACTI: Architectural analysis and modeling of caches with deeply-scaled FinFET devices," in *2014 IEEE Computer Society Annual Symposium on VLSI* (IEEE, 2014), pp. 290–295.
[58]M. O'Connor, N. Chatterjee, D. Lee, J. Wilson, A. Agrawal, S. W. Keckler, and W. J. Dally, "Fine-grained DRAM: Energy-efficient DRAM for extreme bandwidth systems," in *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture* (IEEE, 2017), pp. 41–54.
[59]J. Liu, M. Hassanpourghadi, and M. S.-W. Chen, "A 10GS/s 8b 25fJ/c-s 2850um^2 two-step time-domain ADC using delay-tracking pipelined-SAR TDC with 500fs time step in 14nm CMOS technology," in *2022 IEEE International Solid-State Circuits Conference (ISSCC)* (IEEE, 2022), Vol. 65, pp. 160–162.
[60]P. Caragiulo, O. E. Mattia, A. Arbabian, and B. Murmann, "A compact 14 GS/s 8-bit switched-capacitor DAC in 16 nm FinFET CMOS," in *2020 IEEE Symposium on VLSI Circuits* (IEEE, 2020), pp. 1–2.
[61]Y. Kim, H. Kim, D. Ahn, and J.-J. Kim, "Input-splitting of large neural networks for power-efficient accelerator with resistive crossbar memory array," in *Proceedings of the International Symposium on Low Power Electronics and Design* (ACM, 2018), pp. 1–6.