# Multi-Level Encoding and Decoding in a Scalable Photonic Tensor Processor With a Photonic General Matrix Multiply (GeMM) Compiler

Zhimu Guo, Alexander N. Tait, *Member, IEEE*, Bicky A. Marquez, *Member, IEEE*, Matthew Filipovich, Hugh Morison, Paul R. Prucnal, *Life Fellow, IEEE*, Lukas Chrostowski, *Senior Member, IEEE*, Sudip Shekhar, *Senior Member, IEEE*, and Bhavin J. Shastri, *Senior Member, IEEE*

*(Invited Paper)*

***Abstract*—The resurgence of artificial intelligence enabled by deep learning and high performance computing has seen a dramatic increase of demand in the accuracy of deep learning model which has come at the cost of computational complexity. The fundamental operations in deep learning models are matrix multiplications, and large scale matrix operations and data-centric tasks have experienced bottlenecks from current digital electronic hardware in terms of performance and scalability. Recent research on photonic processors have found solutions to enable applications in machine learning, neuromorphic computing and high performance computing using basic photonic processing elements on integrated silicon photonic platform. However, efficient and scalable photonic computing requires an information encoding/decoding scheme. Here, we propose a multi-level encoding and decoding scheme, and experimentally demonstrate it with a wavelength-multiplexed silicon photonic processor. We also discuss the scalability of our proposed scheme by introducing a photonic general matrix multiply compiler, and consider the effects of speed, bit precision, and noise. Our proposed scheme could be adapted to a variety of photonic information processing architectures for photonic neural networks, photonics tensor cores, and programmable photonic.**

Manuscript received 22 February 2022; revised 14 June 2022; accepted 31 July 2022. Date of publication 5 August 2022; date of current version 23 August 2022. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), in part by the Canadian Foundation for Innovation (CFI), and in part by the Queen's University. *(Corresponding author: Zhimu Guo.)*

Zhimu Guo, Bicky A. Marquez, Matthew Filipovich, and Hugh Morison are with the Department of Physics, Engineering Physics and Astronomy, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: 15zg11@queensu.ca; bama@queensu.ca; matthew.filipovich@queensu.ca; hugh.morison@queensu.ca).

Alexander N. Tait is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: atait@ieee.org).

Paul R. Prucnal is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: prucnal@princeton.edu).

Lukas Chrostowski and Sudip Shekhar are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, British Columbia V6T 1Z4, Canada (e-mail: lukasc@ece.ubc.ca; sudip@ece.ubc.ca).

Bhavin J. Shastri was with the Department of Physics, Engineering Physics and Astronomy, Queen's University, Kingston, ON K7L 3N6, Canada.. He is now with the Vector Institute, Toronto, ON M5G 1M1, Canada (e-mail: shastri@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSTQE.2022.3196884.

Digital Object Identifier 10.1109/JSTQE.2022.3196884

***Index Terms*—Integrated optics, matrix decomposition, matrix multiplication, optical computing, optical neural networks, programmable circuits.**

## I. Introduction

ADVANCEMENTS in machine learning (ML) and artificial intelligence (AI) technologies have enabled numerous applications including sophisticated recommendation models, natural language processing, machine vision, augmented reality, and so on [1], [2], [3], [4]. The groundbreaking progress of these AI applications in different fields is enabled by heavy dependence of ML algorithms training on large data sets. Since the interconnection of neurons in artificial neural networks can be described by a matrix and the data being processed can be represented as a vector, training on large data sets with deep neural networks results in large-scale dense matrix-vector multiplications. The improvement in the performance (i.e. accuracy) of many ML applications comes at the cost of higher computational power requirement [5]. As such, there has been significant progress in the development of digital electronic application-specific integrated circuits known as AI accelerators that are dedicated for dense matrix computations [6], [7]. However, modern AI accelerators have seen two major bottlenecks when it comes to energy efficiency: data transfer to and from memory, and large matrix-vector multiplications, and both have imposed strict physical limitations on the scalability and performance of digital electronic AI accelerators.

Integrated photonic processors enabled by silicon photonics have shown promising capabilities in accelerating tensor (i.e., multidimensional vector and matrix) operations [8], [9], [10], [11] by exploiting the high bandwidth of photonic devices (modulators and photodetectors), low latency and minimal energy-delay product due to passive optical waveguides [12]. Some of these processors [9], [10], [11] are scalable and can use the parallel nature of light through wavelength-division multiplexing (WDM) to achieve large-scale interconnects and massively parallel data processing and transfer. Recent developments have proven that the wavelength-multiplexed silicon photonic platform can be operated with up to 7-bit precision [13], and most

recently 8.5-bit precision [14] on each individual multiplication unit. However, recent studies in these photonic processors have also seen an increasing demand for a rigorous photonic programming scheme to facilitate efficient communication between photonic hardware and its control system [8], [9], [12], [15]. A reliable information encoding and decoding scheme is required to interface between the silicon photonic platform and rest of the computing systems.

The core of a programmable system is a viable and efficient information encoding and decoding method that translates the same information between different hardware platforms and media using their own "languages" respectively. For example, in digital electronics, binary scheme is used as the information encoding and decoding method, where every channel in a binary system has one of the two digital states: either "1" or "0". However, the actual switching of the state at the transistor level is achieved through changing the voltage across the transistors. Therefore, the binary scheme maps "1" to a high voltage value and "0" to a low voltage value, and the binary scheme serves as the fundamental for all digital electronic platforms and hardware. Similarly, silicon photonic systems also require such an information encoding and decoding method that can conveniently translate information between digital user interface and analog compute hardware, using the specific physical parameters measured on different silicon photonic hardware platforms.

In this work, we present a feasible information encoding/decoding solution, the multi-level scheme, for WDM photonic processors based on microring resonator (MRR) [11], [16], [17]. Unlike the digital binary information system, the proposed multi-level scheme encodes multiple values as distinct amplitude levels using only a single analog input channel. Instead of using multiple channels to achieve a high bit precision, the multi-bit encoding method from multi-level encoding scheme will enable a higher bandwidth per input channel. By designing a dedicated information system for photonic tensor processors, we aim to take full advantage of photonics to create a fully packaged software/hardware photonic tensor processor solution that is capable of large matrix operations. As a dedicated photonic information system, the proposed multi-level encoding scheme can be generalized to different photonic tensor processors implementing an MRR-based architecture, and will also ensure a high compatibility with these photonic systems to achieve a high scalability on a hardware level. To demonstrate the scalability of our photonic tensor processor, we have implemented a simple general matrix multiplication (GeMM) compiler for the multi-level photonic information system as a software scaling solution. Computation results have verified the viability of this approach, and the computation accuracy is close to ideal for large matrices. A hardware scaling solution is presented, and we have shown an example of the actual implementation of this solution in a later iteration of our photonic tensor processor design.

## II. Multi-Level Information Encoding and Decoding

### A. Photonic Tensor Processing Element

The multi-level information encoding/decoding scheme is designed for the photonic tensor processing element (TPE) shown in Fig 1. This architecture was first proposed by Bangari et al. [11] to perform convolution operations and recently demonstrated by Marquez et al. [18] for vector dot products with limited precision. The photonic TPE includes an array of MRRs, each operating on a distinct resonant wavelength, encoding a row vector $\vec{W}$. Tunable lasers, that are intensity modulated (with variable optical attenuators (VOAs) in our case, or directly modulated laser (DML) diodes [19]), provide carrier signals for encoding the inputs $\vec{X}$ to the MRRs using different wavelengths. For a proof-of-concept demonstration, our TPE processes vectors of size $n$ resulting in $n$ lasers and $n$ coupled MRRs. As shown in Fig. 1(a), the MRRs are in an add/drop configuration and are coupled with two bus waveguides—a shared waveguide for IN-THRU connection, and another one connecting the DROP. While the input vector $\vec{X}$ is encoded via the attenuators as the intensities of the input optical power, the weight vector $\vec{W}$ is encoded as currents to the MRRs that shift their resonances, and redistribute the input optical between the DROP and the THRU ports according to the difference between the resonance of the MRR and the laser wavelength. In short, each input value is encoded onto a channel with a different wavelength, and we use multiple MRRs in parallel, each weighting a different channel. Both the inputs and weights are strictly encoded as the amplitude of the input optical power, as well as the output measurements. Thus, the phase information will be neglected on all optical channels, and we will not discuss any phase-change effects in our photonic TPE. As a proof-of-concept, we exploit the thermo-optic effect for the MRRs tuning [20], [21]. More efficient carrier-depletion effects or phase-change materials [22] can also be used to shift the resonant wavelength of the MRR with the current.

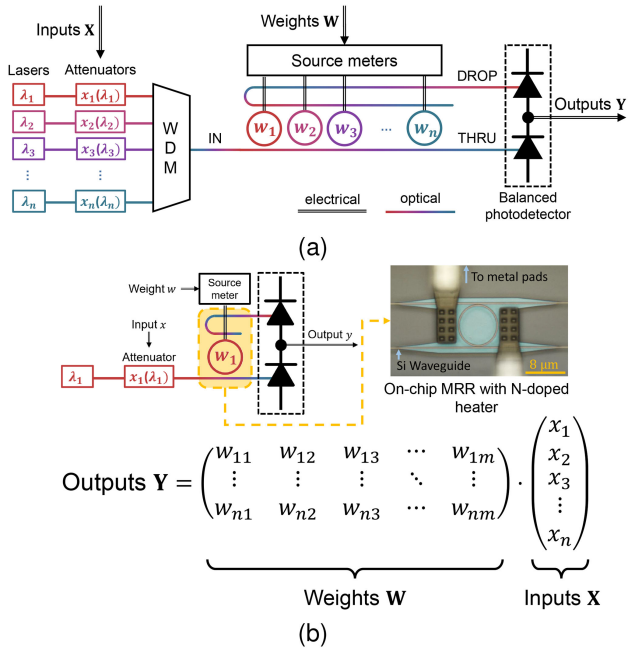



Fig. 1. (a): schematics of an MRR-based photonic TPE for vector dot product between vectors $\mathbb{X}$ and $\mathbb{W}$, along with its control system. (b): general mathematical concept for matrix-vector dot product using MRRs, and an optical micrograph of the fabricated silicon MRR with N-doped heater inside a photonic TPE on a silicon-on-insulator (SOI) platform.

Fig. 1(b) shows the silicon photonic TPE fabricated on a silicon-on-insulator (SOI) wafer with a silicon thickness of 220 nm and a buried oxide thickness of 2 $\mu$m. The bus waveguides have a width of 500 nm. The MRRs have radii of 8.0 $\mu$m, 8.01213 $\mu$m, 8.02426 $\mu$m, 8.03639 $\mu$m, 8.04852 $\mu$m. The gap between the ring and the bus waveguide is 200 nm, yielding a Q factor of $\sim$ 6000, and the free spectral range is around 12 nm for an MRR with 8 $\mu$m radius. The MRRs have N-doped photoconductive heaters [23] that can actuate the weight by thermally tuning the MRR resonance. To implement the N-doped heater, each MRR consists of a circular waveguide is etched to a 90 nm thick pedestal that hosts the phosphorous dopants. A 10 $\mu$m wide N doping section is patterned to follow the MRR, outside of which heavy N++ doping is used to make ohmic contacts. The phosphorous dopant concentrations are N: $5 \times 10^{17}$ cm$^{-3}$ and N++: $5 \times 10^{20}$ cm$^{-3}$. Metal vias and traces are deposited to connect the heater contacts of the MRR weight bank to electrical metal pads.

The TPE control system consists of source meters that provide the current to the MRRs, and a powermeter with a balanced photodetector, all controlled by a computer. The output optical power from both DROP and THRU ports are collected by the two photodetectors in a balanced push-pull configuration that subtracts the THRU port power from the DROP port power, giving us $P_{DROP} - P_{THRU}$ in units of dB [24]. All analog values are passed to and from the computer that regulates the information flow between a user application for ML and the photonic TPE.

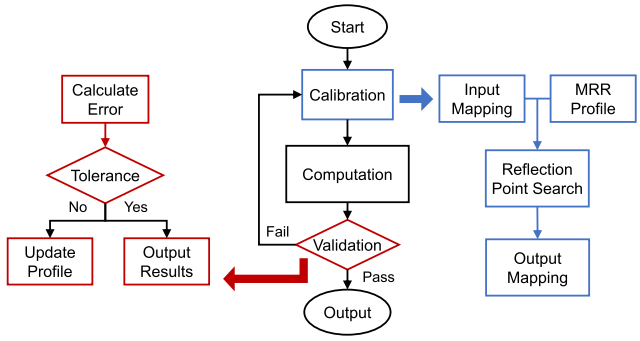



Fig. 2. Operation flowchart for a single MRR, including both calibration and validation stages.

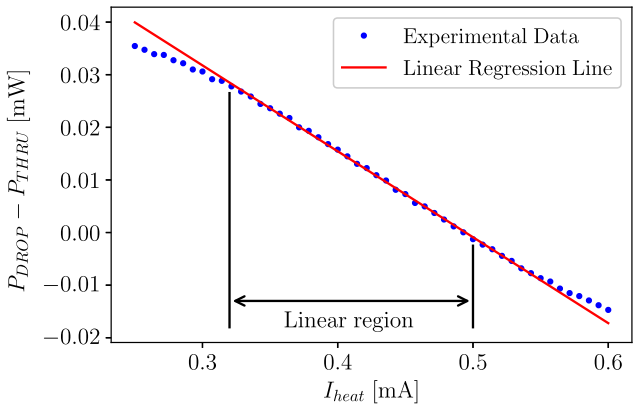



Fig. 3. Experimental data for the MRR profile mapping the measured output, $P_{DROP} - P_{THRU}$, to the applied heating current, $I_{heat}$.

### B. Input and Weight Encoding

The encoding scheme requires each photonic channel to represent numbers with $n$-bit precision using analog signals, and every analog value to be decoded back to its corresponding digital value. The photonic TPE has already shown promising results in its bit precision, and the highest possible precision achieved on a single photonic channel has been verified to be 7-bit [13], and more recently to be 8.5-bit [14]. Here, each photonic channel will include one MRR for multiplication, and one attenuator for input encoding.

The proposed multi-level encoding scheme implements a direct value mapping to translate an $n$-bit digital number to an analog value, and requires *calibration* and *validation* stages before and after the computation, respectively, as shown in Fig. 2. The calibration stage first starts with the inputs to the MRR, which are encoded as the amplitude of the input optical channel modulated by an attenuator. A direct *input mapping* is implemented to encode numerical input values onto the attenuation applied on the input optical channel. Next, the calibration performs a heating current sweep for one MRR at a time under a constant laser power, and compares the currents to the measured outputs, $P_{DROP} - P_{THRU}$, of the MRR. After collecting the heating current sweep data, we will choose a range of heating currents that produce a relatively linear response in optical output power as the MRR profile.

As shown in Fig. 3, the points in the middle of the heating current range produce a relatively linear trend. The relatively linear region is selected using the result of a linear regression of the heating current sweep data. We choose a specific tolerance of standard deviation that we aim to achieve, and manually adjust the heating current range for the linear regression until the standard deviation is under the specified tolerance. Having created the input mapping and the MRR profile, the next important parameter to define is the "zero point," or the "reflection point," of the MRR. The reflection point of the MRR represents the specific current value required to move the resonance of the MRR such that only half of the input optical power couples into the MRR and goes into DROP, while leaving the other half going into THRU. Thus, the linear power difference between DROP and THRU ports, $P_{DROP} - P_{THRU}$, is essentially a constant regardless of the input power. Therefore, we can perform a two-dimensional sweep on both the heating current and input power level to find the reflection point for the MRR, as shown in Fig. 4. The criterion for choosing the reflection point is the spread of power difference values at every current levels. The spread of power differences represents how far away the MRR is from the heating current level that gives the even distribution of power between DROP and THRU ports. A larger spread means the MRR is further away from that current level, and the less even power distribution will pronounce the change in input attenuation in larger magnitudes. On the contrary, the smallest spread means the MRR is almost indifferent to the change in input attenuation, and this only happens when the power distribution

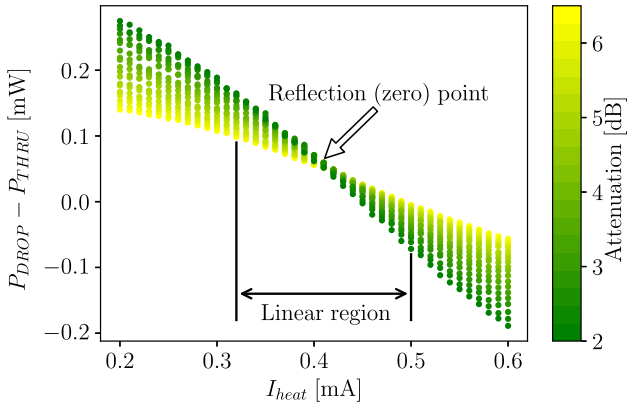



Fig. 4. Experimental data for the sweep that searches for the reflection point in the output transmission for the MRR. The powers from both DROP and THRU ports are measured at the output of the optical circuit, which is equivalent to the location before the signals come into the balanced photodetectors. The laser pump power is a constant 10 dBm, and with the input attenuation the laser pump power is low enough that it will not cause optical nonlinearities.

between THRU and DROP is close to even. Ideally, the constant power difference between DROP and THRU will be zero, but because of insertion losses between the waveguides and MRR, the measured reflection point yields a constant, non-zero power difference. However, for a practical MRR, the heating current levels that create the reflection point and the zero point are related to each other, the difference between these two points are determined by the coupling condition of DROP and THRU ports. Different coulping conditions will introduce different insertion losses on DROP and THRU ports, which breaks the even power distribution between the two ports in the ideal case.

Here we use electrical current instead of electrical power as the calibration metric during the search for the reflection point of an MRR. In theory, thermo-optic effect shifts the resonance of the MRR by applying a heating power to the MRR, and the resonance shift is linear with applied power. When the MRR is on-resonance with the input power, the input light will also induce a small photocurrent that affects the power reading. In addition, the resistance of the MRR will also increase as the temperature increases as a result of thermo-electric effect, consequently affecting the power measurement. On the other hand, if we focus on the small range of power output values around the reflection point, the output values can be approximated as a linear response. This allows us to use current values during the calibration phase with acceptable accuracy. Another benefit of using a tight range of current around the reflection point is that the small range prevents the use of larger currents and higher heat fluctuations created by large changes in the current from weight updates.

To further explain the reason for this non-zero "reflection point," we will take a closer look at an MRR under different coupling conditions: with critical coupling between THRU and DROP ports, and with symmetrical coupling between the two ports. The equations that describe different coulping conditions are originally demonstrated by Stokes et al. [25] and Heebner et al. [26], and are later re-derived by Bogaerts et al. [27]. Here, we assume the coupling coefficient between the MRR and THRU port is $r_1$, the coupling coefficient between the MRR and DROP is $r_2$, the loss in the MRR is $a$, and the detuning is $\phi$. The detuning can be calculated as following:

$$\phi = \frac{2\pi n_{eff}}{\lambda} \cdot 2\pi R = \frac{4\pi^2 R n_{eff}}{\lambda}. \tag{1}$$

Here, $n_{eff}$ is the effective refractive index of the MRR, $R$ is the radius of the MRR, and $\lambda$ is the wavelength of the input optical signal. Then we calculate the THRU port transmission, $T_{THRU}$, as following:

$$T_{THRU} = \frac{r_2^2 a^2 - 2r_1 r_2 a \cos\phi + r_1^2}{1 - 2r_1 r_2 a \cos\phi + (r_1 r_2 a)^2}, \tag{2}$$

and the DROP port transmission, $T_{DROP}$, as following:

$$T_{DROP} = \frac{(1-r_1)^2(1-r_2)^2 a}{1 - 2r_1 r_2 a \cos\phi + (r_1 r_2 a)^2}. \tag{3}$$

Finally, we can calculate the insertion loss, $IL$, for the MRR as following:

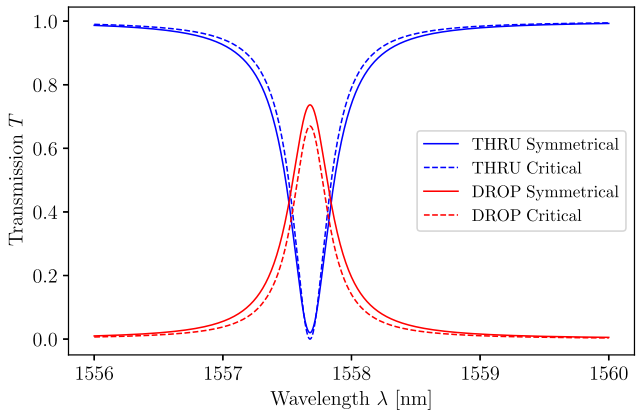
$$IL = 10 \log \frac{T_{THRU} + T_{DROP}}{1.0}. \tag{4}$$

The transmission curves plotted using (2)–(4) for both coupling conditions are shown in Fig. 5(a), as well as the insertion loss curves for both coupling conditions shown in Fig. 5(b). As is shown here, there is a non-zero insertion loss at resonance in both coupling conditions, meaning the magnitude of DROP port transmission will always be less than that of the THRU port transmission. As a result, the "reflection point," which is calculated as the difference between DROP and THRU port powers at half THRU port transmission, will be non-zero in a real-world MRR with losses regardless of which coupling condition. In addition, we choose symmetrical coupling condition for all MRRs in our photonic TPE because of fabrication variation. It is hard to hit an exact coupling value, $r$, because the as-fabricated gap strongly affects $r$. On the other hand, it is easy to make $r_1 = r_2$ by making it symmetric because the gaps usually come out the same. Most of the MRRs that have been fabricated for our weight banks are over coupled, meaning $(1 - a) \ll (1 - r)$. This is not optimal in terms of Q-factor, but it takes the loss, $a$, out of (2) and 3. Thus, we can end up with an expression that has good extinction ratio and also is robust to fabrication-sensitive parameters.

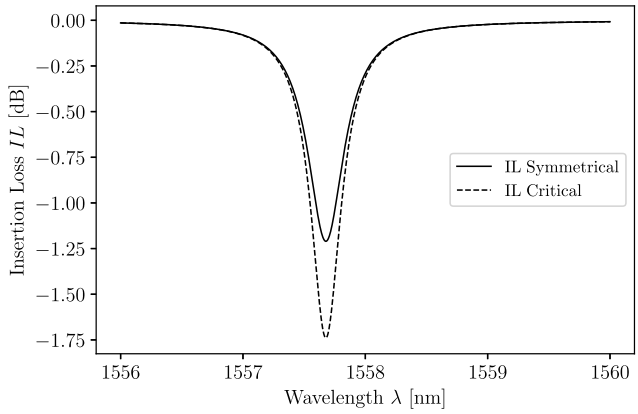
Now we can combine the "reflection point" location with the MRR profile to choose a proper heating current range and map that range to the other set of inputs that are encoded as heating currents to the MRR. The selected heating current range should center around the reflection point so that we can encode same range of positive and negative numbers. Since multiplication between two numbers can also be interpreted as one value being weighted by another, we call the mapping between the second set of inputs and heating current the *weight mapping*.

### C. Output Decoding and Calibration

Finally, the *output mapping* is created using both the weight mapping and the input mapping. This step generates random numbers for both the heating current on the MRR and the attenuator, and the product of the two is represented as the power difference between the DROP and THRU ports of the MRR as $P_{DROP} - P_{THRU}$. The measured outputs from the MRR




(a)




(b)

Fig. 5. (a) Transmission curves of the THRU (blue) and DROP (red) ports of a lossy MRR under either symmetrical coupling condition (solid lines) or critical coupling condition (dashed lines). For the MRR dimensions, we choose an MRR with a 8 $\mu$m radius and an effective refractive index of $n = 2.82$, the loss is $a = 0.99$ and the coupling coefficient between the MRR and the THRU port is $r_1 = 0.97$. For symmetrical coupling condition, we choose the coupling coefficient between the MRR and DROP port to be $r_2 = r_1$, whereas for critical coupling condition we have $r_2 a = r_1$. (b) Insertion loss (IL) curves (black) for both coupling conditions on the same plot, calculated using (4).

are mapped to the range of desired digital values after a linear regression, and the parameters of the regression are used as the output mapping to transform all measured optical output power to the numerical values, thus concluding the entire calibration stage for a single MRR.

The calibration stage is executed once at the start of the photonic system, and then the photonic TPE enters the computation-validation cycle. To demonstrate our proposed multi-level encoding and decoding scheme using a single channel, the calibration is performed using only one tunable laser and one MRR. For larger scale photonic TPEs, calibration will require switching on all the optical channels and only calibrate one channel at a time. This will take into account the constant optical power offset contributed from all the other channels at the output. The validation stage keeps track of the laser inputs, the MRR inputs, and the outputs, together with the three mapping profiles obtained from calibration stage. To reduce compute latency and operation complexity, validation on the output values will not be executed on every output. Here we can implement our control system to sample the outputs at a fixed frequency (i.e., every 5 minutes), and every sampled output is compared to the expected output calculated with the set of MRR and laser inputs, and the system will trigger a re-calibration if the measured output failed to match the expected output within tolerance.

The step that consumes the most time during a validation-recalibration process is the laser frequency sweep to redefine all the resonances of the MRRs due to resonance shifts over time. This is directly constrained by the tuning speed of the tunable laser (TLS). In our experiments the tuning speed of the TLS is 100 nm/s, and the typical free spectral range for our MRR designs with an 8 $\mu$m radius is around 20 nm. Therefore, it takes about 0.2 seconds to complete a frequency sweep to redefine all the MRR resonances. Other steps include the differential comparison between measured and expected values during the validation stage, and laser frequency resets after the resonance calibration. These steps take significantly less time when compared to the TLS frequency sweep. Therefore, we estimate that each validation-recalibration process will take around 0.2 seconds. In addition, our further system stability testing results showed that such validation-recalibration process would only be required hourly, making the time lost during this process insignificant compared to our system's actual uptime.

## III. Operational Rules

### A. Precision Flexibility

The multi-level encoding scheme only provides finite precision for number representation, and the total number of different values is also limited. On the other hand, the range of user requested values can vary depending on the specific application intended for the photonic TPE. However, because the proposed encoding and decoding scheme takes advantage of direct value mapping, the range of digital value that the analog signals are mapped to is arbitrary. In addition, the photonic TPE also supports multi-bit precision during operation, and the switch between different bit-precision only requires a system re-calibration. Therefore, the photonic TPE can be flexible with the value mapping and bit-precision during the encoding and decoding process. For example, if the software requires high computational accuracy but relatively small numerical ranges for input and output values, then the photonic TPE can use lower bit-precision for faster re-calibration, and fit the smaller numerical ranges with better computational accuracy. Here, computational accuracy is defined as the difference between measured and expected outputs. For lower bit-precision, each digital output value can have a larger analog output range, which can greatly reduce inaccuracy due to any kind of signal fluctuations or system instabilities. On the other hand, if the software requires larger numerical input and output ranges but has higher tolerance on accuracy, the photonic TPE can instead incorporate higher bit-precision encoding scheme to cover more values in the large numerical ranges. In this case, we can fit more digital values within the same overall analog output range at the expense of reducing the analog step size for individual digital output values.

The choice for higher or lower bit-precision is largely dependent on the actual application, thus the control system will require user specification to configure the bit-precision used in actual multiplication tasks.

### B. Commutative Property

To guarantee the stability of the system, the multi-level encoding scheme also imposes strict operational rules on the inputs for both the MRRs and the lasers. The input encoding for both sides uses the same direct value mapping between digital and analog values, but the underlying operating mechanisms are different. For the attenuators, different digital values are mapped to different optical power levels through different levels of attenuation, where small digital numbers corresponds to large attenuation, and vice versa. Since non-linearity will occur at high attenuation, we can only operate within a relatively small range of attenuation. As a result, the input optical power will never go to zero. For the MRRs, the digital values are mapped to the applied heating current values, which shift the resonance of the MRRs. The mismatch between the MRR resonance and the laser wavelength determines how the incoming optical power is distributed between DROP and THRU ports, but the total output power will equal the total input power in the ideal lossless case. Because loss is present in a real-world scenario, a higher laser power is more beneficial for a better performance of the photonic TPE. In addition, the heating current range chosen for the MRR will center around the "zero" point where the output power is evenly distributed between DROP and THRU. This means that the output power range is also centered around the zero point, and only spans a limited range on both sides of the zero point. Therefore, the input mapping can only encode numbers to a non-zero optical power range, whereas the weight mapping encodes numbers that centers around zero optical power. As a result, same numbers going through the attenuator will produce a different optical output than those going through the MRR, and the range of available optical outputs is different for the two. Therefore, multiplication of numbers from both sides does not commute, i.e. $a \times b$ does not equal $b \times a$. To circumvent this problem, the multi-level encoding scheme will force the larger number through the lasers when multiplying two numbers with the photonic TPE since higher input power for the MRR will give better output resolution.

### C. Negative Number Encoding

Aside from the non-commutative operation rule mentioned above, we also implement another restriction on the sign of the multiplication. Since only the MRR can encode both positive and negative numbers using left and right of the "zero" point in its output power but the attenuator can only encode positive numbers, any negative number we encounter will be sent to the MRR automatically. In case of two negative numbers during multiplication, both negative signs will be dropped automatically since that is equivalent to two positive number multiplication.

An alternative solution to encode negative numbers in our photonic TPE is to have another photonic TPE with the exact same configuration running in parallel. This will allow us to separate negative and positive multiplication completely within our control system for the TPEs, and dedicate one photonic TPE to process either all positive/negative multiplications, or mixed positive/negative multiplications. For either TPE, negative signs will be dropped everywhere during multiplication, and the control system will take outputs from the one processing mixed positive/negative multiplications as negative values automatically.

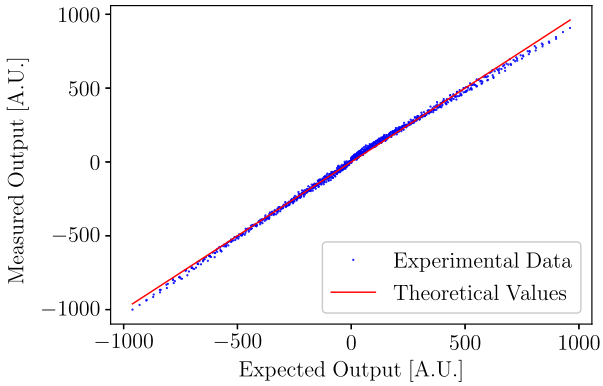



Fig. 6. Output mapping for a 11-bit signed system with 6-bit signed inputs.

## IV. Experimental Demonstration

Here, we implement a 11-bit signed system with 6-bit signed inputs for our proof-of-concept demonstration. First we perform the calibration stage as mentioned above, including creating an input mapping, an MRR profile, and performing a reflection point search. The input mapping uses an attenuation range between 2 dB and 8 dB for mapping $2^5$ positive input digital numbers to their corresponding, linearly spaced, analog optical power levels. From the reflection point search we determine that a heating current of 0.48 mA to the MRR would produce a zero output power calculated from $P_{DROP} - P_{THRU}$. Combining this with the MRR profile which gives us the heating current range that produces a linear output power level, the weight mapping is finished with a heating current range between $[0.37, 0.59]$ mA that fits $2^6$ signed digital numbers.

Next, the output mapping is constructed through sweeping both inputs and weights across all possible values using both the input mapping and the weight mapping. All possible input/weight combinations include $2^5 \times 2^6 = 2048$ pairs, but only a subset of combinations that meet the aforementioned commutative property is selected. The input number range is chosen to be $[0, 31]$, and the weight number range is $[-31, 31]$. The choice of values inside the matrices is based on the selected precision for the system, which is a 6-bit signed integer system as an example. This range is only a digital representation of the measured analog values, and the example demonstrates how the matrix dot product will work based on an arbitrary value range selection. However, this value range selection can be any numerical range that centers around zero depending on the application, and in many situations, the common choice will be the normalized range of $[-1, 1]$. We collect the experimental results as shown in Fig. 6. Here, the expected output is calculated by multiplying the input number with weight number directly

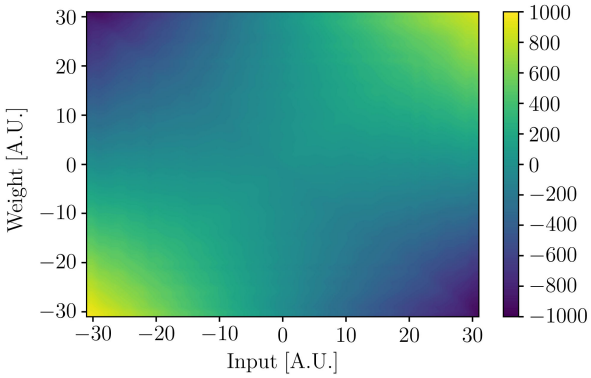



Fig. 7. Multiplication results with full range of 6-bit signed inputs and weights with the implementation of above mentioned operational rules. Here the inputs range between $-31$ and $+31$, and the weights also have the same range. Different colors in the colorbar represents the product of a weight value and an input value, with purple representing the smallest and the yellow representing the largest.

inside the control computer. The measured output is converted from the measured optical output power, $P_{DROP} - P_{THRU}$, to the desired output number range via a re-scaling. The re-scaling of experimental data first performs a linear regression using both the measured output power and the expected output values, and then it compares the slope and intercept of experimental data to a theoretical slope of 1.0 and intercept of 0.0. After the re-scaling, the measured experimental data is converted to measured output values that were in the same range as the expected outputs.

Having fully characterized the photonic TPE that includes an MRR and an attenuator, we now incorporate the sign rule and include full positive and negative numbers for both the input and weight. The result of full 6-bit signed multiplication is shown in Fig. 7. Here, both input and weight go from $[-31, 31]$, and the experimentally measured output is shown as colored contour maps on the two-dimensional grid of weight versus input. The measured output values range between $(-1000, 1000)$, and the standard deviation calculated from the measured outputs is $9.34 \times 10^{-6}$.

The precision adjustment can be easily made at the generation of the direct mapping stage during calibration. The calibration starts with weight and reflection sweeps, which will estimate the usable heating current range for the MRRs. The next step is to decide how many analog levels we need to represent all digital values up to the chosen precision. We demonstrated an 11-bit signed system with half-precision for weights and inputs. However, the system can be easily adapted to lower precision levels, such as 8-bit signed precision with half-precision for the weights and inputs. In this case, we only need to redo the direct mapping for the weights and inputs to accommodate fewer analog levels.

## V. GeMM Compiler and Scalability

Having demonstrated the functionality and performance of a single photonic TPE, we will focus on scaling up our system to accommodate higher computing capacity and throughput. Scaling up a processing element architecture generally involves two approaches: hardware scaling and software scaling.

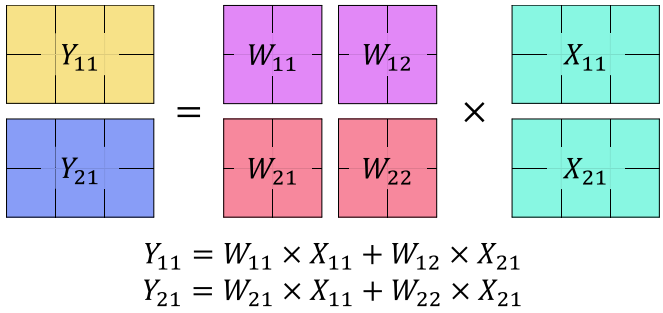



Fig. 8. Matrix dot product using a GeMM compiler.

### A. GeMM Compiler for Photonic TPE

First, we demonstrate our solution for software scaling. One common software up-scaling approach for tensor processors is the use of a General Matrix Multiplication (GeMM) compiler [28], which helps a software to map different matrix operations to specific hardware architectures to optimize structure utilization and computation efficiency. As the modern data science industry continues to develop and the computation volume and complexity increases, most data-focused compute hardware finds it helpful to implement a dedicated compiler to efficiently perform sophisticated matrix multiplications. There are many different designs for GeMM compilers depending on their targeted hardware platforms [29], [30], [31], but the basic operating rule for any GeMM compiler focuses on the most prevalent matrix multiplication, matrix dot product, and its mathematical form can be expressed as (5),

$$\mathbf{Y} = \alpha \mathbf{W} \cdot \mathbf{X} + \beta \mathbf{Z}. \tag{5}$$

Here, $\mathbf{W}$, $\mathbf{X}$ and $\mathbf{Z}$ are input matrices, both $\alpha$ and $\beta$ are scaling constants, and $\mathbf{Y}$ is the output matrix. The math is simple, but the main focus of GeMM compilers is mapping the mathematical expression to the topology of different hardware platforms. Because the sizes of the matrices from data-focused tasks often exceed the physical sizes of the actual compute hardware, GeMM compilers need to first break down these large matrices into smaller matrices or vectors. How the matrices are broken down depends on the core/thread count of the actual hardware, and the overall task of matrix multiplication will be done in multiple batches. Once the matrices are divided, GeMM compilers need to send specific values from the current data batch to the compute units used by the task. After one iteration of computation is finished, GeMM compilers will then collect all the results and send out the next batch. As an example, we have a simple matrix dot product between two matrices $\mathbf{W}$ and $\mathbf{X}$ as shown in Fig. 8.

We divided matrix $\mathbf{W}$ into four batches each containing four elements, and matrix $\mathbf{X}$ into two batches each containing six elements. The number of compute units used for this task will be six, matching the number of elements in the largest data batch. The GeMM compiler will first send out the first data batches from both matrix $\mathbf{W}$, $W_{11}$, and matrix $\mathbf{X}$, $X_{11}$ to all the compute units to calculate the dot product $W_{11} \times X_{11}$. For the second round of operation, the GeMM compiler will send out $W_{12}$ and $X_{21}$ instead, and the same procedure is repeated for

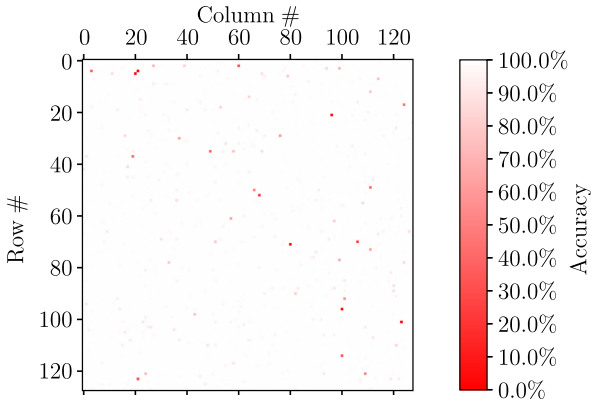



Fig. 9. Accuracy of a matrix dot product computation between two matrices both of size $128 \times 128$. Here, the color grid on the left shows the computation accuracy of each element in the output matrix, where white means 100% accurate and red means the output accuracy is zero.

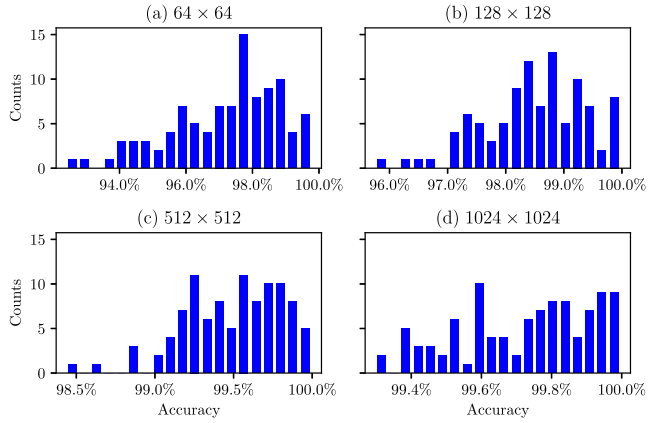



Fig. 10. Distribution of the matrix dot product accuracy collected from trials with different matrix sizes, each including 100 computations using randomly generated matrices. Here, we simulated four different matrix sizes, including (a) $64 \times 64$, (b) $128 \times 128$, (c) $512 \times 512$, and (d) $1024 \times 1024$. We used 6-bit signed inputs and weights for all the computations.

the third and fourth iterations. Once all input data batches are cycled through all compute units and the results are collected, the GeMM compiler then send out the results in two batches to calculate the elements in the output matrix $\mathbf{Y}$. Having calculated all the elements, the GeMM compiler then reconstructs matrix $\mathbf{Y}$ with all the results and sends it back to the user.

Because of this divide and conquer technique, GeMM compilers have enabled many modern tensor processors to achieve a compute capability far beyond their physical topology limit with a high efficiency. Examples of this include NVIDIA's Tensor Core [6] and Google's TPU [7]. Therefore, our software scaling solution can take full advantage of GeMM compiler's promises and enable a high volume computation on a small but efficient physical architecture.

To demonstrate this idea, we implement a simple GeMM compiler that can break down large matrices and schedule computation tasks among different MRRs of our photonic tensor processor. First, we perform a matrix dot product between two matrices each of size $128 \times 128$ using a photonic TPE consisting of five MRRs, and all matrix elements are randomly generated and are encoded using 6-bit signed precision. Since the input matrices are larger than the size of the photonic TPE, the matrices are broken down into many vectors each containing five elements during the many iterations of computation. The output matrix is shown in Fig. 9, where each pixel on the matrix plot represents the computational accuracy of each output matrix element from this single trial, presented in different colors shown on the scale on the right. Here the accuracy of each element in the output matrix is calculated as following:

$$Accuracy = 1 - \frac{Measured - Target}{Measured}. \tag{6}$$

As shown here, this single trial achieved high accuracy for the majority of the matrix elements, with only a few exceptions represented as the red pixels. The average accuracy across all matrix elements from this trial is 99.71%, with a standard deviation of 0.0285. However, we notice a few elements in the output matrix that have an accuracy value of zero. This is likely due to the fact that the expected output for those elements are digital zeros, but our photonic TPE measures non-zero analog values at those points. Such behavior indicates that there is noticeable noise in our photonic circuits, and optimizing our control system to account for such noise will require further work.

### B. System Scalability

Having validated the functionality of the GeMM compiler, we verify the performance of our photonic tensor processor using the multi-level encoding scheme, together with the GeMM compiler. In the first test we vary the sizes of input matrices from $64 \times 64$ to $1024 \times 1024$ while maintaining the same bit-precision for all the matrix element as signed 6-bit. We run the matrix dot product computation with randomly generated elements for each matrix size 100 times, and collect the average accuracy of each output matrix into a histogram as shown in Fig. 10. As shown here, the spread of average computation accuracy tightens as the matrix size increases, indicating that the multi-level encoding scheme and our photonic tensor processor see a performance improvement over larger matrices.

A second test is designed to investigate the performance change as a result of changing the bit-precision of matrix elements. We start the test using only 3-bit signed values for all matrix elements, and increase to the original 6-bit signed precision. The sizes of matrices used in this test are the same $128 \times 128$ for all 100 randomly generated computations, and the results from the second test are shown in Fig. 11. Here, trials using only 3-bit signed precision generate the largest spread of average accuracy, whereas 4-bit signed precision and above produce comparable results. The results show that multi-level encoding scheme exhibits the best performance when using higher bit-precision, but the accuracy slightly decreases for lower bit-precision.

After performing the two performance tests using different parameters, we condense the histograms shown in Fig. 10 and Fig. 11 by calculating the overall average accuracy and its standard deviation for every 100 trials with a specific parameter as shown in Fig. 12. Here, Fig. 12(a) shows the overall accuracy

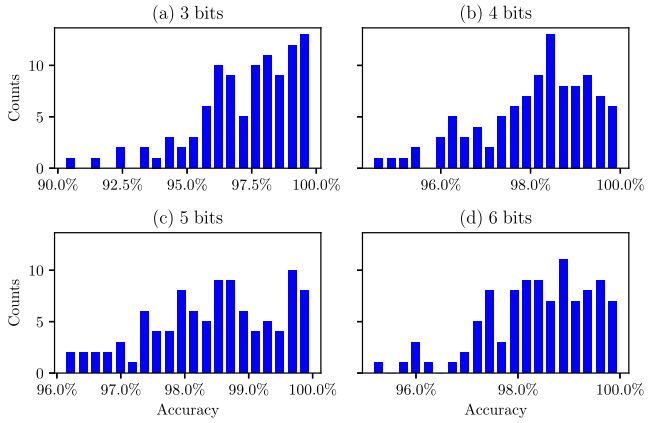



Fig. 11. Distribution of the matrix dot product accuracy collected from trials with different bit-precisions, each including 100 computations using randomly generated matrices. Here, we simulate four different bit-precisions, including (a) 3 bits, (b) 4 bits, (c) 5 bits, and (d) 6 bits to encode input values. We used a matrix size of $128 \times 128$ for all computations.

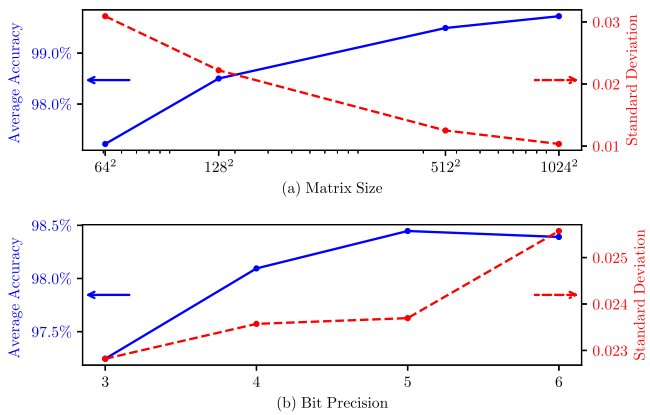



Fig. 12. (a) Average accuracy and its standard deviation calculated from the trials with different matrix sizes, as shown in Fig. 10. (b) Average accuracy and its standard deviation calculated from the trials with different bit precisions, as shown in Fig. 11.

from the first test using four different matrix sizes, from $64 \times 64$ to $1024 \times 1024$. We see a clear upward trend in the overall average accuracy as aforementioned, together with a decreasing trend for the standard deviation from the overall average accuracy calculation. For the second test, the overall average accuracy also increases with higher bit-precision, but we also see a small increase in standard deviation from the calculation as shown in Fig. 12(b). Because the change in standard deviation in Fig. 12(b) is one magnitude smaller than that in Fig. 12(a) and the average accuracy is similar for trials using more than 3-bit precision, the small increase in standard deviation can be a random result since all matrices are randomly generated in all trials. The improved average accuracy in both performance tests is likely a result of larger sample size when running randomized trials. Randomized trials require greater sample sizes to better achieve the ideal normal distribution of test samples, and as the matrix sizes increase these test matrices include more randomized values which contributes to a better test sample distribution. As test sample distribution approaches the ideal normal distribution, the accuracy results obtained from such random trials can faithfully represent the true accuracy achievable on our photonic TPE. Therefore, an improved accuracy results over larger test matrix sizes indicates that our system can achieve above 99.5% accuracy.

The standard deviation in Fig. 12(b) represents the accuracy value fluctuation over multiple repeated trials—the increase in standard deviation results from the noise within our analog system. At higher precision levels, the same level of analog noise will be more likely to cause a misrepresentation of each digital value. Lower precision only requires fewer analog levels to include all the digital values, whereas higher precision will require more analog levels within the same analog range. As a result, the system is more susceptible to noise at higher precision. We have observed a larger fluctuation in average accuracy values over multiple tests, leading to a slight increase in the standard deviation value.

Aside from software scaling, hardware scaling is also a crucial part in boosting the computation capacity of our photonic tensor processor. The hardware architecture for a single photonic TPE is shown in Fig. 1, which contains an array of five MRRs sharing both a common THRU connection and a common DROP connection. The photonic TPE is capable of performing five multiplications simultaneously using five sets of inputs through the same bus waveguide, each set encodes one number through the attenuator as the "input" and the other through the source meters as the "weight". Thus, the single photonic TPE can compute a dot product between two vectors each with five elements within a single iteration. However, this is only one single photonic TPE, and its architecture can be easily duplicated on chip. In addition, because different copies of the same photonic TPE have their own bus waveguide for inputs, the same laser sources can be used in a multiplexer/splitter fashion to provide the same copies of all signal carriers for all the photonic TPEs. The multiplexer is implemented using a WDM multiplexer that combines all the individual laser sources from separate waveguides, and the splitter evenly distributes the combined signal among all photonic TPEs. On the other hand, most hardware scaling solution will benefit from a higher level of integration for lower latency and higher compute throughput. In our current design for the photonic TPE, the input mapping still relies on external attenuators to encode different input values as different optical power levels. However, same effect can be achieved by using the THRU port output of an on-chip MRR. By tuning the MRR on and off resonance, the THRU port output will carry different output power depending on the wavelength mismatch between the optical signal and the MRR. Therefore, by replacing the attenuators with on-chip MRRs for input encoding, the control mechanism can be applied to both the input encoding MRRs and the multiplication MRRs. Additionally, the balanced photodetectors can also be integrated on chip, and will only require a bias voltage from the external source meters. The output of the balanced photodetectors is in the form of different current levels, which can also be monitored through the same sourcing and measurement units. Thus, both information encoding and decoding will be uniformly implemented through the external source meters for both inputs and weights.

To illustrate this hardware scaling idea, we consider the following design for a scaled-up version of the photonic tensor processor containing four TPEs, as shown in Fig. 13. Each of the photonic TPE contains four MRRs with only THRU ports for input encoding, and another four MRRs with both DROP and THRU connections for multiplication. Four laser sources are used to provide carrier signals for each of the input and multiplication MRR in every TPE. Since each TPE has its own separate bus waveguide, only four lasers will be required to drive all four TPEs. The summation of all photonic channels within each TPE is done by the integrated balanced photodetectors on chip. Both information encoding and decoding will rely on source meters to provide either a heating current or a bias voltage, and to measure the photo current output from the balanced photodetectors.

The large matrix decomposition and dot product are done in simulation using the experimental data collected from the single MRR experiment. However, we have also performed testing using multiple MRRs and tried to qualitatively observe the effects of thermal crosstalk between neighboring MRRs. Our preliminary results showed that the magnitude of thermal crosstalk mainly depends on the MRR spacing. With a spacing of around 150 um, the crosstalk effect becomes insignificant relative to other noise sources within our system. The most effective way to minimize thermal crosstalk in our system is to create larger spacing between MRRs; however, this will reduce the compute density of our device. The other solution will require extensive calibration to be performed simultaneously across all active MRRs and sophisticated monitoring procedures during operation. As a result, the reduced compute speed due to the added calibration and monitor steps will also hamper the compute density.

On the other hand, the best solution to thermal crosstalk would be to eliminate thermal tuning and implement the carrier-depletion effect. The carrier-depletion effect not only offers a high tuning speed that can enable fast weight updates but also generates significantly less heat, allowing for a more compact photonic TPE design to achieve higher compute density.

### C. Estimated Energy Consumption

Our photonic TPE design implements a multi-wavelength approach that uses multiple MRRs. As a demonstration we showed the performance and the multi-level encoding/decoding scheme for a single channel TPE, but for multi-channel TPE designs, each MRR will strictly operate on a separate wavelength. First we only consider the tuning power during the operation, which will be the major energy consumption during a photonic MAC operation. For a small photonic TPE, we assume it has a size of $n \times n$, giving us a total of $n^2$ MRRs and $n$ balanced photodetectors for photonic MAC operations. The total number of MACs per cycle for this small photonic TPE will be $n^2$ MACs, and the total energy consumption per cycle will be $n^2 \cdot E_{MAC}$ if we assume each MAC consumes energy equals $E_{MAC}$.Now we consider a scaled up photonic TPE that is $x$ times larger in both dimensions compared to the small photonic TPE, giving us a size of $xn \times xn = x^2 \cdot n^2$. Under the same assumption

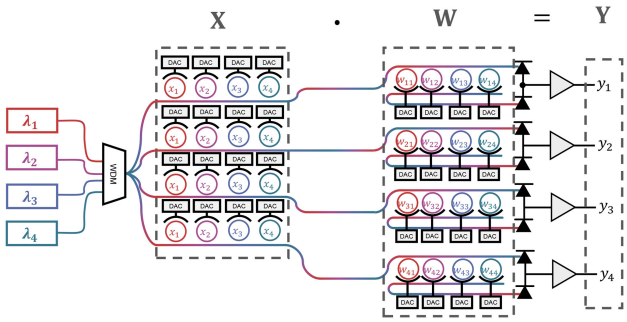



Fig. 13. Scaled-up design of a photonic tensor processor containing four photonic TPEs, each capable of performing a vector dot product with four elements simultaneously. All photonic TPEs in this design integrate input mapping on-chip through the input encoding MRRs, and also retain the same MRR weight bank design as shown before.

for energy consumption, we arrive at the total energy per cycle for the larger photonic TPE as $x^2 \cdot (n^2 E_{MAC})$. If we want to compute a matrix multiplication between two matrices, both of size $m \times l$ with $m \geq xn$ and $l \geq xn$, then it will take $ml$ cycles to complete the operation on the small TPE, whereas it only takes $\frac{ml}{x^2}$ cycles on the scaled-up TPE. If we compute the total energy consumed by tuning the MRRs during the operation, the total energy will be identical since the total workload remains the same.

Next, we will consider the I/O energy consumption in our system. For the larger scale system concept shown in Fig. 13, we chose to use one set of input encoding MRRs for each TPE. As a result, the actual tuning power on the input encoding process will scale linearly with the input matrix size, which is on the same order of magnitude scaling as compared to the photonic MAC. However, we only implement one set of balanced photodetector for accumulating all the computed results. Therefore, the energy consumption scaling for the output optoelectrical conversion will be sublinear compared to the input and weight matrix sizes. If we break down the input and output energy consumption for a single photonic MAC, the input energy consumption per MAC will not decrease for larger photonic TPEs, but the output energy consumption per MAC will decrease significantly. Recent investigation by Al-Qadasi et al. [32] has also quantified this estimation, where for a thermally tuned MRR-based photonic TPE, the energy per MAC was calculated to be around 1.2 pJ/OP for a network size of 15 MRRs. The energy per MAC will decrease to around 1 pJ/OP for a larger network size of 85 MRRs. However, if we can improve the thermal tuning design by adding insulators to the heaters inside the MRRs, the energy per MAC can drop significantly down to around 0.3 pJ/OP for the smaller network size of 15 MRRs, and down to less than 0.1 pJ/OP for the larger network size of 85 MRRs.

Therefore, the total energy consumption will be the slightly less for the large TPE to complete the same workload as the output optoelectrical conversion will be more efficient. However, the small improvement in energy per MAC is outmatched by improved designs of the heaters, and as a result the overall energy consumption for both the small and the large systems are similar. The main difference between operating the small

and the large TPEs comes from the number of cycles required and how the energy consumption is distributed over time. Here, the larger photonic TPE will require more energy within a short time window, whereas the small TPE uses much less energy per cycle and spread out the energy consumption over longer periods of time.

## VI. Speed and Precision Analysis

The MRRs shown in Fig. 1 implemented thermal tuning using N-doped heaters, and our photodetectors were SiGe-based PIN junction photodetectors. The tuning speed of N-doped heaters is up to millisecond scale, which can be a limitation in certain scenarios and applications. However, in many other deep learning applications the update rate of the weights can be much slower, especially during inference or convolutions. During inference, the photonic TPE will be loaded with pre-trained weights. Thus, the photonic TPE can perform MAC operations at the speed limit of the photodetectors, which is shown to be 56 GHz for an avalanche photodetector [33] and 67 GHz for a PIN photodetector [34]. In case of convolutions as demonstrated by Feldman et al. [22], the convolution filters only require a slower update rate compared to the inputs. Therefore, the relatively slow tuning speed of the weights inside the photonic TPE can satisfy a high-speed MAC operation for inference or convolution.

However, in the case of matrix tiling the photonic TPE computation speed will also be limited by the weight updates. Because we are using a wavelength-multiplexed approach, the speed bottleneck for our system during a matrix tiling process will be affected by three factors: weight update speed, detection speed, and the physical size of our photonic TPE. The weight update speed determines how fast the photonic TPE can be updated for the next batch of weights. The detection speed determines how fast the TPE can process all batches of inputs before adjusting the weights. The size of the photonic TPE will affect the number of input batches per weight batch. In a wavelength-multiplexed setup where a single balanced photodetector is paired with multiple MRRs, we can increase the number of MRRs as long as their resonances can all fit within their free spectral ranges. With more MRRs in the weight bank, large matrices require less tiling to finish the computation. Also, we can implement data batching and the weights inside the photonic TPE will not be updated until all the inputs have been processed through the TPE. Therefore, larger photonic TPEs will go through all the inputs using fewer cycles and require more often weight updates when compared to smaller ones. As a result, smaller TPEs rely more on the speed of the photodetector to process all the inputs, but larger TPEs rely more on the weight update speed of the MRRs once all the inputs are processed.

Given these three factors that bottleneck the speed of our photonic TPE, there will be an optimal photonic TPE size that balances the latency between weight updates and photodetection. Currently, we are using the thermo-optic effect in our system, which operates on a millisecond time scale, and the photodetectors in our system have been verified to achieve 10 GHz. Assuming thermal tuning, the MRRs take 1 ms, then the optimal size for our photonic TPE can have no more than a single MRR per pair of balanced photodetectors for a matrix with a size of $1024 \times 1024$. In this scenario, the speed of photonic TPE is bottlenecked by the weight update speed because the photodetector is $10^7$ times faster than the thermally tuned MRRs. However, suppose we were to use the carrier-depletion effect to modulate the optimal photonic weights at up to 56 GHz and use a fast PIN photodetector that operates at 67 GHz. In that case, TPE size will consist of around 850 MRRs. In conclusion, thermal tuning the MRR will create a speed bottleneck from weight updates during matrix tiling, but narrowing the speed gap between weight modulation and photodetection will require larger photonic TPE sizes to take full advantage of that fast weight update capability.

Recent analysis on signal resolution in silicon photonic neural network by Tait [35] summarizes the relation between laser pump power, signal frequency, and bit precision. In the middle of all three terms are different dominating types of noise in different operating regimes of the silicon photonic system. Our silicon photonic system implements an O/E/O operating regime, where the first part is the optical signal from a tunable laser, and then optical weighting uses MRR weight banks with thermal tuning. After the weight bank is the optoelectrical conversion by the balanced photodetector. For such a photonic circuit, there are three major noise regimes that affect the interaction between laser pump power, signal frequency, and bit precision: thermal regime, shot regime, and relative intensity noise (RIN) regime. In the thermal regime, the dominant noise is known as Johnson-Nyquist noise which comes from the random movement of electrons within the photodetector. Here, the noise equivalent power increases exponentially with higher bit precision, and the relation between laser pump power $P_{l_{therm}}$, signal frequency $f$, and bit precision in thermal regime $B$ can be written as:

$$P_{l_{therm}}(f, B) = \sqrt{f} \cdot \frac{J^*(B)}{\eta_{net}}, \quad J^*(B) \propto 2^{\frac{3}{2}B}. \tag{7}$$

Here, $\eta_{net}$ is the transmission efficiency of our photonic circuit, and $J^*$ represents the Johnson-Nyquist noise at the given precision $B$. During the operation of the MRR weight bank inside our photonic TPE, the input laser pump power will remain a constant value. As is shown here, there is a trade-off between signal frequency and the bit precision of our system at a given laser pump power level. Thus, higher frequency operations will require lower bit precision to maintain system stability.

During the optoelectrical conversion, photon shot noise will be the dominant noise and this is called shot noise regime. Shot noise comes from the randomness in photon detection, and in this regime we still have the same relation between laser pump power $P_{l_{shot}}$, signal frequency $f$, and bit precision $B$ as is shown in the thermal regime. Here we have:

$$P_{l_{shot}}(f, B) = f \cdot \frac{E_{shot}(B)}{\eta_{net}}, \quad E_{shot}(B) \propto 2^{3B}. \tag{8}$$

As shown here, the same trade-off between signal frequency and bit precision still remains. In addition to thermal and shot noise regimes, the carrier laser power output also has random changes that create relative intensity noise. In RIN regime, the noise is independent of laser pump power, but the frequency-precision relation gives us the maximum signal frequency that

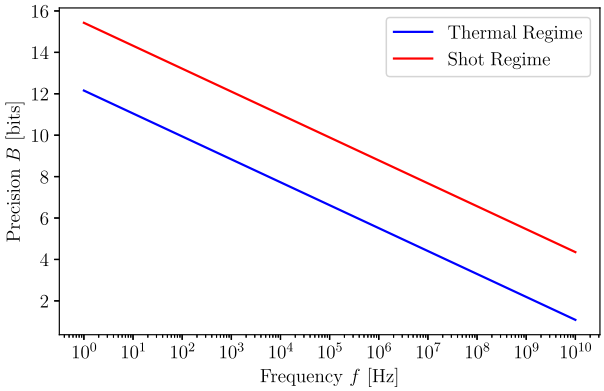



Fig. 14. The trade-off between operating frequency and precision under a constant laser pump power of 10 mW, and a round trip loss of 15 dB. The transmission efficiency was taken to be 0.1, and we consider both thermal regime and shot noise regime. For the thermal regime, we consider the situation where we are using less than the full designed bandwidth for our single channel. For the shot noise regime, we consider the shot noise amplitude in a typical analog photonic system, which will be larger than the noise amplitude in the an ideal system nearing its physical limit.

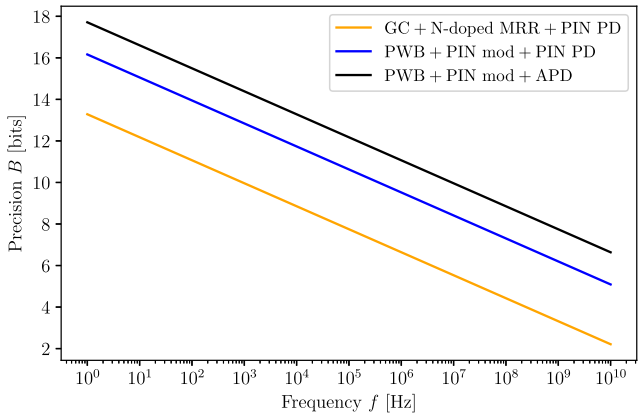



Fig. 15. The frequency-precision trade-off comparison between multiple photonic systems with different components. In this paper, the photonic TPE design implemented grating couplers (GC), MRRs with N-doped heaters (N-doped MRR), and PIN junction photodetectors (PIN PD). However, recent work has shown that we can replace these components with ones that have higher efficiency and speed, like photonic wirebonds (PWB), PIN junction modulators (PIN mod), and avalanche photodetectors (APD). This plot gave an estimation on how the frequency-precision trade-off will look like compared to our current design, and we will be implementing these more advanced components in our future designs.

can be obtained at a certain bit precision:

$$f \leq F_{RIN}(B), \quad F_{RIN}(B) \propto 2^{-3B}. \tag{9}$$

Therefore, the signal frequency has a device-specific upper limit at any given bit precision, and the upper limit is independent of laser pump power.

To demonstrate the aforementioned trade-off between operating frequency and precision under a constant laser pump power of 10 mW and with a 15 dB round trip loss, we plot the operating frequency and the expected precision in both thermal and shot noise regimes as is shown in Fig. 14. Here, we chose a transmission efficiency of 0.1 for our analog photonic circuit, and we included a wide range of frequency values ranging from 1 Hz to 10 GHz. As shown in Fig 14, the trade-off between operating frequency and precision is well pronounced and the thermal regime contributed to the upper limit for our system precision across all frequency values.

In this paper we demonstrate a single channel system, but we can also scale up our photonic TPE by adding more MRRs and more rails to perform more operations simultaneously. Similar structures for this scale-up idea can be found in the paper by Bangari et al. [11]. However, if we were to scale up our photonic TPE to include multiple channels, then we will need to include noise added by fan-in and fan-out effects. These effects can be categorized into three subcategories: singular case (with only one non-zero input), uncorrelated case (all inputs are independent), identical case (all inputs are same). More specifically, the correlation of inputs affects fan-in gain and therefore signal-to-noise ratio (SNR) [35]. When considering fan-out loss in a multi-channel system, signal root mean square (RMS) value and SNR decrease proportionally. However, accounting for fan-in gain, SNR only increases sub-linearly which results in an overall decrease of signal RMS with more channels. Therefore, laser input power needs to increase sub-linearly to maintain the same level of SNR with more channels. As a result, with the same optical input power and precision level, we should expect the max frequency to drop with more channels due to the additional MRRs, and the fan-in and fan-out effects on different types of noises. In addition, extra MRRs on the bus waveguide will introduce an insertion loss to all signals, but this loss is only measured at around 0.01 dB per MRR when it is off resonance [36].

On the other hand, we have been using grating couplers for optical coupling on our chips and the N-doped heaters on the MRRs for thermal tuning the weight bank. As previously mentioned, grating couplers have a high insertion loss at around 15 dB, but recent work has already shown that photonic wirebonds can be implemented reliably for a much more efficient on/off chip coupling. The round trip loss of a photonic device using photonic wirebonds can be as low as 2 dB [37], which greatly increases the available precision at any given frequency. As shown in Fig 15, by replacing grating couplers for photonic wirebonds, we can achieve up to a 3-bit improvement on the precision across all frequencies. Simultaneously, we can replace the N-doped MRRs with PIN junction modulators for higher speed modulation using carrier depletion effect [17]. Moreover, our analog photonic circuit operate with low laser pump power to avoid nonlinearities during weighting. At low laser pump power, if we were to implement an avalanche photodetector that has active avalanche gain in our current designs, and we can further reduce the thermal noise inside the photodetectors. By replacing the PIN photodetectors with avalanche photodetectors, we can receive a further improvement of around 2 bits on the available precision across all frequencies.

## VII. Conclusion

We have demonstrated the proposed multi-level encoding/decoding scheme for an MRR-based photonic TPE, and the experimental results have verified the feasibility of such

implementation. We have also noted some unique characteristics of the photonic TPE architecture, and by taking advantages of its flexibility we have refined and improved the details of the multi-level encoding scheme. We also combined multi-level encoding scheme with a simple GeMM compiler, and explored the scalability of our photonic tensor processor. The results from larger scale matrix computations have verified that the proposed multi-level encoding scheme can achieve a high level of computational accuracy while providing up to 6-bit signed precision. Combining the multi-level encoding/decoding scheme with a GeMM compiler can serve as the operation foundation allowing us to explore larger-scale ML applications using MRR-based photonic tensor processors.

## ACKNOWLEDGMENT

The authors thank Mohammed Al-Qadasi, Thomas Ferreira de Lima, and Jagmeet Singh for suggestions and experimental support.

## REFERENCES

[1] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019.
[2] K. R. Bokka, S. Hora, T. Jain, and M. Wambugu, *Deep Learning for Natural Language Processing*. Birmingham, U.K.: Packt Publishing, 2019.
[3] L. Shao, H. P. H. Shum, and T. Hospedales, "Editorial: Special issue on machine vision with deep learning," *Int. J. Comput. Vis.*, vol. 128, no. 4, pp. 771–772, 2020.
[4] L. Abdi and A. Meddeb, "Driver information system: A combination of augmented reality, deep learning and vehicular ad-hoc networks," *Multimedia Tools Appl.*, vol. 77, no. 12, pp. 14673–14703, 2018.
[5] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," 2017, *arXiv:1605.07678*.
[6] S. Markidis, S. W. D. Chien, E. Laure, I. B. Peng, and J. S. Vetter, "Nvidia tensor core programmability, performance and precision," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. Workshops*, 2018, pp. 522–531.
[7] N. P. Jouppi et al., "In-datacenter performance analysis of a tensor processing unit," in *Proc. 44th Annu. Int. Symp. Comput. Architecture*, ACM, 2017, pp. 1–12.
[8] Y. Shen et al., "Deep learning with coherent nanophotonic circuits," *Nature Photon.*, vol. 11, pp. 441–446, Jul. 2017.
[9] J. Feldmann et al., "Parallel convolutional processing using an integrated photonic tensor core," *Nature*, vol. 589, pp. 52–58, Jan. 2021.
[10] M. Miscuglio and V. J. Sorger, "Photonic tensor cores for machine learning," *Appl. Phys. Rev.*, vol. 7, Sep. 2020, Art. no. 031404.
[11] V. Bangari et al., "Digital electronics and analog photonics for convolutional neural networks (DEAP-CNNS)," *IEEE J. Sel. Topics Quantum Electron.*, vol. 26, no. 1, pp. 1–13, Jan./Feb. 2020.
[12] B. J. Shastri et al., "Photonics for artificial intelligence and neuromorphic computing," *Nature Photon.*, vol. 15, pp. 102–114, Jan. 2021.
[13] C. Huang et al., "Demonstration of scalable microring weight bank control for large-scale photonic integrated circuits," *APL Photon.*, vol. 5, no. 4, 2020, Art. no. 040803.
[14] W. Zhang et al., "Silicon microring synapses enable photonic deep learning beyond 9-bit precision," *Optica*, vol. 9, pp. 579–584, 2022.
[15] P. Prucnal, B. Shastri, and M. Teich, *Neuromorphic Photonics*. Boca Raton, FL, USA: CRC Press, Jan. 2017.
[16] A. N. Tait et al., "Neuromorphic photonic networks using silicon photonic weight banks," *Sci. Rep.*, vol. 7, no. 1, pp. 1–10, 2017.
[17] A. N. Tait et al., "Silicon photonic modulator neuron," *Phys. Rev. Appl.*, vol. 11, Jun. 2019, Art. no. 064043.
[18] B. A. Marquez et al., "Photonic pattern reconstruction enabled by on-chip online learning and inference," *J. Phys., Photon.*, vol. 3, Feb. 2021, Art. no. 024006.
[19] D. Liang et al., "Fully-integrated heterogeneous DML transmitters for high-performance computing," *J. Lightw. Technol.*, vol. 38, no. 13, pp. 3322–3337, Jul. 2020.
[20] A. N. Tait et al., "Feedback control for microring weight banks," *Opt. Exp.*, vol. 26, pp. 26422–26443, Oct. 2018.
[21] L.-W. Luo, G. S. Wiederhecker, K. Preston, and M. Lipson, "Power insensitive silicon microring resonators," *Opt. Lett.*, vol. 37, no. 4, pp. 590–592, 2012.
[22] J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. P. Pernice, "All-optical spiking neurosynaptic networks with self-learning capabilities," *Nature*, vol. 569, pp. 208–214, May 2019.
[23] H. Jayatilleka et al., "Wavelength tuning and stabilization of microring-based filters using silicon in-resonator photoconductive heaters," *Opt. Exp.*, vol. 23, pp. 25084–25097, Sep. 2015.
[24] M. S. Hai, M. N. Sakib, and O. Liboiron-Ladouceur, "A 16 silicon-based monolithic balanced photodetector with on-chip capacitors for 25 front-end receivers," *Opt. Exp.*, vol. 21, pp. 32680–32689, Dec. 2013.
[25] L. F. Stokes, M. Chodorow, and H. J. Shaw, "All-single-mode fiber resonator," *Opt. Lett.*, vol. 7, pp. 288–290, Jun. 1982.
[26] J. E. Heebner, R. Grover, and T. A. Ibrahim, *Optical Microresonators: Theory, Fabrication, and Applications*, 1st ed., London, U.K.: Springer, 2008, doi: 10.1007/978-0-387-73068-4.
[27] W. Bogaerts et al., "Silicon microring resonators," *Laser Photon. Rev.*, vol. 6, no. 1, pp. 47–73, 2012.
[28] J. J. Dongarra, J. D. Croz, S. Hammarling, and I. S. Duff, "A set of level 3 basic linear algebra subprograms," *ACM Trans. Math. Softw.*, vol. 16, pp. 1–17, Mar. 1990.
[29] V. Kelefouras, A. Kritikakou, I. Mporas, and V. Kolonias, "A high-performance matrix–matrix multiplication methodology for and architectures," *J. Supercomputing*, vol. 72, pp. 804–844, Mar. 2016.
[30] C. Jhurani and P. Mullowney, "A interface and implementation on Nvidia GPUs for multiple small matrices," *J. Parallel Distrib. Comput.*, vol. 75, pp. 133–140, 2015.
[31] S. A. Hassan, M. M. Mahmoud, A. Hemeida, and M. A. Saber, "Effective implementation of matrix–vector multiplication on Intel's multicore processor," *Comput. Lang., Syst. Struct.*, vol. 51, pp. 158–175, 2018.
[32] M. A. Al-Qadasi, L. Chrostowski, B. J. Shastri, and S. Shekhar, "Scaling up silicon photonic-based accelerators: Challenges and opportunities," *APL Photon.*, vol. 7, 2022, Art. no. 020902, doi: 10.1063/5.0070992.
[33] M. Huang et al., "56GHZ waveguide Ge/Si avalanche photodiode," in *Proc. IEEE Opt. Fiber Commun. Conf.*, Optical Society of America, 2018, pp. 1–3.
[34] H. Chen et al., "100-Gbps rz data reception in 67-Ghz Si-contacted germanium waveguide p-i-n photodetectors," *J. Lightw. Technol.*, vol. 35, no. 4, pp. 722–726, Feb. 2017.
[35] A. N. Tait, "Quantifying power in silicon photonic neural networks," *Phys. Rev. Appl.*, vol. 17, May 2022, Art. no. 054029.
[36] A. N. Tait et al., "Microring weight banks," *IEEE J. Sel. Topics Quantum Electron.*, vol. 22, no. 6, pp. 312–325, Nov./Dec. 2016, Art. no. 5900214.
[37] N. Lindenmann et al., "Connecting silicon photonic circuits to multicore fibers by photonic wire bonding," *J. Lightw. Technol.*, vol. 33, no. 4, pp. 755–760, Feb. 2015.

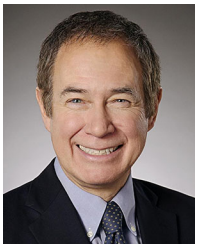
**Zhimu Guo** received the B.A.Sc. degree in engineering physics and computing option and the M.A.Sc. degree from Queen's University, Kingston, ON, Canada, where he is currently working toward the Ph.D. degree. His research focuses on the junction of the hardware and software for computer systems. He is also looking forward to exploring new technologies in the quantum computing realm, including integrated neuromorphic photonic processors for deep learning.

**Alexander N. Tait** (Member, IEEE) received the Ph.D. degree from Lightwave Communications Research Laboratory, Department of Electrical Engineering, Princeton University, Princeton, NJ, USA, under the direction of Paul Prucnal. He is currently an Assistant Professor of electrical and computer engineering with Queen's University, Kingston, ON, Canada. He was a NRC Postdoctoral Fellow with the Quantum Nanophotonics and Faint Photonics Group, National Institute of Standards and Technology, Boulder, CO, USA.

**Bicky A. Marquez** (Member, IEEE) received the bachelor's degree from the Central University of Venezuela, Caracas, Venezuela, in 2012, the master's degree from the Venezuelan Institute for Scientific Research, Parroquia Macarao, Venezuela, in 2014, and the Ph.D. degree in optics and photonics from Bourgogne-Franche-Comté University, Besançon France in 2018, where she worked for Professor Laurent Larger. Her research interests include nonlinear and complex dynamical systems, machine learning, and AI photonic hardware. She likes to spend her free time by traveling and painting/drawing.

**Matthew Filipovich** received the B.A.Sc. degree in engineering physics from Queen's University, Kingston, ON, Canada, where he is currently working toward the master's degree in engineering physics. His research interests include investigating different approaches for neural network training in situ using neuromorphic photonics, including designing a circuit for executing the direct feedback alignment training algorithm.

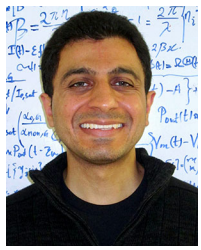
**Hugh Morison** received the B.A.Sc. degree in engineering physics with an option in computing from Queen's University of Technology, Kingston, ON, Canada, where he is currently working toward the graduation degree researching neuromorphic silicon photonic systems. He joined the Shastri Lab with Queen's after the B.A.Sc. degree. His research interests include novel computing systems, artificial intelligence, and experimental demonstrations of silicon photonic neural networks (ANN tasks and network dynamics).



**Paul R. Prucnal** (Life Fellow, IEEE) received the A.B. degree (graduating *summa cum laude*) in mathematics and physics from Bowdoin College, Brunswick, ME, USA, and the M.S., M.Phil., and the Ph. D. degrees in electrical engineering from Columbia University, New York, NY, USA. After his Doctorate, he joined the faculty with Columbia University, where, he was a Member of the Columbia Radiation Laboratory, he performed groundbreaking work in OCDMA and self-routed photonic switching. In 1988, he joined the Faculty with Princeton University, Princeton, NJ, USA. He has authored or coauthored more than 350 journal articles and book chapters and holds 28 U.S. patents. His research on optical CDMA initiated a new research field in which since then more than 1000 papers have been published, exploring applications ranging from information security to communication speed and bandwidth. In 1993, he invented the Terahertz Optical Asymmetric Demultiplexer, the first optical switch capable of processing terabit per second pulse trains. He is the author of the book *Neuromorphic Photonics* and the Editor of the book *Optical Code Division Multiple Access: Fundamentals and Applications*. He was an Area Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS. He is a Fellow of the Optical Society of America (OSA) and the National Academy of Inventors (NAI), and a Member of honor societies, including Phi Beta Kappa and Sigma Xi. He was the recipient of the 1990 Rudolf Kingslake Medal for his paper entitled Self-routing photonic switching with optically-processed control, received the Gold Medal from the Faculty of Mathematics, Physics and Informatics, Comenius University, for leadership in the field of Optics 2006 and has won multiple teaching awards at Princeton, including the E-Council Lifetime Achievement Award for Excellence in Teaching, the School of Engineering and Applied Science Distinguished Teacher Award, and The President's Award for Distinguished Teaching. He is instrumental in founding the field of neuromorphic photonics and developing the photonic neuron, a high-speed optical computing device modeled on neural networks and integrated optical circuits to improve the wireless signal quality by cancelling radio interferences.

**Lukas Chrostowski** (Senior Member, IEEE) is currently a Professor of electrical and computer engineering with the University of British Columbia, Vancouver, BC, Canada. He has authored or coauthored more than 300 journal and conference publications. His research interests include silicon photonics devices, optoelectronics and lasers, including design fabrication and test, for applications in optical communications, computing, biophotonics, and quantum information. He coauthored the textbook *Silicon Photonics Design (Cambridge University Press, 2015)*. He was the Program Director of the NSERC CREATE Silicon Electronic-Photonic Integrated Circuits research training program in Canada (2012–2018).

**Sudip Shekhar** (Senior Member, IEEE) received the B.Tech. degree from the Indian Institute of Technology Kharagpur, Kharagpur, India, in 2003, and the M.Sc. and Ph.D. degrees from the University of Washington, Seattle, WA, USA, in 2005 and 2008, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada. . From 2008 to 2013, he was a Research Scientist with the Circuits Research Laboratory, Intel Corporation, Hillsboro, Oregon. He then joined the ECE department in 2013.



**Bhavin J. Shastri** (Senior Member, IEEE) received the Ph.D. degree in electrical engineering (photonics) from McGill University, Montreal, QC, Canada, in 2012. He is currently an Assistant Professor of engineering physics with Queen's University, Kingston, ON, Canada, and a Faculty Affiliate with the Vector Institute for Artificial Intelligence, Canada. He was an Associate Research Scholar (2016–2018) and Banting/NSERC Postdoctoral Fellow (2012–2016) with Princeton University, Princeton, NJ, USA. He has authored or coauthored more than 70 journal articles and 100 conference proceeding, seven book chapters, and given more than 65 invited talks and lectures including five keynotes and three tutorials. His research interests include silicon photonics, photonic integrated circuits, neuromorphic computing, and machine learning. He is a coauthor of the book (CRC Press, 2017) *Neuromorphic Photonics*, a term he helped coin.

Dr. Shastri was the recipient of the 2022 SPIE Early Career Achievement Award and the 2020 IUPAP Young Scientist Prize in Optics for his pioneering contributions to neuromorphic photonics from ICO. He is a Senior Member of Optica (formerly OSA) and IEEE, recipient of the 2014 Banting Postdoctoral Fellowship from the Government of Canada, the 2012 D. W. Ambridge Prize for the top graduating Ph.D. student at McGill, an IEEE Photonics Society 2011 Graduate Student Fellowship amongst others awards.