Bitwise Neural Network Acceleration Using Silicon Photonics

Kyle Shiflett
Ohio University
Athens, Ohio, USA
ks117713@ohio.edu

Avinash Karanth
Ohio University
Athens, USA
karanth@ohio.edu

Ahmed Louri
George Washington University
Washington, D.C., USA
louri@gwu.edu

Razvan Bunescu
The University of North Carolina at Charlotte
Charlotte, USA
rbunescu@uncc.edu

ABSTRACT

Hardware accelerators provide significant speedup and improve energy efficiency for several demanding deep neural network (DNN) applications. DNNs have several hidden layers that perform concurrent matrix-vector multiplications (MVMs) between the network weights and input features. As MVMs are critical to the performance of DNNs, previous research has optimized the performance and energy efficiency of MVMs at both the architecture and algorithm levels. In this paper, we propose to use emerging silicon photonics technology to improve parallelism, speed and overall efficiency with the goal of providing real-time inference and fast training of neural nets. We use microring resonators (MRRs) and Mach-Zehnder interferometers (MZIs) to design two versions (all-optical and partial-optical) of hybrid matrix multiplications for DNNs. Our results indicate that our partial optical design gave the best performance in both energy efficiency and latency, with a reduction of 33.1% for energy-delay product (EDP) with conservative estimates and a 76.4% reduction for EDP with aggressive estimates.

CCS CONCEPTS

• **Computer systems organization → Parallel architectures**;
• **Hardware → Emerging optical and photonic technologies**.

KEYWORDS

deep neural networks; silicon photonics; optical signal processing

ACM Reference Format:
Kyle Shiflett, Avinash Karanth, Ahmed Louri, and Razvan Bunescu. 2021. Bitwise Neural Network Acceleration Using Silicon Photonics. In *Proceedings of the Great Lakes Symposium on VLSI 2021 (GLSVLSI '21), June 22–25, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3453688.3461515

GLSVLSI '21, June 22–25, 2021, Virtual Event, USA.

ACM ISBN 978-1-4503-8393-6/21/06…$15.00
https://doi.org/10.1145/3453688.3461515

1 INTRODUCTION

The combined effects of escalating power densities due to higher transistors, and the performance limitation of instruction-level parallelism gave rise to multicore systems. Yet, due to the breakdown of Dennard scaling, multicore systems too have been vexed by the power barrier, impeding performance advances [4]. In order to continue performance scaling, chip architects have shifted their focus towards application-specific accelerator designs that surpass the efficiency and functionality of general purpose processors. Machine learning (ML) architectures such as deep neural networks (DNNs) are of particular interest due to their unparalleled accuracy on contemporary applications such as speech recognition and image classification.

DNNs are formed by placing several highly linked layers between the inputs and outputs of the network, which allows for the model to accomplish great precision by controlling the weights of each connection. The hidden layers of these networks may involve several thousands of concurrent matrix-vector multiplications (MVMs) between the network weights and the set of input features. MVM and dot products are fundamental operations in DNN architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), and are used throughout model training and inference.

There are design challenges in DNN architectures that can detrimentally affect the computing speeds and power efficiency. First, real world applications demand millions of MAC operations in each network layer, and with DNNs comprised of a multitude of hidden layers, scaling of DNNs poses serious challenges. Second, in order to maintain high levels of parallelism, data distribution must be done efficiently in these systems [10]. In the electronic-based accelerators, large broadcast buses used for parallel data distribution are limited by electronic clock rates and consume excess power. These fundamental physical limitations prevent the effective scaling of hardware accelerators to maximally exploit the parallelism found in neural networks.

Emerging technology, such as silicon photonics, is capable of producing high processing bandwidths with high power efficiency [3]. The high amount of parallelism, energy efficiency, and ease of broadcast/multicast capabilities of silicon photonics are well suited for the design of highly efficient and scalable neural network accelerators [7, 8]. Harnessing the properties of light, linear transformations can efficiently be performed on data sets at high rates. While substantial prior work has been published on the use

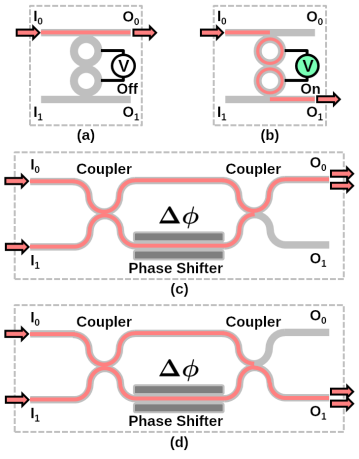

Figure 1: (a) shows dual-MRRs with V_{off}, (b) shows dual-MRRs with V_{on}, (c) shows MZI with both inputs appearing at O_0 (additive), and (d) shows MZI with inputs appearing at O_1.

of silicon photonics for hardware acceleration, none of the prior work have analyzed the device properties sufficiently to provide system-level exploration on how efficient DNN engines can be implemented with current device technology. Prior works such as PIXEL [8] and PCNNA [7] fail to address the bit-level parallelism in MVMs, which we directly address in this work. This work reduces optical communication overhead compared to PIXEL's mesh of x-y crossbars by using a simplified tree distribution network. Furthermore, by performing bit-level operations we avoid precision limitations of purely analog systems like PCNNA.

In this paper, we leverage the unique properties of silicon photonics to design efficient matrix-vector multiplication (MVM) and accumulation for use in fast neural network accelerators. The designs are based on the use of microring resonators (MRR), MZIs, lasers, and optical waveguides integrated with electronic processing to perform highly parallel MVM and accumulation functionality. MRRs and MZIs are well developed technologies that have the minimal footprint and bandwidth density required for high speed optical processing. We design two versions of our optical-electrical hybrid matrix multipliers. The first design uses MRRs to perform bitwise *AND* operation with electronic processing for summation (O-E-E). The second design uses MRRs to perform bitwise AND operation, but also has MZIs to perform optical accumulation, with a final electrical summation (O-O-E). We perform a thorough design-space exploration that evaluates power, latency, and area requirements for different versions of our designs with respect to both inputs and number of bits.

2 BACKGROUND

2.1 Photonic Devices

2.1.1 Microring Resonator. Microring resonators (MRRs) are a promising technology that have been used for modulation, demodulation, and switching in optical interconnects. MRRs are desirable due to their small footprint (7.5 µm radius) and low energies

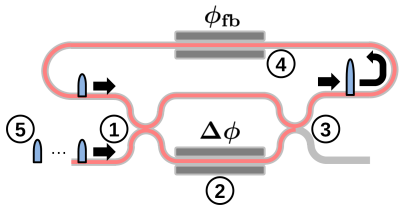

Figure 2: Bitwise MZI optical accumulator with differential phase shift $\Delta\phi$ and feedback phase shift ϕ_{fb}.

(<100 fJ/bit including necessary thermal tuning) [6]. Figure 1(a) shows the basic operation of double MRR when no voltage (V_{off}) is applied. The incoming signal from input port I_0 arrives at output port O_0. When voltage is applied to double MRR (V_{on}), if the incoming signal is in resonance with the ring resonator, the signal will appear at output port O_1. With an incoming optical signal A (A=I_0), and an applied ring voltage B as the two inputs to the double MRR, the signal appearing at output port O_1 will mimic an optical *AND* gate (Y = A AND B) as shown in Figure 1(a,b). For the optical *AND* operation, presence of an optical signal corresponds to A=1, and absence of optical signal corresponds to A=0. The same is true for the MRR weight (voltage), an "on" voltage corresponds to B=1, and an "off" voltage corresponds to B=0.

2.1.2 Mach-Zehnder Interferometer. Mach-Zehnder Interferometers (MZIs) [2] manipulate two light signals by coupling the signals, applying phase shifts in its waveguide arms, and then coupling the shifted signals before appearing at the output ports. MZIs are 2-input 2-output port devices, and are depicted in Figure 1(c,d). Depending on the phase shifts applied in the phase-shifting arms, the MZI can operate in 3 states. The phase delay $\phi_i = (2\pi/\lambda)n_i L_i$, where λ is wavelength of the optical signal, n_i is the refractive index of arm i, and L is the path length of arm i and $\Delta\phi = \phi_0 - \phi_1$ is the differential phase shift. The first state is the bar-state ($|\Delta\phi| = \pi$), which passes input I_0 to output O_0 and passes input I_1 to output O_1. The second state is the cross-state ($|\Delta\phi| = 0$), which passes input I_0 to output O_1 and passes input I_1 to output O_0. The third state is the tunable-state, where the two output ports can have varying amplitudes of the combined input signals, depending on the phase shifts applied in the MZI arms ($|\Delta\phi| = \pi/2$ or $|\Delta\phi| = 3\pi/2$). This state is useful when modulating the two signals, or combining two signals into a single output port, as shown in Figure 1(c,d).

2.1.3 MZI Accumulator. We create an MZI adder that performs optical bitwise accumulation as shown in Figure 2. The optical accumulator is formed by introducing a feedback waveguide from the top output of the MZI back to the top input of the MZI. The top arm of the MZI accumulator has no phase shifter, and will have $\phi_0 = 0$. This means that the phase shifter on the bottom arm of the MZI accumulator (ϕ_1) will be solely responsible for controlling the value of $\Delta\phi$. The feedback arm of the MZI accumulator also has a phase shifter (ϕ_{fb}), which will tune the accumulation signal so it will be in phase with the next input for further accumulation.

2.2 Bitwise Matrix-Vector Multiplication

Since MVM, and subsequently dot products, is the reoccurring operation that is found in DNN architectures, it is necessary to

Table 1: Bit position (BP) distribution for a 4-bit multiplication across 4 Processing Elements (PEs).

BP	PE	Sum
0	PE_0	$x_0^0 w_{00}^0$
1	PE_1	$x_0^0 w_{00}^1 + x_0^1 w_{00}^0$
2	PE_2	$x_0^0 w_{00}^2 + x_0^1 w_{00}^1 + x_0^2 w_{00}^0$
3	PE_3	$x_0^0 w_{00}^3 + x_0^1 w_{00}^2 + x_0^2 w_{00}^1 + x_0^3 w_{00}^0$
4	PE_0	$x_0^1 w_{00}^3 + x_0^2 w_{00}^2 + x_0^3 w_{00}^1$
5	PE_1	$x_0^2 w_{00}^3 + x_0^3 w_{00}^2$
6	PE_2	$x_0^3 w_{00}^3$

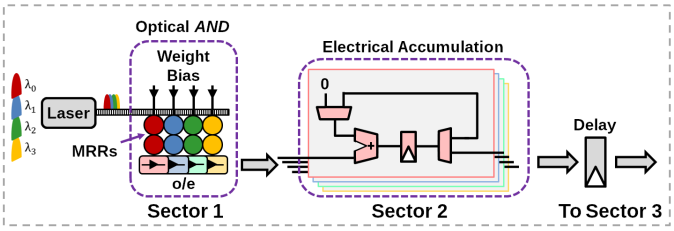

Figure 3: PE design with optical *AND* and electrical accumulation (O-E-E), shown operating on 4 wavelengths.

explore the data distribution of this operation to exploit a sufficient amount of parallelism. Let's consider the equation for the outputs generated in a 4-input multilayer perceptron (MLP) as shown in Equation 1.

$$f\left(\begin{bmatrix} w_{00} & w_{01} & w_{02} & w_{03} \\ w_{10} & w_{11} & w_{12} & w_{13} \\ w_{20} & w_{21} & w_{22} & w_{23} \\ w_{30} & w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}\right) = \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} \quad (1)$$

In this equation, w_{ij} represents a weight associated with an input feature x_j. The bias is represented by b_i, whereas h_i is the activation of the neuron i on the hidden layer. The linear transformation of the input is fed through a nonlinear activation function f to obtain the output h_i. Now let's consider the equation for a single neuron shown in Eq. 2.

$$h_0 = f\left(x_0 w_{00} + x_1 w_{01} + x_2 w_{02} + x_3 w_{03} + b_0\right) \quad (2)$$

In order to accelerate this expression, we must further decompose into the bit-level representation of the data in order to fully understand how the hardware will handle these mathematical operations. Let's take the multiplication $x_0 w_{00}$, we will assume for this example that each feature x_j and weight w_{ij} is a 4-bit value, where the notation is x_j^{bit} and w_{ij}^{bit}, with 0 being the most significant bit (MSB) and 3 being the least significant bit (LSB).

The sum for each bit position (BP) is displayed in Table 1. Note this does not take bit-carries into consideration yet, as they will be handled in a different level. These sums are arranged into n processing elements (PEs) for n bit multiplication. By arranging the bit-level *AND*s in this manner, and executing an *AND* operation in each clock cycle, exploitable parallelism becomes more obvious. In each cycle, the same x_j^{bit} is used by each PE, implying that we can broadcast this bit to all PEs. Also, if the weight bits are viewed as a matrix, where the rows are the PE, and the columns are the cycle number, it is seen that the same 4 bits are used every cycle, but in a different arrangement. This arrangement is a circulant matrix, which is the same column vector being shifted by one position each cycle. Using this circulant matrix arrangement reduces retransmission, and x_j^{bit} broadcasting allows for bit-level data parallelism.

2.3 Electrical Accelerator

The accelerators are split into 3 sectors, and follow the BP distribution described in Table 1. Sector 1 (S1) contains the bitwise *AND* functionality for $x_j^{bit} w_{ij}^{bit}$. Sector 2 (S2) contains the bit-level accumulation of the *AND*s performed in S1, and provides the result for each BP of the entire multiplication. In our 4-bit example, PE_0 will be fed BP_0 ($x_0^0 w_{00}^0$) in cycle 1 which will then be redirected to the BP_0 output. Due to each BP result being output at a different cycle, delay logic must be added to certain BP lines in order to synchronize all BP results. So in this case, BP_0 must undergo a 3 cycle delay, BP_1 must undergo a 2 cycle delay, and BP_2 must undergo a 1 cycle delay to ensure all BPs are available to Sector 3 (S3) at the same time.

This is the computation of the first multiplication term for h_0, which is ($h_0^0 = x_0 w_{00}$). This hardware is replicated for a total of four times in parallel to compute the values of h_0^1, h_0^2, and h_0^3. S3 is responsible for the final additions with carries and activation function implementation to obtain output h_0. To compute the output h_0, the sum ($h_0^0 + h_0^1 + h_0^2 + h_0^3 + b_0$) will be calculated using an adder tree. The result from the adder tree will then be fed into the activation function hardware to obtain the final result h_0. The activation hardware is a hybrid piecewise linear with bit-level mapping design from [12]. Although the main activation function for CNNs used is the ReLu function (a compare with 0), the hybrid piecewise linear hardware implementation is included to handle hyperbolic tangent or sigmoid for flexibility. All 3 sectors of this design are electrical, and the whole design will be referred to as E-E-E as a fundamental comparison for our hybrid optical-electrical accelerator designs.

3 PHOTONIC ACCELERATOR DESIGN

In this section, we propose two photonic accelerators (O-E-E) and (O-O-E) which use photonic devices for different sectors (S1 and S2) and electrical devices for summation (S3).

3.1 Optical-Electrical-Electrical (O-E-E)

Figure 3 shows the design layout for the Optical-Electrical-Electrical (O-E-E) accelerator PE. For our photonic designs, an off-chip laser with a MRR bank modulates the signals for each input feature x_j, where it is split using a series of 3 dB Y-splitters [13] to each PE. We can submit a light pulse bit that may contain up to several different wavelengths at the same instance using wavelength-division multiplexing (WDM). WDM allows for large amounts of information to be efficiently broadcasted on the same waveguide, increasing bandwidth densities beyond the possibilities of electrical wires.

Each input feature x_j is mapped to a separate wavelength, and the number of wavelengths of the system is equal to the number of

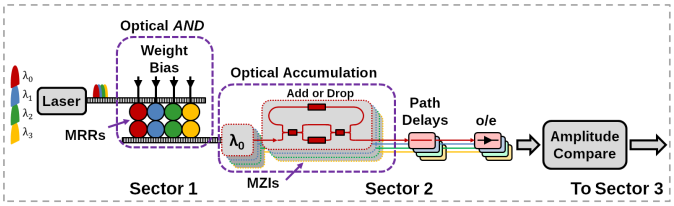

Figure 4: PE design with optical *AND* and optical accumulation (O-O-E), shown operating on 4 wavelengths.

input features. So for a 4-input system, feature x_0 will be carried by wavelength λ_0, feature x_1 will be carried by wavelength λ_1, and so on. Each bit of the input feature x_j^{bit} is transmitted as an optical pulse on wavelength λ_j, and the entire input feature x_j is transmitted as a pulse train.

The MRRs shown in the "Optical *AND*" box in Figure 3, along with the waveguide tree splitter, make up S1. The circulant weight matrix controls the voltage of the MRRs, providing optical *AND* functionality between bit x_j^{bit} and w_{ij}^{bit}, and there is a circulant matrix associated with each wavelength to control these MRRs. Since an optical pulse can contain several different wavelengths using WDM, i.e. containing information for multiple inputs x, we can exploit further parallelism by performing optical *AND*s on multiple wavelengths using the same waveguide.

The second box "Electrical Accumulation" shows the accumulation logic described in the electrical accelerator section. The optical bits must undergo an optical-to-electrical (o/e) conversion, achieved with photodetectors, before entering the electrical logic. Next, the bit will either go to the output BP it is mapped to, or will undergo accumulation. By adding the incoming bit with 0 selected by the multiplexer, this allows it to pass through unchanged. The accumulation can occur by forwarding the output of the adder back into its input. Once all BPs are calculated, the delay logic, consisting of D flip-flops (DFFs) on select BP lines, will then provide the result to the S3 adder tree. The electrical logic in S2 is replicated for each input, and corresponds to its respective wavelength.

3.2 Optical-Optical-Electrical (O-O-E)

Figure 4 shows the design layout for the Optical-Optical-Electrical (O-O-E) accelerator PE. As in the O-E-E accelerator, S1 consists of the off-chip laser source, and the double MRR optical *AND* functional units. After the optical *AND* is carried out, the accumulation must occur, shown in the second box "Optical Accumulation". Optical accumulation is implemented with MZIs, where the optical bit can be fed back into the MZI or dropped to the output waveguide. The waveguide path that connects the upper output of the accumulation MZI back to its input is chosen such that the length of the path incurs a delay precisely in sync with the next incoming optical bit, and includes the phase shifter ϕ_{fb} to tune the optical pulse as described in Section 2.1.3.

Accumulation with photonics is achieved by feeding an optical bit back into an MZI, which results in an additive analog signal that is proportional to the sum of the input signals. Some BP lines must undergo a delay as well, but instead of DFFs limited by the electronic

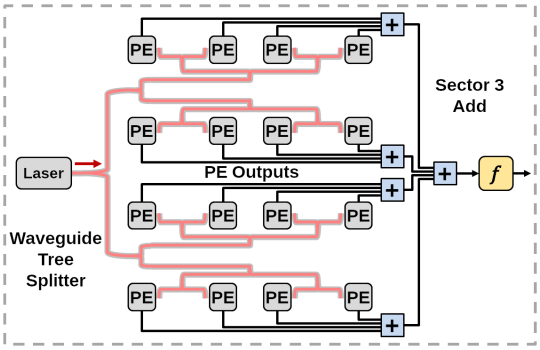

Figure 5: 16 Processor Element (PE) accelerator with waveguide tree splitter, Sector 3 (S3) summation, and activation function.

Table 2: Photonic losses used in O-E-E and O-O-E designs.

Component	Loss
Single MRR[6]	-0.045 dB
Double MRR[14]	-0.1 dB
Straight Waveguide	-1.0 dB/cm
90° Waveguide Bend[11]	-0.06 dB
180° Waveguide Bend[11]	-0.07 dB
Waveguide Y-Splitter[13]	-0.3 dB
Directional Coupler[5]	-0.8 dB
Photodetector[1]	-0.61 dB
Photodetector Sensitivity[1]	-12.7 dBm

clock rate, the waveguide of these lines is simply lengthened by the correct amount for the desired delay. This avoids extra delay circuitry and keeps the latency low.

Once the delays are performed, the BP lines undergo an o/e conversion before being fed to the S3 adder tree. The o/e conversion performed on the optical bits in this design is more involved than in the O-E-E design. Since the light pulses can have different amplitudes to represent their values, we must be able to extract this analog information and convert it into the digital domain. The photodiodes used as photodetectors provide an electrical current that is proportional to the amount of photons absorbed, so comparator logic within the o/e modules is needed to convert the light amplitudes into their binary values.

3.3 Scaling for larger architectures

Scaling up the O-E-E and O-O-E designs to higher PE counts still requires that the optical signal be distributed to each PE, and we have chosen to do this with a waveguide tree splitter for two reasons. First, by using a splitter we can ensure that each wavelength is still available to all PEs. Second, by using a tree design, the PEs are all the same distance from the laser source, meaning that they will observe their optical inputs at the same time. Figure 5 shows the layout of a 16-PE design with the waveguide tree splitter for the distribution of broadcasted data.

4 EVALUATION

Each component used was simulated to obtain their energy/bit, area, and propagation delays for a compute-front evaluation of the

architecture designs. Both electrical and photonic components were modelled using the DSENT simulator [9], which provides cosimulation of photonic and electrical circuits.By using the DSENT simulator, we strive to show a fair comparison between electrical and photonic components. Electrical components were modelled using 22nm technology (Bulk22LVT) provided by DSENT, and photonic components were modelled from the referenced literature in this section. The designs were then evaluated based on the number of inputs (2^0 to 2^6) and number of bits (2^0 to 2^6) to show scaling for all architectures.

Let's take the MRR *AND* logic of a 16-input 16-bit O-O-E (Conservative) design for example. The overall number of MRRs in the design is calculated by $2 \times n_b \times n_{in}$, where n_b is the number of bits per input, and n_{in} is the number of inputs. There will be 512 MRRs, and with each operating at 100 fJ/bit (including ring heating) for 16 bits, they will collectively consume 81.9 nJ. This is assuming an always-on voltage, which will be the MRR worst-case energy since bias weights can contain several 0s. The propagation delay can be calculated for optical components based of the path length (PL) that a single light pulse must travel. We will calculate the path length of the photonic splitter. The 1x16 tree splitter is made up of waveguide Y-splitters cascaded together, and is shown in Figure 5. The total PL for a node at the end splitter from the laser source is estimated to be $PL_Y = 14$ mm, and with the group refractive index of $n_g = 3$ at 1550 nm wavelength, the propagation delay will then be $D_Y = PL_Y \times n_g \div c = 140$ ps, where c is the speed of light. In a similar evaluation methodology, an electrical 16-bit carry-lookahead adder will have a gate count of 1064, and using the Bulk22LVT technology from DSENT at 2 GHz will consume 1.26 pJ.

For optical designs, aggressive and conservative estimates were used for evaluation. These estimates rely on only 3 changed parameters: optical modulation frequency, MRR energy/bit, and MZI energy/bit. In the conservative designs, the parameters are 10 GHz modulation frequency, 100 fJ/bit for MRR, and 450 fJ/bit for MZI. In the aggressive designs, the parameters are 12 GHz optical frequency, 50 fJ/bit for MRR, and 100 fJ/bit for MZI. 7.5 µm radius MRRs have been shown to have modulation energy as low as 7 fJ/bit [6] with tuning energy less than 100 fJ/bit. MZIs have been demonstrated at 32.4 fJ/bit [2], which was taken into consideration when determining conservative and aggressive estimates for our photonic designs. Photonic losses used in our photonic designs are shown in Table 2.

Our accelerator designs were evaluated on the following CNN architectures: AlexNet, ZFNet, ResNet-34, VGG-16, and GoogleNet. These architectures were broken down to their shapes at each convolution layer to determine the amount of data that needs to be distributed to each PE, as well at the total number of multiplies, additions, and activation functions necessary for an inference. For example, the first convolution layer of VGG-16 has input shape [224x224x3] (length x width x channels) with 64 feature maps and receptive field of size [3x3], which would require 3,211,264 kernel dot products, each resulting in 27 multiplications (3x3 receptive field x3 channels) and 26 additions. We can then map these numbers to our design and evaluate for a given bit precision, and apply it with our energy and latency results on a per-device basis.

Energy Efficiency: The energy consumption was calculated for all designs on a per-device level. Table 3 shows the energy consumed for each component in a 16-input 16-bit design. The O-O-E (A) design consume the lowest amount of energy, though the optical designs are consuming substantial energy due to the high number of MRRs and MZIs. In general, though, with larger number of bits, the energies for the optical designs scale well. O-O-E (A) consumes 53.8% the energy that the E-E-E designs does. Our results also show that for a waveguide tree splitter to broadcast 16 values (one per wavelength) at 16-bits each, only 2.8 pJ is consumed (this includes modulation, leakage, ring tuning, and laser), whereas to perform an electrical broadcast on the same amount of data would require 390 pJ to distribute. The energies are assumed with always-active components, which gives a worst-case estimation since signals are not always held high for real-world operations.

Table 3: Energy breakdown by component for E-E-E, O-E-E, and O-O-E designs for a 16 input, 16 bit MVM [nJ].

Component	E-E-E	O-E-E (C)	O-E-E (A)	O-O-E (C)	O-O-E (A)
E-Bcast	0.39				
O-Bcast		0.0028	0.0036	0.0028	0.0036
PE wire	2.41	0.98	0.98	0.44	0.44
E-AND	1.86				
E-Acc	33.3	33.3	33.3		
E-Delay	293.0	293.0	293.0		
o/e		0.073	0.073	0.007	0.007
MRR		52.4	26.2	52.4	26.2
MZI				236	52.4
λ-Filter				26.2	13.1
S3 Sum	179	179	179	179	179
Act Func	0.059	0.059	0.059	0.059	0.059
Total	**510.0**	**558.8**	**532.6**	**494.1**	**274.2**

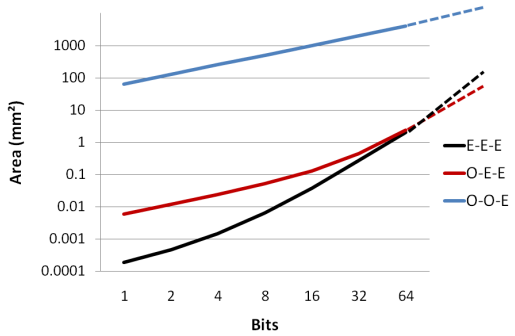

Figure 6: Area analysis of E-E-E, O-E-E, and O-O-E designs for 16 inputs and varying bits (1-64).

Area: Figure 6 shows the area occupied by each design. Since photonic devices can be quite large when compared to CMOS components, optical designs will have considerably larger areas than their electrical counterparts. Although the areas for our photonic designs are quite large, they scale well when compared to all-electrical implementations. The dashed lines in Figure 6 show an area projection beyond what was simulated, and while E-E-E begins to trend upwards with increasing bits, the O-O-E design scaling remains steady.

Latency: The latency results for each design are shown in Figure 7. The latencies of E-E-E, O-E-E (C), and O-E-E (A) are similar because they are limited by the clock rate of their electrical components.

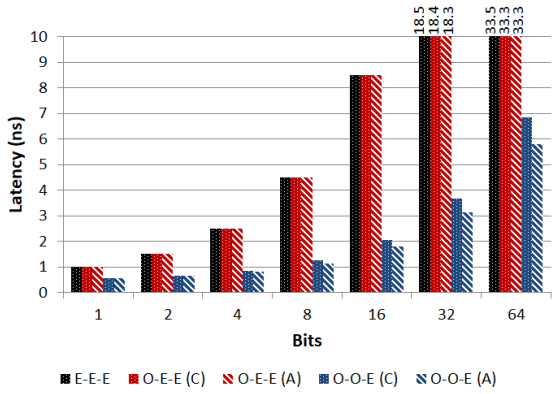

Figure 7: Latency for a 16 input MVM and varying bits (1-64).

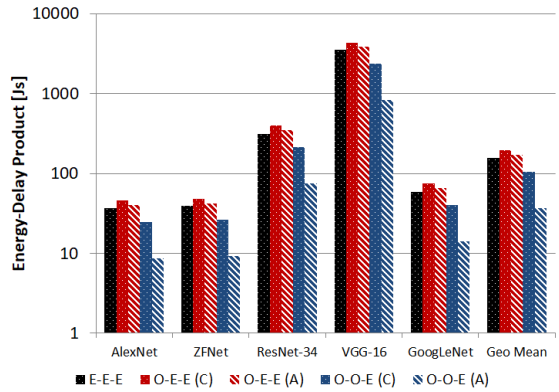

Figure 8: CNN architecture energy-delay product (EDP) evaluation for a 16 PE design with 16 bit precision.

The benefits of photonics can be seen in the O-O-E designs, where the high optical rates allow for fast data manipulation. The O-O-E latencies scale well compared to E-E-E and O-E-E designs. At 16-inputs and 16-bits, O-O-E (C) is 77.8% faster than E-E-E, and O-O-E (A) is 80.8% faster than E-E-E. If we scale the design up to 64 bits, O-O-E (C) is 79.4% faster than E-E-E, and O-O-E (A) is 82.7% faster than E-E-E. These are computational latencies, assuming that operands are fetched without significant penalty. This will change depending on the memory system used, however these results show the low latency and dense computation abilities of silicon photonics.

Energy-Delay Product: The energy-delay product (EDP) is a useful performance metric that captures the trade-off between energy efficiency and speed of a system. Figure 8 shows the EDP for a 16-PE design with 16-bit precision evaluated for various CNN architectures. It can be seen that O-O-E (C) and O-O-E (A) provide a low EDP compared to the E-E-E and O-E-E designs. In O-O-E, the MRRs for optical *AND* combined with the MZIs for optical accumulation allow the design to outperform E-E-E and O-E-E across all CNN architectures. On average when compared to E-E-E, the EDP of O-O-E (C) is 33.1% lower and O-O-E (A) is 76.4% lower.

5 CONCLUSIONS

In this paper, we proposed two electrical-optical hybrid MVM accelerators for use with DNNs. Our proposed architectures utilize emerging photonic devices in a manner that leverages their low-energy low-latency properties for MVM computation at the bit-level. We found that the O-O-E design gave the best performance, with a reduction of 33.1% for EDP with conservative estimates and a 76.4% reduction for EDP with aggressive estimates. The optical designs have demonstrated efficient scaling in energy, area, and latency.

ACKNOWLEDGMENTS

This research was partially supported by NSF grants CCF-1513606, CCF-1702980, CCF-1703013, CCF-1812495, CCF-1901165, CCF-1901192, and CCF-1953980.

REFERENCES

[1] Daniel Benedikovic, Léopold Virot, Guy Aubin, Farah Amar, Bertrand Szelag, Bayram Karakus, Jean-Michel Hartmann, Carlos Alonso-Ramos, Xavier Le Roux, Paul Crozat, Eric Cassan, Delphine Marris-Morini, Charles Baudot, Frédéric Boeuf, Jean-Marc Fédéli, Christophe Kopp, and Laurent Vivien. 2019. 25 Gbps low-voltage hetero-structured silicon-germanium waveguide pin photodetectors for monolithic on-chip nanophotonic architectures. *Photon. Res.* 7, 4 (Apr 2019), 437–444. https://doi.org/10.1364/PRJ.7.000437

[2] J. Ding, R. Ji, L. Zhang, and L. Yang. 2013. Electro-Optical Response Analysis of a 40 Gb/s Silicon Mach-Zehnder Optical Modulator. *Journal of Lightwave Technology* 31, 14 (July 2013), 2434–2440. https://doi.org/10.1109/JLT.2013.2262522

[3] Po Dong, Wei Qian, Hong Liang, Roshanak Shafiiha, Ning-Ning Feng, Dazeng Feng, Xuezhe Zheng, Ashok V. Krishnamoorthy, and Mehdi Asghari. 2010. Low power and compact reconfigurable multiplexing devices based on silicon microring resonators. *Opt. Express* 18, 10 (May 2010), 9852–9858. https://doi.org/10.1364/OE.18.009852

[4] Hadi Esmaeilzadeh, Emily Blem, Renee St. Amant, Karthikeyan Sankaralingam, and Doug Burger. 2011. Dark Silicon and the End of Multicore Scaling. In *Proceedings of the 38th Annual International Symposium on Computer Architecture* (San Jose, California, USA) *(ISCA '11)*. ACM, New York, NY, USA, 365–376. https://doi.org/10.1145/2000064.2000108

[5] R. K. Gupta, S. Chandran, and B. K. Das. 2017. Wavelength-Independent Directional Couplers for Integrated Silicon Photonics. *Journal of Lightwave Technology* 35, 22 (2017), 4916–4923.

[6] Guoliang Li, Xuezhe Zheng, Jin Yao, Hiren Thacker, Ivan Shubin, Ying Luo, Kannan Raj, John E. Cunningham, and Ashok V. Krishnamoorthy. 2011. 25Gb/s 1V-driving CMOS ring modulator with integrated thermal tuning. *Opt. Express* 19, 21 (Oct 2011), 20435–20443. https://doi.org/10.1364/OE.19.020435

[7] Armin Mehrabian, Yousra Al-Kabani, Volker J Sorger, and Tarek El-Ghazawi. 2018. PCNNA: A Photonic Convolutional Neural Network Accelerator. *2018 31st IEEE International System-on-Chip Conference (SOCC)* (Sep 2018). https://doi.org/10.1109/socc.2018.8618542

[8] K. Shiflett, D. Wright, A. Karanth, and A. Louri. 2020. PIXEL: Photonic Neural Network Accelerator. In *2020 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 474–487.

[9] C. Sun, C. O. Chen, G. Kurian, L. Wei, J. Miller, A. Agarwal, L. Peh, and V. Stojanovic. 2012. DSENT - A Tool Connecting Emerging Photonics with Electronics for Opto-Electronic Networks-on-Chip Modeling. In *2012 IEEE/ACM Sixth International Symposium on Networks-on-Chip*. 201–210. https://doi.org/10.1109/NOCS.2012.31

[10] V. Sze, Y. Chen, T. Yang, and J. S. Emer. 2017. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. *Proc. IEEE* 105, 12 (2017), 2295–2329.

[11] Shijun Xiao, Maroof H. Khan, Hao Shen, and Minghao Qi. 2007. Modeling and measurement of losses in silicon-on-insulator resonators and bends. *Opt. Express* 15, 17 (Aug 2007), 10553–10561. https://doi.org/10.1364/OE.15.010553

[12] B. Zamanlooy and M. Mirhassani. 2014. Efficient VLSI Implementation of Neural Networks With Hyperbolic Tangent Activation Function. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 22, 1 (Jan 2014), 39–48. https://doi.org/10.1109/TVLSI.2012.2232321

[13] Yi Zhang, Shuyu Yang, Andy Eu-Jin Lim, Guo-Qiang Lo, Christophe Galland, Tom Baehr-Jones, and Michael Hochberg. 2013. A compact and low loss Y-junction for submicron silicon waveguide. *Opt. Express* 21, 1 (Jan 2013), 1310–1316. https://doi.org/10.1364/OE.21.001310

[14] Linjie Zhou, Richard Soref, and Jianping Chen. 2015. Wavelength-selective switching using double-ring resonators coupled by a three-waveguide directional coupler. *Opt. Express* 23, 10 (May 2015), 13488–13498. https://doi.org/10.1364/OE.23.013488