ACS Photonics

pubs.acs.org/journal/apchd5

Article

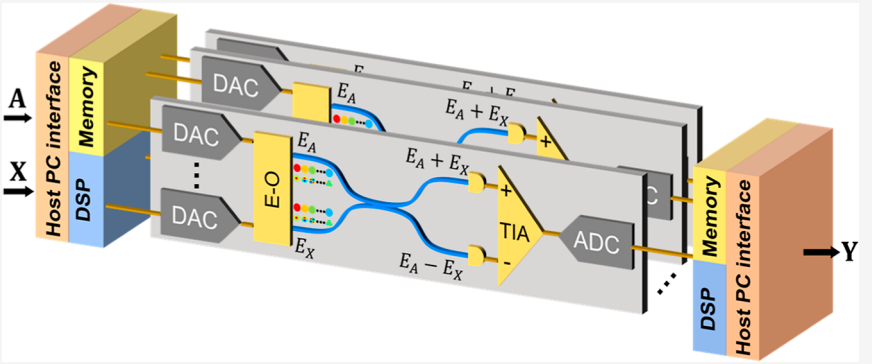
Coherent General-Purpose Photonic Matrix Processor

Zheyuan Zhu,* Alireza Fardoost, Fatemeh Ghaedi Vanani, Andrew B. Klein, Guifang Li,* and Shuo S. Pang*

Cite This: https://doi.org/10.1021/acsphotonics.3c01694

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: Matrix computations are at the heart of scientific computing, especially in models involving large-scale linear systems. As the scale and complexity of the problems grow, energy-efficient matrix computation becomes critical in these applications. Meanwhile, the advantages of miniaturizing conventional digital electronic processors, predicted by the Dennard scaling, diminish in post-Moore's law era. Analogue photonic devices based on passive and high-throughput interconnects are becoming promising alternatives as next-generation energy-efficient computing units. However, the limited reconfigurability and precision of an analogue photonic computing device make it unsuitable for scientific computing applications. Here, we report a general-purpose analogue photonic matrix processing unit (MPU) based on coherent analogue photonic cores, which perform signed multiplications, with reconfigurability and memory provided by digital electronics. Combined with error management strategies, our photonic MPU can perform tasks conventionally dominated by floating-point digital processors, elevating analog photonic-based platforms toward scientific computing applications. We have experimentally demonstrated its feasibilities in a range of computing tasks, including matrix multiplication and inversion as well as solving finite-difference partial differential equations.

KEYWORDS: *matrix processing, digital, photonic processors, analog computing, reconfigurability, MPU*

INTRODUCTION

Numerical computing plays an essential role in addressing many of the challenges in today's society. From modeling of financial markets to astrophysics, large-scale dynamic problems are routinely solved numerically. As all fields of science evolve, more computing power is required to process ever-increasing sets of equations. Typically, solving these systems requires intensive matrix multiplication with millions of elements and trillions of multiply accumulate (MAC) operations. Yet, conventional digital processors, such as CPUs and GPUs, cannot keep up with the growing demand for computing power in post-Moore's law era.[1−3] Alternative computing schemes have been proposed, and in particular, analog accelerators have demonstrated high throughput and energy efficiency[4−8] in various large-scale computing scenarios.

Unlike digital processors that fetch data/instructions and execute them at rising edges of the clock signals, analog accelerators rely on their intrinsic physical processes to model specific mathematical operations. For example, in memristor crossbar arrays, multiplications are performed as the voltage is applied to the rows, and the conductance of the cells produces currents on the readout columns. The currents from all memristor cells along the column accumulate following Kirchoff's law, implementing MAC operation.[9] Photonic accelerators can operate at greater bandwidth and parallelism thanks to the numerous orthogonal degrees-of-freedom in optical signals,[10] thus they could offer higher computing throughput than their electronic counterparts. Currently, analogue photonics has found success in artificial intelligence (AI)-related computing paradigms, including neuromorphic

Received: November 21, 2023
Revised: February 26, 2024
Accepted: February 27, 2024

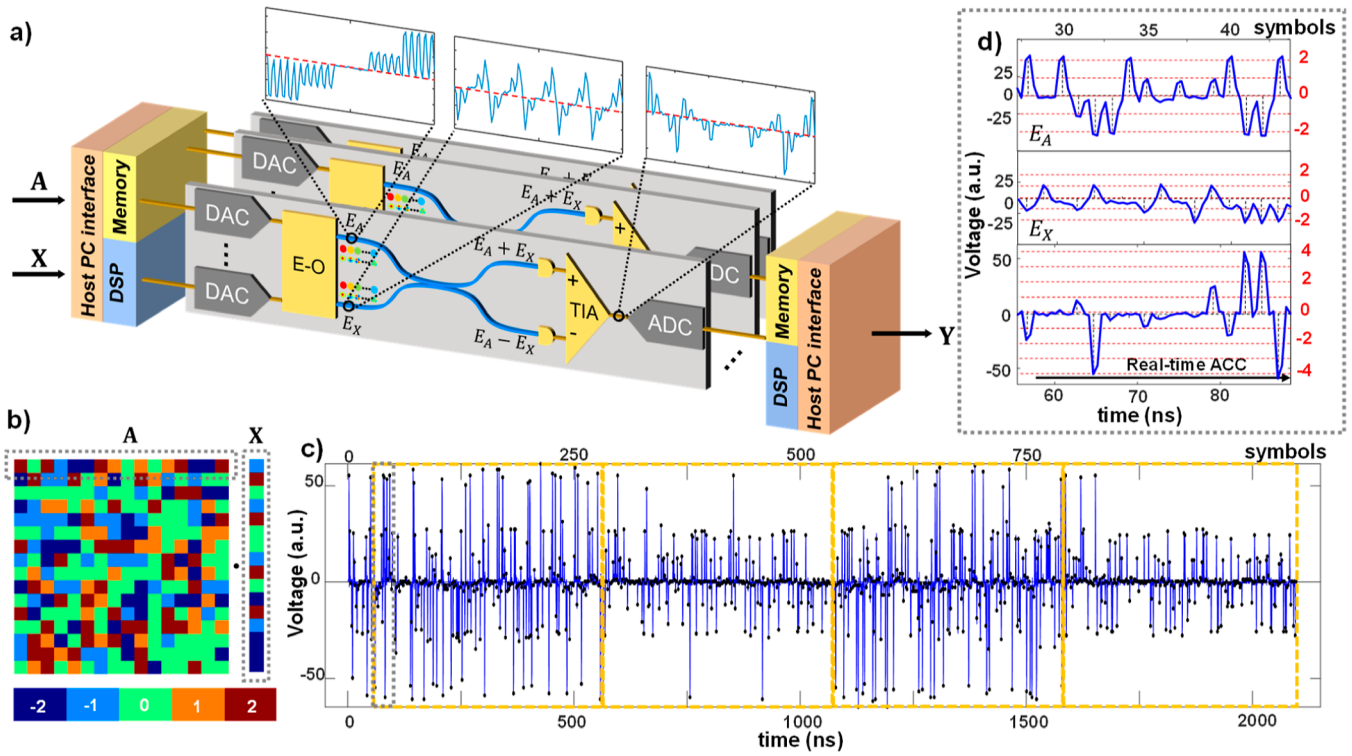


Figure 1. Schematics of the coherent photonic MPU. (a) The MPU consists of an array of photonic cores for matrix processing, along with peripheral electronics for analog/digital interfaces and real-time signal processing. Each photonic core consists of a multidimensional interference unit that calculates the dot product via coherent mixing between the multiplexed electronic fields representing two vectors. (b,c) Matrix-vector multiplications on the photonic core with peripheral digital electronics and custom firmware. For the exemplary 16 × 16 matrix-vector multiplications in (b), coherent mixing outputs from the computing packet (c) are digitized and accumulated (ACC) in real time. (d) Waveform and digitization results corresponding to the multiplication between the first row of **A** and the vector **X**.

computing, reservoir computing, and optical artificial neural networks (ANNs). Photonic neuromorphic computing mimics the neurobiological systems with photonic spiking neurons,[11,12] with the potential to achieve superior speeds and scales than its electronic counterparts.[13−15] In reservoir computing, fixed characteristic photonic reservoirs emulating complex dynamics are constructed, which can reach steady states in the subnanosecond timescale.[16,17] Optical ANN implements the topology of one or more network layers with photonic memory and interconnects. Both feed-forward[18−21] and recurrent[22−24] architectures can be implemented. Since the ANN weights do not update during inference, the power consumption of optical ANN could be lower than that of electronic processors.

Despite the promising throughput and efficiency of analog photonics, hybrid systems combining the advantages of photonics with the flexibility of electronics have yet to mature into practical general-purpose matrix accelerators.[25,26] Numerical computing applications involving linear systems typically require dynamic updates on both the input vector and the coefficient matrix. Photonic memories consume ∼2 orders of magnitude higher in power[14,27] and at least ∼1 order of magnitude longer in latency[20,28] when updating the coefficient matrix than performing passive multiplications. In addition, while photonics is suitable for performing MAC operations, the elementwise nonlinear operations, such as the activations in ANN, are more efficiently handled by conventional digital electronics.[29] Moreover, although low-precision ANN inference has shown success on analog photonic platforms,[30,31] the fixed-point precision often leads to stagnation in iterative solutions of many inverse problems in science and engineering.[32]

Representing the input and coefficient matrixes with directly modulated signals enables dynamically programmable photonic accelerators. Toward this end, works have been done to represent the matrix elements either with a cascaded secondary modulation[33] or through a coherent mixing crossbar array.[34] Nevertheless, multistage cascaded modulation introduces higher loss for the weight elements downstream, limiting the scalability. Recently, the concept of coherent photonic MAC operation with direct input matrix/vector encoding and large-scale fan-out offers a pathway toward flexible and scalable photonic accelerators,[31,35,36] yet this concept has only been applied to low-precision ANN inference scenarios. Here, we propose a photonic matrix processing unit (MPU) based on coherent multidimensional analog photonic cores for matrix-vector multiplication, digital electronics for data storage and reconfigurability, and a fixed-point linear algebra library[32] for error management. We have successfully deployed scientific computing examples on the MPU, including matrix inversion and solving partial differential equations, both of which require precisions beyond the native analog precision. Our hybrid photonic−electronic computing paradigm has the potential to bridge the gap in precision and flexibility between analogue photonics and high-performance scientific computing applications.

PRINCIPLE OF OPERATION

Coherent Photonic Processing. Figure 1 shows a schematic diagram of the proposed MPU with an array of photonic cores. Data representing an element in matrix **A** and vector **X** are modulated as analogue optical signals and sent to

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX

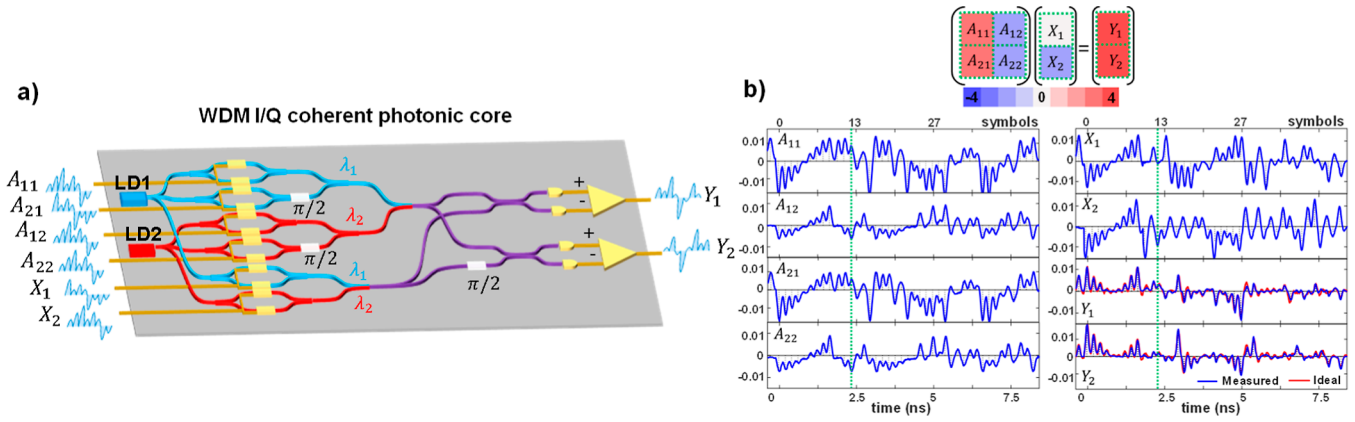


Figure 2. Multidimensional coherent photonic core for matrix processing. (a) Schematics of a photonic core supporting 2 × 2 matrix-vector multiplication using wavelength-division multiplexing (WDM) and quadratures (*I*/*Q*) to encode the matrix and vector elements. (b) 2 × 2 matrix-vector multiplications on the photonic core at the 5 GBd symbol rate. Waveforms match the ideal coherent mixing model (eq 1) with more than 7 signed effective analog levels.

the signal and local oscillator ports on the photonic core, respectively. At the heart of a single photonic core is an interference module that coherently mixes the input optical fields E_A and E_X, and outputs their product using balanced detection

$$V \propto |E_A + E_X|^2 - |E_A - E_X|^2 = 2\mathrm{Re}\{E_A^* E_X\} \quad (1)$$

Equation 1 indicates that the differential readout from coherent detection produces the signed product between the scalar fields E_A and E_X, in which the positive and negative multiplicands can be represented with 0 and 180° phases, respectively.

The photonic core interfaces with peripheral electronics for reconfigurability, storage, and communication with the host PC. Giga-samples-per-second (GSPS) analog-to-digital (A/D) and digital-to-analog (D/A) converters controlled by a field-programmable gate array (FPGA) with custom firmware bridge the analog domain of the optical signals (modulator inputs and detector outputs) and the digital domain (see Methods for details). The signal and local oscillator ports of the core, respectively, receive the computing packet representing a column vector **X** and a matrix row encoded in the time-division multiplexing (TDM) format.

Figure 1b,c shows an exemplary computing packet for four 16 × 16 matrix-vector multiplications, with the matrix and vector elements encoded in TDM. At a symbol rate of 0.5 GBd, we were able to discretize the outputs into 9 equally spaced, signed levels (integers from −4 to +4) with a symbol error rate of <0.01%, matching the dynamic range required by five signed discrete input levels (integers from −2 to 2). The matrix rows are loaded onto the signal port (E_A), while the column vector is loaded repeatedly onto the local oscillator port (E_X). Coherent mixing results are digitized and accumulated in real time every 16 symbols, as shown in Figure 1d.

Multidimensional Photonic Core. The numerous degrees of freedom available in optical signals, such as multiple wavelengths, quadratures, polarizations, spatial modes, and spatial locations, can simultaneously encode the elements in a matrix row and a vector, extending the photonic core to matrix-vector multiplications. To represent an *N*-element vector (or a matrix row vector), the input electrical fields of the signal and local oscillator ports are the superposition of *N* orthogonal modes, i.e., $E_A = \sum_{n=1}^{N} A_{mn}\,\Psi_n$ and $E_X = \sum_{n=1}^{N} X_n\,\Psi_n$. Here, Ψ_n represents the electrical field of an orthogonal base, and $\Psi_i \cdot \Psi_j = \delta_{ij}$, where δ_{ij} denotes the Kronecker delta. During the detection process, the outputs from the photonic core with multiplexed optical inputs naturally accumulate, and the result from each interferometer is the dot product between the *m*-th row vector $\mathbf{A}_m$ and vector **X**.

Figure 2a depicts an exemplary coherent photonic core that performs 2 × 2 matrix-vector multiplication **Y** = **AX** based on WDM and in-phase/quadrature (I/Q) modulation. Here, λ_1 = 1550 nm and λ_2 = 1555 nm together with the two quadratures on each wavelength encode the first $(A_{12},A_{22})^T$ and the second column $(A_{12},A_{22})^T$ of the matrix, as well as the first (X_1) and the second element (X_2) in the vector. The broadband balanced photodiodes accumulate the optical intensity encoded in two wavelengths, producing the signed dot product between the two-element matrix row and the vector. Combining the wavelength and quadratures allows 2 × 2 matrix-vector multiplication in one clock cycle. To extend to the arbitrary matrix and vector sizes, a system-level scheduler can be employed to promote parallelization among multiple cores or to process block matrix operations in serial.

We have experimentally implemented the coherent multi-dimensional photonic core running at 5 GBd. Figure 2b shows the waveforms representing matrix elements A_{11} to A_{22}, vector elements X_1 and X_2, and the output vector elements Y_1 and Y_2, respectively. The precision of the coherent multidimensional photonic core is quantified by the normalized mean-square error (NMSE), defined in eq 2, between the ideal Y_1 (red curve) and measured Y_M (blue curve) multiplication outputs

$$\mathrm{NMSE} = \frac{\frac{1}{N}\sum_{i=1}^{N} |Y_M(t_i) - Y_I(t_i)|^2}{\frac{1}{N}\sum_{i=1}^{N} |Y_I(t_i)|^2} \quad (2)$$

Here, the summation is performed over all *N*-detected symbols extracted at time stamps $\{t_i, i = 1, 2, \ldots, N\}$. The number of effective analog levels, which can be characterized by the ratio between the L2 norms of the error and ground truth, equals $1/\sqrt{\mathrm{NMSE}}$. Hence, the bit-equivalent precision is given by $-\log_2 \sqrt{NMSE}$. Based on the plots in Figure 2b, the NMSEs are 0.0166 and 0.0196, respectively, for Y_1 and Y_2, indicating that the photonic core reaches an effective precision of at least 7 signed analogue levels, which can be represented in an equivalent signed 4-bit fixed-point format.

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX

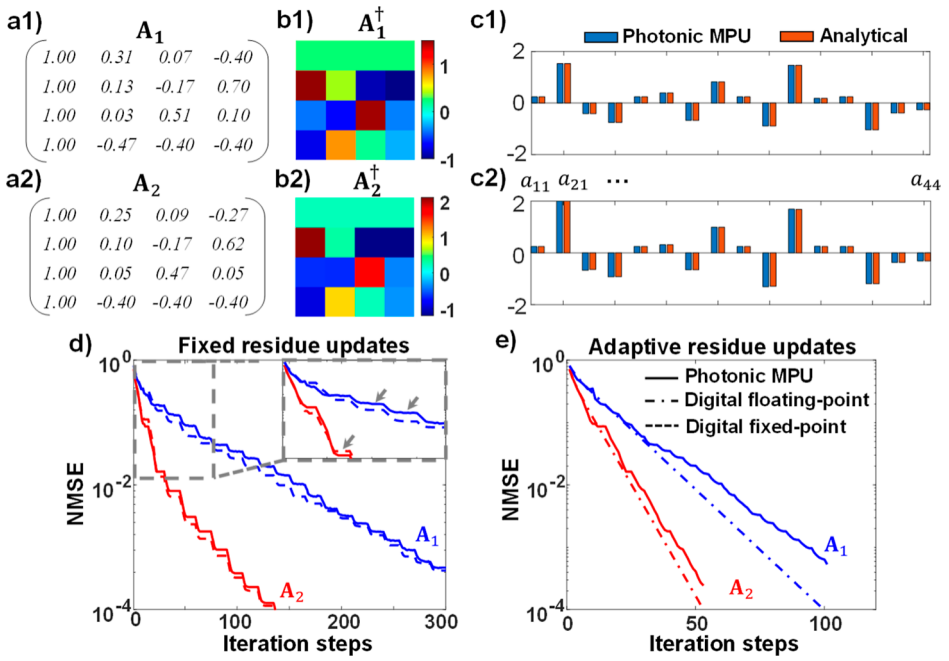


Figure 3. Numerical inversion of two 4 × 4 matrices using Richardson iterations on the photonic core. (a) Two 4 × 4 matrices to be inverted, $\mathbf{A}_1$ and $\mathbf{A}_2$, with condition numbers, κ = 11.1 and κ = 25.0, respectively. (b,c) Comparison between analytical inverses and numerical inverses on the photonic MPU. (d,e) NMSE as a function of iteration steps. Residue update is performed (d) every 15 steps and (e) adaptively, to further refine the fixed-point estimates (solid lines: photonic MPU, dashed lines: digital fixed-point simulation, dot-dashed lines: digital floating point).

Previous photonic computing implementations[18,37,38] have demonstrated sub-100MBd symbol rates. The photonic 2 × 2 matrix-vector core performs four multiplications and two additions in every received symbol. With a 5 GBd symbol rate, the overall throughput reaches 30 giga operations per second, highlighting the advantage of photonic computing in speed. Given the state-of-the-art electronic integrated circuit technology, ADC with four channels at the 56 GBd symbol rate is readily available,[39] matching the throughput of 4 × 4 photonic matrix-vector multiplication. This enables a single photonic core running at a throughput of 0.4 tera operations per second (TOPS), 1 order of magnitude higher than the state-of-the-art digital processors or accelerators.[40,41]

RESULTS

Matrix Inversion. Matrix inversion is one of the routine tasks in matrix computation. Here, we demonstrate numerical matrix inversion that iteratively solves the linear system $\mathbf{I} = \mathbf{A}\mathbf{A}^{\dagger}$ by minimizing the error $|\mathbf{I}\text{-}\mathbf{A}\mathbf{A}^{\dagger}|_F^2$, where $|\cdot|_F$ denotes the Frobenius norm. The iteration starts with a random initial guess of $\mathbf{A}_{(0)}^{\dagger}$ and refines the estimation with Richardson iterations in eq 3

$$\mathbf{A}_{(k)}^{\dagger} \leftarrow (\mathbf{I} - \tau \mathbf{A}^T\mathbf{A})\mathbf{A}_{(k-1)}^{\dagger} + \mathbf{r} \tag{3}$$

Here, $\mathbf{A}_{(k)}^{\dagger}$ is the numerical inverse at the k-th iteration, $\mathbf{A}$ (and $\mathbf{A}^{\mathrm{T}}$) is the matrix to be inverted (and its transpose), $\mathbf{I}$ is the identity matrix with the same size as $\mathbf{A}$, and $\mathbf{r} = \tau\mathbf{A}^T$ is the residue. Given a step size τ, we use the matrix $\mathbf{B} = (\mathbf{I} - \tau A^T\,\mathbf{A})$ in the iterations and deploy the fixed-point matrix−matrix operation $\mathbf{B}\mathbf{A}_{(k)}^{\dagger}$ on the photonic core.

The results of the numerical inversion of two 4 × 4 matrixes on the photonic MPU are shown in Figure 3a. Both matrices are modified from a 4 × 4 discrete cosine transform matrix with different scaling coefficients on their eigenvalues. The condition numbers κ of $\mathbf{A}_1$ and $\mathbf{A}_2$ are 11.1 and 25.0, respectively. The analytical inverses, $\mathbf{A}_1^{-1}$ and $\mathbf{A}_2^{-1}$, are calculated with floating-point LU decomposition and are considered the ground truth. The photonic MPU with TDM encoding was used for 4 × 4 matrix-vector multiplications to match the throughput of peripheral electronics. We combined two levels of precision decomposition to expand the inputs from its native range [−2, +2] to [−8, +8] in the firmware. The columns of $\mathbf{A}_k^{\dagger}$ are treated as independent vectors and cycle through the signal port (E_{X}), while the local oscillator port (E_{A}) receives the elements in $\mathbf{B}$ along the rows.

The error of the solution at step k is quantified by the NMSE between the analytical and the numerical inverses under the Frobenius norm, $|\cdot|_F$, defined in eq 4

$$\mathrm{NMSE} = \frac{|\mathbf{A}-1 - \mathbf{A}_{(k)}^{\dagger}|_F^2}{|\mathbf{A}-1|_F^2} \tag{4}$$

Figure 3d plots the NMSEs of the numerical inverses of $\mathbf{A}_1$ and $\mathbf{A}_2$ on the photonic MPU. The fixed-point matrix-vector multiplications on the photonic MPU introduce cumulative errors that stall the iterations, as shown by the flattening trend of the NMSE in the inset of Figure 3d. To overcome the stagnation, we updated the residue term $\mathbf{r}$ in the Richardson iteration with $\mathbf{r} = \mathbf{I} - \mathbf{A}\mathbf{A}_{(k)}^{\dagger}$ after 15 iterations. This is equivalent to solve $\mathbf{r} = \mathbf{A}^{\dagger}$ in a new equation $\mathbf{r} = \mathbf{A}\mathbf{A}^{\dagger}$ to correct the cumulative error. The summation of these recursive solutions to the residue problems asymptotically approaches the true inverse $\mathbf{A}^{-1}$[32] (see S4 in the Supporting Information).

The residue update can be applied adaptively when the convergence rate slows down. Figure 3e compares the error curves of photonic MPU with an adaptive residue update and that of a floating-point processor (i.e., CPU). We terminated the iterations at a maximum of 50 steps for $\mathbf{A}_2^{\dagger}$ and 100 steps for $\mathbf{A}_1^{\dagger}$, giving the NMSEs of 4.1 × 10^{-4} and 6.4 × 10^{-4}, respectively, both of which are beyond the native signed 4-bit precision of the

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX

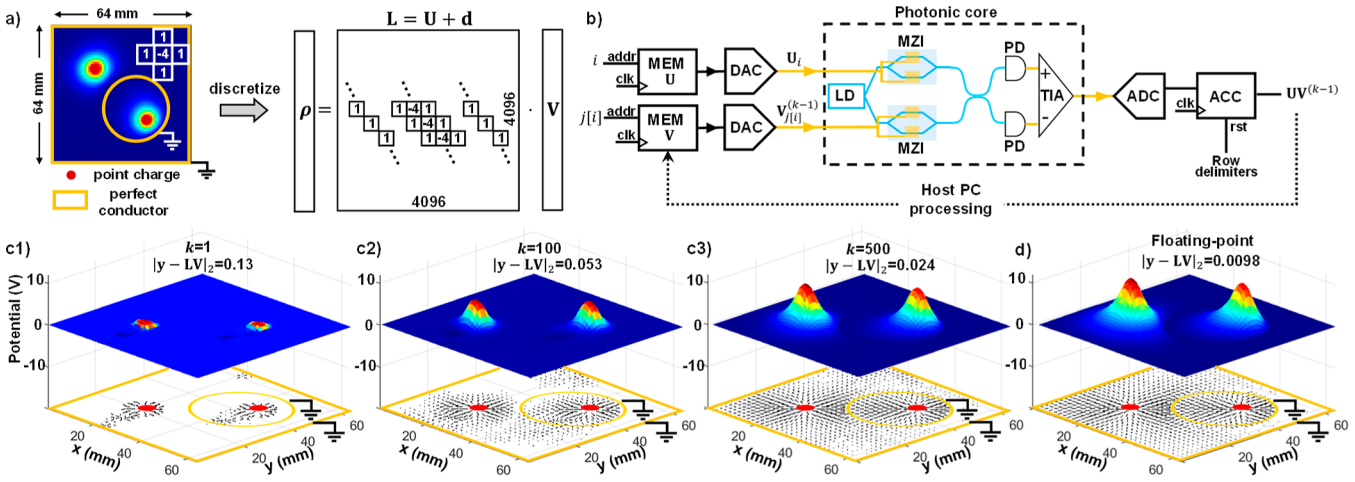


Figure 4. Numerical solution of an electrostatic problem deployed on a photonic MPU. (a) Overview of the electrostatic problem, which consists of two point charges and a grounded conducting shell, surrounded by perfect conducting boundary conditions in the solution domain. The Laplacian operator **L**, electrostatic potential **V**, and charge distribution $\boldsymbol{\rho}$ are all discretized on a finite element grid. Off-diagonal elements in **L** are represented as a sparse coefficient matrix **U** in the CSR format. (b) Analog photonic and digital electronic schemes for solving the Poisson equation with sparse matrix-vector multiplication. The nonzero elements in **U** are streamed onto the signal port of the photonic core via DAC. The indexes drive the address port of the block memory storing the dense vector **V**, thus streaming the corresponding vector elements onto the local oscillator port of the photonic core. The results are accumulated in real time and fed to the next iteration after host PC processing. All clock ports are synchronized with ADC and DAC sampling clocks. (c) Solutions from the photonic MPU after iterations 1, 100, and 500, respectively (Supporting Information Video 1). The electrical potential is depicted as the 3D surface, and the electrical field lines are plotted below. The residual error of the photonic solution, $|\mathbf{y}\text{-}\mathbf{LV}|_2$, indicates a precision better than 5.3 bits. (d) Floating-point solution for comparison.

photonic MPU. The final solutions and elementwise errors from photonic MPU are shown in Figure 3b,c. Because the ideal error decay rate of a fixed-point iterative solver is the same as that of a floating-point solver,[32] the fixed-point estimation $\mathbf{A}^{\dagger}$ can be arbitrarily close to the analytical solution with sufficient iterations.

Numerical Solution of Partial Differential Equations. Solving partial differential equations (PDEs) is a central problem in physics modeling. Numerical solutions of PDEs are typically carried out by discretizing space (and/or time) and approximating the partial derivatives as a linear superposition of neighboring grid points in a finite difference form, which amounts to solving a sparse linear problem. In this example, we demonstrate the solution of the electrostatic equation in free space, given a predefined charge density.

The electrostatic potential, V, induced from a given static charge distribution, ρ, in free space (vacuum permittivity ε_0) is described by Gauss's Law, which is expressed as eq 5 in the differential form

$$\nabla^2 V = -\frac{\rho}{\varepsilon_0} \tag{5}$$

Here, both potential V and the charge density ρ are discretized onto a uniform 2D grid. The discrete Laplacian operator is expressed by five stencils around each grid point, and the finite-difference coefficients are arranged into a sparse matrix **L**, as shown in Figure 4a. The domain of the electrostatic equation comprises a 64×64 mesh with a uniform grid size of 1 mm. The finite-difference coefficient matrix **L** contains 4096×4096 elements, among which 19162 elements are nonzero, giving a sparsity of 0.11%. The charge density ρ consists of two unitary point sources centered at (20 and 20 mm) and (44 and 44 mm), respectively. A grounded perfectly conducting shell with a radius of 16 mm centered at (40 mm and 40 mm) is also introduced into the solution domain. Dirichlet boundary condition with $V = 0$ is applied to all four boundaries of the solution domain, which is equivalent to setting the boundaries to grounded conductors. The electrostatic problem is thus translated into a system of linear equations $\mathbf{LV} = \mathbf{y}$, where **V** and $\mathbf{y} = -\rho/\varepsilon_0$ represent the vectorized potential and charge density, respectively, after discretization.

The linear system of equation $\mathbf{LV} = \mathbf{y}$ is iteratively solved with the weighted Jacobi method due to the highly sparse and strictly diagonal coefficient matrix **L**.[42] From an all-zero vector $\mathbf{V}^{(0)}$, each iteration k updates the solution according to eq 6

$$\mathbf{V}^{(k)} \leftarrow \omega \mathbf{d}^{-1} \odot (\mathbf{r} - \mathbf{U}\mathbf{V}^{(k-1)}) + (1 - \omega)\mathbf{V}^{(k-1)} \tag{6}$$

Here, vector **d** stores the diagonal elements in matrix **L**, $\mathbf{U} = \mathbf{L}\text{-}\mathrm{diag}(\mathbf{d})$ is a sparse matrix containing only the off-diagonal elements in **L**, and $\odot$ denotes the elementwise product. A weight $\omega = 0.9$ was chosen for enhanced stability under the presence of computational errors. The multiplications between **U** and $\mathbf{V}^{(k-1)}$ are deployed on the photonic MPU, and the elemental multiplications, additions, and solution updates are carried out on a digital computer. We incorporate the residual corrections in the Jacobi method (see S4 in the Supporting Information) and update the residue term **r** with the cumulative error $\mathbf{r} = \mathbf{y}\text{-}\mathbf{LV}^{(k)}$ every five iterations.

For the numerical solution of PDEs, the photonic MPU with TDM encoding is loaded with sparse matrix processing firmware supporting 256 nonzero elements per computing packet with signed 3-bit precision. Figure 4b illustrates the principle of spare-dense matrix-vector multiplication on the photonic MPU. The matrix **U** is represented in the compressed sparse row (CSR) format, which stores the nonzero elements, $\mathbf{U}_i$, along with their indexes, $j[i]$, in each row. The entirety of dense vector $\mathbf{V}^{(k-1)}$ is stored in the FPGA block memory. The nonzero elements, $\mathbf{U}_i$, are streamed in real time onto the signal port of the photonic core via DAC. The indexes $j[i]$ drive the address port of the block memory to load the corresponding elements, $\mathbf{V}_{j[i]}^{(k-1)}$, onto the local oscillator port of the photonic core via another DAC channel. After all nonzero elements in each matrix row are

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX

depleted, a delimiting pulse resets the accumulator and outputs one element in the result vector. Figure 4c plots the electrical potential computed from the photonic MPU at iteration steps k = 1, 100, and 500, respectively. The photonic MPU reaches a residual error, $|\mathbf{y}\text{-}\mathbf{LV}|_2$, of 0.024, achieving an indistinguishable solution from the floating-point iterative solver (Figure 4d, see Supporting Information Video 1 for the comparison between the photonic MPU and floating-point solutions at each iteration).

For numerical solution of PDEs, converting the sparse finite-difference coefficient matrix to dense and adopting systolic tensor core structures[43] would be extremely inefficient for large grids. Since the number of nonzero elements in each row of the finite difference matrix $\mathbf{U}$ is always four (except for the rows on the domain boundary) when using the fie-stencil Laplacian operator, we can expand the multidimensional photonic core with four wavelengths to encode each row of $\mathbf{U}$. Meanwhile, multiple rows can be processed synchronously by using parallel photonic cores in the MPU. The vector elements can be fanned out with parallel DAC channels driven from a common FPGA block memory. Thus, parallel multidimensional photonic cores with digital memory can potentially scale to large sparse finite-difference matrices, extending the applicability of the photonic MPU to scientific computing.

CONCLUSIONS AND DISCUSSION

We have demonstrated a flexible photonic matrix processing unit based on a coherent, multidimensional photonic core that supports arbitrary, signed multiplications. Using digital electronics for precision control, memory, and reconfiguration, our photonic platform can be tailored to a variety of numerical computing tasks. Combined with residual iterations for error management, we have shown that our photonic MPU can perform several numerical computing tasks beyond its native hardware precision limit, including matrix inversion, feedback control, and solving partial differential equations, reaching solutions with errors comparable to those on a floating-point processor. It is worth noting that the residual iterations converge at the same rate as the floating-point gradient-descent iterations.[32] The computing overhead from the residue adjustment step is thus inversely proportional to the number of iterations between the two adjustments.

Based on the latest silicon photonics foundry process, coherent photonic MPU supporting 16 × 16 matrix-vector multiplication is available with the industry standard on-chip modulator, photodiode, and interferometer footprints[44] (see S2 in the Supporting Information). The MPU can be practically scaled up by using dense wavelength-division multiplexing (DWDM) and large-scale parallel fan-outs. Because the multiplication and accumulation in photonic cores are passive, the only energy consumption arises from the electrical power in driving the lasers, modulators, ADCs, and coherent receivers as well as the memory access for the matrix/vector elements. For $N \times N$ matrix-vector multiplications, the overall energy efficiency asymptotically approaches a floor of ∼10fJ/MAC for matrix size N on the order of hundreds (see S3 in the Supporting Information), suggesting a 10^1−10^2-fold higher efficiency than electronic accelerators (∼pJ/MAC).[45] For $N \times N$ matrix−matrix multiplications using fan-outs, the power consumptions of lasers, modulators, ADCs, and coherent receivers scale with N^2, while the number of MACs is N^3. Hence, the overall energy consumption per MAC is proportional to $1/N$, which is a strong indication that coherent photonic MAC operations can surpass digital electronic counterparts by orders of magnitude on large scales. We envision an array of photonic MPU nodes with dedicated DSP units to form the backbone of high-speed, energy-efficient computing in data center applications.

METHODS

Coherent Photonic Processing Core. The photonic core consisted of two zero-chirp Mach−Zehnder modulators (MZMs, JDSU IOAP-MOD9140) as the vector elements and two I/Q modulators (Sumitomo Osaka Cement Co., Ltd., T.SBX1.5-10-S-FK) as the matrix elements. Each modulator port was driven by an amplifier, JDSU H301. A total of four 2 × 2 fiber couplers (Fiber Store PLC-202-ST) were used as interferometers and for wavelength mixing in WDM. The balanced photodiodes for coherent detection were Thorlabs BDX1BA mounted on the BDX1EVB evaluation board. The input lasers were 1550 nm tunable sources (Santec TSL-210) with a maximum output power of 8 mW amplified by erbium-doped fiber amplifiers (EDFAs, Amonics AEDF-18-B-FA). We designated 0 dBm (1 mW) average power on the photodiode to ensure the signed 8-bit SNR of the readout. The maximum optical output of the Amonics EDFA was 18 dBm at 1550 nm, which provided sufficient optical power to drive all MZMs in the core.

The MZMs were controlled by DACs to generate the symbol sequences representing the elements in the vector and matrix rows. The DAC output voltages were skewed to compensate for the nonlinear distortion of each MZM, yielding uniform spacing between adjacent analogue input (and output) levels. The photocurrent from the balanced detector pair was converted into an analogue voltage signal via a transimpedance amplifier (TIA) and then digitized with an A/D converter for digital signal processing (DSP). The DSP routines on the FPGA extracted the digitized analogue symbols, accumulated them in real time, and translated them into the multiplication results for streaming back to the host PC.

Electronic Interface with the Photonic MPU. The photonic core was complemented with a firmware and software suite supporting both dense and sparse matrix-vector multiplications. The digital electronics for PCIe communication, block memory, and signal processing routines, including symbol extraction, accumulation, and compensation of modulator nonlinearity, were embedded in the customized firmware in the Xilinx VC707 FPGA evaluation board, which was connected to the PCIe expansion slot of the host PC. A dual-channel D/A converter daughter card (Euvis FMC2662) was attached to the FMC port of the FPGA evaluation board to drive the signal and local oscillator ports of the photonic core. The A/D converter (GaGe CobraMax) was connected to another PCIe expansion slot, and the samples from the A/D converter were directly streamed into the FPGA for processing. The digitized samples were delimited by the FPGA and accumulated in real time to produce the elements in the result vector $\mathbf{Y}$, which were streamed back to a host PC. The communications between the host PC, FPGA evaluation board, and A/D converter were implemented through PCIe Gen2 × 8, which supports a continuous stream throughput of 2GSa/s for 8-bit samples.

For the multidimensional photonic core running at 5 GBd, the input and output ports of the coherent multidimensional core were connected to an arbitrary wave generator (AWG, Keysight M8195A) for signal generation and an oscilloscope (Keysight Infiniium 95004Q) for signal detection. The linearized inputs for each analog level were transferred onto the AWG memory.

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX

The waveform corresponding to the computing packet was then synthesized and loaded onto the modulator ports. The recorded waveforms were downloaded onto a host PC for symbol extraction and post-processing.

Software Backend for the Photonic MPU. The software consists of a low-level backend and a high-level application interface for implementing various numerical computing algorithms in the experiments. The backend sends the precompensated multiplication symbols and retrieves the accumulated multiplication results from the FPGA block memory (see Section S1 of the Supporting Information). In the case that high-precision inputs beyond signed 3 levels are necessary, the backend automatically decomposes the matrix $\mathbf{A}$ and vector $\mathbf{X}$ with base β (a configurable parameter) as $\mathbf{A} = \beta\,\mathbf{A}_H + \mathbf{A}_L$ and $\mathbf{X} = \beta\,\mathbf{X}_H + \mathbf{X}_L$, where $\mathbf{A}_H$, $\mathbf{A}_L$, $\mathbf{X}_H$, and $\mathbf{X}_L$ all have a range of −2 to +2, and $\beta = 3$ is the base matching the calibrated quantization levels on the photonic MPU. The decomposed matrices and vectors were loaded onto the photonic core by alternating the two range levels. This decomposition gave rise to four partial results, $\mathbf{A}_H\mathbf{X}_H$, $\mathbf{A}_H\mathbf{X}_L$, $\mathbf{A}_L\mathbf{X}_H$, and $\mathbf{A}_L\mathbf{X}_L$, which were scaled by β^2, β, β, and 1, respectively, before digital summation.

To interface with high-level applications, the low-level backend provided a packaged C library including dense and sparse *matmul*() functions. Matrix inversion and PDE solvers were implemented in Python by calling these backend functions. Details of the high-level algorithms are provided in Sections S3 and S4 of the Supporting Information.

■ ASSOCIATED CONTENT

Data Availability Statement

The data supporting the findings in this study are available in the manuscript and its Supporting Information. Raw data sets are available from the corresponding author upon reasonable request.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsphotonics.3c01694.

Comparison of the photonic MPU and floating-point solutions at each intermediate iteration of the Jacobi method for solving PDE (MP4)

Construction and calibration of the photonic core, power consumption, scalability of the MPU, and fixed-point Richardson and Jacobi method with residue correction for solving PDEs (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Zheyuan Zhu − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States;* orcid.org/0000-0001-9992-135X; Email: zheyuan.zhu@ucf.edu

Guifang Li − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States*; Email: li@ucf.edu

Shuo S. Pang − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States*; Email: pang@ucf.edu

Authors

Alireza Fardoost − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States*

Fatemeh Ghaedi Vanani − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States*

Andrew B. Klein − *CREOL, The College of Optics and Photonics, University of Central Florida, Orlando, Florida 32816-2700, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsphotonics.3c01694

Funding

This work was supported by the Office of Naval Research (N00014-20-1-2441), National Science Foundation (1932858), and Army Research Office (W911NF2110321).

Notes

The authors declare no competing financial interest.

■ REFERENCES

(1) Horowitz, M. 1.1 Computing's Energy Problem (and What We Can Do about It). In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*; IEEE, 2014; Vol *50*, pp 10−14..

(2) Theis, T. N.; Wong, H.-S. P. The End of Moore's Law: A New Beginning for Information Technology. *Comput. Sci. Eng.* **2017**, *19* (2), 41−50.

(3) Williams, R. S. What's Next? [The End of Moore's Law]. *Comput. Sci. Eng.* **2017**, *19* (2), 7−13.

(4) Miller, D. A. B. Attojoule Optoelectronics for Low-Energy Information Processing and Communications. *J. Lightwave Technol.* **2017**, *35* (3), 346−396.

(5) Shastri, B. J.; Tait, A. N.; Ferreira de Lima, T.; Pernice, W. H. P.; Bhaskaran, H.; Wright, C. D.; Prucnal, P. R. Photonics for Artificial Intelligence and Neuromorphic Computing. *Nat. Photonics* **2021**, *15* (2), 102−114.

(6) de Lima, T. F.; Peng, H.-T.; Tait, A. N.; Nahmias, M. A.; Miller, H. B.; Shastri, B. J.; Prucnal, P. R. Machine Learning With Neuromorphic Photonics. *J. Lightwave Technol.* **2019**, *37* (5), 1515−1534.

(7) Li, C.; Belkin, D.; Li, Y.; Yan, P.; Hu, M.; Ge, N.; Jiang, H.; Montgomery, E.; Lin, P.; Wang, Z.; Song, W.; Strachan, J. P.; Barnell, M.; Wu, Q.; Williams, R. S.; Yang, J. J.; Xia, Q. Efficient and Self-Adaptive in-Situ Learning in Multilayer Memristor Neural Networks. *Nat. Commun.* **2018**, *9* (1), 2385.

(8) Ielmini, D.; Wong, H.-S. P. In-Memory Computing with Resistive Switching Devices. *Nat. Electron.* **2018**, *1* (6), 333−343.

(9) Li, C.; Hu, M.; Li, Y.; Jiang, H.; Ge, N.; Montgomery, E.; Zhang, J.; Song, W.; Dávila, N.; Graves, C. E.; Li, Z.; Strachan, J. P.; Lin, P.; Wang, Z.; Barnell, M.; Wu, Q.; Williams, R. S.; Yang, J. J.; Xia, Q. Analogue Signal and Image Processing with Large Memristor Crossbars. *Nat. Electron.* **2017**, *1* (1), 52−59.

(10) Fardoost, A.; Vanani, F. G.; Zhu, Z.; Doerr, C.; Pang, S.; Li, G. A High-Speed Photonic Tensor Accelerator. In *2022 IEEE Photonics Conference (IPC)*; IEEE, 2022; pp 1−2..

(11) Ferreira de Lima, T.; Shastri, B. J.; Tait, A. N.; Nahmias, M. A.; Prucnal, P. R. Progress in Neuromorphic Photonics. *Nanophotonics* **2017**, *6* (3), 577−599.

(12) Peng, H.-T.; Nahmias, M. A.; de Lima, T. F.; Tait, A. N.; Shastri, B. J. Neuromorphic Photonic Integrated Circuits. *IEEE J. Sel. Top. Quantum Electron.* **2018**, *24* (6), 1−15.

(13) Chakraborty, I.; Saha, G.; Sengupta, A.; Roy, K. Toward Fast Neural Computing Using All-Photonic Phase Change Spiking Neurons. *Sci. Rep.* **2018**, *8* (1), 12980.

(14) Chakraborty, I.; Saha, G.; Roy, K. Photonic In-Memory Computing Primitive for Spiking Neural Networks Using Phase-Change Materials. *Phys. Rev. Appl.* **2019**, *11* (1), 014063.

(15) Feldmann, J.; Youngblood, N.; Wright, C. D.; Bhaskaran, H.; Pernice, W. H. P. All-Optical Spiking Neurosynaptic Networks with Self-Learning Capabilities. *Nature* **2019**, *569* (7755), 208−214.

(16) Van der Sande, G.; Brunner, D.; Soriano, M. C. Advances in Photonic Reservoir Computing. *Nanophotonics* **2017**, *6* (3), 561−576.
(17) Brunner, D.; Soriano, M. C.; Mirasso, C. R.; Fischer, I. Parallel Photonic Information Processing at Gigabyte per Second Data Rates Using Transient States. *Nat. Commun.* **2013**, *4* (1), 1364.
(18) Shen, Y.; Harris, N. C.; Skirlo, S.; Prabhu, M.; Baehr-Jones, T.; Hochberg, M.; Sun, X.; Zhao, S.; Larochelle, H.; Englund, D.; Soljačić, M. Deep Learning with Coherent Nanophotonic Circuits. *Nat. Photonics* **2017**, *11* (7), 441−446.
(19) Bangari, V.; Marquez, B. A.; Miller, H.; Tait, A. N.; Nahmias, M. A.; de Lima, T. F.; Peng, H.-T.; Prucnal, P. R.; Shastri, B. J. Digital Electronics and Analog Photonics for Convolutional Neural Networks (DEAP-CNNs). *IEEE J. Sel. Top. Quantum Electron.* **2020**, *26* (1), 1−13.
(20) Feldmann, J.; Youngblood, N.; Karpov, M.; Gehring, H.; Li, X.; Stappers, M.; Le Gallo, M.; Fu, X.; Lukashchuk, A.; Raja, A. S.; Liu, J.; Wright, C. D.; Sebastian, A.; Kippenberg, T. J.; Pernice, W. H. P.; Bhaskaran, H. Parallel Convolutional Processing Using an Integrated Photonic Tensor Core. *Nature* **2021**, *589* (7840), 52−58.
(21) Lin, X.; Rivenson, Y.; Yardimci, N. T.; Veli, M.; Luo, Y.; Jarrahi, M.; Ozcan, A. All-Optical Machine Learning Using Diffractive Deep Neural Networks. *Science* **2018**, *361* (6406), 1004−1008.
(22) Tait, A. N.; de Lima, T. F.; Zhou, E.; Wu, A. X.; Nahmias, M. A.; Shastri, B. J.; Prucnal, P. R. Neuromorphic Photonic Networks Using Silicon Photonic Weight Banks. *Sci. Rep.* **2017**, *7* (1), 7430.
(23) Mohammadi Estakhri, N.; Edwards, B.; Engheta, N. Inverse-Designed Metastructures That Solve Equations. *Science* **2019**, *363* (6433), 1333−1338.
(24) Tait, A. N.; Ferreira de Lima, T.; Nahmias, M. A.; Miller, H. B.; Peng, H.-T.; Shastri, B. J.; Prucnal, P. R. Silicon Photonic Modulator Neuron. *Phys. Rev. Appl.* **2019**, *11* (6), 064043.
(25) Wetzstein, G.; Ozcan, A.; Gigan, S.; Fan, S.; Englund, D.; Soljačić, M.; Denz, C.; Miller, D. A. B.; Psaltis, D. Inference in Artificial Intelligence with Deep Optics and Photonics. *Nature* **2020**, *588* (7836), 39−47.
(26) Zhou, H.; Dong, J.; Cheng, J.; Dong, W.; Huang, C.; Shen, Y.; Zhang, Q.; Gu, M.; Qian, C.; Chen, H.; Ruan, Z.; Zhang, X. Photonic Matrix Multiplication Lights up Photonic Accelerator and Beyond. *Light: Sci. Appl.* **2022**, *11* (1), 30.
(27) Ríos, C.; Youngblood, N.; Cheng, Z.; Le Gallo, M.; Pernice, W. H. P.; Wright, C. D.; Sebastian, A.; Bhaskaran, H. In-Memory Computing on a Photonic Platform. *Sci. Adv.* **2019**, *5* (2), No. eaau5759.
(28) Rios, C.; Stegmaier, M.; Hosseini, P.; Wang, D.; Scherer, T.; Wright, C. D.; Bhaskaran, H.; Pernice, W. H. P. Integrated All-Photonic Non-Volatile Multi-Level Memory. *Nat. Photonics* **2015**, *9* (11), 725−732.
(29) Peserico, N.; Shastri, B. J.; Sorger, V. J. Integrated Photonic Tensor Processing Unit for a Matrix Multiply: A Review. *J. Lightwave Technol.* **2023**, *41* (12), 3704−3716.
(30) Xu, X.; Tan, M.; Corcoran, B.; Wu, J.; Boes, A.; Nguyen, T. G.; Chu, S. T.; Little, B. E.; Hicks, D. G.; Morandotti, R.; Mitchell, A.; Moss, D. J. 11 TOPS Photonic Convolutional Accelerator for Optical Neural Networks. *Nature* **2021**, *589* (7840), 44−51.
(31) Hamerly, R.; Bernstein, L.; Sludds, A.; Soljačić, M.; Englund, D. Large-Scale Optical Neural Networks Based on Photoelectric Multiplication. *Phys. Rev. X* **2019**, *9* (2), 021032.
(32) Zhu, Z.; Klein, A. B.; Li, G.; Pang, S. Fixed-Point Iterative Linear Inverse Solver with Extended Precision. *Sci. Rep.* **2023**, *13* (1), 5198.
(33) Giamougiannis, G.; Tsakyridis, A.; Moralis-Pegios, M.; Pappas, C.; Kirtas, M.; Passalis, N.; Lazovsky, D.; Tefas, A.; Pleros, N. Analog Nanophotonic Computing Going Practical: Silicon Photonic Deep Learning Engines for Tiled Optical Matrix Multiplication with Dynamic Precision. *Nanophotonics* **2023**, *12* (5), 963−973.
(34) Youngblood, N. Coherent Photonic Crossbar Arrays for Large-Scale Matrix-Matrix Multiplication. *IEEE J. Sel. Top. Quantum Electron.* **2023**, *29*, 1−11.
(35) Sludds, A.; Bernstein, L.; Hamerly, R.; Soljacic, M.; Englund, D. R. A Scalable Optical Neural Network Architecture Using Coherent Detection. In *AI and Optical Data Sciences*; Kitayama, K., Jalali, B., Eds.; SPIE, 2020; Vol. *11299*, p 16..
(36) Chen, Z.; Sludds, A.; Davis, R.; Christen, I.; Bernstein, L.; Ateshian, L.; Heuser, T.; Heermeier, N.; Lott, J. A.; Reitzenstein, S.; Hamerly, R.; Englund, D. Deep Learning with Coherent VCSEL Neural Networks. *Nat. Photonics* **2023**, *17* (8), 723−730.
(37) Huang, C.; Fujisawa, S.; de Lima, T. F.; Tait, A. N.; Blow, E. C.; Tian, Y.; Bilodeau, S.; Jha, A.; Yaman, F.; Peng, H.-T.; Batshon, H. G.; Shastri, B. J.; Inada, Y.; Wang, T.; Prucnal, P. R. A Silicon Photonic-Electronic Neural Network for Fibre Nonlinearity Compensation. *Nat. Electron.* **2021**, *4* (11), 837−844.
(38) Roques-Carmes, C.; Shen, Y.; Zanoci, C.; Prabhu, M.; Atieh, F.; Jing, L.; Dubček, T.; Mao, C.; Johnson, M. R.; Čeperić, V.; Joannopoulos, J. D.; Englund, D.; Soljačić, M. Heuristic Recurrent Algorithms for Photonic Ising Machines. *Nat. Commun.* **2020**, *11* (1), 249.
(39) Sun, K. A 56-GS/s 8-Bit Time-Interleaved SAR ADC in 28-Nm CMOS. Ph.D. Thesis, Southern Methodist University, 2018. https://scholar.smu.edu/engineering_electrical_etds/12/.
(40) Srivastava, N.; Jin, H.; Liu, J.; Albonesi, D.; Zhang, Z. MatRaptor: A Sparse-Sparse Matrix Multiplication Accelerator Based on Row-Wise Product. In *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*; IEEE, 2020; pp 766−780..
(41) Keckler, S. W.; Dally, W. J.; Khailany, B.; Garland, M.; Glasco, D. GPUs and the Future of Parallel Computing. *IEEE Micro* **2011**, *31* (5), 7−17.
(42) Saad, Y. *Iterative Methods for Sparse Linear Systems*; SIAM, 2003.
(43) Miscuglio, M.; Sorger, V. J. Photonic Tensor Cores for Machine Learning. *Appl. Phys. Rev.* **2020**, *7* (3), 031404.
(44) Bogaerts, W.; Pérez, D.; Capmany, J.; Miller, D. A. B.; Poon, J.; Englund, D.; Morichetti, F.; Melloni, A. Programmable Photonic Circuits. *Nature* **2020**, *586* (7828), 207−216.
(45) Jouppi, N. P.; Borchers, A.; Boyle, R.; Cantin, P.; Chao, C.; Clark, C.; Coriell, J.; Daley, M.; Dau, M.; Dean, J.; Gelb, B.; Young, C.; Ghaemmaghami, T. V.; Gottipati, R.; Gulland, W.; Hagmann, R.; Ho, C. R.; Hogberg, D.; Hu, J.; Hundt, R.; Hurt, D.; Ibarz, J.; Patil, N.; Jaffey, A.; Jaworski, A.; Kaplan, A.; Khaitan, H.; Killebrew, D.; Koch, A.; Kumar, N.; Lacy, S.; Laudon, J.; Law, J.; Patterson, D.; Le, D.; Leary, C.; Liu, Z.; Lucke, K.; Lundin, A.; MacKean, G.; Maggiore, A.; Mahony, M.; Miller, K.; Nagarajan, R.; Agrawal, G.; Narayanaswami, R.; Ni, R.; Nix, K.; Norrie, T.; Omernick, M.; Penukonda, N.; Phelps, A.; Ross, J.; Ross, M.; Salek, A.; Bajwa, R.; Samadiani, E.; Severn, C.; Sizikov, G.; Snelham, M.; Souter, J.; Steinberg, D.; Swing, A.; Tan, M.; Thorson, G.; Tian, B.; Bates, S.; Toma, H.; Tuttle, E.; Vasudevan, V.; Walter, R.; Wang, W.; Wilcox, E.; Yoon, D. H.; Bhatia, S.; Boden, N. In-Datacenter Performance Analysis of a Tensor Processing Unit. In *Proceedings of the 44th Annual International Symposium on Computer Architecture - ISCA '17*; ACM Press: New York, New York, USA, 2017; pp 1−12..

https://doi.org/10.1021/acsphotonics.3c01694
ACS Photonics XXXX, XXX, XXX−XXX