Photonic-computing error correction through optical encoder and decoder calibrations

Adam Carstensen[*] and Babak Vosoughi Lahijani[†]
Department of Electrical and Photonics Engineering, Technical University of Denmark, Ørsteds Plads 343, DK-2800 Kgs. Lyngby, Denmark

(Received 23 February 2026; revised 10 July 2026; accepted 30 September 2026; published 20 November 2026)

Photonic processors have emerged as an attractive platform for fast and energy-efficient matrix-vector multiplication. However, they are susceptible to error due to their analog nature. Here, we present an error-correction technique that implements a correction offset to the optical encoders and decoders of photonic processors. Our proposed method is general-purpose, does not require introducing any additional components to the photonic network, and can address errors stemming from unbalanced losses, 50/50 beam-splitter deviations, digital-to-analog conversion inaccuracies, and any unknown sources. In particular, we show that our method is highly effective in mitigating unbalanced-loss errors, an error source that has not been a primary focus of previous error-correction techniques in photonic computing. Using this approach, we achieve over 90% error reduction in large triangular meshes, overcoming a key obstacle to highly accurate photonic processors for information processing.

DOI: 10.1103/71gp-fpwf

I. INTRODUCTION

Digital electronic processors have reached a plateau in computing efficiency, with only marginal improvements over the past decade [1,2]. At the same time, the demand for computation has increased dramatically, driven in large part by artificial intelligence (AI) workloads [3]. A large fraction of these workloads, such as neural network training and inference, are dominated by linear algebra operations, in particular, matrix-vector multiplications (MVMs) [4–6]. Beyond AI, linear operations are central to computing and form the basis for a wide range of applications, such as signal processing, communications, scientific simulations, and optimization. Therefore, fast and efficient MVM is a key objective for next-generation information-processing systems. Analog photonic computing offers a promising path toward this goal, as photonic devices can perform linear operations inherently and in parallel [7]. In particular, photonic matrix-vector multiplication processors (PMVMPs) can achieve computing speeds and energy efficiencies beyond what is possible with digital electronics [8–12]. PMVMPs have been demonstrated using various approaches, such as cross-bar architectures [13], spatial light modulators [14,15], unitary optical processors [16,17], wavelength multiplexing [18], and integrated Mach–Zehnder interferometers (MZI) [19–22]. Each of these methods has its own challenges to overcome in order to achieve widespread adoption [10,23]. However, a common requirement for all photonic computing approaches is that computing errors must remain low enough to avoid significant performance degradation [24,25], an issue that is central to analog computing.

Among various photonic computing approaches, integrated meshes of MZIs have received particular attention due to their scalable nature [26] and universal programmability [27–33]. Despite these strengths, MZI-based photonic processors suffer from nonidealities such as unbalanced losses [27,31], digital-to-analog conversion (DAC) inaccuracies [27], 50/50 beam-splitter (BS) deviations [24,27,28,34–37], unwanted phase shifts [27], and reflections. Integrated MZI meshes are typically implemented using triangular [30] or rectangular [31] meshes. Although the rectangular mesh offers improved robustness against unbalanced losses, it is not immune to unbalanced-loss errors, limiting its performance in applications that require high computational accuracy. To reduce the errors arising from variations in the 50/50 BSs, many other mesh architectures have been proposed [34–36,38–42], often at the cost of higher insertion losses. In addition, algorithmic approaches such as self-configuration [27,28], hardware error correction [43], transferable learning [25], and *in situ* backpropagation [44,45] have likewise been proposed to alleviate the 50/50 BS errors. These methods are tailored to specific errors and lack the flexibility to extend to other types of errors. Furthermore, no error-correction method has yet been proposed for errors stemming from

[*]Contact author: adamc@dtu.dk
[†]Contact author: bala@dtu.dk

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

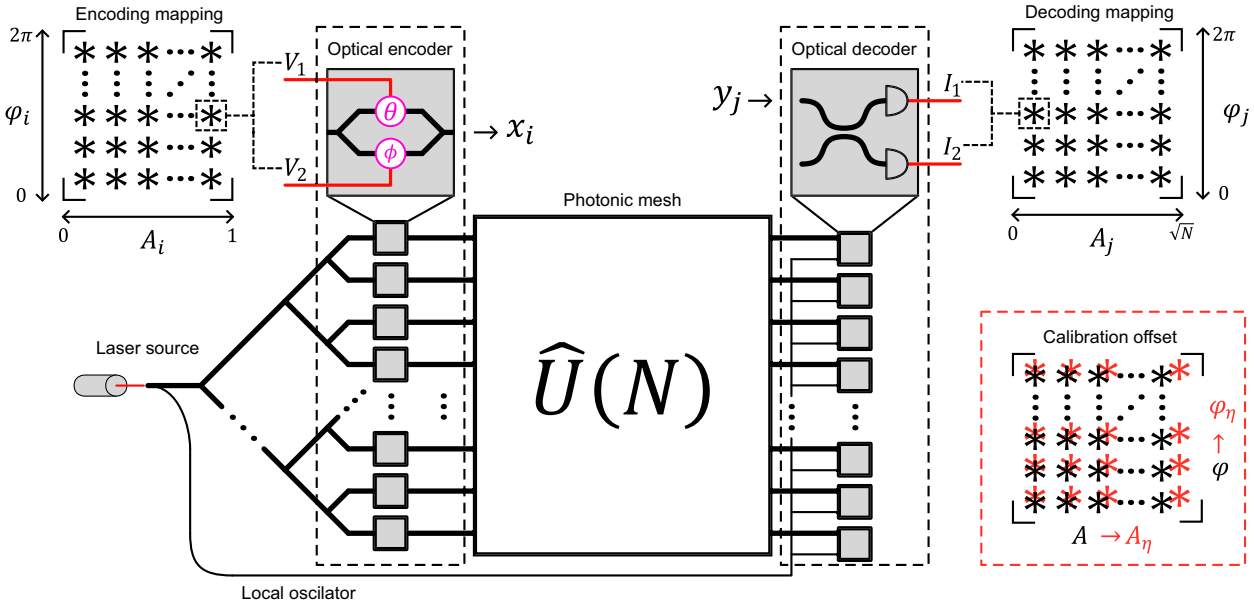


FIG. 1. Schematic representation of a photonic matrix-vector multiplication processor consisting of an optical splitting tree that distributes the input laser source to an optical encoder, a photonic mesh, and an optical decoder. Vector data is encoded from digital information onto optical fields defined by the encoder mapping. The output optical vector is converted back into digital data defined by the decoder mapping. The encoder and decoder mappings are shown here as black "∗" representing voltage and current pairs as discrete points for simplicity. Scaling these mappings along the amplitude axis and shifting them along the phase axis corresponds to applying a complex factor η to each input and output, providing $2N$ complex degrees of freedom (DOF) for error correction.

unbalanced losses and unwanted reflections, which can significantly degrade the performance of PMVMPs.

To conduct MVM in the optical domain, the vector data, x_i, stored in the electrical domain, must be encoded into the amplitudes, A_i, and phases, φ_i, of the ingoing optical fields. MVM occurs as optical fields propagate and interfere throughout the photonic mesh, eventually reaching the output. The output optical fields, y_j, described by their amplitudes, A_j, and phases, φ_j, represent the MVM on the input vector by the photonic mesh and must be detected and converted back to the electrical domain for digital storage. This is done through decoders that can detect the optical signals and recover their amplitude and phase information. An illustration of such a PMVMP is shown in Fig. 1. In this work, we introduce an error-correction method using optical encoders and decoders that are intrinsic to PMVMPs and demonstrate how it diminishes computational errors across various error sources and for different photonic mesh structures.

II. ENCODING MAPPINGS

Encoding can be achieved with various devices, such as an optical setup machine [19,33,45,46], off-chip modulators [47], amplitude-only modulators [13,48,49], directional-coupler-based complex encoders [50], a series of tunable couplers with phase shifters [51], or, as illustrated in Fig. 1, a series of Mach–Zehnder modulators (MZMs), which are widely employed in high-speed photonic computing [11,52]. Regardless of the encoding method, correct encoding requires device calibration and, consequently, a function that maps the desired optical amplitudes and phases to corresponding encoder voltages.

A typical mapping between tuning voltages and encoded amplitude and phase of such a MZM device is illustrated in Fig. 2. The MZM consists of two Y-splitters and two phase

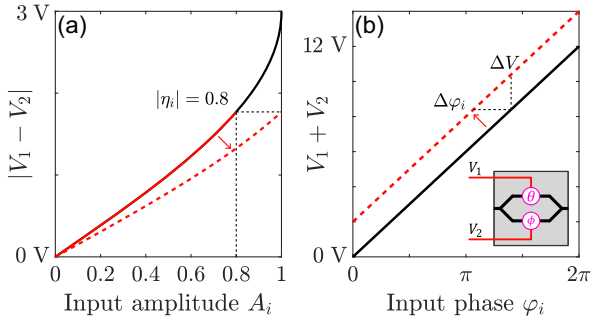


FIG. 2. Example of a Mach–Zehnder modulator for encoding complex input data ($x_i = A_i e^{i\varphi_i}$) and its conversion mappings to tuning voltages (V_1 and V_2). (a) The amplitude is encoded from the difference in tuning voltages mapped by the solid line. Re-scaling this mapping to the dashed line implements a correction factor of $|\eta_i|$ in amplitude (a correction factor of $|\eta_i| = 0.8$ is chosen for illustrative purposes). (b) The phase is encoded from the sum of the tuning voltages mapped by the solid line. Shifting this mapping to the dashed line corresponds to adding a constant phase shift of $\Delta\varphi_i$ to the encoded phase φ_i.

shifters positioned in the interferometer arms. By tuning the relative phase between the arms, constructive or destructive interference is achieved, resulting in amplitude modulation of the output. Analog MZMs are typically fabricated in $LiNbO_3$, owing to its strong Pockels effect and high-speed response [53,54]. For such MZMs with a π-phase offset in one arm, the modulated amplitude can be represented by a sine function of the differential drive voltage $A_i = \sin|\pi\frac{V_1-V_2}{2V_\pi}|$ [the solid curve in Fig. 2(a)]. While this mapping reflects the underlying mechanism of the MZM, it can be adjusted to accommodate the errors of the following photonic mesh. In doing so, we can choose a mapping such that the amplitude is scaled by a factor $|\eta_i|$, i.e., $A_i \to |\eta_i|A_i$ [the dashed line in Fig. 2(a)], resulting in a mapping from amplitude to voltage expressed by $|V_1 - V_2| = \frac{2V_\pi}{\pi}\sin^{-1}|\eta_i A_i|$. This corresponds to a rescaling of the x axis. Thus, the solid line in Fig. 2(a), which one would typically measure from device calibration, can be rescaled, forming the dashed line. We show later how this approach proves effective in reducing MVM errors.

In addition to amplitude, the MZM also enables the encoding of the phase, expressed as $\varphi_i = \pi\frac{V_1+V_2}{2V_\pi}$ [the solid line in Fig. 2(b)]. Similar to amplitude, we can adjust the phase shift by translating the solid line left and right on the x axis ($\Delta\varphi_i$), which can be implemented by simply adding a voltage bias $\Delta V/2$ to each arm. Thus, combined with the offset of the amplitude calibration, a complex correction factor, η_i, can be chosen to encode complex input data ($x_i \to \eta_i x_i$), providing extra degrees of freedom to mitigate photonic computing errors. This speaks to a general idea that any encoder can be represented by some mapping of voltage pairs to amplitude and phase, as illustrated by the black $*$ in the top-left of Fig. 1. Thus, by scaling this mapping along the amplitude axis and shifting it along the phase axis, we can implement our correction factor, η_i, as illustrated by the red $*$ in the bottom-right of Fig. 1.

III. DECODING MAPPINGS

Decoding can be achieved with various schemes such as a reverse optical setup machine [19,33,45,46], an extra unitary layer [50], a four-detector homodyne detection (HD) [52], or, as illustrated in Fig. 1, a decoder consisting of multiple HD channels, allowing measurement of field quadratures through interfering the output signal with a local oscillator [51]. Correctly decoding the output signal requires calibration of the decoder components and, consequently, a function that maps the measured photocurrents to the complex vector data, y_j.

An HD mapping between the measured photocurrents and the output vector data y_j is illustrated in Fig. 3. After interference, the two fields are detected, generating subsequent photocurrents. If the local oscillator is phase-modulated (switching between 0 and π), the real and imaginary parts of the data, y_j, can be measured, which are proportional to photocurrents ($\mathrm{Re}\{y_j\} \propto I_1^+ - I_1^-$ and $\mathrm{Im}\{y_j\} \propto I_2^+ - I_2^-$). The proportionality constant then accounts for various factors such as signal strength of the local oscillator, optical losses, detector responsivity, uneven beam-splitting ratio, relative phase difference of the fanned-out local-oscillator branches, and phase variations due to fabrication imperfections, leading to a mapping from photocurrents to complex vector data. However, in order to accommodate the error of the preceding photonic network, one could choose other proportionality constants, $|\eta_j'|e^{\Delta\phi_j}$, such that the amplitude is scaled by a factor $|\eta_j'|$, i.e., $A_j \to |\eta_j'|A_j$, and the phase undergoes a phase change of $\Delta\varphi_j$, i.e., $\varphi_j \to \varphi_j + \Delta\varphi_j$. As illustrated in Fig. 3, this corresponds to rescaling the real and imaginary axes by $|\eta_j'|$ and rotating the complex plane by $\Delta\varphi_j$. This decoding mapping is ideal when the last-layer phase shifters of the PMVMP are implemented virtually through postprocessing [26,52,55]. However, if PMVMP includes a physical last layer of phase shifters [19,46,56], it may be preferable to apply phase correction

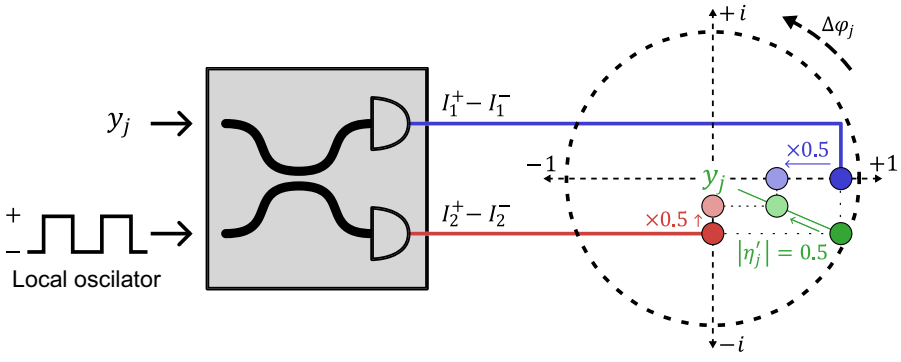


FIG. 3. Example of a homodyne detection scheme for decoding, where the output data, y_j, is extracted by interfering with an oscillating reference signal (local oscillator), thus measuring the quadratures of the signal from the differences in photocurrents. Scaling of both the real and imaginary axes by some factor corresponds to implementing an amplitude-correction factor exemplified here as $|\eta_j'| = 0.5$. Implementing phase corrections can also be achieved by rotation of the complex plane.

($\Delta\varphi_j$) directly through the last-layer phase shifters and reserve decoding mapping only for amplitude correction. In both cases, we can implement a complex correction factor to each output data ($y_j \to \eta'_j y_j$), providing additional degrees of freedom to mitigate errors. This approach is generalizable to any decoding device, as illustrated in the top-right of Fig. 1, where each $*$ represents a pair (or more) of measured photocurrents that can be mapped to an amplitude and phase. Similar to the encoding mapping, by scaling this mapping along the amplitude axis and shifting it along the phase axis, we can implement our correction factor, η'_j, as illustrated in the bottom-right of Fig. 1.

IV. ERROR-CORRECTION METHOD

As exemplified for a MZM in the former section, complex correction factors can be applied to a PMVMP through encoders, which effectively corresponds to applying a diagonal matrix, D, with correction factors, η_i, in front of the PMVMP's transformation, i.e., $\hat{U} \to \hat{U}D$. This likewise applies at the decoder level, equivalent to applying a diagonal matrix of correction factors, η'_j, at the back of the PMVMP's transformation, i.e., $\hat{U} \to D'\hat{U}$.

Equipped with these extra degrees of freedom, we seek to choose these correction factors such that we minimize the error in our MVMs. The nonideal matrix transformation $\hat{U}$ performed by the PMVMP can be directly measured by probing the system with canonical basis vectors, i.e., inputting vectors $[1,0,0,\ldots,0]$, $[0,1,0,\ldots,0]$, $\ldots$, $[0,0,\ldots,0,1]$ into the system. Thus, we attain $\hat{U}$, which can be compared against the ideal matrix transformation U. Because of nonidealities in the PMVMP, there will be a mismatch between these transformations, which we should seek to minimize. In the Appendix, we derive that to globally minimize the matrix error (defined from the Frobenius norm), the correction factors for the encoders should be found as:

$$\eta_i = \frac{\hat{v}_i^* \cdot v_i}{||\hat{v}_i||^2}, \tag{1}$$

where $\hat{v}_i$ and v_i are the column vectors of $\hat{U}$ and U, respectively. Similarly, in order to find the optimal correction factors for the decoders, we can choose η'_j according to Eq. (1), where $\hat{v}_j$ and v_j are row vectors of $\hat{U}$ and U, respectively.

The encoder and decoder calibrations can be applied in tandem by simply applying the input correction, resulting in an updated matrix $\hat{U} \to \hat{U}D$, and then applying the output correction, or vice versa, for even stronger corrections. Implementing both corrections in tandem effectively yields $\hat{U} \to D'\hat{U}D$. These corrections can, in principle, be applied iteratively, alternating between the input and output corrections, in which case they converge exponentially to the joint minimum of the two diagonal correction matrices. However, as shown in Supplementary Material [57], most of the error reduction is achieved after applying each correction once, which is the scheme used throughout this work.

V. PERFORMANCE

Loss remains a major challenge in photonic integrated circuits [58], as light waves experience attenuation due to material absorption, scattering due to sidewall roughness, fabrication imperfections, and mode coupling. In PMVMPs, beyond reducing the overall signal-to-noise ratio, loss introduces another problem that is equally, if not more, significant: its contribution to computational error in the form of unbalanced losses [31]. These errors can, in principle, be corrected by introducing additional sources of loss at the cost of increasing the overall complexity, footprint, and loss of the PMVMP [38,41]. However, the error-correction method described in the previous section can easily be used to alleviate the unbalanced-loss errors without introducing additional components.

In order to assess the performance of our method on unbalanced-loss errors, we consider triangular and rectangular meshes [30,31], with each unit cell subject to losses ranging from 0 to 1 dB. We consider an ensemble of unitary transformations randomly configured following the Haar measure [36,37] and find correction factors to minimize the matrix error according to Eq. (1) for each transformation. We use the normalized definition of the Frobenius norm $||U||^2 = \frac{1}{N}\sum_{ij}|u_{ij}|^2$. This gives us a matrix error definition equivalent to the definition used in other works [27,34]:

$$\mathcal{E} = ||\hat{U} - U||, \tag{2}$$

where we normalize our matrices to account for any overall loss such that $\hat{U} \to \hat{U}/||\hat{U}||$. For PMVMPs to reach real-world applications such as AI-acceleration, matrix errors must remain low to avoid significant performance degradation [24,25]. Figure 4(a) shows the average matrix error for ($N = 32$) triangular and rectangular meshes. As shown, the matrix errors increase with the unit-cell loss for both meshes, while the triangular mesh exhibits significantly higher errors than the rectangular mesh, which would render it ineffective for most, if not all, processing tasks. This sets hard-to-meet requirements for unit-cell losses for the triangular mesh, as most state-of-the-art MZIs have insertion losses of around 0.1 to 0.2 dB [22,51,59–64]. This vulnerability to unbalanced-loss errors can be alleviated by implementing a rectangular mesh at the cost of more complex configuration algorithms. Nevertheless, even for the rectangular mesh, unbalanced losses still provide a non-negligible error contribution.

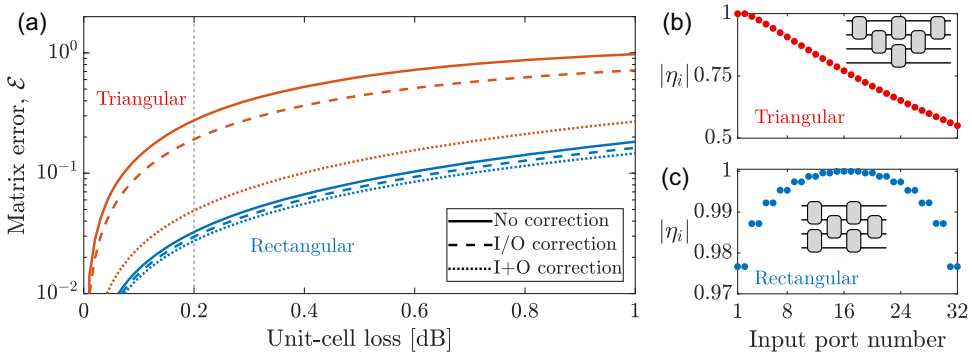


FIG. 4. (a) The average matrix error for a $N = 32$ universal multiport interferometer for the triangular mesh (red) or rectangular mesh (blue) with no error corrections (solid line), with error correction on either the encoder or decoder (dashed line), and with error correction on both the encoder and decoder (dotted line). Deviations in matrix error are generally low and are presented in Supplementary Material [57]. (b),(c) The normalized average correction factor amplitude for each input port for the triangular and rectangular mesh with a 0.2 dB unit-cell loss using both input and output correction. A small $N = 4$ triangular and rectangular mesh is shown in each figure for conceptual clarity. The matrix error and correction factor have been averaged across 10^5 Haar-random unitary matrices as a function of a constant loss of each unit cell in the mesh. The unit-cell loss is the only nonideality in this simulation, and the nonideal matrix has been normalized to account for the overall loss $\hat{U} \to \hat{U}/||\hat{U}||$. Note that the symmetric nature of the meshes leads to effectively the same performance for correcting on either the encoder or the decoder.

Our error-correction method strongly reduces the unbalanced-loss errors, as seen in Fig. 4(a), with the strongest error reduction occurring when combining corrections to both the encoder (input) and the decoder (output). For unit-cell losses of 0.2 dB, this brings the average matrix error from 0.28 to 0.05 for the triangular mesh and from 0.033 to 0.028 for the rectangular mesh, corresponding to 82% and 15% error reduction, respectively. The underlying mechanism can be understood through Figs. 4(b) and 4(c), which show the average amplitude-correction factors of the encoders. The architecture of the triangular mesh leads to high propagation losses when passing through the upper I/O ports (low port numbers) and lower propagation loss for lower I/O ports (high port numbers). Thus, our error-correction method introduces larger amplitude-correction factors, $|\eta_i|$, (closer to one) for low port numbers and smaller correction factors for high port numbers, as seen in Fig. 4(b). In contrast, in a rectangular mesh, there are fewer unit cells at the top and bottom edges of the mesh. This leads to lower propagation losses when passing through the upper and lower I/O ports (high and low port numbers); thus, as seen in Fig. 4(c), the error correction applies smaller correction factors to the upper and lower modulators and larger correction factors to the middle modulators to account for unbalanced losses. It should be noted that a rectangular mesh has a lower overall matrix error because the unit cells are distributed more uniformly across the mesh, leading to correction factors closer to 1. We do not show the phase-correction factors, as they are quite small and on average zero for unbalanced-loss errors.

To provide a broader picture of the application of our error-correction method beyond unbalanced-loss errors and to investigate its performance as a function of the PMVMP size, we additionally consider other sources of error, such as 50/50 BS deviations and DAC inaccuracies, as well as Gaussian-distributed fluctuations in the matrix elements as a general model for MVM error. Figure 5 shows the relative error reduction $\Delta\mathcal{E}/\mathcal{E}$ for various error types as a function of PMVMP size, where $\Delta\mathcal{E}$ is the difference between the corrected and uncorrected errors.

For the BS error, we perturb each beam splitter by introducing a random deviation from the ideal 50/50 splitting ratio. Specifically, the beam-splitter amplitudes are parametrized as $\cos(\pi/4 + \alpha)$ and $\sin(\pi/4 + \alpha)$, where α denotes the deviation from the ideal mixing angle of $\pi/4$ and is sampled independently from a zero-mean normal

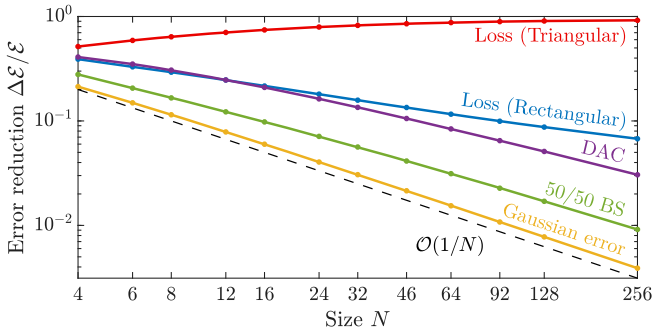


FIG. 5. The average relative matrix error reduction for various nonidealities such as unbalanced losses in a triangular mesh (red), unbalanced losses in a rectangular mesh (blue), DAC errors (magenta), 50/50 beam-splitter deviations (green), and Gaussian-distributed errors in matrix elements (yellow). To account for overall losses, the nonideal matrices have been normalized $\hat{U} \to \hat{U}/||\hat{U}||$. The matrix errors are averaged over $10^3 \times 256/N$ Haar-random unitary matrices. Deviations in error reductions are generally low and are presented in Supplementary Material [57].

distribution with a standard deviation of $\sigma_{\mathrm{BS}} = 0.02$ rad for each beam splitter [27,34,43]. To first order, this corresponds to a 2% deviation in the power splitting ratio, consistent with values reported for foundry components [65]. Digital-to-analog conversion inaccuracies arise from the error introduced when a continuous signal between 0 and 2π is quantized into 2^B discrete levels ($\sigma_{\mathrm{DAC}} = 2\pi/\sqrt{12} \times 2^{-B}$) [27]. Similar to previous work, we assume DACs with 12-bit precision [27]. Both 50/50 BS and DAC errors have already been well studied [27,34,36,43], and subsequent self-configuration algorithms have been developed to drastically reduce these errors [27,28]. Despite this, our error-correction method yields a notable error reduction in both cases, with reductions of around 40% for DAC and 30% for 50/50 BS in small meshes to a few percent for large meshes, as shown in Fig. 5.

So far, we have considered well-understood nonidealities as individual error sources. However, in PMVMPs, many other error sources, whose explicit modeling is nontrivial or whose physical origins are not fully characterized, may be present. Thus, in order to capture a worst-case scenario, we consider 1% Gaussian noise to the real and imaginary parts of each matrix element as a randomly distributed perturbation without making any assumptions about their physical origin. For such random errors, we achieve approximately 20% error reduction for small meshes and less than 1% for large meshes.

We note that the relative error reduction presented in Fig. 5 is largely independent of the strength of the underlying error source, such as the unit-cell loss, beam-splitter deviation, DAC precision, or Gaussian noise, and depends primarily on the mesh size N. This independence holds throughout the regime of practical interest and breaks down when the absolute errors approach order unity.

Note that only unbalanced-loss errors depend on the mesh architecture. For the other error sources, we observe that the triangular and rectangular meshes yield statistically identical matrix errors and hence identical average error reductions, as both comprise the same components assembled in differing positions. In addition, in most cases, the effectiveness of our method decreases with increasing PMVMP size, consistent with the trend observed in other error-correction methods [27,28,43]. In our case, this reduction can be understood from the fact that unitary matrices have N^2 degrees of freedom (DOF), while our method only adds $2N$ new complex DOF, leading to a greater mismatch in DOF for larger PMVMPs. Thus, for uncorrelated matrix errors, we expect a scaling of $\mathcal{O}(1/N)$ in error reduction, which is exactly the scaling that we observe for Gaussian errors presented in Fig. 5. However, for other error types, we see a better scaling than $\mathcal{O}(1/N)$. We attribute this to correlated errors in the matrix elements, which can be corrected more effectively by the encoder and decoder devices. This holds more strongly for unbalanced-loss errors, particularly for the triangular mesh, where we achieve around 50% error reduction for small meshes and over 90% for medium to large meshes. An interesting observation is that the relative error reduction for the triangular mesh increases with N. This distinctive characteristic arises from the triangular shape of the mesh, where the primary contributor to unbalanced losses is the variation in propagation loss when optical fields pass through different I/O ports. Since these differences in propagation loss can be readily corrected at the I/O ports where the encoders and decoders are present, our method delivers the most effective correction. Furthermore, as the network size increases, the unbalanced-loss error increases, resulting in a correspondingly larger relative error correction. Currently, the triangular mesh is only implemented for small PMVMPs with $N = 4$ to 6 [19,20,45,46,64]. However, with the correction method presented in this work, large-scale PMVMPs can implement triangular meshes with errors that rival those of rectangular meshes.

VI. DISCUSSION

Analog photonic computing relies on continuous wave quantities, i.e., amplitude and phase, thereby significantly affected by imperfections. As a result, error correction is essential for their performance. In this work, we have demonstrated that by offsetting the calibration of the encoder and decoder devices of a PMVMP, extra complex degrees of freedom can be attained to correct computing errors. In contrast to the rich body of existing work on error correction, our work presents a method that is applicable to a variety of errors, including unbalanced loss, DAC, 50/50 BS, and any unknown errors. For example, unwanted phase shifts and reflections at the interface between different components are common sources of errors in photonic integrated circuits. Although the precise modeling of these nonidealities is deferred to future work, the method we presented in this work can reduce errors arising from these and other unmodeled or unknown sources. The general-purpose nature of our error-correction method makes it appealing to a wide range of PMVMP architectures, such as unitary optical converters [16,66], recirculating meshes [56,67], and even nonunitary architectures such as coherent cross-bar designs [13,68] and singular-value decomposition meshes [29,47,69]. In general, any photonic system that shares the PMVMP structure of an encoder-transformation-decoder, such as free-space diffractive optical processors with programmable input modulators and pixel-array detectors [70], can admit the same closed-form correction of Eq. (1). In particular, photonic networks with predefined transformations that lack tunability [71] may benefit from our work, and our method can prove pivotal in reducing fabrication-induced errors, as well as providing some level of programmability. Beyond its broad applicability, a key advantage of our work is that it is not restricted to unitary errors [28] and proves very powerful against nonunitary errors, such as unbalanced

losses, allowing for error reductions of more than 90% for medium to large triangular meshes. Furthermore, our method does not impose any overhead on photonic processor speed or energy efficiency, since the correction is applied through recalibration of existing encoder and decoder devices rather than through additional components or operations. The associated one-time calibration requires only $\mathcal{O}(N)$ probing measurements, which is negligible compared with the at least $\mathcal{O}(N^2)$ steps required for the full device configuration [27–31].

In this work, we have considered complex MVMs, as photonic networks naturally implement complex linear operations. However, there is a substantial body of work that traditionally performs real-valued MVMs using light, such as real-valued photonic tensor cores [11,12,48,49] and diffractive optical processors [15,70]. In such cases, our method still remains applicable by computing the correction factors in the same way using Eq. (1) but disregarding the phase-correction factors and only using the amplitude-correction factors, i.e., $|\eta_i|$ and $|\eta'_j|$. Furthermore, the extra degrees of freedom of encoder and decoder calibrations we introduced in this work could also be beneficial in all-optical neural networks [51]. These systems cannot differentiate between an ideal and a nonideal transfer matrix operation. Thus, while optimal correction factors may not be essential, the system can still be trained with these extra degrees of freedom. This regime is particularly relevant for diffractive neural networks [15,72,73], which are typically trained end-to-end on input-output features without an explicit target transformation.

ACKNOWLEDGMENTS

We thank S. Stobbe for valuable discussions. We acknowledge financial support from the European Union's Horizon research and innovation programme (Grant No. 101098961—NEUROPIC) and the Innovation Fund Denmark (Grant No. 4356-00007B—EQUAL).

DATA AVAILABILITY

The code and generated data that support the findings of this article are openly available [74].

APPENDIX: MINIMIZATION OF MATRIX ERROR

The Frobenius norm $||\cdot||$ of a matrix is canonically defined by the elementwise squared sum of the elements. This can also be written as the sum of the column vectors squared (we omit the $1/N$ factor for simplicity):

$$||A||^2 = \sum_{i,j} |a_{ij}|^2 = \sum_i ||w_i||_2^2. \tag{A1}$$

Thus, we can write the matrix error as the sum of vector differences:

$$\mathcal{E}^2 = ||\hat{U}D - U||^2 = \sum_i ||\hat{v}_i\eta_i - v_i||_2^2 = \sum_i \mathcal{E}_i^2. \tag{A2}$$

In order to minimize the error, we simply have to minimize each of the vector errors $\mathcal{E}_i$. The 2-norm of a vector can be written as the inner product with itself $||w||^2 = \langle w|w\rangle$. We can thus write our vector error as:

$$\mathcal{E}_i^2 = \langle v_i|v_i\rangle + |\eta_i|^2\langle\hat{v}_i|\hat{v}_i\rangle - \eta_i\langle v_i|\hat{v}_i\rangle - \eta_i^*\langle\hat{v}_i|v_i\rangle. \tag{A3}$$

Minimizing Eq. (A3) can easily be done by first finding the phase such that $\arg(\eta_i) = \arg(\langle\hat{v}_i|v_i\rangle)$. Thereafter, the amplitude can easily be found to be:

$$|\eta_i| = \frac{|\langle\hat{v}_i|v_i\rangle|}{||\hat{v}_i||^2}. \tag{A4}$$

By combining the amplitude and phase parts, we arrive at Eq. (1). The same derivation can be done to minimize the decoder correction factors, η'_j, by instead considering the row vectors of $\hat{U}$ and U.

[1] J. Shalf, The future of computing beyond Moore's law, Philos. Trans. R. Soc., A **378**, 20190061 (2020).
[2] K. Rupp, 50 years of microprocessor trend data (2021), https://github.com/karlrupp/microprocessor-trend-data.
[3] E. Strubell, A. Ganesh, and A. McCallum, Energy and policy considerations for modern deep learning research, in *Proceedings of the AAAI Conference on Artificial Intelligence* (2020), Vol. 34, pp. 13693–13696.
[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. **25**, 1097 (2012).
[5] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, Language models are few-shot learners, Adv. Neural Inf. Process. Syst. **33**, 1877 (2020).
[6] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, Mastering the game of Go without human knowledge, Nature (London) **550**, 354 (2017).
[7] P. L. McMahon, The physics of optical computing, Nat. Rev. Phys. **5**, 717 (2023).
[8] H. Zhou, J. Dong, J. Cheng, W. Dong, C. Huang, Y. Shen, Q. Zhang, M. Gu, C. Qian, H. Chen, Z. Ruan, and X. Zhang, Photonic matrix multiplication lights up photonic accelerator and beyond, Light Sci. Appl. **11**, 30 (2022).
[9] N. C. Harris, J. Carolan, D. Bunandar, M. Prabhu, M. Hochberg, T. Baehr-Jones, M. L. Fanto, A. M. Smith, C. C. Tison, P. M. Alsing, and D. Englund, Linear programmable nanophotonic processors, Optica **5**, 1623 (2018).
[10] T. Fu, J. Zhang, R. Sun, Y. Huang, W. Xu, S. Yang, Z. Zhu, and H. Chen, Optical neural networks: Progress and challenges, Light Sci. Appl. **13**, 263 (2024).

[11] S. R. Ahmed, R. Baghdadi, M. Bernadskiy, N. Bowman, R. Braid, J. Carr, C. Chen, P. Ciccarella, M. Cole, J. Cooke *et al.*, Universal photonic artificial intelligence acceleration, Nature (London) **640**, 368 (2025).

[12] S. Hua, E. Divita, S. Yu, B. Peng, C. Roques-Carmes, Z. Su, Z. Chen, Y. Bai, J. Zou, Y. Zhu, Y. Xu, C.-K. Lu, Y. Di, H. Chen, L. Jiang, L. Wang, L. Ou, C. Zhang, J. Chen, W. Zhang, H. Zhu, W. Kuang, L. Wang, H. Meng, M. Steinman, and Y. Shen, An integrated large-scale photonic accelerator with ultralow latency, Nature (London) **640**, 361 (2025).

[13] M. Moralis-Pegios, G. Giamougiannis, A. Tsakyridis, D. Lazovsky, and N. Pleros, Perfect linear optics using silicon photonics, Nat. Commun. **15**, 5468 (2024).

[14] T. Wang, S.-Y. Ma, L. G. Wright, T. Onodera, B. C. Richard, and P. L. McMahon, An optical neural network using less than 1 photon per multiplication, Nat. Commun. **13**, 123 (2022).

[15] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, All-optical machine learning using diffractive deep neural networks, Science **361**, 1004 (2018).

[16] R. Tang, R. Tanomura, T. Tanemura, and Y. Nakano, Ten-port unitary optical processor on a silicon photonic chip, ACS Photonics **8**, 2074 (2021).

[17] R. Tang, T. Tanemura, and Y. Nakano, Integrated reconfigurable unitary optical mode converter using MMI couplers, IEEE Photonics Technol. Lett. **29**, 971 (2017).

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

[19] H. Zhang, M. Gu, X. D. Jiang, J. Thompson, H. Cai, S. Paesani, R. Santagati, A. Laing, Y. Zhang, M. H. Yung, Y. Z. Shi, F. K. Muhammad, G. Q. Lo, X. S. Luo, B. Dong, D. L. Kwong, L. C. Kwek, and A. Q. Liu, An optical neural chip for implementing complex-valued neural network, Nat. Commun. **12**, 457 (2021).

[20] J. Carolan, C. Harrold, C. Sparrow, E. Martín-López, N. J. Russell, J. W. Silverstone, P. J. Shadbolt, N. Matsuda, M. Oguma, M. Itoh, G. D. Marshall, M. G. Thompson, J. C. F. Matthews, T. Hashimoto, J. L. O'Brien, and A. Laing, Universal linear optics, Science **349**, 711 (2015).

[21] S. Lin, Y. Zhang, Z. Wu, S. Zeng, Q. Gao, J. Li, X. Yu, and S. Yu, Power-efficient programmable integrated multiport photonic interferometer in CMOS-compatible silicon nitride, Photonics Res. **12**, A11 (2024).

[22] N. C. Harris, G. R. Steinbrecher, M. Prabhu, Y. Lahini, J. Mower, D. Bunandar, C. Chen, F. N. C. Wong, T. Baehr-Jones, M. Hochberg, S. Lloyd, and D. Englund, Quantum transport simulations in a programmable nanophotonic processor, Nat. Photonics **11**, 447 (2017).

[23] S. Gupta and J. Xavier, Neuromorphic photonic on-chip computing, Chips **4**, 34 (2025).

[24] M. Y.-S. Fang, S. Manipatruni, C. Wierzynski, A. Khosrowshahi, and M. R. DeWeese, Design of optical neural networks with component imprecisions, Opt. Express **27**, 14009 (2019).

[25] S. K. Vadlamani, D. Englund, and R. Hamerly, Transferable learning on analog hardware, Sci. Adv. **9**, eadh3436 (2023).

[26] Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljačić, Deep learning with coherent nanophotonic circuits, Nat. Photonics **11**, 441 (2017).

[27] R. Hamerly, S. Bandyopadhyay, and D. Englund, Accurate self-configuration of rectangular multiport interferometers, Phys. Rev. Appl. **18**, 024019 (2022).

[28] R. Hamerly, S. Bandyopadhyay, and D. Englund, Stability of self-configuring large multiport interferometers, Phys. Rev. Appl. **18**, 024018 (2022).

[29] D. A. B. Miller, Self-configuring universal linear optical component, Photonics Res. **1**, 1 (2013).

[30] M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, Experimental realization of any discrete unitary operator, Phys. Rev. Lett. **73**, 58 (1994).

[31] W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, Optimal design for universal multiport interferometers, Optica **3**, 1460 (2016).

[32] S. Pai, I. A. D. Williamson, T. W. Hughes, M. Minkov, O. Solgaard, S. Fan, and D. A. B. Miller, Parallel programming of an arbitrary feedforward photonic network, IEEE J. Sel. Top. Quantum Electron. **26**, 1 (2020).

[33] D. A. B. Miller, Setting up meshes of interferometers–reversed local light interference method, Opt. Express **25**, 29233 (2017).

[34] R. Hamerly, S. Bandyopadhyay, and D. Englund, Asymptotically fault-tolerant programmable photonics, Nat. Commun. **13**, 6831 (2022).

[35] D. A. B. Miller, Perfect optics with imperfect components, Optica **2**, 747 (2015).

[36] S. Pai, B. Bartlett, O. Solgaard, and D. A. B. Miller, Matrix optimization on universal unitary photonic devices, Phys. Rev. Appl. **11**, 064044 (2019).

[37] N. J. Russell, L. Chakhmakhchyan, J. L. O'Brien, and A. Laing, Direct dialling of Haar random unitary matrices, New J. Phys. **19**, 033007 (2017).

[38] K. H. R. Mojaver, B. Zhao, E. Leung, S. M. R. Safaee, and O. Liboiron-Ladouceur, Addressing the programming challenges of practical interferometric mesh based optical processors, Opt. Express **31**, 23851 (2023).

[39] J. R. Basani, S. K. Vadlamani, S. Bandyopadhyay, D. R. Englund, and R. Hamerly, A self-similar sine–cosine fractal architecture for multiport interferometers, Nanophotonics **12**, 975 (2023).

[40] S. A. Fldzhyan, M. Y. Saygin, and S. P. Kulik, Optimal design of error-tolerant reprogrammable multiport interferometers, Opt. Lett. **45**, 2632 (2020).

[41] F. Marchesin, M. Hejda, T. Melendez Carmona, S. Di Carlo, A. Savino, F. Pavanello, T. Van Vaerenbergh, and P. Bienstman, Braided interferometer mesh for robust photonic matrix-vector multiplications with non-ideal components, Opt. Express **33**, 2227 (2025).

[42] S. A. Fldzhyan, M. Y. Saygin, and S. S. Straupe, Low-depth, compact, and error-tolerant photonic matrix-vector multiplication beyond the unitary group, Opt. Express **32**, 46239 (2024).

[43] S. Bandyopadhyay, R. Hamerly, and D. Englund, Hardware error correction for programmable photonics, Optica **8**, 1247 (2021).

[44] T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, Training of photonic neural networks through in situ backpropagation and gradient measurement, Optica **5**, 864 (2018).

[45] S. Pai, Z. Sun, T. W. Hughes, T. Park, B. Bartlett, I. A. D. Williamson, M. Minkov, M. Milanizadeh, N. Abebe, F. Morichetti, A. Melloni, S. Fan, O. Solgaard, and D. A. B. Miller, Experimentally realized in situ backpropagation for deep learning in photonic neural networks, Science **380**, 398 (2023).

[46] Y. Zheng, R. Wu, Y. Ren, R. Bao, J. Liu, Y. Ma, M. Wang, and Y. Cheng, Photonic neural network fabricated on thin film lithium niobate for high-fidelity and power-efficient matrix computation, Laser Photon. Rev. **18**, 2400565 (2024).

[47] Z. Xu, T. Zhou, M. Ma, C. C. Deng, Q. Dai, and L. Fang, Large-scale photonic chiplet Taichi empowers 160-TOPS/W artificial general intelligence, Science **384**, 202 (2024).

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

[49] B. Dong, F. Brückerhoff-Plückelmann, L. Meyer, J. Dijkstra, I. Bente, D. Wendland, A. Varri, S. Aggarwal, N. Farmakidis, M. Wang *et al.*, Partial coherence enhances parallelized photonic computing, Nature (London) **632**, 55 (2024).

[50] R. Qiu, A. Eldebiky, G. L. Zhang, X. Yin, C. Zhuo, U. Schlichtmann, and B. Li, OplixNet: Towards area-efficient optical split-complex networks with real-to-complex data assignment and knowledge distillation, in *Proceedings of the 2024 Design, Automation & Test in Europe Conference & Exhibition* (IEEE, 2024), pp. 1–6.

[51] S. Bandyopadhyay, A. Sludds, S. Krastanov, R. Hamerly, N. Harris, D. Bunandar, M. Streshinsky, M. Hochberg, and D. Englund, Single-chip photonic deep neural network with forward-only training, Nat. Photonics **18**, 1335 (2024).

[52] Y. Xie, X. Ke, S. Hong, Y. Sun, L. Song, H. Li, P. Wang, and D. Dai, Complex-valued matrix-vector multiplication using a scalable coherent photonic processor, Sci. Adv. **11**, eads7475 (2025).

[53] C. Wang, M. Zhang, X. Chen, M. Bertrand, A. Shams-Ansari, S. Chandrasekhar, P. Winzer, and M. Lončar, Integrated lithium niobate electro-optic modulators operating at CMOS-compatible voltages, Nature (London) **562**, 101 (2018).

[54] M. Xu, M. He, H. Zhang, J. Jian, Y. Pan, X. Liu, L. Chen, X. Meng, H. Chen, Z. Li, X. Xiao, S. Yu, S. Yu, and X. Cai, High-performance coherent optical modulators based on thin-film lithium niobate platform, Nat. Commun. **11**, 3911 (2020).

[55] Z. Du, K. Liao, T. Dai, Y. Wang, J. Gao, H. Huang, H. Qi, Y. Li, X. Wang, X. Su, X. Wang, Y. Yang, C. Lu, X. Hu, and Q. Gong, Ultracompact and multifunctional integrated photonic platform, Sci. Adv. **10**, eadm7569 (2024).

[56] Y. Zhu, Y. Liu, X. Yang, K. Liu, X. Hua, M. Luo, J. Liu, S. Chang, S. Zhang, M. Wu *et al.*, Versatile silicon integrated photonic processor: A reconfigurable solution for next-generation AI clusters, arXiv:2504.01463.

[57] See Supplementary material at http://link.aps.org/supplemental/10.1103/71gp-fpwf for iterative corrections and simulation deviations.

[58] C. A. Rosiek, G. Arregui, A. Vladimirova, M. Albrechtsen, B. Vosoughi Lahijani, R. E. Christiansen, and S. Stobbe, Observation of strong backscattering in valley-Hall photonic topological interface modes, Nat. Photonics **17**, 386 (2023).

[59] L. Chen and Y.-K. Chen, Compact, low-loss and low-power 8×8 broadband silicon optical switch, Opt. Express **20**, 18977 (2012).

[60] K. Suzuki, R. Konoike, J. Hasegawa, S. Suda, H. Matsuura, K. Ikeda, S. Namiki, and H. Kawashima, Low-insertion-loss and power-efficient 32×32 silicon photonics switch with extremely high-Δ silica PLC connector, J. Lightwave Technol. **37**, 116 (2019).

[61] C. Taballione, R. van der Meer, H. J. Snijders, P. Hooijschuur, J. P. Epping, M. de Goede, B. Kassenberg, P. Venderbosch, C. Toebes, H. van den Vlekkert, P. W. H. Pinkse, and J. J. Renema, A universal fully reconfigurable 12-mode quantum photonic processor, Mater. Quantum Technol. **1**, 035002 (2021).

[62] C. Taballione, M. C. Anguita, M. de Goede, P. Venderbosch, B. Kassenberg, H. Snijders, N. Kannan, W. L. Vleeshouwers, D. Smith, J. P. Epping, R. van der Meer, P. W. H. Pinkse, H. van den Vlekkert, and J. J. Renema, 20-mode universal quantum photonic processor, Quantum **7**, 1071 (2023).

[63] D. U. Kim, Y. J. Park, D. Y. Kim, Y. Jeong, M. G. Lim, M. S. Hong, M. J. Her, Y. Rah, D. J. Choi, S. Han, and K. Yu, Programmable photonic arrays based on microelectromechanical elements with femtowatt-level standby power consumption, Nat. Photonics **17**, 1089 (2023).

[64] Y. Zheng, H. Zhong, H. Zhang, L. Song, J. Liu, Y. Liang, Z. Liu, J. Chen, J. Zhou, Z. Fang, M. Wang, L. Li, R. Wu, and Y. Cheng, Electro-optically programmable photonic circuits enabled by wafer-scale integration on thin-film lithium niobate, Phys. Rev. Res. **5**, 033206 (2023).

[65] J. C. Mikkelsen, W. D. Sacher, and J. K. S. Poon, Dimensional variation tolerant silicon-on-insulator directional couplers, Opt. Express **22**, 3145 (2014).

[66] M. Markowitz, K. Zelaya, and M.-A. Miri, Learning arbitrary complex matrices by interlacing amplitude and phase masks with fixed unitary operations, Phys. Rev. A **110**, 033501 (2024).

[67] D. Perez, I. Gasulla, F. J. Fraile, L. Crudgington, D. J. Thomson, A. Z. Khokhar, K. Li, W. Cao, G. Z. Mashanovich, and J. Capmany, Silicon photonics rectangular universal interferometer, Laser Photon. Rev. **11**, 1700219 (2017).

[68] G. Giamougiannis, A. Tsakyridis, Y. Ma, A. Totović, M. Moralis-Pegios, D. Lazovsky, and N. Pleros, A coherent photonic crossbar for scalable universal linear optics, J. Lightwave Technol. **41**, 2425 (2023).

[69] R. Tang, R. Tanomura, T. Tanemura, and Y. Nakano, Lower-depth programmable linear optical processors, Phys. Rev. Appl. **21**, 014054 (2024).

[70] T. Zhou, X. Lin, J. Wu, Y. Chen, H. Xie, Y. Li, J. Fan, H. Wu, L. Fang, and Q. Dai, Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit, Nat. Photonics **15**, 367 (2021).

[71] V. Nikkhah, A. Pirmoradi, F. Ashtiani, B. Edwards, F. Aflatouni, and N. Engheta, Inverse-designed low-index-contrast structures on a silicon photonics platform for vector–matrix multiplication, Nat. Photonics **18**, 501 (2024).

[72] C. Qian, X. Lin, X. Lin, J. Xu, Y. Sun, E. Li, B. Zhang, and H. Chen, Performing optical logic operations by a diffractive neural network, Light Sci. Appl. **9**, 59 (2020).

[73] Z. Fan, C. Qian, Y. Jia, Y. Feng, H. Qian, E.-P. Li, R. Fleury, and H. Chen, Holographic multiplexing metasurface with twisted diffractive neural network, Nat. Commun. **15**, 9416 (2024).

[74] A. Carstensen and B. Vosoughi Lahijani, Photonic-computing error correction through optical encoder and decoder calibrations: Matlab code (2026), 10.5281/zenodo.18608080.