Hardware error correction for programmable photonics

Research Article Vol. 8, No. 10 / October 2021 / Optica 1247

Hardware error correction for programmable photonics

Saumil Bandyopadhyay,[1,*] Ryan Hamerly,[1,2] and Dirk Englund[1]

[1]Research Laboratory of Electronics, MIT, 50 Vassar Street, Cambridge, Massachusetts 02139, USA
[2]NTT Research Inc., PHI Laboratories, 940 Stewart Drive, Sunnyvale, California 94085, USA
*Corresponding author: saumilb@mit.edu

Received 2 March 2021; revised 1 July 2021; accepted 18 August 2021 (Doc. ID 424052); published 27 September 2021

Programmable photonic circuits of reconfigurable interferometers can be used to implement arbitrary operations on optical modes, providing a flexible platform for accelerating tasks in quantum simulation, signal processing, and artificial intelligence. A major obstacle to scaling up these systems is static fabrication error, where small component errors within each device accrue to produce significant errors within the circuit computation. Mitigating this error usually requires numerical optimization dependent on real-time feedback from the circuit, which can greatly limit the scalability of the hardware. Here we present a deterministic approach to correcting circuit errors by locally correcting hardware errors within individual optical gates. We apply our approach to simulations of large scale optical neural networks and infinite impulse response filters implemented in programmable photonics, finding that they remain resilient to component error well beyond modern day process tolerances. Our results highlight a potential way to scale up programmable photonics to hundreds of modes with current fabrication processes.

https://doi.org/10.1364/OPTICA.424052

1. INTRODUCTION

Integrated photonics is a key technology for optical communications and is advancing rapidly for applications in sensing, metrology, signal processing, and computation. Programmable photonic circuits of optical interferometers, which can implement arbitrary filters and passively compute matrix operations on optical modes, are the optical analog to field programmable gate arrays (FPGAs) and enable photonic circuits to be flexibly reconfigured post-fabrication by software [1,2]. Experimental demonstrations of these circuits have already shown working systems operating on up to tens of optical modes, which have been used to accelerate tasks in quantum simulation [3–7], mode unscrambling [8–12], signal processing [13,14], combinatorial optimization [15], and artificial intelligence [16].

While scaling up these systems to hundreds or thousands of modes would be immensely beneficial, doing so will require precise fabrication of tens of thousands of optical interferometers. Unfortunately, static component errors induced by process variation introduce errors that rapidly accrue for larger systems, limiting their usefulness for many applications. This is because the decomposition [17,18] and optimization techniques used to program these circuits assume that all of the components are ideal; thus, any component errors result in a programming of the wrong operation. Component imprecision therefore has serious implications for the future of these systems; for example, beam splitter variation as small as 2%, which is a typical wafer-level variance [19], has been shown to degrade accuracy by nearly 50% for feedforward circuits used to implement classifiers for the MNIST image recognition task [20]. Alternative programmable architectures, such as recirculating waveguide meshes consisting of triangular or hexagonal Mach–Zehnder interferometer (MZI) lattices [13,21,22], are similarly susceptible to component-induced error; device variation within these circuits introduces errors that will alter the response of phase-sensitive filters [23]. These systems' degrees of sensitivity to component variation make their control challenging when scaling up to large numbers of modes.

Hardware errors are usually compensated for with numerical optimization. A number of global optimization approaches have been proposed in the past, including nonlinear optimization [24–28], gradient descent [29], and *in situ* backpropagation and training for neural networks [30]. These strategies, however, are time consuming and can scale poorly with circuit size. Moreover, it is often inefficient to retrain hardware settings for each individual chip. For many tasks, such as machine learning, model training is energy intensive; if the same model parameters are broadcast to thousands of chips within a data center, retraining the model for each chip with a unique set of component imprecisions will be very costly. One can instead employ progressive algorithms making use of local feedback [31,32]; however, these algorithms, which iteratively optimize the settings of one device at a time, require $O(N^2)$ tap photodiodes to monitor the optical power within each individual interferometer. This requirement greatly increases the number of electrical lines and overall power consumption of the system.

This focus on *in situ* approaches reveals a critical roadblock for programmable photonics compared to electronic FPGAs.

An FPGA does not optimize hardware settings in real-time off readings taken directly from the chip; rather, control software takes for granted that the logic gates are ideal and maps the requested function into a netlist that can be placed and routed within the chip. A similar capability for programmable photonics would greatly improve the scalability of these systems; if this were the case, a desired optical function could be trained once on an idealized software model and ported over to many chips. The challenge for programmable photonics is that unlike FPGAs, photonic circuits are analog systems that are far more sensitive to errors within individual components. Enabling this level of scalability will therefore require the ability to deterministically correct hardware errors in photonic chips.

If a unitary operation is realizable by an imperfect photonic circuit, it should not require optimization to deduce the required settings; rather, a small perturbation in the device behavior due to component deviation should translate directly to a small perturbation in the interferometer's phase settings to recover the original unitary. This insight has led us to consider a local approach that corrects hardware errors one at a time within each optical gate composing the circuit. In this paper, we present an approach to directly correct hardware errors for a programmable photonic circuit. Our algorithm outperforms previous approaches in several key respects: (1) it is flexible, requiring only a one time device calibration to directly compute the hardware settings for any given unitary; (2) for sufficiently low hardware errors, the computed settings yield the exact unitary desired; and (3) our approach requires minimal overhead and does not make use of additional interferometers or internal detectors within every device. Our analysis is focused on feedforward programmable circuits that implement arbitrary unitary matrices, as these systems have the most demanding requirements for fabrication precision. However, our approach is a local error correction strategy that individually corrects each 2×2 optical gate within the circuit. It therefore does not assume any particular structure to the circuit and can be generalized to any programmable architecture making use of interferometers, including feedforward circuits with redundant devices and recirculating waveguide meshes.

2. HARDWARE ERROR CORRECTION

Local error correction requires characterization of each phase shifter and passive splitter in the photonic circuit. The calibration is performed once with the results stored in a lookup table; any arbitrary function can then be programmed by computing the settings for an ideal set of MZIs and converting them, one by one, to the corresponding settings for an imperfect device. In Supplement 1, Section I, we describe how to calibrate these parameters using detectors only at the circuit outputs; assuming these errors are known, we can proceed with error correction as follows.

The fundamental optical gate of a programmable photonic circuit is a 2×2 MZI composed of an external phase shifter on one input, two 50-50 beam splitters, and an internal phase shifter on one of the modes between the splitters [Fig. 1(a)]. This device is an electrically programmable beam splitter capable of performing a 2×2 unitary operation $T_{ij}(\theta, \phi)$ on optical modes i, j parameterized by the external phase shift ϕ and the internal phase shift θ.

On an integrated photonics platform, the 50-50 splitters can be realized by a directional coupler or multimode interferometer (MMI); the operation of these splitters can be described by a 2×2 matrix:

$$\begin{bmatrix} \cos(\pi/4+\alpha) & i\sin(\pi/4+\alpha) \\ i\sin(\pi/4+\alpha) & \cos(\pi/4+\alpha) \end{bmatrix}, \tag{1}$$

where α describes the deviation from an ideal 50-50 splitting behavior. For an ideal splitter $\alpha = 0$, this matrix reduces to

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}. \tag{2}$$

The overall operation $T_{ij}(\theta, \phi)$ performed by a single ideal MZI is therefore

$$\begin{aligned} T_{ij}(\theta,\phi) &= \frac{1}{2}\begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}\begin{bmatrix} e^{i\theta} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}\begin{bmatrix} e^{i\phi} & 0 \\ 0 & 1 \end{bmatrix} \\ &= ie^{i\theta/2}\begin{bmatrix} e^{i\phi}\sin(\theta/2) & \cos(\theta/2) \\ e^{i\phi}\cos(\theta/2) & -\sin(\theta/2) \end{bmatrix}, \end{aligned} \tag{3}$$

where θ, ϕ are single-mode phase shifts on the top arm.

Higher dimensional matrix operations can be implemented with this unit cell by applying the Clements [18] and Reck [17] decompositions [Fig. 1(b)]. These algorithms decompose an arbitrary N-dimensional unitary U into a product of $N(N-1)/2$ two-dimensional unitaries computed by interference between nearest-neighbor optical modes, followed by phase shifts on the output modes corresponding to a diagonal matrix D:

$$U = D\prod T_{ij}(\theta, \phi). \tag{4}$$

We now analyze the impact of fabrication error. If the MZI has imperfect splitters with errors α, β, the operation of the MZI must now be parameterized with four variables $T'_{ij}(\theta, \phi, \alpha, \beta)$ [Fig. 1(c)]:

$$ie^{i\theta/2}\begin{bmatrix} e^{i\phi}(\cos(\alpha-\beta)\sin(\theta/2)+ & \cos(\alpha+\beta)\cos(\theta/2)+ \\ i\sin(\alpha+\beta)\cos(\theta/2)) & i\sin(\alpha-\beta)\sin(\theta/2) \\ e^{i\phi}(\cos(\alpha+\beta)\cos(\theta/2)- & -\cos(\alpha-\beta)\sin(\theta/2)+ \\ i\sin(\alpha-\beta)\sin(\theta/2)) & i\sin(\alpha+\beta)\cos(\theta/2) \end{bmatrix} \tag{5}$$

$$= \begin{bmatrix} \cos\beta & i\sin\beta \\ i\sin\beta & \cos\beta \end{bmatrix}\hat{T}(\theta,\phi)\begin{bmatrix} \cos\alpha & ie^{-i\phi}\sin\alpha \\ ie^{i\phi}\sin\alpha & \cos\alpha \end{bmatrix}. \tag{6}$$

In the limit $\alpha, \beta \to 0$, the second term of each entry in the matrix $T'_{ij}(\theta, \phi, \alpha, \beta)$ drops out and we recover the expected transformation for an ideal device. Naturally, implementing the usual decomposition on these imperfect devices will not yield the desired unitary:

$$D\prod T'_{ij}(\theta,\phi,\alpha,\beta) \neq D\prod T_{ij}(\theta,\phi). \tag{7}$$

To program into an imperfect circuit a desired unitary $U = \prod_{T_{ij}}(\theta, \phi)$, we apply local corrections $\theta \to \theta', \phi \to \phi'$ to each device such that $T'_{ij}(\theta', \phi', \alpha, \beta) = T_{ij}(\theta, \phi)$.

Figure 2(a) illustrates our approach. We begin by finding θ' such that the magnitudes of the entries of $T'_{ij}(\theta', \phi', \alpha, \beta)$ equal those of $T_{ij}(\theta, \phi)$. This condition produces the following expression for θ' (Supplement 1, Section III.A):

$$\theta' = 2\arcsin\sqrt{\frac{\sin^2(\theta/2)-\sin^2(\alpha+\beta)}{\cos^2(\alpha-\beta)-\sin^2(\alpha+\beta)}}. \tag{8}$$

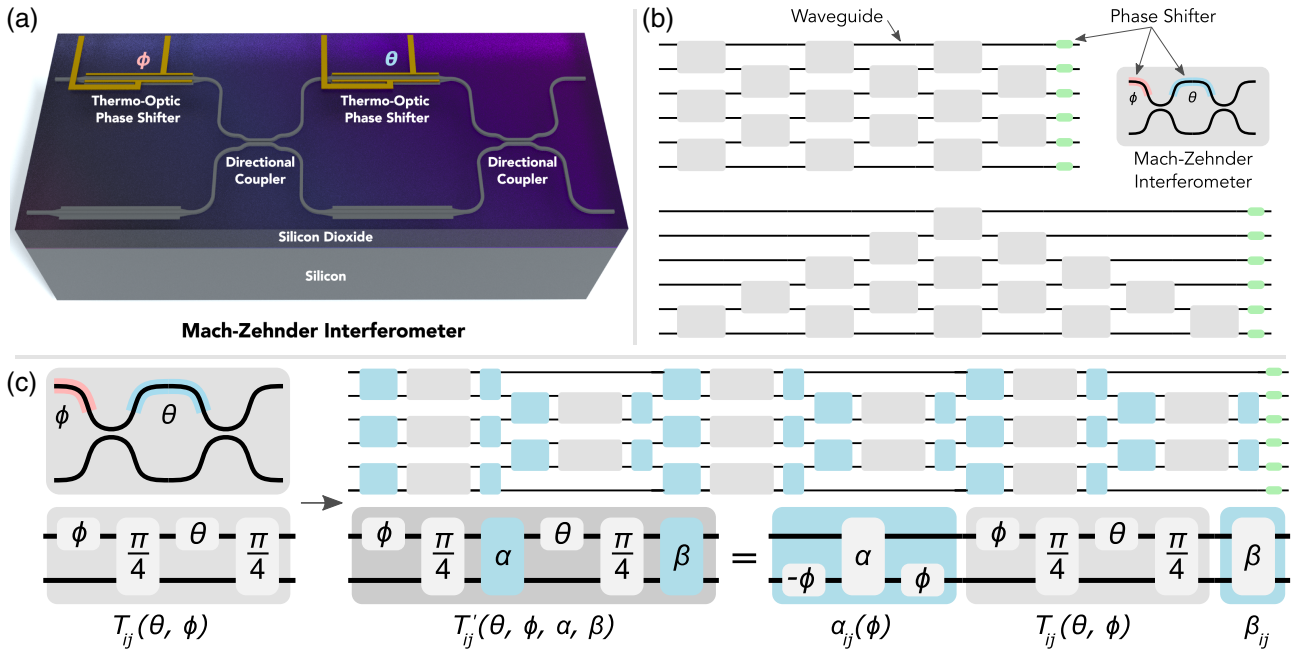

Fig. 1. (a) A Mach–Zehnder interferometer (MZI) on the silicon-on-insulator platform is composed of two 50-50 splitters, implemented with directional couplers, and an external phase shifter ϕ and internal phase shifter θ implemented with thermo-optic phase shifters. These devices act as electrically controlled 2×2 optical gates in programmable photonics. (b) Arbitrary higher-dimensional matrix operations can be implemented by connecting $N(N-1)/2$ MZIs in a rectangular (top) or triangular (bottom) configuration. The Reck (triangular) and Clements (rectangular) decompositions [17,18] describe a procedure for computing the phase settings for each MZI, but they assume the components are ideal. (c) A realistic MZI implemented on a photonics platform will have splitting errors α, β for the two directional couplers within the interferometer. The effect of these hardware errors is to left- and right-multiply each programmable 2×2 unitary $T_{ij}(\theta, \phi)$ implemented by an MZI by error matrices β_{ij}, $\alpha_{ij}(\phi)$. Applying the standard decomposition for ideal components to these imperfect optical gates will not produce the correct gate operation.

Component errors restrict the range over which θ is physically realizable. The above expression has a solution only if $\sin^2(\theta/2) > \sin^2(\alpha+\beta)$ and if $\sin^2(\theta/2) < \cos^2(\alpha-\beta)$. This restricts θ to the range

$$2|\alpha+\beta| < \theta < \pi - 2|\alpha-\beta|. \tag{9}$$

If the matrix decomposition requires θ outside this range, the error is minimized by setting $\theta' = 0$ (if $\theta < 2|\alpha+\beta|$) or $\theta' = \pi$ (if $\theta > \pi - 2|\alpha-\beta|$).

Assuming we can physically implement the required value of θ', the magnitudes of the elements of $T'_{ij}(\theta', \phi', \alpha, \beta)$ and $T_{ij}(\theta, \phi)$ are now the same, but each element of T'_{ij} will have an undesired extraneous phase $\xi_a, \xi_b, \xi_c, \xi_d$ relative to the corresponding term in T_{ij} that must be corrected. We can therefore rewrite $T'_{ij}(\theta', \phi', \alpha, \beta)$ as

$$T'_{ij} = ie^{i\theta'/2}\begin{bmatrix} e^{i\phi'}e^{i\xi_a}\sin(\theta/2) & e^{i\xi_b}\cos(\theta/2) \\ e^{i\phi'}e^{i\xi_c}\cos(\theta/2) & -e^{i\xi_d}\sin(\theta/2) \end{bmatrix} \tag{10}$$

$$= ie^{i\theta'/2}\begin{bmatrix} e^{i\xi_b} & 0 \\ 0 & e^{i\xi_d} \end{bmatrix}\begin{bmatrix} e^{i(\phi'+\xi_a-\xi_b)}\sin(\theta/2) & \cos(\theta/2) \\ e^{i(\phi'+\xi_a-\xi_b)}\cos(\theta/2) & -\sin(\theta/2) \end{bmatrix}, \tag{11}$$

where the simplification in the second line originates from unitarity requiring that $\xi_a + \xi_d = \xi_b + \xi_c$. We correct the phase errors in T'_{ij} by setting $\phi' = \phi + \xi_b - \xi_a$ and by applying additional phases $\psi_1 = -\xi_b + (\theta - \theta')/2$, $\psi_2 = -\xi_d + (\theta - \theta')/2$ to the top and bottom output modes, respectively. Applying these corrections sets $T'_{ij}(\theta', \phi', \alpha, \beta)$ exactly equal to $T_{ij}(\theta, \phi)$.

Expressions for the phase errors ξ_a, ξ_b, ξ_d can be constructed by setting the complex arguments of the elements of T_{ij} equal to those of $T'_{ij}(\theta', \phi', \alpha, \beta)$. From this, we find that

$$\phi' = \phi + \arctan\left[\frac{\sin(\alpha-\beta)}{\cos(\alpha+\beta)}\tan\left(\theta'/2\right)\right] - \arctan\left[\frac{\sin(\alpha+\beta)}{\cos(\alpha-\beta)}\cot\left(\theta'/2\right)\right], \tag{12}$$

$$\psi_1 = -\arctan\left[\frac{\sin(\alpha-\beta)}{\cos(\alpha+\beta)}\tan\left(\theta'/2\right)\right] + \left(\theta-\theta'\right)/2, \tag{13}$$

$$\psi_2 = \arctan\left[\frac{\sin(\alpha+\beta)}{\cos(\alpha-\beta)}\cot\left(\theta'/2\right)\right] + \left(\theta-\theta'\right)/2. \tag{14}$$

The errors $\theta - \theta'$, $\phi' - \phi$, ψ_1, ψ_2 as a function of θ for an example MZI with two 52-48 ($\alpha = \beta = 0.02$) splitters are shown in Fig. 2(b). While the corrections to θ and ψ_1 are small ($\sim$0.1 rad), the errors for ϕ and ψ_2 are quite substantial. In particular, for low device reflectivities ($\theta \approx 0$), the phase corrections required can exceed 1 rad.

Generally, we cannot apply the auxiliary phases ψ_1, ψ_2 locally to the device being corrected, since the output modes do not have phase shifters. In most cases, one of the two can be incorporated into the external phase shifter setting of an MZI in the subsequent column. The other phase can be applied by observing that

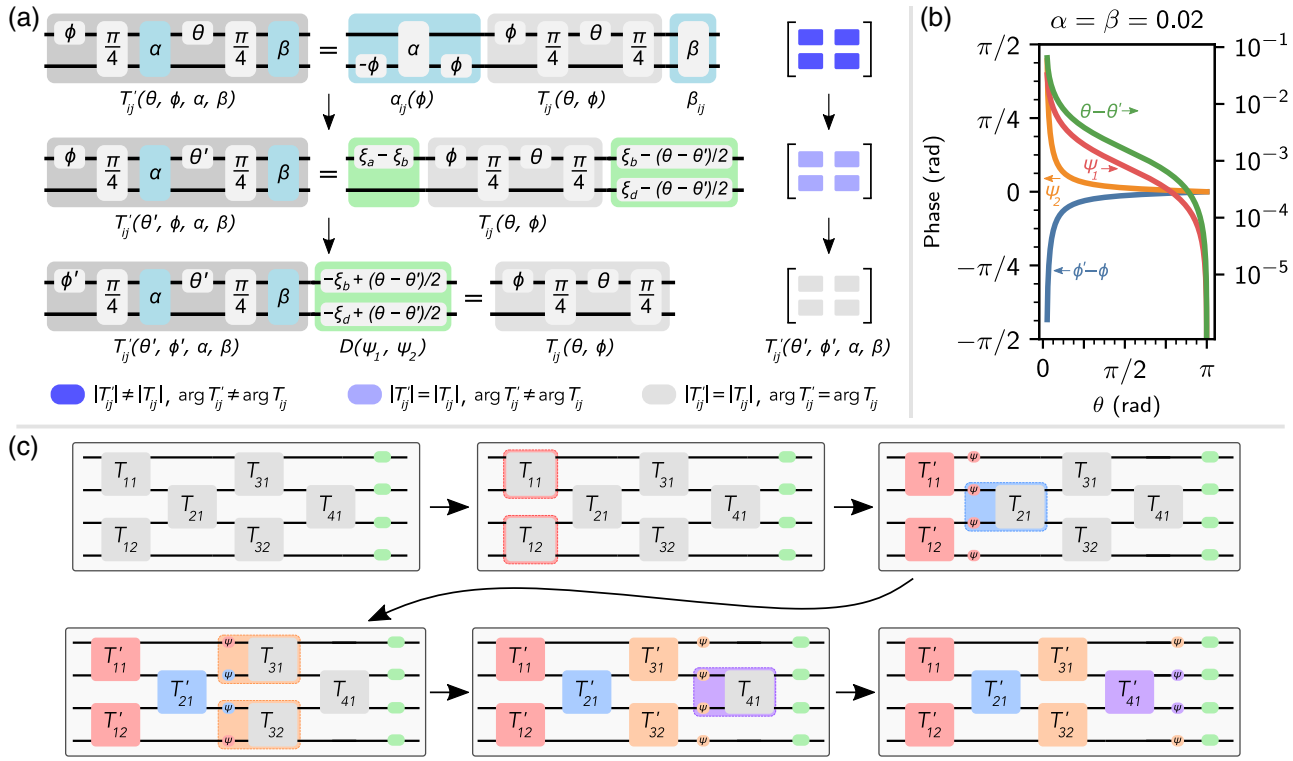

Fig. 2. (a) Fabrication-induced errors within each MZI can be corrected by applying local corrections $\theta \to \theta', \phi \to \phi'$ to the device. We first correct θ to set the magnitudes of the elements of T_{ij} equal to T'_{ij}. Once the amplitude terms are set correctly, we apply phase corrections to the input and outputs of the device to correct phase errors between T_{ij} and T'_{ij}. (b) Corrections $\phi' - \phi, \theta - \theta', \psi_1, \psi_2$ applied to an MZI with two 52-48 beam splitters ($\alpha = \beta = 0.02$). The arrows on the plot indicate which vertical axis each curve corresponds to. (c) Procedure for programming a unitary with hardware errors on a 4×4 rectangular unitary circuit. We first program each MZI to the (θ, ϕ) setting obtained with the standard decomposition in [18]. Each MZI is then converted $T_{ij} \to T'_{ij}$ to the settings for an imperfect device one column at a time. At each step we propagate the output phase shifts ψ_1, ψ_2 forward in the circuit until the entire network is corrected.

$$T_{ij}(\theta, \phi) \begin{bmatrix} e^{i\psi_1} & 0 \\ 0 & e^{i\psi_2} \end{bmatrix} = \begin{bmatrix} e^{i\psi_2} & 0 \\ 0 & e^{i\psi_2} \end{bmatrix} T_{ij}(\theta, \phi + \psi_1 - \psi_2). \tag{15}$$

Using this fact, we can propagate the auxiliary phases forward, through all of the columns of the network, out to the phase shifters D located on the output modes of the circuit. This procedure, illustrated in Fig. 2(c), produces a modified output phase screen D' such that

$$U = D \prod T_{ij}(\theta, \phi) = D' \prod T'_{ij}(\theta', \phi', \alpha, \beta). \tag{16}$$

Depending on the component imperfections and the required value of θ, we may also be able to program θ' such that $|T'_{ij}(\theta', \phi', \alpha, \beta)| = |T_{ij}(\theta, \phi)|$ if the condition in Eq. (9) is satisfied. If every MZI in the circuit satisfies the condition in Eq. (9), we can recover the exact unitary desired. However, if some MZIs in the circuit cannot realize the required splitting, that exact unitary is not physically realizable by the device. In this case, correcting the phases ϕ', ψ_1, ψ_2 and setting θ' as close to the required value as possible minimizes the gate error $\| T_{ij} - T'_{ij} \|$.

We can summarize the algorithm for programming of a matrix U as follows:

(1) Calibrate all phase shifters and splitter errors α, β with the procedure in Supplement 1, Section I and store in a lookup table.
(2) Calculate the required values for θ, ϕ assuming ideal components, using the procedure described by Reck [17] or Clements [18].
(3) For each device, set $\theta \to \theta'$ using the expression in Eq. (8). If $\theta < 2|\alpha + \beta|$, set $\theta' = 0$; if $\theta > \pi - 2|\alpha - \beta|$, set $\theta' = \pi$.
(4) Apply phase corrections ϕ', ψ_1, ψ_2 as given in Eqs. (12)–(14). Propagate ψ_1, ψ_2 forward to the output phase screen D with the expression in Eq. (15).

We have illustrated this procedure for the example of feed-forward unitary circuits, but the same principles apply for other architectures. Each optical gate within any programmable circuit can be corrected to the required 2×2 unitary operation T_{ij} with the aforementioned procedure. The expressions provided assume a specific form for the MZI (Fig. 1), but they can be easily modified to apply to other designs, such as the dual-drive tunable basic unit (TBU) used in recirculating architectures [33].

3. DISCUSSION

A. Hardware Performance

We analyzed the performance of error correction through numerical simulations of programmable photonic circuits with fabrication imperfections. Results were obtained with a custom simulation package written using NumPy [34]. Further details are included in Section V of Supplement 1. Our results assume that the circuits

are unitary (lossless); we have also considered the effect of variable optical losses in hardware in Section IV of Supplement 1.

Figure 3(a) shows the matrix error (relative error per entry) $\epsilon = (\sum_{ij} |U_{\text{hardware, }ij} - U_{ij}|^2/N)^{1/2}$ for 100 Haar random unitaries implemented on 100 randomly generated $N = 32$-mode unitary circuits with mean beam splitter transmission $\eta = (50 \pm \sigma_{\text{BS}})\%$. The beam splitter errors are independently sampled from a Gaussian distribution; for large N, the distribution shape will not greatly affect the results. We find that error correction reduces ϵ significantly, sometimes by more than an order of magnitude. This improvement is larger for circuits with small splitting errors, as they are more likely to satisfy Eq. (9) and program the required θ' for all devices within the circuit. However, even for circuits with large σ_{BS}, where many MZIs may not be programmable to the required θ, the improvement in ϵ is substantial, as all errors in ϕ, ψ_1, ψ_2 can always be corrected.

In Fig. 3(b), we show ϵ with and without error correction for circuit sizes $N = \{64, 128, 256\}$. For these simulations, we chose a beam splitter variation of $\sigma_{BS} = 2\%$, which is a typical wafer-level variance [19]. While the improvement in ϵ diminishes for larger N, we still find substantial improvement gained in our approach for up to 256 modes. For large unitary circuits, most MZIs need to be programmed to reflectivities close to $\theta \approx 0$ [35]; the increasing fraction of devices that cannot be programmed to the required splitting accounts for the increase in ε with N. Nevertheless, there is always some improvement in ϵ, as any phase errors introduced by the components can be corrected. Our results suggest that substantial performance improvements can still be achieved by error correction for circuits with hundreds of modes, which is well beyond the size of the current state of the art ($N = 64$) in programmable photonics [36].

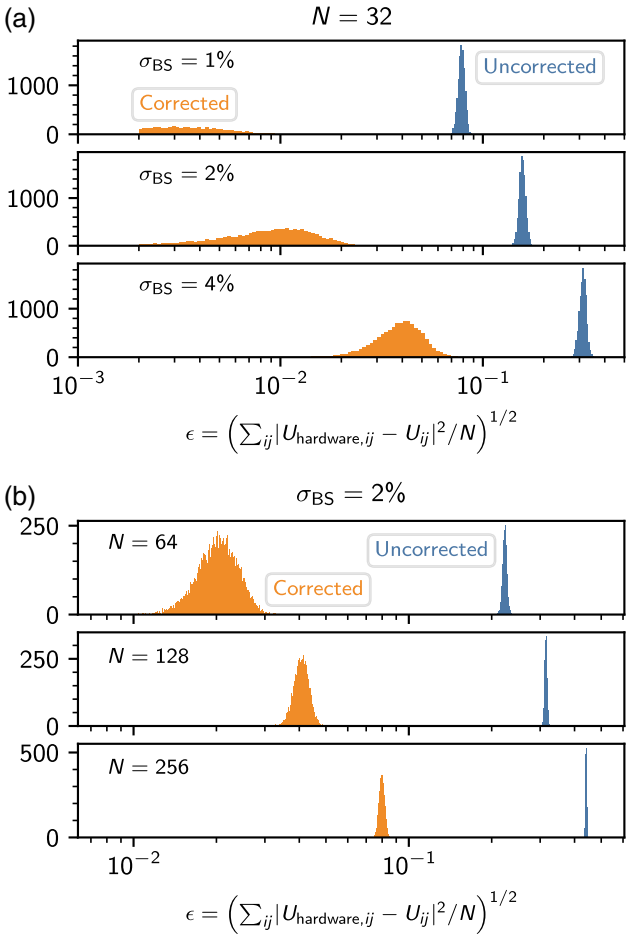

Fig. 3. (a) Matrix error ϵ before and after correction for 100 random unitaries implemented on 100 random circuits with varying beam splitter statistics. (b) Matrix error ε before and after correction for $N = \{64, 128, 256\}$ with a beam splitter variation $\sigma_{BS} = 2\%$.

B. Application: Optical Neural Networks on Feedforward Programmable Circuits

To further benchmark the performance of our error correction protocol, we applied this approach to simulations of a programmable photonic system, namely, a two-layer neural network conducting inference with a feedforward programmable photonic circuit. The architecture of the neural network is similar to that studied in [16,20,32], where forward inference is optically computed through passive interference within a unitary photonic circuit coupled with an electrical or electro-optic nonlinearity [37]. Optical machine learning is a key application area for photonic error correction, as model training is both time consuming and energy intensive, making it impractical to retrain on each individual piece of hardware with a unique set of fabrication errors. Preferably, a model would be highly optimized once in software, after which corrections are applied within the hardware to restore the original software-trained model from any fabrication-induced errors.

The neural networks we benchmark are based on the architecture described in [32]. Using the Neurophox package, we trained two-layer neural networks with $N = \{36, 64, 144, 256\}$ neurons to recognize low-frequency Fourier features of handwritten digits from the MNIST task. The activation function between layers was assumed to be a modReLU function implemented using an electro-optic nonlinearity [37,38]. Further details on the network architecture and training are included in Supplement 1.

Figure 4 shows the median classification accuracy for 300 randomly generated circuits as a function of the beam splitter statistics $\eta = (50 \pm \sigma_{\text{BS}})\%$. The smaller circuits ($N = 36, 64$) exhibit roughly 95%–96% accuracy after training, while the larger circuits ($N = 144, 256$) exhibit a slightly higher model accuracy of ∼97%. The larger circuits, however, are less resilient to errors; without error correction, classification accuracy drops to below 90% for all circuit sizes at a splitter variation as low as ∼3%.

Hardware error correction extends this cutoff to more than 6%, which is well beyond modern-day process tolerances [19]. Moreover, without correction the classification accuracy drops significantly at even typical wafer-level variances (2%). However, with error correction there is almost no drop in accuracy at these variances and less than 1% accuracy loss for beam splitter variations as high as 4%. We expect this margin for fabrication error will prove important as optical neural networks scale up. These results suggest that error correction in programmable photonics can enable high-accuracy neural networks of up to hundreds of modes within current-day process tolerances.

C. Application: Tunable Dispersion Compensators on Recirculating Waveguide Meshes

While our analysis has focused on feedforward programmable photonic meshes, our results can also be applied to recirculating architectures useful in RF and optical signal processing. These recirculating meshes, which are usually configured in hexagonal or triangular lattices, enable implementation of finite impulse

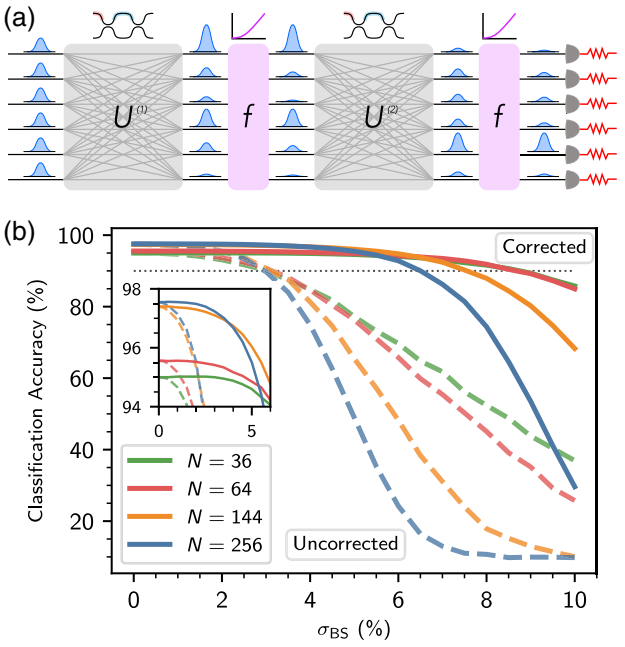

Fig. 4. (a) We simulated the effect of component error on two-layer optical neural networks for the MNIST task. Matrix–vector products are calculated optically in the photonic circuit, and modReLU-like activation functions are implemented electro-optically [37,38]. (b) Median accuracy for 300 unitary circuits as a function of σ_{BS} with and without correction for a photonic image classifier for the MNIST task with $N = \{36, 64, 144, 256\}$ neurons. Error correction significantly improves the fabrication tolerance of the neural network to beyond current-day process tolerances, even for systems with hundreds of modes. As the inset shows, even circuits with 4% splitter error preserve the baseline performance within 1%.

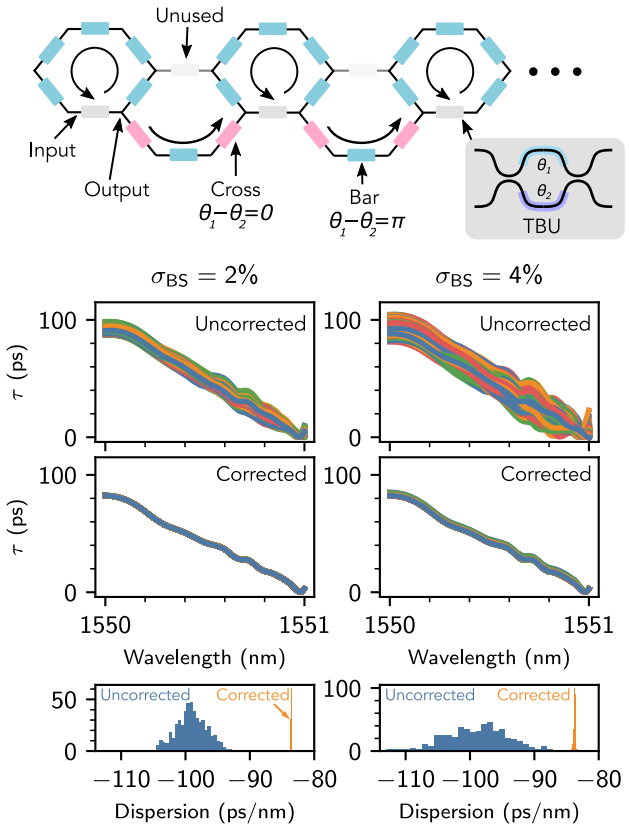

Fig. 5. Top: simulations of a tunable dispersion compensator (TDC) implemented on a recirculating waveguide mesh with 15 tunable-coupling ring resonators coupled serially to one another. Bottom: after training the mesh parameters to implement a fixed linear group delay dispersion on an ideal model, small beam splitter errors will introduce variations in the implemented group delay τ profile. Plotted are the group delay profiles for 500 randomly generated circuits before and after correction. Correcting the settings of each TBU restores the desired performance, eliminating the need to retrain on the hardware. Also displayed is the distribution of the group delay dispersion before and after correction.

response (FIR) and infinite impulse response (IIR) filters by configuring waveguides into asymmetric MZIs and ring resonators, respectively [13,21,22]. Unlike the feedforward architectures, the programming of these structures usually cannot be determined analytically and must be found through optimization [26–28]. Since optimization can be time consuming for complex systems, error correction can enable optimizing these circuit parameters on idealized models and then porting them over to hardware without retraining. As an example, we simulated the performance of an IIR filter functioning as a tunable dispersion compensator (TDC) on a hexagonal waveguide lattice [22]. TDC modules are of interest for numerous applications, including compensating for chromatic dispersion in optical communication links [39] and enabling high-dimensional quantum key distribution (QKD) with temporal modes [40].

We implemented the TDC using an architecture similar to the tunable-coupling ring array described in [14]. Programmable dispersion is achieved by individually tuning the coupling and resonance of each ring in a chain of 15 resonators coupled serially to one another. Each ring is implemented with a single MZI (often referred to as the TBU) in a hexagonal mesh acting as the coupler, while five other TBUs are programmed to the bar state to implement feedback. For simplicity we do not simulate routing within the hexagonal mesh, but instead simulate the transfer function of each individual filter implemented using TBUs with fabrication imperfections. Using the constrained optimization by linear approximations (COBYLA) routine in SciPy [41,42], we trained the TBU parameters on an idealized model to implement a group delay dispersion of −85 ps/nm over the bandwidth of a 50 GHz channel.

Figure 5 shows the group delay τ profiles for 500 randomly generated TDC modules implemented using TBUs with $\sigma_{BS} = \{2, 4\}\%$ before (top) and after (bottom) error correction. Similar to optical neural networks, precise implementation of a TDC requires accurate phase control throughout the circuit. Fabrication errors introduce spurious phases at each resonance, which results in significant variation of the dispersion profile for even slight component errors. As our results show, correcting the parameters of each TBU locally is sufficient to restore the desired dispersion profile.

While we can correct the coupling and phase parameters for each ring, we cannot correct for errors in the closed feedback loop, which is implemented by programming each TBU to the bar state. Any error $\alpha \neq \beta$ will introduce some loss at each TBU programmed to the bar state, as the bar transmission is reduced to $\cos^2(\alpha - \beta)$. The remainder of the light is directed into unused couplers in the circuit, effectively incurring loss. This alters the critical coupling condition, resulting in the slight spread in the corrected dispersion profile observed in our simulations for $\sigma_{BS} = 4\%$.

Our simulations assume α, β are independent, Gaussian random variables; in practice, however, α, β for a single device are strongly correlated [43,44] and the bar state will be nearly perfect. Therefore, our simulations likely overestimate the loss incurred at each TBU programmed to the bar state.

D. Scalability and Outlook

We have presented an approach for characterizing and correcting for hardware errors in programmable photonic circuits. To conclude, we analyze the expected improvement our technique enables and how it will perform as these circuits scale up.

For a unitary photonic circuit, applying the Reck or Clements decomposition produces an average matrix error ε of (Supplement 1, Section II.A)

$$\langle \epsilon \rangle \approx \sigma_{BS}\sqrt{2(N-1)}. \tag{17}$$

If we can correct all errors in θ, then $\epsilon_{\text{corrected}} \to 0$. We can therefore estimate the expected $\epsilon_{\text{corrected}}$ by computing the fraction of MZIs that cannot be programmed to the required splitting value, i.e., the condition in Eq. (9).

The distribution of phase shifter settings for a unitary circuit can be related to the Haar measure on the unitary group [35]. The probability that an MZI is programmed to a value $\theta < \xi$ is (Supplement 1, Section II.C)

$$P(\theta < \xi) = \sum_{k=1}^{N-1} \frac{2(N-k)}{N(N-1)} \left(1 - \cos^{2k}(\xi/2)\right) \tag{18}$$

$$\approx \frac{N+1}{12}\xi^2. \tag{19}$$

We disregard the probability that an MZI is programmed to a splitting $\theta > \pi - 2|\alpha - \beta|$, which is negligibly small for large N [35]. Error correction cannot fix the splitting error if $\theta < 2|\alpha + \beta|$; therefore,

$$\langle \epsilon_{\text{corrected}} \rangle \approx \left(\frac{1}{2} P(\theta < 2|\alpha+\beta|) \langle \varepsilon^2 \rangle\right)^{1/2} \tag{20}$$

$$= \sigma_{\text{BS}}^2 \sqrt{\frac{2\left(N^2-1\right)}{3}}. \tag{21}$$

We find that error correction effectively reduces the hardware error from ϵ to $\approx (1/\sqrt{6})\epsilon^2$. The expected error improvement is

$$\frac{\langle \epsilon \rangle}{\langle \epsilon_{\text{corrected}} \rangle} \approx \frac{\sqrt{3}}{\sigma_{\text{BS}}\sqrt{N+1}}. \tag{22}$$

$\langle \epsilon \rangle$ and $\langle \epsilon_{\text{corrected}} \rangle$ as a function of N are plotted in Fig. 6(a). We consider $\sigma_{\text{BS}} = 1.2\%$, which is the state of the art reported in [19], as well as more relaxed tolerances $\sigma_{\text{BS}} = \{2, 4\}\%$. For σ_{BS} as high as 4%, error correction produces at least a factor of two (and often more) improvement in the error for circuits as large as $N = 500$. We therefore expect our approach to have wide applicability in the near term as the size of programmable photonic circuits scale up.

Error correction also greatly improves the optical bandwidth of unitary circuits. Since directional couplers are highly wavelength sensitive, dense wavelength-division multiplexing (DWDM) requires re-fabricating the same circuit with components optimized at each wavelength channel. Our approach, however, enables the use of the same hardware across a wide wavelength range. In Fig. 6(b), we show the expected hardware errors for large circuits across a 100 nm bandwidth using the optimal splitter ($\sigma_{\text{BS}} = 1.2\%$) design in [19]. We find that the corrected error for an $N = 256$ circuit across a 60 nm bandwidth (1520–1580 nm) will be lower than the *uncorrected* error at the design wavelength $\lambda = 1550$ nm. Even lower errors could be achieved using multimode interferometer (MMI) couplers; these devices have large bandwidths but often suffer from static splitting imbalances [45], i.e., α, β are invariant to wavelength, but $\langle \alpha \rangle$, $\langle \beta \rangle \neq 0$. A circuit with large-bandwidth MMI couplers can thus use error correction to achieve a large instantaneous bandwidth, for instance to compute over many parallel wavelength channels.

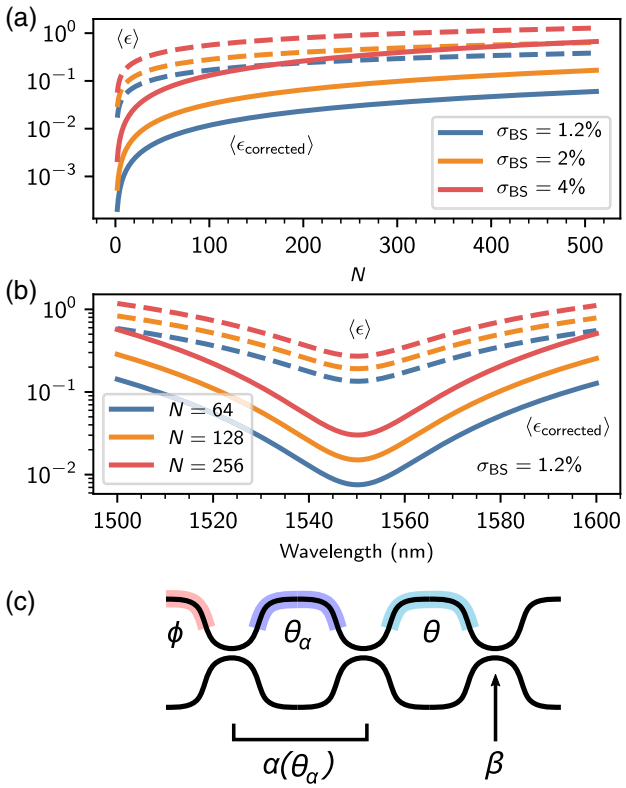

Fig. 6. (a) $\langle \epsilon \rangle$, $\langle \epsilon_{\text{corrected}} \rangle$ as a function of circuit size N for $\sigma_{\text{BS}} = \{1.2, 2, 4\}\%$. (b) Average circuit error as a function of wavelength for $N = \{64, 128, 256\}$ using the optimal directional coupler design in [19]. (c) Redundant MZI for implementing perfect optical gates. One of the two beam splitters is an MZI that can be tuned to implement an error $\alpha(\theta_\alpha)$ that compensates for an error β.

The results in Fig. 6(a) suggest a fundamental error bound achievable with local correction for unitary circuits. Our approach yields results comparable to those achieved with self-configuration procedures [9,32] but does not require a specific structure for the circuit or photodiodes within each device. If the condition in Eq. (9) is satisfied, local correction obtains $\epsilon_{\text{corrected}} = 0$ in $O(1)$ time. If this condition is not satisfied, it is sometimes possible to achieve a larger reduction in error with a *global* optimization approach [24,29]. However, these approaches, which require photodiodes within each device or output measurements whose number scales nonlinearly with the number of modes, become increasingly inaccessible experimentally as N scales up. Local correction requires minimal overhead and can guarantee a minimum error given certain guarantees on the component performance,

making it ideal for standardizing performance across large numbers of chips.

Moreover, this error bound applies only to feedforward, unitary circuits with no redundant devices. ε lower than this bound can be achieved by incorporating additional, redundant MZIs; for instance, one can implement "perfect" optical gates by incorporating an additional phase shifter into the MZI, as shown in Fig. 6(c). This device can be trained with optimization to implement any desired unitary $T_{ij}(\theta, \phi)$ perfectly [46,47]. The error correction formalism enables calculation of these settings analytically. One of the two constituent splitters is a passive component with error β, while the other splitter is an MZI that implements a tunable error $\alpha(\theta_\alpha)$. Any desired 2×2 unitary with a required splitting θ can then be implemented by setting θ_α such that $2|\alpha(\theta_\alpha) + \beta| < \theta < 2|\alpha(\theta_\alpha) - \beta|$ and correcting the resultant phase errors (Supplement 1, Section III.B).

Not all optical gates within the circuit necessarily need to incorporate redundancy. High accuracy unitary circuits have been demonstrated by incorporating only a few extra MZIs into the circuit, which can be trained using nonlinear optimization [24] or gradient descent [29]. Error correction serves an important purpose for these circuits, as one can optimize the hardware settings once on an ideal model and port the settings over to many devices. For recirculating meshes the phase shifter settings are not constrained by the Haar measure, and so the benefit gained from error correction is not expected to diminish with increasing N. We therefore expect error correction to play an important role in scaling up the size of these circuits as well.

The motivation for photonic error correction assumes the hardware is re-programmed infrequently, for instance to implement a weight matrix in a neural network. Other applications, such as mode unscrambling, require real-time configuration robust to device error. We have recently discussed error-resilient self-configuration approaches in [48,49].

4. CONCLUSION

In conclusion, we have presented a protocol to correct for hardware errors in programmable photonic circuits. Unlike optimization-based approaches, our protocol utilizes a one-time calibration procedure to flexibly implement any desired functionality up to the limits of the hardware. We find that applying our approach to key application areas of programmable photonics, such as optical neural networks and programmable coupled-ring systems, enables resilience to fabrication errors well beyond modern-day process tolerances. Error correction also greatly reduces the overhead for programmable photonics that require optimization to deduce the hardware settings, as it eliminates the need to retrain for each individual set of hardware with unknown fabrication errors. Current process tolerances suggest that our approach enables improved functionality for systems of up to hundreds of modes, providing a new avenue for scaling up programmable photonics.

Note: after submission of this manuscript, a related work on error correction [50] was posted to the arXiv.

Funding. National Science Foundation (1745302); Air Force Office of Scientific Research (FA9550-16-1-0391, FA9550-20-1-0113); Intelligence Community Postdoctoral Research Fellowship Program.

Acknowledgment. The authors are grateful to Hugo Larocque and Alexander Sludds for helpful comments on the manuscript.

Disclosures. SB, RH, DE: US Provisional Patent Applications 63/151,103, 63/196,301 (P).

Data availability. The data that support the plots in this paper are available from the corresponding author upon reasonable request.

Supplemental document. See Supplement 1 for supporting content.

REFERENCES

1. W. Bogaerts, D. Pérez, J. Capmany, D. A. B. Miller, J. Poon, D. Englund, F. Morichetti, and A. Melloni, "Programmable photonic circuits," Nature **586**, 207–216 (2020).
2. 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**, 1623–1631 (2018).
3. 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**, 447–452 (2017).
4. J. Wang, S. Paesani, Y. Ding, R. Santagati, P. Skrzypczyk, A. Salavrakos, J. Tura, R. Augusiak, L. Mančinska, D. Bacco, D. Bonneau, J. W. Silverstone, Q. Gong, A. Acín, K. Rottwitt, L. K. Oxenløwe, J. L. O'Brien, A. Laing, and M. G. Thompson, "Multidimensional quantum entanglement with large-scale integrated optics," Science **360**, 285–291 (2018).
5. X. Qiang, X. Zhou, J. Wang, C. M. Wilkes, T. Loke, S. O'Gara, L. Kling, G. D. Marshall, R. Santagati, T. C. Ralph, J. B. Wang, J. L. O'Brien, M. G. Thompson, and J. C. F. Matthews, "Large-scale silicon quantum photonics implementing arbitrary two-qubit processing," Nat. Photonics **12**, 534–539 (2018).
6. C. Sparrow, E. Martín-López, N. Maraviglia, A. Neville, C. Harrold, J. Carolan, Y. N. Joglekar, T. Hashimoto, N. Matsuda, J. L. O'Brien, D. P. Tew, and A. Laing, "Simulating the vibrational quantum dynamics of molecules using photonics," Nature **557**, 660–667 (2018).
7. J. Carolan, C. Harrold, C. Sparrow, E. Martin-Lopez, 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**, 711–716 (2015).
8. D. A. B. Miller, "Self-configuring universal linear optical component," Photon. Res. **1**, 1–15 (2013).
9. D. A. B. Miller, "Self-aligning universal beam coupler," Opt. Express **21**, 6360–6370 (2013).
10. A. Annoni, E. Guglielmi, M. Carminati, G. Ferrari, M. Sampietro, D. A. Miller, A. Melloni, and F. Morichetti, "Unscrambling light-automatically undoing strong mixing between modes," Light Sci. Appl. **6**, e17110 (2017).
11. A. Ribeiro, A. Ruocco, L. Vanacker, and W. Bogaerts, "Demonstration of a 4×4-port universal linear circuit," Optica **3**, 1348–1357 (2016).
12. M. Milanizadeh, P. Borga, F. Morichetti, D. Miller, and A. Melloni, "Manipulating free-space optical beams with a silicon photonic mesh," in *IEEE Photonics Society Summer Topical Meeting Series (SUM)* (2019), pp. 1–2.
13. L. Zhuang, C. G. H. Roeloffzen, M. Hoekman, K.-J. Boller, and A. J. Lowery, "Programmable photonic signal processor chip for radiofrequency applications," Optica **2**, 854–859 (2015).
14. J. Notaros, J. Mower, M. Heuck, C. Lupo, N. C. Harris, G. R. Steinbrecher, D. Bunandar, T. Baehr-Jones, M. Hochberg, S. Lloyd, and D. Englund, "Programmable dispersion on a photonic integrated circuit for classical and quantum applications," Opt. Express **25**, 21275–21285 (2017).
15. M. Prabhu, C. Roques-Carmes, Y. Shen, N. Harris, L. Jing, J. Carolan, R. Hamerly, T. Baehr-Jones, M. Hochberg, V. Čeperić, J. D. Joannopoulos, D. R. Englund, and M. Soljačić, "Accelerating recurrent Ising machines in photonic integrated circuits," Optica **7**, 551–558 (2020).
16. 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**, 441–446 (2017).

17. M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental realization of any discrete unitary operator," Phys. Rev. Lett. **73**, 58–61 (1994).
18. W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walsmley, "Optimal design for universal multiport interferometers," Optica **3**, 1460–1465 (2016).
19. J. C. Mikkelsen, W. D. Sacher, and J. K. S. Poon, "Dimensional variation tolerant silicon-on-insulator directional couplers," Opt. Express **22**, 3145–3150 (2014).
20. 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**, 14009–14029 (2019).
21. D. Pérez, I. Gasulla, and J. Capmany, "Field-programmable photonic arrays," Opt. Express **26**, 27265–27278 (2018).
22. D. Pérez, I. Gasulla, J. Capmany, and R. A. Soref, "Reconfigurable lattice mesh designs for programmable photonic processors," Opt. Express **24**, 12093–12106 (2016).
23. I. Zand and W. Bogaerts, "Effects of coupling and phase imperfections in programmable photonic hexagonal waveguide meshes," Photon. Res. **8**, 211–218 (2020).
24. R. Burgwal, W. R. Clements, D. H. Smith, J. C. Gates, W. S. Kolthammer, J. J. Renema, and I. A. Walmsley, "Using an imperfect photonic network to implement random unitaries," Opt. Express **25**, 28236–28245 (2017).
25. J. Mower, N. C. Harris, G. R. Steinbrecher, Y. Lahini, and D. Englund, "High-fidelity quantum state evolution in imperfect photonic integrated circuits," Phys. Rev. A **92**, 032322 (2015).
26. A. López, D. Pérez, P. DasMahapatra, and J. Capmany, "Auto-routing algorithm for field-programmable photonic gate arrays," Opt. Express **28**, 737–752 (2020).
27. D. Pérez-López, "Programmable integrated silicon photonics waveguide meshes: optimized designs and control algorithms," IEEE J. Sel. Top. Quantum Electron. **26**, 1–12 (2020).
28. D. Pérez-López, A. Lépez, P. DasMahapatra, and J. Capmany, "Multipurpose self-configuration of programmable photonic circuits," Nat. Commun. **11**, 6359 (2020).
29. S. Pai, B. Bartlett, O. Solgaard, and D. A. B. Miller, "Matrix optimization on universal unitary photonic devices," Phys. Rev. Appl. **11**, 064044 (2019).
30. T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, "Training of photonic neural networks through in situ backpropagation and gradient measurement," Optica **5**, 864–871 (2018).
31. D. A. B. Miller, "Setting up meshes of interferometers–reversed local light interference method," Opt. Express **25**, 29233–29248 (2017).
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**, 1–13 (2020).
33. D. Pérez-López, A. M. Gutierrez, E. Sánchez, P. DasMahapatra, and J. Capmany, "Integrated photonic tunable basic units using dual-drive directional couplers," Opt. Express **27**, 38071–38086 (2019).
34. C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," Nature **585**, 357–362 (2020).
35. 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).
36. N. C. Harris, R. Braid, D. Bunandar, J. Carr, B. Dobbie, C. Dorta-Quinones, J. Elmhurst, M. Forsythe, M. Gould, S. Gupta, S. Kannan, T. Kenney, G. Kong, T. Lazovich, S. Mckenzie, C. Ramey, C. Ravi, M. Scott, J. Sweeney, O. Yildirim, and K. Zhang, "Accelerating artificial intelligence with silicon photonics," in *Optical Fiber Communication Conference (OFC)* (OSA, 2020), paper W3A.3.
37. I. A. D. Williamson, T. W. Hughes, M. Minkov, B. Bartlett, S. Pai, and S. Fan, "Reprogrammable electro-optic nonlinear activation functions for optical neural networks," IEEE J. Sel. Top. Quantum Electron. **26**, 1–12 (2020).
38. M. Arjovsky, A. Shah, and Y. Bengio, "Unitary evolution recurrent neural networks," in *Proceedings of the 33rd International Conference on Machine Learning*, Vol. **48** of Proceedings of Machine Learning Research (PMLR, 2016), pp. 1120–1128.
39. C. K. Madsen and G. Lenz, "Optical all-pass filters for phase response design with applications for dispersion compensation," IEEE Photon. Technol. Lett. **10**, 994–996 (1998).
40. J. Mower, Z. Zhang, P. Desjardins, C. Lee, J. H. Shapiro, and D. Englund, "High-dimensional quantum key distribution using dispersive optics," Phys. Rev. A **87**, 062322 (2013).
41. M. J. D. Powell, "Direct search algorithms for optimization calculations," Acta Numer. **7**, 287–336 (1998).
42. P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vanderplas, D. Laxalde, J. Perktold, and R. Cimr, "SciPy 1.0: fundamental algorithms for scientific computing in Python," Nat. Methods **17**, 261–272 (2020).
43. Y. Yang, Y. Ma, H. Guan, Y. Liu, S. Danziger, S. Ocheltree, K. Bergman, T. Baehr-Jones, and M. Hochberg, "Phase coherence length in silicon photonic platform," Opt. Express **23**, 16890–16902 (2015).
44. Z. Lu, J. Jhoja, J. Klein, X. Wang, A. Liu, J. Flueckiger, J. Pond, and L. Chrostowski, "Performance prediction for silicon photonics integrated circuits with layout-dependent correlated manufacturing variability," Opt. Express **25**, 9712–9733 (2017).
45. H. Guan, Y. Ma, R. Shi, X. Zhu, R. Younce, Y. Chen, J. Roman, N. Ophir, Y. Liu, R. Ding, T. Baehr-Jones, K. Bergman, and M. Hochberg, "Compact and low loss 90° optical hybrid on a silicon-on-insulator platform," Opt. Express **25**, 28957–28968 (2017).
46. K. Suzuki, G. Cong, K. Tanizawa, S.-H. Kim, K. Ikeda, S. Namiki, and H. Kawashima, "Ultra-high-extinction-ratio 2 — 2 silicon optical switch with variable splitter," Opt. Express **23**, 9086–9092 (2015).
47. M. Wang, A. Ribero, Y. Xing, and W. Bogaerts, "Tolerant, broadband tunable 2×2 coupler circuit," Opt. Express **28**, 5555–5566 (2020).
48. R. Hamerly, S. Bandyopadhyay, and D. Englund, "Stability of self-configuring large multiport interferometers," arXiv:2106.04363 (2021).
49. R. Hamerly, S. Bandyopadhyay, and D. Englund, "Accurate self-configuration of rectangular multiport interferometers," arXiv:2106.03249 (2021).
50. S. P. Kumar, L. Neuhaus, L. G. Helt, H. Qi, B. Morrison, D. H. Mahler, and I. Dhand, "Mitigating linear optics imperfections via port allocation and compilation," arXiv:2103.03183 (2021).