

PHOTONICS Research

Generalized robust training scheme using genetic algorithm for optical neural networks with imprecise components

RUI SHAO,¹ GONG ZHANG,^{1,2}  AND XIAO GONG^{1,3}

¹Department of Electrical & Computer Engineering, National University of Singapore, Singapore, Singapore

²e-mail: zhanggong@nus.edu.sg

³e-mail: elegong@nus.edu.sg

Received 3 December 2021; revised 2 March 2022; accepted 18 March 2022; posted 18 March 2022 (Doc. ID 449570); published 22 July 2022

One of the pressing issues for optical neural networks (ONNs) is the performance degradation introduced by parameter uncertainties in practical optical components. Hereby, we propose a novel two-step *ex situ* training scheme to configure phase shifts in a Mach–Zehnder-interferometer-based feedforward ONN, where a stochastic gradient descent algorithm followed by a genetic algorithm considering four types of practical imprecisions is employed. By doing so, the learning process features fast convergence and high computational efficiency, and the trained ONN is robust to varying degrees and types of imprecisions. We investigate the effectiveness of our scheme by using practical machine learning tasks including *Iris* and MNIST classifications, showing more than 23% accuracy improvement after training and accuracy (90.8% in an imprecise ONN with three hidden layers and 224 tunable thermal-optic phase shifters) comparable to the ideal one (92.0%). © 2022 Chinese Laser Press

<https://doi.org/10.1364/PRJ.449570>

1. INTRODUCTION

Implementation of neuromorphic photonics on a silicon photonic integrated chip is gradually becoming a promising technology for deep learning accelerators, which utilizes photonic processors to function as artificial intelligence (AI) cores [1–4]. The realization and advancement of integrated programmable photonic processors [5–13] provide a feasible strategy for the construction of optical neural networks (ONNs) [14,15]. Compared to electronics, neuromorphic photonics has advantages of well-known high-bandwidth, and ultralow energy consumption due to negligible energy for light propagation with encoded information [16]. With the rapid advancement of the complementary metal–oxide–semiconductor (CMOS)-compatible silicon-on-insulator (SOI) platform [17–19], integrated silicon waveguides [20] and optical modulators such as Mach–Zehnder interferometers (MZIs) [21–26] and micro-ring resonators (MRRs) [27–29] can be easily formed as programmable processors for the construction of integrated ONNs [21,22,30,31] and other similar deep learning networks such as convolutional neural networks (CNNs) [6,32,33] and recurrent neural networks (RNNs) [2,34].

However, there remain challenges in the precise control of device performance and achieving excellent uniformity for various components in neuromorphic photonic chips. For example, there is a 15% reduction of vowel classification accuracy with the nanophotonic processor [22] and limited accuracy

(about 88%) in handwriting image recognition using the photonic CNN chip [33]. The major problem is non-ideal photonic components, which leads to uncertain performance of the required functionality. In previous research, a few optimization procedures have been reported aiming at restoring the fidelity of the unitary matrix by using numerical initialization of parameters in MZIs [35,36]. However, these strategies mainly focused on the fidelity of the implemented unitary matrix instead of the desired functionality of the ONN. Hence, the effects of imprecise components could be underestimated. Also, these optimizations require precise characterization of each device separately and are actualized after fabrication, leading to extra computational power consumption and suffering from scalability problems in mass production.

Other methods adopted physical architecture modification to mitigate the effects of imprecisions. A double MZI configuration was proposed to compensate for fabricated MZIs with imperfect splitting ratios without calibration [37]. Shokraneh *et al.* [38] reported a diamond mesh of MZIs, which forms a symmetrical architecture to resist imprecisions in the ONN, and Fang *et al.* [39] demonstrated an FFTGrid architecture that has lower sensitivity to imprecisions. However, these methods require additional cascaded MZIs or waveguide crossings to interconnect MZIs, increasing the size of optical programmable processors and structure complexity. *In situ* gradient training [40] and self-configuration of rectangle MZIs [41–43] are promising in principle, but they suffer from complex

experimental measurements and the limitation of specific network structures.

To address the above-mentioned issues, in this paper, we propose a two-step *ex situ* training scheme in an MZI-based ONN. The first stochastic gradient descent algorithm training step aims to obtain optimal phase shift settings in perfect MZIs for the benefit of fast convergence, while the goal of the following genetic algorithm (GA) training step is to find the optimal configurations considering parameter imprecisions in MZIs. In doing so, updated phase shifts can be realized with the ability to maintain classification accuracy in imprecise ONNs, and hyper-parameters in the GA step can be modified according to varying degrees and types of imprecisions. The conventional *ex situ* gradient training method is done on an idealized ONN. After training, the optimal weights are applied to the physical device. Without error corrections, the performance of the ONN would be severely degraded. Our two-step *ex situ* training based on a gradient algorithm and GA can improve the overall performance of the ONN with practical errors. This *ex situ* training method requires only characterizations of MZI samples, and one-time robust training of the ONN can be used on a batch of chips, similar to the use of deep learning. The model is trained by servers, and then these parameters can be used in local hardware. In a mass production scenario, while it is not possible to test and train each of the chips, our *ex situ* method with GA training provides a cost-effective way.

We perform the training scheme in a feedforward photonic neural network implemented by the mesh of MZIs with

tunable thermo-optic phase shifters and demonstrate its effectiveness in practical learning tasks, including *Iris* [44] and MNIST [45] classifications. With the advantages of robustness, parallelization, and black-box optimization in multi-objective GA, our proposed scheme provides an efficient and generalized training method for imprecision-resistant optical neuromorphic computing platforms.

2. ONN ARCHITECTURE AND TRAINING SCHEME

A. Constructions of ONN

The typical ONN is a feedforward sequential processing flow comprising an input layer with artificial neurons, a series of hidden layers, activation layers, and an output layer, as shown in Fig. 1(a). The continuous wave laser source and optical amplifier generate an optical signal and split it into different waveguide channels. The input image is encoded into the optical signal in the form of $A \exp(j\theta)$ using optical attenuators and modulators, where A and θ are the amplitude and phase of the signal, respectively. Then, signals go through the optical interference unit (OIU) and optical nonlinear unit (ONU). After propagating in the network, optical output signals are converted to electrical signals by using photodetectors for the subsequent information storage and processing. Here we use an *ex situ* (software-based) training method to update neurons. The elements to be trained in the ONN chip are phase shifts in each MZI controlled by voltage settings.

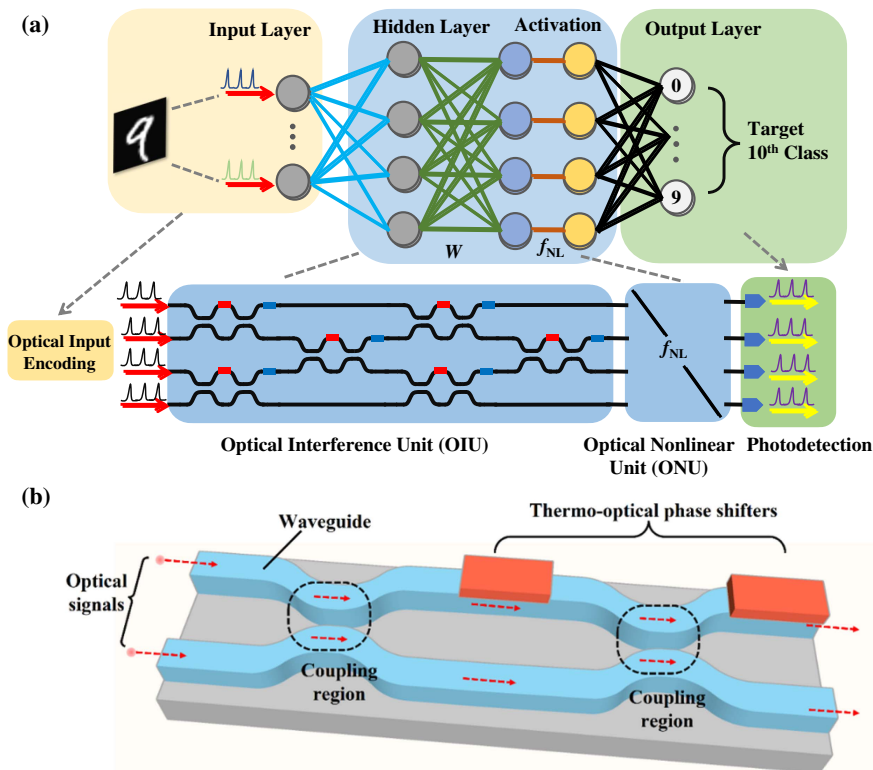


Fig. 1. (a) Illustration of artificial neural network (ANN) architecture for image recognition implemented by photonic units, including optical input encoding parts, optical interference units, optical nonlinear units, and photodetectors. (b) Demonstration of a programmable Mach-Zehnder interferometer consisting of directional couplers and thermo-optical phase shifters.

The paramount part in the ONN is function as the synaptic weight W is the OIU, which can realize the multiply–accumulate operation (MAC) of input optical signals depicted in Fig. 1(a). It constructs a programmable MAC block from N input modes to N output modes. The operation block can be decomposed into a mesh of MZIs, as demonstrated in Fig. 1(b). Each MZI consists of two phase shifters parameterized (θ, ϕ) and two 50:50 directional couplers. The modulation of phase shifters is implemented by tuning the temperature of the rib waveguides according to the thermal-optic effect [46]. Consequently, the overall scattering matrix of each MZI is derived as

$$\begin{aligned} U_{\text{MZI}} &= R_\phi S R_\theta S \\ &= \frac{1}{2} \begin{bmatrix} \exp(j\phi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \begin{bmatrix} \exp(j\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \\ &= j \exp\left(j\frac{\theta}{2}\right) \begin{bmatrix} \exp(j\phi) \sin \frac{\theta}{2} & \exp(j\phi) \cos \frac{\theta}{2} \\ \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \end{bmatrix}, \quad (1) \end{aligned}$$

where U_{MZI} , R_ϕ , R_θ , and S are the scattering matrices of MZI, external phase shifter, internal phase shifter, and directional coupler, respectively. This indicates that the MZI unit can be characterized by the special unitary group of degree 2 [SU(2)] transformation, providing a unitary interference of two input modes assuming that the cell is lossless [47]. Hence, by configuring (θ, ϕ) , any rotation of two-dimensional unitary groups can be realized. For N -dimensional unitary transformation $U(N)$, each N input mode must interfere with others for N output modes. The feasible arrangement of MZIs for mode connections was first proposed by Reck *et al.* [48], demonstrating a triangular topology of the MZI array. Followed by Clements *et al.* [49], a more compact topology with the same number of MZIs was described. Here we adopt the Clements topology to implement the compact ONN as shown in the diagram of the OIU in Fig. 1(a). Arbitrary synaptic weight matrix W can be realized by two unitary matrices (U and V^\dagger) and one diagonal matrix Σ factored out as $W = U\Sigma V^\dagger$, using a physical instantiation of the singular value decomposition (SVD), where V^\dagger means the Hermitian transformation of V [50]. In addition, the implementation of the nonlinear activation function of the ONN is essential for the network, and

B. Quantized Parameter Imprecisions

Since there are severe impacts of parameter imprecisions, the distorted scattering matrix of the MZI caused by these errors is consequently obtained. There are four main types of imprecisions in devices, including phase shift error, insertion loss, drift of the coupling coefficient, and photodetection noise. From previous experimental measurements [22], the phase errors $\{\delta\theta, \delta\phi\}$ can be modeled as random Gaussian distributed variables $G_p(\mu = 0, \sigma)$ where the expectation μ is zero, and the standard deviation σ is typically in the range of 0.05 rad. To have a more precise analysis of the parameters (μ, σ) , we have to consider the source of phase errors. The source can be divided into two parts. The first one can be treated as phase variation $\Delta\Phi$ caused by thermal effects from neighboring phase shifters [52,53]. The i th affected phase shifter $\Delta\Phi_i$ can be calculated by the adjacent phase shifters $\{\Phi_k\}$, defined by $\Delta\Phi_i = \sum_{k \neq i} C_{k \rightarrow i} (\Phi_k + \Delta\Phi_k)$. $C_{k \rightarrow i}$ is the thermal effect coefficient from the k th shifter to the i th shifter, which is determined by the distance between two phase shifters, and $\Delta\Phi_k$ is the phase shift variation of the surrounded k th shifter. The coefficient can be derived from the heat conduction equation [52] or measured from experiments [53]. Since $C_{k \rightarrow i}$ is about 0.065 with a distance of 80 μm , the term $\sum_{k \neq i} C_{k \rightarrow i} \Delta\Phi_k$ can be ignored. The other variation source is from the fabrication imperfection and digital to analog converter (DAC) limited precision. They can be regarded as standard deviations σ_{MZI} and σ_p , respectively. In this chip [22], an individual MZI has a far lower noise value of $\sigma_{\text{MZI}} \rightarrow 5 \times 10^{-3}$ rad. If a 12-bit DAC is used, the DAC resolution ΔV is about 1.2×10^{-3} V for 5 V voltage and $\sigma_p < \eta \Delta V$, where η is the voltage–phase conversion coefficient, typically about 1.6 rad/V. The overall standard deviation $\sigma_{\text{all}} = \sigma_{\text{MZI}} + \sigma_p$ is lower than 6.9×10^{-3} .

Another error in the MZI is insertion loss α . It can be assumed as a constant attenuation coefficient for each MZI. The perturbed scattering matrix of the MZI array $U'(N)$ is characterized as $U'(N) = \alpha^m U(N)$ for the case where the number of MZIs is m and the insertion loss α . Also, the imprecise width of the waveguide could change the coupling region of directional couplers, causing the coupling ratio error ϵ [36]. This error can be calculated by measuring the extinction ratio E of the MZI in the crossbar state, where $\epsilon = 10^{-E/20}$. In summary, the scattering matrix of the MZI with errors is expressed as

$$\begin{aligned} U'_{\text{MZI}} &= \alpha R_\phi S_\epsilon R_\theta S_\epsilon = \frac{1}{2} \alpha \begin{bmatrix} \exp(j\delta\phi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{1+\epsilon} & j\sqrt{1-\epsilon} \\ j\sqrt{1-\epsilon} & \sqrt{1+\epsilon} \end{bmatrix} \begin{bmatrix} \exp(j\delta\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{1+\epsilon} & j\sqrt{1-\epsilon} \\ j\sqrt{1-\epsilon} & \sqrt{1+\epsilon} \end{bmatrix} \\ &= \frac{1}{2} \alpha \begin{bmatrix} [(-1+\epsilon) + (1+\epsilon)\exp(j\delta\theta)] \exp(j\delta\phi) & j\sqrt{1-\epsilon^2} [1 + \exp(j\delta\theta)] \exp(j\delta\phi) \\ j\sqrt{1-\epsilon^2} [1 + \exp(j\delta\theta)] & (1+\epsilon) + (-1+\epsilon)\exp(j\delta\theta) \end{bmatrix}. \quad (2) \end{aligned}$$

we utilize the combination of electro-optic hardware platforms to realize the nonlinear function in the activation layer in our work [51].

Typically, after the optical–electrical conversion, there is photodetection noise $\{\delta_D\}$, experimentally following Gaussian distribution $G_D(\mu = 0, \sigma = \sigma_D)$. The practical received

output O'_i with imprecisions is expressed as $O'_i = (1 + \delta_D)O_i$ where O_i is the ideal output vector of the i th sample at the output layer. Since the sources of these noises are different and independent of each other, these noises are considered as independent variables and are taken into account simultaneously when training the noisy ONN.

There are also other potential drifts existing in experiments, including optical input encoding, device aging, temperature, and learning duration. In the practical MZI-based ONN chip [21,22,31], N input signals are encoded by N cascaded MZIs for N input ports, and these input signals can be inferred from the other port of the MZI. Hence, the effect of optical input drifts can be minimized by real-time monitoring. Device aging mainly exists in electric-optic phase shifters where slab Si waveguides are doped to form PN junctions [54]. As mentioned before, we adopt thermal phase shifters, which typically use TiN as heaters to modulate phases. Hence, device aging can be minimized. The temperature drift can be controlled by a thermoelectric controller to ensure long-term temperature stability better than 0.01 K in experiments. As the GA training is conducted on computers, the computations in the *ex situ* training, which takes about 3–4 h, can also be accelerated by large servers and cloud computing. The learning duration will not have any impact on experimental performance.

By using the quantized parameter imprecisions, the degradation of ONN performance can be directly evaluated on the basis of the network's accuracy in the specific dataset. Distorted scattering matrices of the MZI array were set as synaptic weights in hidden layers. Then the imprecise ONN was applied to perform a machine learning task in the supervised learning way. Classification accuracy of the affected ONN in different error ranges was obtained as depicted in Fig. 2. Figure 2(a) demonstrates the accuracy degradation caused by phase shift error and MZI loss. The typical phase shift error is about 0.05 rad, which lowers the classification accuracy by about 4%. For silicon photonics, the loss of each MZI is about 0.05–0.1 dB, which indicates that the accuracy would drop by about 1%. Figures 2(b) and 2(c) indicate the impacts of the extinction ratio and photodetection noise on accuracy, respectively. The extinction ratio of experimentally measured MZI can reach 20 dB and photodetection noise about 0.05, reducing the accuracy by about 11% and 0.7%, respectively. This indicates that in practical cases, the coupling ratio error would contribute more to accuracy degradation than MZI loss or photodetection noise.

C. Workflow of the Training Scheme

Without any calibration steps or extra imprecise network training, the ONN is typically sensitive to parameter imprecisions, hindering the use of photonic chips in machine learning. Here we propose a network training scheme using GA training considering practical imprecisions existing in optical components. The training flow is illustrated in Fig. 3. First, neural synaptic weights in the ONN are iterated and trained by using the gradient stochastic descent algorithm. Based on the backpropagation of loss, the classification accuracy can rapidly converge to the maximum, and the optimal phase shifts of MZIs are obtained. Then, we consider the effect of parameter imprecisions, and the GA is applied for the optimization of neuron weights

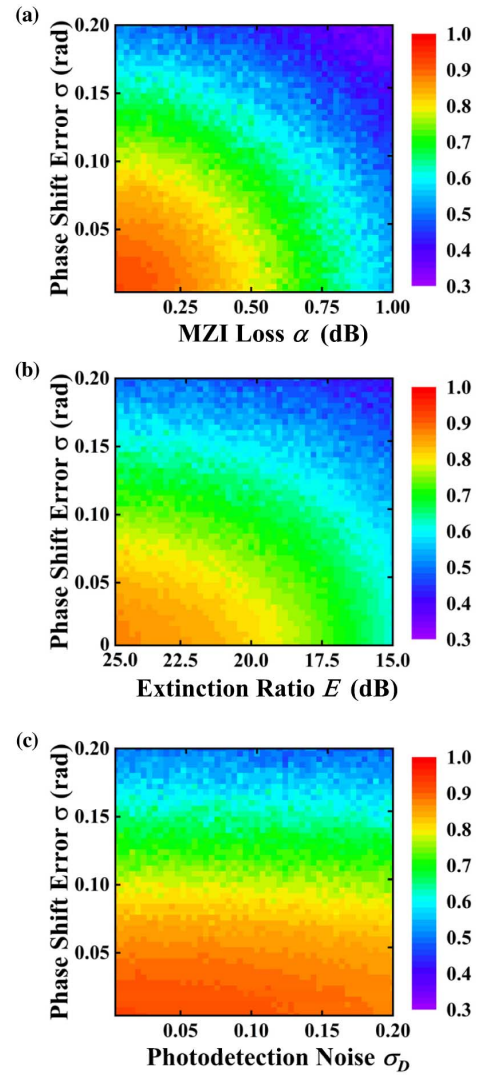


Fig. 2. Heat map of classification accuracy in the MNIST dataset with the imprecise ONN chip. (a) Classification performance between phase shift error σ and per MZI loss α . (b) Effects of phase shift error σ and extinction ratio E . (c) Impacts of phase shift error σ and photodetection noise σ_D on the final achieved network performance.

learning. The optimal phase shifts $\{\xi\}$ are set as references. Compensatory phase shifts $\{\Delta\xi\}$ are added on $\{\xi\}$, and the compensated phase shift array $\{\xi + \Delta\xi\}$ is defined as an individual in the GA process. The compensated phase shift range of $\{\Delta\xi\}$ is based on the phase shift error standard deviation σ . Randomly generating compensated phase shifts, γ individuals $\{\xi + \Delta\xi_{i \in \{1,2,\dots,\gamma\}}\}$ are achieved as the initial population in the GA training stage, where γ is the number of individuals in one generation. Each individual will produce a diverse ONN. Then we define average classification accuracy in M imprecise chips with parameter imprecisions $\{\sigma_{\text{all}}^l, \alpha^l, E^l, \sigma_D^l\}$ as the fitness function $f(O)$, where $l \in \{1, 2, \dots, M\}$, and O means the output vector of the individual. To ensure that the numerical optimization can offer a significant improvement on practical hardware, we define the fitness in GA as the average accuracy in a large number of ONNs with randomly sampled noises. In the GA, several operators are applied, including selection,

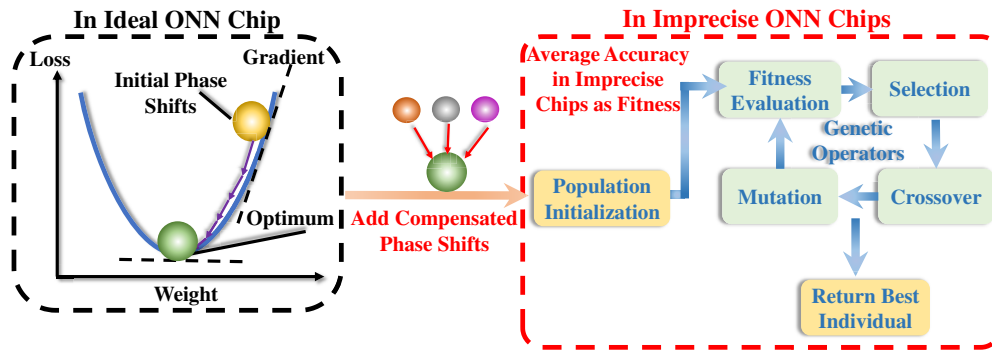


Fig. 3. Training flow of the ONN with parameter imprecisions using the genetic algorithm. Two major stages are involved and illustrated, including gradient training of the ideal ONN and genetic training in the imprecise chips.

crossover, and mutation [55]. Roulette method selection is adopted in the selection stage, meaning that the individual with higher fitness is more likely to be chosen. Also, an individual can be selected repeatedly in one evolution stage. By continuously generating new individuals, the optimum individual can ensure the average accuracy to be close to the ideal one by compensating for all imprecisions. In this way, the genetics-based trained ONN is shown to have enhanced robustness in erroneous cases.

3. SIMULATION, RESULTS, AND DISCUSSION

A. Software Implementation

Two types of datasets are chosen to validate our training scheme. One is the *Iris* flower dataset, which consists of 50 samples from three species with four separate features. The other is the MNIST dataset of handwritten digits with the training set and validation set in the proportion of 500:100. Here we export the first eight dominant features of each image in the MNIST dataset using principal component analysis (PCA). Each input image datum $I = \{I_{i \in \{1,2,\dots,N\}}\}$, where I_i means the value of the i th feature and $N = 4$ in *Iris* and $N = 8$ in MNIST, is encoded into the optical signal $X = \{X_{i \in \{1,2,\dots,N\}} = A_i \exp(j\theta_i)\}$. Here we apply only the intensity to be modulated, hence defining $A_i = I_i$ and $\theta_i = 0$. The ONN contains three hidden layers in the Clements topology, each followed by an activation layer using a rectified linear unit (ReLU) function and an output layer with the square function in the activation layer corresponding to the photodetector measurement. After the input signal is encoded and propagates through the ONN, the output layer exports an L -dimensional vector $O = \{O_{i \in \{1,2,\dots,L\}}\}$. A one-hot encoding [56] scheme is used to distinguish the class of the signal, implying that if $\text{Max}(O) = O_k$, the input signal is identified as the k th class. In the initial gradient descent training stage, we use categorical cross-entropy (CCE) as the loss function, which is given by $C(O, \hat{O}) = -\sum_{m=1}^n \hat{O}_m \log O_m$, where n is the number of classes, O is the obtained vector at the output layer, and \hat{O} is the ground truth vector in the form of $(0, \dots, 1, \dots, 0)$. We use the Neuroptica Python package [57], which has been adopted in Refs. [36,39,51], to simulate the ONN and apply the gradient descent algorithm to train the ideal ONN without practical errors. The training curves of loss, as well as accuracies in training and test datasets of *Iris* and MNIST in 500 epochs

are plotted in Figs. 4(a) and 4(b), where the maximum accuracies in these training datasets can reach 97.1% and 92.0%, respectively. After that, GA is applied in the gradient-based trained ONN. The optimum phase shift array after gradient-based training is selected as the initial base $\{\xi\}$, and the extra phase shifts $\{\Delta\xi\}$ are added to it. We define $M = 50$, which means the average classification accuracy in 50 imprecise chips is set as the fitness function. The parameter imprecisions are set as $\{\sigma_{\text{all}} \in [0.04, 0.05], \alpha \in [0.05, 0.1]$,

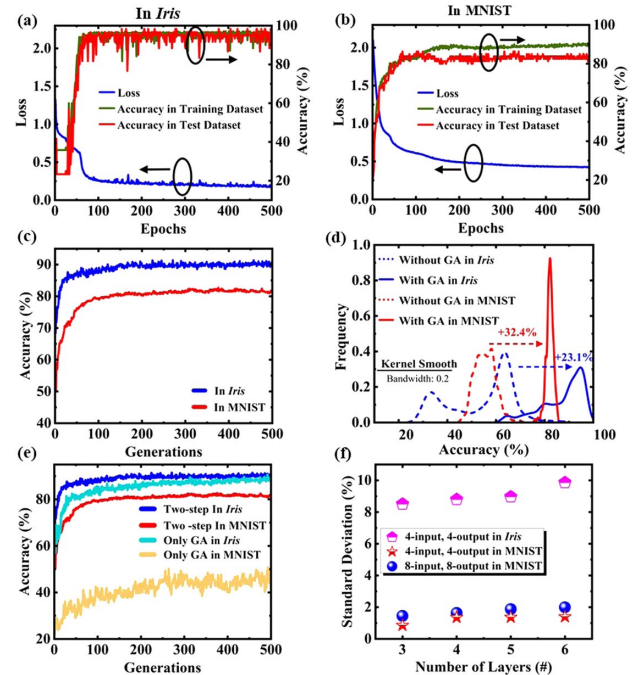


Fig. 4. Training curves of the ideal ONN using gradient descent algorithm in (a) *Iris* and (b) MNIST datasets, including loss curve as well as accuracies in training and test datasets. (c) Maximum accuracy in each generation at the GA training stage considering imprecise optical components. The optimal individual can have 91.4% and 82.7% accuracy in *Iris* and MNIST datasets, respectively. (d) Accuracy distribution with and without GA training in *Iris* and MNIST datasets. (e) Comparisons of training curves between the two-step training method and the only GA training method in *Iris* and MNIST datasets. (f) Standard deviations of accuracy distributions in *Iris* and MNIST datasets with different numbers of layers and different layer widths.

$$\begin{aligned} \text{Typical error range: } \{ \sigma_{\text{all}} \in [0.04, 0.05], \alpha \in [0.05, 0.1], E \in [13, 15], \sigma_D \in [0.04, 0.05] \}; \\ \text{Low error range: } \{ \sigma_{\text{all}} \in [0.004, 0.005], \alpha \in [0.04, 0.05], E \in [20, 23], \sigma_D \in [0.0009, 0.0011] \}. \end{aligned} \quad (3)$$

$E \in [13, 15], \sigma_D \in [0.04, 0.05]$ }, which can typically be measured from experiments [22]. $\{\Delta\xi\}$ is set in the form of $\{\Delta\xi \in [-7\sigma_{\theta=\phi}, +7\sigma_{\theta=\phi}]\}$ since most of the phase shift errors are distributed in this range in the Gaussian profile. Hence, $\{\Delta\xi \in [-0.35, 0.35]\}$. Then, 50 initial individuals are generated as the first generation. After GA training in 500 generations, the best individual with the highest fitness is exported, as shown in Fig. 4(c). To avoid local optima, we have adopted some methods to judge the resulting optima, including repeating training and adding randomness in GA. In the end, we test the best individual in 200 imprecise chips. The classification accuracy distribution with and without GA training in two types of datasets in 200 chips is plotted in Fig. 4(d). The results show that the GA training scheme is universally effective for different datasets, where the average accuracy in the imprecise chips is enhanced by about 23.1% (from 68.3% to 91.4%) for the *Iris* dataset and 32.4% (from 50.3% to 82.7%) for the MNIST dataset. The more complicated MNIST dataset requires more neurons at the input layer, leading to the increase in network size and the use of more phase shifters. Therefore, the convergence speed of GA training in the MNIST dataset is slightly slower than that in the *Iris* dataset, and there is also a larger difference in accuracy between the ideal ONN and the GA trained ONN (97.1% and 92% in the ideal ONN compared to 91.4% and 82.7% after GA training in *Iris* and MNIST datasets, separately). From the accuracy distribution in the imprecise chips, as shown in Fig. 4(d), it is reported that the distribution can be approximated as a Gaussian distribution profile with the expectation of the distribution close to the maximum average accuracy in the GA training stage.

Since GA is a heuristic method to generate high-quality solutions to search problems by relying on bio-inspired operators, it strongly relies on initial individuals. Hence, the adoption of the gradient descent algorithm in the first training of the ideal ONN helps to find optimal individuals so that the training based on GA would quickly converge to global optima instead of local optima. As shown in Fig. 4(e), the two-step training method is faster to converge than only GA training. In addition, we analyze the reason for the different standard deviations of the accuracy distribution in Fig. 4(d). Figure 4(f) compares the standard deviations in different numbers of layers and layer widths. The results show that a greater number of layers and larger layer widths lead to larger standard deviations. A more complex ONN would lead to more significant changes in accuracy, while the difference in accuracy distribution in Fig. 4(d) is mainly related to the type of dataset.

B. Analysis of Hyper-Parameters

The dominant factor in the GA training scheme is parameter imprecision range. It determines the degradation of ONN performance since larger ranges of imprecisions obviously increase the randomization of the network's functionality. Hence, it is necessary to survey the training scheme in different imprecision ranges. We compare the scheme in two types of imprecision ranges as defined below.

The first case is the typical error ranges that we took from this chip [22] to test the validation of GA training in previous sessions. By applying a more precise phase shift error model $G_p(\mu = \Delta\Phi, \sigma_{\text{all}})$, the standard deviation of phase shift errors can be remarkably reduced by about eight times as compared to that reported in Ref. [22]. Considering the fact that the network is *ex situ* trained, the optimal phase shifts of the network are known, and hence, $\Delta\Phi$ is assumed as a constant, which is known right after the neuron weights training stage. The value of σ_{all} has a much more significant impact on the GA accuracy degradation. The constant $\Delta\Phi$, typically only 2% of the applied phase, has little impact on GA training. The GA is robust enough to absorb the 2% error. Hence, here we set $\mu = \Delta\Phi = 0$ when training and testing the ONN. Note here that setting to zero does not mean that we treat the thermal cross talk as non-existent. The critical step to consider thermal cross talk is to calculate $\Delta\Phi$ from the neighboring phase shift settings and compensate for accordingly after GA. Also, the MZI loss can be reduced and the extinction ratio can be improved using an advanced fabrication process and other optimization steps [58,59]. In this research [22], the photodetection noise can be extremely small in the range of 0.1%. Therefore, we obtain much lower error ranges, which are then tested and compared to the typical one. It is clearer to use MNIST to exhibit the effects of hyper-parameters since the accuracy distribution in MNIST is more concentrated than that in *Iris*. The GA training curves and accuracy distribution in the MNIST dataset are illustrated in Figs. 5(a) and 5(b), respectively, where the average accuracy in the imprecise chips is enhanced from 85.5% to 90.8% in the condition of an extremely low error range, which is much closer to the ideal ONN's accuracy (92%). Moreover, the lowest accuracy in the imprecise chip is 88.0%, which ensures the lower limit of ONN performance.

Since the training scheme is a pure software method, various hyper-parameters in GA training make a significant impact on the overall robustness of the ONN. Therefore, we analyze the effects of these hyper-parameters containing the compensated phase shift range $\{\Delta\xi\}$, the number of imprecise chips M , and the population in each generation. To illustrate the effects of these hyper-parameters more universally, we use the MNIST dataset and the typical error ranges $\{\sigma_{\text{all}} \in [0.04, 0.05], \alpha \in [0.05, 0.1], E \in [13, 15], \sigma_D \in [0.04, 0.05]\}$ for demonstration. Figure 6(a) demonstrates the training curves in the condition of different compensated phase shift ranges, where smaller ranges have more severe impacts on the training results. In Fig. 6(b), it can be observed that when the compensated phase shift range is $\{\Delta\xi \in [-5\sigma_{\theta=\phi}, 5\sigma_{\theta=\phi}]\}$, the average accuracy after GA training can achieve the maximum, and the best accuracy distribution is obtained. It is noted that if $\Delta\xi \in [-\sigma_{\theta=\phi}, \sigma_{\theta=\phi}]$, the average accuracy after training is only about 73%, which is much less than in other ranges, indicating that if the compensated phase shift range is too small, the

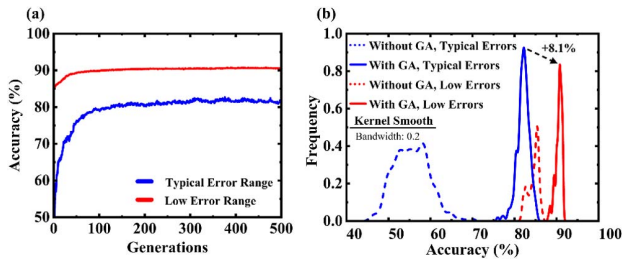


Fig. 5. (a) Accuracy training curves in the MNIST dataset during the GA training stage in the condition of typical error ranges $\{\sigma_{\text{all}} \in [0.04, 0.05], \alpha \in [0.05, 0.1], E \in [13, 15], \sigma_D \in [0.04, 0.05]\}$ and experimentally measured low error ranges $\{\sigma_{\text{all}} \in [0.004, 0.005], \alpha \in [0.04, 0.05], E \in [20, 23], \sigma_D \in [0.0009, 0.0011]\}$. (b) Accuracy distribution of imprecise chips in two error range cases.

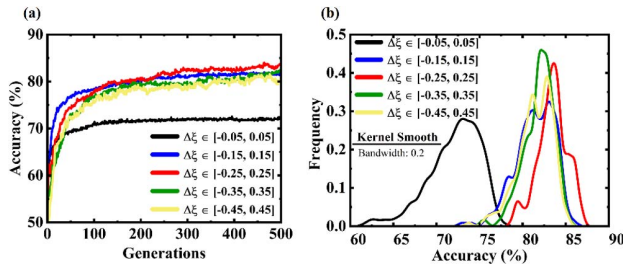


Fig. 6. (a) Accuracy training curves in the GA training stage in different compensated phase shift ranges $\{\Delta\xi\}$. (b) Effects of compensated phase shift ranges $\{\Delta\xi\}$ on the accuracy distribution in imprecise chips.

GA training scheme cannot search out the individual with high robustness in a small solution space. Also, if the compensated phase shift range is larger than $\pm 5\sigma_{\theta=\phi}$, such as $\Delta\xi \in [-7\sigma_{\theta=\phi}, 7\sigma_{\theta=\phi}]$, it would increase the uncertainty of the obtained solutions, leading to a reduction in average accuracy.

When we change the number of imprecise chips, the number of imprecise chips used to evaluate the fitness of the individual has a smaller impact on the accuracy distribution than the compensated phase shift range. As shown in Fig. 7(a), for different numbers of chips, the training curves converge to the same results. Also, Fig. 7(b) shows a similar accuracy distribution in different numbers of chips, indicating that 30 imprecise chips are sufficient to estimate the robustness of the individual in these error ranges. The irrelevance between

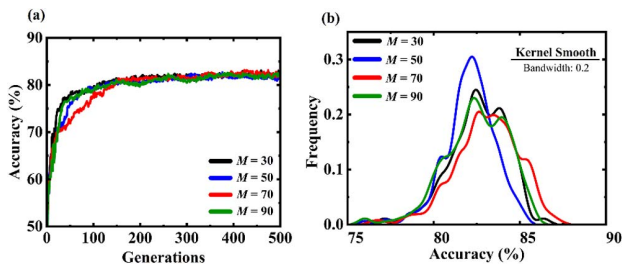


Fig. 7. (a) Accuracy training curves in the GA training stage using different numbers of imprecise chips M . (b) Effects of the number of imprecise chips M on accuracy distribution.

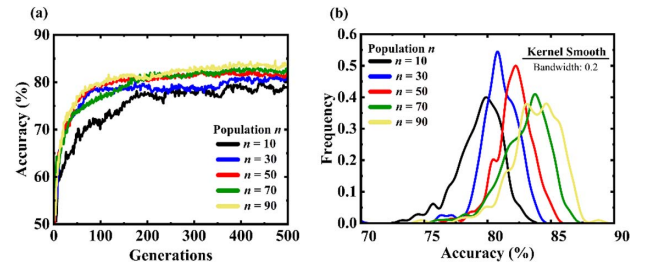


Fig. 8. (a) Accuracy training curves in the GA training stage in the condition of different populations. (b) Effects of different populations in evolution on the accuracy distribution in imprecise chips.

the number of chips and classification accuracy can significantly enhance the computational efficiency in the GA training step. Regarding the effects of different populations in each generation, as depicted in Fig. 8, the curves point out that increasing the number of individuals impressively enhances the maximum accuracy. However, the computation time also increases exponentially as the population rises. Figure 8(a) reports that more individuals can converge to the optima more quickly. The average value of the accuracy distribution in Fig. 8(b) tends to saturate when the population increases to 90, suggesting that the number of individuals in the range of 70–90 is sufficient and is a good balance between computation cost and the improved robustness of the ONN chip.

C. Comparison to SA and PSO

In the self-learning process of weights in ONNs, there are also alternative approaches that can replace the GA to train neurons based on similar evolutionary algorithms, such as simulated annealing (SA) and particle swarm optimization (PSO) [60]. However, the training process of ONNs has the feature of multiple variables updating, which can restrain the convergence speed and training performance of the algorithms. To demonstrate the efficiency of the GA in this situation, we implement these three algorithms in the same conditions and evaluate their performance in imprecise chips. Because the MNIST dataset requires more neurons and layers than *Iris*, *Iris* can minimize the training performance induced by hyper-parameters. It is better to use *Iris* to compare GA, SA, and PSO. We ensure that the training curves in all three algorithms converge to a specific value with the same epochs and then compare the trained phase shift settings in imprecise chips. The training curves depicted in Figs. 9(a) and 9(b) indicate that the GA

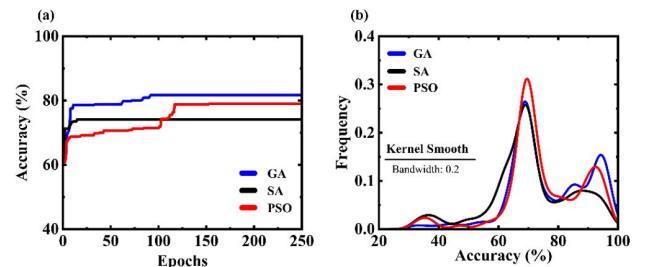


Fig. 9. (a) Accuracy training curves in three heuristic algorithms with the accuracy converging to a particular value. (b) Accuracy distribution of three algorithms in the same imprecise chips.

training scheme has a faster convergence speed than PSO and the highest average accuracy in imprecise chips. Although the SA training method can rapidly converge at an earlier stage, it tends to get stuck in local optima and can reach only about 74% accuracy. The PSO training has the lowest convergence speed and relatively lower accuracy compared to GA. These results prove that the GA training scheme is more suitable for multivariate function optimization since all of the phase shifts in MZIs need to be trained simultaneously. In addition, if the MNIST dataset is used and the network is scaled up, after-trained accuracy in SA and PSO reduces and reaches only about 60%. The accuracy distribution also demonstrates the ascendancy of GA, showing that cases with accuracy lower than 40% can almost be eliminated, and more proportions of cases are distributed in the range of 80% to 93% compared to SA and PSO. In short, the GA training not only has faster convergence speed and better solutions in high-dimensional solution space but also realizes the trained phase shifts with better robustness to practical component imprecisions.

4. CONCLUSION

To sum up, we propose and demonstrate a two-step *ex situ* robust training method for an MZI-based ONN with three hidden layers and 224 imprecise tunable thermo-optic phase shifters. The simulation results show more than 23% accuracy enhancement in both *Iris* and MNIST datasets and comparable accuracy of 90.8% in the imprecise ONN to the ideal accuracy of 92.0%. Our method is an *ex situ* training method of the ONN, which means that the method is used on a computer model of the ONN. The *ex situ* step can provide trained weight parameters used as initial individuals for the second GA step. The weight parameters configured in realistic ONN chips can achieve improved practical accuracy. Furthermore, the comparison of GA, PSO, and SA demonstrates the superiority of GA in the training of imprecise ONNs. Our proposed scheme could also be applied to other network architectures where the phase shift is the only element to be configured, such as CNNs and RNNs. Our method has great potential in optical linear programmable processors [5,8,23–25], ONN accelerators [21,22,31], and photonic quantum information applications [23,61–63]. With this paper, we provide an error-resistant training scheme that is generalized and efficient for practical photonic neuromorphic computing platforms with imperfect components.

Funding. Ministry of Education - Singapore (MOE2018-T2-1-137, R-263-000-C84-112); National Research Foundation Singapore (QEP-P3, QEP-P2, Quantum Engineering Programme 1).

Disclosures. The authors declare no conflicts of interest.

Data Availability. Data presented in this paper are available in Refs. [44,45]. Other underlying data are not publicly available at this time but may be obtained from the authors upon reasonable request.

REFERENCES

- B. J. Shastri, A. N. Tait, T. Ferreira de Lima, W. H. P. Pernice, H. Bhaskaran, C. D. Wright, and P. R. Prucnal, "Photonics for artificial intelligence and neuromorphic computing," *Nat. Photonics* **15**, 102–114 (2021).
- J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, and D. Brunner, "Reinforcement learning in a large-scale photonic recurrent neural network," *Optica* **5**, 756–760 (2018).
- 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**, 1004–1008 (2018).
- J. Robertson, M. Hejda, J. Bueno, and A. Hurtado, "Ultrafast optical integration and pattern classification for neuromorphic photonics based on spiking VCSEL neurons," *Sci. Rep.* **10**, 6098 (2020).
- N. C. Harris, G. R. Steinbrecher, M. Prabhu, Y. Lahini, J. Mower, D. Bunandar, C. Chen, F. N. C. Wong, T. Baehr-Jones, and M. Hochberg, "Quantum transport simulations in a programmable nano-photonic processor," *Nat. Photonics* **11**, 447–452 (2017).
- J. Feldmann, N. Youngblood, M. Karpov, H. Gehring, X. Li, M. Stappers, M. Le Gallo, X. Fu, A. Lukashchuk, and A. S. Raja, "Parallel convolutional processing using an integrated photonic tensor core," *Nature* **589**, 52–58 (2021).
- B. Shi, N. Calabretta, and R. Stabile, "Deep neural network through an InP SOA-based photonic integrated cross-connect," *IEEE J. Sel. Top. Quantum Electron.* **26**, 7701111 (2019).
- N. C. Harris, J. Carolan, D. Bunandar, M. Prabhu, M. Hochberg, T. Baehr-Jones, M. L. Fanto, A. M. Smith, C. C. Tison, and P. M. Alsing, "Linear programmable nanophotonic processors," *Optica* **5**, 1623–1631 (2018).
- L. Zhuang, C. G. H. Roeloffzen, M. Hoekman, K.-J. Boller, and A. J. Lowery, "Programmable photonic signal processor chip for radio-frequency applications," *Optica* **2**, 854–859 (2015).
- J. Notaros, J. Mower, M. Heuck, C. Lupo, N. C. Harris, G. R. Steinbrecher, D. Bunandar, T. Baehr-Jones, M. Hochberg, and S. Lloyd, "Programmable dispersion on a photonic integrated circuit for classical and quantum applications," *Opt. Express* **25**, 21275–21285 (2017).
- C. Taballione, T. A. W. Wolterink, J. Lugani, A. Eckstein, B. A. Bell, R. Grootjans, I. Visscher, J. J. Renema, D. Geskus, and C. G. H. Roeloffzen, "8x8 programmable quantum photonic processor based on silicon nitride waveguides," in *Frontiers in Optics 2018* (OSA, 2018), paper JTU3A.58.
- D. Pérez, I. Gasulla, L. Crudgington, D. J. Thomson, A. Z. Khokhar, K. Li, W. Cao, G. Z. Mashanovich, and J. Capmany, "Multipurpose silicon photonics signal processor core," *Nat. Commun.* **8**, 636 (2017).
- J. Wang, F. Sciarrino, A. Laing, and M. G. Thompson, "Integrated photonic quantum technologies," *Nat. Photonics* **14**, 273–284 (2020).
- Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, "All-optical neural network with nonlinear activation functions," *Optica* **6**, 1132–1137 (2019).
- T.-Y. Cheng, D.-Y. Chou, C.-C. Liu, Y.-J. Chang, and C.-C. Chen, "Optical neural networks based on optical fiber-communication system," *Neurocomputing* **364**, 239–244 (2019).
- R. Stabile, G. Dabos, C. Vagionas, B. Shi, N. Calabretta, and N. Pleros, "Neuromorphic photonics: 2D or not 2D?" *J. Appl. Phys.* **129**, 200901 (2021).
- X. Qiang, X. Zhou, J. Wang, C. M. Wilkes, T. Loke, S. O'Gara, L. Kling, G. D. Marshall, R. Santagati, and T. C. Ralph, "Large-scale silicon quantum photonics implementing arbitrary two-qubit processing," *Nat. Photonics* **12**, 534–539 (2018).
- M. Teng, A. Honardoost, Y. Alahmadi, S. S. Polkoo, K. Kojima, H. Wen, C. K. Renshaw, P. LiKamWa, G. Li, and S. Fatthpour, "Miniaturized silicon photonics devices for integrated optical signal processors," *J. Lightwave Technol.* **38**, 6–17 (2020).
- C. Baudot, M. Douix, S. Guerber, S. Crémer, N. Vulliet, J. Planchot, R. Blac, L. Babaud, C. Alonso-Ramos, D. Benedikovich, D. Pérez-Galacho, S. Messaoudène, S. Kerdiles, P. Acosta-Alba, C. Euvrard-Colnat, E. Cassan, D. Marris-Morini, L. Vivien, and F. Boeuf, "Developments in 300 mm silicon photonics using traditional

- CMOS fabrication methods and materials," in *IEEE International Electron Devices Meeting (IEEE)*, (2017), pp. 34.33.31–34.33.34.
20. D. P. López, "Programmable integrated silicon photonics waveguide meshes: optimized designs and control algorithms," *IEEE J. Sel. Top. Quantum Electron.* **26**, 8301312 (2019).
 21. 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).
 22. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Solja, "Deep learning with coherent nanophotonic circuits," *Nat. Photonics* **11**, 441–446 (2017).
 23. J. Carolan, C. Harrold, C. Sparrow, E. Martín-López, N. J. Russell, J. W. Silverstone, P. J. Shadbolt, N. Matsuda, M. Oguma, and M. Itoh, "Universal linear optics," *Science* **349**, 711–716 (2015).
 24. A. Ribeiro, A. Ruocco, L. Vanacker, and W. Bogaerts, "Demonstration of a 4x4-port universal linear circuit," *Optica* **3**, 1348–1357 (2016).
 25. P. L. Mennea, W. R. Clements, D. H. Smith, J. C. Gates, B. J. Metcalf, R. H. S. Bannerman, R. Burgwal, J. J. Renema, W. S. Kolthammer, and I. A. Walmsley, "Modular linear optical circuits," *Optica* **5**, 1087–1090 (2018).
 26. D. Pérez-López, E. Sánchez, and J. Capmany, "Programmable true time delay lines using integrated waveguide meshes," *J. Lightwave Technol.* **36**, 4591–4601 (2018).
 27. A. N. Tait, A. X. Wu, T. F. De Lima, E. Zhou, B. J. Shastri, M. A. Nahmias, and P. R. Prucnal, "Microring weight banks," *IEEE J. Sel. Top. Quantum Electron.* **22**, 312–325 (2016).
 28. S. Ohno, K. Toprasertpong, S. Takagi, and M. Takenaka, "Si microring resonator crossbar array for on-chip inference and training of optical neural network," arXiv:2106.04351 (2021).
 29. F. Denis-Le Coarer, M. Sciamanna, A. Katumba, M. Freiburger, J. Dambre, P. Bienstman, and D. Rontani, "All-optical reservoir computing on a photonic chip using silicon-based ring resonators," *IEEE J. Sel. Top. Quantum Electron.* **24**, 7600108 (2018).
 30. S. Ohno, K. Toprasertpong, S. Takagi, and M. Takenaka, "Demonstration of classification task using optical neural network based on Si microring resonator crossbar array," in *European Conference on Optical Communications (ECOC)* (IEEE, 2020), pp. 1–4.
 31. F. Shokraneh, S. Geoffroy-Gagnon, M. S. Nezami, and O. Liboiron-Ladouceur, "A single layer neural network implemented by a 4x4 MZI-based optical processor," *IEEE Photon. J.* **11**, 4501612 (2019).
 32. Y. Jiang, W. Zhang, F. Yang, and Z. He, "Photonic convolution neural network based on interleaved time-wavelength modulation," *J. Lightwave Technol.* **39**, 4592–4600 (2021).
 33. 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 TeraFLOPs per second photonic convolutional accelerator for deep learning optical neural networks," arXiv:2011.07393 (2020).
 34. G. Mourgias-Alexandris, G. Dabos, N. Passalis, A. Totovic, A. Tefas, and N. Pleros, "All-optical WDM recurrent neural networks with gating," *IEEE J. Sel. Top. Quantum Electron.* **26**, 6100907 (2020).
 35. C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex, "Using an imperfect photonic network to implement random unitaries," *Phys. Rev. Lett.* **119**, 170501 (2017).
 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. D. A. B. Miller, "Perfect optics with imperfect components," *Optica* **2**, 747–750 (2015).
 38. F. Shokraneh, S. Geoffroy-Gagnon, and O. Liboiron-Ladouceur, "The diamond mesh, a phase-error- and loss-tolerant field-programmable MZI-based optical processor for optical neural networks," *Opt. Express* **28**, 23495–23508 (2020).
 39. M. Y. S. Fang, S. Maniaturuni, C. Wierzynski, A. Khosrowshahi, and M. R. DeWeese, "Design of optical neural networks with component imprecisions," *Opt. Express* **27**, 14009–14029 (2019).
 40. 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).
 41. R. Hamerly, S. Bandyopadhyay, and D. Englund, "Accurate self-configuration of rectangular multiport interferometers," arXiv:2106.03249 (2021).
 42. S. Bandyopadhyay, R. Hamerly, and D. Englund, "Hardware error correction for programmable photonics," *Optica* **8**, 1247–1255 (2021).
 43. R. Hamerly, S. Bandyopadhyay, and D. Englund, "Robust zero-change self-configuration of the rectangular mesh," in *Optical Fiber Communication Conference (OSA)*, (2021), paper Tu5H.2.
 44. A. Asuncion and D. Newman, *UCI Machine Learning Repository* (Irvine, 2007).
 45. L. Deng, "The MNIST database of handwritten digit images for machine learning research," *IEEE Signal Process. Mag.* **29**, 141–142 (2012).
 46. N. C. Harris, Y. Ma, J. Mower, T. Baehr-Jones, D. Englund, M. Hochberg, and C. Galland, "Efficient, compact and low loss thermo-optic phase shifter in silicon," *Opt. Express* **22**, 10487–10493 (2014).
 47. B. Yurke, S. L. McCall, and J. R. Klauder, "SU(2) and SU(1,1) interferometers," *Phys. Rev. A* **33**, 4033–4054 (1986).
 48. M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental realization of any discrete unitary operator," *Phys. Rev. Lett.* **73**, 58–61 (1994).
 49. W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, "Optimal design for universal multiport interferometers," *Optica* **3**, 1460–1465 (2016).
 50. F. Shokraneh, M. S. Nezami, and O. Liboiron-Ladouceur, "Theoretical and experimental analysis of a 4x4 reconfigurable MZI-based linear optical processor," *J. Lightwave Technol.* **38**, 1258–1267 (2020).
 51. 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**, 7700412 (2020).
 52. Y. Zhu, G. L. Zhang, B. Li, X. Yin, C. Zhuo, H. Gu, T.-Y. Ho, and U. Schlichtmann, "Countering variations and thermal effects for accurate optical neural networks," in *IEEE/ACM International Conference on Computer-Aided Design (ACM, 2020)*, pp. 1–7.
 53. I. I. Faruque, G. F. Sinclair, D. Bonneau, J. G. Rarity, and M. G. Thompson, "On-chip quantum interference with heralded photons from two independent micro-ring resonator sources in silicon photonics," *Opt. Express* **26**, 20379–20395 (2018).
 54. S. V. Reddy Chittamuru, I. G. Thakkar, and S. Pasricha, "Analyzing voltage bias and temperature induced aging effects in photonic interconnects for manycore computing," in *ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP)* (IEEE, 2017), pp. 1–8.
 55. H. Zhang, J. Thompson, M. Gu, X. D. Jiang, H. Cai, P. Y. Liu, Y. Shi, Y. Zhang, M. F. Karim, G. Q. Lo, X. Luo, B. Dong, L. C. Kwek, and A. Q. Liu, "Efficient on-chip training of optical neural networks using genetic algorithm," *ACS Photon.* **8**, 1662–1672 (2021).
 56. P. Cerda, G. Varoquaux, and B. Kégl, "Similarity encoding for learning with dirty categorical variables," *Mach. Learn.* **107**, 1477–1494 (2018).
 57. S. Geoffroy-Gagnon, "Flexible simulation package for optical neural networks," 2021, <https://gitlab.com/simongg/neurooptica>.
 58. J. F. Bauters, M. L. Davenport, M. J. R. Heck, J. K. Doyle, A. Chen, A. W. Fang, and J. E. Bowers, "Silicon on ultra-low-loss waveguide photonic integration platform," *Opt. Express* **21**, 544–555 (2013).
 59. S. Chen, H. Wu, and D. Dai, "High extinction-ratio compact polarisation beam splitter on silicon," *Electron. Lett.* **52**, 1043–1045 (2016).
 60. T. Zhang, J. Wang, Y. Dan, Y. Lanqiu, J. Dai, X. Han, X. Sun, and K. Xu, "Efficient training and design of photonic neural network through neuroevolution," *Opt. Express* **27**, 37150–37163 (2019).
 61. B. J. Metcalf, J. B. Spring, P. C. Humphreys, N. Thomas-Peter, M. Barbieri, W. S. Kolthammer, X.-M. Jin, N. K. Langford, D. Kundys, and J. C. Gates, "Quantum teleportation on a photonic chip," *Nat. Photonics* **8**, 770–774 (2014).
 62. A. Crespi, R. Osellame, R. Ramponi, V. Giovannetti, R. Fazio, L. Sansoni, F. De Nicola, F. Sciarrino, and P. Mataloni, "Anderson localization of entangled photons in an integrated quantum walk," *Nat. Photonics* **7**, 322–328 (2013).
 63. H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, and Y. Hu, "Quantum computational advantage using photons," *Science* **370**, 1460–1463 (2020).