Check for updates

Accelerating silicon photonic parameter extraction using artificial neural networks

ALEC M. HAMMOND, EASTON POTOKAR, AND RYAN M. CAMACHO* (ID)

Electrical & Computer Engineering Department, Brigham Young University, Provo, UT 84604, USA
*camacho@byu.edu

Abstract: We present a novel silicon photonic parameter extraction tool that uses artificial neural networks. While other parameter extraction methods are restricted to relatively simple devices whose responses are easily modeled by analytic transfer functions, this method is capable of extracting parameters for any device with a discrete number of design parameters. To validate the method, we design and fabricate integrated chirped Bragg gratings. We then estimate the actual device parameters by iteratively fitting the simultaneously measured group delay and reflection profiles to the artificial neural network output. The method is fast, accurate, and capable of modeling the complicated chirping and index contrast.

1.   Introduction

Interest in integrated optics continues to grow as silicon photonics provides an affordable platform for areas like telecommunications, quantum information processing, and biosensing [1]. Silicon photonic devices typically contain features with sub-micron dimensions, owing to the platform's high index contrast and years of complementary metal-oxide-semiconductor (CMOS) process refinement.

While small features enable several innovative and scalable designs, they also induce an increased sensitivity to fabrication defects [2]. A fabrication defect of just one nanometer, for example, can cause a nanometer shift in the output spectrum of the silicon photonic device [3]. Understanding and characterizing these process defects is essential for device modeling and variability analysis [4,5]. Predicting and compensating for such sensitivity in the optical domain is difficult because typical simulation routines are computationally expensive and in many cases, prohibitive.

To overcome these challenges, we propose a new parameter extraction method using artificial neural networks (ANN). We train an ANN to model the complex relationships between integrated chirped Bragg gratings (ICBG) [6,7] and their corresponding reflection and group delay profiles. We use the trained ANN to extract the physical parameters of various fabricated ICBGs using a nonlinear least squares fitting algorithm — a task that is computationally prohibitive using traditional simulation routines. We find that the proposed routine produces spectra that matches well the experimental reflection and group delay profiles for the ICBGs.

Our work builds upon previous efforts that extract integrated photonic device parameters using analytic models. Chrostowski *et al.*, for example, extract the group index across a wafer with 371 identical microring resonators (MRR) using an analytic formula describing the free spectral range (FSR) [8]. Similarly, Chen *et al.* derive both the effective and group indices from MRRs by fitting the full, analytic spectral transfer function to the experimental data [9]. Melati *et al.* extract phase and group index information from small lumped reflectors knows as point reflector optical waveguides (PROW) using various analytic formulas [10].

Perhaps most similar to this work, Xing et al. build a regression model from data generated by an eigenmode solver that relates waveguide design parameters (e.g. width and thickness) to their corresponding effective indices [11]. They subsequently use this regression model in

addition to an analytic transfer matrix and a fitting routine to extract the average waveguide width and thickness from various Mach-Zhender Interferometer (MZI) devices. Our ANN parameter extraction method is fast, just like the analytic and regression models, but capable of modeling much more complicated devices, like ICBGs.

The rest of the paper is outlined as follows: first, we describe the data generation process necessary to train the ANN. Next, we describe the process of training the ANN. We then discuss the ICBG device design, fabrication, testing, and data calibration. Finally, we describe our nonlinear fitting algorithm and present our experimental results.

2.   Relevant background

ANNs model the relationship between inputs and outputs by cascading various nonlinear computational units known as neurons [12]. For a particular layer D_k, the output for each neuron within that layer consists of a nonlinear activation function $f(x)$ that transforms a weighted sum of the output of the neurons from the previous layer $\mathbf{D_{k-1}}$ and a bias term b_i such that

$$\mathbf{D_i} = f(W_k\mathbf{D_{k-1}} + b_i), \tag{1}$$

where the linear transformation W_k contains a weight for every possible mapping from neurons in the previous layer to neurons in the current layer [13].

ANN training algorithms like backpropagation tune the weights of these neurons until the functional mapping adequately models the corresponding training set [14]. Several factors, like the number of neurons, the number of layers, the activation functions, and even the training set itself, influence the training accuracy and speed of the ANN. In addition, the ANN may learn unintentional biases if the training set insufficiently represents the function space [15].

Consequently, it is important to adequately describe the ICBG using parameters that are simple and intuitive for the designer, but also comprehensive and descriptive in order to fully span the design space. To accomplish this, we parameterized the ICBG's design space using the length of the first ICBG period (a_0), the length of the last ICBG period (a_1), the number of gratings (NG), and the ICBG's corrugation width ($\Delta w = w_1 - w_0$), and the wavelength (λ). Figure 1 illustrates
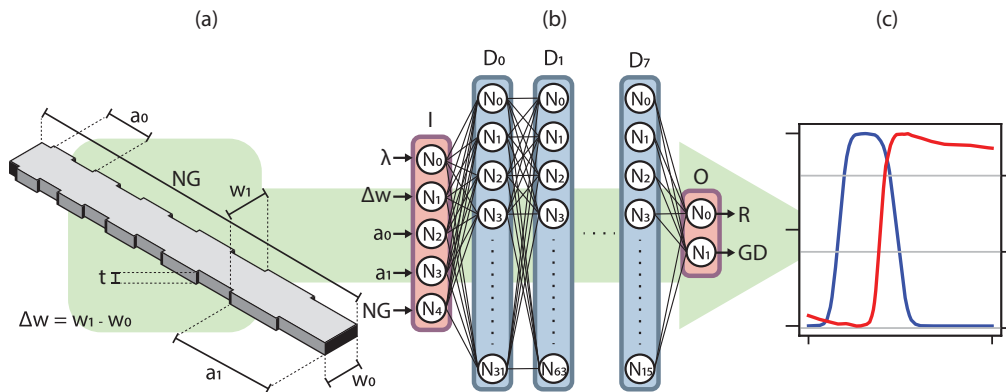

Fig. 1. Summary of the ICBG parameterization scheme and corresponding ANN architecture. (a) The ICBG is parameterized by the optical wavlength (λ) the length of the first period (a_0), the length of the last period (a_1), the corrugation width difference (*Deltaw*), and the number of grating periods (NG). (b) The trained ANN architecture that models the ICBG's inputs and the corresponding reflection (R) and group delay profiles (GD). The ANN consists of 8 deep layers with 32, 64, 128, 256, 128, 64, 32, and 16 neurons respectively. Each layer uses hyperbolic tangent activation functions. (c) The corresponding reflection and group delay profiles for the particular ICBG.

an ICBG with each of these design parameters, along with our chosen ANN architecture. We trained the ANN to output the reflection and group delay spectra of the simulated ICBG.

3. Data generation

To accurately and efficiently simulate the the ICBG reflection and group delay responses for our training-set, we used a layered-dielectric media transfer matrix method (LDMTMM) accelerated by machine learning waveguide models [16]. We could have opted to use other methods involving coupled mode theory or even fully vectorial Maxwell equations solvers [17]. Generally speaking, each method exchanges certain degrees of accuracy for computational simplicity and speed. The LDMTMM method, however, provides sufficiently comparable results to fully vectorial methods at a fraction of the computational cost [1].

The method discretizes the ICBG into individual dielectric slabs, models each slab as an ideal waveguide, and propagates the fields through each slab using a transfer matrix. The effective index of each section is modeled using another ANN that parameterizes the wavelength as a function of waveguide width and thickness. This process is repeated for every wavelength point of interest.

We simulated over 100,000 grating configurations at 250 wavelength points from 1.45 μm to 1.65 μm resulting in approximately 25,000,000 training points (i.e. 100,000 gratings x 250 wavelength points = 25 million training samples). We swept through 10 different corrugation widths, 11 different ICBG lengths, and 961 different chirping patterns. More information regarding the dataset generation process is found in [16].

For a tool intended to perform parameter extraction on fabricated devices, it is important to use a generalized and abstracted model insensitive to minor fabrication defects. For example,
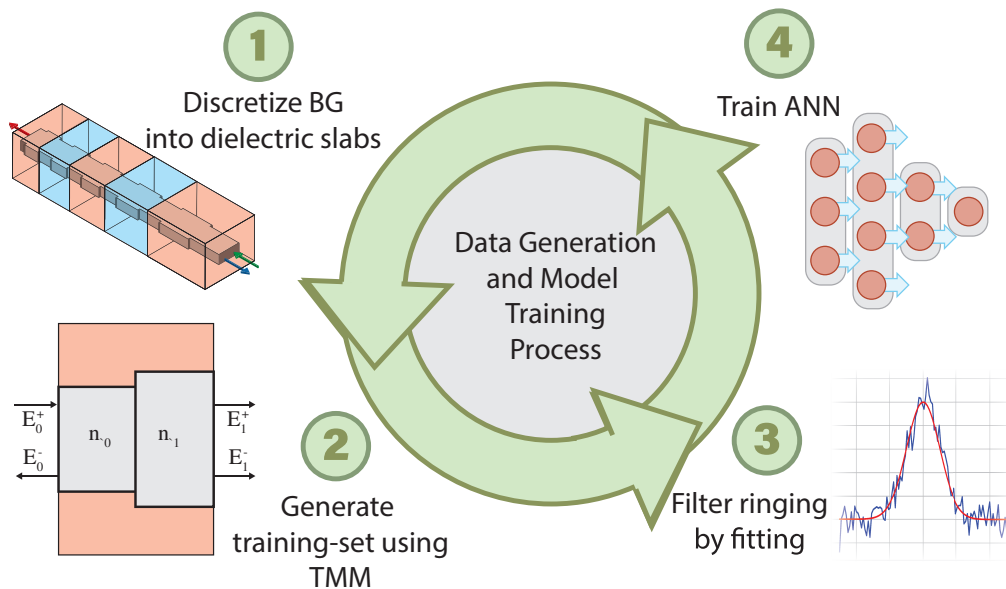

Fig. 2. ANN modeling process. First, several different ICBGs are discretized into individual dielectric layers (1). Then, the reflection and group delay profiles are simulated using the transfer matrix method (2). The apodization dependent ringing is then filtered by fitting the curves to modified Gaussians (3). This dataset is then fed into a ANN training algorithm (4). Often, this process must be repeated until the ANN can suitably express a large enough ICBG design space.

small changes in the ICBG apodization profile greatly alter the expected spectral ringing, but not necessarily the spectral bandwidth (assuming the index contrast is not dramatically altered). Furthermore, parameterized high frequency ringing is rather difficult to capture efficiently using ANNs. We overcame these challenges by fitting the LDMTMM group delay and reflection profiles prior to training to a generalized skewed Guassian function of the form

$$f(\lambda, \lambda_0, \sigma, \beta, a, p, c) = \frac{a\sigma}{\gamma} e^{\frac{-\beta|\lambda-\lambda_0|^p}{\gamma}} + c \tag{2}$$

where

$$\gamma = \frac{2\sigma}{1 + e^{-\beta(\lambda-\lambda_0)}} \tag{3}$$

Our resultant dataset corresponds to a much larger and practical parameter space and significantly alleviates the ANN training process.

Once the dataset was generated and processed, we proceeded to train the ANN. Often, this process must be repeated until a suitable parameter space is simulated. This design flow is illustrated in Fig. 2.

4. ANN training

To identify a suitable ANN architecture, we performed a hyper-parameter optimization (HPO), where several ANNs with different architectures were simulated simultaneously. We swept through common ANN architecture components, like the number of layers, the number of neurons for each layer, each neuron's activation function, and the batch size. We concurrently trained 1200 different ANNs on 1200 cores using Brigham Young University's Fulton Supercomputing Lab and the TensorFlow package [18]. Each simulation took approximately 12 hours. Figure 3
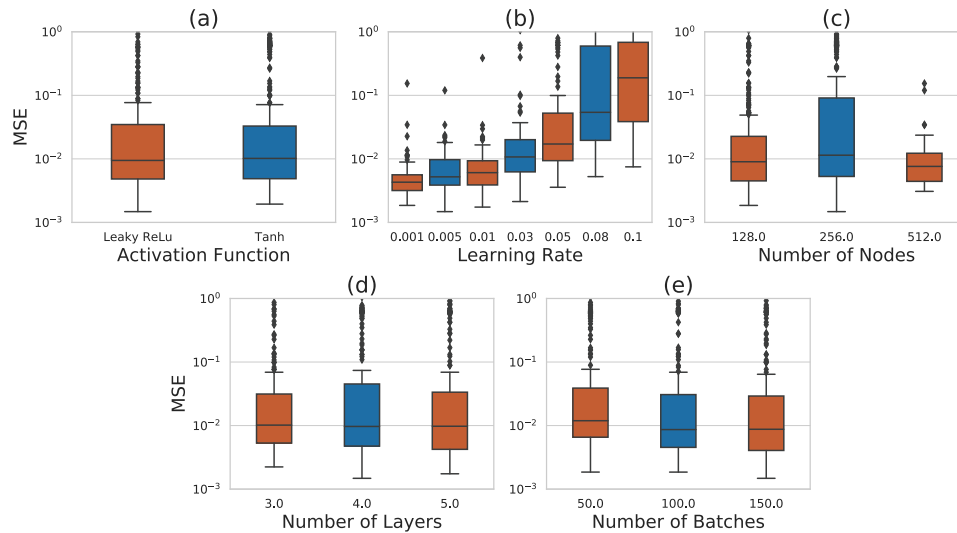

Fig. 3. Hyper-parameter optimization used to determine a suitable architecture for the ANN. We swept through various parameters like the activation function (a), the optimizer's learning rate (b), the number of neurons (c), the number of layers (d), and the number of batches per epoch (e). Each box and whisker plot illustrates the distribution of a particular parameter with reference to its MSE after the final epoch. While some parameters showed little influence (number of layers, activation functions, etc) others greatly affected the MSE convergence (learning rate).

illustrates the HPO's results. We measured the accuracy and effectiveness of each network by tracking the mean squared error (MSE) and coefficient of determination (R^2) for all simulated permutations.

From the HPO, we chose to train an ANN with 8 layers. Each layer had 32, 64, 128, 256, 128, 64, 32, and 16 neurons respectively. Each neuron used a leaky ReLu activation function. No dropout was used.

5.   Device fabrication, measurement, and calibration

We designed 11 different ICBGs each with a linear chirp of 6 nm. We designed 5 of the devices with a reversed chirping, such that their resultant group delay profiles would be mirror images of their counterparts. Some devices were 750 grating periods long and the others were 250. We chose corrugation widths of 30 nm and 50 nm. The devices with less than 750 grating periods had too little SNR to reliably extract their grating parameters.

To efficiently extract the reflection, transmission, and group delay profiles of the same ICBG, we designed an interrogator circuit using various Y-branches, directional couplers, and grating couplers. Figure 4 illustrates the circuit. We used the grating couplers to direct light on and off
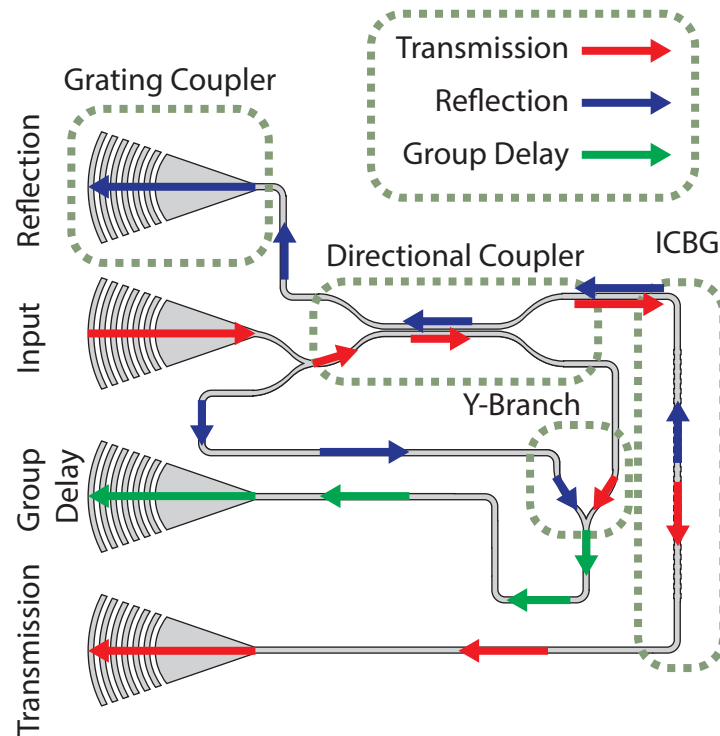

Fig. 4. Interrogation circuit used to extract the reflection, transmission, and group delay profiles of a single ICBG simultaneously. First, light enters the chip via the second grating from the top. It continues through a Y-branch splitter and a directional coupler until it reaches the Bragg grating. The light transmitted through the Bragg grating leaves the chip via the fourth grating coupler (light path in red). The Light that is reflected by the Bragg grating returns through the directional coupler, where half of it is routed off the chip via the first grating coupler (light path in blue). The other half of the reflected light is interfered with the original transmission signal using the directional coupler and an additional Y-branch (light path in green). This interference pattern is routed off of the chip with the third grating coupler. The group delay is then extracted from this interference pattern.

of the chip. We routed the light using the Y-branches and directional couplers. We interfered the reflection signal with the original reference signal using a Mach-Zhender Interferometer (MZI) in order to measure the group delay information.

Our devices were fabricated at the University of Washington in collaboration with the University of British Colombia and the SiEPIC program on a 150 mm silicon-on-insulator (SOI) wafer with 220 nm thick silicon on 3 μm thick silicon dioxide and a hydrogen silsesquioxane resist (HSQ, Dow-Corning XP-1541-006). Electron beam lithography was performed using a JEOL JBX-6300FS system operated at 100 keV energy [19], 8 nA beam current, and 500 μm exposure field size. The silicon was removed from unexposed areas using inductively coupled plasma etching in an Oxford Plasmalab System 100. Cladding oxide was deposited using plasma enhanced chemical vapor deposition (PECVD) in an Oxford Plasmalab System 100.

To characterize the devices, a custom-built automated test setup [1] with automated control software written in Python was used. An Agilent 81600B tunable laser was used as the input source and Agilent 81635A optical power sensors as the output detectors. The wavelength was swept from 1500 to 1600 nm in 10 pm steps. A polarization maintaining (PM) fibre was used to maintain the polarization state of the light, to couple the TE polarization into the grating couplers [20]. A polarization maintaining fibre array fabricated by PLC Connections (Columbus OH, USA) was used to couple light in/out of the chip. The devices with 10 nm corrugation widths failed to provide sufficient index contrast needed to measure a noticeable grating response, and were omitted from the rest of the analysis.

To estimate the reflection and group delay profiles from the measurement data, we calibrated out the band-limited spectral responses induced by the grating couplers, directional couplers, and Y-branches. Figure 5 illustrates this process for both the reflection and group delay data. For the reflection measurements, we first fit the data outside of the ICBG's bandwidth to a fourth order polynomial. We use this polynomial fit to remove the couplers' responses. We then relocate the noise floor by fitting, once again, the data outside of the ICBG's bandwidth. We note that this method is sensitive to fabrication defects within the grating couplers, which shift the center band of the coupler's response.

Various methods involving windowed Fourier transforms and curve fits are commonly used to extract the group delay. We opted to estimate the free spectral range (FSR) of the interferometer using a peak-tracking algorithm. From the FSR, along with the relative path length difference ($L_r ef$) of approximately 200 μm, we can estimate the group delay (τ) using

$$\tau(\lambda) = \frac{(L_{ref} - L(\lambda)) \cdot n_g(\lambda)}{c} \tag{4}$$

where

$$L(\lambda) = \frac{\lambda^2}{FSR \cdot n_g(\lambda)} \tag{5}$$

and $n_g(\lambda)$ is the group index of the reference arm waveguide. Given a sampling width of 10 pm, the maximum peak detection error is within 20 pm. The peak tracking method consequently predicts the group delay with 10 fs tolerances, well within the error induced by small scattering defects.
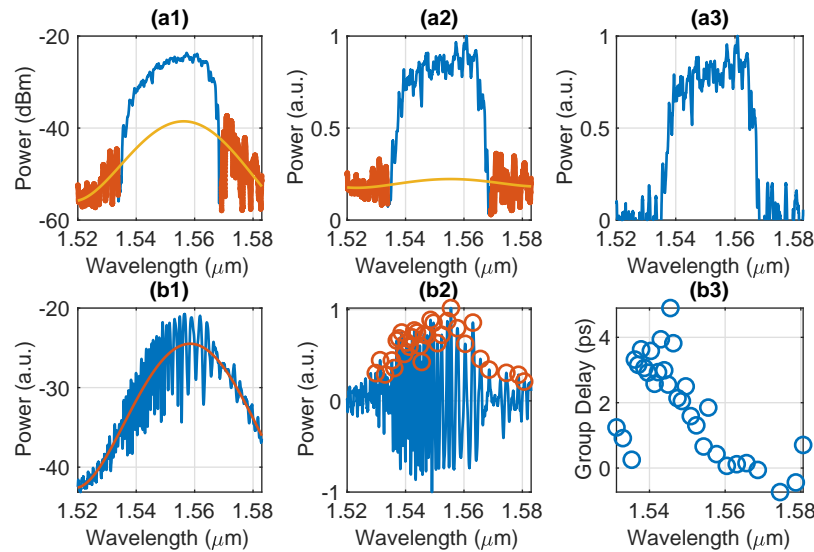
Fig. 5. Calibration process used to extract the measured reflection and group delay responses. The reflection data is first fit to a fourth order polynomial outside of the expected bandwidth in order to remove the grating couplers' transfer function (a1). Next, the data is once again fit to a fourth order polynomial outside of the device's bandwidth to identify the noise floor (a2). The data is then normalized to unit power (a3). Similar to the reflection data, the group delay data is also fit to a fourth order polynomial to remove the grating couplers' response (b1). Next, the FSR is approximated using a peaktracking algorithm (b2). From the FSR, the group delay is estimated (b3).

6. Experimental results

To estimate the actual fabrication parameters of the ICBGs, we used the ANN in conjunction with a nonlinear least squares fitting routine within the SciPy package [21]. The routine initializes by calling the ANN using the original design parameters. The simulated reflection and group delay profiles are directly compared to the measurement data. From the residuals, the algorithm decides whether the current design is sufficiently similar to the measurement data or if further simulation is needed. The sufficiency criteria is determined by a difference in iteration mean-squared-error (MSE) of 10^{-8}. Figure 6 illustrates this procedure.

Since the ICBG has fabrication limits that can be cast as parameter bounds, we chose to run a Trust Region Reflective (TRF) optimization algorithm within the nonlinear solver [22]. Specifically, we bounded the first and last ICBG periods (a_0 & a_1) between 312 nm and 328 nm and the corrugation width between 1 nm and 50 nm. The number of periods was fixed.

We chose to extract parameters of three different ICBGs. After just 5 minutes of optimization on a Macbook Air 2012 (1.8 GHz Intel Core i5, 4 GB 1600 MHz DDR3 RAM), the solver converged on new parameters for all three devices that more reasonably reflect the measurement data. Figure 7 illustrates the algorithm's results compared to the fabrication data and the original design spectra. Table 1

Not only do the algorithm's profiles match the data much better, but the extracted parameter differences are expected from the processes used to fabricate the devices. For example, the algorithm predicts a slightly wider chirping bandwidth and smaller corrugation width for all three devices. The E-beam raster grid's resolution approaches the chirping resolution of the ICBG (1 nm), so "snapping" from one grid point to the next results in slightly wider chirping
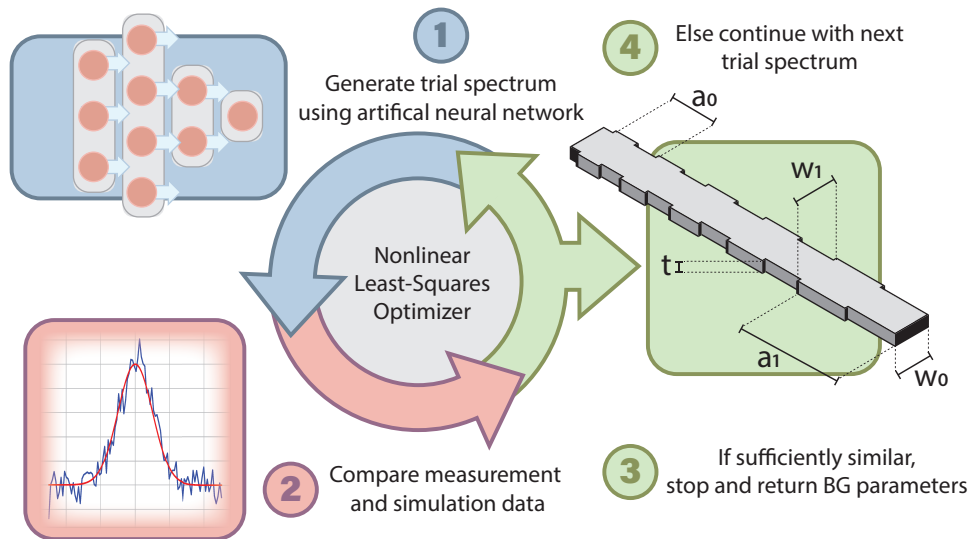
Fig. 6. Efficient and robust method to extract fabricated ICBG device parameters using ANNs and a nonlinear least-squares optimizer. First, the ANN simulates reflection and group delay spectra for the device's initial design parameters (1). Then, the simulations are compared directly to the measured data (2). If the results are sufficiently similar, the optimizer returns the device parameters (3). If not, the optimizer strategically simulates a new set of device parameters based on the residual error (4).
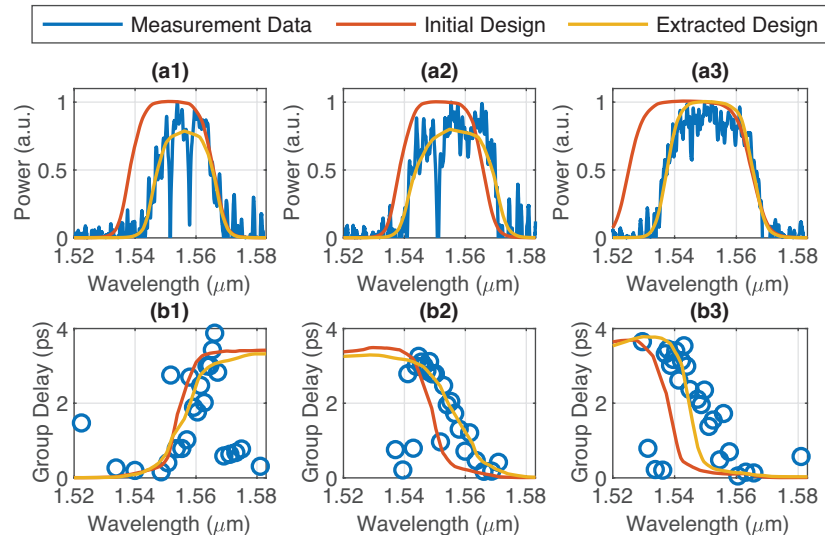

Fig. 7. The extracted reflection (a1, a2, a3) and group delay (b1, b2, b3) profiles (yellow) compared to the initial design profiles (red) and the calibrated measurement data (blue).

Table 1. Parameter extraction results for three separate devices. The design parameters are compared directly to the algorithm's extracted parameters for each device.

Design Parameters	Device 1		Device 2		Device 3	
	Design	Extracted	Design	Extracted	Design	Extracted
a_0 (nm)	324	324.3	318	315.8	318	320.1
a_1 (nm)	318	317.0	324	325.6	324	323.7
NG	750	750	750	750	750	750
Δw (nm)	30	13.7	30	15.7	50	39.3

bandwidths. The E-beam's resolution, along with the etch process, also tend to round the sharp ICBG corners, resulting in lower net corrugation width.

Other small differences between the extracted parameter sets and the fabricated data, like the fabry-perot resonances, are difficult to model with the current ANN abstraction. It would require a much more sophisticated, and possibly impractical, parameterization to capture these defects. Despite these small discrepancies, the fitting algorithm and ANN demonstrate a strong ability to extract parameters for complex silicon photonic devices.

7. Discussion

We demonstrate a novel silicon photonic parameter extraction method using artificial neural networks. Our method is capable of extracting parameters for complicated devices, like integrated chirped Bragg gratings, without sacrificing the speed of traditional analytic methods. To validate our method, we fabricated and measured various integrated chirped Bragg gratings and extracted the actual parameters.

We are confident in the method's accuracy, but also note several important considerations that other researchers should take into account when implementing this approach. First, biases induced by a dataset or the training process itself may not be readily apparent. While several training techniques exist to monitor such problems [13], they do not guarantee that the model is free from bias. Consequently, "black-box" implementations of this method should be prefaced by careful boundary case testing and verification as was performed for the devices presented here. The current method's rigorous data generation process and training procedure is further described in [16] and adds significant confidence to this approach.

Second, this model's parameterization scheme was rather basic and meant to characterize a few essential ICBG parameters, rather than the device's complete structural makeup. Future work could continue to parameterize this model to encompass parameters like apodization and etch depth. Despite this shortcoming, however, the current model sufficiently describes the parameters that most directly affect the ICBG's reflection and group delay profiles.

Machine learning models exhibit a unique property known as "transfer learning", whereby someone can take a preexisting model trained over a particularly narrow parameter space and expand its knowledge domain by continuing to train over a new dataset [23]. This possibility enables model "crowdsourcing", where developers and designers simulate datasets relevant to their particular problem and share a common model between everyone. We encourage this mindset when both training, sharing, and using this parameter extraction framework.

Acknowledgments

Disclosures

The authors declare that there are no conflicts of interest related to this article.

References

1. L. Chrostowski and M. Hochberg, *Silicon Photonics Design: From Devices to Systems* (Cambridge Univ. Press,, 2015).
2. W. Bogaerts and L. Chrostowski, "Silicon Photonics Circuit Design: Methods, Tools and Challenges," Laser Photonics Rev. **12**(4), 1700237 (2018).
3. S. K. Selvaraja, "Wafer-scale fabrication technology for silicon photonic integrated circuits," Ph.D. thesis (Ghent University, 2011).
4. 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**(9), 9712 (2017).
5. W. A. Zortman, D. C. Trotter, and M. R. Watts, "Silicon photonics manufacturing," Opt. Express **18**(23), 23598–23607 (2010).
6. X. Wang, W. Shi, H. Yun, S. Grist, N. A. F. Jaeger, and L. Chrostowski, "Narrow-band waveguide Bragg gratings on SOI wafers with CMOS-compatible fabrication process," Opt. Express **20**(14), 15547–15558 (2012).
7. M. J. Strain and M. Sorel, "Design and Fabrication of Integrated Chirped Bragg Gratings for On-Chip Dispersion Control," IEEE J. Quantum Electron. **46**(5), 774–782 (2010).
8. L. Chrostowski, X. Wang, J. Flueckiger, Y. Wu, Y. Wang, and S. T. Fard, "Impact of Fabrication Non-Uniformity on Chip-Scale Silicon Photonic Integrated Circuits," in *Optical Fiber Communication Conference*, (OSA, San Francisco, California, 2014), p. Th2A.37.
9. X. Chen, Z. Li, M. Mohamed, L. Shang, and A. R. Mickelson, "Parameter extraction from fabricated silicon photonic devices," Appl. Opt. **53**(7), 1396 (2014).
10. D. Melati, A. Alippi, and A. Melloni, "Waveguide-Based Technique for Wafer-Level Measurement of Phase and Group Effective Refractive Indices," J. Lightwave Technol. **34**(4), 1293–1299 (2016).
11. Y. Xing, J. Dong, S. Dwivedi, U. Khan, and W. Bogaerts, "Accurate extraction of fabricated geometry using optical measurement," Photonics Res. **6**(11), 1008 (2018).
12. K. Hornik, "Approximation capabilities of multilayer feedforward networks," Neural Networks **4**(2), 251–257 (1991).
13. S. S. Haykin, *Neural Networks and Learning Machines* (Prentice Hall, 2009).
14. Y. Lecn, "A Theoretical Framework for Back-Propagation," in Proceedings of the 1988 Connectionist Models Summer School (1988), pp. 21–28.
15. J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," Neural Networks **61**, 85–117 (2015)..
16. A. M. Hammond and R. M. Camacho, "Designing Silicon Photonic Devices using Artificial Neural Networks," arXiv:1812.03816 [physics] (2018).
17. R. Helan, *Comparison of Methods for Fiber Bragg Gratings Simulation* (IEEE, 2006), pp. 161–166.
18. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* (2015).
19. R. J. Bojko, J. Li, L. He, T. Baehr-Jones, M. Hochberg, and Y. Aida, "Electron beam lithography writing strategies for low loss, high confinement silicon optical waveguides," J. Vac. Sci. Technol. B **29**(6), 06F309 (2011).
20. Y. Wang, X. Wang, J. Flueckiger, H. Yun, W. Shi, R. Bojko, N. A. Jaeger, and L. Chrostowski, "Focusing sub-wavelength grating couplers with low back reflections for rapid prototyping of silicon photonic circuits," Opt. Express **22**(17), 20652–20662 (2014).
21. E. Jones, T. Oliphant, and P. Peterson, "SciPy: Open source scientific tools for Python," (2001).
22. M. A. Branch, T. F. Coleman, and Y. Li, "A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems," SIAM J. Sci. Comput. **21**(1), 1–23 (1999).
23. S. J. Pan and Q. Yang, "A Survey on Transfer Learning," IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010).