Finding the Right Deep Neural Network Model for Efficient Design of Tunable Nanophotonic Devices

Jung, Minwoo; Kojima, Keisuke; Koike-Akino, Toshiaki; Wang, Ye; Zhu, Dayu; Brand, Matthew

TR2022-047 May 20, 2022

Abstract

We develop generative deep neural networks that explore relevant statistical structures to expedite a complex inverse design of nanophotonic on-chip wavelength demultiplexer. Our design, targeting at telecomm-wavelengths, is electrically switchable via liquid crystal tuning.

Conference on Lasers and Electro-Optics (CLEO) 2022

Finding the Right Deep Neural Network Model for Efficient Design of Tunable Nanophotonic Devices

Minwoo Jung[1,2], Keisuke Kojima[1], Toshiaki Koike-Akino[1,*], Ye Wang[1], Dayu Zhu[1,3], Matthew Brand[1]

[1] *Mitsubishi Electric Research Laboratories (MERL), 201 Broadway, Cambridge, MA 02139, USA.*
[2] *Department of Physics, Cornell University, Ithaca, NY 14853, USA.*
[3] *School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.*
**koike@merl.com, brand@merl.com*

Abstract: We develop generative deep neural networks that explore relevant statistical structures to expedite a complex inverse design of nanophotonic on-chip wavelength demultiplexer. Our design, targeting at telecomm-wavelengths, is electrically switchable via liquid crystal tuning.

1. Introduction: In recent years, generative deep neural networks (DNN) have been successfully applied to time-efficient inverse design of photonic devices [1]. As the complexity of a system grows, there arise multiple possible ways to model how the system variables relate to each other physically and statistically. The performance of a generative DNN model can depend significantly on the underlying statistical structure (a.k.a., Bayesian graphs), thus searching for a right Bayesian-inference model prior to parameter tuning could potentially provide a more systematic way of improving the DNN performance. In this work, we adopted an advanced approach called AutoBayes [2] to explore different network structures in the context of inverse design of complex photonic devices, focusing on a tunable wavelength splitter.

2. LC-Tunable Wavelength Splitter: Our target device structure is based on the compact on-chip wavelength de-multiplexer (1 input and 2 output ports), which is electrically tunable with a liquid crystal (LC) over nanophotonic circuits, such that the outputs are swapped when the LC is on. Using the LumOpt numerical package provided by Lumerical, we first perform the adjoint optimization method [3] for the following target responses: For $T_{1;\mathrm{ON}}(\lambda_1)$, $T_{2;\mathrm{ON}}(\lambda_2)$, $T_{1;\mathrm{OFF}}(\lambda_2)$, and $T_{2;\mathrm{OFF}}(\lambda_1)$, the target is T_{high}, while for $T_{1;\mathrm{ON}}(\lambda_2)$, $T_{2;\mathrm{ON}}(\lambda_1)$, $T_{1;\mathrm{OFF}}(\lambda_1)$, and $T_{2;\mathrm{OFF}}(\lambda_2)$, the target is T_{low} where $T_{1/2;\mathrm{ON/OFF}}$ is the transmission of silicon-on-insulator (SOI) waveguide mode from the input port to the output port 1/2, depending on the LC condition of either ON (e-axis along out-of-plane direction) or OFF (e-axis perpendicular to the input waveguide). All transmission values are averaged over 5nm bandwidth around the target wavelengths $\lambda_{1,2} \pm 2.5$nm. Fig. 1(a) shows an example device topology exhibiting average extinction ratio (ER) of 16.5dB for $\bar{\lambda} = (\lambda_1 + \lambda_2)/2 = 1517$nm and $\Delta\lambda = |\lambda_1 - \lambda_2| = 47$nm, and the resulting device response is drawn in Fig. 1(b).

3. DNN Models: The goal of the inverse design in this device is to generate a useful device topology (denoted as T), given a desired device performance, e.g., transmission spectra (denoted as S). In principle, however, the users of a generative model may not know or care about the entire spectral response. The users' demand would be most likely in the form of some partial information of the spectra (denoted as S'), such as transmission levels at specific wavelengths $S' = [\bar{\lambda}, \Delta\lambda, ER]$. Depending on how we make use of these system variables T, S, and S', the resulting DNN architecture will vary. Fig. 2(a) illustrates three possible ways to construct a pair of a generative model and its inference model (see [2] for the details on how the interpretation of these graph pairs is made). The first model 'Simple $S' \to T$' assumes that the device topology T depends only on the partial spectra information S'. Then, the second model 'ACVAE $S' \to T$' assumes the existence of a latent variable Z, and ACVAE (adversarial

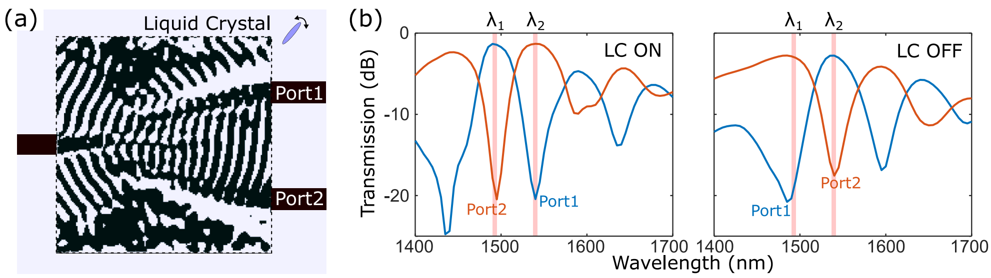

Fig. 1. (a) Optimized device topology (black: silicon, pink: LC, dashed box: optimization area) and (b) transmission spectra of an example device generated by DNN.

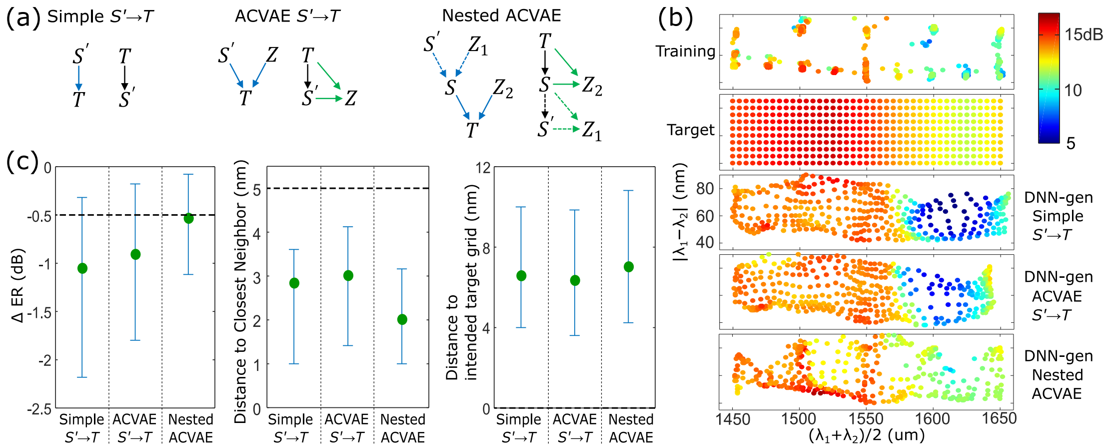

Fig. 2. (a) Three DNN architectures. (b) Extinction ratio (colormap) of training dataset devices, validation targets, and DNN-generated devices out of three architectures. (c) Comparison of three DNNs; left: extinction ratio difference between generated devices and intended target values showing that the nested ACVE model generates the best ER on average, middle: distance (in $\bar{\lambda}$-$\Delta\lambda$ plane) from each generated point to its closest neighbor, right: distance between the generated devices and the intended target grid point. The dots are median values, and the bars are 25% and 75% quantiles. Horizontal dashed lines correspond to the intended ideal performance.

conditional variational auto-encoder [1])-type DNN architecture is required. The third model 'Nested ACVAE' assumes a step-by-step statistical dependence from S' to S and to T.

4. Design Method: The DNN training dataset was prepared by the following procedure: (1) optimize a device topology targeted for $\bar{\lambda} = 1550$nm starting from a random initial condition, given a fixed separation $\Delta\lambda = 45$nm; (2) take the topology optimized at $\bar{\lambda} = 1550$nm as the initial condition for the optimization targeting for $\bar{\lambda} = 1575$nm; (3) cascade the previous step for every 25nm of target $\bar{\lambda}$ values; and (4) cascade the design for $\Delta\lambda = 80$nm at $\bar{\lambda} = 1500$nm and 1600nm, and $\Delta\lambda = 60$nm and 80nm at $\bar{\lambda} = 1450$nm and 1550nm. The steps (1)–(4) ensure smoothness in T across varying S' within a series of such cascades. The adjoint optimization produces many ($\sim$100) intermediate sub-optimal results en route to the final optimal design.

5. Results: Fig. 2 (b) shows the distribution of the training dataset devices (topmost). Also, it shows the target S' values that we used to generate new inverse designs by trained networks (second topmost). To be specific, the target S' were given by ($\bar{\lambda}$, $\Delta\lambda$) grid points at every 5nm in each dimension with the target ER values that are 0.5dB higher than the values available in the training dataset. Lastly, we display the results of network validations for three different DNN models. Fig. 2 (c) then compares the performance of four DNN models in terms of the ER performance of the generated devices (left), uniform spread of generated datapoint distribution (middle), and the agreement between the intended ($\bar{\lambda}$,$\Delta\lambda$) values and the generated points (right). The dashed lines illustrate the intended performance. 'Nested ACVAE' shows the best performance compared to other models in regard to the ER, where half of the generated devices exceed the training data. In terms of smoothness of the interpolation or, in other words, uniformity of generated data point distribution, the other two networks are slightly better as seen in Fig. 2 (b). Overall, AutoBayes model exploration reveals that the inclusion of S helps to improve the network outputs, yet slightly reduces the smoothness of the interpolation. To design one device using the adjoint method takes about 1.5 day using a computing cluster. Even though it takes 4 hours to train the DNN, to generate and validate the 410 devices takes about 9 hours. This shows that DNN has the potential to cover the whole target parameter space in a short time, without using the adjoint method to design each device individually.

6. Summary: We demonstrated that DNN model exploration could be utilized for the efficient inverse design of tunable nanophotonic wavelength splitters. Specifically, the nested ACVAE model found in AutoBayes showed superior performance than state-of-the-art ACVAE model, achieving an average extinction ratio of about 13 dB over wide wavelengths even with a limited number of training data.

References

1. Y. Tang et. al., "Generative deep learning model for inverse design of integrated nanophotonic devices," Laser Photonics Rev. **14**(12), 2000287 (2020).
2. A. Demir, T. Koike-Akino, Y. Wang, D. Erdogmus, "AutoBayes: Automated Bayesian graph exploration for nuisance-robust inference," IEEE Access **9**, 39955–72 (2021).
3. A. Y. Piggott et. al., "Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer," Nat. Photon. **9**, 374-7 (2015).