Supplemental document accompanying submission to *Optics Express*

**Title:** Neural Networks enabled Forward and Inverse Design of Reconfigurable Metasurfaces

**Authors:** Ibrahim Tanriover, Wisnu Hadibrata, Jacob Scheuer, Koray Aydin

**Submitted:** 9.05.2021 07:56:38
Neural Networks enabled Forward and Inverse Design of Reconfigurable Metasurfaces: supplemental document

Ibrahim Tanriover¹, Wisnu Hadibrata¹, Jacob Scheuer² and Koray Aydin¹*

¹ Department of Electrical and Computer Engineering, Northwestern University, Evanston, Illinois 60208, United States
² School of Electrical Engineering, Department of Physical Electronics, Tel-Aviv University, Tel Aviv 69978, Israel
a) Corresponding author: aydin@northwestern.edu
1. Methods

Full electromagnetic simulations based on finite difference time domain (FDTD) method are performed using commercial simulation package Lumerical FDTD Solutions to generate the dataset used in this work. Square unit cell geometry (same periodicity along both of the horizontal direction) is adopted wall-to-wall distance between adjacent nanodisks to be 500 nm. Periodic boundary conditions are applied at all horizontal boundaries (x and y), and perfectly matching layers (PMLs) are employed at vertical boundaries (z). Substrate thickness is assumed to be infinite by placing bottom PML boundary inside the substrate, and the built-in sapphire (Al2O3, $n_{\text{sub}} \approx 1.72$) model is used for the substrate. TM polarized plane wave is used as light source, which is also located inside the substrate illuminating towards +z direction. The optical response (complex reflection and transmission coefficients) is obtained from near field monitors located above (transmission) and below (reflection) the unit cell structure. Operating spectral range is defined from 2 μm to 4 μm wavelengths, and the optical response is sampled at 501 data points with equal wavelength spacing. The radius is changed from 250 nm to 1500 nm with a constant step size of 50 nm. The height is tuned from 40 nm to 200 nm with 20 nm step size. 30 complex refractive index values are chosen randomly 2+0j to 10+10j, where 5 of them are forced to have 0 (zero) k to shift the dataset a bit toward dielectric side (See Supplement 1 for employed complex refractive indices). The resulted values are set as the material indices at the midpoint of training spectral range ($\lambda=3$ μm), and fitted by, and obtained from, the simulation program itself for the rest of the spectrum. These, the fitted $n(\lambda)+jk(\lambda)$ values are used in the feature matrix. As a result, 30x26x9=7020 samples are created as dataset. The simulations took approximately 2 days (48 hours). 1/10 of these samples are randomly separated as validation set, and the rest is used in training. The total size of the entire dataset is 366.7 MB.

For all complex-valued networks network Keras-Complex library “complexnn” [1] and Keras framework are used. We train all the models using Adam optimizer [2], mean squared error (MSE) as the loss function and ‘tanh’ as activation function.
2. Refractive Indices of Data Set

Figure S1 represents the complex refractive indices at the mid-point of training spectral range ($\lambda=3$ μm). 30 $n$-$k$ pairs are randomly selected (from 1 to 10 for $n$ and 0 to 10 for $k$), where 5 of them were forced to have $k=0$ in order to increase networks accuracy on predicting lossless dielectric materials. The complex refractive indices are set for $\lambda=3$ μm, and fitted over entire training spectral range (from 2 μm to 4 μm) by the commercial simulation program (Lumerical FDTD Solutions). The fitted values are used in the training and validation processes.

Fig. S1. Complex refractive indices at $\lambda=3$ μm used in the training data set.
3. Comparison of Complex-Valued and Real-Valued Networks

Figure S2 represents a comparison of complex-valued and real-valued neural networks (CVNNs and RVNNs). The networks have the same structure of 5 hidden layers with 20,80,120,80, and 20 neurons, as indicated in the Figure 1a of the main script.

Although the implementation of CVNN seems to be similar to a real-valued NN with doubled number of channels at each step, the calculations performed within the complex-valued layers are not linearly independent [3]. Comparing complex multiplication of our complex-valued feature x+jy with complex-valued weight u+jv, with scalar vector multiplication of the corresponding vectors [x y] and [u v], respectively, provides a good example to understand the main difference. The former calculation results in a complex-valued output having 2 degrees of freedom as real and imaginary part. When represented as phasors, this multiplication corresponds to scaling and rotating the complex-valued features. The latter, on the other hand, results in a 2x2 matrix having 4 degrees of freedom corresponding to reflecting, shearing, scaling and rotating the initial complex-valued feature. Real-valued networks lose the mathematical correlation between real and imaginary components due to these additional degrees of freedom.

Fig. S2. Comparison of real- and complex-valued networks. a, b The training curve of CVNNs and RVNNs that are modelling complex transmission($S_{21}$) and reflection($S_{11}$) coefficients. The training curves along 150 epochs are shown in the insets, where the y-axes are linear. c, d Probability density function (Pdf) and cumulative density function (Cdf) of mean square error (MSE) values for the CVNN and RVNN models of $S_{21}$, where 95% confidence borders are indicated by blue (CVNN, MSE=2.0x10^{-4}) and red (RVNN, MSE=3.1x10^{-4}) dashed lines. The highest error (MSE) values noted as 1.2x10^{-3} and 2.4x10^{-3} for CVNN and RVNN, respectively.
Table S1. Comparison of mean and standard deviation of MSE values of CVNN and RVNN

|                | CVNN        | RVNN        |
|----------------|-------------|-------------|
| Mean (μ)       | 9.60x10^-5  | 9.80x10^-5  |
| Standard deviation (σ²) | 8.74x10^-5  | 1.13x10^-4  |

As seen in the Figure S2a,b, CVNNs have higher learning rate (|Δ(MSE)| per epoch). Although the real-valued network closes the gap as the models converge, complex-valued network maintains higher accuracy.

Another aspect that is improved by the use of complex-valued networks is consistency. Probability density function (Pdf) and standard deviation (σ²) of MSE values are two important indicators of the consistency of models’ accuracy over the dataset. As seen in Figure S2c and Table S1, CVNN has narrower Pdf and smaller σ² than RVNN. Another important measure is cumulative density function (Cdf). As seen in Figure S2c,d, the 95% of sample sets (combinations of R,H, and n+jk) have MSE ≤ 2.0x10^-4 and MSE ≤ 3.1x10^-4, and maximum error are noted as 1.2x10^-3 and 2.4x10^-3 for complex-valued and real-valued networks, respectively. These results, overall, indicate higher consistency of CVNN’s predictions than that of RVNN.

Fig. S3. Comparison of overall MSEs of real- and complex-valued networks with respect to number of training samples. The networks are modelling complex transmission(521) coefficient. The network architecture and training conditions are the same with the networks represented in Figure S2.

An alternative way to observe the improvement provided by the use of CVNNs is to examine the networks’ performance with respect to amount of training data as shown in Figure S3, where we compare the overall training errors of CVNN and RVNN. As seen in the figure, the CVNN noticeably outperforms the RVNN in the small-data regime, and the difference evanesce as the networks are saturated with training data.

All in all, the results provided above indicates the advantage of preserving the mathematical correlation of complex values. However, as the problem here is low dimensional and only 1 of the 4 input parameters is complex-valued, the improvements provided by complex-valued networks are small. Larger improvements are expected for more complex problems.
4. Additional Examples from the Validation Set

Figure S4 represents the reflection response (complex reflection coefficient, reflectance and phase shift) of examples from the validation set, of which the transmission response is shown in the main script.

Fig. S4.  a, c Comparison of predicted (real part: blue and imaginary part: red) and simulated reflection coefficients of exemplary samples from the validation set. a Plasmonic material (n~k), H=140 nm, and R=1500 nm. c Dielectric material (n constant, k=0), H=160 nm, and R=600 nm. b, d Corresponding transmission and phase values of unit cells that are indicated at a and c, respectively.

Fig. S4. a, c Comparison of predicted (real part: blue and imaginary part: red) and simulated reflection coefficients of exemplary samples from the validation set. a Plasmonic material (n~k), H=140 nm, and R=1500 nm. c Dielectric material (n constant, k=0), H=160 nm, and R=600 nm. b, d Corresponding transmission and phase values of unit cells that are indicated at a and c, respectively.
5. Spectral Generalizability

In this work, we applied our previously proposed wavelength normalization method [4] with CVNNs. In the previous study, we have shown that wavelength scaling of geometric parameters provide spectral generalizability. However, this normalization causes one-to-many mapping problem by creating identical input sets corresponding different operation wavelengths. The problem can be solved by including wavelength as an explicit feature, which in turn, destroys spectral generalizability.

Here, to provide spectral generalizability without sacrificing performance of our models, we replace operation wavelength ($\lambda$) a pseudo-feature, which we define as a feature that defines but doesn’t discriminates the sample (the unit structure, in this case). Based on this definition, we define wall-to-wall distance ($d$) as our pseudo-feature. It is a feature since it defines the unit structure, and it does not discriminate a sample from others as it is constant over the data set. It is also normalized to operation wavelength ($d/\lambda$) similar to the other dimensional parameters.

Table S2. Comparison of overall mean squared error values of models employing pseudo-feature and operation wavelength

|             | with $d/\lambda$ | with $\lambda$ |
|-------------|------------------|----------------|
| MSE         | 9.82x10^{-5}     | 9.47x10^{-5}   |

Using $d/\lambda$ instead of $\lambda$, we re-train the CVNN that models complex transmission coefficients of our unit structures. As indicated in Table SII, the overall performance is preserved in the training spectral range.

Fig. S5. Comparison of predicted and simulated transmission and phase responses of two nanodisks operating outside the training spectral range of our transmission model. a Lossless dielectric ($n=3.48$, $k=0$) b Plasmonic material ($n+jk=5.32+j2.76$ at $\lambda=1.55$ μm) Corresponding $R$-$H$ pair is $R=400$ nm - $H=100$ nm, and wall to wall distance is 250 nm.

Figure S5 exhibits prediction capability of our model (with $d/\lambda$) outside the training spectral range for dielectric (S5a) and plasmonic (S5b) materials. As seen in the figure, the network accurately predicts both amplitude and phase of transmitted light outside the training range. Around resonances, it maintains high accuracy yet exhibits slight deviations. Note that the phase shift predictions are projected from training range to operation range with respect to difference between reference values (phase shift introduced by bare substrate) at each spectral range.
6. General Purpose Inverse Network Example Parameters

Supplementary Figure S6 indicates the \( R-H \) and \( n-k \) pairs corresponding to results provided in Figure 5 in main script.

Fig. S6. a The complex refractive indices and b predicted dimensional parameters (\( R-H \) pair) corresponding to results shown in Figure 5b,c in main script.

References

1. J. S. Dramsch, C. Trabelski, O. Bilaniuk, D. Serdyuk, “Complex-Valued Neural Networks in Keras with Tensorflow,” 2019, figshare. Software.
2. D. P. Kingma, J. L. Ba, “Adam: A Method for Stochastic Optimization. In 3rd International Conference on Learning Representations,” ICLR 2015 - Conference Track Proceedings; 2015; pp 1–15, arXiv:1412.6980
3. J. S. Dramsch, M. Lüthje, and A. N. Christensen, "Complex-valued neural networks for machine learning on non-stationary physical data," Comput. Geosci. 146, 104643 (2021).
4. I. Tanrıöver, W. Hadibrata, K. Aydin, “Physics-Based Approach for a Neural Networks Enabled Design of All-Dielectric Metasurfaces,” ACS Photonics 2020, 7 (8), 1957–1964.