Machine learning of all-dielectric core-shell nanostructures: The critical role of the objective function in inverse design

Journal:	*Nanoscale*
Manuscript ID	NR-ART-08-2023-004392
Article Type:	Paper
Date Submitted by the Author:	31-Aug-2023
Complete List of Authors:	Hoxie, David; University of Alabama at Birmingham College of Arts and Sciences, Physics Bangalore, Purushotham; University of Alabama at Birmingham College of Arts and Sciences, Computer Science Appavoo, Kannatassen; University of Alabama at Birmingham College of Arts and Sciences, Physics

SCHOLARONE™
Manuscripts

ARTICLE

Machine learning of all-dielectric core-shell nanostructures: The critical role of the objective function in inverse design

David J. Hoxie[*a], Purushotham V. Bangalore[b], Kannatassen Appavoo[c]

To integrate nanophotonics into light-based technologies, it is critical to elicit a desired optical response from its fundamental component, a nanoresonator. Because a nanoresonator's optical resonance depends strongly on its base material and structural features, machine learning has been comtemplated for enhancing the design and optimization process. However, its accuracy in searching the vast parameter space of nanophotonics still pose unresolved questions. Here we show how the choice of objective functions, in combination with trained neural networks, can drastically change the optimization process — even for a simple nanophotonic structure. To assess how different objective functions select the correct structural parameters that generate a desired optical Mie response, we use a simple core-shell, all-dielectric nanostructure as benchmark. By controlling the proportion of training data, which represents the "experience" level, we also quantify how the various objective functions perform in finding the ground-truth parameters. Our findings demonstrate that certain objective functions exhibit improved accuracy when used with highly "experienced" neural networks. Surprisingly, we also find other objective functions that perform better when paired with less "experienced" neural networks. Taken together, our results emphasize that it is critical to understand how neural networks are coupled to optimization schemes, as evident even when a simple core-shell nanostructure is used.

Introduction

To enhance the performance of an optoelectronic device, a successful strategy is to tailor photonic elements to be active at selected wavelengths. Such optically-resonant devices have been developed in many applications, such as telecommunications[1], energy conversion[2], detection of cancer cells[3], chemical and biological sensing[4], and all-optical signal modulation[5, 6-10]. In nanophotonics, optical responses (i.e., resonances) are controlled by selecting the material and the structural parameters of the fundamental photonic element, a nanoresonator. While free electrons in metallic nanostructures provide strong plasmonic resonances[9], the polarizable bound charges in all-dielectric nanostructures provide strong Mie resonances[9,11,12], with low parasitic loss in the visible. With resonances that can be both electric and magnetic in nature, all-dielectric nanostructures provide high spectral tunability[9,13]. In fact, this enhanced tunability obtained by the presence of a magnetic resonance is still under intense investigation — with goals of higher Q-factors[14-18] and better figure-of-merit (FOM)[17,19, 20-23].

[a.] Department of Physics, University of Alabama at Birmingham, Birmingham AL 35294

[b.] Department. of Computer Science, University of Alabama at Birmingham, Birmingham AL 35294

[c.] Department of Physics, University of Alabama at Birmingham, Birmingham AL 35294

Tailoring an optical response using an all-dielectric nanostructure (i.e., designing a target spectrum) requires searching a vast parameter space. To search this high-dimensional parameter space, machine learning has been touted to offer several benefits over traditional optimization scheme, which rely mostly on optical theory[19-21]. Examples are stochastic optimizations such as particle swarm optimizations[24], genetic algorithms[25-28], and Bayesian optimization[29]. With the availability of cheaper and faster computational resources, unsupervised machine learning methods allow for further classification, identification[22] and optimization[23] without a labelled set of data. A key step in the overall optimization process is to quantify how the proposed solution (i.e., computed spectra) deviate from the target. Since computing (or simulating) spectra can be time consuming[30-33], neural networks have been employed to reduce optimization time[34]. For example, Vai et al. showed how to employ inverted neural networks for optimizing microwave circuits[34]. Such inverted neural networks, as outlined by Hinton et al. [26,34-38], with a similar (but not identical) backpropagation process provides another efficient means of gradient descent optimization. Important for our work, Purifoy et al. demonstrated that such neural networks — trained via optical transfer matrix theory — can generate accurate Mie optical response for multi-layered core-shell nanostructures[26]. With the neural networks being computationally efficient, consideration for training time was lessened [26]. While such methods show promise, they also open new questions as they involve gradient descent — which are known to be influenced by the initialization process and the overall structure of the network[39,40]. Moreover, difficulties in finding the target are compounded with using common objective functions. For example, it is known that sum of squares perform poorly when multiple local minima are present[41]. Convexity of the objective function can also alter the target of the optimization process[36]. For these reasons, probabilistic solutions are often pursued[35,36,41]. Unfortunately, as far as we know, little work has been conducted to understand the interplay between the neural network and the objective function. Understanding this interplay and how critical it is to improve the reliability and accuracy of the inverse design process in nanophotonics system is lacking.

In this letter, we show how a transfer-matrix trained neural network, when paired with various objective functions, predicts vastly different optical responses — even for a simple core-shell all-dielectric nanostructure. We use this architecture specifically because there are only few structural parameters to tune while it can still host multiple resonances. The resonances are a result of the complex interaction among the electric and magnetic multipoles. Here we tune the diameter of the
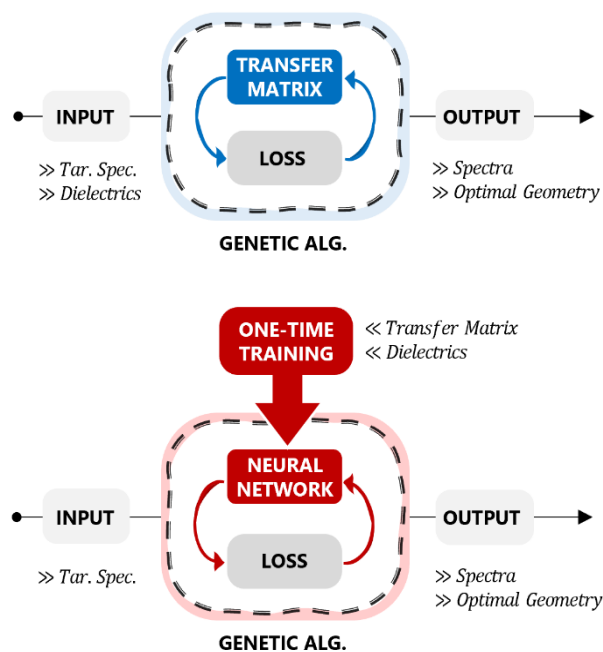

Figure 1: Schematics depicting the inverse design process and comparing **(top)** the traditional computational flow with a genetic (evolutionary) algorithm paired with transfer matrix theory (or an electromagnetic simulation) to **(bottom)** a neural network that has undergone a one-time training to correctly capture transfer matrix theory and then paired with an evolutionary algorithm. The loss function, i.e., the objective function, when coupled to the neural network is the main subject of this work. Note that the top scheme requires using the various dielectric constants of the different materials at the input.

core and the thickness of the shell. Using only these two parameters enables us to manageably explore the parameter space while knowing the ground truth. Knowing the ground truth, in turn, allows us to quantify how accurate the optimization process is. Additionally, we quantify the optimization process when we pair different objective functions with various level of experience of the neural network (i.e., volume of training data). Our results show that while there are objective functions that, as expected, perform well on highly experienced neural networks, there are also select objective functions which perform better on less experienced neural networks. We also note that the time-efficiency of using a trained neural network (vs. corresponding transfer matrix theory) scales favourably by two orders of magnitude, as investigated later.

Generating Optical Spectra of Core-Shell Nanostructures for Training Datasets

We first compare the neural network assisted with an evolutionary optimization algorithm (Figure 1, red) to its corresponding transfer matrix framework (Figure 1, blue). Here we chose an evolutionary algorithm as

       ARTICLE

optimization method to ensure the parameter space is uniformly explored, similar to the procedure by Purifoy et al.[26]. The stochastic nature of the optimization process thus provides a method to quantify how often the target is obtained (reliability). Specifically, our goal is to find the optimized core diameter and shell thickness in an all-dielectric core-shell nanostructure that produces the closest spectral match to a desired target — here, e.g., a Gaussian spectrum centred at 600 nm. To generate our training datasets, we use transfer matrix series solutions outlined by Qiu et al., using Mie theory applied to multi-layer core-shell nanostructures[30].
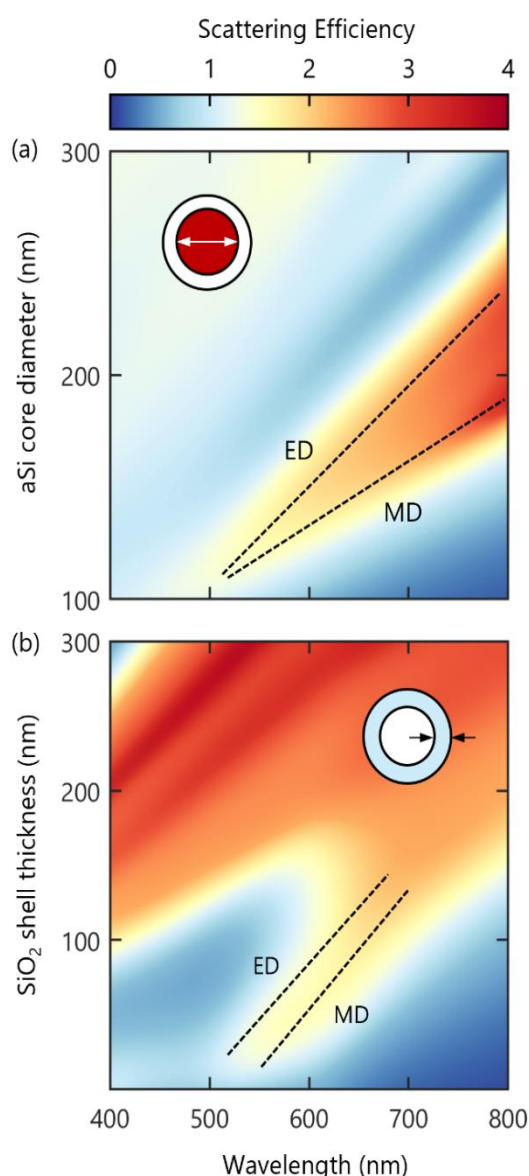

Figure 2 Mie-resonant modes of core-shell nanoparticle. (a) Pseudo-colour plots of scattering efficiency for core-shell nanostructures as the core diameter is varied while the shell thickness remains constant at 50 nm, and **(b)** scattering efficiency of core-shell nanostructures with core diameter of 137 nm and varying glass shell thickness. Electric dipole (ED) and magnetic dipole (MD) are highlighted in both figures.

Mie theory For a homogeneous spherical nanostructure, the scattering cross-sections for extinction and scattering σ_{sca} are obtained for a particular wave number k by expansion up to the order n with the coefficients α_n and β_n. These coefficients describe the constructive or destructive interference of the incident waves with the induced electric and magnetic fields within the nanostructure[42].

$$\sigma(k)_{sca} = \frac{2\pi}{k^2} \sum_{n=1}^{n} (2n+1)\{\sin^2 \alpha_n + \sin^2 \beta_n\}$$

Multi-layer core-shells require a more comprehensive approach as the added shell layers affect the scattering profile[42].

Transfer matrix Qiu et al. showed that scattering solutions for multi-layer core-shell nanostructures are obtained by a series expansion of transverse magnetic and electric modes. The scattering cross-sections solutions are a series of radial Bessel functions[30]:

$$\sigma(\lambda, r_{\sigma,l})_{sca} = \frac{\lambda^2}{8\pi} \sum_{\sigma} \sum_{l=1}^{\infty} (2l+1)\left|1 - r_{\sigma,l}\right|^2$$

where σ_{sca} is the scattering cross-section, $\sigma_{ext,abs}$ are found via $\sigma_{sca} = \sigma_{ext} - \sigma_{abs}$. The cross sections are a function of λ which is the optical wavelength, l represents the various transverse channels for the electric and magnetic modes, σ represents the transverse magnetic or transverse electric mode and finally, $r_{\sigma,l}$ is the reflectivity Ratio for the various layers.

Resonances and Materials Response The scattering cross-section is often re-casted as an efficiency, with the scattering efficiency defined as:

$$Q(\lambda, r)_{sca} = \frac{\sigma(\lambda)_{sca}}{\sigma_p(r)}$$

where σ_p is the physical cross-sectional area of the core-shell and r is the radius of the core-shell. The scattering efficiency depends critically on changes to core diameter and shell thickness, as described by Mie theory[42]. In addition, the lower-order scattering modes are highly sensitive to shell thickness. Using transfer matrix, we calculate these scattering modes, as shown in **Figure 2**. We use material properties from references [43-45] for our calculations. Hereafter, we simplify our notation and refer to $Q(\lambda_{400...800\,nm}, r_{aSi}, r_{gl})_{sca,apx}$ as $Q_{cal,\lambda}$. In other words, $Q_{cal,\lambda}$ is the calculated spectra containing 201 scattering efficiencies equally distributed between minimum ($\lambda_{min} = 400$ nm) and maximum ($\lambda_{max} = 800$ nm) wavelengths. We refer to the target of scattering efficiencies, i.e., $Q(\lambda_{400...800\,nm})_{tgt}$, simply as $Q_{tgt,\lambda}$.

Figure 2 highlight the first-order multipolar resonances, highlighting the electric (ED) and magnetic dipole (MD) lines in dashed lines. These resonances are characterised by plotting the respective spectra for the transverse magnetic (TM) and transverse electric (TE) modes of the scattered plane waves[2,46,47]. Figure 2a shows how the scattering efficiency changes with core diameter. The prominent electric dipole scattering modes (ED) is visible around 500 nm and redshifts as core size increases. Similarly, the magnetic dipole (MD) resonance near 500 nm gradually redshifts as the core size increases. The magnetic mode redshifts faster than the electric mode and is therefore more sensitive to structural changes. These trend in redshift are similar with increasing glass thickness. In other words, the spectral distance between the magnetic and electric resonances remains constant as the aSi diameter changes, when the glass shell thickness reaches 50 nm. Figure 2b highlights how the glass layer has a more complex effect on the scattering efficiency than the core size. For example, the higher order resonances appear as the shell thickness of glass ranges from 150 nm-170 nm, with resonance wavelength ranging between 400 nm and 600 nm. Another resonance grows when the glass thickness ranges from 170 nm to 300 nm — with resonances in the 600-800 nm wavelength range. It is also apparent that the glass layer tends to decrease the first-order resonances when the core reaches 170 nm. In all, these changes highlight the impact the extra shell has on generating more complex interactions, with enhanced higher-order resonances than are observed with a thinner shell.

Neural Network Training

The neural network learns the relation between the diameter of the core and thickness of the shell to the corresponding scattering cross-sections. In other words, the weights and biases of the linear equations constituting the neural network are tuned for predicting the spectra that would have been obtained by exact calculations, which require other computational methods[42,48-52].

Overview of neural network architecture The neural network we employ is trained to reproduce scattering efficiency, given only the structural parameters of the nanostructure as input. The neural network predicts the scattering efficiency of a core-shell nanostructure for varying core diameters and shell thicknesses (i.e., glass and air environment). The neural network follows the architecture used by Peurifoy et al.[53], and consists of four fully connected hidden layers, each with 250 nodes. The final output layer has 201 nodes and are the wavelength-dependent scattering efficiency of the core-
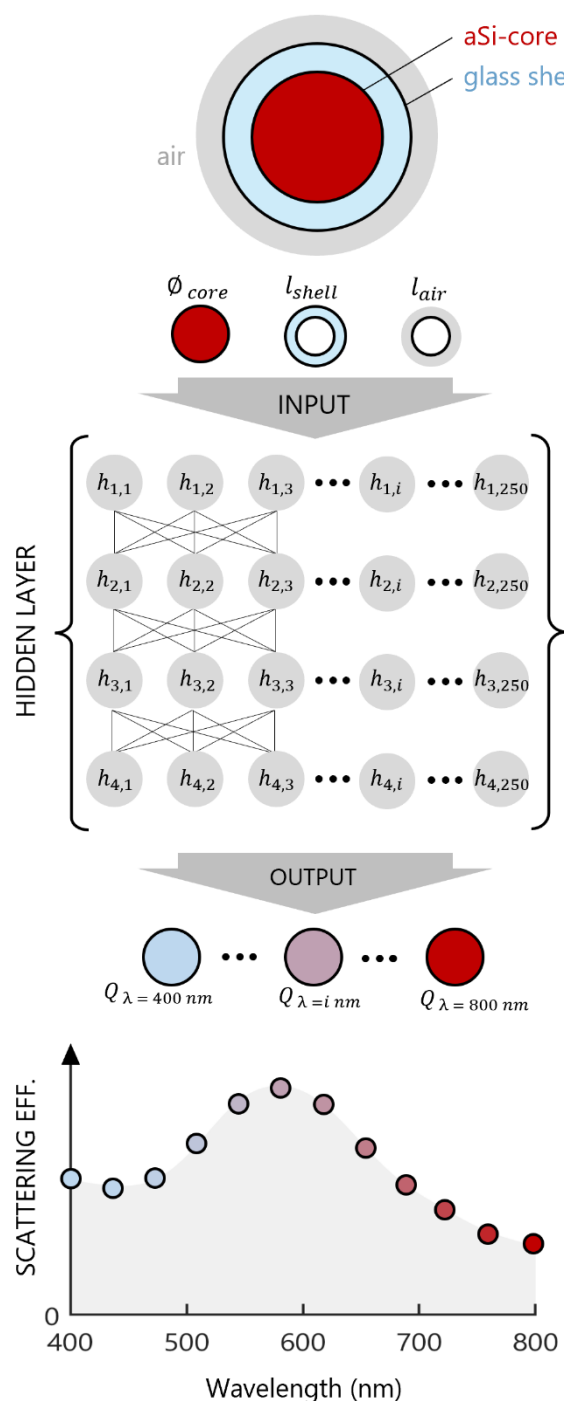

Figure 3. Visual representation of neural network, a) network architecture with three inputs, core size (ϕ_{core}), glass thickness (l_{shell}) and air shell thickness (l_{air}), the inputs feed into 4 layers of 250 nodes, the hidden layers then lead to 200 outputs representing the wavelength dependent spectral points (λ_i). Each spectral output corresponds to a specific scattering efficiency for a given wavelength.

shell nanostructure, i.e., its spectra (**Figure 3**). We follow the settings initially prescribed, with the ''patience' setting at 10 % and 'layers' set to 4. The MATLAB version is 2017b and the python version is 3.6.1. and TensorFlow is 1.2. An approximated training time for 20% of the data is an hour, where we make use of 16 Gb of memory. We
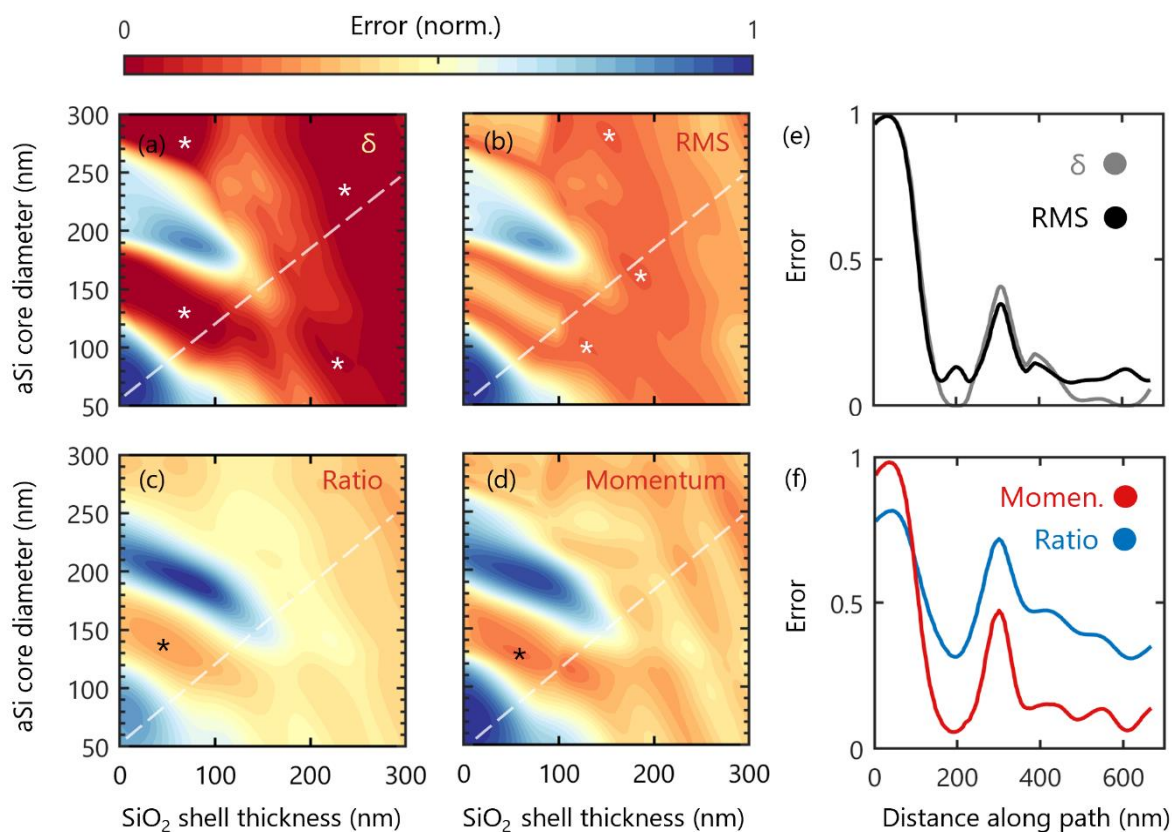
Figure 4 Surface plots depicting error calculated using the different objective functions **(a)** Delta, **(b)** RMS, **(c)** Ratio, and **(d)** Momentum as a function of core diameter and glass shell thickness. Plots are normalized to the maximum error with blue being regions of largest error and red regions of minimum error. Uniformity of color in the plot represent a lack of gradient in the error topography. Plots (e) and (f) are the error as a function of distance along the dashed line in plots a-d, starting from shell thickness at 0 nm and core diameter at 50 nm..

created several neural networks that have been trained with various percentage of total training data, i.e., various "experience" levels. We use a transfer matrix method to calculate the spectra of core-shell nanostructure, and we subsequently use the spectra, in conjunction with the core-shell label, for generating multiple training sets.

Each dataset has a fixed number of nanostructure-spectrum pairs, and contains 150,600 (100%), 120,716 (80%), 91,290 (60%), and 61,152 (40%) spectra. For each dataset that was created, we evenly divided the parameter space for both the core diameter and the shell layer thickness between minimum and maximum (boundary) values. This method ensures both a uniform density of data points over the parameter space and that data points are provided at every optimization bounds. Each dataset is hereafter split using the traditional 80%-20% training-validation scheme and trained up to 1000 epochs using the default early termination provided with the ScatterNet code. We note that each dataset is used specifically to train its own neural network. The different population of dataset allows us to determine how untrained data points presented to the neural network affect the quality of the optimization. In other words, we are able to gauge how

efficient the neural network is in predicting spectra it has not been trained on, allowing us to determine how reliable and accurate the optimization is.

Evolutionary Algorithm and its Role in the Inverse Design Process

Objective functions The objective function evaluates how closely a proposed solution, $Q_{cal,\lambda}$, resembles the target solution $Q_{tgt,\lambda}$. Here, we plot the 2D surface map of the objective function values for each core and shell values to generate the analytical ground truth. Our core-shell nanostructure only has the a-Si core and glass shell, which have a wavelength-dependent permittivity values. We explore the parameter space by mapping each pair of core-diameter and shell-thickness to each objective function (**Figure 4**). The error maps of the different objective function enable us to highlight their ground truth (where the error is minimum), denoted by stars. Plotted are the traditional weighted root mean squared (RMS), a weighted delta function (Delta), signal-to-noise ratio (Ratio), and a custom objective function (Momentum) figure 4 d.

Root Mean Square (RMS): The RMS objective function, O_{RMS}, is the L2 norm, i.e., the magnitude of the distance between the scattering efficiencies.

$$O_{RMS} = \sqrt{\sum_\lambda \frac{(Q_{cal,\lambda} - Q_{tgt,\lambda})^2}{N}}$$

Delta (δ)**:** The Delta objective function is the root mean square at only one wavelength, i.e., a single datapoint. The error O_{delta} is effectively the absolute distance, where $Q_{cal,\ \lambda=600}$ is the calculated scattering efficiency and $Q_{targ,\ \lambda=600}$ is the target scattering efficiency, at the $\lambda_0 = 600$ nm wavelength and N is unity.

$$O_{Delta} = |Q_{cal,\lambda_0} - Q_{tgt,\lambda_0}|$$

Ratio: The Ratio objective function evaluates the average distance between the calculated scattering efficiency, $Q_{cal,\lambda}$, and target scattering efficiency, $Q_{tgt,\lambda}$. The Ratio objective function, O_{Ratio}, seeks to find the averaged lowest variance with respect to the target[26,54-56].

$$O_{Ratio} = \frac{\Sigma_\lambda (Q_{cal,\lambda} - Q_{cal,\lambda} Q_{tgt,\lambda})}{\Sigma_\lambda Q_{cal,\lambda} Q_{tgt,\lambda}}$$

Momentum: The Momentum function $O_{Momentum}$ is our custom objective function. We designed the Momentum objective function to better quantify the differences between the calculated spectrum and target. We incorporate both the shape and amplitude differences of both the target and calculated spectra. Our approach is similar to that of Spline fitting approaches[57] and spectra-pooling by Baxter et al.[58].

Our custom objective function incorporates two measures for evaluating the difference in spectra. The first one, $M_{1,\lambda}$, is the square difference between $Q_{calc,\lambda}$ and $Q_{tgt,\lambda}$.

$$M_{1,\lambda} = (Q_{calc,\lambda} - Q_{tgt,\lambda})^2$$

The second measure, $M_{2,\lambda}$, is the square difference between the slope of $Q_{cal,\lambda}$, and the slope of $Q_{tgt,\lambda}$.

$$M_{2,\lambda} = \left(\frac{d}{d\lambda} Q_{cal,\lambda} - \frac{d}{d\lambda} Q_{tgt,\lambda}\right)^2$$

We then apply a method to weigh the measures. Rather than choosing arbitrary weighting factors, we partition the range of wavelength where each measure is evaluated. The method of averaging over the partitioned regions (split spectra) is similar to pooling methods in machine learning.

$$M_{1\ or\ 2\ Avg,j} = \sum_{\lambda_i}^{\lambda_f} M_{1\ or\ 2,\lambda}$$

The N features (peaks or troughs in the spectra) correspond to regions, $j = 1, 2, 3$, of interest in the target. λ_i is the first wavelength of the region which $M_{1\ avg,j}$ and $M_{2\ avg,j}$ are evaluated over, and λ_f is the last wavelength of that same region. Specifically, our target is a gaussian shape centred at 600 nm (mean, $\mu = 600$ nm; standard deviation, $\sigma = 20$ nm). Following extensive testing, we find that defining the central region width (i.e., $j = 2$) of $1/2\sigma$ provides the best optimization results, as discussed later. In other words, our first region $j = 1$ is $\lambda_{min = 400\ nm} \leq \lambda < \mu - 1/2\ \sigma$. The second region, $j = 2$ is ranges between $\mu - 1/2\ \sigma \leq \lambda \leq \mu - 1/2\ \sigma$. The third region $j = 3$ ranges between $\mu + 1/2\ \sigma < \lambda \leq \lambda_{max = 800\ nm}$.

The two measures of the pooled regions are combined by taking the product of these two measures in a given region (i.e., attention), which gives a weight w for the region j:

$$w_j = M_{1\ avg,j}\ M_{2\ avg,j}$$

The weights are combined through multiplication, as often carried out in Bernoulli trials and random-walk models.

$$W_{vol} = w_{i=1} w_{i=2} w_{i=3\dots N} = \prod_j^N w_j$$

We multiply the three different weights to obtain W_{vol}. However, since a particular weight of zero could affect the overall product, we also sum the total of each weight.

$$W_{sum} = w_{i=1} + w_{i=2} + w_{i=3\dots N} = \sum_j^N w_j$$

To keep the objective function as convex as possible, we take the squared sum of the sum and product terms. Thus, the final objective value that we are minimizing, $O_{Momentum}$ is given by:

$$O_{Momentum} = W_{vol}^2 + W_{sum}^2 + 2W_{vol}W_{sum}$$

Evolutionary Algorithm Setup

We test the accuracy of the four different objective functions by using an evolutionary algorithm. We also quantify how the starting populations change our results. Our optimization method is an evolutionary algorithm based on Matrix Laboratory and Simulink (MATLAB®). To avoid potential issues regarding local
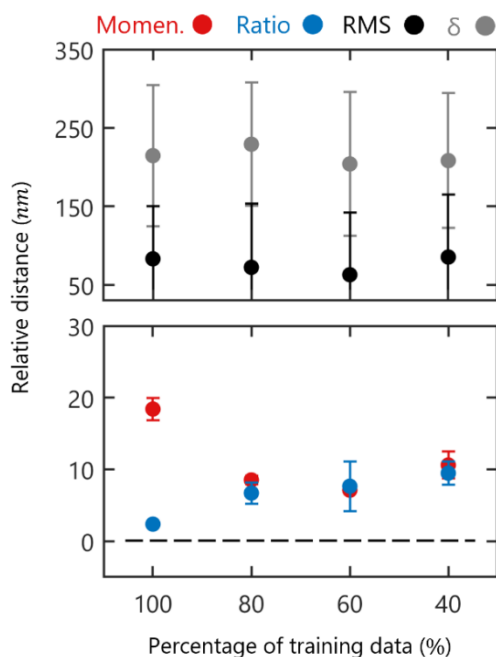
Figure 5: Plots comparing the relative Euclidean distance (from ground truth) for each objective function used, i.e., the Momentum (red circle), Ratio (blue circle), Root Mean Square, (black circle) and Delta (grey circle), as the amount of training data provided to the neural network is decreased. Error bars represent the deviation from the target solution. Note that the results are plotted for differing amount of 80% of entire data training data used on the network.

minima, we set a fix number of generations. Each optimization carried out the exact same number of iterations to ensure the optimization would not prematurely terminate due to a local minimum. For accuracy testing, the initial points (i.e., structural parameters) are varied randomly within the genetic algorithm starting population. After each optimization, the genetic algorithm provides the three structural parameters — that will subsequently generate the spectra either via ML or transfer matrix to best match the target. With the air layer ignored (since it does not contribute to the scattering), we can then evaluate each solution predicted by the genetic algorithm with respect

to the surface maps of Figure 4. For the full optimization to be accurate, we expect the algorithm to always yield a point closest to the white/black star in the different maps in figure 4.

Optimization Accuracy

To understand how the genetic algorithm navigates through the error surface to finally propose a solution for each of the four optimization frameworks (in terms of the aSi and glass layer), we use a metric that compares each optimization run with the ground truth (with variance plotted as error bars in **Figure 5**). This metric is defined as the L2 norm between our target (location of stars in figure 4), and how close each final solution provided by the genetic algorithm is. To measure how often we obtain the correct structural parameters with respect to the target, we run optimizations with a population of 200 individuals using different starting initial positions. These population optimization scheme is repeated a thousand times. We note that for each of these optimizations, the initial population is distributed differently in terms of diameter/shell-thickness pair. Finally, we then average the various obtained L2-norms, as shown in Figure 5.

Optimization Timing

Further, we explore the effects of population size within the genetic algorithm framework by timing how long the optimization takes to converge to its proposed solution. For this test, we kept the initial positions fixed this time. The time taken to complete an optimization is recorded. We repeat this optimization scheme five times. Next, we discount the minimum and maximum times; the remaining three times are then averaged over to obtain the quoted computational time (see **Figure 6**). Moreover, we measured this optimization time for three different population sizes, that is using 50, 100 and 200 individuals. The algorithm uses population conservation.

We mitigate any discrepancies in the timing that may arise due to the use of shared computational resources, in our case, the Cheaha compute cluster at the University of Alabama Birmingham[59]. To mitigate this effect, we ran the optimizations at the highest priority request on Cheaha (the UAB High performance Computing resource), along with having a random loop within the code to issue a minor delay before the start of each timing experiment to efficiently sample the cluster computational load over time. The computational resources allocated for use in this work are as follows: Intel Xeon Gold 6248R 3.00GHz. As benchmark, we find the amount of time to generate 150,600 spectra using the transfer matrix is approximately one hour. To mitigate discrepancy in traditional computational workloads where programs will load and unload data into memory inconsistently, we dropped the shortest and longest times of each test as standard practice.

Results and Discussion

Figure 4, a-d demonstrates clearly how the different objective functions influence where the optimum core-shell parameters of the nanostructure is located (i.e., global minimum), as marked with stars. The objective functions are normalized for visual inspection (the optimization however uses the actual, unnormalized error values. The various areas of minimum error in the surface affect how a gradient-based optimization behave. If the objective function is globally convex (i.e., having an absolute minimum), then we expect the evolutionary algorithm to perform as well or better than a traditional, non-stochastic gradient descent algorithms. The surface plots in Figure 4 a-d also allow us to explore how convex, flat or concave each objective function is. The ideal objective function would have a single minimum at the target location, and a negative gradient at every other point on the surface.

Examining the loss in one dimension — i.e., taking a representative trace (dashed lines in Figure 4) — illustrates the mechanisms behind the contrasting results of the objective functions. The Delta and RMS traces highlight sections where the error values are similar. This uniformity in error topography prevents the genetic algorithm from looking beyond those local minima. In contrast, for the more diverse topography seen in the Momentum and Ratio objective functions traces allow the genetic algorithm to converge to the global minimum. In other words, the large error variations help ensure mutations within the population of the algorithm to use more points to sample.

Figure 5 compares how changes to the neural network accuracy affects the optimization accuracy. The results clearly indicate the vast improvement in accuracy for

the Momentum and Ratio objective functions over the Delta and RMS objective functions. The Momentum and Ratio Objective functions both have high accuracy. Further we see the Momentum and Ratio have low variations in the accuracy (error bars). The Momentum and Ratio objective functions are sensitive to changes in the neural network accuracy. The Delta and RMS objective functions have poor accuracy and large deviations (error bars). We also see the Delta and RMS objective functions are not influenced by the changes in the neural network accuracy. One probable explanation for how the neural network accuracy does not affect the results of the RMS and Delta functions, may be the inaccuracy of the RMS and Delta may be so large it obscures the additional variance caused by the neural network. When we compare the Momentum to the Ratio objective functions, we see that as the neural network experience decreases (i.e., with less training data used), the Momentum objective function shows a decrease in standard deviation to the target spectra, while also
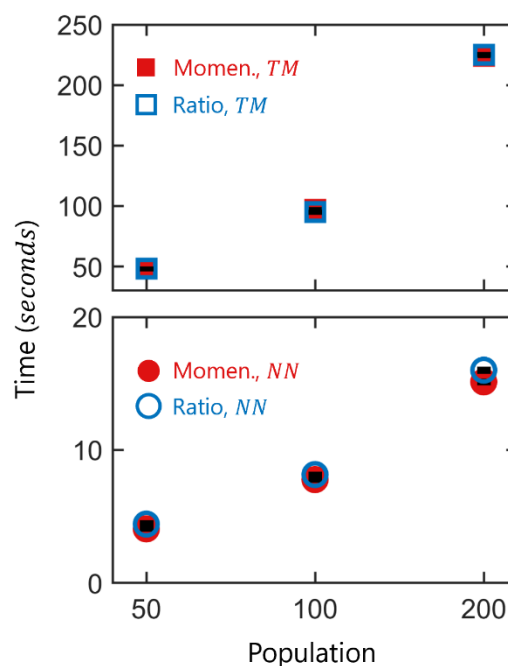

Figure 6: Plot demonstrating the average computational time for varying evolutionary algorithm populations, the x axis is not to scale. The colour red indicates the momentum objective function and blue represents the ratio objective function. The circle represents the network calculations, while the squares represent the transfer matrix calculations.

increasing in accuracy until a minimum accuracy is reached at 60% of the training data. Conversely, as a decrease in neural network training data the Ratio and objective function increases in deviation from the target signal. We also see the Ratio objective function linearly decreases in accuracy as the training data is lowered.

We see that the Momentum loss becomes similar to the Ratio method as neural network is trained on less amount of data. We believe that this convergence is caused by smoothing of the local minima as the less experienced neural network would more likely underfit. Therefore, the differences between the Momentum and Ratio methods become less apparent. With the neural network trained on less data, the Ratio objective function have larger outliers from the target solution. Yet the Momentum objective function appears to have the smallest variation from the target solution. However, as the neural network becomes trained on lesser amount of data (Figure 5, 60%), the depth of the local minima modifies the optimization process and decrease the accuracy of the various objective functions.

We now examine the efficiency (time for optimization scheme to reach a solution) of the two best objective functions as we change the population within our framework (Figure 6). Figure 6 demonstrates how a change in the population of the evolutionary algorithm affects the time required for a given optimization. The squares indicate that transfer matrix theory (in contrast to neural network) was used to evaluate the objective function. The circles represent optimizations where the neural network is used in calculation of the objective function. The colours indicate which objective function was used during the optimization, red is the Momentum objective function and blue is the Ratio objective function.

The results indicate that the increased population from 50 to 100 individuals has a nearly linear increase in time for both the neural network optimization and transfer matrix optimization. The large difference in time shows the computational performance of the neural network optimization is much more efficient than the transfer matrix optimization. Yet the rates of increase for the time between 50 to 100 individuals shows suggests that each optimization scales similarly with the increased population. Once the population increases beyond 100 individuals the computational time increases at a faster rate for the transfer matrix optimization when compared to the neural network optimization. This increase in efficiency shows the neural network optimization has an advantage in scaling efficiency with population. This efficiency gain can be attributed to several factors like bandwidth from the material data required in the transfer matrix, memory capacity and computational complexity. Within the same framework, i.e., neural network or transfer matrix, there is no appreciable computational times for these two objective functions. The transferability of our model is limited by the scatter net framework. The current framework of scatternet only allows for the input of each core shell size, thus any interchange of material

properties would bias the model or average the weights over different materials as there would be no input to distinguish between materials.

Conclusion

We demonstrate how the accuracy of an optimization process utilizing a neural network assisted with an evolutionary algorithm can be improved for the inverse design process. The results show evidence that the combination of a neural network's experience in conjunction with a given objective function act as a means of regularization. If plenty of training data is available for the training of the neural network, one would choose the use a highly trained neural network and employ a Ratio objective function. Often, the lack of available data limits the networks experience. In such a case one should choose to employ the Momentum objective function.

Our work highlights that one must factor and apply domain knowledge for the optimization of nanophotonic structures, as demonstrated by our prototypical core-shell nanostructure. We provide evidence on how the choice of an objective function yields different global optimal designs even when the physics is unchanged. The discrepancy in global optima could be attributed to some objective functions highlighting different features of the core-shell nanostructure asymmetric spectra. In the future, it will be interesting to explore how the use of other tailored optimizations functions affect other types of optimization process such as stochastic gradient-based ones.

Author Contributions

DJH and KA designed the project. PVB and DJH conducted the machine learning portion. All authors contributed in writing the manuscript.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

pertaining to their choice of objective function and computational methods.

Notes and references

1   Kildishev Alexander, V., Boltasseva, A. & Shalaev Vladimir, M. Planar Photonics with Metasurfaces. *Science* **339**, 1232009, (2013).

2   Liu, S. *et al.* Huygens' Metasurfaces Enabled by Magnetic Dipole Resonance Tuning in Split Dielectric Nanoresonators. *Nano Letters* **17**, 4297-4303, (2017).

3   Loo, C., Lowery, A., Halas, N., West, J. & Drezek, R. Immunotargeted Nanoshells for Integrated Cancer Imaging and Therapy. *Nano Lett.* **5**, 709-711, (2005).

4   Vollmer, F. & Arnold, S. Whispering-gallery-mode biosensing: label-free detection down to single molecules. *Nature Methods* **5**, 591-596, (2008).

5   Zheludev, N. I. & Kivshar, Y. S. From metamaterials to metadevices. *Nature Materials* **11**, 917-924, (2012).

6   Sinev, I. *et al.* Polarization control over electric and magnetic dipole resonances of dielectric nanoparticles on metallic films. *Laser & Photonics Reviews* **10**, 799-806, (2016).

7   Temple, T. L. & Bagnall, D. M. Optical properties of gold and aluminium nanoparticles for silicon solar cell applications. *Journal of Applied Physics* **109**, 084343, (2011).

8   Hsu, C. W. *et al.* Transparent displays enabled by resonant nanoparticle scattering. *Nat Commun* **5**, 3152, (2014).

9   Staude, I. *et al.* Tailoring Directional Scattering through Magnetic and Electric Resonances in Subwavelength Silicon Nanodisks. *ACS Nano* **7**, 7824-7832, (2013).

10   Shcherbakov, M. R. *et al.* Ultrafast All-Optical Switching with Magnetic Resonances in Nonlinear Dielectric Nanostructures. *Nano Letters* **15**, 6985-6990, (2015).

11   Kruk, S. & Kivshar, Y. Functional Meta-Optics and Nanophotonics Governed by Mie Resonances. *ACS Photonics* **4**, 2638-2649, (2017).

12   Zhang, C. *et al.* Lighting up silicon nanoparticles with Mie resonances. *Nature Communications* **9**, 2964, (2018).

13   Babicheva, V. E. & Moloney, J. V. Lattice effect influence on the electric and magnetic dipole resonance overlap in a disk array. *Nanophotonics* **7**, 1663-1668, (2018).

14   Ilchenko, V. S. & Matsko, A. B. Optical resonators with whispering-gallery modes-part II: applications. *IEEE Journal of Selected Topics in Quantum Electronics* **12**, 15-32, (2006).

15   Huckabay, H. A. & Dunn, R. C. Whispering gallery mode imaging for the multiplexed detection of biomarkers. *Sensors and Actuators B: Chemical* **160**, 1262-1267, (2011).

16   Kuznetsov, A. I., Miroshnichenko, A. E., Brongersma, M. L., Kivshar, Y. S. & Luk'yanchuk, B. Optically resonant dielectric nanostructures. *Science* **354**, (2016).

17   Yang, Y., Kravchenko, I. I., Briggs, D. P. & Valentine, J. All-dielectric metasurface analogue of electromagnetically induced transparency. *Nature Communications* **5**, 5753, (2014).

18   Sui, C. *et al.* High Q-Factor Resonance in a Symmetric Array of All-Dielectric Bars. *Applied Sciences* **8**, (2018).

19   Englund, D., Fushman, I. & Vuckovic, J. General recipe for designing photonic crystal cavities. *Optics Express* **13**, 5961-5975, (2005).

20   Akahane, Y., Asano, T., Song, B.-S. & Noda, S. Fine-tuned high-Q photonic-crystal nanocavity. *Optics Express* **13**, 1202-1214, (2005).

21   Jiao, Y., Fan, S. & Miller, D. Demonstration of systematic photonic crystal device design and optimization by low-rank adjustments: An extremely compact mode separator. *Optics letters* **30**, 141-143, (2005).

22   Valleti, S. M. P. *et al.* Bayesian Learning of Adatom Interactions from Atomically Resolved Imaging Data. *ACS Nano* **15**, 9649-9657, (2021).

23   Zandehshahvar, M. *et al.* Manifold learning for knowledge discovery and intelligent inverse design of photonic nanostructures: breaking the geometric complexity. *ACS Photonics* **9**, 714-721, (2022).

24   Kumar, M. S., Menabde, S. G., Yu, S. & Park, N. Directional emission from photonic crystal waveguide terminations using particle swarm optimization. *Journal of The Optical Society of America B-optical Physics* **27**, 343-349, (2010).

25   Goh, J., Fushman, I., Englund, D. & Vučković, J. Genetic optimization of photonic bandgap structures. *Optics Express* **15**, 8218-8230, (2007).

26   Peurifoy, J. *et al.* Nanophotonic particle simulation and inverse design using artificial neural networks. *Sci Adv* **4**, eaar4206, (2018).

27   Liu, C., Maier, S. A. & Li, G. Genetic-Algorithm-Aided Meta-Atom Multiplication for Improved Absorption and Coloration in Nanophotonics. *ACS Photonics* **7**, 1716-1722, (2020).

28   Geremia, J. M., Williams, J. & Mabuchi, H. Inverse-problem approach to designing photonic crystals for cavity QED experiments. *Physical Review E* **66**, 066606, (2002).

29   Sakurai, A. *et al.* Ultranarrow-Band Wavelength-Selective Thermal Emission with Aperiodic Multilayered Metamaterials Designed by Bayesian Optimization. *ACS Central Science* **5**, 319-326, (2019).

30   Qiu, W., DeLacy, B. G., Johnson, S. G., Joannopoulos, J. D. &´, M. S. i. Optimization of broadband optical response of multilayer nanospheres. *Opt. Express* **20**, (2012).

31   Jiang, J. & Fan, J. A. Global Optimization of Dielectric Metasurfaces Using a Physics-Driven Neural Network. *Nano Letters* **19**, 5366-5372, (2019).

32   Yao, K., Unni, R. & Zheng, Y. Intelligent nanophotonics: merging photonics and artificial intelligence at the nanoscale. *Nanophotonics* **8**, 339-366, (2019).

33 Zeng, Z., Venuthurumilli, P. K. & Xu, X. Inverse Design of Plasmonic Structures with FDTD. *ACS Photonics* **8**, 1489-1496, (2021).

34 Vai, M. M., Shuichi, W., Bin, L. & Prasad, S. Reverse modeling of microwave circuits with bidirectional neural network models. *IEEE Transactions on Microwave Theory and Techniques* **46**, 1492-1494, (1998).

35 Mahendran, A. & Vedaldi, A. in *Proceedings of the IEEE conference on computer vision and pattern recognition.* 5188-5196.

36 Bao-Liang, L., Kita, H. & Nishikawa, Y. Inverting feedforward neural networks using linear and nonlinear programming. *IEEE Transactions on Neural Networks* **10**, 1271-1290, (1999).

37 Baird, L., Smalenberger, D. & Ingkiriwang, S. in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* 966-971 vol. 962.

38 Hinton, G., Rumelhart, D. & Williams, R. Learning representations by back-propagating errors. *Nature* **323**, (1986).

39 He, K., Zhang, X., Ren, S. & Sun, J. in *Proceedings of the IEEE international conference on computer vision.* 1026-1034.

40 Glorot, X. & Bengio, Y. in *Proceedings of the thirteenth international conference on artificial intelligence and statistics.* 249-256 (JMLR Workshop and Conference Proceedings).

41 Bishop, C. M. *Neural Networks For Pattern Recognition*. (Oxford University Press, 1995).

42 Bohren, C. F. & Huffman, D. R. *Absorption and Scattering of Light by Small Particles*. (John Wiley & Sons, 2008).

43 Palik, E. *Handbook of Optical Constant of Solids*. (1985).

44 Schott. Zemax 2017-01-20b. **Zemax**.

45 Pierce, D. T. & Spicer, W. E. Electronic Structure of Amorphous Si from Photoemission and Optical Studies. *Physical Review B* **5**, 3017-3029, (1972).

46 Feng, T., Xu, Y., Zhang, W. & Miroshnichenko, A. E. Ideal Magnetic Dipole Scattering. *Physical Review Letters* **118**, 173901, (2017).

47 Evlyukhin, A. B., Reinhardt, C. & Chichkov, B. N. Multipole light scattering by nonspherical nanoparticles in the discrete dipole approximation. *Physical Review B* **84**, 235429, (2011).

48 Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems* **2**, 303-314, (1989).

49 Hornik, K. Some new results on neural network approximation. *Neural Networks* **6**, 1069-1072, (1993).

50 Zainuddin, Z. & Pauline, O. Function approximation using artificial neural networks. *WSEAS Transactions on Mathematics* **7**, 333-338, (2008).

51 Elfwing, S., Uchibe, E. & Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks* **107**, 3-11, (2018).

52 Guang-Bin, H., Saratchandran, P. & Sundararajan, N. A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation. *IEEE Transactions on Neural Networks* **16**, 57-67, (2005).

53 <2018 ScienceAdv - Soljacic - Original Peurifoy article.pdf>.

54 Blitz, J. P. & Klarup, D. G. Signal-to-Noise Ratio, Signal Processing, and Spectral Information in the Instrumental Analysis Laboratory. *Journal of Chemical Education* **79**, 1358, (2002).

55 Zhang, Z. & McElvain, J. S. Optimizing Spectroscopic Signal-to-Noise Ratio in Analysis of Data Collected by a Chromatographic/Spectroscopic System. *Analytical Chemistry* **71**, 39-45, (1999).

56 So, S., Mun, J. & Rho, J. Simultaneous Inverse Design of Materials and Structures via Deep Learning: Demonstration of Dipole Resonance Engineering Using Core–Shell Nanoparticles. *ACS Applied Materials & Interfaces* **11**, 24264-24268, (2019).

57 Khoram, E., Qian, X., Yuan, M. & Yu, Z. Controlling the minimal feature sizes in adjoint optimization of nanophotonic devices using b-spline surfaces. *Optics Express* **28**, 7060-7069, (2020).

58 Baxter, J. *et al.* Plasmonic colours predicted by deep learning. *Scientific Reports* **9**, 8074, (2019).

59 *UAB Research Computing Docs*, <https://uabrc.github.io/> (2020).