Design and optimization of large-area meta-optics

Maksym Zhelyeznyakov

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2023

Reading Committee:
Arka Majumdar, Chair
Nathan Kutz
Manjeri Anantram

Program Authorized to Offer Degree:
Electrical and Computer Engineering

University of Washington

Abstract

Design and optimization of large-area meta-optics

Maksym Zhelyeznyakov

Chair of the Supervisory Committee:
Arka Majumdar

In recent years, quasi-periodic arrays of dielectric scatterers, also known as metasurfaces have been shown to manipulate electromagnetic scattering with sub-wavelength spatial resolution. These devices have proven to be incredibly useful for reducing the form factor of conventional optical devices, as well as creating novel optical devices that are not possible to realize with conventional refractive optics. The design of these devices is however non-trivial. Conventional forward design methods are useful for generating only a small subset of possible designs. More complicated designs as well as optimization of existing forward designs requires solving the full-wave Maxwell's equations many times in order to converge to an optimal device design. This thesis outlines three main advancements in computational electromagnetics for the design of metasurfaces. Firstly, we extend the Generalized Multi-sphere Mie Scattering (GMMT) method to ellipsoidal scatterers. This method solves Maxwell's equations and computes gradients with respect to geometric design parameters of metasurfaces composed of ellipsoidal, dielectric scatterers analytically. We use this method to fully inverse design a metasurface lens, a meta-grating that focuses light at different focal planes for different polarizations, and improve the efficiency of a forward designed lens from 25.59% to 32.00%. The second method is a data-driven method that predicts electromagnetic fields from wavelength-scale cylindrical pillars, obtaining a low-dimensional representation of the data via the singular value decomposition. This method fits a differentiable neural-network model of the input geometries and configurations of the metasurface scatterers to the low-dimensional representations of the output field. This framework was used to design a device that produces an annular focal spot for the $\lambda = 400nm$ wavelength, and a focal spot for the $\lambda = 633nm$ wavelength. Finally, we present a data-free machine learning framework for predicting electromagnetic field responses from metasurfaces by using physics informed neural networks. This method predicts full-fields from 1D meta-gratings by a neural network that is trained by minimizing the residual norm of the Maxwell operator. We use this technique to design a set of large-area ($1mm$) meta-lenses exhibiting higher efficiency than the ones designed under the local phase approximation. We validate our designs experimentally, and see a maximum intensity improvement of up to 53%.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Number Page

ACKNOWLEDGMENTS

A PhD was a great way to spend my 20s. I'm lucky to have met the remarkable people who made it a positive experience. First and foremost I would like to express my gratitude to my advisor Arka Majumdar for his mentorship and the freedom I was granted throughout my PhD. I have the utmost respect for him as as scientist, and consider him to be a great friend and mentor. I'd also like to give my thanks to Steven Bruton who provided me with guidance, support, and helped me with the mathematical challenges. I'd like to thank my predecessor Alan Zhan, who started the inverse design project in Arka's group. I would also like to express my appreciation for Junsuk Rho for hosting me for my semester abroad in South Korea.

Working with Arka's group, I was fortunate to have a great group of co-workers. I would especially like to thank James Whitehead and David Rosser for our weekly lunches. They made the COVID lockdowns much more bearable. I would like to thank Johannes Fröch for helping me with my experiments and Abhi Saxena for our helpful conversations. I'd like to acknowledge all of my friends and colleagues in Arka's group: Elyas Bayati, Yueyang Chen, Shane Colburn, Jiaju Zheng, Taylor Fryett, Andrew Wolfram, Albert Ryou, Shreyas Shah, Forest Miller, Anna Wirth-Singh, Hannah Rarick, Arnab Manna, Rui Chen, David Sharp, Luocheng Huang, Roger Fang, Christ Munley, Ammod Shanker, and Quentin Tanguy. I would also like to thank Jaebum Noh for being a good friend during my study semester in South Korea.

I would finally like to thank my friends, parents, and grandparents. I would not be where I am without their love and support.

DEDICATION

to my family and friends

Chapter 1

INTRODUCTION

Humans have been manipulating light since the dawn of civilization. Some of the earliest examples of this date back 3000 years before the *yang-sui* bronze age, burning mirrors of ancient China that were used to light cooking fires [79]. Since then, humanity has developed other optical devices with a wide range of functionalities. Since Galileo Galilei's pioneering work, telescopes have allowed astronomers to observe the very largest objects in the universe. Microscopes help biologists and life-scientists understand living things on the micrometer scale. Corrective lenses fix peoples poor eyesight. Driving mirrors enable safer vehicles. With the advent of smart-phones, cameras are now accessible to the average individual everywhere wherever they go.

The operation of these devices can be understood by the principle of refraction - or Snell's law. given by:

$$n_1 \sin\theta_1 = n_2 \sin\theta_2. \tag{1.1}$$

where n_1 and n_2 refractive indices media and θ_1 and θ_2 are the angle that a ray of light makes with the surface normal of the boundary between two media. For spherical lenses, it is possible to derive the lens-maker's equation [78] from eq. (1.1):

$$\frac{1}{F} = (n-1)\left(\frac{1}{R_1} - \frac{1}{R_2} + \frac{(n-1)d}{nR_1R_2}\right) \tag{1.2}$$

where n is the refractive index of the lens, R_1 and R_2 are the radius of curvatures for the lens surfaces, F is the focal length, and d is the lens thickness. If we consider only spherical, plano-convex lenses and take limit as $R_2 \to \infty$, we get the relationship:

$$F = (n-1)R \tag{1.3}$$

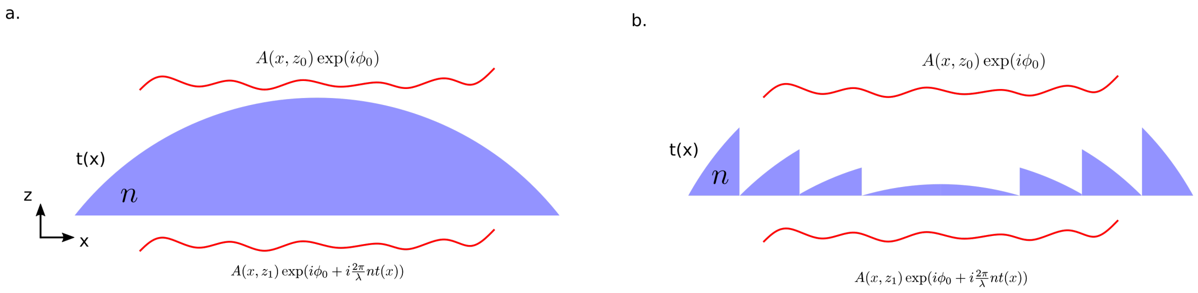


Figure 1.1: **a.** Refractive lens with a thickness function $t(x)$ made from a material with refractive index n. **b.** Fresnel lens. Due to the cyclic nature of phase, these devices achieve the same focal length, but the Fresnel lens is significantly less bulky.

Now let us say we want to miniaturize an optical system. Generally, micro-lenses are assumed to be spherical since they are manufactured exploiting the surface tension effect of a liquid material state resulting in a smooth surface [116]. Thus, in the following analysis we assume the lenses to be spherical. Given a lens aperture size D, the minimum possible focal length is given by $F = \frac{D}{2(n-1)}$, limited by the geometry of the sphere. We note that it is possible to reduce the focal length further by using parabolic lenses, but manufacturing them at the micro-scale is very challenging. Furthermore, refractive optical elements can generally perform only one function. Optical systems typically have several optical elements such as multiple lenses, mirrors, and prisms, making miniaturization challenging.

On the other hand, diffractive optics function in a fundamentally different way. Instead of treating light as a ray, wave-optics treats light as an electromagnetic (EM) wave with some amplitude A, phase ϕ, and wavelength λ. Consider a field propagating along a focal axis z. As this field interacts with an optical device, it picks up a phase delay:

$$\phi_d(x) = \frac{2\pi}{\lambda}nt(x) \tag{1.4}$$

where n is the refractive index of the device, and $t(x)$ is the thickness of the device at coordinate x. This way, by understanding the wave-like nature of light, we can describe optical devices as functions that imprint a phase onto a field. In fact, this understanding led to the creation of the Fresnel lens (show in Fig. 1.1) that significantly reduces the bulk of a large aperture, short focal length lens.

1.1 Metamaterials

Electromagnetic metamaterials are quasi-periodic arrays of sub-wavelength scale scatters (meta-atoms) [114]. Metamaterials are generally three-dimensional element, where the meta-atoms are arranged in periodic structure in all three-dimensions. Unfortunately, the difficulty in fabrication has precluded widespread of use of metamaterials. Their two-dimensional counterpart metasurfaces have drawn significant interest in the last few years. There are several main reasons for this wide-spread use of metasurfaces. Metasurfaces are extremely thin. This is appealing because as mentioned in the previous section, conventional optics are bulky. Metasurfaces are capable of controlling the amplitude, phase, and polarization of light in two dimensions by varying the geometric parameters of individual meta-atoms. They have also been shown to be compatible with the standard CMOS manufacturing processes[123, 104, 57]. Lastly, metasurface features are sub-wavelength, which suppresses all diffraction orders greater than the 0^{th} order. For all these reasons, metasurfaces make a compelling candidate for optical miniaturization. In recent years, metasurfaces have shown great promise in shrinking existing optical elements such as lenses [23, 119, 10], vortex beam generators [52, 119], and holograms [128]. They have also been used to create optical elements with new functionalities such as polarization multiplexed devices [8, 127], wavelength multiplexed devices [112, 124], and polarimeters [12]. Furthermore, metasurfaces are an extremely powerful platform for point-spread function engineering [27] as they allow for sub-wavelength scale control of the electromagnetic field response in two dimensions.

1.2 Designing a metamaterial

We can group metamaterial design into two broad categories - forward and inverse design. Forward design idealizes metasurfaces as ultra-thin and passive phase-masks. Although is possible to modulate both amplitude and phase responses of metamaterials, for most of the devices, we desire to have near-unity transmision response from meta-atoms, thus only modulating the phase a device. This means that for forward design, a phase function that encapsulates the performance of a device is required. Once the phase function is chosen, a material platform must be chosen to fabricate the device. The platform is generally selected based on the desired wavelength of operation.

After selecting the phase function to be implemented and the material platform, we must build a library of meta-atoms with desired phase and amplitude responses which requires the using the local phase [44] or locally periodic [80] approximation (LPA). Both terms are used interchangeably in the literature, and we will let the reader decide which one they prefer. Fig.1.2 shows the simulation strategy behind the LPA. We simulate the full electromagnetic field from a meta-atom given some current source with normalized amplitude. The boundary conditions of the simulation are periodic. Then a complex field E_{measured} is measured some distance away from the scatterer. The complex transmission coefficients t are then given by

$$t = \text{mean}(E_{\text{measured}}) \tag{1.5}$$

and the amplitude A and phase ϕ responses are given by

$$A = |t| \tag{1.6}$$

and

$$\phi = \angle t. \tag{1.7}$$

For forward design, we generally create a lookup table of scatterer geometries that span a $0 - 2\pi$ phase shift, with near-unity amplitude. Fig.1.2 shows an example of a unit cell with a rectangular meta-atom. In this case the modifiable parameters are H, the height of the meta-atom, P the periodicity of the unit cell and D, the side-length of the square meta-atom. There

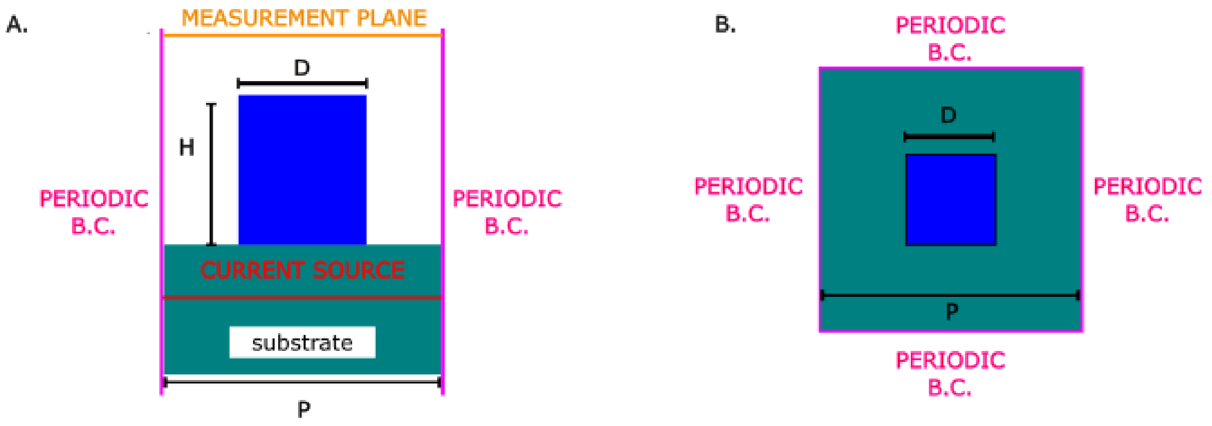


Figure 1.2: **a.** side view of rectangular scatterer being simulated under periodic boundary conditions. **b.** top view. H is meta-atom height, D is the square side length (usually the parameter modified during the design process), and P is the meta-device lattice constant.

are many different types of meta-atom geometries to chose from depending on the application. When fabricating meta-surfaces, H and P generally remain fixed, and the planar geometry gets modified to achieve a desired phase shift curve. Fig. 1.3 shows two example amplitude/ phase curves achieved by fixing the H and P of a metasurface, and modifying the diameters of cylindrical pillars. After the scatterer library is computed, the chosen phase function is discretized on a grid with resolution corresponding to the metamaterial periodicity, and meta-atom geometries are matched to their corresponding phase on the grid.

The other way of designing a meta-material is through inverse design. The inverse design method offers a pathway for designing meta-devices by specifying a figure of merit (FOM), written in terms of the design parameters, which characterizes the device's performance. The FOM is then either minimized or maximized via some optimization algorithm such as particle swarm, genetic algorithms, or gradient based methods. Genetic algorithms and particle swarm methods have successfully been used in optimizing photonic systems [3, 99, 34]. These methods however converge very slowly, and are typically too expensive to use for large-area metamaterial systems. Gradient based inverse design on the other hand uses the

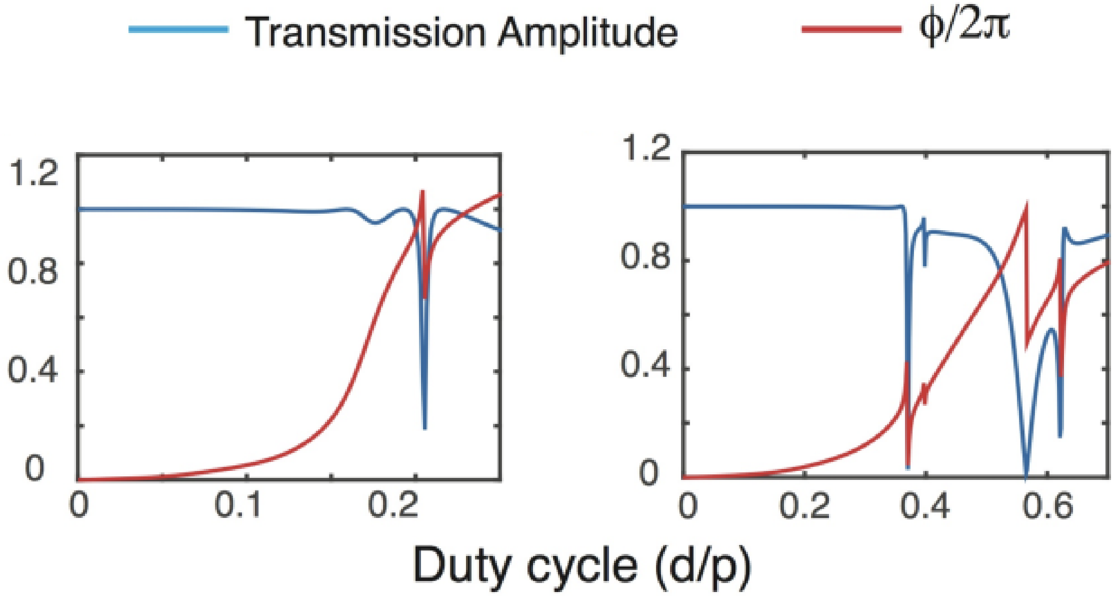


Figure 1.3: Example amplitude and phase curves for circular pillars. Duty cycle is the ratio between the pillar diameter and periodicity. Both sweeps are done at $\lambda = 0.633\mu m$. Figure credit Alan Zhan et. al.[118].

physics of the system, and converges much quicker. Fig. 1.4 shows a basic inverse design scheme. We start with an initial condition - an a priori arrangement of meta-atoms. This can be just a random set of geometries, a uniform set of geometries, or a forward designed device to be optimized. A forward simulation is then computed by some electromagnetic solver. The gradient of the FOM is computed with respect to the scatterer geometries, and the configuration of the scatters is updated. This process is repeated iteratively until the maximum or minimum value of the FOM is reached.

Inverse design requires running the forward simulation many times, and thus its speed is ultimately limited by the computational efficiency of the forward simulation method. In most existing inverse design works, Maxwell's equations are solved on a meshed grid, and the refractive indices of the device are allowed to change at each point of the grid space [74, 83, 82, 80, 14, 61]. These methods are considered to be accurate, but expensive, and

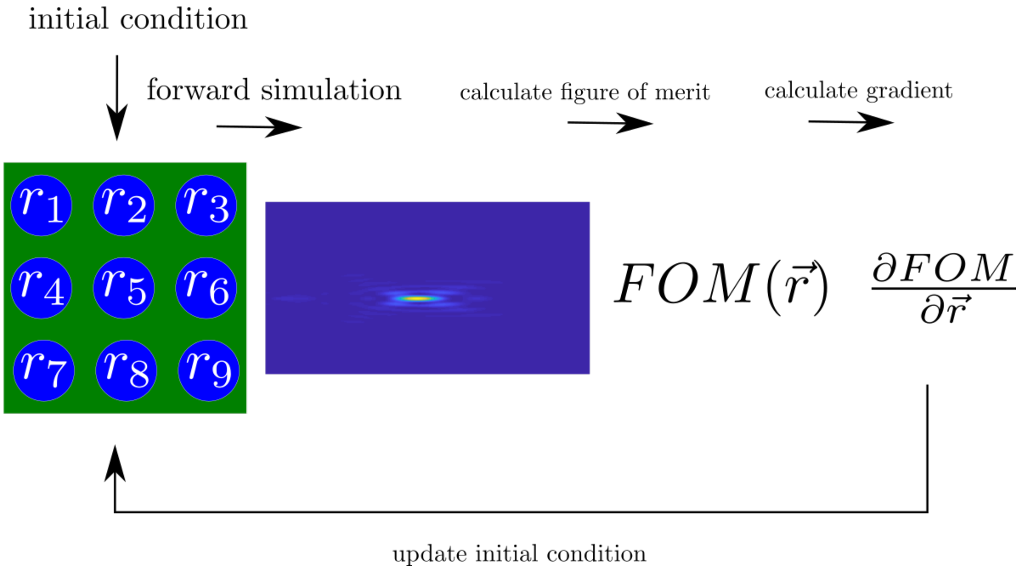


Figure 1.4: Metasurface inverse design. You start with some sort of initial condition on the distribution of the scatterer geometries, perform a forward simulation to get the electromagnetic field response, calculate the FOM from the electric fields, take the gradient with respect to the design parameters, and iteratively update the initial condition.

difficult to generalize to large area metasurfaces.

1.3 Thesis Outline

This thesis primarily focuses of addressing the computational bottlenecks of gradient based inverse design. Specifically, we will focus on different forward simulation methods, and access their performance by optimizing devices generated by forward design, and create new designs that are not possible to generate with forward design. Chapter 2 primarily focuses of the analytical Mie Scattering method. Chapters 3 and 4 focus of deep learning methods.

- **Chapter 2**: This chapter focuses on extending the Generalized Multi-sphere Mie Theory (GMMT) to ellipsoidal scatterers. This work is an extension of the work that was done by my predecessor - Alan Zhan, who used GMMT to design metamaterials using dielectric spherical scatterers [120, 121]. By breaking the symmetry of the sphere, we achieve higher control over the polarization of incident light. To compute the gradient of our FOMs with respect to the meta-atom design parameters we used the adjoint method. We inverse designed a polarization switched metalens. Furthermore, we used this method to optimize the efficiency of a forward designed metalens.

- **Chapter 3**: This chapter focuses on using a data driven approach to model electromagnetic scattering from meta-atoms that doesn't use the LPA. We present a forward simulation framework that is 10^4 times faster and 15 times more memory efficient than mesh-based solvers, and isn't constrained to specific scatterer geometries. We explored field patterns generated by plane wave scattering from dielectric, cylindrical pillars. We used the singular value decomposition to obtain low-dimensional representation of the scattering data, and generated a linear and neural network model fitting the scatterer geometries to the low dimensional representation of our data. To validate this model, we inverse designed two optical elements: a wavelength multiplexed device that focuses light to a point for $\lambda = 400nm$ and produces an annular beam for $\lambda = 633nm$ and an extended depth of focus lens.

- **Chapter 4**: This chapter focuses on modeling electromagnetic scattering by 1 dimensional meta-structures by the use of physics informed neural networks [85]. Unlike most other deep-learning methods, this method does not rely on generating any scattering data. Instead, we trained the neural network by minimizing the l_1 norm residual of the Maxwell partial differential equation operator

  $$\nabla^2 \cdot \quad + \omega^2 \epsilon \cdot \quad + i\omega J$$

  in order to fit a neural network model to a set of parameterized ϵ distributions. We demonstrate the accuracy of our model by designing a 1mm aperture cylindrical meta-lens exhibiting higher performance than a lens that was designed via the use of the LPA. We also experimentally validated the maximum intensity improvement of the inverse designed meta-lens.

Chapter 2

ELLIPSOIDAL T-MATRIX SCATTERING

The contents of this chapter are adapted from [127], *Maksym V. Zhelyeznyakov, Alan Zhan, and Arka Majumdar. Design and optimization of ellipsoid scatterer-based metasurfaces via the inverse t-matrix method. OSA Continuum, 3(1):89–103, Jan 2020*, with permission from the authors.

2.1 Introduction

The design of optical elements made of quasi-periodic arrays of sub-wavelength scatterers, also known as metasurfaces, is a promising area of research. The miniatuarization of existing optical elements such as lenses [115, 89, 23, 118, 10], freeform optics [117], and retroreflectors [8] has already been shown using metasurfaces. Furthermore, multi-functional optical elements[9, 51, 109, 110, 40, 108, 107] and new point spread function engineering methods [27, 28] have been demonstrated using metasurfaces. Until recently, however, these metasurfaces have generally been designed intuitively, termed here as forward design. Libraries of complex transfer coefficients of individual scatterers are pre-computed, and arranged in a periodic lattice to approximate a desired phase profile. The properties of these scatterers are computed with periodic boundary conditions and the metasurfaces are designed under the "local phase approximation": the scattering in any small region is taken to be the same as the scattering from a periodic surface [115, 89, 45]. This approximation neglects inter-scatterer coupling which is significant for metasurfaces composed of scatterers with rapidly varying geometries or with low refractive index [15]. Moreover, it is not always possible to know the phase-profile a priori, and in these cases forward design methods cannot be used.

Inverse design methods use a figure of merit (FOM) written in terms of adjustable geometric scatterer parameters to iteratively optimize the scatterers of a metasurface to implement a desired functionality. The process starts with an arbitrary initial scatterer configuration. Then the electric field scattered off the device, the FOM, and the gradient of the FOM with respect to the scatterer design parameters are computed. The scatterer geometries are then iteratively updated in the direction that optimizes the FOM. Thus, inverse design methods offer a clear path to create optical elements with unintuitive phase functions. Different optimization methods such as particle swarm optimization [100], genetic algorithm [4, 35, 31], and gradient based methods [120, 41, 83, 86, 82, 49, 80, 61, 36] have already been applied to design both integrated photonic elements and free space metasurface optics. One specific direction is to exploit Mie scattering of spherical scatterers to perform the inverse design [120, 121]. This approach allows large-area metasurface design without relying on the local phase approximation, and thus accurately models the inter-scatterer coupling. Currently, this approach is restricted to spherical scatterers, for which the radii are the only free parameters available. We did not find a radius range over which these spherical scatterers smoothly span a $0 - 2\pi$ phase shift without suffering considerable optical losses, a common requirement when designing metaphotonic structures.

In this chapter, we will outline an inverse-design and optimization method for ($\sim 40\lambda$ in diameter) metasurfaces using transition matrix scattering theory, an extension of Generalized Multi-sphere Mie Theory (GMMT). Specifically, we extended a previously reported inverse design method [120] to include ellipsoidal scatterers. We first show the feasibility of using this method for inverse-designing single wavelength metasurfaces lenses. We then demonstrate the effectiveness of this method for designing non-intuitive devices without a known phase function by optimizing a polarization multiplexed lens that switches the location of the focal spots based on the incident light polarization. Finally we demonstrate the efficacy of inverse design techniques for optimization, by improving the efficiency of a high numerical aperture lens starting with a forward designed metalens as the initial condition.

2.2 The Adjoint Method

The simplest way of calculating the gradient of a function with respect to N number of modifiable parameters is to independently vary a single parameter and monitor the sensitivity of the function with respect to the change. This however becomes very costly as N increases. The adjoint method provides a more efficient way of calculating the gradient. Let

$$\mathbf{A(P)}\ \mathbf{x(P)} = \mathbf{b(P)} \tag{2.1}$$

be a linear system we are trying to solve and $g(\mathbf{x(p)})$ be the objective function of the system being optimized. The gradient we'd like to compute is given by:

$$\frac{\partial g}{\partial \mathbf{P}} = \{\frac{\partial g}{\partial P_1}, \frac{\partial g}{\partial P_2}, ..., \frac{\partial g}{\partial P_n}\} \tag{2.2}$$

By using the chain rule we can write the gradient as

$$\frac{\partial g}{\partial \mathbf{P}} = \frac{\partial g}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \mathbf{P}} \tag{2.3}$$

Generally $\frac{\partial g}{\partial \mathbf{x}}$ is easy to calculate, because F is written in terms of $\mathbf{x}$. Calculating second term is more involved. To do this, we differentiate with respect to $\mathbf{p}$ and solve for $\frac{\partial \mathbf{x}}{\partial \mathbf{P}}$

$$\frac{\partial}{\partial \mathbf{P}}[\mathbf{Ax}] = \frac{\partial \mathbf{A}}{\partial \mathbf{P}}\mathbf{x} + \mathbf{A}\frac{\partial \mathbf{x}}{\partial \mathbf{P}} = \frac{\partial \mathbf{b}}{\partial \mathbf{P}} \tag{2.4}$$

$$\frac{\partial \mathbf{x}}{\partial \mathbf{P}} = \mathbf{A}^{-1}\left[\frac{\partial \mathbf{b}}{\partial \mathbf{P}} - \frac{\partial \mathbf{A}}{\partial \mathbf{P}}\mathbf{x}\right] \tag{2.5}$$

thus the full derivative of g becomes

$$\frac{\partial g}{\partial \mathbf{P}} = \frac{\partial g}{\partial \mathbf{x}}\mathbf{A}^{-1}\left[\frac{\partial \mathbf{b}}{\partial \mathbf{P}} - \frac{\partial \mathbf{A}}{\partial \mathbf{P}}\mathbf{x}\right] \tag{2.6}$$

Now by parenthesiing the first term we can define the "adjoint" term:

$$\boldsymbol{\lambda}^T = \frac{\partial g}{\partial \mathbf{x}}\mathbf{A}^{-1} \tag{2.7}$$

Multiplying by $\mathbf{A}$ on the right hand side and transposing we get

$$\mathbf{A}^T\lambda = \left[\frac{\partial g}{\partial \mathbf{x}}\right]^T \tag{2.8}$$

Grouping everything together, the total derivative of the objective becomes:

$$\frac{\partial g}{\partial \mathbf{P}} = \lambda^T \left[\frac{\partial \mathbf{b}}{\partial \mathbf{P}} - \frac{\partial \mathbf{A}}{\partial \mathbf{P}} \mathbf{x} \right] \tag{2.9}$$

If $\mathbf{A(P)}$ and $\mathbf{b(P)}$ are generally given analytically, their derivatives are easy to compute. This leaves us with needing to compute two linear problems, i.e. solving for $\mathbf{x}$ and λ, which can be done using well understood methods for solving linear systems.

A rigorous treatment of Mie scattering theory is a subject for entire textbooks, and can be found elsewhere [62, 70, 33], but we will briefly summarize it here. Mie scattering analytically treats the scattering from spherical particles by decomposing the incident E_{in} and the scattered field E_{out} as a weighted sum of spherical vector wave functions (SWVF) $\psi^{(1)}$ and $\psi^{(3)}$ respectively. For the single sphere case, the electric fields can be written as

$$E_{in}(\vec{r}) = \sum_n a_n \psi^{(1)}(\vec{r} - \vec{r}_0) \tag{2.10}$$

$$E_{out}(\vec{r}) = \sum_n b_n \psi^{(3)}(\vec{r} - \vec{r}_0) \tag{2.11}$$

With a_n and b_n are the expansion coefficients related by T-matrices:

$$b = Ta$$

These T-matrices can be computed analytically. In order to generalize this theory to include multiple-sphere scattering, we write the incident field onto each sphere as a sum of the incident fields and the scattered fields from all the other spheres

$$E_{in}^{i} = E_{in} + \sum_{i \neq i'} E_{out}^{i'} \tag{2.12}$$

where i is the i'th sphere. This leads to a coupled system of linear equations for b:

$$Mb = Ta \tag{2.13}$$

with

$$M = I - TW \tag{2.14}$$

where W is the matrix that couples the interaction between spheres, and depends only on the positions of the spheres with respect to each other [33]. Thankfully, a code base exists that can handle multi-sphere scattering using Lorenz-Mie theory, called CELES. Using the ideas from the previous section, we can calculate the gradient of a FOM defined in terms of the scattered electric field [120]. Let $g((b(p)))$ be our FOM. We can define the gradient by:

$$\frac{\partial g}{\partial P} = 2Re\left[\frac{\partial g}{\partial P}\frac{\partial b}{\partial P}\right] \tag{2.15}$$

where since we're taking derivatives of real numbers with respect to complex numbers, we take the $2Re$ part of the derivative by Wirtinger's definition [75]. By combining equations 2.13 and 2.14, taking the derivative, an solving for $\frac{\partial b}{\partial P}$ we get:

$$\frac{\partial b}{\partial P} = M^{-1}\left[\frac{\partial T}{\partial P}a + \frac{\partial T}{\partial P}Wb\right] \tag{2.16}$$

Similarly to the process in the previous section, the adjoint term can be defined as

$$M^T\lambda = \left[\frac{\partial g}{\partial b}\right]^T \tag{2.17}$$

and the full derivative becomes

$$\frac{\partial g}{\partial P} = 2Re\left[\lambda^T\left(\frac{\partial T}{\partial P}a + \frac{\partial T}{\partial P}Wb\right)\right] \tag{2.18}$$

where $\frac{\partial T}{\partial P}$ can be calculated analytically.

This approach for the optimization and design of large scale metasurfaces without the use of LPA. In [120], this approach was used to design a singlet and a doublet lens, with one and two layers of spherical scatterers respectively. However this approach is limited to only spherical scatterers. Furthermore, we were not able to find a parameter range for spherical scatterers spanned a $0 - 2\pi$ phase shift without considerable losses, which is a common requirement when designing metaphotonic structures, so we extended this framework to ellipsoidal scatterers.

In order to do this, we must first take a look at the structure of the T-matrices [103, 73, 30]:

$$T = RgQ(Q)^{-1} \tag{2.19}$$

Where Q is given by

$$Q = \begin{bmatrix} \bar{P} & \bar{R} \\ \bar{S} & \bar{U} \end{bmatrix} \tag{2.20}$$

Where $\bar{P}, \bar{R}, \bar{S}$ and $\bar{U}$ are square submatrices given by

$$\bar{P}_{lml'm'} = -ikk_s J^{(21)}_{lml'm'} - ik^2 J^{(12)}_{lml'm'}, \tag{2.21}$$

$$\bar{R}_{lml'm'} = -ikk_s J^{(11)}_{lml'm'} - ik^2 J^{(22)}_{lml'm'}, \tag{2.22}$$

$$\bar{S}_{lml'm'} = -ikk_s J^{(22)}_{lml'm'} - ik^2 J^{(11)}_{lml'm'}, \tag{2.23}$$

$$\bar{U}_{lml'm'} = -ikk_s J^{(12)}_{lml'm'} - ik^2 J^{(21)}_{lml'm'}, \tag{2.24}$$

with

$$J^{(pq)}_{lml'm'} = (-1)^m \int_S dS\hat{n}(\mathbf{r}) \cdot \Psi^{(1)}_{p,l',m'}(k_s r, \theta, \phi) \times \Psi^{(3)}_{q,l,-m}(kr, \theta, \phi) \tag{2.25}$$

The terms $\Psi^{(1)}_{p,l',m'}$ and $\Psi^{(3)}_{q,l,-m}$ in equation 2.25 are given by

$$\mathbf{\Psi}^{(\nu)}_{1lm}(\mathbf{r}) = \frac{e^{im\phi}}{\sqrt{2l(l+1)}} b_l(kr) \left[im\pi_{lm}(\theta)\hat{\theta} - \tau_{lm}(\theta)\hat{\phi} \right], \tag{2.26}$$

$$\begin{aligned} \mathbf{\Psi}^{(\nu)}_{2lm}(\mathbf{r}) = & \frac{e^{im\phi}}{\sqrt{2l(l+1)}} \Bigg\{ l(l+1) \frac{b_l(kr)}{kr} P_l^{|m|}(cos\theta)\hat{r} \\ & + \frac{1}{kr} \frac{\partial (krb_l(kr))}{\partial (kr)} \left[\tau_{lm}(\theta)\hat{\theta} + im\pi_{lm}(\theta)\hat{\phi} \right] \Bigg\}, \end{aligned} \tag{2.27}$$

where

$$\pi_{lm}(\theta) = \frac{P_l^{|m|}(cos\theta)}{sin\theta}, \tau_{lm}(\theta) = \frac{\partial P_l^{|m|}(cos\theta)}{\partial \theta}. \tag{2.28}$$

$P_l^m(x)$ is the associated Legendre polynomial. j_l is the spherical Bessel function of order l, and b_l is either a spherical Bessel function (j_l) for $\nu = 1$ or spherical Hankel function of the first kind ($h_l^{(1)}$) of order l for $\nu = 3$, depending on whether RgQ or Q is being computed.

In order to extend this method to ellipsoids, we must parameterize the ellipsoidal surface term dS in equation 2.25 in spherical coordinates, and take the gradient of the T-matrices of the ellipsoidal scatterers in terms of their free parameters - (a, b, c, ϕ), where a, b and c are

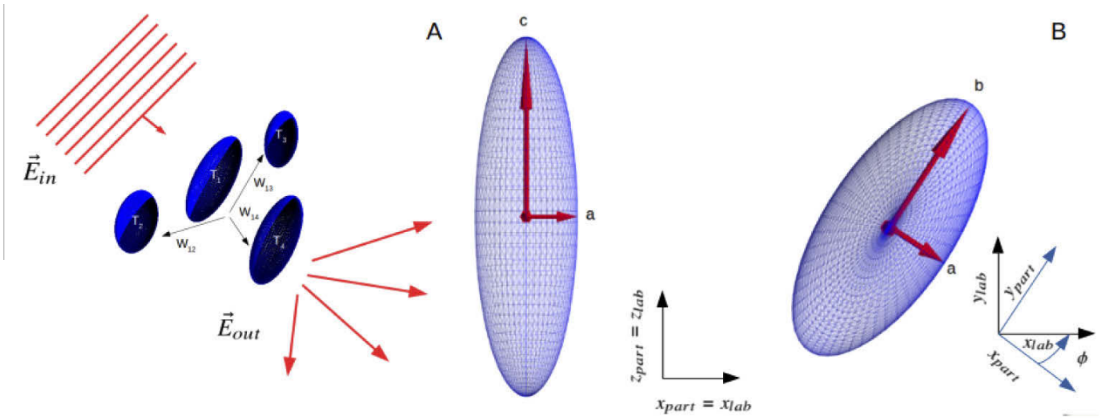


Figure 2.1: **A**.Mie scattering schematic. Light is incident onto the set of ellipsoidal scatterers.Each scatterer has an associated T-matrix. The incident field onto each scatterer is described by the incident field E_{in} and the scattered fields from all other scatterers. The inter-particle coupling is represented by the matrix W which describes the coupling between ellipsoids. **B**. Design parameters for ellipsoidal scatterers. The semi-major axes are taken to always be aligned with the particle frame: semi-major axis$a$is aligned with the x_{part} axis,b with the y_{part} axis, and c with the z_{part} axis. The rotation ϕ is about the z-axis, with the counterclockwise direction defined as a positive rotation.

the semi-major axes of the ellipsoid, and ϕ is its rotation, as shown in Fig. 2.1. In spherical coordinates, the product of the unit normal and the infinitesimal area is:

$$dS\hat{n}(\mathbf{r}) = r^2 sin(\theta)\sigma(\mathbf{r})d\theta d\phi, \tag{2.29}$$

and σ is given by:

$$\sigma(\mathbf{r}) = \hat{r} - \hat{\theta}\frac{1}{r}\frac{\partial r}{\partial \theta} - \hat{\phi}\frac{1}{rsin\theta}\frac{\partial r}{\partial \theta}. \tag{2.30}$$

In this case, r is parameterizing the surface of a particle, and for an ellipsoid in spherical coordinates, r is given by:

$$r(\theta,\phi) = \left[sin^2\theta\left(\frac{cos^2\phi}{a^2} + \frac{sin^2\phi}{b^2}\right) + \frac{cos^2\theta}{c^2}\right]^{-1/2} \tag{2.31}$$

To compute RgQ rather than Q, we simply need to replace $\mathbf{\Psi}^{(3)}$ in the J integrals with $\mathbf{\Psi}^{(1)}$.

The derivative of the T-Matrix of a particle with respect to some parameter p is given by:

$$\frac{\partial T}{\partial P} = \left(\frac{\partial RgQ}{\partial P} - T\frac{\partial Q}{\partial P}\right)Q^{-1}, \tag{2.32}$$

Hence we need to find the derivatives of the sub-matrices $\bar{P}$, $\bar{R}$, $\bar{S}$, and $\bar{U}$ with respect to P. This requires us to take the derivatives of the surface integrals J with respect to the parameter P.

The expressions for the derivatives with respect to a spatial variable (a, b, c) are as follows where P represents any of the ellipsoid axes:

$$\begin{aligned}\frac{\partial J^{(11)}_{lml'm'}}{\partial P} = &- i\iint \alpha_{lml'm'}\left(m'\pi_{l'm'}\tau_{lm} + m\pi_{lm}\tau_{l'm'}\right)\\ &\left[r\left(k\frac{\partial b_l}{\partial P}j_{l'} + k_s b_l\frac{\partial j_{l'}}{\partial P}\right) + 2b_l j_{l'}\right]rsin\theta d\theta d\phi,\end{aligned} \tag{2.33}$$

$$\begin{aligned}\frac{\partial J^{(12)}_{lml'm'}}{\partial P} = &\iint \alpha_{lml'm'}\Bigg\{\left[\frac{\partial \bar{R}^{(12)}_{lml'm'}}{\partial r} + (\Theta^{(12)}_{lml'm'}E_\theta + \Phi^{(12)}_{lml'm'}E_\phi)\frac{\partial \rho_{l,l'}}{\partial r}\right]\frac{\partial r}{\partial P}\\ &+\left(\Theta^{(12)}_{lml'm'}\frac{\partial E_\theta}{\partial P} + \Phi^{(12)}_{lml'm'}\frac{\partial E_\phi}{\partial P}\right)\rho_{l,l'}\Bigg\}d\theta d\phi,\end{aligned} \tag{2.34}$$

$$\begin{aligned}\frac{\partial J^{(21)}_{lml'm'}}{\partial P} =& \iint \alpha_{lml'm'}\Bigg\{\left[\frac{\partial \bar{R}^{(21)}_{lml'm'}}{\partial r} + (\Theta^{(21)}_{lml'm'}E_\theta + \Phi^{(21)}_{lml'm'}E_\phi)\frac{\partial \rho_{l,l'}}{\partial r}\right]\frac{\partial r}{\partial P} \\ &+ \left(\Theta^{(21)}_{lml'm'}\frac{\partial E_\theta}{\partial P} + \Phi^{(21)}_{lml'm'}\frac{\partial E_\phi}{\partial P}\right)\rho_{l,l'}\Bigg\}d\theta d\phi,\end{aligned} \tag{2.35}$$

$$\begin{aligned}\frac{\partial J^{(22)}_{lml'm'}}{\partial P} =& \iint \alpha_{lml'm'}\Bigg\{\left[\frac{\partial \bar{R}^{(22)}_{lml'm'}}{\partial r} + \frac{\partial \Theta^{(22)}_{lml'm'}}{\partial r}E_\theta + \frac{\partial \Phi^{(22)}_{lml'm'}}{\partial r}E_\phi\right]\frac{\partial r}{\partial P} \\ &+ \Theta^{(22)}_{lml'm'}\frac{\partial E_\theta}{\partial P} + \Phi^{(22)}_{lml'm'}\frac{\partial E_\phi}{\partial P}\Bigg\}d\theta d\phi,\end{aligned} \tag{2.36}$$

where we have defined:

$$\alpha_{lml'm'} = \frac{(-1)^m(1+(-1)^{m'-m})(1+(-1)^{l'+l+1}}{2\sqrt{l(l+1)l'(l'+1)}}e^{i(m'-m)\phi} \tag{2.37}$$

k and k_s are the k vectors of light in the medium surrounding the particle, and in the particle itself. Then we define:

$$E_\theta = \frac{cos^2\phi}{a^2} + \frac{sin^2\phi}{b^2} - \frac{1}{c^2} \tag{2.38}$$

$$E_\phi = \frac{1}{b^2} - \frac{1}{a^2}, \tag{2.39}$$

$$\rho_{l,l'} = r^3 j_{l'} b_l, \tag{2.40}$$

Now, we can define the specific terms used to construct each J surface integral. For $J^{(12)}$, we define:

$$\begin{aligned}\frac{\partial \bar{R}^{(12)}_{lml'm'}}{\partial r} =& \frac{sin\theta}{k}(mm'\pi_{l'm'}\pi_{lm} + \tau_{l'm'}\tau_{lm})\Bigg(j_{l'}\frac{\partial(krb_l)}{\partial(kr)} \\ &+r\left(k_s\frac{\partial j_{l'}}{\partial r}\frac{\partial(krb_l)}{\partial(kr)} + kj_{l'}\frac{\partial}{\partial r}\left(\frac{\partial(krb_l)}{\partial(kr)}\right)\right)\Bigg),\end{aligned} \tag{2.41}$$

$$\Theta^{(12)}_{lml'm'} = -\frac{sin\theta}{k}l(l+1)P_l^{|m|}\tau_{l'm'}, \tag{2.42}$$

$$\Phi^{(12)}_{lml'm'} = -i\frac{sin\theta}{k}l(l+1)m'P_l^{|m|}\pi_{l'm'}. \tag{2.43}$$

For $J^{(21)}$, we define:

$$\begin{aligned}\frac{\partial \bar{R}^{(21)}_{lml'm'}}{\partial r} =& -\frac{sin\theta}{k_s}(mm'\pi_{l'm'}\pi_{lm} + \tau_{l'm'}\tau_{lm})\Bigg(\frac{\partial(k_srj_{l'})}{\partial(k_sr)}b_l \\ &+r\left(k_s\frac{\partial}{\partial r}\left(\frac{\partial(k_srj_{l'})}{\partial(k_sr)}\right)b_l + k\frac{\partial b_l}{\partial r}\frac{\partial(k_srj_{l'})}{\partial(k_sr)}\right)\Bigg),\end{aligned} \tag{2.44}$$

$$\Theta^{(21)}_{lml'm'} = \frac{sin\theta}{k_s} l'(l'+1) P^{|m'|}_{l'} \tau_{lm}, \tag{2.45}$$

$$\Phi^{(21)}_{lml'm'} = -i\frac{sin\theta}{k_s} l'(l'+1) m P^{|m'|}_{l'} \pi_{lm}. \tag{2.46}$$

Finally, for $J^{(22)}$ we define:

$$\begin{aligned}\Theta^{(22)}_{lml'm'} =& i\frac{r^2 sin\theta}{kk_s}\Bigg(m'l(l+1)\frac{\partial(k_s r j_{l'})}{\partial(k_s r)} b_l P^{|m|}_l \pi_{l'm'} \\ &+ ml'(l'+1) j_{l'} \frac{\partial(krb_l)}{\partial(kr)} P^{|m'|}_{l'} \pi_{lm}\Bigg)\end{aligned} \tag{2.47}$$

$$\begin{aligned}\Phi^{(22)}_{lml'm'} =& \frac{r^2 sin\theta}{kk_s}\Bigg(l'(l'+1) j_{l'} P^{|m'|}_{l'} \frac{\partial(krb_l)}{\partial(kr)} \tau_{lm} \\ &- l(l+1)\frac{\partial(k_s r j_{l'})}{\partial(k_s r)} \tau_{l'm'} b_l P_{lm}\Bigg)\end{aligned} \tag{2.48}$$

and the three derivative terms:

$$\begin{aligned}\frac{\partial \bar{R}^{(22)}_{lml'm'}}{\partial r} =& -i\frac{sin\theta}{kk_s}(m'\pi_{l'm'}\tau_{lm} + m\pi_{lm}\tau_{l'm'}) \\ &\Bigg(k\frac{\partial}{\partial r}\left(\frac{\partial(krb_l)}{\partial(kr)}\right)\frac{\partial(k_s r j_{l'})}{\partial(k_s r)} \\ &+ k_s\frac{\partial}{\partial r}\left(\frac{\partial(k_s r j_{l'})}{\partial(k_s r)}\right)\frac{\partial(krb_l)}{\partial(kr)}\Bigg)\end{aligned} \tag{2.49}$$

$$\begin{aligned}\frac{\partial \Theta^{(22)}_{lml'm'}}{\partial r} =& i\frac{sin\theta}{kk_s}\Bigg(ml'(l'+1)P^{|m'|}_{l'}\pi_{lm}\Bigg(2r\frac{\partial(krb_l)}{\partial(kr)} j_{l'} \\ &+ r^2\Bigg(k\frac{\partial}{\partial r}\left(\frac{\partial(krb_l)}{\partial(kr)}\right) j_{l'} + k_s\frac{\partial j_{l'}}{\partial r}\frac{\partial(krb_l)}{\partial(kr)}\Bigg)\Bigg) \\ &+ m'l(l+1)P^{|m|}_l \tau l'm'\Bigg(2rb_l\frac{\partial(k_s r j_{l'})}{\partial(k_s r)} \\ &+ r^2\Bigg(k\frac{\partial b_l}{\partial r}\frac{\partial(k_s r j_{l'})}{\partial(k_s r)} + k_s b_l\frac{\partial}{\partial r}\left(\frac{\partial(k_s r j_{l'})}{\partial(k_s r)}\right)\Bigg)\Bigg)\Bigg)\end{aligned} \tag{2.50}$$

$$\begin{aligned}\frac{\partial \Phi^{(22)}_{lml'm'}}{\partial r} =&\frac{sin\theta}{kk_s}\Bigg(l'(l'+1)P^{|m'|}_{l'}\tau_{lm}\Bigg(2r\frac{\partial(krb_l)}{\partial(kr)}j_{l'}\\&+r^2\Bigg(\frac{\partial}{\partial r}\left(\frac{\partial(krb_l)}{\partial(kr)}\right)j_{l'}+k_s\frac{\partial j_{l'}}{\partial r}\frac{\partial(krb_l)}{\partial(kr)}\Bigg)\Bigg)\\&-l(l+1)P^{|m|}_{l}\tau_{l'm'}\Bigg(2rb_l\frac{\partial(k_srj_{l'})}{\partial(k_sr)}\\&+r^2\Bigg(k\frac{\partial b_l}{\partial r}\frac{\partial(k_srj_{l'})}{\partial(k_sr)}\frac{\partial}{\partial r}\left(\frac{\partial(k_srj_{l'})}{\partial(k_sr)}\right)\Bigg)\Bigg)\Bigg).\end{aligned} \tag{2.51}$$

Now with these J integrals, we can compute the quantity $\frac{\partial T}{\partial P}$ for a given axis of an ellipsoid in its own particle frame where a, b, and c are aligned along the x_{part}, y_{part}, and z_{part} axes.

In addition to computing the response of the T-Matrix of the ellipsoid to the contraction or extension of one of its axes, we are also interested in its response to rotations about the z_{part} axis. To do this we will first define the transformation of the T-Matrix or a derivative matrix from the particle frame to some rotated lab frame that has new axes x_{lab} and y_{lab}, but shares $z_{lab} = z_{part}$. Given some rotation angle ϕ_{rot}, we can then define our new axes:

$$x_{lab} =x_{part}cos(\phi_{rot}) + y_{part}sin(\phi_{rot}) \tag{2.52a}$$

$$y_{lab} =-x_{part}sin(\phi_{rot}) + y_{part}cos(\phi_{rot}) \tag{2.52b}$$

$$z_{lab} =z_{part} \tag{2.52c}$$

The general form of this orthogonal transformation in three dimensions can be represented by the Euler angles α, β, and γ. The general transformation of an element of an operator O from the particle frame to the lab frame can be written as[73]:

$$O^{lab}_{plmp'l'm'}(\alpha,\beta,\gamma) = \sum_{m_1=-l}^{l}\sum_{m_2=-l'}^{l'} D^{l}_{mm_1}(\alpha,\beta,\gamma)O^{particle}_{plm_1p'l'm_2}D^{l'}_{m_2m'}(-\gamma,-\beta,-\alpha), \tag{2.53}$$

where the D operator is a Wigner D-function. It can be represented as:

$$D^{l}_{m'm}(\alpha,\beta,\gamma) = e^{-im'\alpha}d^{l}_{m'm}(\beta)e^{-im\gamma}, \tag{2.54}$$

where $d^{l}_{m'm}(\beta)$ is Wigner's (small) d-matrix given by:

$$d^{l}_{m'm}(\beta) = \langle l,m'|\, e^{-i\beta J_y}\, |l,m\rangle\,. \tag{2.55}$$

However, as we are only concerned with rotations about the z axis, we can simplify our expressions knowing that α is our only nonzero angle, and equation (56) becomes:

$$O^{lab}_{plmp'l'm'}(\alpha,0,0)=\sum_{m_1=-l}^{l}\sum_{m_2=-l'}^{l'}D^{l}_{mm_1}(\alpha,0,0)O^{particle}_{plm_1,p'l'm_2}D^{l'}_{m_2m'}(0,0,-\alpha). \tag{2.56}$$

In this case, our D operator has a much simplified form:

$$D^{l}_{m'm}(\alpha,0,0)=e^{-im'\alpha}\delta_{m'm} \tag{2.57a}$$

$$D^{l}_{m'm}(0,0,\gamma)=e^{-im\gamma}\delta_{m'm}. \tag{2.57b}$$

Combining equations 2.56 and 2.57, we obtain a simple expression transforming O from the particle frame to the lab frame:

$$O^{lab}_{plmp'l'm'}(\alpha)=e^{i(m'-m)\alpha}O^{particle}_{plmp'l'm'}. \tag{2.58}$$

Equation 2.58 is applicable to for transforming both T-matrices and the derivative matrices computed in the particle frame into the lab frame. It also gives us a prescription for computing the derivative matrix with respect to the particle's angular orientation. We already have derivatives characterizing the response of the particle to contractions and extensions of its principal axes, and can now rotate these to a lab frame where the particle has an arbitrary angular orientation relative to the z axis. We can now compute the derivative with respect to the particle's angular orientation α as:

$$\frac{\partial T^{lab}_{plmp'l'm'}(\alpha)}{\partial\alpha}=i(m'-m)e^{i(m'-m)\alpha}T^{particle}_{plmp'l'm'}. \tag{2.59}$$

With equations 2.32 and 2.59, we have characterized the derivatives of the T-Matrix representing an ellipsoid with respect to its axes and orientation. These integrals are implemented in MATLAB, and performed using Gaussian quadrature.

Its worth noting that under the T-matrix formalism, entire scattering particles are treated as single multi-pole sources at the particle center [91]. Therefore electric fields outside of the sphere circumscribing a non-spherical particle are not guaranteed to converge. This limits us to particle densities where the circumscribing spheres of the ellipsoids do not overlap with surfaces of nearby ellipsoids.

This is the first time these derivatives were calculated as far as we know. So to validate that these derivatives were in fact accurate, we created a set of 216 ellipsoids with geometries corresponding to permutations of a, b, c between $50 - 300nm$ in steps of $50nm$. Then we computed the T-matrices and their derivatives for each individual ellipsoid analytically and numerically. We denote the numerical derivative of the T-matrices with respect to some parameter $P \in \{a, b, c, \phi\}$ as $\frac{\partial T^N}{\partial P}$

$$\frac{\partial T^N}{\partial P} = \frac{T(P + \Delta P) - T(P)}{\Delta P} + O(\Delta P^2) \tag{2.60}$$

The analytical derivative is denoted as $\frac{\partial T^A}{\partial P}$. We then compared the numerical and analytical derivative of each T-matrix, and define the mean error as

$$error = mean\left(\sum_{ij} \left| \frac{\partial T^N_{i,j}}{\partial P} - \frac{\partial T^A_{i,j}}{\partial P} \right| \right) \tag{2.61}$$

where the indices I, j are the individual elements of the T-matrix. We varied the step sizes for Δa,Δb,Δc from 10 to 10^{-4} nm and for $\Delta\phi$ from 10^{-1} to 10^{-5} radians. Fig. 2.2 shows that as the step size is reduced, the numerical derivative converges closer to the analytical derivative as expected.

2.3 Inverse design and optimization of meta-optics

Using the aforementioned TMM formalism and adjoint optimization method, we inverse designed a high numerical aperture (NA) lens, optimized a forward designed lens, and inverse designed a polarization switched device.

2.3.1 High numerical aperture lens

The high numerical aperture metasurface lens was designed for a $915nm$ incident wavelength, at refractive index 3.56. This design space was picked after an extensive parameter search. The search was done by simulating libraries of ellipsoidal scatterers under periodic boundary

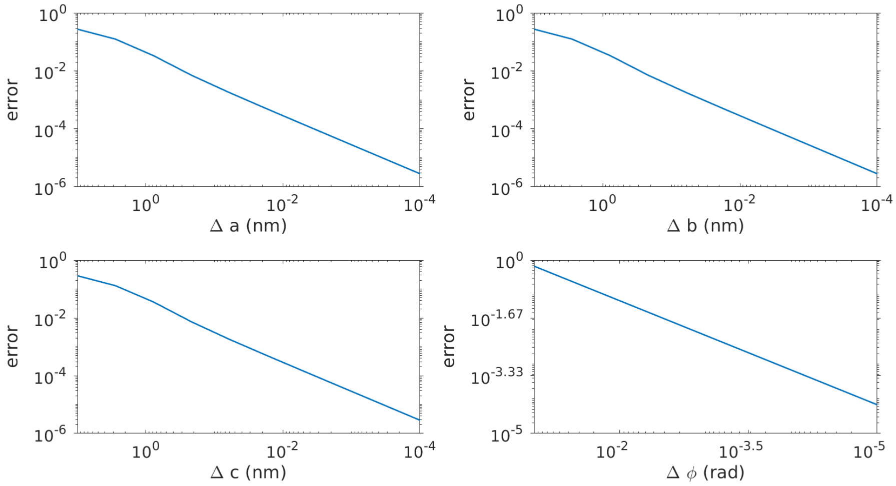


Figure 2.2: Verification of the analytical T-matrix derivatives. **A** shows the error between the analytical T-matrix derivative and the numerical derivative with respect to semi-major axis a, **B** with respect to b, **C** with respect to c. **D** shows the T-matrix derivative with respect to the azimuthal rotation of the ellipsoid ϕ. As the step size of the numerical approximation to the derivative gets smaller, the mean error between numerical and analytical derivative gets closer to 0, which implies that the analytical derivatives are valid

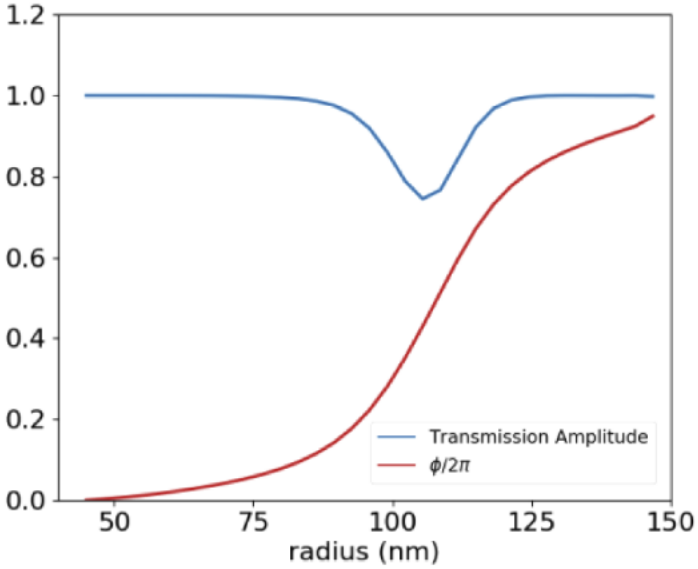


Figure 2.3: Transmission of individual scatterers with periodic boundary conditions as a function of the radius of the ellipsoids (semi-major axes a=b). Ellipsoid height is fixed to be $600nm$. The lattice contant is $450nm$.

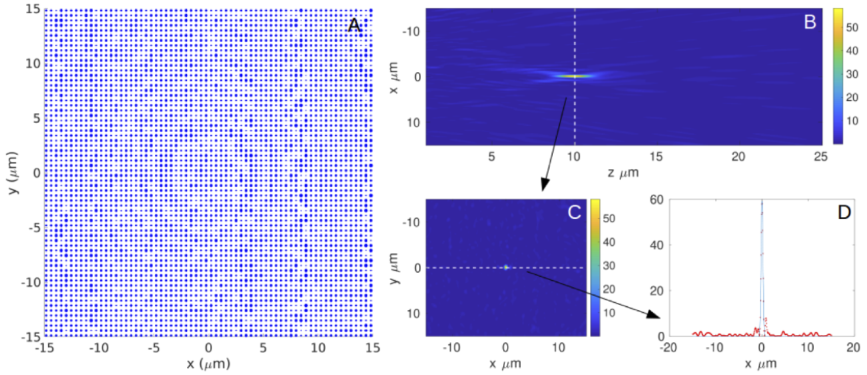


Figure 2.4: **A** Final distribution of scatterers with periodicity $450nm$ for the inverse designed lens. semi-major axes a and b are allowed to range between 40 and $150nm$. Semi-major axis c is allowed to range between 40 and $300nm$.**B** the field cross-section in the x-z plane at $y = 0\mu m$, **C** the cross-section in the x-y plane at $z = 10\mu m$. **D** shows the Gaussian fit to the field at the focal spot $z = 10\mu m$ along $x = 0\mu m$. In order to calculate the lens efficiency, the full-width at half-maximum (FWHM) was calculated for the fitted Gaussians. The integral of the field intensity around the disk $d = 3 \times FWHM$ about the center of the focal spot was calculated, and then divided by the total incident field intensity. The units of all plots are arbitrary light intensity units. The efficiency of the inverse designed lens was calculated to be 3.38%

conditions with rigorous coupled wave analysis (RCWA) [66]. In simulation, the heights, pitch, and refractive index of the ellipsoids were fixed, while varying the a and b semi-major axes of the ellipsoids concurrently. Fig. 2.3 shows the plot of the chosen parameter space. The lens designed was $30\mu m$ in diameter with a focal length of $10\mu m$. In order to design a lens, we use the FOM:

$$g = (I^T - I^A(x, y, z = F))^2 \tag{2.62}$$

where I^T is some target intensity value at the focal spot of the lens, and I^A was the actual intensity calculated by T-matrix theory $I^A(x, y, z) = E^\dagger(x, y, z)E(x, y, z)$, where the † is the complex conjugate operator. The initial conditions for this lens were chosen to be $a = b = 100nm$ and $c = 300nm$. The lattice periodicity was $450nm$. The radii were allowed to vary between $40nm$ and $150nm$ for a, b and $40 - 300nm$ for axis c. The constraints were chosen such that the circumscribing spheres of the ellipsoids would never overlap the surface of the nearby ellipsoids. Fig. 2.4 summarizes the results. 2.4B shows theres a clear focal spot at $10\mu m$. The efficiency of this device was calculated by fitting a Gaussian to the field profile at the focal spot $z = 10\mu m$ and $x = 0$ as shown in 2.4D. Then we found the full-width at half maximum of the Gaussian, and integrated the intensity of the field at that focal spot, and divided it over the total intensity of the incident light. This quantity is defined as the efficiency η of the lens:

$$\eta = \frac{\int\int_\Omega E(x, y, z = F)^\dagger E(x, y, z = F)dxdy}{\int\int_{x,y} E(x, y, z = 0)^\dagger E(x, y, z = 0)dxdy} \tag{2.63}$$

$$\Omega := x^2 + y^2 < (3 \times FWHM)^2$$

Ω is the surface around the focal spot which we integrate over. The efficiency is calculated to be 3.38% for this lens.

2.3.2 Inverse design of polarization switched focal length lens

We then designed a lens with a diameter of 40 μm, and focal lengths of 20 μm (NA $\sim$ 0.71) and 30 μm (NA $\sim$ 0.55) for the x and y polarizations respectively. The lattice constant for

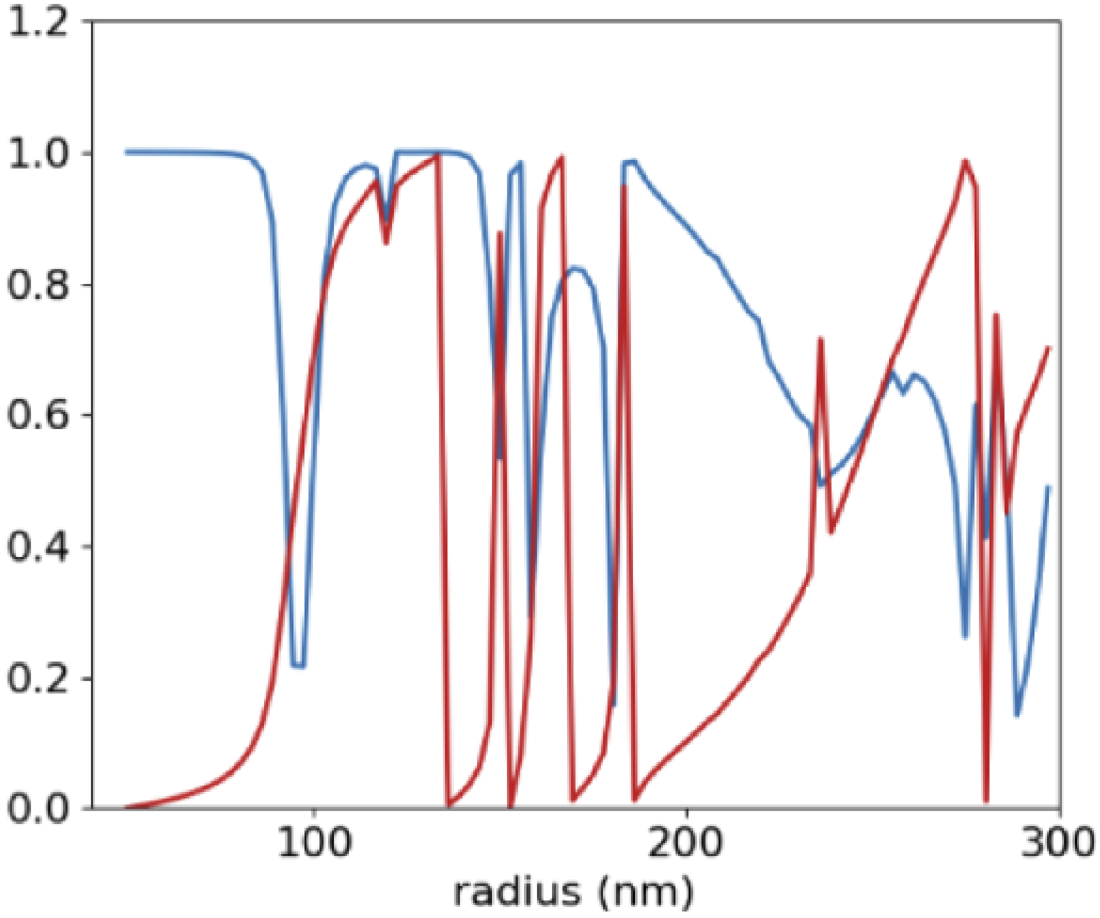


Figure 2.5: Transmission plot for the ellipsoids used for the polarization switched device. Ellipsoid height is fixed at 715nm and the lattice constant is 650nm.

this lens was taken to be $650nm$. The transmission plots of the design space can be seen in Fig. 2.5. Semi-major axis radii a and b were allowed to range between $40nm$ and $292.5nm$. Semi-major axis radius c was allowed to range between $40nm$ and $357.5nm$. The azimuthal rotation around the z axis of the scatterers, was allowed to range from $-\pi/2$ to $\pi/2$.

The optimization problem was framed as a min-max optimization problem [80]. For this optimization, we write the total FOM as a sum of FOM's for each polarization, given by

$$g = g_x + g_y \tag{2.64}$$

with g_x and g_y being the figures of merit for the x and y polarizations respectively, and are

$$g_x = (I^T_{max}(0,0,20\mu m) - I^A(0,0,20\mu m) + I^T_{min}(0,0,30\mu m) - I^A(0,0,30\mu m))^2 \tag{2.65}$$

$$g_y = (I^T_{max}(0,0,30\mu m) - I^A(0,0,30\mu m) + I^T_{min}(0,0,20\mu m) - I^A(0,0,20\mu m))^2 \tag{2.66}$$

Here, I^T_{max} is some arbitrary large value (we chose 200), denoting the fact that light intensity at that spot should be maximized, while I^T_{min} is a regularization term, denoting that the field intensity at that point should be kept small. To design this device, we minimize the maximum (worst) FOM iteratively until we converge to a local minimum:

$$\min_{P\in\{a,b,c,\phi\}} \max(g_x, g_y) \tag{2.67}$$

The performance of the final device is shown in Fig. 2.6. There is a clear focal spot at $z = 20\mu m$ for x-polarized light, and no focal spot at $z = 30\mu m$ (Fig. 2.6B) and for x-polarized light, we see a focal spot at $z = 20\mu m$ and no focal spot at $z = 30\mu m$ (Fig. 2.6E).

We calculated the efficiency of this lens for each polarization using the method described in the previous section, and found values of $\eta = 2.31\%$ for the x-polarization and $\eta = 3.38\%$ for y polarization. Another relevant quantities we can define to characterize the performance of this device are the contrast ratios of the focal spots. We define and report two different contrast quantities. The first one is the ratio between the value of intensity at the focal spot, where light should be maximized, to the ratio of light at the focal spot of the orthogonal polarization. We found the values for these ratios to be $\frac{I(0,0,30\mu m)}{I(0,0,20\mu m}=8.75$ for y polarized light and $\frac{I(0,0,20\mu m)}{I(0,0,30\mu m}=5.11$ for x polarized light. The second ratio we define to be the intensity at the focal spot for one polarization to the intensity at that same spot for the orthogonal polarization. We found these values to be $\frac{I_x(0,0,20\mu m}{I_y(0,0,20)}=5.58$ and $\frac{I_y(0,0,30\mu m}{I_x(0,0,30)}=5.92$.

2.3.3 Metasurface lens optimization

Finally, we discuss the optimization of a forward designed metalens using our inverse design method. We used the library of scatterers computed RCWA shown in 2.3. We discretized the design space in the x-y plane, using a scatterer periodicity of 450 nm, and by using the phase equation for a lens given by

$$\phi(x,y)=\frac{2\pi}{\lambda}(\sqrt{(x^2+y^2)+F^2}-F) \tag{2.68}$$

we placed a scatterer at each discrete point (x,y), with a phase response closest to the phase needed to focus light. The lens we designed has a diameter of $30\mu m$ and a focal length of $10\mu m$. This devices performance is summarized in Fig. 2.7. By using the same approach from the previous sections, we calculated the device's efficiency to be 25.59%.

To optimize this device, we started off with the scatterer distribution given by the forward design as the initial condition and maximize the light intensity at a the focal spot. The performance of the optimized device is summarized in Fig 2.7. The efficiency of this device was calculated to be 32.00%, which is a 6.41% improvement over the forward design lens. On average, each individual scatterer was changed by approximately $3.03nm$ along the a axis, $4.8nm$ along the b axis, and $0.17nm$ along the c axis. The standard deviations for each axis are $3.53nm$, $4.89nm$, $0.42nm$ respectively. The maximum changes for each axis were $33.42nm$, $33.43nm$ and $5.57nm$ respectively. Its worth noting that this improvement implies that lenses designed by the conventional forward design methods are not necessarily locally optimal in the metasurface design space, even for high contrast designs. We can also see that the initial conditions are very important for the final design, as starting with identical ellipsoids, the final design provides very low efficiency. In fact, based on our analysis, we

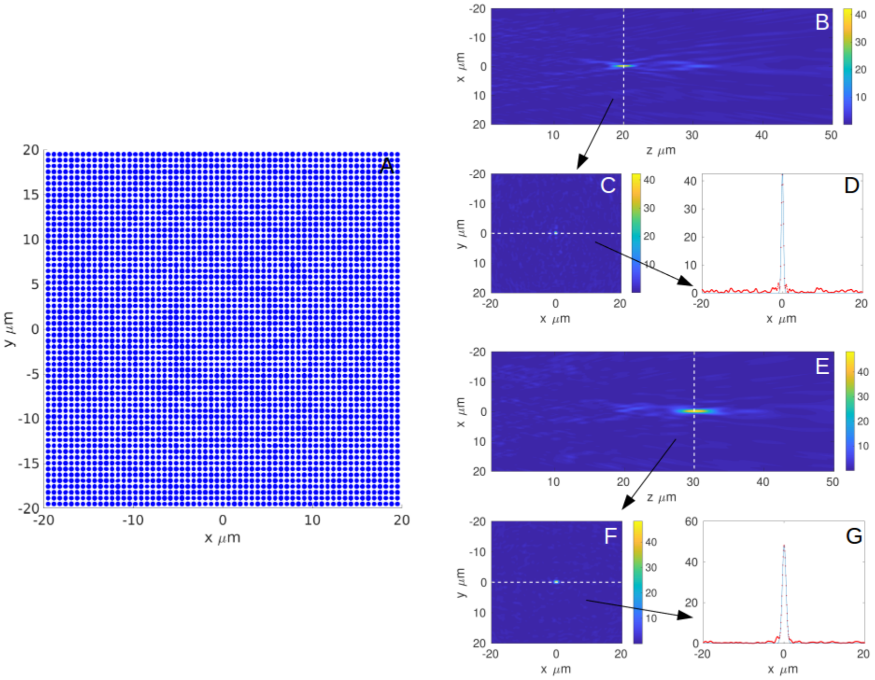


Figure 2.6: **A** Scatterer distribution of the polarization multiplexed lens. Lattice periodicity is 650 nm, radii were limited to range from 40 nm to 292.5 nm for the a and b axes, and 0 to 357.5nm for the c axis. For the initial condition, all of the semi-major axis radii were set to 250nm, and the rotations were set to 0 radians. In the final parameter distribution, the scatterers look very similar, and indeed, the minimum semi-major axis radius in the design is $\sim$ 205nm and the maximum is $\sim$ 289nm. **B-D** Are field distributions correspond to x-polarized light, and **E-G** correspond to y-polarized light. **B,E** are scattered field slices in the x-z plane at $y = 0\mu m$. **C,F** are x-y profiles at each focal spot. **C** is a slice at $z = 20\mu m$, and **E** is a slice at $z = 30\mu m$. **D,G** are Gaussian fits at each focal spot.

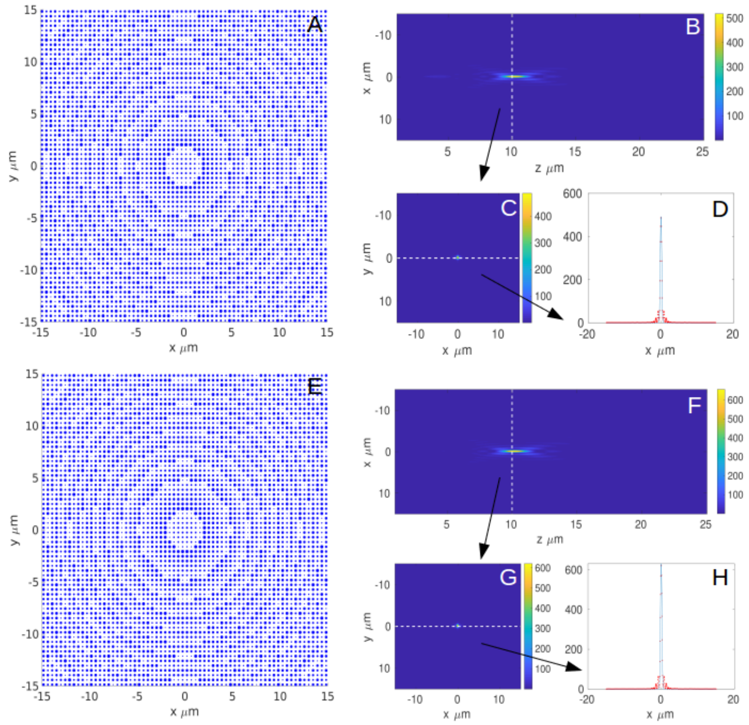


Figure 2.7: Figs. **A-D** correspond to the forward designed lens, and Figs. **E-H** to the optimzied lens. **A,E** are the scatterer distributions. **E,F** are the x-z slices of the resulting field profile at $y = 0\mu m$. **C,G** correspond to the x-y field slice at $z = 10\mu m$. **D,H** are the Gaussians fitted to the field profiles at their focal spot with $y = 0\mu m$. The forward design lens efficiency was determined to be 25.59%, and the optimzied efficiency was calculted to be 32.00%

believe that our inverse design method will be more suitable for optimization type of problem, where the initial conditions are developed based on intuition and prior knowledge.

2.4 Discussion

We demonstrated a new optimization method for designing large area dielectric metasurfaces made of ellipsoidal scatterers based on the adjoint method and a generalization of GMMT. Starting from an array of identical ellipsoidal scatterers, we designed a high NA ($\sim$0.83) lens and a polarization multiplexed lens that focuses light at $30\mu m$ and $20\mu m$ based on the polarization of incident light. Although polarization was the parameter that we chose to optimize over, the same approach can be used to design angle or wavelength multiplexed devices. Furthermore, by modifying the FOM to finely sample the wavelength or angle of the incident light, broad-band and broad-angle devices can be designed at the expense of simulation time. We have also shown that starting with a forward-designed lens as an initial condition, a higher efficiency design can be obtained via optimization.

We note that all the reported devices were designed at refractive index n=3.56. As our method requires the bounding spheres of ellipsoidal scatterers not to overlap other ellipsoidal scatterers, we are limited by the aspect ratio and density of the ellipsoids, and only with high index ellipsoids we can maintain low density of scatterers while spanning the whole 0 to 2π phase. Unfortunately, there is currently no straightforward way to fabricate these structures with such a high index. One solution could be to use a high index resin in additive manufacturing [59]. It is also possible to fabricate cylindrical scatterers at high refractive indices by using traditional lithography. This would require a further generalization of the T-Matrix method to expand the incident and scattered fields in terms of spheroidal wavefunctions instead of SVWF or by using the plane wave coupling method [92].

Chapter 3

DATA-DRIVEN ACCELERATION OF MAXWELL'S EQUATIONS FOR INVERSE DESIGN OF DIELECTRIC METAMATERIALS

The contents of this chapter are adapted from [125], *Maksym V. Zhelyeznyakov, Steven Brunton, and Arka Majumdar*. Deep Learning to Accelerate Scatterer-to-Field Mapping for Inverse Design of Dielectric Metasurfaces Maksym V. Zhelyeznyakov, Steve Brunton, and Arka Majumdar ACS Photonics 2021 8 (2), 481-488 DOI: 10.1021/acsphotonics.0c01468, with permission from the authors.

3.1 Introduction

The objective of this chapter is to describe a forward simulation method for inverse design that is faster than grid based methods [80, 14, 61, 42, 74, 83, 82, 72], is not restricted to spheroidal particles like the method described in the previous chapter, and is more accurate than methods relying on LPA [46, 66]. We leveraged several data-driven modeling and machine learning techniques [17], which are being adopted in the field of optics and photonics [131, 105], with examples in fiber lasers [37, 18, 7, 6, 106, 13, 88] and metamaterial antennas [50].

The electromagnetic (EM) response $\vec{\mathcal{E}}$ to a incident current $\vec{J}$ is given by Maxwell's equation:

$$\nabla \times \nabla \times \vec{\mathcal{E}}(\mathbf{x}) - \omega^2 \epsilon(\mathbf{x})\mu(\mathbf{x})\vec{\mathcal{E}}(\mathbf{x}) + i\omega\mu(\mathbf{x})\vec{J}(\mathbf{x}) = 0 \tag{3.1}$$

where ω is the angular frequency of the current source, $\epsilon(\mathbf{x})$ is the dielectric permittivity distribution, $\mu(\mathbf{x})$ is the magnetic permeability distribution (assumed to be unity here as we will primarily work with dielectric non-magnetic materials) and the vector $\mathbf{x}$ is the position vector. This implies that the field response $\vec{\mathcal{E}}(\mathbf{x})$ only depends on the distribution of $\epsilon(\mathbf{x})$. A forward simulation of Maxwell's equation thus entails the prediction of the spatial EM modes as a function of the scatterer geometry and position. Here, we first use high-fidelity EM simulations to generate data, which are then used to find a simple mapping between $\epsilon(\mathbf{x})$ and $\vec{\mathcal{E}}(\mathbf{x})$ exploiting the singular value decomposition (SVD) and neural networks [19]. We note that, a number of previous works used neural networks to predict the spectral responses from metallic [71, 58, 69] and dielectric [81, 54, 87, 64, 38, 61, 5] scatterers of various geometries. In these problems, the unit cells are identical, and hence there is no need to invoke LPA. However, for imaging applications, where the unit cells are spatially varying, invoking LPA results in inaccuracy. Our work aims to mitigate this challenge by using a data-driven framework to predict the spatial responses from dielectric circular cylinders, while including the effects of their nearest neighbors. Another recent work has applied data driven techniques to accelerate iterative finite difference frequency domain (FDFD) solvers [95]. While accurate, this method is however still memory intensive. Our work provides an alternative, interpolative method for simulating field responses from electromagnetic scatterers by fitting a differentiable model that maps the geometry of the scatterer and its closest neighbors to its EM field response. This model speeds up our forward simulation by estimating local patches of the EM field from the radius of a cylindrical scatterer and its surrounding neighbors. We found that this method can simulate a mesh with 1.2 million discrete points 10^4 times faster than conventional grid-based solvers, and is memory inexpensive enough that it can be run on a laptop. We use this framework to inverse design two devices, both of which are unintuitive under the forward methodology: a multi-wavelength metasurface that produces an annulus beam for one wavelength, and focuses light at a different wavelength, as well as an extended depth of focus lens.

3.2 Methods

The goal of this work is to develop a fast and accurate proxy for the forward simulation that is differential and may be used for inverse design. Fig. 3.1 shows the schematic of our

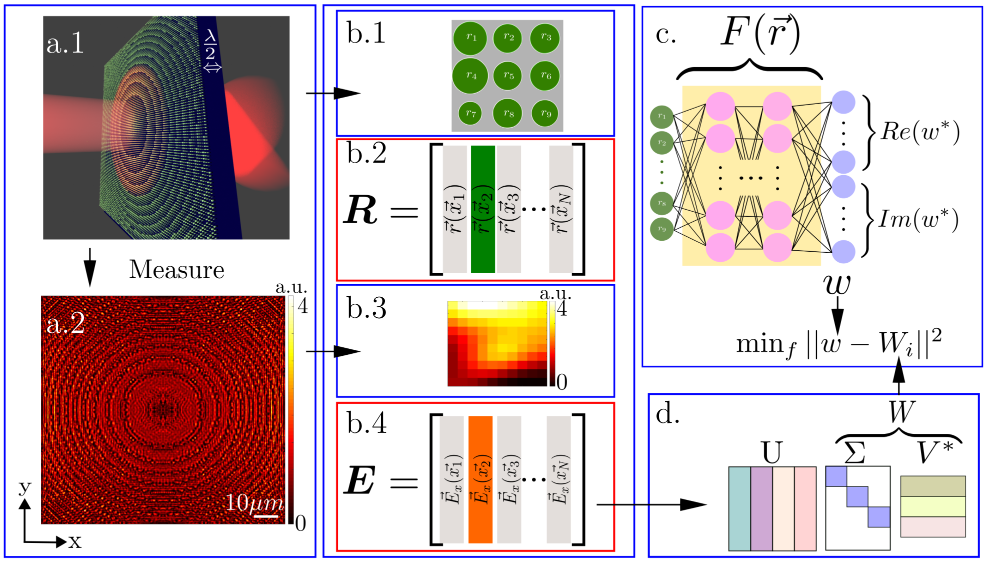


Figure 3.1: Overview of method. **a.1.** Sample forward designed metasurface. **a.2.** Near-field response of metasurface for $\lambda = 633$ nm **b.1-4** Parsing the data. **b.1.** Iterate through each pillar except the ones in the edges, and gather the surrounding pillar radii. **b.2.** Pillar radii and recorded and stacked into matrix $\mathbf{R}$. **b.3.** Field response in a square region with dimension of the pitch p corresponding to the central pillar. **b.4** Electric fields are vectorized and stacked into a matrix $\mathbf{E}$. **c.** We create a neural network that predicts a vector $\mathbf{w}$ corresponding to the column of matrix **d.** $\mathbf{W}$ is constructed as the product $\mathbf{\Sigma V}^*$, where $\mathbf{\Sigma}$ and $\mathbf{V}$ are taken from the SVD of $\mathbf{E}$.

strategy to build a differentiable map $G : \mathbb{R}^9 \to \mathbb{C}^{100}$ that predicts the electric field over a square area with dimension p from the dielectric permittivity distribution $\epsilon(\mathbf{x})$, modeled as 9 cylinders. Here p is the periodicity of the metasurface, and each square area (unit cell) has been discretized into a 10×10 grid. The corresponding field being predicted is flattened into a 100×1 vector. We will explore two models: a low-dimensional linear regression model based on the singular value decomposition (SVD) and a deep neural network model to fit G.

3.2.1 Training data

To train these models, we first generated a data set consisting of forward simulations of several physical devices, in our case meta-lenses. These lenses were designed via forward design. The intuition is that the lens is arguably the simplest physical device, and will likely provide a useful basis to interpolate future devices. We forward designed 10 lenses of diameter $\sim 50\mu m$ with focal lengths varying from $10 - 100\, \mu m$ [21]. The lens design parameters are summarized in Table 3.1. All lenses are intended to function with a current source wavelength $\lambda = 0.633\, \mu m$. The material refractive index was set to $n = 2$, corresponding to silicon nitride, our material of choice for visible wavelength operation [26]. These dimensions correspond to exactly 113 pillars on each axis of the metasurface. All the scatterers were computed with RCWA package S4 [66]. A sample lens of focal length $F = 50\, \mu m$ is shown in Fig.3.1.a.1. We simulated the EM response of each lens using an x-polarized plane wave ($\lambda = 0.633\, \mu m$) with the field monitor $\lambda/2$ away from the scatterers using Lumerical finite difference time domain (FDTD) software. An example field is shown in Fig.3.1.a.2. Only the $\mathcal{E}_x$ component was recorded due to minimal contributions to the total field power from other vector components, which is a result of the circular symmetry of the scatterers. However, the process could easily be generalized to predict the entire vector-field. The resolution of the simulation was chosen to be $0.04431\, \mu m^3/vox$ in order to balance computational time, memory requirements, and accuracy. This results in (10×10) field points in each square unit cell with dimension $p = 443nm$ corresponding to each pillar. Once all of the field data were gathered, we constructed two matrices: $\mathbf{R} \in \mathbb{R}^{9\times N}$ for the radii, and $\mathbf{E} \in \mathbb{C}^{100\times N}$ for

Table 3.1: Parameters used to forward design the training data-set. F: focal length; h: height of the pillars; n: material index; λ: current source wavelength. Lens diameter D is chosen to be the closest integer multiple of the periodicity p.

Parameter	F	D	p	h	n	λ
Value	$10 - 100\,\mu m$	$50.0703\,\mu m$	$0.4431\,\mu m$	$0.633\,\mu m$	2	$0.633\,\mu m$

the electric fields, with N being the number of scatterers. The matrix $\mathbf{R}$ was created by iterating over pillar location $\mathbf{x_i}$, and storing the radii of the pillar and its 8 nearest neighbors as a column vector, shown in Figs. 3.1 b.1 and 3.1.b.2. The pillars on the edges of the metasurface do not have neighbors and were neglected. Similarly, the matrix $\mathbf{E}$ was created by iterating over each pillar location, extracting the field in the unit cell with centroid $\mathbf{x_i}$, and storing it as a flattened 100×1 column vector, shown in Figs 3.1.b.3 and 3.1.b.4. We note that in this work, we consider a scalar field, whose polarization axis is the same as the the incident polarization. This results in two matrices having $N = (113-2)^2 \times 10 = 123210$ columns.

3.2.2 Linear regression model

We first explore the low-dimensional structure of the matrix $\mathbf{E}$, which will facilitate learning a map between the columns of $\mathbf{R}$ and $\mathbf{E}$. Patterns in the rows and columns of $\mathbf{E} \in \mathbb{C}^{M\times N}$ are extracted via the singular value decomposition (SVD) [19]:

$$\mathbf{E} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* \tag{3.2}$$

where $\mathbf{U} \in \mathbb{C}^{M\times M}$ and $\mathbf{V} \in \mathbb{C}^{N\times N}$ are unitary matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{M\times N}$ is a diagonal matrix, with non-negative entries on the diagonal and zeros off the diagonal. The columns of $\mathbf{U}$ can be thought of as a set of orthonormal basis vectors with which to represent the columns

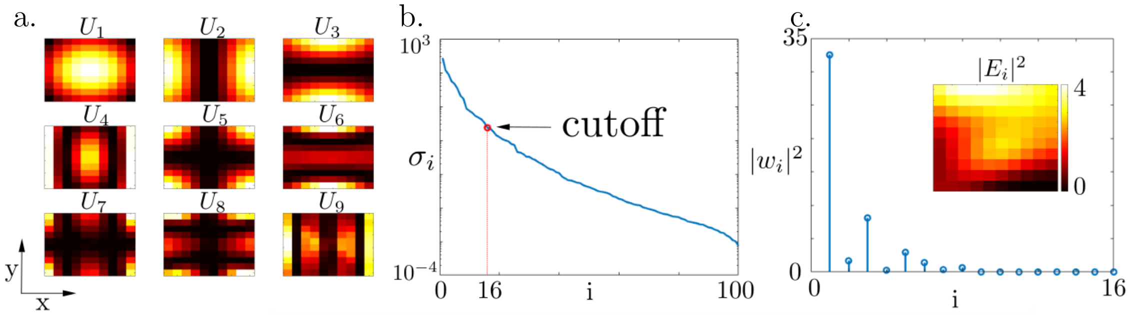


Figure 3.2: Singular value decomposition of simulated data. **a.** First 9 left hand singular vectors **U** of **E** matrix. **b.** Singular value decay of the diagonal matrix $\mathbf{\Sigma}$. Red circle represents the cut off order we used to reconstruct the electric fields. The order 16 cutoff was chosen because it captures 99% total energy of the electric field. Any further contribution from modes with order q > 16, contributes to less than 1% to the total energy in the field. **c.** Plot of the absolute values squared of a random vector $(\mathbf{\Sigma V^*})_\mathbf{i} = \mathbf{w_i}$ that reconstructs some random $p \times p$ field. $\mathbf{w_i}$ represents the weights of the left hand singular vectors **U**.

of $\mathbf{E}$. These columns of $\mathbf{U}$ are arranged hierarchically in terms of how much variance they capture in $\mathbf{E}$, as quantified by the corresponding diagonal element of $\mathbf{\Sigma}$. Fig.3.2 a. shows the square of the absolute value of the first 9 column vectors $\mathbf{u_j}$, reshaped from $\mathbb{C}^{100\times 1}$ to $\mathbb{C}^{10\times 10}$. Definite patterns are observed in these vectors, implying a low-dimensional representation of our data. The rows of $\mathbf{V}^*$ correspondingly provide a hierarchical basis for the rows of $\mathbf{E}$. Each column of the matrix $\mathbf{\Sigma V}^*$ determines the exact combination of the columns of $\mathbf{U}$ required to reproduce the corresponding column of $\mathbf{E}$. Guided by the SVD, it is possible to write an approximate matrix $\tilde{\mathbf{E}}$ as:

$$\tilde{\mathbf{E}} = \mathbf{U_q W} \tag{3.3}$$

where $\mathbf{W} = \mathbf{\Sigma_q V_q^*}$, and the subscript $q < M$ is the truncation order of the matrix approximation. The first q columns of $\mathbf{U}$ are arranged to form U_q, the first $q \times q$ sub-block of $\mathbf{\Sigma}$ is extracted to form $\mathbf{\Sigma_q}$ and the first q rows of $\mathbf{V}^*$ are taken to form $\mathbf{V_q^*}$. It can be shown that $\tilde{\mathbf{E}}$ is the best rank$-q$ approximation to the matrix $\mathbf{E}$, in the Frobenius norm [94]. We choose a truncation value of $q = 16$, shown as the red circle in Fig.3.2.b., as the rank 16 approximation $\tilde{\mathbf{E}}$ captures 99% of the variance in the matrix $\mathbf{E}$.

We will now construct a regression map to estimate columns of $\mathbf{E}$ from columns of $\mathbf{R}$. Specifically, we estimate the matrix $\mathbf{W}$, which will be used to reconstruct $\tilde{\mathbf{E}}$. Instead of using the columns of $\mathbf{R}$ directly as features, we will create an augmented feature vector comprised of monomials constructed from the radii. This feature matrix $\mathbf{\Theta}$ is constructed by vertically concatenating integer Hadamard powers of $\mathbf{R}$:

$$\mathbf{\Theta} = \begin{bmatrix} 1 \\ \mathbf{R} \\ \mathbf{R^{\circ 2}} \\ \vdots \\ \mathbf{R^{\circ m}} \end{bmatrix}. \tag{3.4}$$

Where $\mathbf{R}^{\circ m}$ are the element-wise powers of $\mathbf{R}$:

$$R^{\circ m} = \begin{bmatrix} r_{1,1}^{m} & \dots & r_{1,N}^{m} \\ \vdots & \ddots & \vdots \\ r_{M,1}^{m} & \dots & r_{M,N}^{m} \end{bmatrix} \tag{3.5}$$

Thus, we set up a linear system:

$$\mathbf{W} = \mathbf{\Xi}\mathbf{\Theta} \tag{3.6}$$

and solve for $\mathbf{\Xi}$:

$$\mathbf{\Xi} \approx \mathbf{W}\mathbf{\Theta}^{\dagger} \tag{3.7}$$

where the superscript † denotes the Moore-Penrose pseudo-inverse [94]. The matrix $\tilde{\mathbf{E}}$ can then be approximated by a *generalized* linear regression problem:

$$\tilde{\mathbf{E}} \approx \mathbf{E}_{\text{pred}} = \mathbf{\Xi}\mathbf{\Theta} \tag{3.8}$$

We varied the number of features used to train this linear model by changing the number of powers m used to construct $\mathbf{\Theta}$. To train this linear model, we used 80% of the data. After creating $\mathbf{W}$ and $\mathbf{\Theta}$, we extracted a random set of 98568 columns from each matrix in order to fit the matrix $\mathbf{\Xi}$. We used the other 24642 columns for validation. Fig. 3.3a depicts the qualitative performance of our linear model for $m = 10$. Column **I** represents a randomly chosen vector $\mathbf{E_i}$ at some position $\mathbf{x_i}$, column **II** is the corresponding predicted vector $\mathbf{e}_{\text{pred}} \in \mathbf{E}_{\text{pred}}$, and column **III** is the absolute difference squared $|\epsilon|^2 = |\mathbf{E_i} - \mathbf{e}_{\text{pred}}|^2$. Note that we are comparing complex numbers, while plotting their intensities. Thus, although the intensities may not look extremely similar, their errors can be relatively small. All values were normalized to have a maximum absolute value of 1 and the same colorbar across all figures. Fig. 3.3b shows the probability density functions (PDFs) of the error distributions $|\mathbf{E}_{\text{pred}} - \mathbf{E}|^2$ as we increase m from 1 to 10. As the number of features increases, the PDF becomes tighter. Fig. 3.3c is a plot of the relative error defined as:

$$\frac{||\mathbf{E}_{\text{pred}} - \mathbf{E}_{\text{test}}||_F^2}{||\mathbf{E}_{\text{test}}||_F^2} \tag{3.9}$$

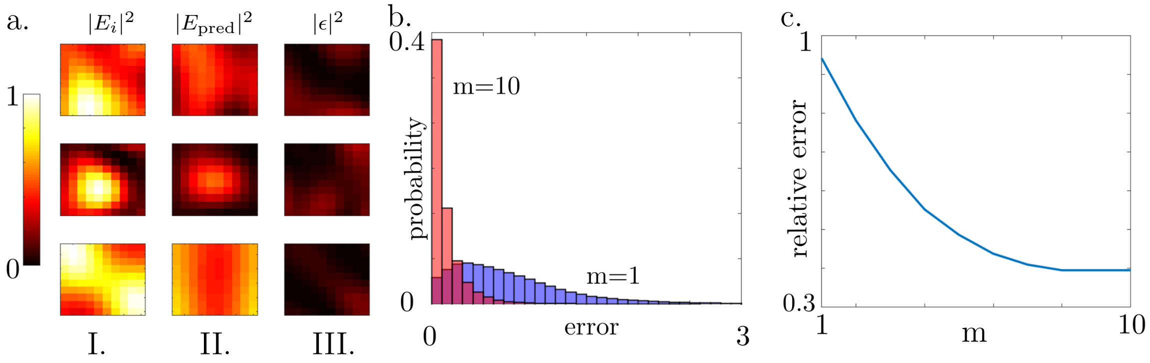


Figure 3.3: **a.** Column **I.** represents the field simulated by FDTD, column **II.** is the electric field predicted using the linear model, and Column **III.** is the difference between the two. Each row represents the field corresponding to the same set of 9 radii. **b.** Probability density functions of relative errors between the predicted matrix $\mathbf{E}_{\text{pred}}$ and the true matrix $\mathbf{E}$. The blue plot corresponds to a feature matrix with only $m = 1$, and the red plot represents the feature matrix constructed with powers up to 10. **c.** Plot of the relative errors in the Frobenius norm between $\mathbf{E}_{\text{pred}}$ and $\mathbf{E}$. The x axis represents the power term in the radius features.

as a function of m. Both plots show the error between our model and the FDTD simulation decreasing as the number of features in each matrix increases, so the model converges to the actual physics of the system. The final relative error for the linear model converged to ≈ 0.395. As a side note, we have attempted using monomial expansions up to order 2 of column vectors of $\mathbf{R}$ as input features for our model, by using powers of column vectors of $\mathbf{R}$ and cross terms in between radii, but found no significant improvements in relative error when fitting the model.

3.2.3 Neural network model

To improve on the generalized linear model, we construct a deep neural network (DNN), shown in Fig. 3.4. We hypothesize that the DNN would learn a non-linear transformation of the input features that better capture the physics of the system. The model was trained by using 80% of the data set, while keeping 20% for validation. The architecture was implemented in TensorFlow [2] and optimized using the Adam optimizer [55]. The DNN architecture consists of 11 fully connected layers, each followed by a ReLU activation function. The first layer of the network is the input layer with 9 neurons corresponding to each radius. The second layer has 100 neurons, which was doubled with each subsequent layer until 1600, and then cut in half until the second to last layer again had 100 neurons. The final layer had 32 neurons, with the first 16 elements corresponding to the real components of the vector $\mathbf{w}$ and the last 16 components corresponding to the imaginary components of $\mathbf{w_i}$. The outputs were arranged in this manner due to TensorFlow's limitations when designing complex-valued neural networks. The objective function used was a mean squared error between the output vector, and the corresponding vector from $\mathbf{W}$, shown in detail in Figs. 3.1c. and 3.1d. The network was trained until the mean squared error of the verification data set stopped being minimized in order to avoid over-fitting. Once the network was trained, we computed the electric field response of the training data by feeding the test data set into the neural network to compute $\mathbf{W}_{\text{pred}}$, and Eq. (3.3) to compute $\mathbf{E}_{\text{pred}}$. The quality of our prediction can again be summarized by the relative error between $\mathbf{E}_{\text{pred}}$ and

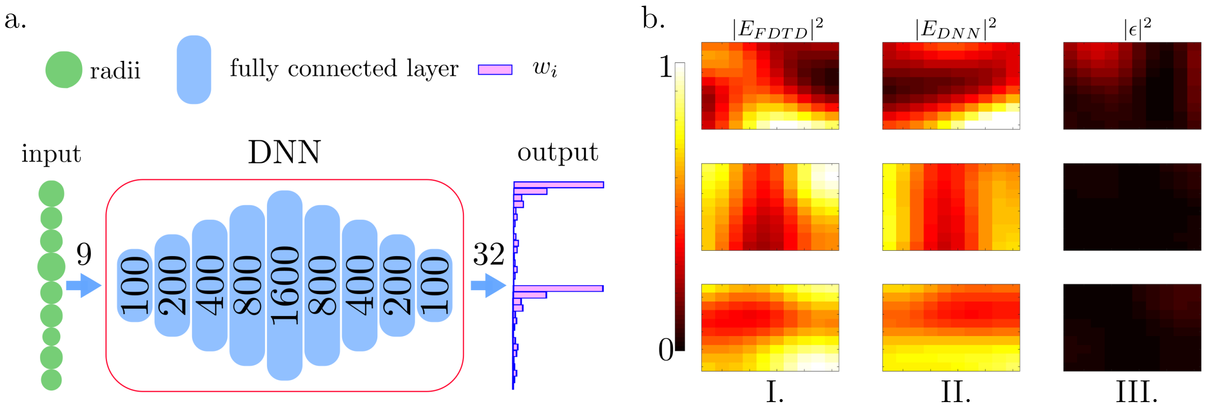


Figure 3.4: **a.** The input into the DNN is 9 radii. The DNN architecture consists of 9 fully connected layers. The first layer starts off with 100 neurons, and each subsequent layer doubles the number of neurons until 1600, then number of neurons per layer is halved until the final layer has 100 neurons. All layers are followed by a ReLU activation function. The output has 32 elements. **b.** The performance of the DNN model. Column **I** is the field simulated by FDTD. **II.** is the field reconstructed by Eq. (3.3) from the predicted vector $\mathbf{w_i}$. **III.** Difference between fields.

$\mathbf{E}_{\text{test}}$ in the Frobenius norm, given in Eq. (3.9), which was computed to be ≈ 0.26. The same metric calculated by using a predicted field from the local phase approximation gives a relative error of ≈ 1.35. We use the DNN model to inverse design our devices. As a side note, appendix B.2 includes a convergence study between number of nearest neighbors included in the neural network model, and the accuracy of the predicted nearfield. The relative error of our test data sets turned out to be lowest when using only nearest neighbors.

3.3 Results

To test the utility of our model, we inverse designed two meta-optical devices. Motivated by stimulated emission depletion (STED) microscopy [98], the first device we inverse designed was a wavelength multiplexed lens that focuses light with $\lambda = 633$ nm, and creates a annulus beam at the focal plan for $\lambda = 400$ nm. The second is an extended depth of focus (EDOF) lens that focuses light over $100 - 350\,\mu m$ along the optical axis. The optimization process was implemented in TensorFlow [2]. We used the DNN model and Eq. (3.3) to predict the nearfields of the designed devices. The farfields were then calculated by using the angular propagation method [39]. The gradients with respect to radii were calculated by using TensorFlow's auto-differentiation, and updated by the adam optimizer. To design the multi-wavelength lens, we had to predict nearfields for two wavelength. Hence, we repeated the procedure outlined in section 2 to create one more data driven model to predict the field response for a $\lambda = 400$ nm current source. This model was trained on the same dataset of metasurface lenses designed to focus light for the 633 nm wavelength, however the electric field responses were gathered from FDTD simulations at 400 nm. Once trained, the relative error defined by Eq. (3.9) computed on the test dataset for $\lambda = 400$ nm was ≈ 0.37. The difference between the two wavelengths can be explained by the relatively non-smooth transmission of $\lambda = 400$ nm E-fields over this range of radii when compared to the $\lambda = 633$

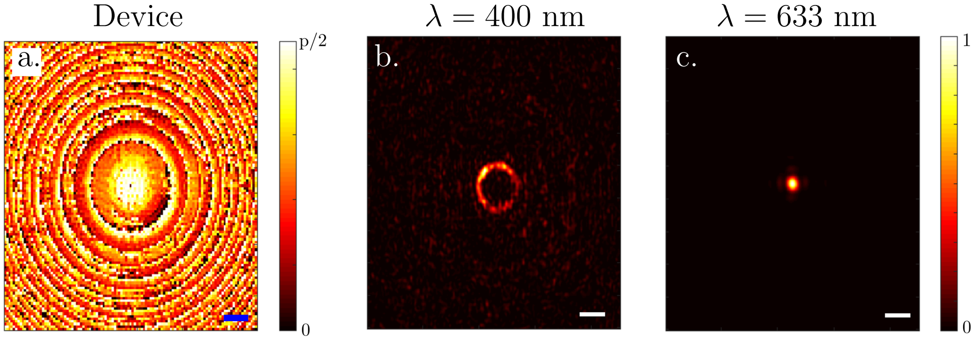


Figure 3.5: **a.** Optimized multi-functional device. Scale bar is $5\mu m$ **b.** FDTD result for $\lambda = 0.4\,\mu m$. **c.** FDTD result for $\lambda = 0.633\,\mu m$. Scale bars are $2\,\mu m$. Units are normalized so the maximum intensity is equal to 1.

nm case. To optimize the lens, we defined two figures of merit for each wavelength as:

$$\mathrm{FOM}_{400} = -\sum_{m=0}^{19} I\left(c\cos\left(m\frac{2\pi}{20}\right), c\sin\left(m\frac{2\pi}{20}\right), 50\,\mu m\right) \tag{3.10}$$

$$\mathrm{FOM}_{633} = -20 \times I(0, 0, 50\,\mu m), \tag{3.11}$$

Where the function $I(x, y, z)$ is the intensity of the electric field at (x, y, z) coordinate given by $I(x, y, z) = \mathcal{E}^*(x, y, z)\mathcal{E}(x, y, z)$. The constant $c = 1.5\,\mu m$ corresponding to the radius of the annular beam at the focal spot. The tuple $(c\cos\big(i\frac{2\pi}{20}\big), c\sin\big(i\frac{2\pi}{20}\big))$ is the parametrization of a circle in the $x - y$ plane, that we discretized over 20 points on the circle. The factor of 20 on FOM_{633} is chosen as a normalization factor to ensure the integral of the intensity over the annulus is the same as the intensity at focal spot. The quantity optimized was then:

$$\max(\mathrm{FOM}_{400}, \mathrm{FOM}_{633}) \tag{3.12}$$

with respect to the radii distribution. We set our initial radius distribution to be the same as the forward designed lens for $\lambda = 633$ nm and $f = 50\,\mu m$. The designed device is shown in Fig. 3.5a. To verify the design, we computed the nearfield response of the radii distribution in Lumerical FDTD and propagated the nearfield to the focal plane using angular spectrum method. Figs. 3.5b and 3.5c show the meta-optic's response to 400 nm and 633 nm wavelength at the focal plane respectively. The efficiency η of the metasurface was calculated to be 26.82% for $\lambda = 400nm$. The formal definition is given in appendix B.1. We quantify the annulusal functionality of the metasurface as the ratio between the power confined in the annulus to the power confined in the center of the annulus. This ratio $\eta_\circ$ was calculated to be 58.47 for $\lambda = 633nm$, formally defined in appendix B.1. The EDOF lens was designed by using a lens with $f = 100\,\mu m$ as a starting condition. Our intent was to design an EDOF lens to focus from $50\,\mu m$ to $100\,\mu m$. We defined the figure of merit for the EDOF as:

$$\mathrm{FOM}_{\mathrm{EDOF}} = -\sum_{m=0}^{10} \log(I(0, 0, 50 + m \times dz)) \tag{3.13}$$

where $dz = 10\,\mu m$. Thus we aim to maximize the intensity at the center of 10 equi-spaced $x - y$ planes. The resulting device is shown in Fig. 3.6a. The intent behind the logarithmic

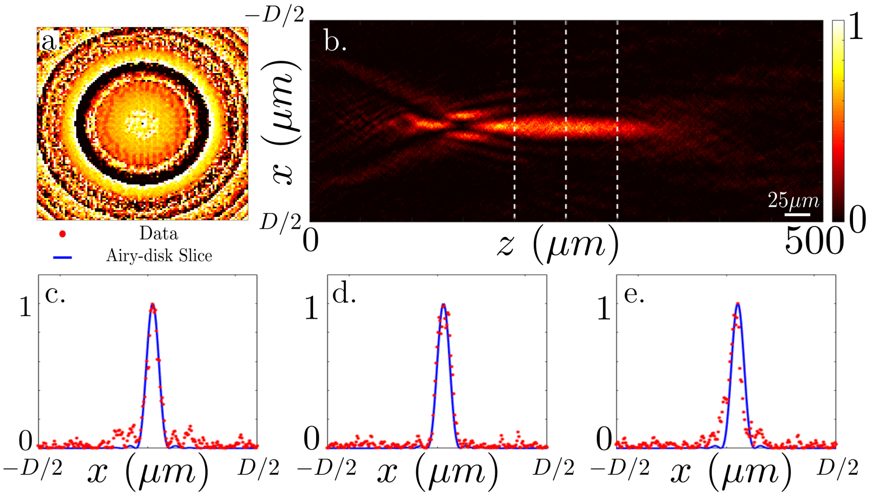


Figure 3.6: EDOF lens. **a.** Device. **b.** Simulated field. Simulation is a result of FDTD and angular spectrum propagation. **c-e** Slices of field in **b.** along the dashed lines corresponding to $200\,\mu m$, $250\,\mu m$ and $300\,\mu m$ respectively. The red dots are the simulated data. The blue lines are the Airy disk profiles corresponding to the diffraction limit. All intensities are normalized by the corresponding maximum intensity.

sum was to equalize the importance of each term along the z-axis. Without the logarithm term, the optimization would prioritize a single focal spot, since such a device would minimize the figure of merit. After optimization, this figure of merit converged to a lens focusing from $100\,\mu m$ to $350\,\mu m$ as shown in Fig. 3.6b, corresponding to a numerical aperture varying from 0.07-0.25. We attribute the longer depth of focus than what was intended to the physical nature of wave propagation. Figs. 3.6c-3.6e show slices of the electric field at $200\,\mu m$, $250\,\mu m$ and $300\,\mu m$ along the optical axis. The red dots correspond the the simulated data, and the blue line corresponds to the Airy disk corresponding to the diffraction limited focal spot. We note that clearly, there are additional side-lobes in the EDOF design, and thus the total energy in the main lobe suffers. However, a different figure of merit can be designed to reduce the side-lobes, depending on the desired application.

3.4 Discussion

This chapter outlines a data driven methodology for forward simulation of Maxwell's equations to design optical metasurfaces. Our model does not make the local phase approximation, and thus the inter-scatterer coupling is well accounted for. While, the model is not as accurate as a complete full-wave simulation, it is significantly faster. A single forward simulation of a square area of dimensions $50\mu m \times 50\mu m$ at 44.31 nm resolution takes approximately 12 seconds with our method versus approximately 3.1 hours using Lumerical FDTD software in the same computer. FDTD also requires a 58.95 GB initialization mesh and 29.6GB of RAM for the same simulation, while our method only requires 3.75GB for the same problem, and can be run on a mid-range laptop. It takes approximately 16 hours to gather the data required for training our neural network model (10 FDTD simulations). Depending on the design problem, and based on our results from optimization using our DNN models as a forward simulator, it takes approximately 100 iterations for an inverse design problem to converge. Under these assumptions, we can estimate that we need about 100 forward simulations to do naive gradient based design, which would take 310 hours to complete by using a full FDTD simulation. Furthermore, adjoint optimization requires 2 simulation passes per optimization step, which will make the whole optimization process to take least 620 hours. Even when considering the time it takes to gather the data, and train our model, the overall design process is sped up significantly. The angular propagation step that transforms the nearfields to the farfields adds an additional 6GB of memory and 26 seconds of optimization time to the optimizaton per farfield plane used. For the EDOF lens, this results in an additional 60GB of memory required during the optimization process. We note using a different propagation method such as the Rayleigh-Sommerfield method that does not require storing the full farfield, but only the field at the point where the FOM is calculated would significantly reduce the memory requirements for the propagation.

It is worth noting that our method is inherently interpolative, and thus is only as accurate as the data that we feed into it. Therefore the current model is limited to predicting fields from lens-like devices, under a specific refractive index, constrained to a subspace of possible geometries. If one wanted to design a metasurface using this method for a different refractive index, or with different scatterer geometries, the model would need to be retrained. One way we could improve this model is by using additional data to train it. In our future work, we hope to improve the accuracy of this model by simulating random arrangements of scatterers and using this as our training data set in addition to the data set from lenses. We also emphasize that the reported efficiency of the designed lenses is low, which remains a challenge for low index materials [16]. However, full wave simulations have reported efficiency increase of the metasurface lenses especially when all the coupling between scatterers are exactly accounted for [72, 25]. Our model could be improved by better accounting for the coupling between scatterers using more data, especially EM field responses from scatterers with rapidly varying geometries, since the scatterer geometries of lenses vary slowly in space. One specific direction will be to capture the physics to predict the full vectorial field, in contrast to the scalar field modeled here. Modelling the second nearest neighboring scatterers could also be an interesting path forward. Utilizing techniques such as transfer learning [130] , we could utilize the features learned from our previous models that include only information from the nearest scatterers, and try to generalize the model for second and even third nearest neighbors. Furthermore, adding additional constraints such as assumptions about energy conservation to the model training process, could further increase the accuracy of the model.

Chapter 4

INVERSE DESIGN USING PHYSICS INFORMED NEURAL NETWORKS

The contents of this chapter are adapted from [126], *Maksym V. Zhelyeznyakov, Johannes E. Fröch, Anna Wirth-Singh, Jaebum Noh,Junsuk Rho, Steven L. Brunton, and Arka Majumdar. Large area optimization of meta-lens via data-free machine learning, 2022, arxiv preprint*, with permission from the authors.

4.1 Introduction

To address the computational bottleneck of large-area inverse design, in this chapter we will introduce a physics-informed neural network (PINN), which can replace a traditional FDTD/FDFD solver to predict the electric field distribution for a given dielectric distribution. We note that a large number of works already used artifical neural networks to predict spectral responses of meta-optics of varying scatterer geometries [71, 58, 81, 54, 87, 64, 38, 61, 5]. However, these works used largely periodic structures for which LPA is accurate. We present a solution via PINNs[84, 67] for lenses and devices with spatially varying scatterer geometries, where it is necessary to model the whole electric field from several scatterers and their neighbors. PINNs solve partial differential equations (PDEs) by minimizing a loss function constructed from the PDE itself. This loss function is generally some norm of the residual [84] or an energy function derived from the PDE [53]. PINNs have already seen wide usage in the field of fluid mechanics [85, 129, 90], biology [111], and solving stochastic PDEs [122]. In electromagnetic inverse problems, PINNs have also been employed to design meta-optics and nanophotonic devices [24, 68]. These works, however, did not demonstrate a simulation speedup, and are limited to the inverse design of very small devices. We also note that pre-trained PINNs have been used to design small gratings[22]; however their methodology is limited to small gratings that deflect light fields to specific angles, and thus cannot be readily used for the inverse design of general meta-optics.

In our work, we train PINNs to predict the electric fields from a parameterized set of dielectric meta-atoms corresponding to rectangular pillars. We then use this as a surrogate model to design cylindrical meta-lenses operating in the visible with a diameter of 1 mm ($\sim 1500\lambda$). Large area meta-optics are simulated by partitioning the simulation region into groups of 11 meta-atoms, with the outermost meta-atoms overlapping. After simulation, the fields are stitched together. Our PINNs do not require a training data set. They are trained by randomly generating distributions of dielectric meta-atoms ϵ, feeding them into a neural network NN, and minimizing the residual of the linear Maxwell PDE operator

$$||A_{\mathrm{Maxwell}}(\epsilon)NN(\epsilon) - b||_1 \tag{4.1}$$

over the neural network training parameters. This means our PINNs are trained without ever invoking a forward numerical simulation of Maxwell's equations during the training process. Numerical simulations are invoked only to test the neural network performance (see next section and appendixC.5). Once trained, this method can calculate the full electromagnetic field response from a 1 mm diameter cylindrical meta-lens at $\sim$ 630nm in approximately 3 seconds on a graphics processing unit (GPU). Furthermore, we demonstrate a theoretical and experimental improvement of the maximum intensity of cylindrical metalenses over their forward designed hyperboloid counterparts, signifying the improvement over using LPA. We emphasize that the size of the meta-lens, on which we demonstrate the intensity enhancement of over 50%, is at least one order of magnitude larger than any other inverse designed lens that does not rely on the LPA. We note that the reported method is robust enough to handle even larger meta-optics, with simulation time scaling only linearly with the aperture of the cylindrical lens (see appendix).

4.2 Deep neural network proxy to Maxwell's equations

Our problem statement is summarized in Fig. 4.1. The monochromatic electromagnetic scattering equation for an inhomogeneous, non-magnetic material is given by:

$$\nabla \times \nabla \times \mathcal{E}(x) - \omega^2 \epsilon(x) \mathcal{E}(x) = i\omega \mathcal{J}(x). \tag{4.2}$$

In the 2D case, assuming out of plane polarization $(0, 0, \mathcal{E}_z)$, and the double curl vector identity, $\nabla \times \nabla \times = \nabla(\nabla \cdot) - \nabla^2$ we can simplify Eq. (4.2) to:

$$\nabla^2 \mathcal{E}_z(x) + \omega^2 \epsilon(x) \mathcal{E}_z(x) = -i\omega \mathcal{J}_z \tag{4.3}$$

where $\mathcal{E}_z$ and $\mathcal{J}_z$ are scalar fields. Equation (4.3) is defined over all space, with boundary conditions at $|x| \to \infty$. To simulate this equation, we discretize it on a Yee grid [113] by replacing the ∇ operator with a matrix, and treating the field $\mathcal{E}_z(x)$ and current $\mathcal{J}_z$ as vectors E and J at discrete values of x. Similarly, we treat the dielectric distribution $\epsilon(x)$ as a diagonal matrix ε. To truncate the simulation to a finite domain, we use perfectly matched boundary layers (PML), by making the transformation on the partial derivative operators $\frac{\partial}{\partial x} \to \frac{1}{1+i\frac{\sigma(x)}{\omega}} \frac{\partial}{\partial x}$. Making these substitutions, Eq. (4.3) becomes:

$$\left[D_x^h D_x^e + D_y^h D_y^e + \omega^2 \varepsilon \right] E = -i\omega J \tag{4.4}$$

with matrices $D_x^h, D_x^e, D_y^h, D_y^e$ being the matrix representations of corresponding derivative operators on a Yee grid with incorporated PML boundaries. See appendix C.4 for a more detailed description of the matrices. These matrices were extracted from a modified version of the package angler [47] with constants c, ϵ, μ set to 1 and the length scale set to μm. To build a neural network proxy to solve Eq. (4.4), we employ a PINN. PINNs generally use the coordinates of the computational grid as the input to the neural network, and then minimize the residual of the physical equations by approximating the target quantity being solved for with a neural network. This approach is slow since it effectively functions as an iterative solver re-parametrized over neural network weights and biases. It also required retraining the neural network for all different dielectric distributions. Our approach is to build a proxy solver that predicts the field E from a dielectric distribution ε. We pre-train the PINN to predict fields from inputs ε before optimizing our meta-lenses. The minimization problem to train the PINN becomes:

$$\min_{\theta} f(\varepsilon;\theta) \quad \text{where} \quad f(\varepsilon;\theta) = \left|\left|\left[D_x^h D_x^e + D_y^h D_y^e + \omega^2 \varepsilon\right] NN(\varepsilon;\theta) + i\omega J\right|\right|_1 \tag{4.5}$$

with $NN(\varepsilon;\theta)$ being the output field from the PINN, and $||\cdot||_1$ is the vector l_1 norm. Here θ refers to the weights and biases of the neural network NN. A lower physics informed loss indicates that the neural network is actually satisfying the PDE, and thus predicting the field more accurately. We re-emphasize that there is no data term in $f(\epsilon;\theta)$, which simplifies the neural network training process. Furthermore, we believe that it mitigates the accumulation of error in the gradients during the inverse design process observed by Chen et. al. [22]. Fig. 4.1 outlines the general strategy for building the proxy model. During each epoch, 10 (batch size) dielectric distributions consisting of rectangular pillars of height $h = 0.6\mu m$ with dielectric constant 4 (corresponding to SiN), are generated from 11 random pillar half-widths per batch. The operation wavelength is $\lambda = 0.633\mu$m. The neural network architecture chosen is a UNET, shown in Fig. 4.1 **a** and 4.1 **b**, due to its relatively good performance with scattering problems[22]. The model is trained for 5×10^5 epochs using the ADAM optimizer [56] with a learning rate set to 5×10^{-4}. The final residual of the fields predicted by the neural network are of the order of ~ 0.5, compared to the numerical residual produced by FDFD which is on the order of 10^{-16}. Although there is a large difference, in the next section we show that this still produces a simulator which is capable of outperforming the LPA when optimizing the efficiency of a metalens. Fig. 4.2 **a** shows an example of a field predicted from a random set of pillars by the neural network, by a 2D FDFD code, and their difference, showing good qualitative match. A more quantitative measure at the errors is shown in Fig. 4.2 **b**, where we show the point-wise error probability density functions for the relative error between the complex fields predicted by FDFD and that predicted by neural network and field predicted under LPA, and the absolute error between pillar-wise

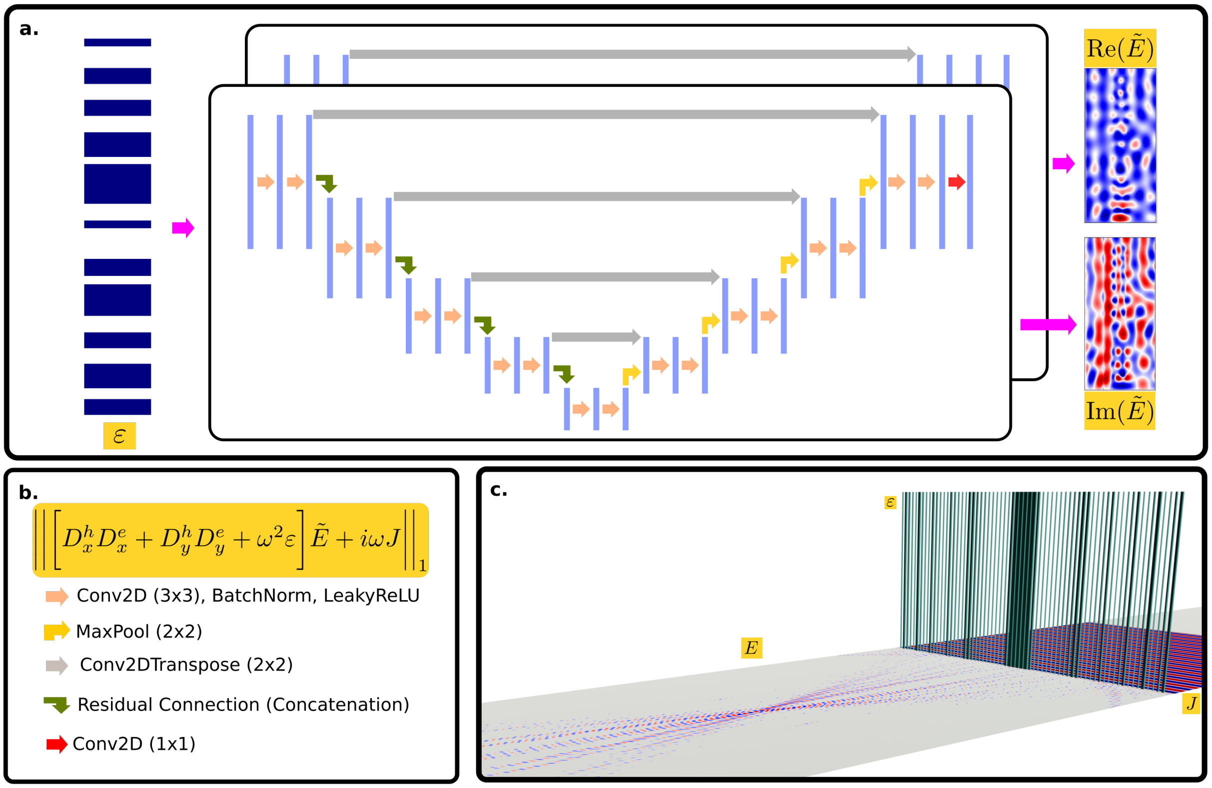


Figure 4.1: **a.** Neural network schematic. ε distributions of 11 pillar meta-optics are meshed by randomly generating sets of pillar half-widths of height $h = 0.6\mu$m with a dielectric constant 4 corresponding to SiN. The background medium is air. The loss function is the $||\cdot||_1$ norm of the residual of Eq. (4.2). **b.** Neural network architecture. Encoder layers are down-sampled by a maxpool operation with a 2×2 kernel. The decoder part of the network is up-sampled by the Conv2DTranspose operation with a 2×2 kernel. **c.** Render of the system under optimization. A current J is incident on a cylindrical metalens with dielectric distribution ε, with output response E.

average transmission coefficients. See supplement C.2 for a more detailed description of the pillar wise transmission coefficient error. The relative error is expressed as:

$$\text{mean}\left(\frac{|E_{\text{approx}} - E_{\text{FDFD}}|^2}{\max(|E_{\text{FDFD}}|^2)}\right). \tag{4.6}$$

For the PINN, E_{approx} is the field predicted from a set of 11 pillars. For the LPA, E_{approx} is fields predicted from the same set of pillars, and then stitched together over the same region. See fig. C.2 for a visual explanation. This error can be interpreted as the % difference between the FDFD predicted field and the approximate field, either predicted by neural network without making any LPA or the field under LPA. The mean expected relative error for the neural network is $\mu = 0.025$ with a standard deviation of $\sigma = 0.0073$. When using the LPA over the same region, we get a mean relative error of $\mu = 0.17$ with a standard deviation of 0.078. Thus, based on the relative field error, our method is 6.8× more accurate than the LPA. For the pillar-wise transmission coefficient error, we get an expected error of $\mu = 0.051$ for the neural network with a standard deviation of $\sigma = 0.033$ and for the LPA method we get an expected error of $\mu = 0.38$ with a standard deviation of 0.14. Thus, based on the transmission coefficient error, our method is 7.2× more accurate than the LPA.

4.3 Device optimization

The optimization process based on automatic differentiation functionality of PyTorch for large area meta-optics is outlined in Fig. 4.3. The forward problem is solved via a pre-trained PINN. Since the input into the neural net is a meshed grid of pillars, a differentiable map from pillar half-widths to meshed geometries must be generated. This is achieved by generating Gaussian functions centered around pillar centers, with standard deviations of pillar half-widths in the x dimension, and pillar height in the y dimension, and then using a modified softmax function to transform the Gaussians into rectangles with slightly rounded edges, making them differentiable via automatic differentiation (see appendix C.1). The meshed structures are fed into two separate neural networks that have been pre-trained to predict the complex electric field. The fields are then stitched together with regions of

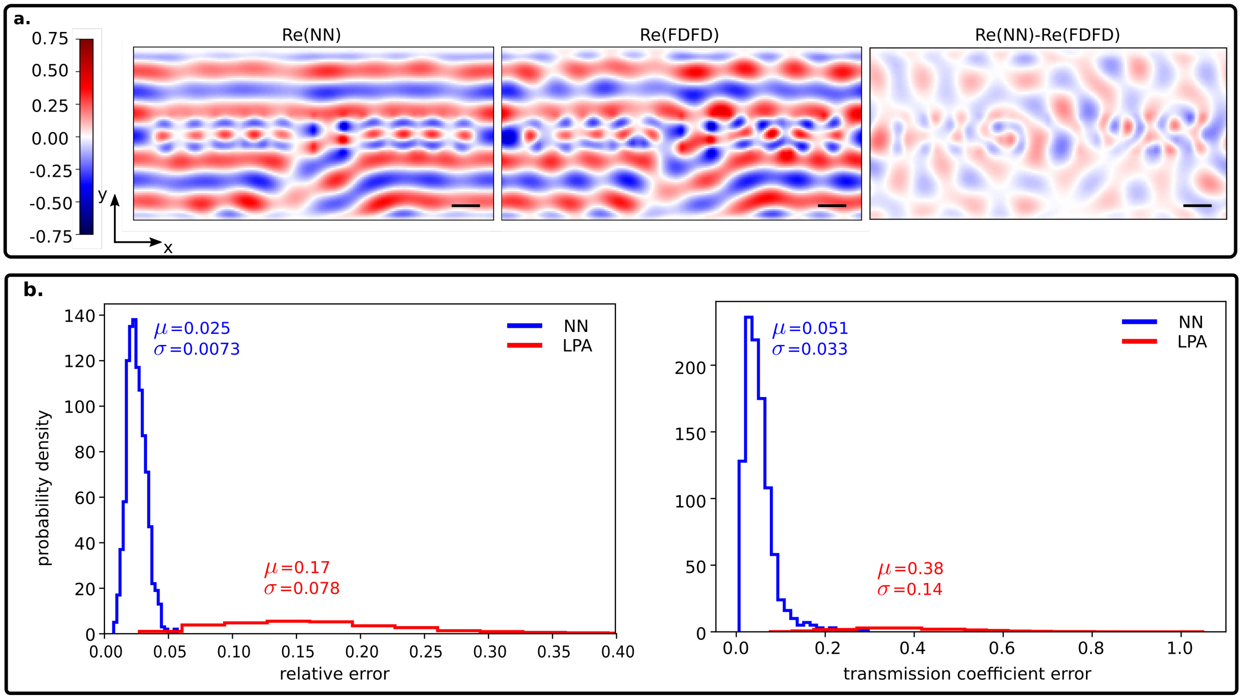


Figure 4.2: **a.** Real part of fields predicted by (left) neural network, (center) FDFD, and (right) the difference between the true and predicted fields. **b.** Comparison between the performance of the proposed neural network and LPA methods. (left) Shows the relative error between FDFD predicted fields and the fields predicted by LPA. (right) Error comparisons between the transmission coefficients predicted by LPA and the neural net.

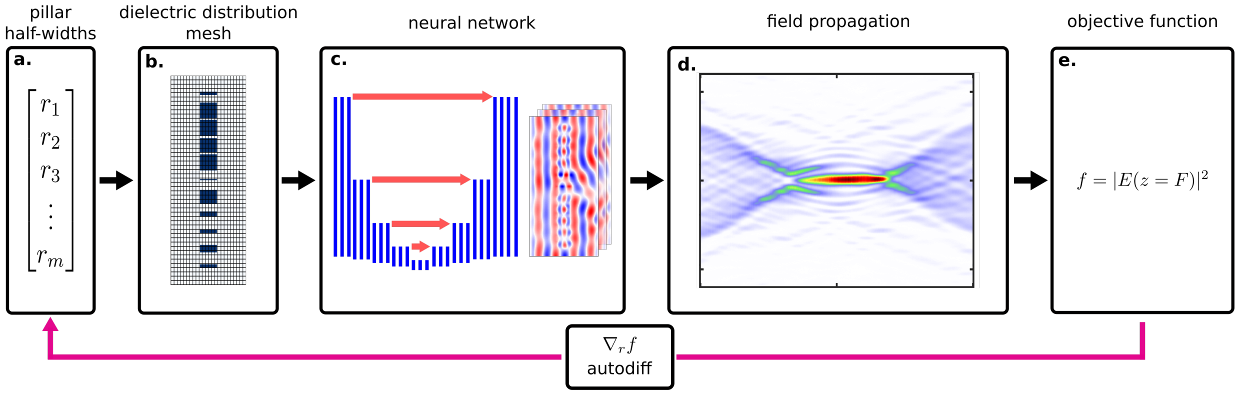


Figure 4.3: Optimization strategy of 2D meta-optics with PINNs. **a.** We start with a vector, which contains a list of all pillar half-widths, characterizing the meta-optic. These half-widths are then batched into groups of 11 with an overlap of 1 pillar on each side. The choice of 11 pillars was made based on the GPU memory required to train the PINN. **b.** The half-widths are meshed into dielectric distributions which get fed into the **c.** The neural network predicts patches of fields which are then stitched together, and **d.** propagated via the angular spectrum method. **e.** The objective function is formed from the resulting field, and backpropagated using PyTorch's automatic differentiation functionality to update the initial radius distribution.

the outer half-widths overlapping. The total field is then propagated using the angular spectrum method. The propagated field is used to calculate the FOM f from Eq. (4.5). We use automatic differentiation to compute the gradients of the FOM with respect to the input half-widths $\nabla_{\vec{r}} f$, and iteratively update them with the ADAM optimizer[56].

4.4 Results

We used PINN surrogate model to optimize 9 different lenses, all with 1 mm aperture, with focal lengths ranging from 250-1500 μm in increments of 250 μm. The minimum feature size is set to 75 nm, to ensure fabricability. To compare our optimization approach, we also generated lenses according to the hyperboloid phase equation:

$$\phi(x, y) = \frac{2\pi}{\lambda}\left(\sqrt{x^2 + F^2} - F\right) \tag{4.7}$$

The phase is implemented under LPA using SiN (refractive index 2), a wavelength of 0.633μm, and periodicity of $p = 0.443\mu$m. We then optimize the lens employing our PINN to increase the intensity at the focal spot, i.e., the FOM is given by:

$$f = \left|E(x = 0, z = F)\right|^2 \tag{4.8}$$

Fig. 4.4**a** and Fig. 4.4**b** show the intensity profile of a forward designed and optimized lens with $F = 500\mu$m focal length. Fig.4.4**c** shows the normalized intensity slice at the focal spot of both lenses. As seen in Fig. 4.4 **d** the maximum intensities at the focal spots improve in every case. Fig. 4.4 **e** shows that the efficiency all improves in all except for the lens with highest NA. We validated our designs by fabricating and experimentally testing the meta-lenses using a microscope (details of fabrication and characterization in Methods). Fig. 4.5 shows an example of the inverse optimized device. Fig. 4.5**a-c** shows the scanning electron micrographs (SEMs) of the fabricated optimized lens with focal length $F = 500\mu$m. Fig.**5d** shows the distribution of the dielectric pillar half-widths of the same forward and optimized lens. signifying the two designs are very different. Fig.4.5**e** shows the focal spot intensities of the lenses integrated over a $r = 3\times$FWHM region at the focal spot, which yields

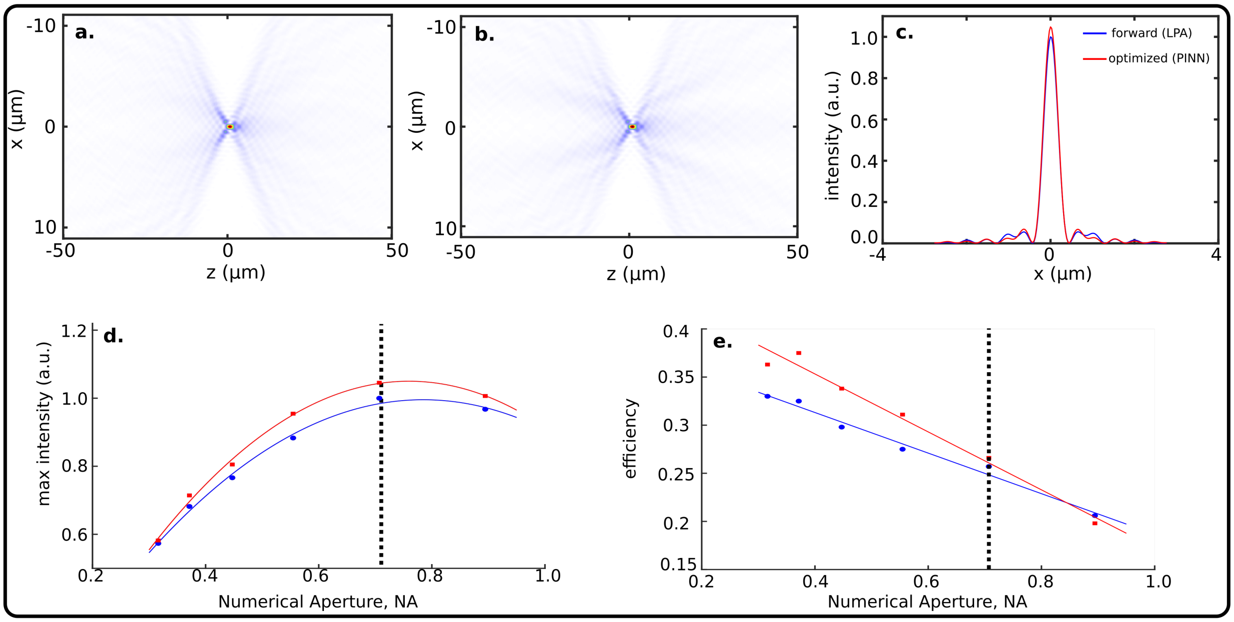


Figure 4.4: Efficiency and intensity sweeps of forward designed lenses and optimized lenses. **a.** Focal spot intensity profile of a forward designed lens with focal length $F = 500\mu\text{m}$. **b.** Focal spot intensity profile of an optimized lens. **c.** Slices of intensity profiles for both lenses. The intensity was normalized such that the maximum intensity of the forward designed lens is 1. The theoretical performance improvement is $\sim 3\%$. **d.** Maximum intensity at the focal spot vs lens numerical aperture (NA). Intensities are normalized such that the maximum of the largest forward designed intensity is set to 1. **e.** Theoretically computed efficiencies of the lenses vs NA. The solid lines are visual aids for the trend and do not correspond to a theoretical prediction.

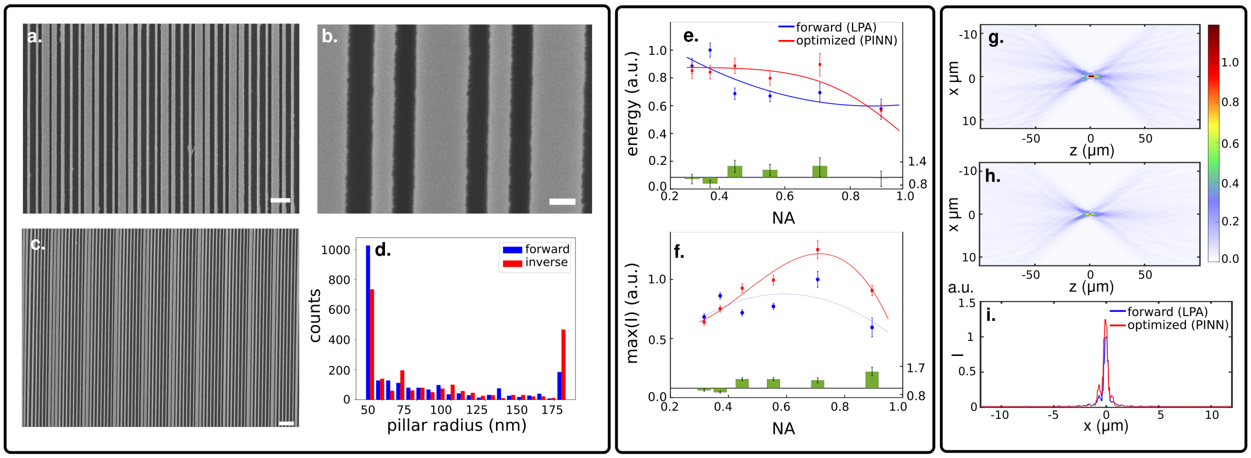


Figure 4.5: **a.-c.** Scanning Electron Microscope (SEM) images of the fabricated SiN metalens with focal length $F = 500\mu$m. The scale bars correspond to $1\mu m$, $0.1\mu m$, and $1\mu m$, respectively. **d.** Counts of pillar half-widths of the forward and inverse designed lens. **e.** Measured intensity contained in the region given by $3\times$ FWHM of the focal spot vs lens numerical aperture. The units are normalized to the largest intensity integral of the forward design. **f.** Maximum intensity at the focal spot. The inverse designed lenses outperform the forward designed lenses for NA > 0.44. The lines are visual aids and not fits to a theoretical model. Units are normalized to the largest intensity of the forward designed lens. In the NA $= 0.9$ ($F = 250\mu$m) case an improvement of 53% is observed. **g.** Experimentally measured field intensities of the forward designed lens and **h.** of the inverse designed lens. **i.** Intensity slice at the focal spot. The intensities are normalized such that the maximum intensity of the forward designed lens is 1. The intensity of the inverse designed lens focal spot is 1.25.

a quantitative value to compare the lens efficiency [11] among different devices. Fig.4.5**f** plots the maximum intensity plot as a function of the lens NA. For optimized lenses with NA $>$ 0.44, we see improvements of more than 25%, with a maximum improvement of 53% for the NA $= 0.9$ lens. The experimentally determined intensity integral, which is analogous to the efficiency of a lens, on has improvements of more than 18% in all cases except for the NA=0.9 case. This is because the FWHM of the optimized lens at the NA=0.9 case is actually smaller than the FWHM of the forward designed lens, leading to a smaller integration area when computing the energy.

Fig. 4.5**g** shows experimentally measured field profiles of the forward designed $F = 500\mu$m meta-lens. Fig. 4.5**h** shows the same for an optimized lens. Fig.4.5**i** is the slice of the focal spot intensity profile along the $z = F$ plane. In all these figures, the intensity is normalized such that the maximum intensity of the forward designed lens is 1.

4.5 Methods

4.5.1 Fabrication

All devices described and discussed in Figure 5 (forward and PINN designed) were fabricated on the same substrate. First a $\sim$ 700 nm SiN film was deposited on a quartz wafer using plasma enhanced chemical vapor deposition (SPTS Delta LPX PECVD). A thin film of a polymer resist (ZEP 520-A) and a thin film of a discharging polymer layer (DisCharge H2O) were subsequently spun onto the sample. We then used electron beam lithography (JEOL JBX-6300FS, 100 keV, 2nA) to write the various structures. After development, a short descum step (Glow Research, Autoglow, 12 s, 100 W) was used to remove remaining resist residues and subsequently a layer of 60 nm AlOx was deposited using a home-built e-beam evaporator. After overnight lift-off in warm NMP and a further plasma cleaning step in O2, we used inductively coupled reactive ion etching (Oxford Instruments PlasmaLabSystem100) with a fluorine gas chemistry to transfer the pattern from the AlOx hard mask into the underlying SiN layer. The final thickness of the etched layer indicated a pillar height of

$\sim$650 nm.

4.5.2 Experiment set up

For intensity measurements, light from a HeNe laser was transmitted through the backside of the chip and measured on the device side using a translatable microscope relay setup. In detail, the sample was mounted at a fixed position on a kinematic holder, allowing the fine adjustment for pitch and yaw, as well as the lateral position. Light was transmitted through the substrate side and would propagate entirely in air. The resulting focusing pattern was measured using a microscope setup consisting of a Nikon 100X LU Plan Fluor objective with 0.9 NA (equal or higher to the NA of the meta-optic), a tube lens (Thorlabs), and a camera (Allied Vision ProSilica GT1930), which were mounted on a programmable automated translation stage (NewPort). Frames were acquired at specific intervals of the movement, which allowed the reconstruction of the intensity vs focal distance.

4.6 Discussion

We have developed a PINN to use as a proxy surrogate model for simulating the full Maxwell's equations to design dielectric meta-optics. We used the PINN to optimize pillar half-widths to maximize the intensity at the focal spot of 1 mm aperture cylindrical meta-lenses. We demonstrated experimental improvements of the maximum intensity of the lenses up to 53%. We also want to note that this method was useful for the inverse design of extended depth of focus lenses[14] (see appendix C.6). This model did not use the LPA, but simulated meta-atoms by splitting up the device into chunks with overlapping boundaries, and stitching the chunks together to approximate the full field response. We emphasize that FDFD simulations were never carried out to train the PINN, and we only minimized the residual of the PDE itself to train the network. The PINN training took approximately 2 hours on our machine. In our studies, this method provided approximately a 3-5x speedup over conventional FDFD with overlapping boundary conditions, and was much simpler to use as a forward simulator for optimization problems since it can be used as a simple map from ϵ to E-field with gradients computed by automatic differentiation.

We would also like to note that the theoretical intensity and efficiency improvements are quite a bit smaller than their experimental counterparts. While we don't have a clear explanation for this discrepancy, the theoretical and experimental trends in lens improvement are similar. We hypothesise that the inverse designed lenses may be more tolerant to fabrication imperfections. The inverse design solution we introduced in this chapter can be integrated into various computationally intensive tasks which require mate-optical inverse design such as the end-to-end optimization of computational imaging systems and the design of optical neural networks[96, 60]. It is worth noting, however, that this method is not a general numerical solver. It is limited to predicting electromagnetic field responses from fixed source, material, and boundary parameters. Source type and k-vector, dielectric constant, geometry type (rectangular pillar of fixed height), and boundary conditions must all remain constant for this method to work. If any of these parameters are modified, the PINN must be retrained. Furthermore, the method we presented was only implemented under a 2D approximation. Extending this method to 3 dimensions would take significant effort due to the fact that the electric field $\mathcal{E}$ could no longer be treated as a scalar field, and the full vector nature would have to be modeled. On a $n \times n$ grid in 2D, the Maxwell operator $\left[\nabla^2 + \omega^2\epsilon\right]$ results in a $n^2 \times n^2$ matrix, while for a $n \times n \times n$ 3D grid the Maxwell operator $\left[\nabla \times \nabla \times -\omega^2\epsilon\right]$ result in $9n^3 \times 9n^3$ square matrices due to the additional 2 vector field components that must be modeled. However, these operators are sparse with number of nonzero elements that scale as $\sim 38n^3$ in 3D, making small problems still manageable. The other problem with generalizing this method to 3D is the large null-space of the $\nabla \times \nabla \times$ operator which results in slow convergence of numerical methods[77, 76]. Its highly likely that this could also affect the training of the PINN, and require regularization or preconditioning which deflates the null space of this operator to properly converge onto a solution. On the other hand, in this work we showed that machine-precision numerical accuracy of numerical solvers may be not be needed for inverse design methods with FDFD. Solvers could be sped up by relaxing the relative error tolerance, such that iterative solvers converge quicker for predicting the forward and adjoint problems.

Chapter 5

CONCLUSION AND OUTLOOK

The inverse design and optimization of metamaterials is an academically interesting problem. They provide a way of controlling the response of electromagnetic fields in 2 dimensions, by modifying scatterer geometries. In a way, one can think of each metasurface scatterer as an individual knob you can twist, but in doing so, you modulate the electromagnetic response of a metasurface in a non-trivial way. To engineer electromagnetic scattering from metasurfaces, sophisticated simulation methods are and optimization techniques are required. In this thesis we outlined three separate forward simulation methods that can be used for inverse design. None of these methods rely on the LPA to model scattering from dielectric metasurface scatterers and provide a significant computational resource reduction for inverse design and metasurface optimization problems. Furthermore, we used these techniques to design new optical meta-devices, as well as optimized existing designs for higher performance.

In the first chapter, we gave a brief overview of existing optical devices. We described the operation of conventional refractive optics via Snell's law, and motivated the use of diffractive optical elements for miniaturization of optical systems via wave optics. We motivated the use of metamaterials for optical miniaturization and realization of novel optical systems. We described the common design strategies for metamaterials - forward and inverse design, and argued that inverse design is of particular importance for designing metamaterials manufactured on low refractive material platforms as well as cases where the geometric gradients of individual scatterers are not negligible. The design of metamaterials still has a number of outstanding challenges. The efficient design of high numerical aperture lenses [20, 126], correcting for chromatic and geometric aberrations [101, 102, 8, 26], low index metasurface design [118, 126], multi-functional metasurface design [65, 29], design of metasurfaces for feature-specific imaging [97], volumetric metasurface design [121], and optical neural network design [63] are all examples of outstanding problems in this field that can benefit from more advanced inverse design techniques.

In chapter 2, we introduced the inverse design of meta-materials based on GMMT, which solves the forward scattering problem for arrays of spheroidal scatterers. Although the method is promising, and was used to design new multifunctional metasurfaces, as well as optimzie the efficiency of an existing metasurface, the method does not provide enough of a computational resource relief to be an efficient method for inverse design of very large area systems. This problem can be alleviated by splitting up the simulation region into chunks, and neglecting the scattering from scatterers which are very far away from each other, reducing the dimensionality of the problem. Furthermore, in its current state, this method only works for geometries that can be smoothly parameterized in spherical coordinates, which limits us to designing devices that are not easily manufacturable. It is possible to extend this method to arbitrary geometries by using the boundary element method [43, 1]. This method is also limited to sparsely packed particles, which offers another challenge. This challenge can also be alleviated by using the plane-wave expansion method [93, 32].

In chapter 3, we outlined a data-driven approach to modeling scattering from metasurface scatterers. We simulated scattering from a set of metamaterial lenses, pre-processed the data using the SVD, and trained a neural network to predict the basis coefficients that reconstruct the original field in terms of the singular vectors found the SVD from a single scatterer, and its nearest neighbors. This method was used to design a multi-functional lens that focuses light for one wavelength and creates a toroidal focal spot for a different wavelength, as well as optimizing an extended depth of focus lens. Its extremely fast and memory efficient, making it easy to run on a mid-range laptop. The drawback to this method however was the fact that we were never able to see an efficiency improvement of a lens after optimization. One interesting avenue to pursue from this project is the application of data-scientific methods to understanding how complex systems like this behave. Methods like dynamic mode decomposition, principle component analysis, proper orthogonal decomposition, as well as a multitude of other methods could be applied to understanding how statistical properties of the distribution of scatterer geometries affects spatial and frequency domain modes of the electromagnetic field scattering. This could lead to a novel understanding of metasurfaces as well as other complex, highly coupled systems, and lead to better meta-device designs.

In chapter 4, we outlined the methodology for using a physics informed neural network to replace a conventional numerical simulator for the inverse design of 1-dimensional metasurface structures. In this work we were able to achieve almost a 5x speedup in the forward simulation of electromagnetic fields. We almost demonstrated an experimental improvement in the maximum intensity value at the lens focal spot of at most 53%, which is at least an order of magnitude larger than previous works. Its important to note that this method does not rely on the local phase approximation, and instead incorporates the scattering from near-neighbor particles to improve the design method. Furthermore, we want to highlight the importance of experimentally verifying inverse design strategies, as it is very easy to make simple mistakes that lead you to believe you have a good method, when in reality you do not C.7. In its current state, this method can only predict 2D fields from a 1D metasurface, so the natural extension of this method would be to train a neural network to predict full-vectorial 3D field scattering from 2D metasurface devices. This however increases the complexity of the problem, and would require increased computational resources, as well as new techniques for training the neural network in order to get the simulation method to converge properly. Another avenue to pursue is perhaps to relax the convergence condition of the simulation method for inverse design. As we have shown, the residual of the field computed with the neural network, is on the order of ~ 0.5, versus a numerical simulator which has machine level precision. This precision, however, may be overkill for inverse design problems.

BIBLIOGRAPHY

[1] `http://github.com/homerreid/scuff-EM`.

[2] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[3] Francesco Aieta, Mikhail A. Kats, Patrice Genevet, and Federico Capasso. Multiwavelength achromatic metasurfaces by dispersive phase compensation. *SCIENCE*, 347(6228):1342–1345, MAR 20 2015.

[4] Francesco Aieta, Mikhail A. Kats, Patrice Genevet, and Federico Capasso. Multiwavelength achromatic metasurfaces by dispersive phase compensation. *Science*, 347(6228):1342–1345, 2015.

[5] Sensong An, Clayton Fowler, Bowen Zheng, Mikhail Y. Shalaginov, Hong Tang, Hang Li, Li Zhou, Jun Ding, Anuradha Murthy Agarwal, Clara Rivero-Baleine, Kathleen A. Richardson, Tian Gu, Juejun Hu, and Hualiang Zhang. A novel modeling approach for all-dielectric metasurfaces using deep neural networks, 2019.

[6] U Andral, J Buguet, R Si Fodil, F Amrani, F Billard, E Hertz, and Ph Grelu. Toward an autosetting mode-locked fiber laser cavity. *JOSA B*, 33(5):825–833, 2016.

[7] U Andral, R Si Fodil, F Amrani, F Billard, E Hertz, and Ph Grelu. Fiber laser mode locked through an evolutionary algorithm. *Optica*, 2(4):275–278, 2015.

[8] Amir Arbabi, Ehsan Arbabi, Yu Horie, Seyedeh Mahsa Kamali, and Andrei Faraon. Planar metasurface retroreflector. *Nature Photonics*, 11:415 EP –, Jun 2017.

[9] Amir Arbabi, Yu Horie, Mahmood Bagheri, and Andrei Faraon. Dielectric metasurfaces for complete control of phase and polarization with subwavelength spatial resolution and high transmission. *Nature Nanotechnology*, 10:937 EP –, Aug 2015.

[10] Amir Arbabi, Yu Horie, Alexander J. Ball, Mahmood Bagheri, and Andrei Faraon. Subwavelength-thick lenses with high numerical apertures and large efficiency based on high-contrast transmitarrays. *Nature Communications*, 6:7069 EP –, May 2015. Article.

[11] Amir Arbabi, Yu Horie, Alexander J. Ball, Mahmood Bagheri, and Andrei Faraon. Subwavelength-thick lenses with high numerical apertures and large efficiency based on high-contrast transmitarrays. *Nature Communications*, 6(1):7069, May 2015.

[12] Ehsan Arbabi, Seyedeh Mahsa Kamali, Amir Arbabi, and Andrei Faraon. Full-stokes imaging polarimetry using dielectric metasurfaces. *ACS Photonics*, 5(8):3132–3140, Aug 2018.

[13] Thomas Baumeister, Steven L Brunton, and J Nathan Kutz. Deep learning and model predictive control for self-tuning mode-locked lasers. *JOSA B*, 35(3):617–626, 2018.

[14] Elyas Bayati, Raphaël Pestourie, Shane Colburn, Zin Lin, Steven G. Johnson, and Arka Majumdar. Inverse designed metalenses with extended depth of focus. *ACS Photonics*, 7(4):873–878, Mar 2020.

[15] Elyas Bayati, Alan Zhan, Shane Colburn, Maksym Viktorovich Zhelyeznyakov, and Arka Majumdar. Role of refractive index in metalens performance. *Appl. Opt.*, 58(6):1460–1466, Feb 2019.

[16] Elyas Bayati, Alan Zhan, Shane Colburn, Maksym Viktorovich Zhelyeznyakov, and Arka Majumdar. Role of refractive index in metalens performance. *Appl. Opt.*, 58(6):1460–1466, Feb 2019.

[17] S. L. Brunton and J. N. Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control.* Cambridge University Press, 2019.

[18] Steven L Brunton, Xing Fu, and J Nathan Kutz. Self-tuning fiber lasers. *IEEE Journal of Selected Topics in Quantum Electronics*, 20(5), 2014.

[19] Steven L. Brunton and J. Nathan Kutz. *Frontmatter*, pages i–iv. Cambridge University Press, 2019.

[20] Steven J. Byrnes, Alan Lenef, Francesco Aieta, and Federico Capasso. Designing large, high-efficiency, high-numerical-aperture, transmissive meta-lenses for visible light. *Opt. Express*, 24(5):5110–5124, Mar 2016.

[21] Hou-Tong Chen, Antoinette Taylor, and Nanfang yu. A review of metasurfaces: Physics and applications. *Reports on Progress in Physics*, 79, 05 2016.

[22] Mingkun Chen, Robert Lupoiu, Chenkai Mao, Der-Han Huang, Jiaqi Jiang, Philippe Lalanne, and Jonathan A. Fan. High speed simulation and freeform optimization of nanophotonic devices with physics-augmented deep learning. *ACS Photonics*, Aug 2022.

[23] Wei Ting Chen, Alexander Y. Zhu, Vyshakh Sanjeev, Mohammadreza Khorasaninejad, Zhujun Shi, Eric Lee, and Federico Capasso. A broadband achromatic metalens for focusing and imaging in the visible. *Nature Nanotechnology*, 13(3):220–226, 2018.

[24] Yuyao Chen, Lu Lu, George Em Karniadakis, and Luca Dal Negro. Physics-informed neural networks for inverse problems in nano-optics and metamaterials. *Opt Express*, 28(8):11618–11633, April 2020.

[25] Haejun Chung and Owen D. Miller. High-na achromatic metalenses by inverse design. *Opt. Express*, 28(5):6945–6965, Mar 2020.

[26] Shane Colburn, Alan Zhan, Elyas Bayati, James Whitehead, Albert Ryou, Luocheng Huang, and Arka Majumdar. Broadband transparent and cmos-compatible flat optics with silicon nitride metasurfaces [invited]. *Opt. Mater. Express*, 8(8):2330–2344, Aug 2018.

[27] Shane Colburn, Alan Zhan, and Arka Majumdar. Metasurface optics for full-color computational imaging. *Science Advances*, 4(2), 2018.

[28] Shane Colburn, Alan Zhan, and Arka Majumdar. Metasurface optics for full-color computational imaging. *Science Advances*, 4(2), February 2018.

[29] Zi-Lan Deng, Yaoyu Cao, Xiangping Li, and Guo Ping Wang. Multifunctional metasurface: from extraordinary optical transmission to extraordinary optical diffraction in a single structure. *Photon. Res.*, 6(5):443–450, May 2018.

[30] Adrian Doicu, Thomas Wriedt, and Jurij A. Eremin. *Light Scattering by Systems of Particles: Null-Field Method with Discrete Sources*. Springer, 2006.

[31] Massimo Donelli. Design of broadband metal nanosphere antenna arrays with a hybrid evolutionary algorithm. *Opt. Lett.*, 38(4):401–403, Feb 2013.

[32] Amos Egel, Krzysztof M. Czajkowski, Dominik Theobald, Konstantin Ladutenko, Alexey S. Kuznetsov, and Lorenzo Pattelli. SMUTHI: A python package for the simulation of light scattering by multiple particles near or between planar interfaces. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 273:107846, oct 2021.

[33] Amos Egel, Lorenzo Pattelli, Giacomo Mazzamuto, Diederik S. Wiersma, and Uli Lemmer. Celes: Cuda-accelerated simulation of electromagnetic scattering by large ensembles of spheres. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 199:103–110, 2017.

[34] Victor Egorov, Michal Eitan, and Jacob Scheuer. Genetically optimized all-dielectric metasurfaces. *Opt. Express*, 25(3):2583–2593, Feb 2017.

[35] Victor Egorov, Michal Eitan, and Jacob Scheuer. Genetically optimized all-dielectric metasurfaces. *Opt. Express*, 25(3):2583–2593, Feb 2017.

[36] Jonathan A. Fan. High performance metasurfaces based on inverse design (conference presentation), 2017.

[37] Xing Fu, Steven L Brunton, and J Nathan Kutz. Classification of birefringence in mode-locked fiber lasers using machine learning and sparse representation. *Optics express*, 22(7):8585–8597, 2014.

[38] Li Gao, Xiaozhong Li, Dianjing Liu, Lianhui Wang, and Zongfu Yu. A bidirectional deep neural network for accurate silicon color design. *Advanced Materials*, 31(51):1905467, 2019.

[39] Joseph W Goodman. Introduction to fourier optics. *Introduction to Fourier optics, 3rd ed., by JW Goodman. Englewood, CO: Roberts & Co. Publishers, 2005*, 1, 2005.

[40] Huijie Guo, Jing Lin, Meng Qiu, Junxia Tian, Qian Wang, Yue Li, Shulin Sun, Qiong He, Shiyi Xiao, and Lei Zhou. Flat optical transparent window: mechanism and realization based on metasurfaces. *Journal of Physics D: Applied Physics*, 51(7):074001, feb 2018.

[41] Paul Hansen and Lambertus Hesselink. Accurate adjoint design sensitivities for nano metal optics. *Opt. Express*, 23(18):23899–23923, Sep 2015.

[42] Paul Hansen and Lambertus Hesselink. Accurate adjoint design sensitivities for nano metal optics. *Opt. Express*, 23(18):23899–23923, Sep 2015.

[43] M. T. Homer Reid and S. G. Johnson. Efficient Computation of Power, Force, and Torque in BEM Scattering Calculations. *ArXiv e-prints*, July 2013.

[44] Liyi Hsu, Matthieu Dupré, Abdoulaye Ndao, Julius Yellowhair, and Boubacar Kanté. Local phase method for designing and optimizing metasurface devices. *Opt. Express*, 25(21):24974–24982, Oct 2017.

[45] Liyi Hsu, Matthieu Dupré, Abdoulaye Ndao, Julius Yellowhair, and Boubacar Kanté. Local phase method for designing and optimizing metasurface devices. *Opt. Express*, 25(21):24974–24982, Oct 2017.

[46] Liyi Hsu, Matthieu Dupré, Abdoulaye Ndao, Julius Yellowhair, and Boubacar Kanté. Local phase method for designing and optimizing metasurface devices. *Opt. Express*, 25(21):24974–24982, Oct 2017.

[47] Tyler W. Hughes, Momchil Minkov, Ian A. D. Williamson, and Shanhui Fan. Adjoint method and inverse design for nonlinear nanophotonic devices, 2018.

[48] John David Jackson. *Classical electrodynamics*. Wiley, New York, NY, 3rd ed. edition, 1999.

[49] J.S. Jensen and O. Sigmund. Topology optimization for nano-photonics. *Laser & Photonics Reviews*, 5(2):308–321, 2011.

[50] M. C. Johnson, S. L. Brunton, N. B. Kundtz, and J. N. Kutz. Extremum-seeking control of a beam pattern of a reconfigurable holographic metamaterial antenna. *Journal of the Optical Society of America A*, 33(1):59–68, 2016.

[51] Seyedeh Mahsa Kamali, Ehsan Arbabi, Amir Arbabi, Yu Horie, MohammadSadegh Faraji-Dana, and Andrei Faraon. Angle-multiplexed metasurfaces: Encoding independent wavefronts in a single metasurface under different illumination angles. *Phys. Rev. X*, 7:041056, Dec 2017.

[52] Ebrahim Karimi, Sebastian A. Schulz, Israel De Leon, Hammam Qassim, Jeremy Upham, and Robert W. Boyd. Generating optical orbital angular momentum at visible wavelengths using a plasmonic metasurface. *Light: Science & Applications*, 3(5):e167–e167, May 2014.

[53] Sharmila Karumuri, Rohit Tripathy, Ilias Bilionis, and Jitesh Panchal. Simulator-free solution of high-dimensional stochastic elliptic partial differential equations using deep neural networks. *Journal of Computational Physics*, 404:109120, 2020.

[54] Yashar Kiarashinejad, Mohammadreza Zandehshahvar, Sajjad Abdollahramezani, Omid Hemmatyar, Reza Pourabolghasem, and Ali Adibi. Knowledge discovery in nanophotonics using geometric deep learning. *Advanced Intelligent Systems*, 2(2), 2019.

[55] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[56] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[57] Nanxi Li, Zhengji Xu, Yuan Dong, Ting Hu, Qize Zhong, Yuan Hsing Fu, Shiyang Zhu, and Navab Singh. Large-area metasurface on cmos-compatible fabrication platform: Driving flat optics from lab to fab. *Nanophotonics*, -1, 04 2020.

[58] Xiaozhong Li, Jing Shu, Wenhua Gu, and Li Gao. Deep neural network for plasmonic sensor modeling. *Opt. Mater. Express*, 9(9):3857–3862, Sep 2019.

[59] Samuel Clark Ligon, Robert Liska, Jürgen Stampfl, Matthias Gurr, and Rolf Mülhaupt. Polymers for 3d printing and customized additive manufacturing. *Chemical Reviews*, 117(15):10212–10290, Aug 2017.

[60] Xing Lin, Yair Rivenson, Nezih T Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406):1004–1008, July 2018.

[61] Zin Lin, Victor Liu, Raphael Pestourie, and Steven G. Johnson. Topology optimization of freeform large-area metasurfaces. *Opt. Express*, 27(11):15765–15775, May 2019.

[62] Yu lin Xu. Electromagnetic scattering by an aggregate of spheres. *Appl. Opt.*, 34(21):4573–4588, Jul 1995.

[63] Che Liu, Qian Ma, Zhang Jie Luo, Qiao Ru Hong, Qiang Xiao, Hao Chi Zhang, Long Miao, Wen Ming Yu, Qiang Cheng, Lianlin Li, and Tie Jun Cui. A programmable diffractive deep neural network based on a digital-coding metasurface array. *Nature Electronics*, 5(2):113–122, Feb 2022.

[64] Dianjing Liu, Yixuan Tan, Erfan Khoram, and Zongfu Yu. Training deep neural networks for the inverse design of nanophotonic structures. *ACS Photonics*, 5(4):1365–1369, Apr 2018.

[65] Mingze Liu, Wenqi Zhu, Pengcheng Huo, Lei Feng, Maowen Song, Cheng Zhang, Lu Chen, Henri J. Lezec, Yanqing Lu, Amit Agrawal, and Ting Xu. Multifunctional metasurfaces enabled by simultaneous and independent control of phase and amplitude for orthogonal polarization states. *Light: Science & Applications*, 10(1):107, May 2021.

[66] Victor Liu and Shanhui Fan. S^4 : A free electromagnetic solver for layered periodic structures. *Computer Physics Communications*, 183(10):2233–2244, 2012.

[67] Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. Deepxde: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021.

[68] Lu Lu, Raphaël Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven Johnson. Physics-informed neural networks with hard constraints for inverse design. *SIAM Journal on Scientific Computing*, 43:B1105–B1132, 11 2021.

[69] Wei Ma, Feng Cheng, and Yongmin Liu. Deep-learning-enabled on-demand design of chiral metamaterials. *ACS Nano*, 12(6):6326–6334, Jun 2018.

[70] Daniel W. Mackowski and Michael I. Mishchenko. Calculation of the t matrix and the scattering matrix for ensembles of spheres. *J. Opt. Soc. Am. A*, 13(11):2266–2278, Nov 1996.

[71] Itzik Malkiel, Michael Mrejen, Achiya Nagler, Uri Arieli, Lior Wolf, and Haim Suchowski. Plasmonic nanostructure design and characterization via deep learning. *Light: Science & Applications*, 7(1):60, Sep 2018.

[72] Mahdad Mansouree, Hyounghan Kwon, Ehsan Arbabi, Andrew McClung, Andrei Faraon, and Amir Arbabi. Multifunctional 2.5d metastructures enabled by adjoint optimization. *Optica*, 7(1):77–84, Jan 2020.

[73] Michael I. Mishchenko, Larry D. Davis, and Andrew A. Lacis. *Scattering, Absoprtion, and Emission of Light by Small Particles.* NASA Goddard Institute for Space Studies, 2002.

[74] Sean Molesky, Zin Lin, Alexander Y. Piggott, Weiliang Jin, Jelena Vucković, and Alejandro W. Rodriguez. Inverse design in nanophotonics. *Nature Photonics*, 12(11):659–670, November 2018.

[75] Raghavan Narasimhan and Yves Nievergelt. *Functions of Several Complex Variables*, pages 151–160. Birkhäuser Boston, Boston, MA, 2001.

[76] Greg Newman and Chester Weiss. Electromagnetic induction in a generalized 3d anisotropic earth, part 2: The lin preconditioner. *Geophysics*, 68:922–30, 05 2003.

[77] Gregory A. Newman and David L. Alumbaugh. Three-dimensional induction logging problems, Part 2: A finite-difference solution. *Geophysics*, 67(2):484–491, 01 2002.

[78] Frank L. Pedrotti, Leno M. Pedrotti, and Leno S. Pedrotti. *Introduction to Optics*. Cambridge University Press, 3 edition, 2017.

[79] John Perlin and Mark Z. Jacobson. *Chapter 4, Burning Mirrors*, page 37–40. New World Library, 2022.

[80] Raphaël Pestourie, Carlos Pérez-Arancibia, Zin Lin, Wonseok Shin, Federico Capasso, and Steven G. Johnson. Inverse design of large-area metasurfaces. *Opt. Express*, 26(26):33732–33747, Dec 2018.

[81] John Peurifoy, Yichen Shen, Li Jing, Yi Yang, Fidel Cano-Renteria, Brendan G. DeLacy, John D. Joannopoulos, Max Tegmark, and Marin Soljačić. Nanophotonic particle simulation and inverse design using artificial neural networks. *Science Advances*, 4(6), 2018.

[82] Alexander Y. Piggott, Jesse Lu, Konstantinos G. Lagoudakis, Jan Petykiewicz, Thomas M. Babinec, and Jelena Vučković. Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer. *Nature Photonics*, 9(6):374–377, Jun 2015.

[83] Alexander Y. Piggott, Jan Petykiewicz, Logan Su, and Jelena Vučković. Fabrication-constrained nanophotonic inverse design. *Scientific Reports*, 7(1):1786, May 2017.

[84] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[85] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481):1026–1030, 2020.

[86] Bing Shen, Peng Wang, Randy Polson, and Rajesh Menon. An integrated-nanophotonics polarization beamsplitter with 2.4 × 2.4 mm2 footprint. *Nature Photonics*, 9:378 EP –, May 2015.

[87] Sunae So, Trevon Badloe, Jaebum Noh, Jorge Bravo-Abad, and Junsuk Rho. Deep learning enabled inverse design in nanophotonics. *Nanophotonics*, 9(5):1041–1057, 2020.

[88] Chang Sun, Eurika Kaiser, Steven L Brunton, and J Nathan Kutz. Deep reinforcement learning for optical systems: A case study of mode-locked lasers. *arXiv preprint arXiv:2006.05579*, 2020.

[89] Shulin Sun, Qiong He, Shiyi Xiao, Qin Xu, Xin Li, and Lei Zhou. Gradient-index metasurfaces as a bridge linking propagating waves and surface waves. *Nature Materials*, 11(5):426–431, May 2012.

[90] A.M. Tartakovsky, C. Marrero, Paris Perdikaris, Guzel Tartakovsky, and D. Barajas-Solano. Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems. *Water Resources Research*, 56:e2019WR026731, 05 2020.

[91] R. H. T.Bates. Analytic constraints on electromagnetic field computations. *IEEE Transactions on Microwave Theory and Techniques*, 23(8):605–623, 1975.

[92] Dominik Theobald, Amos Egel, Guillaume Gomard, and Uli Lemmer. Plane-wave coupling formalism for t-matrix simulations of light scattering by nonspherical particles. *Phys. Rev. A*, 96:033822, Sep 2017.

[93] Dominik Theobald, Amos Egel, Guillaume Gomard, and Uli Lemmer. Plane-wave coupling formalism for t-matrix simulations of light scattering by nonspherical particles. *Physical Review A*, 96(3):033822, 2017.

[94] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra.* SIAM, 1997.

[95] Rahul Trivedi, Logan Su, Jesse Lu, Martin F. Schubert, and Jelena Vuckovic. Data-driven acceleration of photonic simulations. *Scientific Reports*, 9(1):19728, Dec 2019.

[96] Ethan Tseng, Shane Colburn, James Whitehead, Luocheng Huang, Seung-Hwan Baek, Arka Majumdar, and Felix Heide. Neural nano-optics for high-quality thin lens imaging. *Nature Communications*, 12(1):6493, Nov 2021.

[97] Ethan Tseng, Shane Colburn, James Whitehead, Luocheng Huang, Seung-Hwan Baek, Arka Majumdar, and Felix Heide. Neural nano-optics for high-quality thin lens imaging. *Nature Communications*, 12(1):6493, Nov 2021.

[98] Giuseppe Vicidomini, Paolo Bianchini, and Alberto Diaspro. Sted super-resolved microscopy. *Nature Methods*, 15(3):173–182, Mar 2018.

[99] Ke Wang, Jie Zhao, Qiang Cheng, Di Sha Dong, and Tie Jun Cui. Broadband and broad-angle low-scattering metasurface based on hybrid optimization algorithm. *Scientific Reports*, 4(1):5935, Aug 2014.

[100] Ke Wang, Jie Zhao, Qiang Cheng, Di Sha Dong, and Tie Jun Cui. Broadband and broad-angle low-scattering metasurface based on hybrid optimization algorithm. *Scientific Reports*, 4(1):5935, Aug 2014.

[101] Shuming Wang, Pin Chieh Wu, Vin-Cent Su, Yi-Chieh Lai, Mu-Ku Chen, Hsin Yu Kuo, Bo Han Chen, Yu Han Chen, Tzu-Ting Huang, Jung-Hsi Wang, Ray-Ming Lin, Chieh-Hsiung Kuan, Tao Li, Zhenlin Wang, Shining Zhu, and Din Ping Tsai. A broadband achromatic metalens in the visible. *Nature Nanotechnology*, 13(3):227–232, Mar 2018.

[102] Shuming Wang, Pin Chieh Wu, Vin-Cent Su, Yi-Chieh Lai, Cheng Hung Chu, Jia-Wern Chen, Shen-Hung Lu, Ji Chen, Beibei Xu, Chieh-Hsiung Kuan, Tao Li, Shining Zhu, and Din Ping Tsai. Broadband achromatic optical metasurface devices. *Nature Communications*, 8(1):187, August 2017.

[103] P. C. Waterman. Symmetry, unitarity, and geometry in electromagnetic scattering. *Phys. Rev. D*, 3:825–839, Feb 1971.

[104] Reinoud F. Wolffenbuttel and M. Amir Ghaderi. Design of large-area metasurfaces for the mid-ir and suited for cmos-compatible fabrication by masked lithography. *Photonics and Nanostructures - Fundamentals and Applications*, 51:101050, 2022.

[105] Rachel Won. Intelligent learning with light. *Nature Photonics*, 12(10):571–573, 2018.

[106] RI Woodward and Edmund JR Kelleher. Towards "smart lasers": self-optimisation of an ultrafast pulse source using a genetic algorithm. *Scientific reports*, 6:37616, 2016.

[107] Shiyi Xiao, Holger Mühlenbernd, Guixin Li, Mitchell Kenney, Fu Liu, Thomas Zentgraf, Shuang Zhang, and Jensen Li. Helicity-preserving omnidirectional plasmonic mirror. *Advanced Optical Materials*, 4(5):654–658, 2016.

[108] Shiyi Xiao, Jiarong Wang, Fu Liu, Shuang Zhang, Xiaobo Yin, and Jensen Li. Spin-dependent optics with metasurfaces. *Nanophotonics*, 6(1):215–234, 2017.

[109] Shiyi Xiao, Fan Zhong, Hui Liu, Shining Zhu, and Jensen Li. Flexible coherent control of plasmonic spin-hall effect. *Nature Communications*, 6(1):8360, Sep 2015.

[110] Bowen Yang, Tong Liu, Huijie Guo, Shiyi Xiao, and Lei Zhou. High-performance meta-devices based on multilayer meta-atoms: interplay between the number of layers and phase coverage. *Science Bulletin*, 64(12):823–835, 2019. SPECIAL TOPIC: Electromagnetic Metasurfaces: from Concept to Applications.

[111] Alireza Yazdani, Lu Lu, Maziar Raissi, and George Em Karniadakis. Systems biology informed deep learning for inferring parameters and hidden dynamics. *PLoS Comput Biol*, 16(11):e1007575, November 2020.

[112] Weimin Ye, Franziska Zeuner, Xin Li, Bernhard Reineke, Shan He, Cheng-Wei Qiu, Juan Liu, Yongtian Wang, Shuang Zhang, and Thomas Zentgraf. Spin and wavelength multiplexed nonlinear metasurface holography. *Nature Communications*, 7(1):11930, Jun 2016.

[113] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14(3):302–307, 1966.

[114] Nanfang Yu and Federico Capasso. Flat optics with designer metasurfaces. *Nature Materials*, 13(2):139–150, Feb 2014.

[115] Nanfang Yu, Patrice Genevet, Mikhail A. Kats, Francesco Aieta, Jean-Philippe Tetienne, Federico Capasso, and Zeno Gaburro. Light propagation with phase discontinuities: Generalized laws of reflection and refraction. *Science*, 334(6054):333–337, 2011.

[116] Wei Yuan, Li-Hua Li, Wing-Bun Lee, and Chang-Yuen Chan. Fabrication of microlens array and its application: A review. *Chinese Journal of Mechanical Engineering*, 31(1):16, Feb 2018.

[117] Alan Zhan, Shane Colburn, Christopher M. Dodson, and Arka Majumdar. Metasurface freeform nanophotonics. *Scientific Reports*, 7(1):1673, 2017.

[118] Alan Zhan, Shane Colburn, Rahul Trivedi, Taylor K. Fryett, Christopher M. Dodson, and Arka Majumdar. Low-contrast dielectric metasurface optics. *ACS Photonics*, 3(2):209–214, 2016.

[119] Alan Zhan, Shane Colburn, Rahul Trivedi, Taylor K. Fryett, Christopher M. Dodson, and Arka Majumdar. Low-contrast dielectric metasurface optics. *ACS Photonics*, 3(2):209–214, 2016.

[120] Alan Zhan, Taylor K. Fryett, Shane Colburn, and Arka Majumdar. Inverse design of optical elements based on arrays of dielectric spheres. *Appl. Opt.*, 57(6):1437–1446, Feb 2018.

[121] Alan Zhan, Ricky Gibson, James Whitehead, Evan Smith, Joshua R. Hendrickson, and Arka Majumdar. Controlling three-dimensional optical fields via inverse mie scattering. *Science Advances*, 5(10):eaax4769, 2019.

[122] Dongkun Zhang, Lu Lu, Ling Guo, and George Em Karniadakis. Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems. *Journal of Computational Physics*, 397:108850, 2019.

[123] Yichen Zhang, Zhihai Wu, Jun Xia, Jun Wu, Keyang Yang, Chengkun Dong, Guodong Tong, Hao Zhang, Rui Yang, and Yong Luo. Infrared metasurface absorber based on silicon-based cmos process. *Opt. Express*, 30(18):32937–32947, Aug 2022.

[124] Maksym V. Zhelyeznyakov, Steve Brunton, and Arka Majumdar. Deep learning to accelerate scatterer-to-field mapping for inverse design of dielectric metasurfaces. *ACS Photonics*, 8(2):481–488, Feb 2021.

[125] Maksym V. Zhelyeznyakov, Steve Brunton, and Arka Majumdar. Deep learning to accelerate scatterer-to-field mapping for inverse design of dielectric metasurfaces. *ACS Photonics*, 8(2):481–488, 2021.

[126] Maksym V. Zhelyeznyakov, Johannes E. Froch, Anna Wirth-Singh, Jaebum Noh, Junsuk Rho, Steven L. Brunton, and Arka Majumdar. Large area optimization of meta-lens via data-free machine learning, 2022.

[127] Maksym V. Zhelyeznyakov, Alan Zhan, and Arka Majumdar. Design and optimization of ellipsoid scatterer-based metasurfaces via the inverse t-matrix method. *OSA Continuum*, 3(1):89–103, Jan 2020.

[128] Guoxing Zheng, Holger Mühlenbernd, Mitchell Kenney, Guixin Li, Thomas Zentgraf, and Shuang Zhang. Metasurface holograms reaching 80% efficiency. *Nature Nanotechnology*, 10(4):308–312, Apr 2015.

[129] Yinhao Zhu, Nicholas Zabaras, Phaedon-Stelios Koutsourelakis, and Paris Perdikaris. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, 394:56–81, 2019.

[130] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. 11 2019.

[131] Darko Zibar, Henk Wymeersch, and Ilya Lyubomirsky. Machine learning under the spotlight. *Nature Photonics*, 11(12):749–751, 2017.

Appendix A

THE ANGULAR SPECTRUM METHOD

For the sake of completeness and self-containment, in this section we will outline the angular spectrum method for propagating electromagnetic fields in a linear, homogeneous, isotropic, and non-dispersive medium. A more detailed derivation can be found in [39]. In a linear, homogeneous, isotropic, and non-dispersive medium, electromagnetic waves obey the wave equation [48]:

$$\nabla^2 u(x,y,z,t) - \frac{n^2}{c^2}\frac{\partial^2 u(x,y,z,t)}{\partial t^2} = 0 \tag{A.1}$$

where $u(x,y,z,t)$ is an arbitrary scalar field corresponding to any of the vector components of the electric or magnetic fields, (x,y,z) are spatial coordinates and t is the time coordinate, n is the refractive index of the medium, and c is the speed of light. Using separation of variables, we can obtain the time dependent solution $T(t) = A(x,y,z)\exp(-i2\pi\nu t)$, where ν is the frequency of the wave. Isolating the spatial dependence we can obtain the Helmholtz equation:

$$\nabla^2 U(x,y,z) + k^2 U(x,y,z) = 0 \tag{A.2}$$

where

$$k = \frac{2\pi\nu}{c}. \tag{A.3}$$

Eq. A.2 can be written as:

$$\nabla^2_{x,y} U(x,y,z) + \frac{d^2}{dz^2} U(x,y,z) + k^2 U(x,y,z) = 0. \tag{A.4}$$

Taking the Fourier transform of eq. A.4 along the x, y coordinates we get:

$$-k_x^2 \underset{\sim}{U}(k_x,k_y,z) - k_y^2 \underset{\sim}{U}(k_x,k_y,z) + \frac{d^2}{dz^2}\underset{\sim}{U}(k_x,k_y,z) + k^2 \underset{\sim}{U}(k_x,k_y,z) = 0 \tag{A.5}$$

where $\underset{\sim}{U}$ is the Fourier transform in the x, y coordinates, and k_x, k_y are the corresponding spatial frequencies. Since we want to obtain a transform from $U(x, y, 0)$ to $U(x, y, z)$, we assume at $z = 0$ the initial condition is $\underset{\sim}{U}(k_x, k_y, 0)$. We also know that in homogeneous, isotropic, linear, non-dispersive materials:

$$\nabla \cdot \vec{E} = 0. \tag{A.6}$$

Combining everything, we obtain the general solution to eq. A.5:

$$\underset{\sim}{U}(k_x, k_y, z) = \underset{\sim}{U}(k_x, k_y, 0) \exp\Big(i\sqrt{k^2 - k_x^2 - k_y^2}z\Big) \tag{A.7}$$

so in general for a field propagating in z,

$$U(x, y, z) = \mathcal{F}^{-1}\Big[\underset{\sim}{U}(k_x, k_y, 0) \exp\Big(i\sqrt{k^2 - k_x^2 - k_y^2}z\Big); (k_x, k_y)\Big]_{x,y} \tag{A.8}$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform from k_x, k_y to x, y. This can also be expressed as a convolution

$$U(x, y, z) = U(x, y, 0) \bigotimes \exp\Big(i\sqrt{k^2 - k_x^2 - k_y^2}z\Big). \tag{A.9}$$

Its straightforward to implement either A.8 or A.9 using any standard programming language. Furthermore, every operation is continuous and well defined, and can therefore be used with automatic differentiation.

Appendix B

APPENDIX FOR CHAPTER 3

B.1 Definition of efficiency metrics for the designed devices.

The standard definition of efficiency for a metasurface lens with a given focal length f is:

$$\eta = \frac{\int\int_{\Omega} \mathcal{E}^*(x,y,z=f)\mathcal{E}^*(x,y,z=f)dxdy}{\int\int_{x,y} \mathcal{E}^*(x,y,z=0)\mathcal{E}^*(x,y,z=0)dxdy} \tag{B.1}$$

$$\Omega := x^2 + y^2 < (3 \times FWHM)^2 \tag{B.2}$$

where Ω is the surface around the focal spot which we integrate over, and FWHM is the full width half maximum of a Gaussian fitted to the focal spot.

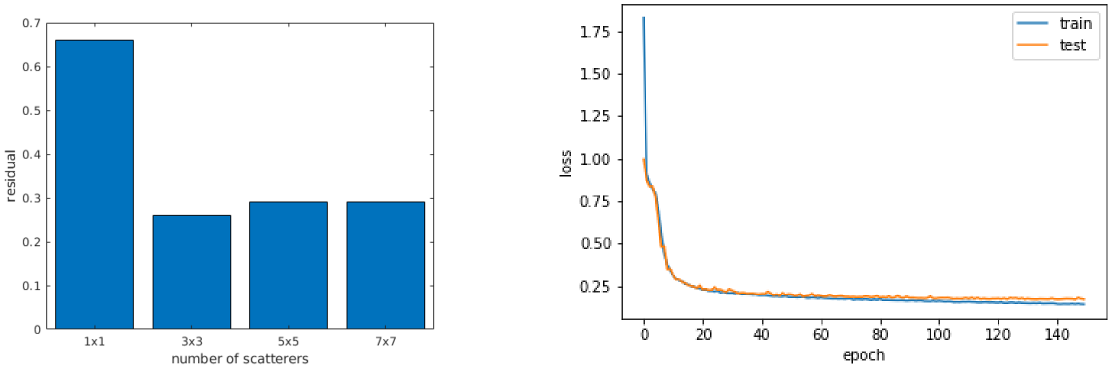
We quantify the functionality of the annular metasurface as the ratio between the power confined in the annulus to the power confined in the center of the annulus. More formally we define $\eta_\circ$ as:

$$\eta_\circ = \frac{\int\int_{\Omega_\circ} \mathcal{E}^*(x,y,z=F)\mathcal{E}^*(x,y,z=F)dxdy}{\int\int_{\Omega_.} \mathcal{E}^*(x,y,z=F)\mathcal{E}^*(x,y,z=F)dxdy} \tag{B.3}$$

$$\Omega_\circ := (r_\circ + \delta r)^2 < x^2 + y^2 < (r_\circ + \delta r)^2 \tag{B.4}$$

$$\Omega_t := x^2 + y^2 < \delta r^2 \tag{B.5}$$

Here, $\Omega_\circ$ is the surface representing the annulus and Ω_t is the surface representing the center of the annulus. $r_\circ$ is the radius of the annulus, defined in the optimization procedure as $1.5\,\mu m$. δr is the thickness over which we integrate, which we define as $\delta r = \frac{1}{2}$FWHM calculated for the $\lambda = 633$ nm case. $\eta_\circ$ was found to be equal to 58.47. All integrals are taken over the $\lambda = 400$ nm field. Another possible metric of interest would be the fraction of power contained inside the surface Ω_t, but in the $\lambda = 633$ nm case. We calculate this metric by switching Ω for Ω_t in Eq. B.1. This metric gives an efficiency of 10.55%.



(a) Model accuracy vs number of input features

(b) DNN loss

Figure B.1: **a.** Shows the model accuracy as number of features increases by including no nearby neighbors (1x1 block), nearby neighbors only (3x3 block), second order (5x5 block) and 3rd order (7x7) block nearest neighbors. **b.** Plot of the DNN objective function as a function of training epoch.

B.2 Data Driven deep neural network model and training

Throughout this work, we assumed that the majority of the nearfield coupling is accounted for by incorporating scattering from nearby pillars only. To test this, we created separate DNN models with single scatterer radius (1x1 block), nearby neighbors (3x3 block), as well as second order (5x5 block) and third order neighbors (7x7 block) as inputs. Fig. B.1a summarizes our results. The relative error between the predicted fields and the test field set is largest when only a single scatterer radius is included in the input. When the nearest neighbors are incorporated, the relative error drops significantly. Perhaps somewhat unexpectedly, including second and third order nearest neighbors increases the relative error in our test data set. This can be interpreted as the models over-fitting the fields to a larger number of inputs, and could be fixed by adding more data to the training.

Fig. B.1b shows the objective loss of the DNN model with 3x3 pillar radii as inputs vs the training epoch. Since around epoch 50, the test data set loss starts to diverge from the train data set loss, we use this point as our training cut-off to avoid overfitting. Fig B.2 shows the histograms of the field errors predicted from a set of radii by the DNN and the linear models. The error here is defined by:

$$||E_{FDTD}(x_i) - E_{pred}(x_i)||_2^2 \tag{B.6}$$

The histogram is tighter for the DNN model, thus showing it indeed performs better than the linear model we used.

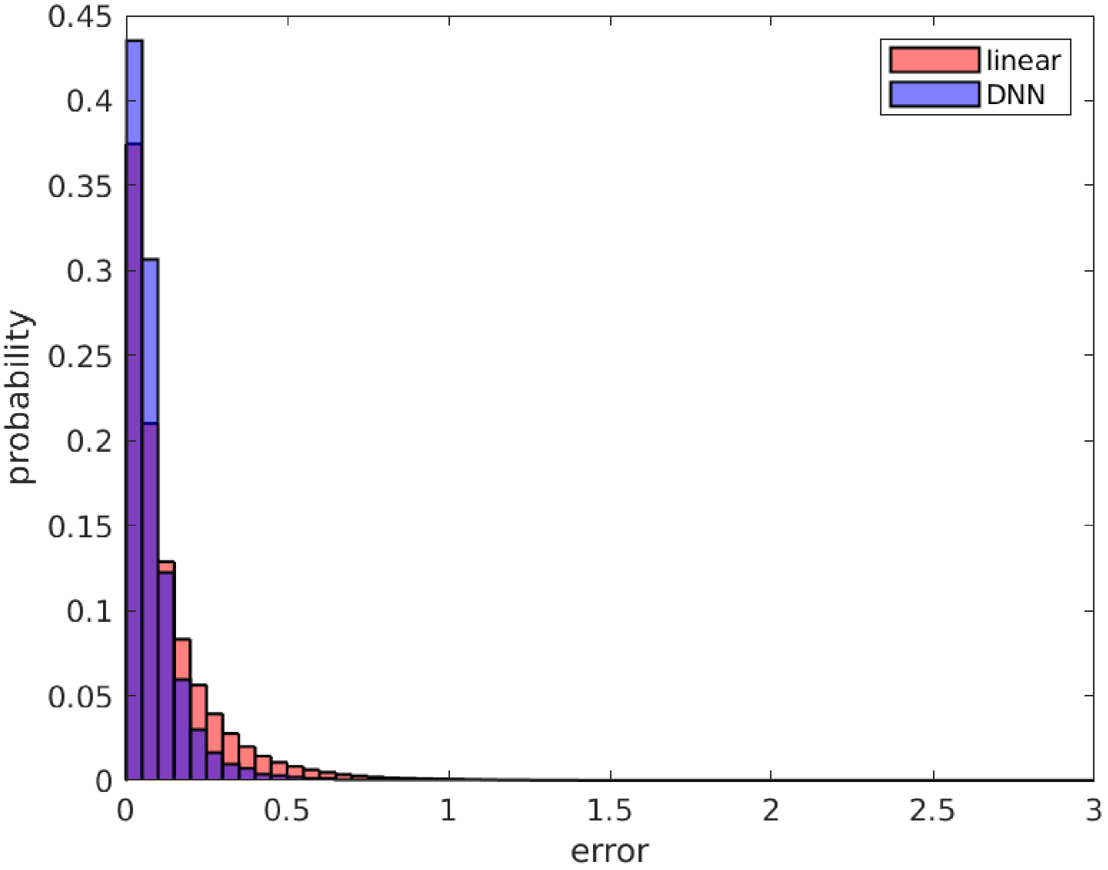


Figure B.2: Error histograms of the DNN and Linear models. Y axis is normalized to the probability of finding a certain error value

B.3 Inverse design with the DNN model

To inverse design both of the devices outlined in chapter 3, we use the DNN model as our forward simulation method to compute the nearfield response of the device. Then the nearfields are propagated to the focal plane of interest using the angular spectrum method implemented in TensorFlow. The final fields were used to compute the figures of merit for both of the devices.

In the case of the wavelength-multiplexed device, we used a forward designed lens that

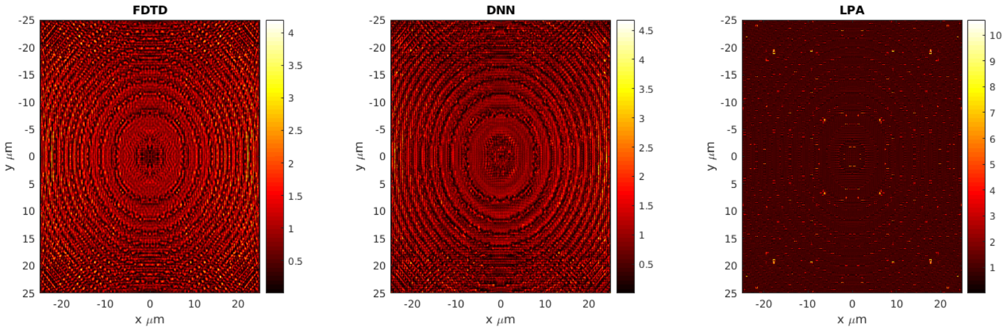


Figure B.3: Nearfields for the $f = 50\mu m$ lens computed with Lumerical FDTD (left), the DNN model (middle), and using RCWA under the local phase approximation (right).

focuses light at focal length $f = 50\mu m$ as the initial condition. Fig. B.3 shows the nearfields of the initial condition computed with three different methods: full FDTD simulation, our DNN model, and RCWA stitching under the LPA. The fields predicted by our method are much more similar to FDTD than those predicted with LPA.

Fig. B.4 summarizes the the design of the annular device. The left-most figure is the initial condition of the device (a $50\mu m$ focal length lens), the second shows the resulting, optimized device, and the third shows the difference between the two. On average, the mean absolute difference between the initial device and the final device is 29.5 nm with a standard deviation of 24.9 nm.

We terminate the optimization after 150 iterations, since the FOM improves only marginally after this point. The final value of the FOM is -1.99×10^3. The final FOM calculated via FDTD is -788.

Similarly, the EDOF device was designed by using a forward designed lens that focuses light at $f = 100\mu m$ as the initial condition. Fig. B.5 shows the nearfields calculated with the different methods. Fig. B.6 summarizes the optimization of the device.

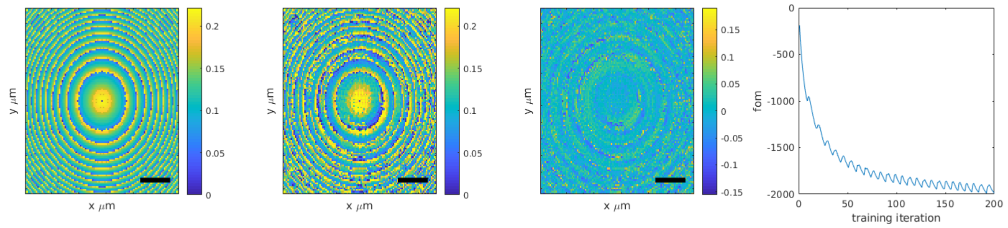


Figure B.4: From left to right we show: the initial device, the final designed device, the difference in radii between the initial device and the final device, and the figure of merit during optimization. The scale bars correspond to $10\mu m$.

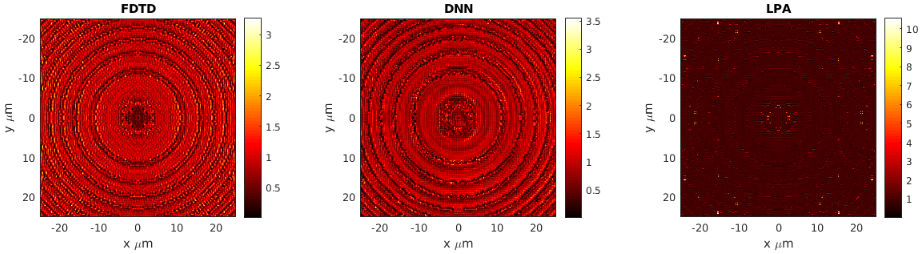


Figure B.5: Nearfields for the $f = 100\mu m$ lens computed with Lumerical FDTD (left), the DNN model (middle), and using RCWA under the local phase approximation (right).

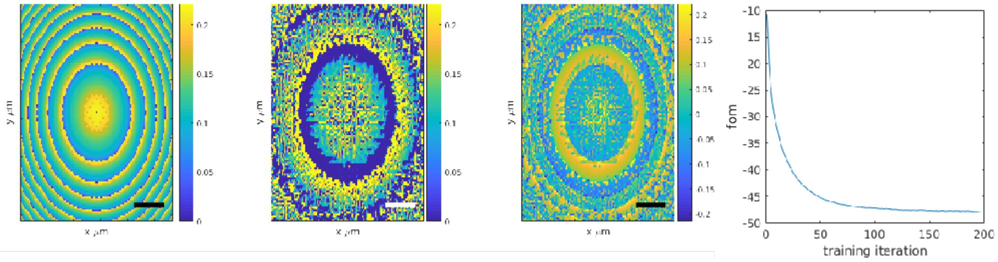
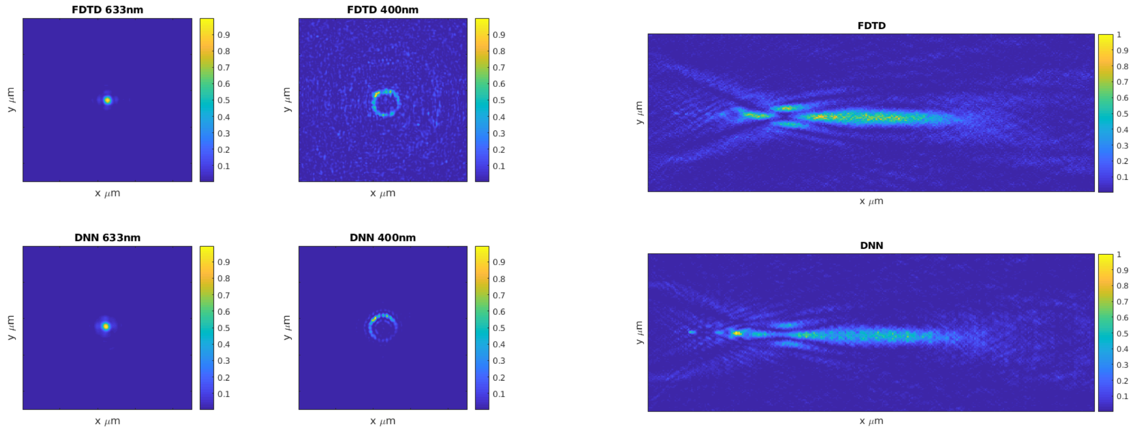


Figure B.6: From left to right we show: the initial device, the final designed device, the difference in radii between the initial device and the final device, and the figure of merit during optimization. The scale bars are $10\mu m$



(a) Wavelength switched device

(b) EDOF device

Figure B.7: **a.** Farfields of wavelength switched device and **b** EDOF device. Top is the field calculated via FDTD and bottom is the field calculated using our method

The final FOM value we calculate for the EDOF lens with our method is -48, vs the FDTD method -24. It is clear from the differences in the FOM's between FDTD and our method that our method does not capture all of the physics of the system. However, as shown from the farfields in Figs. B.7a and B.7b, this is still sufficient to perform inverse design of metasurfaces with alternate functionalities that are impossible to generate with forward design only.

B.4 Data-driven model performance at different wavelengths

We tested the DNN model's performance 10nm away from its operating wavelength $\lambda = 643nm$, and calculated the relative error on the test data set to be ~ 0.5. Then we inverse designed the same wavelength switched device as outlined in chapter 2, at $\lambda = 390nm$ for the annular beam and $\lambda = 643nm$ for the focal spot. Fig. B.8 shows the performance of this device. Note that even with a relative error that is twice than that of the original model, its robust enough for inverse design.

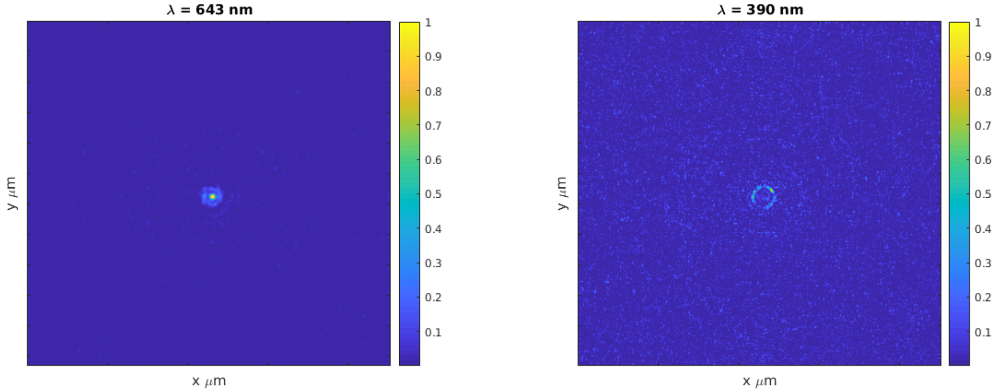


Figure B.8: inverse designed wavelength switched device at $\lambda = 643nm$ and $\lambda = 390nm$

Appendix C

APPENDIX FOR CHAPTER 4

C.1 Problem set up

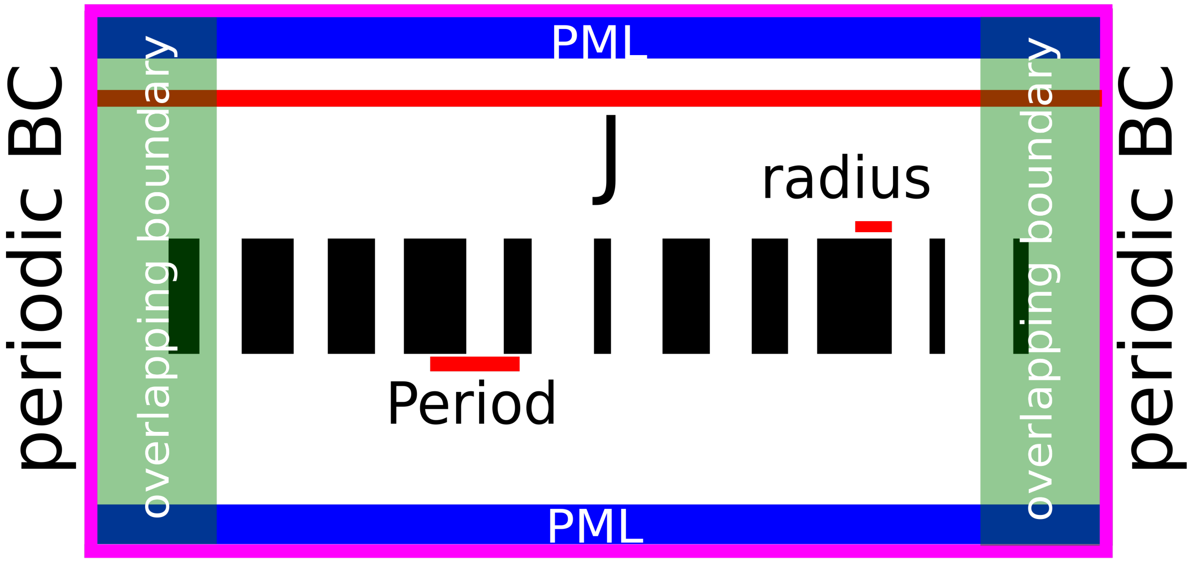


Figure C.1: Simulation problem setup

The neural networks are trained to predict electric field responses of distributions of dielectric scattereres, here SiN pillars, from a plane wave current source of wavelength $\lambda = 633nm$. The resolution is set to be 16 pixels per period, with each period being 0.443μm. The boundary conditions along the x-direction are set to be periodic. The boundary conditions along the y-direction are set to be 10 grid points of a perfectly matched layer (PML). The simulation domain in the x direction is 12×period = 5.316μm, and 6× period = 2.658μm along the y direction. When simulating field responses of large area metasurfaces, the simulation region is split up into groups of 11 periods with the outermost periods overlapping. Each region is simulated with a trained neural network as shown in C.1. The field responses of the innermost 9 pillars are stitched together.

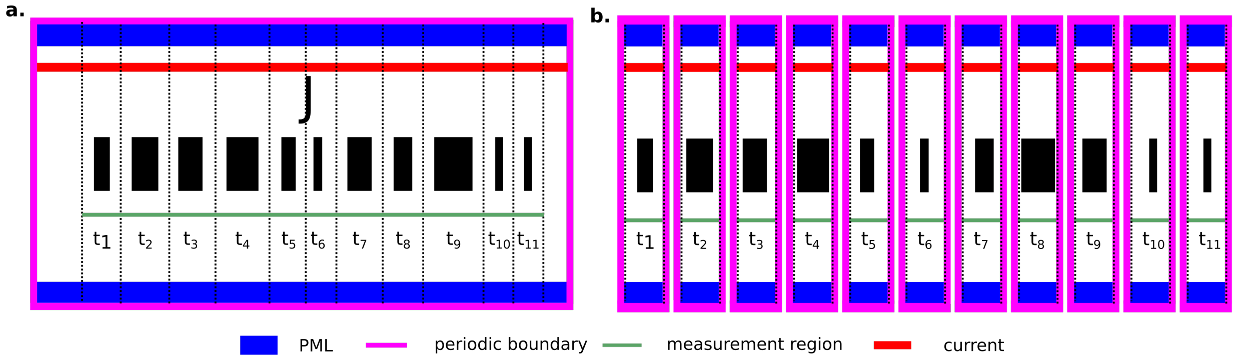


Figure C.2: Calculating pillar-wise transmission coefficients for **a.** FDFD and neural network simulations and **b.** LPA simulations.

C.2 Pillar-wise transmission coefficient error

Here, we summarize how we compute the average pillar-wise transmission error (i.e. main text Figure 2 **b** right hand side). For every set of 11 pillars we compute the transmission coefficients for each pillar using FDFD, PINNs, and LPA. Fig. C.2 (left) shows how transmission coefficients for each batch are computed for FDFD and PINN. Fig. C.2 (right) shows how transmission coefficients are computed under the LPA. For the FDFD and PINN case, we simulate the full field over the simulation region defined in the previous section. Then we measure field 6 pixels (0.13μm) away from the meta-atom. Then that field is averaged over a single period corresponding to the location of the meta-atom. This gives us 11 transmission coefficients. For the LPA, we simply simulate the 11 scatterers under periodic boundary conditions, and extract the average field the same distance away from each pillar. The mean transmission coefficient error is thus given by

$$\frac{1}{11}\sum_{i=1}^{11}|t_i^{FDFD} - t_i^{\text{approx}}|^2 \tag{C.1}$$

where t_i^{FDFD} is the transmission coefficient computed with FDFD and t_i^{approx} is computed by either the neural network or LPA.

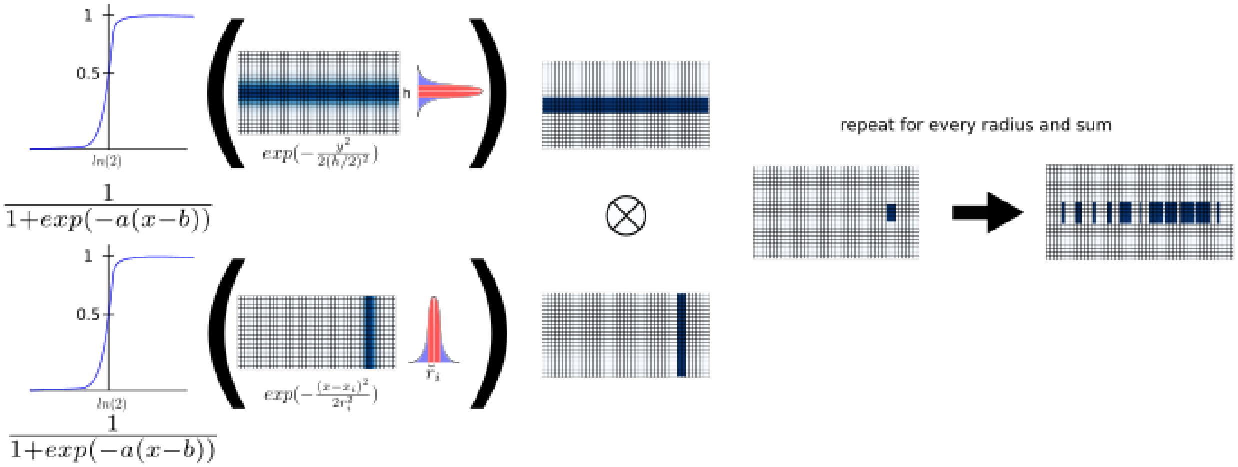


Figure C.3: transforms carried out to make a differentiable map from $r \to \epsilon$

C.3 Mesh reparametrization

To mesh a set of pillars, we first generate a grid of coordinates $x \in [-6p, 6p]$ and $y \in [-\frac{3}{2}p, \frac{3}{2}p]$, p being the pitch of the meta-optics. Since we have a fixed set of pillars, their centroid locations are given by $x_i = \frac{np}{2}$ where n is the pillar index running from -5 to 5. The first transform we define is just a shifted gaussian function on this grid:

$$T_1(r_i, x_i) = \exp\big(-\frac{(x-x_i)^2}{2r_i^2}\big) \tag{C.2}$$

Similarly, the second one defines the height:

$$T_2 = \exp\big(-\frac{y^2}{2(h/2)^2}\big) \tag{C.3}$$

The modified softmax is defined by:

$$T_3(a, b, x) = \frac{1}{1+\exp(-a(x-b))} \tag{C.4}$$

a denotes the aggressiveness of the softmax function, and b is the point above which the function goes to 1, and below which the function goes to 0. Here, we chose $a = 100$ and

$b = \log_e(2)$. The aggressiveness was experimentally determined, with larger values causing gradients to become too steep, and lower values causing things to not resemble pillars as much. $b = \log_e(2)$ is chosen because a gaussian function drops to $\log_e(2)$ of its max values after 1 standard deviation of its input variable, hence creating pillars of size r. Thus, 1 meshed batch of radii can be written as:

$$\sum_{i=1}^{11} T_3(T_1(r_i, x_i), 100, \log_e(2)) T_3(T_2, 100, \log_e(2)) \tag{C.5}$$

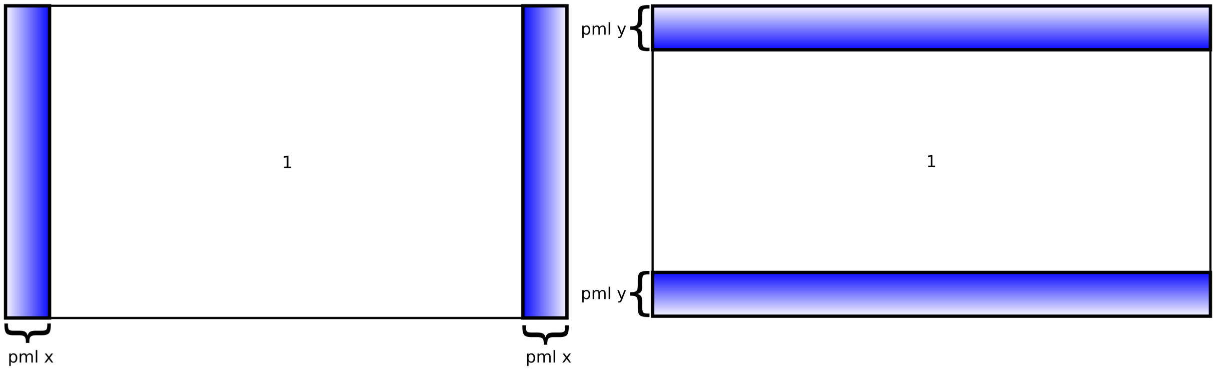


Figure C.4: Description of setup for scale matrices S_x and S_y. Left hand side is for the x derivative and right hand side is for the y derivative. PML scaling is computing on a meshed grid, then flattened and embedded in a diagonal matrix.

C.4 Form of FDFD linear system

In this section we give a brief summary of formulating the FDFD linear system on a Yee grid for completeness. The full details of the method can be found elsewhere [1, 2, 3, 4, 5], so we only give a brief description here. The boundary conditions of Maxwell's equations are defined at $|x| \to \infty$, so perfectly matched layer (PML) boundary conditions must be implemented to truncate the simulation region to a finite size. To do this, scale matrices S_x and S_y need to be generated. They are constructed by creating scale factors

$$s_w(l) = \begin{cases} 1 - i\frac{\sigma_w(l)}{\omega_o \epsilon_0} & \text{inside } w \text{ normal pml} \\ 1 & \text{otherwise} \end{cases} \tag{C.6}$$

Where w is the coordinate normal (x or y), l is the distance inside the PML from the PML interface, ω_o is the operating angular frequency, and ϵ_0 is the permittivity of free space. $\sigma_w(l)$ is given by

$$\sigma_w(l) = \sigma_{w,\max}\left(\frac{l}{d}\right)^m \tag{C.7}$$

and $\sigma_{w,max}$ is

$$\sigma_{w,\max} = \frac{(m+1)\ln \mathrm{R}}{2\eta_0 d} \tag{C.8}$$

d is the thickness of the PML. η_0 is the vacuum impedance $\eta_0 = \sqrt{\mu_0/\epsilon_0}$. R is the target reflection coefficient. A good reference for PML boundaries is Shin et. al.[4]. The package we used, angler[3], uses the convention $m = 4$ and $\ln(\mathrm{R}) = -12$. Furthermore we modified the constants inside the package such that $\mu_0 = \epsilon_0 = \eta_0 = 1$. The matrices S_x and S_y are created by computing s_w on a meshed grid, flattening it, and embedding it into a diagonal matrix:

$$S_w = \begin{bmatrix} \frac{1}{s_w^{1,1}} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{s_w^{2,2}} & 0 & \cdots & 0 \\ 0 & 0 & \frac{1}{s_w^{3,3}} & \cdots & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & \cdots & \frac{1}{s_w^{N,N}} \end{bmatrix} \tag{C.9}$$

The numerical derivative matrices on a yee grid, with periodic boundary conditions Δ_x and Δ_y are:

$$\boldsymbol{\Delta}_{\mathbf{x}} = \begin{matrix} & \overbrace{\hphantom{-\tfrac{1}{dx}\quad -\tfrac{1}{dx}\quad \ddots\quad -\tfrac{1}{dx}}}^{N_y} \\ (N_x-1)N_y\left\{\vphantom{\begin{matrix}\tfrac{1}{dx}\\\tfrac{1}{dx}\\\ddots\\\tfrac{1}{dx}\\\tfrac{1}{dx}\end{matrix}}\right. & \begin{bmatrix} -\frac{1}{dx} & & & \frac{1}{dx} & & & & \\ & -\frac{1}{dx} & & & \frac{1}{dx} & & & \\ & & \ddots & & & \ddots & & \\ & & & -\frac{1}{dx} & & & \frac{1}{dx} & \\ & & & & -\frac{1}{dx} & & & \frac{1}{dx} \\ \frac{1}{dx} & & & & & -\frac{1}{dx} & & \\ & \ddots & & & & & \ddots & \\ & & \frac{1}{dx} & & & & & -\frac{1}{dx} \end{bmatrix} \end{matrix} \tag{C.10}$$

$$\mathbf{\Delta_y} = \begin{array}{cc} & \overbrace{\hphantom{-\frac{1}{dy}\quad\frac{1}{dy}\quad\frac{1}{dy}\quad\ddots\quad-\frac{1}{dy}}}^{N_y} \\ N_y\left\{\vphantom{\begin{matrix}-\frac{1}{dy}\\-\frac{1}{dy}\\\ddots\\\frac{1}{dy}\end{matrix}}\right. & \\ & \end{array} \begin{bmatrix} -\frac{1}{dy} & \frac{1}{dy} & & & & & & & & \\ & -\frac{1}{dy} & \frac{1}{dy} & & & & & & & \\ & & \ddots & \ddots & & & & & & \\ \frac{1}{dy} & & & -\frac{1}{dy} & 0 & & & & & \\ & & & & -\frac{1}{dy} & \frac{1}{dy} & & & & \\ & & & & & -\frac{1}{dy} & \frac{1}{dy} & & & \\ & & & & & & & \ddots & \ddots & \\ & & & & & & \frac{1}{dy} & & & -\frac{1}{dy} \end{bmatrix} \tag{C.11}$$

here N_x and N_y are the grid sizes in the x and y directions and dx and dy are the grid spacings. The derivative matrices D_x^e, D_y^e, D_x^h, and D_y^h can then be constructed as:

$$D_x^e = S_x \Delta_x \tag{C.12}$$

$$D_y^e = S_y \Delta_y \tag{C.13}$$

$$D_x^h = S_x(-\Delta_x^{\dagger}) \tag{C.14}$$

$$D_y^h = S_y(-\Delta_y^{\dagger}) \tag{C.15}$$

where the † operator is the Hermitian transpose.

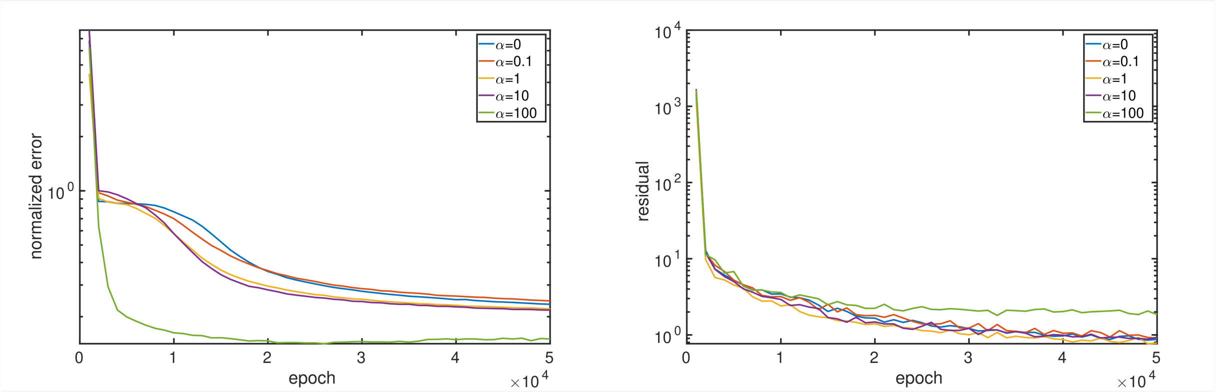


Figure C.5: Comparison of neural network performance for different values of α in the data loss term. **Left:** epoch vs normalized error given by eq. C.19 and **right:** the residual. Both both plots are the done on the test data set.

C.5 Neural network training

In this section we describe the effect of adding data to the PINN loss function. Given a PINN loss:

$$f(\varepsilon;\theta) = \left|\left|\left[D_x^h D_x^e + D_y^h D_x^e + \omega_o^2 \varepsilon\right] NN(\varepsilon;\theta) + i\omega_o J\right|\right|_1 \tag{C.16}$$

and a data loss

$$g(\varepsilon;\theta) = \left|\left|E_{FDFD} - E_{NN}(\varepsilon;\theta)\right|\right|_1 \tag{C.17}$$

we can form a total loss function:

$$h(\varepsilon;\theta) = f(\varepsilon;\theta) + g(\varepsilon;\theta) \tag{C.18}$$

Where ε is the input dielectric distribution, θ are the trainable parameters of the neural network, and α is the ”strength” of the data term. Fig. C.5 shows the test loss vs epoch of neural networks trained with various α parameters. We generated a dataset of 10000 fields by directly simulating our problem with random pillar arrangements using angler [3]. The training was done on 9900 fields, and the test was done on the remaining 100. The left hand side of Fig. C.5 shows the accuracy of the neural network given by the 2 norm relative error between the fields simulated by FDFD and the neural net:

$$\frac{||E_{FDFD} - NN||_2^2}{||E_{FDFD}||_2^2} \tag{C.19}$$

on the test data set. The right hand side shows the same information except for the residual given by C.16. We note that, while some data parameter α may marginally improve how well the neural network predicts a field, for values of α where this improvement becomes significant, the physics informed loss takes a penalty. We argue that having a lower physics informed loss is a much better scenario for optimization problems because it means the fields satisfy the PDE better, and thus more accurately represent the predicted physical quantity. Furthermore, the improvement from including a data term is marginal, and increases the complexity of neural network training, that it is not worth adding.

C.6 Extended depth of focus (EDOF) Lens

We also designed an EDOF lens through a standard max-min objective approach, where we computed the intensity of the field produced by our lens at discrete equidistantly spaced points along a focal line, on an interval between two different focal lengths:

$$\begin{aligned} f &= \{f_1, f_2, ..., f_{10}\} \\ f_i &= E^\dagger(0, z_i)E(0, z_i) \\ &\max_{r} \min_{f_i} f \end{aligned} \tag{C.20}$$

We then used forward design to generate a lens with focal length 2050μm and a diameter of 1000μm, and optimized an EDOF lens to extend its focus between 2000 and 2100 microns. Fig. C.6 shows a summary of the results for the inverse designed EDOF lens. To characterize the performance of the lens, we define the depth of focus as the point where the distance at which the focal spot intensity reaches 1/2 of its original value, see Fig. C.6 c and f. The EDOF device has a theoretically predicted depth of focus of 93μm and an experimentally measured depth of focus of 97μm. The depth of focus of the forward designed lens is $\sim$20μm measured both in experiment and theory.

C.7 Failed designs and the lack of experimental results

We want to emphasize the importance of experimentally verifying inverse design methods. Initially we designed our inverse design method by extracting derivative matrices from angler [3], and comparing our results to the FDFD results from the same package in order to make sure our results were consistent and accurate. Fig. C.7 summarizes our results. We note there there is a large discrepancy between the theory and experiment here. This is due to the fact that angler and our first iteration of our inverse design methodology was predicted the complex conjugate of the electric field instead of the electric field, leading to angular spectrum propagation to propagate the fields in the opposite direction. We would never have realized this error if we did not conduct experimental testing of our devices. We encourage more inverse design papers in the future to manufacture and test their inverse design methods in order to produce reliable and accurate designs.

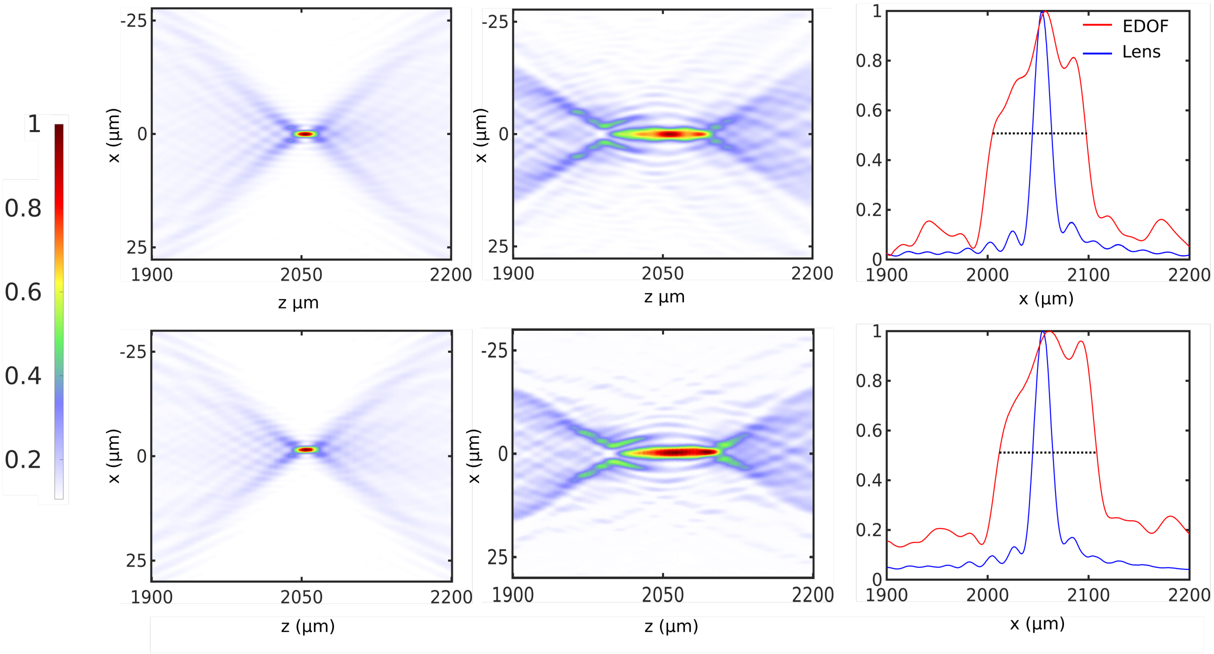


Figure C.6: EDOF lens inverse design. **a. - c.** represent theoretical results, and **d. - f.** are experimentally measured results. **a., d.** are forward designed lenses with focal length 2.05 mm. **b.,e.** are the optimized EDOF lenses. **c., f.** are slices along the z axis with $x = 0\mu$m. The red lines are the EDOF and the blue lines are the lens. The black line is plotted at the full width half maxima of the EDOF lens, which is how we define the depth of focus[5]. The theoretical and experiment depth of focus for the forward designed lens is 20μm. The EDOF lens has a theoretical depth of focus of 93μm and an experimentally measured depth of focus of 97μm

.

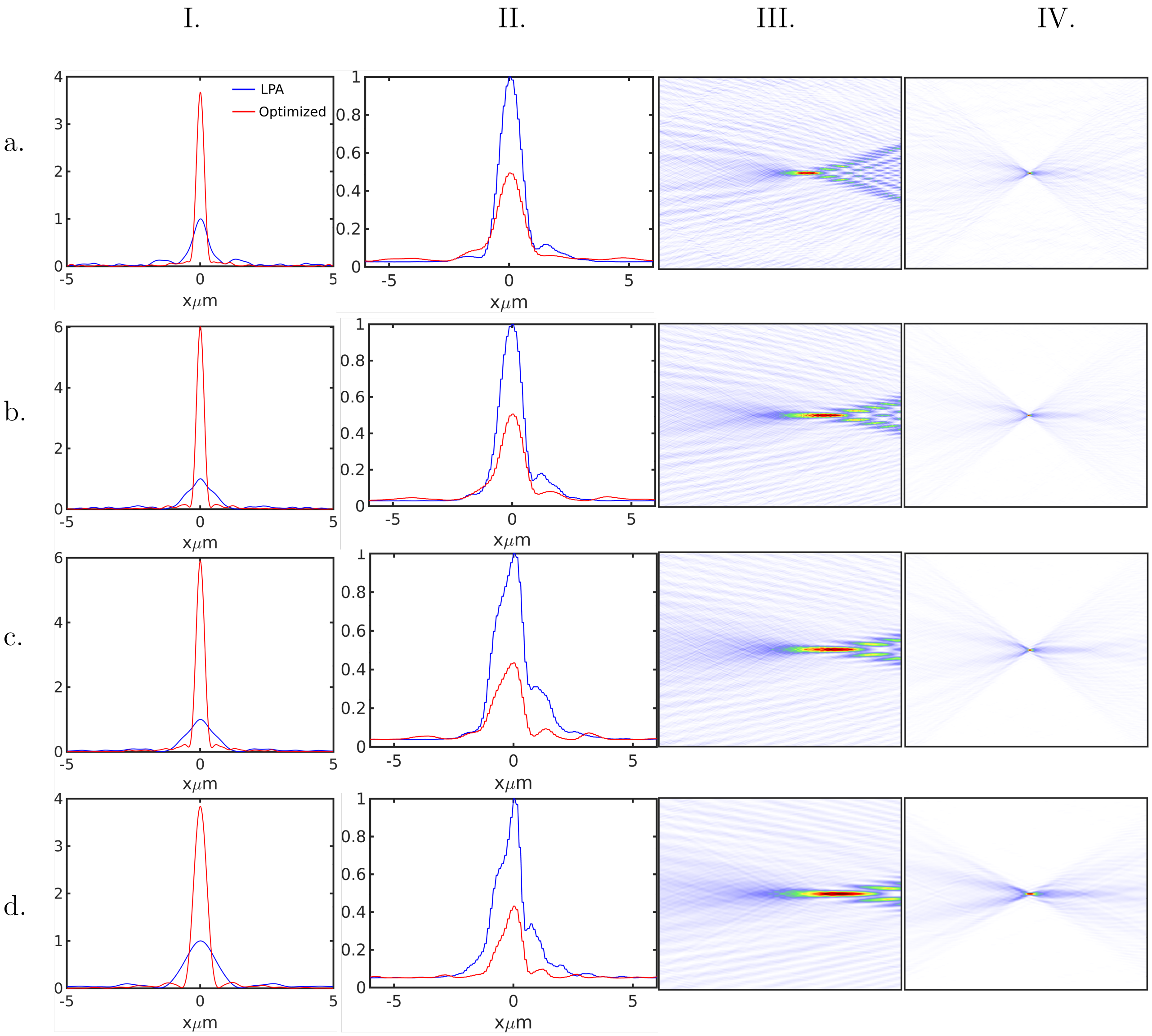


Figure C.7: **I** Theoretically predicted intensities at the focal spot. **II.** Experimentally measured intensities at the focal spot. **III.** Theoretically predicted focal spot for forward design **IV** and inverse design. **a.** 200 μm focal length. **b.** 500 μm **c.** 750 μm **d.** 1000μm.

C.8 Simulation resource and speed comparisons

	PINN				FDFD	
	Batch Size 1	Batch Size 10	Batch Size 20	Batch size 40	Overlapping BCs	Full Simulation
average time per chunk (s)	0.006	0.004	0.004	0.005	0.021	
total sim time (s)	1.17	1.04	1.17	1.13	5.29	11.43
Ram Usage (GB)	5.29	5.29	5.29	5.29	1.77	36.7
GPU Memory (GB)	1.62	2.88	4.58	7.17		

Table C.1: Resource and speed comparisons between the PINN approach and the FDFD approach for the forward simulation of a 1mm lens. The PINN approach is about 5x faster than the FDFD approach when overlapping boundary conditions are used and 10x faster when we don't use overlapping boundary conditions. Furthermore, the overlapping boundary method is more memory efficient, and thus useful for running inverse design on machines with low RAM.

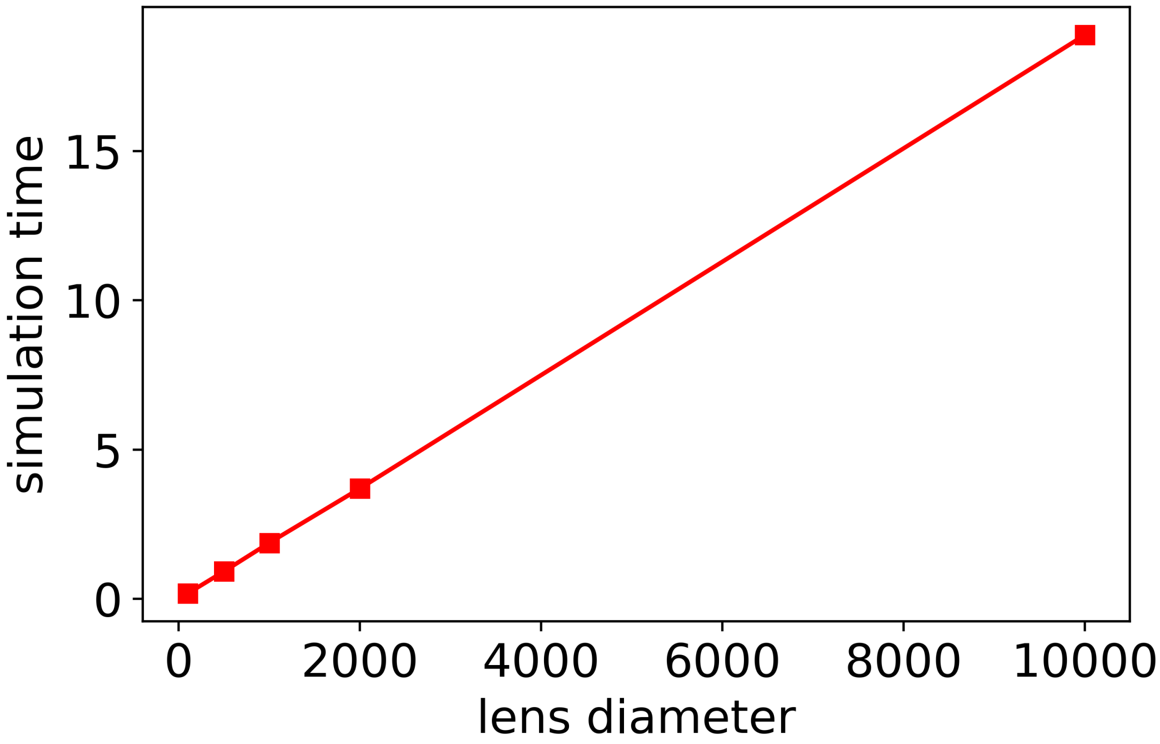


Figure C.8: Comparison of neural network simulation time vs lens diameter. Data is taken for lens diameters of 100, 500, 1000, 2000, and 10000 μm lens diameters. Simulation time increases linearly with lens diameter.

C.9 LPA Data

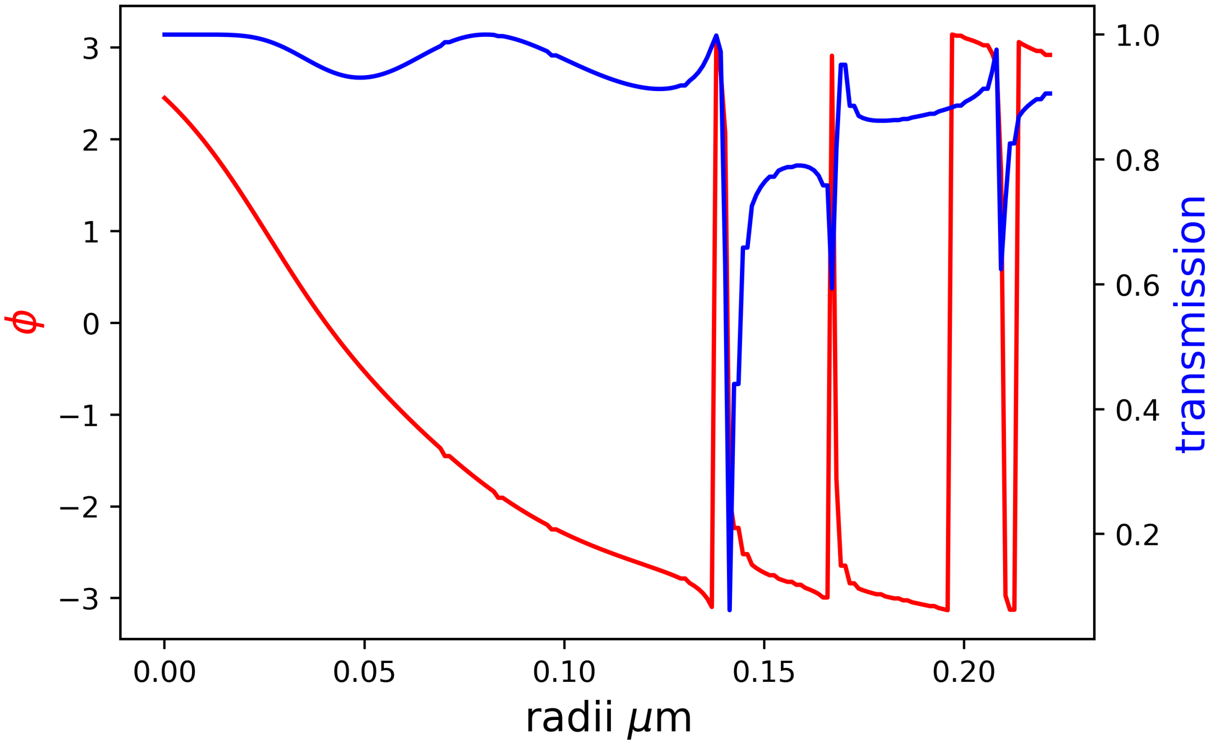


Figure C.9: Phase and amplitude transmission for the pillars with height $h = 0.6$, refractive index $n = 2$, and periodicity 0.443μm at operating wavelength $\lambda = 0.633\mu$m

BIBLIOGRAPHY

[1] Rumpf, R. Electromagnetic and photonic simulation for the beginner: Finite-difference frequency-domain in MATLAB (R); Artech House: Norwood, MA, 2022

[2] Rumpf, R. C.; Garcia, C. R.; Berry, E. A.; Barton, J. H. Finite-difference frequency-domain algorithm for modeling electromagnetic scattering from general anisotropic objects. Prog. Electromagn. Res. B Pier B 2014, 61, 55–67

[3] Hughes, T. W.; Minkov, M.; Williamson, I. A. D.; Fan, S. Adjoint method and inverse design for nonlinear nanophotonic devices. 201

[4] Shin, W.; Fan, S. Choice of the perfectly matched layer boundary condition for frequency-domain Maxwell's equations solvers. Journal of Computational Physics 2012, 231, 3406–3431

[5] Bayati, E.; Pestourie, R.; Colburn, S.; Lin, Z.; Johnson, S. G.; Majumdar, A. Inverse Designed Metalenses with Extended Depth of Focus. ACS Photonics 2020, 7, 873–878. 17