BLASCHKE PRODUCT NEURAL NETWORK (BPNN): A PHYSICS-INFUSED NEURAL NETWORK FOR PHASE RETRIEVAL OF MEROMORPHIC FUNCTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Numerous physical systems are described by ordinary or partial differential equations whose solutions are given by holomorphic or meromorphic functions in the complex domain. In many cases, only the magnitude of these functions are observed on various points on the purely imaginary $j\omega$-axis since coherent measurement of their phases is often expensive. However, it is desirable to retrieve the lost phases from the magnitudes when possible. To this end, we propose a physics-infused deep neural network based on the Blaschke products for phase retrieval. Inspired by the Helson and Sarason Theorem, we recover coefficients of a rational function of Blaschke products using a Blaschke Product Neural Network (BPNN), based upon the magnitude observations as input. The resulting rational function is then used for phase retrieval. We compare the BPNN to conventional deep neural networks (NNs) on several phase retrieval problems, comprising both synthetic and contemporary real-world problems (e.g., metamaterials for which data collection requires substantial expertise and is time consuming). On each phase retrieval problem, we compare against a population of conventional NNs of varying size and hyperparameter settings. Even without any hyper-parameter search, we find that BPNNs consistently outperform the population of optimized NNs in scarce data scenarios, and do so despite being much smaller models. The results can in turn be applied to calculate the refractive index of metamaterials, which is an important problem in emerging areas of material science.

1 INTRODUCTION

Numerous physical systems are described by ordinary or partial differential equations whose solutions are given by holomorphic or meromorphic functions $f(z) = |f(z)| \exp(j\theta(z))$ in the complex domain, where $|f(z)|$ and $\theta(z)$ are respectively the magnitude and phase of $f(z)$. In many cases, only $|f(j\omega)|$, the magnitude of $f(z)$, is observed on a finite set of points (Ω) of the purely imaginary $j\omega$-axis, since the coherent measurement of their phases may be expensive (see Ch. 13 of Stern (2016)) or require significant expertise (King, 2009). However, it is desirable to retrieve the lost phase $\theta(j\omega)$ from measurements of the magnitudes $|f(j\omega)|$ for $\omega \in \Omega$, since the phase often encodes important information. For example in both spectroscopy and optical imaging, estimation of $\theta(j\omega)$ is often not possible. However, phase information determines the optical constants in the spectroscopy of materials (Lucarini, 2005), and is more important than $|f(j\omega)|$ in optical imaging (Stern, 2016). The phase retrieval problem is also important in a number of other disciplines, including holography (Fienup, 1982), x-ray crystallography (Nugent, 2007), astronomy (Stark, 1987), blind deconvolution (Shechtman et al., 2015)), and coherent diffractive imaging (Latychevskaia, 2018).

Many techniques for phase retrieval have been developed in the literature. The Kramers–Kronig formula, see Lucarini (2005) (also known as Sokhotski–Plemelj formula), is noteworthy and is derived by the application of the Hilbert Transform to $\log f(j\omega) = \log |f(j\omega)| + j\theta(j\omega)$. Unfortunately, for phase retrieval, this elegant formula requires the knowledge of the magnitude $|f(j\omega)|$ along the entire $j\omega$-axis. However, as mentioned above, acquiring these values is often only possible only when Ω is a finite set with a small number of elements. This severely limits the application of Kramers–Kronig's formula in many applications. Another approach is based on the compressed sensing, but it requires sparsity assumptions, and stringent assumptions on the design or projection

matrix (i.e., Restricted Isometry Property (RIP) (Candes, 2008), mutual incoherence (Foucart & Rauhut, 2013), etc) that often do not hold for meromorphic functions encountered in practice.

To overcome the above limitations, phase retrieval can also be cast as a data-driven regression problem, where we wish to learn a mapping from the magnitude of a signal, $X \in R^N$, to its phase, $\Theta \in \mathbb{C}^N$. If the number of frequencies ($|\Omega|$) measured is N then the phase can be represented as a vector of complex numbers $\Theta = [\exp(\theta(j\omega_1)), \cdots, \exp(\theta(j\omega_N)] \in \mathbb{C}^N$, where $\exp(\theta(j\omega_i))$ is the phase at i-th frequency. Similarly, the signal's corresponding magnitude vector can be given by $|f(j\omega_i)|, i = 1, 2, \cdots, N$, and the goal is to infer Θ based upon X. We can then learn this mapping using a data-driven model (Please see for instance Figure 1 below for an example).

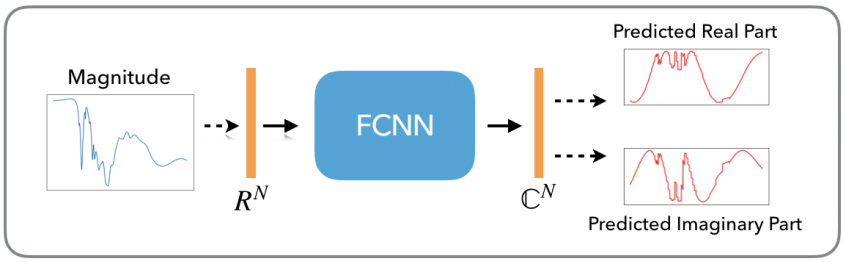


Figure 1: Example of the deep neural network approach to phase retrieval problem. FCNN refers fully-connected neural network here but can be replaced by any machine learning model. The magnitude is encoded into an input of size N while the output is an N-dimensional complex vector of corresponding phases.

With a sufficient amount of training data, it is likely that a universal function approximator (i.e. a neural network), can learn the underlying function well within a tolerance range. For instance, deep learning has shown great success on phase retrieval problems in the imaging area where abundant training data is available (Sinha et al., 2017; Metzler et al., 2018). But sufficient training data can be expensive to acquire in many practical applications.

In this paper, we depart from the existing literature by exploring the applications of physics-infused neural networks to this setting. The infusion of physics reduces the number of training examples to a small enough amount that is practical for many measurements and design applications of interest. The main idea of the approach is described next. Obviously $|\exp(j\theta(j\omega))| = 1$ for any value of the phase $\theta(j\omega)$. We observe that conditions of a Theorem of Helson and Sarason are then satisfied indicating that $\exp(j\theta(j\omega))$ can be approximated by a rational function whose numerator and denominator are Blaschke products (Garcia et al., 2016). This is in turn used to design a neural network for calculating these Blaschke products which in turn can be used for phase retrieval.

The outline of this paper is given next. In Section 2, we briefly review conventional phase retrieval methods, and existing applications of the neural network to phase retrieval problems. In Section 3, we briefly review Blaschke products and the Helson-Sarason Theorem. In Section 4, we propose our approach for baking these results into neural networks resulting in Blaschke Product Neural Networks. In Section 5, we provide experimental results on various synthetic and metamaterials datasets demonstrating the performance improvements of our construction.

2 Related Work

We first briefly review existing methods of phase retrieval applicable to phase retrieval of analytic and meromorphic functions of interest.

Perhaps the most important result in this area is the elegant calculation of Kramers and Kronig (also known as Sokhotski–Plemelj formula) obtained by the application of the Hilbert Transform to $\log f(j\omega) = \log |f(j\omega)| + j\theta(j\omega)$ and is given by

$$\theta(j\omega) = \frac{2\omega}{\pi} P.V. \int_0^\infty \frac{\log |f(j\beta)|}{\beta^2 - \omega^2} d\beta,$$

where P.V. denotes Cauchy's principal value. We note that this can also be derived by invoking the Cauchy integration formula along a path containing the imaginary $j\omega$-axis.

However, there is a major obstacle in the application of this elegant formula. In many important practical scenarios, the set Ω of frequencies (points on $j\omega$-axis) that we have measurements of $|f(j\omega)|$ has a small cardinal number. In such cases, the approximation of Kramers-Kronig integral by a Riemann Sum produces unsatisfactory results. Other approaches used to analytically extend or extrapolate $|f(j\omega)|$ have also demonstrated limited success (Chalashkanov et al., 2012).

Another approach to phase retrieval is by sparse representation. This approach assumes that it is a priori known that $f(z)$ has a sparse representation in a known dictionary of functions $\mathbb{D} = \{\Phi_i(z), i = 1, 2, \cdots, M\}$ i.e. it can be written as a linear sum of elements of $\mathbb{D}$ with only a small number of non-zero coefficients. Under this assumption, many techniques are developed to recover these coefficients and the phase of $f(z)$. Again, this is an interesting proposal but the underlying assumptions may not hold in many problems of interest. In particular, neither a dictionary $\mathbb{D}$ may be known nor classes of functions of interest may be sparse in any fixed dictionary. Recently, some new techniques based on deep learning have been developed to learn the structure of the underlying signal which we want to recover its phase. That is, deep neural networks are used to encode the prior knowledge on the structure of the signal instead of hard-coded assumptions such as sparsity (Jagatap & Hegde, 2019). However, these methods require a significant number of training samples from the magnitude of the underlying signal, which limits the applicability of these approaches on material science applications.

Recently, DL has demonstrated its capability to solve the phase-retrieval problem, especially in the imaging area: Sinha et al. (2017) achieved lensless imaging for phase objects utilizing CNN. Xue et al. (2019) constructed a bayesian convolution neural network to achieve phase imaging while quantifying the uncertainty of the network output. Metzler et al. (2018) combined regularization-by-denoising framework with a convolution neural network for noisy phase retrieval for images. Goy et al. (2018) achieved low photon count phase retrieval with a CNN as well. To avoid the difficulty of getting a large set of paired phase and intensity measurements, Zhang et al. Zhang et al. (2021) used an unpaired dataset for phase-retrieval. Other works of Tayal et al. (2020); Houhou et al. (2020) have also used deep learning for end-to-end phase recovery. None of these works use the underlying physics in their deep learning approach or they require a large number of training samples; hence, are not directly applicable to the physics applications we are interested in this paper. We took a different path of applying physical prior knowledge to our phase retrieval process.

One closely related line of work to ours is solving partial differential equations using neural networks (Dissanayake & Phan-Thien, 1994; Berg & Nyström, 2018; Zhang & Bilige, 2019; Chen et al., 2018). However, these methods typically need a lot of data and require careful selection of hyper-parameters; plus, they do not consider directly solving the phase retrieval problem. Our approach, on the other hand, needs no specific hyper-parameter search algorithm, and achieves high-accuracy result with a very limited training samples.

3 Mathematical Background

We briefly review the mathematical background required in the rest of this paper. In particular, we review Blaschke products and the Helson-Sarason Theorem. We note that the standard presentation of the Blaschke products is motivated by inner functions in Hardy spaces presented over the unit disk whose boundary is the unit circle (Rudin, 1987; Garcia et al., 2016). In our case, we are interested in right hand half plane whose boundary is the $j\omega$-axis. These two are equivalent using the complex Mobius transformation $(z+1)/(1-z)$. In this light, our exposition is closer to Akutowicz (1956).

Let $a_1, a_2, a_3, \cdots$ be an infinite sequence of numbers for which $\Re(a_k) > 0$ for all k for which

$$\sum_{j=1}^{\infty} \frac{\Re(a_j)}{|a_j|^2 + 1} < \infty,$$

then for any integer n, any complex function given by

$$B(z) = \left(\frac{1+z}{1-z}\right)^n \prod_j \frac{|a_j - 1|}{a_j - 1} \frac{|a_j + 1|}{a_j + 1} \frac{z - a_j}{z + \bar{a}_j}$$

is referred to as a infinite Blaschke Product. If the set $\{a_1, a_2, \cdots, a_k\}$ is finite, $B(z)$ is referred to as a finite Blaschke product. For a finite Blaschke Product, it is clear that $\sum_{j=1}^{k} \frac{\Re(a_j)}{|a_j|^2+1} < \infty$ and therefore this condition can be dropped.

For both finite and infinite Blaschke products, it is easy to see that $|B(z)| = 1$ for $z = j\omega$ on the purely imaginary axis. In this paper, we are only interested in finite Blaschke products because of the following approximation Theorem of Helson and Sarason.

Theorem 1 *Let $u(j\omega)$ be any continuous function on the $j\omega$-axis for which $|u(j\omega)| = 1$ for all ω. Let $\epsilon > 0$. Then there exists finite Blaschke products $B_1(j\omega)$ and $B_2(j\omega)$ such that*

$$\|u(j\omega) - \frac{B_1(j\omega)}{B_2(j\omega)}\|_\infty < \epsilon,$$

where the $\|\cdot\|_\infty$ denotes the standard L^∞ norm computed over the $j\omega$ axis.

Let $b(j\omega) = \exp(j\theta(j\omega))$. Clearly $|b(j\omega)| = 1$. Applying the Helson-Sarason Theorem, for analytic or meromorphic functions $f(j\omega)$ (with no poles on the imaginary axis), the phase $b(j\omega)$ can be approximated by a rational function of finite Blaschke products. *The main theme of this paper is that these Blaschke products can be in turn computed using neural networks with a small amount of data*.

From the expression of finite Blaschke products, It can be easily observed that $B_i(j\omega) = A_i P_i(j\omega)/P_i^*(j\omega)$ for $i = 1, 2$ where A_i a complex number with $|A_i| = 1$ and $P_i(j\omega)$ is a complex polynomial of ω and $P_i^*(j\omega) = \bar{P}_i(-j\omega)$, where $\bar{P}_i(j\omega)$ is constructed from $P_i(j\omega)$ by replacing all its coefficients with their respective conjugates. It is easy to see that $[P^*(j\omega)]^* = P(j\omega)$. Letting $A = A_1/A_2$, the approximating rational function can be written as $AP_1(j\omega)P_2^*(j\omega)/P_1^*(j\omega)P_2(j\omega)$ with $|A| = 1$. Thus,

$$b(jw) \approx A\frac{P_1(jw)P_2^*(jw)}{P_1^*(jw)P_2(jw)}. \tag{1}$$

3.1 Blaschke Product Neural Network

Inspired by the above, we model the phase function by equation1, and propose the Blaschke Product Neural Network (BPNN). A BPNN is a neural network with the magnitude of observations as the input and the coefficients of the rational function of Blaschke products as the output. In this paper, we only present BPNNs which are fully connected neural networks (FCNN), although such a restriction is not necessary. We next describe the operation of BPNNs by discussing the forward and backward pass mechanisms.

Forward Pass The output of BPNN are Blaschke coefficients that are in turn used to construct predicted phase function $\tilde{b}(j\omega)$. Without loss of generality, let $\Omega = \{\omega_1, \omega_2, \cdots, \omega_N\}$ be the set of frequencies for which experimental results measuring the magnitude of signals $|f(j\omega)|$ is performed, and their corresponding phases must be retrieved. It can be assumed that $\omega_0 < \omega_1 < \cdots < \omega_N$. The BPNN then produces estimates $\tilde{b}(j\omega_i)$ of the phase $b(j\omega_i)$. This gives an estimate

$$\tilde{f}(j\omega_i) = |f(j\omega_i)|\tilde{b}(j\omega_i).$$

Backward Pass After predictions are generated with forward pass , the prediction error is computed as

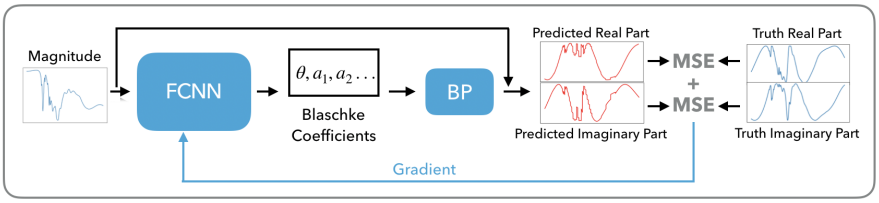
$$L = \frac{1}{N}\sum_{i=1}^{N} |f(j\omega_i) - \tilde{f}(j\omega_i)|^2.$$

The training seeks to minimize L over all training samples in a standard manner using the stochastic gradient descent algorithm and its variants. Cross-validation and Testing can also be done in a standard format.

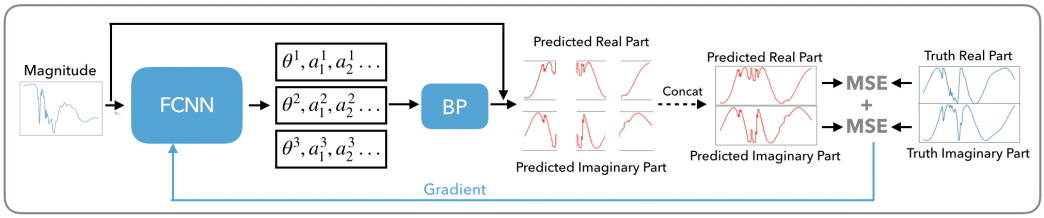
An important hyper-parameter of interest is number of degrees for complex polynomials $P_1(j\omega)$ and $P_2(j\omega)$. This hyper-parameter can either be cross-validated or chosen based on the practitioner's knowledge about the complexity of the specific phase retrieval problem. In our experiments, we

found out that complex phase retrieval problems may require larger degree polynomials, and calculating the value of these high degree polynomials over large frequency ranges (100THz - 1500THz and above) may lead to the overflow problem. To overcome this problem, we propose a piece-wise implementation of BPNN.

Piece-wise Implementation of BPNN In this approach, given a large frequency range, the whole frequency band is partitioned into non-overlapping contiguous segments. The Blaschke method is then applied to each segment (see figure2b). The exact method of partitioning and degrees of corresponding Blaschke polynomials for each sequence are hyper-parameters that can be adjusted. We will provide two examples of applications of Piecewise BPNNs to metamaterial datasets in our numerical experiments.



(a) Illustration of BPNNs



(b) Illustration of Piece-wise Implementation of BPNN

Figure 2: Structure of BPNN (and Piecewise BPNN): BPNN is an end-to-end model. Predictions are constructed by the combination of the Blaschke coefficients output of FCNN and values of input magnitude. In the Piecewise BPNN case, FCNN generates the sets of Blaschke coefficients for all segments (partitions) of the frequency range, and predictions are made for each segment separately and the results are concatenated.

4 Experiments

We conduct simulations on five phase retrieval datasets, two of which are synthetic, while the remaining three are metamaterial datasets. A detailed listing of the datasets is shown in Table 1. We endeavor to test the BPNN on diverse, yet complete, datasets, and thus we choose datasets that include both holomorphic and meromorphic complex functions. Additionally, the selected metamaterial datasets are comprehensive, and well represent the discipline.

4.1 Datasets

4.1.1 Synthetic I: Polynomial ODE

We first select points $\omega_0 < \omega_1 < \omega_2 < \cdots < \omega_N$ on the $j\omega$ axis and fix these values. The first synthetic dataset is the solution to a ordinary differential equation (ODE) given by:

$$\frac{df}{dz} = p(z),$$

where $p(z)$ is a complex polynomial of degree $m = 4$ with its roots lying on the unit circle. For a fixed $p(z)$ randomly selected, we solve the above ODE on the $j\omega$-axis using the Euler method with the real and imaginary part of the initial condition $f(\omega_0)$ uniformly selected in $[-1, 1]$ for 1050 randomly generated initial values. The values $(f(j\omega_1), f(j\omega_2), \cdots, f(j\omega_n))$ are then calculated

Table 1: List of phase retrieval problems in Experiments section. (# Frequency) shows total number of frequency points considered for the problem. Please see relevant sections for detailed descriptions.

PHASE RETRIEVAL PROBLEM	TYPE	# FREQUENCY
Polynomial ODE (4.1.1)	Synthetic	200
Lorentzian response function (4.1.2)	Synthetic	1000
Metamaterial absorber (4.1.3)	Metamaterial	256
Huygens' metasurface (4.1.4)	Metamaterial	2001
Membrane metasurface (4.1.5)	Metamaterial	2001

and their magnitudes and phases are used in this experiment, of which 50 are used for training and 1000 for testing.

4.1.2 SYNTHETIC II: LORENTZIAN RESPONSE FUNCTION

The second synthetic dataset is generated by considering a causal model of a non-magnetic material, with a frequency-dependent complex dielectric permittivity defined as a sum of Lorentzian oscillators:

$$\epsilon_r(\omega) = \epsilon_\infty + \sum_i^N \frac{\omega_{\epsilon,p,i}^2}{\omega_{\epsilon,o,i}^2 - \omega^2 - j * \omega_{\epsilon,s,i}\omega}.$$

A simple Lorentzian model of the dielectric function is applicable to a wide range of resonant material systems with response functions that must obey Kramers-Kronig relations. Using the transfer-matrix method, Markoš & Soukoulis (2008), the complex transmission coefficient $t(\omega)$ can be computed directly from the dielectric function via the closed form equation:

$$t(\omega) = \left[\cos\left(\frac{n\omega l}{c}\right) - \frac{i}{2}\left(z + \frac{1}{z}\right)\sin\left(\frac{n\omega l}{c}\right)\right]^{-1}, \tag{2}$$

where $n = \sqrt{\epsilon_r}$, $z = \sqrt{1/\epsilon_r}$, and c, l are pre-defined physical constants. We generate a dataset of complex transmission coefficients t by uniformly sampling Lorentzian parameters for ϵ_r optimized for the frequency range 100-500 THz. We consider a fixed set of four Lorentzian oscillators, and generate 50 sample transmission spectra for training, and 2000 spectra for the test dataset.

4.1.3 METAMATERIAL I: METAMATERIAL ABSORBER

Metamaterials are structured materials consisting of periodic arrays of resonant elements that derive their spectral properties from their unit cell geometry Smith et al. (2000). With effective material properties that can consist of resonances in both the electric permittivity and magnetic permeability, and hence more unknown material parameters, metamaterials often represent a more complicated phase-retrieval problem than many conventional systems.

The first metamaterial dataset considers a metal-based absorber geometry with a metallic ground plane backing a dielectric spacer layer, designed to operate in the millimeter wave (50-200 GHz) frequency range. This metamaterial absorber (MMA) dataset has 55,000 pairs of MMA geometry and complex frequency dependent reflection coefficient $r(\omega)$ ($t(\omega)$ is zero due to the metal backing). The data are obtained with commercial computational electromagnetic simulation software (CST Microwave Studios).

The input parameters include the four geometric dimensions of the top metal layer a, w_1, w_2, g, material properties of a top dielectric layer $\epsilon, \tan\delta$, the conductivity of the metal σ_{metal}, and thicknesses for all layers $d_{cap}, d_{sub}, d_{m1}, d_{m2}$. The output is the complex reflection coefficient $r(\omega)$, with 1000 frequency points for each of the real and imaginary part of $r(\omega)$ evaluated from 0 – 200 GHz.

4.1.4 METAMATERIAL II: HUYGENS' METASURFACE

Huygens' metasurfaces made of dielectric materials represent another class of metamaterials Decker et al. (2015), with underlying properties that are different than their metallic counterparts, including for example the lack of Ohmic losses Liu et al. (2017). When individual designs are combined into supercells, all-dielectric metasurfaces can exhibit very rich spectral phenomena. The Huygens' metasurface dataset consists of the complex reflection and transmission response derived from a supercell of elliptical resonators.

We adopt the Hyugens' metasurface dataset from the work of Deng et al. (2021). The metasurface supercell geometry has a periodicity p, including four elliptical SiC resonators with consistent height h. Each elliptical resonator has a major axis radius r_{ma}, a minor axis radius r_{mi}, and a rotational angle θ. The fourteen geometry parameters define inputs of computational simulations and yield complex reflection and transmission responses. This ADM dataset is particularly interested in the frequency-dependent reflection and transmission range from 100 - 500 THz. The phase information thus can be inferred directly from the complex reflection or transmission. Out of a total of 1000 samples, we randomly sample 80 for training and reserve the rest for testing.

4.1.5 METAMATERIAL III: MEMBRANE METASURFACE

The membrane metasurfaces are similar to the Huygens' metasurfaces Yang et al. (2020). However, instead of having four stand-alone elliptical resonators in the supercell, it consists of a thin slab of SiC with regions of empty holes. The SiC slab filling the supercell has a thickness defined by h, and a periodicity p. In the SiC slab, we placed four elliptical holes where the elliptical SiC resonators were placed. The four holes thus have the same geometrical parameters r_{ma}, r_{mi}, and θ. The fourteen geometrical parameters share the same range as the elliptical resonators dataset. Similarly, the input of these datasets contains fourteen geometry parameters, and the outputs are complex reflection and transmission with 2001 frequency points range from 100 THz to 500 THz. Out of a total of 7331 samples, we randomly sample 80 for training and reserve the rest for testing.

4.2 BENCHMARKING NEURAL NETWORKS

Comparison is done on each dataset with training datasets of increasing size. For each phase retrieval problem and each training dataset, we compare BPNN to a family of NNs of various sizes and structures. For each structure, we search its hyper-parameters of training to guarantee we have NNs close to optimal. And we record test error at each training epoch and report the best test error. We note that recording test error is uncommon but this guarantees that the reported performance is close to its limit.

We used NNs of same size hidden layers. The range of hidden layer size in search is [32,64,128,256] and the number of hidden layers in search is [1,2,3,4]. For hyper-parameters of training, we search dropout rate in the list of [0., 0.05, 0.1, 0.15, 0.2] and learning rate in the list of [1e-4, 5e-4, 1e-3, 5e-3, 1e-2]. Each NNs are trained with 6000 epochs for guarantee of convergence. And the training is done with 3 random initialization and we report the best test error. Huygens' metasurface dataset is more complex than the other datasets, so the range of hidden layer size in search is increased to [64,128,256,512] while keep ranges of other searched parameters same. Unlike the extensive searching for NNs, We do not search hyper-parameters of BPNN and only use one BPNN for each phase retrieval problem. We also train BPNN with 3 random initialization on each dataset and report its best test error.

4.3 RESULTS

For each dataset, we compare the family of regular neural networks, each with its best searched hyper-parameters to BPNN without hyper-parameters search. All y-axis are in dB scale for clearness of report and we use legend label (A, B): A refers to size of hidden layer while B refers to number of hidden layers.

Synthetic Datasets We use BPNN of 4 roots with a fully-connected neural network of 2 hidden layers, each of size 64 and no dropout. It is obvious from figure[3] that BPNN outperforms a family of optimized neural networks by a large margin. This supports our theoretical justification of using

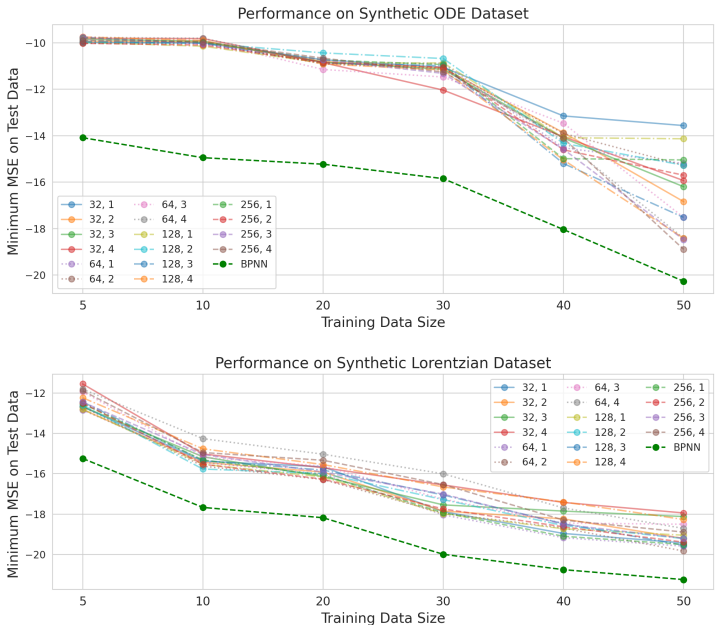


Figure 3: Performance comparison of a family of neural networks, each with its best hyper-parameters and BPNN on two synthetic phase retrieval problems with training set of various size.

BPNN when the phase functions are precisely meromorphic functions. The outstanding performance of BPNN on the Lorentzian dataset implies its value on phase retrieval problems in metamaterials. To confirm this, we conduct more experiments on real metamaterial datasets.

MMA Similar to synthetic datasets, we also use BPNN with 4 roots while using a fully-connected neural network with 2 hidden layers of size 64. No dropout nor weight-decay was added. Although without any regularization techniques for neural networks, BPNN still demonstrate superior performance to all the optimized neural networks. We can the clear advantage due to BPNN's more justified prior of function space. When the loss converges quickly, amount of required data for every unit decrease of neural network's test loss increase exponentially while BPNN's efficient learning can save large amount of efforts for collecting those extra data.

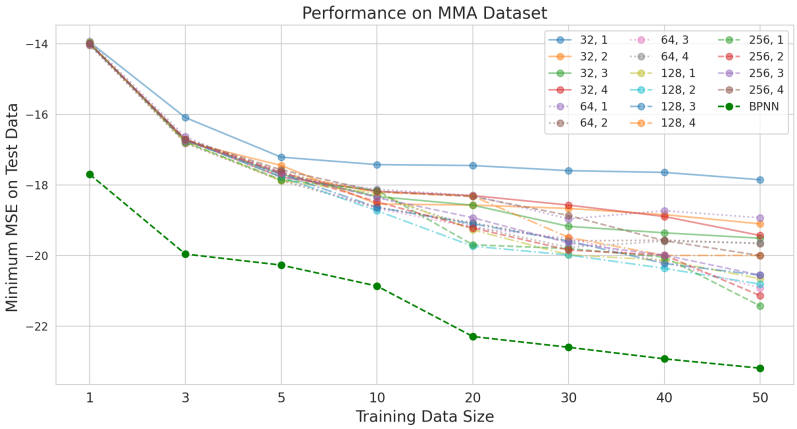


Figure 4: Performance comparison of best performing neural network and BPNN on MMA dataset.

ADM, Membrane Given that ADM dataset and Membrane dataset are much more complex than the synthetic datasets and MMA dataset, we use a Piecewise BPNN with frequencies partitioned into 20 equal-length sequences and we assign each sequence with 3 roots. We note that this is the most naive split for Piecewise BPNN and more reasonable split can lead to improved performance. We find that the best errors of regular neural networks are close to BPNN's best error. However, we also find that the median error of BPNN is much smaller than the best neural network(the neural network with smallest test MSE error across all architectures)'s median error. This shows that BPNN is making more good predictions by the sacrifice of small amounts of very bad predictions. This property can be valuable when the prediction problem at hand is on-or-off type due to BPNN's higher probability of making good predictions.

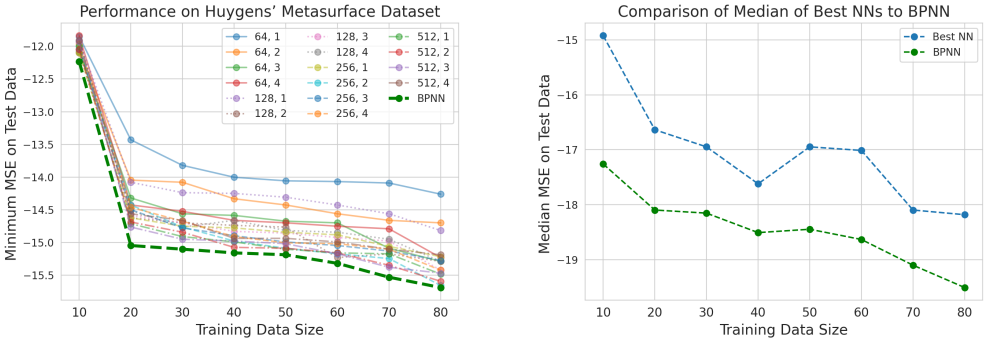


Figure 5: On the left is the performance comparison of a family of neural networks, each with its best hyper-parameters and BPNN on the ADM dataset. On the right is the comparison of median MSE loss of the best neural network in the large family and median MSE loss of BPNN.

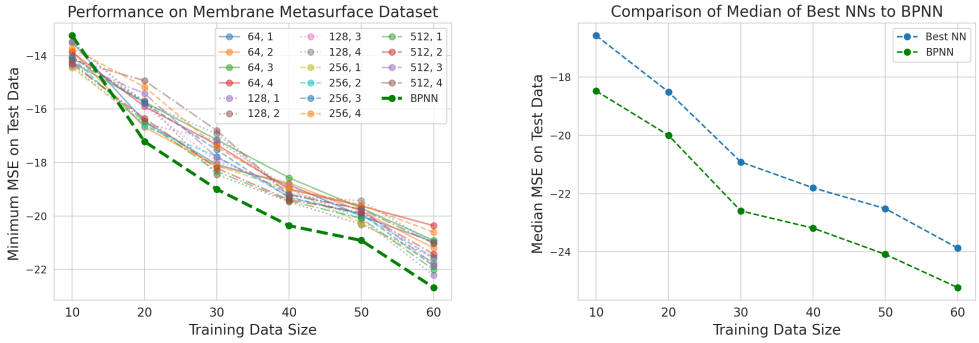


Figure 6: On the left is the performance comparison of a family of neural networks, each with its best hyper-parameters and BPNN on the membrane metasurface dataset. On the right is the comparison of median MSE loss of the best neural network in the large family and median MSE loss of BPNN.

5 Conclusion

We presented a physics-infused neural network referred to as Blaschke Product Neural Network(BPNN) for phase retrieval problem of holomorphic and meromorphic functions. Through extensive experiments on synthetic and metamaterial datasets, we have found that the data required for training neural networks is significantly less than expected when the underlying functions are holomorphic or meromorphic. By baking the physics into neural networks, even less training data is required and the performance is further improved.

6 REPRODUCIBILITY STATEMENT

We upload four datasets. Due to restriction of maximum supplementary size of 100MB, we include three datasets in supplementary: two synthetic datasets and Huygens' metasurface dataset. Membrane metasurface can be accessed with this link(https://figshare.com/s/966f196d1847269a447c). Details about hyper-parameter search space and the process of searching can be found in experiment section. Included in the supplementary we also have our implementation of the BPNN and neural networks by Pytorch with easy option of the hyper-parameters.

REFERENCES

Edwin J. Akutowicz. On the determination of the phase of a fourier integral, i. *Transactions of the American Mathematical Society*, 83(1):179–192, 1956. ISSN 00029947. URL http://www.jstor.org/stable/1992910.

Jens Berg and Kaj Nyström. A unified deep artificial neural network approach to partial differential equations in complex geometries. *Neurocomputing*, 317:28–41, 2018.

Emmanuel J Candes. The restricted isometry property and its implications for compressed sensing. *Comptes rendus mathematique*, 346(9-10):589–592, 2008.

N. M. Chalashkanov, S. J. Dodd, and J. C. Fothergill. Limitations of kramers-kronig transform for calculation of the DC conductance magnitude from dielectric measurements. In *2012 Annual Report Conference on Electrical Insulation and Dielectric Phenomena*. IEEE, October 2012. doi: 10.1109/ceidp.2012.6378768. URL https://doi.org/10.1109/ceidp.2012.6378768.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 6572–6583, 2018.

Manuel Decker, Isabelle Staude, Matthias Falkner, Jason Dominguez, Dragomir N Neshev, Igal Brener, Thomas Pertsch, and Yuri S Kivshar. High-efficiency dielectric huygens' surfaces. *Advanced Optical Materials*, 3(6):813–820, 2015.

Yang Deng, Simiao Ren, Kebin Fan, Jordan M Malof, and Willie J Padilla. Neural-adjoint method for the inverse design of all-dielectric metasurfaces. *Optics Express*, 29(5):7526–7534, 2021.

MWMG Dissanayake and Nhan Phan-Thien. Neural-network-based approximations for solving partial differential equations. *communications in Numerical Methods in Engineering*, 10(3):195–201, 1994.

J. R. Fienup. Phase retrieval algorithms: a comparison. *Applied Optics*, 21(15):2758, August 1982. doi: 10.1364/ao.21.002758. URL https://doi.org/10.1364/ao.21.002758.

Simon Foucart and Holger Rauhut. An invitation to compressive sensing. In *A mathematical introduction to compressive sensing*, pp. 1–39. Springer, 2013.

Stephan Ramon Garcia, Javad Mashreghi, and William T. Ross. Finite blaschke products: a survey, 2016.

Alexandre Goy, Kwabena Arthur, Shuai Li, and George Barbastathis. Low photon count phase retrieval using deep learning. *Physical review letters*, 121(24):243902, 2018.

Rola Houhou, Parijat Barman, Micheal Schmitt, Tobias Meyer, Juergen Popp, and Thomas Bocklitz. Deep learning as phase retrieval tool for cars spectra. *Optics Express*, 28(14):21002–21024, 2020.

Gauri Jagatap and Chinmay Hegde. Phase retrieval using untrained neural network priors. *NeurIPS Workshop on Deep Inverse Blind Submission*, 2019.

Frederick King. *Hilbert transforms*. Cambridge University Press, Cambridge, UK New York, 2009. ISBN 978-0-521-51720-1.

Tatiana Latychevskaia. Iterative phase retrieval in coherent diffractive imaging: practical issues. *Applied Optics*, 57(25):7187, August 2018. doi: 10.1364/ao.57.007187. URL https://doi.org/10.1364/ao.57.007187.

Xinyu Liu, Kebin Fan, Ilya V Shadrivov, and Willie J Padilla. Experimental realization of a terahertz all-dielectric metasurface absorber. *Optics express*, 25(1):191–201, 2017.

Valerio Lucarini. *Kramers-Kronig relations in optical materials research*. Springer, Berlin New York, 2005. ISBN 3-540-23673-2.

Peter Markoš and Costas M. Soukoulis. *Wave Propagation: From Electrons to Photonic Crystals and Left-Handed Materials*. Princeton University Press, stu - student edition edition, 2008. ISBN 9780691130033. URL http://www.jstor.org/stable/j.ctt7s2wj.

Christopher Metzler, Phillip Schniter, Ashok Veeraraghavan, et al. prdeep: robust phase retrieval with a flexible deep network. In *International Conference on Machine Learning*, pp. 3501–3510. PMLR, 2018.

Keith A. Nugent. X-ray noninterferometric phase imaging: a unified picture. *Journal of the Optical Society of America A*, 24(2):536, February 2007. doi: 10.1364/josaa.24.000536. URL https://doi.org/10.1364/josaa.24.000536.

Walter Rudin. *Real and Complex Analysis, 3rd Ed.* McGraw-Hill, Inc., USA, 1987. ISBN 0070542341.

Yoav Shechtman, Yonina C. Eldar, Oren Cohen, Henry Nicholas Chapman, Jianwei Miao, and Mordechai Segev. Phase retrieval with application to optical imaging: A contemporary overview. *IEEE Signal Processing Magazine*, 32(3):87–109, May 2015. doi: 10.1109/msp.2014.2352673. URL https://doi.org/10.1109/msp.2014.2352673.

Ayan Sinha, Justin Lee, Shuai Li, and George Barbastathis. Lensless computational imaging through deep learning. *Optica*, 4(9):1117–1125, 2017.

David R Smith, Willie J Padilla, DC Vier, Syrus C Nemat-Nasser, and Seldon Schultz. Composite medium with simultaneously negative permeability and permittivity. *Physical review letters*, 84 (18):4184, 2000.

Henry Stark. *Image recovery*. Academic Press, Orlando, 1987. ISBN 9780323145978.

Adrian Stern (ed.). *Optical Compressive Imaging*. CRC Press, November 2016. doi: 10.1201/9781315371474. URL https://doi.org/10.1201/9781315371474.

Kshitij Tayal, Chieh-Hsin Lai, Vipin Kumar, and Ju Sun. Inverse problems, deep learning, and symmetry breaking. *arXiv preprint arXiv:2003.09077*, 2020.

Yujia Xue, Shiyi Cheng, Yunzhe Li, and Lei Tian. Reliable deep-learning-based phase imaging with uncertainty quantification. *Optica*, 6(5):618–629, 2019.

Quanlong Yang, Sergey Kruk, Yuehong Xu, Qingwei Wang, Yogesh Kumar Srivastava, Kirill Koshelev, Ivan Kravchenko, Ranjan Singh, Jiaguang Han, Yuri Kivshar, et al. Mie-resonant membrane huygens' metasurfaces. *Advanced Functional Materials*, 30(4):1906851, 2020.

Run-Fa Zhang and Sudao Bilige. Bilinear neural network method to obtain the exact analytical solutions of nonlinear partial differential equations and its application to p-gbkp equation. *Nonlinear Dynamics*, 95(4):3041–3048, 2019.

Yuhe Zhang, Mike Andreas Noack, Patrik Vagovic, Kamel Fezzaa, Francisco Garcia-Moreno, Tobias Ritschel, and Pablo Villanueva-Perez. Phasegan: a deep-learning phase-retrieval approach for unpaired datasets. *Optics Express*, 29(13):19593–19604, 2021.