



Phase retrieval and design with automatic differentiation: tutorial

ALISON WONG,^{1,*} BENJAMIN POPE,^{2,3} LOUIS DESDOIGTS,¹ PETER TUTHILL,¹
BARNABY NORRIS,¹ AND CHRIS BETTERS¹

¹Sydney Institute for Astronomy (SIfA), School of Physics, The University of Sydney, Sydney, NSW 2006, Australia

²School of Mathematics and Physics, The University of Queensland, St Lucia, QLD 4072, Australia

³Centre for Astrophysics, University of Southern Queensland, West Street, Toowoomba, QLD 4350, Australia

*Corresponding author: a.wong@sydney.edu.au

Received 31 May 2021; revised 9 July 2021; accepted 9 July 2021; posted 12 July 2021 (Doc. ID 432723); published 6 August 2021

The principal limitation in many areas of astronomy, especially for directly imaging exoplanets, arises from instability in the point spread function (PSF) delivered by the telescope and instrument. To understand the transfer function, it is often necessary to infer a set of optical aberrations given only the intensity distribution on the sensor—the problem of *phase retrieval*. This can be important for post-processing of existing data, or for the design of optical phase masks to engineer PSFs optimized to achieve high-contrast, angular resolution, or astrometric stability. By exploiting newly efficient and flexible technology for *automatic differentiation*, which in recent years has undergone rapid development driven by machine learning, we can perform both phase retrieval and design in a way that is systematic, user-friendly, fast, and effective. By using modern gradient descent techniques, this converges efficiently and is easily extended to incorporate constraints and regularization. We illustrate the wide-ranging potential for this approach using our new package, Morphine. Challenging applications performed with this code include precise phase retrieval for both discrete and continuous phase distributions, even where information has been censored such as heavily saturated sensor data. We also show that the same algorithms can optimize continuous or binary phase masks that are competitive with existing best solutions for two example problems: an apodizing phase plate coronagraph for exoplanet direct imaging, and a diffractive pupil for narrow-angle astrometry. The Morphine source code and examples are available open-source, with an interface similar to the popular physical optics package Poppy. © 2021 Optical Society of America

<https://doi.org/10.1364/JOSAB.432723>

1. INTRODUCTION

A. Phase Retrieval

Phase retrieval is the task of recovering phase information from a point spread function (PSF) and is relevant to a range of areas in optical science and engineering, including electron microscopy, crystallography, coherent diffractive imaging, and astronomy. Famously in astronomy, phase retrieval was applied to precisely measure and correct for the Hubble Space Telescope mirror aberration shortly after its launch [1]. Correcting for phase aberrations is essential in pushing space telescopes to achieve the high angular resolution and contrast necessary to detect planets around other stars [2].

The phase retrieval problem arises from the inability of detectors (i.e., cameras) to measure the phase of incoming radiation directly. The PSF is the far-field diffraction pattern of the telescope, which at focus (for ideal optics) is the Fourier transform of the electric field distribution in the pupil plane. Unlike measurements made at radio wavelengths where electric fields can be measured directly, optical detectors measure

an intensity distribution given by the squared modulus of the complex PSF convolved with the source intensity distribution. This measurement suffers a loss of information at the detector: taking the square modulus eliminates phase so that the original electric field cannot be recovered by reversing the mathematical operations using an inverse Fourier transform.

Phase retrieval is an intrinsically difficult problem, as for geometric reasons, it is not well posed; however, in practice, a number of adequate solutions have been found [3]. Two early methods still used today are the Gerchberg–Saxton (GS) algorithm [4] and the hybrid input–output (HIO) algorithm [5]. Both are iterative methods that start with an initial phase estimate and then alternate propagation between the focal and pupil planes, each time imposing constraints so that the solution conforms to known information. Further work has seen the development of the HIO algorithm to incorporate sparsity constraints [6]. A comparison of these iterative methods (GS and HIO) is given in [7].

Numerous other approaches have been taken to solve phase retrieval owing to its central importance in so many areas of

science. Although a more complete review of the extensive literature in this area is beyond the scope here, the interested reader is directed to matrix completion [8] and Wirtinger flow [9]; semi-definite programming with sparsity constraints [10–12]; or greedy algorithms with sparsity constraints [13,14], as summarized in [15]. Deep learning has proven successful in phase retrieval [16–21], and while we will use the underlying technology of automatic differentiation in the present work, we focus exclusively on using deterministic optical simulation.

B. Automatic Differentiation

Gradient-based search algorithms provide another strategy that can be applied to the phase retrieval problem. While straightforward gradient descent can perform poorly on nonlinear problems, becoming trapped in local minima, it is often the only practical method for optimizing over a high-dimensional parameter space such as in neural networks, or large and complex phase patterns. Automatic differentiation, or autodiff, has made the well-known gradient-based “backpropagation” algorithm [22] newly efficient, and is a key enabling technology in the burgeoning field of deep learning [23,24]. Autodiff is extremely well suited to optical simulation and optimization. A neural network consists of a series of alternating linear transformations of a vector, generally followed by nonlinear element-wise operations. This is isomorphic to the linear propagation (Fourier or Fresnel) and local phase and amplitude masking in optical simulation.

Interestingly, the analogy also runs in reverse: it is possible to implement a neural network for image classification entirely using optical phase and amplitude masks, so that light (for example) from an image of a handwritten digit or shoe will diffract onto a spot corresponding to this classification [25]. This deep isomorphism means that the same software libraries and optimization approaches that are efficient for neural network implementations can be straightforwardly applied to optical simulation.

Autodiff allows the gradient-based optimization of deterministic optical simulations in high dimension. This is analogous to but distinct from its use in neural networks, which has been the most common application in the field so far, and instead, we are interested in using it for the physical optics simulation itself. Autodiff has been applied to geometric optics [26], including gravitational lensing [27,28], and to interferometric imaging [29].

A framework was established by Jurling and Fienup [30] applying algorithmic differentiation to phase retrieval, in which the differentiation rules for common operations were obtained by hand, with subsequent work tackling the coronagraph [31]. One of the major themes of the present paper is to build upon this idea applying modern automatic differentiation libraries, and flexibly linking it with practical, standard optical models. This can then be used for data analysis tasks, and extending this phase retrieval idea to hardware design. Because of the popularity of neural networks, there has recently been a very considerable public and private investment in both software and hardware for autodiff, which we can here exploit for physical optics. There are many Python implementations of autodiff,

including theano [32], PyTorch [33], TensorFlow [34], autograd [35], and JAX [36] which closely follows the NumPy API [37]. Similarly, the interpreted language Julia has native support for autodiff [38]. These libraries offer much greater flexibility for phase retrieval than was possible in the original framework [30], with work done in Molnar and Nikolic [39] demonstrating that JAX can be applied to the phase retrieval problem in radio astronomy.

Gradient-based techniques require only a differentiable objective function and so can be trivially applied to a wide variety of tasks. In this paper, we look at applying these techniques to challenges encountered in optics, which are naturally primed for this framework, as the underlying operations are straightforwardly differentiable. These tasks are therefore very well suited to gradient descent optimizations or even deep learning solutions. With regularization terms, or with a thresholded parametrization, we show that it is straightforward to optimize not just continuous, but also discrete phase masks. We will address example problems in both phase retrieval and phase design in the context of astronomy.

In the first half of this paper, we apply methods of phase retrieval similar to [20], but in the context of optical astronomy. Phase retrieval in astronomy presents its own set of unique challenges, such as nonlinearities introduced by detector saturation.

We demonstrate phase retrieval with gradient descent, including the effects of saturation. We will take the saturation model and linear detector sensitivity map as known, but note that these can be parametrized, differentiated, and learned as part of the same procedure. We also demonstrate how constraints can be added into the retrieval process, which allows for the incorporation of domain knowledge. The advantages in phase retrieval provided by having exact derivatives enable us to straightforwardly generalize to phase optic design, enabling fast gradient-based optimization of optical masks with respect to arbitrary objective functions.

C. Designing Astronomical Phase Optics

Astronomy requires high precision and low noise observations, relying heavily on the performance of optical instrumentation. Innovative design coupled with painstaking construction is therefore necessary to provide these quality measurements. We draw a distinction between phase retrieval, where a phase screen consistent with a PSF is recovered, and phase design, where a phase mask is optimized to achieve a PSF maximizing some figure of merit. Phase design can be used to develop application-specific phase masks, to optimize some component of a much larger optical system.

The challenge of phase design is related to that of phase retrieval, and much of the machinery used in phase retrieval can be re-purposed for this task. The key adjustment is that optimization is performed with respect to some carefully constructed objective function instead of optimizing on a metric measuring the difference between the observed and estimated PSFs. The difficulty often lies in the balancing of various components in the objective function.

Phase design using a differentiable physical optics simulation has been successfully demonstrated in other areas of optics:

for example, in adaptive optics [40]; the DeepOptics project, optimizing optical components in cameras in the areas of super-resolution imaging and in extending depth of field [41]; or WaveBlocks in microscopy [42]. In each of these cases, a physical optics simulation is implemented in an autodiff framework and used to design phase optics by gradient descent. We will be applying a related approach for the first time to astronomical optics.

In this paper, we demonstrate phase design on two test problems. The first is in the design of an apodizing phase plate (APP) coronagraph, and the second is in the design of a diffractive pupil for the Toliman space mission.

Coronagraphs are essential to high-contrast imaging and are prominently used in exoplanetary imaging. The goal of a coronagraph is to suppress on-axis starlight for some regions (or all) of the field of interest so that a fainter target, such as a planet, may be more easily identified. Among the most common and simplest forms of coronagraph design are those that result in a dark hole, such as an annular dark zone, within the PSF. These are characterized by the inner and outer working angles and evaluated by their contrast ratio. Due to their symmetry, these are generally easier to optimize compared to alternatives such as those enforcing a D-shaped region or a rectangular dark region.

An APP pupil plane coronagraph enforces a dark hole in the PSF purely by inducing a phase change [43] in the pupil plane. This results in full throughput of all starlight, and is distinct from a large family of alternative technologies in which light encounters obstructions in the pupil and/or focal plane. APP coronagraphs can be practically implemented with vector-APPs [44–48], which apply liquid crystal technologies to manufacture intricate phase patterns.

APP coronagraphs redistribute unwanted starlight rather than remove it and despite their simplicity, have proven popular. They have been widely used on large ground-based telescopes world-wide, since the original tests at the MMT Observatory 6.5 m telescope [49]. They have since been used to directly image the exoplanets β Pic b [50] and HD 100546 b [51], and many other planets and substellar objects [52].

A less familiar application for phase design, re-examined here, is the diffractive pupil for the Toliman space telescope [53]. This is an ambitious mission to detect exoplanets in the α Cen A/B binary system by way of very high precision relative astrometry between α Cen A and α Cen B. To reach the precision required for exoplanetary detection, the Toliman mission takes the novel approach of using a diffractive pupil to create a richly featured PSF. Observations of the binary system will then result

in overlapping fringes, which can be used to make the required precise relative positional measurements.

A candidate pupil for the Toliman mission and its corresponding PSF is shown in Fig. 1 and will be used in the TinyTol pathfinder mission, due to launch in August 2021. This will be a payload in the first CubeSat to be launched by CUAVA (The Australian Research Council Training Center for CubeSats, UAVs, and their Applications), ISTI (the Image, Spectograph and TinyTol Instrument). TinyTol has an 18 mm F/8.3 aperture and is a pathfinder/demonstrator for the larger Toliman telescope, which is expected to have a ~ 10 cm aperture.

The TinyTol pupil is composed of discrete regions with a π phase step difference [53]. By construction, there are equal areas of zero and π phases, which induces a null at the origin of the PSF. This aids in diverting light into the fringes of the pattern and also compensates for light leakage, which will manifest as a central peak.

The final design requirements for the Toliman pupil are subject to ongoing review, but for the purposes of this paper, there are two other major design considerations. First, the PSF must span a large enough region of the detector, being uniformly filled with sharp features, so that the fringes from each star in the α Cen system overlap. This allows for the relative positional measurement to be made with respect to those fringes. Second, the pattern ascribed to the pupil must be made of reasonably sized contiguous regions to ensure the design is within realistic fabrication limits.

2. PHASE RETRIEVAL

A. Methods

Here we demonstrate phase retrieval on test problems to illustrate practical challenges encountered in astronomical phase retrieval.

1. Pupil Recovery by Gradient Descent

As proof of concept, we perform phase retrieval from a target PSF with gradient descent for three cases:

1. TinyTol: the phase solution is known (Fig. 1).
2. Quantum monodromy (QM) pattern [54]: maintains a uniform density of points as a function of radius. Each of the peaks are constructed from a Gaussian with equal height and width, approximately the same size and shape of speckles. These properties were determined to be desirable for the Toliman PSF. This serves as an example of a realistic target PSF with desired properties, but where the phase

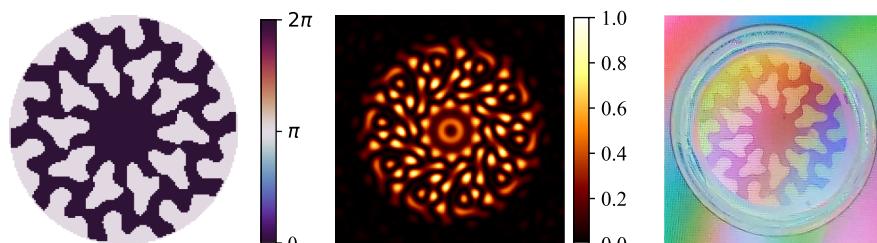


Fig. 1. Left: TinyTol diffractive pupil. Center: corresponding normalized PSF generated through simulation. Right: photograph of the fabricated TinyTol pupil through two polarizing filters to display the phase pattern.

solution is unknown and only an approximate solution is assumed to exist.

3. Faces of two coauthors: an arbitrary scene standing in for the PSF; here the phase solution is unknown and may not exist.

The first column in Fig. 2 shows each of the target PSFs (assumed monochromatic). To recover the phase in the pupil plane, we initialized a pupil (128×128 pixels) of small and random phase uniformly distributed $\in [-0.5, 0.5]$ radians from which the corresponding PSF (256×256 pixels) was calculated and normalized to a total flux of one. The objective was to minimize the error between the phase-estimated PSF and the target PSF. We used the mean absolute error (MAE) for the TinyTol and faces cases and the mean squared error (MSE) for the QM pattern, which were experimentally found to work well. For the phase retrieval problem, TensorFlow was used to calculate the gradients of each pixel in the pupil plane with respect to the objective function, and the Adam [55] algorithm was used to obtain an adaptive learning rate. While slower than

the Morphine gradient descent implementation, TensorFlow was more convenient to later adapt for neural network implementations. The learning rates were initialized to one (TinyTol), 10^7 (QM pattern), and 10^3 (faces), and were stepped down by a factor of 10 if the solution diverged. Additionally, the reconstruction of the QM pattern benefited from warm restarts where the learning rate was temporarily increased to drive the descent out of local optima. The recovered pupils are shown on the right, and the corresponding PSF shown in the center. Tasks solvable by gradient descent are also amenable to deep learning methods. Advantages of deep learning methods are that the neural networks can model more complex, nonlinear relationships and can optimize globally—an improvement over gradient descent, which is a local optimizer. Due to the connectivity of neurons in the network, the networks are able to optimize the pixels describing the phase in the pupil plane concurrently rather than independently. This approach of simultaneously optimizing structures at a range of scales has proven fruitful in mechanical design [56], and we investigated whether this would bear similar advantages for multiscale optimization of phase masks. For

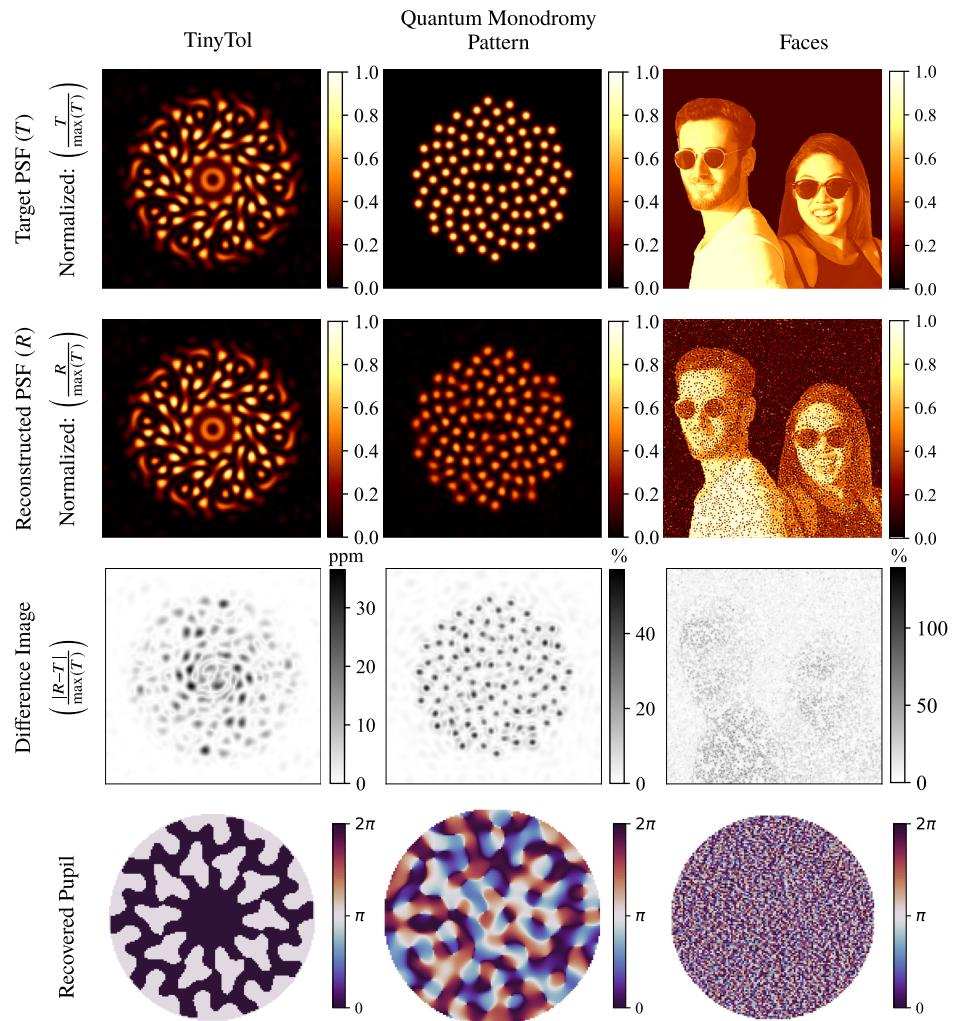


Fig. 2. Phase retrieval with gradient descent and the Adam optimizer for three cases. TinyTol: phase solution is known. Quantum monodromy pattern: realistic PSF where the phase solution is unknown and may not exist. Faces of two coauthors: arbitrary PSF where the phase solution is unknown and may not exist. From top: target PSF, then reconstructed PSF, both normalized to maximum value; relative difference image; and recovered phase pupil.

comparison, we implemented a dense neural network and a convolutional neural network, but found no benefit for these particular phase retrieval problems.

2. Pupil Recovery through Saturation

Detector saturation is an issue that often needs to be considered in astronomical phase retrieval. This occurs when the flux count is too high for the detector and results in a nonlinear detector response—typically observed at the central peak of the PSF. Autodiff phase retrieval with saturation has previously been demonstrated [30], and here we apply this to a severely saturated TinyTol PSF. The target PSF was the same as shown in Fig. 2 but with the maximum value clipped to 5×10^5 counts in the unnormalized image. This corresponded to a maximum value of 1.2×10^{-4} in the normalized (sum to one) image and clipped the maximum value to $\sim 23\%$ of the original maximum. A histogram of the pixel values in the PSF is shown in Fig. 4. The demonstration shows how this method can accurately reconstruct pupil plane phases though nonlinear non-flux conservative processes such as pixel saturation.

Phase retrieval for the saturated PSF followed the same processes as outlined above but with modification to the objective function. Before calculating the MAE between the estimated PSF and the target PSF, the estimated PSF was clipped to the maximum value of the target PSF. Results are shown in Fig. 4.

3. Constrained Optimization

Phase retrieval can benefit from domain knowledge, which can be incorporated into the phase retrieval algorithm in the form of constraints. These can reduce the size of the search space, making the optimization process more efficient while also ensuring that the solution is realistic and physical. For example, this could be to meet fabrication requirements, impose symmetry, or reflect prior knowledge of the optical system. In practice, any constraint can be used provided it is differentiable.

A practical example could be demonstrated in the reconstruction of the TinyTol pupil, which is known to be strictly binary. While it is not shown here, a constraint could be added to force phase values to $\pm \frac{\pi}{2}$. Additionally, symmetry constraints could be imposed, as TinyTol has 10-fold rotational symmetry.

For illustrative purposes, we demonstrate phase retrieval on the QM pattern where we constrain the phase ϕ to four evenly spaced values. The objective function used was

$$\frac{1}{N} \sum_{(x,y)} (\psi(x, y) - \hat{\psi}(x, y))^2 + c \sum_{(p,q)} \left| \left(\phi(p, q) \bmod \frac{\pi}{2} \right) - \frac{\pi}{4} \right|. \quad (1)$$

Here the first term is the MSE of the target PSF ψ and the estimated PSF $\hat{\psi}$, in dimensionless normalized counts and summed over pixels in the focal plane (x, y) , and the second is the regularization term used to implement the phase constraint, where (p, q) are the pixels in the pupil plane summed within the pupil support. The dimensionless regularization constant c was used to adjust the relative weights of the two terms. This was initially set to 10^{-14} and increased every ~ 1000 epochs until it reached 2.5×10^{-13} . This allowed the algorithm to find an early fit to the PSF before slowly introducing the phase constraint.

B. Discussion and Results

1. Pupil Recovery by Gradient Descent

Figure 2 shows successful recovery of the TinyTol pupil. This is perhaps unsurprising as the problem had a known solution. This solution was recovered in approximately ~ 15 min on a MacBook Pro with 8 i9 cores and 32 GB of RAM. To improve efficiency, symmetry and phase constraints could have been imposed. We attained a MAE in the PSF of 5.03×10^{-11} and a MAE in the pupil of 0.0024 radians.

Although the TinyTol pupil contains only two possible phase states, we allowed phases to range continuously between 0 and 2π radians. Applying a constraint enforcing binarity on the recovered pupil results in a MAE in the pupil of 4.25×10^{-8} radians. Given that the support consisted of 12874 pixels, a single flipped pixel (the minimum possible error) would contribute 2.4×10^{-4} radians of error. This makes the error found in our binary-enforced pupil likely a machine precision error and indicates that the recovery itself is essentially perfect.

Next we performed an investigation into phase recovery for a PSF where there is no known solution. We recovered the QM intensity pattern, which has speckled features characteristic of typical PSFs, although in this instance, the replication of the target was imperfect. Our algorithm's notable success was in yielding a PSF with the same structure as the QM pattern; however, at some level of detail, the peaks did not have as clean a structure as the target. This is seen in Fig. 2 where the peak intensities do not match the target due to light bleeding between the peaks of the recovered PSF. This illustrates an intrinsic limitation: over a finite diameter pupil with an applied phase function, the space of solutions that can be obtained is bounded. The specific example here required generation of a Gaussian function in the PSF, an outcome that cannot be obtained perfectly with a regular optical system. Despite this, the close approximation between the target and the recovered PSF points to practically valuable outcomes from this method.

Initializing the gradient descent with different seeds produced different, but equally good, results (Fig. 3). This is indicative of a complex search space with numerous local minima, which explains why this descent benefited from warm restarts.

Our “faces” example presents a test case where the algorithm is presented an outcome that is demonstrably unobtainable. A perfect phase solution cannot exist due to the spatial frequency structure of the target: both large smooth expanses and hard edges are problematic. The reconstructed PSF, although imperfect, is clearly recognizable to be human faces, and key structural features of the original faces are clear. However, the overall texture is speckled, which is characteristic of the speckles observed in typical PSFs. As expected, the recovered pupil has high spatial frequencies to distribute light broadly across the PSF.

Additionally, we attempted reconstructions with a dense neural network and a convolutional neural network for this same task, where the output of the networks was a proposed phase pattern, similar to methods used by Hoyer [56]. The results we obtained were marginally worse than those found via gradient descent, and the models took longer to optimize. This was surprising, as generally neural networks have the capacity to optimize globally and therefore should attain a better solution.

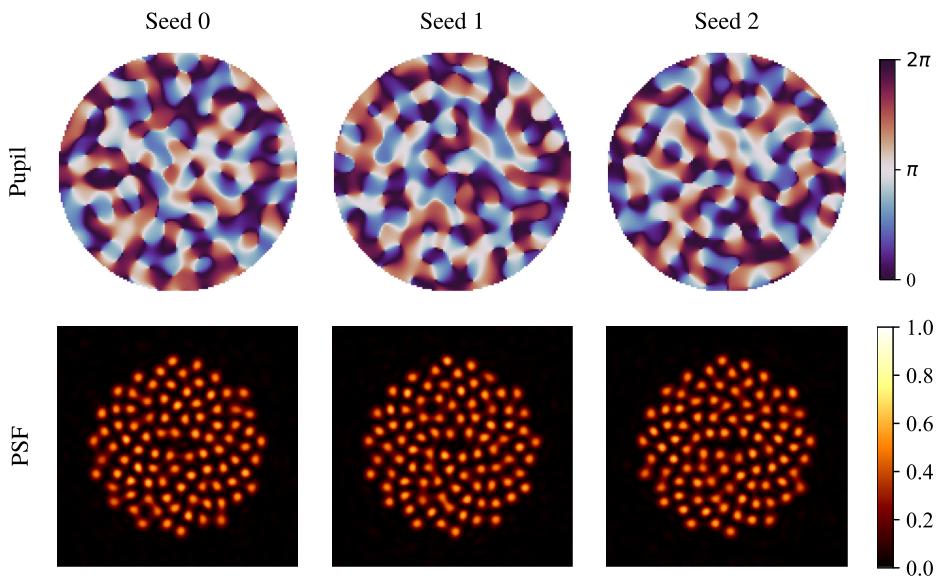


Fig. 3. Phase retrieval with gradient descent and the Adam optimizer on the quantum monodromy pattern with different initializations. Each presents an approximately equally good solution, with the mean absolute error (MAE) differing from case to case by a few parts per million.

However, we believe that in practice, the added complexity of the networks hindered the optimization process, and the increased complexity was unnecessary for the task.

While applying gradient descent to phase retrieval has been demonstrated previously, these results served as a baseline comparison for later experiments involving the handling of PSF saturation and the integration of domain knowledge in the form of constraints.

2. Pupil Recovery through Saturation

Figure 4 shows the results of phase retrieval from a saturated PSF. We have found that we were able to successfully recover the

TinyTol pupil achieving a MAE of 3×10^{-10} in the PSF and 0.005 radians in the pupil. Admittedly, the error in the pupil is two times worse than in the unsaturated pupil. However, applying a binary constraint to the reconstructed pupil reduces the MAE to 4.25×10^{-8} , which is on the scale of machine precision as before.

3. Constraints

Figure 5 shows the results of our reconstruction of the QM pattern when restricted to a four-phase coloring. It is important to note that we did not enforce a strict phase constraint. The metric in Eq. (1) simply penalizes phase that deviates from the

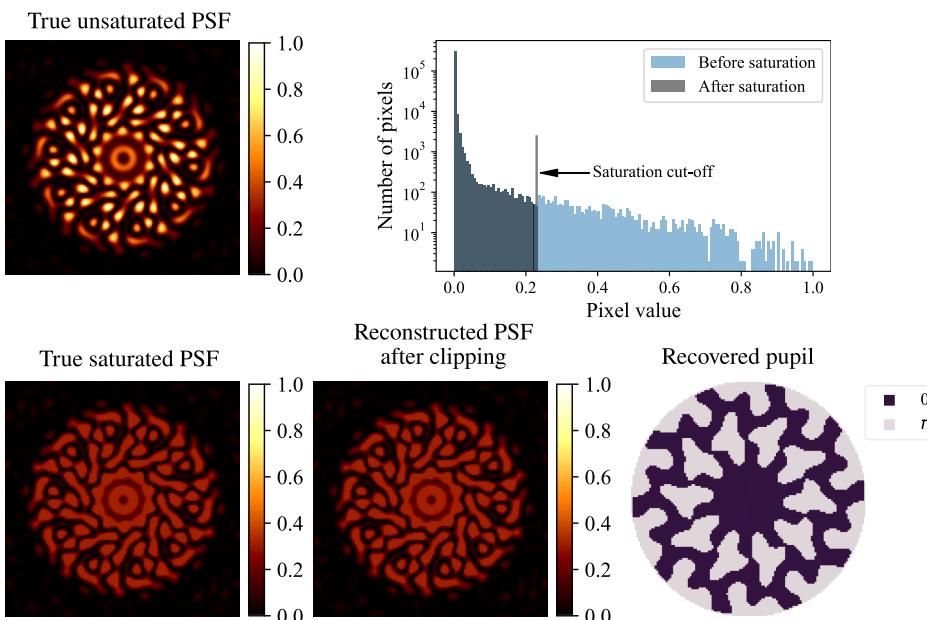


Fig. 4. Phase retrieval of a saturated PSF. The pixels in the true TinyTol PSF were saturated by clipping the values such that the maximum value was $\sim 23\%$ of the original maximum. Phase retrieval was performed by comparing a clipped reconstructed PSF with the saturated PSF.

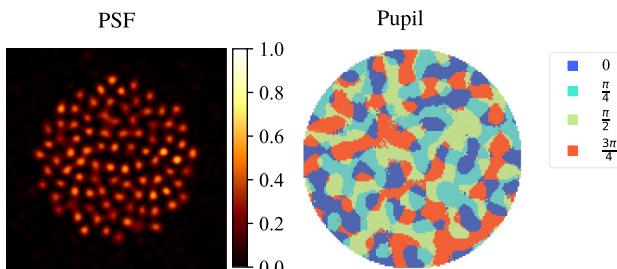


Fig. 5. Phase retrieval with the constraint given in Eq. (1). Here we enforce a four-phase coloring on the pupil, which has less than a percentage increase in MAE compared with the results in Fig. 3. The constraint was not strictly enforced, but rather it was imposed by adding a penalty to the objective function for any deviations from the constraint. The constraint could later be enforced on the resultant pupil. For the example shown above, this would have negligible effect.

four allowed phase values, and the strength of the constraint is controlled by the coefficient c . If the final pupil does not result in a strict phase set, this can be enforced at a later stage by rounding the phase values. We performed this final step on our four-phase QM pupil (Fig. 5), but it had negligible effects on the solution.

As expected, the four-phase solution is not as good as the unconstrained solution shown in Fig. 3, as the four-phase constraint limits the complexity of the allowed pupil. The peaks are not as uniform and there is more bleeding, which is reflected in a slightly higher MAE, but this increase is less than a percentage.

In this demonstration, the phases were initialized with seed 0 and interestingly, the pupil maintains features evident in the seed 0 pupil shown in Fig. 3. This indicates that the algorithm search remains in the vicinity of the seed 0 solution, and never leaves. To escape local optima, the algorithm can be run with different initializations, or warm restarts applied during the fitting process. Given the similarity in quality of the solutions produced by different seeds, we believe that the problem is likely underconstrained and multiple equally good solutions exist.

The examples shown in this section demonstrate how easily gradient descent can be applied to phase retrieval, while also illustrating the capabilities and limitations of this method in the presence of saturation or strong phase constraint conditions. It also lays down the necessary foundational work required to extend the phase retrieval method to the task of phase design.

3. PHASE DESIGN

We now consider an application of these gradient-descent methods to designing phase optics for astronomy, considering two examples: a scalar APP coronagraph (Section 3.A), and a diffractive pupil for astrometry (Section 3.B).

A. Apodizing Phase-Plate Coronagraph

While APP coronagraphs are powerful tools for direct imaging, designing such systems is nontrivial. The difficulty of the problem is that the error space is non-convex, and iterative methods can struggle to escape local optima. Previous approaches have applied phase iteration, e.g., [43,57], modified GS algorithms [58], or modified approaches to shaped-pupil coronagraph

design modulating both phase and amplitude [59]. These studies have shown that APP instruments are globally optimal for pupil plane coronagraphs [52].

In this section, we will show how gradient-based phase design provides a powerful and straightforward framework for optimizing a toy model: a monochromatic, scalar APP coronagraph with a multi-component objective function. We can apply gradient descent to either the coefficients in a modal basis or phase values on a pixelized grid, and straightforwardly achieve competitive coronagraph designs.

We will use Morphine for optical simulation, which is a fork of the popular Poppy library [60] using the JAX autodiff library [36] in place of NumPy. An advantage of this is that many instruments are already simulated in Poppy, which copes comfortably with multiple planes. These gradient descent methods are therefore easily extensible to designing hybrid coronagraphs, with phase and amplitude optics in conjugate planes.

1. Constructing a Basis

The simplest mode basis that can be used for phase design is the pixel basis, where each mode is the phase value of a single pixel in a grid. While this basis is complete up to high spatial frequency, it is quite inefficient for optimization, as the dimensionality is n^2 for an $n \times n$ pixel pupil plane grid. A modal basis is often preferable, as it does not scale with the grid size.

It is often useful to construct a symmetric basis to match the symmetries of the pupil, e.g., the rotational symmetry enforced by the spider arms that hold the secondary mirror. Further, symmetry reduces the complexity of the solution space, and a lower-dimensional basis can be used to improve computational efficiency. Here we have designed a basis with four-fold symmetry for a pupil support with four spiders.

A basis was created using logarithmic radial harmonic functions (LRHFs) [61], which were chosen as they contain a wide range of spatial frequencies and can therefore encode features at a range of scales and diffract light over a wide area of the focal plane. Each mode is defined such that

$$\phi(a, b, c) \equiv \cos(a \ln(r) + b\theta + c), \quad (2)$$

where r is the radius, θ is the angle from the x axis, and $-30 < a < 30 \in \mathbb{Z}$, $b = 4B$ for $0 < B < 8 \in \mathbb{Z}$, and $0 < c < 2\pi$ are constants. Samples are shown in Fig. 6. These modes have fine structure close to the origin that grows larger at greater radii.

We combined the LRFHs with circular modes defining rings at equally spaced contours. These were defined by

$$\sin(r\pi i) \quad (3)$$

and

$$\cos(r\pi i) \quad (4)$$

for $0 < i < 50 \in \mathbb{Z}$.

The resultant basis was optimized by applying principal component analysis (PCA) to reduce the basis down to 500 modes. Select modes from the optimized basis are shown in Fig. 6.

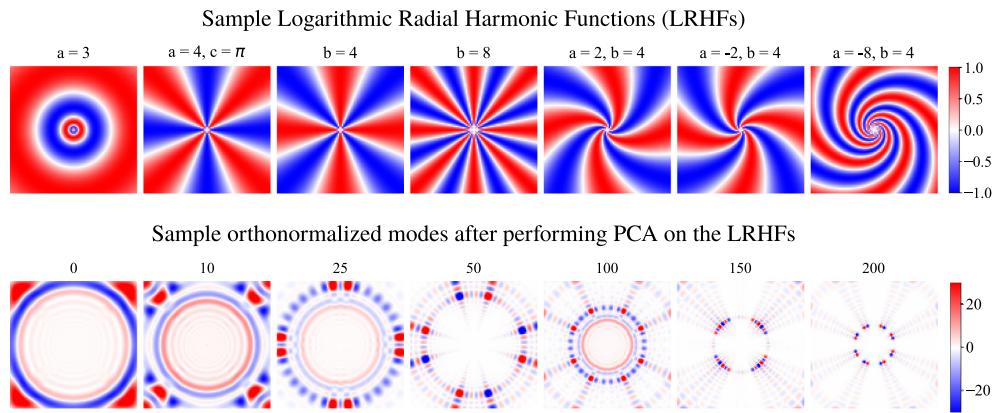


Fig. 6. Top: example modes created using Eq. (2). Constant values are zero where not otherwise specified. a controls the tightness of the spirals, b controls the number of spokes, and c controls the rotation. Bottom: sample modes after performing PCA. This was the mode basis used for coronagraph optimization.

The choice of basis was for demonstrative purposes only and is not necessarily optimal. In practice, the design of the basis should be informed by the task objectives.

2. Algorithm

Here we demonstrate naïve APP coronagraph design with gradient descent for an annular coronagraph where the aim was to suppress light between 3 and $7 \lambda/D$. We used the 500 mode basis shown in Fig. 6 and simulated an 8 m telescope at a wavelength of $1.6 \mu\text{m}$ with a detector pixelscale of 5×10^{-3} arcseconds. The pupil plane is sampled on a 768×768 pixel grid and the focal plane image 256×256 pixels. A 2.36 m radius secondary mirror obscures $\sim 8.7\%$ of the pupil. The four spiders are placed at right angles to each other to maintain four-fold symmetry.

The objective function balances two competing demands. The first is to minimize the peak value in the annulus and the second is to maximize the central peak. The objective function is then

$$a \times \max(\text{value in annulus}) - b \times \text{central peak}, \quad (5)$$

where a and b were constants used to weight the two metrics. Initially b was large to enforce a strong central peak; otherwise, the algorithm could resort to throwing light off the detector. a and b were tuned by hand during optimization, with the goal of maximizing a without the solution diverging. We also hand-tuned the learning rate η , starting at $\eta = 10^{-4}$ and decreasing by a factor of 10 every few thousand epochs until reaching $\eta = 10^{-6}$. The PSF was normalized by dividing by the peak value in the Airy disk obtained from the unobstructed pupil (i.e., no secondary mirror or spiders).

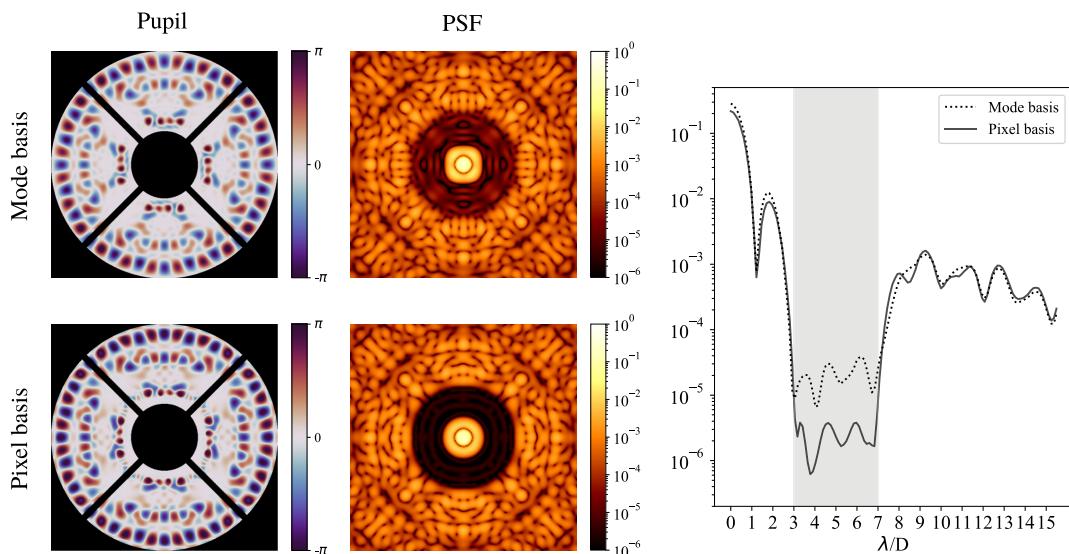


Fig. 7. APP coronagraph optimization. Left column: pupil plane phases of two coronagraph designs. Middle column: corresponding PSFs. Top row: coronagraph design and resultant PSF when optimizing using the modes shown in Fig. 6. Bottom row: results of further optimization performed in the pixel basis, initializing with the solution obtained in the mode basis. Right panel: radial profile of the two PSFs, from the center to the outer edge of the field of view.

Table 1. Comparison of the Coronagraph Solutions from the Mode Basis and the Pixel Basis^a

	Peak	Mean in the Annulus	5σ in the Annulus
Mode basis	0.284	2.677×10^{-6}	4.639×10^{-5}
Pixel basis	0.217	2.975×10^{-7}	4.546×10^{-6}

^aFlux values are normalized to those simulated from an unobstructed pupil, i.e., no spiders or secondary. The Airy disk would have a peak value of one.

Optimization was first conducted in the mode basis, as the 500 modes limited the complexity of the search space. The mode coefficients were initialized to small values uniformly randomly distributed $\in [0, 10^{-3}]$, and gradient descent was performed on the mode coefficients. The results are shown in Fig. 7.

Further optimization was conducted in the pixel basis. To maintain the four-fold symmetry, only pixels in a single quadrant were optimized. For a pupil with 768×768 pixels, a quadrant contained $384 \times 384 = 147456$ pixels, although $\sim 30\%$ were not in the pupil's support. The added complexity of the pixel basis allowed for a more optimal solution to be reached as shown in Fig. 7.

3. Discussion and Results

We found a two-stage approach to be effective: optimizing first in the symmetrized mode basis, before switching to the pixel basis. Using the mode basis, we were able to efficiently navigate the simpler solution space and identify a promising solution. Transferring to the pixel basis then allowed the solution space to be more finely searched. Interestingly, converting to the pixel basis did not result in a large deviation from the general shape recovered in the modal basis.

During the fitting process, the learning rate and the metric weightings [α and b in Eq. (5)] were tuned by hand. b was kept constant, and α was periodically increased to darken the region in the annulus. The purpose of the experiment was not to produce an optimal coronagraph, but to demonstrate that this method could be used for coronagraph design.

Table 1 compares the coronagraph solutions from the mode basis and the pixel basis. While the pixel basis achieves a darker annulus, the mode basis does maintain a higher peak. We did not use a metric to identify the optimal trade-off between these terms in the objective function, but recognize that one could be used.

B. Diffractive Pupil

In this section, we adopt the nomenclature for a pupil mask optimized for wide, astrometrically useful PSFs to be a “diffractive pupil” [53,62]. These make an interesting test case here due to the intrinsic complexity of the designs intended to produce relatively elaborate and strongly featured PSFs. Here we apply our phase design methods to the diffractive pupil for the Toliman space mission. The original TinyTol pupil was designed using the GS algorithm to try to achieve a patch of sharp peaks (a “bed of nails” pattern) by iteratively setting the amplitude of the complex wavefront to a top-hat function in the focal plane, and then re-applying the binary phase constraint in the pupil plane until convergence was reached. Here we performed optimization using the TinyTol constraints to compare the two optimization methods. We use 10-fold symmetry to match the existing TinyTol pupil.

One of the key differences between the TinyTol pupil and the Toliman pupil is that the Toliman pupil is designed for obstruction by a secondary mirror with radius 14.7% of the primary and furthermore with spiders to hold this secondary mirror. However this mission is presently in design phase, and the exact specifications are yet to be determined, so a somewhat simplified system (for example, omitting the spiders) was adopted here for illustrative purposes. We do, however, enforce eight-fold symmetry, which is appropriate for a system with four spider arms.

1. Basis

We constructed a basis with 10-fold symmetry to match the symmetry in the existing TinyTol pupil. This was constructed with the same methods as the basis used in the coronagraph design outlined in Section 3.A.1 but for 10-fold symmetry, again using PCA to optimize for the best 250 modes. Randomly generated samples are shown in Fig. 8 to illustrate the complexity of potential patterns spanned by the basis. In the design stage, the phases in the central region of the pupil (at $r < 0.35R_0$, where R_0 is the pupil radius) were set to zero to avoid high-frequency structures developing toward the center of the pupil—a consequence of the LRHF.

The first phase in designing the basis for the Toliman pupil followed the same process but with eight-fold symmetry. However, to generate a pattern that could be more easily fabricated, additional steps were taken to remove some of the high-frequency structures by convolving the mode basis with a Gaussian with $\sigma = 7$ pixels. This basis was then multiplied by the support before PCA was applied to retrieve the optimal

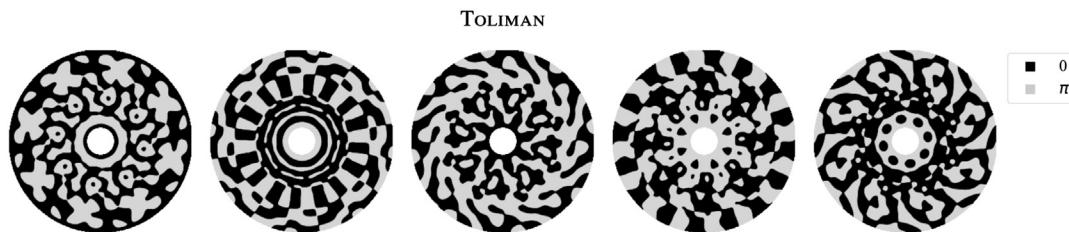


Fig. 8. Randomly generated samples from the Toliman mode basis samples. The underlying modes are continuous, but to enforce binary phase, using the CLIMB algorithm, positive values are mapped to π , and negative values are mapped to zero. The mode basis was convolved with a Gaussian of $\sigma = 7$ pixels before optimization to filter out the highest spatial frequencies.

80 modes. A random sample of draws from this basis is shown in Fig. 8.

2. Methods: CLIMB

In many cases it can be essential to design binary masks in phase or amplitude, whether for reasons of fabrication or physics. This is the case for the Toliman mission, and also for many kinds of the widely used shaped-pupil coronagraphs, which are binary in amplitude [63]. Unfortunately, derivatives cannot be taken though discrete-valued functions, as their gradients are necessarily either zero or infinity. So that we can design binary masks using the above symmetrized modal basis methods, we introduce an algorithm we call CLIMB: continuous latent-image mask binarization. A latent phase pattern is generated as a sum of a continuous mode basis designed as above. The binary mask is then produced by thresholding this latent phase pattern to π where it is positive, and zero when negative. We can therefore continuously optimize over the coefficients in a small mode basis, while generating a high-resolution binarized mask.

To propagate derivatives through this thresholding, it is necessary to give the binary pattern soft edges, where the mask ansatz assumes a floating-point value $\in (0, 1)$ only at the boundaries of regions. We describe this in pseudocode in Algorithm 1. We therefore generate our continuous pattern on a $3N \times 3N$ grid, which we downsample to the final $N \times N$ grid by applying the following prescription to each 3×3 subsampled pixel with phase values z . If all nine pixels are positive, return π , or if negative, return zero. Otherwise, we fit a 2D plane to the phase values, and calculate the line of intersection of this plane with the $z = 0$ plane. We then return the fractional area of the unit square above this line, i.e., for which this locally linearized $\phi > 0$. In practice, because this line can in general cut any two adjacent or opposite edges, this is first done by reorienting the square to a standard form and integrating. This results in a smooth transition from zero to unity as the edge of a binary region moves over a pixel, which means that gradients can propagate effectively even though the mask is zero or unity nearly everywhere.

3. Diffractive Pupil Mask with CLIMB

The goal was to design a diffractive pupil where the corresponding PSF was a circular flattop envelope around a “bed of nails” of sharp peaks, which are important for the Toliman diffractive-pupil astrometry concept. The key to this challenge was balancing the different metrics.

To encourage peaks in the PSF, one objective function used was gradient energy (GE), given in Eq. (6). While it is important to maximize GE, by itself, it is not a useful metric, as maximum GE is achieved with an Airy disk. This is counterproductive to the “circular flattop” requirement. Alternatively, a flattop radially weighted GE [FTRWGE, Eq. (7)] can be used, which encourages peaks at larger radii up to the maximum radius r_0 . However, used alone, it too leads to global minima that are not ideal, tending towards ring-shaped PSFs of radius r_0 . It was found that a combination of the GE and FTRWGE metrics performed well and resulted in a more even spread of light:

Algorithm 1. Continuous Latent Image Mask Binarization

Input: $3N \times 3N$ pixel float-value map Z
Output: $N \times N$ soft-thresholded binary mask ζ which can propagate gradients
 Initialize ζ empty.
 Fill each element ζ_{ij} by rebinning corresponding 3×3 block of Z :
for each 3×3 subgrid $z \in Z$: **do**
 if $\text{all}(z) > 0$
 $\zeta_{ij} \leftarrow 1$ **then**
 else if $\text{all}(z) < 0$
 $\zeta_{ij} \leftarrow 0$ **then**
 else
 Fit a plane $ax + by + cz = 0$ by least squares:
 $a, b, c \leftarrow \text{LeastSquares}(z)$
 Catch division by zero:
 if any $\{a, b, c\}_k = 0$ **then** $\{a, b, c\}_k \leftarrow \epsilon_{\text{machine}}$
 Find intercepts of the nodal line with the unit square:
 $x_1 = (-b - c)/(a)$
 $x_2 = -c/(a)$
 Reorient to a standard square:
 $x_1, x_2 \leftarrow \min(x_1, x_2), \max(x_1, x_2)$
 $x_1, x_2 \leftarrow \max(x_1, 0), \min(x_2, 1)$
 Calculate integral A in standard form:
 $A \leftarrow x_1 - (c/b)x_2 - (0.5a/b)x_2^2 + (c/b)x_1$
 $+ (0.5a/b)x_1^2$
 Catch edge cases involving inappropriate zero or NaN values.
 Enforce orientation:
 if $\text{mean}(z) > 0$ **then**
 $\zeta_{ij} \leftarrow A$
 else
 $\zeta_{ij} \leftarrow (1 - A)$
 end
end
end

$$\text{GE} = \sum_{(x,y)} \nabla \psi(x, y) = \sum_{(x,y)} \sqrt{\left(\frac{\partial \psi(x, y)}{\partial x}\right)^2 + \left(\frac{\partial \psi(x, y)}{\partial y}\right)^2} \quad (6)$$

$$\begin{aligned} \text{FTRW GE} = \sum_{(x,y)} \sqrt{\left(\frac{\partial \psi(x, y)}{\partial x}\right)^2 + \left(\frac{\partial \psi(x, y)}{\partial y}\right)^2} \\ \cdot \sqrt{x^2 + y^2} \cdot \text{circ}(x, y), \end{aligned} \quad (7)$$

where

$$\text{circ}(x, y) = \begin{cases} 1 & \text{if } \sqrt{x^2 + y^2} \leq r_0 \\ 0 & \text{if } \sqrt{x^2 + y^2} > r_0 \end{cases},$$

and ψ is the PSF as defined earlier. In conjunction to the GE metrics, additional metrics were used to enforce the circular flattop shape of the PSF. The radius of the main disk was approximately $10\lambda/D$, in accordance with the existing TinyTol pupil.

A summary of metrics is given below, and were used where the PSFs were normalized to a total flux of one. Figure 9 shows a map of the regions.

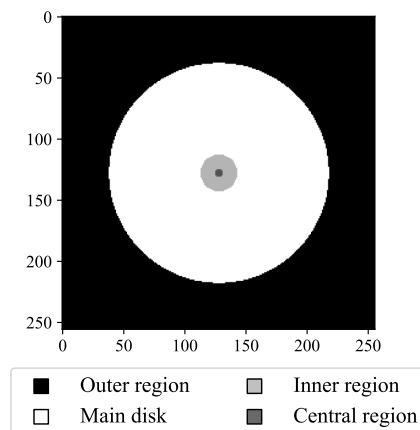


Fig. 9. Defined regions referred to in Eqs. (8) and (9). Ideally the TinyTol and Toliman PSFs should have uniform peaks in the main disk and be suppressed in the outer, inner, and central regions.

- **Metric 1: reduce light in the outer region.** This ensures maximum light in the central PSF and reduces broadband effects. m_1 was defined as the largest value in the outer region.

- **Metric 2: reduce light in the inner region.** This ensures more light in the disk of the PSF and also helps to account for leakage, which will result in a central peak. m_2 was defined as the largest value in the inner region.

- **Metric 3: reduce light in the central region.** This also contributed to ensuring an even spread of light around the disk. Otherwise, the solution tended toward enforcing the first Airy ring. m_3 was defined to be the largest value in the central region.

- **Metric 4: reduce the maximum peak in the main disk.** This was to keep the light evenly spread and ensure that no subset of speckles dominated. m_4 was defined to be the largest value in the circular disk.

- **Metric 5: high GE.** Combined the GE and FTRWGE equations.

Ultimately, the final objective function used for optimizing the 10-fold symmetric TinyTol pupil was

$$\begin{aligned} \mathcal{L}_{\text{TINYTOL}} = & 15 \max(m_1, 0.0573) \\ & + 5 \times 10^3 \max(m_2, 5 \times 10^{-5}) \\ & + 10^3 \max(m_3, 5.48 \times 10^{-4}) \\ & + 10^4 \max(m_4, 1.2 \times 10^{-4}) \\ & - 10^5 \text{GE} - 50 \text{FTRWGE}. \end{aligned} \quad (8)$$

When optimizing the eight-fold symmetric Toliman pupil, the same equation was used but with slightly different constants. Minor changes were found by experimentation to suit the new basis, resulting in the following objective function:

$$\begin{aligned} \mathcal{L}_{\text{TOLIMAN}} = & 10 \max(m_1, 0.0573) \\ & + 5 \times 10^3 \max(m_2, 5 \times 10^{-5}) \\ & + 10^3 \max(m_3, 5.48 \times 10^{-4}) \\ & + 2 \times 10^4 \max(m_4, 1 \times 10^{-4}) \\ & - 2 \times 10^5 \text{GE} - 50 \text{FTRWGE}. \end{aligned} \quad (9)$$

4. Optimization

Fitting for both TinyTol and Toliman was done by performing a gradient descent on the coefficients in the mode basis. We performed a simple fit under monochromatic conditions. Similar to the results in Fig. 3, the initialization affected the solution. We repeated gradient descent for multiple seeds (30 seeds for TinyTol, 50 seeds for Toliman). For TinyTol, we ran each fit for 250 epochs with a learning rate of $\eta = 0.05$. For Toliman, we ran for 500 epochs with a learning rate of $\eta = 0.05$, which was dropped down to $\eta = 0.01$ after 200 epochs. A selection of results are shown in Fig. 10.

Computations were performed using the Morphine library [64]. It took ~ 5.5 min to produce a TinyTol pupil (250 modes and 250 epochs) and ~ 3 min to produce a Toliman pupil (80 modes and 500 epochs) on a MacBook Pro with 8 i9 cores and 32 GB of RAM.

5. Discussion and Results

The solutions that arose from various seeds resulted in approximately the same overall merit, but their performance in individual metrics varied. Currently, there is no global metric for choosing the optimum pupil, so these were chosen on visual appeal. Interestingly, the resultant family of solutions generally had similar structure—large features at large radii and smaller features toward the center.

We used a very simple gradient descent without an adaptive learning rate. During optimization, local gradients of the loss function sometimes exhibited sharp increases, which resulted in divergent solutions that never recovered. In these cases, the solution could be either discarded for solutions from other initializations or rerun with a lower learning rate.

C. Practical Implementation

This section has illustrated the potential to harness modern machine automatic differentiation algorithms to the hitherto challenging domain of phase design in optical systems. The assumption at the outset is that masks manipulate only phase, and it is worth a brief digression here into what potential real-world implementations or methods of fabrication may be used to realize them. The short answer is that the machinery is quite general, and should prove of value wherever the pursuit of metrics quantifying desired outcomes is available. Dynamically controllable examples include the deformable mirrors in AO systems, which furthermore can allow for the superimposition of atmospheric correction while also adding a given phase mask pattern.

Modern lithographic methods can etch static designs onto various transmissive or reflective optical surfaces that are either flat or curved. These systems provide spatial resolutions in the single-micrometer regime, providing realistic ways to fabricate very fine features and sharp structures. Novel technologies such as multi-twist retarders [65] (used to fabricate the pupil in Fig. 1) can be applied to transmissive optics with similar spatial resolutions that provide nearly achromatic response through modifications in geometric phase. As long as the phase mask is fit-for-purpose in its optical setting, the algorithms here can be used to formulate the design.

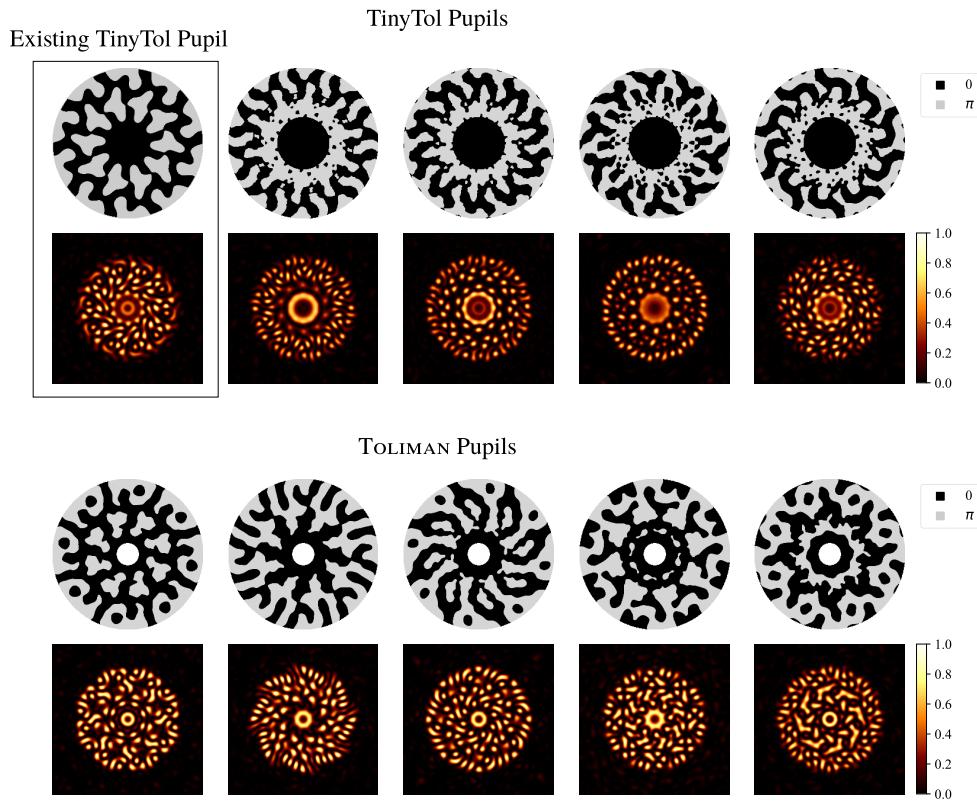


Fig. 10. Sample diffractive pupils produced using the TinyTol [Eq. (8)] and Toliman [Eq. (9)] metrics. Solutions were initialized with different seeds. Each solution achieves similar results on the objective function, although the relative contributions of the different optimized terms vary. TinyTol color scales are normalized to the maximum value in the existing TinyTol pupil, while Toliman PSFs are normalized to the maximum across the ensemble.

4. OPEN SCIENCE

We have made the Morphine source code freely available on GitHub [66], available under a GPLv3 open-source license. Implementations of the key calculations are provided as Jupyter notebooks [67], with GitHub links in the captions of each relevant figure. We encourage other researchers to reproduce, test, extend, and apply our work.

5. CONCLUSION AND FURTHER WORK

We have taken the highly optimized automatic differentiation tools that underpin the modern deep learning revolution and applied them to optical simulation for astronomy. By using packages such as JAX and Tensorflow to construct optical simulations and take derivatives, we have extended the work first done by Jurling [30] in the realm of phase retrieval to more complex and constrained situations. We have shown that this permits optimization of very many parameters, including various modal bases and dense pixel grids, as well as through a neural network to perform global optimizations of the phases. With these tools, the reconstruction of constrained pupil plane phases is simple, even in cases with information loss through non-flux conservative processes such as saturation, and with arbitrary and unusual PSFs.

The power of this method can be easily extended to complex imaging systems, where we have adapted the popular standard

optical simulation package Poppy with autodiff capabilities, as Morphine. An advantage of this approach is that existing simulations in Poppy are easy to adapt to be differentiable, and take advantage of JAX's just-in-time compilation, accelerated linear algebra, and GPU support. Using Morphine, we showed promising results in astronomical phase mask design, specifically for the construction and optimization of an APP coronagraph. Basic gradient descent methods find solutions close to current state of the art.

The gradient descent approach can even be used to optimize discrete phase maps, whether by regularization or through soft thresholding. In the latter case, the simple CLIMB algorithm proposed here allows simultaneous imposition of symmetry and binarization, and would extend well to shaped-pupil coronagraphs and other coded aperture optimization problems.

A limiting factor in the performance of current coronagraphs is their sensitivity to tip–tilt “jitter” and other low-order wavefront errors. It will be straightforward in our framework to optimize coronagraphs that are robust to these random errors by stochastic gradient descent over an ensemble of noise realizations. Similarly, by batching over wavelength diversity, we expect to be able to optimize broadband performance. By optimizing over phases in multiple conjugate planes, by analogy to multi-plane light conversion [68], we expect that gradient descent approaches will enable the design of new and complex hybrid coronagraphs, imaging spectrographs, and astrophotonic devices.

There is no reason to limit future work to simply phase retrieval: any aspect of the imaging system that can be parametrized and differentiated can in principle be learned, calibrated, or optimized as part of the same approach. For example, the inter-pixel sensitivity flat field map could be jointly learned together with a PSF model.

Funding. Breakthrough Initiatives; Jet Propulsion Laboratory.

Acknowledgment. This research was supported by contributions from the Breakthrough Watch program, which is managed by the Breakthrough Initiatives and sponsored by the Breakthrough Prize Foundation. This work was performed in part under contract with the Jet Propulsion Laboratory (JPL) funded by NASA through the Sagan Fellowship Program executed by the NASA Exoplanet Science Institute. This research made use of NASA's Astrophysics Data System. This research made use of JAX [36]; TensorFlow [34]; Poppy, an open-source optical propagation Python package originally developed for the James Webb Space Telescope project [60]; the IPython package [69]; NumPy [37]; matplotlib [70]; and SciPy [71]. BJSP is grateful to the School of Physics and Sydney Institute for Astronomy at the University of Sydney for hosting him as a visiting researcher during the COVID-19 pandemic. We acknowledge and pay respect to the Gadigal people of the Eora Nation. It is upon their unceded, sovereign, ancestral lands that the University of Sydney is built. BJSP acknowledges the traditional owners of the land on which the University of Queensland is situated, the Turrbal and Jagera people. We pay respects to their Ancestors and descendants, who continue cultural and spiritual connections to Country.

Disclosures. The authors declare no conflicts of interest.

Data Availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request. See also [66,67].

REFERENCES

1. J. R. Fienup, J. C. Marron, T. J. Schulz, and J. H. Seldin, "Hubble Space Telescope characterized by using phase-retrieval algorithms," *Appl. Opt.* **32**, 1747–1767 (1993).
2. M. Ygouf, L. M. Mugnier, D. Mouillet, T. Fusco, and J. L. Beuzit, "Simultaneous exoplanet detection and instrument aberration retrieval in multispectral coronagraphic imaging," *Astron. Astrophys.* **551**, A138 (2013).
3. A. H. Barnett, C. L. Epstein, L. F. Greengard, and J. F. Magland, "Geometry of the phase retrieval problem," *Inverse Prob.* **36**, 094003 (2020).
4. R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik* **35**, 237–246 (1972).
5. J. R. Fienup, "Reconstruction of an object from the modulus of its Fourier transform," *Opt. Lett.* **3**, 27–29 (1978).
6. S. Mukherjee and C. S. Seelamantula, "An iterative algorithm for phase retrieval with sparsity constraints: application to frequency domain optical coherence tomography," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2012), pp. 553–556.
7. J. R. Fienup, "Phase retrieval algorithms: a comparison," *Appl. Opt.* **21**, 2758–2769 (1982).
8. E. J. Candès, Y. Eldar, T. Strohmer, and V. Voroninski, "Phase retrieval via matrix completion," arXiv:1109.0573 (2011).
9. E. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: theory and algorithms," arXiv:1407.1065 (2014).
10. Y. Shechtman, Y. C. Eldar, A. Szameit, and M. Segev, "Sparsity based sub-wavelength imaging with partially incoherent light via quadratic compressed sensing," *Opt. Express* **19**, 14807–14822 (2011).
11. H. Ohlsson, A. Y. Yang, R. Dong, and S. S. Sastry, "Compressive phase retrieval from squared output measurements via semidefinite programming," *IFAC Proc. Vol.* **45**, 89–94 (2012).
12. I. Waldspurger, A. D'aspremont, and S. Mallat, "Phase recovery, maxcut and complex semidefinite programming," *Math. Program.* **149**, 47–81 (2015).
13. S. Bahmani, P. Boufounos, and B. Raj, "Greedy sparsity-constrained optimization," in *Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)* (2011), pp. 1148–1152.
14. Y. Shechtman, A. Beck, and Y. C. Eldar, "Gespar: efficient phase retrieval of sparse signals," *IEEE Trans. Signal Process.* **62**, 928–938 (2014).
15. Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging: a contemporary overview," *IEEE Signal Process. Mag.* **32**(3), 87–109 (2015).
16. A. Kappeler, S. Ghosh, J. Holloway, O. Cossairt, and A. Katsaggelos, "Ptychnet: CNN based Fourier ptychography," in *IEEE International Conference on Image Processing (ICIP)* (2017), pp. 1712–1716.
17. L. Boominathan, M. Maniparambil, H. Gupta, R. Baburajan, and K. Mitra, "Phase retrieval for Fourier ptychography under varying amount of measurements," in *Fourier Ptychography, Computational Imaging Using Deep Learning* (2018).
18. C. Metzler, P. Schniter, A. Veeraraghavan, and R. Baraniuk, "prDeep: robust phase retrieval with a flexible deep network," in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy and A. Krause, eds., Vol. **80** of *Proceedings of Machine Learning Research PMLR* (2018), pp. 3501–3510.
19. G. Jagatap and C. Hegde, *Phase Retrieval using Untrained Neural Network Priors* (2019).
20. F. Wang, Y. Bian, H. Wang, M. Lyu, G. Pedrini, W. Osten, G. Barbastathis, and G. Situ, "Phase imaging with an untrained neural network," *Light Sci. Appl.* **9**, 77 (2020).
21. Y. Nishizaki, R. Horisaki, K. Kitaguchi, M. Saito, and J. Tanida, "Analysis of non-iterative phase retrieval based on machine learning," *Opt. Rev.* **27**, 136–141 (2020).
22. Y. LeCun, D. Touretzky, G. Hinton, and T. Sejnowski, "A theoretical framework for back-propagation," in *Proceedings of the 1988 Connectionist Models Summer School (CMU)* (1988), Vol. **1**, pp. 21–28.
23. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**, 436–444 (2015).
24. A. Gunes Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: a survey," arXiv:1502.05767 (2015).
25. X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, "All-optical machine learning using diffractive deep neural networks," *Science* **361**, 1004–1008 (2018).
26. B. M. Sutin, "An optical toolbox for astronomical instrumentation," *Proc. SPIE* **9911**, 99112J (2016).
27. W. R. Morningstar, L. Perreault Levasseur, Y. D. Hezaveh, R. Blandford, P. Marshall, P. Putzky, T. D. Rueter, R. Wechsler, and M. Welling, "Data-driven reconstruction of gravitationally lensed galaxies using recurrent inference machines," *Astrophys. J.* **883**, 14 (2019).
28. K. Karchev, A. Coogan, and C. Weniger, "Strong-lensing source reconstruction with variationally optimised Gaussian processes," arXiv:2105.09465 (2021).
29. I. Czekala and R. Loomis, *iancze/MPoL: Pip Installable Package* (2020).
30. A. S. Jurling and J. R. Fienup, "Applications of algorithmic differentiation to phase retrieval algorithms," *J. Opt. Soc. Am. A* **31**, 1348–1359 (2014).
31. S. D. Will, T. D. Groff, and J. R. Fienup, "Jacobian-free coronagraphic waveform control using nonlinear optimization," *J. Astron. Telesc. Instrum. Syst.* **7**, 019002 (2021).
32. Theano Development Team, "Theano: a Python framework for fast computation of mathematical expressions," arXiv:abs/1605.02688 (2016).
33. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: an imperative style, high-performance deep learning library," arXiv:1912.01703 (2019).
34. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur,

- J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* (2015).
35. D. Maclaurin, D. Duvenaud, and R. P. Adams, "Autograd: effortless gradients in NumPy," in *ICML 2015 AutoML Workshop* (2015), Vol. **238**.
36. J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, and S. Wanderman-Milne, *JAX: Composable Transformations of Python+NumPy Programs* (2018).
37. C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Rio, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature* **585**, 357–362 (2020).
38. J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, "Julia: a fast dynamic language for technical computing," arXiv:1209.5145 (2012).
39. M. Molnar and B. Nikolic, "A generalized approach to redundant calibration with Jax," *Tech. Rep.* 84 (Hydrogen Epoch of Reionization Array Consortium, 2020) HERA Memo #84.
40. I. Vishniakou and J. D. Seelig, "Differentiable model-based adaptive optics with transmitted and reflected light," *Opt. Express* **28**, 26436–26446 (2020).
41. V. Sitzmann, S. Diamond, Y. Peng, X. Dun, S. Boyd, W. Heidrich, F. Heide, and G. Wetzstein, "End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging," *ACM Trans. Graph.* **37**, 114 (2018).
42. J. Page and P. Favaro, "Learning to model and calibrate optics via a differentiable wave optics simulator," arXiv:2005.08562 (2020).
43. J. L. Codona, M. A. Kenworthy, P. M. Hinz, J. R. P. Angel, and N. J. Woolf, "A high-contrast coronagraph for the MMT using phase apodization: design and observations at 5 microns and 2 λ/D radius," *Proc. SPIE* **6269**, 560–569 (2006).
44. F. Snik, G. Otten, M. Kenworthy, M. Miskiewicz, M. Escuti, C. Packham, and J. Codona, "The vector-APP: a broadband apodizing phase plate that yields complementary PSFs," *Proc. SPIE* **8450**, 224–234 (2012).
45. G. P. L. Otten, F. Snik, M. A. Kenworthy, M. N. Miskiewicz, M. J. Escuti, and J. L. Codona, "The vector apodizing phase plate coronagraph: prototyping, characterization and outlook," *Proc. SPIE* **9151**, 577–586 (2014).
46. G. P. P. L. Otten, F. Snik, M. A. Kenworthy, M. N. Miskiewicz, and M. J. Escuti, "Performance characterization of a broadband vector apodizing phase plate coronagraph," *Opt. Express* **22**, 30287–30314 (2014).
47. G. P. P. L. Otten, F. Snik, M. A. Kenworthy, C. U. Keller, J. R. Males, K. M. Morzinski, L. M. Close, J. L. Codona, P. M. Hinz, K. J. Hornburg, L. L. Brickson, and M. J. Escuti, "On-sky performance analysis of the vector apodizing phase plate coronagraph on MagAO/Clio2," *Astrophys. J.* **834**, 175 (2017).
48. S. P. Bos, D. S. Doelman, K. L. Miller, and F. Snik, "New concepts in vector-apodizing phase plate coronagraphy," *Proc. SPIE* **11448**, 114483W (2021).
49. M. A. Kenworthy, J. L. Codona, P. M. Hinz, J. R. P. Angel, A. Heinze, and S. Sivanandam, "First on-sky high-contrast imaging with an apodizing phase plate," *Astrophys. J.* **660**, 762–769 (2007).
50. S. P. Quanz, M. R. Meyer, M. A. Kenworthy, J. H. V. Girard, M. Kasper, A.-M. Lagrange, D. Apai, A. Boccaletti, M. Bonnefoy, G. Chauvin, P. M. Hinz, and R. Lenzen, "First results from very large telescope NACO apodizing phase plate: 4 μm images of the exoplanet β pictoris b," *Astrophys. J.* **722**, L49–L53 (2010).
51. S. P. Quanz, A. Amara, M. R. Meyer, M. A. Kenworthy, M. Kasper, and J. H. Girard, "A Young protoplanet candidate embedded in the circumstellar disk of HD 100546," *Astrophys. J.* **766**, L1 (2013).
52. D. S. Doelman, F. Snik, E. H. Por, S. P. Bos, G. P. P. L. Otten, M. Kenworthy, S. Y. Haffert, M. Wilby, A. J. Bohn, B. J. Sutliff, K. Miller, M. Ouellet, J. de Boer, C. U. Keller, M. J. Escuti, S. Shi, N. Z. Warriner, K. Hornburg, J. L. Birkby, J. Males, K. M. Morzinski, L. M. Close, J. Codona, J. Long, L. Schatz, J. Lumbres, A. Rodack, K. Van Gorkom, A. Hedglen, O. Guyon, J. Lozi, T. Groff, J. Chilcote, N. Jovanovic, S. Thibault, C. de Jonge, G. Allain, C. Vallée, D. Patel, O. Côté, C. Marois, P. Hinz, J. Stone, A. Skemer, Z. Briesemeister, A. Boehle, A. M. Glauser, W. Taylor, P. Baudoz, E. Huby, O. Absil, B. Carlomagno, and C. Delacroix, "Vector-apodizing phase plate coronagraph: design, current performance, and future development [invited]," *Appl. Opt.* **60**, D52–D72 (2021).
53. P. Tuthill, E. Bendek, O. Guyon, A. Horton, B. Jeffries, N. Jovanovic, P. Klupar, K. Larkin, B. Norris, B. Pope, and M. Shao, "The TOLIMAN space telescope," *Proc. SPIE* **10701**, 432–441 (2018).
54. A. Bolsinov, H. Dullin, and A. Veselov, "Spectra of sol-manifolds: arithmetic and quantum monodromy," *Commun. Math. Phys.* **264**, 583–611 (2006).
55. D. Kingma and J. Ba, "Adam: a method for stochastic optimization," in *International Conference on Learning Representations* (2014).
56. S. Hoyer, J. Sohl-Dickstein, and S. Greydanus, "Neural reparameterization improves structural optimization," arXiv:1909.04240 (2019).
57. J. L. Codona and R. Angel, "Imaging extrasolar planets by stellar halo suppression in separately corrected color bands," *Astrophys. J.* **604**, L117–L120 (2004).
58. G. J. Ruane, E. Huby, O. Absil, D. Mawet, C. Delacroix, B. Carlomagno, and G. A. Swartzlander, "Lyot-plane phase masks for improved high-contrast imaging with a vortex coronagraph," *Astron. Astrophys.* **583**, A81 (2015).
59. E. H. Por, "Optimal design of apodizing phase plate coronagraphs," *Proc. SPIE* **10400**, 236–247 (2017).
60. M. D. Perrin, R. Soummer, E. M. Elliott, M. D. Lallo, and A. Sivaramakrishnan, "Simulating point spread functions for the James Webb Space Telescope with WebbPSF," *Proc. SPIE* **8442**, 84423D (2012).
61. P. A. Fletcher and K. G. Larkin, "Direct embedding and detection of RST invariant watermarks," in *Information Hiding*, F. A. P. Petitcolas, ed. (Springer, 2003).
62. O. Guyon, E. A. Bendek, J. A. Eisner, R. Angel, N. J. Woolf, T. D. Milster, S. M. Ammons, M. Shao, S. Shaklan, M. Levine, B. Nemati, J. Pitman, R. A. Woodruff, and R. Belikov, "High-precision astrometry with a diffractive pupil telescope," *Astrophys. J. Supp.* **200**, 11 (2012).
63. S. Tanaka, K. Enya, L. Abe, T. Nakagawa, and H. Kataza, "Binary-shaped pupil coronagraphs for high-contrast imaging using a space telescope with central obstructions," *Publ. Astron. Soc. Jpn.* **58**, 627–639 (2006).
64. B. J. S. Pope, L. Pueyo, Y. Xin, and P. G. Tuthill, "Kernel phase and coronagraphy with automatic differentiation," *Astrophys. J.* **907**, 40 (2021).
65. R. K. Komanduri, K. F. Lawler, and M. J. Escuti, "Multi-twist retarders: broadband retardation control using self-aligning reactive liquid crystal layers," *Opt. Express* **21**, 404–420 (2013).
66. B. J. S. Pope and L. Desdoigts, "Morphine," GitHub, 2021, <https://github.com/benjaminpope/morphine>.
67. A. Wong, B. J. S. Pope, and L. Desdoigts, "Phase retrieval and design," GitHub, 2021, https://github.com/alipwong/phase_retrieval_and_design.
68. N. K. Fontaine, R. Ryf, H. Chen, D. T. Neilson, K. Kim, and J. Carpenter, "Multi-plane light conversion of high spatial mode count," *Proc. SPIE* **10744**, 120–125 (2018).
69. F. Pérez and B. E. Granger, "IPython: a system for interactive scientific computing," *Comput. Sci. Eng.* **9**, 21–29 (2007).
70. J. D. Hunter, "Matplotlib: a 2D graphics environment," *Comput. Sci. Eng.* **9**, 90–95 (2007).
71. E. Jones, T. Oliphant, and P. Peterson, *SciPy: Open Source Scientific Tools for Python* (2001).