

# Hybrid Propagation Physics for The Design and Modeling of Astronomical Observatories: a Coronagraphic Example

Jaren N. Ashcraft<sup>a</sup>, Ewan S. Douglas<sup>b</sup>, Daewook Kim<sup>a,b,c</sup>, A.J.E. Riggs<sup>d</sup>

<sup>a</sup>James C. Wyant College of Optical Sciences, University of Arizona, Meinel Building 1630 E. University Blvd., Tucson, AZ. 85721, USA

<sup>b</sup>Department of Astronomy and Steward Observatory, University of Arizona, 933 N. Cherry Ave., Tucson, AZ 85721, USA

<sup>c</sup>Large Binocular Telescope Observatory, University Of Arizona, 933 N. Cherry Ave. Tucson, AZ 85721, USA

<sup>d</sup>Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA

**Abstract.** For diffraction-limited optical systems an accurate physical optics model is necessary to properly evaluate instrument performance. Astronomical observatories outfitted with coronagraphs require physical optics models to simulate the effects of misalignment and diffraction, and polarization. Accurate knowledge of the observatory’s point-spread function (PSF) is integral for the design of high-contrast imaging instruments and simulation of astrophysical observations. The typical state of the art is to model the misalignment, ray aberration, and diffraction across multiple software packages, which complicates the design process. Gaussian Beamlet Decomposition (GBD) is a ray-based method of diffraction calculation that has been widely implemented in commercial optical design software. By performing the coherent calculation with data from the ray model of the observatory, the ray aberration errors can be fed directly into the physical optics model of the coronagraph, enabling a more integrated and open-source model of the observatory. We develop a formal algorithm for the transfer-matrix method of GBD, and evaluate it against analytical results and traditional physical optics model to assess the suitability of GBD for high-contrast imaging simulations. Our GBD simulations of the observatory PSF, when compared to the analytical Airy function, have a sum-normalized RMS difference of  $\approx 10^{-6}$ . These fields are then propagated through a Fraunhofer model of a exoplanet imaging coronagraph where the mean residual numerical speckle contrast is  $4 \times 10^{-11}$  within a factor of  $\approx 5$  of the equivalent Fraunhofer-only model, indicating a suitability for high-contrast imaging simulation and integrated sensitivity analysis of diffraction, polarization, and misalignment impacts for exoplanet detection. We developed this algorithm in an open-source software package and outlined a path for its continued development to increase the fidelity and flexibility of diffraction simulations using GBD.

**Keywords:** physical optics modeling, coronagraphs, exoplanets, high-contrast imaging, diffraction, Gaussian Beamlet Decomposition.

\*Jaren N. Ashcraft, [jashcraft@arizona.edu](mailto:jashcraft@arizona.edu), Ewan S. Douglas, [douglass@arizona.edu](mailto:douglass@arizona.edu)

## 1 Introduction

### 1.1 Astrophysical Motivation

Integrated models of optical observatories are highly beneficial to their design and use.<sup>1,2</sup> Accurate observatory models permit powerful insight into predicting the as-built performance of a given instrument. However, the accuracy of these models is fundamentally limited by the physics employed. To facilitate high-yield scientific observations, astronomical observatories are nominally

designed to operate in the diffraction limit ~~of the telescope~~ where wavefront aberrations are small. Diffraction-limited optical observatories are necessarily modeled with diffraction integrals derived from the Huygens-Fresnel principle to support the wave-like behavior of light. The paraxial and scalar assumptions that angles of incidence are small and that polarization is negligible<sup>?</sup> are made to ease the computational burden on the model. The resultant Fresnel and Fraunhofer diffraction integrals are accurate providing these conditions are met. If the performance of the observatory is limited by a factor outside the assumptions made, then the model will be ignorant of it. ~~An example of this is the linear and shift-invariant assumption imposed on diffraction models of astronomical observatories. Ray aberrations (e.g. coma, astigmatism) have a field dependence, and consequently change across an observatory's field of view. However, diffraction integrals assume shift-invariance. This means that the aberrations do not change across the field of view and a separate ray trace model must be used to capture this effect. An example of this is the limitation of coronagraphs to polarization aberrations.<sup>?, ?, ?, ?</sup> The aforementioned diffraction integrals are incapable of determining polarization, so a separate model must be made to include the effects of polarization on the field's complex amplitude.~~

Integrating optical models from different regimes in physics has become a popular method by which to overcome this limitation. Linking ray trace models to diffraction models in particular can overcome the paraxial and scalar assumption imposed by the Fresnel and Fraunhofer diffraction integrals. ~~In the prior example, to capture the influence of optical aberrations a new ray trace must be performed and the optical path difference of the rays must be translated to a diffraction model for each point of interest in the field of view. Similarly, diffraction integrals are incapable of determining the effects of optical polarization. For example,~~For example, the Daniel K. Inoyue Solar Telescope (DKIST) supports a suite of polarimetric instrumentation that is sensitive to the

influence of optical polarization. To support this regime of optical physics the scalar assumption is not sufficient, so the polarization state is propagated along geometric ray paths using polarization ray tracing<sup>?,?</sup> to determine the influence of polarization aberrations on the optical beam.<sup>?</sup> Modern space telescopes also require integrated models to accurately predict the instrument behavior. For example, the optical models of the James Webb Space Telescope incorporate the influence of generally dynamic thermal, structural and optical effectsmodels simultaneously to produce an accurate library of the observatory's jitter.<sup>?</sup> High-contrast imaging instruments that use coronagraphs, designed to separate exoplanets from diffracted starlight, are in dire need of integrated physical optics modeling from their inception. These coronagraphic instruments aim to discern targets that are orders of magnitude dimmer than greater than ten orders of magnitude different in brightness-with-respect-to their host star.<sup>?,?,?</sup> Understanding all sources of error is of paramount importance to the functionality of the instrument.

High-contrast imaging instruments have been successfully deployed on the ground (e.g. SCExAO,<sup>?</sup> MagAO-X,<sup>?</sup> NIRC2,<sup>?</sup> GPI,<sup>?</sup> SPHERE<sup>?</sup>) and in space (e.g. NICMOS,<sup>?</sup> NIRCcam<sup>?,?</sup>) to pursue the direct detection of extrasolar planets, debris, and protoplanetary disks. The Decadal Survey on Astronomy and Astrophysics 2020 (Astro2020) recommends pursuing these instruments for a future 6 meterm diameter infrared/optical/visible (IROUV) flagship observatory for the progression of astrophysical sciences.<sup>?</sup> Currently a gap exists between the theoretical limit on coronagraph performance and what realizations of these instruments are actually able to achieve.<sup>?</sup> This is in part due to the sensitivity of coronagraphs to low-order aberrations worsening the raw contrast achievable near the inner working angle (IWA).<sup>?</sup> To understand this system-level limitation, the connection between low-order aberrations and coronagraph performance should be analyzed early in the design process.

Presently the optical design of observatories is done in a ray-tracing engine<sup>?,?</sup> (e.g. CODE V, Zemax OpticStudio) because it is more suitable to optimizing the shapes of observatory mirrors. Upon reaching a diffraction-limited optical design, the system is then assumed to be well-represented by a paraxial diffraction model. The wavefront maps produced by the ray trace model of the observatory and the contributions from the imperfect polishing of the observatory mirrors are sent to a **linearized** physical optics propagator to examine the image plane electric field in the presence of diffraction from structure in the beam and phase errors on the optics.

Many tools have been developed to simulate the performance of high-contrast imaging instrumentation. Tiny Tim is one of the first of these widely-used packages used to simulate the Hubble Space Telescope (HST) instrument point-spread functions (PSFs).<sup>?</sup> The tool generates aberrated PSFs based on the instrument, observation, and dynamic aberrations for a given observing scenario, enabling highly accurate simulations of the observatory performance. However, it only considers aberrations that are conjugate to the exit pupil of the observatory. This limits the model's ability to capture out-of-pupil effects, like the Talbot effect and speckles from optical surfaces.<sup>?</sup> To capture these effects, optical propagation packages **integrate integrated** optical models of observatories by adding Fresnel diffraction to the PSF simulation, enabling the modeling of plane-to-plane diffraction effects. **Open-source Packages that currently support Fresnel diffraction include:** PROPER,<sup>?</sup> Physical Optics Propagation in PYthon (POPPY),<sup>?,?,?</sup> **and** High-Contrast Imaging in Python (HCIPy),<sup>?</sup> **AOTools,<sup>?</sup> and prysm<sup>?,?</sup>**. Using these tools, near-field diffraction that limits high-contrast imaging can be modeled, and focal-plane wavefront sensing methods can be tested. **The Fast Linearized Coronagraph Optimizer (FALCO) package<sup>?</sup> uses the same propagation physics to aid the design of these instruments by optimizing the commands sent to deformable mirrors to correct for the near-field diffraction effects that could limit the Roman Space Telescope**

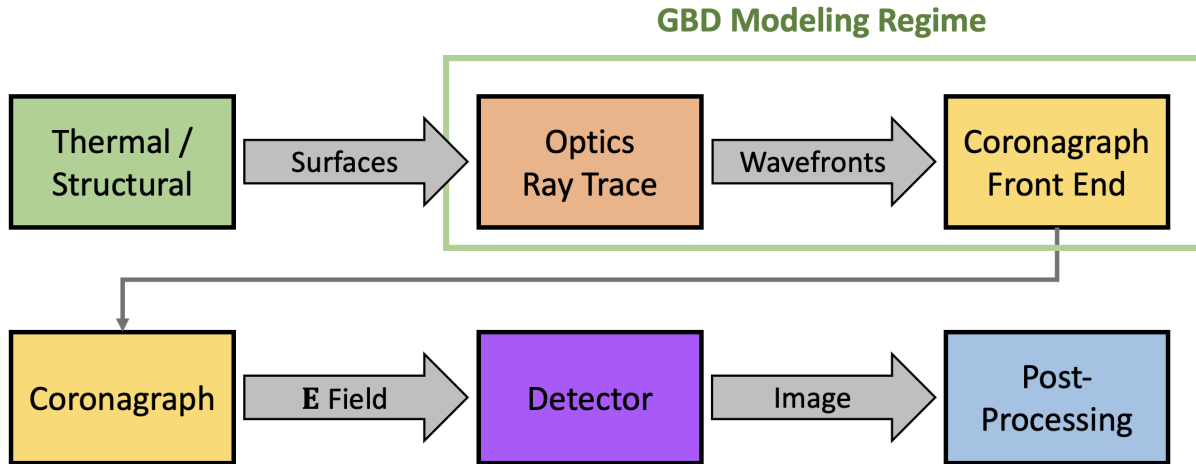
~~(Roman) Coronagraph Instrument's sensitivity.~~

These open-source physical optics propagation tools form the cornerstone of high-contrast imaging instrument modeling and design. The open-source framework means that the codes are accessible to anyone, so the physics are completely verifiable by the scientific community.<sup>?</sup> It is in the scientific community's best interest to continue to develop open-source propagation physics modules to increase the scope of and further integrate our observatory models.

Commercial optical design codes offer the ability to make diffraction calculations based on ray data, but their physical optics simulation techniques are not as transparent or versatile as the open-source propagation codes that are used to design coronagraphs for astronomical observatories. The current open-source physical optics codes used for observatory modeling are also limited in their scope because of the Fresnel approximation, which is incapable of accurately modeling the field after fast-focusing and highly aspheric surfaces.<sup>?,?</sup> As observatories get larger, their optics may become faster (i.e. lower  $F\#$ ) and more aspheric to fit within an available volume. Some coronagraph architectures capable of Earth-like exoplanet detection (e.g. phase-induced amplitude apodization, or PIAA (PIAA)<sup>?</sup>) employ mirrors that apodize the pupil with highly aspheric mirrors, and require tailored propagators in order to be included in physical optics models.<sup>?</sup> In the regime where the contribution of these surfaces is best represented by a ray trace, a diffraction calculation must be made to appropriately model the optical field at the image plane. ~~To continue the development of integrated optical models, exploring the possibilities and limitations of new propagation physics is desirable.~~

~~An example of the typical integrated modeling pipeline for astronomical observatories outfitted with coronagraphs is shown in Figure 1. The observatory is typically designed and modeled in ray trace software (e.g. CODE V, Zemax OpticStudio) to accurately model the wavefront in the~~

observatory's exit pupil. Upon finalizing the design, the complex-valued exit pupil is decomposed into a functional representation (e.g. a set of polynomial coefficients, such as the Zernikes<sup>?</sup>) and passed to the entrance pupil of a coronagraph model constructed in an open-source, Fourier-based physical optics propagator (e.g. POPPY, PROPER, HCIPy). The front-end model computes the complex field distribution at the coronagraph mask, and then propagates the field past the mask to the image plane. The flux field is taken by a model of the detector (e.g. EMCCD Detect,<sup>?</sup> Pyxel<sup>?</sup>) to create a simulated raw science image that can be post-processed (e.g. PyKLIP,<sup>?</sup> NMF imaging<sup>?</sup>).



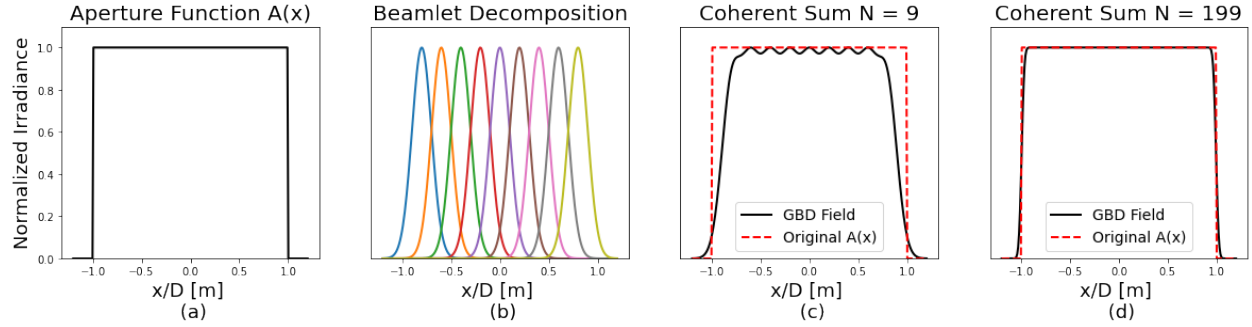
**Fig 1** Modeling flow inspired by the Structural Thermal Optical Performance (STOP) modeling process for the Roman Coronagraph.<sup>?</sup> This diagram illustrates the different modeling regimes required to create a simulated image of an observatory. We aim to further integrate this modeling pipeline by creating an open-source Gaussian Beamlet Decomposition platform to unify the ray trace model of the observatory with the diffraction model of the coronagraph.

To bridge the gap between commercial ray tracing engines and open-source physical optics propagation codes, we investigate the viability of a ray-based diffraction calculation called Gaussian Beamlet Decomposition (GBD) for designing observatories with coronagraphs. Traditionally, GBD operates using the complex ray tracing algorithm described in the works by Greynolds<sup>?</sup> and Harvey et al<sup>?</sup>. This technique has been previously implemented in FRED,<sup>?,?</sup> and possibly in

CODE V,<sup>?</sup> but an exact method of its implementation in these software packages is not clearly available in the literature. An alternative approach called the transfer matrix algorithm was recently developed to improve GBD's viability for precision diffraction simulation by Worku and Gross. An alternative approach to GBD was recently developed to improve its viability for precision diffraction simulation by Worku and Gross.<sup>?,?,?</sup> However, their implementation is not public and has not yet been formally evaluated as a tool to augment the modeling of astronomical observatories or high-contrast imaging instrumentation. To formally evaluate GBD as a modeling tool for astronomical instrumentation, a complete algorithm for its implementation is derived here.

## 1.2 Gaussian Beamlet Decomposition

GBD is a method of physical optics propagation that approximates the propagated field as a finite sum of coherent Gaussian beams. This method has been implemented in various optical design packages<sup>?</sup> to perform coherent calculations on non-paraxial systems. The operating principle of GBD is to decompose the field in the entrance pupil of an optical system (Figure 2a) into a finite set of Gaussian beams (Figure 2b). Their coherent sum (Figure 2c-d) approximates the initial field decomposition.



**Fig 2** Illustration of the operating principle of GBD in one dimension. The aperture function for traditional imaging systems is shown in (a) as a top-hat function of uniform amplitude. The decomposition of this function into a discrete set of Gaussian beams is shown in (b), which shows nine evenly-spaced Gaussian profiles before their coherent summation, which is shown in (c). The coherent sum (black) of these nine beamlets shows that the beamlets are incapable of perfectly reconstructing the original aperture function (dashed red), specifically the sharp edge and uniform amplitude. More beamlets are needed for a more accurate reconstruction, which is shown in (d) for 199 beamlets. The amplitude ripple has virtually vanished, and the field near the aperture edges is almost entirely recovered.

Under-sampling the field in the entrance pupil leads to artifacts in the decomposition. A characteristic amplitude ripple based on the period of the beamlet decomposition remains in the field. Due to the soft edges of the Gaussian beams, they cannot completely reconstruct the field of a sharp aperture edge. Using a larger number of smaller beamlets decreases the beamlet distribution period and increases the slope of the Gaussian beams, allowing for the mitigation of both of these effects (shown in Figure 2d).

Upon decomposing the initial wavefront into a sufficient set of Gaussian beams, we can use their analytical linear propagation laws<sup>?</sup> (described in Section 2 of this manuscript) to compute the coherent field at any arbitrary plane in the optical system. Fourier transform-based propagation methods derived from the Huygens-Fresnel principle typically assume the field is scalar and the optical system is paraxial.<sup>?</sup> This is usually appropriate for stellar~~astronomical~~ coronagraphs, which operate on slowly focusing beams with diffraction-limited optics, but may not be for next-generation observatories whose large apertures may necessitate relatively fast telescope optics.



Observatories are designed and modeled in ray trace software (e.g. CODE V, Zemax OpticStudio) to accurately model the wavefront in the exit pupil. Upon finalizing the design, the complex-valued exit pupil is decomposed into a functional representation (e.g. a set of polynomial coefficients, such as the Zernikes<sup>?</sup>) and sent to an open-source Fourier-based physical optics propagator (e.g. POPPY, PROPER, HCIPy) to generate the complex field at the image plane. The flux field is taken by a model of the detector (e.g. EMCCD Detect, Pyxel<sup>?</sup>) to create a simulated raw science image that can be post-processed (e.g. PyKLIP,<sup>?</sup> NMF imaging<sup>?</sup>).

GBD ~~aims to~~ computes the same complex optical field without making the paraxial assumption across the ~~observatory instrument~~. Rather, the ~~coherent propagated~~ field of a ~~generally astigmatic~~ Gaussian beams is derived from the ray data directly. Doing so imposes the paraxial assumption about a single beamlet ~~instead of the entire observatory~~, which is a much less stringent approximation. Gaussian beams are technically infinite in extent, but extremely localized around the beam waist. Consequently, the contribution of the field very far from the Gaussian is negligible. ~~This locality~~ ~~This parameter~~ enables the ~~simulation of the~~ optical system to generally be non-paraxial, ~~providing that the nearby (parabasal) rays that define the beamlet have a small optical invariant~~  $H^{?,?}$  where  $h_{1,2}$  are the coordinates of the parabasal rays on the plane transverse to propagation with an origin defined by the central ray and  $\theta_{1,2}$  are the angles between the parabasal rays and the central ray. By making the diffraction calculation directly from ray data, GBD circumvents the need for ~~translating decomposing~~ the wavefront to a physical optics propagator ~~into a set of~~ ~~polynomials~~ and imposing the paraxial assumption on the optical system. Instead, the ray trace model can be directly integrated into the diffraction model.

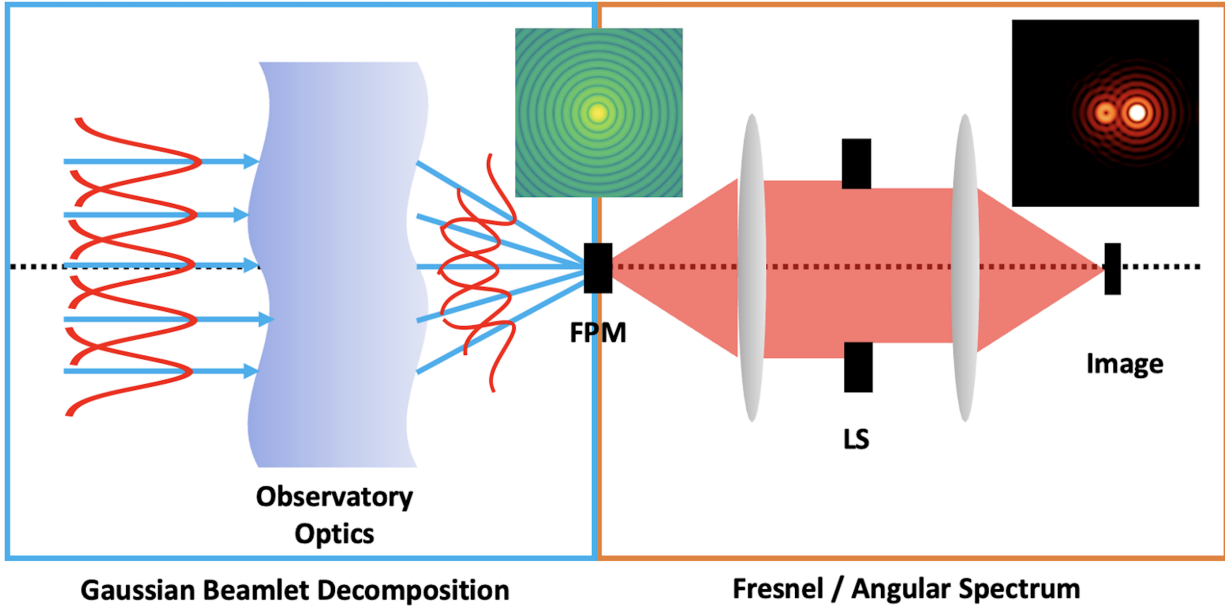
There are two main approaches that exist in the literature to implement GBD: the complex ray tracing method, and the transfer matrix method. The complex ray tracing method was recently de-

scribed by Harvey et al<sup>?</sup> in their seminal paper about implementing GBD in Photon Engineering's non-sequential ray tracing software FRED. This method traces waist and divergence rays to compute the complex field at the plane of interest using Arnaud's method of complex ray tracing.<sup>?,?</sup> Through FRED, the complex ray tracing method has seen widespread use for nonparaxial coherent beam analysis.

**Another GBD approach,** the transfer matrix method was developed by Worku and Gross<sup>?,?,?</sup> to mitigate GBD's inability to simulate sharp-edge diffraction and add new utility to the technique. This formulation of GBD works by computing the differential ray transfer matrix for a given ray path and then using that data to solve the Gaussian beam solution to the general Collins' integral.<sup>?</sup> Worku and Gross have leveraged the general Collins' integral to provide alternative conditions to the Gaussian beam solution to modify the decomposition, such as truncated<sup>?</sup> and pulsed<sup>?</sup> beamlet decomposition. The option of modifying the **Gaussian** beamlets to overcome the limitations of GBD makes the transfer matrix method extremely attractive for use in high-contrast imaging where preservation of high-spatial frequency content is important. Of particular interest are mirror segment gaps and opto-mechanical structures that obscure the primary mirror. Therefore, we elect to investigate Worku and Gross's transfer matrix method of GBD for the work presented in this manuscript. **Our goal is to publicize the transfer matrix method by developing an algorithm for its implementation, and then use it to characterize GBD's suitability for high-contrast imaging simulation. Our work can then be used as a platform with which to study the suitability of Worku and Gross' modified GBD by future investigators.**

### 1.3 Hybrid Propagation Physics

The ray-based nature of GBD introduces problems in modeling the electric field when the rays are vignetted. Structure in the field where the initial decomposition occurs (typically, the entrance pupil) can be well-represented by Gaussian beams as long as the structure of interest is larger than the beamlets used in decomposition. Diffraction from structure in intermediate planes (between the pupil and focal planes) is challenging to represent if the beamlets diverge considerably. However, secondary beamlets can be traced from these intermediate structures to aid in the accuracy of the simulation.<sup>2</sup> At the focal plane of a diffraction-limited system, the rays are highly concentrated while the diffracted field spreads out considerably (e.g. the Airy disk). Because of GBD's reliance on ray tracing, it cannot represent diffraction from structure in the focal plane well without re-decomposing the field.<sup>2</sup> For the case of a Lyot-type coronagraph all rays are vignetted at the focal plane mask (FPM) and the field decomposition is ~~consequently~~ lost. To circumvent this we compute the field before the FPM with GBD and propagate it through the remaining coronagraph with traditional diffraction integrals, where we expect the low-order aberrations to be small and the paraxial assumption to be valid. This *hybrid* method (shown in Figure 3) enables the user to alter the propagation physics for the electric field based on where it is the most appropriate; GBD will simulate the fast beams in the fore-optics and paraxial diffraction will propagate the field through the coronagraphic imaging optics.



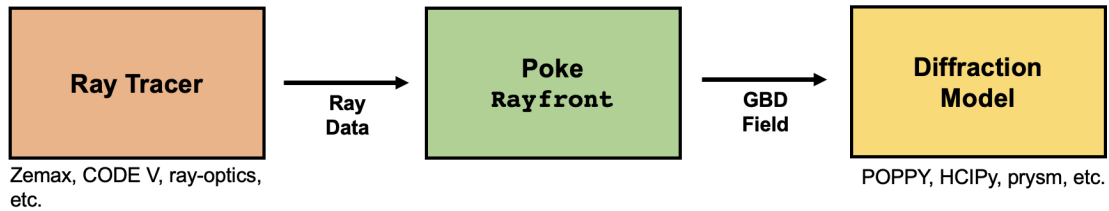
**Fig 3** Diagram demonstrating the hybrid propagation physics model. The observatory optics are best described by a ray-based propagation model, so GBD is used to compute the field at the image plane before the coronagraph focal plane mask (FPM). The field array is then passed to a paraxial diffraction model which propagates it past the FPM and through the remainder of the coronagraph. This propagation scheme permits the user to model the influence of the observatory optics non-paraxially, without losing accuracy after propagating past the focal plane mask.

The end result of such a model allows for direct integration of the ray trace model with the physical optics model, without imposing the paraxial approximation on the observatory. Like Fresnel and Fraunhofer diffraction, GBD is an approximation to diffraction physics. The decomposition of the field into Gaussian beams does not have an analytical solution. Therefore, undesirable artifacts can be introduced into the field if the decomposition is not well-understood and the sampling is insufficient.<sup>?</sup> To better understand the impact of a GBD PSF on high-contrast imaging simulations, we develop a hybrid propagation ~~model~~~~module~~ to compare GBD to an equivalent Fraunhofer diffraction model.

To our knowledge, the transfer matrix method of GBD has not seen widespread implementation. Given its obvious benefits, we believe that this is because the transfer matrix method is not well-understood by the scientific community. We aim to remedy this by presenting a vectorizeable

algorithm for the transfer matrix method, and open-sourcing our simulation platform for future investigators to use. This manuscript is the first work, to our knowledge, to provide the explicit mathematics of GBD *and* provide our code as an object-oriented module for more widespread use.

Our GBD module was built in the Poke<sup>?</sup> Python package currently available on Github. Poke was originally developed to be a polarization ray tracing module to study the influence of polarization aberrations on astronomical coronagraphs,<sup>?</sup> but we expanded it to include GBD. Poke operates by using ray tracer API's to trace a raybundle through every surface in the optical system. The relevant ray data is stored in a Rayfront Python object and can be loaded into a Python environment and interacted with independent of the raytracer that generated the data. The Rayfronts can also be compiled into binary file types using the msgpack<sup>?</sup> package and distributed to any interested investigator, effectively open-sourcing the physical optics calculations done on ray data.



**Fig 4** Illustration of Poke’s use as an interface to open-source ray-based physical optics. Poke only requires an interface between a ray tracing engine (orange, left) to generate and save a Rayfront object, which includes all ray data necessary for the GBD calculation. Currently, Poke supports sequential systems in CODE V and Zemax, but we are working on adding support for open-source packages that have ray tracers, like ray-optics.<sup>?</sup> The GBD field that Poke computes can then be sent to any open-source diffraction modeling package (yellow, right) to complete the hybrid propagation model.

In this study we conduct ray traces in Zemax, which are then saved as a Poke Rayfront object. Poke performs the GBD simulations using the saved ray data to generate the field at the focal plane of the telescope. This data is exported to a coronagraph model built using HCIPy.

In [section 2](#) we outline the mathematics of Gaussian beam propagation and differential ray tracing used to perform GBD simulations. In [section 3](#) we present the mathematical algorithm

for the transfer-matrix method of GBD that we developed for this investigation. In [section 4](#) we compare the results of observatory PSFs produced by GBD with one produced using traditional diffraction methods, and analyze the artifacts that remain in the field. In [section 5](#) we assess the suitability of GBD for high-contrast imaging models and establish a road map for our module's continued development.

## 2 Preliminary Mathematical Methods

~~To assess the viability of GBD for high-contrast imaging simulation we compare the PSF produced by GBD to other diffraction methods that represent the state of the art. Traditional GBD is known to not reconstruct high spatial frequency information because of its inability to recreate sharp edges exactly and residual ripple that is left from the beamlet distribution,<sup>2</sup> but this has not been quantitatively studied in the literature.~~

In this section we review the ~~necessary mathematical tools that we need to formulate our GBD algorithm. This includes the~~ propagation equations for a single Gaussian beam, ~~how to use~~ review differential ray tracing ~~to arbitrarily compute the parameters necessary~~ for Gaussian beam propagation, and examine a few methods of decomposing the entrance pupil field in the optical system to improve the simulation's sensitivity to high-spatial frequencies.

### 2.1 *Propagation of a single Gaussian beam* ~~Gaussian Beam Parameters~~

~~To propagate an arbitrary optical beam the field must be decomposed into a finite sum of Gaussian beamlets which are then independently propagated.~~ The equation for a single Gaussian Beam is

parameterizeddescribed entirely by the complex beam parameter  $Q(z)$ ,<sup>?</sup>

$$U(r, z) = \frac{U_o}{Q(z)} \exp\left[ik \frac{r^2}{2Q(z)}\right], \quad (1)$$

where  $U_o$  is the amplitude,  $k$  is the wavenumber, and  $r$  is the radial coordinate in the plane perpendicular to propagation. The complex parameter  $Q(z)$  describes the beam's  $1/e$  field radius (the “waist”  $w_o$ ) and wavefront radius of curvature  $R(z)$ ,

$$Q(z)^{-1} = \frac{1}{R(z)} + i \frac{\lambda}{\pi w(z)^2}. \quad (2)$$

$Q(z)$  is a convenient expression of the Gaussian beam because it fully encapsulates the information required to describe the transverse electric field of the beam as it propagates. The real part of  $Q(z)$  is related to the radius of curvature of the wavefront,

$$R(z) = z \left(1 + \left(\frac{Z_o}{z}\right)^2\right), \quad (3)$$

where  $Z_o$  is the Rayleigh range and  $z$  is the longitudinal propagation distance. The imaginary part of  $Q(z)$  is related to the beam waist radius,

$$w(z) = w_o \sqrt{1 + \left(\frac{z}{Z_o}\right)^2}. \quad (4)$$

In the paraxial regime  $Q(z)$  can be propagated using the ABCD ray transfer matrices of geometrical optics<sup>?</sup>.

$$Q(z)_o^{-1} = \frac{C + DQ_i^{-1}}{A + BQ_i^{-1}} \quad (5)$$

288 For the generally astigmatic case,  $Q(z)$  is a 2x2 matrix  $\mathbf{Q}(z)$  that encodes the complex curvature  
 289 in two orthogonal directions and how they couple into each other.<sup>?,?</sup>

$$\mathbf{Q}(z)^{-1} = \begin{pmatrix} Q(z)_{xx}^{-1} & Q(z)_{xy}^{-1} \\ Q(z)_{yx}^{-1} & Q(z)_{yy}^{-1} \end{pmatrix} \quad (6)$$

290 This treatment allows for greater versatility in the beamlet propagation, but requires that the  
 291 elements of the ray transfer matrices are also 2x2 matrices. The propagation formula shown in  
 292 equation 5 has a similar matrix extension.

$$\mathbf{Q}(z)_o^{-1} = (\mathbf{C} + \mathbf{D}\mathbf{Q}_i^{-1})(\mathbf{B} + \mathbf{A}\mathbf{Q}_i^{-1})^{-1} \quad (7)$$

293 We can then express the propagated Gaussian beam with equation 8

$$U(\mathbf{r}) = \frac{U_o}{\sqrt{\det|\mathbf{A} + \mathbf{B}\mathbf{Q}_1^{-1}|}} \exp\left[\frac{-ik}{2} \mathbf{r}^T \mathbf{Q}_2^{-1} \mathbf{r}\right], \quad (8)$$

294 Where  $\mathbf{r}$  is the radial coordinate in the plane transverse to propagation. The formulation for the  
 295 propagation of the complex curvature matrix allows for modeling Gaussian beams as they propa-  
 296 gate along generally skew ray paths in non-axially symmetric optical systems. This is an important  
 297 utility for diffraction modeling of wavefront aberrations introduced by system misalignment or  
 298 thermal deformations, which generally break optical system symmetry.

## 299 2.2 Computing the Differential Ray Transfer Matrix

300 The ABCD ray transfer matrix is a useful and concise method for analyzing properties of ray paths  
 301 along optical systems. In the regime of geometrical optics, a generally skew ray can be traced



302 through a system using 4x4 ABCD ray transfer matrices.<sup>?</sup> These matrices model simple optical  
 303 elements (e.g. thin lenses) with ease by operating on an input column vector that represents a light  
 304 ray. The simplest ray transfer matrix that describes a paraxial and orthogonal optical system is a  
 305 2x2 operator that maps an input ( $i$ ) spatial and angular coordinate to the appropriate output ( $o$ ),

$$\begin{pmatrix} y_o \\ m_o \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} y_i \\ m_i \end{pmatrix}, \quad (9)$$

306 where  $y$  is the spatial coordinate transverse to the propagation direction, and  $m$  is the slope in that  
 307 dimension. The elements of the ABCD matrix and ray vectors are real-valued scalars. To account  
 308 for skew ray paths, the position and angle in the dimension orthogonal to  $y$  and the direction of  
 309 propagation must be tracked, adding two dimensions to the matrix calculus. A 4x4 ABCD matrix  
 310 describes a nonorthogonal system with tilts and decenters that map generally skew input rays to  
 311 generally skew output rays,

$$\begin{pmatrix} x_o \\ y_o \\ l_o \\ m_o \end{pmatrix} = \begin{pmatrix} A_{xx} & A_{xy} & B_{xx} & B_{xy} \\ A_{yx} & A_{yy} & B_{yx} & B_{yy} \\ C_{xx} & C_{xy} & D_{xx} & D_{xy} \\ C_{yx} & C_{yy} & D_{yx} & D_{yy} \end{pmatrix} \begin{pmatrix} x_i \\ y_i \\ l_i \\ m_i \end{pmatrix}. \quad (10)$$

312 For simplicity, it is convenient to represent that radial position in the plane transverse to prop-  
 313 agation ( $x, y$ ) and the corresponding direction in the dimension ( $l, m$ ) as a position and angle  
 314 vector respectively ( $\mathbf{r}, \boldsymbol{\theta}$ ). The ABCD matrix can similarly be condensed into 2x2 sub matrices

315 that operate on each spatial dimension, yielding a familiar notation,

$$\begin{pmatrix} \mathbf{r}_o \\ \theta_o \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} \begin{pmatrix} \mathbf{r}_i \\ \theta_i \end{pmatrix}. \quad (11)$$

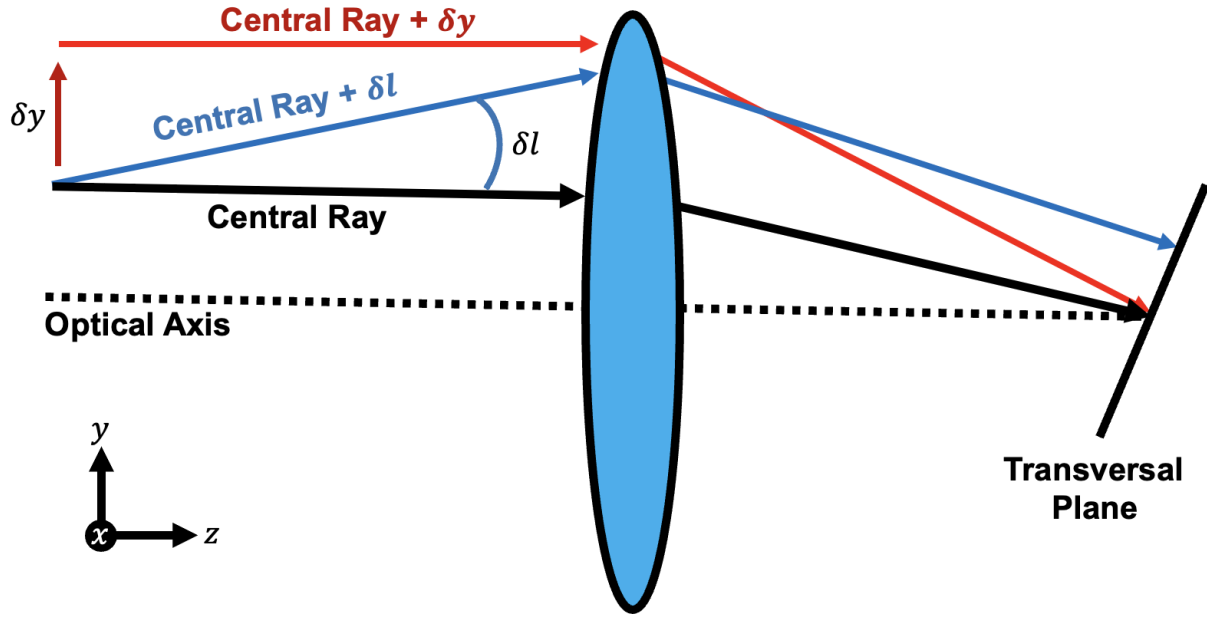
316 This description is powerful because it communicates the elegance and simplicity of ray trans-  
317 fer matrices. All dimensions transverse to propagation are accounted for, but the calculus to propa-  
318 gate a ray is still the same. The ray transfer matrices for simple and paraxial optical elements (e.g.  
319 thin lens, distance) are well known,<sup>?</sup> and were used in concert with GBD in a prior investigation  
320 with paraxial systems.<sup>?</sup> However, our aim is to use GBD to model *non*-paraxial optical system  
321 diffraction. Therefore, we need a method of computing the ray transfer matrix for **an arbitrary**  
322 **a-generally** skew ray-path. A simple dimensional analysis of the matrix relation in Equation 11  
323 is a good place to start understanding how to construct an arbitrary ABCD matrix. Because the  
324 position element  $\mathbf{r}$  of the ray vector must be in units of distance, and the angular element  $\theta$  must be  
325 dimensionless, the units of the ABCD matrix elements are constrained.  $\mathbf{A}$  and  $\mathbf{D}$  must be dimen-  
326 sionless and transform the ray position and angle through the optical system, indicating that they  
327 represent magnification.  $\mathbf{B}$  and  $\mathbf{C}$  must have units of distance and inverse distance, respectively.  $\mathbf{B}$   
328 operates on an angle, and is therefore a metric of propagation distance through an optical system  
329 given some ray angle.  $\mathbf{C}$  operates on a position, and is therefore an indicator of the amount of  
330 refraction a ray experiences given a position in the entrance pupil.

331 Stone and Forbes’<sup>?</sup> work in differential ray tracing for inhomogeneous media was instructive  
332 in terms of deriving a method to construct the ABCD matrix. They illustrate the construction of  
333 individual optical elements through ray derivatives in a generally 4x4 matrix through derivatives of

surface data. Using this method the position and angular derivatives of an optical surface are taken and arranged in a matrix like in Equation 11. The matrix product of the optical elements is then the final differential ray transfer (or ABCD) matrix. Their method is functional if the analytical expression of the optical elements are known, but could be very computationally intensive if the optical system has many elements. Instead, we approximate the differential matrix by tracing additional rays, and compute the finite difference of the ray coordinates and directions at the input and output of the optical system to approximate the derivative.

Our implementation of the differential ray tracing technique in poke utilizes a user-specified ray tracer (e.g. Zemax OpticStudio) the Zemax OpticStudio ray-tracing engine that propagates rays by computing Snell's Law at each surface. The ray coordinates of interest are on the source plane (where the field decomposition is done, e.g. the entrance pupil) and the detector plane (where the propagated field is evaluated, eg. the location of a camera sensor). A local coordinate system is configured on both planes, and the ray positions and directions are computed on these planes. A local coordinate system is defined for both the input and output surface, and the ray data is computed on both of these surfaces. An element of the ray transfer matrix can be computed by determining the ratio of the differential parameters from the data on the detector plane to the source plane. An example of computing the element  $A_{y,y}A_{x,x}$  is given by equation 12. In this example the central ray (shown in black on figure 5) is traced along with a ray with a differential addition in input  $y$  coordinate (shown in red on figure 5). The difference in  $y$  coordinates of the central and differential rays determine the derivative.

$$A_{yy} = \frac{\partial y_{detector}}{\partial y_{source}} = \frac{y_{detector,+y} - y_{detector,o}}{y_{source,+x} - y_{source,o}} \quad (12)$$



**Fig 5** Diagram illustrating differential ray tracing in the 2D case for a simple lens system. The central ray (black) is propagated along with two paraxial rays with a differential addition in the  $y$  direction (red) and another with a differential addition in the  $l$  direction (blue). To determine the ABCD matrix the ray data is computed on the transversal plane, which is normal to the central ray.

The ray transfer matrix for a non-orthogonal optical system has 16 unknowns (see ABCD matrix in equation 10), and each ray yields 4 quantities. To solve for every element of the matrix 4 linearly independent rays must be traced. The simplest ray set is geometrically orthogonal,<sup>7</sup> where copies of the central ray  $(x, y, l, m)$  are modified by a differential quantity  $(\delta)$  in each of the 4 ray coordinates, two in position  $(\delta x, \delta y)$  and two in slope  $(\delta l, \delta m)$ . The differential ray set is given by equation 13

$$\begin{pmatrix} x + \delta x \\ y \\ l \\ m \end{pmatrix}, \begin{pmatrix} x \\ y + \delta y \\ l \\ m \end{pmatrix}, \begin{pmatrix} x \\ y \\ l + \delta l \\ m \end{pmatrix}, \begin{pmatrix} x \\ y \\ l \\ m + \delta m \end{pmatrix}, \quad (13)$$

360 which are traced in addition to the central ray of interest. ~~is propagated along with four rays that~~  
 361 ~~enable the computation of the local derivatives in position  $(x, y)$  and angle  $(l, m)$ .~~ The full differ-  
 362 ential ray transfer matrix is given by Equation 14. The ray transfer matrix is purely a function of  
 363 the Cartesian position of the ray  $(x, y)$  and slope of the ray in those directions  $(l, m)$  at the input  
 364 and output of the optical system. ~~An example of propagating a ray from the source plane to the~~  
 365 ~~detector plane is shown in equation 14. Here the subscript  $S$  refers to the coordinate on the source~~  
 366 ~~plane, and the subscript  $D$  refers to the coordinate on the detector plane.~~

$$\begin{pmatrix} x_D & y_D & l_D & m_D \end{pmatrix} = \begin{pmatrix} \frac{\partial x_D}{\partial x_S} & \frac{\partial x_D}{\partial y_S} & \frac{\partial x_D}{\partial l_S} & \frac{\partial x_D}{\partial m_S} \\ \frac{\partial y_D}{\partial x_S} & \frac{\partial y_D}{\partial y_S} & \frac{\partial y_D}{\partial l_S} & \frac{\partial y_D}{\partial m_S} \\ \frac{\partial l_D}{\partial x_S} & \frac{\partial l_D}{\partial y_S} & \frac{\partial l_D}{\partial l_S} & \frac{\partial l_D}{\partial m_S} \\ \frac{\partial m_D}{\partial x_S} & \frac{\partial m_D}{\partial y_S} & \frac{\partial m_D}{\partial l_S} & \frac{\partial m_D}{\partial m_S} \end{pmatrix} \begin{pmatrix} x_S \\ y_S \\ l_S \\ m_S \end{pmatrix} \quad (14)$$

367 The differential ray tracing method ~~using positive-valued differentials shown above~~ is the “for-  
 368 ward” finite difference method of approximating the ray data derivative. In practice similar results  
 369 would be achieved from the “reverse” finite difference method where the  $\delta$ ’s are negatively val-

ued. A more accurate computation uses the “central” finite differences method where two rays are traced for each dimension orthogonal to propagation in addition to the central ray, and their finite differences are averaged. This is remarkably similar in complexity to the description of the aforementioned complex ray tracing algorithm in Section 1 where as many as nine rays are traced to propagate for the generally astigmatic Gaussian beam.<sup>?</sup> ~~The same rays can be traced, but by virtue of a different implementation, we have more possibilities for development through the general Collins’ integral.<sup>?,?</sup>~~ Because Gaussian beamlets are propagated along these geometric ray paths, computing the differential ray transfer matrix enables the propagation of a beamlet through an arbitrary optical system that does not meaningfully scale in computation time with the number of optical surfaces. ~~Using this matrix, any open-source diffraction model can be linked to a ray trace model of an observatory.~~

### 2.3 Entrance Pupil Spatial Decomposition

~~The final variable to constrain in GBD is how to appropriately decompose the field in the entrance pupil into a finite set of Gaussian beams. This problem is illustrated in 1D earlier in figure 2, but the actual decomposition is a 2D problem. The operating principle of GBD is to decompose the field in the optical system’s entrance pupil into a coherent sum of Gaussian beamlets.~~ The fundamental ~~Gaussian mode mode Gaussian~~ does not represent a complete set,<sup>?</sup> and therefore the decomposition of the field is not unique. We must carefully consider how the beamlets are distributed in the entrance pupil for accurate diffraction calculations.

Various sample schemes exist in the literature, with different strengths and weaknesses. The *even Cartesian* sample scheme (shown on the left in figure 6) described in Harvey et al<sup>?</sup> is the most straightforward, where the beamlets lie evenly spaced along a Cartesian grid. The ray coordinates

in the entrance pupil are then computed from an overlap factor (OF) which describes the overlap of the beamlets'  $1/e$  waist radii  $w_o$ ,

$$OF = \frac{N_g 2w_o}{W}, \quad (15)$$

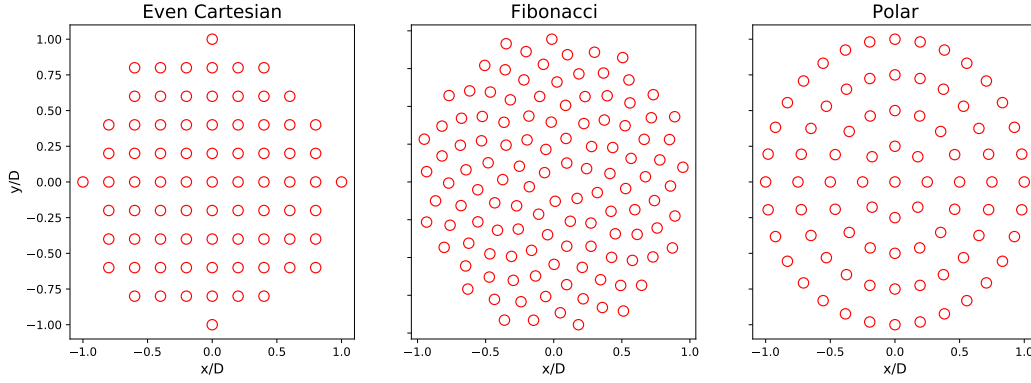
where  $N_g$  is the number of Gaussian beamlets across an aperture, and  $W$  is the width of the aperture. This feature is easy to implement and understand, but for under-sampled cases it introduces artifacts due to the ripple from the distribution and soft edge left by the Gaussian beamlets.

The *Fibonacci* sample scheme (shown on the middle in figure 6) introduced by Worku and Gross places the beamlets along a Fibonacci spiral, which results in a more accurate decomposition for circular apertures.<sup>?</sup> The distribution of the beamlets is even along polar angles on the spiral. The polar distribution of the beamlets is given by a position  $R$  and angle  $\Theta$ ,

$$R = \frac{W}{2} \sqrt{N_g}, \quad (16)$$

$$\Theta = \frac{2\pi}{\phi^2} N_g, \quad (17)$$

where  $\phi$  is the golden ratio and  $N_g$  is the total number of beamlets to trace. Even polar sampling (shown on the right in figure 6) is also a viable method to increase the accuracy of the decomposition for fewer beamlets assuming the optical system has a circular aperture,<sup>?</sup> but was not explored in this study due to the apparent advantages of the Fibonacci sample scheme, which are shown in section 4

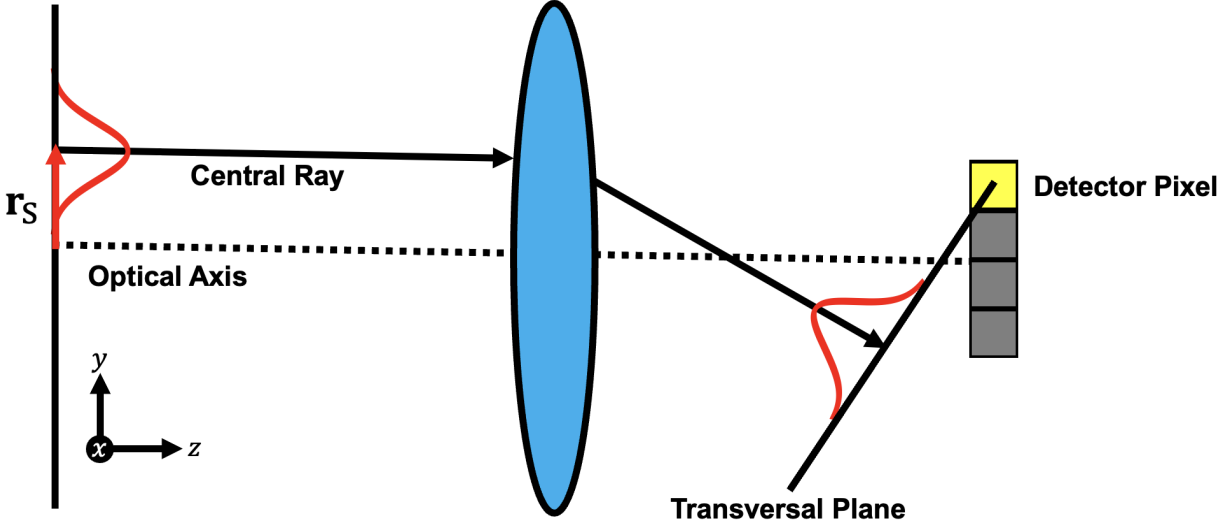


**Fig 6** Illustration of the Even (left), Fibonacci (middle), and Polar (right) sample schemes for decomposing the field at the entrance pupil of the optical system with approximately the same number of beamlets in each figure. Quantifying the ramifications of these sample schemes is of paramount importance to accurate diffraction simulation.

### 3 The Proposed Beamlet Propagation algorithm

The transfer matrix method of GBD is a flexible variant to characterize because of the recent developments by Worku and Gross to include modified GBD.<sup>?</sup> However, because the implementation is not public we must derive the full algorithm using the concepts described in section 2. The basic concept of propagating a single beamlet through an arbitrary optical system is illustrated in figure 7. A Gaussian beam is placed at some position  $r_S$  in the source plane where the initial decomposition occurs. The central ray (shown in black on figure 7) that emanates from the peak of this Gaussian beam is traced to the detector plane using a ray tracing engine. The data of the central ray at the source plane ( $\mathbf{r}_{ray,S}, \mathbf{k}_{ray,S}$ ) and detector plane ( $\mathbf{r}_{ray,D}, \mathbf{k}_{ray,D}$ ) (shown in figure 8) are used to propagate the ray to the "transversal plane", which is normal to the central ray and intersects the detector pixel where we are evaluating the field (shown in yellow on figure 7). The differential rays are also traced to the transversal plane in order to compute the ABCD matrices that describe the transformation of the Gaussian beam through an arbitrary ray path, and compute the final field incident on the detector pixel.





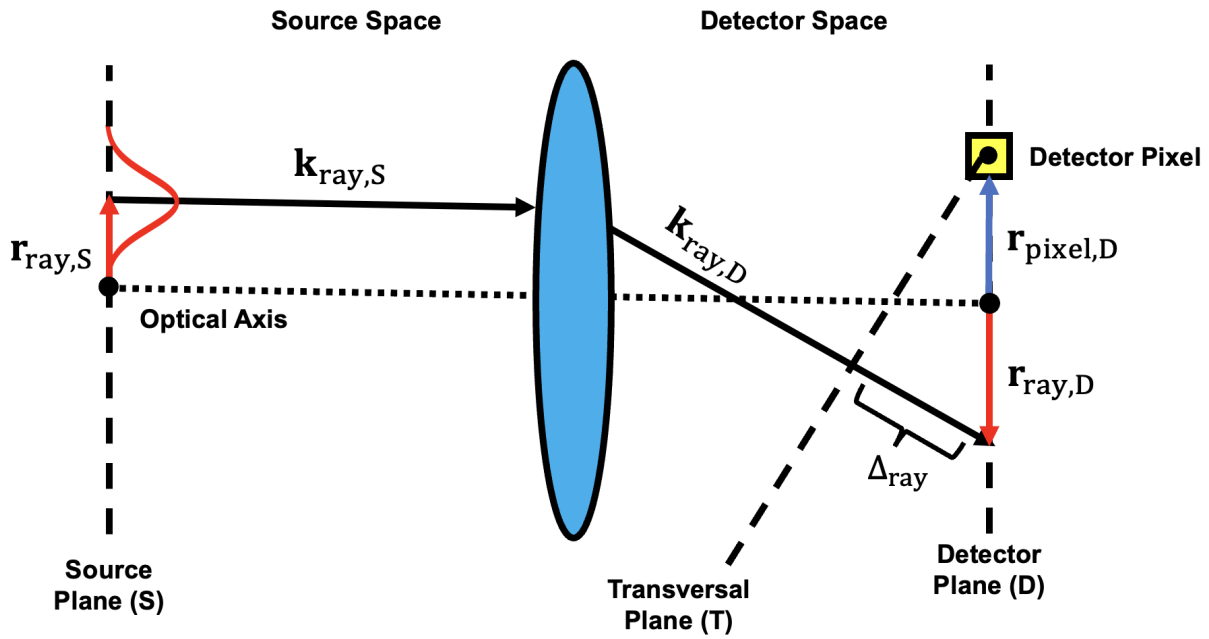
**Fig 7** Illustration of the concept for transfer-matrix Gaussian beamlet decomposition. The principle is to propagate the central ray that emanates normal to the peak of the Gaussian beam (red) in the entrance pupil to the transversal plane that intersects a given point on a detector (yellow pixel). The Gaussian field is evaluated at this intersection, and the process is repeated for each pixel and each ray. The coherent superposition of the Gaussian beams yields the field observed on the detector.

We present the **mathematical algorithm** in its entirety below for a single beamlet because the propagation method is identical for each beamlet. Doing this for many beamlets is as simple as coherently summing the computed fields at the detector pixels. We refer to several vectors in the algorithm below that are referred to using the convention  $\mathbf{x}_{y,z}$ .  $\mathbf{x}$  is data type of the vector, typically  $\mathbf{r}$  if a position or  $\mathbf{k}$  if a direction.  $y$  is what item the vector belongs to, "cen" means it belongs to the central ray, "+x" means it belongs to the +x differential ray, "pix" means a pixel on the detector plane.  $Z$  refers to the coordinate system of the vector,  $D$  for "detector", or  $T$  for "transversal". The relevant parameters used in the propagation algorithm are shown in figure 8, and are referenced throughout the procedure below.

### 3.1 Propagating rays to the transversal plane

A GBD simulation begins by running a ray trace through an optical system in the user's preferred

design code. GBD needs only compute the ray data at the plane where the decomposition occurred (the source plane) and the plane where we choose to observe the field (the detector plane). This is shown for the simple case of a lens with the detector positioned far from focus on figure 8. The source plane (leftmost dashed line) is where the rays used for GBD are started. The detector plane (rightmost dashed line) intersects the ray defined by  $\mathbf{k}_{ray,D}$  and includes the pixel that we are evaluating the field at, positioned at coordinate  $\mathbf{r}_{pixel,D}$



**Fig 8** Illustration of relevant ray parameters necessary to propagate the ray data to the transversal plane, where we evaluate the Gaussian field that intersects with a detector pixel. A ray that defines a Gaussian beam is placed on the source plane (leftmost vertical dashed line) with some direction ( $\mathbf{k}_{ray,S}$ ) and position ( $\mathbf{r}_{ray,S}$ ) and propagated through the system. At the detector plane (rightmost vertical dashed line), the ray's direction and positions ( $\mathbf{k}_{ray,D}$  and  $\mathbf{r}_{ray,D}$  respectively) have been transformed by the optical system. To propagate the rays to the transversal plane (tilted dashed line) that intersects the pixel at  $\mathbf{r}_{pixel,D}$  where the field evaluation occurs, it must be moved a distance  $\Delta_{ray}$  along the ray path in detector space.

Propagating a ray to the transversal plane entails a simple free-space propagation. To do so, we find the intersection of the line defined by the central ray  $\mathbf{k}_{ray,D}$  and the plane normal to the central ray of the Gaussian beam  $\mathbf{k}_{cen,D}$  and intersecting the detector pixel  $\mathbf{r}_{pixel,D}$ . The plane is defined by equation 18

$$\mathbf{k}_{cen,D} \cdot (\mathbf{r} - \mathbf{r}_{pixel,D}) = 0, \quad (18)$$

442 and the line along the ray is defined by equation 19

$$\mathbf{r} = \mathbf{r}_{ray,D} + \mathbf{k}_{ray,D} \Delta_{ray}. \quad (19)$$

443 Where  $\cdot$  denotes the dot product and  $\mathbf{r}$  is the space of all points that satisfy equations 18 and  
 444 19. To find the intersection of these points, we substitute  $\mathbf{r}$  in equation 18 for equation 19, which  
 445 yields the relation in equation 20

$$\mathbf{k}_{cen,D} \cdot (\mathbf{r}_{ray,D} + \mathbf{k}_{ray,D} \Delta_{ray} - \mathbf{r}_{pixel,D}) = 0. \quad (20)$$

446 Solving this equation for  $\Delta_{ray}$  yields the distance the ray needs to be propagated along it's own  
 447 path in free space to intersect the transversal plane.

$$\Delta_{ray} = - \frac{\mathbf{k}_{cen,D} \cdot (\mathbf{r}_{ray,D} - \mathbf{r}_{det,D})}{\mathbf{k}_{cen,D} \cdot \mathbf{k}_{ray,D}} \quad (21)$$

448 This procedure must be conducted for the central ray, and the four differential rays in order to  
 449 propagate them all to the transversal plane. The differential rays are propagated by substituting the  
 450 direction and position of the differential ray for  $\mathbf{r}_{ray,D}$  and  $\mathbf{k}_{ray,D}$  respectively, where the subscript  
 451 *ray* denotes the ray being traced, described in detail in table 1

subscript ( $_{ray}$ )	Description
$cen$	Central Ray
$+x$	Differential ray with additional $+\delta x$ in x coordinate
$+y$	Differential ray with additional $+\delta y$ in y coordinate
$+l$	Differential ray with additional $+\delta l$ in l direction cosine
$+m$	Differential ray with additional $+\delta m$ in m direction cosine

**Table 1** Table of subscripts describing the five rays that define the propagation of a generally astigmatic Gaussian beam. The central ray emanates from the peak of the beamlet and propagates through the optical system along with the closely spaced differential rays.

We use the distances computed using equations 21 to update the positions of their respective rays by invoking equation 19. The propagated ray position is given in equation 22  $\mathbf{r}'_{ray,D}$

$$\mathbf{r}'_{ray,D} = \mathbf{r}_{ray,D} + \mathbf{k}_{ray,D} \Delta_{ray}. \quad (22)$$

Doing so for all rays in table 1 yields the positions of the central ray and associated differential rays on the transversal plane. We can then use these rays to evaluate the ray transfer matrix that the Gaussian beam experiences by propagating to the transversal plane. But before the Gaussian field is evaluated, we must transform the coordinates computed above from the basis vectors of the detector plane to the transversal plane.

### 3.2 Transformation to the transversal plane

Any 3 dimensional coordinate system can be expressed with a series of basis vectors. An orthogonal basis in 3D is given by three unit vectors that point along the three orthogonal dimensions. On the detector plane, we define the basis vectors  $\alpha, \beta,$  and  $\gamma$  (shown on figure 9 to the right of the detector plane), to be the directions along the  $x, y,$  and  $z$  direction respectively in the coordinates of the detector plane.

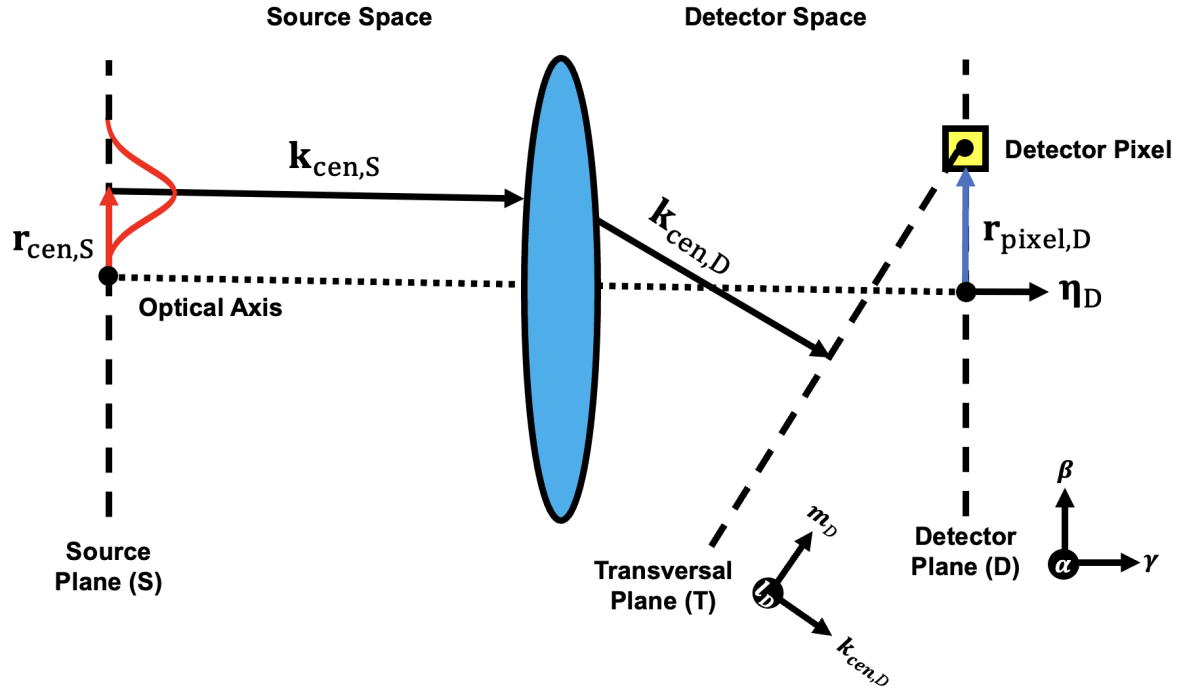


Fig 9 Caption

Similarly, the transversal plane is defined by basis vectors  $\mathbf{l}_D, \mathbf{m}_D, \mathbf{k}_{cen,D}$ , which are the analogous  $x, y$ , and  $z$  directions for the transversal plane. To compute these direction cosines, we start by taking the cross product of the central ray  $\mathbf{k}_{cen,D}$  with the surface normal of the detector  $\eta_D$  (shown in black on figure 8)

$$\mathbf{l}_D = \mathbf{k}_{cen,D} \times \eta_D. \quad (23)$$

A feature of equation 23 is that the  $\mathbf{l}_D$  is orthogonal to both the central ray and detector normal. Similarly, to determine our final basis vector which is mutually orthogonal to both the central ray and  $\mathbf{l}_D$  basis vector we compute their cross product

$$\mathbf{m}_D = \mathbf{k}_{cen,D} \times \mathbf{l}_D. \quad (24)$$

These vectors form a complete basis that describe the coordinate system of the plane normal to the central ray. Therefore, we can construct an orthogonal transformation matrix that performs a rotation of basis from the detector plane to the transversal plane. This matrix is shown in equation 25

$$\mathbf{O} = \begin{pmatrix} l_{\alpha} & l_{\beta} & l_{\gamma} \\ m_{\alpha} & m_{\beta} & m_{\gamma} \\ k_{\alpha} & k_{\beta} & k_{\gamma} \end{pmatrix}, \quad (25)$$

Where  $\mathbf{l}_D, \mathbf{m}_D$ , and  $\mathbf{k}_{cen,D}$ , are written as  $l, m, k$  respectively for brevity and are shown in equation 25 as their components projected onto the vectors  $\alpha, \beta$ , and  $\gamma$ . The matrix  $\mathbf{O}$  gives us the tool we need to express our ray coordinates in terms of the transversal plane basis. This is done by a simple multiplication of the ray position and direction vectors ( $\mathbf{r}'_{ray,D}, \mathbf{k}_{ray,D}$ ) with the matrix, shown in equations 26 and 27

$$\mathbf{r}_{ray,T} = \mathbf{O} \mathbf{r}'_{ray,D}, \quad (26)$$

$$\mathbf{k}_{ray,T} = \mathbf{O} \mathbf{k}_{ray,D}. \quad (27)$$

The same must also be done for the pixel coordinate of interest  $\mathbf{r}_{pixel,D}$  to evaluate the Gaussian field later,

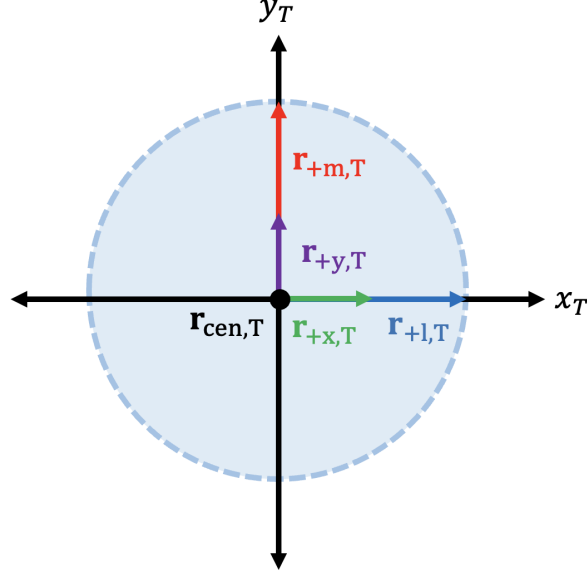
$$\mathbf{r}_{pixel,T} = \mathbf{O} \mathbf{r}_{pixel,D}, \quad (28)$$

483 .

484 The  $T$  subscript in equations 26-28 denotes that they are expressed in the transversal plane  
485 basis. Now that the ray data is expressed in the appropriate coordinate system, we can use them to  
486 compute the differential ray transfer matrix to the transversal plane.

### 487 3.3 Computing the differential ray transfer matrix

488 To compute the differential ray transfer matrix for a Gaussian beam we require the five ray coordi-  
489 nates and directions derived in the previous subsection ( $\mathbf{r}_{ray,T}$ , and  $\mathbf{k}_{ray,T}$ ), as well as the data for  
490 the same ray on the source plane ( $\mathbf{r}_{ray,S}$ , and  $\mathbf{k}_{ray,S}$ ). The differential ray transfer matrix is calcu-  
491 lated using the methods discussed in section 2, and is repeated here to show the full calculation.  
492 The matrix in equation 29 has elements described by linear approximations of a derivative of ray  
493 data. Each element of the matrix is given by the form  $a_{ray,B}$  where  $a$  denotes the data (position  
494  $x, y$  or direction cosine  $l, m$ ),  $ray$  denotes which of the 5 rays the data is from ( $+x, +y, +l, +m, cen$ ,  
495 given in table 1), and  $B$  which denotes the plane the data was taken from ( $S$  for source plane,  $T$  for  
496 transversal plane). An illustration of the ray position coordinates on the transversal plane is shown  
497 in figure 10. The differences in the positions of the differential rays with the position of the central  
498 ray gives the information required to compute the elements of equation 29.



**Fig 10** Example of the five rays position coordinates on the transversal plane after propagating through a rotationally symmetric optical system. The rotational symmetry means the rays stay in their plane of incidence, in this case, the  $+m$  and  $+y$  rays stay in the  $y - z$  plane, and the  $+l$  and  $+x$  rays stay in the  $x - z$  plane. Along the vertical  $y_T$  axis, the  $+m$  and  $+y$  ray positions are shown in red and purple, respectively. Along the horizontal  $x_T$  axis, the  $+l$  and  $+x$  rays are shown in blue and green, respectively. The central ray coordinate acts as the coordinate origin of the transversal plane, given by  $\mathbf{r}_{cen,T}$ . The light blue region that inscribes the position vectors is the area that is represented by the differential ray trace, and is effectively what we enforce the paraxial approximation over.

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \frac{x_{+x,T}-x_{cen,T}}{x_{+x,S}-x_{cen,S}} & \frac{x_{+y,T}-x_{cen,T}}{y_{+y,S}-y_{cen,S}} & \frac{x_{+l,T}-x_{cen,T}}{l_{+l,S}-l_{cen,S}} & \frac{x_{+m,T}-x_{cen,T}}{m_{+m,S}-m_{cen,S}} \\ \frac{y_{+x,T}-y_{cen,T}}{x_{+x,S}-x_{cen,S}} & \frac{y_{+y,T}-y_{cen,T}}{y_{+y,S}-y_{cen,S}} & \frac{y_{+l,T}-y_{cen,T}}{l_{+l,S}-l_{cen,S}} & \frac{y_{+m,T}-y_{cen,T}}{m_{+m,S}-m_{cen,S}} \\ \frac{l_{+x,T}-l_{cen,T}}{x_{+x,S}-x_{cen,S}} & \frac{l_{+y,T}-l_{cen,T}}{y_{+y,S}-y_{cen,S}} & \frac{l_{+l,T}-l_{cen,T}}{l_{+l,S}-l_{cen,S}} & \frac{l_{+m,T}-l_{cen,T}}{m_{+m,S}-m_{cen,S}} \\ \frac{m_{+x,T}-m_{cen,T}}{x_{+x,S}-x_{cen,S}} & \frac{m_{+y,T}-m_{cen,T}}{y_{+y,S}-y_{cen,S}} & \frac{m_{+l,T}-m_{cen,T}}{l_{+l,S}-l_{cen,S}} & \frac{m_{+m,T}-m_{cen,T}}{m_{+m,S}-m_{cen,S}} \end{bmatrix} \quad (29)$$

For convenience the total ray transfer matrix can be organized into a tensor composed of 2x2 sub-matrices shown on the left hand side of equation 29. These are exactly the non-orthogonal representations of the ray transfer matrix discussed in section 2. With the ray transfer matrix solved, we can finally compute the Gaussian field.



### 3.4 Computing the Gaussian field

The central ray emanates normal to the peak of the Gaussian, and as such locates where the peak is at every point in the optical system. Therefore, it serves as the coordinate origin for our Gaussian field evaluation on the transversal plane. With the position vectors effectively transformed in equations 26 and 28, we can define our radial coordinate  $\mathbf{r}_o$  as the simple difference of these two positions to center the coordinate system

$$\mathbf{r}_o = \mathbf{r}_{pix,T} - \mathbf{r}_{cen,T}. \quad (30)$$

Finally, we can call upon the equation for a propagated Gaussian beam (equation 8) to perform the field evaluation?

$$U(\mathbf{r}_o) = \frac{U_o}{\sqrt{\det|\mathbf{A} + \mathbf{B}\mathbf{Q}_1^{-1}|}} \exp\left[\frac{-ik}{2} \mathbf{r}_o^T \mathbf{Q}_2^{-1} \mathbf{r}_o + ik\Delta_{cen} + ik\Phi_{opd}\right] \quad (31)$$

Where  $\mathbf{Q}_2^{-1}$  is the propagated complex curvature matrix given by equation 7

$\mathbf{r}_o^T$  is the transpose of the coordinate from equation 30,  $\Delta_{cen}$  is the propagated distance of the central ray to the transverse plane, and  $\Phi_{opd}$  is the optical path difference experienced by the ray, which we get from the ray tracing engine used.

The procedure outlined above is repeated for each beamlet and pixel on the detector. From a computational perspective this procedure is somewhat daunting. Repeating this process means the computation complexity scales with  $N_{beamlets} \times N_{pixels}$ , which can quickly become prohibitive. Fortunately, the algorithm described is entirely vector and matrix operations so it is simple to take advantage of array broadcasting rules in Python to vectorize the computation, and parallel comput-

ing to accelerate it. In our implementation, the vector operations described above are broadcasted such that the field of a single beamlet at all pixels are computed simultaneously. This operation exists in a loop over the number of beamlets. For preliminary efforts in computational acceleration, see [Appendix A](#).

Note that there is no additional phase associated with a decenter or shift of the initial Gaussian beam in the final formula (Equation 31), unlike prior formulations of this problem.<sup>2,2</sup> This is because the field is computed on a series of tilted planes, which accounts for this phase factor. Note the subtlety that the propagation procedure does not depend on the elementary field of choice until the electric field is evaluated. The assumption of this propagation method is that the elementary field is paraxial about the area encompassed by the differential rays. In principle, as long as a field's propagation formula is known analytically via the general Collins' integral it can be propagated with this method. Thus, GBD is a *special case* of the method described. Because of the history of modeling resonators with ray transfer matrices, we know the ABCD propagation laws for Laguerre-Gaussian<sup>2</sup> and Hermite-Gaussian<sup>2</sup> beams, which form complete sets. This indicates that the beamlet decomposition method would be of higher accuracy when decomposing the field into modes of higher spatial order. The formula for flattened elliptical Gaussian beams is also known,<sup>2</sup> which are capable of mitigating the soft-edge effect imposed by the traditional beamlet decomposition.<sup>2</sup> This method also works with Worku and Gross's half- and quarter-truncated Gaussian beamlets.<sup>2</sup>

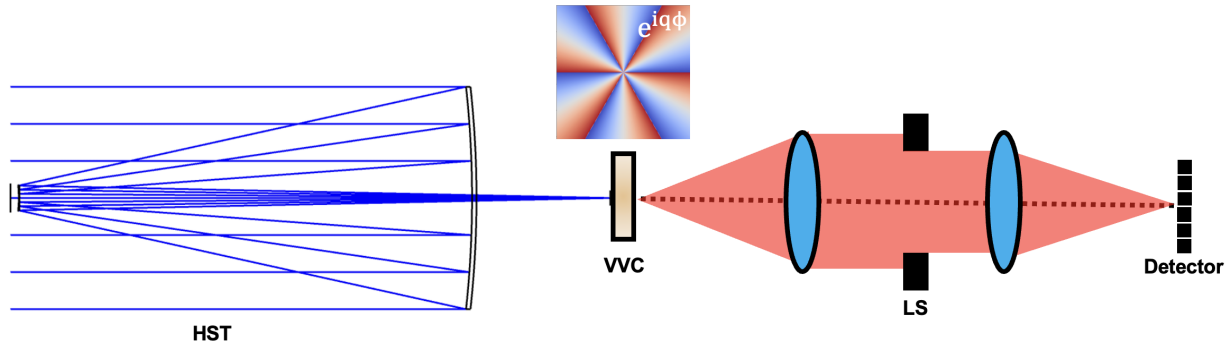
## 4 Results

To evaluate GBD as a viable physical optics propagation technique we must benchmark its performance v.s. traditional diffraction simulations for a given observatory **coupled to a vortex coro-**

nagraph. The fiducial observatory used in this study is a Ritchey-Chretien (RCs) objective based on the Hubble Space Telescope (HST) using an unobscured aperture. This model is constructed in Zemax OpticStudio, using the system prescription is given in Table 2 and is illustrated in figure 11.

Surface	RoC [m]	Conic Constant	Distance [m]	Semi-Aperture [m]
M1	-11.0400	-1.00230	-4.90607	1.20000
M2	-1.35800	-1.49686	6.40620	0.14056

**Table 2** Optical system prescription for the RC telescope based on the HST used in this investigation. All distances are given in meters, and the semi-aperture is equivalent to half of the diameter.



**Fig 11** Illustration of the hybrid propagation physics model used to produce the results in this section. The system prescription in table 2 is loaded into Zemax OpticStudio and shown on the left (labeled HST). The phase of the vector vortex coronagraph (VVC) is shown in the middle. The GBD PSF is computed at this plane and propagated through the coronagraph using HCIPy to arrive at the final image at the detector plane.

The PSF simulations conducted in section 4.2below are monochromatic simulations at  $1.65 \mu\text{m}$  on a detector with  $256 \times 256$  pixels over  $1 \times 1 \text{ mm}^2 \times 2 \text{ mm}$ . The coronagraphic simulations conducted in section 4.3 are were conducted at the same wavelength, but with  $1600 \times 1600$  pixels across  $8 \times 8 \text{ mm}^2 \times 512 \times 512$  pixels across  $10 \times 10 \text{ mm}$  to better sample the vortex mask and reach the desired contrast levels for both the Hybrid and Fraunhofer model.

#### 4.1 The Fiducial Coronagraph

Our goal in this study is to assess the feasibility of GBD to integrate ray models of observatories into physical optics models of coronagraphs accurately. To quantify this, we propagate the images produced by GBD and the analytical Airy function through a charge-26 vortex coronagraph (VC).<sup>?,?</sup> The optical VC is the general case of the vector vortex coronagraph<sup>?</sup> which has shown great promise for future missions to image Earthlike exoplanets.<sup>?</sup> These coronagraphs are excellent at rejecting low-order spatial modes, while transmitting high-spatial frequency content. We expect that traditional GBD will have some difficulty in accurately modeling high-spatial frequency content, and that it will manifest in the focal plane of this fiducial coronagraph if the error is limiting. If not, then we can conclude that GBD is suitable for high-contrast imaging simulation.

The complex amplitude of the charge- $q$  VC focal plane mask is given by

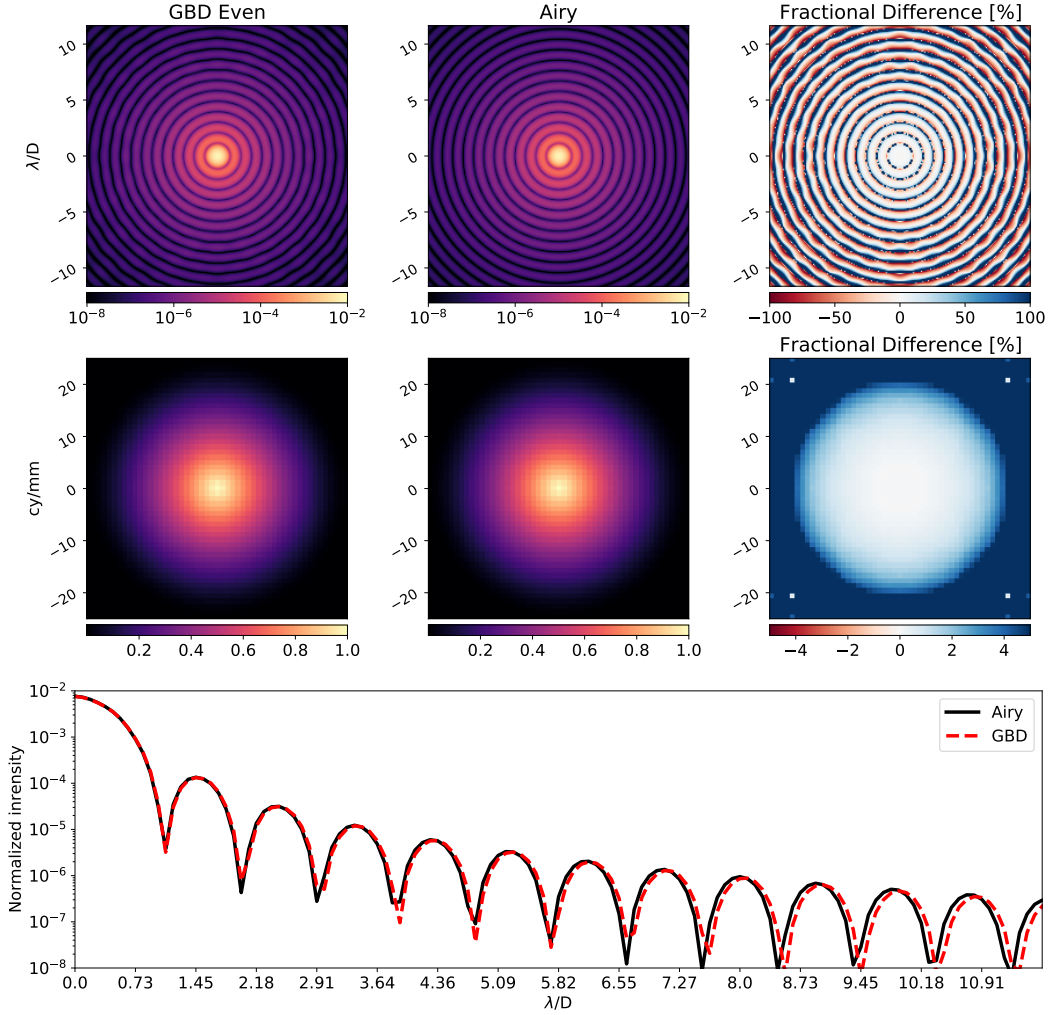
$$U(x, y) = \exp\left[iq \arctan\left(\frac{y}{x}\right)\right], \quad (32)$$

and the transmission is unity everywhere except for the center pixel, where it is 0. This is because there is a singularity in the phase ramp at this location, so it must be masked out. We chose the VC because of its ability to effectively reject on-axis starlight at a given wavelength. Should GBD introduce undesirable artifacts into the PSF, it should be visible in the coronagraphic focal plane. Modeling a VC accurately is challenging computationally, because the on-axis starlight is only completely rejected if the focal plane is infinitely sampled. The singularity at the center must also be sampled highly in order to accurately sample the rapid change in phase immediately around it without discretization errors. These require very large arrays ( $\geq 16k \times 16k$  arrays) for meaningful starlight rejection, which considerably slows the simulation. To overcome this computational

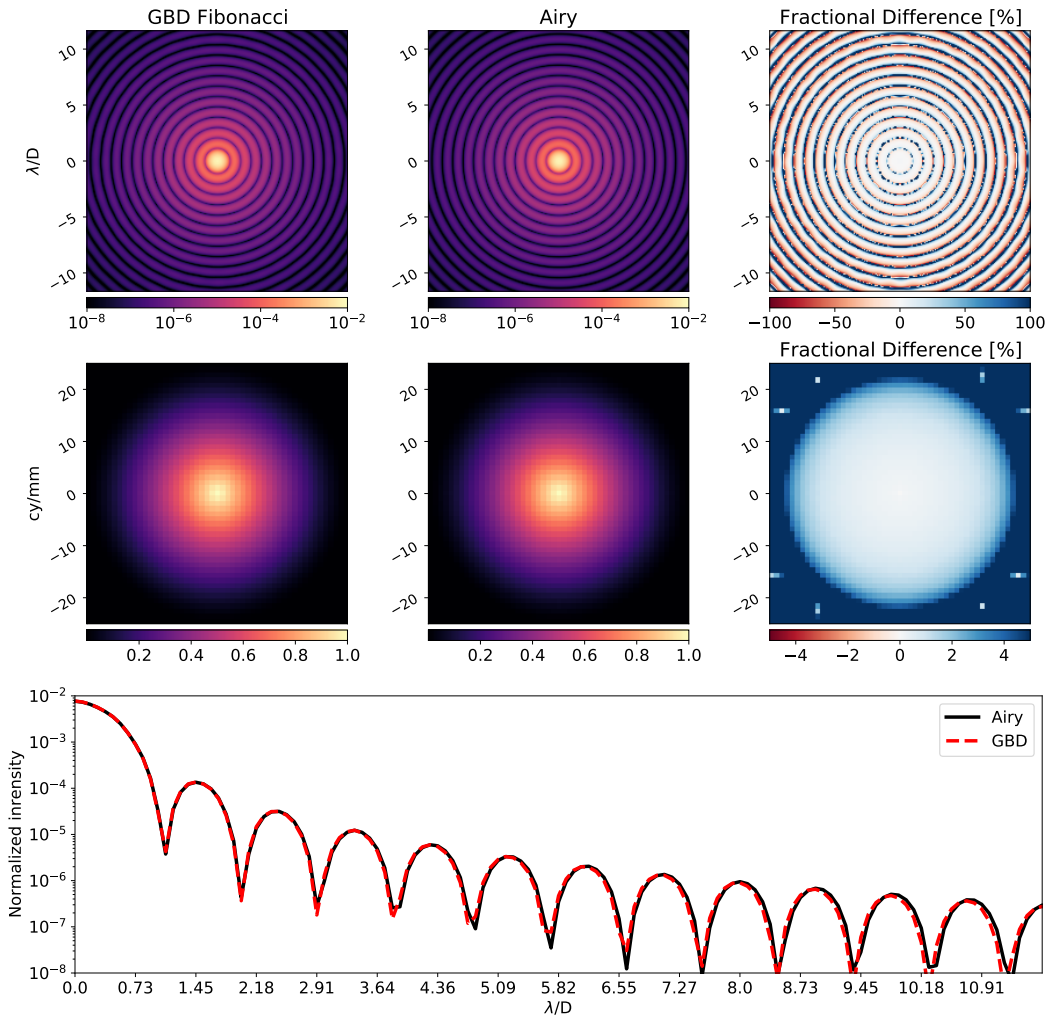
burden, a multi-step propagation algorithm can be used to sample the central singularity higher than the rest of the field. HCIPy<sup>7</sup> has this algorithm implemented in the `VortexCoronagraph` class, which accepts a user-specified wavefront and then outputs the wavefront before the lyot stop. We can define our GBD PSF as an HCIPy wavefront and propagate it through the coronagraph to complete our hybrid propagation model and analyze the image plane residuals. ~~we can supply with the PSF produced by GBD in order to complete our hybrid propagation model and analyze the influence on the resultant image plane.~~

## 4.2 The Observatory PSF

First we examine GBD's ability to construct the Airy pattern. The Airy pattern represents the "ideal" diffraction-limited observatory image for a circular aperture. The analytical solution for this image is known and available in POPPY, so we have a point of comparison that is not limited by numerical simulation errors. Shown in figures 12 and 13 below are the PSF simulations using the differential ray tracing methods detailed in Section 2 for the Even and Fibonacci sample schemes. We also plot a comparison of the Modulation Transfer Function (MTF) to better illustrate how the transfer of individual spatial frequencies is affected by GBD.

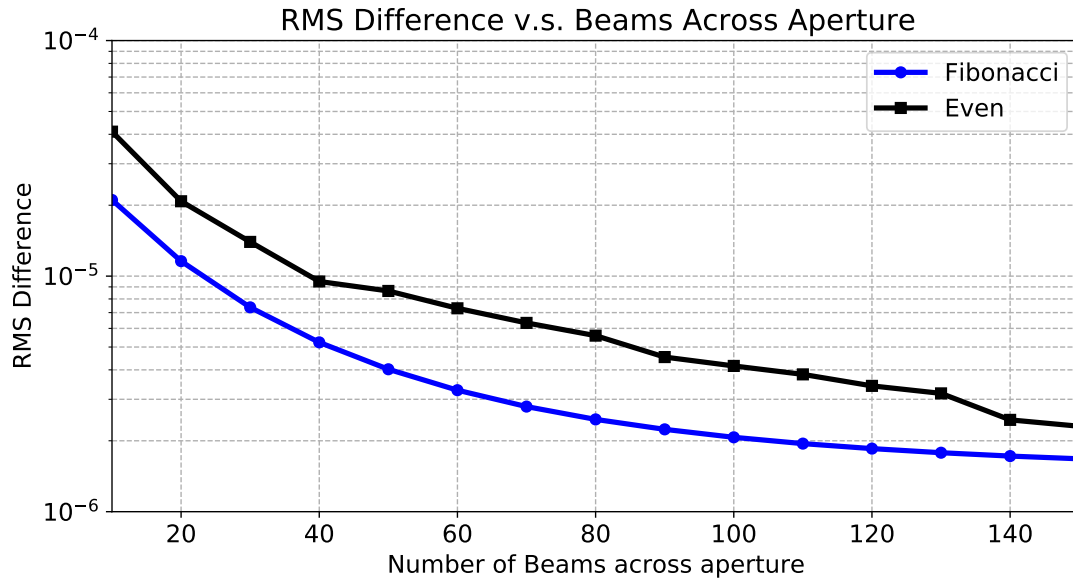


**Fig 12** Comparisons of the PSF (top) and MTF (middle) for GBD with even sampling (left) and analytical Airy pattern (middle). The PSFs are given in units of normalized intensity, such that the sum of the energy in the PSF is unity. The fractional difference is plotted on the right, and the azimuthally averaged radial profile is plotted on the bottom. The MTF is plotted out to the cutoff frequency of the HST of around 25 cy/mm. The radial oscillation in the fractional difference PSF is indicative of the artifacts introduced by GBD. In the MTF it is apparent that frequencies below 20 cy/mm are well-maintained, but the higher spatial frequencies increase in fractional difference. The effect on the PSF is revealed in the radial profile, where the PSF appears to spread out at larger angular separations. The RMS difference of the PSF data is  $2.3 \times 10^{-6}$ .



**Fig 13** Comparisons of the PSF (top) and MTF (middle) for GBD with Fibonacci sampling for the same number of beamlets as figure 12 (left) and analytical airy pattern (middle). The PSFs are given in units of normalized intensity, such that the sum of the energy in the PSF is unity. The fractional difference is plotted on the right, with the MTF plotted out to the cutoff frequency of the HST of around 25 cy/mm. The azimuthally averaged radial profile is plotted on the bottom. The fractional difference PSF is noticeably free of the ripples present in Figure 12, and the fractional difference MTF shows that frequencies out to 23 cy/mm are well-maintained. The spreading present in Figure 12 is noticeably absent in the radial profile. The RMS difference of the PSF data is  $1.6\text{e-}6$ , indicating that this distribution results in a more accurate simulation.

is how the entrance pupil was decomposed. In both the PSF and MTF dimensions it is clear that the Fibonacci sampling is the superior decomposition method for this aperture. For more undersampled cases there may be a tradeoff in accuracy. To understand this, in Figure 14 we examine how well the analytical Airy function is reconstructed for the two sample schemes as a function of the number of beamlets.



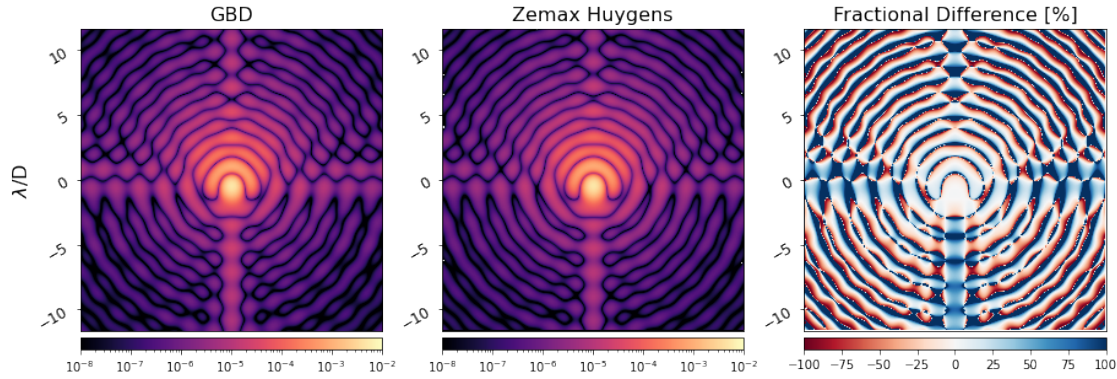
**Fig 14** The RMS difference between the sum-normalized GBD PSF and analytical Airy function for the Fibonacci (blue circles) and Even (black squares) sample schemes. For the circular aperture it is clear that the Fibonacci sample scheme is more accurate for every case given a circular aperture. However, the returns are diminishing as the number of beamlets increases.

The Fibonacci sampling clearly wins out in terms of performance given a fixed number of beamlets, which translates directly to computation time. This also means that using the Fibonacci sample scheme can yield a simulation of the same accuracy as the even sample scheme with fewer beamlets. By judicious choice in sample scheme, the computational complexity of GBD can be lessened or the simulation accuracy can be increased.

To further test GBD's capability to model a more realistic observatory PSF, we add obscurations in the entrance pupil that correspond to the secondary mirror and supporting spiders. We also tilt



the secondary mirror by 0.05 degrees to aberrate the beam. By doing so, we simultaneously test our algorithm's ability to capture aberrations from optical misalignment. The results of this simulation are shown in Figure 15



**Fig 15** Simulations of an aberrated PSF with secondary support structure in the entrance pupil of our HST model given in Table 2. The GBD PSF (left) and Zemax Huygens PSF (right) have very similar structure, with the dominant difference in structure being the features from the sharp-edge diffraction from the spiders (shown in fractional difference on the right). This simulation helps quantify the degree to which GBD's difficulty in modeling sharp-edge diffraction effects. The RMS difference of the PSF data is  $7.5\text{e-}6$ .

The data in figure 15 was simulated in a comparison between our proposed algorithm and Zemax's Huygens PSF analysis tool. In this simulation we vignette any Gaussian Beam that has any of its differential rays vignetted. Our results show extremely similar structure in the PSF from both the aberration induced by the secondary mirror misalignment and the structure from the HST spiders. The fractional difference reveals that the first couple "rings" of the PSFs are almost identical, indicating that the low-order aberrations were sufficiently simulated by GBD. There is a larger fractional difference in the dimmer structures that result from the sharp edges of the secondary obscuration and spiders, which is a known challenge for GBD simulations. This result can be further improved by careful implementation of alternative beamlet profiles, such as Worku and Gross's truncated beamlets,<sup>2</sup> or higher-order transverse electric field modes (e.g. Hermite, Laguerre-Gaussian). However, the result using fundamental Gaussian modes is encouraging, since

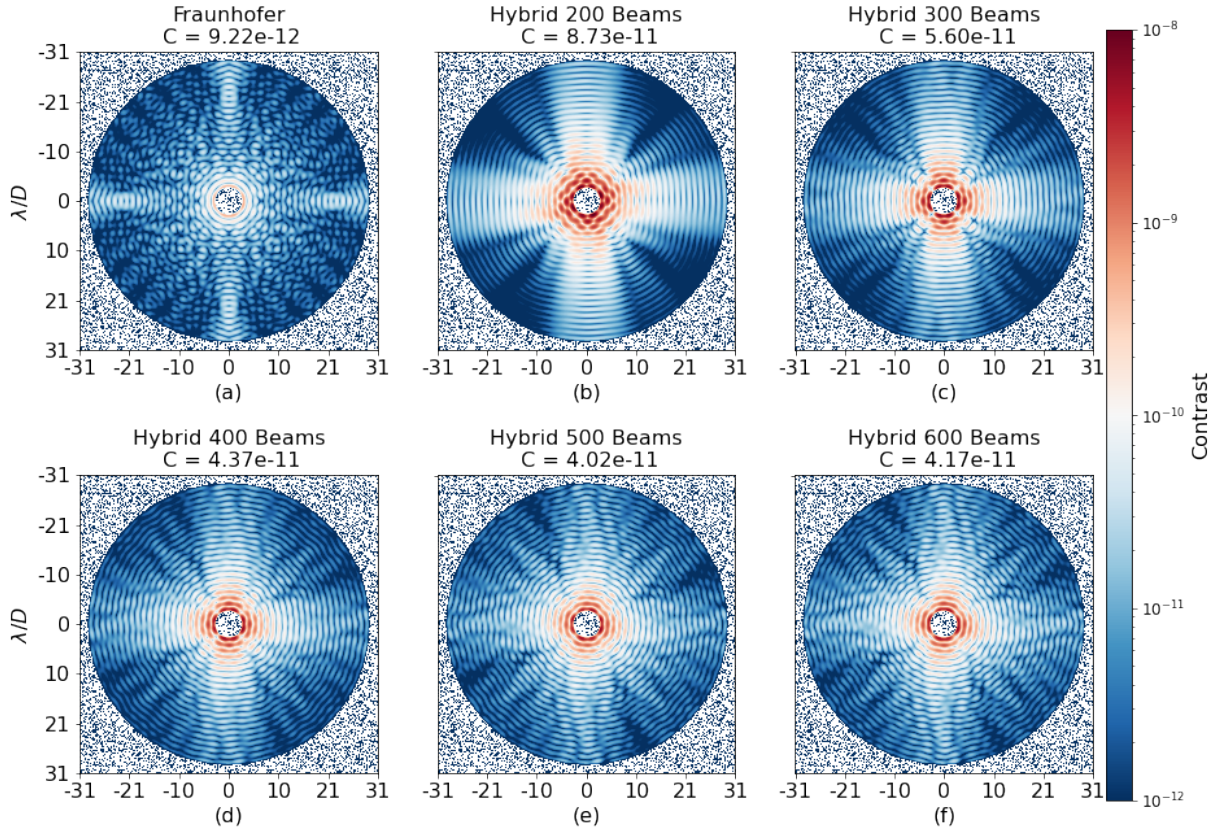
the RMS difference of the PSFs is within the same order of magnitude as the results shown in Figures 12 and 13. Now that our algorithm’s ability to perform PSF simulations has been verified, we can analyze the degree to which it introduces artifacts in high-contrast imaging simulations.

### 4.3 Coronagraphic Response

In traditional VCs, the on-axis field from an unvignetted circular aperture should be entirely rejected. The VC tends to be insensitive to aberrations of low-order, but high-spatial frequency signals pass through relatively unperturbed. We expect from the observatory PSF simulations that GBD does not trace high-spatial frequency information well due to the soft edges and amplitude ripples introduced by the beamlet decomposition. Any meaningful errors from this step should manifest as a high-spatial frequency error and pass through the coronagraph unperturbed. To formally assess GBD’s suitability for high-contrast imaging, we construct the PSF of the fiducial observatory with a circular aperture using GBD and compare it to a Fraunhofer model of the same system. Both PSFs are propagated with Fraunhofer diffraction through the vortex coronagraph and the coronagraphic focal plane is compared to assess the presence, if any, of high-spatial frequency content introduced by GBD.

Parameter	Value
Wavelength ( $\lambda$ )	1.65 $\mu\text{m}$
Number of Pixels ( $N_{pixels}$ )	1600512
Instantaneous FoV ( $\Delta x$ )	4.9519.5 $\mu\text{m}$

**Table 3** Simulation parameters for the result shown in Figure 16.



**Fig 16** Comparison of the Coronagraphic PSF from 3-30  $\lambda/D$  generated by a solely Fraunhofer diffraction model (a) and the proposed Hybrid propagation scheme for a varying number of beamlets across the 2.4m **HSThubble** aperture with an overlap factor of **24.7**. (b) was generated with 200 beamlets, (c) with 300 beamlets, (d) with 400 beamlets, **(e) with 500 beamlets, and (f) with 600 beamlets across the aperture**. These data were generated by producing a PSF with the given propagation scheme and then propagating it through HCIPy's VortexCoronagraph class with a topological charge of **26**. To reflect the residual starlight's astrophysical flux ratio "contrast", the PSF data are normalized to the maximum of an off-axis PSF at  $\approx 16\lambda/D \approx 25\lambda/D$  that propagates through the VortexCoronagraph. The mean contrast of the masked region (C) is shown on each image. The residuals **from the Hybrid propagation (b-f) are brighter than the equivalent Fraunhofer simulation, and minimize at the 500 beamlet case**. This is particularly apparent near the inner working angle, where we see a signal of  $\approx 1 \times 10^{-8}$  for the 200 beamlet case decrease to  $\approx 5 \times 10^{-9}$  in the 500 beamlet case. **from the Fibonacci distribution are clearly visible as high-frequency rings in b-d, which are at greater than earthlike exoplanet ( $10^{-10}$ ) contrast. However, it is encouraging that with more beamlets the prominent circles shown in (b) become around the same level as the background in (d).** This is indicative of GBD's ability to be further improved with **more computational resources or** a more judicious choice of beamlets.

inaccuracies in the propagation physics. The Hybrid propagation model generally minimizes in  
 average contrast until the 600 beamlet case (Figure 16f) where there is a slight increase.using the  
 Fibonacci distribution from Figure 13 leaves a prominent pair of rings near the core of the PSF,  
 and some distributed errors of lower energy at  $\lesssim 10^{-9}$ . The residuals from traditional Fraunhofer  
 diffraction leave some low-energy features near the core of the PSF, which can be improved with  
 greater sampling but the rest of the field is largely below  $10^{-11}$  contrast. The decreasing residual  
 energy with the number of beamlets used in the hybrid propagation scheme is encouraging, but  
 presently limited by the runtime to compute a higher-sampled simulation. The increase in av-  
 erage contrast in figure 16f suggests that we have determined the limit that traditional GBD can  
 be used to practically mimic a PSF given this sampling of the focal plane. The algorithm out-  
 lined in Section 3 is carried out as-written with some computational acceleration done by taking  
 advantage of Python's ability to vectorize matrix operations and other Python packages to acceler-  
 ate the exponential calculation numpy's array broadcasting operations, and numexpr's accelerated  
 exponential calculation, discussed in detail in Appendix A. We have not formally explored parallel  
 processing packages on Central Processing Units (CPU's) or Graphical Processing Units (GPU's)  
 to accelerate this computation, but expect that these could make higher-sampled simulations more  
 feasible to minimize the artifacts that remain in GBD PSF's for high-contrast imaging simulations.  
 Now that we have established an open-source platform for GBD, Worku's Modified GBD<sup>7</sup> could  
 be developed to minimize the number of beamlets required to accurately retrieve high-spatial fre-  
 quency information. Exploring higher order spatial modes and the astigmatic fundamental mode  
 of Gaussian Beams could also yield greater accuracy for the same or less computational complex-  
 ity. The result in Figure 16 using traditional GBD shows that we can reduce the residuals to below  
 $10^{-9}$  Contrast near the inner working angle, which is sufficient to detect some exo-earths.<sup>7</sup> It is

also worth noting that this result suggests that with sufficient sampling, GBD can presently be used to simulate the PSFs for systems with less stringent contrast floors. However, to get equivalent performance of traditional diffraction physics we have created a platform for the algorithm’s development in an open-source environment and will outline a road map for the most pressing optimizations that can be conducted by future investigations. ~~will explore alternative beamlet solutions and accelerated computing in future work.~~

## 5 Summary and Conclusions

Diffraction-limited optical systems require an accurate physical optics model to simulate the performance of the instrument. We formally describe an alternative implementation of the Gaussian Beamlet Decomposition physical optics propagation technique that is well-suited to PSF simulation. We illustrate the degree to which GBD is capable of accurately modeling the PSF of a Hubble-like astronomical observatory, and show that it introduces little traceable artifacts in the “hybrid” simulation of a vortex coronagraph. We have demonstrated a new means of integrated coronagraphic observatory modeling which reaches ~~near~~ state-of-the-art  ~~$10^{-9}$~~  contrast levels ~~at~~  $< 10^{-9}$  with an numerical average contrast of  $4 \times 10^{-11}$ . ~~We also outline a path forward for the ongoing development of this technique to make it suitable for simulations of Earth-like exoplanet signals at contrasts of  $10^{-10}$ .~~ To our knowledge this manuscript is the first of its kind to explicitly publish the full transfer matrix GBD method, and our hope in doing so is that future investigators can contribute to the development of this technique to further improve its accuracy.

## 5.1 Future Work

The simulations presented in this manuscript, while accurate, necessitated highly-sampled simulations that were computationally intensive. The longest simulation used as many as 360,000 Gaussian beamlets across 2,560,000 pixels, which created several phase arrays that were approximately 100TB in size. This simulation was done on an AMD Ryzen9 3950X processor and took  $\approx 2420$  hours to complete. For GBD to be a practical diffraction technique for high-contrast instrument sensitivity analysis it would be ideal to generate a field of the same accuracy in a shorter amount of time. Preliminary efforts in accelerated computing are discussed in Appendix A. Through careful optimization of our algorithm we were able to decrease the runtime considerably.

Modified GBD is already known to increase simulation accuracy for fewer beamlets,<sup>2</sup> and as such is the next natural step in the development of our GBD module. Worku and Gross’s work on truncated Gaussian beams showed high accuracy in reconstructing the field after 2D polygonal apertures, and “squeezed” the half-truncated beamlets in the azimuthal direction to increase the accuracy of a field after a circular aperture. Hexagonal apertures are of particular interest to astronomical telescopes because of their use in telescopes such as the W.M. Keck Observatory and James Webb Space Telescope. Understanding how the truncated beamlets are able to reconstruct the field after segmented apertures is another important step in developing GBD for high-contrast imaging simulations. It is also worth repeating the notion in Section 2 that nothing about the proposed algorithm requires the beamlets to be strictly Gaussian. Therefore, considering alternative beamlets to use in the field decomposition may increase the accuracy beyond what traditional GBD is capable of.



The Beamlet decomposition algorithm~~Our GBD module~~ would also benefit from more consideration into leveraging how parallelizeable it is. The contribution from each beamlet at each pixel is computed independently. With thoughtful consideration to the structure of our code and wide library of parallel processing packages available to us in Python on CPU's and GPU's,<sup>?,?</sup> the beamlet computation could be rapidly accelerated. Preliminary experiments on GPU's have shown a decrease in runtime by a factor of  $\approx 16$ . Several investigators have been exploring the Jax Python package for its ability to perform automatic differentiation and parallel computing in support of physical optics simulations.<sup>?,?,?</sup> Automatic differentiation could be extremely useful for future Beamlet Decomposition algorithms to improve the accuracy of the ABCD matrix computation. Parallelization and vectorization is the most natural path forward for our beamlet decomposition algorithm because of the independence of the beamlet operations. Jax makes this process simple with the *pmap* and *vmap* functions, while also allowing for just-in-time compilation for accelerated computing.

The use of a beamlet decomposition algorithm completely integrates a diffraction model with a ray model of an optical system, resulting in a more physically complete modeling pipeline. Consequently, other ray-based analyses can be integrated directly into the diffraction model. Polarization ray tracing<sup>?</sup> is a natural extension to beamlet decomposition simulations because it can trace the complex amplitude of individual beamlets for generally vectorial field propagation through optical systems. This capability was demonstrated by Worku and Gross<sup>?</sup> in the context of high-numerical aperture microscope objectives, but has the potential to be a powerful simulation tool for astronomical telescopes, including the next generation giant segmented mirror telescopes (Thirty Meter Telescope, Extremely Large Telescope, Giant Magellan Telescope)<sup>?</sup> and the Astro2020-recommended IROUV space observatory, which may be sensitive to polarization aberrations.

## 5.2 Open-Source Science and Engineering

This research was inspired by POPPY, a Python physical optics module originally developed to simulate the James Webb Space Telescope. Our goal is to expand the capabilities of POPPY by investigating new propagation physics. We are developing the GBD module in a self-contained package with interfaces to other popular diffraction codes (POPPY, HCIPy) to provide ray information to diffraction models. The code used to prototype the GBD method can be found in the  $Po\vec{k}\vec{E}$  repository on GitHub<sup>1,2</sup>. This repository is intended to be purely experimental as we develop the propagation physics for other high-contrast imaging packages that have substantive support.

## 6 Appendix A: Accelerated Computing

### 6.1 Accelerated Computing

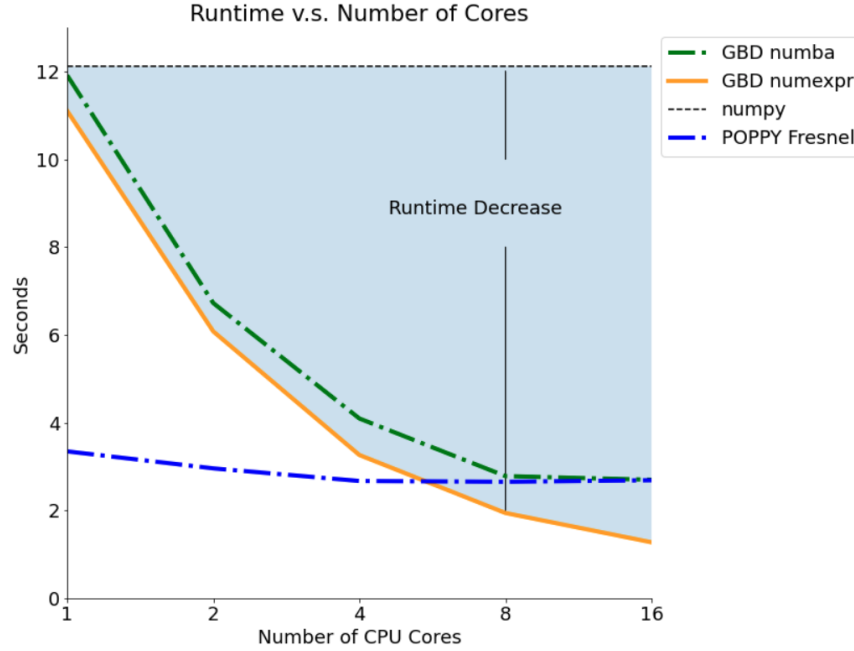
The independence of the Gaussian beamlet operations are uniquely suited to the exploration of multi-threaded computation to accelerate diffraction simulations. Accelerated computing is integral to diffraction modeling to enable rapid and precision simulation of small signals. The time to conduct traditional Fourier-based diffraction modeling is set by the complexity of the system. The sampling of each optical element and the number of total optical elements increase the complexity and number of Fast Fourier Transforms (FFTs) used, resulting in more computation time. GBD circumvents the FFT entirely by tracing rays to propagate through the optical system in a fraction of the time of the FFT. GBD's diffraction calculation at the plane of interest computes an exponential of a complex-valued array that scales with the number of beamlets and sampling of the image plane, resulting in longer computation times. Preliminary explorations into accelerated computing were conducted using the numexpr<sup>2</sup> and numba<sup>2</sup> Python packages that showed

---

<sup>1</sup><https://github.com/Jashcraf/poke>



favorable computation time decreases by multi-threading the operation on a CPU. Both packages operate by pre-compiling a given function into machine code that the program calls and breaking up the operation into chunks of arrays that a CPU core can handle efficiently. The key computational advantage of GBD is the ability to do diffraction calculations in parallel. Numba was the first package explored due to its ease of implementation. The package works by applying a decorator to a Python function that processes the large array of interest, and then specifying the number of central processing unit (CPU) cores for the process to use. The distribution of the information stored within the array is handled automatically by numba, and results in considerable runtime decreases for GBD. On a 16 core 2.4GHz CPU, runtime for a simulation of 1876 Gaussian beamlets through a coronagraph to simulate a 256x256 pixel focal plane was sped up by a factor of 5, which approached POPPY's Fresnel diffraction runtime. This experiment was repeated using the numexpr package which showed an even greater decrease in computation time, consistent with results for accelerating Fresnel diffraction in POPPY.<sup>?</sup> The comparison in runtime vs. number of CPU cores is shown in Fig. 6.



**Fig 17** Run time comparison for a 50x50 grid of Gaussian beamlets on a 256 x 256 detector grid v.s. number of CPU cores used. Numexpr considerably accelerates the exponential calculation given multiple processing cores, and is an encouraging path forward for further parallelization.

We anticipate even greater speedups on GPU's. In particular, the creation of the phase array from the ray data is the slowest operation and needs to be parallelized for accelerated diffraction modeling. Using the Cupy python package, adding support for GPU's was made quite simple given its compatibility with the numpy API. Preliminary experiments using our beamlet decomposition algorithm show a  $\approx 16\times$  decrease in computation time across all cases tested. We have added GPU support though Cupy in our poke package and plan to publish a formal study quantifying the degree to which GBD can benefit from accelerated computing in a future report. We intend to add GPU compatibility to our GBD module in future work.

## 7 Appendix B: Sample Poke code

The code used to conduct these simulations is located on the Poke repository on GitHub in the GBD\_Paper directory. The `test_transversal.py` script was used to conduct the simula-

tions in this manuscript. These simulations were done during a very early release (v0.1.0) of Poke without much optimization for runtime or a user-friendly design. This version has been moved to the legacy branch of the repository as  $Pok\vec{E}$  and the GBD module is actively developed.

## 8 Acknowledgments

This work benefited greatly from the insight of Dr. Norman Girma Worku, who pioneered the development of the transfer-matrix method of Gaussian Beamlet Decomposition. We thank Trenton Brendel, Weslin Pulin, and Brandon Dube for helpful discussion on the fundamental physics and design of the  $Pok\vec{E}$  API. Thanks to Kian Milani for the guidance with matplotlib. Thanks to Marcos Esparza, Kevin Derby, and Hyukmo Kang for help proof-reading this manuscript. Thanks to Sebastiaan Haffert for guidance on the use of HCIPy’s VortexCoronagraph model. This research made use of several open-source Python packages, including POPPY,<sup>?</sup> HCIPy,<sup>?</sup> prsym,<sup>?</sup> numpy,<sup>?</sup> matplotlib,<sup>?</sup> ipython,<sup>?</sup> astropy<sup>2,?,?.?</sup> scipy,<sup>?</sup> numba,<sup>?</sup> and numexpr.<sup>?</sup> This research made use of High Performance Computing (HPC) resources supported by the University of Arizona TRIF (Technology and Research Initiative Fund), UITS, and Research, Innovation, and Impact (RII) and maintained by the UArizona Research Technologies department. This work was supported by a NASA Space Technology Graduate Research Opportunity.

**Jaren N. Ashcraft** (He/They) is a Ph.D. Candidate at the University of Arizona’s Wyant College of Optical Sciences working with the UA Space Astrophysics Laboratory and Large Optics Fabrication and Testing Group. He received his B.S. degree in Optical Engineering from the University of Rochester in 2019, and M.S. degree in Optical Sciences from the University of Arizona in 2022. He is a recipient of a NASA Space Technology Graduate Research Opportunities award.

---

<sup>2</sup><http://www.astropy.org>

**Ewan S. Douglas (He/they)** is an Assistant Professor of Astronomy at the University of Arizona, and an Assistant Astronomer, Steward Observatory. Ewan completed a postdoctoral appointment at the Massachusetts Institute of Technology Department of Aeronautics and Astronautics, received master's and doctoral degrees in Astronomy from Boston University, and a bachelor's degree in physics from Tufts University. Dr. Douglas' specializes in space telescopes and instrumentation for high-contrast imaging of debris disks and extrasolar planets.

**Daewook Kim** is a faculty of optical sciences and astronomy at the University of Arizona. His research area covers precision optical engineering, optics fabrication, and freeform metrology including interferometry and deflectometry. He is the chair of the Optical Manufacturing and Testing (SPIE) and Optical Fabrication and Testing (OPTICA) conferences. He has served as an associate editor for the Optics Express journal. He is a senior member of OPTICA and SPIE Fellow.

**Dr. A J Eldorado Riggs** is an optical engineer at the Jet Propulsion Laboratory. His research focuses on the high-contrast imaging of exoplanets, in particular mask optimization and wavefront sensing and control for the Coronagraph Instrument on the Nancy Grace Roman Space Telescope. He received his B.S. in physics and mechanical engineering from Yale University in 2011 and his Ph.D. in mechanical and aerospace engineering from Princeton University in 2016.

## List of Figures

- 1 Modeling flow inspired by the Structural Thermal Optical Performance (STOP) modeling process for the Roman Coronagraph.<sup>?</sup> This diagram illustrates the different modeling regimes required to create a simulated image of an observatory. We aim to further integrate this modeling pipeline by creating an open-source Gaussian Beamlet Decomposition platform to unify the ray trace model of the observatory with the diffraction model of the coronagraph.
- 2 Illustration of the operating principle of GBD in one dimension. The aperture function for traditional imaging systems is shown in (a) as a top-hat function of uniform amplitude. The decomposition of this function into a discrete set of Gaussian beams is shown in (b), which shows nine evenly-spaced Gaussian profiles before their coherent summation, which is shown in (c). The coherent sum (black) of these nine beamlets shows that the beamlets are incapable of perfectly reconstructing the original aperture function (dashed red), specifically the sharp edge and uniform amplitude. More beamlets are needed for a more accurate reconstruction, which is shown in (d) for 199 beamlets. The amplitude ripple has virtually vanished, and the field near the aperture edges is almost entirely recovered.

3 Diagram demonstrating the hybrid propagation physics model. The observatory optics are best described by a ray-based propagation model, so GBD is used to compute the field at the image plane before the coronagraph focal plane mask (FPM). The field array is then passed to a paraxial diffraction model which propagates it past the FPM and through the remainder of the coronagraph. This propagation scheme permits the user to model the influence of the observatory optics non-paraxially, without losing accuracy after propagating past the focal plane mask.

4 Illustration of Poke's use as an interface to open-source ray-based physical optics. Poke only requires an interface between a ray tracing engine (orange, left) to generate and save a Rayfront object, which includes all ray data necessary for the GBD calculation. Currently, Poke supports sequential systems in CODE V and Zemax, but we are working on adding support for open-source packages that have ray tracers, like ray-optics.<sup>2</sup> The GBD field that Poke computes can then be sent to any open-source diffraction modeling package (yellow, right) to complete the hybrid propagation model.

5 Diagram illustrating differential ray tracing in the 2D case for a simple lens system. The central ray (black) is propagated along with two paraxial rays with a differential addition in the  $y$  direction (red) and another with a differential addition in the  $l$  direction (blue). To determine the ABCD matrix the ray data is computed on the transversal plane, which is normal to the central ray.

6 Illustration of the Even (left), Fibonacci (middle), and Polar (right) sample schemes for decomposing the field at the entrance pupil of the optical system with approximately the same number of beamlets in each figure. Quantifying the ramifications of these sample schemes is of paramount importance to accurate diffraction simulation.

7 Illustration of the concept for transfer-matrix Gaussian beamlet decomposition. The principle is to propagate the central ray that emanates normal to the peak of the Gaussian beam (red) in the entrance pupil to the transversal plane that intersects a given point on a detector (yellow pixel). The Gaussian field is evaluated at this intersection, and the process is repeated for each pixel and each ray. The coherent superposition of the Gaussian beams yields the field observed on the detector.

8 Illustration of relevant ray parameters necessary to propagate the ray data to the transversal plane, where we evaluate the Gaussian field that intersects with a detector pixel. A ray that defines a Gaussian beam is placed on the source plane (leftmost vertical dashed line) with some direction ( $\mathbf{k}_{ray,S}$ ) and position ( $\mathbf{r}_{ray,S}$ ) and propagated through the system. At the detector plane (rightmost vertical dashed line), the ray's direction and positions ( $\mathbf{k}_{ray,D}$  and  $\mathbf{r}_{ray,D}$  respectively) have been transformed by the optical system. To propagate the rays to the transversal plane (tilted dashed line) that intersects the pixel at  $\mathbf{r}_{pixel,D}$  where the field evaluation occurs, it must be moved a distance  $\Delta_{ray}$  along the ray path in detector space.

9 Caption

10 Example of the five rays position coordinates on the transversal plane after propa-  
gating through a rotationally symmetric optical system. The rotational symmetry  
means the rays stay in their plane of incidence, in this case, the  $+m$  and  $+y$  rays  
stay in the  $y - z$  plane, and the  $+l$  and  $+x$  rays stay in the  $x - z$  plane. Along the  
vertical  $y_T$  axis, the  $+m$  and  $+y$  ray positions are shown in red and purple, respec-  
tively. Along the horizontal  $x_T$  axis, the  $+l$  and  $+x$  rays are shown in blue and  
green, respectively. The central ray coordinate acts as the coordinate origin of the  
transversal plane, given by  $\mathbf{r}_{cen,T}$ . The light blue region that inscribes the position  
vectors is the area that is represented by the differential ray trace, and is effectively  
what we enforce the paraxial approximation over.

11 Illustration of the hybrid propagation physics model used to produce the results in  
this section. The system perscription in table 2 is loaded into Zemax OpticStudio  
and shown on the left (labeled HST). The phase of the vector vortex coronagraph  
(VVC) is shown in the middle. The GBD PSF is computed at this plane and prop-  
agated through the coronagraph using HCIPy to arrive at the final image at the  
detector plane.



12 Comparisons of the PSF (top) and MTF (middle) for GBD with even sampling (left) and analytical Airy pattern (middle). The PSFs are given in units of normalized intensity, such that the sum of the energy in the PSF is unity. The fractional difference is plotted on the right, and the azimuthally averaged radial profile is plotted on the bottom. The MTF is plotted out to the cutoff frequency of the HST of around 25 cy/mm. The radial oscillation in the fractional difference PSF is indicative of the artifacts introduced by GBD. In the MTF it is apparent that frequencies below 20 cy/mm are well-maintained, but the higher spatial frequencies increase in fractional difference. The effect on the PSF is revealed in the radial profile, where the PSF appears to spread out at larger angular separations. The RMS difference of the PSF data is  $2.3\text{e-}6$ .

13 Comparisons of the PSF (top) and MTF (middle) for GBD with Fibonacci sampling for the same number of beamlets as figure 12 (left) and analytical airy pattern (middle). The PSFs are given in units of normalized intensity, such that the sum of the energy in the PSF is unity. The fractional difference is plotted on the right, with the MTF plotted out to the cutoff frequency of the HST of around 25 cy/mm. The azimuthally averaged radial profile is plotted on the bottom. The fractional difference PSF is noticeably free of the ripples present in Figure 12, and the fractional difference MTF shows that frequencies out to 23 cy/mm are well-maintained. The spreading present in Figure 12 is noticeably absent in the radial profile. The RMS difference of the PSF data is  $1.6\text{e-}6$ , indicating that this distribution results in a more accurate simulation.

895 14 The RMS difference between the sum-normalized GBD PSF and analytical Airy  
896 function for the Fibonacci (blue circles) and Even (black squares) sample schemes.  
897 For the circular aperture it is clear that the Fibonacci sample scheme is more accu-  
898 rate for every case given a circular aperture. However, the returns are diminishing  
899 as the number of beamlets increases.

900 15 Simulations of an aberrated PSF with secondary support structure in the entrance  
901 pupil of our HST model given in Table 2. The GBD PSF (left) and Zemax Huygens  
902 PSF (right) have very similar structure, with the dominant difference in structure  
903 being the features from the sharp-edge diffraction from the spiders (shown in frac-  
904 tional difference on the right). This simulation helps quantify the degree to which  
905 GBD's difficulty in modeling sharp-edge diffraction effects. The RMS difference  
906 of the PSF data is  $7.5e-6$ .

16 Comparison of the Coronagraphic PSF from  $3-30 \lambda/D$  generated by a solely Fraunhofer diffraction model (a) and the proposed Hybrid propagation scheme for a varying number of beamlets across the 2.4m HSThubble aperture with an overlap factor of 24.7. (b) was generated with 200 beamlets, (c) with 300 beamlets, (d) with 400 beamlets, (e) with 500 beamlets, and (f) with 600 beamlets across the aperture. These data were generated by producing a PSF with the given propagation scheme and then propagating it through HCIPy's VortexCoronagraph class with a topological charge of 26. To reflect the residual starlight's astrophysical flux ratio "contrast", the PSF data are normalized to the maximum of an off-axis PSF at  $\approx 16\lambda/D \approx 25\lambda/D$  that propagates through the VortexCoronagraph. The mean contrast of the masked region (C) is shown on each image. The residuals from the Hybrid propagation (b-f) are brighter than the equivalent Fraunhofer simulation, and minimize at the 500 beamlet case. This is particularly apparent near the inner working angle, where we see a signal of  $\approx 1 \times 10^{-8}$  for the 200 beamlet case decrease to  $\approx 5 \times 10^{-9}$  in the 500 beamlet case. from the Fibonacci distribution are clearly visible as high-frequency rings in b-d, which are at greater than earthlike exoplanet ( $10^{-10}$ ) contrast. However, it is encouraging that with more beamlets the prominent circles shown in (b) become around the same level as the background in (d). This is indicative of GBD's ability to be further improved with more computational resources or a more judicious choice of beamlets.

17 Run time comparison for a 50x50 grid of Gaussian beamlets on a 256 x 256 de-  
tector grid v.s. number of CPU cores used. Numexpr considerably accelerates  
the exponential calculation given multiple processing cores, and is an encouraging  
path forward for further parallelization.

## List of Tables

- 1 Table of subscripts describing the five rays that define the propagation of a gen-  
erally astigmatic Gaussian beam. The central ray emanates from the peak of the  
beamlet and propagates through the optical system along with the closely spaced  
differential rays.
- 2 Optical system prescription for the RC telescope based on the HST used in this  
investigation. All distances are given in meters, and the semi-aperture is equivalent  
to half of the diameter.
- 3 Simulation parameters for the result shown in Figure 16.