PROPER: An Optical Propagation Library for IDL

John E. Krist
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA, USA 91109

Abstract

PROPER is a library of IDL (Interactive Data Language) routines for simulating optical propagation in the near and far fields using Fourier-based Fresnel and angular spectrum methods. The goal of PROPER is to provide a free, easy-to-use, and versatile means for simulating systems that require diffraction-based rather than geometrical analyses, such as spatial filtering systems with intermediate optics (e.g. a stellar coronagraph for extrasolar planet imaging). It has routines for creating complex apertures and obscurations, wavefront errors (defined by Zernikes, power spectra, or user-supplied maps), amplitude modulators (e.g. coronagraphic occulters), simple lenses, and deformable mirrors. The routines automatically select which propagator (near or far-field) is best at each surface based on analytically propagating a Gaussian pilot beam. The library includes a comprehensive manual and is distributed as IDL source code.

Keywords: Optical modeling, diffraction, coronagraphs

1. INTRODUCTION

The performance of some optical systems must be evaluated in terms of the diffraction and propagation of light between elements. In some cases, even using Fraunhoffer diffraction calculations (pupil plane to image plane or the inverse using a single Fourier transform) is not sufficient, as the effects of intermediate surfaces may be significant. For example, they are especially critical to the performance of coronagraphic telescopes designed for imaging planets around stars, such as the Terrestrial Planet Finder[1]. In these systems, the central star's diffraction pattern produced by the telescope must be suppressed using optical nulling or filtering techniques that typically involve a sequence of components. In addition, scattered light created by wavefront errors must be reduced using devices such as deformable mirrors. Because wavefront phase errors transform into amplitude errors as they propagate (and vice-versa) due to the Talbot effect, the effects of intermediate optics are important.

Modeling the propagation of light through an optical system is usually done in one of two ways: (1) calculating the path of individual beams though the optical components (ray tracing), or (2) calculating the changes in the electromagnetic field as it travels (physical optics propagation, or POP). Ray tracing is typically used to design the system and determine its basic optical properties, such as magnification, low-order aberrations, vignetting, etc., but it cannot predict the effects of diffraction. Conversely, POP is concerned with how the electromagnetic wavefront is diffracted and modified as it travels, but it does not (usually) determine things like aberration changes caused by a shifted component. Hybrid codes exist that use the two methods[2] – ray tracing to determine aberrations, beam sizes, and the like, and POP to compute the diffraction effects. Alternative propagation algorithms also exist that combine the two, such as Gaussian beam decomposition[3].

A number of POP codes exist. Many well-known commercial ray-tracing programs now include physical optics propagation calculations, including Code V[4] and Zemax. Their POP systems are in addition to the simple, far-field (Fraunhoffer) calculations they have always made. However, these packages are costly, have steep learning curves, and are not easy to integrate into exploratory modeling systems (e.g. wavefront control algorithm testing). Many POP programs have been developed by research institutions and companies for internal use and are typically not available to or designed for use by the public. Free, publicly-available codes are few. Among these are LightPipes, which is a simple set of individual C programs that can be chained together to propagate a wavefront through a system, and Arroyo[5], a C++ library with an emphasis on atmospheric propagation for adaptive optics modeling.

2. PROPER

PROPER is a library of optical propagation procedures and functions for IDL (Interactive Data Language), a commercial data analysis environment commonly used in the astronomy and physics communities. PROPER is intended for

Optical Modeling and Performance Predictions III, edited by Mark A. Kahan,
Proc. of SPIE Vol. 6675, 66750P, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.731179

exploring diffraction effects in optical systems within a flexible and easy-to-use framework. It is a set of wavefront propagation routines – it is not a ray tracing system – and thus is not suitable for detailed design work. An optical system is described by a series of PROPER library function and procedure calls within a user-written IDL routine (hereafter called the *prescription*). These calls may be interleaved with additional user-written code, taking advantage of the wide range of mathematical, array processing, file input/output, and graphical routines available in IDL, providing a versatile system for modeling. PROPER includes procedures to create apertures and obscurations (circular, elliptical, rectangular, polygonal, and hexagonal arrays) and apply aberrations (low-order Zernikes polynomials, error maps defined by power-spectrum-density profiles, deformable mirrors, and user-defined wavefront error maps). An interface to the FFTW library is also provided for higher-performance Fourier transform routines. The PROPER routines are provided as IDL source code, so users can see what is happening "behind the scenes", debug the procedures, and (hopefully) suggest ways to improve the code. PROPER and its 140 page comprehensive manual is available for free from www.openchannelsoftware.com/projects/PROPER.

3. PROPAGATION METHOD

PROPER uses Fourier-transform-based near field (near the beam waist) and far field propagators to compute the complex-valued, two-dimensional wavefront distribution at any distance along the optical axis (because PROPER lacks ray tracing abilities, the system must be unfolded into a linear sequence of surfaces). It implements the computational paradigm described by Lawrence[6] in which separate operators are used for far-to-near, near-to-near, and near-to-far field propagations implemented with the Fresnel and angular spectrum algorithms. By chaining these operators into a sequence, the wavefront can be propagated through an arbitrary number of surfaces.

In the far field, the wavefront phase is referenced relative to a spherical surface with its center of curvature at the beam waist. This allows large phase curvatures caused by defocus to be modeled without aliasing in the computational array (i.e. the minimum separation between fringes in the phase-wrapped representation of the wavefront is equal to or greater than two samples). The grid sampling of the wavefront in the far field changes with propagation distance. In the near field, where the wavefront curvature is small, the phase is referenced to a plane, and the sampling remains constant between near-to-near field propagations. The user selects the sampling by specifying the diameter of the entrance pupil relative to the computational grid.

A lens operator, representing a thin refractive or reflective surface with power, is used to impart a radially-quadratic phase change. This has the effect of altering the curvature of the wavefront, causing the beam to converge or diverge with additional propagation distance. Any real systems that might contain conic, aspheric, or other non-parabolic lenses or mirrors must be represented using these simple lenses. Future versions of the library may integrate ray tracing to support other lens types.

The wavefront properties (diameter, radius of curvature, distance from the waist) are determined at any location along the optical axis by analytically propagating a Gaussian pilot beam at the same time. These characteristics determine which propagator to use. A point along the optical axis is considered to be in the near field when it is within twice the Rayleigh distance, $z_R=\pi\omega_0^2/\lambda$, from the beam waist, where ω_0 is the radius at the beam waist.

4. PROPER FUNCTIONS AND PROCEDURES

4.1 Prescription Definition and Execution Routines

A PROPER prescription starts with `prop_begin`, which initializes the wavefront and ends with prop_end, which returns the final image, either the intensity or complex field. A prescription is executed using `prop_run`. A "state" system is also available that allows the user to skip repeated propagations through the front end of an optical system when only the back end changes (e.g. the pattern on a deformable mirror is changed).

The general prescription definition and execution routines are:

prop_begin	Define initial beam properties and create wavefront array
prop_define_entrance	Define entrance pupil and renormalize wavefront to unit intensity
prop_end	Terminate propagation sequence
prop_end_savestate	Terminate saving state information and clean up state files
prop_init_savestate	Initialize state saving system

prop_is_statesaved()	Check if state applicable to current run exists
prop_run	Execute a prescription
prop_state	Read in saved state if it exists, else save current state

4.2 Wavefront Phase and Amplitude Modifying Routines

The primary intended use of the PROPER package is simulating the sensing and control of wavefront errors. Thus, there are a number of routines that involve modifying the phase and amplitude of the wavefront (aperture and obscuration mask routines are listed in another category). Examples are shown in Section 6. The functions are:

prop_add_phase	Add a phase error map to the current wavefront
prop_divide	Divide the wavefront amplitude by a value or 2D array
prop_dm	Modify the wavefront using a deformable mirror
prop_errormap	Read in an error map from a file and apply it to the current wavefront
prop_lens	Alter wavefront curvature due to a lens or mirror
prop_multiply	Multiply the wavefront amplitude by a value or 2D array
prop_propagate	Propagate the wavefront a specified distance
prop_psd_errormap	Create an error map defined by a power spectral density profile
prop_zernikes	Add phase aberrations defined by Zernike polynomials

4.3 Query Functions

A number of functions are available for determining the current characteristics of the wavefront being propagated and the current propagation state. Note that the results from some of these routines are only valid at the focus of an unaberrated system:

prop_get_amplitude()	Return the amplitude portion of the current wavefront
prop_get_beamradius()	Get the current radius of the pilot beam
prop_get_distancetofocus()	Get the distance from the current location to the focus (beam waist)
prop_get_fratio()	Get the current focal ratio of the pilot beam
prop_get_gridsize()	Get the size of the wavefront array grid
prop_get_nyquistsampling()	Get the Nyquist sampling criterion for the current wavefront
prop_get_phase()	Return the phase portion of the current wavefront
prop_get_refradius()	Get the current reference surface radius
prop_get_sampling()	Get the sampling of the wavefront in meters
prop_get_sampling_arcsec()	Get the sampling of the wavefront in arcseconds
prop_get_sampling_radians()	Get the sampling of the wavefront in radians
prop_get_wavefront()	Return the complex-valued wavefront array
prop_get_wavelength()	Get the wavelength of the propagation in meters
prop_is_statesaved()	Check if state applicable to current run is saved

4.4 Shape Drawing, Aperture & Obscuration Pattern Routines

Routines are provided that return an image containing a filled shape or multiply the wavefront by a mask with a certain shape. The edges of the shapes are antialiased (the value of a pixel along the edge of a shape is proportional to the area of the pixel covered by that shape, ranging from 0.0 to 1.0). Examples are shown in Section 5. These functions are:

prop_circular_aperture	Multiply the wavefront by a circular aperture (dark outside)
prop_circular_obscuration	Multiply the wavefront by a circular obscuration (dark inside)
prop_ellipse	Return an image containing a filled ellipse
prop_elliptical_aperture	Multiply the wavefront by an elliptical aperture (dark outside)
prop_elliptical_obscuration	Multiply the wavefront by an elliptical obscuration (dark inside)
prop_hex_aperture	Return an image containing a multisegmented hexagonal aperture
prop_polygon	Return an image containing a filled polygon
prop_rectangle	Return an image containing a filled rectangle
prop_rectangular_aperture	Multiply the wavefront by a rectangular aperture (dark outside)

prop_rectangular_obscuration	Multiply the wavefront by a rectangular obscuration (dark inside)
prop_rounded_rectangle	Return an image containing a rounded rectangle aperture mask

4.5 Error Map Input & Output Routines

IDL has an wide array of file input/output routines, and the free IDL Astronomy User's Library (which is required for running PROPER) has a number of procedures for reading and writing images in commonly used scientific formats (e.g. FITS). The PROPER routines listed here are expressly intended for reading and writing wavefront error maps:

prop_errormap	Read an error map from a FITS file and apply it to the wavefront
prop_readmap	Read an error map from a FITS file and return it as an image
prop_writemap	Write out a phase or amplitude error map to a FITS file

4.6 Utility Routines

PROPER has a number of utility routines that do simple things to the wavefront array (resample or shift) or perform related functions:

prop_fit_zernikes	Fit Zernike polynomials to an error map
prop_magnify()	Resize an image using damped sinc interpolation
prop_noll_zernikes	Generate a table of Noll-ordered Zernike polynomials
prop_print_zernikes	Print a table of Noll-ordered Zernike polynomials
prop_radius()	Return array of distances of wavefront array elements from optical axis
prop_resamplemap()	Resample an error map using cubic convolution interpolation
prop_rotate()	Rotate and/or shift an image via interpolation
prop_shift_center()	Shift the center of an array to the lower left of the array
prop_use_fftw	Enables/disables use of FFTW fft routines

4.7 Detector Modeling Routines

The `prop_pixellate` routine will integrate the final wavefront intensity image onto square pixels of finite area and spacing. It is useful for modeling images observed with a detector.

4.8 Other Routines

The only routine in this category, `prop_8th_order_mask`, generates a band-limited occulter mask for simulating high performance stellar coronagraphs. Because its source code can be examined by the user, it provides a useful example of how to implement other occulters.

5. CREATING APERTURES AND MASKS

Apertures and obscurations create the diffraction pattern that is unavoidable in even a perfect optical system. PROPER has a number of functions to generate simple or complex aperture patterns and masks using ellipses, rectangles, polygons, or hexagonal arrays (Figure 1). The edges of the shapes are antialiased (the transmission value of a pixel is proportional to its area that is covered by the edge)

6. ADDING WAVEFRONT ABERRATIONS

Because a primary purpose of PROPER is to provide a means to explore wavefront sensing and control, the library contains a number of routines to generate realistic phase and amplitude aberrations (Figure 2). Low-spatial-frequency aberrations, such as figuring or alignment errors, are often described using Zernike polynomials. Circular Zernike polynomial aberrations may be added normalized to an unobscured, circular aperture (up to an arbitrary order) or a centrally-obscured circular aperture (up to Z_{22}). Mid- or high-spatial-frequency errors, such as those caused by polishing, can be added by generating an error map based on a parameterized power spectral density (PSD) specification. A PSD describes how much aberration exists at each spatial frequency. Such wavefront aberrations are especially significant to high-contrast astronomical imaging as they dominate the scattered light background after the diffraction pattern has been suppressed by a coronagraph or some other device. Along with Zernike and PSD-based aberrations, the user may add error maps either from a file or as an IDL array. All aberrations may be applied either as phase or amplitude errors (e.g. non-uniform reflective coatings).

To simulate wavefront control mechanisms, PROPER includes a model of a deformable mirror that has uniformly-spaced actuators on a square grid. It includes a measured actuator influence function to reproduce the deformation of the mirror surface caused by an actuator poke. The user may specify either the piston of each actuator or the desired surface height at each actuator, in which case the routine will solve for the required actuator piston including influence function effects.

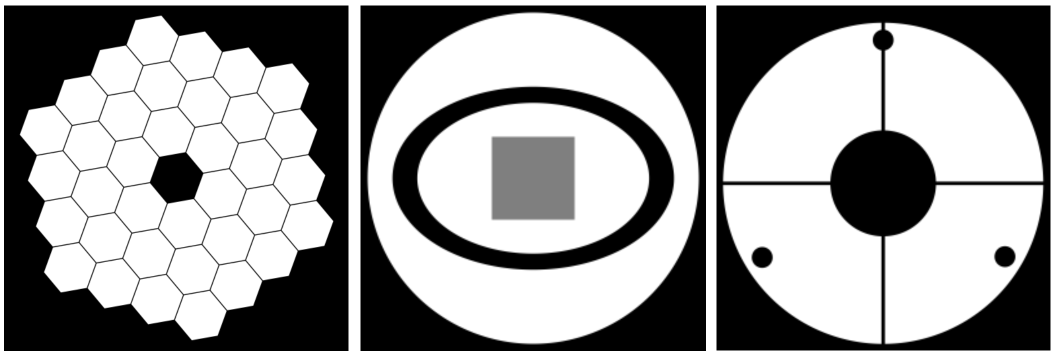

Figure 1. Examples of apertures created by PROPER library routines.

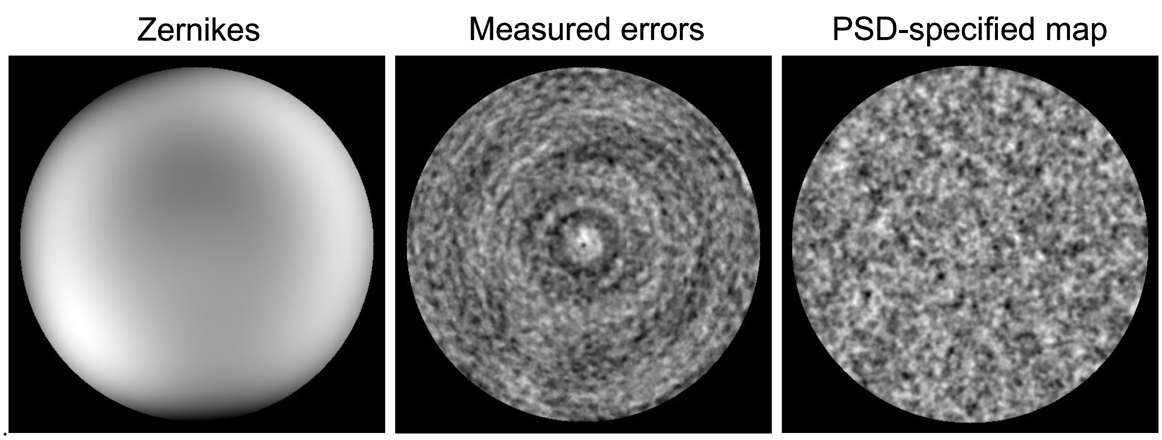

Figure 2. Examples of aberration maps that can be used or created by the PROPER library. (Left) Low-order Zernike polynomials. (Middle) A measured surface error map derived from interferometer measurements. (Right) A surface error map created by the `prop_psd_errormap` routine based on a PSD derived from the measured error map in the middle. While the PSD cannot capture the spatial correlation of the circular zones in the actual surface, it does reasonably simulate the level and spatial scale of the mid-frequency ripples.

7. EXAMPLES

7.1 A simple stellar coronagraph

A stellar coronagraph (Figure 3) suppresses the diffraction pattern of a star created by a telescope so that faint objects near it can be seen. The star is focused onto a occulting mask in the first image plane. Because the wavefront at the image plane can be equated to the Fourier transform of the preceding pupil, the occulter acts as a high-pass filter, blocking the low-spatial-frequency components of the pupil (unocculted sources are not affected). After the occulter, a new image of the pupil is formed which has the remaining light concentrated along the sharp edges of aperture and any obscurations. A mask (the Lyot stop) blocks these regions, effectively removing the high-spatial-frequency components of the original pupil (and the wings of the star's point spread function). The beam is then reimaged onto the detector.

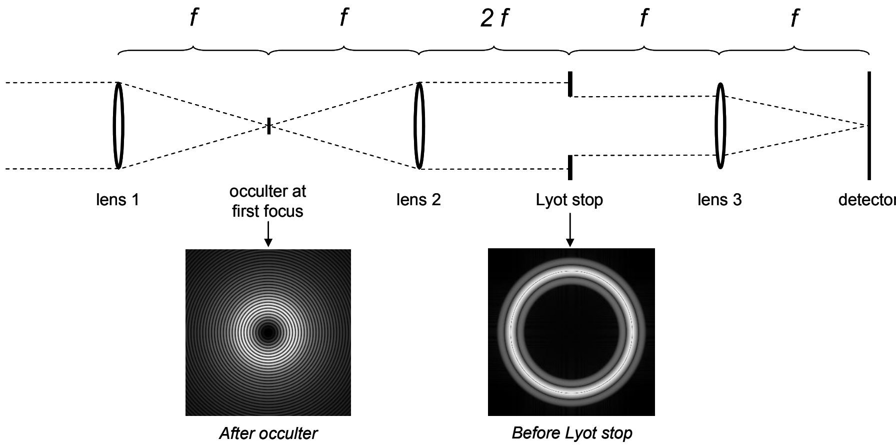

Figure 3. Schematic diagram of a simple stellar coronagraph. The computed image intensity after the occulter and before the Lyot stop are shown for an 8^{th}-order band limited circular occulter.

The IDL code below implements the coronagraph shown in Figure 3 using calls to PROPER routines (those with "prop_" prefixes). For simplicity, the focal lengths of the lenses are the same and no aberrations are included, though they can be easily added to any or all surfaces. An band-limited occulter is used, and the Lyot stop masks out the outer 50% of the pupil radius.

```
pro coronagraph, wavefront, wavelength, grid_size, sampling, PASSVALUE=optval

diam = 0.1            ;-- telescope diameter in meters
f_lens = 24 * diam    ;-- focal length of objective
beam_ratio = 0.4      ;-- pupil diameter / grid diameter

prop_begin, wavefront, diam, wavelength, grid_size, beam_ratio

prop_circular_aperture, wavefront, diam/2
   prop_define_entrance, wavefront
   prop_lens, wavefront, f_lens, 'objective'

prop_propagate, wavefront, f_lens, 'occulter'
   occrad = 4.0 ;-- occulter radius in wavelength/diam units
   prop_8th_order_mask, wavefront, occrad, /CIRCULAR

prop_propagate, wavefront, f_lens, 'pupil imaging lens'
   prop_lens, wavefront, f_lens, 'pupil imaging lens'

prop_propagate, wavefront, 2*f_lens, 'lyot stop'
   prop_circular_aperture, wavefront, 0.50, /NORM

prop_propagate, wavefront, f_lens, 'reimaging lens'
   prop_lens, wavefront, f_lens, 'reimaging lens'

prop_propagate, wavefront, f_lens, 'final focus'

prop_end, wavefront, sampling

return
end
```

7.2 Terrestrial Planet Finder wavefront sensing and control

The Terrestrial Planet Finder Coronagraph (TPF-C) is a proposed coronagraphic space telescope designed to image Earth-like planets around nearby stars. Such planets are 10^{10} times fainter than the stars they orbit. TPF's optimized coronagraph would suppress the diffraction pattern of the telescope to a level below that of the scattered light produced by the surface errors in the optics. The scattered light itself would be reduced by using deformable mirrors (DMs) to correct the wavefront errors. Two DMs in sequence have the ability to correct for both phase and amplitude wavefront errors[7].

As described in a previous article[8], a PROPER prescription was developed that described a simplified but realistic version of TPF-C in which each surface was given realistic phase and amplitude errors. A single DM was used to compensate for phase and amplitude errors in a limited region on one side of the occulted star, creating a dark half-hole in which imaging contrast was dramatically improved. The DM pattern to create the hole was determined through an iterative image-plane-intensity wavefront sensing and control algorithm using images simulated using PROPER. Since then, a modified version of that prescription was created that uses two DMs in sequence. With two DMs and using a variant of the same wavefront sensing and control algorithm, a hole that is dark on both sides was created with improved contrast (Figure 4). It is such experiments in wavefront control for which PROPER is well suited.

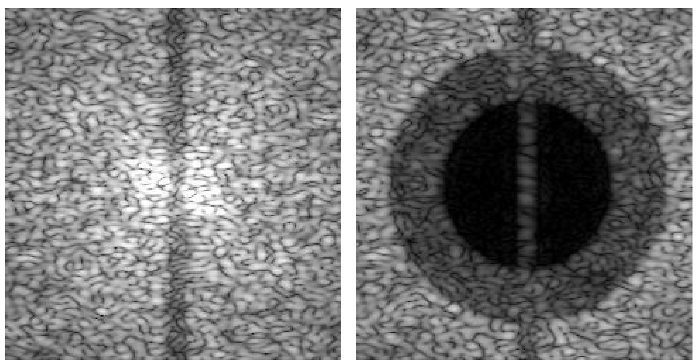

Figure 4. (Left) Simulated monochromatic image (λ=0.5 μm) of an occulted star observed with TPF-C. Each optic had both phase and amplitude errors defined by PSDs. The dark line down the center is caused by the linear occulter. The surrounding speckles are caused by phase and amplitude wavefront errors. (Right) TPF-C image after wavefront correction by the pair of DMs and an interative sensing and control algorithm. Phase errors were corrected over the larger circle, while both phase and amplitude errors were corrected in the smaller one.

7.3 James Webb Space Telescope NIRCam coronagraph

The James Webb Space Telescope is an infrared-optimized telescope currently scheduled for launch in 2013. It uses a hexagonally-segmented primary mirror 6.5 m in diameter. Its Near-Infrared Camera (NIRCam) will provide imaging over a wavelength range of λ=0.65 – 5.0 μm. NIRCam will include a simple Lyot coronagraph[9] to provide high contrast imaging capabilities, especially of extrasolar planets seen in thermal emission at λ=4.6 μm. PROPER was used to establish the final NIRCam occulter and Lyot stop designs and tolerances (Figure 5) and to determine the imaging performance of the coronagraph through end-to-end simulations (Figure 6) with realistic wavefront errors.

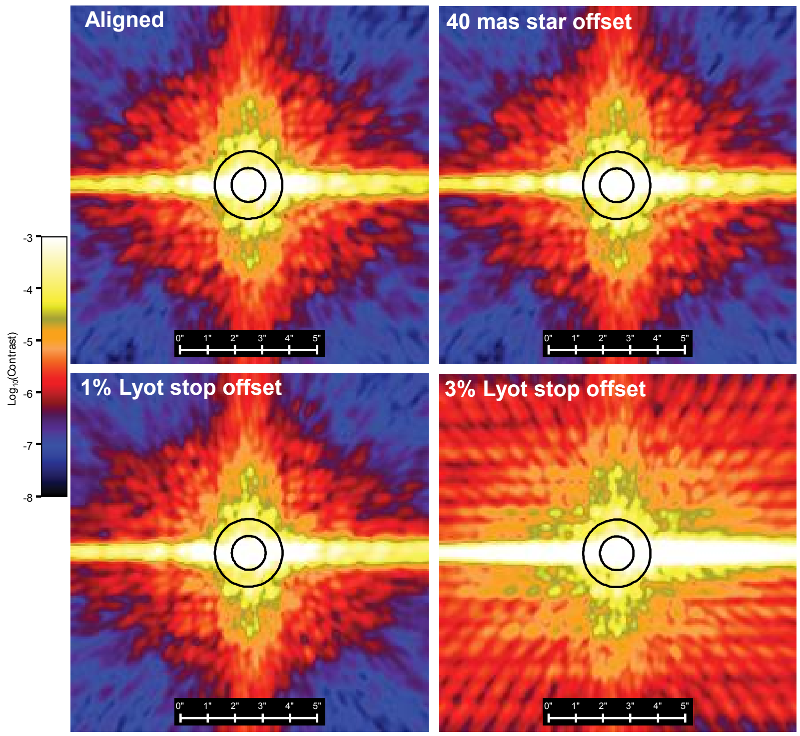

Figure 5. Simulated images of an occulted star in the NIRCam coronagraph through filter F460M (λ=4.6 µm). A linear occulter was used, and its transmission pattern was subsequently divided out, causing the white horizontal line. The circles at the center of each image correspond to radii of 4λ/D and 8λ/D. These simulations were used to determine the sensitivity of the coronagraph to misalignments of the occulter and Lyot stop.

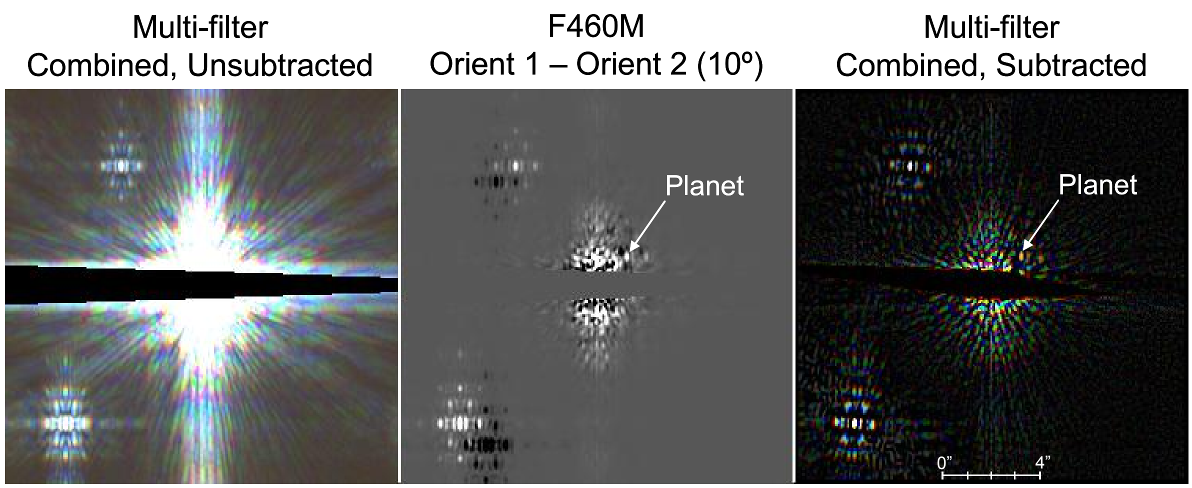

Figure 6. Simulated JWST NIRCam coronagraphic observations of a nearby (4 parsecs) star using a wedged-shaped occulter. Images were simulated through three filters (F360M, F410M, and F460M centered at λ=3.6 µm, 4.1 µm, and 4.6 µm) at two roll orientations of the telescope separated by 10º. A planet with twice the mass of Jupiter and two field stars are also included. (LEFT) Multicolor (F360M=blue, F410M=green, F460M=red) observations at the first orientation; (MIDDLE) Subtraction of F460M image taken at the second orientation from the first. (RIGHT) Multicolor subtracted images. The planet appears red because its flux peaks in the F460M filter. The subtractions at each filter were obtained using an iterative procedure that solves for the common rotating objects (field stars & planet) and static objects (residual light pattern from the occulted star). All of the images have the same static JWST wavefront aberrations plus random 5 nm RMS errors to represent exposure-to-exposure instabilities. In addition, the images at the second roll have an additional 20 nm RMS to represent orientation-induced changes to the wavefront.

8. LIMITATIONS

The PROPER library was primarily designed for be a easy to use and flexible way to study wavefront propagation, sensing, and control. It does have limitations that are primarily due to its inability to handle real-world systems that cannot be described by simple thin lenses. It cannot, for instance, predict the effects of multiple conic optics except in the purely paraxial case. It also cannot handle phase aberrations that are large enough to cause aliasing in the computational grid relative to the reference sphere (it is assumed that the wavefront does not deviate significantly from a spherical wavefront produced by a well-corrected system). Some optical design programs, such as Zemax, use ray tracing to determine the optimal reference surface.

9. ACKNOWLEDGEMENTS

PROPER was developed with the support of the NASA Terrestrial Planet Finder Program at the Jet Propulsion Laboratory. Zemax is a trademark of Zemax Development Corporation. Code V is a trademark of Optical Research Associates, Inc. Interactive Data Language is a product of ITT Visual Information Solutions.

REFERENCES

1. D. Coulter, "NASA's Terrestrial Planet Finder Missions," in *Proc. of the SPIE*, 5487, 1207 (2004).
2. G. N. Lawrence, "Integrating Geometrical and Physical Optics with the Lensgroup Operator Method,", in *Proc. of the SPIE*, 3780, 56 (1999).
3. B. D. Stone and T. J. Bruegge, "Practical considerations for simulating beam propagation: A comparison of three approaches," in *Proc. of the SPIE*, 4832, 359 (2002).
4. T. J. Bruegge, M. P. Rimmer, and J. D. Targove, "Software for Free-space Beam Propagation," in *Proc. of the SPIE*, 3780, 14 (1999).
5. M. C. Britton, "Arroyo," in *Proc. of the SPIE*, 5497, 290 (2004).
6. G. N. Lawrence, "Optical Modelling," in *Applied Optics and Optical Engineering*, 11, 125 (1992).
7. S. B. Shaklan and J. G. Green, "Reflectivity and optical surface height requirements in a broadband coronagraph," *Applied Optics*, 45, 5143 (2006).
8. J. Krist, J. Trauger, D. Moody, "Studying a simple TPF-C," in *Proc. of the SPIE*, 6265, 111 (2006).
9. J. Krist, et al., "Hunting Planets and Observing Disks with the JWST NIRCam Coronagraph," in *Proc. of the SPIE*, 6693 (2007).