The Astronomical Journal, 164:4 (32pp), 2022 July https://doi.org/10.3847/1538-3881/ac4017

Analytic Light Curves in Reflected Light: Phase Curves, Occultations, and Non-Lambertian Scattering for Spherical Planets and Moons

Rodrigo Luger[1,2,5], Eric Agol[2,3], Fran Bartolić[1,4], and Daniel Foreman-Mackey[1]
[1] Center for Computational Astrophysics, Flatiron Institute, New York, NY, USA; rluger@flatironinstitute.org
[2] Virtual Planetary Laboratory, University of Washington, Seattle, WA, USA
[3] Department of Astronomy, University of Washington, Seattle, WA, USA
[4] Centre for Exoplanet Science, University of St. Andrews, St. Andrews, UK
Received 2021 March 31; revised 2021 October 22; accepted 2021 October 29; published 2022 June 6

Abstract

We derive efficient, closed-form, differentiable, and numerically stable solutions for the flux measured from a spherical planet or moon seen in reflected light, either in or out of occultation. Our expressions apply to the computation of scattered light phase curves of exoplanets, secondary eclipse light) curves in the optical, or future measurements of planet–moon and planet–planet occultations, as well as to photometry of solar system bodies. We derive our solutions for Lambertian bodies illuminated by a point source, but extend them to model illumination sources of finite angular size and rough surfaces with phase-dependent scattering. Our algorithm is implemented in Python within the open-source starry mapping framework and is designed with efficient gradient-based inference in mind. The algorithm is ∼4–5 orders of magnitude faster than direct numerical evaluation methods and ∼10 orders of magnitude more precise. We show how the techniques developed here may one day lead to the construction of two-dimensional maps of terrestrial planet surfaces, potentially enabling the detection of continents and oceans on exoplanets in the habitable zone.[6]

Unified Astronomy Thesaurus concepts: Exoplanets (498); Light curves (918); Occultation (1148); Eclipses (442); Analytical mathematics (38)

1. Introduction

Despite recent advances in instrumentation and the dawn of thirty-meter-class telescopes and kilometer-wide interferomer arrays, extrasolar planets will remain unresolved point sources for decades to come. Nevertheless, modulations in the light received from these distant bodies due to their rotation, changing illumination, and eclipses by their host stars or other bodies in the system can be harnessed to reconstruct two-dimensional views of their surfaces. In particular, next-generation space-based telescopes such as the Large UV/Optical/IR Surveyor (LUVOIR) may enable us to measure variations in the reflected light signature of terrestrial planets in the habitable zone, which can be used to map their surfaces and indirectly infer the presence of clouds, continents, oceans, and perhaps even life.

There is an extensive literature on techniques for mapping exoplanet surfaces based on their phase curves (e.g., Russell 1906; Lacis & Fix 1972; Knutson et al. 2007; Cowan & Agol 2008; Oakley & Cash 2009; Berdyugina & Kuhn 2017; Heng et al. 2021; Luger et al. 2021b, 2021c) and occultation light curves (e.g., Williams et al. 2006; Rauscher et al. 2007, 2018; de Wit et al. 2012; Majeau et al. 2012), both in thermal and reflected (scattered) light. In particular, much attention has been given to techniques for mapping Earth-like planets from visible-light reflected phase curves (e.g., Ford et al. 2001; Kawahara & Fujii 2010, 2011; Fujii & Kawahara 2012; Aizawa et al. 2020; Kawahara 2020). Unlike thermal phase curves, which primarily encode (often degenerate) information about longitudinal surface brightness variations (Russell 1906), reflected light curves often contain information about the full two-dimensional surface albedo distribution (e.g., Kawahara & Fujii 2010).

Occultation light curves in reflected light can encode even more information about the surface. Thus far, these have been studied primarily within our solar system. Mutual occultations among the Galilean moons of Jupiter have been extensively studied to infer surface properties of the moons and to refine their ephemerides (e.g., Arlot et al. 1974, 2014; Aksnes et al. 1984; de Kleer et al. 2017; Saquet et al. 2018; Morgado et al. 2019; Bartolić et al. 2022). Farther out in the solar system, mutual occultations of Pluto and Charon in the late 1980s were used to confirm Charon's existence (Stern 1992), establish the sizes and orbital parameters of the two bodies (Tholen & Buie 1990), and infer their surface properties (Marcialis 1990). In particular, Dunbar & Tedesco (1986) developed an efficient analytic algorithm to model Pluto–Charon occultation light curves in reflected light assuming uniform surfaces and used it to infer the two body's average geometrical albedos. Later, Buie et al. (1992) used a maximum entropy approach to reconstruct two-dimensional maps of the two bodies and Reinsch et al. (1994) analyzed the complete mutual occultation data set to infer longitudinal maps of Pluto's albedo.

Many studies have analyzed real Earth reflected light curves to infer surface properties of our planet as an exercise in preparation for the mapping of exoplanets. Cowan et al. (2009, 2011) analyzed visible-light disk-integrated light curves of the Earth taken by the Deep Impact spacecraft to produce longitudinal maps of the surface, harnessing multiband observations to disentangle static surface brightness features from temporally variable clouds. A transit of Earth by the

[5] Flatiron Fellow.
[6] https://github.com/rodluger/starrynight

Moon was observed as well (Livengood et al. 2011), although this data has yet to be exploited for mapping purposes. More recently, Jiang et al. (2018) and Fan et al. (2019) used data from the L1-stationed DSCOVR satellite to infer surface and cloud properties of the Earth, and Luger et al. (2019b) analyzed background scattered light in TESS photometry to reconstruct a cloud map of the Earth.

There have also been many developments on the open-source software front. These include ReflectDirect, a reflected light-curve analysis suite (Schwartz et al. 2016); samurai, a tool for rotational unmixing of reflected light curves (Lustig-Yaeger et al. 2018); spiderman, an efficient discretization scheme on the sphere that enables fast computation of exoplanet phase curves and occultation light curves (Louden & Kreidberg 2018); exocartographer, a Bayesian framework for doing inference on exoplanet phase curves based on a HEALPix (Górski et al. 2005) discretization scheme (Farr et al. 2018); the sot package for spin–orbit tomography of exo-Earths (Kawahara 2020; Kawahara & Masuda 2020); and neural_exocartography, a tool for mapping exoplanet surfaces with neural learned regularization (Asensio Ramos & Pallé 2021). Of particular relevance to the present work, Haggard & Cowan (2018) presented EARL (Exoplanet Analytic Reflected Lightcurves), a Mathematica code for computing analytic, closed-form solutions for the phase curve of a Lambert sphere, i.e., one that scatters light isotropically, in the case that the surface albedo distribution is characterized by either a sum of delta functions or a sum of spherical harmonics. And finally, Luger et al. (2019a) introduced starry, a light-curve-modeling package that computes thermal phase curves and occultation light curves, as well as their derivatives, analytically from a spherical harmonic expansion of the surface brightness.

The present paper is an extension to the starry algorithm, adapting it to model phase curves and occultation light curves in reflected light. The expressions we derive are analytic: they may all be expressed in closed form in terms of algebraic operations involving trigonometric functions and (at times) elliptic integrals. We derive numerically stable recursion relations for the efficient evaluation of all expressions and code them within an autodifferentiation framework to enable the computation of accurate derivatives for use in gradient-based inference and optimization schemes. Our code is fully open source, comprehensively unit tested, and supplemented with an extensive API documentation and suite of tutorials. As in all papers in the starry series, in the caption of each of the figures we provide links (inside parenthesis) to the exact Python scripts that generated them. Next to many of the equations we also provide links (as footnote) to Jupyter notebooks containing detailed derivations and/or validations.[7]

2. Overview

Our goal in this paper is to derive analytic expressions for the flux received by a distant observer from a sphere of non-uniform albedo illuminated by a monochromatic source that may or may not be occulted by a (possibly different) spherical body. This applies, for example, to the case of planetary phase curves, secondary eclipse (occultation) light curves, and moon–moon, planet–moon, and planet–planet occultations (in the solar system or not; e.g., Cabrera & Schneider 2007; Luger et al. 2019b), all seen in reflected light. We derive all expressions in the limit that the reflecting body is Lambertian, i.e., it scatters light isotropically, but we relax this assumption in later sections. We model the general case of an intensity that varies across the surface of the body according to a spatially dependent albedo A. Throughout this paper, we will take A to mean the *spherical* albedo, the fraction of power incident on a body at a given wavelength that is scattered back out to space (in all directions). Note that the spherical albedo is closely related to the Bond albedo: the Bond albedo is the stellar flux-weighted integral of $A(\lambda)$ over all wavelengths λ (see, e.g., Seager 2010).

As in Luger et al. (2019a), we compute fluxes by first expanding the surface in terms of spherical harmonics. While in Luger et al. (2019a) we expanded the emissivity of the surface, here we instead expand the spherical albedo A. Specifically, if $\boldsymbol{y}$ is the vector of spherical harmonic coefficients describing the albedo anywhere on the surface and $\tilde{\boldsymbol{y}}$ is the spherical harmonics basis (Equation (A1)), the albedo A at a point (x, y) on the sky-projected disk of the body is given by the dot product

$$A(x, y) = \tilde{\boldsymbol{y}}^{\top}(x, y)\,\boldsymbol{y}. \tag{1}$$

The flux measured from this body is proportional to the surface integral over the projected disk of the albedo A times the illumination profile $\mathcal{I}$ of the surface, given by Lambert's law as

$$\mathcal{I}(\vartheta_{\rm i}) = \mathcal{I}_0 \max(0, \cos\vartheta_{\rm i}), \tag{2}$$

where $\vartheta_{\rm i}$ is the angle between the incident radiation and the surface normal, and $\mathcal{I}_0$ is the peak illumination. We show in Appendix A.2 that in the case of, say, a planet illuminated by its host star,

$$\mathcal{I}_0 = \frac{f_s}{\pi r_{\rm s}^2} \tag{3}$$

where $r_{\rm s}$ is the distance between the planet and the star (in units of the planet's radius) and f_s is the stellar flux measured at the observer (in arbitrary units). Following the convention in Luger et al. (2019a), we assume throughout this paper that $f_s = 1$, so all fluxes are defined as a fraction of the flux of the illumination source at the observer.

The piecewise nature of the illumination function at the day/night terminator makes the problem of computing the visible flux particularly difficult; most studies to date have tackled the problem numerically, either via Monte Carlo integration (e.g., Ford et al. 2001) or by discretizing the surface and computing the relevant integrals by summing over the visible pixels (e.g., Kawahara & Fujii 2010; Fujii & Kawahara 2012). Recently, Haggard & Cowan (2018) developed an analytic framework for computing light curves of unocculted bodies illuminated by a point source. In this paper, we rederive their solution under the starry framework and extend it, for the first time, to the case where the body is occulted by another spherical body, which may or may not be the illumination source. We also extend the solution to the case of an extended illumination source and to non-Lambertian scattering.

In Luger et al. (2019a), we reduced the problem of computing the flux from an occulted body in thermal (emitted) light to a series of efficient, analytical operations involving trigonometric functions of the position and size of the occultor and certain complete elliptic integrals. In the case of reflected

[7] https://github.com/rodluger/starrynight/blob/5f79

Table 1
List of Common Symbols Used in this Paper

Symbol	Description	References
	Frames of reference	
$\mathcal{F}_0$	frame in which surface map is specified	Appendix A.1
$\mathcal{F}$	observer (sky) frame	Appendix A.1
$\mathcal{F}'$	integration frame (occultor present)	Appendix C
$\mathcal{F}''$	integration frame (no occultor)	Appendix B
	Lines and surfaces	
S	region of integration enclosed by $\mathcal{PQT}$	Appendix C
$\mathcal{P}$	integration path along occultor limb	Appendix C
$\mathcal{Q}$	integration path along body limb	Appendix C
$\mathcal{T}$	integration path along terminator	Appendix C
	Special functions	
arctan 2	quadrant-aware arctangent	(A10)
F	incomplete elliptic integral of the first kind	(C47)
${}_2F_1$	Gauss hypergeometric function	Appendix C.6.4
E	incomplete elliptic integral of the second kind	(C48)
Γ	gamma function	Appendix B
Π	incomplete elliptic integral of the third kind	(C69)
	Operators and symbols	
V	Vieta summation operator	(C40)
Δ	pairwise difference operator	(C32)
$\int_\phi$	vectorized integral	(C31)

Table 2
List of Common Scalar Quantities Used in this Paper

Symbol	Description	References
	Integers	
l	spherical harmonic degree	(A2)
m	spherical harmonic order	(A2)
n	vector index	⋯
μ	spherical harmonic index, $\mu = l - m$	(A5)
ν	spherical harmonic index, $\nu = l + m$	(A5)
	Coordinates	
x	Cartesian x coordinate on the plane of the sky	⋯
y	Cartesian y coordinate on the plane of the sky	⋯
z	Cartesian z coordinate, $z = \sqrt{1 - x^2 - y^2}$	(A6)
	Geometrical parameters	
b	semiminor axis of terminator ellipse	(A9)
b_c	complement of b, $b_c \equiv \sqrt{1 - b^2}$	Appendix A.2
b_o	occultor impact parameter, $b_o = \sqrt{x_o^2 + y_o^2}$	Appendix A.1
k^2	elliptic parameter	(C46)
I	bodyinclination	Appendix A.1
n	elliptic characteristic	(C70)
r_o	occultor radius	Appendix A.1
r_s	distance to illumination source, $r_s = \sqrt{x_s^2 + y_s^2 + z_s^2}$	Appendix A.2
x_o	occultor x position	Appendix A.1
x_s	illumination source x position	Appendix A.2
y_o	occultor y position	Appendix A.1
y_s	illumination source y position	Appendix A.2
z_s	illumination source z position	Appendix A.2
Λ	body obliquity	Appendix A.1
θ	angle of rotation of the terminator ellipse	(A19), (A21)
Θ	body rotational phasse	Appendix A.1
	Intensities and fluxes	
a	orbital semimajor axis	Section 4.1
A	albedo (spherical)	(1)
f	reflected flux during an occultation	Appendix C
f_0	reflected flux outside of an occultation	(B6)
$\hat{f}_0$	complement of reflected flux outside of an occultation	(B7)
f_1–f_{14}	case-dependent reflected flux during occultation	Appendix C
f_I	intensity-weighted flux during an occultation	(C5)
f_T	thermal flux during an occultation	(A3)
f_{T0}	thermal flux outside of an occultation	(A7)
f_S	reflected flux over integration region S	(C12)
I	polynomial intensity at a point on the surface	(A12)
$\mathcal{I}$	true intensity at a point on the surface	(A11)
R_p	planet radius	Section 4.1
$R_\star$	stellar radius	Section 4.1
ϑ_i	polar angle of incidence	Figure 9
ϑ_r	polar angle of reflection	Figure 9
σ	Oren–Nayar surface roughness coefficient	Section 4.2
τ	Angular extent of terminator past $\pi/2$	(4)
ϕ_i	azimuthal angle of incidence	Figure 9
ϕ_r	azimuthal angle of reflection	Figure 9

light, however, the change in the limits of integration due to the unilluminated nightside breaks many of the symmetries that simplified the flux calculation. In particular, the limits of integration now depend on the solution to a quartic equation specifying the points of intersection between the occultor and the day/night terminator, and the solution to those integrals is now a function of *incomplete* elliptic integrals. The procedure for computing the flux is therefore significantly more complex. We therefore defer all calculations to the Appendix, and devote the body of the paper to validating and demonstrating applications of our approach.

This paper is organized as follows. In Section 3 we present sample light curves computed using our algorithm, validate it against numerical integration, and discuss its performance in terms of computational speed and precision. In Section 4 we extend the model to apply to illumination sources of finite size and surfaces that scatter light anisotropically. We discuss implications, applications, and limitations of our model in Section 5 and summarize our findings in Section 6. For convenience, Tables 1–4 at the end list all symbols and variables used in the text, with descriptions and links to the equations in which they are defined.

3. Reflected Light Curves in Starry

3.1. Sample Light Curves

Figure 1 shows a sample application of the algorithm developed in this paper: a reflected light phase curve of the Earth over the course of one year. The model is computed using the methodology in Appendix B from an $l = 25$ spherical harmonic expansion of the cloudless Earth, where the oceans are given an albedo of zero and the continents an albedo of unity (note, however, that since the light curve is normalized, the model does not depend on the value of the latter). The Earth is assumed to be a perfect Lambertian scatterer, so effects like the phase dependence of Rayleigh scattering and specular reflection (glint) from the oceans are neglected (but see Section 4 for an extension of the model to non-Lambertian

Table 3
List of Common Vector Quantities Used in this Paper

Symbol	Description	References
	Bases	
$\tilde{\boldsymbol{g}}$	Green's basis	(A4)
$\tilde{\boldsymbol{p}}$	polynomial basis	(A8)
$\tilde{\boldsymbol{y}}$	spherical harmonic basis	(A1)
	Angles and angular parameters	
$\boldsymbol{q}$	cosine-like parameter of $\boldsymbol{\alpha}$	(C51)
$\boldsymbol{\alpha}$	modified angle along occultor limb	(C43)
$\boldsymbol{\lambda}$	angle along occulted body limb	Appendix C.9
$\boldsymbol{\phi}$	angle along occultor limb	Appendix C.6
$\boldsymbol{\xi}$	angle along terminator	Appendix C.8
	Integrals	
$\boldsymbol{r}^{\top}$	unocculted solution in emitted light	Appendix A.1
$\boldsymbol{s}^{\top}$	occultation solution in emitted light	Appendix A.1
$\mathbb{i}$	helper integral	(C42)
$\mathbb{j}$	helper integral	(C45)
$\mathbb{k}$	helper integral	(B3)
$\mathbb{p}^{\top}$	primitive integral	(C27)
$\mathbb{q}^{\top}$	primitive integral	(C29)
$\mathbb{r}^{\top}$	unocculted solution in reflected light	(B1)
$\mathbb{s}^{\top}$	occultation solution in reflected light	(C26)
$\mathbb{t}^{\top}$	primitive integral	(C28)
$\mathbb{u}$	helper integral	(C56)
$\mathbb{w}$	helper integral	(C58)
	Other vector quantities	
$\boldsymbol{a}$	vector of albedo values on a discrete surface grid	(12)
$\boldsymbol{d}$	data vector	Section 5.1
$\boldsymbol{f}$	vector of flux values	Section 5.1
$\boldsymbol{i}$	illumination profile in polynomial basis	(A15)
$\boldsymbol{x}''$	solution to quartic in terminator frame	(C3)
$\boldsymbol{y}$	vector of spherical harmonic coefficients	Appendix A.1
$\boldsymbol{\mu}$	prior mean	Section 5.1

Table 4
List of Common Matrices Used in this Paper

Symbol	Description	References
	Linear operators	
$\boldsymbol{A}$	change-of-basis matrix: $\tilde{\boldsymbol{y}} \rightarrow \tilde{\boldsymbol{g}}$	Appendix A.1
$\boldsymbol{A_1}$	change-of-basis matrix: $\tilde{\boldsymbol{y}} \rightarrow \tilde{\boldsymbol{p}}$	Appendix A.1
$\boldsymbol{A_2}$	change-of-basis matrix: $\tilde{\boldsymbol{p}} \rightarrow \tilde{\boldsymbol{g}}$	Appendix A.2
$\boldsymbol{C}$	posterior covariance matrix	Section 5.1
$\boldsymbol{I}$	illumination operator	(A17)
$\boldsymbol{P}$	pixelization operator	(12)
$\boldsymbol{P}^{+}$	inverse pixelization operator	(14)
$\boldsymbol{R}$	rotation matrix: $\mathcal{F}_0 \rightarrow \mathcal{F}$	Appendix A.1
$\boldsymbol{R}'$	rotation matrix: $\mathcal{F} \rightarrow \mathcal{F}'$	Appendix A.1
$\boldsymbol{R}''$	rotation matrix: $\mathcal{F} \rightarrow \mathcal{F}''$	Appendix A.1
$\boldsymbol{X}$	starry design matrix	Section 5.1
$\boldsymbol{\Lambda}$	prior covariance matrix	Section 5.1
$\boldsymbol{\Sigma}$	data covariance matrix	Section 5.1
	Integrals	
$\boldsymbol{G}$	anti-exterior derivative of $\tilde{\boldsymbol{g}}$	(C25)
$\mathbb{H}$	helper integral	(C75)
$\mathbb{L}$	helper integral	(B3)
$\mathbb{M}$	helper integral	(B3)

scatterers). The observer is assumed to be along the ecliptic, so the illumination source is along the x–z plane of a right-handed Cartesian coordinate system, with $\hat{z}$ pointing toward the observer and $\hat{x}$ pointing to the right on the sky. The axis of rotation of the Earth is therefore tilted clockwise away from $\hat{y}$ by 23°.5. The images at the top show snapshots of the disk of the Earth throughout the observation; below each one, we plot in blue the normalized phase curve at that phase over a single rotation. The orange dots correspond to a brute force numerical solution, obtained by discretizing the disk on a grid of $\sim 10^5$ points and summing over the dayside. The models agree to within the numerical precision of the brute force solution (about 100 ppm of the planetary flux in this case).

While the dominant signal in the phase curve is the sine-like envelope due to the changing phases of the Earth, the local behavior of the light curve at each phase is complex and varies significantly over the course of the year. Unlike phase curves in thermal light, which primarily encode low-order spatial information (since the region of integration is always the full disk), phase curves in reflected light encode information at different scales depending on the phase. At crescent phase, the region of the disk contributing to the total flux is a narrow lune; these measurements therefore encode information primarily about high-l modes. At full phase, the region of integration is the full disk, so these measurements encode information about low-l modes. Furthermore, because of the obliquity of the Earth, the orientation of the crescent lune changes relative to features on the surface over the course of one orbit, changing the relative contribution of different portions of the surface to the flux and increasing the overall information content of the observation. As we will show in Section 5.2, the information content of reflected light phase curves is overwhelmingly higher than that of phase curves in thermal light, particularly for planets with significant obliquity.

Figure 2 shows another light curve of the rotating Earth, but this time taken during an occultation by the Moon. The map of the Earth is the same as before, but the observer is now along the equatorial plane of the Earth. The top panel is a reproduction of Figure 7 in Luger et al. (2019a) for the case of thermal light, where the Moon is seen to travel across the disk of the Earth from southwest to northeast, progressively occulting South America (dip), the Atlantic (peak), and Africa (dip). As before, the blue curve is the analytic solution and the orange dots correspond to the numerical solution.

The bottom panel of the figure shows a light curve for the same occultation geometry, but seen instead in reflected light, with the Sun to the top left and slightly out of the page, corresponding to some point during northern summer. Note the same dip-peak-dip pattern, albeit with significantly different amplitudes. In particular, the transit across South America is deeper, because it occurs close to local noon, when the illumination is highest; conversely, the transit across Africa occurs close to local dusk, when the illumination is close to zero. As before, the light curves computed using starry agree to within the numerical precision of the brute force solution.

Our last sample light curve is Figure 3, which shows a secondary eclipse light curve of the Earth as it is occulted by the Sun. The model for the Earth is the same as above, and the observer is now close to the ecliptic, but slightly misaligned so that the Earth is occulted behind a solar latitude of 30° (i.e., at a solar impact parameter of 0.5). The observation takes place at

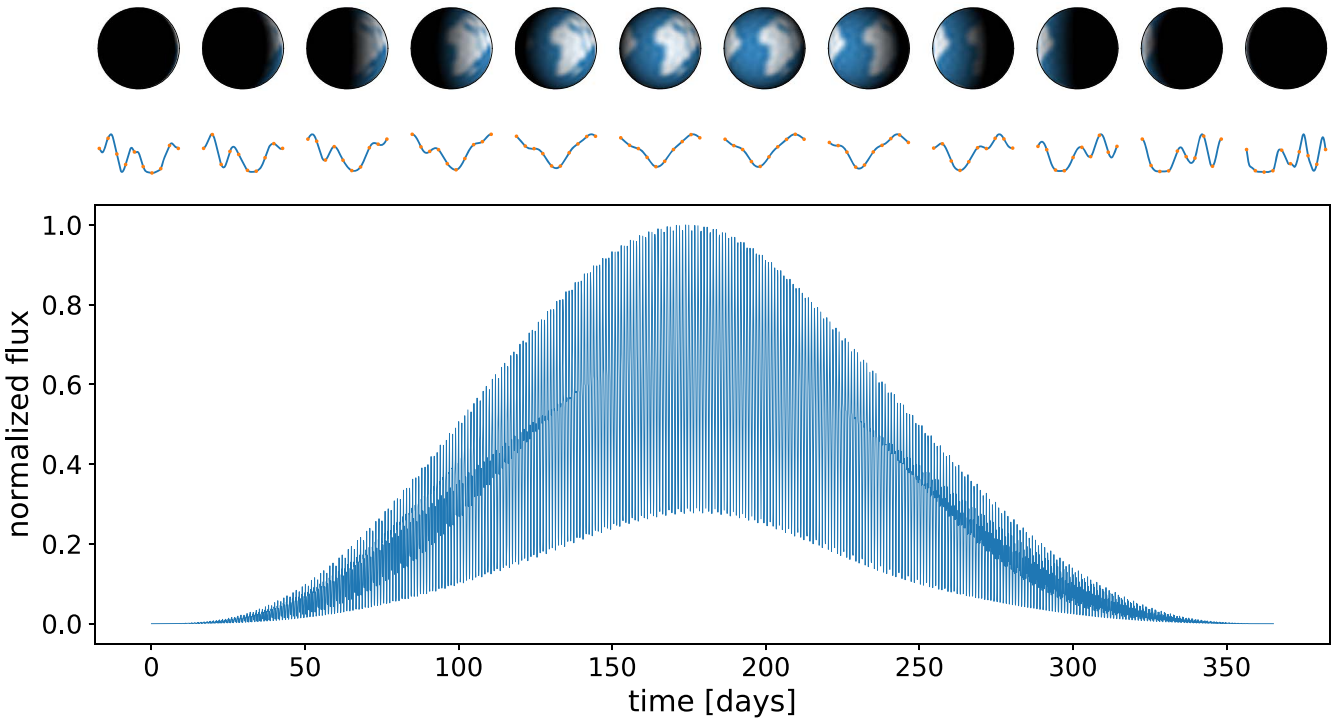

Figure 1. Mock reflected light phase curve of the cloudless Earth expanded to spherical harmonic degree $l = 25$, viewed along the ecliptic. The main plot shows the phase curve over the course of one year. The images at the top show the corresponding progression of the phases of the Earth, from new phase to full phase and back to new phase. Below each image we show a normalized 24 hr segment of the light curve at that phase (blue). Orange dots correspond to the flux computed from brute force numerical integration on a grid of $\sim 10^5$ points.[8]

the June solstice, so the Earth is tilted by 23°.5 out of the page. As before, the top panel shows the light curve in thermal light; this is similar to the top panel of Figure 13 in Luger et al. (2019a). The orange dots again correspond to the numerical solution. The center panel shows the same light curve in reflected light. Because the observation occurs very close to full phase, the normalized light curves look very similar to each other. The bottom panel shows the difference between the two (reflected minus thermal), which is only on the order of a few percent. In fact, because the illumination profile is proportional to the cosine of the viewing angle, μ, and the reflection is assumed to be isotropic, the (normalized) secondary eclipse light curve in reflected light is to good approximation equal to a limb-darkened thermal occultation light curve with linear limb darkening coefficient $u_1 = 1$. As we will see later, for very close-in planets, this approximation breaks down, since the illumination phases at secondary eclipse ingress and egress are sufficiently different from full phase.

3.2. Performance

As we discuss in the Appendix, the model for phase curves and occultation light curves in reflected light may be expressed analytically in terms of purely algebraic and trigonometric functions and in some cases incomplete elliptic integrals of the first, second, and third kinds. We have derived efficient and numerically stable recursion relations to compute the relevant expressions and their derivatives. At times, these involve the evaluation of certain expressions numerically, especially when doing so leads to either a speed-up or a significant gain in numerical precision. In particular, as we discuss in Appendix C.1, the integration boundaries during an occultation sometimes depend on the solution to a quartic equation. While this can be solved in closed form, the analytic solution can often be very unstable. We therefore solve the quartic numerically, attaining a precision for the roots within a few orders of magnitude of machine (double) precision.

Figures 4 and 5 summarize the precision and computation time of the starry algorithm for two typical scenarios: a phase curve evaluation (Figure 4) and an occultation evaluation (Figure 5). Blue points correspond to the reflected light algorithm developed in this paper, while purple points correspond to the thermal light algorithm from Luger et al. (2019a) for the same occultation geometry, but without an illumination source. Solid and dashed lines correspond to evaluations without and with gradient propagation, respectively (see Section 3.3 for details). The orange and red dots correspond to numerical evaluation of the flux: brute force integration by summation on a grid of $\sim 10^6$ points (orange) and two-dimensional adaptive Gaussian quadrature using the dblquad function in scipy (Jones et al. 2001) with both absolute and relative error tolerances set to 10^{-3} (red). In both figures, the vertical axis corresponds to the evaluation time in seconds for a single flux computation, while the size of the points is proportional to the base-10 log of the relative error. For the starry solutions, the latter is estimated as the max-min difference in the flux over one thousand evaluations in which the input parameters are perturbed within an order of magnitude of machine epsilon; this is therefore a probe of the condition number of the starry algorithm and captures only error due to numerical instabilities. It is worth emphasizing that this is a measurement of the precision of the algorithm, rather than the accuracy, because it would be computationally intractable to compute a solution more accurate than this using a different

[8] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/earthphase.py

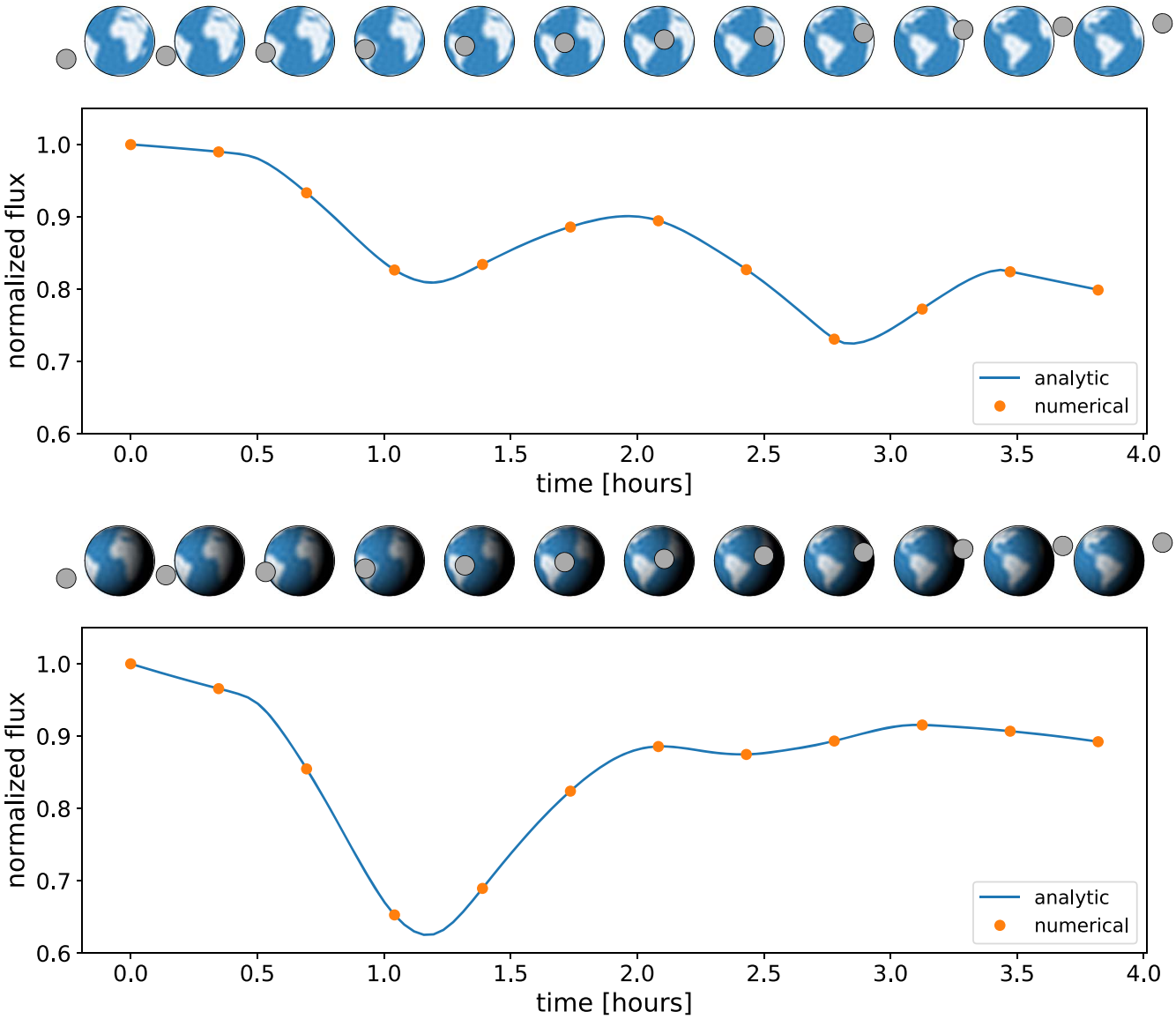

Figure 2. Mock light curves of the Moon occulting a rotating, cloudless Earth expanded to spherical harmonic degree $l = 25$. Black curves show the analytic solution; orange dots correspond to brute force numerical integration on a grid of $\sim 10^5$ points. The top panel shows the light curve in emitted light and is the same as in Figure 7 in Luger et al. (2019a). The bottom panel (this work) shows the same light curve in reflected light during northern summer.[9]

algorithm. We argue that this measurement can be interpreted to mean that the algorithm is also accurate, but detailed quantification of this difference is beyond the scope of this paper. For the numerical solutions, the error is estimated as the difference between the numerical flux and the starry flux.

For both phase curves and occultations, the starry reflected light algorithm is 1–2 orders of magnitude slower than the emitted light algorithm, owing primarily to the increased complexity of the reflected light model. For phase curves (Figure 4), the thermal solution vector $\boldsymbol{r}^\top$ (Equation (A7)) is a constant that can be precomputed, while the analogous vector in the reflected light case, $\mathbb{r}^\top$ (Equation (B2)), must be evaluated recursively each time. For occultations (Figure 5), the slower evaluation in the reflected light case is primarily due to the time spent solving the quartic equation for the points of intersection between the occultor and the day/night terminator of the illuminated body (Appendix C). This contributes the same overhead at all map degrees l, resulting in a gentler scaling in l than for the thermal case; for large l, the evaluation time for the two algorithms is within a factor of 2–3. In terms of precision, the algorithms are comparable, particularly for occultations. For both phase curves and occultations, the numerical error up to $l = 10$ is less than one part per trillion (10^{-12}) for both thermal and reflected light curves.

Compared to either numerical evaluation method, the starry reflected light solutions are 4–5 orders of magnitude faster and about 10 orders of magnitude more precise. While different grid sizes and different error settings for the numerical integration change the balance slightly between these numbers, the starry solution is always many orders of magnitude faster and more precise than either method. In particular, because of the complicated integration boundaries (see, for example, Figure 15), two-dimensional Gaussian quadrature struggles to reach adequate accuracy in a reasonable amount of time, while integration on a grid fails to capture the curvature of the

[9] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/earthmoon.py

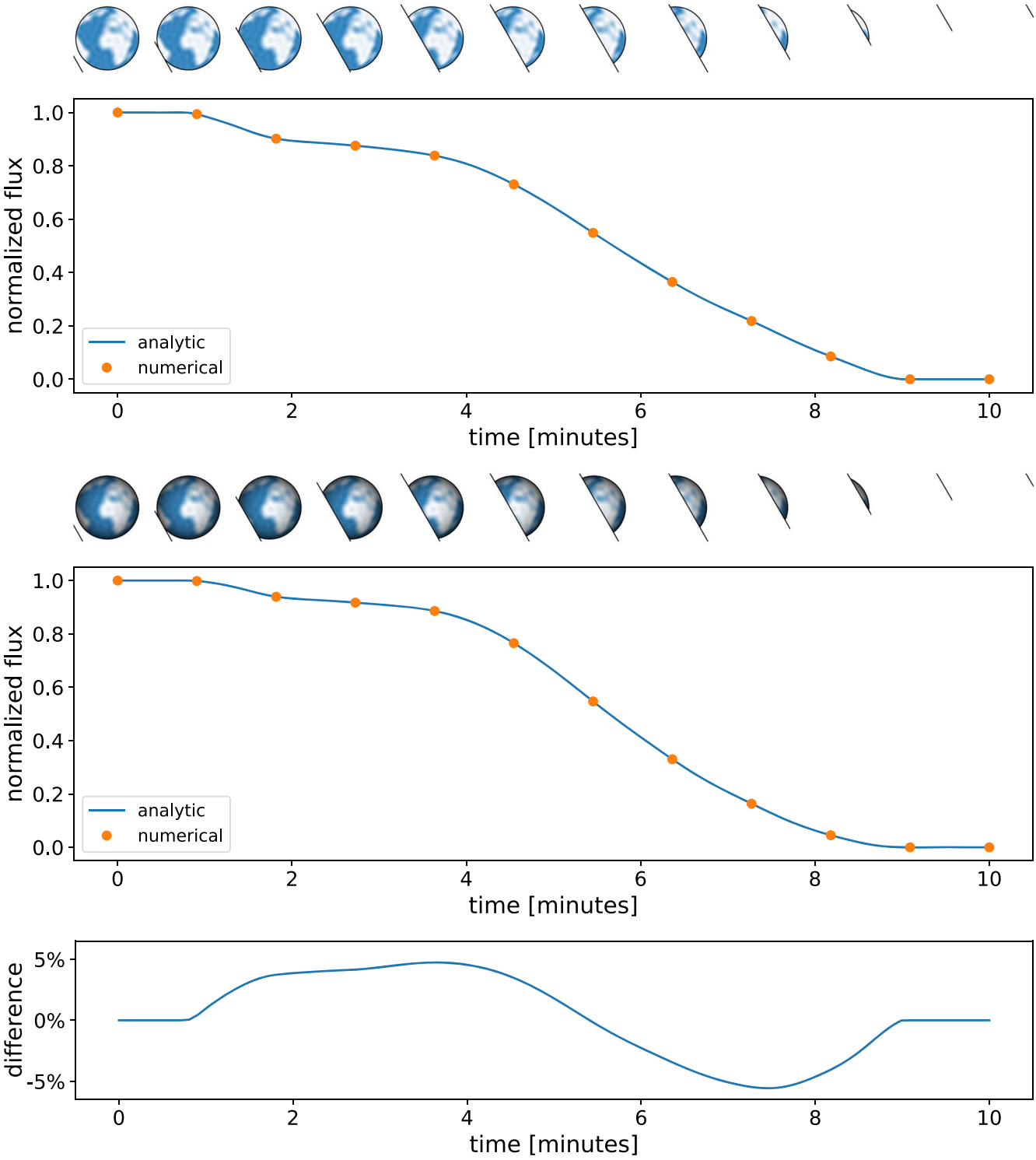

Figure 3. Mock secondary eclipse ingress light curves of the cloudless Earth expanded to spherical harmonic degree $l = 25$, viewed from an orientation where the Earth is occulted behind a solar latitude of 30°. Black curves show the analytic solution; orange dots correspond to brute force numerical integration on a grid of $\sim 10^5$ points. The top panel shows the light curve in emitted light and is similar to Figure 13 in Luger et al. (2019a). The middle panel (this work) shows the same light curve in reflected light. The bottom panel shows the difference between the normalized reflected and emitted light curves.[10]

integration boundaries. Moreover, neither method yields the gradient of the solution with respect to the input parameters, which can be extremely useful for optimization and inference problems (see Section 3.3) below.

Note, importantly, that as we mentioned above, the reported error of the starry solution is only the *numerical* error of the algorithm: it does not capture any systematic error due to, say, an error in the derivation of the method. To this end, we rely on the Jupyter notebooks containing derivations and validations of the main equations in the Appendix, whose links (as footnote) appear below to the equation.We have also developed an

[10] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/earthsun.py

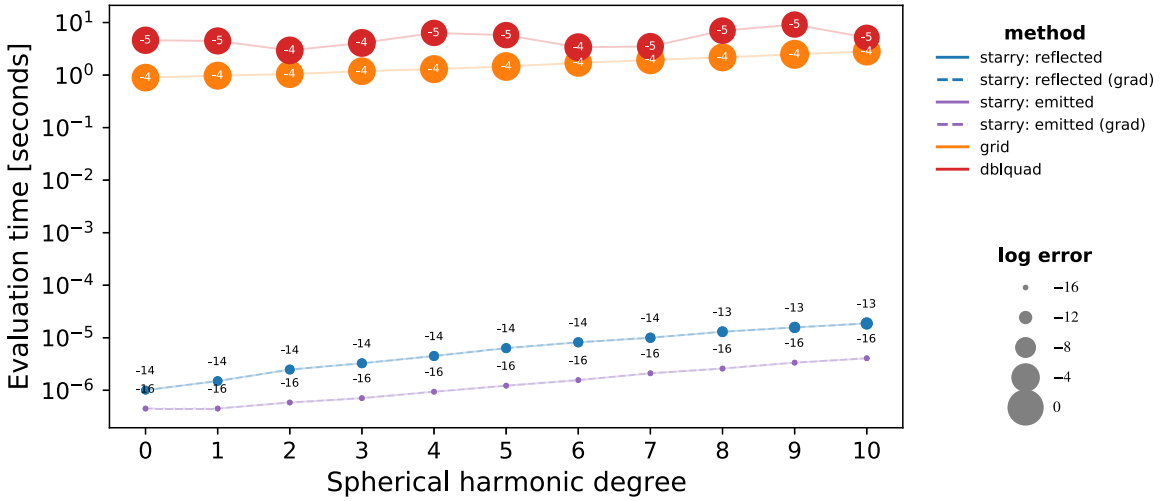

Figure 4. Evaluation time (vertical axis) and numerical precision (point size) for a single flux evaluation in the absence of an occultor as a function of spherical harmonic degree for different methods. In purple we show results for the emitted light starry algorithm from Luger et al. (2019a; solid: no gradient; dashed: with gradient), and in blue we show results for the reflected light algorithm from this paper (solid: no gradient; dashed: with gradient). For comparison, in we also show results for discrete integration on a grid (orange) and for numerical integration using two-dimensional Gaussian quadrature (red); neither of these include gradient evaluations. The reflected light algorithm is comparable in efficiency and precision to the emitted light algorithm. It is ∼5 orders of magnitude faster and ∼10 orders of magnitude more precise than numerical integration.[11]

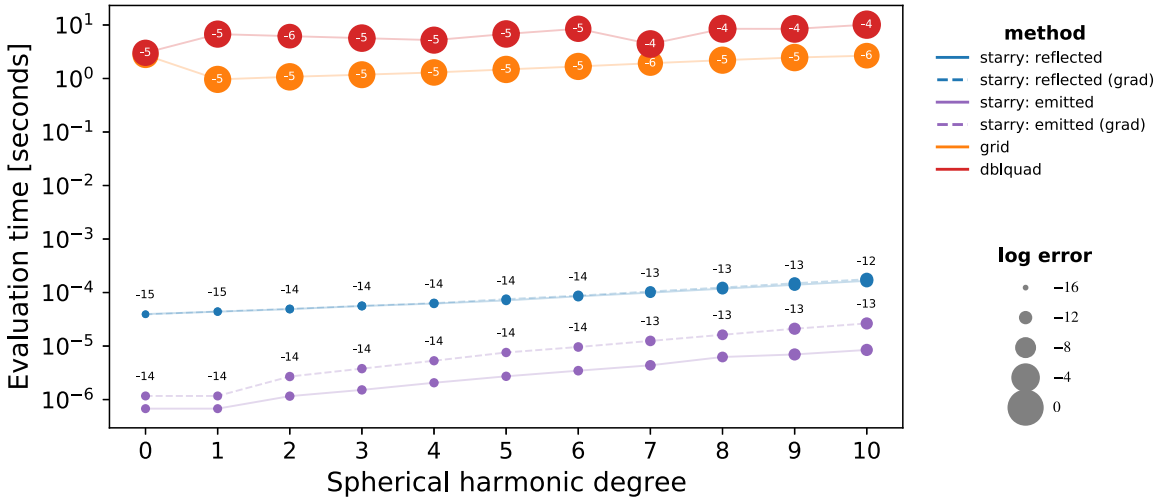

Figure 5. Same as Figure 4, but for an occultation evaluation in which the occultor intersects the terminator (case 6 in Appendix C). The reflected light algorithm is around one order of magnitude slower and comparably precise to the emitted light algorithm. It is ∼4 orders of magnitude faster and ∼10 orders of magnitude more precise than numerical integration.[12]

extensive suite of unit tests comparing the starry solution to the numerical solution over a large grid of input parameter values, and verified that the solutions agree to within the precision of the numerical method. That said, there are specific cases in which the algorithm presented in the Appendix suffers from numerical instabilities. These generally happen due to division by small numbers or catastrophic cancellation in the recursion, and often occur near configurations involving grazing occultations, near-total occultations, terminator semiminor axis $b \approx 0$ or $b \approx 1$, etc. To mitigate these, we introduce various tolerance parameters in the code to either nudge the inputs away from these singular points or switch to a different evaluation method.

Table 5
Tolerance Parameters Used in the Code

Symbol	Description	Value
ϵ_0	If ξ is this close to $\frac{n\pi}{2}$, compute $\mathbb{t}_2$ in the limit $\xi = \frac{n\pi}{2}$	10^{-13}
ϵ_1	If $\lvert\sin\theta\rvert$ or $\lvert\cos\theta\rvert$ are less than this value, set to this value	10^{-12}
ϵ_2	If k^2 is within this value of unity, nudge it away	10^{-12}
ϵ_3	If $\lvert b_o - r_o\rvert$ is less than this value, nudge b_o away from r_o	10^{-8}
ϵ_4	If b_o is within this amount of $r_o - 1$, nudge it away	10^{-8}
ϵ_5	If b_o is within this amount of $r_o + 1$, nudge it away	10^{-8}
ϵ_6	If $\lvert\sin\alpha\rvert$ is less than this value, set to this value	10^{-8}
ϵ_7	If b is within this value of zero, nudge it away	10^{-8}
ϵ_8	If b_o is within this amount of $1 - r_o$, nudge it away	10^{-7}
ϵ_9	If two quartic roots are this close, eliminate one of them	10^{-7}
ϵ_{10}	If b is within this value of unity, set it to unity	10^{-6}
ϵ_{11}	If θ is within this amount of $\frac{\pi}{2}$ when $r_o = 1$, nudge it away	10^{-5}

[11] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/speed.py
[12] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/speed.py

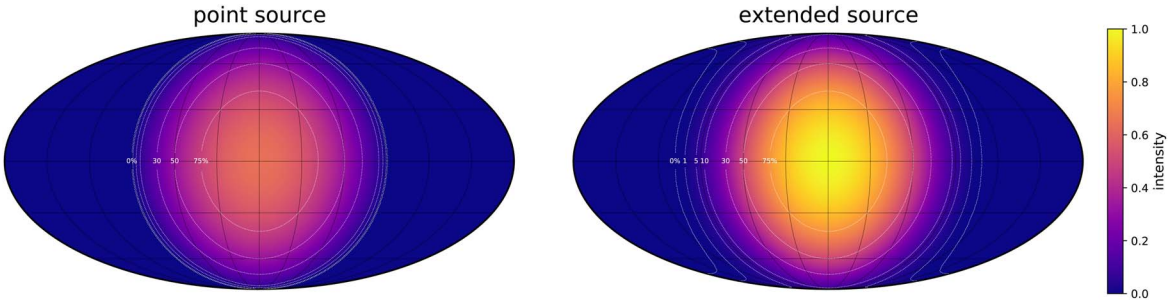

Figure 6. Normalized surface intensity on Kelt-9b viewed in a Mollweide projection assuming a point illumination source (left) and accounting for the finite extent of the star (right). The day/night terminator extends about 17° past where it is in the point-source case. The substellar intensity is higher in the extended source case, because the subplanet point on the star is closer to the planet than in the case where the star is a point source located at the center of the star.[13]

These parameters are outlined in Table 5 at the end. In the vicinity of the cases described in that table, the precision of the starry algorithm will be reduced to (roughly) the value of the tolerance parameter, which in extremely rare cases can be as high as 10^{-5}.[14]

3.3. Implementation and Usage

The algorithm presented in this paper has been implemented in the Python package starry, which can be installed from GitHub or via the Python package manager pip. The algorithm is coded in a mixture of C++ with forward automatic differentiation using the Eigen library (Guennebaud & Jacob 2010) and Python with backward differentiation using just-in-time compiled theano operations (Theano Development Team 2016). The user interface, however, is purely in Python. The theano backend facilitates integration with the exoplanet modeling package (Foreman-Mackey et al. 2020) and in particular with pymc3 (Salvatier et al. 2016) for inference with gradient-based Markov Chain Monte Carlo (MCMC) schemes such as Hamiltonian Monte Carlo (HMC; Duane et al. 1987) and No-U-Turn Sampling (NUTS; Hoffman & Gelman 2011). Complete documentation and an extensive library of tutorials is available online. The links next to each of the figures (inside parenthesis) point to the Python scripts used to generate them and may also help in learning how to use starry.[15,16]

4. Extensions

The algorithm discussed above and derived in the Appendix computes light curves in the limit that (1) the body is illuminated by a point source and (2) the body is an ideal Lambertian scatterer. Both of these assumptions can be relaxed within starry, and below we discuss modifications to the code to allow for this.

4.1. Extended Illumination Source

In the limit that the angular size of the star as seen from the planet is small, the illumination profile on the surface of the planet will decrease as the cosine of the angle between the surface normal and the star, reaching zero at the day/night terminator, an angle $\pi/2$ away from the substellar point. However, if the star is sufficiently large and the planet is sufficiently close-in, rays originating from near the limb of the star will reach points on the planet surface beyond this angle. If the stellar radius $R_\star$ is larger than the planet radius R_p, the angular extent of the true day/night terminator past $\pi/2$ is given by

$$\tau = \arcsin\left(\frac{1 - R_p/R_\star}{a/R_\star}\right), \tag{4}$$

where a the semimajor axis of the orbit (where we implictly assume the eccentricity is zero). For planets like the Earth, this quantity is only about 0°.26, resulting in a negligible effect on the planet's light curve. However, for very close-in planets, the effect can be significant (Knuth et al. 2017; Carter 2019). For instance, the hot Jupiter Kelt-9b has $R_p/R_\star = 0.083$ and $a/R_\star = 3.16$ planet (Wong et al. 2019). Assuming zero eccentricity and ignoring any stellar oblateness (see Ahlers et al. 2020), the day/night terminator extends $\tau \approx 17°$ past the limb of that planet. Figure 6 shows the illumination profile of this planet in a Mollweide projection, with the substellar point at the center, for the point-source approximation (left) and accounting for the finite size of the star (right). In addition to the displaced day/night terminator, the main difference between the two profiles is the substellar intensity, which is significantly higher in the extended source case. This is due to the simple fact that, in the point-source case, the illumination source is placed at the center of the star, which is one stellar radius farther from the planet than the point closest to the planet (the subplanetary point) in the extended source case. Once accounting for this difference, the fractional change in the intensity on the planet away from the substellar point is similar in both cases, and the intensity anywhere beyond $\pi/2$ is less than one-tenth the peak value.[17]

The illumination profile in the extended source case may be computed as the two-dimensional integral of the point-source illumination profile over the visible portion of the stellar disk. While this integral may in theory be computed analytically (see, for instance, Kopal 1954, who derived series solutions to this problem), the resulting profile on the planet surface will not in general be exactly expressible in terms of spherical harmonics, a necessary condition for the starry algorithm. For simplicity, we therefore compute the illumination profile for extended sources by averaging the contribution of

[13] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/extended.py
[14] https://github.com/rodluger/starry/tree/master/tests
[15] https://github.com/rodluger/starry
[16] https://starry.readthedocs.io
[17] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/tau.ipynb

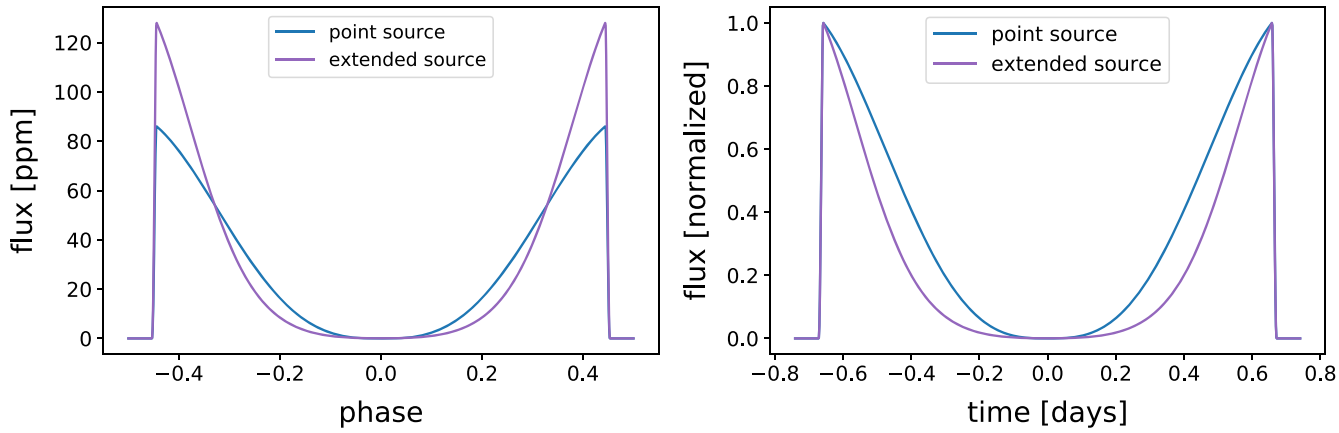

Figure 7. Reflected light phase curve model for Kelt-9b, assuming a spherical albedo $A = 0.2$. The transit (phase zero) is not included. Blue is the starry model assuming a point-source illumination; purple accounts for the finite size of the star. The left panel shows the two models in parts per million of the stellar flux; the right panel shows the models normalized so their maximum value is unity. The primary effect of the extended source size is to increase the planet flux near full phase, since the stellar surface is on average slightly closer to the planet, and to change the overall curvature of the phase curve. The shape of secondary eclipse, however, is relatively insensitive to the point-source approximation (see Figure 8).[18]

source_npts point sources uniformly distributed across the projected disk corresponding to the portion of the stellar surface visible from the planet, where source_npts is a user-supplied value. In the limit $R_p \ll R_\star$, this is a spherical cap centered at the subplanet point with radius $R_\star \cos\tau$. In Figure 6 we set source_npts $= 300$, but in practice we find that $\sim$30 points are sufficient for even the most extreme cases such as Kelt-9b. Note, importantly, that while this method allows one to account for the effect of stellar limb darkening on the illumination profile of the planet, this has not been implemented in starry.

Figure 7 shows the practical implications of the finite stellar size for the phase curve of Kelt-9b. The left panel shows the reflected light phase curve of the planet in parts per million, assuming a spherical albedo of 0.2, for the point-source approximation (blue) and including the effect of the extended source with source_npts $= 300$ (purple). The increased illumination at the substellar point results in a $\sim$30 ppm increase in the value of the phase curve close to full phase ($\pm 1/2$). Close to a phase of zero, the extended source results in decreased flux, since the portion of the star illuminating the planet (the region close to the limb) is slightly *farther* away, by a factor of $\sqrt{1 + (a/R_\star)^{-2}}$. This results in a steeper phase curve, which can be seen in the right panel, where the light curves have been normalized to their maximum value.

While modeling the extended size of the star is essential to getting the shape of the phase curve correct, the same is not true for secondary eclipse. Figure 8 shows the normalized secondary eclipse model for Kelt-9b under the point-source approximation (blue) and the extended source model (purple). Neglecting the fact that the *depth* of secondary eclipse is significantly different between the two models (see the left panel of Figure 7), the difference in *shape* between the two curves is almost negligible. For reference, the figure shows two additional models one might consider using to fit a secondary eclipse light curve: a uniform (unilluminated) disk (solid orange) and a disk whose intensity falls as μ, the cosine of the viewing angle (dashed orange). Both can be computed using the classical Mandel & Agol (2002) transit model; the latter corresponds to a linearly limb-darkened sphere and is functionally equivalent to a Lambertian sphere seen at full phase. However, neither approximation is particularly good, since Kelt-9b changes illumination phase significantly from ingress to egress owing to its proximity to the star.

[18] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/kelt9b.py

4.2. Non-Lambertian Scatterers

The second assumption we now seek to relax is that of Lambertian scattering. A perfect Lambert sphere reflects light isotropically, so the measured intensity at a point on the surface is strictly proportional to the product of the cosine of the angle of incidence and the cosine of the viewing angle. While this is convenient from a modeling standpoint, it is hardly ever true in practice. For planets and moons in particular, there is often a strong phase dependence in the scattering. Rayleigh scattering in planetary atmospheres is preferentially in the forward/backward direction, while clouds and oceans can contribute strong specular reflection. Moreover, rough surfaces can have complex scattering behavior due to changes in the orientation of the surface normal on small scales and effects such as multiple reflections and self-shading.

In principle, any of these processes can be accounted for in the starry algorithm by modifying the linear operator $\boldsymbol{I}$ (Equation (A17)), which in the Lambertian case simply weights the spherical harmonic expansion of the albedo by the cosine-like illumination profile to obtain the observed intensity at a point on the surface (see Equations (A18) and (A20)). For non-Lambertian scattering, this matrix must also account for the phase dependence of the reflection: in particular, it will depend not only on the angle between the surface normal and the incident radiation, ϑ_i, but also on the angle between the surface normal and the reflected radiation (i.e., the direction toward the observer), ϑ_r. It may also depend on the azimuthal angles of the incident and reflected rays, ϕ_i and ϕ_r, respectively. These four angles are shown in Figure 9, showing the incoming radiation (source) vector $\boldsymbol{s}$ and the outgoing radiation (viewer) vector $\boldsymbol{v}$ in a frame in which the z-axis points along the surface normal.

Treatment of a generalized, flexible scattering model is beyond the scope of this paper; see Heng et al. (2021) for recent results on this front. However, as an example of how a scattering model may be incorporated into the starry algorithm, we consider in detail the case of the rough surface scattering

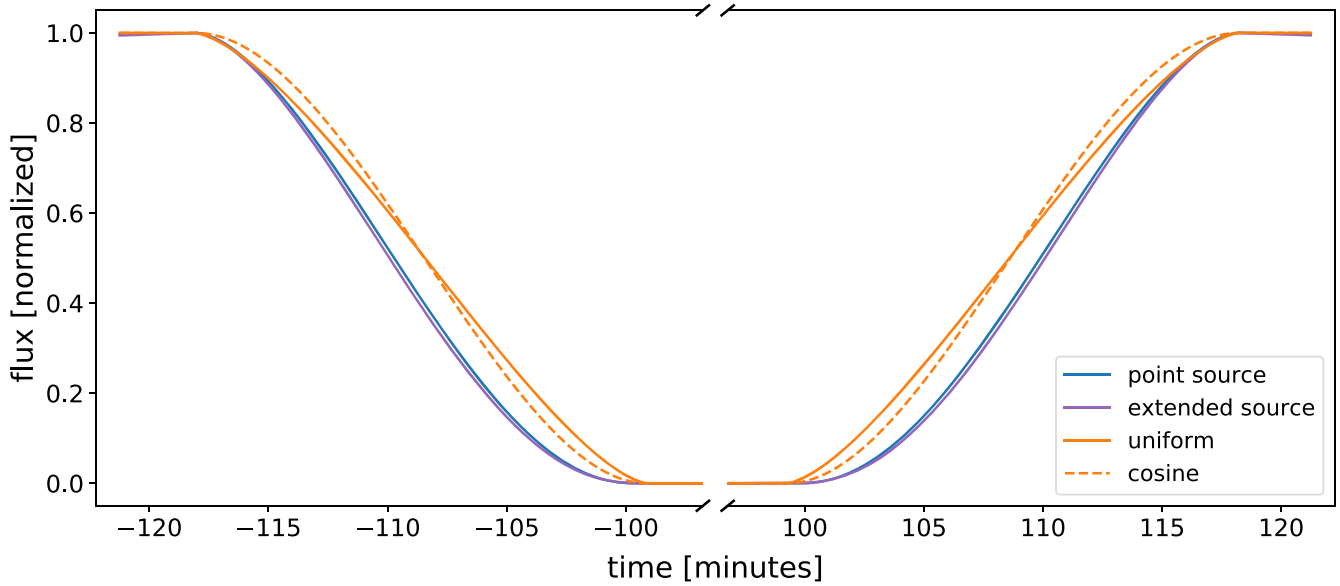

Figure 8. Normalized reflected light secondary eclipse model for Kelt-9b, assuming a spherical albedo $A = 0.2$. Blue is the starry model assuming a point source illumination; purple accounts for the finite size of the star. The orange curves show approximate models computed using the Mandel & Agol (2002) model: a sphere of uniform intensity (solid orange) and a sphere whose intensity falls as the cosine of the viewing angle from the center of the planet disk (dashed orange). Because Kelt-9b is so close to its host star, the illumination phase changes significantly from ingress to egress, and neither approximation accurately captures the behavior of the light curve. On the other hand, the point-source approximation agrees well with the extended source solution, modulo the difference in the depth (see Figure 7).[19]

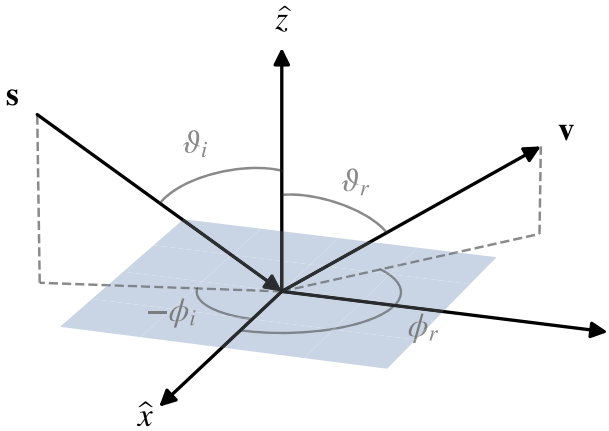

Figure 9. Scattering geometry for non-Lambertian reflection. Based on Figure 3 of Oren & Nayar (1994). The incident radiation is labeled $\boldsymbol{s}$ and the outgoing radiation is labeled $\boldsymbol{v}$. The shaded region is a small patch of surface, oriented so that the normal vector points along $\hat{z}$. The four angles relevant to the computation of the emergent intensity are also indicated.[20]

model of Oren & Nayar (1994), commonly used in computer graphics applications and solar system body modeling (e.g., Morgado et al. 2019). In this model, the surface is treated as a collection of a large number of Lambertian facets oriented at random angles relative to the average surface normal, whose net contribution to the total intensity can depart significantly from the Lambertian case. While the general model accounts for interreflections, shadowing, and an arbitrary distribution of facet orientations, in its simplest form the intensity observed at a point (x, y) on the (projected) surface of a body of unit spherical albedo may be approximated as (see Equation (30) in Oren & Nayar 1994)

$$\mathcal{I} = \mathcal{I}_{\rm Lamb}\{c_0 + c_1 \max(0, \cos(\phi_{\rm r} - \phi_{\rm i}))\sin\alpha\tan\beta\}, \quad (5)$$

where

$$c_0 = 1 - 0.5\left(\frac{\sigma^2}{\sigma^2 + 0.33}\right) \quad \alpha = \max(\vartheta_{\rm r}, \vartheta_{\rm i})$$
$$c_1 = 0.45\left(\frac{\sigma^2}{\sigma^2 + 0.09}\right) \quad \beta = \min(\vartheta_{\rm r}, \vartheta_{\rm i}), \quad (6)$$

and the angles $\vartheta_{\rm i}$, $\vartheta_{\rm r}$, $\phi_{\rm i}$, and $\phi_{\rm r}$ are all implicit functions of x, y, and the illumination source position. The term $\mathcal{I}_{\rm Lamb}$ is the Lambertian illumination profile, given by Equation (2). At a given point on the surface, and for a given source position, the intensity $\mathcal{I}$ is therefore a function of a single parameter, σ, defined as the standard deviation in radians of the distribution of facet angles (which is assumed to be a zero-mean Gaussian).[21,22]

In order to incorporate this scattering model into starry, we must weight the spherical harmonic expansion of the albedo, $\boldsymbol{y}$, by Equation (5) instead of Equation (2). Weighting by Equation (2) is (relatively) straightforward, since $\mathcal{I}_{\rm Lamb}$ is a piecewise function of the $l = 1$ spherical harmonics (see Appendix A.2). The function we must integrate when computing fluxes is therefore exactly expressible in terms of spherical harmonics and thus starry-integrable. However, Equation (5) cannot be expressed exactly in terms of spherical harmonics, so we must instead approximate it. To this end, we evaluate Equation (5) on a grid of x and y spanning the unit disk, as well as the illumination phase, parametrized by b, the semiminor axis of the elliptical segment defining the day/night

[19] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/kelt9b.py
[20] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/scattering.py
[21] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/OrenNayar.ipynb
[22] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/OrenNayar.ipynb

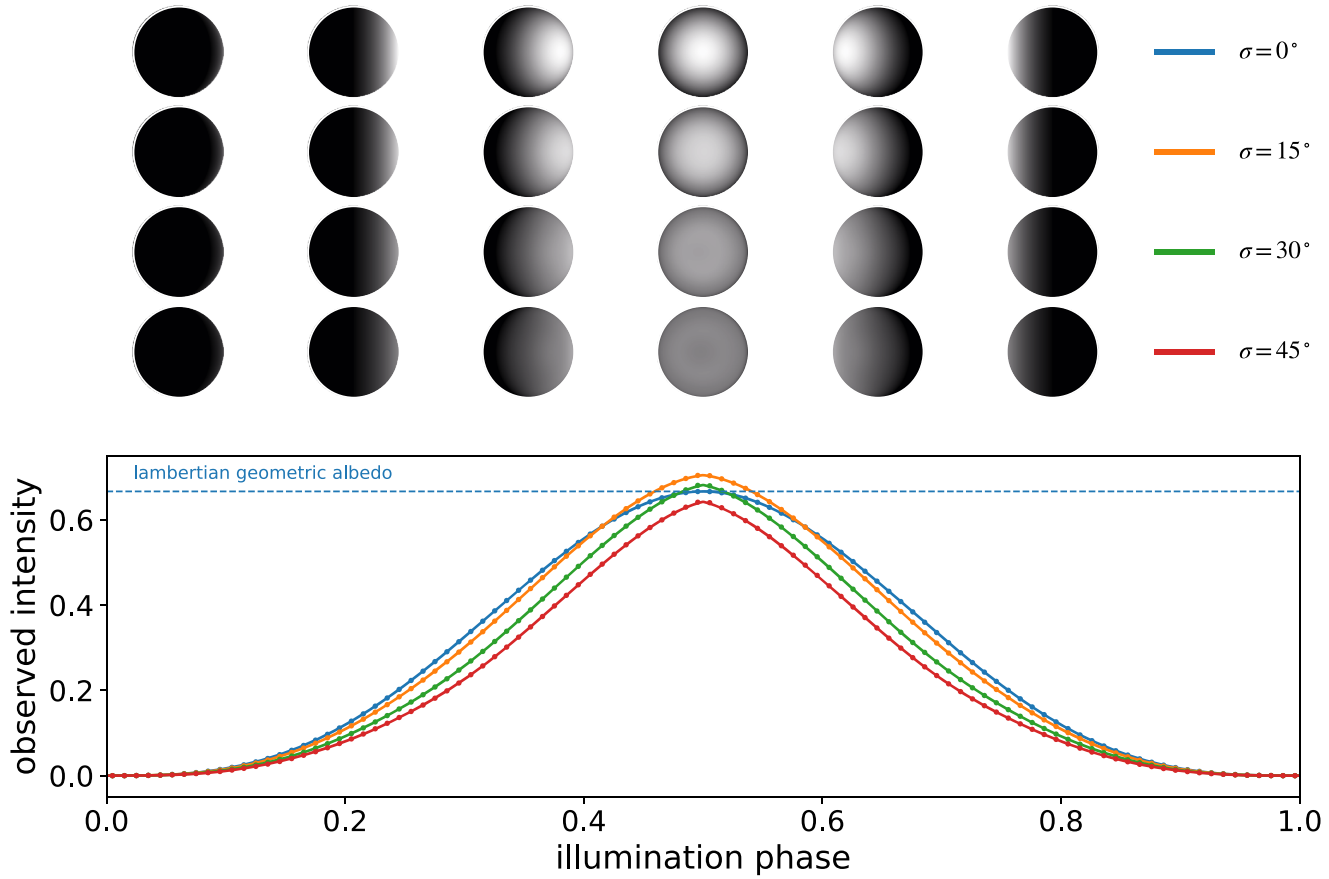

Figure 10. Intensity measured from a sphere at varying illumination phase under the Oren & Nayar (1994) scattering model. The top panel shows spheres rendered with different surface roughness coefficients ranging from $\sigma = 0°$ (the Lambertian case) to $\sigma = 45°$. The bottom panel shows the corresponding phase curves for a sphere of unit spherical albedo illuminated by a point source, computed analytically from a degree 5 expansion of the scattering law. Dots correspond to the intensity computed numerically directly from Equation (30) in Oren & Nayar (1994).[23]

terminator (see Appendix A.2). We then fit to this a polynomial of total degree 5 in x, y, and $z \equiv \sqrt{1 - x^2 - y^2}$ and degree 5 in b and degree 4 in $b_c \equiv \sqrt{1 - b^2}$. Then, for a given value of b and b_c, we construct the operator $\boldsymbol{I}$ out of the polynomial coefficients in the same way as we constructed the Lambertian operator in Appendix A.2. More details about our approximation can be found in the Jupyter notebook accompanying Equation (5).

Figure 10 shows spheres of unit albedo with different surface roughness coefficients σ and their corresponding phase curves. The sphere in the top row is perfectly Lambertian; its phase curve (blue) peaks at a value of $2/3$, equal to the geometric albedo of a Lambert sphere. Increasing the surface roughness results in a greater relative contribution of flux from the limb of the object near full phase, since there now exist facets reflecting light directly back toward the observer (remaining rows and curves in the phase curve plot). Conversely, less light is scattered back to the observer at the subillumination point. These competing effects lead to phase curves that peak at a super-Lambertian value for small roughness coefficients ($\sigma = 15°$, orange) and at a sub-Lambertian value for large roughness coefficients ($\sigma = 45°$, red). We validate our calculations by computing the phase curves by numerically integrating the Oren & Nayar (1994) model over the visible disk; these are shown as the small dots in the figure, which agree to within 350 ppm of the body's flux for $\sigma = 45°$.

Our implementation of the scattering model extends just as easily to occultations and to cases where the surface does not have uniform albedo. Figure 11 shows a visible-light observation of the occultation of Io by Europa on 2009 December 4 taken by the PHEMU09 campaign (Arlot et al. 2014). The trajectory of Europa relative to Io, computed from the JPL Horizons database using ephemerides from Folkner et al. (2014) is shown at the top. We fit to this data a starry occultation model with the scattering law discussed above. For simplicity, we set the surface map equal too an $l = 15$ spherical harmonic expansion of the Galileo global color mosaic of Io (Becker & Geissler 2005) and use ephemerides from the JPL Horizons database, allowing for a small static x–y offset between Europa and Io as in Arlot et al. (2014) due to the uncertainty in the database. In total, we fit for five parameters: the two Cartesian offset terms, the flux contribution from Europa, the average albedo of Io, and the average surface roughness of Io, σ. The model is displayed in blue and closely matches the data. Note, however, that this is meant simply as a demonstration of the starry algorithm, as our model for the surface is approximate at best, given differences in the wavelength band between the Galileo observations and those of the PHEMU09 campaign, changes in the albedo of Io since the Galileo measurements, and the fact that the orientation and extent of any shadows due to volcanoes on the surface are likely different between the Galileo and PHEMU09

[23] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/oren_nayar.py

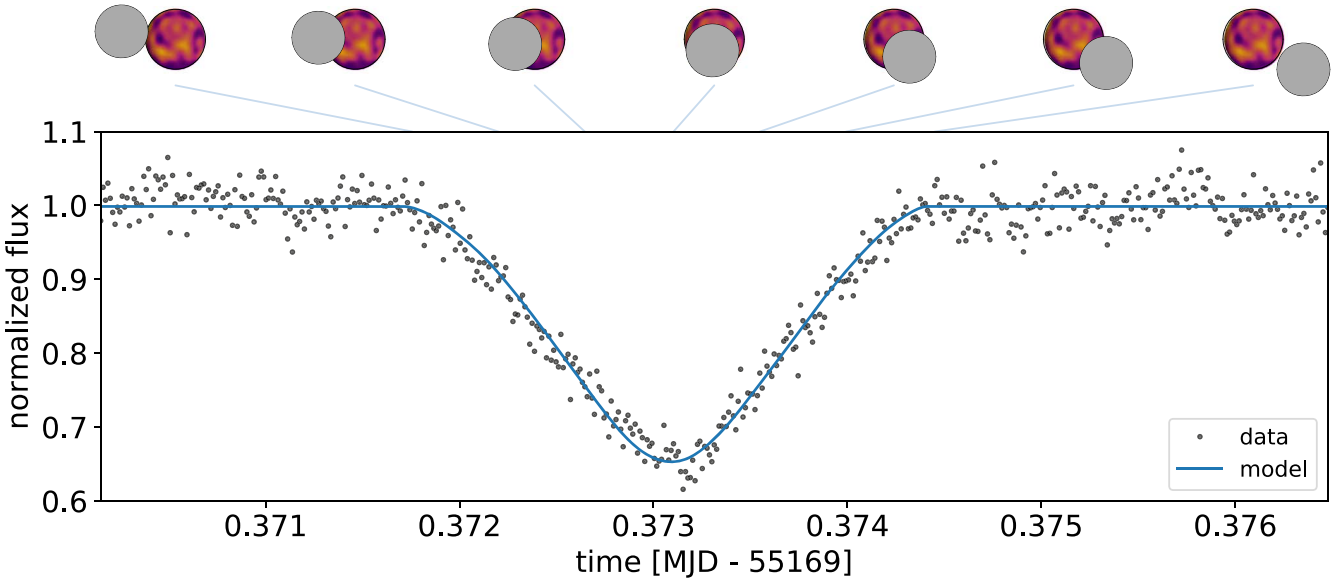

Figure 11. Visible-light occultation of Io by Europa observed on 2009 December 4 by the PHEMU09 campaign (Arlot et al. 2014). The blue line is the starry model, based of an $l = 15$ spherical harmonic fit to the Galileo global color mosaic of Io (Becker & Geissler 2005), an $l = 5$ expansion of the Oren & Nayar (1994) scattering law, and orbital information from the JPL Horizons database. See text for details.[24]

observations. Furthermore, proper modeling would entail the joint analysis of all light curves of Io taken in a given season, for which we can afford to simultaneously fit for the surface map without risk of overfitting (Bartolić et al. 2022).[25]

5. Discussion

5.1. Linearity

In the Appendix we derive closed-form expressions for the flux as a function of the spherical harmonic expansion of the albedo, $\boldsymbol{y}$: Equation (A18) for occultations and Equation (A7) for phase curves. Inspection of those equations reveals that they are both *linear* in $\boldsymbol{y}$: the flux is simply the dot product of several matrices and the vector of spherical harmonic coefficients. We may therefore write both expressions in the form

$$f = \boldsymbol{x}^\top \boldsymbol{y}, \tag{7}$$

where f is a scalar representing the model for the flux at a particular point in time and $\boldsymbol{x}^\top$ is a row vector equal to $\mathbb{s}^\top \boldsymbol{A_2 I A_1 R' R}$ (in the case of an occultation) or $\mathbb{r}^\top \boldsymbol{I A_1 R'' R}$ (in the case of a phase curve; see Appendix A.2 for details on what each of the terms represent). Now, if we let $\boldsymbol{f}$ be the vector of values of f for each point in the timeseries and construct the matrix $\boldsymbol{X}$ out of the stacked row vectors $\boldsymbol{x}^\top$, we may write our model for the entire timeseries as the dot product

$$\boldsymbol{f} = \boldsymbol{X}\boldsymbol{y}. \tag{8}$$

The linearity of the starry model is useful in several ways. For one, it can be exploited to cheaply compute the same model for different input vectors $\boldsymbol{y}$. This is useful for multiband light curves, where the same matrix $\boldsymbol{X}$ dots into several vectors $\boldsymbol{y}$, one for each observation band, or for time-dependent models, in which the model for the flux might be the Taylor series

$$\boldsymbol{f} = \boldsymbol{X}\left(\boldsymbol{y}(t)|_{t=t_0} + \frac{d\boldsymbol{y}'(t)}{dt}\bigg|_{t=t_0} t + \frac{1}{2}\frac{d^2\boldsymbol{y}'(t)}{dt^2}\bigg|_{t=t_0} t^2 + \cdots\right), \tag{9}$$

where $\boldsymbol{y}(t)$ is a time-dependent representation of the surface map, which we expand about $t = t_0$ (Luger et al. 2019b); see also Kawahara & Masuda (2020) for an alternative linear model for time-dependent maps. But perhaps even more importantly, linear models are particularly useful for inference, since under Gaussian noise properties the posterior is analytic. In particular, if our light-curve measurements are given by the data vector $\boldsymbol{d}$ whose noise model is specified by the covariance matrix $\boldsymbol{\Sigma}$, and we place a Gaussian prior on $\boldsymbol{y}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Lambda}$, the posterior mean may be written

$$\hat{\boldsymbol{y}} = \boldsymbol{C}(\boldsymbol{X}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{d} + \boldsymbol{\Lambda}^{-1}\boldsymbol{\mu}), \tag{10}$$

where $\boldsymbol{C}$ is the posterior covariance, given by

$$\boldsymbol{C} = (\boldsymbol{X}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{X} + \boldsymbol{\Lambda}^{-1})^{-1}. \tag{11}$$

Because of this linearity, and the analyticity of the starry model, inference on data sets comprising thousands of points and spherical harmonic degree $l \leqslant 20$ takes a *fraction of a second* on a typical computer. Full posterior inference with starry can thus be faster than the numerical evaluation of a single forward model (see Figure 5).

It is important to note, however, that the starry model is linear only in the map coefficients $\boldsymbol{y}$. In any real application, there will be uncertainty in the inputs of $\boldsymbol{X}$, such as the orbital parameters, the occultor parameters, the scattering law, etc. These parameters must typically be sampled over, since the model is a nonlinear function of them. However, the analyticity —and in particular, the differentiability—of the starry model makes sampling via gradient-based MCMC easy. Moreover, the linearity of the model with respect to $\boldsymbol{y}$ allows one to efficiently marginalize over those parameters when sampling

[24] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/io_europa.py
[25] https://ssd.jpl.nasa.gov/horizons.cgi

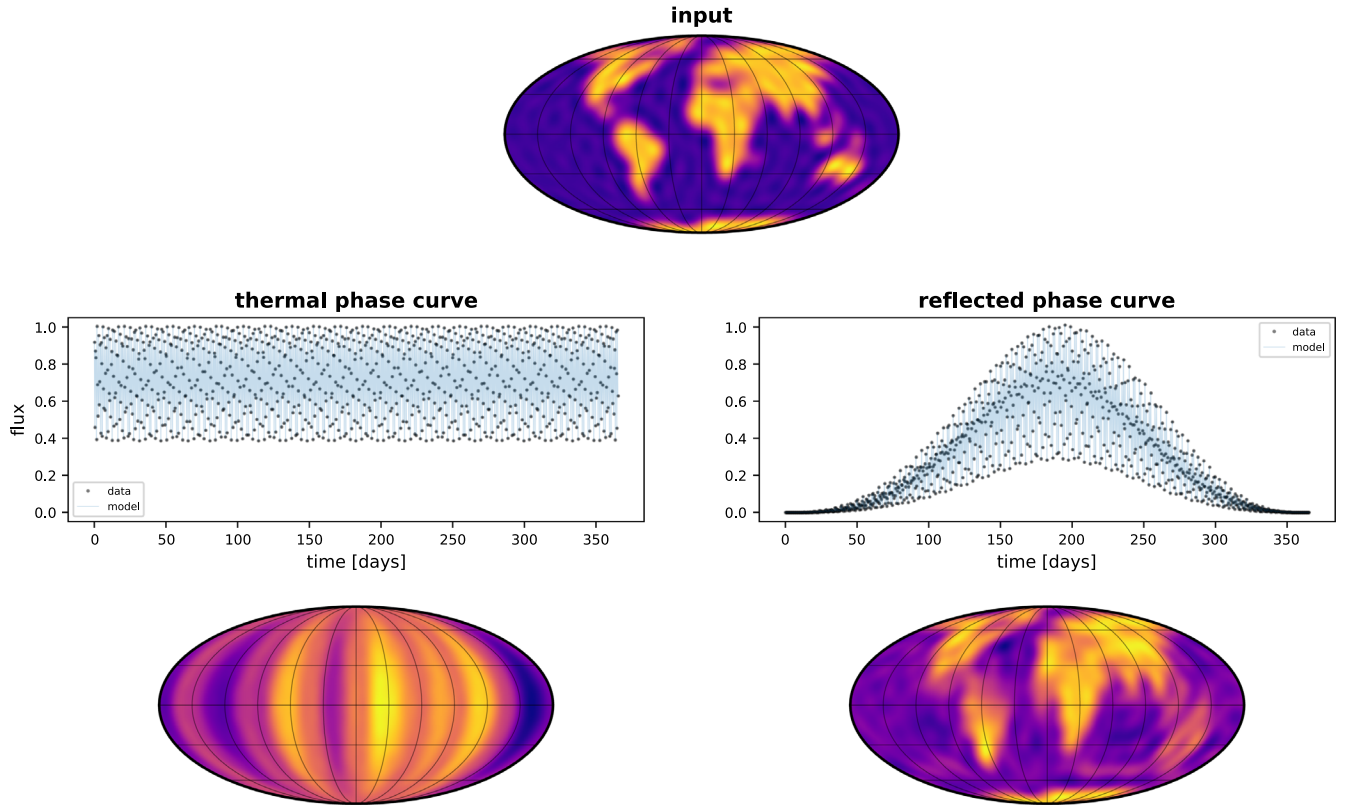

Figure 12. Example of an inference problem for a thermal phase curve (center left) and a reflected phase curve (center right). In both cases, a mock phase curve is generated from an $l = 20$ expansion of the cloudless Earth (top) with 1000 evenly spaced points over the course of one year and an extremely small photometric uncertainty of 1 ppm. The observer sits along the ecliptic and the obliquity of the Earth is set to 23°.5. Data is shown as the black points, and the maximum likelihood starry model is shown in blue. While both models fit the data equally well, the same is not true of the inferred surface maps (bottom row): only in the reflected case are the continental outlines recovered. The thermal phase curve problem is extremely ill-conditioned, but the analogous problem in reflected light is much better posed.[26]

over the nonlinear parameters. Tutorials on how to do this can be found in the starry documentation.[27]

5.2. The Information Content of Reflected Light Curves

One of the fundamental difficulties with the mapping problem is that the process of inferring a two-dimensional map from a light curve is almost always ill-posed. This has been known since at least the turn of the last century, when Russell (1906) discussed how, because of symmetry, all odd harmonics above $l = 1$ are in the null space for thermal phase curves of spherical bodies, meaning those terms do not contribute at all to the disk-integrated flux. As discussed in Luger et al. (2019a, 2021c), the problem is even more ill-posed than that: for any even degree $l > 0$, there are $2l + 1$ modes on the *surface* (one for each value of m), but only 2 Fourier modes in the *light curve* (i.e., a sine and a cosine). Thus, for every mode that can be constrained from the light curve, there are far more modes that cannot, a problem that only gets worse as l increases.

The left panel of Figure 12 shows this issue in practice. We generate a mock thermal light curve (center left) from an $l = 20$ expansion of the Earth (top) with 1000 points over the course of one year with an exquisite photometric precision of 1 ppm. The Earth is given an obliquity of 23°.5 on the plane of the sky but is viewed along the ecliptic, rotating edge-on with an inclination of 90°. The data are shown in black, and in blue is the posterior mean model (Equation (10)), in which we assume a prior variance of 10^{-3} for all spherical harmonic coefficients (and zero covariance). The corresponding surface map is shown at the bottom. As expected, this looks nothing like the true map of the Earth. For a body seen rotating edge-on, the information content of the light curve is strictly longitudinal. While the inferred map captures the average brightness of the Earth at each longitude fairly well, it is missing all latitudinal information. This is independent of the signal-to-noise or the cadence of the data set—it is a fundamental limitation of phase curves in thermal light.[28]

The same is not true for the case of reflected light phase curves. In the right panel of Figure 12 we show the same mock light curve and perform the same inference step, but this time for observations in reflected light; the light curve is similar to that in Figure 1. Because of the presence of a day-night terminator beyond which features on the surface contribute zero flux, none of the symmetry arguments above apply. In particular, the facts that (1) the Earth is seen at different phases and (2) the terminator is inclined relative to the rotational axis mean that the region of the surface contributing to the phase curve is always changing, resulting in a complex light curve that encodes

[26] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/inference.py
[27] https://starry.readthedocs.io
[28] This choice is equivalent to assuming a flat prior on the power spectrum with power 10^{-3} in each degree l of the surface map. This particular value is roughly the average power per degree in the input map, although the results in the figure are not sensitive to this choice. For a detailed discussion of how to set one's priors in a realistic setting in which the true map is not known, see Luger et al. (2021a).

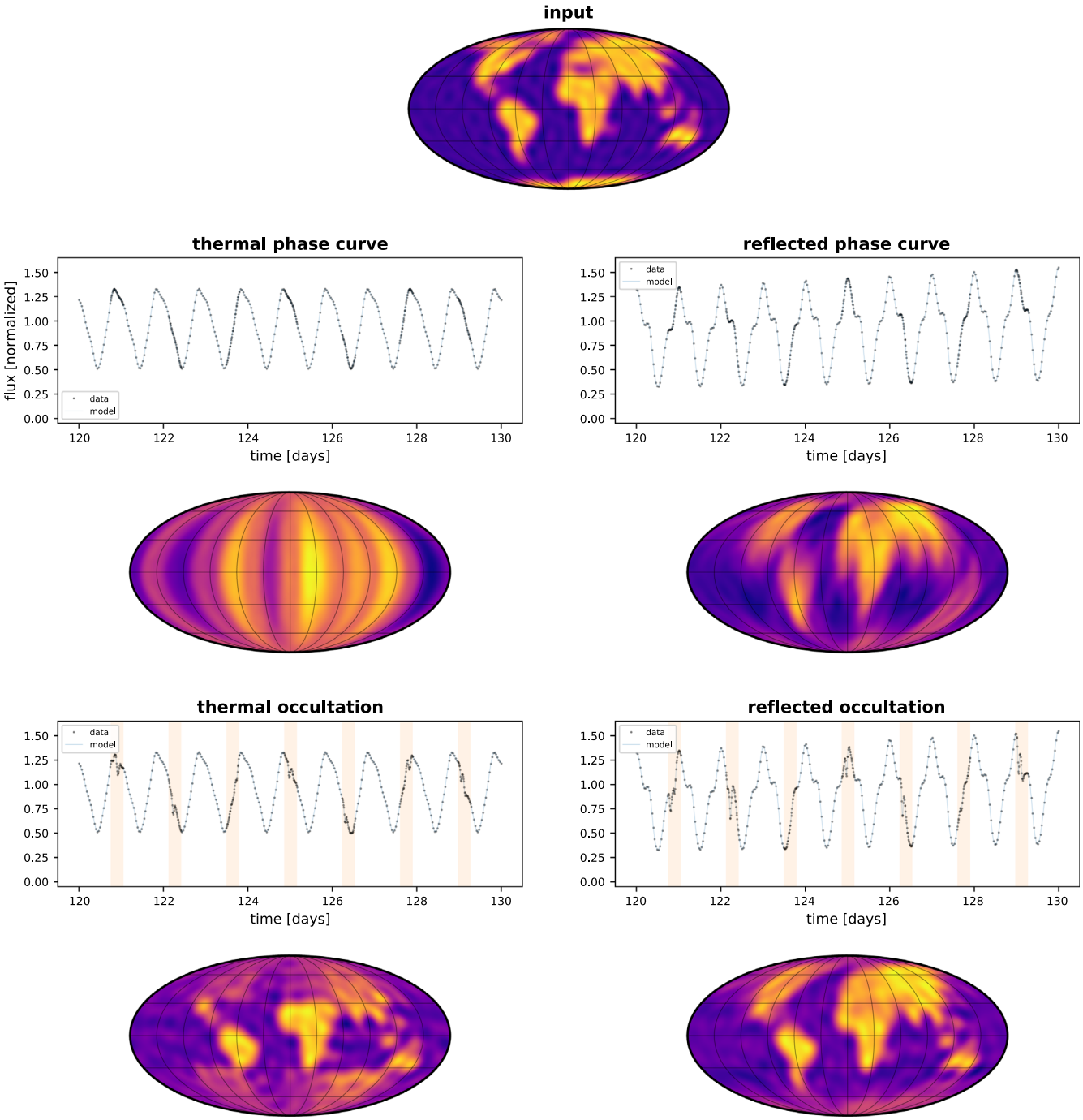

Figure 13. Similar to Figure 12, but for 500 high signal to noise observations taken over ten days near quadrature. At the top we show the thermal and reflected phase curves and the corresponding inferred maps. At the bottom, we show the same light curves but this time including seven equatorial occultations by a (very) short period moon one-quarter the size of the planet. The presence of the occultations significantly increases the fidelity of the recovered maps near the equator (the path of the occultor). While in the thermal case there is significant overfitting at high latitudes, in the reflected case the map accurately recovers features across the entire planet.[29]

significantly more information than its thermal counterpart. The result is an inferred map that is largely faithful to the true map (bottom). While the continental outlines are somewhat fuzzy and some artifacts are present at high latitudes, it is clear that the inference problem is much less ill-posed in this case.

Unlike thermal light curves, reflected light phase curves have the potential to robustly constrain two-dimensional maps of exoplanets. This result is not new, and has been discussed at length in the literature (e.g., Fujii & Kawahara 2012; Berdyugina & Kuhn 2017; Luger et al. 2019b; Aizawa et al. 2020; Kawahara 2020). In particular, Kawahara & Fujii (2010) demonstrated the uniqueness of their inferred map from mock reflected light curves of the Earth. As we argued above, for specific geometrical configurations, the mapping problem in reflected light can actually be well-posed, meaning it has no null space up to a certain degree l. Reflected light phase curves of terrestrial planets with James Webb Space Telescope

[29] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/inference_occ.py

(JWST) and future direct imaging missions thus have the potential to reveal detailed information about their surfaces.

Nevertheless, at finite signal to noise and for limited observation duration or cadence, there may still be significant degeneracies in the reflected phase curve problem. In Luger et al. (2019a), we argued that occultations can be used to break many of these degeneracies, since they directly probe the surface at scales inaccessible to phase curves. The same is true in reflected light. Figure 13 demonstrates this for mock observations of the same Earth-like planet as in Figure 12, but this time taken over ten days near quadrature, when the disk of the planet is seen at half phase. The top rows show the thermal and reflected phase curves and the inferred maps, which look similar to those in the previous figure. At the bottom we show the same light curve, but this time including seven equatorial occultations by a moon one-quarter the size of the planet. The moon's period and occultation duration are unrealistically short, but the inferred maps at the bottom show the exquisite constraining power of these occultations. In the thermal case, the presence of the moon allows us to infer the two-dimensional distribution of surface features along its equatorial occultation path; however, there is little information in the light curve about features at higher latitudes, and the starry model overfits. Conversely, in the reflected light case, there is information about all latitudes and longitudes, and the inferred map is largely faithful to the true map.

It should be kept in mind that the kinds of observations mentioned above will be very challenging for exoplanets, even with next-generation observatories such as HabEx or LUVOIR. Observations of occultations of planets by moons, in particular, are likely several decades away at least. There is some hope that planet–planet occultations may be detectable in the near future for specific planetary systems such as TRAPPIST-1 (Luger et al. 2019b), but at extremely limited signal-to-noise. Even phase curve observations will be difficult because of their limited signal-to-noise, and in practice many degeneracies will likely remain. Several studies have found that color information can greatly help in the interpretation of reflected phase curves (Cowan et al. 2009; Kawahara & Fujii 2011; Lustig-Yaeger et al. 2018; Kawahara 2020), while others have explored in detail the best kinds of priors to assume (e.g., Aizawa et al. 2020; Asensio Ramos & Pallé 2021). Future maps of exoplanets—particularly terrestrial ones in the habitable zone—will require every tool in the toolbox.

5.3. Limitations

There are a few limitations to the starry model that are important to bear in mind. The primary limitation concerns the maximum spherical harmonic degree of the model. While the expressions derived here are valid at arbitrary degree, we find that their numerical stability quickly degrades above $l \sim 20$–25 for occultations and $l \sim 35$–40 for phase curves. The same is true for the model in thermal light (Luger et al. 2019a), and is due to (1) the large condition number of the change-of-basis matrix from spherical harmonics to polynomials and (2) instabilities in the many recursion relations used to evaluate the solution vectors $\mathbb{s}^\top$ and $\mathbb{r}^\top$. In principle, one could improve the numerical stability by evaluating all expressions at higher floating-point precision, but in practice the computational cost of this becomes quickly prohibitive. However, it is important to keep in mind that our current best image of an exoplanet is the $l = 1$ map of HD189733b (Knutson et al. 2007; de Wit et al. 2012; Majeau et al. 2012). Next-generation facilities such as JWST may allow us to probe surface modes as small as $l = 5$ for some planets (R. Luger et al. 2022, in preparation), but even with future telescopes such as LUVOIR, it is extremely unlikely we will do better than $l = 20$. If cases arise requiring a resolution smaller than about $180°/20 = 9°$ on the surface, the starry algorithm will have to be revisited.

The second limitation concerns the flexibility of the starry model. While we presented ways to capture non-Lambertian scattering in starry (Section 4.2), there are certain aspects of light curves in reflected light that cannot be captured by the model. One example of this is shadowing. Craters on the moon or volcanoes on Io can cast large shadows visible from space, particularly if viewed near crescent phase. Unfortunately, there is no way to model this within the starry framework. Another example is multiple scattering, as in optically thick atmospheres, for which a proper radiative transfer model must be used. There is also the case of specular reflection, or "glint," which is a pronounced signal for the Earth due its oceans (Robinson et al. 2014) and on Titan due to its hydrocarbon lakes (Barnes et al. 2011). In principle, glint could be modeled in the same way as non-Lambertian scattering, by constructing the linear operator $\boldsymbol{I}$ in such a way as to downweight portions of the projected disk where $\vartheta_i \neq \vartheta_r$. In practice, however, if the size of the glint spot is small (which is typically the case), an expansion at extremely high l ($l \sim 360$ in the case of the Earth) would be required, which would not work for the reasons above. Instead, it may be possible to combine the starry algorithm with the formalism of Haggard & Cowan (2018), who derived analytic expressions for phase curves of delta function maps (δ-map), to model glint. As discussed in Lustig-Yaeger et al. (2018), glint mapping is an extremely powerful way to not only map terrestrial planets but also to confirm their habitability via the presence of an ocean. Lastly, the presence of time-variable features, such as clouds, dust storms, or seasonal variations of vegetation are not accounted for in the model, although the documentation discusses how one may approach the modeling of temporal features.[30]

We would also like to emphasize that while spherical harmonics are a convenient basis for the purpose of computing light curves, they have a significant drawback when it comes to modeling real planetary surfaces: it can be difficult to strictly enforce physical values of the albedo everywhere on the surface map when doing inference. That is because there is no analytic way to determine whether a spherical harmonic representation is positive-valued (or restricted to a given range) everywhere on the sphere. Instead, this must be checked numerically, by evaluating the function on a discrete grid. This makes it somewhat cumbersome to implement positivity as a prior when doing inference; in particular, this prior cannot be expressed as a Gaussian, so the analytic expression for the posterior discussed in Section 5.1 will generally have nonzero support for negative albedo values. This is particularly problematic when the data is not very constraining and the posterior is prior-dominated, as positivity can be an *extremely* informative prior (e.g., Fienup 1978). We therefore recommend that in such cases users of the starry algorithm use HMC/MCMC to do inference, either (1) sampling over the spherical harmonic coefficients $\boldsymbol{y}$ and imposing a uniform prior in the range [0, 1] on the albedo values $\boldsymbol{a}$ evaluated on a discrete grid

[30] https://starry.readthedocs.io

on the sphere or (2) sampling over the albedo values $\boldsymbol{a}$ with the same uniform prior, but using $\boldsymbol{y}$ to compute the actual light curve model (Bartolić et al. 2022). In both cases, there exists a linear operator that transforms between $\boldsymbol{y}$ and $\boldsymbol{a}$:

$$\boldsymbol{a} = \boldsymbol{P}\boldsymbol{y} \tag{12}$$

and

$$\boldsymbol{y} = \boldsymbol{P}^{+}\boldsymbol{a}, \tag{13}$$

where

$$\boldsymbol{P}^{+} = (\boldsymbol{P}^{\top}\boldsymbol{P} + \lambda \mathbf{I})^{-1}\boldsymbol{P}^{\top} \tag{14}$$

is the pseudoinverse of $\boldsymbol{P}$, with (small) regularization parameter λ and where $\mathbf{I}$ is the identity matrix. Each row of the matrix $\boldsymbol{P}$ is constructed from the value of each of the spherical harmonics at the corresponding point on the grid; both $\boldsymbol{P}$ and its inverse may be precomputed for efficiency. In both cases, the grid should be fine enough to ensure positivity over most of the sphere but not so fine as to throttle the computation; as a rule of thumb, we find that grids with ∼4 times as many pixels as spherical harmonic coefficients are sufficient. The documentation includes tutorials on how to implement this in practice.[31]

Finally, while the algorithm presented here can be used to model planet–planet and planet–moon occultations in reflected light, we only account for the physical blocking of light rays from the planet by the occultor; we do not account for the attenuation of the reflected light due to the *shadow* of the occultor on the planet. This can be an important effect at certain orbital phases (e.g., Cabrera & Schneider 2007). However, since the shadow of the occultor is not necessarily circular, it is significantly harder to model in closed form. A proper treatment of this and other effects, such as occultations and shadows due to rings (e.g., Arnold & Schneider 2004), is deferred to future work.

6. Conclusions

We have presented an efficient, numerically stable, closed-form algorithm for computing phase curves and occultation light curves of spherical bodies in reflected (scattered) light. This algorithm is an extension of the algorithm presented in Luger et al. (2019a) for light curves in thermal light and is generally applicable to exoplanetary phase curves, secondary eclipses, and occultations by moons and other planets, as well as to light curves of planets and moons in our solar system. We derive the solution for the case of a Lambert sphere illuminated by a point source, but extend it to the case of an extended illumination source and non-Lambertian scattering parametrized by a surface roughness coefficient. The algorithm is ∼4–5 orders of magnitude faster and ∼10 orders of magnitude more precise than other numerical approaches for computing these light curves. The algorithm is also differentiable in all of the model parameters, enabling inference with efficient gradient-based samplers such as HMC, and linear in the spherical harmonic coefficients describing the surface albedo, enabling fast, closed-form solutions for the albedo posterior distribution under a Gaussian noise model. We implement the algorithm within the starry software, an open-source Python package for inferring surface maps of unresolved celestial bodies. The algorithm is coded in a combination of C++ and Python compiled using the theano package (Theano Development Team 2016). The interface was designed specifically for compatibility with the exoplanet modeling package (Foreman-Mackey et al. 2020) and the pymc3 inference suite (Salvatier et al. 2016).

Upcoming telescopes will enable measurements of exoplanet phase curves and secondary eclipses at unprecendented precision. While the JWST will be primarily sensitive to thermal emission from exoplanets (which can currently be modeled with starry), next-generation direct imaging facilities such as the LUVOIR will enable measurements in reflected light, in particular for terrestrial planets in the habitable zone. Because of the changing illumination pattern over the course of an orbit of the planet, phase curves and occultation light curves in reflected light contain vastly more information about the two-dimension albedo distribution of the body than their thermal counterparts. With careful modeling, light curves in reflected light are likely to give us the first images of potentially habitable exoplanets, enabling the detection of clouds, continents, oceans, and perhaps even life.

We would like to thank Nicolas Cowan, Christina Hedges, and the Astronomical Data Group at the Center for Computational Astrophysics for many thought-provoking discussions that helped shape this paper. R.L. is supported by a Flatiron Fellowship at the Flatiron Institute, a division of the Simons Foundation.

The software presented in this work is open source under the MIT License and is available at https://github.com/rodluger/starry, with documentation and tutorials hosted at https://starry.readthedocs.io. The code used to generate the figures in this paper is hosted at https://github.com/rodluger/starrynight.

Software: astroquery (Ginsburg et al. 2013, 2019), Eigen v3 (Guennebaud & Jacob 2010), exoplanet (Foreman-Mackey et al. 2020), pybind11 (Jakob et al. 2017), pymc3 (Salvatier et al. 2016), scipy (Jones et al. 2001), starry (Luger et al. 2018), theano (Theano Development Team 2016).

Appendix A
The Problem

This paper closely follows the notation and formalism introduced in Luger et al. (2019a). While we include all of the relevant equations and definitions below, the reader is encouraged to review Luger et al. (2019a) before proceeding. To improve the readability of this paper, Tables 1–4 at the end list the principal symbols and quantities used throughout the text, with links to the sections and equations in which they are defined. Because of the large number of symbols used in this paper, we adopt the following conventions: scalars are represented by regular lowercase or occasionally uppercase letters (i.e., x or X), vectors are represented by boldface lowercase letters ($\boldsymbol{x}$), and matrices and other linear operators are represented by boldface capital letters ($\boldsymbol{X}$). With a few exceptions, Greek letters are reserved for angular quantities and may be either scalars (α) or vectors ($\boldsymbol{\alpha}$). Script font is typically used to denote curves or frames of reference ($\mathcal{F}$). Primes are used to distinguish between frames of reference (x and x' are used to denote the same quantity, but in frames $\mathcal{F}$ and $\mathcal{F}'$, respectively). Tildes are used to denote basis vectors ($\tilde{\boldsymbol{y}}$). Finally, blackboard vectors ($\mathbb{x}$) correspond to solutions to the various "primitive" integrals that arise in the occultation problem.

A.1. Review of the starry Algorithm in Emitted Light

Without loss of generality, assume the body whose flux we wish to compute has radius unity and sits at the origin of a

[31] https://starry.readthedocs.io

right-handed Cartesian coordinate system in some frame $\mathcal{F}_0$. In this frame, the surface (emitted) intensity field of the body is described by a vector $\boldsymbol{y}$ of coefficients in the spherical harmonic basis $\tilde{\boldsymbol{y}}$:

$$\tilde{\boldsymbol{y}}(x, y) = \left(Y_{0,0} \;\; Y_{1,-1} \;\; Y_{1,0} \;\; Y_{1,1} \;\; Y_{2,-2} \;\; Y_{2,-1} \;\; Y_{2,0} \;\; Y_{2,1} \;\; Y_{2,2} \;\; \cdots\right)^{\top}, \tag{A1}$$

where the component at index n is the spherical harmonic $Y_{l,m}(x, y)$ with

$$\begin{aligned} l &= \lfloor \sqrt{n} \rfloor \\ m &= n - l^2 - l. \end{aligned} \tag{A2}$$

The spherical harmonics are traditionally expressed in spherical coordinates, but for our purposes it is more convenient to express them in Cartesian coordinates on the sky-projected disk, in which case they are simply polynomials in x, y, and z (see Appendix A in Luger et al. 2019a).

An observer views the body from a large distance in the sky frame $\mathcal{F}$, in which the x-axis points to the right, the y-axis points up, and the z-axis points out of the sky toward the observer. Following Luger et al. (2019a), if an occultor of radius r_o is located at sky position (x_o, y_o), we compute the visible thermal flux f_T from

$$f_T = \boldsymbol{s}^{\top}\boldsymbol{A}\boldsymbol{R}'\boldsymbol{R}\boldsymbol{y}, \tag{A3}$$

where, from right to left, $\boldsymbol{R} = \boldsymbol{R}(I, \Lambda, \Theta)$ is a Wigner rotation matrix that rotates $\boldsymbol{y}$ from $\mathcal{F}_0$ to the sky frame $\mathcal{F}$ given the body's inclination I, obliquity Λ, and rotational phase Θ (Appendix C in Luger et al. 2019a), $\boldsymbol{R}' = \boldsymbol{R}'(x_o, y_o)$ rotates the body on the plane of the sky into the integration frame $\mathcal{F}'$, in which the occultor lies along the $+y'$-axis, $\boldsymbol{A}$ (Equation (B13) in Luger et al. 2019a) is the change-of-basis matrix from $\tilde{\boldsymbol{y}}$ to the *Green's basis* $\tilde{\boldsymbol{g}}$ in which the integrals are computed, whose component at index n is

$$g_n(x, y) = \begin{cases} \dfrac{\mu+2}{2} x^{\frac{\mu}{2}} y^{\frac{\nu}{2}} & \mu, \nu \text{ even} \\ z(x, y) & \mu = \nu = 1 \\ 3x^{l-2} y z(x, y) & \nu \text{ odd}, \mu = 1, \dfrac{\mu+\nu}{2} \text{ even} \\ z(x, y)(-x^{l-3} + x^{l-1} + 4x^{l-3}y^2) & \nu \text{ odd}, \mu = 1, \text{ odd} \\ z(x, y)\left(\dfrac{\mu-3}{2} x^{\frac{\mu-5}{2}} y^{\frac{\nu-1}{2}} - \dfrac{\mu-3}{2} x^{\frac{\mu-5}{2}} y^{\frac{\nu+3}{2}} - \dfrac{\mu+3}{2} x^{\frac{\mu-1}{2}} y^{\frac{\nu-1}{2}}\right) & \text{otherwise,} \end{cases} \tag{A4}$$

with

$$\begin{aligned} \mu &\equiv l - m \\ \nu &\equiv l + m \end{aligned} \tag{A5}$$

and

$$z(x, y) \equiv \sqrt{1 - x^2 - y^2}, \tag{A6}$$

and $\boldsymbol{s}^{\top} = \boldsymbol{s}^{\top}(b_o, r_o)$ is the vector of solutions to the integral over the projected visible disk of the body for each term in $\tilde{\boldsymbol{g}}$ (Equation (26) in Luger et al. 2019a), with $b_o = \sqrt{x_o^2 + y_o^2}$.

If instead no occultor is present, we compute the total visible thermal flux f_{T_0} from this body as

$$f_{T_0} = \boldsymbol{r}^{\top}\boldsymbol{A}_1\boldsymbol{R}''\boldsymbol{R}\boldsymbol{y}, \tag{A7}$$

where, as before, $\boldsymbol{R} = \boldsymbol{R}(I, \Lambda, \Theta)$ rotates the body from $\mathcal{F}_0$ to the sky frame $\mathcal{F}$, $\boldsymbol{R}''$ rotates the body on the plane of the sky into the integration frame $\mathcal{F}''$, $\boldsymbol{A}_1$ (Equation (B11) in Luger et al. 2019a) is the change-of-basis matrix from the spherical harmonic basis $\tilde{\boldsymbol{y}}$ to the *polynomial basis* $\tilde{\boldsymbol{p}}$ in which the integrals are computed, whose component at index n is

$$\tilde{p}_n(x, y) = \begin{cases} x^{\frac{\mu}{2}} y^{\frac{\nu}{2}} & \mu, \nu \text{ even} \\ x^{\frac{\mu-1}{2}} y^{\frac{\nu-1}{2}} z(x, y) & \text{otherwise,} \end{cases} \tag{A8}$$

and $\boldsymbol{r}^{\top}$ is the vector of solutions to the integral over the projected visible disk of the body for each term in $\tilde{\boldsymbol{p}}$ (Equation (19) in Luger et al. 2019a).[32]

A.2. Adapting the Algorithm to the Reflected Light Case

In order to compute light curves in reflected light, we must make two modifications to the starry algorithm. First, the expressions above assume that the coefficient vector $\boldsymbol{y}$ describes the *emissivity* of the body, which (in the absence of limb darkening) is assumed to be Lambertian, i.e., all points on the surface emit equally in all directions. Here, we wish to derive the solution for the flux in the case of Lambertian reflectance, in which case the vector $\boldsymbol{y}$ is taken to describe the spherical albedo of the surface, A.

Second, we must explicitly model the illumination of the body. We assume the body is illuminated by a point-like source whose flux measured by the observer is unity. In this case, the observed intensity at any point on the surface is proportional to the cosine of the angle ϑ_i between the incident light and the surface normal. Points for which $\vartheta_i \geqslant \pi/2$ are unilluminated and therefore have an intensity of zero. If the point-like illumination source is placed at sky coordinates (x_s, y_s, z_s) in units of the radius of the illuminated body, the day/night terminator on the body is a half-ellipse of semimajor axis unity

[32] In Luger et al. (2019a), $\mathcal{F}'' = \mathcal{F}'$, so this rotation is trivial: $\boldsymbol{R}''$ is just the identity matrix.

that is fully described by its (signed) semiminor axis,

$$b = -\frac{z_s}{r_s}, \tag{A9}$$

where $r_s = \sqrt{x_s^2 + y_s^2 + z_s^2}$ is the distance to the source, and the angle by which its semimajor axis is rotated away from the $+x$-axis,

$$\theta = -\arctan 2(x_s, y_s), \tag{A10}$$

where $\arctan 2(a, b)$ is the quadrant-aware arctangent of a/b. Given this formulation, and assuming that $r_s \gg 1$, it is straightforward to show that the illumination $\mathcal{I}$ at a point (x, y) on the projected disk of the body is given by the function

$$\mathcal{I}(b, \theta, r_s; x, y) = \max(0, I(b, \theta, r_s; x, y)), \tag{A11}$$

where

$$\begin{aligned} I(b, \theta, r_s; x, y) &= \frac{1}{\pi r_s^2} \cos \vartheta_i \\ &= \frac{1}{\pi r_s^2}(-b_c \sin\theta x + b_c \cos\theta y - bz(x, y)), \end{aligned} \tag{A12}$$

with $b_c \equiv \sqrt{1 - b^2}$ and $z(x, y) = \sqrt{1 - x^2 - y^2}$. The illumination $\mathcal{I}$ is a unitless quantity, normalized such that the integral of $A\mathcal{I}$ over the unit disk is equal to the flux measured by the observer as a fraction of the flux of the illumination source.[33] In particular, if we place the illumination source along the $+z$-axis at $(0, 0, 1)$, the body is seen at full phase, so $b = -1$, $b_c = 0$, and

$$\mathcal{I}_{\text{full}}(x, y) = \frac{\sqrt{1 - x^2 - y^2}}{\pi}. \tag{A13}$$

Multiplying this by the albedo and integrating over the unit disk, we obtain the reflected flux measured by the observer in units of the flux of the illumination source:

$$\begin{aligned} f_{\text{full}} &= \int_{-1}^{1} \int_{-\sqrt{1-x^2}}^{\sqrt{1-x^2}} A \frac{\sqrt{1 - x^2 - y^2}}{\pi} \, dy \, dx \\ &= \frac{2}{3} A, \end{aligned} \tag{A14}$$

which is precisely the geometric albedo of a Lambert sphere of spherical albedo A (see, e.g., Seager 2010).[33,34,35,36,37,38]

In principle, our task is now straightforward: weight each of the terms in the Green's basis (Equation (A4)) and integrate them over the visible portion of the body's disk to obtain the reflected light solution vector, $\mathbb{s}^\top$. Unfortunately, the piecewise nature of Equation (A11) makes direct evaluation of these integrals extremely difficult in practice. We find that it is more tractable to weight our basis terms by the function I (Equation (A12)) and to modify the limits of integration to exclude the nightside of the body, where I is (unphysically) negative. In particular, since I is just a polynomial in x, y, and z (x, y), we can express it as a vector $\boldsymbol{i}(b, \theta)$ in the polynomial basis $\tilde{\boldsymbol{p}}$. Recalling the structure of the basis (Equation (A8)), we may write

$$\boldsymbol{i}(b, \theta, r_s) = \frac{1}{\pi r_s^2} \begin{pmatrix} 0 \\ -b_c \sin\theta \\ -b \\ b_c \cos\theta \end{pmatrix}. \tag{A15}$$

This fact allows us to construct a linear operator $\boldsymbol{I}$ to weight a map vector in the polynomial basis by the illumination profile. If we think about how each of the terms in $\tilde{\boldsymbol{p}}$ transforms under $\boldsymbol{I}$,

$$\begin{aligned} &\begin{pmatrix}1\\0\\0\\0\end{pmatrix} \to \begin{pmatrix}\boldsymbol{i}_0\\\boldsymbol{i}_1\\\boldsymbol{i}_2\\\boldsymbol{i}_3\\0\\0\\0\\0\end{pmatrix} \quad \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\\boldsymbol{i}_0\\0\\0\\\boldsymbol{i}_1\\\boldsymbol{i}_2\\\boldsymbol{i}_3\\0\end{pmatrix} \\ &\begin{pmatrix}0\\0\\1\\0\end{pmatrix} \to \begin{pmatrix}\boldsymbol{i}_2\\0\\\boldsymbol{i}_0\\0\\-\boldsymbol{i}_2\\\boldsymbol{i}_1\\0\\\boldsymbol{i}_3\end{pmatrix} \quad \begin{pmatrix}0\\0\\0\\1\end{pmatrix} \to \begin{pmatrix}0\\0\\0\\\boldsymbol{i}_0\\0\\0\\\boldsymbol{i}_1\\\boldsymbol{i}_2\end{pmatrix}, \end{aligned} \tag{A16}$$

we can compose $\boldsymbol{I}$ out of these column vectors:

$$\boldsymbol{I}(b, \theta, r_s) = \frac{1}{\pi r_s^2} \begin{pmatrix} 0 & 0 & -b & 0 & \cdots \\ -b_c \sin\theta & 0 & 0 & 0 & \cdots \\ -b & 0 & 0 & 0 & \cdots \\ b_c \cos\theta & 0 & 0 & 0 & \cdots \\ 0 & -b_c \sin\theta & b & 0 & \cdots \\ 0 & -b & -b_c \sin\theta & 0 & \cdots \\ 0 & b_c \cos\theta & 0 & -b_c \sin\theta & \cdots \\ 0 & 0 & b_c \cos\theta & -b & \cdots \\ 0 & 0 & b & b_c \cos\theta & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{A17}$$

where the dimensions of the matrix are $((l + 2)^2, (l + 1)^2)$, where l is the spherical harmonic degree of the map (this operator raises the degree of the map by one). Note, again, that this weighting is valid only on the dayside hemisphere (see Equation (A11)), as the operator $\boldsymbol{I}$ weights points on the

[33] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[34] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[35] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[36] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[37] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[38] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb

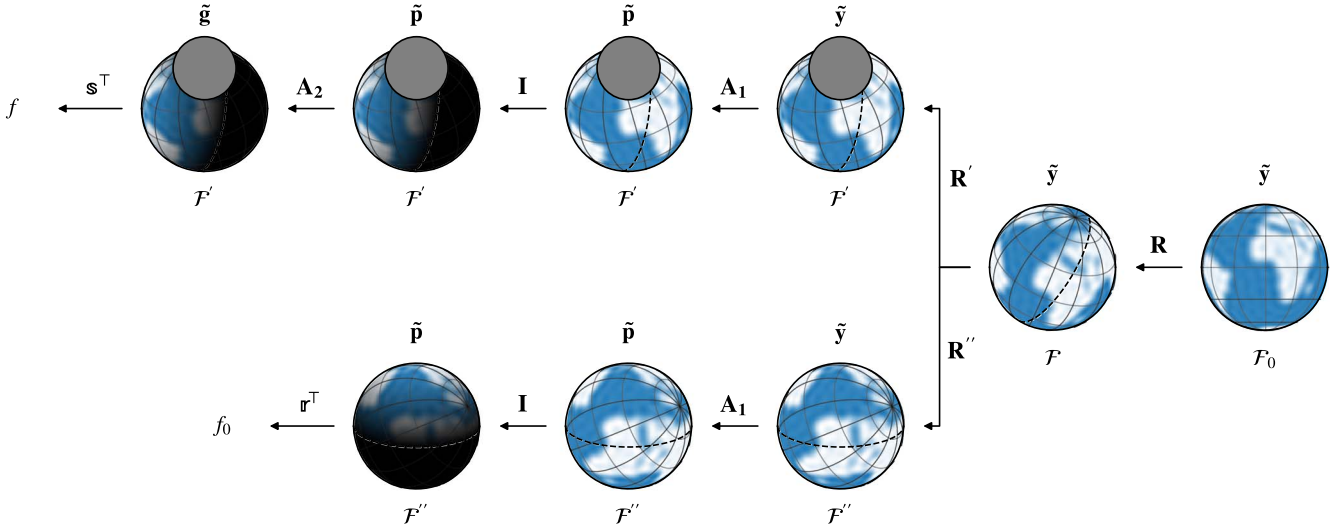

Figure 14. How starry computes the flux from a body in reflected light, tracking each of the linear transformations from the input map (far right) to the output (far left). The label below each map denotes the reference frame, while the label above each map denotes the basis in which the map is represented. Arrows indicate linear operations and are labeled accordingly. The upper branch corresponds to the occulted case (Equation (A18)), while the lower branch corresponds to the case where the body is unocculted (Equation (A20)). See text for details.[39]

nightside by a *negative* amount, which is clearly unphysical. As we will see momentarily, we account for this by excluding the nightside from the integration region in our flux integrals.[40,41]

We may now re-write Equations (A3) and (A7) to account for this illumination transformation. The flux during an occultation is now given by

$$f = \mathbb{s}^\top(b, \theta', b_o, r_o)\boldsymbol{A_2}\boldsymbol{I}(b, \theta', r_s)\boldsymbol{A_1}\boldsymbol{R}'(x_o, y_o)\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}, \tag{A18}$$

where

$$\theta' = \arctan 2(x_o, y_o) - \arctan 2(x_s, y_s) \tag{A19}$$

is the angle of the terminator in the frame $\mathcal{F}'$. Note that we made use of the fact that $\boldsymbol{A} = \boldsymbol{A_2}\boldsymbol{A_1}$ (Equation (14) in Luger et al. 2019a), where $\boldsymbol{A_1}$ transforms from the spherical harmonic basis $\tilde{\boldsymbol{y}}$ to the polynomial basis $\tilde{\boldsymbol{p}}$, and $\boldsymbol{A_2}$ transforms from $\tilde{\boldsymbol{p}}$ to the Green's basis $\tilde{\boldsymbol{g}}$.

Similarly, the flux when there is no occultation is now given by

$$f_0 = \mathbb{r}^\top(b)\boldsymbol{I}(b, \theta'', r_s)\boldsymbol{A_1}\boldsymbol{R}''(x_s, y_s)\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}, \tag{A20}$$

where

$$\theta'' = 0 \tag{A21}$$

is the angle of the terminator in the frame $\mathcal{F}''$, by construction. The transformation $\boldsymbol{R}'' = \boldsymbol{R}''(x_s, y_s)$ rotates the body through an angle $\arctan 2(x_s, y_s)$ so the semimajor axis of the terminator is aligned with the x''-axis; as will become clear in Appendix B below, this greatly simplifies the integration step.

Note that in both equations we replaced the integral vectors $\boldsymbol{r}^\top$ and $\boldsymbol{s}^\top(b_o, r_o)$ with the vectors $\mathbb{r}^\top(b)$ and $\mathbb{s}^\top(b, \theta', b_o, r_o)$, respectively. As we mentioned above, we must modify the integration limits to exclude the nightside, where the weighting by $\boldsymbol{I}$ is unphysical. The vectors $\mathbb{r}^\top$ and $\mathbb{s}^\top$ correspond to these modified integrals, which we devote the rest of this paper to computing.

Figure 14 summarizes the transformations involved in the two equations above. Starting on the right with a map vector $\boldsymbol{y}$ in the spherical harmonic basis $\tilde{\boldsymbol{y}}$, defined in some observer-independent frame $\mathcal{F}_0$, we first rotate it via $\boldsymbol{R}$ to the sky frame $\mathcal{F}$, in which the body is viewed by the observer. If an occultor is present (upper branch of the figure), we rotate the map from $\mathcal{F}$ via $\boldsymbol{R}'$ to the frame $\mathcal{F}'$, in which the occultor lies along the $+y'$-axis. We then apply $\boldsymbol{A_1}$ to change basis to $\tilde{\boldsymbol{p}}$ and $\boldsymbol{I}$ to weight the map by the illumination. Finally, we change basis via $\boldsymbol{A_2}$ to the Green's basis, in which we compute and dot the integrals $\mathbb{s}^\top$. If, on the other hand, there is no occultation (lower branch of the figure), we instead rotate the map via $\boldsymbol{R}''$ to the integration frame $\mathcal{F}''$, in which the terminator is parallel to the x''-axis. We then apply $\boldsymbol{A_1}$ to change basis to $\tilde{\boldsymbol{p}}$, apply the illumination transform $\boldsymbol{I}$, and finally dot in the solutions to the surface integrals $\mathbb{r}^\top$.

Appendix B
The Solution: No Occultation

Before we tackle configurations involving occultations, we must address the simpler problem of computing the total visible flux from an unocculted body in reflected light (Equation (A20)). This problem was originally solved by Haggard & Cowan (2018) and subsequently by Luger et al. (2019b), but for completeness we present the detailed derivation in the starry formalism here.

As we discussed above, we perform the integration in a frame $\mathcal{F}''$ in which the semimajor axis of the terminator is aligned with the x''-axis, with the illumination source at $y'' \geqslant 0$.

[39] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/frames.py
[40] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb
[41] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/illumination.ipynb

The solution vector may then be computed from

$$\mathbb{r}^\top(b) = \int_{-1}^{1}\int_{b\sqrt{1-x''^2}}^{\sqrt{1-x''^2}} \tilde{\boldsymbol{p}}(x'', y'')\, dy''\, dx'', \tag{B1}$$

which is identical to Equation (20) in Luger et al. (2019a) except for the lower integration limit of the inner integral. The lower limit is now the equation describing the terminator, which ensures we always exclude the nightside from the integration region. Equation (B1) may be solved analytically in terms of purely trigonometric and algebraic functions of b. The component of $\mathbb{r}^\top$ at index n is given by

$$\mathbb{r}_n(b) = \begin{cases} \dfrac{(1-b^{\frac{\nu+2}{2}})}{2}\mathbb{L}_{\frac{\mu}{2},\frac{\nu}{2}} & \dfrac{\mu}{2} \text{ even} \\ \mathbb{k}_{\frac{\nu-1}{2}}(b)\mathbb{M}_{\frac{\mu-1}{2},\frac{\nu-1}{2}} & \dfrac{\mu-1}{2} \text{ even} \\ 0 & \text{otherwise} \end{cases}, \tag{B2}$$

where the components of $\mathbb{k}$, $\mathbb{L}$, and $\mathbb{M}$ are given by

$$\begin{aligned} \mathbb{k}_j(b) &= \int_b^1 a^j\sqrt{1-a^2}\, da \\ \mathbb{L}_{i,j} &= \frac{\Gamma\left(\frac{i+1}{2}\right)\Gamma\left(\frac{j+1}{2}\right)}{\Gamma\left(\frac{i+j+4}{2}\right)} \\ \mathbb{M}_{i,j} &= \frac{\Gamma\left(\frac{i+1}{2}\right)\Gamma\left(\frac{j+4}{2}\right)}{\Gamma\left(\frac{i+j+5}{2}\right)}, \end{aligned} \tag{B3}$$

where Γ is the gamma function. Given initial conditions

$$\begin{aligned} \mathbb{k}_0(b) &= \frac{\arccos(b) - bb_c}{2} & \mathbb{L}_{0,0} &= \pi & \mathbb{M}_{0,0} &= \frac{4}{3} \\ \mathbb{k}_1(b) &= \frac{b_c^3}{3} & \mathbb{L}_{0,1} &= \frac{4}{3} & \mathbb{M}_{0,1} &= \frac{3\pi}{8}, \end{aligned} \tag{B4}$$

we may compute all the required higher order terms from the recurrence relations

$$\begin{aligned} \mathbb{k}_j(b) &= \frac{b^{j-1}b_c^3 + (j-1)\mathbb{k}_{j-2}(b)}{j+2} \\ \mathbb{L}_{0,j} &= \left(\frac{j-1}{j+2}\right)\mathbb{L}_{0,j-2} \\ \mathbb{M}_{0,j} &= \left(\frac{j+2}{j+3}\right)\mathbb{M}_{0,j-2} \\ \mathbb{L}_{i,j} &= \left(\frac{i-1}{i+j+2}\right)\mathbb{L}_{i-2,j} \\ \mathbb{M}_{i,j} &= \left(\frac{i-1}{i+j+3}\right)\mathbb{M}_{i-2,j}. \end{aligned} \tag{B5}$$

Once $\mathbb{r}^\top$ is known, the observed total flux in reflected light is computed from (see Equation (A20))

$$f_0 = \mathbb{r}^\top(b)\boldsymbol{I}(b, \theta'', r_s)\boldsymbol{A_1}\boldsymbol{R}''(x_s, y_s)\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}. \tag{B6}$$

Finally, for future reference, we can also compute what we will call the *complement* of case 0:

$$\hat{f}_0 = (\boldsymbol{r}^\top - \mathbb{r}^\top(b))\boldsymbol{I}(b, \theta'', r_s)\boldsymbol{A_1}\boldsymbol{R}''(x_s, y_s)\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}. \tag{B7}$$

This is the flux contribution from the unphysical nightside (if we were to integrate over it), where our polynomial illumination (Equation (A12)) function yields *negative* intensities. This quantity will be useful in negating the unphysical contribution in the integrals of the following section.[42,43,44,45]

Appendix C
The Solution: Occultation

The integration in the unocculted case presented above is relatively straightforward, since the boundaries of integration are always the half-ellipse defining the terminator and the half-circle defining the upper limb of the body (Equation (B1)). When an occultor is present, however, the integration boundaries are far less trivial, since they may or may not include sections of the terminator, sections of the limb of the body, and sections of the limb of the occultor. The integration regions may also be disjoint; for instance, in case 7 of Figure 15, the portion of the dayside that is unocculted consists of two separate regions.

Whereas in Luger et al. (2019a) we compute the observed flux by always integrating over the unocculted portion of the disk, here we find that it is often easier and more computationally efficient to compute the integral of the intensity over the *simplest* region—meaning the one with the fewest boundaries—and combine it with the formalism from Appendix B to compute the visible flux. These integrals may be over the unocculted dayside, the occulted dayside, the unocculted nightside, or the occulted nightside; in the case of the latter three, a bit of algebra (Appendices C.3–C.4) is needed to relate these to the observed flux. Additionally, in some cases we can avoid computing new integrals entirely, as the solution can be obtained from a combination of the classical starry solution vector $\mathbb{s}^\top$ and the formalism from the unocculted case (Appendix B).

After exhaustive experimentation, we identified in total 14 families of geometrical configurations for the occultation problem, each defined by a distinct combination of integration boundaries; these are shown in Figures 15 and 17. Together, these cases encompass all possible occultation configurations, for any illumination angle, occultor size, and occultor position.

Before we discuss how to compute the occultation integrals, we must first develop a procedure to identify the relevant case given the occultor impact parameter $b_o = \sqrt{x_o^2 + y_o^2}$ and radius r_o and the terminator semiminor axis b and angle θ' in the frame $\mathcal{F}'$. Then, once the case is determined, we must identify the relevant integration boundaries, which depend on the points of intersection between the limb of the body, the limb of the occultor, and the terminator. We do so in the following sections.

[42] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/PhaseCurve.ipynb
[43] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/PhaseCurve.ipynb
[44] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/PhaseCurve.ipynb
[45] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/PhaseCurve.ipynb

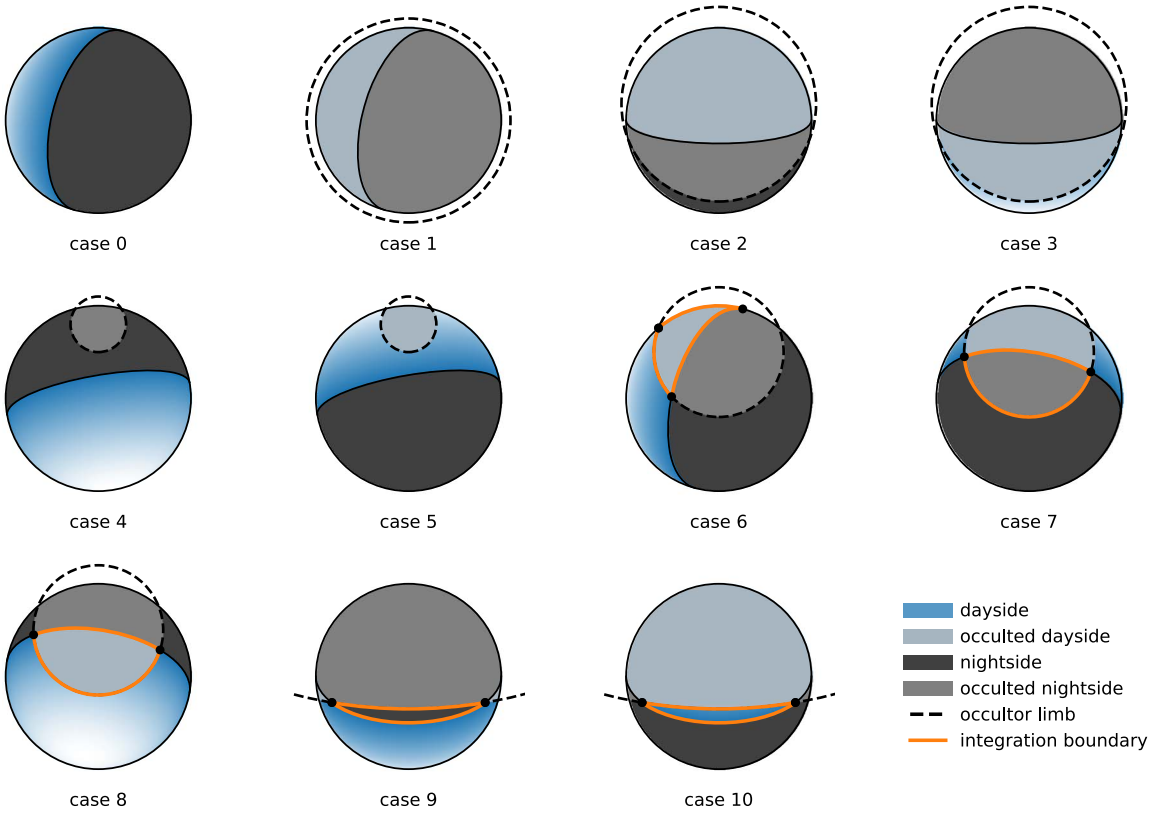

Figure 15. The 10 principal families of cases of occultations in reflected light. In these figures, the body with the solid outline is the one whose flux we are interested in, and the body with the dashed outline is the occultor. The nightside of the occulted body is colored black (dark gray if occulted), and the dayside is colored blue (bluish-gray if occulted). Case 0 is the unocculted case (Appendix B), while cases 1–5 involve configurations in which the limb of the occultor does not intersect with the terminator at any point, so the visible flux may be computed in terms of classical starry integrals. The remaining cases require integration along the orange boundary (the curves $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$ of Appendix C.3), which include the terminator. These involve the evaluation of incomplete elliptic integrals and are derived below. Note, finally, that there are additional subcases not shown above. For instance, cases 4, 5, 7, and 8 also encompass configurations in which the occultor does not intersect the limb of the occulted body. However, as this distinction does not affect the procedure for computing the flux in these cases (see text), we omit these subcases from the figure.[46]

C.1. Case Determination

The key to identifying the case corresponding to a given configuration is to determine whether or not the limb of the occultor intersects the terminator of the body, and if so, the points of intersection. While we perform the integration in frame $\mathcal{F}'$, finding the points of intersection with the terminator is easier if we temporarily switch to the frame $\mathcal{F}''$, in which the terminator is parallel to the x''-axis. In this frame, the equations defining the terminator and the limb of the occultor are, respectively,

$$y_1''(x'') = b\sqrt{1 - x''^2}$$
$$y_2''(x'') = y_o'' \pm \sqrt{r_o^2 - (x'' - x_o'')^2}\,, \tag{C1}$$

where

$$x_o'' = b_o \sin\theta'$$
$$y_o'' = b_o \cos\theta' \tag{C2}$$

are the coordinates of the occultor in $\mathcal{F}''$. We wish to find the vector of N points $\boldsymbol{x}'' = (x_0, x_1, \cdots, x_{N-1})^\top$ for which $y_1''(x_n'') - y_2''(x_n'') = 0$. Following Luger et al. (2017), we may express this condition as the quartic equation

$$Ax''^4 + Bx''^3 + Cx''^2 + Dx'' + E = 0, \tag{C3}$$

with coefficients

$$A = (1 - b^2)^2$$
$$B = -4x_o''(1 - b^2)$$
$$C = -2(b^4 + r_o^2 - 3x_o''^2 - y_o''^2 - b^2(1 + r_o^2 - x_o''^2 + y_o''^2))$$
$$D = -4x_o''(b^2 - r_o^2 + x_o''^2 + y_o''^2)$$
$$E = b^4 - 2b^2(r_o^2 - x_o''^2 + y_o''^2) + (r_o^2 - x_o''^2 - y_o''^2)^2. \tag{C4}$$

Although closed-form solutions to quartic equations exist (see, e.g., Hughes & Chraibi 2011, who solve for the area of overlap between two ellipses analytically), they are prone to significant numerical instabilities. Instead, we solve for the roots of the quartic numerically by casting it as an eigenvalue problem (e.g., Edelman & Murakami 1995) and polish the results with a few iterations of Newton's method. We find that this is reasonably computationally efficient and yields roots with precision within a couple orders of magnitude of machine epsilon (see Section 3.2).[47,48,49,50]

[46] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/cases.py
[47] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/quartic.ipynb
[48] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/quartic.ipynb
[49] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/quartic.ipynb
[50] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/quartic.ipynb

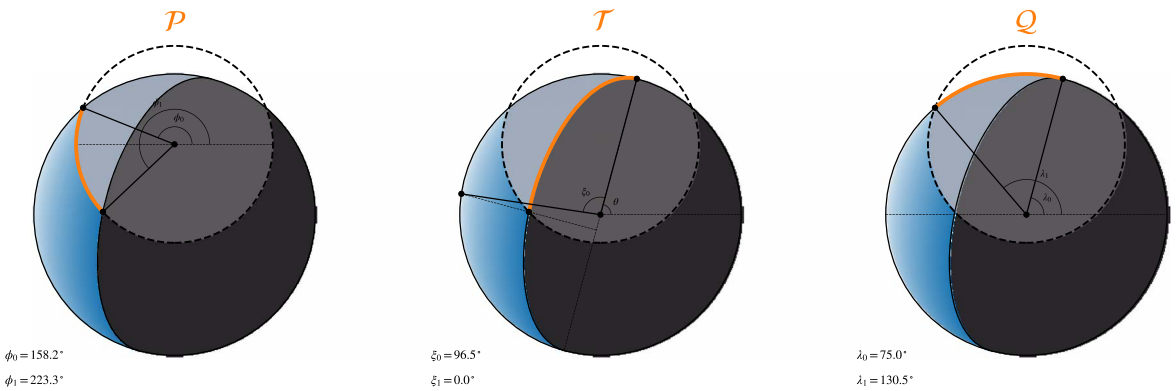

Figure 16. Geometry of an occultation in reflected light, corresponding to case 6 in Figure 15. The surface integral over the occulted portion of the dayside (bluish-gray region) is computed from the line integrals of the antiderivatives of the surface intensity map along the boundary curves $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$. See text for details .[51]

In general, the quartic defined by Equation (C3) has $N = 4$ (potentially degenerate) roots, some of which may be complex, and some of which correspond to intersections with the wrong half of the terminator ellipse (i.e., the section of the terminator on the far side of the body). After excluding the unphysical solutions, we are still left with anywhere between zero and four roots.

Cases with zero roots (case 1–case 5) are treated in Appendix C.2, while cases with one or two roots (case 6–case 10) are treated in Appendix C.3. Cases with three or four roots (case 11–case 14) are rarely encountered in practice, but are possible for some pathological configurations; these are treated in Appendix C.4.

C.2. Cases 1–5

Cases 1–5 (see Figure 15) involve configurations in which the occultor does not intersect with the terminator of the occulted body, and are therefore fairly straightforward to solve. In particular, we can use the original emitted light solution from Luger et al. (2019a), provided we weight the map by our polynomial illumination function:

$$f_\mathrm{I} = \boldsymbol{s}^\top(b_\mathrm{o}, r_\mathrm{o})\boldsymbol{A_2 I}(b, \theta', r_\mathrm{s})\boldsymbol{A_1 R'}(x_\mathrm{o}, y_\mathrm{o})\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}, \quad \text{(C5)}$$

where $\boldsymbol{s}^\top(b_\mathrm{o}, r_\mathrm{o})$ is the emitted light solution vector (Equation (26) in Luger et al. 2019a). The flux f_I is the flux one would measure from a body whose surface map is weighted by the illumination function $\boldsymbol{I}(b, \theta', r_\mathrm{s})$ during an occultation. Note that this is not necessarily the *observed* flux, since this may include the unphysical negative contribution from the nightside. We must compute the actual observed flux on a case-by-case basis.

Case 1 corresponds to any complete occultation of the body ($b_\mathrm{o} \leqslant r_\mathrm{o} - 1$), so the solution for the flux is trivial:

$$f_1 = 0. \quad \text{(C6)}$$

Case 2 corresponds to occultations in which the occultor blocks *all* of the dayside of the body and *some* of the nightside. In this configuration, the unocculted part of the disk consists only of nightside, so the solution is again trivial:

$$f_2 = 0. \quad \text{(C7)}$$

Conversely, case 3 corresponds to occultations in which the occultor blocks *all* of the nightside of the body and *some* of the dayside. Since the visible portion of the disk consists only of dayside, we can simply use the weighted solution in emitted light (Equation (C5)):

$$f_3 = f_\mathrm{I}. \quad \text{(C8)}$$

Case 4 involves any occultation in which the occultor blocks *only* the nightside of the body (regardless of whether or not it intersects with the limb of the body). Since the nightside intensity is zero everywhere, this case is also trivial, as the flux is equal to the flux in the no occultation case (Equation (B6)):

$$f_4 = f_0. \quad \text{(C9)}$$

Finally, case 5 involves any occultation in which the occultor blocks *only* the dayside of the body (regardless of whether or not it intersects with the limb). We first compute the illumination-weighted flux f_I as above, then negate the unphysical nightside contribution using Equation (B7):

$$f_5 = f_\mathrm{I} - \hat{f}_0. \quad \text{(C10)}$$

C.3. Cases 6–10

Cases 6–10 (see Figure 15) correspond to configurations in which the limb of the occultor intersects with the terminator at either one point (case 6) or two points (cases 7–10). Because of these intersections, we cannot simply reweight the emitted light solution, as the integration boundaries are now different. In general, we may compute the flux by integrating the components of the Green's basis $\tilde{\boldsymbol{g}}$ over the region S bounded by three curves, which we denote $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$. These are shown in orange in Figure 15 and presented in more detail in Figure 16. Curve $\mathcal{P}$ is a segment of the limb of the occultor, parametrized by the angle $\phi \in [\phi_0, \phi_1]$; curve $\mathcal{T}$ is a segment of the terminator, parametrized by the angle $\xi \in [\xi_0, \xi_1]$; and curve $\mathcal{Q}$ is a segment of the limb of the occulted body, parametrized by the angle $\lambda \in [\lambda_0, \lambda_1]$. The endpoints ϕ_0, ϕ_1, ξ_0, ξ_1, λ_0, and λ_1 are functions of the solutions to the quartic from Appendix C.1 and will be presented in Appendix C.5.

Let $\mathbb{s}^\top$ be the integral of $\tilde{\boldsymbol{g}}^\top$ over S:

$$\mathbb{s}^\top(b, \theta', b_\mathrm{o}, r_\mathrm{o}) = \iint_{S(b, \theta', b_\mathrm{o}, r_\mathrm{o})} \tilde{\boldsymbol{g}}^\top(x', y')\, dx'\, dy'. \quad \text{(C11)}$$

[51] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/geometry.py

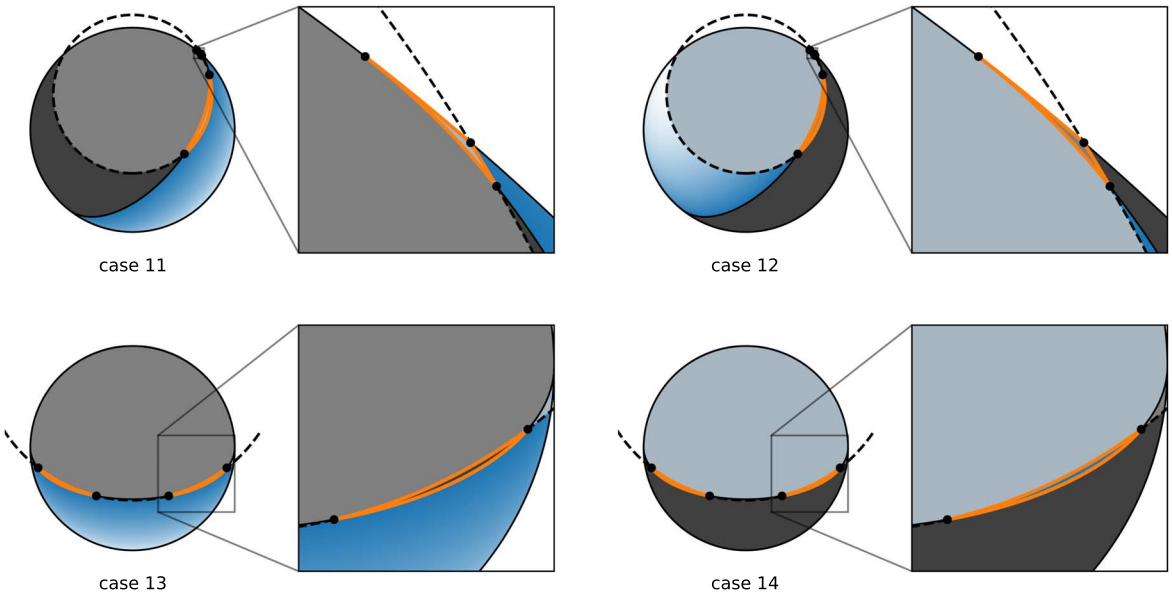

Figure 17. Four additional families of occultations in reflected light, involving rare triple (cases 11 and 12, top) and quadruple (cases 13 and 14, bottom) intersections between the limb of the occultor and the terminator of the occulted body. All four cases involve integration over two disjoint regions (bounded by the orange curves in the figure). The insets next to each case show a zoomed-in version of four such regions. See text for more details.[52]

We defer the solution to Equation (C11) to Appendix C.5 below, as it is quite lengthy. Given $\mathbb{s}^{\top}$, the flux $f_{\rm S}$ over the integration region S is computed from Equation (A18):

$$f_{\rm S} = \mathbb{s}^{\top}(b, \theta', b_{\rm o}, r_{\rm o})\boldsymbol{A_2 I}(b, \theta', r_{\rm s})\boldsymbol{A_1 R'}(x_{\rm o}, y_{\rm o})\boldsymbol{R}(\mathrm{I}, \Lambda, \Theta)\boldsymbol{y}. \tag{C12}$$

Note again that this is not necessarily the *observed* flux, which we must compute on a case-by-case basis below.

Case 6 corresponds to configurations in which the limb of the occultor intersects the terminator at a single point. The integration region (see Figures 15 and 16) is the occulted portion of the dayside, which is bounded by all three curves $\mathcal{P}$, $\mathcal{Q}$, and $\mathcal{T}$. The total flux may be computed by subtracting the occulted flux $f_{\rm S}$ from the total dayside flux f_0:

$$f_6 = f_0 - f_{\rm S}. \tag{C13}$$

Cases 7–10 involve two points of intersection between the occultor limb and the terminator. Cases 7 and 8 correspond to occultors that block some of the nightside and some of the dayside, but *neither* of the extrema of the terminator ellipse. In case 7 a lens-shaped region is formed by the intersection of the occultor limb and the terminator on the *nightside*, while in case 8 this region is formed on the *dayside*. In case 7, we begin by computing the flux over the unocculted region, $f_{\rm I}$, which includes the spurious nightside contribution. We then remove this contribution by noting that it is equal to the total nightside contribution, $\hat{f}_0$, minus the occulted nightside flux, $f_{\rm S}$:

$$f_7 = f_{\rm I} - (\hat{f}_0 - f_{\rm S}). \tag{C14}$$

Case 8, on the other hand, is equivalent to case 6, since the integration region consists of occulted dayside:

$$f_8 = f_0 - f_{\rm S}. \tag{C15}$$

Cases 9 and 10 correspond to occultors that also block some nightside and some dayside, along with *both* of the extrema of the ellipse; these are therefore exclusively for large occultors ($r_{\rm o} > 1$). Case 9 involves occultations in which only a small lens-shaped region of the nightside is visible. The total flux is the visible dayside plus unphysical nightside contribution, $f_{\rm I}$, minus the nightside contribution, which we compute from Equation (C12):

$$f_9 = f_{\rm I} - f_{\rm S}. \tag{C16}$$

Conversely, case 10 involves occultations in which only a small lens-shaped region of the dayside is visible. In this case, we may compute the observed flux from Equation (C12) directly:

$$f_{10} = f_{\rm S}. \tag{C17}$$

C.4. Cases 11–14

Cases 11–14 correspond to (rare) configurations involving three or four roots to Equation (C3) and are illustrated in Figure 17. All four involve integration over two disjoint regions (see the figure). Cases 11 and 12 involve three points of intersection between the terminator and the occultor limb. In case 11, the regions of integration S_1 and S_2 are the occulted portion of the dayside, so the solution is similar to that of cases 6 and 8:

$$f_{11} = f_0 - (f_{\rm S_1} + f_{\rm S_2}), \tag{C18}$$

where $f_{\rm S_1}$ and $f_{\rm S_2}$ are computed from Equation (C12) for each of the integration regions. Conversely, in case 12 the two regions are the occulted portion of the nightside, so the solution is similar to that of case 7:

$$f_{12} = f_{\rm I} - (\hat{f}_0 - (f_{\rm S_1} + f_{\rm S_2})). \tag{C19}$$

[52] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/figures/pathological.py

Finally, cases 13 and 14 involve four points of intersection between the terminator and the occultor limb. The regions of integration in case 13 are the visible portion of the nightside, so this case is equivalent to case 9:

$$f_{13} = f_{\rm I} - (f_{\rm S_1} + f_{\rm S_2}). \tag{C20}$$

Conversely, the regions of integration in case 14 are the visible portion of the dayside, so this case is equivalent to case 10:

$$f_{14} = f_{\rm S_1} + f_{\rm S_2}. \tag{C21}$$

C.5. Computing the Integrals $\mathbb{s}^\top$

In the previous sections, we discussed how to identify the case corresponding to a specific configuration of the occultor and the illumination source. We showed how in some cases (1–5; Appendix C.2) the total flux may be computed by exploiting the classical starry integrals (Equation (C5)). In all other cases (6–14; Appendices C.3 and C.4), however, the flux computation involves evaluation of Equation (C12), where the solution vector $\mathbb{s}^\top$ is the vector of integrals (in the frame $\mathcal{F}'$) of each of the terms in the Green's basis $\tilde{\boldsymbol{g}}$ over a region S of the projected disk of the occultor (Equation (C11)). As in Luger et al. (2019a), the approach to computing $\mathbb{s}^\top$ is to use Green's theorem to transform the surface integrals into line integrals along the curves $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$ (see Figure 16). Specifically, we write

$$\begin{aligned}\mathbb{s}^\top &= \iint_S \tilde{\boldsymbol{g}}^\top(x', y')\, dx'\, dy' \\ &= \oint \boldsymbol{G}^\top(x', y') \cdot d\boldsymbol{r}(x', y'),\end{aligned} \tag{C22}$$

where $\boldsymbol{G}(x', y')$ is a vector of two-dimensional Cartesian vectors chosen such that its exterior derivative is $\tilde{\boldsymbol{g}}$,

$$\frac{d\boldsymbol{G}_{y'}(x', y')}{dx'} - \frac{d\boldsymbol{G}_{x'}(x', y')}{dy'} = \tilde{\boldsymbol{g}}(x', y'), \tag{C23}$$

and

$$d\boldsymbol{r}(x', y') = \left(\frac{dx'}{d\varphi}\right) d\varphi\, \hat{\boldsymbol{x}}' + \left(\frac{dy'}{d\varphi}\right) d\varphi\, \hat{\boldsymbol{y}}', \tag{C24}$$

where φ is the parametrized angle along the integration path and the integral is taken in a counter-clockwise direction relative to the center of the integration region. Luger et al. (2019a) showed that one possible solution to Equation (C23) consists of the vector whose nth component is given by

$$\boldsymbol{G}_n(x', y') = \begin{cases} x'^{\frac{\mu+2}{2}} y'^{\frac{\nu}{2}}\, \hat{\boldsymbol{y}}' & \mu,\ \nu \text{ even} \\ \dfrac{1 - z(x', y')^3}{3(1 - z(x', y')^2)}(-y'\, \hat{\boldsymbol{x}}' + x'\, \hat{\boldsymbol{y}}') & \mu = \nu = 1 \\ x'^{l-2} z(x', y')^3\, \hat{\boldsymbol{x}} & \nu \text{ odd},\ \mu = 1,\ l \text{ even} \\ x'^{l-3} y' z(x', y')^3\, \hat{\boldsymbol{x}} & \nu \text{ odd},\ \mu = 1,\ l \text{ odd} \\ x'^{\frac{\mu-3}{2}} y'^{\frac{\nu-1}{2}} z(x', y')^3\, \hat{\boldsymbol{y}} & \text{otherwise,} \end{cases} \tag{C25}$$

where the indices l, m, μ, ν are given by Equations (A2) and (A5).[53]

[53] In this section, we deliberately drop the dependence of $\mathbb{s}^\top$ and the primitive integrals on the geometrical parameters b, θ', $b_{\rm o}$, $r_{\rm o}$ for clarity.

We showed in the previous sections that there are at most three curves $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$ bounding a given closed surface of integration (see Figure 16). We may therefore express Equation (C22) as

$$\mathbb{s}^\top = \mathbb{p}^\top + \mathbb{t}^\top + \mathbb{q}^\top, \tag{C26}$$

where we define the primitive integrals

$$\mathbb{p}^\top = \int_\phi \boldsymbol{G}^\top(x'_{\rm p}, y'_{\rm p}) \cdot d\boldsymbol{r}(x'_{\rm p}, y'_{\rm p}), \tag{C27}$$

$$\mathbb{t}^\top = \int_\xi \boldsymbol{G}^\top(x'_{\rm t}, y'_{\rm t}) \cdot d\boldsymbol{r}(x'_{\rm t}, y'_{\rm t}), \tag{C28}$$

$$\mathbb{q}^\top = \int_\lambda \boldsymbol{G}^\top(x'_{\rm q}, y'_{\rm q}) \cdot d\boldsymbol{r}(x'_{\rm q}, y'_{\rm q}) \tag{C29}$$

to be the line integrals of $\boldsymbol{G}$ along each of the curves $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{Q}$, respectively," where the coordinates along each curve are parametrized in terms of φ as follows:

$$\begin{aligned} x'_{\rm p} &= r_{\rm o} \cos\varphi \\ x'_{\rm t} &= \cos\theta' \cos\varphi - b \sin\theta' \sin\varphi \\ x'_{\rm q} &= \cos\varphi \\ y'_{\rm p} &= b_{\rm o} + r_{\rm o} \sin\varphi \\ y'_{\rm t} &= \sin\theta' \cos\varphi + b \cos\theta' \sin\varphi \\ y'_{\rm q} &= \sin\varphi, \end{aligned} \tag{C30}$$

and we define

$$\begin{aligned}\int_{\boldsymbol{\varphi}} &\equiv \int_{\varphi_0}^{\varphi_1} + \int_{\varphi_2}^{\varphi_3} + \cdots + \int_{\varphi_{N-2}}^{\varphi_{N-1}} \\ &\equiv \sum_{i=0}^{\frac{N}{2}-1} \int_{\varphi_{2i}}^{\varphi_{2i+1}}\end{aligned} \tag{C31}$$

to be the sum of definite integrals between pairs of limits φ_i arranged in a vector $\boldsymbol{\varphi}$ of length N. For future reference, it will also be useful to define the operator

$$\Delta\boldsymbol{x} \equiv \sum_{i=0}^{\frac{N}{2}-1} (x_{2i+1} - x_{2i}), \tag{C32}$$

which sums the difference of successive pairs of values in a vector $\boldsymbol{x} = (x_0, x_1, x_2, x_3, \cdots, x_{N-1})^\top$. This will come in handy when computing definite integrals. Specifically, if g is the antiderivative of some function f, we may use the fundamental theorem of calculus to compute the integral of f over the interval(s) given by the vector of limit pairs $\boldsymbol{\varphi}$:

$$\begin{aligned}\int_{\boldsymbol{\varphi}} f(\varphi) d\varphi &= \int_{\varphi_0}^{\varphi_1} f(\varphi) d\varphi + \int_{\varphi_2}^{\varphi_3} f(\varphi) d\varphi + \cdots \\ &\quad + \int_{\varphi_{N-2}}^{\varphi_{N-1}} f(\varphi) d\varphi \\ &= \Delta\boldsymbol{g}(\boldsymbol{\varphi}),\end{aligned} \tag{C33}$$

where $\boldsymbol{g}$ is the vector given by

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{\varphi}) = (&g(\varphi_0), g(\varphi_1), g(\varphi_2), g(\varphi_3), \cdots, \\ &g(\varphi_{N-2}), g(\varphi_{N-1}))^\top.\end{aligned} \tag{C34}$$

Note that most of the cases (1–10) involve integration over a single closed region, so Equations (C31) and (C32) reduce to

$$\int_{\varphi} \equiv \int_{\varphi_0}^{\varphi_1} \tag{C35}$$

and

$$\Delta \boldsymbol{x} \equiv x_1 - x_0. \tag{C36}$$

For cases 11–14, we must integrate over two disjoint regions, so we sum over two pairs of limits. In the next three sections, we derive the solutions to each of the primitive integrals $\mathbb{p}^{\top}$, $\mathbb{t}^{\top}$, and $\mathbb{q}^{\top}$.[54,55,56,57,58,59]

C.6. The Integral Along the Occultor Limb, $\mathbb{p}^{\top}$

In this section we present a solution to Equation (C27). The first order of business is to derive expressions for the integration limits ϕ. Depending on the integration case, these limits will correspond to the point of intersection between the limb of the occultor and the limb of the occulted body and/or the point of intersection between the limb of the occultor and the terminator of the occulted body. The former is given by (see Equation (24) in Luger et al. 2019a)

$$\phi_0 = \frac{\pi}{2} \pm \left(\arcsin\left(\frac{1 - r_o^2 - b_o^2}{2 b_o r_o} \right) - \frac{\pi}{2} \right), \tag{C37}$$

where the sign is chosen such that the point $(r_o \cos\phi_0, b_o + r_o \sin\phi_0)$ is on the dayside of the occulted body, and the latter (of which there may be multiple) is given by

$$\boldsymbol{\phi_1} = \theta' + \arctan 2(b\sqrt{1 - \boldsymbol{x}''^2} - y_o'', \boldsymbol{x}'' - x_o''), \tag{C38}$$

where $\boldsymbol{x}''$ are the roots of the quartic (Equation (C3)). These angles are then wrapped to the range $[0, 2\pi)$ and sorted into the vector $\boldsymbol{\phi}$ such that the integration is always performed in a counter-clockwise sense about the center of the integration region. The left panel in Figure 16 shows a configuration in which the lower integration limit $\phi_0 = 158.^{\circ}2$ corresponds to the point of intersection between the limbs of the two bodies and the upper integration limit $\phi_1 = 223.^{\circ}3$ corresponds to the limb-terminator intersection. Both angles are measured counter-clockwise from the line $x' = b_o$.[60,61]

In order to evaluate the integral in Equation (C27), we follow the reparametrization tricks of Appendix D.2.3 in Luger et al. (2019a). The algebra is long and tedious, so we merely present the result (alongside the usual validation links). The nth component of $\mathbb{p}^{\top}$ is

$$\mathbb{p}_n = \begin{cases} 2(2r_o)^{l+2} \begin{cases} V\left(\frac{\mu+4}{4}, \frac{\nu}{2}, 0; \mathbb{i}\right) & \frac{\mu}{2} \text{ even} \\ V\left(\frac{\mu+2}{4}, \frac{\nu}{2}, 0; \mathbb{u}\right) & \frac{\mu}{2} \text{ odd} \end{cases} & \mu, \nu \text{ even} \\ \mathbb{p}_2 & \mu = \nu = 1 \\ \beta(2r_o)^{l-1} \begin{cases} V\left(\frac{l-2}{2}, 0, 0; \mathbb{j}\right) - 2V\left(\frac{l-2}{2}, 0, 1; \mathbb{j}\right) & l \text{ even} \\ V\left(\frac{l-3}{2}, 1, 0; \mathbb{j}\right) - 2V\left(\frac{l-3}{2}, 1, 1; \mathbb{j}\right) & l \text{ odd} \neq 1 \end{cases} & \mu = 1 \\ \beta(2r_o)^{l-1} \begin{cases} 2V\left(\frac{\mu-1}{4}, \frac{\nu-1}{2}, 0; \mathbb{j}\right) & \frac{\mu-1}{2} \text{ even} \\ 2V\left(\frac{\mu-1}{4}, \frac{\nu-1}{2}, 0; \mathbb{w}\right) & \frac{\mu-1}{2} \text{ odd} \end{cases} & \mu > 1 \end{cases} \quad \mu, \nu \text{ odd}, \tag{C39}$$

where $\beta = (1 - (b_o - r_o)^2)^{\frac{3}{2}}$ and we define the Vieta operator

$$V(u, v, w; \boldsymbol{x}) \equiv \sum_{i=0}^{u+v} \mathcal{A}_{u,v,i} x_{u+w+i} \tag{C40}$$

as the dot product of a vector $\boldsymbol{x}$ and the vector of Vieta's theorem coefficients, where (see Equation (D34) in Luger et al. 2019a)

$$\mathcal{A}_{u,v,i} = \sum_{j=\max(0,u-i)}^{\min(u+v-i,u)} \binom{u}{j} \binom{v}{u+v-i-j} \times (-1)^{u+j} \left(\frac{b_o - r_o}{2r_o} \right)^{u+v-i-j}. \tag{C41}$$

[54] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/Greens.ipynb

[55] The components of the vectors $\mathbb{p}^{\top}$ and $\mathbb{q}^{\top}$ are analogous to the primitive integrals $\mathcal{P}$ and $\mathcal{Q}$ defined in Equations (30)–(32) in Luger et al. (2019a), although the integration limits of both and the sense of integration of $\mathcal{P}$ are different.

[56] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/Greens.ipynb

[57] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/Greens.ipynb

[58] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/Greens.ipynb

[59] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/Greens.ipynb

[60] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/pT.ipynb

[61] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/pT.ipynb

The vectors $\mathbb{i}$, $\mathbb{j}$, $\mathbb{u}$, and $\mathbb{w}$ are solutions to specific integrals, which we compute recursively below. As in Luger et al. (2019a) the $n = 2$ term of $\mathbb{p}^\top$, $\mathbb{p}_2$, is handled separately; we also compute this below.[62,63,64]

Note that several of the cases in Equation (C39) are identical to those in Equation (D35) of Luger et al. (2019a), provided we replace their integrals $\mathcal{I}$ and $\mathcal{J}$ with our integrals $\mathbb{i}$ and $\mathbb{j}$, respectively. The integrals themselves are similar, except for a change in the limits of integration, which are no longer symmetric about zero. As we will see, this leads to the dependence of these expressions on *incomplete* elliptic integrals. Note also that the integrals $\mathbb{u}$ and $\mathbb{w}$ are new, as certain cancellations in Luger et al. (2019a) resulted in the corresponding cases contributing zero net flux (last case in Equation (D35) of Luger et al. 2019a).

C.6.1. The Vector $\mathbb{i}$

The components of the vector $\mathbb{i}$ are given by the integral

$$\mathbb{i}_v(\boldsymbol{\alpha}) = \int_{\boldsymbol{\alpha}} \sin^{2v}\varphi d\varphi, \tag{C42}$$

for $v \in [0, v_{\max}]$, where we define the helper angle

$$\boldsymbol{\alpha} \equiv \frac{\boldsymbol{\phi}}{2} + \frac{\pi}{4}. \tag{C43}$$

The integral in the expression above is the same as that in Equation (D38) of Luger et al. (2019a), except for a change in the limits of integration. As in Luger et al. (2019a), we can compute the vector $\mathbb{i}$ recursively given a trivial lower boundary condition:

$$\begin{aligned}\mathbb{i}_0(\boldsymbol{\alpha}) &= \Delta\boldsymbol{\alpha}\\ \mathbb{i}_v(\boldsymbol{\alpha}) &= \frac{1}{2v}((2v-1)\mathbb{i}_{v-1}(\boldsymbol{\alpha}) - \Delta(\sin^{2v-1}\boldsymbol{\alpha}\cos^{2v-1}\boldsymbol{\alpha})),\end{aligned} \tag{C44}$$

where the last expression is valid for all $v > 0$. We find that this algorithm is generally stable, except when $\sin\boldsymbol{\alpha}$ is small. In that limit, we evaluate $\mathbb{i}_{v_{\max}}(\boldsymbol{\alpha})$ by numerical integration of Equation (C42) using Gauss–Legendre quadrature with 100 points. We then recurse downward by substituting $v \to v+1$ in Equation (C44) and solving for $\mathbb{i}_v(\boldsymbol{\alpha})$.[65]

C.6.2. The Vector $\mathbb{j}$

The components of the vector $\mathbb{j}$ are given by the integral

$$\mathbb{j}_v(k^2, \boldsymbol{\alpha}) = \int_{\boldsymbol{\alpha}} \sin^{2v}\varphi\left(1 - \frac{\sin^2\varphi}{k^2}\right)^{\frac{3}{2}} d\varphi, \tag{C45}$$

where

$$k^2 \equiv \frac{1 - r_o^2 - b_o^2 + 2b_o r_o}{4b_o r_o}. \tag{C46}$$

The integral in this expression is again the same as that in Equation (D39) of Luger et al. (2019a), except for a change in the limits of integration. In that paper, we computed all terms $\{\mathbb{j}_0, \cdots, \mathbb{j}_{v_{\max}}\}$ from a three-term recurrence relation and two boundary conditions. In the case of upward recursion, the boundary conditions $\mathbb{j}_0$ and $\mathbb{j}_1$ were computed analytically from the complete elliptic integrals $K(k^2)$ and $E(k^2)$. In cases where upward recursion was not numerically stable, we evaluated $\mathbb{j}_{v_{\max}}$ and $\mathbb{j}_{v_{\max}-1}$ via a quickly convergent series expansion and recursed downward.

In order to solve Equation (C45), it is possible to replace the complete elliptic integrals $K(k^2)$ and $E(k^2)$ in the lower boundary conditions (Equation (D46) in Luger et al. 2019a) with the incomplete elliptic integrals

$$F(\psi \mid m) \equiv \int_0^\psi \frac{d\varphi}{\sqrt{1 - m\sin^2\varphi}} \tag{C47}$$

and

$$E(\psi|m) \equiv \int_0^\psi \sqrt{1 - m\sin^2\varphi}\, d\varphi, \tag{C48}$$

which we compute from the *el*2 parameterization of Bulirsch (1965), then use the same upward recursion relation to obtain analytic solutions for all $\mathbb{j}_v$:

$$\begin{aligned}\mathbb{j}_0(k^2, \boldsymbol{\alpha}) &= \frac{1}{3}\left(2\left(2 - \frac{1}{k^2}\right)\Delta\boldsymbol{E}\left(\boldsymbol{\alpha} \,\middle|\, \frac{1}{k^2}\right)\right.\\ &\quad \left. + \left(\frac{1}{k^2} - 1\right)\Delta\boldsymbol{F}\left(\boldsymbol{\alpha} \,\middle|\, \frac{1}{k^2}\right) + \Delta z_0(k^2, \boldsymbol{\alpha})\right)\\ \mathbb{j}_1(k^2, \boldsymbol{\alpha}) &= \frac{1}{15}\left(\left(-3k^2 + 13 - \frac{8}{k^2}\right)\Delta\boldsymbol{E}\left(\boldsymbol{\alpha} \,\middle|\, \frac{1}{k^2}\right)\right.\\ &\quad \left. + \left(3k^2 - 7 + \frac{4}{k^2}\right)\Delta\boldsymbol{F}\left(\boldsymbol{\alpha} \,\middle|\, \frac{1}{k^2}\right) + \Delta z_1(k^2, \boldsymbol{\alpha})\right)\\ \mathbb{j}_v(k^2, \boldsymbol{\alpha}) &= \frac{1}{2v+3}(2(v + 1 + (v-1)k^2)\,\mathbb{j}_{v-1}(k^2, \boldsymbol{\alpha})\\ &\quad - (2v-3)k^2\mathbb{j}_{v-2}(k^2, \boldsymbol{\alpha}) + \Delta z_v(k^2, \boldsymbol{\alpha})),\end{aligned} \tag{C49}$$

where the last expression is valid for all $v > 1$ and

$$\begin{aligned}z_0(k^2, \boldsymbol{\alpha}) &= \frac{\sin\boldsymbol{\alpha}\cos\boldsymbol{\alpha}\, \boldsymbol{q}(k^2, \boldsymbol{\alpha})}{k^2}\\ z_1(k^2, \boldsymbol{\alpha}) &= (3\sin^2\boldsymbol{\alpha} + 4 - 6k^2)z_0(k^2, \boldsymbol{\alpha})\\ z_v(k^2, \boldsymbol{\alpha}) &= k^2\sin^{2v-3}\boldsymbol{\alpha}\cos\boldsymbol{\alpha}\, \boldsymbol{q}(k^2, \boldsymbol{\alpha})^5,\end{aligned} \tag{C50}$$

with

$$\boldsymbol{q}(k^2, \boldsymbol{\alpha}) = \sqrt{1 - \frac{\sin^2\boldsymbol{\alpha}}{k^2}}. \tag{C51}$$

Note that when $k^2 < 1$ we use the reciprocal-modulus transformation to evaluate the elliptic integrals:

$$\begin{aligned}F\left(\psi \,\middle|\, \frac{1}{k^2}\right) &= k\, F(\beta \mid k^2)\\ E\left(\psi \,\middle|\, \frac{1}{k^2}\right) &= \frac{E(\beta \mid k^2) - (1-k^2)F(\beta \mid k^2)}{k}\end{aligned} \tag{C52}$$

[62] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/pT.ipynb
[63] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/pT.ipynb
[64] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/pT.ipynb
[65] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/I.ipynb

with[66,67]

$$\beta = \arcsin\left(\frac{\sin\psi}{k}\right). \tag{C53}$$

In practice, however, we find that this procedure is even more numerically unstable than it was in Luger et al. (2019a). To address this, we express the recurrence structure of the problem as a tridiagonal system with one lower boundary condition $\mathbb{j}_0$ and one upper boundary condition $\mathbb{j}_{v_{\max}}$:

$$\begin{pmatrix} a_0 & 1 & & & & \\ b_1 & a_1 & 1 & & & \\ & b_2 & a_2 & 1 & & \\ & & b_0 & a_3 & 1 & \\ & & & \ddots & \ddots & \ddots \\ & & & & b_{v_{\max}} & a_{v_{\max}} \end{pmatrix} \begin{pmatrix} \mathbb{j}_1 \\ \mathbb{j}_2 \\ \mathbb{j}_3 \\ \mathbb{j}_4 \\ \cdots \\ \mathbb{j}_{v_{\max}-1} \end{pmatrix} = \begin{pmatrix} c_0 - b_0\mathbb{j}_0 \\ c_1 \\ c_2 \\ c_3 \\ \cdots \\ c_{v_{\max}} - \mathbb{j}_{v_{\max}} \end{pmatrix}, \tag{C54}$$

where the recursion coefficients are given by

$$\begin{aligned} a_v(k) &= -2\frac{(v+1)+(v-1)k^2}{2v+3} \\ b_v(k) &= \frac{(2v-3)k^2}{2v+3} \\ c_v(k^2, \boldsymbol{\alpha}) &= \Delta\left(\frac{z_v(k^2, \boldsymbol{\alpha})}{2v+3}\right). \end{aligned} \tag{C55}$$

Solving this matrix system yields values for all intermediate $\{\mathbb{j}_1, \cdots, \mathbb{j}_{v_{\max}-1}\}$. While efficient algorithms exist for solving tridiagonal problems, we obtain far better numerical stability by instead performing traditional LU decomposition. We find that this algorithm is stable in all the regimes that we tested.[68,69]

We evaluate the upper boundary condition $\mathbb{j}_{v_{\max}}$ by numerical integration of Equation (C45) via Gauss–Legendre quadrature with 100 points. While the lower boundary condition may be computed analytically from Equation (C49), in practice we achieve better precision via numerical integration (as above), with negligible effects on computational performance.

C.6.3. The Vector $\mathbb{u}$

The components of the vector $\mathbb{u}$ are given by the integral

$$\mathbb{u}_v(\boldsymbol{\alpha}) = \int_{\boldsymbol{\alpha}} \cos\varphi \sin^{2v+1}\varphi d\varphi. \tag{C56}$$

This integral has an analytic solution for all v:[70]

$$\mathbb{u}_v(\boldsymbol{\alpha}) = \frac{\Delta \sin^{2v+2}\boldsymbol{\alpha}}{2v+2}. \tag{C57}$$

C.6.4. The Vector $\mathbb{w}$

The components of the vector $\mathbb{w}$ are given by the integral

$$\mathbb{w}_v(k^2, \boldsymbol{\alpha}) = \int_{\boldsymbol{\alpha}} \cos\varphi \sin^{2v+1}\varphi \left(1 - \frac{\sin^2\varphi}{k^2}\right)^{\frac{3}{2}} d\varphi. \tag{C58}$$

We may compute it by either upward or downward recursion. In both cases, we compute each of the $\mathbb{w}_v$ from

$$\mathbb{w}_v(k^2, \boldsymbol{\alpha}) = \Delta \boldsymbol{b}_v(k^2, \boldsymbol{\alpha}). \tag{C59}$$

In the upward case, we start with the lower boundary conditions

$$\begin{aligned} \boldsymbol{b}_0(k^2, \boldsymbol{\alpha}) &= \frac{\sin^2\boldsymbol{\alpha}}{5}\left(\frac{1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^3}{1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^2} + \boldsymbol{q}(k^2, \boldsymbol{\alpha})^3\right) \\ \boldsymbol{c}_0(k^2, \boldsymbol{\alpha}) &= \sin^4\boldsymbol{\alpha}\,\frac{\boldsymbol{q}(k^2, \boldsymbol{\alpha})^5}{1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^2}, \end{aligned} \tag{C60}$$

and recurse upward in $\boldsymbol{b}$ and $\boldsymbol{c}$ simultaneously:

$$\begin{aligned} \boldsymbol{b}_v(k^2, \boldsymbol{\alpha}) &= \frac{1}{2v+5}\left(\frac{2v\sin^2\boldsymbol{\alpha}}{1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^2}\right. \\ &\quad \left. \times \boldsymbol{b}_{v-1}(k^2, \boldsymbol{\alpha}) - \boldsymbol{c}_{v-1}(k^2, \boldsymbol{\alpha})\right) \\ \boldsymbol{c}_v(k^2, \boldsymbol{\alpha}) &= \sin^2\boldsymbol{\alpha}\,\boldsymbol{c}_{v-1}(k^2, \boldsymbol{\alpha}) \end{aligned} \tag{C61}$$

for $v > 0$. In the case of downward recursion, we start with the upper boundary conditions

$$\begin{aligned} \boldsymbol{b}_{v_{\max}} &= \frac{\sin^{2v_{\max}+2}\boldsymbol{\alpha}}{4v_{\max}+10}(\boldsymbol{f}_{v_{\max}}(k^2, \boldsymbol{\alpha}) + 2\boldsymbol{q}(k^2, \boldsymbol{\alpha})^3) \\ \boldsymbol{c}_{v_{\max}} &= \frac{1}{2}\boldsymbol{q}(k^2, \boldsymbol{\alpha})^5 \sin^{2v_{\max}}\boldsymbol{\alpha}, \end{aligned} \tag{C62}$$

where

$$\begin{aligned} &\boldsymbol{f}_v(k^2, \boldsymbol{\alpha}) \\ &\equiv \frac{3}{v+1}\,{}_2\boldsymbol{F}_1\left(-\frac{1}{2}, v+1; v+2; 1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^2\right), \end{aligned} \tag{C63}$$

and ${}_2\boldsymbol{F}_1(a, b; c; z)$ is the Gauss hypergeometric function,[71] which we compute via its series definition. We recurse downward in $\boldsymbol{b}$ and $\boldsymbol{c}$ simultaneously:

$$\begin{aligned} \boldsymbol{b}_v(k^2, \boldsymbol{\alpha}) &= \frac{1-\boldsymbol{q}(k^2, \boldsymbol{\alpha})^2}{\sin^2\boldsymbol{\alpha}}\left(1+\frac{5}{2v+2}\right) \\ &\quad \times \boldsymbol{b}_{v+1}(k^2, \boldsymbol{\alpha}) + \frac{\boldsymbol{c}_{v+1}(k^2, \boldsymbol{\alpha})}{v+1} \\ \boldsymbol{c}_v(k^2, \boldsymbol{\alpha}) &= \frac{\boldsymbol{c}_{v+1}(k^2, \boldsymbol{\alpha})}{\sin^2\boldsymbol{\alpha}}. \end{aligned} \tag{C64}$$

C.7. The Term $\mathbb{p}_2$

The final integral we must solve is that corresponding to $\mathbb{p}_2$ ($\mu = \nu = 1$). As in Luger et al. (2019a), this is the integral of the linear limb darkening term, whose solution must be handled separately due to the fact that the corresponding antiderivative in Equation (C25) is not a polynomial in x, y, and $z(x, y)$; also

[66] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/J.ipynb
[67] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/J.ipynb
[68] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/J.ipynb
[69] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/J.ipynb
[70] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/U.ipynb
[71] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/W.ipynb

see Agol et al. (2020). The integral we must solve is

$$\mathbb{p}_2 = \int_\phi \frac{1}{3}\left(\frac{1 - z(r_o\cos\varphi,\, b_o + r_o\sin\varphi)^3}{1 - z(r_o\cos\varphi,\, b_o + r_o\sin\varphi)^2}\right) \times (r_o^2 + b_o r_o \sin\varphi) d\varphi, \tag{C65}$$

where z is the usual Cartesian coordinate (Equation (A6)). The solution is tricky, but fortunately a similar integral was solved in Equation (34) of Pál (2012). Adapting their solution to our formalism, we obtain

$$\mathbb{p}_2 = \frac{1}{3}\left(c_0 + c_1\Delta\boldsymbol{F}\left(\boldsymbol{\alpha}\,\middle|\,\frac{1}{k^2}\right) + c_2\Delta\boldsymbol{E}\left(\boldsymbol{\alpha}\,\middle|\,\frac{1}{k^2}\right) + c_3\Delta\boldsymbol{\Pi}\left(n;\,\boldsymbol{\alpha}\,\middle|\,\frac{1}{k^2}\right)\right), \tag{C66}$$

where[72]

$$\begin{aligned} c_0 &= \Delta\{-\arctan 2(-(b_o - r_o)\cos\boldsymbol{\alpha},\,(b_o + r_o)\sin\boldsymbol{\alpha}) + \boldsymbol{\alpha} \\ &\quad - \frac{4}{3}b_o r_o \sin\boldsymbol{\alpha}\cos\boldsymbol{\alpha}\sqrt{1 - (b_o - r_o)^2 - 4b_o r_o\sin^2\alpha} \\ &\quad + \boldsymbol{\delta}(b_o,\, r_o,\, \boldsymbol{\alpha})\} \\ c_1 &= \frac{1 + b_o^4 - b_o^2(5 + 2r_o^2) + r_o^4 + r_o^2}{3\sqrt{1 - (b_o - r_o)^2}} \\ c_2 &= \frac{(b_o^2 + 7r_o^2 - 4)\sqrt{1 - (b_o - r_o)^2}}{3} \\ c_3 &= \frac{b_o + r_o}{(b_o - r_o)\sqrt{1 - (b_o - r_o)^2}}, \end{aligned} \tag{C67}$$

and[73]

$$\delta(b_o,\, r_o,\, \alpha) = \begin{cases} -2\pi & \alpha > \dfrac{3\pi}{2} \text{ and } b_o > r_o \\ +2\pi & \alpha > \dfrac{3\pi}{2} \text{ and } b_o < r_o \\ 0 & \text{otherwise.} \end{cases} \tag{C68}$$

The quantities $\boldsymbol{F}(\boldsymbol{\alpha}\,|\,1/k^2)$ and $\boldsymbol{E}(\boldsymbol{\alpha}\,|\,1/k^2)$ are the same incomplete elliptic integrals as those in Appendix C.6.2,[74] while

$$\Pi(n;\,\psi\,|\,m) \equiv \int_0^\psi \frac{d\varphi}{(1 - n\sin^2\varphi)\sqrt{1 - m\sin^2\varphi}} \tag{C69}$$

is the incomplete elliptic integral of the third kind, with

$$n = -\frac{4b_o r_o}{(r_o - b_o)^2}. \tag{C70}$$

While stable algorithms exist to evaluate $\Pi(n;\psi\,|\,m)$ (e.g., Bulirsch 1969),[75] we find that the parameterization above has poor numerical stability, particularly in the vicinity of the singular points $b_o = r_o$ and $b_o = 1 + r_o$. In practice, we find that numerical evaluation of Equation (C65) via Gaussian quadrature is more numerically stable and just as computationally efficient as the procedure outlined above.

C.8. The Integral Along the Terminator, $\mathbb{t}^\top$

In this section we present a solution to Equation (C28), the line integral along the day/night terminator of the occulted body. As before, the first thing we must do is derive expressions for the integration limits $\boldsymbol{\xi}$. Depending on the integration case, these limits will corresponds to the point of intersection between the terminator and the limb of the occultor and/or the point of intersection between the terminator and the limb of the occulted body. The former is given by

$$\boldsymbol{\xi_0} = \arctan 2(\sqrt{1 - \boldsymbol{x}''^2},\, \boldsymbol{x}''), \tag{C71}$$

where $\boldsymbol{x}''$ are the roots of the quartic (Equation (C3)).[76] The latter is given by

$$\xi_1 = \begin{cases} 0 & (1 - x_o'')^2 + y_o''^2 < r_o^2 \\ \pi & \text{otherwise.} \end{cases} \tag{C72}$$

As before, these angles are then wrapped to the range $[0,\, 2\pi)$ and sorted into the vector $\boldsymbol{\xi}$ such that the integration is performed counter-clockwise about the center of the integration region.[77] The middle panel of Figure 16 shows a case where $\xi_0 = 96.^\circ5$ corresponds to the point of intersection between the occultor limb and the terminator and $\xi_1 = 0^\circ$ corresponds to the point where the terminator extends onto the backside of the body. Note, importantly, that unlike ϕ, the angle $\boldsymbol{\xi}$ is not measured between the horizontal and a point on the curve of $\mathcal{T}$. Recall that $\boldsymbol{\xi}$ is an angular parameter of the ellipse, so it is measured in the same way as the eccentric anomaly in a Keplerian orbit: it is the angle between the semimajor axis of the ellipse and the perpendicular projection of a point on the ellipse onto the unit circle (see Figure 16).

The solution to Equation (C28) involves repeated application of the binomial theorem. If we define the quantities

$$Z_{j,k}^{u,v}(b,\,\theta') = \binom{u}{j}\binom{v}{k}(-1)^{v-k} b^{u+v-j-k}\sin^{v+j-k}\theta'\cos^{u-j+k}\theta'$$

[72] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/p2.ipynb
[73] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/p2.ipynb
[74] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/p2.ipynb
[75] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/p2.ipynb
[76] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/tT.ipynb
[77] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/tT.ipynb

and

$$\delta(b, \xi) = \begin{cases} 0 & 0 \leqslant \xi < \dfrac{\pi}{2} \\ \pi & \dfrac{\pi}{2} \leqslant \xi < \pi \\ 2|b|b_c & \pi \leqslant \xi < \dfrac{3\pi}{2} \\ \pi + 2|b|b_c & \dfrac{3\pi}{2} \leqslant \xi < 2\pi, \end{cases} \quad , \tag{C73}$$

we may express the solution[78] to the $\mathbb{t}_n$ integral as[79]

$$\mathbb{t}_n = \begin{cases} b\cos\theta' \sum_{j=0}^{\frac{\nu}{2}} \sum_{k=0}^{\frac{\mu+2}{2}} Z_{j,k}^{\frac{\nu}{2},\frac{\mu+2}{2}}(b, \theta')\, \mathbb{H}_{j+k+1,l+1-j-k}(\boldsymbol{\xi}) & \\ -\sin\theta' \sum_{j=0}^{\frac{\nu}{2}} \sum_{k=0}^{\frac{\mu+2}{2}} Z_{j,k}^{\frac{\nu}{2},\frac{\mu+2}{2}}(b, \theta')\, \mathbb{H}_{j+k,l+2-j-k}(\boldsymbol{\xi}) & \mu, \nu, \text{ even} \\ \frac{1}{3}\Delta\left\{\arctan\left(\frac{|b|\sin\boldsymbol{\xi}}{\cos\boldsymbol{\xi}}\right)\right. & \\ \quad \left. - \operatorname{sgn}(\sin\boldsymbol{\xi})\left(\arctan\left(\frac{\left(\frac{\sin\boldsymbol{\xi}}{1+\cos\boldsymbol{\xi}}\right)^2 + 2b^2 - 1}{2|b|b_c}\right) + |b|b_c\cos\boldsymbol{\xi}\right)\right. & \mu = \nu = 1 \\ \left. + \boldsymbol{\delta}(b, \boldsymbol{\xi})\right\} & \\ -bb_c^3 \sin\theta' \sum_{j=0}^{l-2} Z_{0,j}^{0,l-2}(b, \theta')\, \mathbb{H}_{j+1,l+1-j}(\boldsymbol{\xi}) & \\ -b_c^3 \cos\theta' \sum_{j=0}^{l-2} Z_{0,j}^{0,l-2}(b, \theta')\, \mathbb{H}_{j,l+2-j}(\boldsymbol{\xi}) & \nu \text{ odd, } \mu = 1, l \text{ even} \cdot \\ -bb_c^3 \sin^2\theta' \sum_{j=0}^{l-3} Z_{0,j}^{0,l-3}(b, \theta')\, \mathbb{H}_{j+2,l-j}(\boldsymbol{\xi}) & \\ -bb_c^3 \cos^2\theta' \sum_{j=0}^{l-3} Z_{0,j}^{0,l-3}(b, \theta')\, \mathbb{H}_{j,l+2-j}(\boldsymbol{\xi}) & \nu \text{ odd, } \mu = 1, l \text{ odd} \\ -b_c^5 \sin\theta'\cos\theta' \sum_{j=0}^{l-3} Z_{0,k}^{0,l-3}(b, \theta')\, \mathbb{H}_{j+1,l+1-j}(\boldsymbol{\xi}) & \\ bb_c^3 \cos\theta' \sum_{j=0}^{\frac{\nu}{2}} \sum_{k=0}^{\frac{\mu+2}{2}} Z_{j,k}^{\frac{\nu-1}{2},\frac{\mu-3}{2}}(b, \theta')\, \mathbb{H}_{j+k+1,l+1-j-k}(\boldsymbol{\xi}) & \\ -b_c^3 \sin\theta' \sum_{j=0}^{\frac{\nu}{2}} \sum_{k=0}^{\frac{\mu+2}{2}} Z_{j,k}^{\frac{\nu-1}{2},\frac{\mu-3}{2}}(b, \theta')\, \mathbb{H}_{j+k,l+2-j-k}(\boldsymbol{\xi}) & \text{otherwise} \end{cases} \tag{C74}$$

The solution to Equation (C74) depends on the matrix $\mathbb{H}$, whose components are given by the integral

$$\mathbb{H}_{u,v}(\boldsymbol{\xi}) = \int_{\boldsymbol{\xi}} \cos^u\varphi \sin^v\varphi d\varphi. \tag{C75}$$

The $\mathbb{H}$ integral is the same as that in Equation (D27) of Luger et al. (2019a), except for a change in the limits of integration. We can compute this integral recursively given four lower boundary conditions:

$$\begin{aligned} \mathbb{H}_{0,0}(\boldsymbol{\xi}) &= \Delta\boldsymbol{\xi} \\ \mathbb{H}_{1,0}(\boldsymbol{\xi}) &= \Delta\sin\boldsymbol{\xi} \\ \mathbb{H}_{0,1}(\boldsymbol{\xi}) &= -\Delta\cos\boldsymbol{\xi} \\ \mathbb{H}_{1,1}(\boldsymbol{\xi}) &= -\frac{\Delta\cos^2\boldsymbol{\xi}}{2}. \end{aligned} \tag{C76}$$

The remaining terms may be computed[80] by upward recursion using the relations

$$\mathbb{H}_{u,v}(\boldsymbol{\xi}) = \frac{-\Delta(\cos^{u+1}\boldsymbol{\xi}\sin^{v-1}\boldsymbol{\xi}) + (v-1)\mathbb{H}_{u,v-2}(\boldsymbol{\xi})}{u+v} \tag{C77}$$

[78] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/tT.ipynb

[79] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/tT.ipynb

[80] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/H.ipynb

for $u < 2$, $v \geqslant 2$ and

$$\mathbb{H}_{u,v}(\boldsymbol{\xi}) = \frac{\Delta(\cos^{u-1}\boldsymbol{\xi}\sin^{v+1}\boldsymbol{\xi}) + (u-1)\mathbb{H}_{u-2,v}(\boldsymbol{\xi})}{u+v} \quad \text{(C78)}$$

for all remaining terms.[81,82]

C.9. The Integral Along the Occulted Body Limb, $\mathbf{q}^\top$

The final line integral we must solve is the integral along the boundary of the occulted body, Equation (C29). Fortunately, this is also the easiest of the three. The limits of integration $\boldsymbol{\lambda}$ correspond to the point at which the terminator crosses from the dayside to the nightside,

$$\lambda_0 = \begin{cases} \theta' & \cos^2\theta' + (\sin\theta' - b_o)^2 < r_o^2 \\ \theta' + \pi & \text{otherwise.} \end{cases} \quad \text{(C79)}$$

and the point of intersection between the limb of the occultor and the limb of the occulted body,[83]

$$\lambda_1 = \frac{\pi}{2} \pm \left(\arcsin\left(\frac{1 - r_o^2 + b_o^2}{2b_o}\right) - \frac{\pi}{2}\right), \quad \text{(C80)}$$

where the sign is chosen such that the point $(\cos\lambda_1, \sin\lambda_1)$ is on the dayside of the occulted body.[84] As before, these angles are wrapped to the range $[0, 2\pi)$ and placed in the vector λ such that the line integral is taken in the counter-clockwise direction about the center of the integration region. The right panel of Figure 16 shows a case where $\lambda_0 = 75°$ and $\lambda_1 = 130.°5$. Both angles are measured counter-clockwise from the x'-axis.

Given $\boldsymbol{\lambda}$, the solution to Equation (C29) is straightforward:

$$\mathbf{q}_n = \begin{cases} \mathbb{H}_{\frac{\mu+4}{2},\frac{\nu}{2}}(\boldsymbol{\lambda}) & \mu, \nu \text{ even} \\ \frac{1}{3}\Delta\boldsymbol{\lambda} & \mu = \nu = 1 \\ 0 & \text{otherwise,} \end{cases} \quad \text{(C81)}$$

where the matrix $\mathbb{H}$ is given by Equation (C75).[85]

Appendix D
Caveats

The starry code includes a large suite of unit tests that compare the flux computations to numerical models and to various benchmarks for a wide variety of inputs. While we have done our best to develop tests over the full range of occultation configurations, there may be edge cases in which the starry algorithm fails and returns the wrong flux. This could happen, for instance, if the quartic root solver (Appendix C.1) fails to find the points of intersection between the occultor and the day/night terminator, leading to an incorrect case identification (Figure 15) and thus the wrong value for the flux. In the development of the algorithm, these cases would occasionally show up as a single, obvious outlier in a light-curve model. All such cases we encountered have been fixed by adding consistency checks in the root solver and switching to alternate evaluation methods near known singularities. However, it is possible that there may still be rare cases in which this happens, in which case we ask that users raise an issue[86] on GitHub so that we can provide a fix.

[81] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/H.ipynb
[82] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/H.ipynb
[83] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/qT.ipynb
[84] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/qT.ipynb
[85] https://github.com/rodluger/starrynight/blob/5f79b24026133db207a88bee465e5e5080b64bcf/tex/proofs/qT.ipynb
[86] https://github.com/rodluger/starry/issues

ORCID iDs

Rodrigo Luger https://orcid.org/0000-0002-0296-3826
Eric Agol https://orcid.org/0000-0002-0802-9145
Fran Bartolić https://orcid.org/0000-0001-8630-9794
Daniel Foreman-Mackey https://orcid.org/0000-0002-9328-5652

References

Agol, E., Luger, R., & Foreman-Mackey, D. 2020, AJ, 159, 123
Ahlers, J. P., Johnson, M. C., Stassun, K. G., et al. 2020, AJ, 160, 4
Aizawa, M., Kawahara, H., & Fan, S. 2020, ApJ, 896, 22
Aksnes, K., Franklin, F., Millis, R., et al. 1984, AJ, 89, 280
Arlot, J. E., Camichel, H., & Link, F. 1974, A&A, 35, 115
Arlot, J. E., Emelyanov, N., Varfolomeev, M. I., et al. 2014, A&A, 572, A120
Arnold, L., & Schneider, J. 2004, A&A, 420, 1153
Asensio Ramos, A., & Pallé, E. 2021, A&A, 646, A4
Barnes, J. W., Soderblom, J. M., Brown, R. H., et al. 2011, Icar, 211, 722
Bartolić, F., Luger, R., Foreman-Mackey, D., Howell, R. R., & Rathbun, J. A. 2022, PSJ, 3, 67
Becker, T., & Geissler, P. 2005, in 36th Annual Lunar and Planetary Science Conf. (Houston, TX: LPI), 1862
Berdyugina, S. V., & Kuhn, J. R. 2017, arXiv:1711.00185
Buie, M. W., Tholen, D. J., & Horne, K. 1992, Icar, 97, 211
Bulirsch, R. 1965, NuMat, 7, 78
Bulirsch, R. 1969, NuMat, 13, 305
Cabrera, J., & Schneider, J. 2007, A&A, 464, 1133
Carter, J. L. 2019, arXiv:1901.01361
Cowan, N. B., & Agol, E. 2008, ApJL, 678, L129
Cowan, N. B., Agol, E., Meadows, V. S., et al. 2009, ApJ, 700, 915
Cowan, N. B., Robinson, T., Livengood, T. A., et al. 2011, ApJ, 731, 76
de Kleer, K., Skrutskie, M., Leisenring, J., et al. 2017, Natur, 545, 199
de Wit, J., Gillon, M., Demory, B.-O., & Seager, S. 2012, A&A, 548, A128
Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. 1987, PhLB, 195, 216
Dunbar, R. S., & Tedesco, E. F. 1986, AJ, 92, 1201
Edelman, A., & Murakami, H. 1995, MaCom, 64, 763
Fan, S., Li, C., Li, J.-Z., et al. 2019, ApJL, 882, L1
Farr, B., Farr, W. M., Cowan, N. B., Haggard, H. M., & Robinson, T. 2018, AJ, 156, 146
Fienup, J. R. 1978, Proc. SPIE, 149, 72
Folkner, W. M., Williams, J. G., Boggs, D. H., Park, R. S., & Kuchynka, P. 2014, IPNPR, 42, 196
Ford, E. B., Seager, S., & Turner, E. L. 2001, Natur, 412, 885
Foreman-Mackey, D., Luger, R., Czekala, I., et al. 2020, exoplanet-dev/exoplanet: exoplanet v0.3.2, Zenodo, doi:10.5281/zenodo.3785072
Fujii, Y., & Kawahara, H. 2012, ApJ, 755, 101
Ginsburg, A., Robitaille, T., Parikh, M., et al. 2013, doi:10.6084/m9.figshare.805208.v2
Ginsburg, A., Sipöcz, B. M., Brasseur, C. E., et al. 2019, AJ, 157, 98
Górski, K. M., Hivon, E., Banday, A. J., et al. 2005, ApJ, 622, 759
Guennebaud, G., & Jacob, B. 2010, Eigen v3, http://eigen.tuxfamily.org
Haggard, H. M., & Cowan, N. B. 2018, MNRAS, 478, 371
Heng, K., Morris, B. M., Kitzmann, D., et al. 2021, NatAs, 5, 1001
Hoffman, M. D., & Gelman, A. 2011, arXiv:1111.4246
Hughes, G. B., & Chraibi, M. 2011, arXiv:1106.3787
Jakob, W., Rhinelander, J., Moldovan, D., et al. 2017, pybind11: Seamless Operability between C++11 and Python, v2.2, GitHub, https://github.com/pybind/pybind11
Jiang, J. H., Zhai, A. J., Herman, J., et al. 2018, AJ, 156, 26
Jones, E., Oliphant, T., Peterson, P., et al. 2001, SciPy: Open Source Scientific Tools for Python, v1.0.0, http://www.scipy.org/

Kawahara, H. 2020, ApJ, 894, 58
Kawahara, H., & Fujii, Y. 2010, ApJ, 720, 1333
Kawahara, H., & Fujii, Y. 2011, ApJL, 739, L62
Kawahara, H., & Masuda, K. 2020, ApJ, 900, 48
Knuth, K., Placek, B., Angerhausen, D., et al. 2017, Entrp, 19, 559
Knutson, H. A., Charbonneau, D., Allen, L. E., et al. 2007, Natur, 447, 183
Kopal, Z. 1954, MNRAS, 114, 101
Lacis, A. A., & Fix, J. D. 1972, ApJ, 174, 449
Livengood, T. A., Deming, L. D., A'Hearn, M. G., et al. 2011, AsBio, 11, 907
Louden, T., & Kreidberg, L. 2018, MNRAS, 477, 2613
Luger, R., Agol, E., Foreman-Mackey, D., et al. 2018, STARRY: Analytic Computation of Occultation Light Curves v1.0.0, Astrophysics Source Code Library, ascl:1810.005
Luger, R., Agol, E., Foreman-Mackey, D,, et al. 2019a, AJ, 157, 64
Luger, R., Bedell, M., Vanderspek, R., & Burke, C. J. 2019b, arXiv:1903.12182
Luger, R., Bedell, M., Foreman-Mackey, D., et al. 2021a, arXiv:2110.06271
Luger, R., Foreman-Mackey, D., & Hedges, C. 2021b, AJ, 162, 124
Luger, R., Foreman-Mackey, D., Hedges, C., & Hogg, D. W. 2021c, AJ, 162, 123
Luger, R., Lustig-Yaeger, J., & Agol, E. 2017, ApJ, 851, 94
Lustig-Yaeger, J., Meadows, V. S., Tovar Mendoza, G., et al. 2018, AJ, 156, 301
Majeau, C., Agol, E., & Cowan, N. B. 2012, ApJL, 747, L20
Mandel, K., & Agol, E. 2002, ApJL, 580, L171
Marcialis, R. L. 1990, The Pluto–Charon System as Revealed During the Mutual Events (Tucson, AZ: Univ. of Arizona), 1224
Morgado, B., Vieira-Martins, R., Assafin, M., et al. 2019, P&SS, 179, 104736
Oakley, P. H. H., & Cash, W. 2009, ApJ, 700, 1428
Oren, M., & Nayar, S. K. 1994, in Proc. 21st Annual Conf. on Computer Graphics and Interactive Techniques, SIGGRAPH'94, ed. D. Schweitzer, A. Glassner, & M. Keeler (New York: Association for Computing Machinery), 239
Pál, A. 2012, MNRAS, 420, 1630
Rauscher, E., Menou, K., Seager, S., et al. 2007, ApJ, 664, 1199
Rauscher, E., Suri, V., & Cowan, N. B. 2018, AJ, 156, 235
Reinsch, K., Burwitz, V., & Festou, M. C. 1994, Icar, 108, 209
Robinson, T. D., Ennico, K., Meadows, V. S., et al. 2014, ApJ, 787, 171
Russell, H. N. 1906, ApJ, 24, 1
Salvatier, J., Wiecki, T. V., Fonnesbeck, C., et al. 2016, PeerJ Comp. Sci., 2, e55
Saquet, E., Emelyanov, N., Robert, V., et al. 2018, MNRAS, 474, 4730
Schwartz, J. C., Sekowski, C., Haggard, H. M., Pallé, E., Cowan, N. B., et al. 2016, MNRAS, 457, 926
Seager, S. 2010, Exoplanet Atmospheres: Physical Processes (Princeton, NJ: Princeton Univ. Press)
Stern, S. A. 1992, ARA&A, 30, 185
Theano Development Team 2016, arXiv:1605.02688
Tholen, D. J., & Buie, M. W. 1990, BAAS, 22, 1129
Williams, P. K. G., Charbonneau, D., Cooper, C. S., Showman, A. P., & Fortney, J. J. 2006, ApJ, 649, 1020
Wong, I., Shporer, A., Morris, B. M., et al. 2019, AJ, 160, 188