Exoplanet Reflected-light Spectroscopy with PICASO

Natasha E. Batalha1, Mark S. Marley2, Nikole K. Lewis3, and Jonathan J. Fortney1

1 Department of Astronomy and Astrophysics, University of CaliforniaSanta Cruz, Santa Cruz, CA 95064, USA; nbatalha@ucsc.edu
2 NASA Ames Research Center, MS 245-3, Moffett Field, CA 94035, USA
3 Department of Astronomy and Carl Sagan Institute, Cornell University, 122 Sciences Drive, Ithaca, NY 14853, USA

Received 2019 March 14; revised 2019 April 11; accepted 2019 April 12; published 2019 June 14

Abstract
Here we present the first open-source radiative transfer model for computing the reflected light of exoplanets at any phase geometry, called PICASO: the planetary intensity code for atmospheric scattering observations. This code, written in Python, has heritage from a decades-old, well-known Fortran model used for several studies of planetary objects within the solar system and beyond. We have adopted it to include several methodologies for computing both direct and diffuse-scattering phase functions, and have added several updates including the ability to compute Raman scattering spectral features. Here we benchmark PICASO against two independent codes and discuss the degree to which the model is sensitive to a user’s specification for various phase functions. Then, we conduct a full information-content study of the model across a wide parameter space in temperature, cloud profile, signal-to-noise ratio, and resolving power.

Key words: planetary systems – techniques: spectroscopic

1. Introduction

Across all the state-of-the-art pipelines that exist to study atmospheric composition and climate from exoplanets, about half a dozen have been developed for transit science, a few of which are open-source (e.g., Madhusudhan & Seager 2009; Benneke & Seager 2012; Line et al. 2012; Waldmann et al. 2015; Barstow et al. 2017; Zhang et al. 2019). This abundance of model development has overall improved the quality of all of these models, has contributed to interesting model intercomparison studies (e.g., Baudino et al. 2017), and increased accessibility of traditionally private codes.

On the other hand, for observations of reflected light from directly imagined exoplanets, there have only been two such pipelines, neither of which is open source (Lupu et al. 2016; Lacy et al. 2019). An additional branch of models, used for exoplanet Earth science, also exists to compute reflected light from planetary atmospheres (NEMESIS, Irwin et al. 2008; DISORT, Stamnes et al. 1988; PSG, Villanueva et al. 2018). NEMESIS is well vetted, and has been used for retrieving the composition from dozens of observations of Jupiter (e.g., Irwin et al. 2019a), Neptune (e.g., Irwin et al. 2019b), Titan (e.g., Thelen et al. 2019), and many more. DISORT is an open-source forward model, but cannot be used to retrieve atmospheric composition. Note that all retrieval models consist of a versatile and fast-forward model that can be wrapped in a statistical algorithm. DISORT contains several hard-wired assumptions for terrestrial conditions and is not versatile or fast enough to use in a retrieval framework. PSG is the retrieval tool of the ExoMars mission and has been used for other investigations such as Earth and the NASA Infrared Telescope Facility. Last, ray-tracing forward models, such as that of Dyudina et al. (2016), would also be computationally intensive and complex to wrap into a retrieval framework for exoplanets.

With the detection and analysis of reflected light from optical phase curves (Demory et al. 2013; Esteves et al. 2015; Niraula et al. 2018) and optical photometry (Evans et al. 2013; Barstow et al. 2014; Garcia Munoz & Isaak 2015; Webber et al. 2015; Lee et al. 2017), and with the onset of reflected-light direct-imaging missions on the horizon, such has the Wide Field Infrared Survey Telescope (WFIRST) and the European Extremely Large Telescope (ELTs), Spiegel et al. (2013), there has been an increasing demand for an accessible, versatile reflected-light code.

Here, we present the planetary intensity code for atmospheric scattering observations (PICASO). It is available through Github,4 and can be installed through pip or conda. Tutorials for running the code are available online5 along with an in-depth physics tutorial for the derivation of the radiative transfer of the code.6

1.1. The Heritage of the Code

The methodology of PICASO partly originates from the Fortran albedo spectra model described in several studies of planetary objects within the solar system (McKay et al. 1989; Marley & McKay 1999) and beyond (Marley et al. 1999). These models used radiative transfer methods described in Toon et al. (1977, 1989) and only included the capability to compute monochromatic scattered radiation observed at full phase. Later, Cahoy et al. (2010) introduced the capability to compute the monochromatic scattered radiation observed at any phase angle. Since then, the model has been widely used in several studies of exoplanets, including retrievals of exoplanet atmospheres (Lupu et al. 2016; Nayak et al. 2017), sulfur hazes in giant exoplanet atmospheres (Gao et al. 2017), Earth analogs in reflected light (Feng et al. 2018), water absorption in cool giants (MacDonald et al. 2018), and color classification of directly imaged exoplanets (Batalha et al. 2018b), among others.

While the code bifurcated across several of these analyses (e.g., updates to molecular opacities, various ways to regrid the atmosphere, varying phase functions, different sources of scattering and cloud opacity) the bulk of the radiative transfer in the code has remained relatively similar since the original publications of McKay et al. (1989), Marley et al. (1999), and Cahoy et al. (2010). Individual changes for each analysis lack

---

4 Github (https://github.com/natashabatalha/picaso).
5 Code tutorial (https://natashabatalha.github.io/picaso).
6 Physics tutorial (https://natashabatalha.github.io/picaso_dev).
Exoplanet atmospheres exhibit a diverse range in optical properties. and single scattering of a cirrus cloud, and the volcano icon represents the same of aged volcanic aerosols (Thomas & Stamnes 2002). Main point: exoplanet atmospheres exhibit a diverse range in optical properties.

1.2. Exoplanet Diversity and the Need for Versatility

At the foundation of any reflected-light code is an assumption of a scattering phase function, \( p(\cos \Theta) \), used to describe the angular dependence of how light is scattered. Assumptions of phase functions vary widely in complexity (Hansen 1969). The most simplistic assumption is an isotropic scattering phase function. In this case, there is equal probability of arriving photons traveling to scatter in any given direction. Of course, scattering by gases and particles is not isotropic. To account for more realistic phase functions, an asymmetry parameter, \( g \), is usually introduced and used with more complex phase functions. It is important to note that asymmetry values vary widely depending on the specific optical properties of the condensing species (e.g., Morley et al. 2012).

Figure 1 shows distributions of asymmetry parameters and single-scattering albedos from a wide range of giant planet models computed in Batalha et al. (2018b; data are available through Batalha et al. 2018c). All cloud models were computed using the prescription of Ackerman & Marley (2001) for a Jupiter-like system (at solar metallicity, \( 25 \text{ ms}^{-2} \)), with varying semimajor axes (i.e., equilibrium temperatures). For reference, the asymmetry parameters of well-studied cirrus clouds and volcanic aerosols are also shown (Thomas & Stamnes 2002).

At 5 au from a Sun-like star, \( \text{H}_2\text{O}/\text{NH}_3 \) clouds dominate the optical behavior, leading to high asymmetry values/single-scattering albedo. At hotter temperatures, more exotic cloud species such as \( \text{ZnS} \) and \( \text{Na}_2\text{S} \) begin to widen and decrease the distribution of asymmetry values/single-scattering albedos. Exoplanet atmospheres, which cover a broad range in mass and temperature space, exhibit a wide range of optical properties.

To emphasize how these wide ranges of optical properties propagate to the behavior of the phase function, Figure 2 shows a typical phase function computed for a range in asymmetry parameters, \( g \).

The extreme range in these differences motivated the design of the new code. We aimed to create a code where fundamental radiative transfer assumptions, such as that of the phase function, could be easily assessed. We hope that this will guide development of future facilities, and facilitate a symbiotic relationship between future observations and the improvement of theoretical models of directly imaged exoplanets.

1.3. Organization

In what follows we describe the methodology of PICASO in Section 2, with a special emphasis on the new physics and capabilities that have been introduced. Then we analyze the main assumptions made in our calculations of the reflected light in Section 3 in order to show which assumptions the calculations are most sensitive to across a large parameter space in planet temperature, cloud composition, and stellar type. In Section 4 we validate PICASO against two different independent calculations. Then, given the most recent specifics of a future space-based direct-imaging mission, we use PICASO for a full information-content (IC) analysis across signal-to-noise ratio (S/N) and resolving power. Here, we specifically focus on our ability to constrain atmospheric composition and gravity. We end with a discussion and conclusion in Section 6.
2. PICASO: The Forward Model

A full derivation of the radiative transfer of the forward model can be found online.\(^7\) As is with any atmospheric scattering code, we begin with the radiative transfer equation (Goody & Yung 1989),

\[
I(\tau, \mu) = I(\tau_{i+1}, \mu) e^{g/\mu} - \int_{0}^{\delta \tau} S(\tau', \mu) e^{-g/\mu} d\tau'/\mu. \tag{1}
\]

Here, the terms are as follows:

1. \(I(\tau, \mu)\): the azimuthally averaged intensity emergent from the top of an atmospheric layer, \(i\), with opacity, \(\tau\), and outgoing angle, \(\mu\).
2. \(I(\tau_{i+1}, \mu) e^{g/\mu}\): the incident intensity on the lower boundary of the layer attenuated by the optical depth within the layer, \(\delta \tau\).
3. \(S(\tau', \mu)\): the source function, integrated over all layers.

In our formalism, the source function only consists of two components: (1) the single-scattered radiation, and (2) the multiple-scattered radiation, integrated over all diffuse angles. In other words, we do not include a thermal term in the source function. Traditionally, dating back to Toon et al. (1989), the thermal and reflected-light terms have been computed separately. We leave the addition of the thermal component to a future update so that the source function has the form

\[
S(\tau', \mu) = \frac{\omega}{4\pi} F_0 P_{\text{single}}(\mu, -\mu_0) e^{-\tau'/\mu},
\]

\[
\quad + \frac{\omega}{2} \int_{-1}^{1} I(\tau', \mu') P_{\text{multi}}(\mu, \mu') d\mu' \tag{2}
\]

where the first term is the single-scattered radiation, whose behavior is described by the phase function \(P_{\text{single}}(\mu, -\mu_0)\), and the second term is the multiple-scattered radiation, whose behavior is described by \(P_{\text{multi}}(\mu, \mu')\). A schematic of the plane-parallel model is shown in Figure 3.

In addition to basic planetary properties (e.g., stellar spectrum, planet mass, and radius), PICASO takes in as input: (1) a pressure-temperature profile and altitude-dependent abundances (see https://natashabatalha.github.io/picaso/notebooks/1_GetStarted.html, justdoit.atmosphere()), and (2) a cloud profile (single-scattering albedo, asymmetry parameter, and total extinction; see https://natashabatalha.github.io/picaso/notebooks/2_AddingClouds.html, justdoit.clouds()). As further shown in the tutorial, the cloud profile can either be input as a full altitude-dependent profile parameterized or generated from a model such as that of Ackerman & Marley (2001), or it can be input as different cloud layers that are arbitrarily set by additionally supplying the cloud top pressures and vertical extent of each layer. PICASO is designed to accommodate several different input styles in order to be highly customizable for each user.

Our methodology is thoroughly described in Cahoy et al. (2010; see Section 3.2). In short, we follow the source function method in Toon et al. (1989). We first use the two-stream quadrature to solve for the diffuse scattered radiation. Then, we use the resulting two-stream intensity to approximate the source function. There are several other methods of solving this, e.g., the δ-M stream method (Wiscombe 1977), which we will explore in a future release of the code.

PICASO includes several ways of handling the single- and multiple-scattering phase functions, as compared with Cahoy et al. (2010). Additionally, PICASO has been written to include a more physically motivated methodology for Raman scattering. Therefore, we devote Sections 2.1, 2.2, and 2.4 to these specific components, and we refer to Cahoy et al. (2010) for an explanation of the boundary condition formalism, which has not been altered. Finally, in Sections 2.5 and 2.6 we derive the methodology for computing various types of albedos and the planet phase geometry, respectively.

2.1. The Single-scattering Component

For the direct- or single-scattering component, the most widely used form is the Henyey–Greenstein (HG) phase function because it is a function of \(g\), generally resembles “real” phase functions, and is non-negative for all values of \(\Theta\). The one-term HG phase function has the form

\[
p_{\text{HG}} = \frac{1 - g^2}{(1 + g^2 - 2g \cos \Theta)^{3/2}}. \tag{3}
\]

Here, \(g\) is the asymmetry parameter, which is defined as

\[
g = \frac{1}{4\pi} \int_{4\pi} p(\cos \Theta) \cos \Theta d\omega, \tag{4}
\]

where \(\Theta\) is the angle between the original direction and the scattered direction (related to the planet’s phase, \(\alpha\) function via \(\alpha = \pi - \Theta\)). By defining this parameter, we only need to determine the relative proportion of photons that are scattered in the forward versus backward direction (as opposed to each individual intensity).

The asymmetry parameter \(g\) can be any value \(-1 \leq g \leq 1\). In the limit when \(g = 1\), photons approximately continue traveling in their original direction; when \(g = -1\), their directions are reversed; and when \(g = 0\), they are equally likely to travel in the forward or backward direction (i.e., isotropy).

Using Equation (3) for the single-scattering phase function is easily accessed in PICASO,\(^8\) but it is not the default. This is because although Equation (3) captures the observed forward peak relatively well, it fails to capture the additional (but smaller) backward-scattering peak that has been observed on the Moon, Mars, Venus, and Jupiter (Sudarsky et al. 2005). To

\(^7\) https://natashabatalha.github.io/picaso_dev

\(^8\) See OTHG in https://natashabatalha.github.io/picaso/notebooks/4_AnalyzingApproximations.html#What-approximations-exist?
account for this, a second term in the phase function can be introduced,

$$P_{\text{TTHG}}(\cos \Theta) = f P_{\text{OTHG}}(\cos \Theta, g_f) + \left(1 - f\right) P_{\text{OTHG}}(\cos \Theta, g_b),$$

(5)

Here, in addition to having two asymmetry factors ($g_f$ for the forward and $g_b$ for the backward), we also have a new parameter, $f$, which describes the fraction of forward to back scattering. In PICASO, we give $f$ the functional form of

$$f = c_1 + c_2 g_c^2,$$

(6)

where users can specify $c_1$, $c_2$, and $c_3$. By default, PICASO, sets $g_f = \bar{g}$, $g_b = -\bar{g}/2$, and $f = 1 - \bar{g}^2$. $\bar{g}$ is the cloud asymmetry factor that is computed directly from the cloud code eddysed, weighted by the contribution of cloud opacity ($\bar{g} = g_{\text{cld}} \tau_{\text{cld}}/\tau_{\text{scat}}$) (Ackerman & Marley 2001).

However, these values are certainly not universal. Jupiter, Saturn, Uranus, and Neptune all exhibit slightly different forward- and back-scattering peaks (Sudarsky et al. 2005; Dydudina et al. 2016). For observations of exoplanets, these parameters will have to be fit for.

The last component to consider is the effect of Rayleigh scattering, which acts to increase the back-scattering peak. The Rayleigh phase function has the form

$$P_{\text{Ray}}(\cos \Theta) = \frac{3}{4} (1 + \cos^2 \Theta).$$

(7)

In order to incorporate both Rayleigh and the cloud scattering properties, we combine TTHG and Equation (7) by weighting the two phase functions by the fractional opacity of each. In other words, if $\tau_{\text{cld}}$ is the contribution of scattering from clouds, $\tau_{\text{Ray}}$ is the contribution of scattering from Rayleigh, and $\tau_{\text{scat}}$ is the total scattering, the phase function takes the form

$$P_{\text{TTHG-ray}} = \frac{\tau_{\text{cld}}}{\tau_{\text{scat}}} P_{\text{TTHG}} + \frac{\tau_{\text{Ray}}}{\tau_{\text{scat}}} P_{\text{Ray}}.$$  

(8)

This methodology was used in Feng et al. (2018), and a similar methodology was employed in Cahoy et al. (2010). It is also the default methodology used in PICASO. Table 1 has a full summary of the single-scattering methodology.

2.2. Multiple-scattering Component

We cannot use the same forms for phase functions, as we did in Section 2.1, because the multiple-scattering component of the source function (Equation (2)) must be integrated over all diffuse angles, $\mu$.

An additional convenience of the HG phase function is that we can mathematically write it as a series of Legendre polynomials,

$$P_{\text{multi}}(\cos \Theta) \approx \sum_{l=0}^{N-1} \beta_l P_l(\cos \Theta),$$

(9)

where $P_l(\cos \Theta)$ are the polynomials (not to be confused with another phase function), and $\beta_l$ are the moments of the phase function. The moments can be written out as

$$\beta_l = \frac{2l + 1}{2} \int_{-1}^{1} P_l(\cos \Theta) p(\cos \Theta) d \cos \Theta,$$

(10)

which should look familiar, given the previous equation shown for the asymmetry factor (see Equation (4)). Therefore, the moment can just be simplified to $\beta_l = (2l + 1)g_l$.

Expanding this polynomial to simply an $N = 1$ expansion gives us

$$P_{\text{multi}}(\cos \Theta) = 1 + 3\bar{g} \cos \Theta = 1 + 3\bar{g} \mu \mu'$$

(11)

where an azimuthal independence assumption can reduce $\cos \Theta = \mu \mu'$. One drawback of this formalism (similar to the OTHG) is that it fails to capture the behavior of Rayleigh scattering (because $\bar{g}$ is zero for Rayleigh). Therefore, many authors (starting with Snook 1999) have leveraged the fact that the second-order Legendre dependence on $\cos^2 \Theta$ is the same as that of Rayleigh (see Equation (7)). Therefore, by forcing the second moment $\beta_2 = (2l + 1)g_l = g_2$ to yield the Rayleigh phase function, we can accurately account for Rayleigh scattering. We set this new parameter, $g_2 = \tau_{\text{Ray}}/(2\tau_{\text{scat}})$, so that when Rayleigh dominates the total opacity, $g_2$ approaches 1/2, the correct value for Rayleigh (Hansen & Travis 1974).

For multiple-scattering, the two options for PICASO are $N = 1$ and $N = 2$ expansions ($N = 2$ being the default). However, we can also add the $\delta$-Eddington methodology (explained in the following section) to further improve the accuracy.

2.3. $\delta$-Eddington

Low-order Legendre expansions are not adequate enough to represent very high forward-scattering (which is very asymmetric). Given the high scattering asymmetry produced by Mie-scattering particles with sizes larger than typical optical wavelengths (Figure 1), this could be problematic. In order to make lower order approximations more accurate, PICASO leverages the $\delta$-Eddington approximation (Joseph et al. 1976). In this approximation, $g$, $\tau$, $\omega$ (the single-scattering albedo) are all scaled by recognizing that a beam that experiences a high degree of forward-scattering from high-albedo particles

| Key          | Formalism         | Inputs Required                  | Pro/Con                  |
|--------------|-------------------|----------------------------------|--------------------------|
| OTHG         | Equation (3)      | $g$                              | Does not capture back scattering. |
| TTHG         | Equation (5)      | $g_f$, $g_b$, $c_1$, $c_2$, $c_3$ | Captures small back peak.  |
| TTHG-Ray*    | Equation (8)      | $g_f$, $g_b$, $c_1$, $c_2$, $c_3$, $\tau_{\text{Ray}}$, $\tau_{\text{cld}}$ | Captures sharper back scattering caused by Rayleigh. |

*PICASO default.
increases, approximations to the TTHG phase function become progressively worse.

Figure 4 shows a full comparison of all the multiple-scattering phase functions. Note that the \( N = 2 \) expansion with \( \delta \)-scaling reduces the forward peak to regions where the Legendre polynomials should be in higher agreement with the TTHG (i.e., lower asymmetry).

### 2.4. Raman Scattering

A small fraction of photons that are scattered via the well-known Rayleigh process experience a shift to redder wavelengths that is caused by the excitation of rotational and vibration transitions in atmospheric gases. Although some incident stellar photons are shifted, we normalize by the original incident flux when we compute the albedo (see Equation (15)). This discrepancy between the shifted incident radiation and the original incident radiation creates new spectral features in the albedo calculation. These detectable shifts are called “ghost” features in the reflected-light spectra of planetary atmospheres (Price 1977).

Raman scattering has been detected in the reflected light of all solar system gas giants (e.g., Yelle et al. 1987; Karkoschka 1994; Courtin 1999). Recently, it was also suggested that Raman scattering could be an important indicator of the main spectroscopic scatterer in the atmospheres of exoplanets, such as \( \text{H}_2 \) versus \( \text{N}_2 \) (Oklopčić et al. 2016). Additionally, it was shown that varying the stellar spectra will have a non-negligible affect on the reflected light of exoplanets (Oklopčić et al. 2017).

For the studies of exoplanets, the effect of Raman scattering has been approximated using the methodology of Pollack et al. (1986; e.g., Marley et al. 1999; Sudarsky et al. 2005; Cahoy et al. 2010). All of these analyses computed Raman-scattering correction terms for a 6000 K blackbody. The Pollack et al. (1986) approximation captures the overall shape of Raman scattering by \( \text{H}_2 \) (i.e., decreased reflectively toward the blue). However, it fails to capture specific ghost features from the stellar spectrum at higher resolving powers (as the shift of individual stellar spectral lines are resolved).

For PICASO, we modify the Pollack et al. (1986) approximation to include the Raman cross sections of \( \text{H}_2 \) computed by Oklopčić et al. (2016). We also retain the original Pollack et al. (1986) methodology as an option for low-resolution, low \( S/N \) observations.

Following Pollack et al. (1986), we introduce the effect of Raman scattering by adding a correction term to the Rayleigh opacity, \( \tau_{\text{Ray}} \),

\[
f_{\text{Ram}} = \frac{\sigma_{\text{Ray}} + \sigma_{\text{Ram}}(f_{\text{Ram}}/f_{\text{Ray}})}{\sigma_{\text{Ray}} + \sigma_{\text{Ram}}},
\]

where \( \sigma_{\text{Ram},\text{Ray}} \) are the cross sections of both Raman and Rayleigh scattering, respectively, and \( f_{\text{Ram}} \) and \( \lambda \) are the solar spectra at unshifted and shifted wavelengths. Each excitation corresponds to a specific wavelength shift of \( \lambda^{-1} = \lambda^{-1} + \Delta \lambda^{-1} \), where \( \Delta \lambda \) is the wavelength shift. For reference, the strongest transition of \( \text{H}_2 \) (the vibrational fundamental) is \( \Delta \nu = 4161 \text{ cm}^{-1} \).

Unlike Pollack et al. (1986), we use stellar spectral models from Castelli & Kurucz (2003) for this analysis. PICASO uses PySynphot (STScI Development Team 2013) so that users can draw from different stellar databases.

Additionally, we include initial rotational levels ranging from \( J = 0 \) to \( J = 9 \) for \( \text{H}_2 \) only because these were the transitions provided by Oklopčić et al. (2016). The cross sections for any given transition from an initial quantum state of \( \nu = 0, J_n \) to final quantum state of \( \nu_f, J_f \) is given by (see Equation (A4); Oklopčić et al. 2016)

\[
\sigma_{\text{Ram}}(0, J_n, f_j, J_f, \lambda) = \frac{C}{\lambda^4 \nu_j^2}\text{[cm}^2\text{]}.\n\]

The constant \( C \) is given by the values in Table A1 of Oklopčić et al. (2016).

In a future update we will include Raman scattering by \( \text{N}_2 \) and \( \text{He} \), but for this work, \( \text{H}_2 \) is sufficient to study the approximate behavior or Raman scattering.

### 2.5. Computing Different Types of Albedos

PICASO computes three different kinds of albedos: spherical albedo (A\(_s\)), geometric albedo (A\(_g\)), and the Bond albedo (A\(_b\)). The spherical albedo, A\(_s\), is the fraction of incident light that is reflected by a sphere toward all angles, and it can be computed for a planet phase geometry, \( \alpha \), by

\[
A_s(\lambda) = 2 \int_0^{\pi} \frac{F_p(\alpha, \lambda)}{F_{0.0}(\lambda)} \sin \alpha d\alpha,
\]

where \( F_p \) is the emergent flux from the planet, and \( F_{0.0} \) is the flux from a perfect Lambert disk under the same incident flux, \( F_0 \). This spherical albedo, integrated over all angles, can be written in two parts: the geometric albedo, and the phase integral. The geometric albedo is just the ratio of the reflected planet flux at full phase to the incident flux from the perfect
Lambert disk,

\[ A_b(\lambda) = \frac{F_\lambda(\alpha = 0^\circ, \lambda)}{F_0(\alpha = 0^\circ, \lambda)}. \]  

(16)

Then, the phase integral, which is normalized to be 1.0 at full phase, can be written as

\[ q = 2 \int_0^\pi \frac{F_\lambda(\alpha, \lambda)}{F_0(\alpha = 0^\circ, \lambda)} \sin \alpha d\alpha. \]  

(17)

Although we do not show any Bond albedos here, PICASO does contain the functionality to compute them. The Bond albedo is a stellar flux-weighted reflectivity that is integrated by wavelength,

\[ A_b = \frac{\int_0^\infty A_b(\lambda) F_\lambda(\lambda)d\lambda}{\int_0^\infty F_\lambda(\lambda)d\lambda}. \]  

(18)

Therefore, Bond albedos will vary for two planets that have identical spherical/geometric albedos but orbit different stars (Marley et al. 1999). Because a directly imaged exoplanet will never be observed at full phase, the traditional geometric albedo (which arose from solar system heritage) can be somewhat cumbersome, given the ratio to the ideal Lambert disk. Nevertheless, for ease of comparison with the existing literature, we here report results primarily in this framework.

2.6. Planet Phase Geometry

In order to capture the phase dependence, we compute the emergent intensity from the disk at several plane-parallel facets, where each facet has its own incident and outgoing angles. Following Horak & Little (1965), we use a Chebyshev–Gauss integration method to integrate over all the emergent intensities (also used in Cahoy et al. 2010; Madhusudhan & Burrows 2012; Webber et al. 2015). By default, PICASO includes 10 Chebyshev and 10 Gauss angles, which strikes a balance between computational speed and physical accuracy. However, this can easily be modified in the code (see options in justdoit.phase_angle()). Of course, increasing the number of angles increases computing time. However, if the user is particularly interested in capturing scattering at high cosine angles (e.g., near the limbs), then it is necessary to increase the number of integration angles accordingly.

Chebyshev–Gauss angles easily translate into planetary latitude and longitude, making it possible to explore the effect of 3D general circulation models on albedo spectra, as in, e.g., Webber et al. (2015) and Lee et al. (2017). Although we do not currently include this in the set of PICASO tutorials, we will make this jupyter notebook available soon.

3. An Analysis of the Modeling Assumptions

Given our reflected-light model, we now aim to determine which modeling assumptions are most important across a large range in parameter space. In particular, we are interested in sampling a parameter space across approximate temperature, cloud properties, and stellar spectrum (to test effects of Raman scattering).

In order to do so, we require as input the temperature-pressure profiles, cloud structure, and atmospheric composition profile. Batalha et al. (2018b) created a large grid across this particular parameter space. For a planet with a gravity of 25 m s\(^{-2}\), it covered planets with semimajor axes ranging from 0.5 to 5 au around a Sun-like star, metallicities (M/H) of 1–100 times solar, and cloud profiles ranging from \( f_{\text{sed}} = 0.01–6 \).

We use these models published in Batalha et al. (2018b) as input. Briefly, the temperature profiles were computed using the radiative–convective model initially developed by McKay et al. (1989) and later updated by Marley & McKay (1999), Marley et al. (2002), and Fortney et al. (2005, 2008).

The cloud profiles were computed using a Mie-scattering treatment of particle sizes calculated from the model developed by Ackerman & Marley (2001). Each profile was computed using a specific value of \( f_{\text{sed}} \), which is used to tune the sedimentation efficiency of the atmosphere. High values of \( f_{\text{sed}} > 1 \) produce vertically thin clouds with large particles, low values of \( f_{\text{sed}} < 1 \) produce the opposite—vertically thick clouds with small particles. The model of Ackerman & Marley (2001) produces as output single-scattering albedo, cloud extinction, and asymmetry values as a function of atmospheric layer, and the wavelength.

The top panel of Figure 5 contains two plots that show the depth in the atmosphere at which the two-way optical depth encountered by a photon traversing the atmosphere at \( \mu = 0.5 \) is \( \tau = 1 \), which we henceforth denote as a photon-attenuation plot. These two models, chosen from our grid, are computed at solar metallicity with a semimajor axis of 5 au. The left panel contains a model of a cloud-free system, while the right panel contains a system with a cloud sedimentation efficiency of \( f_{\text{sed}} = 3 \). We show these two case studies throughout the analysis of the modeling assumptions.

The shaded regions of the photon-attenuation plot indicate the dominant source of opacity as a function of wavelength: molecular absorption (blue), cloud absorption and scattering (green), or Rayleigh scattering (pink). Over the full illuminated hemisphere of a planet, the angle of incidence of course varies from \( \mu = 0 \) to 1 and scattering within cloud decks can increase the effective path length of a photon through the absorbing gas. Thus no single plot can fully capture the complete complexity inherent in the problem, but we find plots such as these helpful for understanding how the shape of reflected-light spectra can be traced back to the dominant sources of reflectivity and absorption in the atmosphere. This is especially true for cases with an interplay between the muting of strong scattering features (e.g., Raman) from the presence of optical absorbers (e.g., Na and K).

Indeed, such plots are commonly used in solar system planetary science to help illuminate the relative importance of scattering and absorption at different wavelengths (e.g., Sromovsky et al. 2009).

3.1. Sensitivity to Single-scattering Phase Function

We first explore the sensitivity of PICASO to the choice in single-scattering phase function. Figure 6 shows the same planet case as in Figure 5. The same spectrum was run using each of the four ways of representing direct scattering in PICASO.

For a cloud-free case where there are no high-asymmetry scatterers, the two TTHG functions and the function used in Cahoy et al. (2010) are all identical (because \( g_{\text{cd}} = 0 \). However, the cloud-free case shows deviations from the code default (TTHG\_ray) on the order of 10%–30% when the OTHG phase function is used.
For the cloudy case, Cahoy et al.’s phase function closely matches TTHG-ray, except when the Rayleigh-scattering opacity is high toward the blue, where deviations of \( \lesssim 10\% \) are present. Note that all deviations are strongly sensitive to wavelength.

OTHG exhibits the greatest deviation from the other phase functions because it does not account for the back-scattering peak from Rayleigh scattering. Figure 2 shows the Rayleigh contribution as a small back-scattering contribution. It is important to note that the actual Rayleigh phase function is symmetric forward and back, but when it is combined with forward-scattering particles, the net scattering is more forward.

Even for cases that are apparently less asymmetric (semi-major axis, \( a_0 = 0.5 \) au, see Figure 1), the specification for the direct-scattering phase function can still produce spectra that have maximum differences on the order of 100\% for full phase observations and differences of 50\% for phase = 90\°.

Although we do not show the specific effect of changing \( f \), the fraction of forward to back scattering, it will also strongly impact the resultant spectrum. Smaller fractions will produce smaller back-scattering peaks and yield significantly dimmer spectra across wavelength, and vice versa.

Modeling recommendation:

1. Use default specification for direct scattering (TTHG_ray).
2. Fit for the functional form of the fraction \( f \) of forward to back scattering according to the problem being addressed.

### 3.2. Multiple-scattering Phase Function and \( \delta \)-Eddington

Next, we assess PICASO’s sensitivity to the multiple-scattering phase function and to the \( \delta \)-Eddington approximation. Figure 7 shows the modeling sensitivity to the user’s choice for the multiple-scattering phase function. For both cloud-free and cloudy cases, there are \( \lesssim 1\% \) differences when choosing between an \( N = 1 \) or \( N = 2 \) Legendre expansion. The \( N = 2 \) expansion is used to approximate the multiple scattering by Rayleigh scattering. Therefore, for the cases modeled here, the diffuse scattering by Rayleigh is a relatively small contribution to the total reflectivity. As observations increase in precision, we will have to revisit whether this holds true. Studies of the accuracy of Legendre polynomial expansions suggest that they may degrade in accuracy for asymmetric large particles (Zhang et al. 2017). Zhang et al. (2017) also suggested that Chebyshev polynomial expansions are a more accurate alternative. We will save this for a future update, when better data warrant higher accuracy phase functions.

In order to improve the parameterization of these expansions, PICASO leverages the \( \delta \)-Eddington method of scaling the single-scattering albedo, opacity, and asymmetry parameter. When \( g_{\text{dd}} \) is nonzero, the choice of the \( \delta \)-Eddington method impacts the spectra by up to 30\% in some cases. We set \( N = 2 \) \( \delta \)-Eddington as default because it can with relatively high accuracy reproduce observations of Earth (Feng et al. 2018) and Jupiter (Cahoy et al. 2010).

As more diverse populations of exoplanets are observed in reflected light with higher S/N, we will conduct a more thorough investigation of these approximations. Since the publication of the \( \delta \)-Eddington method (Joseph et al. 1976), several other techniques have also been developed to improve the phase function parameterization (e.g., Hu et al. 2000; Iwabuchi & Suzuki 2009; Sorensen et al. 2017). We will consider these in a future update.

Modeling recommendation: For planet cases with some degree of asymmetric cloud scatterers, always use \( N = 2 \) Legendre polynomial expansion with the \( \delta \)-Eddington correction.

### 3.3. Raman Scattering

Figure 8 shows the modeling sensitivity to PICASO’s two methodologies for computing Raman scattering. The Pollack et al. (1986) approximation captures the general behavior of the decline in reflectively toward the blue, but fails to produce any
spectral features. When the Pollack et al. approximation is modified to include cross sections computed from Oklopcic et al. (2016), ghost spectral features are introduced at the ~10% level. Spectral features begin to disappear at $R \sim 50$, but small 1% baseline differences still remain at $R \sim 10$.

For a $T_{\text{eff}} = 6000$ K star (the Pollack et al. default), 10% differences remain through 0.55 $\mu$m (after the Mg I feature at 5200 Å). Cooler stars (toward $T_{\text{eff}} = 2600$ K), with more crowded molecular features, create spectral differences past 0.65 $\mu$m. Cloudy spectra (e.g., Figure 4 right panel) are also sensitive to Raman-scattering features despite the prominent cloud opacity in the blue.

Differences between the calculations here and those shown in Oklopcic et al. (2017; see Figure 4) and Sromovsky (2005; see Figure 17) are attributed to the resolution of the stellar spectrum and the stellar databases that were chosen. A key input to modeling Raman scattering correctly is an accurate, high-resolution stellar spectrum. Oklopcic et al. (2017) used stellar spectra from the Valdes et al. (2004) database, which are computed with $\Delta \lambda = 1$ Å. Sromovsky (2005) used a solar spectrum from the Upper Atmospheric Research Satellite, which had a nominal resolution of 2 Å.

The Castelli & Kurucz (2003) grid used here is computed at a resolution of 10 Å. This lower resolution grid will result in an underestimation of the Raman effect. One additional minor difference can be seen in the $T_{\text{eff}} = 2600$ K spectrum. Around 0.35 $\mu$m, some spectral features appear to have flat tops. This is a result, originally pointed out in Courtin (1999), of instabilities in the solution of the radiative transfer equation that prevent us from allowing Equation (13) to be greater than 1. Despite these differences, our modified Pollack approximation is a much more accurate solution than the original Pollack methodology.

One last subtlety is that Figure 8 makes it seem as if Raman scattering has a dramatic effect on the total energy budget of the atmosphere. This is somewhat exaggerated by the way in which the albedo is defined, by ratioing to the stellar flux. It has to be accounted for in order for the ratio to be correct, but in actuality, is not a huge influence on the energy budget of the atmosphere. The effect of the small deposition of energy into the atmosphere by the small-wavelength shifts that occur for those photons that experience this form of scattering would have to be computed by a complete radiative–convective equilibrium code that carefully tracks the energy budget of the atmosphere. Because PICASO is focused on tracking the reflectivity of the atmosphere as a whole, it is not well suited to this particular task, and doing so is beyond the scope of this paper.

Modeling recommendation: Choose the Pollack et al. methodology with the Oklopcic et al. (2016) cross sections, and choose a stellar spectrum that matches the required level of resolution and accuracy.

4. Benchmark Analysis

In order to benchmark the accuracy of the code, we chose to compare it against the results of Dlugach & Yanovitskij (1974) and Madhusudhan & Burrows (2012). Dlugach & Yanovitskij (1974) computed the intensity of radiation that is diffusely reflected from a semi-infinite homogeneous atmosphere with arbitrary single-scattering phase function. Their analysis focused on the optical properties of Venus and the Jovian planets. Therefore, they carried out calculations for Rayleigh and the HG phase functions with asymmetry parameters ranging from 0 to 0.9, and single-scattering albedos ranging from 0.7 to 1. Madhusudhan & Burrows (2012) provided analytic phase expressions for geometric albedo as a function of single-scattering albedo for both Rayleigh scattering and isotropy in a semi-infinite atmosphere. We compare PICASO against two models (one originating from solar system science, the other originating from exoplanet science) across a wide range in phase function, and the single-scattering albedo is sufficient enough to prove the accuracy of the model.

Figure 9 shows the first comparison against Dlugach & Yanovitskij (1974). Dlugach et al. used a one-term HG phase function for all asymmetric calculations. Therefore, it is important to note that if comparisons are carried out using PICASO’s default (as opposed to using OTHG for the single-scattering phase function), the results will not agree well. This further motivates our choice for inheriting older methodologies of computing phase functions so that fruitful code comparisons are easily accessible.
Using an OTHG phase function, the models agree within 10% for all Rayleigh phase functions and for $g \leq 0.5$. The models start to exhibit 10% differences for $0.5 < g \leq 0.85$. Because the diversity of cases illustrated in Figure 1 falls in this range of asymmetry values, we feel that this can be considered in good agreement. There are a few cases with $g = 0.9$ that exhibit ~40% differences. However, it is not obvious to what these differences could be attributed. Dlugach & Yanovitskij (1974) computed higher geometric albedos when single scattering was lower than 0.98, and lower geometric albedos when single scattering was ~1. Given the complete independence of the two models, there are numerous factors that might contribute to this, including the diffuse-scattering calculation, geometric integration, and the radiative transfer solver. Because most cases fall within a 10% agreement, we consider these two models to be in good agreement.

Figure 10 shows the comparison between the calculations in Madhusudhan & Burrows (2012) and PICALO. Here, we only compare isotropic and Rayleigh cases across a range of single-scattering albedos. Our results are well within a 10% agreement. The largest deviation comes from the computation of very low geometric albedos (~0.01). Such very low single-scattering albedos are well outside the range that is expected for the types of clouds that are expected (Figure 1), although unusual composition particles (e.g., Gao et al. 2017) can be quite dark at some wavelengths. We consider PICALO and the analytic model of Madhusudhan & Burrows (2012) to be in good agreement.

5. IC Analysis of Reflected Light

Currently, an important driver for the creation of PICALO is to determine optimal observing strategies for future direct-imaging missions, such as WFIRST, ELTs, and potential large space-based observatories such as the Large UV/Optical/IR Surveyor (LUVOIR) or the Habitable Exoplanet Observatory (HabEx). For example, determining bandpass ranges, minimum S/N, and instrument resolving powers that maximize the total retrievable information from a planetary reflected-light spectrum will be a critical contribution to the design of future facilities. Throughout this analysis, we focus specifically on the approximate S/N, bandpass, and resolution of the WFIRST Coronagraph Instrument (CGI), which is a technology demonstrator for future concept missions such as LUVOIR or HabEx. Our methodology can be applied to any parameter space, however.

Lupu et al. (2016) and Nayak et al. (2017) began to explore optimal observing strategies by wrapping the original Fortran code that was outlined in Cahoy et al. (2010) and others in a sophisticated retrieval framework. Lupu et al. (2016) focused on our ability to ascertain the presence or absence of clouds and CH4, while Nayak et al. (2017) focused on our ability to constrain planet phase and radius. These studies offered valuable insights into our ability to constrain the atmospheres of exoplanets with reflected light. However, the computational limitations of Markov chain Monte Carlo (MCMC; or similar) methods hinders our ability to rapidly move through a large parameter space in atmospheric diversity, resolution, and S/N.

IC theory offers an alternative to full MCMC methods. IC has been commonly used in Earth and solar system science (e.g., Saioh et al. 2009; Kuai et al. 2010), as well as in exoplanet science (e.g., Line et al. 2012; Batalha & Line 2017; Howe et al. 2017; Batalha et al. 2018a). We use the IC model that was originally developed for transiting exoplanet science. A full description of the methodology can be found in Batalha & Line (2017).

IC theory relies heavily on computing the Jacobian of individual systems, which describes how sensitive the model is to slight perturbations of the state vector parameters at a given initial state. In this analysis we assume that the state vector consists of $\{T(P), \xi_i, g\}$, where $T(P)$ is the pressure-dependent temperature profile, $\xi_i$ is the mixing ratio of species $i$, and $g$ is the gravity. We compute the derivative of the Jacobian using a centered-finite difference scheme. Our $T(P)$ and mixing ratio profiles come from the calculations in Batalha et al. (2018b), so that perturbations shift the entire profile. $T(P)$ and $g$ are perturbed linearly, with 0.1% perturbations. $\xi_i$’s are perturbed in log space, also with 0.1% perturbations. These finite perturbations were chosen to reproduce the results of a full retrieval analysis.

Of course there are several other parameters that contribute to an atmospheric state. Choosing only $T(P)$, $\xi_i$, $g$ is almost certainly too simplistic. For example, as we have seen here, the
cloud asymmetry parameter and the single-scattering albedo will largely contribute to how well we can constrain the atmospheres of exoplanets. Additionally, Lupu et al. (2016) showed that the pressure of the cloud deck will also influence the shape of the spectral features. In order to capture this behavior, we compute the Jacobian across a diversity of initial states ($a_s = 0.5–5.0$ au, $f_{\text{sed}} = 0.1–6$) at a phase angle of 90°.

As shown in Figure 1, this covers a broad diversity of cases in single scattering and asymmetry to compensate for our simplistic state vector. Along with the Jacobian, $K$, we also need an approximation of the error covariance, $S_e$, and $S_o$, the a prior covariance matrix. We take the values of the error covariance matrix from the cases in Nayak et al. (2017) for $S/N = 5–25$. The prior, $S_o$, represents the information we start with for any given system. We assume broad uniform priors for every state vector parameter. In other words, we assume to have very little information about the system before conducting our observation: ±300 K for $T(P)$, ±6 dex for the mixing ratio, and ±100 m s$^{-2}$ for gravity. Given $K$, $S_e$, and $S_o$, we can compute the posterior covariance matrix, which gives the 1σ uncertainty on a state vector parameter after a measurement is made:

$$\hat{S} = (K^TS_e^{-1}K + S_o^{-1})^{-1}. \quad (19)$$

Because of the $S_e^{-1}$ dependence, using large priors guarantees that our estimates for the posterior covariance matrix are solely driven by the model sensitivity (via the Jacobian) and the expected data quality at each wavelength (e.g., $K^TS_e^{-1}K \gg S_o^{-1}$). Additionally, we spot-checked our analysis against the full retrievals done in Lupu et al. (2016) and Nayak et al. (2017), and found that they are in good agreement.

Figure 11 shows a summary of the results of the IC analysis for a subset of $a_s$ and $f_{\text{sed}}$. We focus on this subset because it is the sweet spot in parameter space for a WFIRST-CGI mission. However, we discuss the full parameter space in Sections 5.1 and 5.2. Additionally, Figure 11 shows constant-exposure time contours for both detector-noise and photon-limited observations. Because WFIRST instrumentation has not yet been finalized, we cannot add definitive exposure times on each of these curves. However, we include them all the same to give readers an understanding of the interplay between $S/N$ and resolving power in terms of total time (e.g., for detector-noise-limited observations, it takes equal time to achieve $S/N = 25$ at $R = 40$ as $S/N = 5$ at $R = 120$). As WFIRST instrumentation is solidified, we will perform more robust noise simulations with estimates for integration time.

Overall, our ability to constrain composition and gravity are more dependent on the $S/N$ than on instrument resolving power. Regardless of resolving power, $S/N = 10$ is not sufficient to constrain either composition or gravity (our definition of a constraint is discussed in the following Section 5.1). Generally, an $S/N \sim 20$ is needed to attain robust constraints on composition and gravity.

### 5.1. Sensitivity to Composition

Figure 11 only shows the ability to constrain the abundance of CH$_4$, the dominant absorber at these temperatures. From $a_s = 0.5–5$, there is a transition from alkali-dominated atmospheres (Na and K) toward 0.5 au to CH$_4$-dominated atmospheres toward 5 au. This transition, a result of chemical equilibrium, occurs at about 0.85 au (see Batalha et al. 2018b), where both alkali and CH$_4$ features are comparatively small. WFIRST-CGI, with a proposed wavelength coverage for spectroscopy of $\sim$0.6–0.76 μm, will be primarily focused on the detection of CH$_4$. Therefore, we only show figures for $a_s = 1$ and 3 au because most of the considered targets will fall in this range.

At $S/N = 5$, constraints on CH$_4$ approach the prior value, meaning that the observation does not contribute to the overall knowledge. A definition of a “good” constraint is relatively arbitrary, but we adopt the definition of Feng et al. (2018), which is effectively the ability to constrain the abundance within ±1.0 log units. Although this may seem too stringent a definition, IC analyses tends to be more optimistic than full retrieval analyses because IC cannot pick up on important factors such as degeneracies between state vector parameters.

The difference in being able to detect CH$_4$ at 1 au versus a detection at 3 au comes from the relative size of the molecular features and the cloud composition. At 1 au, even though Na and K are nearly gone, CH$_4$ is still not as pronounced as it is at 3 au because the volume mixing ratio is lower. Additionally, water clouds at higher altitude will weaken the feature.
The cloud parameter $f_{\text{sed}}$ appears to have a weaker effect than the semimajor axis because at moderately high values of $f_{\text{sed}}$, the cloud deck is at low enough pressures to not completely impede the detection of molecular features. For $f_{\text{sed}} \leq 1$, detection of CH$_4$ (or any other molecular feature) will be difficult or impossible because the path length for reflected light through the atmosphere is too short.

5.2. Sensitivity to Gravity

The effect of gravity on reflected light is summarized in Figures 3 and 4 in Lupu et al. (2016). Generally, increasing gravity increases the depth of spectral features and increases reflectively toward the blue. There is also a more subtle effect of gravity on the 0.8 $\mu$m H$_2$ continuum feature (at lower gravity, the feature is stronger). We are not able to leverage this effect because of the WFIRST-CGI spectroscopic wavelength coverage.

For S/N $\leq 10$, the constraint on gravity, approaches the prior, meaning that the observation does not yet contribute to the overall knowledge. It also appears that systems at 1 au are slightly more amenable to gravity characterization than systems at 3 au because at 3 au, when $f_{\text{sed}} \geq 1$, the water cloud reflectively dominates the opacity, while at 1 au Rayleigh scattering still contributes. When the water cloud opacity dominates the opacity, the spectrum is less sensitive to slight perturbations in gravity. This is also why the $f_{\text{sed}} = 3$ cases are better constrained than the $f_{\text{sed}} = 1$ cases.

6. Discussion and Conclusion

Here, we presented an initial release of a reflected-light code called PICASO. PICASO is versatile enough for calculations of reflected-light spectroscopy and for retrievals of directly imaged exoplanet atmospheres. It has been benchmarked against two independent codes from Dlugach & Yanovitskij (1974) and Madhusudhan & Burrows (2012). For isotropic and Rayleigh scattering, PICASO agrees with other codes to well within 10%. For asymmetric scattering, calculations are slightly more discrepant, but well within the bounds of observational precision ($\sim 10\%$ agreement).

PICASO contains several methodologies for computing calculations of reflected light. Specifically, we have focused on highlighting different methodologies for computing single scattering, multiple scattering, and Raman scattering. Within each section, we have provided recommendations for modeling exoplanets, which are also PICASO’s default run settings. A further explanation of this is available in our online radiative transfer tutorial (see footnote 6).

Our IC analysis demonstrates the approximate parameter space in cloud composition, resolving power, and S/N, where we can expect to obtain robust constraints on composition and gravity. We find that in general, we need an S/N $\sim 20$ to attain constraints on composition, where our definition for constraint is attaining a 1$\sigma$ confidence interval of $\pm 1$ log unit on the volume mixing ratio of the dominant absorber (following Feng et al. 2018).

Despite the versatility of the original release, there are still aspects that we are currently working on. Future releases of the code will contain the following:

1. Thermal emission.
2. $\delta$–M stream method (Wiscombe 1977).
3. Raman scattering by N$_2$ and He.
4. Chebyshev polynomial for multiple-scattering phase function.
5. Compatibility with nested sampling algorithm.

Additionally, a robust retrieval analysis will be needed to address degeneracies that cannot be captured in an information-content (IC) analysis. This includes developing methods to constrain radius, directly retrieve the optical properties of the clouds (i.e., the imaginary component of the refractive index), and discern the presence of photochemical hazes. This analysis will additionally be added as a future code release because PICASO contains the modularity and versatility to support it.

We thank Kerri Cahoy for helpful discussion and tracking down various versions of the original albedo code. Additionally, we thank Cornell undergraduate Mark Siebert and Caltech graduate student Danica Adams for being the first beta testers and for pointing out some bugs in the code and installation. N.E.B acknowledges support from the University of California Presidents Postdoctoral Fellowship Program. M.S.M. acknowledges support from GSFC Sellers Exoplanet Environments Collaboration (SEEC), with funding specifically by the NASA Astrophysics Divisions Internal Scientist Funding Model.

Software: numba (Lam et al. 2015), pandas (McKinney 2010), bokeh (Bokeh Development Team 2014), NumPy (van der Walt et al. 2011), IPython (Pérez & Granger 2007), Jupyter, (Kluyver et al. 2016), PySynphot (STScI Development Team 2013), sqlite3 (sqlite3 Development Team 2019), picaso(Batalha 2019).

Appendix A
List of All Modeling Recommendations

In Section 3 we explored PICASO’s sensitivities to single-scattering phase function, multiple-scattering phase function, and Raman-scattering methodology. Throughout the text, we outlined our modeling suggestions. Here, we aggregate these recommendations into a single table. In this version of PICASO, they represent the current radiative transfer defaults.

1. Single scattering:
   (a) Use default specification for direct scattering (TTHG_Ray).
   (b) Fit for the functional form of the fraction, $f$, of forward to back scattering according to the problem being addressed.
   (c) See Table 1 for a list of advantages or disadvantages.

2. Multiple scattering:
   (a) For planet cases with some degree of asymmetric cloud scatterers, always use $N = 2$ Legendre polynomial expansion with the $\delta$-Eddington correction.

3. Raman scattering:
   (a) Choose the Pollack et al. methodology with the Oklopčić et al. (2016) cross sections.
(b) Choose a stellar spectrum that matches the required level of resolution.
(c) The user will experience slight computing-speed losses for a single run depending largely on the stellar/planet resolution that is chosen. However, because these shifts only need to be computed once, adding this Raman-scattering methodology is not a computational burden.

4. Phase geometry:
(a) The default number of integration angles is 10 Gauss and 10 Chebyshev angles. If the user is particularly interested in exploring scattering effects at high cosine angle (e.g., near the planet limb), it would be beneficial to increase the number of planet facets despite the decrease in computation speed.

Appendix B
Opacity Database

For this version of the code, PICASO contains a database of opacities that are hosted on Github. As summarized in Freedman et al. (2008), our molecular opacities are computed on a 1060 point pressure-temperature grid from 0.3 to 1 μm. This database currently contains the molecular opacity from CH₄, CO₂, CrH, FeH, H₂O, H₂S, K, Li, NH₃, Na, Hb, TiO, and VO. Notably, for CH₄ we include the visible methane following Karkoschka (1994). For continuum absorption, we include H bound-free, H free–free, H₂ + CH₄, H₂ + H₂ + H, H₂ + H₂ + He, and H₂ + Ne. We also include Rayleigh scattering from H₂, H, and CH₄, and Raman scattering from H₂ (Kolopčić et al. 2016).

Our opacity database is constructed in SQLite format. SQLite is a user-friendly python-based module for the databases. We provide a full tutorial on how to query and database to another format that can handle much larger data.

Freedman et al.2016, our molecular opacities are computed on a 1060 point pressure–temperature grid from 0.3 to 1 μm. This database currently contains the molecular opacity from CH₄, CO₂, CrH, FeH, H₂O, H₂S, K, Li, NH₃, Na, Hb, TiO, and VO. Notably, for CH₄ we include the visible methane following Karkoschka (1994). For continuum absorption, we include H bound-free, H free–free, H₂ + CH₄, H₂ + H₂ + H, H₂ + H₂ + He, and H₂ + Ne. We also include Rayleigh scattering from H₂, H, and CH₄, and Raman scattering from H₂ (Kolopčić et al. 2016).

Our opacity database is constructed in SQLite format. SQLite is a user-friendly python-based module for the C-library, SQLite. After testing several database formats (json, hdf5, ascii, sqlalchemy), SQLite was chosen because it is a lightweight disk-based database that does not require a separate server process. Additionally, as we expand our opacity database, it will be trivial to port over this smaller SQLite database to another format that can handle much larger data structures. We provide a full tutorial on how to query and construct sqlite3 databases. If users follow our recipe, they can swap in any molecular opacities without needing any code modifications.

ORCID iDs
Natasha E. Batalha https://orcid.org/0000-0003-1240-6844
Mark S. Marley https://orcid.org/0000-0002-5251-2943
Nikole K. Lewis https://orcid.org/0000-0002-8507-1304
Jonathan J. Fortney https://orcid.org/0000-0002-9843-4354

References
Ackerman, A. S., & Marley, M. S. 2001, ApJ, 556, 872
Barstow, J. K., Aigrain, S., Irwin, P. G. J., et al. 2014, ApJ, 786, 154
Barstow, J. K., Aigrain, S., Irwin, P. G. J., & Sing, D. K. 2017, ApJ, 834, 50
Batalha, N. 2019, natashabatalha/picaso: Initial Publication Release, Zenodo, doi:10.5281/zenodo.2647593
Batalha, N. E., Lewis, N. K., Line, M. R., Valenti, J., & Stevenson, K. 2018a, ApJL, 856, L34
Batalha, N. E., & Line, M. R. 2017, AJ, 153, 151
Batalha, N. E., Smith, A. R. W., Lewis, N. K., et al. 2018b, AJ, 156, 158

9 Opacity Tutorial (https://natashabatalha.github.io/picaso/notesbooks/5_SwappingOpacities.html).
Spergel, D., Gehrels, N., Breckinridge, J., et al. 2013, arXiv:1305.5422
sqlite3 Development Team 2019, DB-API 2.0 interface for SQLite databases
Sromovsky, L. A. 2005, Icar, 173, 254
Sromovsky, L. A., Fry, P. M., Hammel, H. B., et al. 2009, Icar, 203, 265
Stamnes, K., Tsay, S.-C., Jayaweera, K., & Wiscombe, W. 1988, ApOpt, 27, 2502
STScI Development Team 2013, pysynphot: Synthetic photometry software package, Astrophysics Source Code Library, ascl:1303.023
Sudarsky, D., Burrows, A., Hubeny, I., & Li, A. 2005, ApJ, 627, 520
Thelen, A. E., Nixon, C. A., Chanover, N. J., et al. 2019, Icar, 319, 417
Thomas, G. E., & Stamnes, K. 2002, Radiative Transfer in the Atmosphere and Ocean (Cambridge: Cambridge Univ. Press)
Toon, O. B., McKay, C. P., Ackerman, T. P., & Santhanam, K. 1989, JGR, 94, 16287
Toon, O. B., Pollack, J. B., & Sagan, C. 1977, Icar, 30, 663
Valdes, F., Gupta, R., Rose, J. A., Singh, H. P., & Bell, D. J. 2004, ApJS, 152, 251
van der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, CSE, 13, 22
Villanueva, G. L., Smith, M. D., Protopapa, S., Faggi, S., & Mandell, A. M. 2018, JQSRT, 217, 86
Waldmann, I. P., Tinetti, G., Rocchetto, M., et al. 2015, ApJ, 802, 107
Webber, M. W., Lewis, N. K., Marley, M., et al. 2015, ApJ, 804, 94
Wiscombe, W. J. 1977, JAtS, 34, 1408
Yelle, R. V., Doose, L. R., Tomasko, M. G., & Strobel, D. F. 1987, GeoRL, 14, 483
Zhang, F., Liu, K., Yang, Q., Wu, K., & Zhao, J.-Q. 2017, AdMet, 2017, 1835169
Zhang, M., Chachan, Y., Kempton, E. M.-R., & Knutson, H. A. 2019, PASP, 131, 034501