



Unifying radiative transfer models in computer graphics and remote sensing, Part II: A differentiable, polarimetric forward model and validation

Katherine Salesin ^{*,a}, Kirk D. Knobelspiesse ^b, Jacek Chowdhary ^c, Peng-Wang Zhai ^d, Wojciech Jarosz ^a

^a Department of Computer Science, Dartmouth College, 15 Thayer Dr, Hanover, NH, 03755, USA

^b Ocean Ecology Laboratory, GSFC/NASA, Code 616, Greenbelt, MD, 20771, USA

^c Department of Applied Physics and Applied Mathematics, Columbia University, 2880 Broadway, New York, NY, 10025, USA

^d Department of Physics, University of Maryland Baltimore County, 1000 Hilltop Circle, Baltimore, MD, 21250, USA



ARTICLE INFO

Keywords:

Radiative transfer
Computer graphics
Remote sensing
Atmospheric and ocean optics
Polarization
Monte Carlo

ABSTRACT

The constellation of Earth-observing satellites continuously collects measurements of scattered radiance, which must be transformed into geophysical parameters in order to answer fundamental scientific questions about the Earth. Retrieval of these parameters requires highly flexible, accurate, and fast forward and inverse radiative transfer models. Existing forward models used by the remote sensing community are typically accurate and fast, but sacrifice flexibility by assuming the atmosphere or ocean is composed of plane-parallel layers. Monte Carlo forward models can handle more complex scenarios such as 3D spatial heterogeneity, but are relatively slower. We propose looking to the computer graphics community for inspiration to improve the statistical efficiency of Monte Carlo forward models and explore new approaches to inverse models for remote sensing. In Part 2 of this work, we demonstrate that Monte Carlo forward models in computer graphics are capable of sufficient accuracy for remote sensing by extending Mitsuba 3, a forward and inverse modeling framework recently developed in the computer graphics community, to simulate simple atmosphere-ocean systems and show that our framework is capable of achieving error on par with codes currently used by the remote sensing community on benchmark results.

1. Introduction

In Part 1 of this work, we examined the history and current state of the art in radiative transfer models in computer graphics and remote sensing and established our motivation for importing knowledge from computer graphics. In this paper, we take first steps toward putting these concepts into action. Our goal in this work is to develop a framework that can match the accuracy and capabilities of multi-angle, spectral, polarimetric instruments on future satellite missions and can leverage the inherent advantages of Monte-Carlo-based solvers (e.g. three-dimensional geometry and spatial heterogeneity) as well as other special techniques imported from computer graphics (see Part 1). Although in this work we primarily focus on forward modeling, a medium-term goal of our research is to explore differentiable rendering as an engine for inverse modeling. Lastly, a long-term goal of this project is to release this software for use by the remote sensing community, so flexibility and ease of use are also important considerations.

To this end, we extend a forward and inverse modeling framework recently developed in the computer graphics community, Mitsuba 3 [1], to perform simulations of interest to remote sensing. We describe the pre-existing capabilities of Mitsuba 3 and its unique advantages in Section 2 and our extensions to it in Section 3. Lastly, we validate our framework on recent benchmark tests of simple atmosphere-ocean systems that conform to the plane-parallel layers assumption in Section 4 and show that our framework is capable of the accuracy expected for forward models in remote sensing. We plan to apply our framework to three-dimensional spatially varying media in future work.

2. Base capabilities of our framework

We chose to build our Monte Carlo radiative transfer framework on top of Mitsuba 3 [1] because it is uniquely suited for tackling this problem:

* Corresponding author.

E-mail address: katherine.a.salesin.gr@dartmouth.edu (K. Salesin).

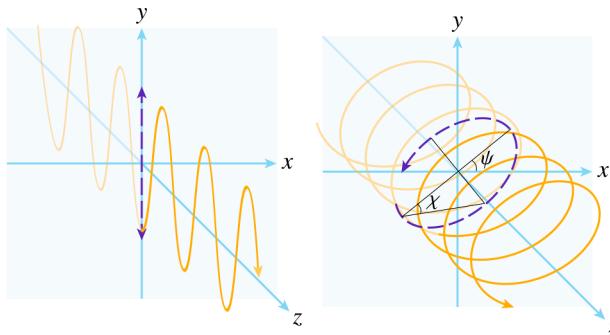


Fig. 1. Light travels along the z axis with linear polarization aligned with the y axis (left) and elliptical polarization with orientation angle ψ and ellipticity angle χ (right).

- **Physical accuracy.** It is a physically based renderer that adheres to radiative transfer principles and solves related problems via Monte Carlo integration. It can optionally simulate polarization via Mueller calculus (see Section 2.1).
- **Differentiability.** Derivatives of arbitrary radiative transfer simulations can be computed via differential radiative transfer (see Part 1, Sec. 4.6).
- **Modularity.** Concepts such as sensors, phase functions, even rendering algorithms themselves, are abstracted into base classes that can be extended to implement specific variants (e.g. a perspective camera or a Rayleigh phase function). These variants are loadable “plug-ins” which can easily be swapped for one another.
- **Speed.** The built-in Dr.JIT [2] just-in-time compiler generates platform-optimized megakernels for parallel execution on CPUs or GPUs via LLVM or CUDA/OptiX, respectively (it can also run on CPUs via traditional thread pools).
- **Accessibility.** It is open source, well maintained, and well documented.

Mitsuba 3 consists of a C++ and Python backend containing core classes, functions, and plug-ins and a Python frontend for calling them. Python bindings to the backend are automatically generated at compile time, allowing it to be loaded as a Python package and its classes and functions to be exposed to the user. The system is flexible: a user can either build higher-level functions in the backend and call them in the frontend, or build higher-level functions in the frontend from low-level backend components. For example, in the frontend, a user could load a specific plug-in (e.g. a Lorenz-Mie phase function) and run tests on that alone, load and render a complete scene (as shown in Section 4), or set up a gradient-descent loop for inverse modeling (which we will explore for the remote sensing context in future work). See the Mitsuba 3 tutorial suite for more examples [1].

The base system is an extensible framework with many built-in capabilities including a library of phase functions, bidirectional scattering distribution functions (BSDFs), emitters, sensors, and rendering algorithms that can simulate elastic scattering by surfaces and scattering media. The BSDF library includes a rough dielectric plug-in that can be used to simulate a rough ocean surface. While the base system supports scattering media that varies three-dimensionally in albedo and density (but not phase function), we leave exploration of spatially varying media for future work.

2.1. Representing polarization

Mitsuba 3 uses Mueller calculus to represent polarization states, which is also common in remote sensing. Mueller calculus uses a Stokes vector $s \in \mathbb{R}^4$ to describe polarization:

$$s = \frac{1}{2} \sqrt{\frac{\epsilon}{\mu}} \begin{bmatrix} I \\ Q \\ U \\ V \end{bmatrix} = \frac{I}{2} \sqrt{\frac{\epsilon}{\mu}} \begin{bmatrix} 1 \\ DP \cos(2\psi) \cos(2\chi) \\ DP \sin(2\psi) \cos(2\chi) \\ DP \sin(2\chi) \end{bmatrix}, \quad (1)$$

where I is the total radiance, Q and U describe the linearly polarized radiance and its orientation relative to a reference frame, and V describes circularly polarized radiance; ϵ and μ are the electric permittivity and magnetic permeability of the medium, respectively.¹ Graphically, one can think of the Stokes vector as representing the average polarization ellipse, with orientation angle ψ and ellipticity angle χ (see Fig. 1).

For the reference frame of Q and U , Mitsuba 3 uses a right-handed coordinate system with the z -axis pointing along the direction of light propagation (as illustrated in Fig. 1). To be consistent with the results presented in Chowdhary et al. [5], for radiance arriving at the sensor we align the x -axis of the reference plane with the plane spanned by the sensor direction and the global z -axis of the scene (see Fig. 5, this plane is shaded in blue beneath the blue sensor direction arrow), and a positive sign of Q indicates parallel to this plane. See the Mitsuba 3 documentation for more details on its polarization conventions [1].

The degree of polarization (DP) and degree of linear polarization (DLP) of the light are defined as:

$$DP = \frac{\sqrt{Q^2 + U^2 + V^2}}{I}, \quad DLP = \frac{\sqrt{Q^2 + U^2}}{I}, \quad (2)$$

where $DP = 0$ is completely unpolarized, $0 < DP < 1$ is partially polarized, and $DP = 1$ is fully polarized light. Any 4-vector that produces $DP < 0$ or $DP > 1$ is not a physically valid Stokes vector. In this work, we focus solely on linear polarization since circular/elliptical polarization is generally ignored in the remote sensing context (see Gassó and Knobelspiesse [6] for a recent counterargument) and modern-day polarimeters typically only collect linear polarization data.

A Stokes vector can be transformed into another by multiplication with a Mueller matrix $M \in \mathbb{R}^{4 \times 4}$ such that $s' = Ms$, which typically represents an interaction of light with a material, such as an optical element (e.g. polarizer), surface, or particle. A Mueller matrix can also rotate the reference frame of a Stokes vector.

Stokes vectors and Mueller matrices are convenient to use in a rendering context because they are natural and physically valid extensions of scalar, unpolarized radiance – in a path or light tracer, any scalar radiance quantity can be substituted for a Stokes vector and any scatter event can be substituted for a Mueller matrix. Stokes vectors may be added to represent a superposition of rays of light, and their addition correctly accounts for their cumulative polarization states. However, there is the extra complication of tracking the reference frames of each Stokes vector and rotating them appropriately before addition or multiplication.

3. Our extensions to the framework

We implemented several extensions to Mitsuba 3 in order to make it useful for a remote sensing context that include, but are not limited to: support for polarized phase functions, Rayleigh phase functions, Lorenz-Mie phase functions with optional distributions of particle size (see

¹ Stokes vectors can have different physical interpretations; we include the factor $1/2\sqrt{\epsilon/\mu}$ to clarify that our Stokes vectors carry radiance, which should be proportional to the flow of electromagnetic energy, which should be proportional to the medium properties ϵ and μ . In practice, this detail becomes relevant upon refraction at a dielectric surface, which changes the solid angle of the diverted beam and its radiance by a factor of the squared relative index of refraction (sometimes called the n^2 law) [3,4].

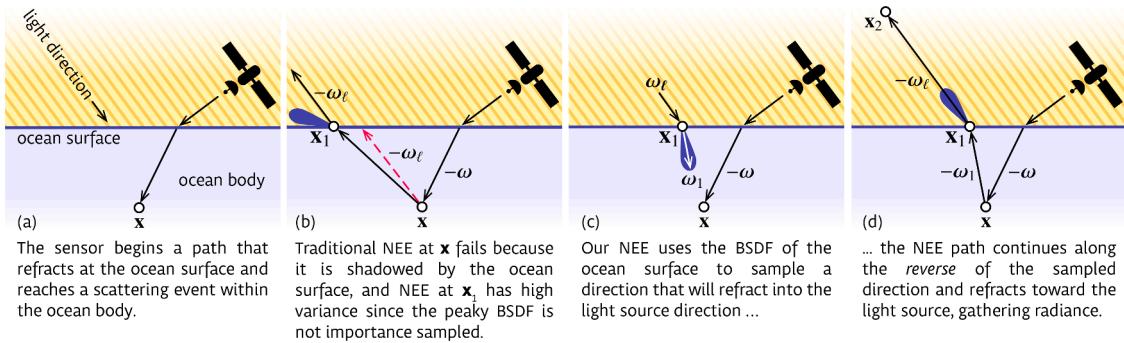


Fig. 2. Traditional next-event estimation (NEE) vs. our modified version to gather radiance within the scattering ocean body in a simple atmosphere-ocean system. Positions and directions match those defined in Eq. (8), and the BSDF lobe at \mathbf{x}_1 for each incoming direction is shown in purple.

Section 3.2), and a custom path tracer tailored for special handling of atmosphere-ocean systems (see Section 3.1).

3.1. Special handling of next-event estimation beneath an ocean surface

We use path tracing to estimate the radiance arriving at the sensor. A path is a set of n positions (“path vertices”) throughout a scene where valid scatter events may occur, i.e. on surfaces and within scattering media, beginning at the sensor and ending at a light source. Path tracing estimates the radiance arriving at the sensor by constructing light-carrying paths vertex by vertex – the next vertex is chosen by sampling a new direction for the path proportional to the current vertex’s out-scattering radiance, preferably.

For example, the outgoing radiance L_o leaving a surface at \mathbf{x} in direction ω^2 is given by:

$$L_o(\mathbf{x}, \omega) = L_e(\mathbf{x}, \omega) + \underbrace{\int_{\Omega} f_s(\mathbf{x}, \omega, \omega') L_i(\mathbf{x}, \omega') d\omega'}_{L_r(\mathbf{x}, \omega) : \text{reflected radiance}}, \quad (3)$$

which is the sum of emitted radiance L_e and reflected radiance L_r . The reflected radiance is an integral of incoming radiance $L_i(\mathbf{x}, \omega_i)$ from all directions ω_i over the sphere of directions Ω . This light is weighted by the surface’s cosine-weighted BSDF f_s . In a vacuum, the incoming and outgoing radiances are related as

$$L_i(\mathbf{x}, \omega) = L_o(r(\mathbf{x}, \omega), \omega) \quad (4)$$

where the ray tracing operator $r(\mathbf{x}, \omega) = \mathbf{x}_z := \mathbf{x} - z\omega$ returns the closest surface point along the ray from \mathbf{x} in direction $-\omega$. Together, Eqs. 3 and 4 form a recursive integral equation, and, as detailed in Part 1, a simple path tracing algorithm would approximate L_r using a 1-sample Monte Carlo estimator by sampling a random direction ω_i at each level of recursion. A straightforward choice is to sample ω_i proportional to the BSDF f_s .

To account for scattering media, Eq. (4) becomes:

$$L_i(\mathbf{x}, \omega) = e^{-\tau(\mathbf{x}, \mathbf{x}_z)} L_o(r(\mathbf{x}, \omega), \omega) + \underbrace{\int_0^z \sigma(\mathbf{x}_t) e^{-\tau(\mathbf{x}, \mathbf{x}_t)} \int_{\Omega} f_p(\mathbf{x}_t, \omega, \omega') L_i(\mathbf{x}_t, \omega') d\omega' dt}_{L_s(\mathbf{x}, \omega) : \text{in-scattered radiance}}. \quad (5)$$

² Recall that our unit vectors always point along the flow of light, so into a scattering event \mathbf{x} when measuring incident radiance L_i , and out when measuring outgoing radiance L_o or L_e .

The outgoing radiance at the end of the ray $L_o(r(\mathbf{x}, \omega), \omega)$ is attenuated by the transmittance, and we must additionally integrate the in-scattered radiance at each point $\mathbf{x}_t := \mathbf{x} - t\omega$ through the medium up to the end of the ray. The medium’s extinction coefficient is σ , and $\tau = \int_0^d \sigma(\mathbf{x}_a + t\omega) dt$ is the medium’s optical depth along the line segment from \mathbf{x}_a to \mathbf{x}_b of length d (we ignore emissive scattering media for brevity). We assume the phase function f_p integrates to the single-scattering albedo over the sphere of directions. A simple path tracing algorithm would first sample a random free-flight distance t , and depending on whether this stays within the medium or hits a surface, would then sample a new direction ω_i proportional to either the phase function f_p or the BSDF f_s .

Radiance from sources other than emissive surfaces, e.g. a directional light over some solid angle or an environment map, can be gathered when a path escapes the scene. The efficiency of this simple routine depends on the roughness of materials within the scene and the relative size and placement of light sources – scenes with a mixture of concentrated (small in size or solid angle) light sources and low-roughness materials are particularly challenging since the integrand becomes more peaky. When this happens, the probability that a path randomly sampled from the BSDF will hit a small light source is low, but when it does, the contribution to the Monte Carlo estimator will be huge, which results in high variance (see Part 1, Fig. 2a). An improvement commonly used in graphics is to include next-event estimation (NEE; or local/directional estimate), and to combine sampling strategies using multiple importance sampling (MIS), as we discussed in Part 1.

Unfortunately, a straightforward application of NEE and MIS provides little benefit in typical atmosphere-ocean systems since they contain particularly tricky light paths that require extra thought: because the ocean surface is refractive, scattering events that occur beneath the ocean surface cannot leverage traditional NEE. To see why, consider a path that has traveled beneath the ocean surface and is situated somewhere within the scattering ocean body: during NEE, a new ray is generated in the direction of the emitter; since this ray does not directly “see” the emitter – the ocean surface is in the way – it returns zero radiance. Simply ignoring this occlusion is unfortunately also not an option since the ray would be refracted at the ocean-atmosphere interface away from the emission source and hence gather no radiance (Fig. 2b). NEE does not help here because light from above the ocean surface cannot reach a scattering event within the ocean body along a straight line. This type of light path is in fact a caustic path. In graphics there are known general solutions to this problem such as bidirectional path tracing [7], Metropolis light transport [8,9], and photon mapping techniques [10–13], as well as specialized NEE techniques that account for one intervening reflection or refraction event [14–16]. While general, these approaches tend to be too heavy-weight for our more simplified setting.

We instead developed an importance sampling scheme to make such

refractive connections which remains light-weight and efficient by taking advantage of the plane-parallel assumptions in the atmosphere-ocean system (Fig. 2(c,d)). Instead of splitting incident radiance into direct and indirect parts, as is done in traditional NEE, we split incident radiance into caustic subpaths that connect a scattering vertex within the ocean to a light source via two straight path segments, and all other subpaths:

$$L_s(\mathbf{x}, \boldsymbol{\omega}) = \underbrace{\int_{\Omega} f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}_i) L_2(\mathbf{x}, \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i}_{L_{\text{caustic}}(\mathbf{x}, \boldsymbol{\omega}) : \text{length-2 caustic radiance}} + \int_{\Omega} f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}_i) L_{3+}(\mathbf{x}, \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i. \quad (6)$$

We approximate the incident radiance along the “other” subpaths using a standard 1-sample Monte Carlo estimator that samples a new direction $\boldsymbol{\omega}_i$ proportional to the phase function f_p . The subset of length-2 caustic paths expands to:

$$L_{\text{caustic}}(\mathbf{x}, \boldsymbol{\omega}) = \int_{\Omega} f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}_1) e^{-\tau(\mathbf{x}, \mathbf{x}_1)} \int_{\Omega} [f_s(\mathbf{x}_1, \boldsymbol{\omega}_1, \boldsymbol{\omega}_2) \times e^{-\tau(\mathbf{x}_1, \mathbf{x}_2)} L_e(\mathbf{x}_2, \boldsymbol{\omega}_2)] d\boldsymbol{\omega}_2 d\boldsymbol{\omega}_1, \quad (7)$$

where $\boldsymbol{\omega}_1$ is the direction entering the scattering event at \mathbf{x} (from the ocean surface) and $\boldsymbol{\omega}_2$ is the emission direction of the light source; then $\mathbf{x}_1 = \mathbf{r}(\mathbf{x}, \boldsymbol{\omega}_1)$ is a point on the ocean surface and $\mathbf{x}_2 = \mathbf{r}(\mathbf{x}_1, \boldsymbol{\omega}_2)$ is a point on the light source.

In the atmosphere-ocean setting, it is common to model incoming solar radiance using a light source that emits constant radiance L_e along a single direction $\boldsymbol{\omega}_e$ from an infinite distance. Using a Dirac delta distribution, defined as $\int \delta(x) dx = 1$ and $\delta(x) = 0$ for all $x \neq 0$, its emitted radiance can be expressed as $L_e \delta(\boldsymbol{\omega} - \boldsymbol{\omega}_e)$. Substituting this for $L_e(\mathbf{x}_2, \boldsymbol{\omega}_2)$ in Eq. 6 collapses the inner integral, yielding:

$$L_{\text{caustic}}(\mathbf{x}, \boldsymbol{\omega}) = \int_{\Omega} f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}_1) e^{-\tau(\mathbf{x}, \mathbf{x}_1)} f_s(\mathbf{x}_1, \boldsymbol{\omega}_1, \boldsymbol{\omega}_e) \times e^{-\tau(\mathbf{x}_1, \mathbf{x}_2)} L_e d\boldsymbol{\omega}_1. \quad (8)$$

Here, \mathbf{x}_2 corresponds to the point where the ray from \mathbf{x}_1 traveling in direction $-\boldsymbol{\omega}_e$ exits the atmosphere. We can then define a simple 1-sample Monte Carlo estimator for Eq. (8) by sampling a direction $\boldsymbol{\omega}_1$ ($\langle L \rangle := \langle L_{\text{caustic}}(\mathbf{x}, \boldsymbol{\omega}) \rangle$ for brevity):

$$\langle L \rangle = \frac{\int_{\Omega} f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}_1) e^{-\tau(\mathbf{x}, \mathbf{x}_1)} f_s(\mathbf{x}_1, \boldsymbol{\omega}_1, \boldsymbol{\omega}_e) e^{-\tau(\mathbf{x}_1, \mathbf{x}_2)} L_e}{p(\boldsymbol{\omega}_1)}. \quad (9)$$

The last question is how to sample a direction $\boldsymbol{\omega}_1$. Standard NEE can only be successfully applied at the ocean surface point \mathbf{x}_1 (Fig. 2b). This would correspond to importance sampling Eq. (9) with a PDF proportional to the phase function f_p . We found this leads to significant variance due to variation in the BSDF term f_s , so we instead developed a strategy that importance samples such two-segment paths proportional to the BSDF: we use the light’s emission direction $\boldsymbol{\omega}_e$ and the BSDF’s existing sampling strategy to sample a new direction $\boldsymbol{\omega}_1$ that refracts *into* the ocean body (Fig. 2c). By the reciprocal nature of light transport, any path starting at \mathbf{x} and traveling along the *reverse* of this BSDF-generated ray will refract toward the emission direction and gather radiance successfully. This sampling step gives us an initial ray to start our subpath: its origin is \mathbf{x} and its direction $-\boldsymbol{\omega}_1$ is the reverse of the BSDF-generated ray. When the path reaches the ocean surface, we evaluate the surface BSDF in the emitter direction $\boldsymbol{\omega}_e$, and additionally evaluate the transmittance through the ocean and atmosphere (Fig. 2d). This method assumes that the surface is planar and spatially homogeneous, i.e. the location of entry/exit at the surface does not matter. It is also important not to double count the radiance along length-2 caustic subpaths: if a path generated by phase function sampling (to estimate the second part of Eq. 6) happens to refract *out* of the ocean at the surface, traditional

NEE should not be performed at that surface scattering event because its radiance is accounted for in Eqs. (7) and (8).

We must also account for this sampling step by dividing the gathered radiance by the sampling PDF. If f_s is a rough microfacet BSDF, a common choice for modeling the ocean surface, a sampling strategy with a PDF perfectly proportional to f_s is available [18,19], which causes $f_s(\boldsymbol{\omega}_1)$ and $p(\boldsymbol{\omega}_1)$ to cancel each other out. Any remaining variance in Eq. (9) is due to the phase function and transmittance evaluations.

We validate our strategy on a test case based on the AOS-II atmosphere-ocean model described in Section 4.1 in Fig. 3. The scene is composed of an ocean surface modeled by a microfacet BSDF and an ocean body beneath filled with a homogeneous scattering medium. We estimate the ocean-leaving radiance (ignoring surface reflection) since that is the only component handled by our custom NEE method, and also examine the effects of varying the roughness of the ocean surface. We compare to the standard NEE strategy that generates successful direct connections to the light only at the ocean surface. Our method is relatively invariant to roughness, whereas the basic NEE exhibits higher variance as the surface approaches specular. In every case, our method results in a variance reduction of three to four decimal orders of magnitude. It should be possible to further reduce the remaining variance by performing MIS between traditional NEE (importance sampling proportional to the phase function) and our NEE (importance sampling proportional to the BSDF), though we leave this for future work.

We believe our two-segment NEE strategy is just the starting point for further improved path construction in plane-parallel systems. Our current strategy can be seen as a (long photon volume \times point) estimator [20] specialized for a planar refractive surface. This suggests that additional variance reduction may be possible by developing e.g. (photon volume \times long beam) strategies, and more. We hope to explore this in the future.

3.2. Lorenz-Mie scattering

We follow the implementation of Mishchenko and Yang [17] and Mishchenko et al. [21] for Lorenz-Mie scattering: they describe in detail how to compute the phase function and scattering and extinction cross sections for a given parameter set by summing the necessary spherical functions using stable recurrence relations and integrating those

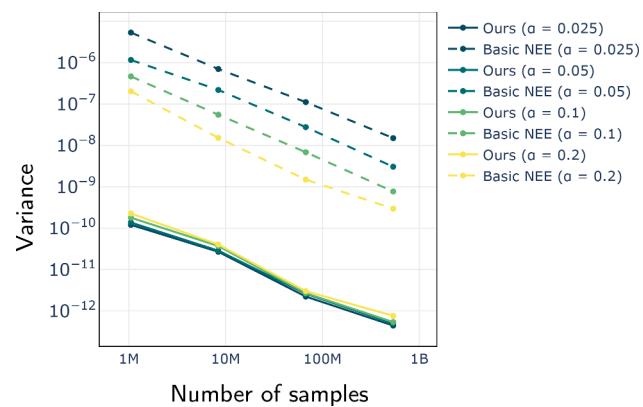


Fig. 3. We demonstrate the variance reduction of our NEE (solid) versus basic NEE (dashed) on the AOS-II atmosphere-ocean model described in Section 4.1 at $\theta_e = 60^\circ$, $\theta_v = 60^\circ$. The scene is composed of a rough ocean surface and a homogeneous ocean body. Here, we have isolated just the ocean-leaving radiance (and ignore surface reflection) to isolate the tricky scenario in question and highlight the difference between the methods. We show the effect of modifying the roughness (α), or mean square slope, of the ocean surface across a range of values: our method performs consistently across all values, while basic NEE becomes more and more noisy as the roughness decreases (the surface approaches specular). In all cases, our method consistently produces 1,000–10,000 \times lower variance than traditional NEE.

quantities over a range of particle sizes using Gaussian quadrature. We implemented size distributions as a base class so that arbitrary distributions can easily be added to our framework, such as gamma, lognormal, power-law, etc.

Because Lorenz-Mie scattering is prohibitively expensive to evaluate for every scatter event, we pre-tabulate Lorenz-Mie phase functions over a range of scattering angles to use at runtime. A Lorenz-Mie phase function typically exhibits strong forward scattering, and we would like to create a PDF for sampling directions from this distribution that matches its shape. To that end, we used an existing capability of Mitsuba 3 that computes a piecewise-linear PDF proportional to tabulated data (see e.g. Pharr et al. [22] for details on constructing, sampling, and evaluating such a PDF). We use the top-left Mueller matrix value to construct the tabulated PDF and draw directional samples from this distribution at runtime. We then evaluate the Mueller matrix elements by linearly interpolating each tabulated element in the sampled direction and weight the result by the sampling PDF. While the tabulation resolution can be chosen arbitrarily, for our results in Section 4 we use tabulated data in 0.1° increments for scattering angles from 0° to 10° and 1° increments from 10° to 180°.

4. Results

To validate the accuracy of our framework, we compare to all of the benchmark results presented in Chowdhary et al. [5]. These benchmark results are simulated top-of-atmosphere and just-above-surface polarimetric measurements for atmosphere-ocean systems over several wavelengths and a range of sun and viewing geometries. Because the majority of the radiative transfer codes evaluated in Chowdhary et al. [5] are deterministic and therefore operate under the plane-parallel layers assumption, we made our atmosphere-ocean systems also conform to that assumption for comparison purposes.

Table 1
Scene composition for atmosphere-ocean models.

	Atmosphere	Ocean Surface	Ocean Body
AOS-I	Yes	Yes	No
AOS-II	No	Yes	Yes
AOS-III	Yes	Yes	Yes
AOS-IV	Yes	Yes	Yes

Table 2
Scene parameters for atmosphere, ocean surface, and ocean body in all atmosphere-ocean models, where applicable, per wavelength (λ). Abbreviations: single-scattering albedo (ω), optical depth (τ), index of refraction (IOR), mean square slope (α^2).

λ	350 nm	450 nm	550 nm	650 nm
Atmosphere				
Phase function: Rayleigh (no depolarization)				
ω	1	1	1	1
τ	0.63031	0.22111	0.097069	0.049188
Ocean surface				
BDSF: isotropic Cox-Munk [26]				
IOR	1.34 – 0i	1.34 – 0i	1.34 – 0i	1.34 – 0i
α^2	0.03884	0.03884	0.03884	0.03884
Ocean body (AOS-I, AOS-II, AOS-III)				
Phase function: Rayleigh (no depolarization)				
ω	0.39644970	0.32846715	0.03253425	0.002932551
τ	3.38	1.37	5.84	34.1
Ocean body (AOS-IV)				
Phase function: detritus-plankton model [27] (see Table 3)				
ω	0.72114137	0.72519084	0.88340267	0.68034101
τ	7.71	5.24	91.34	118.47

Table 3

Parameters for AOS-IV model's bio-optical phase function. Both Lorenz-Mie phase functions are calculated at $\lambda = 550$ nm and are assumed to be spectrally invariant. The low chlorophyll mix ratio is used for $\lambda = 350$ nm and $\lambda = 450$ nm and the high chlorophyll mix ratio is used for $\lambda = 550$ nm and $\lambda = 650$ nm (see Chowdhary et al. [5] for discussion). Particle sizes are integrated over the domain $0.01 – 100\mu\text{m}$. Abbreviations: scattering coefficient (b), scattering cross section (σ), index of refraction (IOR), mix ratio of detritus & minerals to phytoplankton (a_{dm}).

λ	350 nm	450 nm	550 nm	650 nm
b_w	0.0134	0.0045	0.0019	0.0010
b_p	0.0422	0.0335	0.8050	0.8050
Lorenz-Mie scattering parameters				
				Detritus
				Phytoplankton
Size distribution exponent	4.4			3.7
IOR (particles)	1.15			1.04
IOR (medium)	1.0			1.0
σ	1.388×10^{-5}			8.874×10^{-5}
a_{dm} (low chlorophyll)	0.61			–
a_{dm} (high chlorophyll)	0.34			–

4.1. Scene descriptions and parameters

The scene composition of the four atmosphere-ocean systems is illustrated in Fig. 5. Each atmosphere-ocean system contains some combination of a homogeneous scattering atmosphere, a wind-ruffled ocean surface, and a homogeneous scattering and absorbing ocean body (see Table 1 for a summary). We list the material parameters in Tables 2 and 3, which match those used in Chowdhary et al. [5]: for the atmosphere we ignore the effects of molecular absorption by gases and borrow optical depths from literature [23]; for the ocean body we assume the ocean is composed of pure seawater and borrow its scattering and absorption properties from literature [24,25] for AOS-I, AOS-II, and AOS-III (the ocean body for AOS-IV is more complex and described in detail below); the ocean floor is modeled by a fully absorbing surface. The ocean surface is modeled by an isotropic Cox-Munk BSDF [26] with a roughness value corresponding to a wind speed of ~ 7 m/s at 12 m above the surface, which translates to a mean square slope $\alpha^2 = 0.03884$ (the default parameter in Mitsuba 3 is the root mean square slope, α). In order to match the results of Chowdhary et al. [5] as closely as possible, we omit the shadowing-masking term for microfacets and do not enforce conservation of energy.

The final AOS-IV model differs from the other models in that it uses a bio-optical phase function [27] for the ocean that is a weighted average of pure seawater, detritus and mineral particles, and phytoplankton particles. In the following, \mathbf{F} indicates a polarized phase function that expands into a Mueller matrix. This phase function is given by:

$$\mathbf{F}_{\text{ocean}}(\lambda, \theta) = \frac{b_w(\lambda) \mathbf{F}_w(\theta) + b_p(\lambda) \mathbf{F}_p(\theta)}{b_w(\lambda) + b_p(\lambda)}. \quad (10)$$

The pure seawater scattering matrix \mathbf{F}_w is modeled by Rayleigh scattering; b_w and b_p are the scattering coefficients of pure seawater and particles, respectively. The particle scattering matrix \mathbf{F}_p is further subdivided into detritus & mineral and phytoplankton contributions, given by:

$$\mathbf{F}_p(\theta) = \frac{(1 - a_{\text{dm}}) \sigma_{\text{ph}} \mathbf{F}_{\text{ph}}(\theta) + a_{\text{dm}} \sigma_{\text{dm}} \mathbf{F}_{\text{dm}}(\theta)}{(1 - a_{\text{dm}}) \sigma_{\text{ph}} + a_{\text{dm}} \sigma_{\text{dm}}}. \quad (11)$$

The scattering matrices \mathbf{F}_{ph} and \mathbf{F}_{dm} are modeled by Lorenz-Mie scattering with a power-law (or Junge) distribution of particle sizes; a_{dm} is

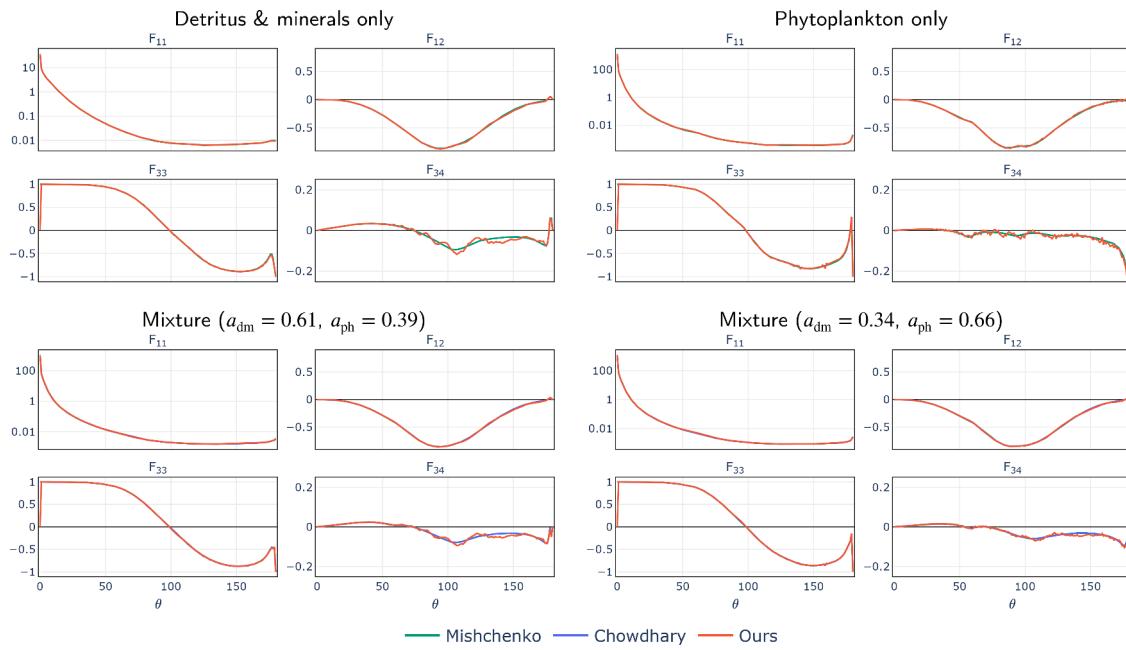


Fig. 4. We compare our Lorenz-Mie scattering computations to two references: results computed using Mishchenko and Yang [17]’s code (top row) and tabulated results provided in Chowdhary et al. [5] (bottom row). We use the Lorenz-Mie scattering parameters in Table 3 to compute the four unique Mueller matrix elements of a collection of spherical particles representing detritus & minerals in the ocean (top-left), phytoplankton in the ocean (top-right), and mixtures of these phase functions at two different ratios (bottom-left and bottom-right, a_{dm} is the fraction of detritus & mineral particles and $a_{ph} = 1 - a_{dm}$ is the fraction of phytoplankton particles). Our results match both references very closely, with some minor artifacts from Gaussian quadrature visible in F_{34} . We use 200 Gaussian quadrature points to integrate over the entire domain of particle sizes for our results (Mishchenko and Yang [17]’s code subdivides the integration domain into 100 subintervals and uses 100 Gaussian quadrature points to integrate over each subinterval; the number of quadrature points used in Chowdhary et al. [5] is unknown).

the fraction of particles represented by detritus & minerals³; σ_{ph} and σ_{dm} are the scattering cross sections of phytoplankton and detritus & mineral particles, respectively. We list the parameters for this phase function in AOS-IV, including those for Lorenz-Mie scattering, in Table 3.

Chowdhary et al. [5] helpfully provide their tabulation of this phase function for AOS-JV in their supplemental material; we show that we can also compute this phase function from the given parameters using our own Lorenz-Mie scattering implementation in Fig. 4. For the AOS-JV results presented in Fig. 20, 19, 13, 12, we use the tabulated data provided by Chowdhary et al. [5] to achieve the most faithful comparison – they provide data in 0.1° increments for scattering angles from 0° to 10° and in 1° increments from 10° to 180°.

4.2. Comparison to benchmark results

Chowdhary et al. [5] use eGAP [28,29] to compute their reference results, an adding-doubling radiative transfer code currently used at NASA/GISS, to an accuracy of 1×10^{-6} for models AOS-I, AOS-II, and AOS-III and 1×10^{-5} for model AOS-IV. They provide results in units of reflectance, defined as:

$$R_i = \frac{\pi s_i}{L_e \mu_\ell}, \quad (12)$$

where $s_i \in \{I, Q, U\}$ is the final Stokes vector component (i.e. radiance value) recorded by the sensor, L_e is the (unpolarized) incoming extra-terrestrial solar flux (set to π in all cases for simplicity), and $\mu_\ell = \cos\theta_\ell$ (see Fig. 5).

We compare reflectance values to the reference results in terms of

relative percent error, defined as:

$$\epsilon_R = \frac{R_i^{\text{ours}} - R_i^{\text{ref}}}{R_i^{\text{ref}}} \times 100 \quad (13)$$

and degree of linear polarization in terms of absolute error (since the quantity itself is a percentage):

$$\epsilon_{\text{DLP}} = \text{DLP}^{\text{ours}} - \text{DLP}^{\text{ref}}. \quad (14)$$

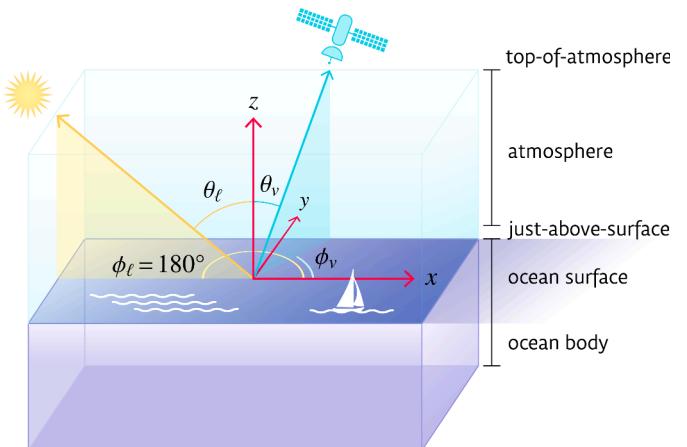


Fig. 5. Each atmosphere-ocean scene in Chowdhary et al. [5] is some combination of a homogeneous atmosphere, a rough ocean surface, and a homogeneous ocean body stacked in parallel layers (see Tables 1 to 3 for the composition and parameters of each). We illustrate the geometric setup of light and viewing angles.

³ We substitute the notation a_{dm} for f_{dm} used in Chowdhary et al. [5] to avoid confusion with our BSDFs (f_s) and phase functions (f_p).

Our stretch target was to achieve $|\varepsilon_R| < 0.01\%$ and $|\varepsilon_{DLP}| < 0.1\%$ for all models and parameter sets, in accordance with the typical error of the deterministic and Monte Carlo codes compared in Chowdhary et al. [5]; these thresholds are beneath the measurement uncertainties for modern-day polarimeters in the range $|\varepsilon_R| < 2\text{--}5\%$ and $|\varepsilon_{DLP}| < 0.2\text{--}0.5\%$ [30–32]. The codes compared in Chowdhary et al. [5] were able to achieve $|\varepsilon_R| < 0.05\%$ in most cases and $|\varepsilon_R| < 0.01\%$ often, with some outliers generally within $|\varepsilon_R| < 0.2\%$. All codes consistently achieved $|\varepsilon_{DLP}| < 0.1\%$, except for very few outliers within $|\varepsilon_{DLP}| < 0.2\%$.

Our results, shown in Figs. 7 to 20, were computed using our Mitsuba 3 framework with $\sim 5.37 \times 10^8$ samples for all parameter sets. Since our Monte Carlo code would naturally exhibit some random noise, we conducted 20 trials per parameter set using a different random seed for each – the mean of each set of trials is connected by dark purple line and a kernel density plot (“violin” plot) illustrating the spread of each set of trials is shown in light purple for each parameter set. The mean of all trials is also equivalent to a Monte Carlo estimate using $20 \times 5.37 \times 10^8$ total samples for that parameter set. Our target error regions of $|\varepsilon_R| < 0.01\%$ and $|\varepsilon_{DLP}| < 0.1\%$ are shown in light blue on each plot, which also gives a visual cue of instances where the y-axis has been scaled to encompass outliers.

For reflectance, the mean errors for *AOS-I* and *AOS-III* remain within our target error threshold of $|\varepsilon_R| < 0.01\%$ in most cases. Individual trials for *AOS-I* and *AOS-III* mostly remain within $|\varepsilon_R| < 0.1\%$ for top of atmosphere and $|\varepsilon_R| < 0.5\%$ for just above surface. *AOS-II* exhibits more consistent error, with error means typically in the range $|\varepsilon_R| < 0.1\%$, but sometimes straying into the $|\varepsilon_R| < 0.5\text{--}1\%$ range as the viewing angle θ_v increases. Error means for *AOS-IV* mostly remain within $|\varepsilon_R| < 0.5\%$ for top of atmosphere and $|\varepsilon_R| < 1\%$ for just above surface, with some outliers in $|\varepsilon_R| < 5\%$ for just above surface. This may be due to a difference in implementation in how the tabulated phase function is sampled and evaluated.

For degree of linear polarization, the majority of all trials remain comfortably within our target error threshold of $|\varepsilon_{DLP}| < 0.1\%$ for *AOS-I* and *AOS-III*, except for some outliers generally within $|\varepsilon_{DLP}| < 0.5\%$ for *AOS-I* and *AOS-III* just above surface. *AOS-II* trials meet our target threshold of $|\varepsilon_{DLP}| < 0.1\%$ for the $\phi_v = 0^\circ$ case; for other geometries the error is generally within $|\varepsilon_{DLP}| < 0.5\%$ and in a few cases within $|\varepsilon_{DLP}| < 1\%$ (at $\theta_\ell = 60^\circ$, $\phi_v = 60^\circ$ and $\theta_\ell = 60^\circ$, $\phi_v = 240^\circ$). *AOS-IV* consistently meets our target error threshold of $|\varepsilon_{DLP}| < 0.1\%$ for top of atmosphere except for some outliers within $|\varepsilon_{DLP}| < 0.5\%$; for just above surface, the error means remain consistently within $|\varepsilon_{DLP}| < 0.5\%$ and some outliers within $|\varepsilon_{DLP}| < 1\%$.

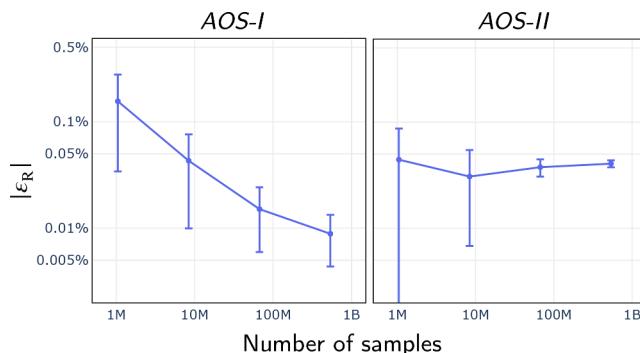


Fig. 6. (Absolute value of) Relative percent error in reflectance, $|\varepsilon_R|$, for *AOS-I* and *AOS-II* models at top of atmosphere, for $\theta_\ell = 30^\circ$, $\theta_v = 30^\circ$, $\phi_v = 60^\circ$. The mean of each set of 20 random trials is connected by dark purple line and the standard deviation of the trials is shown as error bars; both axes are a log scale. The *AOS-I* model shows a pattern of converging toward the reference result, while *AOS-II* shows a slight bias that approaches $|\varepsilon_R| \approx 0.04\%$.

We also ran convergence tests with increasing sample counts in order to assess the convergence behavior of our Monte Carlo estimates. This provides an indication of whether any remaining error at high sample counts is due to variance (which would vanish at even higher sample counts) or bias (a pernicious error that would not vanish at higher sample counts). Examples shown in Fig. 6 show the convergence of the reflectance estimate for one parameter set of the *AOS-I* and *AOS-II* models. The *AOS-I* estimate appears to be converging to the reference result, whereas the *AOS-II* model exhibits some slight bias, converging to $|\varepsilon_R| \sim 0.04\%$.

These validation tests demonstrate that our framework – while exhibiting a small amount of bias in some cases – generally reaches an error threshold on par with deterministic codes currently used by the remote sensing community. Therefore, it should be acceptable for use in remote sensing applications.

4.3. Technical details and performance

All results were generated on a cluster node with 2.90 GHz Intel Xeon Gold 6226R CPUs. Mitsuba 3 can be compiled to run either on CPU (via traditional thread pools or LLVM infrastructure) or GPU (via CUDA/OptiX) – we used the CPU variant with LLVM optimizations. This variant, as well the CUDA/OptiX-based GPU variant, rely on the integrated Dr.Jit [2] just-in-time compiler to generate platform-optimized megakernels for parallel execution. See Jakob et al. [2] for more technical details.

Due to the independence of traced paths, Monte Carlo codes such as ours are “embarrassingly parallel” – in theory, the minimum runtime would be the cost of tracing one path (plus setup of inputs and processing of outputs), if enough cores were available. Therefore, the “efficiency” of our code is primarily a product of 1) how many samples are taken, and 2) how many cores are available. Running one trial of our results in Section 4 (i.e. calculating one final radiance estimate corresponding to one parameter set) with $\sim 5.37 \times 10^8$ samples (i.e. each sample is a full path) and 64 cores generally takes 10 to 40 s total for *AOS I-III* and 1 to 2.5 min for *AOS-IV*, depending on the model and parameters.

If comparing performance in terms of wall clock time, most deterministic codes will have the advantage – however, our framework comes with other advantages, including the inherent advantages of Monte-Carlo-based models discussed in Part 1, as well as the framework-specific advantages discussed in Section 2.

5. Discussion, limitations, & future work

Our framework can theoretically handle most of the complex light-based phenomena listed in the first paragraph of Part 1, with the caveat that appropriate geometry and material properties would need to be provided. Some exceptions include inelastic scattering (e.g. fluorescence), as wavelengths of propagating light are assumed to be fixed, and spatially varying phase functions in scattering media, which would be necessary to simulate media with a spatially varying distribution of particle sizes. These features could be added in theory, but would require significant modifications to the base framework.

The extensive suite of validation tests in Section 4 establishes that our current framework is an acceptable forward model for simple atmosphere-ocean systems. Although these results conform to the assumption of plane-parallel layers for the sake of comparison, we hope to illustrate the relative advantages of our framework on more complicated cases with spatially varying scattering media and three-dimensional geometry in the future. We hope to incorporate several of the relevant threads of research discussed in Part 1, Sec. 4, such as multiple-scattering microfacet models and lower-variance estimators of spectrally and spatially varying scattering media, into our framework in order to push the accuracy and efficiency of our forward model beyond the current state of the art in remote sensing.

Concurrent work, a project called Eradiate [33], has also begun to develop a forward modeling framework for remote sensing that uses Mitsuba 3 as its underlying backbone; however, their project aims to build a user-friendly wrapper of remote-sensing-oriented scripts and tools that use Mitsuba 3 “under the hood.” Our primary focus is to improve the inverse modeling capabilities of Mitsuba 3 on polarimetric measurements and our forward model is a stepping stone from which to begin that exploration.

Differentiable rendering with Mitsuba 3 has already shown promising results in retrieving the roughness parameter of a microfacet model and spatially varying optical depth of a scattering medium in an unpolarized context [34]. Therefore, a natural place for us to begin is to test the retrieval of similar parameters from the simple atmosphere-ocean scenes used in Section 4 in a polarized context. We will also incorporate our Lorenz-Mie scattering model in order to retrieve polarization-sensitive parameters such as size and complex indices of

refraction of aerosols.

While differentiable rendering has many desirable qualities, it is not a panacea: there are several practical issues that will need to be examined. For instance, differentiable rendering inherits the drawbacks of whatever optimizer (e.g. stochastic gradient descent, ADAM [35]) it is used with, such as converging to a local optimum instead of a global one, choosing an appropriate learning rate, momentum, regularizer, etc. Since it is a Monte Carlo approach, determining an appropriate sample count for gradient estimates introduces yet another hyperparameter into the optimization. There is also the question of computational “waste”: a naive optimization would forget intermediate results computed in prior steps and perform a fresh, costly Monte Carlo simulation at every iteration, even though the parameter changes may be very small. To mitigate this, some works have begun to design optimization methods for differentiable rendering that reuse information from prior steps [36–38].

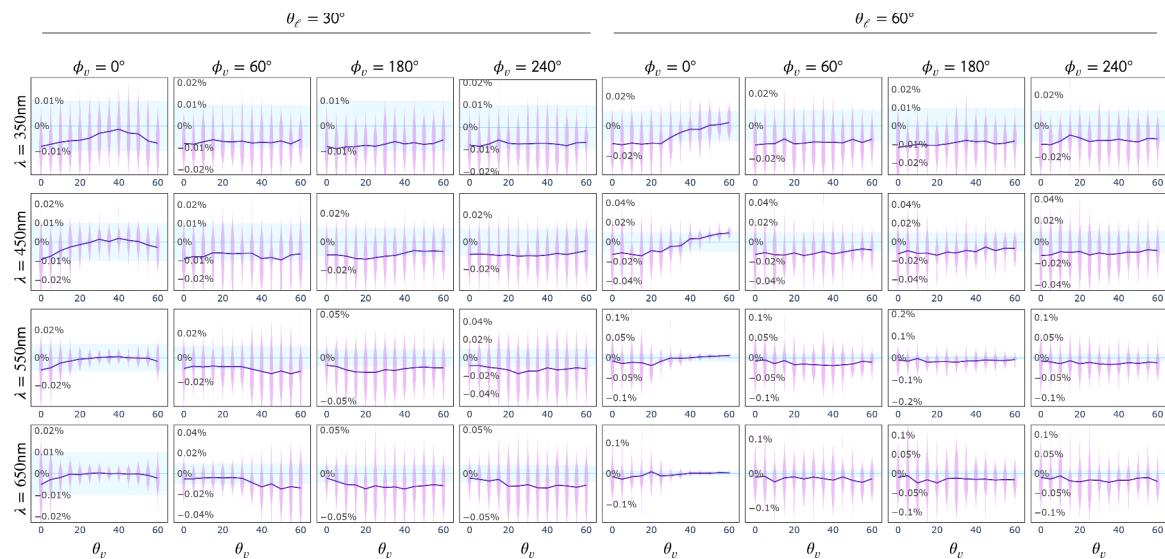


Fig. 7. Relative percent error in *reflectance* (ϵ_R) for Chowdhary et al. [5] AOS-I model at *top of atmosphere*, for all parameter sets. The mean of each set of 20 random trials is connected by dark purple line and a kernel density plot (“violin plot”) illustrating the spread of each set of trials is shown in light purple for each parameter set. Our target error threshold of $|\epsilon_R| < 0.01\%$ is shown in light blue.

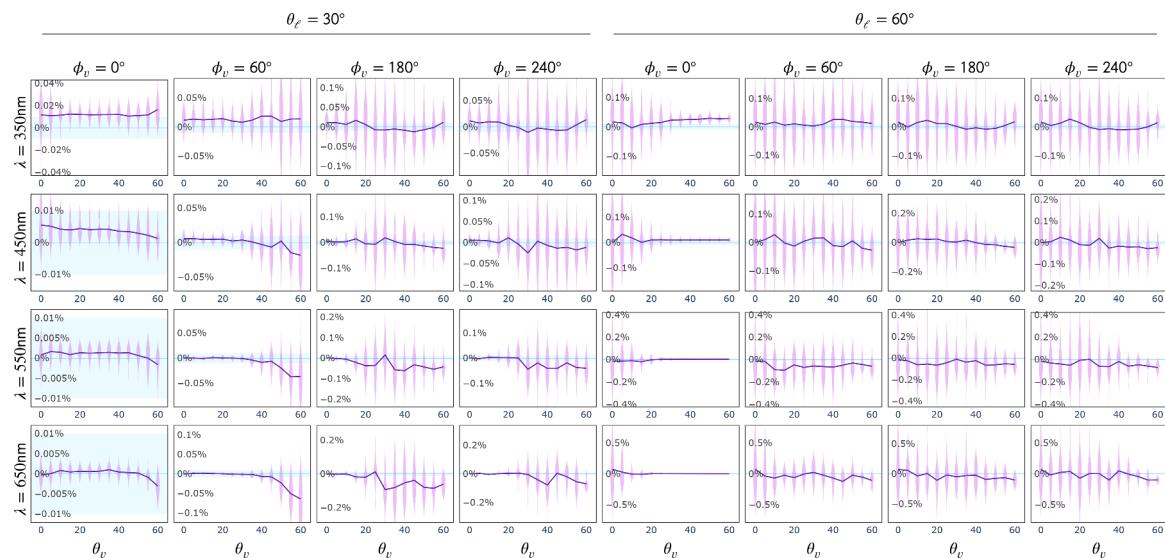


Fig. 8. Same as Fig. 7, but with sensor at *just above surface level* instead of *top of atmosphere*.

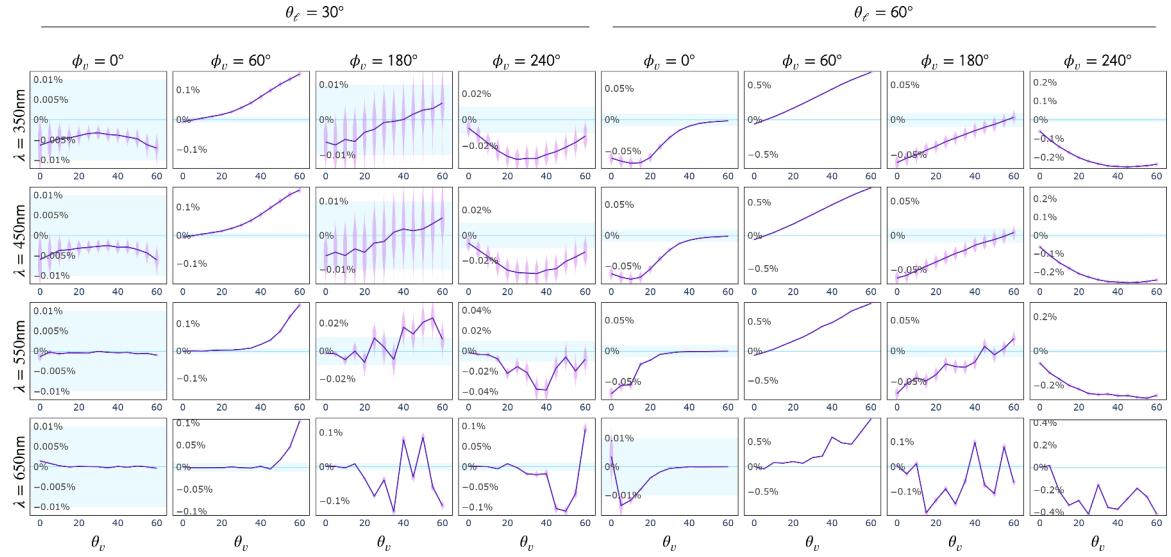


Fig. 9. Same as Fig. 7, but for AOS-II model (note that there is no distinction between just above surface and top of atmosphere for AOS-II because there is no atmosphere).

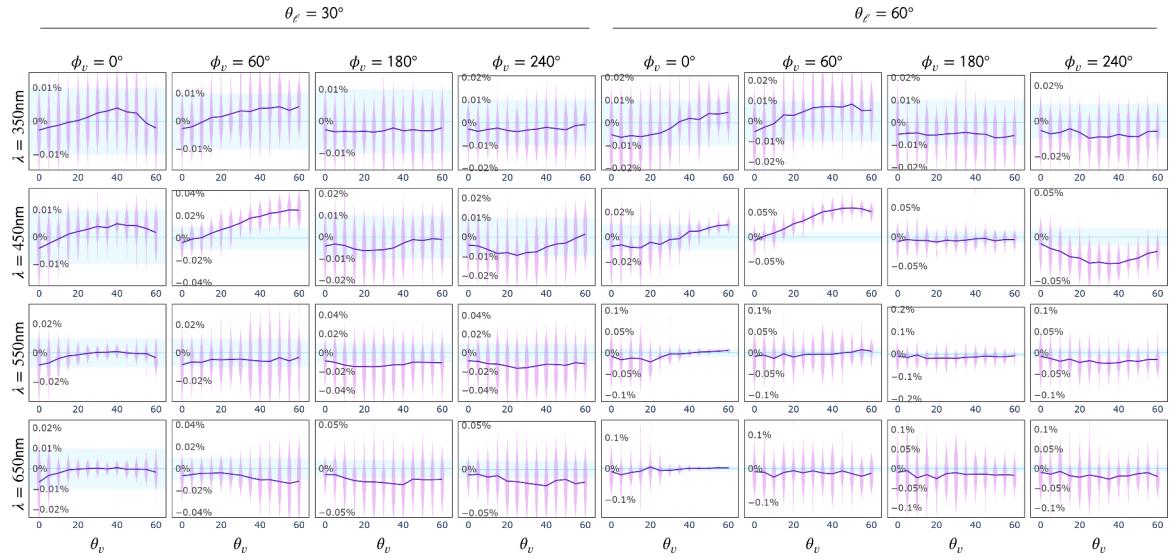


Fig. 10. Same as Fig. 7, but for AOS-III model at top of atmosphere.

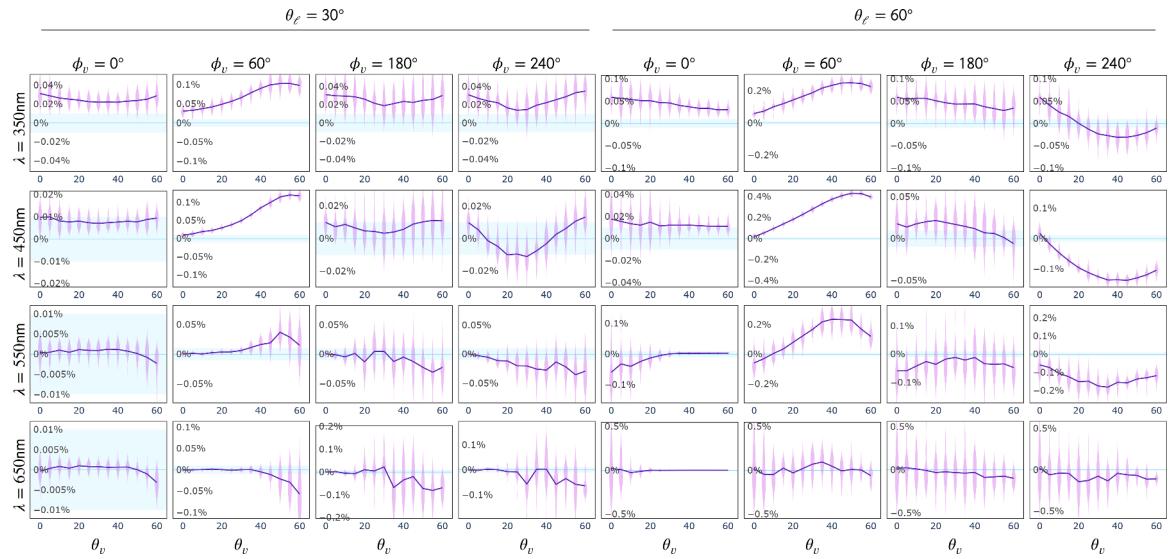


Fig. 11. Same as Fig. 7, but for AOS-III model at just above surface.

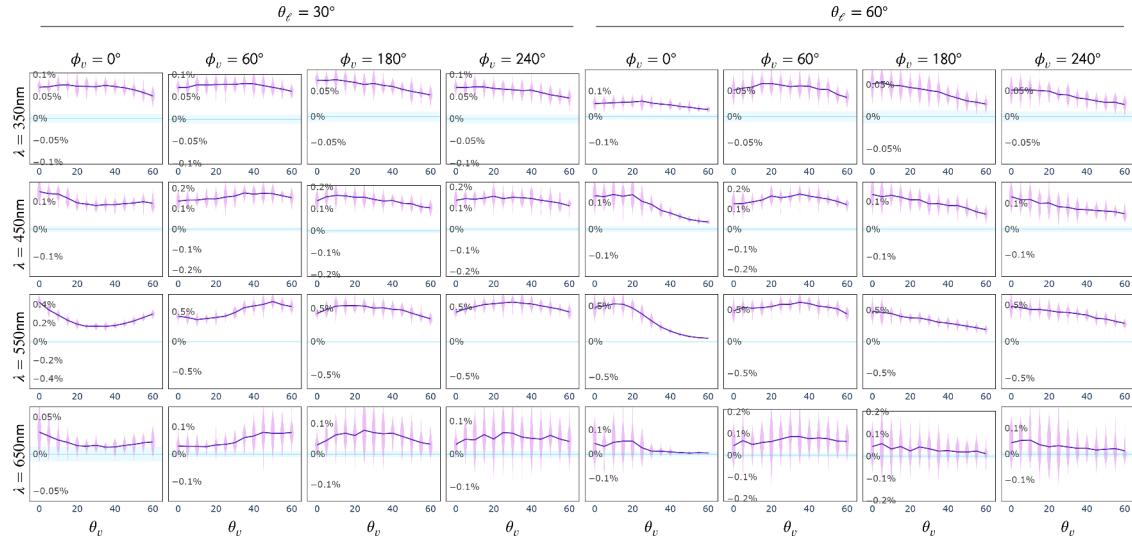


Fig. 12. Same as Fig. 7, but for AOS-IV model at top of atmosphere.

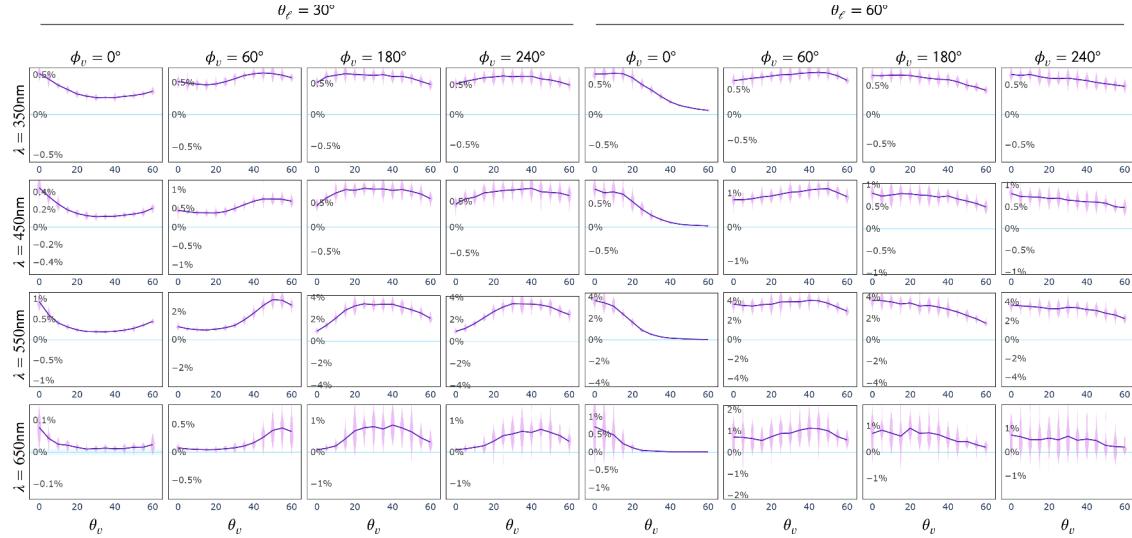


Fig. 13. Same as Fig. 7, but for AOS-IV model at just above surface.

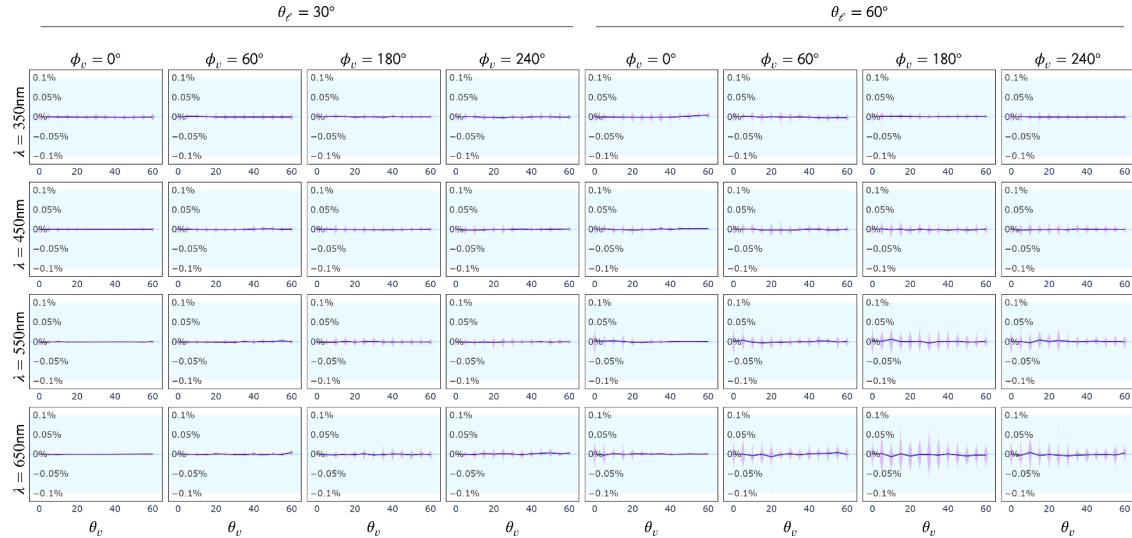


Fig. 14. Absolute error in degree of linear polarization (ϵ_{DLP}) for Chowdhary et al. [5] AOS-I model at top of atmosphere, for all parameter sets. The mean of each set of 20 random trials is connected by dark purple line and a kernel density plot (“violin plot”) illustrating the spread of each set of trials is shown in light purple for each parameter set. Our target error threshold of $|\epsilon_{\text{DLP}}| < 0.1\%$ is shown in light blue.

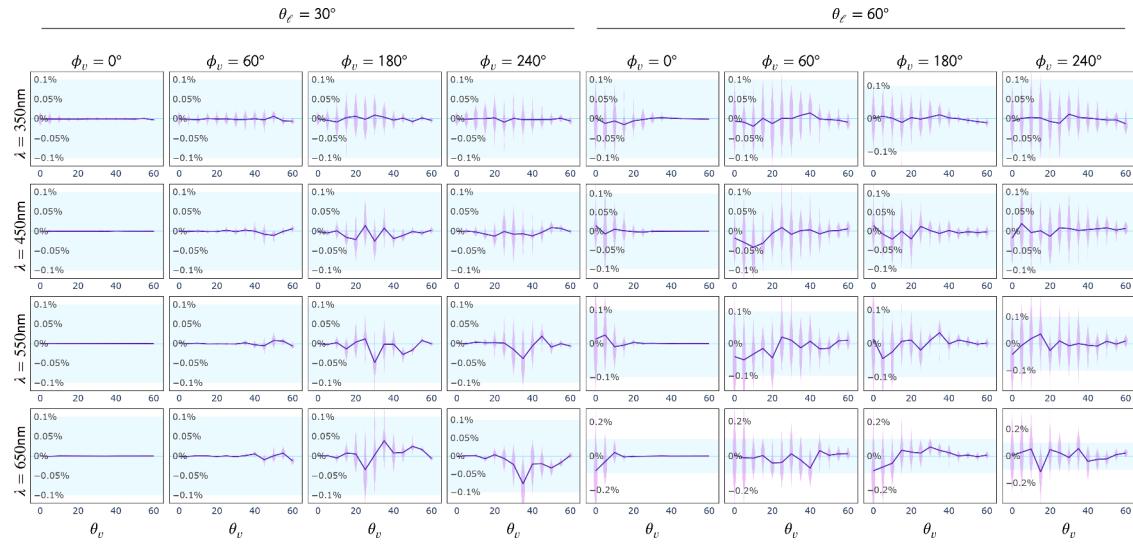


Fig. 15. Same as [fig. 14](#), but with sensor at just above surface level instead of top of atmosphere.

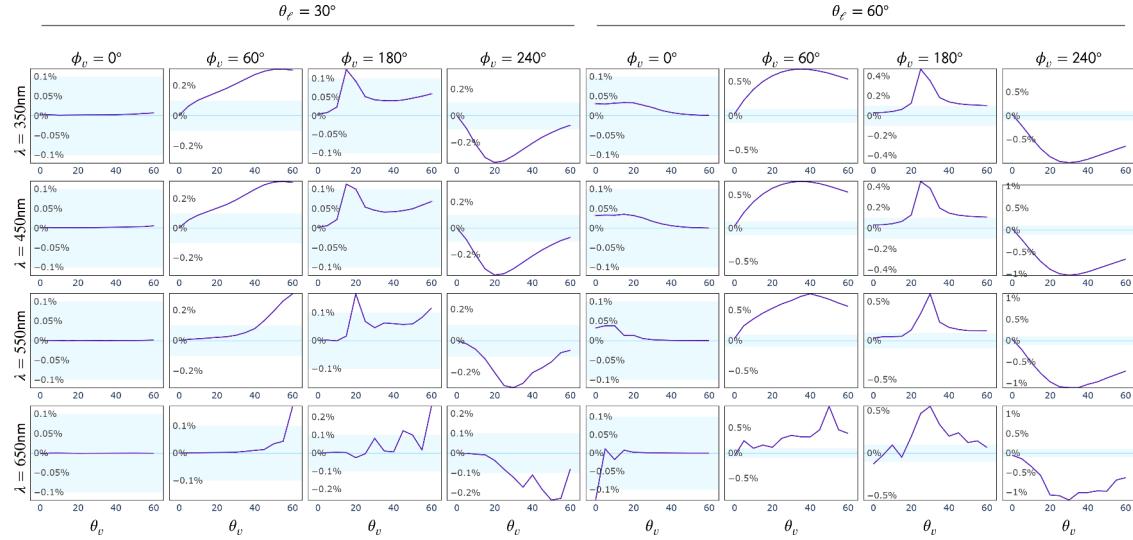


Fig. 16. Same as [Fig. 14](#), but for AOS-II model (note that there is no distinction between just above surface and top of atmosphere for AOS-II because there is no atmosphere).

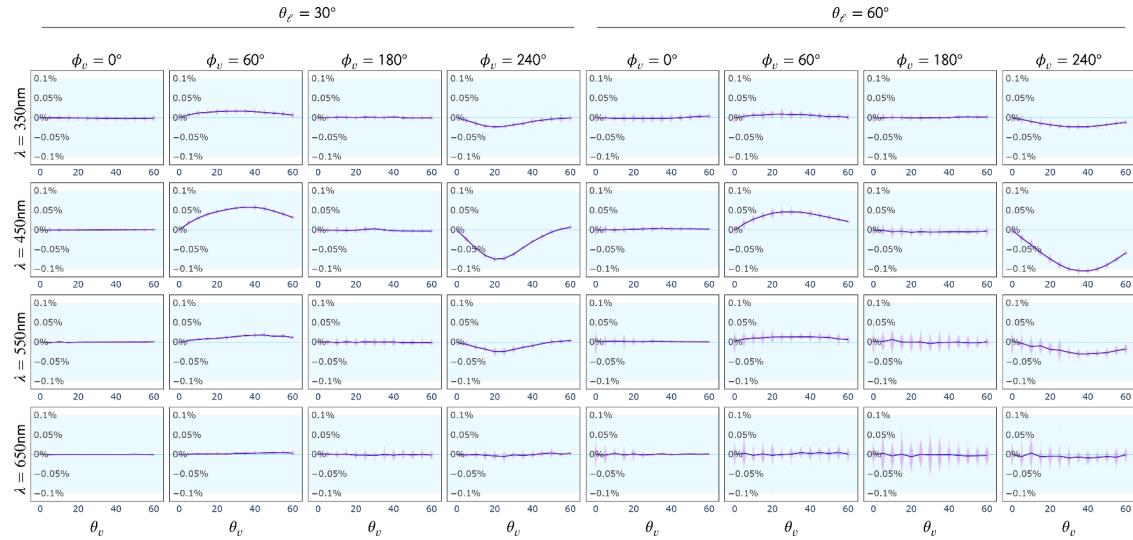


Fig. 17. Same as [Fig. 14](#), but for AOS-III model at top of atmosphere.

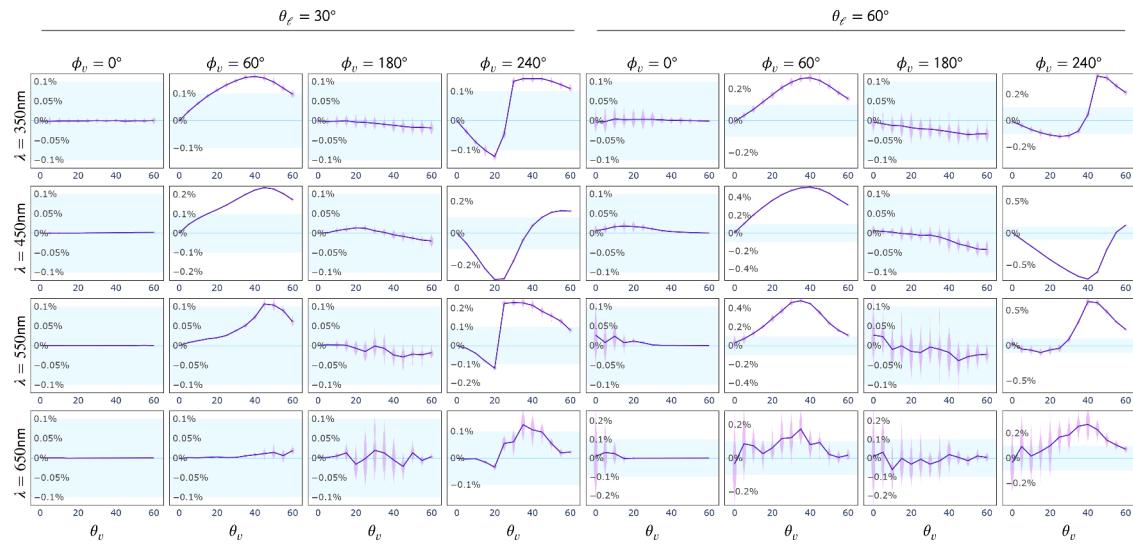


Fig. 18. Same as Fig. 14, but for AOS-III model at just above surface.

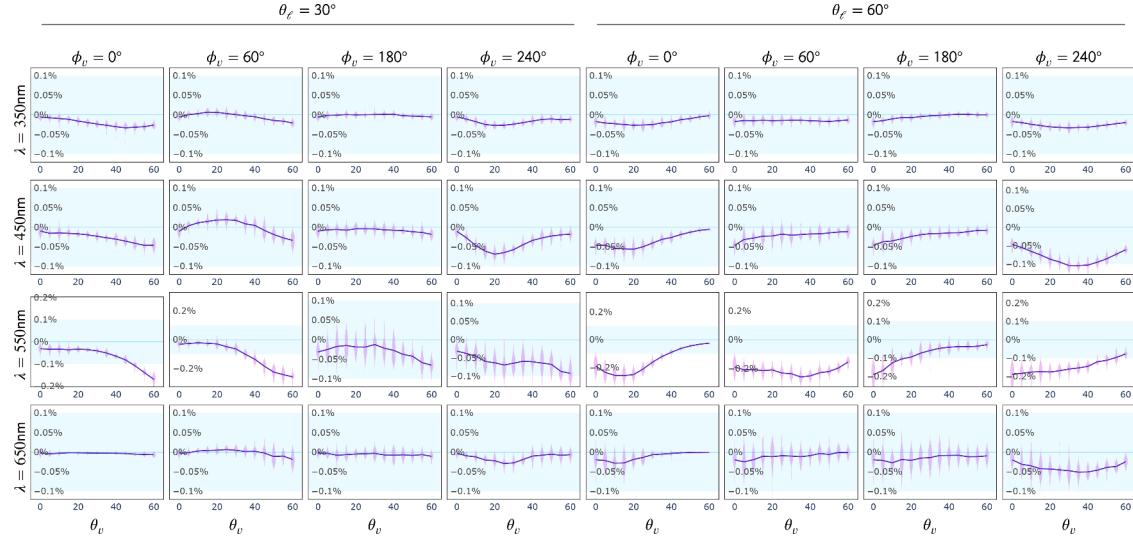


Fig. 19. Same as Fig. 14, but for AOS-JV model at top of atmosphere.

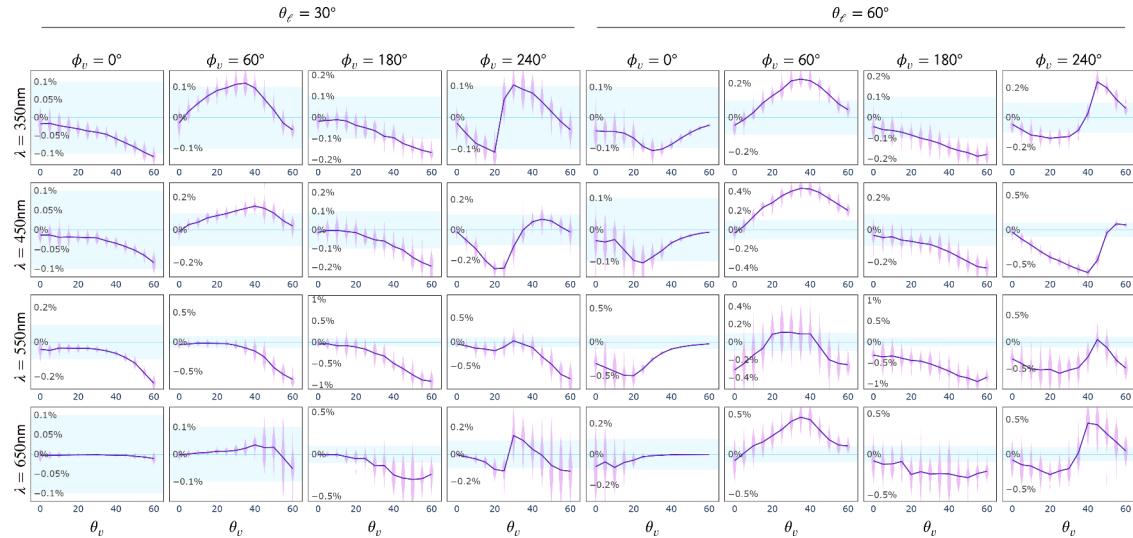


Fig. 20. Same as Fig. 14, but for AOS-JV model at just above surface.

Lastly, a guiding principle of this work is *accessibility*, so our framework will be made available publicly and promptly at the first author's Github page so that it can be used by the scientific community.

Declaration of Competing Interest

None.

Acknowledgments

The authors thank Wenzel Jakob and the rest of the Mitsuba 3 development team for consultations on Mitsuba 3. We also thank Meng Gao for insightful discussions on radiative transfer codes. Funding: This work was supported by the National Aeronautics and Space Administration (NASA) [grant number 80NSSC22K1154]; the National Science Foundation (NSF) grant number 1844538; and a Neukom Institute CompX faculty grant.

References

- [1] Jakob W., Speierer S., Roussel N., Nimier-David M., Vicini D., Zeltner T., Nicolet B., Crespo M., Leroy V., Zhang Z.. Mitsuba 3 renderer. 2022a. <https://www.mit-suba-renderer.org>.
- [2] Jakob W., Speierer S., Roussel N., Vicini D. DR.JIT: a just-in-time compiler for differentiable rendering. ACM Transactions on graphics (Proceedings of SIGGRAPH). 41; 2022. p. 124:1-124:19.10/gqjn7p
- [3] Zhai P-W, Kattawar GW, Hu Y. Comment on the transmission matrix for a dielectric interface. J Quant Spectrosc Radiat Transf 2012;113(16):1981–4.10/f4bthc
- [4] Garcia R. Response to “comment on the transmission matrix for a dielectric interface”. J Quant Spectrosc Radiat Transf 2012;113(17):2251–4.10/f4dsz3
- [5] Chowdhary J, Zhai P-W, Xu F, Frouin R, Ramon D. Testbed results for scalar and vector radiative transfer computations of light in atmosphere-ocean systems. J Quant Spectrosc Radiat Transf 2020;242:106717.10/kn5j
- [6] Gassó S, Knobelispiese KD. Circular polarization in atmospheric aerosols. Atmos Chem Phys 2022;22(20):13581–605.10/kn6p
- [7] Lafortune EP, Willems YD. Bi-directional path tracing. Proceedings of the international conference on computational graphics and visualization techniques (Compugraphics). vol. 93; 1993. p. 45–53.Alvor, Portugal
- [8] Pauly M, Kollig T, Keller A. Metropolis light transport for participating media. Rendering techniques (Proceedings of the eurographics workshop on rendering). Vienna: Springer-Verlag; 2000. p. 11–22.ISBN 3-211-83535-0,10/gfzm93
- [9] Jakob W., Marschner S. Manifold exploration: a Markov chain Monte Carlo technique for rendering scenes with difficult specular transport. ACM Transactions on graphics (Proceedings of SIGGRAPH). 31; 2012. p. 58:1-58:13.10/gfq4p
- [10] Jensen HW, Christensen PH. Efficient simulation of light transport in scenes with participating media using photon maps. Annual conference series (Proceedings of SIGGRAPH). ACM Press; 1998. p. 311–20.10/b64p36
- [11] Jarosz W, Zwicker M, Jensen HW. The beam radiance estimate for volumetric photon mapping. Computer graphics forum (Proceedings of Eurographics). 27; 2008. p. 557–66.10/bjsfsx
- [12] Jarosz W, Nowrouzezahrai D, Sadeghi I, Jensen HW. A comprehensive theory of volumetric radiance estimation using photon points and beams. ACM Trans Graph 2011;30(1). 5:1-5:19.10/fcdh2f
- [13] Jarosz W, Nowrouzezahrai D, Thomas R, Sloan P-P, Zwicker M. Progressive photon beams. ACM Transactions on graphics (Proceedings of SIGGRAPH Asia). 30; 2011. p. 181:1-181:12.10/fn5xzj
- [14] Walter B, Zhao S, Holzschuch N, Bala K. Single scattering in refractive media with triangle mesh boundaries. ACM Transactions on graphics (Proceedings of SIGGRAPH). 28; 2009. p. 1.10/bq66rh
- [15] Hanika J, Droske M, Fascione L. Manifold next event estimation. Computer graphics forum (Proceedings of the eurographics symposium on rendering). 34; 2015. p. 87–97.10/f7mbxk
- [16] Koerner D, Novak J, Kutz P, Habel R, Jarosz W. Subdivision next-event estimation for path-traced subsurface scattering. Proceedings of EGSR (Experimental ideas & implementations). Eurographics Association; 20166pages.ISBN 978-3-03868-019-2/10/gf6rzj
- [17] Mishchenko MI, Yang P. Far-field Lorenz-Mie scattering in an absorbing host medium: theoretical formalism and FORTRAN program. J Quant Spectrosc Radiat Transf 2018;205:241–52.10/gcpvrq
- [18] Walter B, Marschner SR, Li H, Torrance KE. Microfacet models for refraction through rough surfaces. Rendering techniques (Proceedings of the eurographics symposium on rendering). Eurographics Association; 2007. p. 195–206.ISBN 978-3-905673-52-4,10/gfz4kg
- [19] Heitz E, d'Eon E. Importance sampling microfacet-based BSDFs using the distribution of visible normals. Computer graphics forum (Proceedings of the eurographics symposium on rendering). 33; 2014. p. 103–12.10/f6fgph
- [20] Bitterli B, Jarosz W. Beyond points and beams: Higher-dimensional photon samples for volumetric light transport. ACM Transactions on graphics (Proceedings of SIGGRAPH). 36; 2017. 112:1-112:12.10/gfznbr
- [21] Mishchenko MI, Dlugach JM, Lock JA, Yurkin MA. Far-field Lorenz-Mie scattering in an absorbing host medium. II: improved stability of the numerical algorithm. J Quant Spectrosc Radiat Transf 2018;217:274–7.10/gd779h
- [22] Pharr M, Jakob W, Humphreys G. Physically based rendering: from theory to implementation. 4th ed. Cambridge, MA: MIT Press; 2023.ISBN 978-0-262-04802-6
- [23] Bodhaine BA, Wood NB, Dutton EG, Slusser JR. On Rayleigh optical depth calculations. J Atmos Oceanic Technol 1999;16(11):1854–61.10/cvt9q9
- [24] Sogandares FM, Fry ES. Absorption spectrum (340–640 nm) of pure water. I. Photothermal measurements. Appl Opt 1997;36(33):8699–709.10/dvx5sd
- [25] Pope RM, Fry ES. Absorption spectrum (380–700 nm) of pure water. II. Integrating cavity measurements. Appl Opt 1997;36(33):8710–23.10/d8rwcd
- [26] Cox C, Munk W. Measurement of the roughness of the sea surface from photographs of the Sun's glitter. J Opt Soc Am 1954;44(11):838–50.10/ddbn9x
- [27] Chowdhary J, Cairns B, Travis LD. Contribution of water-leaving radiances to multiangle, multispectral polarimetric observations over the open ocean: Bio-optical model results for case 1 waters. Appl Opt 2006;45(22):5542–67.10/d7t5n5
- [28] de Haan JF, Bosma PB, Hovenier JW. The adding method for multiple scattering calculations of polarized light. Astron Astrophys 1987;183(2):371–91.
- [29] Chowdhary J. Multiple scattering of polarized light in atmosphere-ocean systems: application to sensitivity analyses of aerosol polarimetry. New York, NY: Columbia University; 1999. Ph.D. thesis.
- [30] Hasekamp OP, Fu G, Rusli SP, Wu L, Di Noia A, aan de Brugh J, Landgraf J, Martijn Smit J, Rietjens J, van Amerongen A. Aerosol measurements by SPEXone on the NASA PACE mission: expected retrieval capabilities. J Quant Spectrosc Radiat Transf 2019;227:170–84.10/kn5g
- [31] Puthukkudy A, Martins JV, Remer LA, Xu X, Dubovik O, Litvinov P, McBride B, Burton S, Barbosa HMJ. Retrieval of aerosol properties from airborne hyper-angular rainbow polarimeter (AirHARP) observations during ACEPOL 2017. Atmos Meas Tech 2020;13(10):5207–36.10/kn6m
- [32] Dubovik O, Li Z, Mishchenko MI, Tanré D, Karol Y, Bojkov B, Cairns B, Diner DJ, Espinosa WR, Goloub P, Gu X, Hasekamp O, Hong J, Hou W, Knobelspiesse KD, Landgraf J, Li L, Litvinov P, Liu Y, Lopatin A, Marbach T, Maring H, Martins V, Meijer Y, Milinevsky G, Mukai S, Parol F, Qiao Y, Remer L, Rietjens J, Sano I, Stammes P, Stammes S, Sun X, Tabary P, Travis LD, Waquet F, Xu F, Yan C, Yin D. Polarimetric remote sensing of atmospheric aerosols: Instruments, methodologies, results, and perspectives. J Quant Spectrosc Radiat Transf 2019;224:474–511.10/kn6k
- [33] Leroy V., Schunke S., Nollet Y., Misk N., Govaerts Y.. Eradiate radiative transfer model. 2023. <https://www.eradiate.eu/>.
- [34] Vicini D, Speierer S, Jakob W. Path replay backpropagation: Differentiating light paths using constant memory and linear time. ACM Transactions on graphics (Proceedings of SIGGRAPH). 40; 2021. 108:1-108:14.10/gnngwp
- [35] Kingma D.P., Ba J.. Adam: a method for stochastic optimization. arXiv:14126980 [cs]2014;1412.6980.
- [36] Czerninski I, Schechner YY. PARS - Path recycling and sorting for efficient cloud tomography. Intell Comput 2023;2.10/gr2hg7
- [37] Nicolai B, Roussel F, Novák J, Keller A, Jakob W, Müller T. Recursive control variates for inverse rendering. ACM Transactions on graphics (Proceedings of SIGGRAPH). 42; 2023.10/kn6t
- [38] Chang W, Sivaram V, Nowrouzezahrai D, Hashisuka T, Ramamoorthi R, Li T-M. Parameter-space ReSTIR for differentiable and inverse rendering. ACM SIGGRAPH Conference papers. Los Angeles, CA: ACM Press; 2023.10/kn67