

Path Tracing Estimators for Refractive Radiative Transfer

ADITHYA PEDIREDLA, YASIN KARIMI CHALMIANI, MATTEO GIUSEPPE SCOPELLITI, MAYSAM CHAMANZAR, SRINIVASA NARASIMHAN, and IOANNIS GKIOULEKAS, Carnegie Mellon University, USA

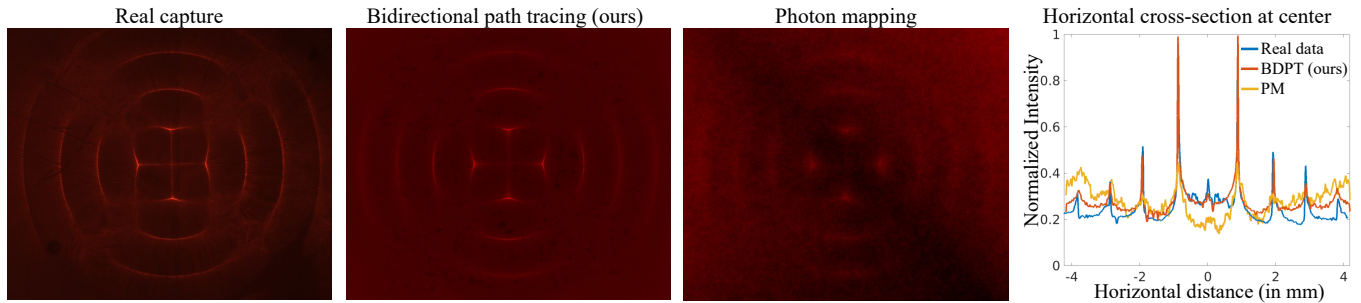


Fig. 1. **Simulating refractive radiative transfer using bidirectional path tracing.** Our technique enables path tracing estimators such as particle tracing, next-event estimation, bidirectional path tracing (BDPT) for heterogeneous refractive media. Unlike photon mapping (PM), path tracing estimators are unbiased and do not introduce artifacts besides noise. The above teaser shows that the proposed path tracing technique matches the real data acquired from an acousto-optic setup (ultrasound frequency: 813 kHz, 3 mean free paths, anisotropy $g = 0.85$) more accurately compared to photon mapping. Notice the complex caustics formed in a scattering medium whose refractive index field is altered using ultrasound. For more details and the reasons behind capture system imperfections, see Section 7.2. Our renderer can accelerate developments in acousto-optics, photo-acoustics, and schlieren imaging applications. The rendering time for BDPT and PM is 2.6 hrs on a cluster of 100 72-core (AWS c5n.18xlarge) machines.

Rendering radiative transfer through media with a heterogeneous refractive index is challenging because the continuous refractive index variations result in light traveling along curved paths. Existing algorithms are based on photon mapping techniques, and thus are biased and result in strong artifacts. On the other hand, existing unbiased methods such as path tracing and bidirectional path tracing cannot be used in their current form to simulate media with a heterogeneous refractive index. We change this state of affairs by deriving unbiased path tracing estimators for this problem. Starting from the refractive radiative transfer equation (RRTE), we derive a path-integral formulation, which we use to generalize path tracing with next-event estimation and bidirectional path tracing to the heterogeneous refractive index setting. We then develop an optimization approach based on fast analytic derivative computations to produce the point-to-point connections required by these path tracing algorithms. We propose several acceleration techniques to handle complex scenes (surfaces and volumes) that include participating media with heterogeneous refractive fields. We use our algorithms to simulate a variety of scenes combining heterogeneous refraction and scattering, as well as tissue imaging techniques based on ultrasonic virtual waveguides and lenses. Our algorithms and publicly-available implementation can be used to characterize imaging systems such as refractive index microscopy, schlieren

imaging, and acousto-optic imaging, and can facilitate the development of inverse rendering techniques for related applications.

CCS Concepts: • **Computing methodologies** → **Computational photography; Rendering.**

Additional Key Words and Phrases: eikonal equation, refractive radiative transfer, gradient index

ACM Reference Format:

Adithya Pediredla, Yasin Karimi Chalmiani, Matteo Giuseppe Scopelliti, Maysam Chamanzar, Srinivasa Narasimhan, and Ioannis Gkioulekas. 2020. Path Tracing Estimators for Refractive Radiative Transfer. *ACM Trans. Graph.* 39, 6, Article 241 (December 2020), 15 pages. <https://doi.org/10.1145/3414685.3417793>

1 INTRODUCTION

Media characterized by continuously-varying refractive index, occurring due to variations in temperature and pressure, or due to heterogeneous mixing of materials, are common in the real world. Examples of such *heterogeneous refractive media* include the atmosphere, biological tissue, crystals, minerals, and transparent plastics. When light travels through such media, it follows curved trajectories due to the continuous refraction. This bending of light can be observed by shining a laser beam on a heterogeneous refractive medium, e.g., as shown in Figure 2.

Light transport in heterogeneous refractive media is described using the *refractive radiative transfer equation* (RRTE) [Ament et al. 2014; Ihrke et al. 2007] that, in addition to light bending due to continuous refraction, also models effects due to volumetric and surface scattering. The light bending effects make this equation significantly more challenging to simulate than its counterpart for homogeneous refractive media, the radiative transfer equation [Novák et al. 2018].

Authors' address: Adithya Pediredla, adithya.k.pediredla@gmail.com; Yasin Karimi Chalmiani, ykarimic@andrew.cmu.edu; Matteo Giuseppe Scopelliti, mscopell@andrew.cmu.edu; Maysam Chamanzar, mchamanz@andrew.cmu.edu; Srinivasa Narasimhan, srinivas@andrew.cmu.edu; Ioannis Gkioulekas, igkioule@andrew.cmu.edu, Carnegie Mellon University, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2020 Association for Computing Machinery.

0730-0301/2020/12-ART241 \$15.00

<https://doi.org/10.1145/3414685.3417793>

Existing rendering algorithms are based on photon mapping techniques; these algorithms are efficient but biased, and can introduce significant artifacts in the output images. By contrast, unbiased algorithms such as path tracing, particle tracing, and bidirectional path tracing, are inefficient or even completely intractable for rendering refractive radiative transfer.

The main challenge in applying these unbiased algorithms to heterogeneous refractive media is the difficulty of performing *direct connections* between two points inside such a medium. Direct connections are needed, e.g., for next event estimation in path tracing and particle tracing, and to connect the source and sensor subpaths in bidirectional path tracing. Connecting two points inside a homogeneous refractive medium is trivial—one only needs to trace the linear segment between the two points. By contrast, connecting two points inside a heterogeneous refractive medium requires finding paths, generally curved, starting and ending at the two points that are solutions of the *eikonal equation*. This equation accounts for the variable (inversely proportional to refractive index) speed of light inside heterogeneous refractive media. The solutions to the eikonal equation are paths satisfying Fermat’s principle, in that they are locally *stationary* with respect to the time it takes light to traverse them or, equivalently, with respect to *optical pathlength*.

In this paper, we address this challenge by developing techniques for computing these stationary paths, efficiently and without bias. Our technique is based on the fact that the eikonal equation, and the ray tracing equations derived from it, are differentiable. Therefore, we show that we can use efficient gradient-based optimization algorithms to search for paths that connect two points, while also satisfying the eikonal equation (Section 4). Special attention is required to account for the fact that, in heterogeneous refractive media, there may exist more than one stationary paths connecting two points: our optimization-based technique in its simple form would only find one of these paths, resulting in bias, and is not practical for enumerating all stationary paths. To ensure unbiasedness and maintain efficiency, we introduce a Monte Carlo technique that forms an unbiased estimate of the total throughput through multiple stationary paths, through repeated random reinitializations of our gradient-based optimization algorithm (Section 4.1). To further improve efficiency, we show how to accelerate direct connections using techniques similar to sphere tracing of signed distance functions [Hart 1996], and inside-outside tests based on fast winding numbers [Barill et al. 2018] (Section 5).

The ability to perform direct connections inside heterogeneous refractive media allows us to extend unbiased rendering algorithms, such as path tracing with next-event estimation and bidirectional path tracing, to simulate refractive radiative transfer. We have implemented these algorithms within a physically-based renderer [Jakob 2010], and validated the accuracy of our implementation using simple scenes with known solutions, as well as real measurements (Figure 1, Section 7). Compared with previous photon mapping techniques, our algorithms can accurately and efficiently reproduce important effects, such as surface and volumetric caustics from *gradient-index* (GRIN) optics. Our algorithms are also straightforward to extend for time-of-flight rendering and spectral rendering.

Besides faithfully rendering visual phenomena, our unbiased rendering algorithms can be an invaluable tool to characterize scientific

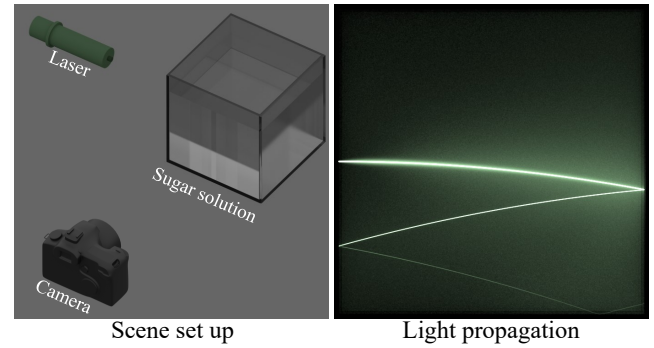


Fig. 2. **Light bending.** This scene is inspired from Ament et al. [2014]. The aquarium shown to the left is filled with a sugar solution whose refractive index increases linearly with the liquid’s depth. A laser beam propagating through this solution refracts and scatters continuously, resulting in a curved light trajectory. Multiple such trajectories appear due to Fresnel reflection of the beam on the aquarium’s wall. The rendering time for this scene is 5.5 hours on a 72-core (AWS c5n.18xlarge) machine.

imaging systems such as refractive index microscopy [Sun et al. 2014], schlieren imaging [Settles and Hargather 2017], and acousto-optic techniques for light delivery and imaging [Chamanzar et al. 2019] that observe, induce, or exploit refractive index heterogeneity. For example, consider the design of virtual optical elements such as waveguides and lenses, formed inside the medium by acousto-optic systems. In these systems, ultrasound induces controlled changes of the refractive index field in a scattering medium (e.g., tissue), and as a consequence, the wavefront of propagating light can be modulated. A beam of light can be confined and steered deep inside the medium [Scopelliti and Chamanzar 2019], and in addition, can be spatially modulated to generate complex spatial patterns for targeted light delivery [Karimi et al. 2019] inside the scattering medium. However, the ability of these systems to achieve desired light throughput depends on optimizing several parameters, including the choice of frequency, voltage, and phase coding of the ultrasound system [Chamanzar et al. 2019]. Our algorithms and implementation are well-suited to emulate such systems: We use our implementation to simulate acousto-optic in-situ light modulation, and reproduce real experimental data much more closely than what was possible by previous photon mapping techniques (Figure 1). We expect that our publicly-available implementation [Pediredla et al. 2020] will be useful for characterizing many imaging systems, developing inverse rendering techniques, or even training deep neural networks for problems involving refractive radiative transfer.

2 RELATED WORK

Rendering heterogeneous refractive media. In geometric optics, the eikonal equation can be used to describe the propagation of light in a medium where the refractive index varies continuously as a function of location [Kravtsov and Orlov 1990]. In particular, given an initial location and direction, the eikonal equation can be transformed into a Hamiltonian system of two partial differential equations, which in turn can be used to trace the path light will follow inside the medium [Muñoz 2014; Stam and Languénou 1996]. These paths will generally be continuous curves, and not straight lines as in a medium with a constant refractive index. Computing

these non-linear light paths typically involves using expensive numerical integration techniques [Sanz-Serna 1992], which introduces a severe, and often prohibitive, computational overhead for rendering algorithms intended to simulate images of light interacting with heterogeneous refractive media. This has motivated the development of techniques for accelerating non-linear ray tracing, including specialized data structures [Ihrke et al. 2007] and analytical approximations to the solutions of Hamilton’s equations [Mo et al. 2015; Steinberg 2020]. Non-linear ray tracing has also been used to visualize gravitational lensing [Gröller 1995; Weiskopf et al. 2004].

Heterogeneous refractive index fields can be encountered not just in transparent, but also in volumetric media. Light traveling through such media not only refracts continuously, but additionally *scatters* at discrete locations, due to interactions with medium microstructures. This complex light transport process can be modeled mathematically using the *refractive radiative transfer equation* (RRTE), a generalization of the standard radiative transfer equation (RTE) [Ishimaru 1978], that accounts for radiometric effects due to both volumetric scattering and continuous refraction [Ament et al. 2014]. Whereas research in computer graphics has produced an array of physically-accurate and efficient algorithms for simulating the RTE (e.g., volumetric path tracing and bidirectional path tracing [Lafortune and Willemis 1996; Novak et al. 2018]), rendering algorithms for simulating the RRTE remain limited. A fundamental challenge in generalizing volumetric path tracing techniques to the RRTE is that these techniques rely critically (e.g., for next-event estimation, or to connect the source and sensor subpaths) on the ability to efficiently compute *direct connections* between points in the medium, i.e., find the path that light will follow to travel from one point inside the medium to the other. When the medium has a constant refractive index, a direct connection is equivalent to simply computing the line connecting the two points. By contrast, when the medium has a heterogeneous refractive index, performing the direct connection requires solving the eikonal equation. Existing RRTE rendering techniques sidestep this issue by adapting photon mapping techniques, which replace direct connections with kernel density estimation [Ament et al. 2014; Cao et al. 2010; Gutierrez et al. 2005, 2006; Haber et al. 2005]. Unfortunately, these algorithms, while efficient, are biased, and can result in images with strong artifacts. Our key technical contribution is a computationally-efficient procedure for performing direct connections in heterogeneous refractive media. In turn, this makes it possible for the first to develop unbiased volumetric path tracing techniques for the efficient and physically-accurate simulation of the RRTE.

Differentiation of specular and refractive ray tracing. At the core of our technique for computing direct connections is the observation that Hamilton’s equations for non-linear ray racing are *differentiable*. Therefore, we can use efficient gradient descent techniques to search for paths that satisfy these equations and, simultaneously, connect two specified points. Our approach can be seen as a generalization of techniques previously-developed in computer graphics to tackle challenging specular and refractive rendering problems. In particular, Chen and Arvo [2000b] showed that the equations controlling how light can travel from one point to another through

an intermediate *specular* reflection are differentiable. The associated derivatives have been used to interactively render reflections of curved reflectors [Chen and Arvo 2000a; Mitchell and Hanrahan 1992], as well as enable sampling of paths with complex specular interactions in general-purpose Markov chain Monte Carlo rendering algorithms [Jakob and Marschner 2012; Kaplanyan et al. 2014]. More recently, these ideas have been used to enable next-event-estimation connections through specular reflections [Hanika et al. 2015; Zeltner et al. 2020], including techniques that account for the possible existence of multiple such connections. Likewise, Walter et al. [2009] showed that the equations governing the process of connecting two points through an intermediate *refractive* interface are differentiable, and used the associated derivatives to accelerate the rendering of caustics in optically-thin media. In both cases, the underlying equations are equivalent to the eikonal equation, restricted to allowing bending of light only at a specific surface. We extend these ideas to the more general form of the eikonal equation, allowing for continuous bending of light at any spatial point.

Imaging techniques involving heterogeneous refractive media. The ability of heterogeneous refractive media to steer light along non-linear paths has been at the core of a variety of imaging techniques. In particular, *gradient-index* (GRIN) optics implement specific heterogeneous refractive index profiles that mimic or extend the focusing properties of a lens. Examples include lenses and lightguides with parabolic refractive profiles [Kapron 1970; Sharma and Ghatak 1981], Maxwell’s fish-eye lens [Merlin 2011], and the Luneburg lens [Luneburg 1966]. GRIN optics can serve as a replacement for standard optics in applications requiring minimal aberrations, or in applications using wavelengths where it is impossible to manufacture regular lenses. More recently, ultrasonic techniques have emerged that mechanically induce pre-programmed refractive index profiles in media that would normally have a homogeneous refractive index. This capability can be used to, e.g., create varifocal lenses that support very high tuning frequencies [Mermillod-Blondin et al. 2008], or to *sculpt* waveguides inside media such as tissue where it is impossible to place conventional lenses [Chamanzar et al. 2019]. We re-visit these techniques in our experiments section, where we show how our algorithms can be used to simulate their applications.

The ubiquity of heterogeneous refractive media in nature has also motivated significant research towards imaging techniques that can extract information about the heterogeneous refractive index profiles underlying image measurements. For example, schlieren photography is a classical technique for visualizing the propagation of light in gasses and other heterogeneous refractive media [Dalziel et al. 2000; Howes 1984; Krehl and Engemann 1995; Settles and Hargather 2017; Wetzstein et al. 2011]. Other techniques use measurements similar to those from schlieren photography, together with dedicated inverse algorithms, to estimate the three-dimensional refractive index profile [Atcheson et al. 2008; Ji et al. 2013; Ma et al. 2014; Xue et al. 2014]. Closely related are techniques in seismic imaging, which exploit the similarity between the propagation of seismic waves below the Earth’s surface and the propagation of light in heterogeneous refractive media, to create subsurface images [Robein 2010]. In particular, travel-time tomography techniques

Algorithm 1: Symplectic integration for ray tracing**Input:** $n(\mathbf{x})$, $\nabla n(\mathbf{x})$, $\text{ray}(\mathbf{x}, \mathbf{v})$, $n\text{Steps}$, $s = \text{step size}$ **Output:** $\text{ray}(\mathbf{x}, \mathbf{v})$ **for** $i = 1 : n\text{Steps}$ **do** $\text{ray.v} += 0.5s \nabla n(\text{ray.x});$ $\text{ray.x} += s \frac{\text{ray.v}}{n(\text{ray.x})};$ $\text{ray.v} += 0.5s \nabla n(\text{ray.x});$ **end**

recover the unknown subsurface refractive index values using gradient optimization, which requires differentiating through the eikonal equation [Benmansour et al. 2010; Leung et al. 2006].

3 BACKGROUND ON REFRACTIVE RADIATIVE TRANSFER

In this section, we provide background on geometric optics and light transport inside media with a heterogeneous refractive index. For details, we refer to Kravtsov and Orvol [1990] for the former, and Ament et al. [2014] for the latter.

3.1 Non-linear ray tracing

The propagation of light inside a medium with continuously-varying refractive index (*heterogeneous refractive medium*) can be described using the eikonal equation. This equation accounts for the fact that the speed of light varies continuously inside the medium, and is inversely proportional to the refractive index. In particular, we can define at any point $\mathbf{x}$ inside the medium the *vector velocity* of light $\mathbf{v}(\mathbf{x}) \equiv \omega(\mathbf{x})/n(\mathbf{x})$, where $\omega(\mathbf{x})$ is a three-dimensional unit-norm vector indicating direction of propagation, and $n(\mathbf{x})$ is the local refractive index. Then, we can rewrite the eikonal equation equivalently as the Hamilton's equations [Ihrke et al. 2007]:

$$\frac{d\mathbf{v}}{ds} = \nabla n(\mathbf{x}), \quad (1)$$

$$\frac{d\mathbf{x}}{ds} = \frac{\mathbf{v}}{n(\mathbf{x})}, \quad (2)$$

where ds is the infinitesimal arc length. Then, given an initial position $\mathbf{x}$ and velocity $\mathbf{v}$, we can trace the path light will follow through the medium by evolving these two equations. The resulting paths will typically be curves, rather than straight lines, and will correspond to *stationary points* (local minima, local maxima, or saddle points) of the *optical pathlength* $\tau \equiv \int n(\mathbf{x}(s)) ds$, where integration is along the path. We thus refer to these paths as *stationary paths*.

In practice, given initial $\mathbf{x}$ and $\mathbf{v}$, we cannot evolve Hamilton's equations analytically except for very simple heterogeneous refractive media. Instead, the two equations need to be solved numerically. Throughout this paper, we use a first-order symplectic integrator to solve for ray propagation. Symplectic integrators are numerically-stable algorithms developed specifically for Hamiltonian systems such as the one defined through Equations (1)-(2) [Sanz-Serna 1992]. Compared to other numerical integration techniques (e.g., Runge-Kutta techniques used by Ament et al. [2014] and Ihrke et al. [2007]), symplectic integrators remain stable and have low error for longer ray propagation distances. We provide details on the integration

procedure in Algorithm 1. Throughout the paper, we use the term *non-linear ray tracing* to refer to the process of using symplectic integration to propagate a ray from the initial position $\mathbf{x}$ and velocity $\mathbf{v}$, in a way that satisfies Hamilton's equations (1)-(2).

3.2 Refractive radiative transfer equation

Light transport inside a scattering medium is typically modeled in computer graphics using the *radiative transfer equation* (RTE) [Chandrasekhar 2013]. The RTE models light emission, absorption, and scattering, as a function of the medium's *absorption coefficient* σ_a , *scattering coefficient* σ_s , and *phase function* f_s . The *refractive radiative transfer equation* (RRTE) [Ament et al. 2014] extends the RTE to model light transport in a medium that additionally has heterogeneous refractive index $n(\mathbf{x})$.

The RRTE employs basic radiance $\bar{L}(\mathbf{x}, \omega) = L(\mathbf{x}, \omega)/n(\mathbf{x})^2$ instead of radiance. Ament et al. [2014] show that basic radiance is conserved as light travels across stationary paths in a transparent heterogeneous refractive medium; this is analogous to the conservation of radiance along linear paths in a transparent homogeneous medium. The RRTE can be written as:

$$\begin{aligned} \frac{d\bar{L}(\mathbf{x}, \omega)}{ds} = & -(\sigma_a + \sigma_s)\bar{L}(\mathbf{x}, \omega) + \sigma_a \bar{L}_e(\mathbf{x}, \omega) \\ & + \sigma_s \int_{\omega'} f_s(\mathbf{x}, \omega' \rightarrow \omega) \bar{L}(\mathbf{x}, \omega') d\omega'. \end{aligned} \quad (3)$$

The RRTE is similar to RTE, with radiance replaced with basic radiance. Another key difference is the streaming differential ds . In a homogeneous refractive medium, where light propagates along (piecewise-) linear paths, ds reduces to directional derivatives (where direction is $\omega = \mathbf{v}/\|\mathbf{v}\|$). However, due to the curved light paths in a heterogeneous refractive medium, ds must also account for change of propagation direction and velocity along a path. Therefore:

$$\frac{d\bar{L}}{ds} = \frac{\partial \bar{L}}{\partial \mathbf{x}} \frac{d\mathbf{x}}{ds} + \frac{\partial \bar{L}}{\partial \mathbf{v}} \frac{d\mathbf{v}}{ds}. \quad (4)$$

3.3 Path-integral formulation of refractive radiative transfer

The RRTE in its integro-differential form (3) does not directly lend itself to the derivation of Monte Carlo rendering algorithms. Additionally, real-world scenes typically contain surface geometry and homogeneous refractive volumes, in addition to heterogeneous refractive volumes. To develop Monte Carlo rendering algorithms for such general scenes, it will be helpful to first derive a path-integral formulation for refractive radiative transfer. We provide here this derivation, which closely follows the derivation of the path integral formulation of radiative transfer [Novák et al. 2018; Veach 1997].

Refractive volume rendering equation. We begin by converting the RRTE into an integral form that we term the *refractive volume rendering equation*, in direct analogy with the volume rendering equation for homogeneous refractive media. Thanks to the similarity between the RRTE and RTE, this can be done using a derivation

similar to Novak et al. [2018], which gives us the equation:

$$\begin{aligned} \bar{L}(\mathbf{x}, \omega) = & \int_{s=0}^z T(\mathbf{x}, \mathbf{x}_s) \left[\sigma_a \bar{L}_e(\mathbf{x}_s, \omega_s) + \right. \\ & \left. \sigma_s(\mathbf{x}_s) \int \bar{f}_s(\mathbf{x}_s, \omega' \rightarrow \omega_s) \bar{L}(\mathbf{x}_s, \omega') d\omega' \right] ds \\ & + T(\mathbf{x}, \mathbf{x}_z) \bar{L}(\mathbf{x}_z, \omega_z), \end{aligned} \quad (5)$$

where:

- $\mathbf{x}_s$ is the point in the medium obtained by non-linear ray tracing using Equations (1)-(2), starting from initial position $\mathbf{x}$ and initial direction $-\omega$, and evolve Hamilton's equations for a geometric distance s .
- $T(\mathbf{x}, \mathbf{x}_s) = \exp^{-\int_0^s \mu_t(\mathbf{x}(s)) ds}$ is the volumetric transmittance along the curved path from $\mathbf{x}$ to $\mathbf{x}_s$.
- $\mathbf{x}_z$ is the point where non-linear ray tracing from initial position $\mathbf{x}$ and initial direction $-\omega$, first intersects the boundary of the medium, and $\mathbf{x}_z = \infty$ if the medium is infinite.

Importantly, we note that volumetric transmittance depends on geometric length and not optical length.

Refractive rendering equation. Radiative transfer for surfaces at a point $\mathbf{z}$ is given by the rendering equation [Immel et al. 1986; Kajiya 1986]. This equation can be modified for refractive radiative transfer by simply changing radiance to basic radiance:

$$\bar{L}(\mathbf{z}, \omega) = \bar{L}_e(\mathbf{z}, \omega) + \int \bar{L}_i(\mathbf{z}, \omega) f_r(\mathbf{z}, \omega' \rightarrow \omega) d\omega'^{\perp}. \quad (6)$$

Here, $\bar{L} = L$ if the surface is interfaced with air, and $\bar{L} = L/n(\mathbf{z})^2$ if the surface is immersed in a heterogeneous volume. $\omega'^{\perp}$ is the projected solid angle measure. $\bar{L}_i$ is the incident basic radiance and $\bar{L}_i(\mathbf{z}, \omega) = \bar{L}(x_{\mathcal{M}}(\mathbf{z}, \omega), -\omega)$, where $x_{\mathcal{M}}(\mathbf{z}, \omega)$ is the intersection of a ray, starting at point $\mathbf{z}$ and direction $-\omega$, with a surface $\mathcal{M}$.

Path integral. The intensity I measured by a sensor equals the inner product (with respect to a measure equal to the product of area and projected solid angle measures) of the sensor importance function $W_e(\mathbf{x}, \omega)$ and the incident radiance on the sensor. By combining and recursively expanding Equations (5)-(6) [Jakob 2016; Veach 1997], we obtain the following path-integral expression for refractive radiative transfer:

$$\begin{aligned} I = & |\langle W_e(\mathbf{x}, \omega), n(\mathbf{x})^2 \bar{L}_i(\mathbf{x}, \omega) \rangle| \\ = & \int_{\mathcal{P}} W_e(\mathbf{x}_0, \mathbf{x}_1) n(\mathbf{x}_k)^2 L_e(\mathbf{x}_k, \mathbf{x}_{k-1}) G(\mathbf{x}_0, \mathbf{x}_1) T(\mathbf{x}_0, \mathbf{x}_1) \\ & \prod_{k=1}^{K-1} G(\mathbf{x}_k, \mathbf{x}_{k+1}) T(\mathbf{x}_k, \mathbf{x}_{k+1}) f_c(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{x}_{k+1}) d\mu(\bar{\mathbf{x}}) \\ = & \int_{\mathcal{P}} f(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}), \end{aligned} \quad (7)$$

where $\bar{\mathbf{x}} = (\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{K-1}) \in \mathcal{P}$ is a light path with k bounces satisfying Equations (1)-(2) between bounces; $\mathcal{P}$ is the space of all such paths; $d\mu(\bar{\mathbf{x}})$ is the total measure of the path; and all other terms that depend on whether the point ($\mathbf{x}_k$) is in medium or on surface and their values are shown in Table 1.

Table 1. Definitions of terms for different types of connections.

symbol	$\mathbf{x}_k$ on surface	$\mathbf{x}_k$ inside medium
$f_c(\mathbf{x}, \mathbf{y}, \mathbf{z})$	$f_r(\mathbf{y}, \omega_{\overrightarrow{xy}} \rightarrow \omega_{\overrightarrow{yz}})$	$\sigma_s(\mathbf{y}) f_s(\mathbf{y}, \omega_{\overrightarrow{xy}} \rightarrow \omega_{\overrightarrow{yz}})$
$T(\mathbf{x}, \mathbf{y})$	1	$T(\mathbf{x}, \mathbf{y})$
$G(\mathbf{x}, \mathbf{y})$	$ N(\mathbf{x}) \cdot \omega_{\overrightarrow{xy}} N(\mathbf{y}) \cdot \omega_{\overrightarrow{yx}} / \ \mathbf{x} - \mathbf{y}\ ^2$	$\left(\int_0^y ds \right)^{-2}$

3.4 Monte Carlo estimators for refractive radiative transfer

Analytically evaluating the path integral in Equation (7) is typically not possible, except for trivial scenes. Monte Carlo rendering algorithms approximate Equation (7) as:

$$I \approx \sum_{n=1}^N \frac{f(\bar{\mathbf{x}}^{(n)})}{p(\bar{\mathbf{x}}^{(n)})}, \quad (8)$$

where $p(\cdot)$ is a probability distribution defined on path space $\mathcal{P}$, and $\{\bar{\mathbf{x}}^{(n)}\}_{n=1}^N$ are paths sampled from this distribution. Several path sampling techniques have been proposed in computer graphics for radiative transfer, including path tracing (PT) [Arvo 1995], bidirectional path tracing (BDPT) [Veach and Guibas 1995] and Metropolis light transport [Veach and Guibas 1997]. These techniques combine computational efficiency and physical accuracy, resulting in intensity estimates that are unbiased and consistent. Our focus in this paper is on the first two of techniques, which sample paths independently; we collectively refer to techniques of this kind as *path tracing estimators*. Our goal is to develop path tracing estimators for refractive radiative transfer. In Section 8, we discuss potential future work on extending Metropolis light transport techniques to the refractive radiative transfer setting.

Path tracing estimators work by first generating subpaths: in the case of PT, this is a single subpath starting at the sensor, whereas in the case of BDPT, there are two subpaths starting on the source and sensor. In radiative transfer, these subpaths are generated by alternating between linear ray tracing, and surface or volume scattering events. Analogously, in refractive radiative transfer, these subpaths are generated by alternating non-linear ray tracing and scattering events. Then, complete paths are formed by performing *direction connections*: in the case of PT, these are connections between vertices of the subpath and points on a light source—often termed next-event estimation—whereas in BDPT, these are connections between any two vertices on the source and sensor subpath. When these direct connections are between points in a homogeneous refractive medium, as is most commonly the case in radiative transfer, these connections require simply connecting the two points by a line. By contrast, performing a direct connection between points in a heterogeneous refractive medium, as needed in refractive radiative transfer, requires finding a curved path connecting the two points, while also satisfying Equations (1)-(2). Algorithms for performing direct connections of this type is the key missing component for enabling the use of path tracing estimators to simulate refractive radiative transfer. In the next section, we develop such an algorithm, which we then integrate into BDPT. Even though our discussion and results focus on the BDPT case, we note that our direct connection

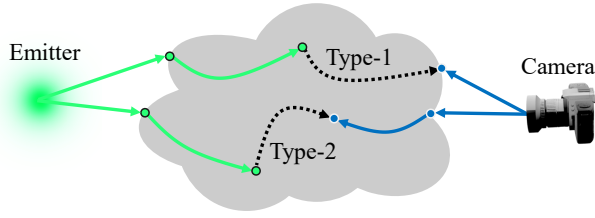


Fig. 3. **Types of direct connections.** Based on the location of the source (green) and sensor subpath (blue) ends, we have two different types of direct connections (dotted curves). **Type-1** direct connections have one vertex on the medium boundary and another one inside the medium. **Type-2** direct connections have both vertices inside the medium.

algorithm can be used to extend any other path tracing estimator to the refractive radiative transfer setting.

4 COMPUTING DIRECT CONNECTIONS

To join the source and sensor path ends, we need to find a curve that terminates at the given two vertices (source and sensor path ends) and satisfies Hamilton's equations (1)-(2) at all the points on the curve. In algebraic geometry, finding this curve is known as the *boundary value problem* (BVP), and non-linear ray tracing is known as the *initial value problem* (IVP). We will solve the BVP by solving several IVP problems. For this, we first identify various connections based on the vertex locations and propose techniques to compute the curve joining these vertices.

Figure 3 identifies two types of direct connections based on the location of the source and sensor path ends. While type-2 connections are a superset of type-1 connections, treating them as different gives run time advantages for type-1 connections. Also, type-1 connections are easier to derive than type-2 connections.

Type-1: A vertex at the boundary of the medium. In this case, one of the vertices is inside the medium and another on the boundary. This case occurs if the medium boundary is non-specular (rough or has texture associated with it) or if the surface geometry is present inside the heterogeneous refractive medium (i.e., participating medium).

Let $\mathbf{x}_0$ and $\mathbf{y}$ be the two endpoints of the source and sensor subpaths (need not be in the same order) that have to be connected via a refractive index field $n(\mathbf{x})$. We assume that we have available an implicit representation $g(\cdot) = 0$ for the surface boundary that $\mathbf{y}$ is located on. We have to determine initial velocity ($\mathbf{v}_0$) such that $\mathbf{x}_{s^*} = \mathbf{y}$, where $\mathbf{x}_{s^*}$ is obtained by solving initial value problem (ray propagation) till geometric distance s^* . For this, we define a loss function $\mathcal{L}(\mathbf{v}_0) = \frac{1}{2} \|\mathbf{x}_{s^*} - \mathbf{y}\|^2$ and optimize it using the BFGS algorithm, which requires the gradient of loss function ($d\mathcal{L}/d\mathbf{v}_0$). For a given initialization $\mathbf{v}_0^0$, we next show the procedure to compute the gradient $d\mathcal{L}/d\mathbf{v}_0$:

$$\frac{d\mathcal{L}}{d\mathbf{v}_0} = (\mathbf{x}_{s^*} - \mathbf{y})^T \frac{d\mathbf{x}_{s^*}}{d\mathbf{v}_0}, \quad (9)$$

$$\frac{d\mathbf{x}_{s^*}}{d\mathbf{v}_0} = \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0} + \frac{\partial \mathbf{x}_{s^*}}{\partial s^*} \frac{ds^*}{d\mathbf{v}_0}. \quad (10)$$

Algorithm 2: Symplectic integration for derivative tracing

Input: $n(\mathbf{x}), \nabla n(\mathbf{x}), H_n(\mathbf{x}), \text{ray}(\mathbf{x}, \mathbf{v}), \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0} = O_3, \frac{\partial \mathbf{v}_{s^*}}{\partial \mathbf{v}_0} = I_3,$
 $nSteps, s = \text{step size}$

Output: $\frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0}, \frac{\partial \mathbf{v}_{s^*}}{\partial \mathbf{v}_0}$

for $i = 1 : nSteps$ **do**

$\text{ray.v} += 0.5s\nabla n(\text{ray.x});$
 $\frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0} += 0.5sH_n(\text{ray.x}) \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0};$
 $\text{ray.x} += s \frac{\text{ray.v}}{n(\text{ray.x})};$
 $\frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0} += s \left(-\frac{\mathbf{v}_s \nabla n(\text{ray.x})}{n(\text{ray.x})^2} \frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} + \frac{1}{n(\text{ray.x})} \frac{\partial \mathbf{v}_s}{\partial \mathbf{v}_0} \right);$
 $\text{ray.v} += 0.5s\nabla n(\text{ray.x});$
 $\frac{\partial \mathbf{v}_{s^*}}{\partial \mathbf{v}_0} += 0.5sH_n(\text{ray.x}) \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0};$

end

Computing $\frac{ds^*}{d\mathbf{v}_0}$ is tricky as the geometric length (s^*) depends on $\mathbf{v}_0$ only at the solution of $\mathcal{L}$ but is not defined for other $\mathbf{v}_0$. To mitigate this problem, without loss of generality, we assume that $\mathbf{x}_{s^*}$ is a point on the medium's boundary. Therefore, $g(\mathbf{x}_{s^*}) = 0$, which makes s^* a function of $\mathbf{v}_0$ for all initial velocity values. We can now compute the missing $ds^*/d\mathbf{v}_0$ by differentiating:

$$\begin{aligned} g(\mathbf{x}_{s^*}) = 0 &\Rightarrow \frac{dg(\mathbf{x}_{s^*})}{d\mathbf{x}_{s^*}} \frac{d\mathbf{x}_{s^*}}{d\mathbf{v}_0} = 0 \\ &\Rightarrow \frac{ds^*}{d\mathbf{v}_0} = -\frac{\frac{dg(\mathbf{x}_{s^*})}{d\mathbf{x}_{s^*}} \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0}}{\frac{dg(\mathbf{x}_{s^*})}{d\mathbf{x}_{s^*}} \frac{\partial \mathbf{x}_{s^*}}{\partial s^*}}, \end{aligned} \quad (11)$$

$$\frac{d\mathbf{x}_{s^*}}{d\mathbf{v}_0} = \left(I_{3 \times 3} - \frac{\frac{\partial \mathbf{x}_{s^*}}{\partial s^*} \frac{dg(\mathbf{x}_{s^*})}{d\mathbf{x}_{s^*}}}{\frac{dg(\mathbf{x}_{s^*})}{d\mathbf{x}_{s^*}} \frac{\partial \mathbf{x}_{s^*}}{\partial s^*}} \right) \frac{\partial \mathbf{x}_{s^*}}{\partial \mathbf{v}_0}. \quad (12)$$

The derivatives of the ray at distance s^* : ($\partial \mathbf{x}_{s^*}/\partial \mathbf{v}_0$) are given in Appendix A. The derivatives ($\partial \mathbf{x}_{s^*}/\partial \mathbf{v}_0, \partial \mathbf{v}_{s^*}/\partial \mathbf{v}_0$) can be computed as part of the symplectic integration procedure used for Hamilton's equations, in parallel with ray tracing, as shown in Algorithm 2.

Type-2: Both vertices inside the medium. When both the vertices are inside the medium, we can retain most of the theory developed for type-1 connections. However, we do not have the implicit surface constraint to restrict the geometric curve length (s^*). To mitigate this problem, we define $s^* := \arg_s \min \|\mathbf{x}_s - \mathbf{y}\|^2$, which is the shortest geometric length the light path has to be propagated to be closest to $\mathbf{y}$. This definition of s^* is equivalent to solving the implicit surface $(\mathbf{x}_s - \mathbf{y})^T \mathbf{v}_s = 0$, which brings us back to the derivations for type-1 connections. The exact gradient for this case is derived in Appendix B. Figure 4 shows a visualization of a gradient descent optimization.

We note that type-2 connections can handle type-1 connections as well. However, we found that using type-1 connections increases

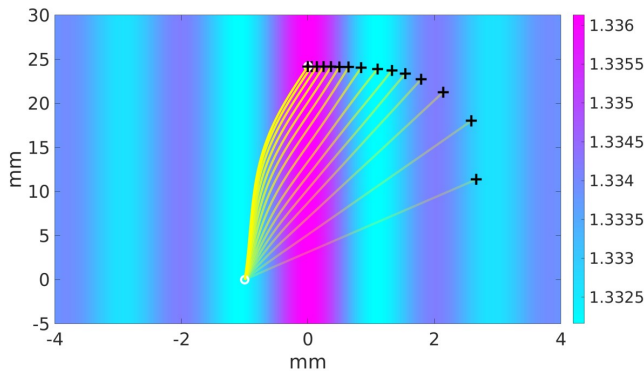


Fig. 4. **Visualization of gradient descent optimization.** The background image is the refractive index field (RIF) of the medium. The white circles mark the location of the vertices between which a direct connection has to be computed. The yellow curve is obtained by solving IVP for a given initial velocity and vertex position. Using gradient descent, we converge to a velocity that makes the yellow curve pass through both the vertices.

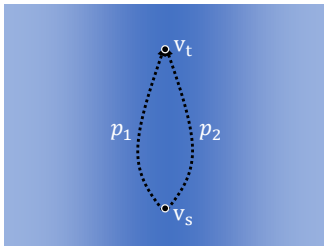


Fig. 5. **Multiple direct connections.** Between a source path end (v_s) and sensor path end (v_t), we can have multiple connections that all satisfy Equations (1)-(2) at all the points on the path. Computing a path deterministically biases the Monte Carlo algorithm. To select all the paths randomly, we initialize the gradient descent technique with random initial velocity $\mathbf{v}_0$.

the number of converged cases by 10%, which decreases the number of times the optimizer must be reinitialized. We think this is because of a better-behaved loss function: Whereas the loss function for type-1 uses the point where the path intersects the surface, the one for type-2 uses the closest point along the path, which may be anywhere in the medium.

4.1 Multiple direct connections

In the case of a medium with a constant refractive index, there is always only one stationary path between any two points. By contrast, in a medium with a heterogeneous refractive index, there can be more than one stationary paths between two points, as shown in Figure 5. In such a case, one would need to compute the total throughput $\sum_{k=1}^K f(C_k)$, where $f(C_k)$ is the throughput of the complete light path formed by connecting the source and sensor subpaths through the k -th stationary path C_k , and K is the number of stationary paths. However, in practice, it is difficult to compute all stationary paths between two points, effectively requiring using the BVP solver with all possible initialization values.

To circumvent this, we compute an unbiased Monte Carlo estimate of the above sum. This requires first having a procedure that can randomly sample each of the K stationary paths with some

Algorithm 3: Multiple direct connections

```

Input:  $\mathbf{x}, \mathbf{y}, n(\mathbf{x}), \nabla n(\mathbf{x}), H_n(\mathbf{x})$ ;
Output:  $C, p(C)$ 
/* Initialization */
sample  $\mathbf{v} \sim \text{Unif}(\text{sphere})$ ;
 $C = \text{Gradient Descent}(\text{ray}(\mathbf{x}, \mathbf{v}), \mathbf{y}, n, \nabla n, H_n)$ ;
 $1/p(C) = 1$ ;
while true do
    sample  $\mathbf{v} \sim \text{Unif}(\text{sphere})$ ;
     $C^* = \text{Gradient Descent}(\text{ray}(\mathbf{x}, \mathbf{v}), \mathbf{y}, n, \nabla n, H_n)$ ;
    if  $\|C^* - C\| < \epsilon$  then
        | break;
    end
     $1/p(C) = 1/p(C) + 1$ ;
end

```

non-zero probability $p(C_k)$; and second, being able to compute the probabilities $p(C_k)$ for all k . Then, after drawing a set of samples $C^{(n)} \in \{C_k, k = 1, \dots, K\}, n = 1, \dots, N$, we estimate the total throughput as $\sum_{n=1}^N f(C^{(n)})/p(C^{(n)})$. In practice, using just one sample ($N = 1$) is sufficient for creating an unbiased estimate.

Our approach for creating the sampling distribution $p(C_k)$ is analogous to the algorithm proposed by Zeltner et al. [2020] for the unbiased estimation of total throughput through multiple possible specular connections. We first randomly sample an initial velocity of $\mathbf{v}_0$ using some probability on the sphere (e.g., uniform). Then we use $\mathbf{v}_0$ to initialize our BVP solver, which will converge to a stationary path $C \in \{C_k, k = 1, \dots, K\}$. The primary challenge is to compute the probability $p(C)$ of sampling this stationary path. This would require integrating the probability we use to sample $\mathbf{v}_0$ over all possible values $\mathbf{v}_0$ for which the above procedure produces the same stationary path C —that is, computing the area of the attraction basin of C . This integral can be approximated by a second Monte Carlo procedure. However, as Zeltner et al. [2020] explains, this can introduce significant bias, because ultimately we need to use the inverse probability $1/p(C)$ in our Monte Carlo estimate of the total throughput. Instead, Zeltner et al. [2020] use an algorithm proposed by Booth [2007] that directly estimates the inverse probability. This algorithm works by continually sampling initial velocities $\mathbf{v}_0$ and reinitializing the BVP solver until we arrive again at the stationary path C ; the number of reinitializations is an unbiased estimate of $1/p(C)$. We use this approach, as well. The procedure is summarized in Algorithm 3, and we refer to Zeltner et al. [2020] for details.

We conclude this section with some notes. First, as our BVP solver uses a gradient descent algorithm, it may fail to converge for some initializations. To maintain unbiasedness, we can treat all cases of non-convergence as corresponding to some virtual *null path* C_0 , for which $f(C_0) \equiv 0$. Then, Algorithm 3 will produce correct estimates for $1/p(C_0)$, as well as the inverse probabilities for all the real stationary paths, ensuring unbiasedness. We note that, when the sampling procedure produces C_0 , the fact that $f(C_0) = 0$ means that it is not necessary to run the subroutine for computing the corresponding

inverse probability. We note that, when using $N = 1$ samples for estimating the total throughput, non-convergence would result in the total light path (including source and sensor subpaths) not contributing to the final image, wasting significant computation. This can be avoided by either using a number of samples $N > 1$ or continuing to draw sample stationary paths C using, e.g., a Russian roulette procedure [Veach 1997]. In practice, in the experiments we show in Section 7, fewer than 1.5% of BVP initializations failed to converge, and thus this issue did not significantly affect performance.

Second, both our discussion and that by Zeltner et al. [2020] assume that there is a *finite* number K of stationary paths C_k . However, it is possible to have infinitely many stationary paths: For example, in the case of heterogeneous refractive media, this can happen when connecting certain points through rotationally-symmetric or spherically-symmetric refractive fields, e.g., two points placed at diametrically opposite locations on the surface of a Maxwell lens [Merlin 2011]. In the case of specular reflections studied by Zeltner et al. [2020], this can happen when connecting points at the foci of converging curved reflectors [Mitchell and Hanrahan 1992], e.g., the foci of an ellipsoidal reflector. In both of these examples, all initializations of the BVP solver (in the heterogeneous refraction case) or specular connection solver (in the specular refraction case of Zeltner et al. [2020]) will result in valid stationary paths after zero gradient iterations. In cases such as these, the subroutine for estimating inverse probabilities in Algorithm 3 will introduce a small bias, directly controlled by the constant ϵ used for terminating the resampling procedure. Effectively, the subroutine will compute not $p(C)$, but the integral of $p(C)$ in some neighborhood around C whose size is controlled by ϵ . In practice, these pathological situations arise rarely and under very idealized, non-generic conditions.

Third, as discussed by Zeltner et al. [2020], one can also use the *biased* estimator $f(C)$, where C is a stationary path sampled using the sampling subroutine of Algorithm 3. Compared to the estimate $f(C)/p(C)$ produced by the complete Algorithm 3, the biased estimator can have significantly improved performance, as it skips running multiple BVP solvers to estimate the inverse probability. Additionally, when there is only one stationary path, then this biased estimator is actually unbiased—in such cases, the inverse probability estimation subroutine of Algorithm 3 would converge after one iteration, resulting in $p(C) = 1$. This situation arose in all of the experiments shown in Section 7, where we have not detected any case of multiple stationary paths, and where both the unbiased and biased estimators produced identical results, albeit with the biased estimator being twice as fast thanks to avoiding the additional BVP solver call for producing $p(C) = 1$.

5 ACCELERATION TECHNIQUES

In this section, we discuss some techniques for accelerating the non-linear ray tracing and direct connection procedures used by our rendering algorithms.

Checking for boundary intersections. When performing non-linear ray tracing inside a heterogeneous refractive medium, it is necessary to check *after each step* whether the ray has departed the medium. In linear ray tracing, this check can be done using a single ray-mesh intersection test for each traced ray, with the cost of each such

intersection test depending on mesh complexity. However, using intersection tests in non-linear ray tracing would be prohibitively expensive, as we would need to perform an intersection test after each ray tracing step, resulting in potentially hundreds of intersection tests per traced ray. We mitigate this issue by replacing intersection tests with efficient inside-outside tests based on fast winding numbers, as proposed by Barill et al. [2018]. To ensure that there is no bias, when the ray is near the boundary (determined as discussed below), we use binary search to find exact intersections.

Even though the tests based on fast winding numbers are significantly faster than intersection tests, having to perform them multiple times for each ray still results in a large computational overhead. We can further reduce this overhead by using a technique inspired by sphere tracing [Bálint and Valasek 2018; Hart 1996; Seyb et al. 2019] to only perform inside-outside tests after several steps of the non-linear ray tracing procedure. In particular, we assume that we have available the signed distance function (SDF) of the heterogeneous refractive medium. At each point inside the medium, the SDF can be used to compute the *minimum* number of steps needed to reach the boundary. Therefore, when performing a sequence of ray tracing steps, we only need to run the inside-outside test when this minimum number of steps is reached, instead of after every step. In our implementation, we precompute the SDF of all heterogeneous refractive volumes in a scene at sparse points on a uniform grid inside the volumes, and interpolate the SDF values using cubic splines. We also use the SDF to determine when, during the ray tracing process, to switch from fast-winding-number tests to exact intersections. We observed that the use of SDFs resulted on a rendering speedup of around 30%.

Approximate normals for Type-1 connections. Type-1 connections require computing surface normals at multiple points on the medium boundary, as these normals are needed to compute the derivatives in Equation (12). However, computing exact surface normals can be expensive, especially for large meshes. We side-step this issue by replacing normals in the derivative calculation with the gradient of the SDF at the corresponding boundary points: this gradient would be exactly equal to the surface normal if we knew the SDF exactly [Sethian 1996], and is otherwise a sufficient approximation when using the computed derivatives to perform gradient descent. The SDF gradients can be computed in constant time ($O(1)$) by interpolating with derivative cubic spline kernels [Unser 1999].

6 IMPLEMENTATION

We have implemented the BDPT algorithm with direct connections as a plugin to the Mitsuba renderer [Jakob 2010]. Our implementation is publicly available [Pediredla et al. 2020]. In this section, we discuss some important implementation details.

Representation of heterogeneous refractive fields. The heterogeneous refractive field is provided as a set of discrete grid values. We use cubic B-splines [Unser 1999] to interpolate these values and obtain a continuously varying refractive index given by

$$n(x, y, z) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{p=0}^{P-1} C_{mnp} \beta(x-m) \beta(y-n) \beta(z-p). \quad (13)$$

Table 2. Default options and parameter values used in our implementation.

parameter	default value
step size	1 mm
direct connection tolerance	1 mm
Russian roulette weight	0.01
precision (for exact intersection with boundary)	1 μm
optimizer	BFGS
max iterations	20
function tolerance (termination)	1 mm
gradient tolerance (termination)	0
parameter tolerance (termination)	0

We chose B-spline interpolation as coefficients (C_{mnp}) can be computed in linear time ($O(n)$) [Unser 1999] and the values can be interpolated in constant time ($O(1)$). Additionally, the gradient and Hessian required for direct connections can also be computed in $O(1)$ time analytically without numerical differentiation.

Optimizer. For the optimization algorithm required for direct connections, we tested a few gradient-based algorithms (standard gradient descent, steepest descent, accelerated gradient descent, conjugate gradient descent, BFGS, LBFGS, DFP, Levenberg-Marquardt), on a few test heterogeneous refractive media. We found that BFGS performs the best in terms of both the number of iterations until convergence and the percentage of tests where it succeeded. We used the BFGS implementation of the Ceres solver [Agarwal et al. [n. d.]], which we integrated with Mitsuba. Various default options used for the optimization algorithm are listed in Table 2.

7 EXPERIMENTS

We use our implementation to perform a series of experiments on different scenes, including robustness tests for our direct connection procedure, visualization of volumetric caustics, time-of-flight and spectral rendering, as well as ultrasonic lensing. In the latter case, we also compare our renderings to real measurements. All rendering experiments were performed using Amazon Web Services (AWS) cloud computing, and we report the corresponding machine specifications and running times throughout this section.

Effect of RIF frequency. We first characterize the effect of a medium's RIF on the convergence rate of IVP and BVP. Intuitively, we expect that increasing the frequency RIF should decrease the convergence rate of both IVP and BVP. In the limiting case of very-high-frequency RIF, the discretized tracing techniques of Algorithms 1 and 2 will always fail for any numerically stable step size.

To quantify this intuition, we ran IVPs and BVPs on RIFs of the form described in Section 7.2, with progressively increased frequencies. We show the results in Figure 6. To compute the maximum step size required for convergence, we first randomly sampled a point and direction inside the RIF. We then computed the ground truth IVP solution by propagating the ray using Algorithm 1 for a distance for 50 mm with a very small step size of 1 μm . Finally, we repeated the IVP for different step size values, and computed the error between the final ray position and the ground truth IVP solution.

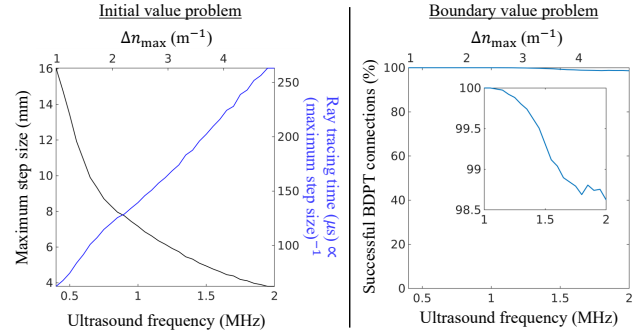


Fig. 6. **Effect of RIF frequency on IVP and BVP.** If the step size used in Algorithm 1 is not small enough, then the symplectic integration technique will fail. The step-size required is a function of the refractive index gradient. An RIF with high frequencies or high refractive index gradients requires a small step size. In the left half of the above figure, we show the maximum step size required for acousto-optic RIFs (discussed in detail in Section 7.2). As the frequency of the acousto-optic RIF, which is proportional to the maximum refractive index gradient is increased, the maximum step size required for the IVP problem to converge decreases, and hence, the ray tracing time increases. An increase in the frequency of RIF also decreases the probability that the direct connections are made, as shown in the figure on the right side.

We repeated the above procedure for several randomly sampled positions and directions and averaged the errors. The largest step size where the average error is less than a threshold of 0.1 μm is the step size for which IVP converges. In agreement with our intuition, we observe in Figure 6 that the largest step size decreases as RIF frequency increases, and that the optimal ray tracing time (number of steps needed for convergence when using the largest step size) increases almost linearly with RIF frequency. In the same figure, we also show the effect of RIF frequency on the success rate of BVPs. As solving BVPs involve solving several IVPs, we fixed the step size to be 1 mm for which the IVP converges for all frequencies. We observe that, as RIF frequency increases, the number of successful direct connections decreases, albeit slightly.

Light propagation in sugar solution. To showcase the bending of light inside a refractive medium, we created in Figure 2 a scene similar to Ament et al. [2014]. The scene comprises an aquarium filled with a solution of sugar in water. The sugar solution concentration is highest at the bottom of the aquarium and lowest at the top. The refractive index of the sugar water is proportional to the concentration of the solution. Hence, the refractive index is lowest at the top of the aquarium and highest at the bottom and varies linearly as we measure RIF from bottom to the top.

Volumetric caustics. Luneburg lenses are spherical GRIN lenses that focus a parallel beam of light to a point on the opposite side on the lens. Visible-range Luneburg lenses are realized with 3D printing technology [Zhao et al. 2016], and are used as spherical retroreflectors on satellites [Shargorodsky et al. 2000]. In Figure 7, we simulate volumetric caustics created by Luneburg lenses with different refractive index profiles. In Figure 8, we use the same scene to perform equal-time comparisons between BDPT and the photon

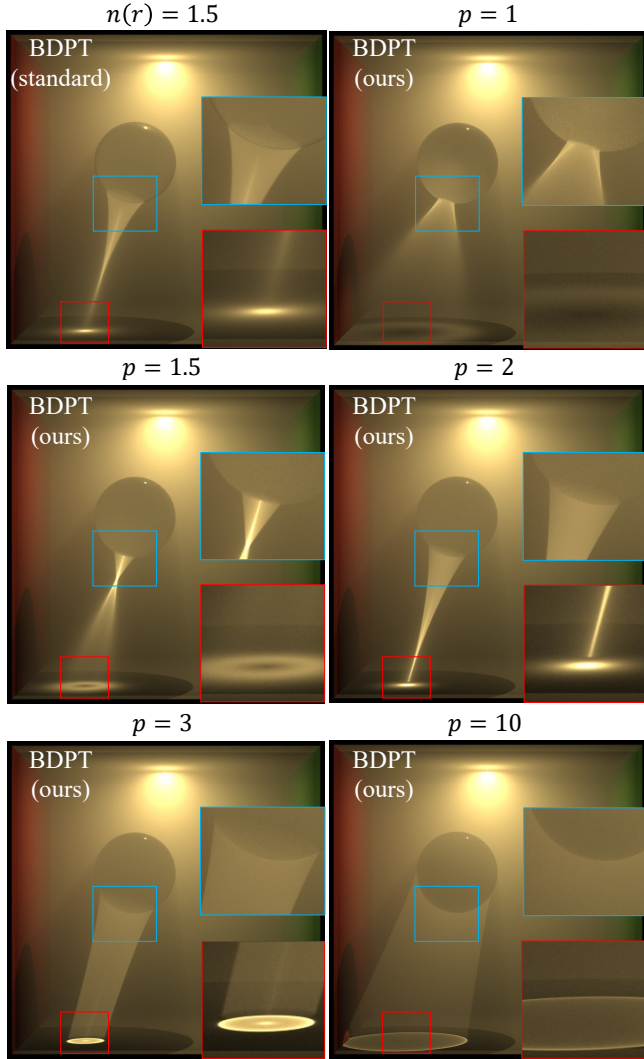


Fig. 7. **Volumetric caustics of various Luneburg lens profiles.** The above volumetric caustic scene is inspired from the dataset by Bitterli [2016]. The scene consists of either a homogeneous glass sphere (rendered with standard BDPT) or a Luneburg lens with refractive index $n(r) = \sqrt[2]{2 - \left(\frac{r}{R}\right)^2}$ (rendered with our BDPT), placed inside a volumetric scattering medium. The observed volumetric caustics are dependent on the refractive index profile of the lens. As the order of the lens increases, its focusing power decreases. The standard and our BDPT had runtimes of 5 min and 10 min, respectively, on a 72-core (AWS c5n.18xlarge) machine.

mapping algorithm of Ament et al. [2014].¹ We observe that BDPT converges faster than photon mapping, even after optimizing the hyper-parameters of the latter, and that photon mapping renderings have blur and other artifacts. We note that advanced photon mapping techniques such as progressive photon mapping [Hachisuka et al. 2008] and stochastic progressive photon mapping [Hachisuka and Jensen 2009] can potentially improve convergence speed compared to the technique of Ament et al. [2014]. However, adapting

¹As the source code of Ament et al. [2014] is not publicly available, we use our own implementation of their technique in Mitsuba.

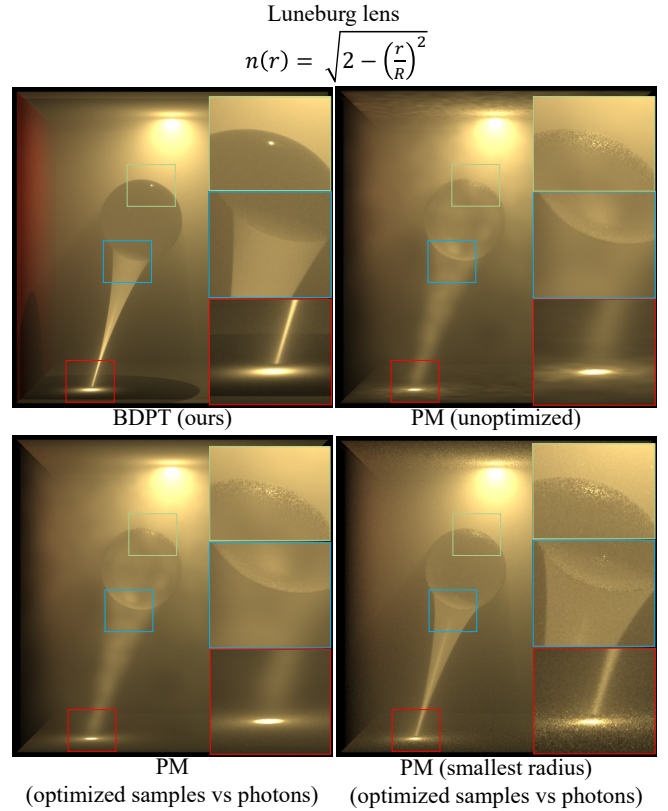


Fig. 8. **Equal runtime comparison with photon mapping.** We compare the photon mapping (PM) technique of Ament et al. [2014] with our BDPT algorithm. As PM has several hyper-parameters, we show results using both the default Mitsuba parameter values, and values optimized using parameter sweeping. We do this parameter fine-tuning for two PM radii, the Mitsuba default and the smallest possible value. We note that the PM result is noisier than BDPT, and also has blur and other artifacts. The runtime for all renderings is 10 min on a 72-core (AWS c5n.18xlarge) machine.

these biased techniques to the RRTE setting is still a matter of future research, beyond the scope of this paper.

Appearance change due to subsurface heterogeneity. Due to prolonged exposure to heat and pressure (a process known as metamorphism [Gillen 2012]), materials such as marble and jade typically exhibit both heterogeneous refractive index and subsurface scattering. In Figure 9, we show an example of this combination: The dragon shape has a refractive index profile that varies as a function of the shape's signed distance function; on the dragon's skeleton, the refractive index equals 1.5, and towards the dragon's surface, it is set to 1.1. We can observe that the caustics on the table, the surface caustics on the dragon, and the appearance of thin geometric features on the dragon mesh all change significantly compared to the case of constant refractive index.

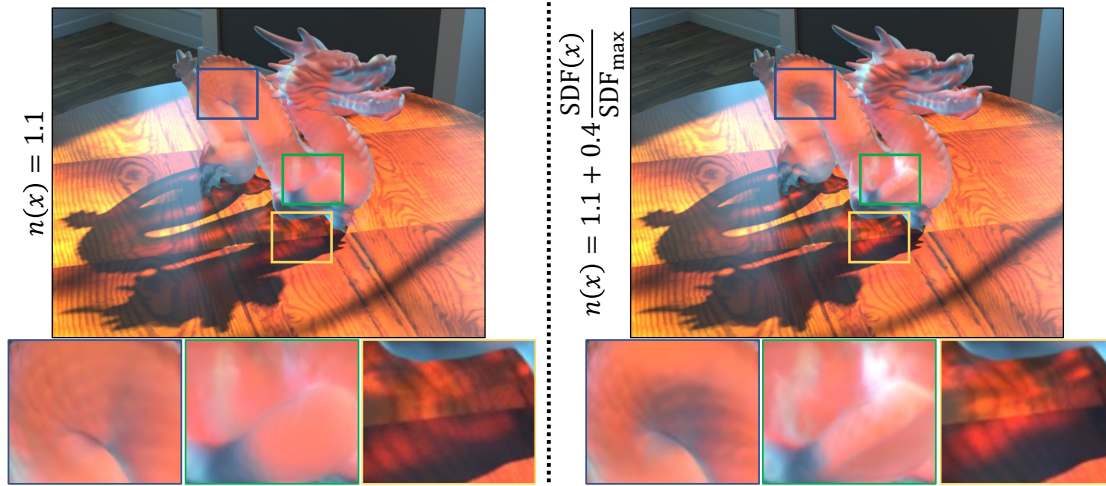


Fig. 9. **Appearance change with subsurface heterogeneity.** Subsurface scattering and refractive index influence the appearance of an object. Here, we observe that sub-surface refractive index heterogeneity also has a strong influence on the appearance of both the object and the caustics produced. The scalar refractive index field, in this case, is assigned similar to the Luneburg lenses. The signed distance function (SDF) is used as a proxy for the radius, and hence, the refractive index is higher deep inside the medium compared to the surface. The homogeneous RIF (left figure) was rendered in 7.33 hours, and the heterogeneous RIF (right figure) was rendered in 39 hours on a 72-core (AWS c5n.18xlarge) machine.

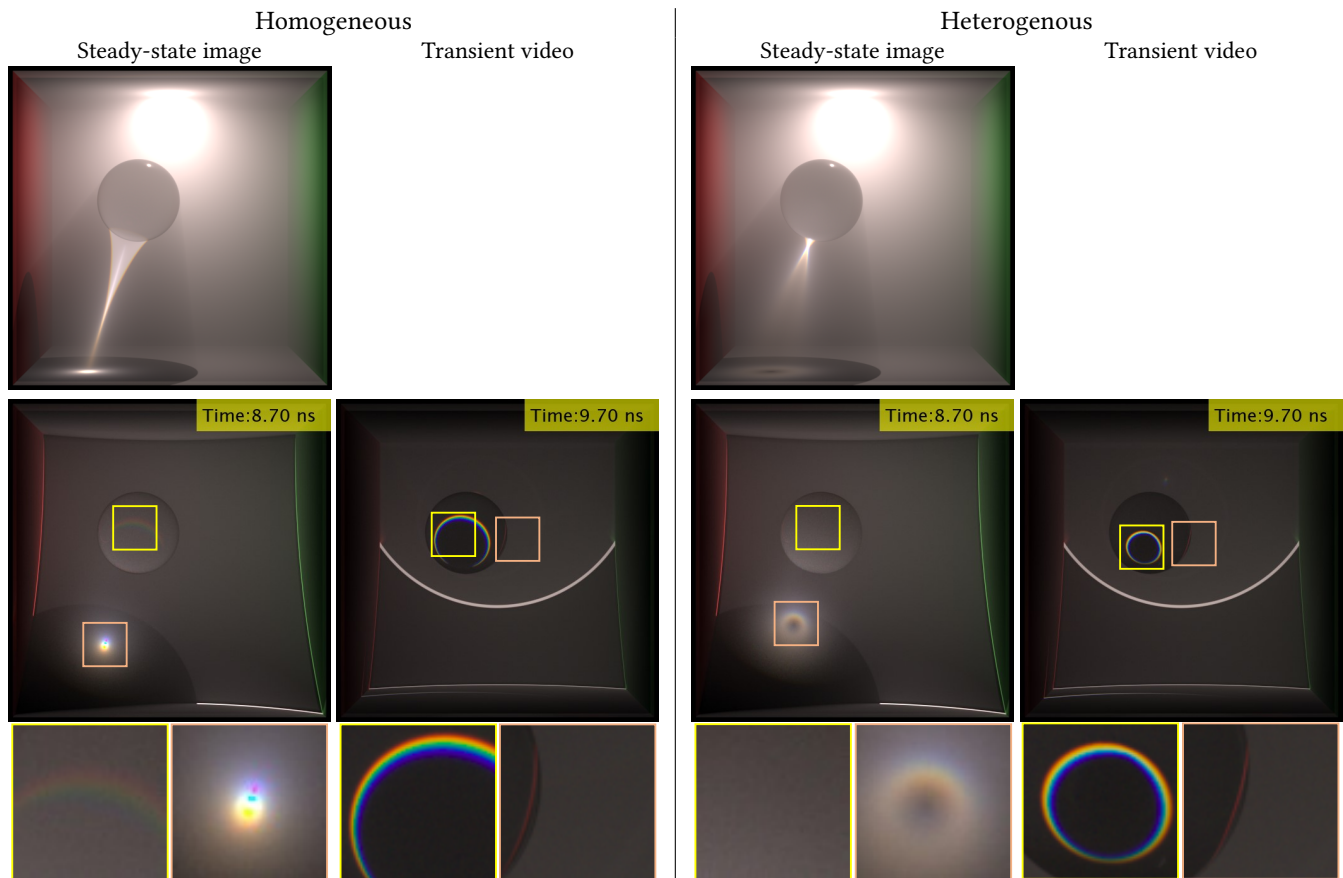


Fig. 10. **Spectral transient rendering.** Transient rendering refers to binning various light paths based on their optical pathlength. The path tracing estimators developed in this paper can be straightforwardly extended to transient rendering. To accurately render spectral dispersion, we used spectral rendering to produce these RGB images. The homogeneous sphere (left) was rendered in 6.18 hours, and the heterogeneous one (right) was rendered in 12.67 hours on a 72-core (AWS c5n.18xlarge) machine. (Please use Adobe Acrobat to view this figure, to see the animated transient video.)

7.1 Extension to time-of-flight rendering

Time-of-flight cameras, which accept or reject photons based on the optical length of their travelled paths, are nowadays common in several applications [Jarabo et al. 2017]. Several rendering algorithms for simulating measurements from such cameras have appeared in recent years [Cammarano and Jensen 2002; Jarabo et al. 2014; Pediredla et al. 2019]. Analogously to Pediredla et al. [2019], we can extend the path integral formulation of Equation (7) to model time-of-flight cameras in a scene containing heterogeneous refractive media, by introducing a pathlength-dependent term:

$$I = \int_{\mathcal{P}} W_{\tau}(\|\bar{\mathbf{x}}\|) f(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}), \quad (14)$$

where $W_{\tau}(\cdot)$ is the camera-dependent *pathlength importance function*, $\|\bar{\mathbf{x}}_n\| \equiv \sum_{k=0}^{n-1} d(\mathbf{x}_k, \mathbf{x}_{k+1})$ and $d(\mathbf{x}_k, \mathbf{x}_{k+1}) \equiv \|\int_0^1 n(\mathbf{y}) d\mathbf{s}\|$. Importantly, when simulating heterogeneous refractive media, it is important to take into account the fact that time-of-flight cameras separate photons based on their *optical* pathlength, that is, pathlength weighted by the refractive index.

In Figure 10, we show RGB steady-state images and transient sequences of both homogeneous and heterogeneous spheres; transient refers to sequences of images where each image records contributions of paths within a narrow range of optical lengths. These images were created using spectral rendering: we first rendered a hyperspectral cube corresponding to multiple wavelengths in the visible range, and then converted this cube into an RGB image through the color transform equations. Spectral rendering is required because the medium is dispersive, that is, its refractive index is a function of wavelength. The refractive index of the homogeneous sphere is modeled after crown glass [Johnson and Christy 1972]. The heterogeneous spheres are modeled as heterogeneous mixtures of crown glass and flint glass, with the resulting refractive index at each location inside the sphere varying as $n(r, \lambda) = rn_{\text{crown}}(\lambda) + (R-r)n_{\text{flint}}(\lambda)$; r is the distance from the sphere center, and R is the outer sphere radius.

From the steady-state images in Figure 10, we observe that the focusing power of the heterogeneous lens is higher than the homogeneous lens, and the caustics formed on the floor are ring-shaped instead of the Gaussian profile of the homogeneous lens. Different frames additionally show different color patterns due to dispersion. In supported PDF viewers, the figure can be animated to show the entire transient sequences. As flint glass has a higher refractive index than crown glass, we notice that the heterogeneous transient exits the sphere later than the homogeneous transient. The transients also show that the chromatic dispersion is slightly less for the heterogeneous case, because flint glass is less dispersive than crown glass. Different chromatic effects can also be observed in the fixed transient frames shown at times 8.7 ns and 9.7 ns. In all cases, red wavelengths travel faster (e.g., exit the sphere or reach the floor earlier) than green and blue wavelengths, as expected from the dispersion properties of real-world materials.

7.2 Simulating in-situ ultrasonically sculpted waveguides

When ultrasonic pressure waves travel through a medium, the density and the medium's refractive index will be locally modulated, and the medium's refractive index becomes heterogeneous.

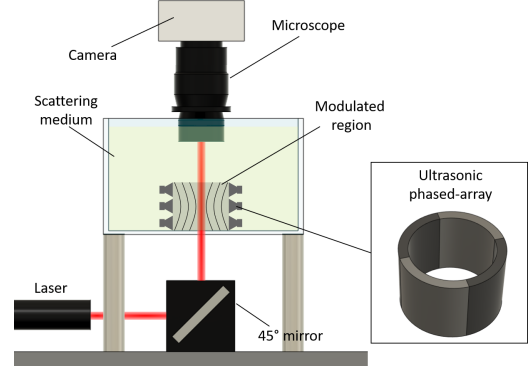


Fig. 11. **Acousto-optic setup.** An ultrasonic transducer is mounted inside a scattering medium (water and intralipid in our experiments). The transducer has four elements. The elements coded in light gray are always on the same phase and out-of-phase with the dark gray elements. This setup is used to modulate the refractive index field inside the scattering medium, which in turn acts as a waveguide for the light traveling through the medium.

The change in refractive index as a function of local density is described by the Lorentz-Lorenz relation [Born and Wolf 2013]. Chamanzar et al. [2019] showed that this phenomenon can be used to sculpt arbitrary spatial refractive index patterns, to shape and control the trajectory of light in transparent and turbid media. They have demonstrated the ability to form virtual ultrasonically-defined optical elements, such as waveguides, within the target medium. These optical elements can be used for light delivery or relaying images through the medium. For example, a virtual relay lens can be sculpted in the medium to relay images of fluorescent targets to an external microscope [Scopelliti and Chamanzar 2019]. These virtual optical components can additionally be configured so that light will form specific caustic patterns inside the medium [Karimi et al. 2019]. This type of ultrasonic modulation of refractive index profiles can be performed using a phased array of ultrasonic transducers from outside the medium, as shown in Figure 11. Our physically accurate rendering algorithms can be used to characterize the performance of these in-situ ultrasonic sculpting techniques, and help assess their utility in different application areas (e.g., tissue and brain imaging).

To demonstrate the applicability of our algorithms for this imaging technique, and to evaluate the accuracy of our algorithms, we used the setup of Figure 11 to perform experiments similar to Karimi et al. [2019]. The refractive index field generated by this ultrasonic array is given by [Karimi et al. 2019]:

$$n(r, \phi, t) = n_o + n_{\text{max}} J_m(k_r r) \cos(m\phi) \sin(\Omega t), \quad (15)$$

where r and ϕ are the radial and azimuthal components of the position $\mathbf{x}$ in cylindrical coordinates induced by the transducer. Note that n is independent of depth. t is time, n_o is the refractive index of the host medium, n_{max} is the maximum amplitude of the refractive index contrast pattern. J_m is the m^{th} -order Bessel function of the first kind, k_r is the radial component of the ultrasound wavevector, and Ω is the angular frequency of the ultrasound. By driving the array elements with a phase and amplitude sequence of $(10 \text{ V} \angle \pi, 10 \text{ V} \angle 0, 10 \text{ V} \angle \pi, 10 \text{ V} \angle 0)$, a caustic pattern of four bright spots can be formed at a depth of $d = 3 \text{ cm}$ through a turbid medium with an

optical thickness of 3 MFP as shown in Figure 1. This caustic pattern corresponds to $m = 2$ in Equation (15).

In Figure 1, we compare the real data we captured against renderings from BDPT and photon mapping. (We refer to Appendix C for more details on these renderings.) The runtime for both BDPT and PM for each θ rendering is approximately 3 hours on a 72-core (AWS c5d.18xlarge) machine. From the renderings, we observe that BDPT more closely matches the real data than PM. However, there are some mismatches between BDPT and real data due to hardware non-idealities in our optical setup:

- *Outer rings of BDPT rendering are not as sharp as the real data.* This mismatch is due to the assumptions in deriving Equation (15). McLeod et al. [2006] derived Equation (15) assuming an infinite-length transducer. However, in reality, the transducer is finite-sized.
- *Real-data intensity is less than BDPT intensity at the boundaries.* This difference is due to the fact that the collimated beam coupled into the medium is a Gaussian beam in experiments, and an ideal collimated beam in renderings.
- *Mismatch in the right side of the intensity profile plot.* As the transducer is symmetric, we would ideally expect real data also to be symmetric. This does not happen in practice as, due to the wire connections made to the transducer, the impedance is higher on one side. This explains the mismatch.

8 CONCLUSION

We introduced Monte Carlo rendering algorithms for simulating the refractive radiative transfer equation (RRTE), describing light transport in media characterized by both volumetric scattering and heterogeneous refractive index. At the core of our algorithms is a procedure that enables efficiently performing direct connections between points in such media by discovering paths connecting the two points while at the same time satisfying Hamilton's equations for continuously-refracting light. We used this procedure as a building block to develop non-linear variants of the volumetric path tracing and bidirectional path tracing algorithms. These new algorithms produce unbiased image estimates and allow us to simulate the RRTE in a way that is both computationally-efficient and physically-accurate. In our experiments, we verified that our algorithms could closely match real-world measurements of materials with scattering and heterogeneous refractive properties, reproducing effects such as complex volumetric caustics that other rendering techniques (e.g., photon mapping) cannot reproduce in equal time.

Even though we focused on path tracing algorithms, our technical and algorithmic results can potentially be used to enable other types of Monte Carlo rendering to simulate the RRTE. For example, a useful by-product of our direct connection technique is the derivatives of solutions to Hamilton's equations with respect to initial conditions. These derivatives can be used to generate *perturbations* of sampled paths for Markov chain Monte Carlo algorithms, analogously to manifold exploration techniques [Jakob and Marschner 2012; Kaplanyan et al. 2014]. By facilitating advances in the physical accuracy and computational efficiency of tools for simulating the RRTE, we believe that our techniques have strong potential for impact in scientific imaging applications involving heterogeneous

refractive media. For example, our rendering algorithms can be used to virtually evaluate the performance limits of existing imaging techniques, without the need for painstaking lab experiments; and to explore, potentially in a data-driven way, new designs that can optimize performance in different application settings. We made our implementation public [Pediredla et al. 2020], to accelerate the adoption of our algorithms in scientific imaging applications, and to facilitate reproducibility and follow-up research.

ACKNOWLEDGMENTS

This work was supported by NSF Expeditions award 1730147, NSF award 1935849, DARPA REVEAL contract HR0011-16-C-0028, and a gift from AWS Cloud Credits for Research.

REFERENCES

- Sameer Agarwal, Keir Mierle, and Others. [n. d.]. Ceres Solver. <http://ceres-solver.org>.
- Marco Ament, Christoph Bergmann, and Daniel Weiskopf. 2014. Refractive radiative transfer equation. *ACM Transactions on Graphics (TOG)* 33, 2 (2014), 17.
- James Arvo. 1995. *Analytic methods for simulated light transport*. Ph.D. Dissertation.
- Bradley Atcheson, Ivo Ihrke, Wolfgang Heidrich, Art Tevs, Derek Bradley, Marcus Magnor, and Hans-Peter Seidel. 2008. Time-resolved 3d capture of non-stationary gas flows. *ACM transactions on graphics (TOG)* 27, 5 (2008), 1–9.
- Csaba Bálint and Gábor Valasek. 2018. Accelerating Sphere Tracing. In *Eurographics (Short Papers)*, 29–32.
- Gavin Barill, Neil G Dickson, Ryan Schmidt, David IW Levin, and Alec Jacobson. 2018. Fast winding numbers for soups and clouds. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 43.
- Fethallah Benmansour, Guillaume Carlier, Gabriel Peyré, and Filippo Santambrogio. 2010. Derivatives with respect to metrics and applications: subgradient marching algorithm. *Numer. Math.* 116, 3 (2010), 357–381.
- Benedikt Bitterli. 2016. Rendering resources. <https://benedikt-bitterli.me/resources/>.
- Thomas E Booth. 2007. Unbiased Monte Carlo estimation of the reciprocal of an integral. *Nuclear science and engineering* 156, 3 (2007), 403–407.
- Max Born and Emil Wolf. 2013. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier.
- Mike Cammarano and Henrik Wann Jensen. 2002. Time Dependent Photon Mapping. In *EGWR*, Paul Debevec and Simon Gibson (Eds.), 135–144. <https://doi.org/10/c54t>
- Chen Cao, Zhong Ren, Baining Guo, and Kun Zhou. 2010. Interactive Rendering of Non-Constant, Refractive Media Using the Ray Equations of Gradient-Index Optics. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 1375–1382.
- Maysamreza Chamanzar, Matteo Giuseppe Scopelliti, Julien Bloch, Ninh Do, Minyoung Huh, Dongjin Seo, Jillian Iafrafi, Vikaas S Sohal, Mohammad-Reza Alam, and Michel M Maharbiz. 2019. Ultrasonic sculpting of virtual optical waveguides in tissue. *Nature communications* 10, 1 (2019), 1–10.
- Subrahmanyam Chandrasekhar. 2013. *Radiative transfer*. Courier Corporation.
- Min Chen and James Arvo. 2000a. Perturbation methods for interactive specular reflections. *IEEE Transactions on Visualization and Computer Graphics* 6, 3 (2000), 253–264.
- Min Chen and James Arvo. 2000b. Theory and application of specular path perturbation. *ACM Transactions on Graphics (TOG)* 19, 4 (2000), 246–278.
- SB Dalziel, Graham O Hughes, and Bruce R Sutherland. 2000. Whole-field density measurements by 'synthetic schlieren'. *Experiments in fluids* 28, 4 (2000), 322–335.
- Cornelius Gillen. 2012. *Metamorphic geology: an introduction to tectonic and metamorphic processes*. Springer Science & Business Media.
- Eduard Gröller. 1995. Nonlinear ray tracing: Visualizing strange worlds. *The Visual Computer* 11, 5 (1995), 263–274.
- Diego Gutierrez, Adolfo Munoz, Oscar Anson, and Francisco J Seron. 2005. Non-linear Volume Photon Mapping. In *Rendering Techniques*, 291–300.
- Diego Gutierrez, Francisco J Seron, Adolfo Munoz, and Oscar Anson. 2006. Simulation of atmospheric phenomena. *Computers & Graphics* 30, 6 (2006), 994–1010.
- Jörg Haber, Marcus Magnor, and Hans-Peter Seidel. 2005. Physically-based simulation of twilight phenomena. *ACM Transactions on Graphics (TOG)* 24, 4 (2005), 1353–1373.
- Toshiya Hachisuka and Henrik Wann Jensen. 2009. Stochastic progressive photon mapping. In *ACM SIGGRAPH Asia 2009 papers*, 1–8.
- Toshiya Hachisuka, Shinji Ogaki, and Henrik Wann Jensen. 2008. Progressive photon mapping. In *ACM SIGGRAPH Asia 2008 papers*, 1–8.
- Johannes Hanika, Marc Droske, and Luca Fascione. 2015. Manifold next event estimation. In *Computer graphics forum*, Vol. 34. Wiley Online Library, 87–97.
- John C Hart. 1996. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer* 12, 10 (1996), 527–545.

- Walton L Howes. 1984. Rainbow schlieren and its applications. *Applied Optics* 23, 14 (1984), 2449–2460.
- Ivo Ihrke, Gernot Ziegler, Art Tevs, Christian Theobalt, Marcus Magnor, and Hans-Peter Seidel. 2007. Eikonal rendering: Efficient light transport in refractive objects. *ACM Transactions on Graphics (TOG)* 26, 3 (2007), 59.
- David S Immel, Michael F Cohen, and Donald P Greenberg. 1986. A radiosity method for non-diffuse environments. *Acm Siggraph Computer Graphics* 20, 4 (1986), 133–142.
- Akira Ishimaru. 1978. *Wave propagation and scattering in random media*. Vol. 2. Academic press New York.
- Wenzel Jakob. 2010. Mitsuba renderer. <http://www.mitsuba-renderer.org>.
- Wenzel Jakob. 2016. Path Space Markov Chain Monte Carlo Methods in Computer Graphics. In *Monte Carlo and Quasi-Monte Carlo Methods*. Springer, 107–141.
- Wenzel Jakob and Steve Marschner. 2012. Manifold exploration: a Markov Chain Monte Carlo technique for rendering scenes with difficult specular transport. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–13.
- Adrian Jarabo, Julio Marco, Adolfo Muñoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. 2014. A framework for transient rendering. *ACM Transactions on Graphics (ToG)* 33, 6 (2014), 1–10.
- Adrian Jarabo, Belen Masia, Julio Marco, and Diego Gutierrez. 2017. Recent advances in transient imaging: A computer graphics and vision perspective. *Visual Informatics* 1, 1 (2017), 65–79.
- Yu Ji, Jinwei Ye, and Jingyi Yu. 2013. Reconstructing gas flows using light-path approximation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2507–2514.
- Peter B Johnson and R-WJPrB Christy. 1972. Optical constants of the noble metals. *Physical Review B* 6, 12 (1972), 4370.
- James T Kajiya. 1986. The rendering equation. In *ACM SIGGRAPH computer graphics*, Vol. 20. ACM, 143–150.
- Anton S Kaplanyan, Johannes Hanika, and Carsten Dachsbacher. 2014. The natural-constraint representation of the path space for efficient light transport simulation. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–13.
- Felix P Kapron. 1970. Geometrical optics of parabolic index-gradient cylindrical lenses. *JOSA* 60, 11 (1970), 1433–1436.
- Yasin Karimi, Matteo Giuseppe Scopelliti, Ninh Do, Mohammad-Reza Alam, and Maysamreza Chamanzar. 2019. In situ 3D reconfigurable ultrasonically sculpted optical beam paths. *Optics express* 27, 5 (2019), 7249–7265.
- Yu A Kravtsov and Yu I Orlov. 1990. *Geometrical optics of inhomogeneous media*. Springer-Verlag.
- P Krehl and S Engemann. 1995. August Toepler—the first who visualized shock waves. *Shock Waves* 5, 1–2 (1995), 1–18.
- Eric P Lafortune and Yves D Willems. 1996. Rendering participating media with bidirectional path tracing. In *Rendering techniques' 96*. Springer, 91–100.
- Shingyu Leung, Jianliang Qian, et al. 2006. An adjoint state method for three-dimensional transmission traveltime tomography using first-arrivals. *Communications in Mathematical Sciences* 4, 1 (2006), 249–266.
- Rudolf Karl Luneburg. 1966. *Mathematical theory of optics*. Univ of California Press.
- Chenguang Ma, Xing Lin, Jinli Suo, Qionghai Dai, and Gordon Wetzstein. 2014. Transparent object reconstruction via coded transport of intensity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3238–3245.
- Euan McLeod, Adam B Hopkins, and Craig B Arnold. 2006. Multiscale Bessel beams generated by a tunable acoustic gradient index of refraction lens. *Optics letters* 31, 21 (2006), 3155–3157.
- Roberto Merlin. 2011. Maxwell's fish-eye lens and the mirage of perfect imaging. *Journal of Optics* 13, 2 (2011), 024017.
- Alexandre Mermillod-Blondin, Euan McLeod, and Craig B Arnold. 2008. High-speed varifocal imaging with a tunable acoustic gradient index of refraction lens. *Optics letters* 33, 18 (2008), 2146–2148.
- Don Mitchell and Pat Hanrahan. 1992. Illumination from curved reflectors. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*. 283–291.
- Qi Mo, Hengchin Yeh, and Dinesh Manocha. 2015. Tracing analytic ray curves for light and sound propagation in non-linear media. *IEEE transactions on visualization and computer graphics* 22, 11 (2015), 2493–2506.
- Adolfo Muñoz. 2014. Higher order ray marching. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, 167–176.
- Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. 2018. Monte carlo methods for volumetric light transport simulation. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 551–576.
- Jan Novak, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. 2018. Monte Carlo Methods for Volumetric Light Transport Simulation. *Computer Graphics Forum* (2018).
- Adithya Pediredla, Yasin Karimi Chalamiani, Matteo Giuseppe Scopelliti, Maysam Chamanzar, Srinivasa Narasimhan, and Gkioulekas Ioannis. 2020. Mitsuba Eikonal Renderer. <https://github.com/cmu-ci-lab/MitsubaER>.
- Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. 2019. Ellipsoidal path connections for time-gated rendering. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.
- Etienne Robein. 2010. *Seismic imaging: a review of the techniques, their principles, merits and limitations*. EAGE publications.
- Jesus M Sanz-Serna. 1992. Symplectic integrators for Hamiltonian problems: an overview. *Acta numerica* 1 (1992), 243–286.
- Matteo Giuseppe Scopelliti and Maysamreza Chamanzar. 2019. Ultrasonically sculpted virtual relay lens for in situ microimaging. *Light: Science & Applications* 8, 1 (2019), 1–15.
- James Albert Sethian. 1996. *Level set methods: Evolving interfaces in geometry, fluid mechanics, computer vision, and materials science*. Vol. 1999. Cambridge University Press Cambridge.
- Gary S Settles and Michael J Hargather. 2017. A review of recent developments in schlieren and shadowgraph techniques. *Measurement Science and Technology* 28, 4 (2017), 042001.
- Dario Seyb, Alec Jacobson, Derek Nowrouzezahrai, and Wojciech Jarosz. 2019. Non-linear sphere tracing for rendering deformed signed distance fields. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–12.
- VD Shargorodsky, VP Vasiliev, NM Soyuzova, VB Burmistrov, IS Gashkin, MS Belov, TI Khorosheva, and E Nikolaev. 2000. Experimental spherical retroreflector on board of the meteor-3M satellite. In *Proceedings of the 12th International Workshop on Laser Ranging*. 1–5.
- Anurag Sharma and AK Ghatak. 1981. A variational analysis of single mode graded-index fibers. *Optics Communications* 36, 1 (1981), 22–24.
- Jos Stam and Eric Languénou. 1996. Ray tracing in non-constant media. In *Rendering Techniques' 96*. Springer, 225–234.
- Shlomi Steinberg. 2020. Accurate Rendering of Liquid-Crystals and Inhomogeneous Optically Anisotropic Media. *ACM Transactions on Graphics (TOG)* 39, 3 (2020), 1–23.
- Teng-Qian Sun, Qing Ye, Xiao-Wan Wang, Jin Wang, Zhi-Chao Deng, Jian-Chun Mei, Wen-Yuan Zhou, Chun-Ping Zhang, and Jian-Guo Tian. 2014. Scanning focused refractive-index microscopy. *Scientific reports* 4 (2014), 5647.
- Michael Unser. 1999. Splines: A perfect fit for signal and image processing. *IEEE Signal processing magazine* (1999).
- Eric Veach. 1997. *Robust Monte Carlo methods for light transport simulation*. Vol. 1610. Stanford University PhD thesis.
- Eric Veach and Leonidas Guibas. 1995. Bidirectional estimators for light transport. In *Photorealistic Rendering Techniques*. Springer, 145–167.
- Eric Veach and Leonidas J Guibas. 1997. Metropolis light transport. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 65–76.
- Bruce Walter, Shuang Zhao, Nicolas Holzschuch, and Kavita Bala. 2009. Single scattering in refractive media with triangle mesh boundaries. In *ACM Transactions on Graphics (TOG)*, Vol. 28. ACM, 92.
- Daniel Weiskopf, Tobias Schafhitzel, and Thomas Ertl. 2004. GPU-based nonlinear ray tracing. In *Computer graphics forum*, Vol. 23. Wiley Online Library, 625–633.
- Gordon Wetzstein, Ramesh Raskar, and Wolfgang Heidrich. 2011. Hand-held schlieren photography with light field probes. In *2011 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 1–8.
- Tianfan Xue, Michael Rubinstein, Neal Wadhwa, Anat Levin, Fredo Durand, and William T Freeman. 2014. Refraction wiggles for measuring fluid depth and velocity from video. In *European Conference on Computer Vision*. Springer, 767–782.
- Tizian Zeltner, Iliyan Georgiev, and Wenzel Jakob. 2020. Specular manifold sampling for rendering high-frequency caustics and glints. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 149–1.
- Yuan-Yuan Zhao, Yong-Liang Zhang, Mei-Ling Zheng, Xian-Zi Dong, Xuan-Ming Duan, and Zhen-Sheng Zhao. 2016. Three-dimensional Luneburg lens at optical frequencies. *Laser & Photonics Reviews* 10, 4 (2016), 665–672.

A RAY DERIVATIVES

Hamilton's equations (1)-(2) can be written in integral form:

$$\mathbf{x}_s = \mathbf{x}_0 + \int_0^s \frac{\mathbf{v}_s}{n(\mathbf{x}_s)} ds, \quad (16)$$

$$\mathbf{v}_s = \mathbf{v}_0 + \int_0^s \nabla n(\mathbf{x}_s) ds. \quad (17)$$

Differentiating these equations with respect to initial velocity ($\mathbf{v}_s$), we get the following equations for the propagation of the derivatives:

$$\begin{aligned} \frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} &= \int_0^s \left(-\frac{1}{n(\mathbf{x}_s)^2} \mathbf{v}_s \nabla n(\mathbf{x}_s) \frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} + \frac{1}{n(\mathbf{x}_s)} \frac{\partial \mathbf{v}_s}{\partial \mathbf{v}_0} \right) ds, \\ \frac{\partial \mathbf{v}_s}{\partial \mathbf{v}_0} &= \mathbf{I} + \int_0^s H_n(\mathbf{x}_s) \frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} ds. \end{aligned} \quad (18)$$

Note that, for computing derivatives, we need to have access to the Hessian of the refractive index field, in addition to the gradient already required for ray propagation. Finally, differentiating Equations (16)-(17) with respect to the geometric distance s :

$$\frac{\partial \mathbf{x}_s}{\partial s} = \frac{\mathbf{v}_s}{n(\mathbf{x}_s)}, \quad \frac{\partial \mathbf{v}_s}{\partial s} = n(\mathbf{x}_s). \quad (19)$$

B DERIVATIVES FOR TYPE-2 CONNECTIONS

We can write the derivative used for type-2 connections as:

$$\frac{d\mathbf{x}_s}{d\mathbf{v}_0} = \left(\frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} - \frac{\partial \mathbf{x}_s}{\partial s} \frac{\mathbf{v}_s^T \frac{\partial \mathbf{x}_s}{\partial \mathbf{v}_0} + (\mathbf{x}_s - \mathbf{y})^T \frac{\partial \mathbf{v}_s}{\partial \mathbf{v}_0}}{\mathbf{v}_s^T \frac{\partial \mathbf{x}_s}{\partial s} + (\mathbf{x}_s - \mathbf{y})^T \frac{\partial \mathbf{v}_s}{\partial s}} \right). \quad (20)$$

C DERIVATIVE AND HESSIAN FOR ACOUSTO-OPTICS

We note that the modulated spatial refractive index profile in Equation (15) changes over time, because ultrasound waves propagate and change over time. The temporal scale of these variations is on the order of microseconds, for ultrasound in the MHz range. Given the much faster temporal dynamics of light and its much faster speed of propagation, we assume that photons emitted at different times would each interact with a temporally-static, but spatially-varying refractive index profile. In practice, the integration time of the sensor is on the order of a few milliseconds, and thus several ultrasonic cycles will be captured and averaged when using continuous-wave illumination. Therefore, the sensor measures:

$$I = \int_t \int_{\mathcal{P}_{n(\mathbf{x},t)}} f_{n(\mathbf{x},t)}(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}) dt. \quad (21)$$

As $n(t)$ is sinusoidal, with an integer number of half-cycles in the integration time, we approximate Equation (21) as

$$I = \int_{\theta=-\frac{\pi}{2}}^{\frac{\pi}{2}} \int_{\mathcal{P}_\theta} f_\theta(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}) dt. \quad (22)$$

We compute the inner integral with Monte Carlo integration. For BDPT, the Monte Carlo integration is extended to the outer integral by sampling $\theta \sim [-\frac{\pi}{2}, \frac{\pi}{2}]$ for each sample. However, such an extension is non-trivial for photon mapping. To make a fair comparison, we have evaluated the outer integral numerically using trapezoidal rule for both photon mapping and BDPT. For this numerical integration, we used 100 θ values uniformly spaced in $[-\frac{\pi}{2}, \frac{\pi}{2}]$.

For the specific case of in-situ acousto-optic spatial modulation in Equation (15), the analytical expression for the scalar refractive index profile is known. Therefore, instead of using spline interpolation, we compute the refractive index gradient and Hessian analytically. To derive these analytical expressions, we first relate cylindrical and Cartesian coordinates as:

$$r = \sqrt{x^2 + y^2}, \quad \phi = \tan^{-1} \left(\frac{y}{x} \right), \quad (23)$$

$$\frac{\partial r}{\partial x} = \frac{x}{r}, \quad \frac{\partial r}{\partial y} = \frac{y}{r}, \quad \frac{\partial \phi}{\partial x} = -\frac{y}{r^2}, \quad \frac{\partial \phi}{\partial y} = \frac{x}{r^2}, \quad (24)$$

$$\frac{\partial^2 r}{\partial x^2} = \frac{y^2}{r^4}, \quad \frac{\partial^2 r}{\partial x \partial y} = -\frac{xy}{r^4}, \quad \frac{\partial^2 r}{\partial y^2} = \frac{x^2}{r^4}. \quad (25)$$

Then, we compute the refractive index gradient as follows:

$$n(r, \phi, t) = n_0 + n_{\max} J_m(K_r r) \cos(m\phi) \\ \frac{\partial n}{\partial x}(r, \phi, t) = n_{\max} \left[J'_m(K_r r) K_r \frac{x}{r} \cos(m\phi) + J_m(K_r r) m \frac{y}{r^2} \sin(m\phi) \right], \quad (26)$$

where

$$J'_m(K_r r) = \frac{m}{K_r r} J_m(K_r r) - J_{m+1}(K_r r). \quad (27)$$

Substituting, we get:

$$\frac{\partial n}{\partial x}(r, \phi, t) = n_{\max} \left[\frac{m}{r} \cos\{(m-1)\phi\} J_m(K_r r) - K_r \cos\phi \cos m\phi J_{m+1}(K_r r) \right]. \quad (28)$$

Similarly,

$$\frac{\partial n}{\partial y}(r, \phi, t) = n_{\max} \left[J'_m(K_r r) K_r \frac{y}{r} \cos(m\phi) - J_m(K_r r) m \frac{x}{r^2} \sin(m\phi) \right] \\ = -n_{\max} \left[\frac{m}{r} \sin\{(m-1)\phi\} J_m(K_r r) + K_r \sin\phi \cos m\phi J_{m+1}(K_r r) \right]. \quad (29)$$

We compute the refractive index Hessian as follows:

$$\frac{1}{n_{\max}} \frac{\partial^2 n}{\partial x^2}(r, \phi, t) = +J_m(K_r r) \left[-\frac{m}{r^2} \cos\{m-2\}\phi + \frac{m^2}{r^2} \sin\{m-1\}\phi \sin\phi \right] \\ + J'_m(K_r r) \left[\frac{m}{r} K_r \cos\{m-1\}\phi \cos\phi \right] \\ - J_{m+1}(K_r r) \frac{K_r y}{r^2} [\sin\phi \cos m\phi + m \cos\phi \sin m\phi] \\ - J'_{m+1}(K_r r) [K_r^2 \cos^2\phi \cos m\phi], \quad (30)$$

$$\frac{1}{n_{\max}} \frac{\partial^2 n}{\partial x \partial y}(r, \phi, t) = -J_m(K_r r) \left[-\frac{m}{r^2} \sin\{m-2\}\phi + \frac{m^2}{r^2} \sin\{m-1\}\phi \cos\phi \right] \\ + J'_m(K_r r) \left[\frac{m K_r}{r} \cos\{m-1\}\phi \sin\phi \right] \\ + J_{m+1}(K_r r) \frac{K_r x}{r^2} [\sin\phi \cos m\phi + m \cos\phi \sin m\phi] \\ - J'_{m+1}(K_r r) [K_r^2 \cos\phi \sin\phi \cos m\phi], \quad (31)$$

$$-\frac{1}{n_{\max}} \frac{\partial^2 n}{\partial y^2}(r, \phi, t) = +J_m(K_r r) \left[-\frac{m}{r^2} \cos\{m-2\}\phi + \frac{m^2}{r^2} \cos\{m-1\}\phi \cos\phi \right] \\ + J'_m(K_r r) \left[\frac{m K_r}{r} \sin\{m-1\}\phi \sin\phi \right] \\ + J_{m+1}(K_r r) \frac{K_r x}{r^2} [\cos\phi \cos m\phi - m \sin\phi \sin m\phi] \\ + J'_{m+1}(K_r r) [K_r^2 \sin^2\phi \cos m\phi]. \quad (32)$$